28 Schriften aus der Fakultät Wirtschaftsinformatik und Angewandte Informatik der Otto-Friedrich-Universität Bamberg

Management von Datenanalyseprozessen

Bernd Knobloch

University of Bamberg Press

28 Schriften aus der Fakultät Wirtschaftsinformatik und Angewandte Informatik der Otto-Friedrich-Universität Bamberg

Contributions of the Faculty Information Systems and Applied Computer Sciences of the Otto-Friedrich-University Bamberg

Schriften aus der Fakultät Wirtschaftsinformatik und Angewandte Informatik der Otto-Friedrich-Universität Bamberg

Contributions of the Faculty Information Systems and Applied Computer Sciences of the Otto-Friedrich-University Bamberg

Band 28

University of Bamberg Press
2018

Management von Datenanalyseprozessen

von Bernd Knobloch

University
of Bamberg
Press
2018

Bibliographische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliographie; detaillierte bibliographische Informationen sind im Internet über http://dnb.d-nb.de/ abrufbar.

Diese Arbeit hat der Fakultät Wirtschaftsinformatik und Angewandte Informatik der Otto-Friedrich-Universität Bamberg als Dissertation vorgelegen.
1. Gutachter: Prof. Dr. Elmar J. Sinz
2. Gutachter: Prof. Dr. Otto K. Ferstl
Tag der mündlichen Prüfung: 04.12.2017

Herstellung und Druck: Digital Print Group, Nürnberg
Umschlaggestaltung: University of Bamberg Press, Larissa Günther
Umschlagbild © Bernd Knobloch

ISSN: 1867-7401
ISBN: 978-3-86309-565-9 (Druckausgabe)
eISBN: 978-3-86309-566-6 (Online-Ausgabe)
URN: urn:nbn:de:bvb:473-opus4-514832
DOI: http://dx.doi.org/10.20378/irbo-51483

Geleitwort

Unter dem Leitbild „Das datengetriebene Unternehmen“ ist in den letzten Jahren die betriebliche Datenanalyse zu einem der zentralen Themen für die effektive und flexible Unternehmensführung geworden. Die Gründe dafür sind vielfältig:

- Der steigende Wettbewerbsdruck sorgt dafür, dass das kurzfristige Erkennen von Änderungen der Marktgegebenheiten noch mehr als bisher zu einer Überlebensfrage wird.
- Der Begriff Digitalisierung, der im Kern die Verbindung von Real- und IT-Welt bezeichnet, ist derzeit in aller Munde. Dabei war, mit dem Fokus auf Automatisierung, Digitalisierung seit jeher eines der Kernthemen der Wirtschaftsinformatik.
- Die Verfügbarkeit von nicht-transaktionsbezogenen Datenströmen ist z.B. durch die Sozialen Medien stark angewachsen.
- Speicher wird immer billiger, Internet und mobile Geräte sind allgemein verfügbar.
- Zur Bearbeitung der Daten steht eine Kombination spezifischer Methoden und Verfahren, z.B. auch aus dem Bereich der Statistik, zur Verfügung.
- Data-Warehouse- und Data-Mining-Systeme bereiten die Daten nutzungsgerecht auf.

Die zentrale Frage lautet: Wie führt man ein Unternehmen auf Basis der modernen Datenanalyse?

Bernd Knobloch stellt die Frage nach den Voraussetzungen für den Einsatz einer effektiven Datenanalyse im Unternehmen, nach dem Management und der zugehörigen Infrastruktur von Datenanalyseprozessen. Management versteht er dabei als Gestaltung und Lenkung. Er bearbeitet beide Teilfragen. Die Gestaltungsaufgabe wird als Formulierung und Lösung eines Konstruktionsproblems dargestellt, das er mithilfe des Aufgabenkonzepts der Organisationslehre löst. Die Lenkungsaufgabe wird kybernetisch als Regelungsproblem mit dem Phasenschema Planung, Steuerung, Durchführung und Kontrolle ver-

standen. Beide Problembereiche behandelt Bernd Knobloch sehr differenziert. Es geht ihm dabei stets darum, Datenanalyseprobleme nicht nur ein einziges Mal zu lösen, sondern ihr Management einschließlich der zugehörigen Infrastruktur im Unternehmen zu etablieren. Die Evaluierung des vorgeschlagenen Ansatzes erfolgt in einer praxisorientierten Fallstudie.

Mit der vorliegenden Arbeit wird der Themenbereich der betrieblichen Datenanalyse gründlich und ganzheitlich bearbeitet. Bernd Knobloch stellt das Thema einerseits aus dem Blickwinkel der aktuellen Forschung dar. Andererseits verfügt er über eine jahrelange, einschlägige Berufserfahrung, welche die Problemrelevanz und die Praxistauglichkeit der Ausführungen sicherstellt. Die Lektüre des Buches kann allen empfohlen werden, die sich in Wissenschaft und Praxis mit Fragen der betrieblichen Datenanalyse beschäftigen.

Bamberg, im Januar 2018

Prof. Dr. Elmar J. Sinz

Vorwort

Die Durchdringung aller Lebensbereiche mit datenverarbeitenden Systemen hat unser Leben, Arbeiten und nicht zuletzt unser Denken verändert. Zu den vielen Facetten dieser Entwicklung zählt insbesondere die mit zunehmender Geschwindigkeit auf uns einströmende Flut an Informationen. Mit deren sinkender Halbwertzeit steigen die Anforderungen an ihre rasche Verarbeitung und Verwertung. Hierfür stehen immer mächtigere Verfahren der Datenanalyse zur Verfügung.

Im Jahre 1998 durfte ich im Rahmen eines Praktikums Bekanntschaft mit dem damals neuen Data-Mining-Ansatz zur „intelligenten" Datenanalyse machen. So faszinierend das Potenzial dieser Analysetechnik war, so schnell zeigten sich beim praktischen Einsatz jedoch ihre Tücken und Komplexität. So reifte die Erkenntnis, dass der Anwender methodische Unterstützung bei der Konzeption und Ausführung solcher Analysen gebrauchen kann. Während meiner Tätigkeit als wissenschaftlicher Mitarbeiter an der Universität Bamberg hatte ich Gelegenheit, den Data-Mining-Ansatz aus Sicht der Wirtschaftsinformatik theoretisch tiefer zu untersuchen und in den breiteren Kontext der Informationsversorgung des Managements einzuordnen. Mit Blick auf die betriebliche Nutzung wurde deutlich, dass zur fundierten Beantwortung fachlicher Fragestellungen mithilfe der Datenanalyse eine anwendungsorientierte Verankerung hilfreich und stets ein Zusammenspiel mehrerer Ansätze nötig ist.

Als freiberuflicher Unternehmensberater konnte ich hierzu in zahlreichen Projekten über viele Jahre hinweg Erfahrungen sammeln, Empfehlungen erarbeiten, weiterentwickeln und auf ihre Tauglichkeit prüfen. In dem gemeinsam mit Peter Neckel verfassten Handbuch „Customer Relationship Analytics" beschreibe ich einen ersten Vorschlag für eine eher pragmatische Analysemethodik, die empirische Fragen aus Anwendungsproblemen ableitet. Die positive Resonanz auf diese Darstellung gab Anlass zu dem Entschluss, dieses Thema im Rahmen einer Dissertation ausführlicher zu betrachten.

Der Anspruch, eine entsprechende Technik zu entwickeln, die zudem in der betrieblichen Praxis auch brauchbar (d.h., leicht verständlich und

nachvollziehbar) ist, stellte sich wiederholt als enorme Herausforderung dar, die zuweilen als zu groß erschien, als dass sie von mir allein zu bewältigen wäre. Den langen und beileibe nicht immer geradlinigen Weg, der schließlich doch noch zu einer fertigen Dissertation führte, haben viele Menschen begleitet. Ihre Einsicht und Einsichten, Unterstützung und konstruktive Kritik, ihr Ansporn und Rückhalt sind unbezahlbar.

Meinem Doktorvater Herrn Prof. Dr. Elmar J. Sinz danke ich herzlich für die inspirierende und geduldige Betreuung meiner Arbeit sowie für die schöne Zeit am Lehrstuhl SEDA. Herrn Prof. Dr. Otto K. Ferstl gebührt mein Dank nicht nur für die Übernahme des Zweitgutachtens, sondern auch für die langjährige Begleitung durch das Thema Data Mining seit meiner Diplomarbeit. Herrn Prof. Dr. Wolfgang Becker danke ich für die Mitwirkung in der Promotionskommission und für den unverzichtbaren betriebswirtschaftlichen Blick auf das Thema. Herrn Dr. Jens Weidner verdanke ich außer einer stets amüsanten und lehrreichen Zeit auch das Interesse für das Thema Data Mining. Überaus angenehm und fruchtbar war die Partnerschaft mit dem CEUS-Team an der Universität Bamberg. Mit Peter Neckel und Tim-Oliver Förtsch verbindet mich eine lange freundschaftliche Zusammenarbeit, und ich schulde ihnen wertvolle thematische Anregungen. Dies gilt gleichermaßen für Herrn Prof. Dr. Thomas Voit, der zudem das Wagnis auf sich nahm, große Teile dieser Arbeit Korrektur zu lesen. Stets zur Stelle war meine Schwester Sabine, die sich als zuverlässige Helferin nicht nur in sprachlichen Belangen erwies.

Bewusst zuletzt, weil aus tiefstem Herzen mein Gruß an jene Menschen, die mir am meisten bedeuten: Tina, Sabine, und meine Eltern. Euch allein verdanke ich die Kraft, diese Arbeit geschafft zu haben!

Ich widme diese Arbeit zwei großartigen Persönlichkeiten, die ihre Fertigstellung leider nicht mehr miterleben durften: Meinen Großeltern Hans Eber und Johanna Friederike Knobloch.

Ködnitz, im März 2018

Bernd Knobloch

Inhaltsüberblick

Inhaltsverzeichnis

Abkürzungsverzeichnis

ASCII	American Standard Code for Information Interchange
AST	Algorithm Selection Tool (Software-Prototyp)
AUC	area under curve
BA	Business Analytics
BI	Business Intelligence
CBR	Case-based Reasoning (Fallbasiertes Schließen)
CRISP-DM	Cross Industry Standard Process for Data Mining
CRM	Customer Relationship Management
CWM	Common Warehouse Metamodel
DeGEval	Deutsche Gesellschaft für Evaluation e.V.
DIN	Deutsches Institut für Normung
DSS	Decision Support System
EDA	explorative Datenanalyse
EIS	Executive Information System
ETL-Prozess	Extraktions-, Transformations- und Ladeprozess (Data Warehousing)
FN	False Negatives [Anzahl falsch negativ Klassifizierter]
FP	False Positives [Anzahl falsch positiv Klassifizierter]
HTN	Hierarchical Task Network
HTTP	Hypertext Transfer Protocol

i.d.R.	in der Regel
i.e.S.	im engeren Sinne
i.w.S.	im weiteren Sinne
ISO	Internationale Organisation für Normung (engl. International Organization for Standardization)
IT	Informationstechnik, Informationstechnologie
Kard.	Kardinalität
KDD	Knowledge Discovery in Databases
KI	Künstliche Intelligenz
KNN	künstliches Neuronales Netz
MDL	Minimum Description Length
MIME	Internet Media Type (*ursprünglich* Multipurpose Internet Mail Extension)
MIS	Management Information System, Managementinformationssystem
MLT	Machine Learning Toolbox (Projekt)
MUS	Managementunterstützungssystem
OCL	Object Constraint Language
ODMG	Object Database Management Group
OLAP	On-Line Analytical Processing
OMG	Object Management Group
OR	Operations Research

OWL	Web Ontology Language
OWL-DL	Web Ontology Language/Description Logic
POS	Point of Sale
Pr/T-Netz	Prädikat/Transitions-Netz
PSM	Problem Solving Method
ROC	Receiver Operating Characteristics
SDWM	Semantisches Data-Warehouse-Modell
SEMMA	proprietäres Datenanalyse-Prozessmodell der Firma *SAS INSTITUTE*, Akronym der Phasen Sample, Explore, Modify, Model, Assess
SERM	Strukturiertes Entity-Relationship-Modell
SOAP	*ursprünglich* Simple Object Access Protocol, *jetzt Eigenname*
SOM	(1) Semantisches Objektmodell; (2) Self-Organizing Maps
SQL	Structured Query Language
SWRL	Semantic Web Rule Language
TN	True Negatives [Anzahl richtig negativ Klassifizierter]
TP	True Positives [Anzahl richtig positiv Klassifizierter]
UGM	User Guidance Module (Software-Prototyp, Projekt)
URL	Uniform Resource Locator
W3C	World Wide Web Consortium

WfMC	Workflow Management Coalition
WfMS	Workflow Management System
WSDL	Web Services Description Language
XML	Extensible Markup Language
ZE	Zeiteinheiten

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

"In data analysis we have no difficulty in complicating problems in useful ways" [Tuke62, 8]

Die fortschreitende Digitalisierung aller Arbeits- und Lebensbereiche, die ubiquitäre Verfügbarkeit computergesteuerter Systeme sowie die Tendenz zu nutzererzeugten Inhalten, z.B. in Social Media, führen zu einem stetigen Anwachsen der in den Unternehmen verfügbaren Datenbestände in operativen und analytischen Anwendungssystemen. Diese Situation wird aktuell mit dem Begriff „Big Data" charakterisiert, der zusätzlich die zunehmende Heterogenität dieser Daten mit oft spezifischen Verarbeitungsanforderungen betont. Gleichzeitig verspricht die Auswertung dieser Daten zur Fundierung betrieblicher Entscheidungen oder zur Eröffnung neuer Umsatzquellen große Potenziale. Im Idealfall soll ein „datengetriebenes Unternehmen" entstehen, das sich durch die Fähigkeit zum schnellen Wandel sowohl im Hinblick auf das operative Geschäft als auch bezüglich seiner strategischen Ausrichtung auszeichnet [Wrob+15, 370-374].

Die Mehrzahl der Industrieunternehmen sieht einer internationalen Umfrage von 2016 zufolge die Datenanalyse als kritischen Erfolgsfaktor. Als wichtigste Anwendungsdomänen werden die vorausschauende und vorbeugende Wartung von Maschinen und Anlagen noch vor der Auswertung von Kunden- und Marketingdaten genannt [LPDK16, 8]. Mehrere verwandte Studien zeigen vergleichbare Befunde [Wrob+15, 372-373]. Auch im Mittelstand werden zunehmend größere Datenbestände ausgewertet [BeUB16, 59-64]. Viele Unternehmen beklagen jedoch verschiedene Umsetzungsbarrieren [Wrob+15, 374]. Aus technischer Sicht bestehen Schwierigkeiten vor allem mit der Datenqualität und mit der Interoperabilität der Komponenten der Analyseinfrastruktur. Aus organisatorischer Sicht besteht einerseits Mangel an Analyseexperten, andererseits an geeigneten Methoden und Vorgehensmodellen [Wrob+15, 374], um wirklich Nutzen aus den vorliegenden Daten zu schöpfen. Aus fachlicher Sicht wird als größte Herausforderung die Definition klarer Anwendungsfälle und Einsatzszenarien genannt [LPDK16, 10]. Die letzten beiden Aspekte werden in der vorliegenden Arbeit genauer untersucht.

1.1 Problemstellung

Die Auswahl, Kombination und Konfiguration der für solche Auswertungen erforderlichen Datenanalyseverfahren variiert mit den jeweiligen Entscheidungssituationen und stellt eine schlecht strukturierte Aufgabe dar, für die – je nach Problemstellung – oft nur in sehr eingeschränktem Umfang Theoriewissen verfügbar ist. Die zur Analysedurchführung realisierten Analyseprozesse sind demzufolge mitunter überaus komplex.

So nennt das Ergebnis einer Expertenbefragung von YANG & WU als eines der zehn drängendsten Forschungsprobleme die bessere Unterstützung des Analyseprozesses durch eine Methodik bzw. durch die Automatisierung der Prozessgestaltung [YaWu06, 602f.]. Als weitere Herausforderung wird die Entwicklung einer einheitlichen Theorie des Data Mining genannt, die Verfahren und Ansätze aus der Statistik, dem Maschinellen Lernen und der Datenbanktechnik vereinen soll [YaWu06, 596]. CAO ET AL. konstatieren eine Lücke zwischen Anwendung und Technik und propagieren ein „Domain-Driven Data Mining“ als datenanalytisches Paradigma [CYZZ10, 16].

In jüngerer Zeit veröffentlichte Forschungsagenden enthalten – offensichtlich im Lichte des durch Data Science stark beförderten explorativen Ansatzes und der von unstrukturierten Daten auferlegten Verarbeitungsanforderungen [Baro13, 73], [Wrob+15, 370-372] – kaum anwendungs- bzw. prozessorientierte Themen (vgl. z.B. [FaBi12, 2f.], [NiSV14], [Wrob+15, 374-376]), wenngleich deren Relevanz zusammen mit der Bedeutung der Datenanalyse eher gewachsen ist. So stellen auch KRIEGEL ET AL. einen verstärkten Bedarf an Nutzerunterstützung für zunehmend komplexere Auswertungen fest [Krie+07, 93]. In diesem Sinne positioniert ZIMMERMANN die Auswahl geeigneter Parameterwerte und die Einbeziehung fachlich-inhaltlicher Aspekte bei der Ergebnisinterpretation als wichtige, aktuell ungelöste Probleme [Zimm14].[1]

[1] Die Wirtschaftsinformatik sieht aktuell nicht-technische Forschungsthemen im Bereich analytischer Systeme u.a. in der Integration mit Geschäftsprozessen, in der flexiblen, benutzergesteuerten Entwicklung sowie in Fragen der Governance [Baar+14, 15-17].

In der Statistik werden anwendungsorientierte Themen schon seit längerer Zeit diskutiert. So weist HAND 1993 in einem viel beachteten Vortrag vor der ROYAL STATISTICAL SOCIETY auf verbreitete Probleme bei der Abbildung des Anwendungsproblems auf eine statistische Fragestellung hin, die er „Fehler der dritten Art" nennt: „giving the right answer to the wrong question" [Hand94]. Er verortet das Problem der Identifizierung der sachlichen Forschungsfrage innerhalb der statistischen Strategie, die über die Auswahl adäquater Verfahren, ihre korrekte Anwendung und die richtige Interpretation ihrer Ergebnisse Auskunft gibt. HUBER fordert 1997, die Anstrengungen von der Entwicklung immer neuer, in der Praxis aufgrund nicht vorhandenen Bedarfs oder unrealistischer Modellannahmen nicht anwendbarer Algorithmen hin zu brauchbaren Meta-Methodiken zu lenken: „(...) I believe that we must begin to think about replacing statistics and statistical methodology by meta-statistics and meta-methodology. We need to create meta-methods for producing and investigating ad hoc solutions to ad hoc problems" [Hube97, 189].

Wie die geschilderten Schwierigkeiten und Herausforderungen zeigen, sind zur Durchführung erfolgreicher Datenanalysen offensichtlich zahlreiche Aspekte zu berücksichtigen, die von der Problemspezifikation über die Prozessgestaltung einschließlich der Auswahl und Konfiguration geeigneter Aufgabenträger bzw. Verfahren bis hin zur Interpretation und Bewertung der Ergebnisse reichen. Diese Beobachtung legt ein umfassendes Konzept zum Management von Datenanalyseprozessen nahe, das deren systematische Planung, Steuerung und Revision abdeckt.

Ein solches Konzept ist derzeit nicht bekannt. Vielmehr existieren mehrere spezifische Ansätze, die jeweils nur Forschern und Praktikern aus den jeweiligen analytischen Disziplinen geläufig sind, obwohl sie inhaltlich und strukturell zum Teil große Ähnlichkeit aufweisen und häufig sehr hohe Relevanz besitzen. Da in der Praxis typischerweise mehrere konkrete Ausprägungen der Datenanalyse kombiniert zum Einsatz kommen, erscheint ein interdisziplinäres Konzept hilfreich.

1.2 Zielsetzung

Diese Situation legt es nahe, existierende Arbeiten aus den relevanten Disziplinen zusammenzutragen, weiterzuentwickeln und in einem gemeinsamen Bezugsrahmen zu vereinen. Die vorliegende Arbeit stellt den Versuch dar, hierfür einen Vorschlag zu unterbreiten. Als Leitkriterien dienen die unmittelbare Anwendbarkeit in der betrieblichen Praxis und die methodisch-theoretische Fundierung des Ansatzes. Ganz im Sinne HUBERs soll er geeignet sein, die Lösung konkreter Sachprobleme im betrieblichen Umfeld datenanalytisch effektiv zu unterstützen.

Datenanalyseprozesse im Sinne höherer Lösungsverfahren zur Bewältigung von Datenanalyseproblemen bilden das Untersuchungsobjekt dieser Arbeit. Datenanalyseprozesse werden als zielgerichtete Folgen von Aufgabendurchführungen (Vorgängen) verstanden, die Analysedaten in problemrelevante Informationen transformieren und dabei auf eine Menge von Ressourcen (Aufgabenträger zur Realisierung geeigneter Datentransformationsverfahren) zurückgreifen.

Untersuchungsziel ist die **Konzipierung einer Methodik zum ganzheitlichen Management von Datenanalyseprozessen**, die einen Modellierungsansatz zur Repräsentation der Prozesse, ein Architekturmodell von Datenanalyseprozessen sowie ein Vorgehensmodell mit detaillierten Vorgaben zur Planung, Steuerung und Revision (Kontrolle) von Datenanalyseprozessen umfasst. Hierbei wird besondere Aufmerksamkeit auf folgende Aspekte gelegt:

- *Integration verschiedener Ansätze der Datenanalyse*: Die Methodik soll dem Anspruch genügen, möglichst alle in der betrieblichen Realität gängigen und relevanten Analyseansätze aufnehmen zu können. Hierzu zählen insbesondere die Methoden der beschreibenden und schließenden Statistik, der explorativen Datenanalyse, der empirischen Sozialforschung, der Datenbanktechnik einschließlich des On-Line Analytical Processing (OLAP), des Knowledge Disvovery in Databases (KDD/Data Mining) und der Data Science sowie unterstützende Ansätze, wie etwa die Datenvisualisierung, das Web- und Information Retrieval.

- *Einbeziehung der Problemspezifikation, der Prozesskonstruktion und der Ressourcen*: Aufgrund der Zielgerichtetheit von Datenanalyseprozessen ist neben der Gestaltung der Prozesse selbst gerade auch ihre systematische Ausrichtung auf Analyseziele und deren Herleitung aus anwendungsbezogenen (betrieblichen) Problemstellungen zu betrachten (Problemspezifikation). Ebenso sind die für die Prozesskonstruktion verfügbaren Ressourcen zu behandeln. Die vier Aspekte Anwendung, Analyse, Prozess, Ressourcen definieren zugleich die Ebenen der Datenanalysearchitektur.
- *Anwendbarkeit in der betrieblichen Praxis*: Die Entwicklung der Methodik soll konsequent mit dem Ziel ihrer tatsächlichen Anwendbarkeit in der betrieblichen Realität erfolgen. Hieraus folgen unmittelbar Anforderungen an die Verständlichkeit, Übersichtlichkeit und Erweiterbarkeit bzw. Anpassbarkeit an spezifische Kontexte. Insbesondere soll auf formale Repräsentationsformen und rigorose Vorgaben zugunsten von Flexibilität und Offenheit weitestgehend verzichtet werden.

Wird ein Ansatz zum Management von Datenanalyseprozessen angestrebt, so ist zu untersuchen, ob die Konzepte des allgemeinen Prozessmanagements auf die Datenanalyse übertragbar sind und inwiefern dies vor dem Hintergrund des praktischen Einsatzes sinnvoll ist.

Weiterhin sind im Lichte der in Abschnitt 1.1 geschilderten Problemstellung insbesondere folgende Fragen zu beantworten:

- Wie kann die Ableitung einer analytischen Fragestellung aus dem Anwendungsproblem methodisch gestützt und zugleich flexibel und verständlich gelingen?
- Wie kann die Zielorientierung überwacht und sichergestellt werden, um den von HAND geschilderten „Fehler der dritten Art“ zu vermeiden?
- Welche Ansätze eignen sich zur Planung von häufig überaus komplexen Datenanalyseprozessen, bei der zahlreiche Einflussfaktoren zu berücksichtigen sind?

- Nach welchen Kriterien und unter Einbeziehung welcher Bewertungsobjekte kann eine ganzheitliche Interpretation und Beurteilung von Datenanalysen aus fachlicher Sicht erfolgen?

Die Entwicklung eines Werkzeug-Konzepts bzw. eines -Prototyps ist nicht das Ziel dieser Arbeit. Gleichwohl wird angesichts der Komplexität von Datenanalyseprozessen schnell deutlich, dass ihr Management von Erfahrungswissen und fallspezifischem Wissen über die aktuelle Analyse profitieren kann, das nur mithilfe eines Software-Werkzeugs effektiv zu erfassen und zu verwalten ist. Zur Auflösung dieses Dilemmas wird eine zweigleisige Strategie verfolgt. Einerseits ist der Modellierungsansatz derart auszulegen, dass er prinzipiell in der Lage ist, das verfügbare Wissen abzubilden. Hierzu ist eine detaillierte Modellierung der Attributebene der Metaobjekttypen durchzuführen, um entsprechende Repräsentationskonstrukte vorzusehen. Andererseits ist die Methodik so zu gestalten, dass sie prinzipiell auch manuell nutzbar ist. Dies ist zum einen durch einen Modellierungsansatz erreichbar, der mit wenigen, intuitiv verständlichen Metaobjekttypen auskommt. Zum anderen sollte das Handlungsschema der Methodik mit Empfehlungen und Heuristiken für den Analytiker angereichert sein, um diesen zusammen mit den Metaphern des Modellierungsansatzes in die Lage zu versetzen, Analyseprozesse unabhängig von einem spezifischen Werkzeug zu managen.

Die Arbeit zielt damit auf den Entwurf eines robusten methodischen Rahmens, der bei Bedarf neu ausgefüllt bzw. erweitert werden kann. Dieser kann zugleich als Grundlage zur Entwicklung eines Software-Werkzeugs dienen, das eine umfassende Unterstützung der Methodik realisiert.

Darüber hinaus soll die Arbeit Beiträge zu den Grundlagen der Datenanalyse beisteuern und ein Instrumentarium für den inner- und überbetrieblichen Diskurs über Informationsbedarf und Datenanalyse bereitstellen.

1.3 Forschungsansatz

Die Entwicklung einer Methodik zum Management von Datenanalyseprozessen stellt ein Konstruktionsproblem im Sinne der gestaltungsorientierten Wirtschaftsinformatik dar [Sinz10, 28]. Sie erfolgt primär deduktiv in drei Schritten [Öste+10, 4].

Im ersten Schritt erfolgt die *Analyse* der Grundlagen und Gestaltungsoptionen von Datenanalyseprozessen. Zunächst werden Wesen, Ziele und Vorgehen bei der Datenanalyse durch Auswertung der Literatur beschrieben. Hierbei wird gezielt auf Grundlagenliteratur zurückgegriffen, um Spezifika einzelner Vorschläge sowie technik- oder methodenzentrierte Einflüsse moderner Ansätze weitestgehend auszublenden. Durch Betrachtung der Grundlagen des Prozessmanagements werden Aufgaben und Kriterien erarbeitet, die in die Methodik einfließen sollen. Anschließend wird auf Grundlage der Untersuchung typischer Vorgehensschemata aus verschiedenen Disziplinen der Datenanalyse ein allgemeingültiges Vorgehensmodell konzipiert, das die Handlungskomplexität bei der Datenanalyse durch Unterstützung evolutionärer Prinzipien beherrschbar machen soll. Hierbei wird auf Erkenntnisse aus der Softwaretechnik zurückgegriffen, um dort erlangte Erfahrungen für die Datenanalyse nutzbar zu machen. Das Vorgehensmodell soll in wenige Phasen gegliedert sein, um die direkte Anwendbarkeit in der Analysepraxis zu fördern. Es dient als Bezugsrahmen für den anschließenden *Entwurf* der Methodik.

Im zweiten Schritt wird das Vorgehensmodell zu einer umfassenden Methodik erweitert. Dazu wird zunächst ein Modellierungsansatz zur Repräsentation von Datenanalyseprozessen entwickelt, dessen Rahmen eine viergliedrige Analysearchitektur aus Anwendungs-, Analyse-, Prozess- und Ressourcenebene bildet. Im Anschluss werden die aus dem Prozessmanagement abgeleiteten Phasen Planung, Steuerung und Revision (Kontrolle) von Datenanalyseprozessen detailliert erläutert. Für jede Phase wird ein Handlungsschema konzipiert, das jeweils einzelne Phasen des zuvor entwickelten Vorgehensmodells konkretisiert und eine anwendungsorientierte Unterstützung bei der Realisierung von Datenanalysen liefern soll. Zur Konzipierung der Methodik werden einschlägige verwandte Arbeiten ausgewertet. Darüber hinaus wird

Grundlagenliteratur zur Unternehmensplanung, zur Entscheidungstheorie, zur Wiederverwendung in der Softwaretechnik und zur Handlungsplanung der Künstlichen Intelligenz auf Lösungsbeiträge untersucht.

Der dritte Schritt dient der Evaluation der vorgestellten Methodik, indem diese auf ein umfassendes Fallbeispiel angewandt und bezüglich ihrer Stärken, Schwächen und Grenzen bewertet wird. Hierbei wird insbesondere untersucht, inwiefern die Methodik ohne Werkzeugunterstützung anwendbar ist und welche Weiterentwicklungspotenziale sich aufzeigen.

1.4 Aufbau der Arbeit

Die Arbeit ist in drei Hauptteile sowie einen Anhang gegliedert, welche dieser Einleitung folgen. Die drei Hauptteile repräsentieren die in Abschnitt 1.3 genannten Schritte und behandeln die Analyse-, Entwurfs- bzw. Evaluationsphase.

Teil A betrachtet die **Grundlagen und Gestaltungsoptionen von Datenanalyseprozessen**.

In Kapitel 2 werden die **Datenanalyse und Datenanalyseprozesse** untersucht, um die Frage nach Ziel und Wesen der Datenanalyse zu beantworten. Datenanalyse wird hierbei als Instrument zur Informationsversorgung positioniert und als Aufgabe definiert, bevor ein Überblick über wichtige Ansätze und Ausprägungen erfolgt. Anschließend wird erörtert, inwiefern die Datenanalyse sich als Prozess darstellt und die Anwendung der Instrumente des Prozessmanagements möglich ist.

Nachdem der Prozesscharakter der Datenanalyse erkannt wurde, beantwortet Kapitel 3 mit einer **Bestandsaufnahme und Empfehlungen zum Vorgehen bei der Datenanalyse** die Frage nach den Eigenschaften solcher Prozesse und ihrer Handhabung. In einer Analyse von Struktur und Verhalten wird festgestellt, dass in der Realität stark von der idealen Vorgehensweise abgewichen wird, wie sie von Prozessmodellen nahegelegt wird. Als Ursache wird die Komplexität realer Analyseabläufe identifiziert, zu deren Handhabung im Anschluss geeignete Maß-

nahmen erörtert und ein evolutionäres Vorgehensmodell für die Datenanalyse vorgestellt werden.

Teil B entwickelt eine **Methodik für das Management von Datenanalyseprozessen**.

In Kapitel 4 wird zunächst ein umfassender Ansatz zur **Modellierung von Datenanalyseprozessen** entwickelt, der die Grundlage für die Methodik bildet. Aus Anforderungen an die Repräsentation von Analyseprozessen wird eine Datenanalysearchitektur hergeleitet, die sich in eine Anwendungsebene, eine Analyseebene, eine Prozessebene und eine Ressourcenebene gliedert. Die Ebenen werden jeweils einzeln beschrieben und um die Darstellung spezieller Sichten auf Datenanalyseprozesse ergänzt.

Kapitel 5 behandelt die methodische **Planung von Datenanalyseprozessen**. Zunächst erfolgt die Einordnung der Prozessplanung als Gestaltungsaufgabe, für die eine Planungsstrategie entwickelt und anhand entsprechender Vorschläge aus der Literatur vier Basisansätze identifiziert werden. Im Anschluss werden Handlungsschemata zur Problemspezifikation (Planung auf Anwendungs- und Analyseebene der Datenanalysearchitektur) sowie zur Prozessspezifikation (Planung auf Prozessebene) vorgestellt und jeweils mit Empfehlungen und Heuristiken erläutert.

In Kapitel 6 wird die **Steuerung von Datenanalyseprozessen** diskutiert, die als Lenkungsaufgabe mit Gestaltungsanteilen charakterisiert wird. Das zugehörige Handlungsschema beschreibt Aufgaben und Vorgehen bei der Prozesssteuerung, für die abschließend drei Ansätze (Steuerungsmodi) diskutiert werden, wie sie in der Datenanalyse von Bedeutung sind.

Kapitel 7 komplettiert die Methodik mit der ganzheitlichen **Revision von Datenanalyseprozessen**. Aufgrund ihres Gestaltungsanteils wird die Kontrollaufgabe als Revision bezeichnet. Das zugehörige Handlungsschema gliedert sich in drei Abschnitte, die sich der Beurteilung der durchgeführten Datenanalysen (Revision i.e.S.), der ganzheitlichen Evaluierung des analytisch gestützten Projekts (Revision i.w.S.) sowie

der Erfahrungssicherung und Prozessverbesserung (lernende Revision) widmen.

Teil C der Arbeit nimmt eine **Evaluation** der vorgestellten Methodik vor.

Hierzu wird in Kapitel 8 eine **Fallstudie** zum Kundenauftragsrückgang in der Konsumgüterindustrie dargestellt, welche die Anwendbarkeit der Methodik untersucht und deren Potenziale und Grenzen herausstellt.

Kapitel 9 beschließt die Arbeit mit **Fazit und Ausblick**.

Der umfangreiche **Anhang** enthält weiterführende Angaben, die zum tieferen Verständnis einzelner Aspekte der Arbeit sowie zur praktischen Anwendung der Methodik hilfreich sind. Anhang A1 gibt einen Überblick über gängige Datenanalysemethoden. Anhang A2 ordnet die in Kapitel 3 vorgestellten Maßnahmen zur Bewältigung der Analysekomplexität allgemeinen Prinzipien der Komplexitätshandhabung zu. Anhang A3 fasst die Phasen und Aufgaben des Vorgehensmodells zur Datenanalyse in einer Übersicht zusammen. Anhang A4 enthält detaillierte Attributschemata zum Modellierungsansatz aus Kapitel 4. Anhang A5 zeigt beispielhafte Deskriptoren, wie sie zur Beschreibung verschiedener Sachverhalte im Rahmen der Methodik eingesetzt werden. Anhang A6 konkretisiert Bedingungen zur korrekten Planung von Analyseprozessen, die in Kapitel 5 genannt werden. Anhang A7 erläutert ausgewählte Kriterien zur Beurteilung von Analyseergebnissen. Anhang A8 fasst die Aufgaben des Handlungsschemas der Methodik in einer Übersicht zusammen.

1.5 Konventionen

Zur besseren Übersichtlichkeit werden Aufgaben des Vorgehensmodells bzw. Schritte der Handlungsschemata jeweils mit einem Buchstaben-Zahlencode gekennzeichnet, wobei der Buchstabe jeweils einen Bezug zur betroffenen Aufgabe bzw. Phase, die Zahlen jeweils die Position der Aufgabe im Schema repräsentiert, z.B. P1 (Aufgabe „Planung der Datenanalysephase“) und P1.3 (Teilaufgabe „Bestimmung einer Verfahrensklasse“ von P1).

Wichtige ***Begriffe***, die innerhalb eines Absatzes definitorisch erklärt werden, sind fett und kursiv gesetzt. `Modellierungsobjekte` des Modellierungsansatzes und deren `Attribute` werden bei expliziter Referenz in nicht-proportionaler Schriftart gesetzt. Code-Fragmente oder Regeln sind in `Courier` dargestellt. Darüber hinaus sind `Bezeichner` oder `formale Ausdrücke` ebenfalls nicht-proportional gesetzt, sofern sie besonders betont werden sollen.

Teil A:

Grundlagen und Gestaltungsoptionen von Datenanalyseprozessen

Der erste Teil dieser Arbeit legt das Fundament für den späteren Entwurf einer Methodik zur Planung von Datenanalyseprozessen. Er gliedert sich in zwei Kapitel. Die Grundlagen von Datenanalysen und Datenanalyseprozessen untersucht Kapitel 2. Es werden die Ziele, grundlegende Ansätze und wichtige Ausprägungen der Datenanalyse betrachtet. Anschließend wird dargestellt, inwiefern sie Prozesscharakter besitzt, und wie sich die Instrumente und Kriterien des Prozessmanagement für die Anwendung auf Datenanalyseprozesse grundsätzlich eignen.

Kapitel 3 macht eine Bestandsaufnahme und Empfehlungen zum Vorgehen bei der Datenanalyse. Nach Darstellung von Struktur und Verhalten von Analyseprozessen wird der Umgang mit Komplexität bei der Prozessdurchführung zunächst aus allgemeiner Sicht erörtert, bevor als erstes Zwischenergebnis ein evolutionäres Vorgehensmodell für die Datenanalyse vorgestellt wird. Dieses wird in Teil B zu einer Methodik konkretisiert und mit Inhalten ausgefüllt.

2 Datenanalyse und Datenanalyseprozesse

Das vorliegende Kapitel beleuchtet Wesen und Inhalt der Datenanalyse und untersucht deren Prozesscharakter. Abschnitt 2.1 leitet eine aufgabenorientierte Definition her. In Abschnitt 2.2 werden Basisansätze und wichtige Ausprägungen der Datenanalyse beleuchtet. Abschnitt 2.3 untersucht die Konzeption von Datenanalyseprozessen. Gegenstand von Abschnitt 2.4 ist das Prozessmanagement für Datenanalysevorhaben.

2.1 Datenanalyse als Instrument der Informationsversorgung

Dieses Teilkapitel leitet das Grundverständnis der Datenanalyse her, wie es dieser Arbeit zugrunde gelegt wird. Abschnitt 2.1.1 betrachtet den Datenanalysebegriff im Allgemeinen, und Abschnitt 2.1.2 definiert grundlegende Begriffe. Zur Präzisierung des Begriffsverständnisses untersucht Abschnitt 2.1.3, welche Ziele die Datenanalyse verfolgt, um schließlich eine zusammenfassende Definition abzuleiten (Abschnitt 2.1.4).

2.1.1 Der Datenanalysebegriff

Unter dem Oberbegriff Datenanalyse finden sich in der Literatur zahlreiche Beschreibungen variierender Ansätze und Methoden, deren Ursprung in unterschiedlichen Disziplinen liegt. Auf eine Definition wird meist verzichtet, da der Zweck des Herausarbeitens bedeutsamer Aussagen aus umfangreichen Datensammlungen offensichtlich scheint [LaSi82, ix]: „Data analysis is concerned with the analysis of data – of any kind, by any means“ [Hube11, 1]. Gleichwohl lohnt eine nähere Betrachtung, um das Wesen der Datenanalyse genauer zu fassen.

Der aus dem Griechischen stammende Begriff *Analyse* bezeichnet die Auflösung eines Ganzen in seine Bestandteile sowie die genaue Untersuchung seiner Einzelheiten [Kien82, 27]. Zweifellos handelt es sich bei der Datenanalyse um eine Form der Untersuchung von durch Daten repräsentierten Sachverhalten [BaGü04, 525], [Zimm95, 3], [Beck01, 48]. PYLE sieht in der Erhebung von Daten eine Form der Zerlegung der Welt in messbare Merkmale [Pyle03, 35]. Nach DREIER kann mit

Datenanalyse „das Ordnen, Zerlegen und Verarbeiten von Daten bezeichnet werden, mit dem Ziel, Antworten auf Forschungsfragen zu finden“ [Drei94, 150].

Somit ist Datenanalyse eine zielgerichtete Datenverarbeitung [Hand99, 1f.], die auf die Beantwortung bestimmter Untersuchungsfragen gerichtet ist [Drei94, 150]. Eine weitergehende Vorstellung begreift sie als systematische Datenerhebungs- und Auswertungsstrategie, die neben der Analyse auch die Datenerfassung sowie die Interpretation der Untersuchungsergebnisse einschließt [EnMT95, 2]. Eine undifferenziertere Auffassung subsumiert unter Datenanalyse „alle Operationen“, die auf zu analysierenden Daten durchgeführt werden können [BaGü04, 63].

Demgegenüber stehen engere Auslegungen, die Datenanalyse auf statistische Auswertungen [LaSi82, ix] oder auf die Explorative Datenanalyse (EDA) [BrFM97] eingrenzen. Auch die aktuell dominierenden englischsprachigen Begriffe wie z.B. Data Analytics oder Data Science implizieren in der Regel spezielle Ausprägungen.[2] Durch Unterordnung der Datenanalyse etwa unter die Statistik wird jedoch die „Möglichkeit einer alternativen, nicht primär statistisch orientierten Betrachtung der Daten ausgeblendet“ [Drei94, 324]. Diese Arbeit nimmt daher eine aufgabenorientierte Perspektive ein, die von konkreten Erscheinungsformen abstrahiert.

2.1.2 Exkurs: Wissen, Information und Daten

Vor einer weiteren Annäherung an die Datenanalyse sind zunächst drei grundlegende Begriffe zu klären, die für die Datenanalyse eine zentrale Rolle einnehmen.

[2] Bereits in der Vergangenheit wurden wiederholt diverse Methodenklassen zur Definition einer „modernen“ oder „intelligenten“ Datenanalyse herangezogen., so etwa computerunterstützte Ansätze wie Data Mining [Pete95], [Zimm95], [ChDü98], [Runk00], oder auch neuere Verfahren der Statistik [LaSi82].

2.1.2.1 Wissen

Gemäß dem Rationalitätsprinzip nach NEWELL [Newe82, 105] lässt sich Wissen als Gesamtheit verhaltenswirksamer Erkenntnisse eines Individuums definieren. Wissen ist somit stets subjektiv und beeinflusst das Verhalten seines Trägers. Nach dieser Sichtweise repräsentiert Wissen sowohl Sachverhalte („wahre" Erkenntnis) als auch Urteile über Sachverhalte („gewisse" Überzeugung) [Wild74, 119], [Oppe95, 195] und kann in der zweiten Ausprägung objektiv falsch sein. Daher werden im Folgenden zwei Wissensbegriffe unterschieden: ***Wissen i.e.S.*** ist die Menge aller von einem Wissensträger als wahr angenommenen Aussagen, die tatsächlich wahr sind. ***Wissen i.w.S.*** schließt zusätzlich seine Überzeugungen als Menge jener Aussagen ein, von denen er glaubt, sie seien wahr. Wissen kann durch Erfahrung, durch logische Prozesse (Nachdenken) oder durch Information erworben werden [Reim91, 6f.], [Oppe95, 8].

2.1.2.2 Information

Unter Information wird im Allgemeinen eine Wissensmitteilung oder Nachricht verstanden [Lyre02, 11f.] und kann sowohl den *Vorgang des Informierens* (Benachrichtigung) als auch den *Inhalt des Informierens* (Nachricht) bezeichnen [Oppe95, 5f.]. Die Informationstheorie kennt verschiedene Ansätze zur Annäherung an den Begriff, aus denen WERSIG die folgende philosophisch-theoretische Fundierung ableitet [Wers96, 221-224]:

- *Information als Handlung*: Information ist die Beschaffung oder Übermittlung des in einer bestimmten Situation benötigten Wissens.
- *Information als kommuniziertes Wissen:* Information reflektiert Wissen über die Welt.
- *Information als Veränderung:* Der Eingang neuen Wissens löst beim Empfänger eine Veränderung aus, die als Information bezeichnet wird.

Zur Erfassung des Informationskonzepts ist das Modell der Semiotik (Zeichenlehre) nach MORRIS [Morr38] gebräuchlich, das die Ebenen Syntaktik, Semantik und Pragmatik unterscheidet. Die *Syntaktik* betrifft das Auftreten einzelner Informationseinheiten, repräsentiert durch Signale oder Zeichen, und deren Beziehungen [Lyre02, 16f.]. Die von SHANNON [Shan48] begründete mathematische Kommunikationstheorie, die oft als Ausgangspunkt informationstheoretischer Überlegungen dient, behandelt potenzielle (noch nicht übertragene) Information unter syntaktischen Gesichtspunkten [Lyre02, 31]. Im zugehörigen Kommunikationsmodell übermittelt ein Sender Nachrichten an einen Empfänger [ShWe49], [Lyre02, 12]. Die *Semantik* betrifft die Bedeutung der Zeichen, die stets auf den Empfänger bezogen ist (Subjektgebundenheit der Information [Lyre02, 209]). Dieser verarbeitet die eingehenden Nachrichten und misst ihnen eine Bedeutung bei. Diese Verarbeitung umfasst typischerweise vom situativen Kontext und vom Vorwissen abhängige Interpretationsvorgänge [LiBa90, 3], [Lyre02, 206].

Die *Pragmatik* beschreibt die Wirkung der Information. Sender und Empfänger erfahren mit jedem Informationsaustausch eine Veränderung bezüglich ihrer semantischen Ebenen. Beim Empfänger führt das übertragene Wissen zu einer Reduzierung von Ungewissheit und nimmt potenziell Einfluss auf künftige Entscheidungen und Handlungen [Wiem73, 4], [PiRe91, 252]. Diese Wirkung entspricht der Intention der Übertragung.[3] Gleichzeitig erhöht sich das Informationspotenzial des Empfängers, da er das erworbene Wissen weiter kommunizieren kann. Beim Sender hingegen ist eine Verringerung des Informationspotenzials bezogen auf den aktuellen Kommunikationspartner festzustellen, da die Wiederholung einer Nachricht für diesen keine Information mehr darstellt (Redundanz) [Lyre02, 205]. Ob die intendierte Wirkung tatsächlich eintritt, kann erst nach der Übertragung beim Empfänger festgestellt werden und hängt von dessen Annahmen,

[3] Die Absichtlichkeit der Übertragung kennzeichnet den Unterschied zwischen unverständlichem Rauschen und verständlicher Information [LiBa90, 31]. Sie besteht z.B. im Hervorrufen einer bestimmten Prägung des Denkens und Handelns beim Empfänger [PiRe91, 252].

Erwartungen und Überzeugungen ab [LiBa90, 13]. Er muss die Information ferner korrekt in vorhandenes Wissen einordnen. Kommuniziertes Wissen ist somit nicht zugleich Information. Information ist keine absolute Größe, sondern existiert nur relativ in Bezug auf die Differenz der semantischen Ebenen von Sender und Empfänger [Lyre02, 32].

Information kann demnach als durch Kommunikation bewirkte Verringerung der Ungewissheit eines Individuums interpretiert werden. Die Ungewissheit entsteht durch die Notwendigkeit zum rationalen Handeln in einer Problemsituation, wofür zunächst nicht ausreichend Wissen zur Verfügung steht [Wers96, 223]. Diese Sichtweise entspricht der in der Betriebswirtschaftslehre vorherrschenden Definition von WITTMANN, der Information als zweckorientiertes Wissen ansieht, genauer als „Wissen, das zur Erreichung eines Zweckes, nämlich einer möglichst vollkommenen unternehmerischen Disposition eingesetzt wird" [Witt59, 14]. Sie stellt den pragmatischen Aspekt in den Vordergrund [PiRe91, 252]. Die Informatik hingegen beschränkt sich auf den semantischen Aspekt, wonach Information die Bedeutung von Zeichen oder Daten repräsentiert, welche durch Interpretations- oder Dekodierungskonventionen bestimmt wird [ANSI66], [Oppe95, 6].[4] Für diese Arbeit wird der pragmatische Informationsbegriff zugrunde gelegt. Zugleich wird die in der Sicht der Informatik zum Ausdruck kommende Rolle von Daten als Rohmaterial betont, aus dem Information gewonnen werden kann.

2.1.2.3 Daten

Die Informatik setzt Informationen gelegentlich mit Daten gleich [Müll00, 5]. So verwendet z.B. DATE die Begriffe explizit synonym [Date95, 4]. Für die Ziele dieser Arbeit ist der Datenbegriff der empirischen Forschung maßgeblich, die Daten als Träger von Aussagen über die empirisch erfassbare Wirklichkeit versteht [Drei94, 3].

---

[4] In der englischsprachigen Literatur sind Informationen im Allgemeinen Daten, die für ihren Empfänger im Hinblick auf dessen Entscheidungen bzw. Handlungen eine Bedeutung haben [Oppe95, 6].

Ein genaueres Verständnis vermittelt die Datentheorie [Coom67], [Galt67]. Daten sind stets Ergebnis einer gezielten Erfassung, bei der nur situativ relevante Sachverhalte Berücksichtigung finden [Jaco91, 14]. Der zugehörige systematische Erhebungsprozess ist jeweils auf ein Untersuchungsproblem abgestimmt und zielt auf die Protokollierung der gemachten Beobachtungen in „geeigneter und abrufbarer Form" [Drei94, 130f.]. Hierbei werden Untersuchungsobjekte (Untersuchungseinheiten, Merkmalsträger, Fälle) im Hinblick auf interessierende Eigenschaften (Variablen, Merkmale) untersucht und deren Ausprägungen (Werte) als Daten erfasst [Drei94, 132], [Benn94, 8]. Erfolgt die Datenerfassung mithilfe vordefinierter Merkmals- und Wertemengen, liefert sie sogenannte strukturierte Daten (z.B. in Tabellenform),[5] andernfalls unstrukturierte Daten (z.B. Text-, Bild-, Video- oder Tonaufzeichnungen). Auch sie beschreiben interessierende Sachverhalte, folgen aber häufig weniger spezifischen Untersuchungszielen (z.B. erfassen Videoaufzeichnungen allgemein das Verhalten von Personen).

Daten stellen eine Repräsentation der untersuchten Welt in geeignete Symbole oder Zeichen dar [Hand99, 8], [Oppe95, 7], [FeSi13, 144f.]. Diese Modellabbildung ist Gegenstand der *Sigmatik*, einer oft vernachlässigten Ebene des semiotischen Modells [Ball00, 13f.]. Die Bedeutung der Symbole kann durch eine zur Repräsentation passende Interpretationsvorschrift erschlossen werden, die Korrespondenzregeln zwischen Zeichen und repräsentierten Gegenständen spezifiziert. Die Interpretation erfolgt stets kontextabhängig [Reim91, 9f.], [Drei94, 42-44].[6]

In dieser Arbeit werden ***Daten*** als zielgerichtet und systematisch erfasste Aussagen über einen ausgewählten Gegenstandsbereich verstanden. Sie umfassen damit auch Aufzeichnungen zu Überzeugungen und Empfindungen, wie z.B. Beurteilungen oder Beschwerden. Eine Einschränkung

---

[5] Für die Erhebung strukturierter Daten ist auch der Begriff *Messung* gebräuchlich, der die systematische Zuordnung von Werten aus einer definierten Wertemenge zu Objekten bezeichnet [Jaco91, 5].

[6] So kann eine Zeichenfolge unterschiedliche Aussagen repräsentieren [FeSi13, 144]. Gleichzeitig kann ein Sachverhalt durch unterschiedliche Daten repräsentiert sein [Reim91, 10].

auf empirisch fassbare Sachverhalte erlaubt der Begriff *empirische Daten*. Die in der Informatik übliche Forderung nach maschineller Verarbeitbarkeit [Müll00, 6], [LeHM95, 200f.] wird hingegen nicht als notwendige Bedingung erachtet, wenngleich (bzw. gerade weil) die manuelle Handhabung praxisrelevanter Daten nicht realistisch erscheint.

2.1.2.4 Beziehung zwischen Wissen, Informationen und Daten

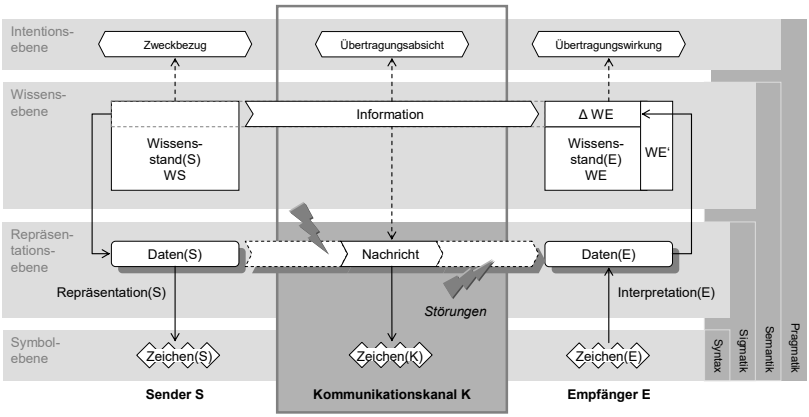


Abbildung 1: Betrachtungsebenen der Begriffe Wissen, Information und Daten (eigene Darstellung)

Abbildung 1 stellt wesentliche Zusammenhänge der erläuterten Begriffe dar. Die als Daten repräsentierten Aussagen können unmittelbares Resultat empirischer Beobachtungen oder Bestandteil des Wissens eines Individuums sein (Repräsentationsebene, Sender). Sie geben den Aussagegehalt (Wissensebene).[7] Auf Symbolebene sind Daten nach einer bestimmten Syntax geordnete Zeichen. Teile des Wissens können mit einer spezifischen Übertragungsabsicht (Intentionsebene) an einen Empfänger übermittelt werden, um dort Information auszulösen. Informationsflüsse werden in Form von aus Zeichenfolgen bestehenden

[7] Automatisch erfasste Transaktions- und Sensordaten sind stets empirisch und repräsentieren (unter der Annahme valider Messinstrumente) wahre Aussagen (vgl. Wissen i.e.S.). Von Personen produzierte Daten wie z.B. Texte können auch Überzeugungen ihrer Verfasser repräsentieren (vgl. Wissen i.w.S.).

Nachrichten realisiert. Daten dienen somit der Dokumentation von Sachverhalten und dem Transport von Information[8] [Jaco91, 14]. Die Information muss seitens des Empfängers durch Anwendung einer geeigneten Interpretationsvorschrift aus den Daten wieder gewonnen werden, indem er ihnen eine Bedeutung beimisst [HoMP01, 210f.]. Diese kann eine Erhöhung seines Wissensstands hervorrufen. Nicht übereinstimmende Repräsentations- und Interpretationsregeln zwischen Sender und Empfänger sowie auf den Kommunikationskanal einwirkende Störungen führen regelmäßig zu Verständnisproblemen, welche die Informationswirkung vereiteln [FeSi13, 144f.].

2.1.3 Ziele der Datenanalyse

Auf Grundlage der vorstehenden Überlegungen können nun die mit der Datenanalyse verfolgten Ziele auf allgemeiner Betrachtungsebene aus der Literatur herausgearbeitet werden.

2.1.3.1 Ableitung von Information und Wissen

Der Zweck jeder Datenanalyse besteht darin, durch das Herausarbeiten der in Daten enthaltenen Aussagen das Wissen über den repräsentierten Gegenstandsbereich zu erweitern [Ehre76, 19], [HeMi94, VI], [FaPS96, 8], [Vell97, 320]. Da hierbei bislang unbekannte Eigenschaften der untersuchten Objekte im Vordergrund stehen [Uthu96, 561], ist dies gleichbedeutend mit der Extraktion der in den Daten enthaltenen Informationen [AnHe85, 1], [ElPr96, 98], [TsHe04, 148] bzw. der Ableitung neuer Informationen [Zimm95, 3], [BaGü04, 63]. In der Regel steht die Analyse im Zusammenhang mit einer Problemsituation, welche eine System- bzw. Zustandsbeschreibung, eine Entscheidung oder eine Prognose erfordert [Drei94, 150], [Beck01, 48], [BaGü04, 98].

[8] Letztlich ist jede Dokumentation als spezielle Form der asynchronen Kommunikation interpretierbar.

2.1.3.2 Fokussierung und Abstraktion der Daten

Information ist für den Empfänger stets relevant und nicht-redundant (vgl. Abschnitt 2.1.2.2). Somit impliziert Datenanalyse als Informationsversorgung die „Vereinfachung und Zusammenfassung (Reduktion) einer zunächst unübersichtlichen Menge" von Daten [Drei94, 150] auf situativ interessierende Aussagen [Ehre76], [HoKl91, 338f.]. Dies lässt sich durch Fokussierung (Filterung) und Abstraktion erreichen. Wichtige Ausprägungen der Abstraktion sind Generalisierung (Ausblenden von Unterschieden) und Aggregation (Ausblenden von Komponenten bzw. Beziehungen) [FeSi13, 154].

Die Aussonderung unbedeutender Fakten oder Details, z.B. durch Verzicht auf die Reproduktion von Einzelwerten zugunsten von Aggregaten (z.B. Summen, Lage-/Streuungsmaße, Häufigkeitsverteilungen) oder durch Transformation der Merkmalsrepräsentation (z.B. Diskretisierung kontinuierlicher Werte), erleichtert die Wahrnehmung der relevanten Aussagen [OpSc84, 3], [TsHe04, 148]. Der mit einer zielgerichteten Abstraktion einhergehende Genauigkeitsverlust ist in der Regel vernachlässigbar [Ehre76, 23f.]. Daher strebt die Datenanalyse häufig nicht nach Ergebnissen, die exakt die Analysedaten wiedergeben, sondern zielt auf eine Approximation, die zugleich ausreichend verallgemeinerbar ist [ElPr96, 93]. Derartige Ergebnisse sind Voraussetzung für fundierte Prognosen und Inferenzen. Nur generalisierte, gesetzartige Aussagen erlauben die Extrapolation in Bereiche, die über die Reichweite der ausgewerteten Daten hinausgehen [Vell97, 323f.].

2.1.3.3 Ordnung des Datenkörpers durch Struktur und Beziehungen

Vor einer möglichen Aggregation müssen Beziehungen zwischen einzelnen Variablen oft erst ermittelt oder explizit hergestellt werden [Ehre76, 8, 91]. Beziehungen zeigen Existenz, Art und Stärke von Zusammenhängen zwischen Bezugsgrößen auf [HeMi94, 320], [ChGl99b, 263] und ermöglichen die fundierte Interpretation empirischer Aussagen [Ehre76, 19]. So erlaubt z.B. erst der Vergleich numerischer Werte mit Ziel-/Normgrößen deren realistische Beurteilung, und die Verknüpfung von Kunden- mit Artikeldaten eröffnet Einblicke in die Kundenpräferenzen. Die Zusammenfassung einzelner

Datenwerte (Abstraktion) und die Ermittlung von Beziehungen schaffen eine Ordnung und Strukturierung des Datenkörpers [Ehre76, 222]. Die Aufdeckung dieser Ordnung in den Daten bzw. zwischen den repräsentierten Objekten wird häufig als Ziel der Datenanalyse formuliert [Ehre76, 85], [LaSi82, ix], [Zimm95, 3].

Viele Phänomene sind auf Ebene der Individualaussagen stark irregulär. Ihre Ordnung lässt oft systematische, verallgemeinerungsfähige Beziehungen zutage treten [Ehre76, 230]. Die Datenanalyse leistet in diesem Sinne die Abtrennung der Zufallserscheinungen von bedeutsamen Regelmäßigkeiten, die in der Menge detaillierter Fakten oft nur schwer erkennbar sind [LaSi82, ix]. Beispielsweise ist das Einkommen einzelner Personen schwer abschätzbar, da es von vielerlei Faktoren abhängt. Gruppiert man die Personen nun etwa nach Berufsgruppen, so gestaltet sich die Schätzung ihres Einkommens weitaus einfacher, da die Angehörigen einer Berufsgruppe gemeinsame Merkmale aufweisen.

2.1.3.4 Herleitung von Mustern und Modellen

Die oben beschriebenen Zusammenfassungen, Beziehungen und Regelmäßigkeiten werden auch als Muster oder Modelle bezeichnet [Ehre76, 121], und die Erkennung von Mustern sowie die Generierung von Modellen bzw. deren Anpassung an gegebene Daten als wichtiges Ziel der Datenanalyse angesehen [Drei94, 150], [HeMi94, 320], [FaPS96,12].

Der Begriff ***Muster*** bezeichnet eine Aussage über eine Untermenge der Daten, die einfacher ist als die Aufzählung aller Elemente der Untermenge.[9] Diese absichtlich vage Definition soll ein möglichst breites Spektrum von Beschreibungsformen abdecken und schließt jede Art von Beziehungen und Regelmäßigkeiten ein [Biss96, 6], [FrPM91, 3]. Ein Muster ist eine lokale Struktur, die von einigen Fällen oder von einer Region des Datenraumes unterstützt wird. Ein ***Modell*** stellt eine

[9] „Given a set of facts (data) *F*, a language *L*, and some measure of certainty *C*, we define a *pattern* as a statement *S* in *L* that describes relationships among a subset *Fs* of *F* with a certainty *c*, such that *S* is simpler (in some sense) than the enumeration of all facts in *Fs*" [FrPM91, 3].

übergeordnete Struktur dar, die Beziehungen zwischen vielen Fällen zusammenfasst [Hand99, 6], [FaUt02, 28]. Es zielt auf eine umfassende Approximation des Systemverhaltens [Zimm95, 4f.]. Ein Muster kann somit, in Abhängigkeit von der Art des Modells, als Bestandteil oder als Instanziierung eines Modells angesehen werden[10] [FaPS96, 11].

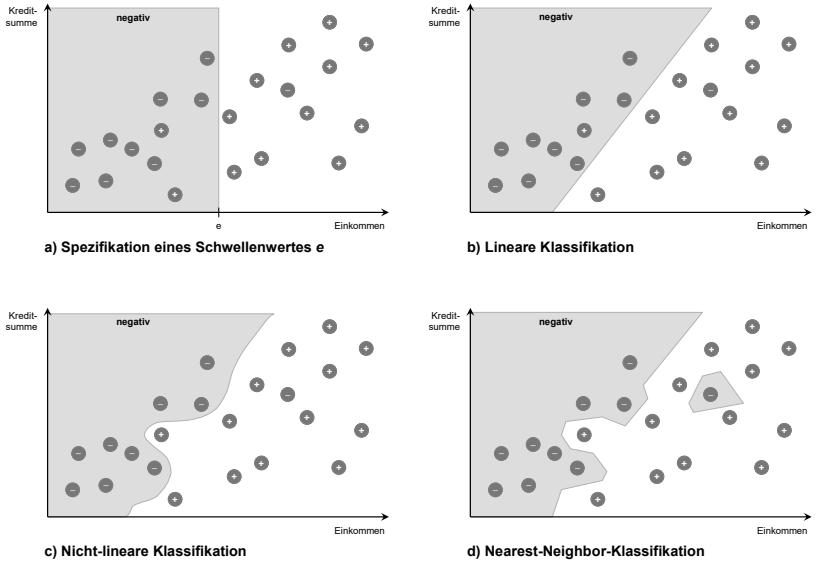


Abbildung 2: Optionen zur Strukturierung einer Datenmenge am Beispiel einer Klassifikation [FaPS96]

Die bisher identifizierten Ziele lassen sich anhand eines Beispiels [FaPS96] illustrieren. Gegeben seien Daten zu ehemaligen Kreditnehmern einer Bank, deren Tilgungsverhalten als positiv (+) oder negativ (–) charakterisiert wurde. Ziel der Analyse ist die Ermittlung von Kriterien zur Prognose des Tilgungsverhaltens, um über Annahme oder Ablehnung künftiger Kreditanträge entscheiden zu können. Zum besseren Verständnis werden nur drei als relevant erachtete Merkmale herausgegriffen und grafisch aufbereitet (Abbildung 2). Aus der

[10] Z.B. ist $f(x) = \alpha x^2 + \beta x$ ein Modell, während $f(x) = 3x^2 + x$ ein Muster darstellt [FaPS96, 11].

Darstellung lassen sich bereits Beziehungen zwischen Einkommen, Kreditsumme und Tilgungsverhalten ablesen. Nun wird der Datenbestand in zwei Klassen geordnet, wobei eine Klasse die positiven, die andere die negativen Fälle repräsentiert.

Die Abbildung zeigt vier Alternativen für eine solche Klassifikation. Während die Genauigkeit, erkennbar an der Menge korrekt zugeordneter Kreditnehmer, von a) nach d) steigt, nimmt die Verständlichkeit der Beschreibung in dieselbe Richtung ab. Die gewählte Klasseneinteilung abstrahiert von einzelnen Fällen und vermittelt Wissen über das allgemeine Tilgungsverhalten von Kreditnehmern. Die erforderliche Genauigkeit wird vom Wesen des vorliegenden Sachproblems bestimmt [Ehre76, 116]. Zur Beschreibung von Tatbeständen werden möglichst einfache Approximationen (im Beispiel etwa die Festsetzung eines Schwellenwertes im Fall a) tendenziell gegenüber exakten Ergebnissen (wie die Nearest-Neighbor-Klassifikation in Fall d) bevorzugt. Prognoseprobleme erfordern eine möglichst hohe Genauigkeit der Ergebnisse.

2.1.3.5 Überprüfung und Generierung von Hypothesen und Theorien

Mit Herleitung von Mustern oder Modellen verlässt die Datenanalyse die empirische Ebene und betritt eine durch Theorien beeinflusste, konzeptuelle Ebene [Drei94, 74f.]. ***Theorien*** sind integrierte Konstrukte aus verknüpften Erkenntnissen. Sie reflektieren das über einen Sachverhalt herrschende Verständnis und können als (vorläufig) bestätigtes Wissen angesehen werden. ***Hypothesen*** hingegen sind theoretische Formulierungen oder Behauptungen, deren Wahrheitsgehalt nicht bewiesen ist [Ehre76, 192-194]. Muster und Modelle haben zunächst rein hypothetischen Charakter und sind hinsichtlich ihres Zutreffens und ihrer Erklärungskraft noch zu testen [ElPr96, 93]. Daten gelten als Indikatoren für theoretische Konstrukte und lassen sich in dieser Relation aus beiden Richtungen betrachten [EnMT95, 3]. Vor diesem Hintergrund besteht das Ziel einer Datenanalyse entweder in der Überprüfung der Gültigkeit von Hypothesen bzw. Theorien oder in deren Generierung [Jaco91, 3], [Drei94, 3, 135].

2.1.3.6 Interpretation

Nach POPPER sind „Sätze über Beobachtungen und über Versuchsergebnisse immer *Interpretationen* der beobachteten Tatsachen ... sie [sind] *Interpretationen im Lichte von Theorien*“ [Popp82, 72]. Damit Daten Auskunft über ein theoretisches Konstrukt geben bzw. ein solches anzeigen können, muss der Analytiker Kenntnis von der (möglichen) Existenz dieses Konstrukts haben. Demnach erfolgt die Auswertung von Daten stets innerhalb eines häufig subjektiven, kognitiven Bezugsrahmens, der vor dem Hintergrund eines konkreten Untersuchungsziels aktiviert wird [Drei94, 133]. Die auf Vorwissen gestützte inhaltliche Interpretation dient der Ergründung der kontextuellen Bedeutung der in den Analyseergebnissen repräsentierten Aussagen sowie der resultierenden Implikationen (vgl. Abschnitt 2.1.2). An ihrem Ende steht die aus den Daten gewonnene Information [HoKl91, 325].

2.1.4 Zusammenfassung des Begriffsverständnisses

Aufbauend auf den vorstehenden Ausführungen wird für den weiteren Gang dieser Arbeit folgendes Verständnis des Datenanalysebegriffs zugrunde gelegt: ***Datenanalyse*** ist die zielorientierte Verarbeitung von Daten zur Gewinnung zweck- und adressatengerechter Information. Ihr Ziel besteht in der Beantwortung spezifischer, im Kontext einer gegebenen Problemsituation auftretender Fragen. Die Datenanalyse umfasst die Interpretation der Daten sowie jede die Interpretation unterstützende Transformation.

Datenanalyse ist eine Aufgabe, die unabhängig von den eingesetzten Verfahren aus Außensicht definiert ist.[11] Sie wird bestimmt durch die Analysedaten (Aufgabenobjekt), das Analyseziel, das einen Informationsbedarf auslösende Problem sowie die resultierenden Analyseergebnisse. Jede Datenanalyse besteht prinzipiell in einer Transformation, welche die Daten in eine Darstellung geringerer Komplexität überführt. Dabei werden irrelevante Sachverhalte ausgeblendet und

[11] Vgl. hierzu die Definition der Aufgabenstruktur bei [FeSi13, 98] sowie die Unterscheidung zwischen Anwendungs- und Verfahrensebene bei [Knob01, 68f.].

durch das Herausarbeiten bedeutsamer Aussagen neue Informationen generiert [Knob02, 341].

Werden mit *Datenanalyse i.e.S.* typischerweise komplexere Transformationen assoziiert, die etwa fortgeschrittene Methoden der Mathematik, Statistik oder Künstlichen Intelligenz erfordern, so schließt obige Definition auch einfachere Formen der Informationsversorgung wie etwa das Berichtswesen, die Benachrichtigung über Ereignisse sowie den Daten- oder Dokumentenabruf (Data Access [BaGü04, 65], Information Retrieval [BaRi11]) ein: Berichte werden durch Aggregation aus Basisdaten erzeugt; eine Benachrichtigung, z.B. über die Schwellenwertüberschreitung einer kritischen Variablen, ist das Ergebnis der Auswertung einer Bedingungsregel; und der Abruf eines Tupels im Rahmen einer Datenbankabfrage stellt eine Verarbeitung der Eingaberelation durch einen Filter dar. Sie werden als *Datenanalysen i.w.S.* in die Betrachtung bewusst einbezogen.

2.2 Ansätze und Ausprägungen der Datenanalyse

Die folgenden Abschnitte geben einen Überblick über Basisansätze (2.2.1) und wichtige Erscheinungsformen der Datenanalyse (2.2.2).

2.2.1 Basisansätze der Datenanalyse

Datenanalysen lassen sich entlang zweier orthogonaler Dimensionen beschreiben [Hand99, 2], einerseits nach dem Theoriebezug im Analyseziel (Abschnitt 2.2.1.1), andererseits nach der Reichweite der Analyseergebnisse (Abschnitt 2.2.1.2). Hieraus resultieren drei Ausrichtungen als Basisansätze der Datenanalyse (Abschnitt 2.2.1.3).

2.2.1.1 Theoriebezug im Analyseziel

Theoretische Konstrukte können sowohl Ausgangs- als auch Endpunkt der Datenanalyse sein (vgl. Abschnitt 2.1.3.5). Zum ersten Fall zählt einerseits die *konfirmatorische Datenanalyse*, deren Ziel die empirische Überprüfung einer existierenden Hypothese ist. Andererseits schließt er auch die Anwendung bestätigter Theorien auf Einzelfälle oder eine

Grundgesamtheit ein. Im zweiten Fall, der *explorativen Datenanalyse*, zielt die Untersuchung auf die Konstruktion empirisch begründeter Hypothesen auf Grundlage gegebener Daten [EnMT95, 1], [Drei94, 15]. Die konfirmatorische und explorative Analyse werden im Folgenden kurz charakterisiert.

Konfirmatorische Datenanalyse

Die konfirmatorische Analyse gilt als „klassischer" Ansatz [HeMi94, 320], [BeLi97, 64], bei dem eine konkrete Vorstellung davon existiert, welche Aussagen die Untersuchung liefern soll. Er umfasst statistische Untersuchungen, die einen postulierten Zusammenhang zwischen mehreren Variablen überprüfen (z.B. Regressions-/Korrelationsanalyse [Küpp99, 52]), ebenso wie einfache Anwendungen der Datenabfrage. Bei letzteren beschränkt sich die Analyse prinzipiell auf ein Abrufen und Zusammenfassen von Datenwerten nach definierten Kriterien [Pyle99, 28]. Es wird gewissermaßen nach Daten gesucht, die zu einem (etwa als SQL-Klausel) spezifizierten Muster passen. Der Suchraum ist dadurch erheblich eingeschränkt [DhSt97, 168].

Die Vorstellungen, Annahmen und Muster, welche die Untersuchung leiten, werden als (implizite) Hypothesen verstanden.[12] Allgemeines Analyseziel des konfirmatorischen Ansatzes ist demnach die Durchsuchung der Datenbestände nach Aussagen, die existierende Hypothesen stützen oder widerlegen. Diese „hypothesengetriebenen" Fragestellungen werden auch als ***Top-down-Probleme*** bezeichnet [BeLi97, 64], [Knob01, 68].

Explorative Datenanalyse

Die explorative Datenanalyse[13] geht nicht von einer gegebenen Theorie bzw. einem genau spezifizierten Analysemodell aus, sondern von der

---

[12] Beispielsweise liegt der Frage, wie viele Käufer eines Artikels A auch einen anderen Artikel B erworben haben, die Annahme eines Verbundkaufzusammenhangs zwischen diesen Artikeln zugrunde.

[13] Dieser Ansatz geht im Wesentlichen auf die Arbeiten von J.W. TUKEY [Tuke62], [Tuke77] und J.-P. BENZÉCRI [Benz73] zurück. Sie haben die datenzentrierte Betrach-

vorhandenen Datenbasis. Sie strebt nach der Entdeckung empirisch begründeter Aussagen, die nach statistischen Maßstäben signifikant sind und auf bisher unbekannte Konzepte oder neue Hypothesen hinweisen [EnMT95, 2f.]. Die Untersuchung soll durch Annahmen und subjektive Präferenzen des Analytikers weitgehend unbeeinflusst bleiben. Daher erfolgt z.B. keine vorherige Festlegung, welche Variablen einen Zusammenhang erklären, und es herrscht keine konkrete Vorstellung darüber, welche Aussagen das Ergebnis enthalten soll [Küpp99, 51]. Es wird nach Mustern gesucht, die vorliegende Daten beschreiben [DhSt97, 168]. Eine Begrenzung des Suchraums liegt prinzipiell nur in Bezug auf den Mustertyp (z.B. Beziehungen, Cluster, Abweichungen) vor [Knob01, 68f.].

Allgemeines Ziel des explorativen Ansatzes ist es, in den Daten verborgene Regelmäßigkeiten sowie Anomalien oder Abweichungen von Regelmäßigkeiten aufzudecken, um daraus Hypothesen abzuleiten [Dree01, 134], [HeMi94, 9]. Solch „datengetriebene“ Fragestellungen sind auch als ***Bottom-up-Probleme*** bekannt [BeLi97, 64].

2.2.1.2 Reichweite der Analyseergebnisse

Das Kriterium Reichweite beantwortet die Frage, ob die Ergebnisse einer Untersuchung nur auf die analysierten Daten zutreffen, oder ob sie auf nicht in den Analysedaten repräsentierte Fälle generalisierbar sein sollen [Benn94, 4f.]. Im ersten Fall spricht die Statistik vom *deskriptiven* Ansatz. Der zweite Fall wird als *schließende* (auch inferenzielle, induktive oder analytische) Statistik bezeichnet und beschäftigt sich mit empirisch begründeten Schlussfolgerungen [HeMi94, 2], [Hand99, 2].

Deskriptive Datenanalyse

Ziel der deskriptiven Analyse ist die Aufbereitung mitunter sehr umfangreichen Datenmaterials zur Ableitung verständlicher Muster, die sich ausschließlich auf die untersuchten Objekte beziehen [Benn94, 5],

tungsweise wiederbelebt, nachdem lange Zeit die am Wahrscheinlichkeitsmodell orientierte schließende Statistik dominierte [HeMi94, 2].

[HeMi94, 2], [EnMT95, 13]. Die Analyse richtet sich auf beobachtbare Eigenschaften der Untersuchungseinheiten [Drei94, 15].

Die Deskription lässt sich weiter in *Mustererkennung* und *Musterbeschreibung* (Beschreibung i.e.S.) differenzieren. Die rein extensionale Aufzählung erkannter Muster (z.B. einer Menge von Kundengruppen oder fragwürdiger Versicherungsfälle) ist von eingeschränktem Wert. Durch intensionale Charakterisierung anhand typischer Eigenschaften können die entdeckten Konzepte näher beschrieben werden. Im induktiven Maschinellen Lernen sind diese Vorgänge als *unüberwachtes Lernen* (*Lernen durch Beobachtung*) und *überwachtes Lernen* (*Lernen aus Beispielen*) bekannt [FrPM91, 15f.]. Die Erkennung der Muster erfolgt unüberwacht, d.h., ohne Vorgabe von Beispielfällen, aufgrund der statistischen Verteilung der Merkmalswerte. Die Beschreibung geschieht überwacht, indem etwa Gemeinsamkeiten oder Unterschiede der ein Muster unterstützenden Fälle (Beispiele) betrachtet werden, um hieraus eine Erklärung des Konzepts abzuleiten [BeLi97, 72, 80f.].

Schließende Datenanalyse

Soll eine Untersuchung Aussagen liefern, die über den Erfahrungsbereich der analysierten Daten hinausgehen, so ist eine inferenzielle Analyse angezeigt [Drei94, 15]. Dies ist immer dann der Fall, wenn die Untersuchung der interessierenden Objekte nicht oder nicht vollständig möglich ist (z.B. aus Kosten- oder Zeitgründen, oder wenn dies die Zerstörung des Untersuchungsobjekts impliziert) [EnMT95, 49]. Eine besondere Form der Inferenz stellt die *Prognose* dar, weshalb häufig auch zwischen Beschreibungs- und Vorhersageaufgaben der Datenanalyse unterschieden wird [BeLi97, 96f.]. Bei der Prognose werden auf Grundlage von Daten zur Vergangenheit oder Gegenwart die Werte ausgewählter Zielvariablen für die Zukunft bestimmt (zeitliche Inferenz) [FaPS96, 12]. Schließende Analysen erlauben die Voraussage (noch) nicht beobachtbarer Eigenschaften[14] und die Übertragung

[14] So wird beispielsweise bei Klassifizierungsaufgaben ein Modell (z.B. Entscheidungsbaum) erzeugt, welches zur Voraussage der Klassenzugehörigkeit neuer Fälle genutzt werden kann. Diese Zuordnung ist eine Prognose, die auf einer hypothetischen Zuordnung beruht. Ob sich z.B. ein in eine bestimmte Risikoklasse eingeordneter Kunde

statistischer Hypothesen auf die Grundgesamtheit, wenn nur eine Stichprobe analysiert wurde [EnMT95, 14].[15]

Schlussfolgerungen geschehen zwangsläufig auf Basis unsicherer Information. Der Einsatz formaler stochastischer Modelle gestattet jedoch eine relativ präzise Bestimmung der mit einer Inferenz verbundenen Ungewissheit (z.B. durch Konfidenzintervalle, Signifikanztests, etc.) [HeMi94, 2], [EnMT95, 49]. Schließende Analysen erfordern somit stets die Einschätzung der Generalisierbarkeit von Analyseergebnissen und der Zulässigkeit von Schlüssen [Benn94, 4f.].

2.2.1.3 Ausrichtungen der Datenanalyse

Entlang der beschriebenen Dimensionen lassen sich drei Basisansätze (***Ausrichtungen***) der Datenanalyse identifizieren, die in Abbildung 3 zusammengestellt sind (vgl. hierzu auch [Drei94, 16]). Mit Ausnahme der Kombination Explorative Analyse / Inferenz (nicht zulässiger Schluss aus unverifizierten Ergebnissen) existieren Ausprägungen für alle Kombinationen.

Die explorative Analyse ist naturgemäß deskriptiv. Ihr Ziel ist die weitgehend hypothesenfreie Beschreibung der Untersuchungsobjekte; Schlussfolgerungen auf andere Fälle sind nicht beabsichtigt. Die erkannten Phänomene sind hypothetischer Natur und können in die Konstruktion von Theorien einfließen (*Hypothesengenerierung*). Die Analyse auf Grundlage vorhandener Theorien oder Hypothesen kann sowohl deskriptiven als auch inferenziellen Zwecken dienen. Im Rahmen der Deskription interessiert insbesondere die Validität der über die Untersuchungseinheiten ermittelten Aussagen (*Hypothesenprüfung* durch konfirmatorische Analyse). Hierzu werden Fälle herangezogen,

tatsächlich gemäß dieser Klassifikation verhält, kann erst durch Beobachtung seines künftigen Verhaltens festgestellt werden [BeLi97, 53f.].

[15] Zuweilen werden Inferenzen nicht der Datenanalyse zugerechnet, sondern als Anwendung ihrer Ergebnisse interpretiert (vgl. u.a. [Ehre76, 110], [Knob01, 76],). Auch Knowledge Discovery in Databases (KDD) behandelt prinzipiell nur Deskriptionsaufgaben [FaPS96, 12]. In dieser Arbeit werden gemäß Abschnitt 2.1.4 sämtliche Transformationen, die geeignet sind Informationen aus Daten ableiten, als Datenanalysen verstanden, der schließende Ansatz also explizit einbezogen.

die nicht in ihre Herleitung eingeflossen sind; eine Generalisierung oder Anwendung der Aussagen auf diese Testfälle ist jedoch nicht das Ziel. Diese Ausweitung erfolgt im Rahmen der Inferenz, um auf Basis einer Stichprobe ermittelte Aussagen auf eine größere Population zu übertragen oder auf andere Fälle anzuwenden (*Generalisierung und Anwendung* durch schließende Analyse) [Dree01, 134f.].

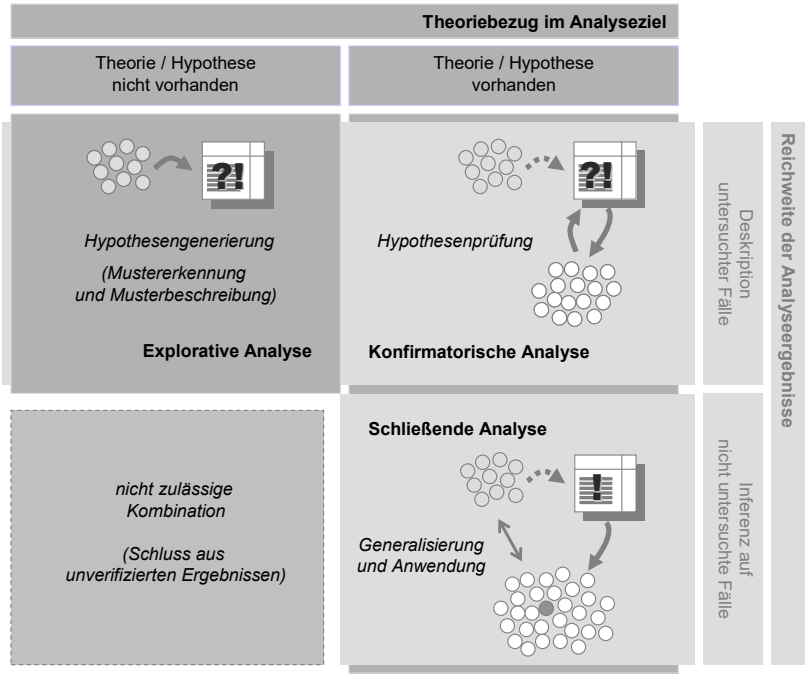


Abbildung 3: Dimensionen und Ausrichtungen der Datenanalyse (eigene Darstellung)

Die Einteilung ist grundsätzlich unabhängig von Methoden. Bestimmte Analyseverfahren können beispielsweise zur explorativen und konfirmatorischen Analyse gleichermaßen eingesetzt werden [HeMi94, VI]. Aus ihr lässt sich eine Reihung mehrerer unterschiedlicher Analysen zu einem idealtypischen Ablauf eines Datenanalyseprojekts ableiten:

1. **explorative Analyse** zur Hypothesengenerierung,
2. **konfirmatorische Analyse** zur Hypothesenprüfung, und
3. **schließende Analyse** zur Anwendung bestätigter Erkenntnisse auf neue Fälle.

2.2.2 Bedeutende Ausprägungen der Datenanalyse

Im Folgenden werden wichtige Erscheinungsformen der Datenanalyse erläutert, die für die betriebliche Informationsversorgung und Entscheidungsunterstützung aktuell von Relevanz sind. In der Literatur finden sich hierzu zum Teil wenig trennscharfe Klassifikationen, die z.B. die Komplexität der Benutzeroberfläche zugehöriger Werkzeuge, die Darstellungsform der Ergebnisse [BaGü04, 64f.] oder die Herkunft der eingesetzten Verfahren [HKMW01], [LiBe11] als Kriterien heranziehen. Daher wird im Folgenden der Versuch einer funktionsorientierten Klassifikation unternommen. Sie orientiert sich an den im Laufe der Zeit entstandenen Managementunterstützungssystemen (MUS) (vgl. [Knob02, 346-348], [GlGD08, 55f.]). Die Darstellung fokussiert weder auf die chronologische Entwicklung noch auf die Systemklassen an sich, sondern vielmehr auf die der jeweiligen Entwicklungsstufe eigenen, wesentlichen funktionalen Neuerungen bzw. Schwerpunkte.

2.2.2.1 Datenerhebung: Empirische Forschung

Die empirischen Wissenschaften, die sich der Beschreibung und Erklärung der Wirklichkeit anhand direkt beobachtbarer Eigenschaften interessierender Untersuchungsobjekte (empirischer Aussagen) widmen, stützen sich zur Erreichung ihrer Ziele auf die Auswertung von Daten. Diese Daten werden im Rahmen eines methodisch geleiteten Forschungsprozesses in der Regel speziell zur Beantwortung der aktuellen Fragestellung erhoben (Primärforschung) [Drei94, 1-3, 74]. Standen in der betrieblichen Datenanalyse lange Zeit bereits vorhandene Daten im Vordergrund (Sekundärforschung), so hat die *Datenerhebung* allein zu Auswertungszwecken im Zuge der Digitalisierung zuletzt wieder an Bedeutung gewonnen [Baro13, 50ff.]. Dabei ist die Primärforschung eine der ältesten Analysedisziplinen und zählt in

vielen Bereichen (z.B. der Marktforschung) zum Standardrepertoire der Informationsversorgung [BeEE09].

Gängige Erhebungsmethoden in der **empirischen Forschung** sind Befragungen, Beobachtungen und Experimente. *Befragungen* sind etwa in der Wahl- und Marketingforschung weit verbreitet, leiden jedoch u.a. an hohen Verweigerungsquoten. *Beobachtungen* sind unabhängig von der Auskunftsbereitschaft der Untersuchungsobjekte möglich und können auch ohne Kenntnis der Betroffenen erfolgen, wodurch die Daten deren natürliches Verhalten widerspiegeln. Zur Beobachtung zählen insbesondere auch alle Formen der automatischen Erhebung und Aufzeichnung[16] (z.B. Transaktions-, Sensordaten, Überwachungsvideos). Sie ist heute praktisch in allen Bereichen verbreitet. *Experimente* überwiegen in den Naturwissenschaften, gelangen aber zunehmend auch in der Sozial- und Marktforschung zum Einsatz. Sie haben den Vorzug, dass sich die Einflussgrößen überwiegend unter Kontrolle des Forschers befinden [Drei94, 3], [HeMi94, 26f]. Zur Datenauswertung werden typischerweise Verfahren der beschreibenden und schließenden Statistik genutzt [Drei94, 130ff.], [Diek07, 658ff.].

2.2.2.2 Datenversorgung: Standardberichtswesen

Unter der Bezeichnung Management Information Systems (MIS) wurden in den 1960-er Jahren die ersten Anwendungssysteme zur Unterstützung der Kontrollaufgabe des Managements vorgestellt. Ziel war die Versorgung von Führungskräften mit detaillierten oder verdichteten, aktuellen und korrekten Daten zur Beschreibung des Zustands der Geschäftsbereiche. Diese Daten wurden in Form periodischer, *standardisierter Berichte* angeliefert (Push-Prinzip). MIS arbeiteten datenorientiert und vergangenheitsbezogen. Die Berichte waren statisch, d.h., ihre Anpassung oder die Erstellung neuer Berichte zur Befriedigung situativer Informationsbedarfe sowie der Rückgriff auf Analysemethoden oder -modelle waren nicht vorgesehen. Vielmehr

[16] Vgl. hierzu etwa die Erfassung von Point-of-Sale-Daten aus Scannerkassen, die explizit als Verfahren der Kundenbeobachtung („Verkaufsdatenbeobachtung“) eingeordnet wird [Thei99, 360], [Fisc93, 38].

stand die *automatisierte Datenbereitstellung* unter Nutzung der damals neuen Möglichkeiten der Informationstechnik im Vordergrund [GlGD08, 55-62].

Ein automatisiertes **Standardberichtswesen** ist in fast jedem Unternehmen im Einsatz. Die Berichte sind meist auf einzelne Funktionsbereiche beschränkt und auf die Unterstützung des unteren und mittleren Managements bei operativen Kontroll- und Entscheidungsaufgaben ausgerichtet (z.B. Controlling- oder Vertriebsinformationssysteme). Moderne Reporting-Werkzeuge bieten Visualisierungsmöglichkeiten zur ansprechenden Präsentation von Kennzahlen und stellen zunehmend auch externe oder Echtzeitdaten bereit [GlGD08, 57-60], [NeKn15, 83f.].

2.2.2.3 Informationsversorgung: On-Line Analytical Processing (OLAP)

Stärker auf die Informationsbedürfnisse der Anwender ausgerichtet waren die in den 1980-er Jahren zunächst für das Top-Management entwickelten Executive Information Systems (EIS). Intuitiv bedienbare, flexible Werkzeuge sollten aktuelle und adressatengerecht aufbereitete Informationen liefern, um die Lage des Unternehmens umfassend darzustellen, ohne den Manager mit irrelevanten Daten zu überlasten. Wichtige funktionale Neuerungen waren *Benachrichtigungen* und *Navigationsmöglichkeiten*. Die selektive Benachrichtigung über relevante Ereignisse, wie etwa Schwellenwertüberschreitungen kritischer Kennzahlen oder das Vorliegen wichtiger Dokumente (z.B. Pressemeldungen), unterstützten die Überwachung komplexer Diskurswelten (Exception Reporting; passive *Informationsversorgung* gemäß Push-Prinzip [MeGr00, 2-5]). Zur Erkundung von Zusammenhängen oder Ursachen bestimmter Phänomene wurden Funktionen zur spontanen explorativen Navigation im Informationsraum bereitgestellt (gerichtete, aktive *Informationsbeschaffung* gemäß Pull-Prinzip) [GlGD08, 74-82].

Ein Großteil der geschilderten Funktionen wird heute meist als **On-Line Analytical Processing (OLAP)** auf Basis von Data Warehouses realisiert. Hauptmerkmal des OLAP, das als Konzept zur dynamischen, interaktiven Datenanalyse durch Entscheidungsträger entwickelt wurde [CoCS93], ist die Navigation im multidimensionalen Datenraum, der

durch Hypercubes aufgespannt wird. Hierbei werden quantitative Daten (Kennzahlen) durch verschiedene qualitative Daten (Dimensionen) beschrieben [BöUl00], [BaGü13, 119ff.]. Der Anwender kann Perspektiven und Aggregationsstufen ändern, multidimensionale Filterkriterien wählen und die angezeigten Informationen somit flexibel variieren (Ad-hoc-Reporting). Ergebnisse von OLAP-Anfragen werden als Kreuztabellen dargestellt und können mit arithmetischen Funktionen wie in Tabellenkalkulationsprogrammen ausgewertet und grafisch aufbereitet werden. OLAP gehört zum State-of-the-Art moderner Entscheidungsunterstützungssysteme und eignet sich sowohl zur Realisierung des Berichtswesens als auch zur explorativen Analyse. Varianten werden aktuell auch unter Bezeichnungen wie Data Discovery oder Visual Analytics vermarktet [NeKn15, 84].

2.2.2.4 Automatisierte Wissensentdeckung: Data Mining

Mit wachsenden Datenmengen in den Unternehmen stieg der Bedarf nach Funktionen zur automatischen Filterung und *Entdeckung* neuer Muster, wie sie seit den 1990-er Jahren mit Knowledge Discovery Systems bereitstehen [Knob02, 347f.]. Die weitgehend autonom arbeitenden Systeme verfolgten das Anliegen, die Exploration sehr großer, hochdimensionaler Datenbestände zu ermöglichen und auch Personen ohne langjährige datenanalytische Ausbildung zugänglich zu machen [Biss96, 5], [ElPr96, 92f.]. Die *Automatisierung* erlaubt den Einsatz überaus komplexer Berechnungskalküle und verschafft dem Analytiker mehr Raum für die Interpretation der Ergebnisse [HeMi94, V], [Hand99, 5], [FaUt02, 30].

Die automatisierte, explorative Datenanalyse im obigen Sinne ist unter dem Namen **Data Mining** bzw. **Knowledge Discovery in Databases (KDD)**[17] bekannt [Küst01, 124]. Nach anerkannter Definition ist KDD

[17] KDD und Data Mining verfolgen dieselbe Zielsetzung, haben aber eine unterschiedliche Reichweite [Knob01, 74]. Während KDD einen Wissensentdeckungsprozess beschreibt, stellt Data Mining lediglich eine Aktivität innerhalb dieses Prozesses dar, welche die eigentliche Datenanalyse umfasst. Vgl. hierzu ausführlich [Knob01, 86ff.]. Sofern eine explizite Unterscheidung zwischen Prozess und Aktivität nicht erforderlich ist, können die Begriffe synonym gebraucht werden.

ein Prozess zur nicht-trivialen Entdeckung gültiger, neuer, potenziell nützlicher und verständlicher Muster in Datenbeständen [FaPS96, 6]. Neuartigkeit, Nützlichkeit und Verständlichkeit rekurrieren auf die Absicht, Information zu erzeugen (vgl. Abschnitt 2.1.2.2). Gültigkeit stellt auf generalisierbare Aussagen ab und fordert eine vorläufige Verifikation, um die Analyseergebnisse als Wissen i.e.S. akzeptieren zu können (Wissensentdeckung). Kern der Definition ist die weitgehend hypothesenfreie Erkennung implizit in den Daten verborgener Muster und Auffälligkeiten. Als grundlegende Aufgaben des Data Mining gelten Abweichungsanalyse (Mustertypen: Änderungen und Abweichungen), Beziehungserkennung (Verknüpfungen, Abhängigkeiten, Sequenzen), Segmentierung (Cluster) und Modellgenerierung (Prognosemodelle) [Knob01, 77]. Seine Methoden stammen hauptsächlich aus der Statistik und dem Maschinellen Lernen [Hand99, 4].

2.2.2.5 Entscheidungsunterstützung: Statistik

Zur Fundierung von Planungs- und Entscheidungsprozessen reicht die Versorgung mit Daten, Informationen und Wissen oft nicht aus. Zur problemgerechten Aufbereitung und Auswertung verfügbarer Daten sind daher in den 1970-er Jahren interaktive *Entscheidungsunterstützungssysteme* (Decision Support Systems, DSS) entstanden, die *Modelle* und *Methoden* zur Lösung strukturierter oder semi-strukturierter Probleme bereitstellten. Ziel war insbesondere die effektive Unterstützung bei der Problemstrukturierung, Alternativensuche und -bewertung. Hierzu wurde ein formallogisches Vorgehen unterstellt, bei dem Probleme in explizite Modelle überführt und mithilfe zugehöriger Methoden gelöst werden. DSS waren auf konkrete Entscheidungssituationen auf operativer und taktischer Ebene zugeschnitten. Sie stützten sich neben der Statistik stark auf Heuristiken und Optimierungsverfahren des Operations Research (OR) [GlGD08, 62-74].

Anwendungsspezifische DSS haben in Stabs- und Fachabteilungen (z.B. zur Absatz- oder Produktionsplanung) sowie als individuelle Lösungen auf Grundlage von Tabellenkalkulationsprogrammen weite Verbreitung gefunden. Statistikwerkzeuge ermöglichen flexible modell- und methodengestützte Auswertungen. **Statistik** ist die Kunst der Erhebung und

Interpretation von Daten von der Planung bis zur Präsentation der Schlussfolgerungen. Sie kann als Basisdisziplin der Datenanalyse angesehen werden und gilt als deren „klassische“ Ausprägung. Die Statistik untersucht in erster Linie quantitative Daten mithilfe mathematisch und wahrscheinlichkeitstheoretisch fundierter, formaler Methoden [Voge91, 1], [Hube11, 1f.]. Im Mittelpunkt stehen Modellerstellung, -optimierung und -beurteilung, wobei die Überprüfung der Annahmen, die exakte Spezifikation der Unsicherheit sowie die Einschätzung der Stabilität der Modelle besondere Beachtung erfahren [ElPr96, 95, 109]. Zunehmende Bedeutung hat in jüngerer Zeit die BAYES-Statistik[18] erlangt, die als einfacher anzuwendende Verallgemeinerung der traditionellen Statistik verstanden werden kann. Sie erweitert den Begriff der Wahrscheinlichkeit zum Konzept der Plausibilität einer Aussage und ist damit in der Lage, auch Probleme der Schätzung unbekannter Parameter komplexer Systeme sowie die statistische Beurteilung der Ergebnisse mithilfe von Konfidenzregionen und Hypothesentests zu behandeln, die mittels traditioneller Statistik nicht lösbar sind [Koch00, 1f.].

2.2.2.6 Wirkungsanalyse: Prognose und Inferenz

Wenngleich Modelle und Methoden zur Beurteilung der künftigen Entwicklung von Sachverhalten sowie der Wirkung von Handlungsalternativen und Entscheidungen von der durch DSS charakterisierten Funktionsklasse bereits abgedeckt sind [GlGD08, 69f.], werden *zukunfts- und wirkungsorientierte Analysen* zunehmend als eigenständige Funktionsklasse betrachtet (Predictive Analytics [Cao16, 1:3]). Dies erscheint vor dem Hintergrund der als Basisansatz positionierten schließenden Analyse (vgl. Abschnitt 2.2.1.3) zulässig. Zudem betonen verschiedene Autoren, dass frühere MUS mehrheitlich vergangenheitsbezogene, statische Betrachtungen ermöglichten und sehen daher eine neue Systemgeneration begründet [GrGe00, 157-176], [GlGD08, 83]. Mit Ausnahme spezieller Induktionen, wie z.B. Schlüsse von der Stichprobe auf die Population, haben die meisten Inferenzen Vorhersagecharakter

[18] Sie beruht auf dem von THOMAS BAYES begründeten Theorem [Koch00, 14].

und sind in diese Funktionsklasse einzuordnen. Empirisch begründete Wirkungsanalysen und Prognosen werden mithilfe von Modellen, die durch Statistik oder Data Mining generiert wurden, mithilfe spezieller Simulationsmodelle oder mithilfe der Szenariotechnik durchgeführt [Knob02, 348].

2.2.2.7 Datentransformation und -speicherung: Data Science

Enorm angewachsene Datenmengen, die zunehmende Verfügbarkeit *unstrukturierter Daten* und die Anforderung, daraus möglichst schnell Informationen abzuleiten („Big Data" [KlTH13]), haben in den 2010-er Jahren zu neuen Herausforderungen bezüglich effektiver und effizienter Speicherung, Transformation und Bereitstellung von Daten geführt [NeKn15, 103-105], [Hand15, 710]. Hierfür wurden verschiedene Lösungen entwickelt, wie etwa die verteilte, parallele Verarbeitung zur Handhabung *großer Datenmengen*, spezielle Datenbankverwaltungssysteme zur Speicherung *unstrukturierter Daten* und zur Beschleunigung von Auswertungen (z.B. Hauptspeicher- und spaltenorientierte Systeme) [KlTH13, 321f.], [Krue+10, 145-148]. Von besonderer Bedeutung sind effiziente Verfahren zur Informationsextraktion, um strukturierte Merkmale und Muster aus unstrukturierten Dokumenten abzuleiten [LaRe08, 12f.].

Die analytische Datenverarbeitung in solchen Systemumgebungen wird als **Data Science** bezeichnet. Eine allgemein akzeptierte Definition ist nicht verfügbar.[19] Data Science wird als *interdisziplinärer Ansatz* zur Gewinnung von Wissen aus Daten verstanden [StSt14, 472], [CaFa16], der ähnlich wie KDD stark explorativ angelegt ist [Hand15, 709], [Dhar13, 67] und als Prozess neben der Analyse auch die Datenerfassung, -haltung, -transformation und Ergebnisverifikation umfasst. Dabei wird der *experimentelle Charakter* betont, wie er auch der

[19] Der Begriff wird NAUR zugeschrieben. Er versteht darunter "the science of dealing with data, once they have been established, while the relation of the data to what they represent is delegated to other fields and sciences" [Naur74, zitiert nach Cao16, 1:4]. Die Positionierung von Data Science im Wortsinn als wissenschaftliche Disziplin mit dem Untersuchungsgegenstand Daten wird von mehreren aktuellen Quellen aufgegriffen [Dhar13, 64], [Cao16, 1:3], [EnMD16, 1].

Explorativen Datenanalyse und dem Maschinellen Lernen eigen ist [Hand15, 708]. Hieraus wird wiederum die Notwendigkeit des Einsatzes einer breiten Palette unterschiedlicher Methoden begründet [EmMD16, 2]. Zugleich ist damit eine unstrukturierte, hoch *agile Vorgehensweise* verbunden. Gegenüber grafischen Werkzeugen werden Skriptsprachen bevorzugt, die schnelle Änderungen und hohe Flexibilität der entwickelten Prozesse garantieren sollen. Damit wird deutlich, dass Data Science kaum für Fachanwender geeignet ist, sondern zur Durchführung breit ausgebildete Informatik- und Statistikexperten erfordert [Baro13, 35-49], [Dhar13, 69].

Die Datengetriebenheit des Ansatzes geht soweit, dass zuweilen ohne konkreten Informationsbedarf allgemeines Domänenwissen generiert wird, das z.B. zur Übersetzung von Texten oder zur Spracherkennung universell einsetzbar ist (vgl. GOOGLE TRANSLATOR, IBM WATSON). Modellerstellung und -verifikation erfolgen hierbei vollautomatisch allein nach Maßgabe der Generalisierbarkeit der Modelle und unter Verzicht auf die Vorgabe theoretischer Konstrukte [Dhar13, 70-73]. Data Science betont die Rolle der Daten als Ressource, aus der auf vielfältige Weise ein Nutzen generiert werden kann. In diesem Sinne ist der Ansatz auch als Vorstufe für andere Analysefunktionen zu sehen, die stärker ergebnisorientiert angelegt sind.

2.2.2.8 Lösung von Anwendungsproblemen: Business Analytics

Ein *integrativer* Ansatz zur betrieblichen Datenanalyse hat sich seit den 2000-er Jahren herausgebildet. Aufbauend auf den im Zuge von Business Intelligence (BI) geschaffenen, integrierten Datenbeständen sollten verschiedene Analysefunktionen zusammengeführt werden, um situativ notwendige Informationen abrufen oder generieren zu können [Knob02, 348-351], [KeBa06, 7-16], [GlGD08, 111f.]. Neben den oben erläuterten generischen MUS-Funktionen sollten auch anwendungsorientierte (konzeptorientierte) Funktionen bereitgestellt werden, die spezielle betriebswirtschaftliche Verfahren (z.B. Balanced Scorecard, Planungsrechnung, etc.) zur Unterstützung bestimmter Managementaufgaben realisieren [KeBa06, 14]. Der Ansatz gilt als interdisziplinäre Weiterentwicklung von DSS bzw. MUS [BiHA16], [HoLP14, 130].

Die auf der Analyseschicht der BI-Architektur angesiedelten Funktionen werden unter dem Begriff **Business Analytics (BA)** zusammengefasst[20] [AyCG14, 194]. Eine breit akzeptierte Definition ist nicht bekannt; lexikalisch ist darunter die systematische, computergestützte Analyse von Daten im betrieblichen Kontext zu verstehen [Oxfo17]. Eine umfassende Literaturstudie [HoLP14] hat die mit dem Begriff assoziierten Bedeutungen untersucht und als allgemeines Ziel die evidenzbasierte Problemerkennung und -lösung in Geschäftssituationen herausgearbeitet [HoLP14, 134]. BA ist demnach *anwendungsorientiert* und mit spezifischen Varianten in vielen Domänen vertreten (z.B. Process / Financial / Customer Analytics). Weiter wird darin ein Paradigma (Philosophie) gesehen, das Evidenzen als primäre Richtschnur für Entscheidung und Problemlösung definiert. Zur Realisierung ist ein Transformationsprozess zur Informationserzeugung auszuführen, der die Erfassung, Aggregation, Analyse und Interpretation von Daten umfasst. Hierzu können eine Sammlung von Praktiken und Verfahren angegeben und nötige Kompetenzen definiert werden. Die *systematische, problemorientierte Vorgehensweise* wird auch von anderen Autoren betont [NeKn06] und zur „Planung, Steuerung, Durchführung und Kontrolle der Datengewinnung und analytischen Informationsnutzung“ konkretisiert [Wint16, 74]. Zwar erfolgt gelegentlich eine methodenzentrierte Einordnung [Gluc16], dennoch dominiert eine realisierungsneutrale Interpretation im Sinne eines auf spezifische Entscheidungen und Informationen ausgerichteten Analyseansatzes [BiHA16], [Wint16, 72].

2.2.2.9 Zusammenfassung

Abbildung 4 zeigt die resultierende Klassifikation der diskutierten Datenanalysefunktionen. Das *Standardberichtswesen* und die ereignisorientierte *Benachrichtigung* bilden die Funktionen des erweiterten *Berichtswesens*. Die Klasse der *Datenanalysen i.e.S.* umfasst *Navigation*, *Entdeckung, modell- und methodengestützte Analysen* sowie *Prognose und Inferenz*. Zusammen mit dem erweiterten Berichtswesen konstituiert sie

[20] Der Begriff wurde durch DAVENPORT [Dave06], [DaHa07] mit seiner Darstellung von Chancen und Notwendigkeit einer analytisch geprägten Unternehmensführung befördert.

die Gruppe der *Datenanalysen i.w.S.* Diese werden flankiert von weiteren, für eine nutzbringende Datenanalyse hilfreichen Funktionen. Dies ist zum Ersten die *Datenerhebung*, die empirische Daten zielgerichtet erfasst und dokumentiert. Zum Zweiten sind Funktionen zur *Speicherung und Transformation* nötig, die insbesondere die effiziente Transformation strukturierter und unstrukturierter Massendaten gewährleisten sollen. Sie können auch direkt zur Realisierung der Analysefunktionen beitragen. Die beiden linken Funktionsklassen schaffen gewissermaßen die Grundlage für effektive Datenanalysen. Zum Dritten kann die *evidenzbasierte Problemlösung* als Zwecksetzung (*Anwendung*) für Datenanalysen im betrieblichen Umfeld dienen (rechts).

Abbildung 4: Einfache Klassifikation wichtiger Datenanalysefunktionen (eigene Darstellung)

In späteren Kapiteln dieser Arbeit wird eine Methodik für die Datenanalyse zur evidenzbasierten Problemlösung (Business Analytics) entwickelt. Sie kann dazu auf alle Funktionen der Klasse der Datenanalysen i.w.S. zurückgreifen. Im Sinne der aufgabenorientierten Definition aus Abschnitt 2.1.4 sind die zugehörigen Verfahren und Techniken zunächst nicht eingeschränkt; vielmehr sollen jeweils die für den gegebenen Zweck am besten geeigneten Verfahren genutzt werden. Es ist zu erwarten, dass künftig neue, verfahrensorientiert motivierte Formen der Datenanalyse entstehen werden. Die Funktionsklassifikation kann helfen, diese nach ihrer Zielsetzung einzuordnen.

Anhang A1 vermittelt einen Überblick über gängige Datenanalysemethoden. Für nähere Ausführungen wird auf die dort angegebene Literatur verwiesen.

2.3 Konzeption von Datenanalyseprozessen

Datenanalyse wird typischerweise als Prozess verstanden. Der vorliegende Abschnitt untersucht, inwiefern diese Sichtweise begründet ist und welche Merkmale Datenanalyseprozesse auszeichnen. Hierzu wird zunächst der Prozessbegriff erörtert (Abschnitt 2.3.1), bevor gezeigt wird, wie dessen Bestimmungselemente auf die Datenanalyse zutreffen (Abschnitt 2.3.2).

2.3.1 Der Prozessbegriff

Der Ursprung des Wortes *Prozess* liegt im Lateinischen und bezeichnet einen Vorgang, aber auch ein Verfahren [Kien82, 364], [RiGr89, 1543]. Darüber hinaus steht der englische Begriff *process* auch für einen technischen Ablauf und eine zielgerichtete Aktivitätsfolge [KlRo88, 436f.], [Oxfo17b]. Die konkrete Bedeutung, die dem Begriff in verschiedenen Disziplinen beigemessen wird, variiert mitunter stark [BeSc95, 278]. Die Wirtschaftsinformatik untersucht insbesondere Geschäftsprozesse, worunter im einfachsten Fall ein ereignisgesteuerter Ablauf von Vorgängen verstanden wird. Im umfassenderen Fall werden zudem die Leistungserstellung und -übergabe sowie die Koordination der beteiligten Ressourcen berücksichtigt [FeSi01, 126].

Gemäß DIN dient ein Prozess der Verrichtung einer Aufgabe und besteht aus einer oder mehreren miteinander verbundenen Aktivitäten [DIN00, 7-3]. Diese transformieren Eingabeobjekte in Ausgabeobjekte [Kuts03, 14]. Ein Prozess beginnt mit einer definierten Startaktivität, die von auslösenden Ereignissen angestoßen wird. Mit Abschluss einer definierten Endaktivität gilt der Prozess als vollzogen, und die Prozessleistung liegt als messbarer Output vor.[21] Gleichzeitig werden

[21] Im Falle eines Prozesses mit nur einer Aktivität kann die Startaktivität zugleich als Endaktivität gekennzeichnet sein.

Nachereignisse erzeugt, die weitere Prozesse auslösen können. Zur Prozessdurchführung werden den einzelnen Aktivitäten Ressourcen zugeordnet [Jung02, 15f.].

Die skizzierten Merkmale können unter den vier Aspekten *Ziele*, *Transformation*, *Verkettung* und *Ressourcen* subsumiert werden, die eine vollständige Beschreibung von Prozessen erlauben [BeSc95, 280]. Die ersten beiden Aspekte repräsentieren die Außensicht, die letzten beiden die Innensicht der dem Prozess zugrunde liegenden Aufgabe [FeSi13, 98]. Diese vier Aspekte werden im Folgenden kurz erläutert.

2.3.1.1 Ziel- und Transformationsaspekt

Ziele sind wesentliche Bestimmungselemente der von einem Prozess zu erfüllenden Aufgabe [BeSc95, 279]. Sie werden in Sach- und Formalziele unterschieden [FeSi13, 72]. Formalziele nehmen auf technische und wirtschaftliche Eigenschaften der Prozessdurchführung und der Prozessleistung Bezug und äußern ich in Qualitäts-, Zeit- und Kostenkriterien. Sachziele beschreiben Art und Zweck der Leistungserstellung und richten sich demnach auf den vom Prozess zu erzeugenden Output [Jung02, 46], [Reif03, 23].

Dieser entsteht im Allgemeinen durch Transformation von Eingabeobjekten in Ausgabeobjekte [BeSc95, 278f.], [Gait83, 23]. Die Transformation wird im Sachziel beschrieben und nimmt somit stets auch Bezug auf den Prozessinput (Transformations- bzw. Aufgabenobjekte), dessen Zustände gemäß Outputspezifikation zu verändern sind [Geis97, 120]. Das zugehörige Input-Output-System wird dabei zunächst als Black-Box betrachtet [Gait83, 20]. Input und Output können materieller oder immaterieller Natur sein [Kuts03, 14]. So transformiert ein Fertigungsprozess Rohstoffe in Produkte (Güter), ein Auftragsbearbeitungsprozess Bestellungen in Fertigungsaufträge (Informationen).

2.3.1.2 Verkettungsaspekt (Prozessstruktur)

Aufgabenspezifikationen können intensional und extensional erfolgen [Berg81, 27ff.]. Während die intensionale Spezifikation das Sachziel als Sollzustand des Aufgabenobjekts beschreibt und keine Aussagen über

die Mittel zur Zielerreichung macht, stellt die extensionale Spezifikation den Ablauf der zur Zielerreichung auszuführenden Aktivitäten dar.[22] Sie weist gewissermaßen den Weg zum gesetzten Ziel, weshalb ein Prozess auch als Lösungsverfahren zur Erfüllung einer Aufgabe angesehen werden kann [Gait83, 55f.], indem er die sachlogische, zeitliche Abfolge von Vorgängen beschreibt. Aus dieser Verkettung resultiert ein Vorgangsnetz [FeSi13, 49], in welchem der Output eines Vorgangs den Input für die nachfolgenden Vorgänge bildet (Erzeuger-Verbraucher-Beziehungen) [BeSc95, 279], [Reif03, 18]. Beispielsweise stellt der Produktionsauftrag als Output der Auftragsbearbeitung den Input für die Arbeitsvorbereitung dar. Die Koordination der Vorgänge erfolgt über Ereignisse (vgl. Abschnitt 2.3.1) [Geis97, 120].

Ein Prozess weist demnach eine innere *Prozessstruktur* auf,[23] die gegebenenfalls durch mehrstufige Zerlegung in hierarchisch gegliederte Teilprozesse aufgedeckt wird [BeSc95, 279f.]. Die Prozessgliederung korrespondiert stets mit einer Aufgabengliederung und der zugehörigen Zielhierarchie, d.h., Teilprozesse erfüllen jeweils Teilaufgaben des Gesamtprozesses und verfolgen entsprechende Teilziele [Gait83, 2, 6]. Teilprozesse können sequenziell, simultan oder überlappt ablaufen [Kuts03, 14f.]. Die Abgrenzung gegenüber vor- und nachgelagerten Prozessen wird als *horizontale Auflösung*, der Aggregationsgrad (Abgrenzung von über- und untergeordneten Prozessen) als *vertikale Auflösung* bezeichnet [Gait83, 79f.]. Ein nicht weiter zerlegter Teilprozess heißt ***Aktivität* (*Schritt*)** [BeSc95, 282], [Jung02, 14].

2.3.1.3 Ressourcenaspekt

Jede Prozessaktivität benötigt zu ihrer Durchführung gewisse Ressourcen [LoDi04, 7]. Neben allgemeinen Betriebsmitteln wie Raum- und Zeitkapazitäten zählen hierzu insbesondere die *Aufgabenträger* als aktive

---

[22] Ein Prozess ist aus dieser Sicht ein Vorgang, für den eine Ablaufbeschreibung existiert [JaBS97, 24].

[23] Manche Autoren, wie z.B. [Dave93, 5], grenzen den Prozessbegriff auf klar strukturierte Vorgangsketten ein. Diese Sichtweise wird an anderer Stelle kritisiert, da z.B. innovative Prozesse damit nicht abbildbar sind [PiFr95, 14].

Ressourcen. Im Allgemeinen sind in den Vollzug von Prozessen mehrere unterschiedliche (maschinelle oder personelle) Aufgabenträger involviert [Reif03, 19]. Sie bearbeiten passive Ressourcen, die als *Leistungpakete* bezeichnet werden [FeHa94, 7]. Organisation und Zuordnung der Ressourcen zu einzelnen Aktivitäten orientieren sich an den Sach- und Formalzielen des Prozesses [Gait83, 16].

2.3.1.4 Zusammenfassung des Begriffsverständnisses

Gemäß vorstehender Ausführungen wird für diese Arbeit folgendes Prozessverständnis zugrunde gelegt: Ein ***Prozess*** dient der Realisierung definierter Ziele (Zielaspekt), die eine Umwandlung von Inputs in Outputs vorsehen (Transformationsaspekt). Er besteht aus mehreren, miteinander verknüpften Aktivitäten (Verkettungsaspekt), die von geeigneten Aufgabenträgern ausgeführt werden und Leistungspakete bearbeiten (Ressourcenaspekt) [BeSc95, 280], [Reif03, 20]. Eine vollständige Prozessbeschreibung umfasst drei Ebenen, die (1) Ziele (Ziel-/Transformationsaspekt), (2) Aufgaben (Verkettung) und (3) Ressourcen berücksichtigen.

Vor dem Hintergrund der lexikalischen Analyse (Abschnitt 2.3.1) ist diese Definition sowohl im Sinne eines Vorgangs (Durchführung einer Transformationsaufgabe) als auch im Sinne eines hierfür geeigneten Verfahrens (Aktivitätenfolge) interpretierbar. Zur Präzisierung wird der Begriff ***Vorgang*** im Folgenden ausschließlich für die Durchführung einer Aktivität bzw. Aufgabe gebraucht [FeSi13, 99], die Durchführung eines Prozesses wird als ***Ablauf*** bezeichnet (Instanzebene). Der Begriff *Prozess* steht grundsätzlich für eine Ablaufbeschreibung auf Typebene.

Eng mit dem Prozessbegriff verknüpft ist der Begriff ***Workflow***. Ein Workflow beschreibt, wie mehrere Aufgabenträger zur Erfüllung einer gemeinsamen Aufgabe kooperieren, indem sie zeitlich und kausal verknüpfte Einzeltätigkeiten ausführen. Er repräsentiert demnach wiederum einen Prozess im Sinne eines Lösungsverfahrens [JaBS97, 17f.], [WfMC99, 8], [PüSi10, 254]. Von einem allgemeinen Prozess unterscheidet er sich im Wesentlichen durch (1) den Fokus auf informationsverarbeitende Aufgaben, (2) die Arbeitsteilung, (3) seine Teilautomatisierung mit häufig hohem Anteil personeller Arbeit, (4) einen

hohen bis mittleren Strukturierungsgrad sowie die (5) umfassende technische Unterstützung und Steuerung, z.B. durch ein Workflow-Management-System [Sinz94, 220], [JaBS97, 490], [Reif03, 2, 69-71].

2.3.2 Datenanalyse als Prozess

In Literatur und Praxis herrscht Übereinkunft darüber, dass Datenanalyse als Prozess zu verstehen ist (vgl. z.B. [AdZa96, 37], [BeLi97, 63], [Vell97, 320f.], [Hand99, 3], [DeHa01, 57]). Prozessmodelle für Analysevorhaben sind u.a. in der empirischen Sozialforschung, der Statistik und der Wissensentdeckung (KDD) verbreitet. Unter besonderer Berücksichtigung der teilautomatisierten und arbeitsteiligen Analysedurchführung bei Nutzung von Analyse-Workbenches wird im KDD häufig auch der Workflow-Begriff gebraucht [KCC+02], [BrJT08], [ZPZL09], [KSBF09], [RüWB10], [Hila+11]. Die einzelnen Aspekte des Prozessbegriffs erfahren in der Literatur unterschiedlich starkes Gewicht. Die folgenden Abschnitte gehen auf die in Abschnitt 2.3.1 identifizierten Prozessmerkmale ein.

2.3.2.1 Datenanalyse als zielgerichtete Datenverarbeitung

Die Datenanalyse wird in Abschnitt 2.1.4 definiert als zielgerichtete Verarbeitung von Daten zur Gewinnung zweckgerechter Information, um Antworten auf bestimmte Fragen im Kontext einer Problemsituation zu liefern. Diese Fragen repräsentieren einen Informationsbedarf als gewünschtes Ergebnis der Datenanalyse [AdZa96, 81] und werden im Folgenden als *Analyseziele* bezeichnet [Säub00, 11], [Knob01, 105]. Die Analyseergebnisse bilden die Leistung (Sachziel) des Analyseprozesses und sollen zur Lösung des vorliegenden *Sachproblems* beitragen [HeMi94, 19]. Formalziele für Datenanalyseaufgaben ergeben sich aus der allgemeinen Forderung nach effizienter Informationserzeugung [FeSi13, 62] sowie aus Qualitätsanforderungen, die problemabhängig an die Analyseergebnisse zu stellen sind. Qualitätskriterien richten sich z.B. auf Vollständigkeit, Gültigkeit und Bestimmtheit der Information und können in geeigneter Form konkretisiert werden (vgl. [Bert75, 43]). Einen Kriterienkatalog enthält Anhang A5.1.

2.3.2.2 Datenanalyse als Transformationsaufgabe

Das allgemeine Ziel der Gewinnung von Information aus Daten in der oben zitierten Definition impliziert eine von der Datenanalyse zu leistende Transformation von Analysedaten (Input) in Analyseergebnisse (Output). Die Analysedaten (Transformationsobjekte) sind gemäß dem Analyseziel, das die als Ergebnis erwarteten „Antworttypen" festlegt, näher zu bestimmen [AdZa96, 82f.], [BeLi97, 95f.], [Knob01, 105]. Datenanalyse ist demnach eine Transformationsaufgabe.[24] Transformation gilt als eine grundlegende Operation der Daten- bzw. Informationsverarbeitung und kann sich sowohl auf den Inhalt als auch auf die Repräsentation der Daten richten [Bert75, 15], [PiRe91, 257f.]. PICOT & REICHWALD unterscheiden inhaltliche Transformationen niedriger und höherer Ordnung. Die erste Klasse bilden Umformungen wie etwa Verdichtung (z.B. Summierung) oder Spezifizierung (z.B. Aufschlüsseln von Kostenabweichungen nach Bezugsgrößen), in die zweite Klasse fallen Urteilen (Subsumieren von Tatbeständen unter Begriffe) und Schließen (Ableiten oder Verwerfen von Urteilen aufgrund anderer Urteile). Die Transformation i.e.S. (Translation, Kodierung) operiert rein syntaktisch und konvertiert Daten zwischen verschiedenen logischen oder physischen Strukturen nach Maßgabe definierter Regeln [Devl97, 205], [HeMi94, 157], [Müll00, 167].

Zur Generierung von Informationen während der Datenanalyse sind alle Transformationen geeignet, welche die in Abschnitt 2.2.2 genannten Funktionen realisieren. Der eigentlichen Analyse vorgelagert sind häufig Operationen, die der Vorbereitung der Daten für die Untersuchung dienen. Sie lassen sich im Wesentlichen auf die Grundfunktionen Auswahl, Separation und Verknüpfung, Normalisierung und Denormalisierung, Aggregation, Umwandlung sowie Ergänzung zurückführen [Devl97, 206ff, 256ff.], [Müll00, 196f.]. Darüber hinaus bedürfen die Analyseergebnisse häufig einer Aufbereitung in Form nachgelagerter

---

[24] Prinzipiell ist jede Aufgabe auf eine Transformationsaufgabe rückführbar [FeSi13, 37-39]. In der Grundform wird der Output ausschließlich aus den eingehenden Inputs abgeleitet (Input-Output-System). Weitere Formen berücksichtigen zusätzlich gespeicherte Informationen oder Zielgrößen. Der letzte Fall (Entscheidungsaufgabe) ist in der Datenanalyse im Rahmen der Interpretation von Bedeutung.

Transformationen, um eine andere Art der Informationsübertragung zu realisieren, die für den Empfänger besser interpretierbar ist [HeMi94, 158]. Hierzu gehören z.B. die Wiederherstellung ursprünglicher Datenrepräsentationen oder die grafische Aufbereitung. Zusammenfassend schließt eine Datenanalyse jedwede Transformation ein, die das Ziel der Ableitung von Informationen aus Daten unterstützt.

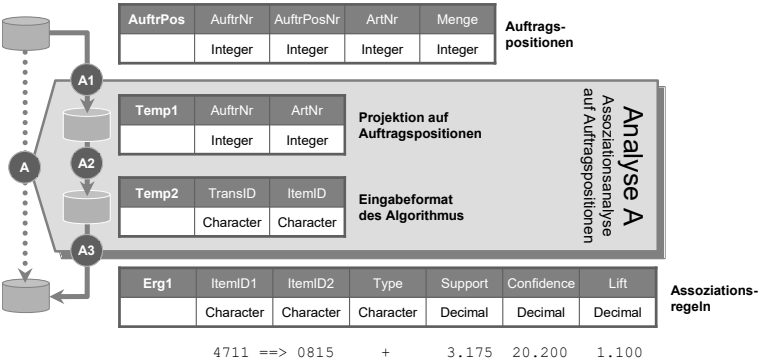


Abbildung 5: Datenanalyse als Datentransformation am Beispiel einer Assoziationsanalyse (eigene Darstellung)

Abbildung 5 zeigt ein einfaches Beispiel einer Datenanalyse als Datentransformation. Aus in relationaler Form vorliegenden Auftragspositionen von Kunden eines Handelsunternehmens (Input) sollen durch Transformation (A) Assoziationsregeln erzeugt werden, die das Verbundkaufverhalten wiedergeben (Output). Diese Regeln zeigen die relative bzw. bedingte Häufigkeit (Kennzahlen „Support“ bzw. „Confidence“) der gemeinsamen Bestellung mehrerer Artikel (Items) in der Form „Wenn Artikel 4711 bestellt wird, dann wird auch Artikel 0815 bestellt“, zusammen mit verschiedenen Gütemaßen („Type“ und „Lift“) an. Die zunächst als Black-Box erscheinende Analyse A erweist sich bei genauerer Betrachtung als Folge mehrerer aufeinander folgender Datentransformationen, die aus den Ausgangsdaten zunächst relevante Attribute selektieren (A1), die Repräsentation dieser Attribute verändern (jeweils vom Datentyp Integer in Character) (A2) und schließlich Analyseergebnisse der gewünschten Form erzeugen (A3). Diese sind anschließend vom Analytiker zu bewerten und zu interpretieren.

2.3.2.3 Datenanalyse als Verkettung mehrerer Schritte

Aus dem Beispiel wird bereits deutlich, dass sich Datenanalyse als mehrstufiger Prozess beschreiben lässt [Vell97, 321]. Die Mehrstufigkeit resultiert aus der Tatsache, dass zu ihrer Realisierung in der Regel verschiedene Transformations- und Interpretationsschritte erforderlich sind [WSG+97, 247]. Weitere Schritte können hinzukommen, wenn mehrere Einzelanalysen gekoppelt werden, um komplexere Fragestellungen zu beantworten. Im Folgenden werden die drei wesentlichen Aspekte erläutert, die eine Verkettung motivieren.

Erweiterter Transformationsbedarf

Effektive Datenanalyse erfordert neben der Anwendung einschlägiger Analysemethoden weitere Transformationen im Rahmen eines umfassenden Analyseprozesses, der jeweils nach Maßgabe des gesetzten Analyseziels und der vorliegenden Daten zu gestalten ist. Hierzu zählen sämtliche der eigentlichen Analyse vorgelagerten Aufgaben, die der Datenselektion, der Fehlerbereinigung und der Translation der Datendarstellung dienen (vgl. die Transformationsfunktionen in Abschnitt 2.3.2.2). Um Aussagegehalt und Relevanz der Analyseergebnisse sicherzustellen, sind diese einer gründlichen Interpretation und Bewertung sowie gegebenenfalls weiterer Transformationen zu unterziehen [FaPS96, 9]. All diese den eigentlichen Analyseschritt säumenden Aufgaben sollen gewährleisten, dass nach Abschluss des Prozesses nützliches Wissen bzw. Information vorliegt [Drei94, 17], [FaPS96b, 82], [BlGG00, 2].

Ein Datenanalyseprozess vereint und strukturiert diese Verarbeitungsschritte und lässt sich auf höchster Ebene in die drei trivialen Phasen (1) *Vorbereitung der Analysedaten*, (2) *Durchführung der Analyse* und (3) *Aufbereitung der Analyseergebnisse* gliedern [Bigu96, 10f.]. Ein auf dieser Ebene beschriebener Prozess konstituiert ein generisches Lösungsverfahren zur Informationsversorgung durch Datenanalyse [KnWe00, 349]. Die eigentlichen Analysemethoden stellen aus dieser Sicht notwendige, jedoch nicht hinreichende Teillösungsverfahren dar [Knob01, 86f.].

Kopplung komplementärer Analyseansätze

In Abschnitt 2.2.1.3 wurde ein idealtypischer Ablauf empirischer Untersuchungen skizziert, der von der Theoriegenerierung (explorative Analyse) über die Theorieprüfung (konfirmatorische Analyse) zur Theorieanwendung auf neue Fälle (schließende Analyse) führt. Seine vollständige Realisierung erlaubt die Ableitung und Nutzung bestätigten Wissens gemäß der empirischen Forschungsmethodik [Ehre76, 194]. Abbildung 6 illustriert, wie sich Theorien im Rahmen eines „empirischen Zyklus" entwickeln und verfeinern lassen [Drei94, 99f.] [AdZa96, 14f.].

Empirische Theorien sind konsistente Aussagensysteme, die der Erklärung von Phänomenen innerhalb eines Gegenstandsbereichs dienen. Zur Überprüfung solcher Theorien werden zunächst hypothetische konzeptuelle Aussagen in empirisch überprüfbare Aussagen operationalisiert und durch Analyse geeigneter Daten verifiziert.[25] Stützen die Analyseergebnisse die Hypothesen, ist die Theorie als vorläufig bewährt zu kennzeichnen [Drei94, 99]. Andernfalls kann die ursprüngliche Theorie entweder verworfen oder im Lichte der Analyseergebnisse modifiziert werden. Die empirisch hergeleiteten Aussagen implizieren konzeptuelle Aussagen, die als neue Hypothesen akzeptiert werden können (Theoriegenerierung). Die entstehende neue Theorie lässt sich wiederum einer Prüfung unterwerfen.

Zwar mag in vielen Fällen ausreichend Domänenwissen vorliegen, auf dessen Grundlage sich eine Theorie formulieren lässt [DeHa01, 59]. Trifft dies aber nicht zu, kann eine explorative Analyse zur Hypothesengenerierung beitragen. Derart gewonnene Hypothesen sind stets zu verifizieren [Vell97, 329f.]. Der kombinierte Einsatz beider komplementärer Analyseansätze birgt demnach großes Potenzial [Tuke62, 62], [SiLK94, 37]. „Keiner dieser beiden Ansätze sollte dabei für sich alleine in der Forschung verwendet werden, wir benötigen für eine angemessene Datenanalyse immer beide, konfirmatorische *und* explorative

[25] Die *Verifikation* dient der Feststellung des Wahrheitswertes einer Hypothese durch Suche nach Beweisen bzw. Gegenbeweisen in geeigneten Daten. Ihr Resultat ist die *Validität* der Aussage [KlZy96, 588].

Datenanalyse“ [Drei94, 300]. Ihre systematische Kopplung führt zu einem *Datenanalysezyklus*, der jeweils eine Hypothesen erzeugende (*bottom up*) und eine Hypothesen verifizierende Analyse (*top down*) im Wechsel ausführt [Knob01, 71].

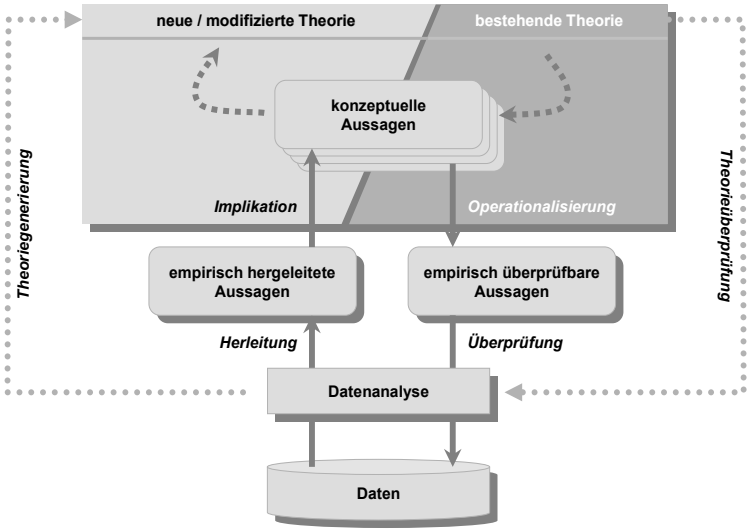


Abbildung 6: Zyklus der Theorieüberprüfung und Theoriegenerierung (eigene Darstellung, vgl. [Drei94, 137])

Da jede Analyse in der Regel neue Erkenntnisse und Fragestellungen aufwirft, entsteht vielfach weiterer Analysebedarf, woraus sich mehrere Iterationen des Analysezyklus ergeben können. Dem Zyklus wohnt das Grundprinzip inne, dass die Ergebnisse einer Analyse eine neue Untersuchung mit jeweils engerem Fokus und konkreteren Analysezielen motivieren [BeLi97, 64, 92]. Diese Kopplung von inhaltlich abhängigen Analysen führt zu Analyseprozessen höherer Ebene (Abbildung 7). Zur begrifflichen Klärung werden Prozesse, die durch Verkettung von Analysen entstehen, im Folgenden auch als ***Analyseketten***, die ihnen untergeordneten (hierarchisch weiter zerlegbaren) Teilprozesse als ***Analyseprozesse i.e.S.*** bezeichnet.

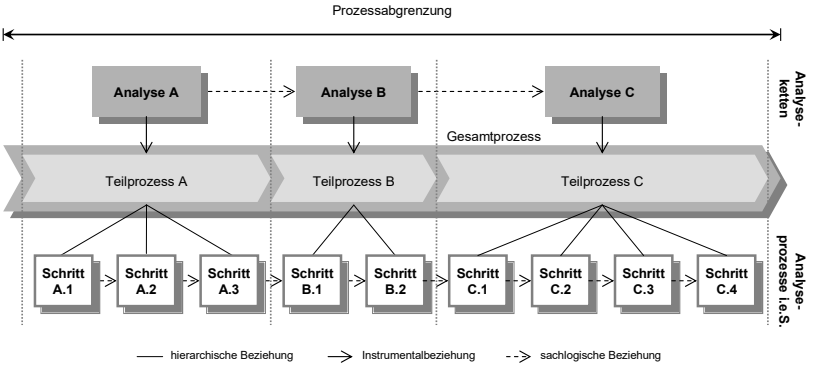


Abbildung 7: Mehrstufige Prozessstrukturierung durch Verkettung (eigene Darstellung)

Verkettung mehrerer Analysen

Die an einer Kette beteiligten Analysen müssen indes nicht zwingend streng alternierend explorative und konfirmatorische Fragen behandeln. Vielmehr entsteht durch Abstraktion vom starren Schema des Analysezyklus die Idee der beliebigen Verkettung von Analysen [BöKU03, 184f.]. Inhalt und Reihenfolge der Einzelanalysen innerhalb einer Kette ergeben sich aus dem zugrunde liegenden Sachproblem und den im Verlauf der Untersuchung erlangten Einsichten [NeK15, 190]. Der „Analyseweg“ ist durch ständige Perspektivenwechsel, variierende Untersuchungsobjekte und, dadurch bedingt, wechselnde Analysemethoden gekennzeichnet[26] [Biss01, 78], [Hand99, 3]. Die isolierte Untersuchung von Teilaspekten eines Sachproblems kann naturgemäß nicht zu dessen ganzheitlicher Ergründung oder Lösung führen. Daher ist es „oft die Kombination oder die Folge einer Reihe von Verfahren (...), die den Erfolg einer Analyse ausmachen“ [Boll96, 261]. Die problemspezifisch gewählte Aneinanderreihung unterschiedlicher Analysefunktionen erlaubt die Beantwortung sehr komplexer Fragestellungen [HaCC98, 2:3].

[26] Aus diesem Grunde wird für Analyseketten zuweilen die Metapher einer Autofahrt oder Ausflugsreise gebraucht, die situations- und interessenabhängig einen zumindest teilweise ungeplanten Verlauf nehmen kann [Biss01, 78], [NeKn15, 204-207]. „What is essentially described here is a voyage of discovery – and it is this sense of discovery that makes modern data analysis so exciting” [Hand99, 3].

Zur Illustration sei ein Beispielszenario aus dem Marketing angeführt [NeKn15, 190]: Zur Planung einer Werbemaßnahme werden zunächst durch Clusteranalyse (1) mögliche Zielgruppen ermittelt. Eine anschließende OLAP-Untersuchung (2) verschafft Einblick in Verhalten und Präferenzen der in einzelnen Segmenten enthaltenen Kunden. Die Berechnung des Kundenwerts (3) unterstützt die Interpretation der Segmente, von denen nun eines oder mehrere als Zielgruppe für das weitere Vorgehen ausgewählt werden. Mithilfe von Bestelldaten der selektierten Kunden kann sodann eine Verbundkaufanalyse (4) zur Identifizierung von Cross-Selling-Potenzialen erfolgen. Hierbei geeignet erscheinende Artikel werden in einer ABC-Analyse (5) anhand ihres Deckungsbeitrags geordnet und wieder durch OLAP (6) näher untersucht. Mit dieser Analysekette lassen sich geeignete Kundengruppen und Artikel für die Werbeaktion erkennen.

Allgemeine Grundprinzipien, die eine Analyseverkettung leiten können, sind unter anderem

- die **mehrstufige Spezialisierung von Analyseziel und -daten**, um Einzelaspekte eines Problems schrittweise näher zu ergründen [AmCo94b, 45], [Vell97, 321], [NeKn06, 103];
- die **Orientierung an Anwendungsprozessen**, aus denen sich Analysefragen ergeben. So enthalten z.B. die Phasen des Managementzyklus spezifische Situationsanalyse-, Prognose-, Entscheidungs-, Kontroll- und Abweichungsanalyseprobleme [Wild74, 37];
- die **Beschreibung, Erklärung und Begründung von Phänomenen**, indem erkannte Muster durch weitere Analysen beschrieben (vgl. extensionale und intensionale Charakterisierung, Abschnitt 2.2.1.2), durch Verknüpfung mit assoziierten Sachverhalten erklärt und kausal begründet werden [Ehre76, 425-427];
- die **mehrstufige Verbesserung der Modellgüte**, bei der Inferenzmodelle durch Aufnahme weiterer oder Fallenlassen bereits enthaltener Einflussgrößen kalibriert werden, um eine höhere

Approximations- bzw. Prognosegenauigkeit zu erreichen [ElPr96, 94f.], [Vell97, 321];

- **Vergleich und Kombination von Methoden und Modellen**, wobei Resultate verschiedener Analysen zur Auswahl des besten Ansatzes gegenübergestellt oder komplementäre Modelle zur Realisierung einer höheren Ergebnisqualität verknüpft werden [BAB+01, 90ff., 104f.];
- der **experimentelle Analyseansatz**, nach dem zahlreiche Analysen probeweise ausgeführt werden, bis letztlich ein akzeptables Ergebnis vorliegt. Er wird eingesetzt, wenn aufgrund unklar definierter Ziele oder komplexer Wechselwirkungen zwischen Fragen, Daten und Methoden kein strukturiertes Vorgehen möglich ist („cookbook fallacy") [Hand99, 3].

Zusammenfassend bleibt festzuhalten, dass Datenanalyseprozesse prinzipiell aus mehreren Teilprozessen auf verschiedenen Ebenen bestehen, deren Struktur sich häufig erst während der Ausführung durch Entscheidungen des Analytikers offenbart [FaPS96, 9].

2.3.2.4 Ressourcenaspekt

Zur Durchführung eines Analyseprozesses ist jeder Aktivität ein Aufgabenträger zuzuordnen, der ein geeignetes Lösungsverfahren realisiert. Während der Großteil der Transformationsaufgaben automatisierbar ist, sind bei Interpretationsaufgaben die menschliche Kognition und Urteilskraft meist unverzichtbar. Daher erweisen sich Datenanalyseprozesse in ihrer Gesamtheit grundsätzlich als teilautomatisiert. Sie werden von personellen Analytikern unter Einsatz mehr oder weniger spezialisierter Analyse- und Visualisierungswerkzeuge realisiert [BrAn96, 45], [Drei94, 33] (Workflow-Charakter der Datenanalyse, vgl. Abschnitt 2.3.1.4). Die Analysedaten tragen die Rolle der Leistungspakete, die von den Prozessaktivitäten verarbeitet und in den gewünschten Output transformiert werden.

2.3.2.5 Zusammenfassung: Datenanalyse als Prozess bzw. Workflow

Wie sich zeigt, hat die Datenanalyse gemäß den in Abschnitt 2.3.1 eingeführten Aspekten Ziele, Transformation, Verkettung und Ressourcen *Prozesscharakter*. Im Lichte der Kriterien aus Abschnitt 2.3.1.4 ist sie angesichts (1) ihrer Ausrichtung auf die Informationserzeugung, (2) der arbeitsteiligen, (3) typischerweise teilautomatisierten Durchführung, (4) eines aus ihrer häufig experimentellen Natur resultierenden, eher mittleren Strukturiertheitsgrads sowie (5) ihrer Abhängigkeit von softwaretechnischer Werkzeugunterstützung als *Workflow* einzuordnen. Die Prozessperspektive erlaubt die Anwendung der Prinzipien des Prozessmanagements.

2.4 Prozessmanagement in Datenanalyseprojekten

Bevor ein vorläufiger Managementansatz für Datenanalyseprozesse diskutiert wird (Abschnitt 2.4.4), ist in den folgenden Abschnitten zunächst die Frage nach dem Inhalt (2.4.1), den Zielen (2.4.2) und Aufgaben (2.4.3) des Prozessmanagements im Allgemeinen zu beantworten.

2.4.1 Der Prozessmanagementbegriff

Unter dem Begriff Prozessmanagement wird eine Vielzahl von Managementmodellen, Führungsprinzipien, Organisations- und Implementierungsmethoden diskutiert [Jung02, 14f.]. Allgemein wird darunter ein Gestaltungsparadigma zur Prozessorganisation[27] verstanden, das sich auf die Sicherung und Verbesserung der Prozessqualität richtet [Rohm98, 52], [Reif03, 31]. Die Reichweite des Begriffsverständnisses unterscheidet sich dabei in Abhängigkeit vom jeweils zugrunde gelegten Managementbegriff [Reif03, 37]. Im engeren Sinne wird darunter nur

[27] Die Prozessorganisation lässt sich auf die von NORDSIECK [Nord31], [Nord34] und HENNING [Henn34] eingeführte getrennte Betrachtung von Aufbau und Ablauf zurückführen, die von GAITANIDES [Gait83] weiter theoretisiert wurde. Wesentliche Impulse für den praktischen Einsatz sind MINTZBERG (Flusssysteme) [Mint79], PORTER (Wertketten) [Port85], DAVENPORT (Prozessinnovation) [Dave93] sowie HAMMER & CHAMPY (Business Reengineering) [HaCh94] zuzuschreiben [PiFr95, 16f.], [Rose96, 7f.].

die Gestaltung und (Fort-) Entwicklung von Prozessen verstanden [GSVR94], [Rohm98], [BeKR00], während im weiteren Sinne auch deren Lenkung eingeschlossen ist [Schw94], [ScNZ95], [ScSe04]. In der vorliegenden Arbeit wird die erweiterte Begriffsauffassung vertreten. Die Einbeziehung des Lenkungsaspekts reflektiert das Verständnis der Wirtschaftsinformatik [Sche98, 54] sowie die Definition der WORKFLOW MANAGEMENT COALITION (WFMC) [Geis97, 119].

2.4.2 Ziele und Instrumente des Prozessmanagements

Prozesse dienen der Erbringung einer Leistung mit definierten Eigenschaften (vgl. Abschnitt 2.3.1.1). Als Maxime des Prozessmanagements gilt die Zufriedenheit des Leistungsempfängers, die sich einstellt, wenn dessen Erwartungen mit den tatsächlichen Eigenschaften der gelieferten Leistung übereinstimmen [GaSV94, 13-15], [ScSe04, 36]. Die Eigenschaften lassen sich den Kategorien Qualität, Zeit und Kosten zuordnen. Unter Qualität wird im Allgemeinen die Eignung eines Gutes verstanden, gegebene Erfordernisse zu erfüllen [Gier00, 24]. Sie richtet sich insbesondere auf die Prozessleistung, die einen Anwendungsnutzen stiften soll. Zeitziele nehmen auf die Termintreue (Zeitpunkt der Leistungsbereitstellung) oder die Durchlaufzeit (Zeitdauer von Prozessauslösung bis Leistungsbereitstellung) Bezug. Prozesskosten quantifizieren den zur Erbringung der Prozessleistung aufgetretenen Ressourcenverbrauch [ScSe04, 203]. Um die Qualität der Prozesse auch langfristig zu gewährleisten, sind zusätzlich Flexibilitätsziele zu verfolgen. Ihr Inhalt ist die Anpassbarkeit der Prozesse an gewandelte Anforderungen und korrespondiert mit der Entwicklungsfähigkeit der Prozesse [Rohm98, 58f., 62]. Das ***Prozessmanagement*** dient zusammenfassend der zielorientierten Gestaltung, Lenkung und Entwicklung von Prozessen im Hinblick auf Qualität, Zeit und Kosten (Prozessparameter) zum Zwecke der Zufriedenstellung des Auftraggebers (Ergebnisparameter)[28] [GaSV94, 3].

[28] Die Ergebnisparameter bewerten die Prozessgüte gewissermaßen „mit der Stimme des Leistungsempfängers“, die Prozessparameter „mit der Stimme des Prozesses“ [Jung02, 92].

Das Prozessmanagement bewegt sich innerhalb dreier Spannungsfelder, die aus dem Zielsystem resultieren [GaSV94, 9]: Zum Ersten ist die Beziehung zwischen der Prozessleistung und den Erwartungen des Leistungsempfängers (Auftraggebers) zu behandeln. Jene richten sich bei der Datenanalyse in erster Linie darauf, dass die Analyseergebnisse Antworten auf die gestellten Fragen geben (Erreichung der Analyseziele) und betreffen damit die *Effektivität* des Prozesses. Zum Zweiten soll die Relation zwischen dem Ressourceneinsatz und dem Prozessergebnis möglichst optimal gestaltet, d.h., die *Effizienz* des Prozesses sichergestellt werden (Formalziele). Zum Dritten ist der Prozess mit gegebenenfalls veränderten Anforderungen in Einklang zu bringen, um *Flexibilität* zu erreichen. Qualitäts-, Zeit- und Kostenziele können gemäß ihrem Beitrag zur Steigerung von Effektivität und Effizienz eingeordnet werden [ScSe04, 173], woraus sich die in Abbildung 8 dargestellte Gliederung ergibt.

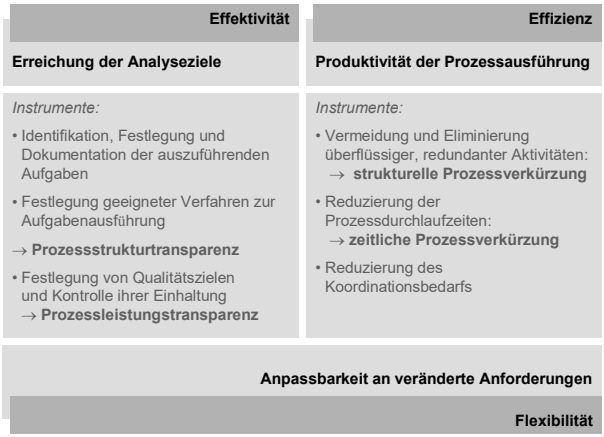


Abbildung 8: Zielkategorien des Prozessmanagements im Kontext der Datenanalyse (vgl. [Gait83, 6f.], [Jung02, 16])

Wesentliche **Instrumente** des Prozessmanagements zur Erreichung der genannten Ziele sind die Schaffung von *Prozessleistungstransparenz* und *Prozessstrukturtransparenz* [GaSV94, 15]. Unter Prozessleistungstransparenz wird die Messung der Prozess- und Ergebnisparameter zur Feststellung des Zielerreichungsgrades verstanden, unter Prozessstruk-

turtransparenz die Dokumentation der Prozessschritte einschließlich ihrer Verknüpfungen und Abhängigkeiten [ScVr94a, 25]. Sie können mithilfe eines Ansatzes zur Repräsentation (Modellierung) der Prozesse realisiert werden, der Struktur und Verhalten einzelner Prozessabläufe sowie der zugrunde liegenden Prozesspläne (Schemata) umfasst und auch die Prozessergebnisse einbezieht. Ein geeigneter Modellierungsansatz wird in Kapitel 4 entwickelt.

Vor ihrer konkreten Anwendung ist eine Operationalisierung der Ziele erforderlich [Rohm98, 59]. Hierbei sind die im Folgenden skizzierten Aspekte zu beachten.

Prozesseffektivität

Effektive Prozesse zeichnen sich durch richtige Aufgabenerfüllung nach Inhalt, Zeitbestimmung und Variabilität aus [Jung02, 16]. Voraussetzung der Prozesseffektivität ist die Prozessstrukturtransparenz, d.h., die Prozessaufgaben müssen identifiziert, spezifiziert und dokumentiert werden, und es sind Regeln und Verfahren zur Prozess- bzw. Aktivitätsdurchführung festzulegen. Ebenso ist Prozessleistungstransparenz zu schaffen, indem Qualitätsziele definiert, Termin- und Zeitpläne erstellt und deren Einhaltung kontrolliert werden [Gait83, 6f.]. Da die Ursachen nicht effektiver Prozesse oft in unklaren Vorgaben liegen, sind nur Prozesse mit definierter Struktur in der Lage, die gewünschten Ergebnisse zuverlässig zu produzieren [Gier00, 24f.].

Prozesseffizienz

Effiziente Prozesse sind durch minimalen Ressourceneinsatz und kostengünstige Aufgabenerfüllung charakterisiert [Jung02, 16]. Ineffizienzen lassen sich mittels Messung von Prozessparametern (Prozessleistungstransparenz) feststellen und beheben, etwa durch Eliminierung nicht erforderlicher Aktivitäten, Wahl des kürzesten Pfades (strukturelle bzw. zeitliche Prozessverkürzung) oder optimale Ressourcenauslastung [Gait83, 6f.]. Ebenso kann Prozessstrukturtransparenz die Effizienz befördern, da intransparente Abläufe oft redundante Vorgänge, erhöhten Koordinationsbedarf und längere Bearbei-

tungszeiten personeller Aufgabenträger zur Folge haben [GaSV94, 2], [Gier00, 21f.].

Prozessflexibilität

Flexible Prozesse besitzen die Fähigkeit zur Anpassung an veränderte Ziele und Einflussfaktoren [Jung02, 16]. Flexibilität nimmt in der Regel auf einzelne oder mehrere Effektivitäts- oder Effizienzaspekte Bezug. Prozessstruktur- und -leistungstransparenz ermöglichen die Erkennung und Verfolgung gewandelter Anforderungen im Hinblick auf Ergebnis- und Prozessparameter sowie deren Überführung in adäquate Umgestaltungsmaßnahmen.

2.4.3 Aufgaben des Prozessmanagements

Das Prozessmanagement umfasst die zielorientierte Prozessgestaltung, -lenkung und -entwicklung (vgl. Abschnitt 2.4.2). Diese Aufgaben werden im Folgenden kurz charakterisiert.

2.4.3.1 Prozessgestaltung

Die Gestaltung eines Prozesses umfasst die Abgrenzung von seiner Umwelt sowie die Festlegung von Verhalten und Struktur [FeMa95, 446]. Hierzu erfolgen die Vorgabe von Analyse- und Formalzielen sowie die Bestimmung der Analysedaten (Transformationsobjekt). Sodann ist eine Prozessstruktur zu konstruieren, die zur Realisierung des gewünschten Verhaltens in der Lage ist. Dieses kann durch Vorgabe von Regeln und Parametrisierung von Prozesskomponenten justiert werden. Sachliche Gestaltungsziele richten sich vornehmlich auf Ergebnisparameter. Als formale Gestaltungsziele sind neben Qualitäts-, Zeit- und Kostenkriterien insbesondere die Sicherstellung von Flexibilität und Lenkungsfähigkeit des Prozesses sowie die Beherrschung von Komplexität zu nennen [Reif03, 36f., 42]. Ergebnis der Gestaltung ist eine Ablaufspezifikation in Form eines Prozess- bzw. Workflow-Schemas [JaBS97, 491] und die Dokumentation der spezifizierten Ziele.

2.4.3.2 Prozesslenkung

Die Lenkung von Prozessen erfolgt in dem von der Gestaltung vorgegebenen Rahmen durch permanente Abfolge der Tätigkeiten Planung, Steuerung und Kontrolle [FeSi13, 3]. Sie soll die Leistungserstellung im Sinne der gesetzten Ziele durch Festlegung, Auslösung und Beurteilung konkreter Prozessaktivitäten sicherstellen [Reif03, 36] und ist im Wesentlichen auf die Prozessparameter gerichtet (Prozesscontrolling) [ScSe04, 173].

Die *Planung* konkretisiert vorgegebene Ziele und bestimmt Auswahl und Reihenfolge der auszuführenden Aktivitäten [Reif03, 39]. Hierbei kommt der Gestaltungscharakter der Planung zum Ausdruck (vgl. dazu näher Abschnitt 5.1). Die *Steuerung* dient der Allokation geeigneter Ressourcen sowie der Koordination des Zusammenwirkens der Aktivitäten. Hierzu leitet sie Anweisungen der Planung an die Aufgabenträger weiter [Gier00, 22], [Kuts03, 16], [ScSe04, 237]. Die *Kontrolle* umfasst zwei Aspekte: Unter dem Handlungsaspekt sichert sie die Zielerreichung durch Erkennung und Korrektur von Zielabweichungen (Leistungsmessung und -beurteilung durch Soll/Ist-Vergleiche, Analyse von Abweichungsursachen, Erarbeitung und Überwachung von Korrekturmaßnahmen), unter dem Lernaspekt strebt sie nach Steigerung von Effektivität und Effizienz durch bessere Beherrschung der Einflussfaktoren (Gewinnung von Erfahrungsdaten) [ScSe04, 215].

2.4.3.3 Prozessentwicklung

Gestaltung und Lenkung geschehen im Rahmen eines fortwährenden Entwicklungsprozesses, um Prozesse auch längerfristig in die Lage zu versetzen, die gestellten Anforderungen zu erfüllen [Reif03, 36]. Hierbei sind einerseits veränderte Leistungsanforderungen seitens des Auftraggebers (Erzeugung anderer Ergebnisse) [Jung02, 91], andererseits gewandelte Bedingungen bei gleichem Analyseziel (z.B. veränderte Daten, neue Verfahren, andere Infrastrukturkomponenten) zu behandeln. In beiden Fällen stehen Analyse- und Formalziele im Fokus der Betrachtung, weshalb Prozessentwicklung als zielgerichtete Umgestaltung eines Prozesses verstanden wird [Reif03, 42].

Die *Umgestaltung* kann einzelne Aktivitäten, Teil- oder Gesamtprozesse betreffen [Jung02, 98]. Grundsätzlich wird zwischen einer vollständigen Neugestaltung (Innovation) und kontinuierlich ablaufenden Verbesserungen unterschieden. Die Kombination beider Ansätze erscheint sinnvoll, da die (Neu-) Gestaltung „optimaler" Prozesse prinzipiell als unmöglich erachtet wird [GaSV94, 3, 11]. Anders als bei Geschäftsprozessen bildet in der Datenanalyse die kontinuierliche Verbesserung jedoch eher die Ausnahme, während die Innovation die Regel darstellt. Die systematische Wiederverwendung und Weiterentwicklung von Analyseprozessen ist in der Praxis wenig verbreitet; sie findet allenfalls vor dem Hintergrund des Erfahrungsschatzes des Analytikers implizit statt.

2.4.4 Ein Regelkreismodell des Datenanalyseprozessmanagements

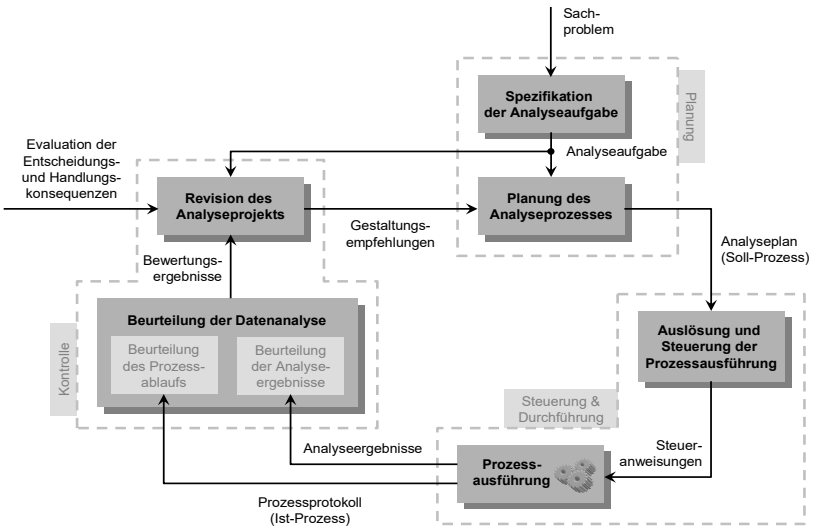


Abbildung 9: Regelkreismodell des Managements von Datenanalyseprozessen (eigene Darstellung)

Auf Grundlage der skizzierten Aufgaben lässt sich ein Regelkreismodell zum Management von Datenanalyseprozessen herleiten, das in Abbildung 9 dargestellt ist. Es repräsentiert ein Zwischenergebnis der

bisherigen Überlegungen und wird im weiteren Verlauf der Arbeit verfeinert. Das Modell orientiert sich an dem für die Prozesslenkungsaufgabe (Abschnitt 2.4.3.2) beschriebenen Zyklus aus Planung, Steuerung und Kontrolle, der als Regelkreis interpretierbar ist [FeSi13, 36]. Die verbleibenden Aufgaben Gestaltung und Entwicklung können schwerpunktmäßig der Phase Planung zugeordnet werden, da ihr die Entscheidung über die Festlegung der vom Prozess zu erreichenden Ziele und die dazu auszuführenden Aktivitäten (Gestaltung) obliegt. Die Entwicklung ist aufgrund ihrer Definition als Umgestaltung analog zu behandeln. Der Umgestaltungsbedarf wird im Rahmen der Kontrolle erhoben.

Ausgangspunkt ist die **Planung** eines Analyseprozesses nach Maßgabe eines vorliegenden *Sachproblems* sowie gegebenenfalls der *Gestaltungsempfehlungen* aus der Revision eines früheren Prozessablaufs. Hierzu wird aus dem Sachproblem eine *Analyseaufgabe* abgeleitet, welche auf die Befriedigung eines problemspezifischen Informationsbedarfs zielt. Diese Aufgabe soll mithilfe des zu planenden Analyseprozesses gelöst werden. Ergebnis der Planung ist ein ausführbarer Prozess im Sinne eines *Analyseplans*, der Aussagen über die Prozessstruktur, die Aufgabenträger und Methoden zur Ausführung der Prozessaktivitäten trifft (Soll-Prozess). Der Analyseplan wird zur Anwendung gebracht, indem die Prozessaktivitäten durch *Steueranweisungen* bei den spezifizierten Aufgabenträgern ausgelöst werden und deren Zusammenwirken gemäß den Planvorgaben gesteuert wird (**Steuerung und Durchführung**).

Während der Prozessdurchführung gemachte Erfahrungen werden im Rahmen der **Kontrolle** detailliert ausgewertet. Konkret werden hierbei zum einen die *Analyseergebnisse* hinsichtlich ihrer Eignung betrachtet, zur Lösung des Sachproblems beizutragen (Bewertung der Ergebnisparameter aus Sicht des Auftraggebers). Zum anderen wird ein über den Prozessablauf geführtes *Prozessprotokoll* (Ist-Prozess) ausgewertet, das Informationen über Schwachstellen des konzipierten Analyseplans beinhaltet und die Messung der Prozessparameter erlaubt. Gemeinsam dienen sie der Beurteilung der realisierten Datenanalyse. Die zugehörigen *Bewertungsergebnisse* gehen, zusammen mit gegebenenfalls vorliegenden Informationen über Nutzen und Kosten von Handlungs-

maßnahmen, die auf Grundlage der Analyseergebnisse ergriffen wurden (*Evaluation der Entscheidungs- und Handlungskonsequenzen*), in die Revision des gesamten Analyseprojekts ein. Im Rahmen einer ganzheitlichen Schwachstellenanalyse werden Verbesserungsvorschläge und *Gestaltungsempfehlungen* für künftige Prozesse entwickelt, um bei wiederholter Anwendung eine höhere Zielerreichung zu ermöglichen.

Die weitere Gliederung dieser Arbeit orientiert sich am Regelkreismodell: Zunächst betrachtet Kapitel 3 das Vorgehen während der Durchführung von Datenanalysen. Die Kapitel 5-7 beschreiben eine Methodik zur Planung (Kapitel 5), Steuerung (Kapitel 6) und Revision (Kontrolle, Kapitel 7) von Datenanalyseprozessen. Sie basiert auf einem umfassenden Modellierungsansatz, der in Kapitel 4 entwickelt wird. Eine Fallstudie zeigt in Kapitel 8 die Anwendung der Methodik.

3 Bestandsaufnahme und Empfehlungen zum Vorgehen bei der Datenanalyse

Der Ablauf von Datenanalyseprozessen weicht in der Regel stark von der idealtypischen Vorgehensweise ab. Dieses Kapitel untersucht die Ursachen dieser Diskrepanz und stellt Empfehlungen zur Unterstützung des Analytikers vor. Abschnitt 3.1 betrachtet Prozessmodelle und stellt sie den in der Praxis zu beobachtenden Abläufen gegenüber. Abschnitt 3.2 erläutert Möglichkeiten zum Umgang mit Komplexität bei der Analysedurchführung. Auf Grundlage der gewonnenen Erkenntnisse schlägt Abschnitt 3.3 ein Vorgehensmodell zur Datenanalyse vor.

3.1 Struktur und Ablauf von Datenanalyseprozessen

MOSTELLER & TUKEY betrachten Datenanalyse als Kunst und sehen hierin eine Ursache für Schwierigkeiten bei dem Versuch, das Vorgehen bei ihrer Durchführung strukturiert darzustellen [MoTu77, 1]. Im vorliegenden Abschnitt wird dieser Versuch unternommen. Hierzu untersucht Abschnitt 3.1.1 Möglichkeiten der Strukturierung von Datenanalyseprozessen und entwickelt ein generisches Prozessmodell. Eine kurze Beschreibung der darin enthaltenen Aufgaben erfolgt in Abschnitt 3.1.2. Das tatsächliche Vorgehen bei der Datenanalyse erläutert Abschnitt 3.1.3.

3.1.1 Prozessmodelle der Datenanalyse

Datenanalysen sind wissensintensive Vorgänge, die von menschlichen Analytikern konzipiert und häufig unter Einsatz mehrerer Software-Werkzeuge ausgeführt werden [BrAn96, 39f.]. Die spezifische Ausgestaltung konkreter Analyseabläufe im Hinblick auf Inhalt und Reihenfolge der involvierten Tätigkeiten variiert in Abhängigkeit von der verfolgten Zielsetzung und von der vorliegenden Systemumgebung. Die Vorgabe einer allgemeingültigen, idealen Struktur von Datenanalyseprozessen auf Detailebene scheint daher nicht realistisch [Drei94, 4]. Um dem Analytiker dennoch eine Orientierung für die Gestaltung effektiver und effizienter Prozesse zu geben, wurden verschiedene *Prozessmodelle* vorgeschlagen [BrAn96, 54f.], [WSG+97, 247]. Da solche

Modelle aus jeweils spezifischen Perspektiven von tatsächlichen Abläufen abstrahieren, resultieren recht unterschiedliche Prozessbeschreibungen. So werden etwa die Prozesse in verschiedene Teilschritte zerlegt, gleiche Aktivitäten unterschiedlich bezeichnet, verschiedene Detaillierungsgrade gewählt, eine anwender- oder technikbezogene Sicht eingenommen.

Bei genauerer Sichtung der Vorschläge lässt sich dennoch eine einheitliche Grundstruktur identifizieren [Säub00, 34]. Zu ihrer Ermittlung erscheint die detaillierte Beschreibung einzelner Modelle nur bedingt hilfreich. Vielmehr sind deren Kernaufgaben in den Vordergrund zu stellen, wozu eine Betrachtung des Datenanalyseprozesses auf höchster Ebene genügt [DeHa01, 63]. Einen interessanten Beitrag zu dieser Diskussion liefert SÄUBERLICH, der fünf ausgewählte Prozessmodelle für KDD auf einheitliche Strukturen untersucht und zu einem „Leitfaden für KDD-Anwendungen" generalisiert [Säub00, 22-39]. Dieser Ansatz wird im Folgenden aufgegriffen und vor dem Hintergrund des in Abschnitt 2.2.2 vertretenen Verständnisses um weitere Prozessmodelle aus anderen Disziplinen der Datenanalyse ergänzt.[29] Als Grundlage der Auswertung dienen die folgenden Vorschläge, für deren Berücksichtigung die jeweils angeführten Charakteristika maßgeblich sind.[30] Auf Detaildarstellungen der einzelnen Modelle wird verzichtet; diese sind in den angegebenen Quellen nachzulesen.

- Das Prozessmodell nach **ADRIAANS & ZANTINGE** [AdZa96, 37-78] repräsentiert ein einfaches, praxisorientiertes Schema, dessen Schwerpunkt auf der *Datenvorbereitung* liegt. Dem Vorschlag liegt ein Data-Mining-Verständnis zugrunde, das von Methoden des *Maschinellen Lernens* geprägt ist, die auch in der Data Science von großer Bedeutung sind.
- Phasenmodelle des *Data Warehousing* umfassen je eine Extraktions-, Transformations- und Ladephase zur Realisierung einer integrierten

[29] SÄUBERLICH untersucht die Modelle von BRACHMAN & ANAND, CHAPMAN ET AL., FAYYAD ET AL., JOHN sowie eine frühere Version des Vorschlags von WIRTH ET AL.

[30] Die Vorschläge sind in alphabetischer Reihenfolge der Originalquellen aufgeführt.

Analysedatenbasis (ETL-Prozess[31]), sowie eine Analysephase zu deren Nutzung. Im Modell von **BAUER & GÜNZEL** [BaGü13, 87-141] wird die Extraktion nach Veränderungen in den Quellsystemen durch eine Monitoring-Aktivität ausgelöst, eine eigene Interpretationsphase ist jedoch nicht vorgesehen.

- In der *Statistik* steht häufig die Prüfung von Hypothesen im Vordergrund, die zunächst zu formulieren sind. **BLEYMÜLLER ET AL.** beschreiben ein einfach gehaltenes Prozessmodell, das für jedwede statistische Untersuchung geeignet sein soll [BlGG00, 2]. Im Rahmen einer eigenen Planungsphase können Hypothesen postuliert und operationalisiert, in der abschließenden Interpretationsphase über deren Annahme oder Ablehnung entschieden werden.

- Wesentliches Merkmal des viel zitierten KDD-Modells nach **BRACHMAND & ANAND** [BrAn96] ist dessen *anwender- und aufgabenspezifische Perspektive*, die den menschlichen Analytiker in den Mittelpunkt rückt. Die Autoren verweisen auf Ergebnisse empirischer *Studien mit Anwendern*, unter deren Berücksichtigung sie ein Prozessmodell entwickeln konnten, das das Vorgehen in realen Projekten reflektiere [BrAn96, 40f, 55].

- Der Forschungsprozess nach **CAPLOW** [Capl71, 40] gilt in den Sozialwissenschaften als grundlegendes Modell für den Ablauf der *empirischen Forschung* [Drei94, 17]. Er geht über die reine Datenanalyse hinaus und bezieht insbesondere die Projektplanung und das Untersuchungsdesign in die Betrachtung ein.

- Der unter dem Namen CRISP-DM (Cross-Industry Standard Process for Data Mining) bekannte Vorschlag von **CHAPMAN ET AL.** [CCK+00] wurde von einem *Industriekonsortium* entwickelt und zielt auf einen branchen- und softwareunabhängigen *Standard für Data-Mining-Prozesse*. Es wird vom Werkzeug IBM SPSS MODELER unterstützt und gilt als das in der Praxis meistgenutzte Prozess-

[31] Die Abkürzung ETL steht für die Stufen Extraktion von Rohdaten aus den Datenquellen, Transformation (i.e.S.) und Laden der aufbereiteten Daten in eine gemeinsame Speicherstruktur [Müll00, 145f.].

modell für KDD. CRISP-DM wird zunehmend auch für andere Formen der Datenanalyse empfohlen und eingesetzt [PrFa13, Kap. 2], [KDnu14].[32]

- Das Prozessmodell von **EMC** [EMC15, 26-30] reflektiert die Sicht eines *Beratungshauses* und stellt die *Anwendung der gewonnenen Erkenntnisse* im Kontext von *Big Data Analytics* in den Mittelpunkt. Der Vorschlag integriert bewährte Praktiken mit interdisziplinären wissenschaftlichen Theorien.[33] Neben der Kommunikation mit dem Auftraggeber wird explizit der Rückgriff auf in der Organisation vorhandenes *Erfahrungswissen* betont.

- **FAYYAD ET AL.** [FaPS96] beschreiben in ihrem grundlegenden Aufsatz, der als einer der am häufigsten zitierten wissenschaftlichen Beiträge zum KDD angesehen werden kann, die Hauptaufgaben der *Wissensentdeckung* und ein zugehöriges einfaches Prozessmodell. Dieser Vorschlag erlangt weitere Bedeutung durch die Beobachtung, dass zahlreiche Werkzeughersteller bei Entwicklung ihrer proprietären Prozessmodelle intensiv auf die KDD-Literatur Bezug nehmen [Wild01, 14].

- Der Beitrag von **JOHN** [John97, 5-23] stellt die *Interaktion* der beiden Rollen des *Domänenexperten* und des Analytikers heraus, die bei komplexen Data-Mining-Vorhaben kooperieren. Von großer Bedeutung sind daher Problemspezifikation sowie Interpretation, während der im Falle unbefriedigender Ergebnisse über eine *Wiederholung des Prozesses* befunden wird.

---

[32] In einer nicht-repräsentativen Umfrage unter den Nutzern des Portals KDNUGGETS vom Oktober 2014 gaben 43% der 200 Teilnehmer an, CRISP-DM für Analytics, Data Mining oder Data Science zu verwenden. Der SEMMA-Prozess von SAS ist mit 8,5% das zweitbeliebteste der explizit genannten Schemata. 27,5% der Befragten gaben an, eine eigene Methodik einzusetzen [KDnu14].

[33] Den Autoren zufolge liegen dem Modell neben der Entscheidungstheorie, dem empirischen Zyklus (Abschnitt 2.3.2.3) und dem CRISP-DM ein Reifegradmodell (DELTA) nach DAVENPORT [DaHM10], der Ansatz der angewandten Informationsökonomie nach HUBBARD [Hubb10] und ein technisches Kompetenzmodell („MAD Skills“) nach COHEN [Cohe+09] zugrunde [EMC15, 28].

- Der so genannte SEMMA-Prozess (Akronym aus den Bezeichnungen der einzelnen Phasen) stellt den Beitrag der Firma **SAS INSTITUTE** [KuKi99] dar, die nicht nur ein gängiges KDD-Werkzeug anbietet, sondern insbesondere mit ihrer weit verbreiteten Standardsoftware für statistische Anwendungen über langjährige Expertise in der Datenanalyse verfügt. Dieses Prozessmodell repräsentiert die *Sichtweise eines Werkzeugherstellers* und reflektiert zugleich *praktische Erfahrungen* mit verschiedenen Ausprägungen der Datenanalyse.
- Den hinsichtlich der Anzahl der betrachteten Phasen umfassendsten Ansatz stellt das aufgabenorientierte Prozessmodell von **WIRTH ET AL.** [WSG+97, 245-247] dar. Es beschreibt den Lebenszyklus eines KDD-Projekts aus der Sicht eines *industriellen Anwenderunternehmens* in neun Stufen, die jeweils in mehrere Aufgaben zerlegt werden können. Neben der eigentlichen Analyse widmen die Autoren jeweils zwei Phasen der Problemspezifikation und der Anwendung des Wissens.

Als Kristallisationskerne für die Aktivitäten eines allgemeinen Prozessmodells dienen die fünf Phasen eines *Modells zur datenanalytisch gestützten Entscheidungsfindung* nach GAUL ET AL. [GaSc89], [GRSS95]. Dieses Modell kann zugleich als Prozessvorlage für die evidenzbasierte Problemlösung (Business Analytics, Abschnitt 2.2.2.8) dienen. Die Phasen werden im Folgenden als *Problemspezifikation*, *Datenvorbereitung*, *Datenanalyse*, *Ergebnisaufbereitung* und *Anwendung des Wissens* bezeichnet (Abbildung 10). Das resultierende Modell ergänzt demnach die in Abschnitt 2.3.2.3 identifizierten drei generischen Prozessphasen der Datenanalyse um einen einleitenden und einen abschließenden Schritt.

Die Aktivitäten der beschriebenen Prozessmodelle sind in der Abbildung den Phasen des oben notierten, allgemeinen Modells gegenübergestellt.[34] Diese Zuordnung gelingt ohne größere Schwierigkeiten, was die Annahme bestätigt, dass alle Modelle im Grunde Varianten desselben Basisprozesses repräsentieren [DeHa01, 63]. Da eine tabellarische Darstellung unter der ungleichen Schrittanzahl der Vorschläge

[34] Die Betrachtungsgranularität der Prozessmodelle entspricht jeweils der Darstellung in den Quellen.

leidet, ist die Zugehörigkeit der Aktivitäten zu den generischen Phasen durch hervorgehobene Linien gekennzeichnet [Säub00, 34-36]. Hierbei wird deutlich, dass die Ansätze eine unterschiedliche Reichweite besitzen. Während alle Modelle die Kernphasen Datenvorbereitung, Datenanalyse und (mit Ausnahme des Vorschlags von BAUER & GÜNZEL) Ergebnisaufbereitung abdecken, sehen nur sieben eine Problemspezifikation vor, und nur drei berücksichtigen die Anwendung des Wissens.

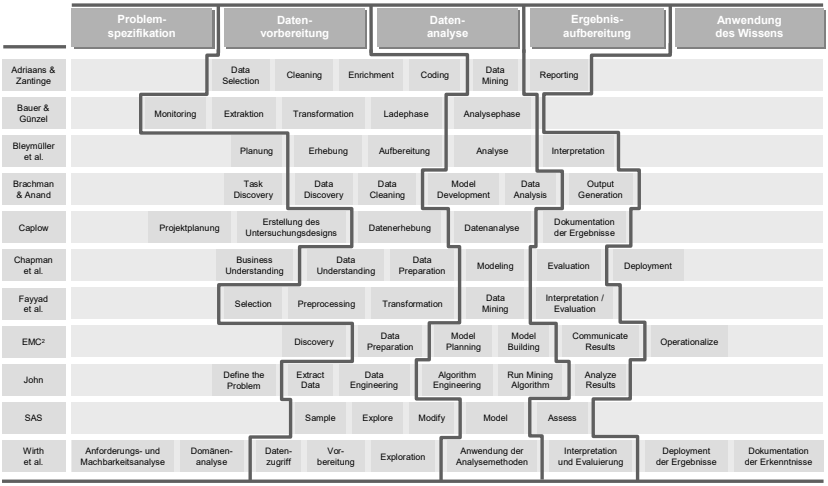


Abbildung 10: Zuordnung von Prozessmodellen zu den generischen Phasen von Datenanalyseprozessen (eigene Darstellung)

Die gewählte Vorgehensweise ist einerseits im Hinblick auf die Vorgabe eines allgemeinen Prozessmodells, andererseits in Bezug auf die abstrakte Darstellung zu kritisieren. Der erste Aspekt ist durch die Dominanz des allen betrachteten Prozessmodellen inhärenten, generischen Dreistufenmodells aus Datenvorbereitung, Analyse und Ergebnisaufbereitung zu entkräften. Die beiden Randphasen ergeben sich zwangsläufig aus den von diesem Muster nicht abgedeckten Aktivitäten. Eine rein induktive Herleitung hätte demnach zu einem ähnlichen Ergebnis geführt. Der zweite Aspekt fokussiert die zunächst unbeantwortete Frage nach der konkreten Ausgestaltung der Phasen [Säub00, 39]. Das allgemeine Modell hat den Charakter eines ***Vorgehens-***

modells, dessen Phasen „nützliche Abstraktionen“ [Somm01, 56] der zur Durchführung einer Analyse erforderlichen Aktivitäten in idealtypischer Reihenfolge darstellen [StGR98, 756]. Ein konkreter Analyseablauf ergibt sich durch situationsspezifische Ausfüllung der generischen Phasen mit den jeweils notwendigen Aufgaben [DeHa01, 65].

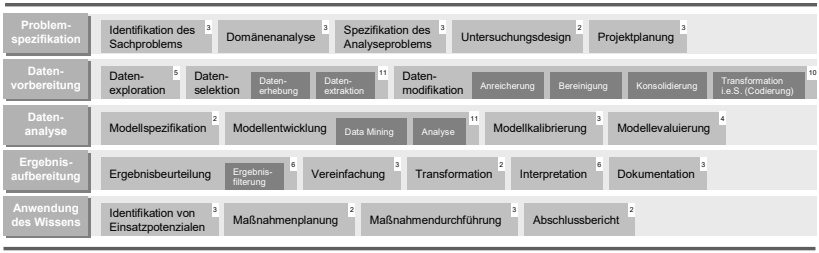


Abbildung 11: Generische Phasen und wichtige Aufgaben von Datenanalyseprozessen (eigene Darstellung)

Vor diesem Hintergrund ermöglicht die vorgenommene Gegenüberstellung die Identifikation von in den Prozessphasen typischerweise auftretenden Aufgaben aus den Schritten der betrachteten Modelle. Abbildung 11 ordnet die aus den Literaturvorschlägen entnommenen Aktivitätsinhalte den generischen Phasen (links) als Aufgaben zu. Die Bezeichnungen wurden vereinheitlicht. Jeder Aufgabe ist jeweils ihre Häufigkeit nachgestellt (weißer Kasten); zusätzlich sind häufig genannte Ausprägungen oder Teilaufgaben schattiert abgebildet.

3.1.2 Prozessaufgaben

Eine detaillierte Beschreibung der in Datenanalyseprozessen auftretenden Aufgaben würde den Rahmen dieser Arbeit sprengen.[35] Im Folgenden wird, geordnet nach den fünf generischen Phasen, ein Überblick über die in Abbildung 11 enthaltenen wichtigsten Aufgaben vermittelt. Hierbei wird zur breiteren Fundierung neben den in Abschnitt 3.1.1 benutzten Quellen auf weitere Literatur zurückgegriffen.

[35] Eine ausführliche Fassung dieses Abschnitts mit detaillierten Erläuterungen einzelner Aufgaben ist in [Knob07] nachzulesen.

3.1.2.1 Problemspezifikation

Die erste Phase dient der Festlegung des Ziels und zugehöriger Erfolgskriterien der Untersuchung. Am Anfang steht die *Identifikation des Sachproblems*, dessen Lösung mithilfe einer Datenanalyse unterstützt werden soll [HeMi94, 19], [Drei94, 17]. Hierbei kann es sich um einen unerwünschten Zustand der Diskurswelt, um ein Entscheidungs- oder Gestaltungsproblem handeln (z.B. Verbesserung des Werbeerfolgs). Im Rahmen einer *Domänenanalyse* kann Hintergrundwissen gesammelt werden, um ein tieferes Verständnis des Problems zu erlangen und dieses zu konkretisieren oder systematisch zu gliedern [WSG+97, 246], [NeKn15, 177-180]. Da sich Sachprobleme in der Regel auf theoretische Aussagen beziehen, sind diese zur Lösung mittels Datenanalyse zunächst in empirisch überprüfbare Aussagen zu überführen. Der Vorgang der „Messbarmachung" theoretischer Konstrukte heißt ***Operationalisierung*** und identifiziert für jedes Konstrukt geeignete empirische Indikatoren [Drei94, 22f., 77] (z.B. Verkaufszahlen beworbener Produkte als Indikator für Werbeerfolg). Auf deren Grundlage kann die *Spezifikation des Analyseproblems* erfolgen. Ein Analyseproblem beschreibt die mit der Datenanalyse beabsichtigte Transformation von Daten in Informationen, bestimmt also Anforderungen an geeignete Analysedaten und an die zu erzeugenden Analyseergebnisse [Knob03a, 339f.].

Für jedes Analyseproblem wird ein *Untersuchungsdesign* erstellt, das nach Maßgabe der zuvor definierten Elemente das Vorgehen bei der Analyse bestimmt. Hierzu gehören neben methodischen Festlegungen insbesondere Entscheidungen zur Analysestrategie (Verkettung von Analysen) sowie die Erstellung eines Prozessplans. Die beschriebenen Aufgaben werden von der *Projektplanung* flankiert, die sich vor allem Organisations-, Ressourcen-, Zeit- und Kostenaspekten widmet [Drei94, 3, 24]. Sie umschließt auch die angestrebte Anwendung des erlangten Wissens und definiert Erfolgskriterien zur späteren Evaluierung des Projekts [DeHa01, 66].

3.1.2.2 Datenvorbereitung

Zweck der zweiten Phase ist die Vorbereitung geeigneter Daten gemäß den Erfordernissen der geplanten Analyse. Sie verfolgt hierzu im

Wesentlichen vier Ziele [FSWS97, 4], [EnTh98, 430]: Zur (1) *Datenbereitstellung bzw. -beschaffung* erfolgt die Identifikation geeigneter Datenquellen und die Auswahl relevanter Daten nach Maßgabe des im Analyseproblem formulierten Bedarfs [Knob01, 89], [HiWi01, 25]. Liegen passende Daten nicht bereits vor (etwa in einem Data Warehouse), muss zunächst eine Datenerhebung durchgeführt werden. Diese Ziele werden von der Aufgabe *Datenselektion* realisiert. Die eigentliche Vorbereitung der bereitgestellten Daten betrifft (2) die *Beseitigung von Datenmängeln* sowie (3) die *Veränderung der Daten* zur Erhöhung der Effektivität und Effizienz der Analyse. Bei Verzicht auf adäquate Vorbereitung der Rohdaten ist mit negativen Auswirkungen auf die Nützlichkeit (insbesondere die Gültigkeit) der Analyseergebnisse zu rechnen [FaPS96, 4], weshalb entsprechende Aufgaben als obligatorische Elemente jedes Analyseprozesses gelten. Sie werden zur *Datenmodifikation* zusammengefasst. Zur systematischen Planung bzw. Durchführung der Aufgaben der Vorbereitungs-, Analyse- und Aufbereitungsphasen ist schließlich (4) ein *Kennenlernen der Daten* durch verschiedene Voruntersuchungen anzuraten. Die zugehörige Aufgabe wird als *Datenexploration* bezeichnet.[36]

Die Notwendigkeit der Datenvorbereitung lässt sich zu großen Teilen auf Datenqualitätsprobleme zurückführen. Da Daten stets Abbildungen realer Sachverhalte sind scheint es naheliegend, ihre Qualität über die korrekte Repräsentation dieser Sachverhalte zu definieren (inhärente Datenqualität) [Engl99, 22]. Da es jedoch im Allgemeinen mehrere verschiedene korrekte Repräsentationen eines Sachverhalts gibt [Jaco91, 72], ist die Qualität von Daten zusätzlich auf ihre Geeignetheit für einen definierten Zweck („fitness for use“) zu beziehen[37] [TaBa98, 54]. Nun stimmt der Zweck der Datenerfassung bzw.

[36] In der Literatur finden sich auch Bezeichnungen wie *data auditing* [Pyle99, 113], *data discovery* [BrAn96, 49] oder *data profiling* [BeLi97, 67] [ABEM15, 131].

[37] Angelehnt an das Total Quality Management wird Datenqualität pragmatisch definiert als die Eigenschaft, konsistent die Erwartungen der Nutzer zu erfüllen [Cros79, 15]. Da die Erwartungen jedoch häufig nicht eindeutig sind und einem Wandel unterliegen können, kann Analysedaten kein objektiver, sondern lediglich ein potenzieller Wert beigemessen werden, der sich erst dann realisiert, wenn die Daten nutzbringend in einer konkreten Untersuchung eingesetzt werden [Engl99, 22].

-erzeugung (insbesondere bei Sekundärdaten) häufig nicht mit der Intention der geplanten Analyse überein (Adäquationsproblem) [HeMi94, 24]. In solchen Fällen müssen vorhandene Daten erst in eine adäquate Repräsentation transformiert werden [Benn94, 39], [Devl97, 7], [KeFi98, 63], [Cham98, 233]. ***Datenmängel*** sind demnach bestimmte Eigenschaften der Daten, die aus analytischer Sicht problematisch sind, weil sie zur Erzeugung unbrauchbarer Resultate führen oder den Einsatz von Analysemethoden behindern [FSWS97, 5]. Die Datenvorbereitung zielt hauptsächlich auf die Erkennung und Beseitigung solcher Mängel. Maßnahmen zur Behandlung des Adäquationsproblems sind spezifisch für eine Analyse, während jene zur Behebung inhärenter Qualitätsprobleme als analyseunabhängig eingeordnet werden können [BöKU03, 173-175].

Eine anwendungsorientierte Klassifikation ordnet Datenqualitätsprobleme den Problemkreisen Verfügbarkeit, Inhalt und Repräsentation zu[38] [Knob01, 91f.]. Sie werden im Folgenden kurz charakterisiert, um wichtige Teilaufgaben der Datenvorbereitung abzuleiten. Der Problemkreis *Verfügbarkeit* bezieht sich auf Umfang und Aussagegehalt der Daten. Obwohl zuweilen sehr umfangreiche Datenbestände vorliegen, sind wichtige Aussagen häufig nicht enthalten oder unterrepräsentiert [DeHa01, 236f., 247f.]. Bei zeitbezogenen Daten stellt die Dynamik der repräsentierten Sachverhalte ein Problem dar, wenn von der Untersuchung zu erfassende Veränderungen in den Daten nicht abgebildet sind [FrPM91, 9], [KrWZ98, 31]. Der Verfügbarkeitsaspekt deckt somit Probleme des Datenvolumens, der Dynamik, fehlender Fälle (Datensätze) und fehlender Merkmale (Attribute) ab [Biss96, 8-10], [Küpp99, 114ff.]. Die Behandlung der beiden erstgenannten Punkte ist Aufgabe der *Datenselektion*, während fehlende Aussagen durch *Anreicherung* aus anderen Quellen oder Berechnungen ergänzt werden können.

Der zweite Problemkreis betrifft den *Inhalt* verfügbarer Daten. Inhaltliche Mängel können Analyseergebnisse verfälschen und liegen vor, wenn ein Teil der Datensätze fehlende Werte, unsichere, ungenaue oder fehlerhafte Werte aufweist, bestimmte Sachverhalte redundant

[38] Vgl. für alternative Systematisierungen z.B. [FSWS97, 5-9] und [KCHK+03].

repräsentiert oder durch semantische Inkonsistenzen (z.B. Mehrfacherfassung von Objekten; Synonyme) gekennzeichnet ist [FrPM91, 9f.], [Biss96, 9]. Nicht oder nicht korrekt gefüllte Datenfelder sind Gegenstand der *Datenbereinigung*, während die Behandlung von Redundanzen und semantischen Inkonsistenzen Aufgabe der *Konsolidierung* ist. Der dritte Problemkreis beschreibt ungeeignete Formen der *Repräsentation*. Daten aus verschiedenen Quellen können häufig aufgrund syntaktischer Inkonsistenzen nicht direkt miteinander verknüpft werden. Dieses Problem kann durch Konsolidierung gelöst werden. Für die vorgesehene Analyse (Methode) ungeeignete Darstellungsformen, Granularitäten oder Datenschemata werden durch *Transformation (i.e.S.)* behandelt [AdZa96, 40, 44-46], [BeLi97, 67-69]. Sie gewährleistet die Durchführbarkeit der Analyse, beeinflusst aber auch deren Effektivität und Effizienz.

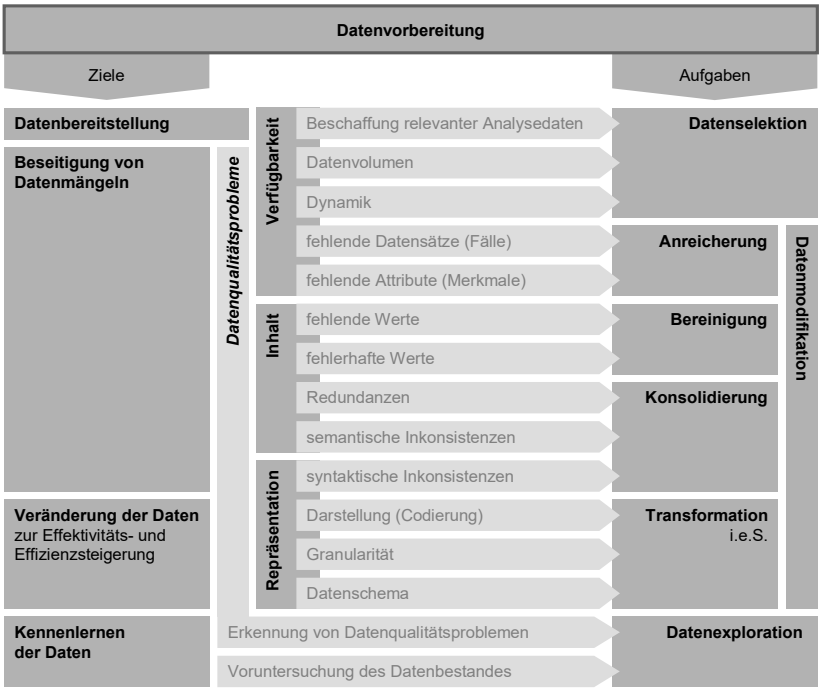


Abbildung 12: Ziele und Aufgaben der Datenvorbereitung, in Anlehnung an [Knob01, 92]

Abbildung 12 stellt die Ziele der Datenvorbereitung den Datenqualitätsproblemen und den zugehörigen Aufgaben gegenüber. Die Datenbereitstellung wird, wie eingangs beschrieben, von der Datenselektion realisiert. Die Datenexploration leistet wichtige Beiträge zur Identifikation weiterer Analyseziele, zum Untersuchungsdesign sowie zur Kontrolle der Daten- und Analyseergebnisqualität (Unterstützung der Interpretation) [Drei94, 300].

3.1.2.3 Datenanalyse

Die dritte Phase wird im Folgenden aus der Perspektive der Modellerstellung betrachtet. Modelle sind integrierte, verifizierte Mustersysteme (vgl. Abschnitt 2.1.3.4), weshalb ihre Herleitung den umfassendsten Fall der Analyse darstellt. In anderen Fällen ist jeweils nur ein Teil der hier beschriebenen Aufgaben relevant. Die Modellerstellung gliedert sich in die Teilaufgaben Spezifikation, Entwicklung, Kalibrierung und Evaluierung [BrAn96, 44-46], [BeLi97, 78-80].

Nach Auswahl eines zum Analyseziel passenden Modelltyps (z.B. Entscheidungsbaum) muss ein geeignetes Berechnungsverfahren gewählt und adäquat parametrisiert werden (*Modellspezifikation*). Die Parametrisierung[39] umfasst etwa die Festlegung der Eingabe- und Zielvariablen (Merkmal, dessen Ausprägungen zu berechnen sind) mit zugehörigen Gewichtungsfaktoren, die Vorgabe statistischer Gütekriterien (z.B. Signifikanzniveau) oder methodenspezifischer Parameter (z.B. Baumtiefe). In der Regel lässt sich nicht sofort eine zufrieden stellende Verfahrenskonfiguration finden; daher wird das gewählte Verfahren solange mit variierenden Einstellungen auf die Daten angewandt, bis ein akzeptables erstes Modell vorliegt [KrWZ98, 31], [Küpp99, 121]. Bei Bedarf ist auch der Austausch von Verfahren oder Modelltyp möglich. Diese *Modellentwicklung* wird auch als Training des Modells bezeichnet [BeLi97, 78f.]. Modellspezifikation und Modell-

[39] Der Begriff Parameter wird in der Literatur sowohl für *Methodenparameter* als auch für *Modellparameter* verwendet und oft nicht klar differenziert. Methodenparameter sind an den Berechnungsverfahren vorgenommene Einstellungen zur Steuerung der Modellerstellung. Modellparameter sind Struktur- und Verhaltenseigenschaften des generierten Modells [HiWi01, 71].

entwicklung verlaufen iterativ und sind stark miteinander verzahnt [HiWi01, 72] (vgl. Verkettungsgründe in Abschnitt 2.3.2.3).

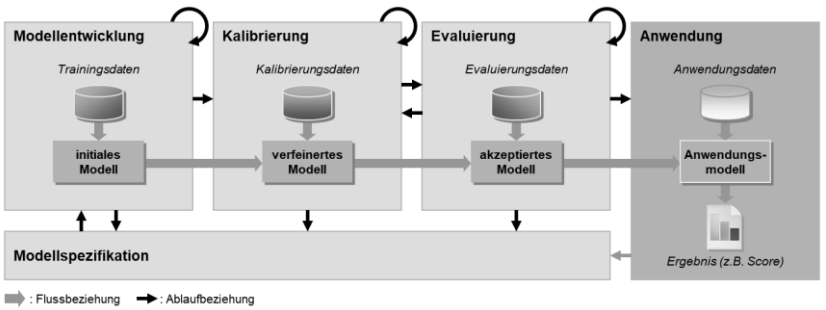


Abbildung 13: Iterative Modellerstellung, in Anlehnung an [BeLi00, 185]

Das Verhalten des generierten Modells wird anschließend durch *Kalibrierung* so justiert, dass neben der erforderlichen Approximations- und Prognosegenauigkeit gleichzeitig ein akzeptabler Generalisierungsgrad erreicht wird [ElPr96, 94f.]. Das zu minimierende Kriterium ist eine gewichtete Summe aus Modellkomplexität und Trainingsfehler. Zu hohe Modellkomplexität birgt das Risiko, dass zufällige Phänomene in den Trainingsdaten fälschlicherweise verallgemeinert werden (Überanpassung). Trainingsfehler treten auf, wenn das Modell gegebene Beispielfälle nicht ausreichend präzise reproduzieren kann (Unteranpassung). Die Kalibrierung erfolgt iterativ durch Prüfung des Modells anhand eines von den Trainingsdaten disjunkten Kalibrierungsdatensatzes und durch entsprechende Anpassung der Methodenparameter [HiWi01, 74]. Das resultierende Modell sollte einer abschließenden *Evaluierung* unterzogen werden, die auf einer eigens hierfür reservierten Datenmenge erfolgt, die nicht in die Modellerstellung eingeflossen ist. Auf diese Weise kann abgeschätzt werden, wie sich das Modell bei Anwendung auf neue Daten verhält [Ehre76, 29], [BeLi97, 79]. Ist die Fehlerrate nicht akzeptabel, muss die Kalibrierung wiederholt werden [BrAn96, 44]. Abbildung 13 illustriert das beschriebene Vorgehen.

3.1.2.4 Ergebnisaufbereitung

Die vierte Phase umschließt mit der Interpretation der Ergebnisse eine zentrale Komponente der Datenanalyse (vgl. Abschnitt 2.1.3.6). Vor der inhaltlichen Deutung können weitere Aufgaben hilfreich sein, um Verständlichkeit und Zugänglichkeit der Analyseergebnisse zur weiteren Verarbeitung zu verbessern [FaPS96b, 83], [BrFa00], [WHKR00, 156f.].

Analyseverfahren produzieren oft zahlreiche Einzelaussagen (z.B. Regeln), die in vielen Fällen zur Lösung des Sachproblems nicht relevant und in ihrer schieren Masse unübersichtlich sind. Daher beginnt die Ergebnisaufbereitung mit einer *Beurteilung*, um den Informationscharakter der Aussagen festzustellen und nicht signifikante, triviale, redundante oder nicht hilfreiche Aussagen auszusondern (Relevanzfilter) [HoKl91, 329], [Bigu96, 14]. Diese Filterung kann zum Teil auf Grundlage statistischer Maße (Modellevaluierung, Abschnitt 3.1.2.3) oder monetärer Bewertungen objektiv erfolgen, erfordert aber in der Regel auch kontextabhängige Erwägungen und ist demnach mit der Interpretation verwoben. Eine weitere *Vereinfachung* umfangreicher Ergebnismengen ist durch Sortierung und Ordnung nach verschiedenen Kriterien möglich, z.B. mittels Aggregation oder hierarchischer Gliederung inhaltlich assoziierter Aussagen (Konstruktion von Domänenmodellen). Sie kann durch Visualisierungswerkzeuge effektiv unterstützt werden [AmCo94b, 45]. Ergebnisse, die in einer schwer verständlichen Repräsentationsform vorliegen oder zur maschinellen Weiterverarbeitung vorgesehen sind, bedürfen häufig einer entsprechenden *Transformation* (vgl. Abschnitt 2.3.2.2). Für den Menschen sind natürlichsprachige oder regelartige Ausdrücke sowie grafische Darstellungen gut lesbar, zur maschinellen Nutzung in Anwendungssystemen eignen sich z.B. deklarative Formalismen oder Programmcode [FrPM91, 13]. Die Übermittlung an bestimmte Endgeräte macht häufig zusätzlich eine gerätespezifische Anpassung notwendig [KoRS02, 47].

Im Rahmen der auf Hintergrundwissen gestützten *Interpretation* sollen die konzeptuelle Bedeutung und die kontextuellen Implikationen der Analyseergebnisse eruiert werden [HoKl91, 325]. Hierbei ist insbesondere ihre Anwendbarkeit auf das vorliegende Sachproblem zu beachten. Die Erkenntnisse der Untersuchung werden in einer

Dokumentation für die Auftraggeber detailliert zusammengefasst [BrAn96, 47].

3.1.2.5 Anwendung des Wissens

Soll Datenanalyse wirksam zur Lösung eines Sachproblems beitragen, so sind die gewonnenen Erkenntnisse in Entscheidungen oder Handlungsmaßnahmen zu überführen, um einen messbaren geschäftlichen Nutzen zu stiften [CHS+97, 12], [KoRS02, 48], [MiCD13, 101]. Diese sind Gegenstand der fünften Phase. Im Idealfall legt die Problemspezifikation bereits Einsatzbereiche fest. Andernfalls müssen Anwendungspotenziale erst identifiziert werden. Dies kann auch dann notwendig werden, wenn die Analyse nicht zu den erwarteten Ergebnissen geführt hat.

Im Allgemeinen lassen sich drei Verwendungsformen des Wissens unterscheiden: (1) die Durchführung singulärer Maßnahmen, (2) die regelmäßige Unterstützung operativer Prozesse [KrWZ98, 31], [Bigu96, 14] sowie (3) die Erbringung von Informationsdienstleistungen. Im ersten Fall werden Analyseergebnisse bei konkreten Entscheidungen des Managements berücksichtigt, etwa zur einmaligen Prognose bestimmter Marktentwicklungen oder zur Konzipierung umfassender Projekte, wie etwa einer Marketingkampagne oder Unternehmensreorganisation. Im zweiten Fall erfolgt die Integration der Ergebnisse (z.B. einer Kundenklassifizierung) in operative Datenbanken oder Anwendungssysteme (z.B. in Form eines Modells zur Realisierung eines automatischen Empfehlungssystems im Online-Handel) [DeHa01, 72], [Knob01, 106]. Der dritte Fall betrifft die Entwicklung von Produkten, die eine Information bereitstellen (Data Products) [StSt14, 472], etwa in Form einer Web-Suchmaschine oder eines mobilen Services zur Verkehrsstauvorhersage.

Die fünfte Phase umfasst folgende Aufgaben: Nach *Identifikation von Einsatzpotenzialen* erfolgt die *Maßnahmenplanung*, die in Abhängigkeit von der gewählten Anwendung mehr oder weniger detailliert ausfällt. Im Allgemeinen ist ein Projektplan zu erstellen, der auch Mittel zur Überwachung von Durchführung bzw. Betrieb sowie zur Erfolgskontrolle enthält [CCK+00, 60]. Nach *Maßnahmendurchführung* ist ein

Abschlussbericht zu erstellen, der sämtliche Erfahrungen aus dem Projekt dokumentiert. Dies schließt eine Evaluation des gesamten Vorhabens (Datenanalyse einschließlich Anwendung des Wissens), Berichte über positive und negative Aspekte der gewählten Vorgehensweise, Verbesserungspotenziale für künftige Projekte und eine Bewertung des ökonomischen Erfolgs ein [WSG+97, 247], [CCK+00, 33], [DeHa01, 72].

3.1.3 Datenanalyse als iterativ-inkrementeller Prozess

Klassische Vorgehensmodelle implizieren eine streng sequenzielle Ordnung der Aktivitäten im Sinne abgeschlossener Phasen, die innerhalb eines Prozesses jeweils nur einmal auftreten. Bei solch kaskadischen Abläufen „fließen" die Bearbeitungsobjekte förmlich von einer Aktivität zur nächsten [Somm01, 24, 57]. Diese Idealvorstellung ist in der Analysepraxis problematisch und in ihrer strikten Form letztlich nicht aufrecht zu erhalten. Datenanalyseprozesse verlaufen typischerweise *iterativ*.[40] Wiederholungen einzelner Schritte und Rücksprünge zu vorherigen Phasen sind üblich, können jederzeit und bezüglich jeder Aufgabe auftreten [BrAn96, 52], [FaPS96, 4, 11], [FaPS96b, 83f.], [ZLKO97, 291], [KrWZ98, 30]. Da im Voraus nicht bekannt ist, welche Mängel und Erkenntnisse die Daten offenbaren, können viele Aktivitäten nicht a priori spezifiziert und der Prozess nicht vollständig linear abgearbeitet werden [DeHa01, 64].

Abbildung 14 zeigt anhand eines einfachen Beispiels, wie die tatsächliche Vorgangsfolge (Instanzebene) von der idealtypischen Reihenfolge abweicht und wie die Phasen aufgrund mehrfach ausgeführter Aufgaben miteinander verzahnt sind. Die Aufgabenwiederholung führt auf Typebene zu Prozessschleifen und Sprüngen. Solche durch Versuch und Irrtum geprägten Prozessabläufe heißen *evolutionär* [Mali00, 265].

---

[40] Im Gegensatz zur Sequenzialität bezeichnet Iterativität die wiederholte Ausführung gewisser Aufgaben oder Teilprozesse, bis ein bestimmtes Abbruchkriterium erfüllt ist [EnLS97, 164].

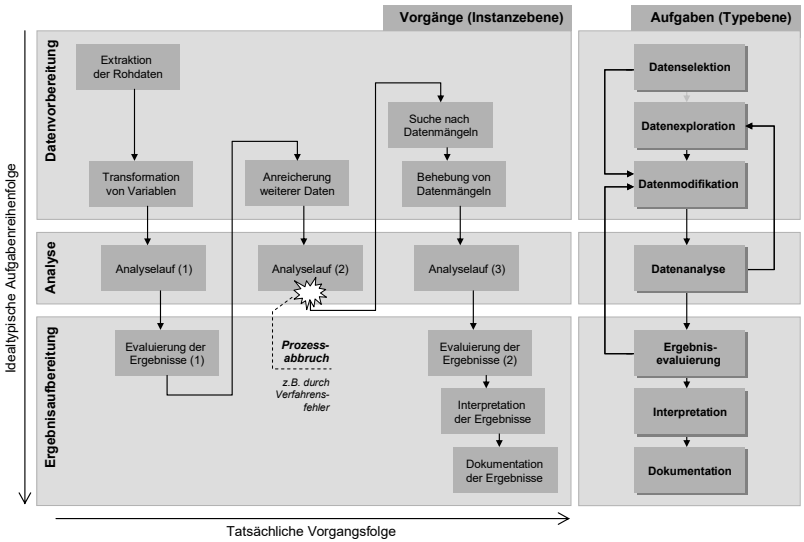


Abbildung 14: Beispiel eines iterativ-inkrementellen Ablaufs von Datenanalyseprozessen (eigene Darstellung, vgl. zur Instanzebene [NeKn15, 166])

Die folgenden Abschnitte untersuchen die Ursachen für die iterative Natur von Datenanalyseabläufen und erläutern wichtige Einflussfaktoren. Die Darstellung ist nach den drei Beschreibungsebenen von Analyseprozessen gegliedert.

3.1.3.1 Ebene der Analyseziele

TUKEY zufolge ist eine gute Datenanalyse extensiv, wird also nicht als isolierte Einzelanalyse geplant und ausgeführt, sondern umfasst stets mehrere Iterationen oder Folgeuntersuchungen [Tuke62, 62]. Analyseprozesse führen demnach selten geradlinig von einer Frage zur Antwort [Vell97, 328]. Als Begründung lässt sich anführen, dass streng sequenzielle Vorgehensweisen nur praktikabel sind, wenn die Anforderungen zu Beginn möglichst umfassend und klar spezifiziert vorliegen [Somm01, 58]. In der explorativen Datenanalyse wird auf eine Hypothesenformulierung weitgehend verzichtet, um die Aufmerksamkeit nicht von vorneherein auf bestimmte Aspekte einzuschränken (Prinzip der Offenheit) [Flic02, 69-71]. Eine vage Vorstellung, welche

Aussagen im Kontext des zu behandelnden Sachproblems relevant sind, gilt zunächst als ausreichend [BrAn96, 45f.], womit ein beträchtliches Maß an Unsicherheit bezüglich des genauen Vorgehens verbunden ist [Ehre76, 429]. Auch bei anderen Analyseansätzen fällt die Entscheidung über den Fortgang der Untersuchung häufig situativ auf Basis bisher erlangter Erkenntnisse. Analyseketten (Abschnitt 2.3.2.3) entwickeln sich dynamisch durch Aneinanderreihung evolvierender Fragestellungen und erzeugen dabei zunehmend detailliertere, umfassendere Beschreibungen der untersuchten Sachverhalte [Tuke62, 4], [AmCo94b, 45], [Hand99, 3].

Ein derart ergebnisgetriebenes Vorgehen führt nicht selten zur Rücknahme bereits gezogener Schlussfolgerungen im Lichte neuer Erkenntnisse [Tuke62, 46], [Vell97, 323]. Datenanalyse hat insoweit *inkrementellen*[41] *und experimentellen Charakter* [Tuke62, 63], [Quin91, x] und kann davon profitieren, wenn alternative Untersuchungsmethoden und konkurrierende Erklärungen ausprobiert und gegebenenfalls wieder verworfen werden [Vell97, 328]. Bei komplexen Untersuchungsgegenständen wird der iterativ-inkrementelle Ansatz für notwendig erachtet, um Theorien und konzeptuelle Aussagen ausreichend verfeinern und begründen zu können [EnMT95, 2] (vgl. empirischer Zyklus, Abschnitt 2.3.2.3). Die vollständige Spezifikation des gesamten Analysepfades erscheint in solchen Situationen kaum möglich, da im Voraus nicht bekannt ist, zu welchen Erkenntnissen die Einzelanalysen führen [DeHa01, 64]. Häufig kann lediglich das initiale Analyseziel mehr oder weniger präzise vorgegeben werden.

3.1.3.2 Ebene der Prozessaufgaben

Iterationen auf Zielebene schlagen sich zum Teil auf die Aufgabenebene nieder, da gewandelte oder neue Analyseziele typischerweise den Einsatz anderer Methoden oder die Modifikation von Analysedaten erfordern [Hand99, 3]. Jedoch konstituiert sich nicht jede Untersuchung

[41] Inkrementelles (schrittweise aufeinander aufbauendes) [Dude17b] Vorgehen steht dem holisitischen Ansatz gegenüber, der alle Aspekte eines Problems ganzheitlich erfasst und zu lösen versucht [Somm01, 64].

durch komplexe Analyseketten; Einzeluntersuchungen mit klar spezifizierten Analyseproblemen sind durchaus von praktischer Relevanz. Auch bei ihnen müssen häufig Aufgaben wiederholt werden, wenn (Zwischen-) Ergebnisse nicht aussagekräftig oder im Hinblick auf bestimmte Gütemaße nicht zufriedenstellend sind [FaPS96, 4]. Wiederholungen sind auch angezeigt, wenn das Analysewerkzeug einen Verfahrensfehler meldet [Ecke04c]. Ursächlich hierfür sind die zahlreichen Freiheitsgrade bezüglich der Auswahl konkreter Analysedaten und -methoden. Prinzipiell können für jede Prozessaktivität verschiedene Datenrepräsentationen herangezogen werden, und im Allgemeinen stehen mehrere verschiedene Methoden zur Auswahl, deren Verhalten in Abhängigkeit von der jeweiligen Parametrisierung stark variiert [WSG+97, 245]. Die Entscheidung, welches Verfahren wie auf welche Daten anzuwenden ist, erfordert viel Erfahrung und Methodenkenntnisse [BrAn96, 46], kann aber häufig nur durch Versuch und Irrtum getroffen werden [HeMi94, 158f.]. Die Auswahl von Eingabevariablen gestaltet sich schwierig, da im Voraus oft nicht abgeschätzt werden kann, welche Einflussgrößen in welcher Weise auf die Analyseziele einwirken [Bigu96, 49], [BeLi97, 78], [KrWZ98, 44]. Datenmängel werden oft erst nach ersten Analyseläufen im Verlauf des Prozesses evident [AdZa96, 84]. Stellt sich heraus, dass die Daten nicht den Anforderungen des Analyseverfahrens genügen, sind Aufgaben der Datenvorbereitung zu wiederholen oder eigens in den Prozess aufzunehmen.

Die Erzeugung von Analyseergebnissen erfolgt somit typischerweise iterativ über die Phasen Datenvorbereitung, Analysedurchführung und Ergebnisaufbereitung hinweg und setzt sich solange fort, bis akzeptable Ergebnisse vorliegen [CFM+97, 147]. Empirische Studien zeigen, dass die Integration der drei Phasen in einen Zyklus zur messbaren Verbesserung der Ergebnisqualität führt [AmCo94, 56-58].[42] Die

[42] Die Wurzeln dieses iterativen Vorgehens können im allgemeinen statistischen Paradigma gesehen werden, demzufolge Variabilität in verschiedene Komponenten zu zerlegen ist (Residuenanalyse) [ElPr96, 85]. Muster in den Residuen sind Hinweise auf Möglichkeiten zur weiteren Verbesserung der Ergebnisse, etwa durch Anpassung des Modells an die Residuen, durch Aufnahme weiterer Variablen oder durch Datentransformation. Das Abbruchkriterium der Iteration ist erfüllt, wenn eine weitere An-

Zirkularität wird gerade als Stärke angesehen, da sie bei konsequenter Anwendung eine permanente Reflexion des gesamten Prozesses und seiner Elemente erzwingt. In eng verzahnten Analyse-/Interpretationszyklen lassen sich dem Analyseziel dienende Methoden und Modelle besser identifizieren als bei klassisch-linearer Vorgehensweise [Flic02, 72]. Zusammenfassend betrachtet entfällt ein beträchtlicher Teil der Aktivitäten eines Datenanalyseprozesses auf Korrekturmaßnahmen. Die Qualität der Aufgabendurchführung, insbesondere jener im Vorfeld der eigentlichen Analyse, hat daher große Bedeutung für die Ergebnisqualität und die Prozesseffizienz [FaPS96, 11], [FaPS96b, 84].

3.1.3.3 Ebene der Ressourcen

Die Instrumentalbeziehung zwischen Aufgaben- und Ressourcenebene bedingt, dass sich der Einfluss der Ressourcen auf das Vorgehen bei der Datenanalyse größtenteils als Konsequenz der auf Aufgabenebene getroffenen Entscheidungen ergibt. Dennoch wirken die Ressourcen auch direkt auf Art und Reihenfolge der ausgeführten Aktivitäten ein. Einerseits kann die Verfügbarkeit neuer oder aktualisierter Daten (passive Ressourcen) dazu anregen, Elemente des Prozesses wiederholt auszuführen [BrAn96, 52]. Andererseits determinieren die Analysewerkzeuge (aktive Ressourcen) die Prozessgestaltung nicht unwesentlich. So stellen Werkzeuge und Verfahren häufig spezielle Anforderungen bezüglich Schema, Skalenniveau oder Codierung der Analysedaten [KrWZ98, 42], [Küpp99, 96], [HiWi01, 57]. Zwar wird die Werkzeugauswahl in der Praxis häufig auf die im Unternehmen bereits im Einsatz befindlichen Systeme eingeschränkt, was die Menge der Verfahrensalternativen auf den ersten Blick limitiert. Die meist breite Palette der in umfassenden Softwareprodukten implementierten Methoden verleitet jedoch geradezu zum Experimentieren mit alternativen Datenrepräsentationen und Verfahren [EnMT95, 3]. Da dieses Ausprobieren häufig unsystematisch erfolgt, ist vielfach mit unbefriedigenden Ergebnissen zu rechnen, die zahlreiche Iterationen zur Folge haben können. Hierin zeigt sich ein dritter Einflussfaktor für iterative

passung zu keiner weiteren Verbesserung führt [HeMi94, 326]. Vgl. die iterative Modellerstellung in Abschnitt 3.1.2.3.

Prozessabläufe aus der Ressourcenebene: Letztlich bestimmen das Verhalten sowie die Kenntnisse und Erfahrungen des menschlichen Analytikers die Vorgehensweise bei der Datenanalyse und somit auch die erreichbare Effizienz [WSG+97, 243f.].

3.2 Umgang mit Komplexität bei der Prozessdurchführung

Die obigen Schilderungen zeigen, welche Schwierigkeiten mit der Datenanalyse verbunden sein können. Dieses Kapitel diskutiert Vorschläge, wie solchen Herausforderungen zu begegnen ist. Hierzu werden zunächst Erfolgskriterien und Fehlerquellen aus der Literatur herausgearbeitet (Abschnitt 3.2.1). Anschließend erfolgt eine theoretische Betrachtung der Prozesskomplexität (Abschnitt 3.2.2) und typischer Komplexitätsgrade (3.2.3). In Abschnitt 3.2.4 werden daraus Ansätze zur Komplexitätsbewältigung abgeleitet und Arbeiten aus der Literatur zugeordnet.

3.2.1 Erfolgskriterien und häufige Fehlerquellen

Wie Erfahrungsberichte und empirische Studien unter Mitwirkung von Analytikern zeigen, werden Zeit- und Kostenbedarf von Analyseprozessen häufig unterschätzt [BrAn96, 38], [KoRS02, 46], [Muns11, 65-67]. Hohe Zeitanteile entfallen auf die Planung der Untersuchung, die Datenvorbereitung und Methodenparametrisierung, aber auch auf Managementaufgaben wie die Verfolgung des Projektfortschritts, die Verwaltung von Daten, Zwischen- und Endergebnissen, sowie auf die Handhabung nicht integrierter Analysewerkzeuge [BrAn96, 41], [BrSW97, 131]. Insgesamt dient ein Großteil der Tätigkeiten nicht unmittelbar der eigentlichen Datenanalyse; ihr Anteil am Gesamtaufwand liegt nur zwischen 5% und 20%[43] [Bigu96, 12], [WiHu96, 2], [Wild01, 16], [Muns11, 65f.]. Die langen Projektlaufzeiten gefährden

[43] Eine von MUNSON [Muns11] unter Praktikern durchgeführte Umfrage weist der Datenselektion 20%, der Datenmodifikation 30% und der Analysephase 14% des Zeitbedarfs zu. 20% verbleiben für Bewertung und Interpretation der Ergebnisse (jeweils Median der Antworten). Ein Vergleich des Autors mit anderen Studien bestätigt im Wesentlichen diese Verteilung.

häufig den Projekterfolg. Ein Datenanalyseprojekt gilt im Allgemeinen als gescheitert, wenn die benötigten Ergebnisse nicht oder nicht zeit- und budgetgerecht geliefert werden können [WSG+97, 244], [DeHa01, 229]. Insgesamt lassen sich sechs Klassen von Ursachen für das Scheitern von Datenanalyseprojekten identifizieren [Pyle04a].

Zum Ersten weckt eine unklare oder fehlerhafte Problemspezifikation vielfach unrealistische Erwartungen, die zwangsläufig enttäuscht werden. Sie können sowohl die Lösung des Sachproblems als auch das Analyseproblem betreffen. Beispielweise kann ein faktisch erfolgreiches Projekt, das eine Reduzierung der Kundenabwanderung um 10% erreicht, als gescheitert erscheinen, wenn aufgrund unklarer Ziele eine höhere oder vollständige Reduzierung erwartet wurde. Ebenso gelten Analysen, die nicht die erwarteten Ergebnisse liefern, in vielen Fällen als erfolglos. Diese zeigen aber bei fachkundiger Ausführung zunächst nur, dass (implizit) formulierte Hypothesen durch die vorliegenden Daten nicht bestätigt werden [DeHa01, 235]. Die explizite Formulierung der Analyseziele hilft, solche Fehlinterpretationen zu verhindern. Werden diese Ziele nicht auf das Sachproblem ausgerichtet, sind die Ergebnisse nutzlos, bezüglich des Ziels aber möglicherweise korrekt. Die Qualität der *Problemspezifikation* ist demnach entscheidend für die Effektivität der Datenanalyse [DeHa01, 257], zumal entsprechende Mängel im Rahmen der Prozessdurchführung häufig nicht seriös kompensierbar sind [HeMi94, 19], [Pyle04a]. Bei explorativ-evolutionären Analysen ist zudem eine fortlaufende Kontrolle ihrer Ausrichtung auf das Sachproblem (*Zielorientierung*) erforderlich, da die Untersuchung andernfalls leicht in ungeplante Richtungen gelenkt wird [ElPr96, 98], [AmCo94, 46].

Zum Zweiten leidet der Erfolg der Projekte häufig unter einem unsystematischen Vorgehen, bei dem der Analytiker die eigene Intuition etablierten Handlungsschemata vorzieht.[44] Letztere werden gerne als Hilfsmittel für unerfahrene Anfänger verschmäht. Die Konsequenz ist vielfach die Vernachlässigung von Datenvorbereitung und Ergebnisaufbereitung zugunsten der Analysephase. In der Praxis ist eine

[44] Dieses Verhalten wird zuweilen durch die Neigung der Analytiker, nicht zielorientierte Untersuchungen ohne Problembezug zu betreiben, gefördert. DELMATER & HANCOCK sprechen hierbei von einem „Spielen“ mit den Analysewerkzeugen [DeHa01, 240].

Konzentration auf technische Belange zu beobachten, die eine Ergebnisoptimierung vorwiegend auf Grundlage von Verfahrens- und Werkzeugparametern verfolgt [Pyle04a]. Mangelhafte Datenvorbereitung kann jedoch gravierendere Fehler verursachen als Versäumnisse bei der Analyse selbst. Unterbleibt zugleich die Verifikation der Ergebnisse, endet die Analyse mit fehlerhaften Resultaten [ElPr96, 98], [DeHa01, 230-232]. Das geschilderte Verhalten verursacht Iterationen im Prozess und Projektverzögerungen [Ecke04c]. Das Ignorieren bzw. die Abwesenheit *einsatzfähiger Prozessmodelle* mit geeigneter Werkzeugunterstützung gilt als Hauptursache für die genannten Schwierigkeiten [WSG+97, 244f.], [DeHa01, 240].

Zum Dritten trägt oft die unsachgemäße *Auswahl und Anwendung von Analyseverfahren* zum Misserfolg einer Untersuchung bei [DeHa01, 235]. Die Vielfalt der verfügbaren Methoden verschärft die Auswahlproblematik selbst für Anwender mit mathematisch-statistischem Hintergrund [EnMT95, 12]. Analytiker verwenden häufig jene Ansätze und Methoden, mit denen sie am besten vertraut sind, die aber nicht zwingend die beste Wahl in der gegebenen Situation darstellen [LaPh94, 23], [Pyle04a].

Zum Vierten besteht stets das Risiko der Fehlinterpretation der Ergebnisse. Zu ihrer korrekten Deutung sind neben Methodenkenntnissen insbesondere profunde *Domänenkenntnisse* erforderlich [Dree01, 135]. Darüber hinaus ist *Hintergrundwissen* über Herkunft und Entstehung der Analysedaten hilfreich für die adäquate Analysedurchführung und Ergebnisinterpretation. Entsprechendes Wissen steht dem Analytiker jedoch häufig nicht in ausreichendem Maße zur Verfügung, weil wichtige Diskussionen mit Fachexperten unterbleiben [Pyle04a].

Zum Fünften ist festzustellen, dass *Dokumentation und Protokollierung* der Analysen und ihrer Ergebnisse meist stark vernachlässigt werden [WSG+97, 245], [Pyle04a]. Ursächlich sind der damit verbundene Zusatzaufwand und das Fehlen einer adäquaten Methodik. Die ausgeführten Aktivitäten und Prozesse sind daher schwer zu rekonstruieren. Fehlerdiagnose, Projektevaluierung und Rechtfertigung gegenüber dem Auftraggeber werden dadurch erschwert. Auch der

spätere Rückgriff auf die Erfahrungen eines erfolgreichen Projekts ist ohne diese Informationen nur schwer möglich [LaPh94, 23], [Pyle04a].

Zuletzt hängt der Erfolg einer Analyse auch in starkem Maße vom ausführenden Analytiker ab (Abschnitt 3.1.3.3). Insofern erscheinen Schulungsmängel als weitere Ursache des Scheiterns. Erhöhter Schulungsbedarf entsteht aber häufig erst durch Analysesoftware, die schwer bedienbar ist oder die Prozessdurchführung nur unzureichend unterstützt [WSG+97, 244f.]. So ist der erfolgreiche Einsatz der Datenanalyse in der betrieblichen Praxis auf die Verfügbarkeit *benutzerfreundlicher Analysewerkzeuge* angewiesen, die sowohl explizit eine methodisch fundierte Vorgehensweise unterstützen als auch eine breite Palette an leistungsfähigen Verfahren anbieten [Gros09b, 8].

Zielorientierung	**Problemspezifikation ermöglicht zielgerichtete Analysewege** klar definierte, realistische Projektziele; Spezifikation des Sachproblems und Ableitung eines geeigneten Analyseproblems; Ausrichtung des Analysepfads an den Zielen
Methodik und Prozessmodelle	**Verfügbarkeit einer Analysemethodik mit einsatzfähigen Prozessmodellen** Prozessmodelle mit allen wichtigen Aufgaben (Vermeidung von Fehlern und Iterationen); Vorgabe eines zielorientierten, methodischen Handlungsplans als Orientierung
Korrekte Verfahrensauswahl	**Unterstützung des Analytikers bei der Auswahl geeigneter Analyseverfahren** Vermeidung der Wahl ungeeigneter Verfahren; Leitlinien zur Auswahlentscheidung
Domänen- und Hintergrundwissen	**bedarfsberechte Versorgung mit Domänen- und Hintergrundwissen** Domänenwissen und Kenntnisse über Herkunft und Wesen der Daten unterstützen die Analyseplanung und helfen bei der Vermeidung von Fehlinterpretationen der Ergebnisse
Protokollierung von Prozesswissen	**ausführliche Dokumentation der Projekte und Prozesse** Unterstützung bei der Protokollierung des Vorgehens zur Evaluierung und Dokumentation; Möglichkeit der Wiederverwendung von Erfahrungswissen in späteren Projekten
Werkzeug-unterstützung	**Unterstützung durch leistungsfähiges Datenanalysewerkzeug** Unterstützung bei der Anwendung der Analyse- und Transformationsverfahren; benutzerfreundliche und methodikgestützte Prozessdurchführung

Abbildung 15: Wichtige Erfolgsfaktoren für Datenanalyseprojekte (eigene Darstellung)

Die vorstehenden Überlegungen führen zu den in Abbildung 15 zusammengefassten wesentlichen Erfolgsfaktoren der Datenanalyse: (1) Die Zielorientierung durch klare Problemspezifikation, (2) die Verfügbarkeit einer Methodik mit einsatzfähigen Prozessmodellen, (3) die Unterstützung bei der Auswahl geeigneter Analyseverfahren, (4) bedarfsgerechte Versorgung mit Domänen- und Hintergrundwissen,

(5) die umfassende Dokumentation der Prozesse und Projekte, sowie (6) die Unterstützung durch ein leistungsfähiges, auf die Methodik ausgerichtetes Werkzeug (vgl. auch [FrPM91, 19], [DeHa01, 299, 266]).

3.2.2 Prozess- und Analysekomplexität

Die geschilderten Schwierigkeiten sind zum großen Teil Ausdruck der Komplexität, die mit einer Datenanalyse häufig verbunden ist. Die Komplexität eines Systems wird im Allgemeinen determiniert durch die Anzahl und Verschiedenheit der Komponenten und Relationen zwischen den Komponenten (Komplexität i.e.S. bzw. Kompliziertheit), die resultierenden Kombinationsmöglichkeiten (Varietät) sowie die Veränderlichkeit der durch diese Faktoren geschaffenen Bedingungen im Zeitverlauf (Dynamik). Aus Perspektive des Analytikers kommt hinzu, dass er sich bei seinen Entscheidungen vor und während der Prozessdurchführung in einer schwach strukturierten Situation befindet und das Problem der Intransparenz der relevanten Bedingungen bewältigen muss [FiWo90, 13-15], [Haus90, 137], [Bron92, 1122], [Mali00, 200f, 257].

Die Komplexität von Datenanalyseprozessen ergibt sich somit aus der objektiven Komplexität der Prozessstruktur (***Prozesskomplexität***) [Gait83, 211] und der subjektiven Komplexität, die während der Handhabung (Planung, Steuerung, Durchführung und Kontrolle) der Prozesse entsteht (***Analysekomplexität*** im Sinne von Problemkomplexität) [Haus90, 137], [Bron92, 1122]. Die wesentlichen Komplexitätstreiber sind in Abbildung 16 zusammengestellt. Auf ihre ausführliche Erläuterung wird verzichtet, vielmehr seien im Anschluss einige ausgewählte Aspekte durch Beispiele illustriert.

Bei der Konzipierung eines Analyseprozesses sind zahlreiche verschiedene Elemente in Einklang zu bringen, zwischen denen Interdependenzen in Form von Instrumentalbeziehungen (z.B. zwischen Analyseproblem und -prozess sowie zwischen Aktivität und Verfahren), Reihenfolgebeziehungen (Verkettung von Analysen), Leistungsverflechtungen (Verknüpfung von Aktivitäten) oder des konkurrierenden Zugriffs auf Ressourcen existieren [Gait83, 160-170], [Vell97, 327]. Die Anforderungen der Analyseverfahren an die Datenrepräsentation

schaffen Prämissen für Datenvorbereitungsaufgaben, die untereinander sowie zur Analysephase besonders starke Abhängigkeiten aufweisen [LaPh94, 22f.], [Hand99, 3], [Müll00, 190].

Abbildung 16: Überblick über wichtige Komplexitätstreiber bei der Datenanalyse (eigene Darstellung)

Aus der Menge aller Kombinationsmöglichkeiten muss eine zulässige, effektive und effiziente Konfiguration gefunden werden [KnWe00, 355], [Knob03a, 345]. Abbildung 17 illustriert das Gestaltungspotenzial anhand eines vereinfachten Beispiels: Stehen nur 3 Datenquellen und 5 Analyseverfahren zur Disposition, resultieren bereits $3 \times 5=15$ Analysemöglichkeiten (Variante a). Da die Daten kaum unmodifiziert ausgewertet werden, steigt die Varietät bei angenommenen 6 Transformationsalternativen auf $3 \times 6 \times 5=90$ (Variante b) an, bei Berücksichtigung von 4 Verfahrensalternativen je Transformation auf $3 \times 6 \times 4 \times 5=360$.[45] Die Einbeziehung der Parametrisierbarkeit jedes

[45] Da im Beispiel die Position der Aufgaben S, M und A durch Typisierung vorgegeben ist, werden die Kombinationsmöglichkeiten bereits stark eingeschränkt. Bei freier

Verfahrens führt schnell zur kombinatorischen Explosion. Die aktuell zulässigen Konfigurationsalternativen sind jeweils eine Teilmenge dieser potenziellen Varietät.

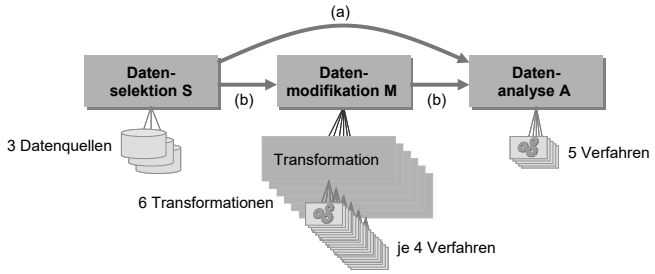


Abbildung 17: Vereinfachtes Beispiel zur Varietät (eigene Darstellung)

Die Dynamik resultiert hauptsächlich aus der Veränderung von Daten, der Produktion von (Zwischen-) Ergebnissen sowie evolvierenden Analysezielen. Sie überträgt sich aufgrund der Interdependenzen auf andere Komponenten und kann dort Korrektur- und Anpassungsmaßnahmen erfordern (parallele Evolution) [Vell97, 327f.]. Trotz theoretischer Möglichkeiten zur Analyse dieser Zusammenhänge ist die vollständige Erfassung der Dynamik eines Prozesssystems nicht realistisch [Gait83, 207], [Mali00, 187].

Informationsmangel (Intransparenz) in Bezug auf Prämissen, Interdependenzen und Dynamik steigert die Problemkomplexität des Analytikers. Während bekannte Problemsituationen durch Rückgriff auf in der Vergangenheit erfolgreiche Handlungsprogramme lösbar sind, muss in unbekannten Situationen eine geeignete Vorgehensweise fallweise entwickelt werden [Gait83, 206]. Einige Beispiele der zahlreichen, im Zuge einer Analyse zu treffenden Entscheidungen sind im Folgenden für wichtige Aufgaben aufgeführt [NeKn15, 159-165]:

- **Spezifikation des Analyseproblems**: Wie kann das Sachproblem operationalisiert werden? Welche Fragestellung soll anhand welcher Daten durch eine Analyse beantwortet werden?

Anordnung von drei Aufgaben mit insgesamt 360 Ausprägungen ergeben sich $360 \times 359 \times 358 = 46.267.920$ Möglichkeiten der unwiederholten Reihung.

- **Datenselektion**: Aus welchen Quellen sollen welche Daten anhand welcher Kriterien selektiert werden? Wie beeinflussen diese Entscheidungen die Analyseergebnisse?
- **Datenexploration**: Welche Voruntersuchungen sind hilfreich, welche verzichtbar? Ab wann kann von fehlerfreien Daten im Sinne der geplanten Analyse ausgegangen werden?
- **Datenmodifikation**: Welche Transformationen sind nötig? Wie beeinflussen die Maßnahmen die Analyse? Wie können Verfälschungen der Ergebnisse vermieden werden?
- **Datenanalyse**: Welcher Analyseansatz und welches Verfahren soll mit welchen Parameterwerten eingesetzt werden? Wie kann die Güte des Modells gemessen und verbessert werden?
- **Interpretation**: Wie kann die Relevanz und Signifikanz der Ergebnisse zuverlässig bewertet, wie ihre Aussagen und Implikationen zuverlässig ergründet werden?
- **Anwendung des Wissens**: Welche Handlungs- und Entscheidungskonsequenzen sind aus den Ergebnissen zu ziehen? Wie kann das Risiko einer Fehlentscheidung minimiert werden?

3.2.3 Komplexitätsgrade von Datenanalyseprozessen

Nicht alle Analyseprozesse sind gleichermaßen komplex. Liegt ein wohl definiertes, stabiles Problem vor, das sich mit wenigen Abweichungen gemäß einem vorgegebenen Plan lösen lässt, ist von geringer Analysekomplexität auszugehen. Aufgrund überschaubarer Interdependenzen sind mühelos Prozessaktivitäten identifizierbar, denen leicht passende Verfahren und Aufgabenträger zugeordnet werden können. Solche Prozesse sind routinemäßig ausführbar, programmierbar oder standardisierbar. In der Datenanalyse gilt dies im Allgemeinen für das Standardreporting und den Daten-/Dokumentenabruf, für einfachere modell- und methodengestützte Analysen sowie für einfache schließende Analysen. Komplexe Prozesse entstehen in Fällen mit instabiler, sich dynamisch entwickelnder Problemdefinition, die hohen Anpassungs- und Innovationsdruck schafft. Verknüpfte Teilprobleme

und zahlreiche Interdependenzen begründen hohen Koordinationsbedarf und erschweren die Aufgabenzerlegung. Die Problemlösung erfolgt innovativ und ist mit Standardprozessen nicht erreichbar [Gait83, 151f., 209]. Analysen mit Navigations- und Entdeckungsschwerpunkt (z.B. umfassende OLAP-Untersuchungen, EDA, Data Mining) sowie fortgeschrittene statistische Modelle und Methoden, Prognose- und Simulationsaufgaben sind typische Beispiele für komplexe Analyseprozesse.

3.2.4 Handhabung der Analysekomplexität

Angesichts der bei Durchführung von Datenanalysen zu erwartenden Schwierigkeiten erscheint es geboten, den Strukturiertheitsgrad der Analyseaufgabe zu erhöhen bzw. die Analysekomplexität zu verringern. Hierzu wird ein mehrstufiger Lösungsansatz verfolgt. Zunächst wird die Möglichkeit geprüft, die Analysekomplexität komplett zu umgehen (Abschnitt 3.2.4.1). Gelingt dies nicht, sind komplexitätsreduzierende Maßnahmen zu ergreifen (Abschnitt 3.2.4.2), die anschließend um Ansätze zur Bewältigung der unvermeidlichen Restkomplexität ergänzt werden (Abschnitt 3.2.4.3). Eine wirksame Lösung besteht letztlich in der Summe mehrerer Einzelmaßnahmen. Ihnen sind jeweils exemplarische Beiträge aus der Literatur zugeordnet. Ein tabellarischer Überblick, der den Ansätzen allgemeine Strategien der Komplexitätsbewältigung zuordnet, findet sich in Anhang A2. Die hier beschriebenen Maßnahmen fundieren und ergänzen die in Abschnitt 3.2.1 diskutierten Erfolgskriterien der Analysepraxis aus theoretischer Sicht.

3.2.4.1 Umgehung von Analysekomplexität

Als Optionen zur Umgehung der Analysekomplexität sind folgende Maßnahmen zu diskutieren:[46]

[46] Die Maßnahmen werden zur einfachen Referenzierung durch ein Kürzel gekennzeichnet, das jeweils aus dem Anfangsbuchstaben der Hauptkategorie der Komplexitätshandhabung (U: Umgehung, R: Reduzierung, B: Bewältigung) und einer fortlaufenden Nummerierung besteht.

Verzicht auf Datenanalysen (U1)

Ein gänzlicher Verzicht auf eine Datenanalyse ist angeraten, wenn die Lösung des Sachproblems dadurch nicht wirksam unterstützt werden kann. Dieser triviale Fall wird nicht weiter betrachtet.

Rückgriff auf vorhandene Informationen bzw. Analyseergebnisse (U2)

Wenn Datenanalyse der Gewinnung entscheidungsrelevanter Informationen dient, liegt ein Verzicht auf Datenanalyse zunächst auch dann nahe, wenn die geforderten Informationen bereits in abrufbarer Form vorliegen. Da der Abruf von Daten bzw. Informationen (Data Access, Information Retrieval) in Abschnitt 2.1.4 der Datenanalyse i.w.S. zugerechnet wird, bleibt dennoch eine – gleichwohl einfache – Datenanalyseaufgabe auszuführen. Als Begründung dient der zugrunde gelegte Informationsbegriff, der stets einen Interpretationsvorgang zur Bewertung der Relevanz gelieferter Daten im situativen Kontext verlangt (Abschnitt 2.1.2.4). Dies gilt gerade auch für den Abruf gespeicherter Ergebnisse früherer Datenanalysen. Da der Problemkontext, in dem sie erzeugt wurden, vermutlich vom aktuellen divergiert, sind diese Ergebnisse zunächst als Daten zu betrachten, deren Informationscharakter für die neue Situation erst festzustellen ist. Der Rückgriff auf vorhandene Dokumente oder frühere Analyseergebnisse führt demnach nicht zur Umgehung, jedoch zur wirksamen Reduzierung von Analysekomplexität.[47] Die Bereitstellung von Analyseergebnissen in einem Data Warehouse diskutieren u.a. [ImMa96], [Chen01], [BöKU03, 76]. Einen Überblick über Konzepte und Techniken des Information Retrieval vermitteln z.B. [BaRi11].

Vollautomatisierung von Datenanalysen (U3)

Die Automatisierung des gesamten Analysevorhabens von der Konzeption bis zur Realisierung würde die Analysekomplexität bis auf die Formulierung der zu beantwortenden Frage eliminieren. Dies erscheint

[47] Es zeigt sich, dass die vertretene Auffassung von Information bzw. Datenanalyse große praktische Relevanz besitzt, da in der Realität jede Informationsbeschaffung mit gewissem Aufwand verbunden ist und stets im aktuellen Problemkontext erfolgt.

insoweit utopisch, als Anwendungssysteme mit der Fähigkeit, alle Belange einer Untersuchung autonom zu bewältigen und gleichzeitig neue oder veränderte Fragestellungen handhaben zu können, trotz großer Fortschritte mit automatisierten Prognosesystemen [Dhar13, 72f.] nicht in Sicht sind. Die Mehrheit (28%) der Teilnehmer an einer nicht repräsentativen Umfrage unter Analytikern vom Mai 2015 kommt zu dem Schluss, dass die Automatisierung von Data-Science-Aufgaben auf Expertenniveau frühestens in 5-10 Jahren möglich ist, die zweitgrößte Gruppe (18,8%) ist der Ansicht, dass dies nie geschehen wird [KDnu15]. Zwar bleibt auch unter den Teilnehmern unklar, was eine Analyse auf Expertenniveau konkret kennzeichnet, jedoch ist zu vermuten, dass die Problemspezifikation nicht-automatisiert bleibt. Ein wissensbasiertes System zum „Invisible Data Mining" von [Hogl03] ist in der Lage, auf gemäß einer Fragegrammatik formulierte Anfragen passende natürlichsprachige Antworten zu liefern, ist jedoch auf eine kleine Menge gegebener Algorithmen und Datenbestände beschränkt.

3.2.4.2 Reduzierung von Analysekomplexität

Die Maßnahmen zur Komplexitätsreduktion und -bewältigung werden jeweils zunächst in Bezug auf das zu handhabende Prozesssystem und anschließend auf die vom Analytiker auszuführenden Tätigkeiten diskutiert. Aufgrund von Interdependenzen sind Querverweise zwischen einzelnen Maßnahmen nicht zu vermeiden.

Begrenzung des Analyseraums bzw. Verkürzung des Analysepfads (R1)

Folgende Maßnahmen eignen sich zur Komplexitätsreduzierung auf Zielebene:

- **Zielorientierung durch klare Problemspezifikation (R1.1)**: Unklar formulierte Analyseprobleme subsumieren gewissermaßen eine ganze Klasse von Einzelproblemen. Eine präzise Problemspezifikation unterstützt den Analytiker dabei, sich auf ein Analyseziel zu konzentrieren und dieses konsequent zu verfolgen [AmCo94, 46], [DeHa01, 257]. Wichtige theoretische Grundlagen zu Problemspezifikation und Operationalisierung liefert die empirische Sozialforschung [Drei94, 66ff.], [Diek07, 186ff.]. Beiträge anderer Disziplinen

verweisen zwar auf die Notwendigkeit dieser Aufgabe, liefern jedoch kaum konkrete Lösungsvorschläge (z.B. [WSG+97, 246], [Enge99, 41], [CCK+00, 16-18], [EMC15, 29]).

- **Prozessabspaltung durch Zieldifferenzierung (R1.2)**: Dynamisch auftretende, neue Analyseziele verlängern bzw. verändern den aktuellen Analysepfad und sollten nur bei hoher Sachproblemrelevanz direkt verfolgt werden [Bend02, 18f.]. Weitere Ziele können registriert und später wieder aufgenommen werden. Eine Methodik, die eine Begrenzung des Analyseraums durch rigorose Zielorientierung unterstützt, ist nicht bekannt.
- **Methodisch gestützte Datenauswahl und -selektion (R1.3)**: Berücksichtigt die Problemspezifikation auch den Datenbedarf für die Analyse, so kann eine Komplexitätsreduktion auch hinsichtlich des Datenraums erreicht werden [FSWS97, 7]. Außer Ansätzen zur Informationsbedarfsanalyse [GoRi09, 79ff.] sind keine Beiträge bekannt, welche die problemorientierte Datenauswahl thematisieren. Analog zu R1.1 wird die Aufgabe der Extraktion von Daten zwar genannt, weiterhin aber von deren Verfügbarkeit ausgegangen (vgl. z.B. [Enge99, 50], [CCK+00, 20]).

Prozessverkürzung (R2)

Die Begrenzung des Prozesssystems kann ebenso auf Aufgabenebene erfolgen:

- **Abspaltung analyseunabhängiger Aktivitäten (R2.1)**: Die Datenmodifikation wird in Abschnitt 3.1.2.2 in analyseunabhängige (die inhärente Datenqualität betreffende) und analysespezifische (die Datenrepräsentation betreffende) Aufgaben differenziert. Gelingt es, erstere auszugliedern und die Analyse auf bereinigten, konsolidierten Datenbeständen zu betreiben, nehmen Umfang und Komplexität des Analyseprozesses stark ab [Müll00, 203]. Die Nutzung von Data Warehouses zur Bereitstellung bereinigter Analysedaten kann in vielen Bereichen als Standard gelten [Uthu96, 565], ist jedoch nicht in allen Fällen möglich.

- **Zugriff auf spezielle Analysedatenbanken (R2.2)**: Bei Nutzung einer Analysedatenbank (Data Mart) [BaGü13, 67f.], die auf bestimmte Klassen von Auswertungen ausgerichtet ist, kann zusätzlich ein Teil der analysespezifischen Transformationen abgespalten werden. Die inhaltliche Fokussierung des Data Marts begrenzt zugleich die Komplexität des Datenraums und damit auch die Handlungsoptionen des Analytikers. Diesem komplexitätsreduzierenden Effekt steht die geringere Flexibilität bezüglich der Analyseziele gegenüber [NeKn15, 168f.].
- **Vermeidung unnötiger Aktivitäten und Iterationen (R2.3)**: Diese häufig infolge von Fehlern oder Informationsmangel auftretenden Probleme lassen sich teilweise durch Bereitstellung von bewährten Praktiken, Heuristiken oder Erfahrungswissen vermeiden, die in geeigneter Form für den Analytiker hinterlegt werden können, etwa in Form der in Abschnitt 3.1.1 diskutierten Prozessmodelle. Diese sehen z.B. eine Datenexploration vor, um frühzeitig Informationen zur Datenqualität zu erhalten. Verbesserte Kenntnisse über das Analyseproblem vermittelt z.B. auch der automatisierte Ansatz zur Datencharakterisierung von [EnTh98], [EnTh98b].
- **Wiederverwendung von Zwischenergebnissen (R2.4)**: Die (wiederholte) Durchführung bestimmter Aktivitäten lässt sich einsparen, wenn die von ihnen erzeugten (Zwischen-) Ergebnisse, wie z.B. transformierte Daten oder Modelle, gespeichert und wiederverwendet werden können. Eine derart systematische Verwaltung von Zwischenergebnissen ermöglicht der objektorientierte KDD-Prototyp CITRUS [BrSW97]. Das verbreitete Werkzeug KNIME unterstützt die Wiederverwendung der Outputs einzelner Aktivitäten [Bert+09, 26].
- **Unterstützung durch integriertes Analysewerkzeug (R2.5)**: Softwarewerkzeuge, die den gesamten Analyseprozess abdecken, machen Datentransformationen überflüssig, die bei Nutzung mehrerer heterogener Systeme häufig zur Anpassung der Datenrepräsentation an werkzeugspezifische Anforderungen nötig werden. Die integrierte Benutzeroberfläche solcher Systeme verbirgt zugleich die Heterogenität der implementierten Verfahren und erleichtert dem

Analytiker ihre Anwendung. Diese Eigenschaften treffen auf viele praxisrelevante Werkzeuge zu [Gros09b, 6], [NeKn15, 87-90].

Reduzierung von Planungsaufwand (R3)

Gelingt es, bestimmte Planungsaufgaben zu eliminieren, lässt sich die Problemkomplexität des Analytikers wirksam reduzieren. Hierzu sind folgenden Maßnahmen geeignet:

- **Wiederverwendung von Prozessplänen, Wiederholung von Prozessabläufen (R3.1)**: Die Neuplanung eines Prozesses kann vollständig unterbleiben, sofern eine Wiederverwendung existierender Prozessschemata möglich ist. Als Idealfall gilt die unmodifizierte Wiederholung eines bereits ausgeführten Prozesses; in der Regel ist jedoch mit Änderungsbedarf zur Anpassung an situative Gegebenheiten zu rechnen. Ist die vollständige Wiederverwendung nicht möglich, kommt eventuell die wiederholte Nutzung von Teilprozessen in Frage. Einen ganzheitlichen Ansatz zur vollständigen oder teilweisen Wiederverwendung von Data-Mining-Prozessen präsentiert [Enge99]. Das Forschungsprojekt MINING MART [MoSE03] entwickelte ein CBR-System[48] zur Nutzung bestehender Prozesse der Datenvorbereitung. Eine einfache Speicherung von Prozessmodulen unterstützen auch mehrere kommerzielle Werkzeuge [Bert+09], [Rapi10].
- **Vereinheitlichung und Standardisierung von Prozessen (R3.2)**: Treten ähnliche Analyseprobleme mehrfach wiederholt auf, lohnt unter Umständen die Entwicklung eines Standardprozesses, der an definierten Stellen fallspezifisch angepasst werden kann.
- **Bereitstellung vereinheitlichter oder abstrakter Prozesspläne (Schablonen) (R3.3)**: Werden wiederverwendbare oder standardisierte (Teil-) Prozesse abstrahiert, so entstehen generische Prozesspläne in Form von Schablonen, die in mehreren Situationen einsetzbar sind [Knob03a, 349]. Aufgrund der notwendigen fallspezifischen Konkretisierung können sie jedoch nur als Grundlage oder Orientierung

[48] CBR steht für Case-based Reasoning (Fallbasiertes Schließen).

bei der Planung dienen. Schablonen sind Bestandteil der auf KDD gerichteten Ansätze von [Enge99], [KSBF10] und [WeRü11].

(Teil-) Automatisierung der Prozessplanung bzw. -ausführung (R4)

Angesichts der üblichen Werkzeugunterstützung der Datenanalyse erscheint die Automatisierung ihrer Planung und Ausführung erstrebenswert. Bei der Prozessausführung ist die vollständige Automatisierung nur für klar spezifizierte Analysen mit hohem Wiederholungsgrad erreichbar. Weit verbreitet sind integrierte Inferenzsysteme, die bestehende Modelle auf neue Fälle anwenden und fortlaufend aktualisieren, etwa als Empfehlungssysteme in Online-Shops oder als Scoring-Systeme, die Geschäftstransaktionen wie z.B. Kreditkartenzahlungen während ihrer Abwicklung bewerten [NeKn15, 169]. Daneben sind bestimmte Teilprozesse, etwa zur Datenvorbereitung bei bekanntem Transformationsbedarf, oft leicht automatisierbar [SiLK95, 284]. Mittlerweile existieren Systeme, die Transformationsanforderungen selbständig erkennen und entsprechend erfüllen [Neck07]. Die automatisierte Analysedurchführung stellt ein nützliches Instrument zur Steigerung von Zuverlässigkeit und Effizienz der Untersuchungen dar. Die Prozesssteuerung wird auf das Anwendungssystem übertragen, die Planungskomplexität bleibt jedoch bestehen und kann sogar höher ausfallen als bei personeller Analysedurchführung. Weil manuelle Eingriffe während des Ablaufs nicht möglich sind, müssen alle Ablaufvarianten und -ausnahmen im Prozessplan abgebildet sein und erfordern daher oft detailliertere Vorgaben als bei interaktiver Abarbeitung.

Somit gewinnt die automatisierte Analyseplanung an Bedeutung. Sie verspricht eine wirksamere Reduzierung der Problemkomplexität sowie die Berücksichtigung einer größeren Anzahl von Alternativen, die umfassendere Prüfung der Einhaltung von Integritätsbedingungen sowie die verbesserte Dokumentation der Prozesse als manuelle Planung [LaPh94, 23]. Realistisch ist aktuell nur die Planung einzelner Prozessabschnitte. Hierzu finden sich zahlreiche Beiträge in der Literatur, die sich auf die Aktionsplanung der Künstlichen Intelligenz [LaPh94], [ZLKO97], CBR [MoSE03] oder Ontologien [DiPS09], [ZPZL09] stützen und die Planung von Aktivitätsfolgen, die Verfahrensauswahl

oder -instanziierung betreffen. Nur in Einzelfällen (z.B. [KSBF10]) haben diese Forschungsprototypen jedoch Eingang in kommerzielle Systeme gefunden.

3.2.4.3 Bewältigung von Analysekomplexität

Die Bewältigung der verbleibenden Prozesskomplexität kann im Wesentlichen durch drei Klassen von Maßnahmen gelingen.

Ordnung des Prozesssystems (B1)

Zur besseren Übersicht der großen Menge zu handhabender Elemente kann eine systematische Ordnung des Prozesssystems hilfreich sein, die in folgenden Formen möglich ist:

- **Strukturierung und Modularisierung (B1.1)**: Eine Strukturierung der Aktivitäten geben die Prozessmodelle aus Abschnitt 3.1.1 vor. Ein entsprechender Vorschlag für Analyseziele oder Sachprobleme ist nicht bekannt. Eine Modularisierung ist stets mit wiederverwendbaren Prozesseinheiten und Schablonen gemäß R3.1 und R3.3 verbunden.
- **Ebenen- und Sichtenbildung, Hierarchisierung (B1.2):** Ein wirksames Instrument der Ordnung ist die Einteilung in Ebenen, die das System jeweils aus einem anderen Blickwinkel vollständig beschreiben. Auf jeder Ebene können Sichten definiert werden, die jeweils nur bestimmte Typen von Systemelementen beschreiben und die Bewältigung der typmäßigen Komplexität[49] unterstützen [Sinz95, 3-5]. So muss der Analytiker jeweils nur jene Systemkomponenten handhaben, die den gerade bearbeiteten Prozessaspekt repräsentieren. CRISP-DM [CCK+00] ist in vier Ebenen (Phasen des Prozessmodells, generische Aufgaben, fallspezifisch spezialisierte Aufgaben, Vorgänge) gegliedert, sieht jedoch keine Sichten vor. Die in Abschnitt 2.3.1.4 zusätzlich zur hier betrachteten Aufgabenebene identifizierten Ziel- und Ressourcenebenen werden

[49] Die typmäßige Komplexität resultiert aus der Artenvielfalt der Systemkomponenten [Sinz95, 5].

von keinem bekannten Ansatz explizit berücksichtigt. Analysewerkzeuge sehen typischerweise verschiedene *views* für Aktivitäten und Operatoren vor, die jedoch eher im Sinne von Ebenen zu verstehen sind. Die hierarchische Zerlegung von Aufgaben in ausführbare Aktivitäten unterstützen z.B. die Ansätze von [WiRe96], [AmCo98], [Enge99, 51f.], [KSBF09].

- **Taxonomien von Aufgaben und Verfahren (B1.3)**: Eine hierarchische Ordnung ist auch in Bezug auf die Prozessbausteine möglich. Die Organisation von Aufgaben und Verfahren, z.B. nach ihren Sachzielen, erleichtert deren Auswahl [Knob03a, 346]. Die meisten Analysewerkzeuge organisieren Operatoren in Form von einfachen Taxonomien. Umfassende Ontologien zur Beschreibung von Prozessbausteinen, die aber mit wenigen Ausnahmen (z.B. [Hila+11]) der automatisierten Verarbeitung dienen, sind für KDD entstanden. Ein Überblick erfolgt in Abschnitt 4.7.1.5.

Methodische Unterstützung des Analytikers (B2)

Die Handhabbarkeit komplexer Systeme hängt unmittelbar von der Verfügbarkeit einer adäquaten Problemlösemethodik ab [Mali00, 239]. Eine effektive methodische Unterstützung sollte auf die Strukturierung des Prozesssystems abgestimmt sein, indem für die einzelnen Ebenen und Sichten jeweils spezialisierte Lösungsmechanismen verfügbar sind.

- **Problemspezifikation (B2.1)**: Neben der Begrenzung des Analyseraums (R1.1) unterstützt die Problemspezifikation insbesondere ein systematisches Vorgehen, das die Voraussetzung für die effektive Erreichung gesetzter Ziele ist.
- **Vorgabe von Prozessschemata (B2.2)**: Auf Aufgabenebene sind Prozessmodelle zur Orientierung des Handelns geeignet (R3.3 und B1.1). Da die konkrete Aktivitätenfolge häufig situativ bestimmt wird, dienen sie in erster Linie als Checkliste für in einzelnen Prozessphasen wichtige Aufgaben [Knob03a, 346]. Stärker auf einen Kontext ausgerichtet sind wiederverwendbare Schemata aus ähnlichen Projekten (R3.1, R3.2), bei denen auch die Reihenfolge der Aktivitäten von größerem Interesse ist.

- **Flexibilisierung des Vorgehens durch evolutionäre Methodik (B2.3)**: Zuweilen kann auf eine Detailstrukturierung von Prozessen verzichtet werden, etwa weil sie einer internen Gliederung nicht zugänglich sind (z.B. OLAP-Navigation) [KnWe00, 359], oder weil die bewusste Öffnung des Vorgehens [Mali00, 265] der iterativ-inkrementellen Natur mancher Datenanalysen besser gerecht wird. Handlungspläne mit gezielt gesetzten Freiheitsgraden verlagern Teile der Planungs- und Steuerungsaufgaben auf spätere Zeitpunkte. In besser strukturierten Fällen kann Flexibilität durch kontextabhängige Konditionalisierung des Ablaufs erreicht werden [Gait83, 182, 216]. Abstrakte Prozessschemata (R3.3), Modularisierung (B1.1) und Hierarchisierung (B.1.2) schaffen die Grundlage für ein evolutionäres Problemlöseverhalten, das durch situative Handhabung von Teilproblemen die integrale Lösung des Gesamtproblems erlaubt [Mali00, 265]. Es sollte durch eine praktikable evolutionäre Methodik unterstützt werden. [DeHa01, 65] präsentieren ein einfaches Prototyping-Modell für Data Mining. [MSMF09] erweitern CRISP-DM um Standards des Software-Engineering und erlauben die Auswahl verschiedener flexibler Lebenszyklusmodelle. Zahlreiche, dem Analyseprozess zur Seite gestellte Managementprozesse machen den Vorschlag jedoch derart unübersichtlich, dass er für einen Einsatz in der Praxis kaum tauglich erscheint.

Bereits TUKEY fordert, grundlegende Denkweisen, Erfahrungen und bewährte Praktiken erfolgreicher Analysevorhaben zu kommunizieren [Tuke62, 6]. Dies legt eine wissensbasierte Unterstützung des Analytikers nahe [EnMT95, 12]:

- **Bereitstellung von Erfahrungs- und Domänenwissen (B2.4)**: Erfahrungswissen, z.B. über Analyseverfahren, die Wirkung von Verfahrensparametern oder häufige Fehlerursachen, verbessert Effektivität und Effizienz des Prozesses und reduziert mit der Ungewissheit auch die Handlungskomplexität [DeHa01, 229]. Ebenso wichtig ist eine breite Basis an Domänenwissen, um den Aussagegehalt der Daten und Ergebnisse sowie Projektrisiken korrekt einschätzen zu können [Bend02, 9]. Einen praxisorientierten Ansatz zum Wissensmanagement in KDD-Projekten präsentieren

[BaRi00]. Einen kollaborativen Ansatz zur Bereitstellung von konkreten Beschreibungen ausgeführter Prozesse in einem Portal verfolgen [VaBl09].

- **Bereitstellung von Handlungsempfehlungen und Best Practices (B2.5)**: Ein höherer Grad der Unterstützung wird erreicht, wenn die Erfahrungen in Form konkreter Handlungsempfehlungen und Heuristiken bereitgestellt werden. Hilfreich sind stets auch aufgabenbezogene bewährte Praktiken [Mali00, 284f.], [DeHa01, 63], [Knob03a, 349]. Vorschläge hierfür liefert z.B. PYLE, der umfangreiche Empfehlungen zu Prozessgestaltung und Verfahrensanwendung strukturiert nach Aktivitäten bereitstellt[50] [Pyle03].
- **Zugriff auf kontextabhängiges Wissen (B2.6)**: Die Anwendbarkeit der Wissenselemente kann gesteigert und für den Analytiker besser beurteilt werden, wenn sie kontextabhängig formuliert und bereitgestellt werden.

Werkzeugbasierte Unterstützung des Analytikers (B3)

Die bereits in Abschnitt 3.2.1 empfohlene Werkzeugunterstützung des Analytikers kann durch folgende Maßnahmen konkretisiert werden:

- **Werkzeugunterstützung der Problemlösemethodik (B3.1)**: Die Vorteile der in B2.3 geforderten Problemlösemethodik lassen sich nur dann ausschöpfen, wenn diese durch ein Analysewerkzeug vollständig unterstützt wird. Hier besteht insgesamt Verbesserungspotenzial, da selbst einfache Prozessmodelle kaum von Werkzeugen umgesetzt werden.[51]
- **Automatisierung von Gestaltungs-, Lenkungs- und Managementaufgaben (B3.2)**: Zur Automatisierung der Planung siehe R4. Die Unterstützung der Prozesslenkung im Sinne eines Workflow-Management-Systems bieten im Wesentlichen alle prozessorientierten Mehrzweck-Analysewerkzeuge [Gros09b, 6]. Funk-

[50] Er dokumentiert die Hinweise in so genannten „action boxes“ und „technical boxes“.

[51] CRISP-DM wird trotz seiner Popularität explizit nur vom IBM SPSS Modeler unterstützt, SAS-Werkzeuge realisieren das herstellereigene SEMMA-Prozessmodell.

tionen zur Unterstützung des Projekt- oder Wissensmanagements (B2.4-B2.6) sind hingegen nicht bekannt.

- **Assistenz- und Beratungsfunktion (B3.3)**: Da die Automatisierung bestimmter Gestaltungsaufgaben nicht realistisch ist, kann das Analysewerkzeug eine Assistentenrolle übernehmen oder den Anwender beratend unterstützen. Solche Systeme haben eine lange Tradition in der Datenanalyse und wurden z.B. für die Statistik [Haux86], [HoKl91], für die Auswertung von Marktforschungsdaten [GaSc94], zur Verfahrensauswahl im Maschinellen Lernen [CSG+92], [LiSt99], [Gira05] oder zur interaktiven Prozessplanung [Enge96], [BePH05] entwickelt.

3.3 Ein Vorgehensmodell für die Datenanalyse

Wie obige Diskussion zeigt, stellt die methodische Unterstützung einen wichtigen Erfolgsfaktor für die Datenanalyse dar. Im vorliegenden Kapitel wird auf Grundlage der bisherigen Erkenntnisse ein evolutionäres Vorgehensmodell (B2.3) entwickelt. Hierzu untersucht Abschnitt 3.3.1 erprobte Methoden der evolutionären Problemlösung. Abschnitt 3.3.2 führt zwei Betrachtungsebenen ein. Das resultierende Vorgehensmodell wird in Abschnitt 3.3.3 erläutert.

3.3.1 Evolutionäre Entwicklung von Analyseergebnissen

Aufgrund der Prozesskomplexität sind viele der im Zuge einer Datenanalyse zu treffenden Gestaltungs- und Lenkungsentscheidungen im Hinblick auf ihre genauen Anforderungen unklar und bezüglich ihrer Wirkungen nicht vollständig abschätzbar [WSG+97, 245], [KrWZ98, 30]. Solche Probleme lassen sich durch evolutionäre Versuchs-Irrtums-Prozesse lösen, wie sie in Abschnitt 3.1.3 beschrieben sind [Mali00, 260f, 268f.]. Wesentliche Stärke des evolutionären Vorgehens ist die Möglichkeit, zunächst mit einer groben Problemspezifikation zu beginnen und diese schrittweise fortzuentwickeln. Das Wissen über das zu lösende Problem wächst zusammen mit der Lösung im Verlauf des Prozesses. Evolutionäre Ansätze sind starren kaskadischen Modellen in komplexen Situationen daher oft überlegen

[Gait83, 212], [Mali00, 281], [Somm01, 59].[52] Anstatt das Vorgehen bei der Datenanalyse „kochbuchartig" als wenig realitätsnahe Folge linearer Schritte zu definieren erscheint es daher folgerichtig, Versuchs-Irrtums-Prozesse durch bewusst geschaffene Freiheitsgrade möglichst effizient zu gestalten [Drei94, 1], [Mali00, 270]. Hierzu wird im Folgenden geprüft, inwiefern sich evolutionäre Problemlöse- und Entwicklungsmethoden aus der Softwaretechnik für die Datenanalyse eignen.

3.3.1.1 Prototyping

Eine Methode, welche den skizzierten Anforderungen genügt, ist das Rapid Prototyping. Gemäß dieser Methode ablaufende Prozesse sind zieloffen, iterativ und inkrementell sowie bezüglich Inhalt und Reihenfolge der Aktivitäten flexibel [DeHa01, 65]. Prototyping beruht auf der wiederholten Entwicklung, Validierung und Revision verschiedener Versionen der zu erstellenden Leistung (*Prototypen*) [FlZü97, 660]. Prototypen unterstützen insbesondere die Problemspezifikation und die frühzeitige Erkennung von Projektrisiken, da fehlende, neue bzw. nicht erfüllbare Anforderungen während des Experimentierens mit vorläufigen Lösungen leicht erkannt werden können. Beim Rapid Prototyping kommt der Geschwindigkeit besondere Bedeutung zu. Daher sind die Prozessphasen Spezifikation, Entwicklung und Validierung zeitlich und inhaltlich derart miteinander verwoben, dass relativ schnell eine erste (vorläufige) Lösung präsentiert werden kann. Diese wird bewertet und über mehrere Versionen hinweg solange modifiziert, bis ein zufriedenstellendes Ergebnis vorliegt [Somm01, 56-59, 181f.] (Abbildung 18).

Beim hier beschriebenen evolutionären Prototyping wird der erste Prototyp schrittweise in die endgültige Lösung überführt [Somm01, 184].[53] Dieses Vorgehen ist in der Datenanalyse bei der iterativen

[52] Die Effektivität des evolutionären Problemlösens ist durch Studien belegt, die zeigen, dass sich in allen Fällen nach relativ geringer Anzahl von Versuchen erfolgreiche Lösungen ergeben [Mali00, 281].

[53] Beim alternativen Wegwerf-Prototyping dient der Prototyp nur der Sammlung von Informationen über das Problem und wird anschließend verworfen [Somm01, 184].

Modellerstellung (vgl. Abschnitt 3.1.2.3) üblich, um über zunehmend bessere Zwischenversionen zu einem nützlichen Ergebnis zu gelangen [DeHa01, 64f.], [MWK+06].

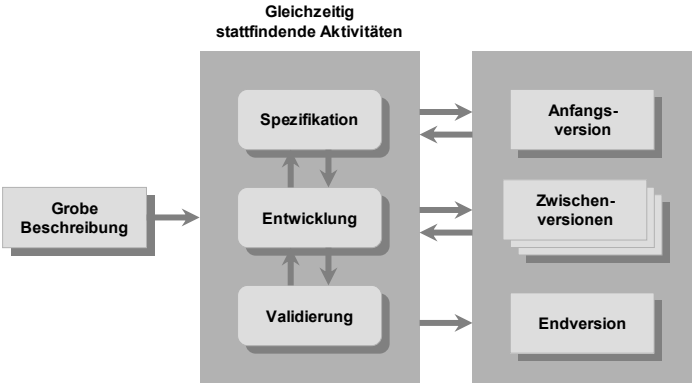


Abbildung 18: Evolutionäre Entwicklung von Problemlösungen in mehreren Versionen [Somm01, 59]

Aus Sicht des Projektmanagements besitzt das evolutionäre Vorgehen eine gravierende Schwäche: Aufgrund des unstrukturierten Ablaufs wird das Prozessschema gewissermaßen unsichtbar (mangelnde Prozessstrukturtransparenz), was Orientierung und Zeitplanung erschwert [DeHa01, 65]. Die Fortschrittsmessung gestaltet sich schwierig, da Zwischenversionen häufig nicht dokumentiert werden.[54] Diese Faktoren können Effizienz und Zielorientierung des Prozesses gefährden. Daher wird empfohlen, rein evolutionäres Vorgehen nur für überschaubare Projekte zu wählen und größere Vorhaben mit Ansätzen zu bewältigen, die die Vorzüge des kaskadischen und des evolutionären Vorgehens kombinieren [Somm01, 59f.]. Zwei grundlegende hybride Ansätze sind das inkrementelle und das spiralförmige Modell.

[54] Weitere Nachteile, die im Rahmen der evolutionären Softwareentwicklung auftreten, wie eine unklare Struktur der erstellten Softwaresysteme oder die Notwendigkeit des Einsatzes teurer, schulungsintensiver Werkzeuge zur Unterstützung des Prozesses [Somm01, 59f.], sind im Datenanalysekontext nicht im selben Maß zu erwarten.

3.3.1.2 Inkrementelles Vorgehensmodell

Beim inkrementellen Vorgehen nach MILLS ET AL. [MNL+80] wird die Prozessleistung in mehrere Arbeitspakete (Teilleistungen) aufgeteilt, die jeweils von einem allgemeinen Entwicklungsprozess bearbeitet werden, der somit insgesamt wiederholt durchlaufen wird. Die Teilleistungen werden einzeln erstellt und anschließend auf erforderliche *Verbesserungen* oder *Erweiterungen* hin untersucht. Solche Anforderungen führen ihrerseits zur Wiederholung bestimmter Aufgaben. Die endgültige Spezifikation liegt beim inkrementellen Ansatz erst mit der letzten Erweiterung vor. Zur Erstellung der einzelnen Arbeitspakete ist die Verwendung verschiedener Prozessmodelle erlaubt. Durch kaskadische Gliederung des Hauptprozesses soll der Ablauf übersichtlicher und einfacher zu handhaben sein als bei rein evolutionärem Vorgehen. Der Ansatz zielt insbesondere auf die Effizienz der Überarbeitung von Arbeitspaketen, indem weitere Anforderungen erst nach Vorliegen erster Ergebnisse berücksichtigt werden [Somm01, 63f., 188]. Im Kontext der Datenanalyse äußern sich Erweiterungen z.B. in Form modifizierter bzw. neuer Analyseziele. Verbesserungen können in Bezug auf Analysedaten, Modelle und Ergebnisse notwendig werden.

3.3.1.3 Spiralmodell

Beim Spiralmodell, für die Softwareentwicklung vorgeschlagen von BOEHM [Boeh88], bewegt sich die Erstellung der Prozessleistung ausgehend von einer ersten Groblösung hin zum endgültigen Ergebnis. Zur Beschreibung des Prozesses wird die Metapher einer Spirale gewählt, in der jede Windung eine *Entwicklungsstufe* der Prozessleistung hervorbringt. Feste Prozessphasen sind nicht vorgegeben, jede Windung ist aber in vier Segmente eingeteilt, die generische Schritte repräsentieren und allen Zyklen der Spirale gemein sind. Innerhalb der Segmente können jeweils beliebige andere Vorgehensmodelle zum Einsatz kommen. Gemeinhin gilt die explizite Berücksichtigung von Projektrisiken durch das hierfür reservierte zweite Segment als wichtiges Merkmal des Spiralmodells [Somm01, 63, 65-67]. Aus Sicht der Datenanalyse erscheint seine Generizität bemerkenswert. So können weitere Windungen sowohl durch die Glieder einer Analysekette auf

Zielebene motiviert sein, als auch infolge von Korrektur- oder Verbesserungsbedarf auf der Aufgabenebene eröffnet werden. Das Spiralmodell bietet große Flexibilität für die Ausgestaltung von Analyseprozessen und erscheint aus diesem Grund insbesondere für umfassende, explorative Untersuchungen geeignet.

3.3.1.4 Eignung für Datenanalyseprozesse

Die Literatur erachtet die iterativ-inkrementelle Vorgehensweise als notwendig und hinreichend zur effektiven Lösung komplexer Datenanalyseprobleme (vgl. Abschnitte 3.1.3 und 3.2.4.3). DELMATER & HANCOCK präsentieren ein Vorgehensmodell für Data Mining, das Prototyping zur Modellerstellung vorsieht [DeHa01, 64-72]. An der Universität Bamberg konnte der Nutzen evolutionärer Vorgehensmodelle für die Datenanalyse auch durch Erfahrungen aus Praxisprojekten belegt werden. So zeigt NECKEL [Neck02] anhand eines Beispiels aus dem Lebensmitteleinzelhandel, wie eine Kundensegmentierung mit Data-Mining-Methoden als Prototyping-Prozess beschrieben werden kann und kommt zu dem Schluss, dass brauchbare Ergebnisse einer Clusteranalyse nur durch wiederholte Erstellung und Evaluierung vorläufiger Ergebnisse zu erzielen sind [Neck02, 91]. SCHENNACH [Sche04] verknüpft die Ideen des Spiralmodells mit CRISP-DM und entwickelt ein KDD-Spiralmodell. Anhand eines Analyseprojekts eines Automobilherstellers wird die prinzipielle Anwendbarkeit des Modells demonstriert, das nach drei Zyklen ein zufrieden stellendes Ergebnis liefert [Sche04, 61ff].

Zur Entwicklung eines Vorgehensmodells für die evolutionäre Datenanalyse werden wesentliche Grundkonzepte der beiden beschriebenen Hybridmodelle herangezogen. Als Bezugsrahmen dient das allgemeine Prozessmodell aus Abschnitt 3.1.1. Die Idee der *Erweiterungen* aus dem inkrementellen Ansatz wird zur Kontrolle von Analyseketten (Zielebene) verwendet. Jedes Arbeitspaket entspricht einem Analyseziel, das entweder zu Projektbeginn vorgegeben oder in seinem Verlauf identifiziert wird. Für jedes Analyseziel wird ein Analyseprozess ausgeführt. Zur Lenkung der Prozesse (Aufgabenebene) dient das *Spiralmodell*. Als Segmente seiner Windungen dienen die

generischen Phasen Datenvorbereitung, Datenanalyse, Ergebnisaufbereitung. Jede Iteration im Prozess entspricht einer neuen Spiralwindung, bei der das Prozessschema jedoch nicht zwingend vollständig abzuarbeiten ist. Allerdings muss für jeden Analyseprozess (Analyseziel) insgesamt jede Phase des Prozessschemas mindestens einmal durchlaufen werden.

Die gewählte Verknüpfung des inkrementellen und des spiralförmigen Vorgehens bietet ein strukturiertes Handlungsschema und ermöglicht gleichzeitig die Nutzung der Stärken der Hybridmodelle. Hierbei ist es für den konkreten Ablauf unerheblich, ob eine neue Spiralwindung wegen einer Prozessiteration oder eines neuen Analyseziels eröffnet wird. Aufgrund der getrennten Betrachtung von Analysezielen werden jedoch die Ursachen der Zyklen nachvollziehbar und eine quantitative Zielkontrolle realisierbar. Durch die Möglichkeit zur frühzeitigen Erkennung nicht zielführender Anschlussuntersuchungen bleibt das Projektrisiko beherrschbar (qualitative Zielkontrolle; vgl. [Somm01, 64f.]).

3.3.2 Differenzierung zwischen Projekt- und Prozessebene

Bei Gegenüberstellung der Prozessmodelle in Abschnitt 3.1.1 fällt auf, dass nicht alle der untersuchten Vorschläge die Randphasen *Problemspezifikation* und *Anwendung des Wissens* in den Ablauf einbeziehen. Diese Feststellung verdient weitere Beachtung.

Genaue Kenntnis des zu lösenden Problems bzw. der zu erreichenden Projektziele ist unbestritten von zentraler Bedeutung für Projekte im Allgemeinen. Für evolutionäre Vorhaben gewinnt eine klare Vision der Projektergebnisse besondere Relevanz zur Sicherung der Zielorientierung (vgl. Abschnitte 3.2.1 und 3.2.4). Gemäß Abschnitt 3.1.2.1 erfolgt die Problemspezifikation ausgehend von einem Sachproblem der Anwendungsdomäne, aus dem mehrere Analyseprobleme resultieren können: Zum einen lassen sich die fachlichen Konstrukte des Sachproblems häufig zu mehreren empirische Begriffen operationalisieren, die zur Lösung des Sachproblems beitragen. Zum anderen führen situativ identifizierte Analyseziele zu weiteren Analyseproblemen. Die 1:n-Beziehung zwischen Sach- und Analyseproblemen legt eine Trennung der sachlichen von den analytischen Belangen nahe.

Somit kann der Tatsache Rechnung getragen werden, dass die Lösung eines Sachproblems mithilfe mehrerer unterschiedlicher Datenanalysen erarbeitet wird.

Diese Differenzierung ist ebenso für die Anwendung des Wissens hilfreich, die der Realisierung der Lösung des Sachproblems dient. Wie in Abschnitt 3.1.2.5 dargelegt, zählt hierzu zwar auch die Anwendung analytisch erzeugter Prognosemodelle, die ihrerseits eine Datenanalyse darstellt. Analyseergebnisse können aber ebenso in Entscheidungen und Handlungsmaßnahmen einfließen, die je nach Form der Wissensverwertung weit über datenanalytische Aspekte hinausgehen können. Die Betrachtung der Wissensanwendung auf Sachebene ermöglicht deren Berücksichtigung im Vorgehensmodell, ohne sie auf analytische Nutzungsformen einzuschränken. Aus ökonomischer Perspektive ist zu bedenken, dass Datenanalysen ebenso wie die Anwendung des Wissens Kosten verursachen. Dieser Aufwand ist regelmäßig nur dann zu rechtfertigen, wenn dem Gesamtprojekt messbare Erträge zuordenbar sind [AdZa96, 81], [BeLi97, 18], [DeHa01, 72]. Um die vielfach vernachlässigte Bewertung des Erfolgs analytisch unterstützter Projekte anzumahnen [Küpp99, 103], [Hanc12, 175], wird das Modell um eine Phase zur Evaluierung des gesamten Projekts ergänzt. Durch Gegenüberstellung der Projektziele mit dem tatsächlich realisierten Erfolg können Verbesserungspotenziale für künftige Vorhaben erkannt werden (vgl. B2.4).

Die Aufgaben Problemspezifikation, Wissensanwendung und Projektevaluierung nehmen nicht nur auf Datenanalyseprozesse, sondern auch auf Sachprobleme Bezug, weshalb sie nicht in allen Prozessmodellen auftreten. Sie betreffen das gesamte Projekt, innerhalb dessen eine oder mehrere Datenanalysen erfolgen. Aus diesem Grund wird im Folgenden zwischen analytisch zu unterstützenden Projekten (Analyseprojekte) und Analyseprozessen differenziert und das Vorgehensmodell in die *Projektebene* und die *Prozessebene* gegliedert (Abbildung 19).

Die Inhalte der Problemspezifikationsphase werden für die Belange der beiden Ebenen spezialisiert. Auf Projektebene erfolgt die umfassende *Planung des Analyseprojekts*, auf Prozessebene wird für jede Untersuchung eine *Planung der Analyse* durchgeführt. Die Phasen *Daten-*

vorbereitung, Datenanalyse und *Ergebnisaufbereitung* des allgemeinen Prozessmodells komplettieren die Prozessebene. Sie werden auf Projektebene zur Aufgabe *Durchführung der Analyse* aggregiert. Auf Projektebene wird die *Anwendung des Wissens* aus dem allgemeinen Modell (in der Abbildung unten) übernommen und die bislang nicht vorgesehene Aufgabe *Evaluierung des Analyseprojekts* ergänzt.

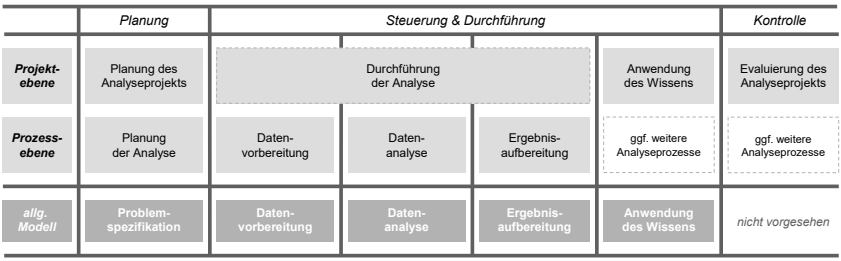


Abbildung 19: Betrachtungsebenen des Vorgehensmodells für die Datenanalyse (eigene Darstellung)

3.3.3 Die Phasen des Vorgehensmodells

Die Aufgabeninhalte der Phasen des allgemeinen Prozessmodells sind in Abschnitt 3.1.2 ausführlich beschrieben. Im Folgenden werden die vier Phasen des Vorgehensmodells aus Projektperspektive betrachtet und Bezüge zur Prozessebene erläutert. In der Durchführungsphase wird zudem auf den spiralförmigen Ablauf eingegangen. Die Darstellung gründet auf einem Handlungsschema für betriebswirtschaftliche Datenanalyseprojekte [Knob01, 102-108] und dem Regelkreismodell des Datenanalyseprozessmanagements aus Abschnitt 2.4.4. Die Korrespondenz der Phasen des Vorgehensmodells mit jenen des Regelkreises ist in Abbildung 19 (oben) dargestellt. Das Schema ist auf Projektebene kaskadisch, d.h., die Ergebnisse einer Phase determinieren die Inhalte der Folgephase.

3.3.3.1 Planung des Analyseprojekts

Datenanalyseprojekte unterscheiden sich nicht grundsätzlich von Projekten[55] in anderen Domänen [DeHa01, 257], sodass bezüglich des Projektmanagements auf die einschlägige Literatur verwiesen werden kann.[56] Inhaltliche Besonderheiten werden im Folgenden skizziert. Die Planung des Analyseprojekts umfasst die Aufgaben *Identifikation des Sachproblems*, *Domänenanalyse* und *Projektplanung* aus Abschnitt 3.1.2.1. Die Aufgaben *Spezifikation des Analyseproblems* und *Untersuchungsdesign* stellen die Verknüpfung mit der Prozessebene her.

Ausgangspunkt eines Analyseprojekts bildet stets die Beschreibung eines aus dem Sachproblem resultierenden fachlichen Handlungsbedarfs, die operationale betriebswirtschaftliche Ziel- und Erfolgskriterien beinhaltet und in der Domänenanalyse ermittelte Rahmenbedingungen der Diskurswelt darstellt [HiWi01, 22f.]. Die eigentliche Projektplanung behandelt organisatorische, rechtliche, technische und ökonomische Aspekte des Projektmanagements. So sind etwa die Zeit- und Ressourcenplanung auszuführen sowie Fragen zu Verantwortlichkeiten, zum Wiederholungsgrad, zu relevanten Rechtsnormen, zur technischen Realisierbarkeit und zur erwarteten Wirtschaftlichkeit des Vorhabens zu klären [Küpp99, 103], [Knob01, 105], [DeHa01, 257]. Diese Entscheidungen sind nicht ohne Kenntnis des Untersuchungsdesigns (Planung von Analyseketten und -prozessen) zu treffen, weshalb Interdependenzen zwischen Projekt- und Prozessebene bestehen. Zur Kontrolle von Projektrisiken ist die Aufnahme weiterer Analyseprobleme jeweils vor dem Hintergrund der Zeit- und Ressourcenrestriktionen des Gesamtprojekts zu prüfen (Zielkontrolle). Ihre Einbeziehung führt zu Erweiterungen im Sinne des inkrementellen Modells.

---

[55] Als ***Projekt*** gilt im Allgemeinen ein abgegrenztes Vorhaben, das durch klare Ziele (Projektaufgabe) definiert ist, unter bestimmten Ressourcenbeschränkungen im Rahmen einer projektspezifischen Organisation ausgeführt wird und im Hinblick auf die Gesamtheit der konkreten Bedingungen typischerweise einmaligen Charakter besitzt [Daen88, 122], [Sche96, 11], [PaRa14, 19f.].

[56] Vgl. z.B. [PaRa14] im Allgemeinen sowie [Schw06] (IT-Projekte) und [DeHa01, 257] (Datenanalyse).

In welchen konkreten Situationen die Datenanalyse die Erfüllung betrieblicher Aufgaben unterstützen und zur Lösung von Sachproblemen beitragen kann, ist vom Analytiker in Kooperation mit Domänenfachleuten zu entscheiden [Küpp99, 106], [FaPS96, 25]. Oftmals kann es sinnvoll sein, zur Problemerkennung ein eigenes Analyseprojekt zu initiieren, wodurch gewissermaßen eine *Schachtelung von Projekten* entsteht (vgl. [Mali00, 267f.]). Prinzipiell lassen sich auf allen Stufen von der Erkennung des Handlungsbedarfs bis zur Problembeschreibung unterstützende Datenanalysen zur Planung der vorgesehenen Untersuchung einsetzen.

3.3.3.2 Durchführung der Analyse gemäß dem Spiralmodell

Diese Phase wird durch die auf Prozessebene definierten Aufgaben ausgefüllt. Der Übergang von der Projekt- zur Prozessebene erfolgt, indem aus dem Sachproblem ein oder mehrere *Analyseprobleme abgeleitet* werden. Auf ihrer Grundlage wird das *Untersuchungsdesign* erstellt, das die Vorgehensweise für das Analyseprojekt definiert. Hierzu zählen Entscheidungen über die Verkettung mehrerer Analysen, die jeweils einzusetzenden Analyseansätze und Verfahren sowie die Auswahl geeigneter Analysedaten. Für jedes Analyseproblem wird ein Analyseprozess konzipiert und ausgeführt. Zur Realisierung der Untersuchung kommt grundsätzlich jede der in Abschnitt 2.2.2 genannten Funktionen in Frage, auf die der Prozess jeweils abgestimmt wird.

Um der häufig evolutionären Natur der Analysedurchführung gerecht zu werden, wird in dieser Phase ein *Datenanalyse-Spiralmodell* eingesetzt (Abbildung 20). Die Spirale wird von innen nach außen durchlaufen und ist in vier Segmente gegliedert, die den Phasen der Prozessebene entsprechen. Sie können fallspezifisch weiter zerlegt werden. Innerhalb eines Zyklus werden nach Maßgabe der Planung auf Projektebene (1) eine Analyse geplant, (2) Analysedaten selektiert und vorbereitet, (3) die Datenanalyse durchgeführt und (4) deren Ergebnisse aufbereitet und bewertet. Die innerhalb der Phasen auszuführenden Aufgaben und die eingesetzten Verfahren variieren mit dem Analyseproblem und dem gewählten Analyseansatz. Bei Bedarf können einzelne Aufgaben übersprungen oder wiederholt werden. Streng sequenzielle Prozessabläufe

umfassen nur eine Spiralwindung, die sich in diesem Fall zu einer linearen Schrittfolge glättet.

Am Ende jedes Zyklus wird über Notwendigkeit und Art der Fortsetzung der Analyse entschieden. Denkbar sind die Wiederholung der aktuellen Analyse zur Verbesserung der erzielten Ergebnisse, der Anschluss einer weiterführenden Analyse mit anderem Analyseziel, die Abspaltung von Teilprojekten durch Zieldifferenzierung (R1.2) und die Beendigung der Untersuchung wegen Erfolg oder Misserfolg. Auf diese Weise ist mit zunehmender Zyklenzahl eine Steigerung der Präzision der Analyseziele und der Qualität der Ergebnisse erreichbar.

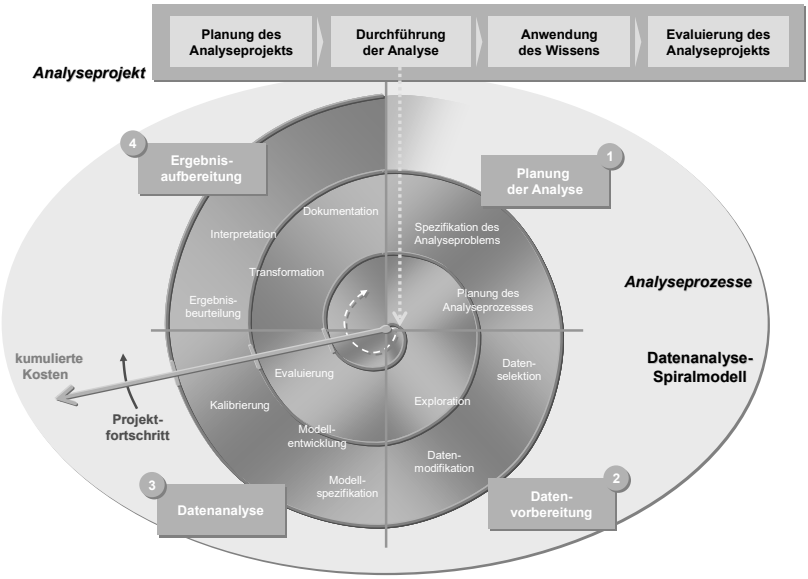


Abbildung 20: Vorgehensmodell für Datenanalyseprojekte (eigene Darstellung)

Für Zwecke des Projektmanagements wird jede Windung ungeachtet ihrer Ursache gezählt,[57] wodurch sich der Gesamtressourcenverbrauch

[57] Der Zykluszähler wird erhöht, wenn der Ablauf ein bereits abgeschlossenes Spiralsegment erneut durchschreitet. Diese Situation kann durch Zuordnung der Aufgaben zu den Spiralsegmenten (bzw. generischen Analyseprozessphasen) leicht automatisiert überwacht werden.

abschätzen lässt. Zusätzlich kann für jedes Analyseproblem die spezifische Zahl der Zyklen dokumentiert werden. Analyseprobleme werden als Inkrement innerhalb der Analysekette separat registriert. Dieses Messverfahren stellt eine Weiterentwicklung der Idee BOEHMs für die Datenanalyse dar. Er schlägt vor, die kumulierten Projektkosten an der Länge einer gedachten Geraden vom Koordinatenursprung zur aktuell äußersten Spiralwindung abzuschätzen und den Projektfortschritt innerhalb eines Zyklus am Winkel dieser Geraden abzulesen [Boeh88, 65].

Der Projekterfolg lässt sich in gewissem Maße durch systematisches *Risikomanagement* beeinflussen. Aufgrund der experimentellen Natur der Datenanalyse können brauchbare Resultate jedoch selbst bei korrekter Ausführung aller notwendigen Schritte auch nach zahlreichen Zyklen nicht garantiert werden [DeHa01, 258]. Zur Vermeidung unkontrollierter Kostenentwicklungen wird eine neue Windung nur betreten, wenn eine Risikoabwägung positiv ausfällt. Sie kann nach Zeit- und Kostenkriterien [Küpp99, 151], [Kafk99, 45] sowie auf Basis einer systematischen Zielkontrolle (vgl. Abschnitt 6.2.3.3) erfolgen. Die konkrete Realisierung dieser Vorstellung bleibt dennoch im Einzelfall schwierig und läuft teilweise der Datengetriebenheit explorativer Analyseansätze entgegen.

3.3.3.3 Anwendung des Wissens

Die Anwendung des Wissens kann zu weiteren Projekten unterschiedlicher Art anregen. Die Realisierung von Entscheidungen und Handlungsmaßnahmen (z.B. einer Betriebsreorganisation) führt häufig zu umfangreichen Vorhaben, die jeweils individuell zu handhaben sind. Die Nutzung analytisch erzeugter Modelle zu Prognose- oder Inferenzzwecken sowie die Bereitstellung von Data Products konstituiert jeweils ein neues Datenanalyseprojekt, das in das aktuelle Projekt verschachtelt ist. Im einfachsten Fall werden Analyseergebnisse in einem Abschlussbericht für den Auftraggeber ausführlich dokumentiert [Drei94, 34]. Anschlussanalysen zur Verifikation der Ergebnisse einer ersten Analyse im Rahmen des Analysezyklus (vgl. Abschnitt 2.3.2.3) sollten hingegen

als Glieder einer Analysekette betrachtet und innerhalb des Spiralmodells behandelt werden.

3.3.3.4 Evaluierung des Analyseprojekts

Bei der Evaluation wird der tatsächlich realisierte Erfolg mit der Zielsetzung des Projekts verglichen [BeLi97, 28f.] und monetär bewertet. Sie umfasst die Beurteilung der Geeignetheit der gewählten Datenanalyse und die Überprüfung der Wirksamkeit der zur Anwendung des Wissens ergriffenen Maßnahmen. In die Bewertung der Datenanalyse gehen die *Analyseergebnisse* sowie der zugehörige *Prozessablauf* mit all seinen Bestimmungsfaktoren (analytischer Ansatz, Daten, Aktivitäten, Verfahren) ein [Knob03a, 342]. Zur *Evaluation der Handlungsmaßnahmen* werden in der Regel weitere Datenanalysen erforderlich [Wild01, 14] (Schachtelung). An die Beurteilung von Analyseprozessen und Handlungskonsequenzen schließt die *Bewertung des ökonomischen Erfolgs* an, der sich aus der Differenz des Ertrags der Maßnahmen und der Summe aller Aufwendungen des Projekts ergibt. Hierbei werden die Kosten der Maßnahmenrealisierung und jene der Analyse einbezogen [BeLi97, 109-111]. Die ganzheitliche ökonomische Bewertung kann somit erst nach Abschluss des gesamten Projekts erfolgen [DeHa01, 72]. Am Ende der Evaluierung steht die *Erfahrungssicherung*, welche die Grundlage für die Realisierung von Lerneffekten bildet und Verbesserungspotenziale für nachfolgende Analyseprojekte eröffnet [BeLi97, 34f.], [Mart98, 24-26] (Regelkreis, vgl. Abschnitt 2.4.4).

3.3.4 Zusammenfassung: Vorgehensmodell zur Datenanalyse

Die Konzeption des Vorgehensmodells basiert auf den in Abschnitt 3.2.4.3 angestellten Überlegungen zur Komplexitätsbewältigung. Hierzu sollen insbesondere die hierarchisch-modulare Strukturierung (B1.1, B1.2), die Vorgabe eines allgemeingültigen Prozessschemas (B2.2) und die Unterstützung flexibler, evolutionärer Vorgehensweisen (B2.3) beitragen. Die gewählte Differenzierung in Projekt- und Prozessebene erscheint aus folgenden Erwägungen sinnvoll: Zum einen erlaubt sie während eines Projekts den Austausch oder Wechsel des Analyseansatzes, ohne eine Änderung des Vorgehens auf Projektebene nach

sich zu ziehen. Zum anderen gestattet sie innerhalb eines Projekts mehrere verkettete Analysen, deren Anzahl sich oft erst im Zuge der Untersuchung ergibt. Die gewählte Modularstruktur vermeidet darüber hinaus, dass durch „flache“ Aneinanderreihung einzelner Projekte der Gesamtzusammenhang aller Aktivitäten des Hauptprojekts aufgetrennt wird. Eine integrale Gesamtsicht ist insbesondere für die Projektevaluierung hilfreich. Bei Schachtelung mehrerer Analyseprojekte berücksichtigt die jedes Projekt abschließende Evaluierung nicht nur die Aktivitäten des betrachteten Projekts, sondern auch jene aller hierarchisch tiefer stehenden Teilprojekte.

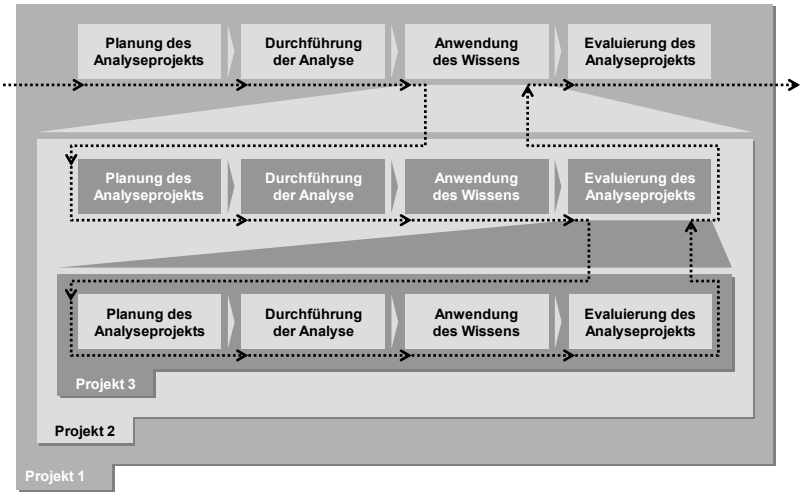


Abbildung 21: Schachtelung von Datenanalyseprojekten (eigene Darstellung; vgl. [NeKn15, 191])

Abbildung 21 zeigt den Ablauf durch die Phasen eines Projekts 1, das mehrstufig zwei Teilprojekte umschließt. Die Bewertung von Projekt *i* wird jeweils in die Beurteilung des übergeordneten Projekts *i-1* einbezogen. Der gezeigte Fall tritt in der Praxis häufig auf, z.B. wenn ein analytisch hergeleitetes Klassifikationsmodell (Projekt 1) für die modellgestützte Auswahl der Zielgruppen für Marketingkampagnen eingesetzt (Projekt 2) und anschließend der Erfolg jeder einzelnen Kampagne gemessen wird (Projekt 3).

Das Modell ist zur Lenkung von Projekten geeignet. Eine strukturierte Auflistung aller Phasen und Aufgaben ist in Anhang A3 enthalten. Eine Erweiterung des Modells zu einer Methodik durch detaillierte Handlungsempfehlungen für die einzelnen Phasen wird in den weiteren Kapiteln der Arbeit entwickelt. Hierbei wird zugleich die enge Verzahnung der Aufgaben der Problemspezifikation auf den beiden Modellebenen aufgelöst.

Teil B:

Eine Methodik für das Management von Datenanalyseprozessen

Zur effektiven und effizienten Handhabung von Datenanalyseprozessen ist eine umfassende methodische Unterstützung erforderlich (vgl. Maßnahme B2 in Abschnitt 3.2.4.3). Als ***Methodik*** *bzw. Methode*[58] *gilt allgemein ein nach Mittel und Zweck planmäßiges Verfahren, das zu technischer Fertigkeit bei der Lösung von Aufgaben führt. Für die Systementwicklung in der Informatik umfasst eine Methodik grundsätzlich eine Sprache zur Repräsentation von Anforderungen und Ergebnissen der Entwicklung sowie eine strukturierte Vorgehensweise, welche eine Folge von Arbeitsschritten sowie zugehörige Regeln und Empfehlungen für das Entwicklungsvorhaben festlegt [JaBS97, 487], [Somm01, 26f.]. Die folgenden Kapitel entwerfen eine Methodik für das Management von Datenanalyseprozessen. In Kapitel 4 wird ein geeigneter Modellierungsansatz (Sprache) entwickelt. Die Kapitel 5-7 zeigen detaillierte Handlungsschemata für das Vorgehen bei der Planung, Steuerung und Revision der Prozesse. Sie dienen der Präzisierung und Erweiterung des Vorgehensmodells aus Abschnitt 3.3*

[58] Der Begriff *Methodik* bezeichnet im eigentlichen Sinne die Methodenlehre der Wissenschaft. Ein planmäßiges, wissenschaftlich fundiertes Vorgehen heißt *Methode* [Kien82, 271]. Zur Abgrenzung vom allgemeinen Verfahrensbegriff wird anstelle von Methode jedoch häufig der Begriff Methodik gebraucht. Dieser Sprachregelung wird auch in dieser Schrift gefolgt.

4 Modellierung von Datenanalyseprozessen

Das vorliegende Kapitel entwickelt einen umfassenden Ansatz zur Modellierung von Datenanalyseprozessen. Zunächst stellt Abschnitt 4.1 die Motivation für einen solchen Ansatz dar. Abschnitt 4.2 leitet mit der Datenanalysearchitektur eine geeignete Struktur für das Modellsystem her. Dessen Ebenen werden in den anschließenden vier Abschnitten detailliert erläutert. Abschnitt 4.7 erweitert die Betrachtung um spezielle Sichten auf Datenanalyseprozesse. Abschnitt 4.8 fasst die wesentlichen Eigenschaften des Ansatzes zusammen.

4.1 Repräsentation von Datenanalyseprozessen

Im Folgenden werden die Ziele und Anforderungen betrachtet, welche die Herleitung des Ansatzes zur Repräsentation von Datenanalyseprozessen leiten.

4.1.1 Ziele der Modellierung

Die Prozessmodellierung gilt als Voraussetzung für das Prozess- bzw. Workflow-Management [Sinz94, 220], [JaBS97, 5] und dient der Herstellung von *Prozessleistungs-* und *Prozessstrukturtransparenz* (vgl. Abschnitt 2.4.2). Die systematische Festlegung und Messung von Ergebnisparametern sowie die detaillierte Dokumentation geplanter Prozesse und realisierter Abläufe ist ohne adäquate Repräsentation nicht möglich. Da das Prozessmanagement eine ganzheitliche Betrachtungsweise impliziert, ist eine möglichst vollständige Erfassung und Abbildung aller wesentlichen Aspekte der jeweiligen Anwendungssituation anzustreben [ScVr94a, 25f.], [HaSt98, 163], [JaBS97, 18].

Prozessmodelle dokumentieren im deskriptiven Sinne zunächst den Fortschritt und die Ergebnisse der *Prozessgestaltung* und schaffen eine Kommunikationsgrundlage für alle Beteiligten. Der *Prozesslenkung* dienen sie im präskriptiven Sinne als Vorlage, indem sie die Reihenfolge der auszuführenden Aktivitäten, die benötigten Ressourcen sowie einzuhaltende Regeln und Restriktionen vorgeben [WfMC99, 8], [Gier00, 20], [Reif03, 38]. Darüber hinaus besitzen sie Protokollfunktion

zur Überwachung des aktuellen Zustands der Prozesse während ihrer Ausführung sowie zur Analyse abgeschlossener Abläufe [WSG+97, 251f.], [Reif03, 72-74, 83]. Sie legen damit die Grundlage für die kontinuierliche Verbesserung im Sinne der *Prozessentwicklung*. Von besonderer Bedeutung sind Modelle im Hinblick auf die automatisierte Steuerung und Ausführung, Verifikation und Validierung (Simulation) von Prozessen [JaBS97, 19], [HaBR08, 48].

4.1.2 Anforderungen an den Modellierungsansatz

Die oben genannten allgemeinen Ziele sind vor dem Hintergrund von Datenanalyseprozessen, wie sie in der vorliegenden Arbeit verstanden werden, zu konkretisieren und zu erweitern. Zunächst sollte der Ansatz geeignet sein, prinzipiell alle Erscheinungsformen der Datenanalyse abzudecken, d.h., er sollte nicht die Spezifika eines analytischen Ansatzes oder einer Verfahrensklasse in den Vordergrund rücken, sondern möglichst offen angelegt sein, um auch künftige Entwicklungen aufnehmen zu können. Dies ist zugleich ganz im Sinne der integrativen, anwendungsorientierten Perspektive, wie sie die evidenzbasierte Problemlösung einnimmt (vgl. Abschnitt 2.2.2.8). Die Einbeziehung der Sachprobleme ist daher von großer Bedeutung. Gemäß dem in Abschnitt 2.3.1 entwickelten Verständnis umfasst eine vollständige Prozessbeschreibung die Ebenen der Ziele, der Aufgaben und der Ressourcen. Sie sind ebenso zu berücksichtigen wie die Aufgaben der drei Phasen Planung, Steuerung und Kontrolle des Vorgehensmodells aus Abschnitt 3.3.

Weitere Anforderungen ergeben sich aus den in Abschnitt 3.2.4 diskutierten Maßnahmen zur Komplexitätsbewältigung. Aus Sicht der Prozessmodellierung sind hier insbesondere folgende Aspekte von Interesse:

- Unterstützung der Problemspezifikation (B2.1) unter Berücksichtigung der Zielorientierung und -differenzierung (R1.1, R12) sowie der Datenquellenauswahl und Datenselektion (R1.3);
- Unterstützung der Wiederverwendung von Analyseergebnissen (U2, R2.4) und Modellierungsartefakten (R3.1);

- Abbildung abstrakter Prozesspläne (R3.3) und Unterstützung der Prozessflexibilität (B2.3 und Abschnitt 2.4.2);
- Strukturierung, Modularisierung und Hierarchisierung des Modellsystems (B1.1, B1.2);
- Abbildung von Aufgaben- und Verfahrenstaxonomien (B1.3);
- Abbildung von Erfahrungs- und Domänenwissen (B2.4);
- Bereitstellung kontextspezifischer Handlungsempfehlungen und Best Practices (B2.5, B2.6);
- Berücksichtigung von Anforderungen einer Werkzeugunterstützung (B3.1).

Der letzte Aspekt umfasst auch Anforderungen bezüglich der automatisierten Analyse und Gestaltung der Workflow-Schemata (B3.2), die eine gewisse formale Fundierung voraussetzen [Rögl09, 501]. Aus der Sicht personeller Aufgabenträger ist die intuitive Verständlichkeit von zentraler Bedeutung, gerade auch angesichts des in der Datenanalyse typischerweise großen Interaktionsbedarfs mit Fachexperten und Auftraggebern. Hierzu tragen u.a. semiformale Repräsentationsformen, nachvollziehbare Metaphern sowie die Verwendung von in der Analysepraxis üblichen Begriffen bei. Ein Modellierungsansatz, der diesen Anforderungen nicht ausreichend genügt, wird für den Einsatz in der betrieblichen Praxis kaum in Erwägung gezogen.

4.2 Die Datenanalysearchitektur

Zur Entwicklung eines Modellierungsansatzes, der die gestellten Anforderungen erfüllen kann, untersucht Abschnitt 4.2.1 die Konzeption von Datenanalysen und leitet daraus vier Ebenen einer Datenanalysearchitektur her. Deren Struktur und der resultierende Nutzen werden in Abschnitt 4.2.2 behandelt.

4.2.1 Konzeption von Datenanalysen

Da Datenanalyse eine Untersuchung darstellt, wird zur ihrer Konzeptualisierung und Strukturierung der von FERSTL vorgeschlagene Begriff *Untersuchungssituation* herangezogen [Fers79, 43f.]. Demnach besteht eine Untersuchung aus einer Folge von vier Aktivitäten:

1. Abgrenzung bzw. Beschreibung des *Untersuchungsobjekts* O durch Angabe bekannter Eigenschaften von O;
2. Festlegung eines *Untersuchungsziels* Z, das sich auf unbekannte Eigenschaften von O richtet;
3. Bestimmung einer Menge L von verfügbaren Lösungsverfahren V_i zur Erreichung des Untersuchungsziels Z;
4. Anwendung mindestens eines Untersuchungsverfahrens V_i aus L.

Ein ***Untersuchungsproblem*** (O, Z) beschreibt das bezüglich eines Untersuchungsobjekts zu verfolgende Untersuchungsziel. Eine ***Untersuchungssituation*** (O, Z, L) entsteht durch Anwendung eines Untersuchungsverfahrens auf ein Untersuchungsproblem und liefert eine Problemlösung.

Datenanalysen kennzeichnet der Umstand, dass das Untersuchungsobjekt O nicht direkt, sondern mittels geeigneter Daten untersucht wird, die ausgewählte Eigenschaften des Untersuchungsobjekts repräsentieren (vgl. Abschnitt 2.1.2.3). Auf dieses Modell O_M können die Verfahren der Datenanalyse angewandt werden. Sie ermöglichen Untersuchungen, die bezüglich des eigentlichen Untersuchungsobjekts nicht oder nur schwer durchführbar sind. Zu diesem Zweck ist das ursprüngliche Untersuchungsproblem, das hier als *Sachproblem* bezeichnet sei, in ein Modellproblem zu überführen, woraus eine *modellgestützte Untersuchungssituation* resultiert (vgl. [Fers79, 79f.] und Abbildung 22 oben). Das ursprüngliche Untersuchungsziel Z wird dabei in ein *Analyseziel* Z_M bezüglich des *Analyseobjekts* O_M transformiert. Das zugehörige modellgestützte Untersuchungsproblem wird als (Daten-)

Analyseproblem[59] bezeichnet. Als Lösungsverfahren LM für das Analyseproblem ist ein *Analyseprozess* zu konstruieren, der ein *Analyseergebnis* für das Analyseproblem (OM, ZM) liefert. Dieses Ergebnis ist danach in eine Lösung bezüglich des Sachproblems (O, Z) zu transformieren.

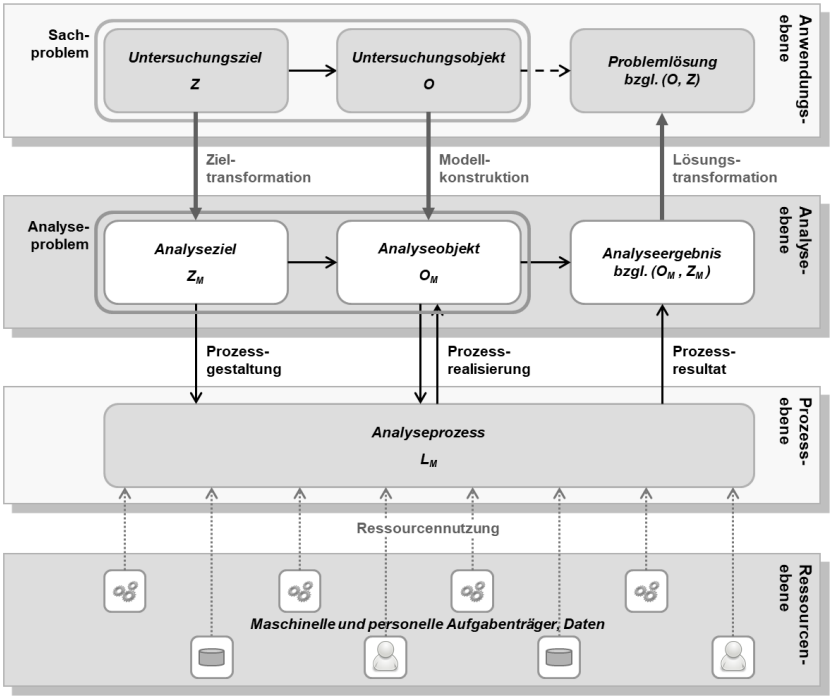


Abbildung 22: Datenanalyse als modellgestützte Untersuchungssituation auf verschiedenen Betrachtungsebenen (in Anlehnung an [Fers79, 80], Erweiterung für Datenanalyse: eigene Darstellung)

[59] Der hier verwendete Begriff des Analyseproblems ist aus der Datenanalyse abgeleitet und nicht mit jenem bei FERSTL identisch. Im dortigen Sinne können Datenanalyseprobleme als Analyse- oder Black-Box-Probleme sowie als Mischform (Grey-Box-Probleme) ausgeprägt sein (vgl. [Fers79, 44-54]). Die Datenanalyse untersucht somit typischerweise existierende Systeme bezüglich unbekannter Struktur- oder Verhaltenseigenschaften.

Ein Analyseprozess transformiert Analysedaten (gemäß Analyseobjekt) in ein Analyseergebnis (gemäß Analyseziel). Seine Realisierung erfolgt durch maschinelle und personelle Aufgabenträger, die als gegeben vorausgesetzt werden (vgl. Abschnitt 2.3.2). Die Eigenschaften dieser aktiven *Ressourcen* nehmen ebenso Einfluss auf die Prozessgestaltung wie die Daten als passive Ressourcen, indem sie die ausführbaren bzw. auszuführenden Transformationen determinieren.

4.2.2 Struktur und Nutzen der Datenanalysearchitektur

Aus dieser Konzeption resultieren vier Betrachtungsebenen von Datenanalyseprozessen, die in Abbildung 22 dargestellt sind. (1) Das Sachproblem, das mit dem eigentlich zu untersuchenden Sachverhalt den Zweck und den Anwendungskontext einer Analyse erfasst, wird auf der *Anwendungsebene* beschrieben. (2) Daraus wird auf der *Analyseebene* ein geeignetes Analyseproblem abgeleitet. (3) Zu dessen Lösung erfolgt auf der *Prozessebene* die Konstruktion eines Analyseprozesses, wofür (4) auf der *Ressourcenebene* beschriebene Ressourcen genutzt werden, die auch zur Prozessdurchführung zur Verfügung stehen. Die vier Betrachtungsebenen bilden die *Architektur von Datenanalyseprozessen* (kurz: ***Datenanalysearchitektur***), die sich am Konzept der Informationssystem-Architektur bzw. dem zugehörigen generischen Architekturrahmen nach SINZ [Sinz97] orientiert. Die Datenanalysearchitektur konkretisiert die Projektebene des Vorgehensmodells aus Abschnitt 3.3 zur Anwendungs- und Analyseebene und ergänzt die Ressourcenebene. Die Analyseebene stellt das Bindeglied zwischen Projekt- und Prozessebene her, die im Vorgehensmodell noch eng miteinander verzahnt sind.

Der Architekturbegriff ist dem Bauwesen entlehnt und umfasst einen Bauplan, der die Komponenten des zu konstruierenden Systems samt ihrer Beziehungen unter allen relevanten Blickwinkeln beschreibt, sowie die Konstruktionsregeln für die Erstellung des Bauplans. Der Bauplan stellt ein *Modellsystem* (Abbild) dar, die Konstruktionsregeln werden in Gestalt von Metamodellen angegeben. Ein *Metamodell* spezifiziert die verfügbaren Bausteine (Metaobjekte), die zulässigen Beziehungen zwischen den Bausteinen (Metabeziehungen) sowie Konsistenzbedingungen für die Kombination von Bausteinen und Beziehungen zur

Erstellung gültiger Modellsysteme. Zum Zweck der Komplexitätsbeherrschung wird das Modellsystem in Modellebenen und zugehörige Sichten gegliedert. Eine *Modellebene* enthält eine vollständige Beschreibung des Systems unter einem definierten Blickwinkel, der jeweils bestimmte, mit der Modellbildung verfolgte Ziele unterstützt [Sinz97, 2-4]. Die Datenanalysearchitektur betrachtet Analyseprozesse unter folgenden Blickwinkeln: (1) Die Anwendungsebene stellt die Zwecksetzung und die fachlichen Rahmenbedingungen für eine Datenanalyse dar („Wozu?"), (2) die Analyseebene zeigt den durch die Analyse zu deckenden Informationsbedarf und den zugehörigen Datenbedarf („Was?"). (3) Die Prozessebene gibt Auskunft über das konkrete Vorgehen bei der Durchführung der Analyse (höheres Lösungsverfahren, „Wie?"), und (4) die Ressourcenebene beschreibt die aktiven und passiven Ressourcen (Aufgabenträger, Daten), mit denen die Untersuchung erfolgt („Womit?"). Alle vier Ebenen gemeinsam bilden eine vollständige Beschreibung einer Datenanalyse im betriebswirtschaftlich-fachlichen Umfeld und schaffen eine Grundlage zu deren ganzheitlicher Planung, Steuerung und Revision.

Jede Modellebene besitzt ein eigenes Metamodell. Zur weiteren Komplexitätsbewältigung auf den einzelnen Ebenen dienen *Sichten*, die jeweils eine Projektion auf das Metamodell der Ebene vornehmen und in der Regel eine unvollständige Beschreibung der Modellebene geben. Alle paarweisen Beziehungen zwischen Modellebenen werden in einem Beziehungsmetamodell spezifiziert, das Metaobjekte der betroffenen Ebenen über Zuordnungsbeziehungen verknüpft. Für Modellebenen oder für Beziehungen zwischen Ebenen können zudem *Strukturmuster* festgelegt werden. Strukturmuster beschreiben partielle Problemlösungen und können heuristisches Modellierungswissen oder Integritätsbedingungen enthalten, welche die Zahl der zulässigen Gestaltungsoptionen einschränken [Sinz97, 3f.]. Sie werden jeweils in einer Bibliothekssicht beschrieben.

Die Darstellung der Metamodelle des Ansatzes folgt dem Meta-Metamodell in Abbildung 23. Seine Bausteine sind Metaobjekttypen (symbolisiert durch ein Rechteck), die durch Metabeziehungen (Kante) verknüpft sind. Als Typen von Metabeziehungen sind Generalisierung

(is_a), Aggregation (is_part_of), Assoziation (connects) und Attribut-Zuordnungsbeziehungen (has) definiert. Einer Metabeziehung können zwei Kardinalitäten in (min, max)-Notation zugeordnet werden, die jeweils die minimal bzw. maximal zulässige Anzahl zu verknüpfender Instanzen der zughörigen Metaobjekttypen bestimmen.

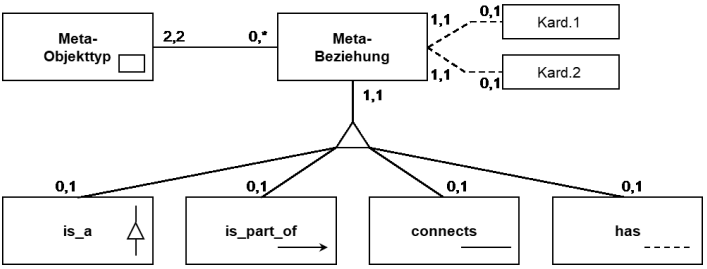


Abbildung 23: Meta-Metamodell, in Anlehnung an [FeSi13, 139], [Böhn01, 262]

Zur Unterstützung der umfassenden Dokumentation aller mit einem Modellbaustein assoziierten Belange enthält jeder Metaobjekttyp eine Reihe von Attributen, die von einem generischen **`Objekttyp`** geerbt werden.[60] Hierzu gehören neben `Name`, `Beschreibung`, `Kommentaren` und `Schlüsselworten` zur Suche insbesondere `Kontext`, `Kontext-regeln` (vgl. Abschnitt 4.7) sowie eine Menge von `Links`, die der Verknüpfung beliebiger Ressourcen dienen. Diese sind hilfreich, um etwa Dokumente oder Diagramme (z.B. Aufzeichnungen; Analyseergebnisse; Datenschemata für Datenquellen) zu hinterlegen. Die Menge aller Attribute ist in Anhang A4 tabellarisch aufgelistet.

4.3 Anwendungsebene: Problemstellung und Zweck der Datenanalyse

Datenanalyse erfolgt nicht zum Selbstzweck, sondern soll die Lösung von Problemen aus der Anwendungsdomäne unterstützen. Solche Probleme werden in Abgrenzung zu Analyseproblemen als ***Sach-***

[60] Im Weiteren sind `Attribute` von Metaobjekttypen bei ihrer ersten Nennung oder expliziten Referenzierung jeweils durch nichtproportionale Schriftart, **`Metaobjekt-typen`** zusätzlich durch Fettdruck gekennzeichnet.

probleme bezeichnet. Sie sind im Sinne von Anwendungs-, fachlichen oder Geschäftsproblemen zu verstehen. Die Anwendungsebene der Datenanalysearchitektur dokumentiert mit Darstellung dieser Sachprobleme den Zweck und die fachlich-organisatorischen Rahmenbedingungen einer Datenanalyse.

Ihre Beschreibung erfolgt zunächst unabhängig von analytischen Erwägungen. Auf diese Weise ist eine rein auf sachliche Überlegungen fokussierte Identifizierung, Konkretisierung und Strukturierung von Sachproblemen gewährleistet. Dies erfolgt in der *Problemstruktursicht*. Hinreichend detailliert beschriebenen Sachproblemen können Datenanalysen zugeordnet werden, wenn diese einen Beitrag zur Problemlösung leisten können. Erfordert das Wesen eines Sachproblems andere Maßnahmen (z.B. Entscheidungen, Handlungen) zu seiner Lösung, können diese alternativ zur Datenanalyse als Lösungsoptionen modelliert werden. Diese Darstellung erfolgt in der *Problemlösungssicht*. Der Speicherung von Modellierungsartefakten widmet sich die *Bibliothekssicht*. Die Anwendungsebene unterstützt somit die umfassende Analyse von Sachproblemen und die Gestaltung geeigneter Lösungsansätze auch außerhalb des Einsatzbereichs der Datenanalyse und stellt in diesem Sinne einen Beitrag zur Informationsverarbeitung in komplexen Situationen bereit. Gemäß dieser erweiterten Aufgabenstellung ist die in Abschnitt 4.2.1 eingeführte Sicht auf ein Sachproblem als Untersuchungsproblem zu generalisieren. Die Rolle von Sachproblemen, die zur Lösung mithilfe einer Datenanalyse geeignet erscheinen, als Ausgangspunkt zur Ableitung von Analyseproblemen bleibt davon jedoch unberührt.

4.3.1 Problemstruktursicht

Ein ***Problem*** wird allgemein als wahrgenommene Diskrepanz zwischen einem unerwünschten *Ausgangszustand* und einem anzustrebenden *Zielzustand* definiert, die durch Operatoren zu überbrücken ist. Im Vergleich zu einer Aufgabe ist ein Problem insbesondere dadurch gekennzeichnet, dass die Transformation des Ist- in den Soll-Zustand durch eine *Barriere* verhindert wird [Dörn79, 10f.], [Gait83, 67], [FiWo90, 12]. Ein Lösungsverfahren zur Überwindung der Diskrepanz ist damit

nicht unmittelbar verfügbar, sondern muss während des Problemlöseprozesses entwickelt werden. Dieses Problemverständnis als Wunsch nach Überwindung eines unbefriedigenden Zustands trifft auf ein breites Spektrum von Situationen in allen Lebensbereichen zu.

Jedes Sachproblem ist explizit oder implizit mit einer Menge von Objekten oder Konzepten der Realität assoziiert, auf deren Merkmale sich die Diskrepanz bezieht. Dabei kann es sich z.B. um die Zufriedenheit (Merkmal) der Kunden (Objekt), den Bestand an Aufträgen, den Wert von Einkäufen oder die Qualität von Produkten handeln. Ein Domänenobjekt, das Gegenstand einer solchen Diskrepanz ist, heißt ***Problemobjekt***, das betroffene Merkmal ***Problemmerkmal***. Die Problembeschreibung erfordert eine detailliertere Betrachtung von Ziel- und Ausgangszustand. Hierbei werden sukzessive zur Modellierung von Sachproblemen geeignete Metaobjekttypen und zugehörige Attribute aufgedeckt.

4.3.1.1 Zielzustand

Ein Ziel kann allgemein als angestrebter künftiger Zustand [Hein91, 13], konkreter als erstrebenswerter Soll-Zustand eines Zielkriteriums [Beck95, 69], [Reif03, 22] definiert werden. HEINEN nennt zusammenfassend die Beschreibungsdimensionen *Inhalt*, *Ausmaß* und *Zeitbezug* von Zielen [Hein91, 14]. Die Zielzustände von Sachproblemen beziehen sich stets auf das Problemmerkmal, das demnach den `Inhalt` des Ziels darstellt. Sein `Ausmaß` wird durch Spezifikation einer konkreten Ausprägung (Wert) des Soll-Zustands definiert. Zusätzlich ist mittels des `Zeitbezug`s festzulegen, bis wann das Ziel zu erreichen ist (konkreter Termin, absoluter Zeitraum oder relative Zeitspanne). Durch Festschreibung von Zielausmaß und Zeitbezug werden das Ziel messbar und die Zielerreichung einer Bewertung zugänglich.

Sofern das Sachproblem nicht der Wertsphäre des Unternehmens entspringt (z.B. Rentabilitätssteigerung), nimmt der Zielzustand auf Veränderungen in der Leistungssphäre Bezug (sachliche Ziele). Um die Konformität des Zielzustands mit den Unternehmenszielen sicherzustellen, sind sachliche Ziele mit betriebswirtschaftlichen Zielen zu verknüpfen (vgl. [DeHa01, 240], [Pyle03, 128]). Diese Verknüpfung

erfolgt über den Zielinhalt, dem durch eine Zweck-Mittel-Relation jeweils ein betriebswirtschaftliches Oberziel zugeordnet wird, das vom sachlichen Ziel unterstützt wird. Die Beschreibung des Zielzustands erfährt damit eine Ergänzung um die Dimension `Wertbeitrag`.[61] Die erläuterten Komponenten der Zustandsbeschreibung am Beispiel eines sinkenden Bonbetrags (Einkaufswert) im Einzelhandel [NeKn06] zeigt Abbildung 24 rechts.

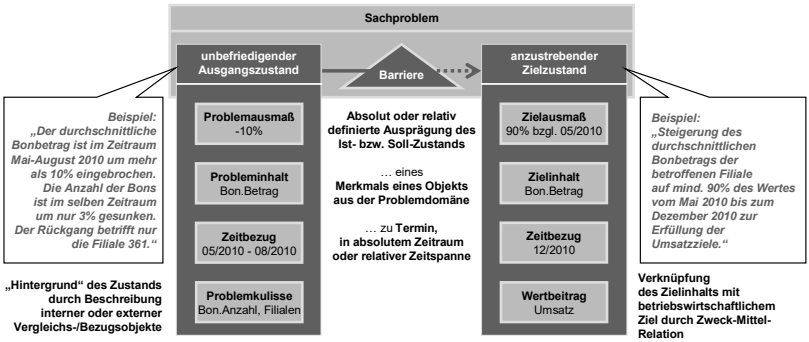


Abbildung 24: Komponenten und Beschreibungselemente von Sachproblemen (eigene Darstellung)

4.3.1.2 Ausgangszustand

Der unbefriedigende Ausgangszustand wird analog zum Zielzustand repräsentiert, indem die Beschreibungselemente `Inhalt`, `Ausmaß` und `Zeitbezug` aus der Zieltheorie auf die Ist-Situation übertragen werden. Zur Unterstützung der Einordnung und Bewertung des Problems wird eine Beschreibung des Hintergrunds des problematischen Zustands ergänzt, die im Folgenden als Problem-`Kulisse` bezeichnet wird. Diese setzt dem Probleminhalt Vergleiche mit unternehmensexternen oder -internen Bezugsobjekten entgegen, um eine objektivere Einschätzung der Situation zu erlauben. Sie sind jeweils im Hinblick auf die konkreten Merkmale zu wählen, die den aktuellen Probleminhalt bilden. Externe Bezugsobjekte sind z.B. der Gesamtmarkt oder

[61] Hierbei wird keine Zielausprägung angegeben, sondern lediglich das zu unterstützende Ziel benannt.

Mitbewerber, interne Bezugsobjekte etwa Produktsparten, Produkte, Organisationseinheiten oder Vertriebsgebiete. Abbildung 24 zeigt links die Komponenten des Ausgangszustands zusammen mit einem Beispiel, das mit der Beschreibung des Zielzustands korrespondiert.

Zur Vervollständigung der Problembeschreibung kann auch der Ausgangszustand durch einen `Wertbeitrag` und der Zielzustand durch eine Problem-`Kulisse` charakterisiert werden. Der `Wertbeitrag` des Ist-Zustands verweist auf jenes Ziel der Wertsphäre, welches durch die Problemsituation primär Beeinträchtigung erfährt. Mit seiner Hilfe ist etwa eine Priorisierung von Problemen möglich. Die `Kulisse` des Soll-Zustands nennt Rahmenbedingungen, die bei der Herstellung der gewünschten Zielsituation zu beachten sind, und kann demnach wichtige Hinweise zur Wahl einer Lösungsoption geben. Weiterhin können beide Zustände durch eine ausführliche `Beschreibung` in Textform ergänzt werden, um etwa Erläuterungen wie in der Abbildung gezeigt bereitzustellen.

4.3.1.3 Problemaspekt

Es wird deutlich, dass die vollständige Beschreibung eines Sachproblems aus zwei Komponenten besteht: aus jeweils einer Repräsentation des Ausgangszustands und des Zielzustands. Beide Komponenten dienen verschiedenen Zwecken und erfordern unterschiedliche Handhabung. Während der Ausgangszustand eine Situationsbeschreibung liefert und Ansatzpunkt für die Untersuchung der Ursachen und Bedingungen des Problems darstellt, weist der Zielzustand den Weg zur Problemlösung, die durch Konstruktion geeigneter Handlungsschritte und -maßnahmen (Operatoren) zu entwickeln ist. Ein Sachproblem wird daher durch zwei über eine sequenzielle Beziehung verknüpfte **`Problemaspekte`** modelliert. Die sequenzielle Beziehung repräsentiert die intendierte Handlungsrichtung, die anfangs noch durch die Problembarriere blockiert ist. Beide Problemaspekte besitzen die gleiche Struktur; ihre abweichende Rolle wird anhand ihres `Typ`s (Zustandsversion) gekennzeichnet: **`Situationsbezogene Problemaspekte`** (`Typ`=„Ist“) repräsentieren Ist-Zustände, **`lösungsbezogene Problemaspekte`** (`Typ`=„Soll“) Soll-

Zustände. Zur einfachen Identifizierung erhält jeder Problemaspekt einen `Namen`.

Problemaspekte sind geeignet, komplexe Probleme zu strukturieren und zugleich Optionen zur Problemlösung aufzudecken. Eine Problemlösung beschreibt eine Transformation vom Ausgangs- in einen Zielzustand. Sie erfolgt häufig über mehrere Zwischenzustände, deren Überbrückung neue Problemaspekte offenlegt. GAITANIDES betrachtet diese „Problemschachtelung" als *Differenzierung*, die nach Teilproblemen oder Unterproblemen erfolgen kann. *Teilprobleme* entstehen durch Ausgrenzung aus einem Gesamtproblem, zu dem sie jeweils in einer Teil-Ganzes-Beziehung stehen. *Unterprobleme* resultieren aus der Zerlegung von Überproblemen und zielen auf deren Lösbarkeit im Sinne einer Mittel-Zweck-Beziehung [Gait83, 70].[62] Die Differenzierung situationsbezogener Problemaspekte unterstützt bei der Identifizierung von Problemursachen und Teilproblemen sowie bei der Konkretisierung von Problembeschreibungen. Die Differenzierung lösungsbezogener Problemaspekte dient der Entwicklung von Problemlösungen mithilfe von Unterproblemen. Zur Dokumentation potenzieller Einflussfaktoren oder relevanter Bezugsobjekte besitzt jeder Problemaspekt eine `Problemdomäne`, die eine Menge von Domänenobjekten umfasst und initial aus dem Problemobjekt besteht.

4.3.1.4 Metamodell

Abbildung 25 zeigt das Metamodell zur Problemstruktursicht mit wichtigen Attributen[63] des Problemaspekts. Zur Analyse der Problemstruktur können jeweils zwei (2,2) Problemaspekte miteinander verknüpft werden. Ein **`Problemaspekt`** kann beliebig viele (0,*) Verknüpfungen aufweisen. Eine **`Verknüpfung`** kann als **`Sequenz`**, als **`Teil-von`**-Beziehung oder als **`Mittel-Zweck`**-Beziehung ausgeprägt sein. Die letzten beiden Arten realisieren eine **`Differenzierung`** eines

[62] Die Problemdifferenzierung ist eine Form der Subsystembildung bzw. Hierarchisierung als Strategie zur Komplexitätsbewältigung (vgl. B1.2 in Abschnitt 3.2.4.3).

[63] Attribute von Metaobjekttypen gehören naturgemäß zu genau einem Metaobjekttyp (1,1). Die vollständige Attributliste enthält Anhang A4.

Vater-Aspekts in Kind-Aspekte. Ein Problemaspekt kann den Typ **`situationsbezogen`** oder **`lösungsbezogen`** annehmen und wird durch ein nach oben bzw. unten gerichtetes Trapez symbolisiert. Bei Verzicht auf die Typisierung wird er als Rechteck repräsentiert.

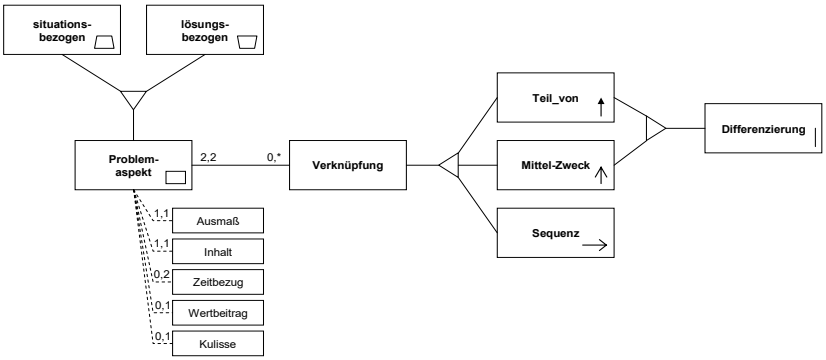


Abbildung 25: Metamodell zur Problemstruktursicht der Anwendungsebene (eigene Darstellung)

4.3.1.5 Problemkarte

Ein Schema auf der Anwendungsebene der Datenanalysearchitektur wird als ***Problemkarte*** bezeichnet. Die Problemkartierung soll alle Belange festhalten, die zum Verständnis des Problems notwendig sind, und alle Schritte benennen, die zu seiner Lösung beitragen. Die Problemkarte repräsentiert demnach Wege zur Problemlösung.[64] Sie wird während des gesamten Projekts fortgeschrieben und dient damit nicht nur zur Orientierung des Vorgehens und zur Ableitung von Analyseproblemen, sondern insbesondere auch zur Dokumentation. Folglich ist die Darstellung des Differenzierungs- und Verknüpfungsgefüges von Problemaspekten explizites Modellierungsziel. Durch Darstellung der Struktur des Problems und möglicher Lösungswege wird

[64] So spricht WILD auch von einer „Problemlandkarte“ als Resultat einer Problemfeldanalyse, die Beziehungen zwischen Problemaspekten abbildet [Wild74, 69]. Sie ist im Sinne einer Karte der kognitiven „Landschaft“ der Problemdomäne zu interpretieren (vgl. auch [Pyle03, 65]).

die Lösungsfindung transparent, nachvollziehbar, kommunizierbar und wiederverwendbar.

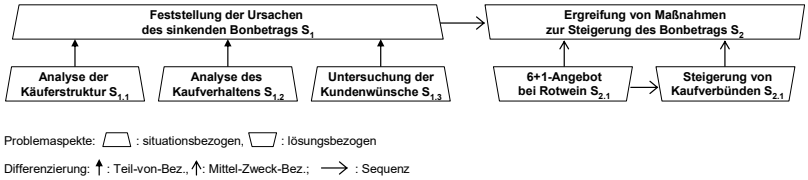


Abbildung 26: Beispiel einer Problemkarte (Problemstruktursicht), in Anlehnung an [NeKn06, 98]

In Abbildung 26 ist eine Problemkarte zum Beispiel des sinkenden Bonbeitrags dargestellt. Die Problemaspekte S_1 und S_2 werden als zur Überbrückung der Diskrepanz eines Sachproblems S sequenziell zu behandelnde Teilprobleme angesehen. Der situationsbezogene Problemaspekt S_1 wird in drei Teilproblemaspekte, der lösungsbezogene Aspekt S_2 in zwei Unterproblemaspekte differenziert.[65] Letztere werden als geeignete Mittel zur Problemlösung erachtet und sind in zeitlicher Folge zu realisieren (Sequenz).

4.3.2 Problemlösungssicht

4.3.2.1 Maßnahme

Die in der Problemstruktursicht vorgenommene Konkretisierung und Differenzierung von Problembestandteilen verfolgt die Absicht, operationale Problemaspekte zu definieren, die einer gezielten Beeinflussung zugänglich sind, um einen Beitrag zur Problemlösung zu erreichen (vgl. [Gait83, 66f.]). Diese Beeinflussung geschieht durch adäquate **Maßnahmen**, die in zweierlei Art ausgeprägt sein können. **Informationsmaßnahmen** dienen der Befriedigung eines durch den Problemaspekt

[65] Die Typunterscheidung der Differenzierungsbeziehungen ist nicht zwingend. Eine Gleichsetzung der Mittel-Zweck-Ordnung mit der Teil-Ganzes-Ordnung kritisiert GAITANIDES jedoch als zu starke Vereinfachung [Gait83, 72f.]. Im Zusammenhang der Problemkartierung wird eine Abstraktion von den Typen als zulässige Vereinfachung erachtet, eine Gleichsetzung der Typen jedoch als nicht zweckmäßig abgelehnt.

aufgeworfenen Informationsbedarfs, **Handlungsmaßnahmen** streben die Ausführung von Handlungen an, welche die Herstellung des vom Problemaspekt beschriebenen Zustands bewirken sollen. Daraus wird deutlich, dass Maßnahmen keine Lösungsverfahren darstellen, sondern vielmehr Lösungsoptionen oder -vorschläge repräsentieren, die noch einer Ausfüllung und Konkretisierung durch Pläne und Anweisungen bedürfen. Informationsmaßnahmen können in vielen Fällen durch Datenanalysen realisiert werden, die auf der Analyseebene zu planen sind. Alternative Formen der Informationsbeschaffung, wie z.B. Literaturrecherchen oder Nachfragen bei fachkundigen Personen, sind nicht Gegenstand der hier vorgestellten Methodik. Ebenso sind Handlungsmaßnahmen separat zu gestalten.

Der Inhalt einer Maßnahme wird durch einen `Namen` und eine `Beschreibung` dokumentiert. Von besonderer Bedeutung für die Maßnahmendurchführung sind organisatorische Rahmenbedingungen, die durch eine Reihe spezifischer Attribute gesetzt werden. Die `Zeitrestriktion` definiert die zur Maßnahmenrealisierung verfügbare Zeit und ist nicht mit dem Zeitbezug des Problemaspekts identisch, da die Knappheit personeller oder finanzieller Ressourcen häufig weit strengere Einschränkungen vorgibt. Sie dient ebenso wie die `Budgetrestriktion`, die den finanziellen Spielraum für die Maßnahme definiert, zur Kontrolle der Effizienz der Maßnahme. Die Verantwortung für die Maßnahme wird durch die Attribute `Organisation` (Organisationseinheit), `Projekt` und `Ansprechpartner` festgelegt. Nach Abschluss einer Maßnahme kann dokumentiert werden, wie ihr `Erfolg` im Ganzen zu beurteilen ist und wie ihre `Bewertung` anhand einzelner Kriterien ausfällt (Evaluierungsergebnis).[66]

Die Zuordnung von Maßnahmen zu Problemaspekten führt die mit der Problemstrukturierung erreichte Annäherung an eine Problemlösung fort und arbeitet den dort skizzierten Lösungsweg gewissermaßen zu einer gangbaren Routenbeschreibung aus, ohne bereits Realisierungsdetails einzelner Maßnahmen zu nennen. Hierzu können in die

[66] Aufgrund der hohen zeitlichen und organisatorischen Spezifität haben Problemaspekte und Maßnahmen Instanzcharakter. Die Ergebnisse einer konkreten Maßnahme sind demnach direkt dort zu hinterlegen.

Problemkarte weitere Verknüpfungen zwischen Problemaspekten eingefügt werden, um die sequenzielle Abfolge der Schritte zur Problemlösung darzustellen. Zur Verdeutlichung ihrer Semantik ist es möglich, Verknüpfungen mit einem `Namen` und einer `Beschreibung` zu versehen.

4.3.2.2 Metamodell

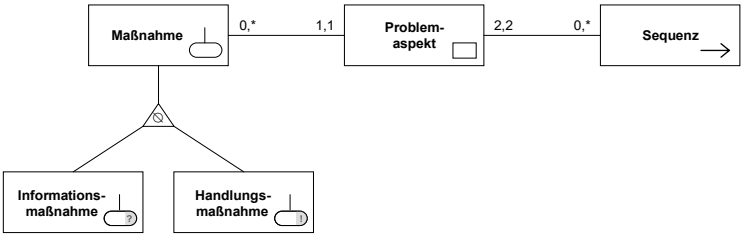


Abbildung 27: Metamodell zur Problemlösungssicht der Anwendungsebene (eigene Darstellung)

In Abbildung 27 ist das Metamodell zur Problemlösungssicht dargestellt. Für jeden **`Problemaspekt`** können mehrere (0,*) **`Maßnahmen`** (abgeflachter Kreis) bestimmt werden, die jeweils eindeutig einem (1,1) Problemaspekt zugeordnet und über ungerichtete Kanten mit diesem verbunden sind. Ihre Ausprägung als **`Handlungsmaßnahme`** oder **`Informationsmaßnahme`** wird durch das Attribut `Typ` gekennzeichnet und grafisch durch ein Ausrufezeichen (!) bzw. ein Fragezeichen (?) symbolisiert. Als Verknüpfung ist in der Problemlösungssicht nur die **`Sequenz`** von Interesse, die jeweils zwei (2,2) Problemaspekte in eine zeitliche oder sachlogische Reihenfolgebeziehung setzt. Problemaspekte können beliebig viele (0,*) sequenzielle Verknüpfungen besitzen. Die Unterscheidung zwischen situations- und lösungsbezogenen Problemaspekten kann vernachlässigt werden.

Abbildung 28 zeigt eine Problemkarte der Problemlösungssicht, die den in Abbildung 26 aufgedeckten Problemaspekten geeignete Maßnahmen zuweist. So sollen die situationsbezogenen Aspekte $S_{1.1}$ und $S_{1.2}$ jeweils durch Datenanalysen, Problemaspekt $S_{1.3}$ durch Marktforschung gehandhabt werden, um benötigte Informationen zu erhalten. Die in der

Problemstruktursicht noch unverknüpften Aspekte erfahren eine Reihung, die bestimmt, dass die Marktforschung erst nach Vorliegen der Ergebnisse von Datenanalyse 1 auszuführen bzw. zu konzipieren ist, und dass Handlungsmaßnahmen erst nach Abschluss aller Informationsmaßnahmen bearbeitet werden. Problemaspekt *S2.1* soll durch eine Werbeaktion gelöst, Aspekt *S2.2* durch eine weitere Werbeaktion und eine umfassende Werbekampagne gehandhabt werden.

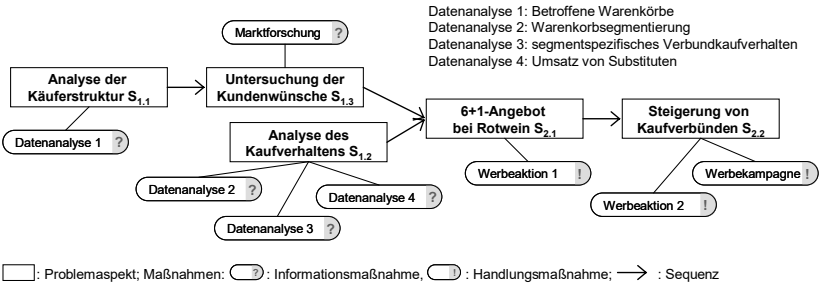


Abbildung 28: Beispiel einer Problemkarte (Problemlösungssicht) (eigene Darstellung)

4.3.3 Bibliothekssicht

Zur Unterstützung der Problemanalyse und Lösungsfindung mithilfe von Erfahrungen aus früheren Projekten können Problemaspekte in einer Wissensbasis abgespeichert und bei Bedarf zum Einsatz in ähnlichen Situationen auf Abruf bereitgestellt werden. Diese Wissensbasis wird auch als ***Fallbibliothek*** bezeichnet und in der *Bibliothekssicht* abgebildet. Auf Anwendungsebene beschränkt sich diese auf den Metaobjekttyp **`Problemaspekt`**, der aufgrund seiner Zuordnung zum Problemobjekt den Ausgangspunkt für die Suche nach Erfahrungswissen bildet. Insbesondere lassen sich ausgehend von einzelnen Problemaspekten alle zugehörigen Maßnahmen abrufen und Problemkarten bzw. Abschnitte von Problemkarten rekonstruieren. Auf die Einzeldarstellung des Metamodells wird wegen der einfachen Struktur der Bibliothekssicht verzichtet (vgl. hierzu Abbildung 30).

4.3.3.1 Problemkennzeichnung (Anwendung)

Der vom Anwender vergebene Name für Modellierungsartefakte ist häufig nicht eindeutig und nicht ausreichend präzise. Um eine einheitliche, intersubjektive Bezeichnung zu erreichen, erhält jeder Problemaspekt eine `Kennzeichnung`, die sich auf ausgewählte Attributsausprägungen stützt und intuitiv verständlich ist. Diese ist von Vorteil, wenn Problemaspekte oder zugeordnete Elemente wie z.B. Analyseprobleme oder -prozesse aus der Bibliothek unkompliziert wieder aufgefunden werden sollen. Eine freie Indizierung mit beliebigen Termen oder Schlagworten aus einem vorgegebenen Vokabular führt typischerweise zu abweichenden Kennzeichnungen für gleichartige oder ähnliche Artefakte durch verschiedene Personen.

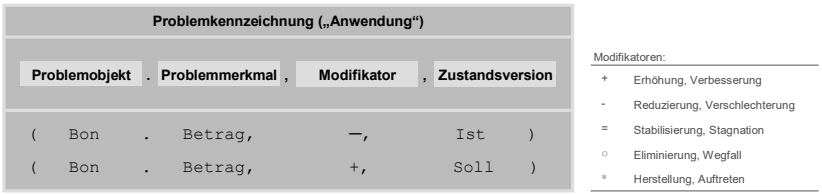


Abbildung 29: Schema und Beispiele zur Problemkennzeichnung („Anwendung”) (eigene Darstellung)

Daher wird zur `Kennzeichnung` das als problematisch erkannte *Merkmal* des betroffenen *Domänenobjekts* in der Notation `Problemobjekt.Problemmerkmal` sowie ein Modifikator und die Zustandsversion verwendet. Die *Zustandsversion* entspricht dem `Typ` des Problemaspekts und nimmt für situationsbezogene Aspekte den Wert „Ist“, für lösungsbezogene Aspekte den Wert „Soll“ an. Der *Modifikator* beschreibt, wie sich das `Ausmaß` des Problemmerkmals verändert hat (Ist-Zustand) bzw. verändern soll (Soll-Zustand). Abbildung 29 zeigt das Kennzeichnungsschema mit gültigen Ausprägungen von Modifikatoren sowie zwei Beispiele. Das erste Beispiel beschreibt einen durch sinkende Bonbeträge charakterisierten situationsbezogenen Problemaspekt, das zweite Beispiel den korrespondierenden lösungsbezogenen Aspekt, der die angestrebte Erhöhung des Einkaufswerts beschreibt. Da diese Problemkennzeichnung den Anwendungskontext von Analysepro-

blemen und -prozessen beschreibt, wird sie im Folgenden auch verkürzt als ***Anwendung*** bezeichnet. Der `Modifikator` wird als eigenständiges Attribut festgehalten. Damit ist die initiale Problemkennzeichnung aus den verfügbaren Attributen ableitbar.

Die Beschreibungselemente stehen in hierarchischer Ordnung, d.h., das Problemobjekt bestimmt den Anwendungsbereich (vgl. `Problemdomäne`), das Problemmerkmal den Anwendungszustand, und der Modifikator weist schließlich auf ein Anwendungsereignis hin, das eine Zustandsänderung repräsentiert, die abhängig von der Zustandsversion entweder eingetreten ist (Ist) oder ausgelöst werden soll (Soll). Somit kann durch stufenweise Abstraktion von den untergeordneten Elementen die von der Kennzeichnung erfasste Klasse von Problemaspekten erweitert (d.h., die mit der Kennzeichnung verbundene Einschränkung relaxiert) werden.

Die `Kennzeichnung` wird auf aus Differenzierung hervorgehende Kind-Problemaspekte vererbt und kann dort nur bezüglich Modifikator und Zustandsversion verändert werden, um aus situationsorientierten Problemaspekten lösungsorientierte Handlungsabsichten abzuleiten. Eine Änderung ist nur einstufig zulässig, d.h., ein Problemaspekt kann nicht die von seinen „Großeltern" vererbte Kennzeichnung verändern. Ein Kind-Aspekt kann jedoch zusätzlich zu den geerbten Anwendungen maximal eine weitere Kennzeichnung definieren. Diese Einschränkungen dienen dazu, die Ausdrucksmächtigkeit von Problemaspekten zu gewährleisten. Änderungen der Problemkennzeichnung sind prinzipiell zu vermeiden und sollten nur dann vorgenommen werden, wenn dies tatsächlich notwendig ist. So sind Problemaspekte denkbar, die unabhängig von ihren Vorfahren auch in anderen Kontexten relevant sind. Beispielsweise ist ein Problemaspekt „Qualitätsprobleme beheben", der ursprünglich zur Behebung von Absatzproblemen aufgedeckt wurde, auch außerhalb dieser Domäne von allgemeiner Bedeutung für das Qualitätsmanagement.

4.3.3.2 Maßnahmenbeschreibung

Zum gezielten Auffinden erfolgreicher Maßnahmen, die bei der Planung künftiger Vorhaben hilfreich sein können, sind Informationen

über das `Unternehmen`, die `Branche` und den `Betriebstyp` nützlich, für die die Maßnahme entwickelt und durchgeführt wird. Während Problemaspekte grundsätzlich für alle Ausprägungen dieser Dimensionen gelten können (bzw. im Falle eines Nichtzutreffens schlicht nicht modelliert bzw. abgerufen werden),[67] sind geeignete Maßnahmen in hohem Maße vom organisatorischen Kontext abhängig, wie er von diesen Attributen beschrieben wird. So ist die Behandlung von Problemen der Produktqualität z.B. von der Branche abhängig, weshalb eine zugehörige Handlungsmaßnahme aus dem Maschinenbau typischerweise nicht bei Versorgungsschwankungen bei einem Energienetzbetreiber geeignet ist. Maßnahmen, die bezüglich eines der Attribute allgemeingültig sind, können durch einen entsprechenden Attributwert explizit als solche gekennzeichnet werden.

4.3.3.3 Rekonstruktion von Problemstrukturen und Lösungsoptionen

Die Rekonstruktion von Problemkarten oder Problemkartenausschnitten (Problemstrukturen) aus einzelnen Problemaspekten erfordert eine Reihe spezifischer Beziehungsattribute. Jeder Problemaspekt speichert die Menge der mit ihm sowohl in der Problemstruktursicht als auch in der Problemlösungssicht verknüpften Objekte. So verfügt mit Ausnahme der Gesamt- oder Überprobleme höchster Ebene jeder Problemaspekt über einen `Vateraspekt`, aus dem er durch Differenzierung abgeleitet ist. Analog besitzen mit Ausnahme der Teil- oder Unterprobleme auf der Blattebene alle Problemaspekte eine Menge von `Kindaspekten`, die aus ihnen durch Teil-von- oder Mittel-Zweck-Beziehungen hervorgehen. Aus den sequenziellen Verknüpfungen ist zu jedem Problemaspekt ferner die Menge eingehender `Vorgänger` und ausgehender `Nachfolger` bekannt. Mithilfe dieser Informationen lassen sich ausgehend von einem Problemaspekt komplexe Problemstrukturen rekonstruieren, indem die jeweiligen Verknüpfungsbezie-

[67] Da z.B. das Problem der Kundenabwanderung in der öffentlichen Verwaltung nicht relevant ist, wird ein entsprechender Problemaspekt für diesen Betriebstyp nicht als Bestandteil einer Problemkarte auftreten. Analog wird dieser Problemaspekt für Projekte in der öffentlichen Verwaltung aufgrund mangelnder Relevanz auch nicht aus einer Wissensbasis abgerufen.

hungen in alle vier „Richtungen“ schrittweise für jeden als Verknüpfungsziel auftretenden Aspekt verfolgt werden. Ebenso ist zu jedem Problemaspekt die Menge der zugeordneten `Maßnahmen` ersichtlich, sodass auch die Rekonstruktion der Problemlösungssicht möglich ist.

4.3.4 Zusammenfassung zur Anwendungsebene

Problemstruktursicht, Problemlösungssicht und Bibliothekssicht bilden eine vollständige Beschreibung der Anwendungsebene. Das integrierte Metamodell zeigt Abbildung 30. Die genannten Attribute der Metaobjekttypen stehen in allen drei Sichten zur Verfügung. Eine ausführliche Zusammenstellung mit Erläuterungen, Datentypen und Kardinalitäten der Attribute findet sich in Anhang A4.1.

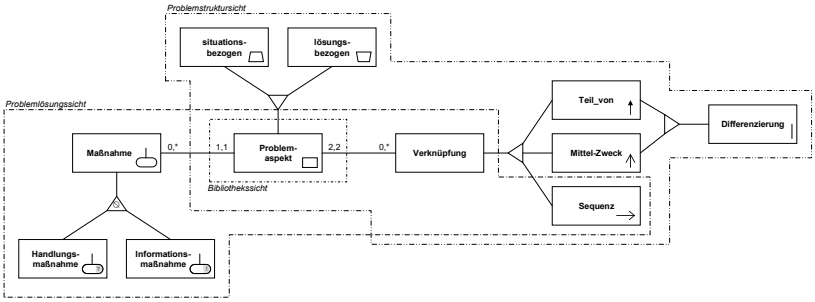


Abbildung 30: Integriertes Metamodell zur Anwendungsebene (eigene Darstellung)

Die Sichtweise, die der Modellierer bei der Erfassung des Objektsystems einnimmt und bei der Spezifikation des Modellsystems zugrunde legt, wird in der Modellierung als *Metapher* beschrieben. Sie bildet zusammen mit dem Metamodell den Gestaltungsrahmen für die Modellierung (Modellierungsansatz) [FeSi13, 136]. Die Metapher für die Anwendungsebene ist die der *Problemkarte*, die Gestalt, Gliederung und Verknüpfung von Problemen bzw. Problemaspekten visualisiert (Problemstruktursicht) und Lösungsoptionen aufzeigt (Problemlösungssicht). Sie bildet eine „mentale Landkarte“, mit deren Hilfe sich der Anwender im Problemraum zurechtfinden, mögliche Wege erkennen und eine Route zur Lösung planen kann.

4.4 Analyseebene: Ziel und Gegenstand der Datenanalyse

Informationsmaßnahmen der Anwendungsebene, die zur Deckung eines Informationsbedarfs vorgesehen sind, erfahren auf der Analyseebene eine Konkretisierung in Form von Analysezielen bzw. Analyseproblemen. Handlungsmaßnahmen finden im Folgenden keine weitere Beachtung. Eine detaillierte Beschreibung des Informationsbedarfs als Spezifikation des zu erreichenden Analyseergebnisses (Output) erfolgt in der *Informationsbedarfssicht* (kurz: *Zielsicht*) in Gestalt eines ***Analyseziels***. Sie kann ohne Berücksichtigung von Analysedaten mit Vertretern der Fachabteilung (Informationsempfänger) entwickelt werden und betrachtet im Sinne eines Fachkonzepts ausschließlich die Aufgabenebene. In der *Informationserzeugungssicht* (kurz: *Problemsicht*) werden zusätzlich inhaltliche und formale Anforderungen an zur Erzeugung der gewünschten Ergebnisse geeignete Analysedaten erhoben. Diese nehmen teilweise auf konkrete Datenquellen Bezug und werden häufig zusammen mit Datenverantwortlichen ausgearbeitet.[68] Diese Spezifikation des Analyse-Inputs wird als ***Analyseobjekt*** bezeichnet, das dem Analyseziel beigefügt wird, woraus ein ***Analyseproblem*** resultiert. Ein Analyseproblem repräsentiert ein modellgestütztes Untersuchungsproblem der Datenanalyse (vgl. Abschnitt 4.2.1). Erfordert die Deckung eines Informationsbedarfs mehrere Analysen, oder ergeben sich aus Analyseergebnissen situativ neue Fragestellungen, können mehrere Analyseziele bzw. -probleme sequenziell verknüpft werden. Derartige Analyseketten bildet die *Verkettungssicht* ab, mit deren Hilfe komplexe Analysestrategien darstellbar sind. Zu Zwecken der Wiederverwendung und Erfahrungssicherung zu speichernde Artefakte repräsentiert wiederum eine *Bibliothekssicht*.

4.4.1 Informationsbedarfssicht (Zielsicht)

Das **`Analyseziel`** spezifiziert Anforderungen an das angestrebte Analyseergebnis und richtet sich auf unbekannte Eigenschaften eines

[68] Diese Aussage gilt im Allgemeinen für Sekundärdaten. Primärdaten, die eigens für die geplante Analyse erhoben werden, können ohne Bezug zur Aufgabenträgerebene spezifiziert werden.

Domänenobjekts. Das betreffende `Domänenobjekt` entspricht dem Untersuchungsobjekt, und als interessierende Eigenschaft in der Rolle des Untersuchungsziels wird primär ein `Merkmal` dieses Domänenobjekts gekennzeichnet. Untersuchungsobjekt und -ziel sind in der Regel zunächst mit dem Inhalt jenes Problemaspekts identisch, dem auf Anwendungsebene die Informationsmaßnahme zugeordnet ist, aus welcher das Analyseziel abgeleitet ist. Im Zuge einer Verfeinerung des Analyseziels kann sich diese Korrespondenz auflösen. Insbesondere ist es hierbei auch möglich, ein initiales Analyseziel durch mehrere Analyseziele zu konkretisieren, die sodann gemeinsam mehr als eine interessierende Eigenschaft betrachten.

Wichtiger Bestimmungsfaktor der zu produzierenden Informationen stellt die `Ausrichtung` der Analyse als explorativ (Induktion, $\triangle$), konfirmatorisch (Verifikation, $\triangledown$) oder schließend (Inferenz, $\triangleright$) dar (vgl. Abschnitt 2.2.1). Sie kennzeichnet die gewünschten Analyseergebnisse als hypothetische Aussagen, als Überprüfung von Aussagen bzw. als Prognosen. Die Ausrichtung ist für die spätere Auswahl geeigneter Analyseansätze relevant und kann durch die gezeigten Symbole visualisiert werden. Erläuterungen zum Analyseziel lassen sich in einer `Beschreibung` dokumentieren. Die inhaltlichen und formalen Anforderungen an die Analyseergebnisse werden durch spezifische Repräsentationselemente abgebildet.

4.4.1.1 Analysefrage

Der gewünschte Inhalt der benötigten Informationen nimmt auf konzeptuelle Aussagen über Objekte und Problemaspekte der Anwendungsdomäne Bezug. Diese enthalten häufig theoretische Konstrukte (z.B. „Käuferstruktur“, „Kaufverhalten“), die einer direkten Behandlung mithilfe der Datenanalyse nicht zugänglich sind, sondern zu diesem Zweck in empirische Aussagen transformiert werden müssen. Diese Transformation wird in Statistik und empirischer Forschung als *Operationalisierung* bezeichnet (vgl. Abschnitt 3.1.2.1) und entspricht der Übersetzung des Untersuchungsziels in der modellgestützten Untersu-

chungssituation.[69] Operationalisierung führt häufig zu mehreren empirischen Aussagen, die als Indikatoren für ein theoretisches Konstrukt in Frage kommen [Drei94, 77]. Das Kaufverhalten kann u.a. durch Kauffrequenzen, Umsatzanteile einzelner Warengruppen oder Verbundkäufe gemessen werden. Das Ergebnis der Operationalisierung wird in Form eines oder mehrerer Analyseziele dokumentiert und jeweils im Attribut `Analysefrage` gespeichert. Die Formulierung einer *Frage in natürlicher Sprache* stellt den direktesten Weg der menschlichen Informationsbeschaffung dar und ist intuitiv sowie unmittelbar kommunizierbar [Hogl03, 31]. Sie eignet sich zur Beschreibung der Inhalte, welche die operationalisierten Aussagen liefern sollen, und wird als `Analysefrage.Fragetext` abgelegt.

Der präzisen Dokumentation der Informationsinhalte dienen weitere Beschreibungselemente der `Analysefrage`. Zu deren Herleitung tragen folgende Überlegungen von McGuff bei: „Think of facts as basic pieces of information that users want to see in the answer to a question, and dimensions as one way for the users to constrain the scope of the question" [McGu98]. *Fakten* entsprechen den (operationalisierten) Merkmalen als Frageziel, die im Allgemeinen sowohl qualitativer als auch quantitativer Natur sein können. *Dimensionen* definieren den Geltungsbereich der Informationen und können einschränkenden oder beschreibenden Charakter aufweisen. Im ersten Fall dienen sie als Filter, der unter anderem die zu betrachtenden Domänenobjekte näher bestimmt, im zweiten als Bezugspunkte für die abzufragenden Merkmalswerte.[70] Als beschreibende Dimensionen kommen qualitative oder quantitative Größen in Frage. Beispiele für qualitative Beschreibungsdimensionen zur Untersuchung des Kaufverhaltens sind Artikel, Warengruppen, Filialen oder Wochentage, nach denen Einkäufe aufgeschlüsselt werden können. Als quantitative Beschreibungsdimensionen können z.B. Warenwert oder Deckungsbeitrag der Artikel dienen. Hierdurch werden

---

[69] Hinweise zur Theorie und Praxis der Operationalisierung gibt Abschnitt 5.4.5.1.

[70] Die Differenzierung von Fakten und Dimensionen ist im OLAP gebräuchlich, wird hier aber bewusst verallgemeinert, um Hinweise für die Frageformulierung zu erarbeiten.

die Fakten in Vergleichs- oder Verhältnisbeziehungen[71] gesetzt, um den Aussagen mehr Gehalt zu verleihen [Küpp05, 176].

Eine ausführliche Analyse der Semantik von Fragen unternimmt HOGL [Hogl03], um daraus eine Anfragesprache für die Wissensentdeckung zu entwickeln, die auch Fachexperten zugänglich ist. Im Ergebnis identifiziert er als Strukturkomponenten von Fragen, die analog auch für die zugehörigen Antworten zutreffen, (1) den *Fragetyp*, der mit der `Ausrichtung` der Analyse korrespondiert; (2) das *Frageobjekt*, das den Typ der als Antwort zu liefernden Aussage (z.B. Zusammenhang) beschreibt und durch (3) *Frageargumente* näher bestimmt wird (etwa zwischen welchen Objekten ein Zusammenhang existiert). Optional kann die Aussage eingeschränkt werden durch (4) die *Fragegruppe*, die ein Gruppierungskriterium definiert, bezüglich dessen die betreffenden Objekte denselben Merkmalswert aufweisen müssen, sowie (5) den *Fragekontext*, der eine Menge allgemeiner Selektionskriterien zulässt [Hogl03, 53-71]. Dem Ansatz liegt die Annahme zugrunde, dass jede Frage einer definierten Fragekategorie angehört, für die jeweils eindeutige Beantwortungsstrategien existieren [Hogl03, 40]. Für die automatisierte Instanziierung von Analyseverfahren ist die Vorgabe einer Fragegrammatik zwar erforderlich. Inwiefern eine solche Einschränkung für den Einsatz in der betrieblichen Praxis tauglich ist, lässt selbst der Autor offen [Hogl03, 119ff.]. Gleichwohl gibt eine derartige Strukturierung wichtige Hinweise für die Frageformulierung, weshalb diese Überlegungen aufgegriffen und weiterentwickelt werden.

Hierzu wird zunächst das Frageobjekt verständlicher als `Analysefrage.Aussagetyp` interpretiert und untersucht, welche Ausprägungen dieses Attribut annehmen kann. Diese ergeben sich aus den grundlegenden Ergebnistypen der Datenanalyse (vgl. [FrPM91], [FaPS96], [Hogl03, 61-63]): *Zusammenhänge* repräsentieren Beziehungen, Abhängigkeiten und Gesetzmäßigkeiten, die auch zur Erklärung oder Vorhersage (Klassifizierung, Regression) dienen können. *Unterschiede* sind in Diskriminierungsregeln darstellbar, die wiederum

[71] Verhältnisbeziehungen zwischen quantitativen Merkmalen werden als Verhältniszahlen ausgedrückt, die als Beziehungs-, Gliederungs- oder Indexzahlen auftreten [Küpp05, 359f.].

zur Klassifizierung von Objekten nutzbar sind. Objekte lassen sich auf Grundlage ihrer *Gemeinsamkeiten* (Ähnlichkeiten) zu Gruppen oder Segmenten zusammenfassen. *Veränderungen* betrachten Objektmerkmale zu verschiedenen Zeitpunkten und sind zum Vergleich zwischen Merkmalsversionen oder mit anderen Größen (z.B. Soll-, Norm- oder Mittelwerte) um *Abweichungen* zu ergänzen. In der Statistik, im Berichtswesen und im OLAP sind zusätzlich verschiedene *Zusammenfassungen* von Merkmalswerten üblich, z.B. in Form von Summen, Durchschnitten, statistischen Maßzahlen [Ehre76, 24f., 34] oder domänenspezifischen Kennzahlen. Beim Datenabruf werden häufig auch *Einzelwerte* von Objekten abgefragt.

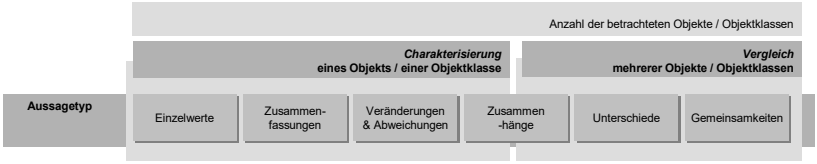


Abbildung 31: Aussagetypen der Datenanalyse (eigene Darstellung)

Abbildung 31 fasst die genannten ***Aussagetypen*** in einer Systematisierung zusammen. Sie unterscheidet Aussagen danach, ob sie einzelne oder mehrere Objekte bzw. Objektklassen charakterisieren, d.h. nach der Anzahl der betrachteten Merkmalsträger. Objektklassen können Gesamt- oder Teilpopulationen der Extension von Domänenobjekttypen sein, die jeweils nach bestimmten Kriterien ausgewählt werden, wie z.B. alle Kunden, Käufer bestimmter Artikel, oder Aufträge mit einem Mindestvolumen aus einem definierten Zeitraum. Demnach beschreiben Einzelwerte, Zusammenfassungen, Veränderungen & Abweichungen einzelne Objekte bzw. Objektklassen. Unterschiede und Gemeinsamkeiten dienen dem Vergleich mehrerer Merkmalsträger. Zusammenhänge können sich auf Merkmale sowohl eines als auch mehrerer Objekte erstrecken.[72] So kann etwa eine Beziehung zwischen der Zufriedenheit eines Kunden und den Eigenschaften der von ihm gekauften Produkte bestehen.

[72] Vgl. hierzu die Unterscheidung zwischen *interfield patterns* und *inter-record patterns* bei [FrPM91, 12].

Inferenzen und die zugehörigen Modelle stellen keinen eigenen Aussagetyp dar, sondern werden durch die `Ausrichtung` der Analyse modelliert. Letztlich treffen Prognosen und Klassifikationen stets Aussagen über unbekannte Einzelwerte von Objekten (z.B. Kundenwert, Antwortwahrscheinlichkeit, Klassenzugehörigkeit).[73] Explorative und konfirmatorische Untersuchungen eigenen sich zur Erzeugung aller Aussagetypen; die Ausrichtung steht demnach grundsätzlich orthogonal zum Aussagetyp. Domänenspezifische Konzepte [Hogl03, 63f.] finden keine Berücksichtigung als Aussagetyp, da sie bereits zuvor in geeignete Merkmale operationalisiert werden. Häufig erfolgt die Verknüpfung mehrerer Aussagen unterschiedlichen Typs in einer Analyse. Typische Vertreter kombinierter Auswertungen sind das betriebliche Berichtswesen und OLAP. Die produzierten Berichte stellen zweckorientiert zusammengefasste Daten bereit [Küpp05, 170].

Vor diesem Hintergrund erweisen sich die oben eingeführten Frageelemente *Fakten* und *Dimensionen* als geeignetes Strukturierungsprinzip, das Fragen nach einzelnen Aussagetypen ebenso aufnimmt wie solche nach Berichten als Aggregate von Einzelaussagen. Die Fakten werden durch einen der eingeführten *Aussagetypen* sowie durch *Aussageargumente* (`Analysefrage.Argumente`) beschrieben. Letztere repräsentieren jene Objektmerkmale, operationalisierten Begriffe oder Kennzahlen, über die empirische Aussagen gewünscht werden. Die Dimensionen werden in Beschreibungs- und Selektionsdimensionen differenziert.[74] *Beschreibungsdimensionen* können qualitativ (Bezugsobjekte), quantitativ (Vergleichs- und Verhältnisgrößen) oder konfigurativ (Ordnungs- oder Sortierkriterien) sein. *Selektionsdimensionen* wählen Objekte bzw. Objektklassen, die anhand sachlicher, zeitlicher und räumlicher Kriterien weiter eingeschränkt werden können. Mithilfe dieser Systematik lassen sich Fragen nach allen Aussagetypen

[73] Der Analyseebene liegt eine strikt aufgabenträgerunabhängige Betrachtung zugrunde. Erfordert der eingesetzte Analyseansatz z.B. zur Berechung der Vorhersage ein Prognosemodell, kann diese Anforderung auf Prozessebene nach Auswahl eines entsprechenden Verfahrens in Form einer mehrstufigen Analyseaufgabe modelliert werden.

[74] Die zugehörigen Attribute heißen `Analysefrage.Beschreibungsdimensionen` und `Analysefrage.Selektionsdimensionen`.

formulieren, wobei die Dimensionen nicht obligatorisch sind. Aussagetyp und mindestens ein Aussageargument sind erforderlich. Abbildung 32 zeigt die Komponenten der Fragestruktur und illustriert ihre Anwendung anhand eines Beispiels, das einen einfachen Bericht zur Entwicklung der vom sinkenden Bonbetrag betroffenen Warenkörbe spezifiziert.

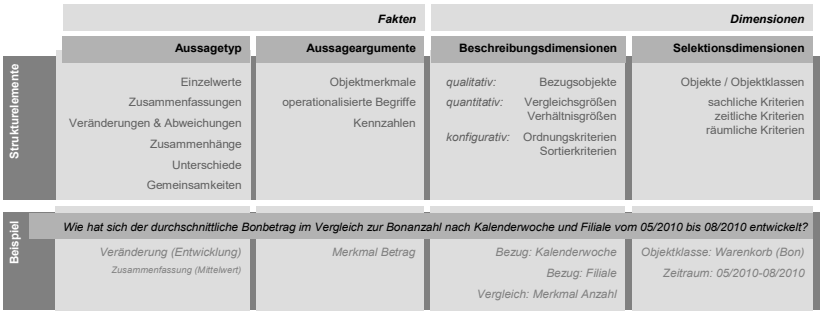


Abbildung 32: Komponenten der Fragestruktur mit Beispiel (eigene Darstellung)

4.4.1.2 Informationsbedarfsprofil

Um die Eignung der durch die Analysefrage inhaltlich beschriebenen Analyseergebnisse im gegebenen Anwendungskontext zu gewährleisten, sind zusätzlich formale Anforderungen an die zu liefernden Informationen zu definieren, die sich in die Klassen Art, Qualität, Menge und Nutzen gliedern lassen. Sie werden als Menge von Deskriptoren in Form von Schlüssel/Wert-Paaren spezifiziert, die gemeinsam das `Informationsbedarfsprofil` der Datenanalyse bilden. Ein Katalog möglicher Deskriptoren ist in Anhang A5.1 enthalten, und Hinweise zur Wahl der Anforderungen finden sich in Abschnitt 5.4.5.1. An dieser Stelle seien lediglich einige Beispiele angeführt. Für oben genannte Analysefrage nach dem Bonbetrag sind z.B. die Deskriptoren `Aussageform: faktisch`, `Repräsentationsform: Tabelle` und `Aggregationsgrad: Filiale` denkbar, die einen Bericht in verständlicher, gut lesbarer Form auf Basis von Ist-Werten (faktischen Informationen) fordern und die Granularität auf die bereits in der Analysefrage enthaltene Bezugsgröße präzisieren.

4.4.1.3 Metamodell

Die Informationsbedarfssicht umfasst als einzigen Metaobjekttyp das **`Analyseziel`**, das in Abbildung 33 mit seinen wichtigsten Attributen im Metamodell dargestellt ist. Die Gesamtheit der Attribute dient der präzisen Beschreibung der von der Analyse zu liefernden Information. Hierzu wird mit einem (1,1) `Domänenobjekt` und einem (1,1) zugehörigen `Merkmal` das Untersuchungsziel in konzeptuellen Begriffen repräsentiert, das durch genau eine (1,1) `Analysefrage` operationalisiert wird. Die Analysefrage wird in Form eines (1,1) `Fragetexts` in natürlicher Sprache ausformuliert und durch mindestens einen (1,*) `Aussagetyp` und mehrere (1,*) `Argumente` definiert. Zusätzlich können beliebig viele (0,*) `Beschreibungsdimensionen` und `Selektionsdimensionen` angegeben werden. Die Analyse wird ferner durch eine (1,1) `Ausrichtung` typisiert. Im `Informationsbedarfsprofil` können beliebig viele (0,*) formale Anforderungen an die Analyseergebnisse gestellt werden. Zur grafischen Repräsentation eines Analyseziels dient ein abgerundetes Rechteck, das im oberen Teil dessen frei definierbaren Namen zeigt. Durch eine Doppellinie abgetrennt werden im unteren Teil in der ersten Zeile Ausrichtung und Merkmal, in der zweiten Zeile das Domänenobjekt benannt, die zusammen eine einheitliche Charakterisierung der repräsentierten Untersuchung ergeben. So beschreibt das in Abbildung 33 rechts als Beispiel gezeigte Analyseziel mit dem Namen *Betroffene Warenkörbe* eine explorative (△) Analyse des *Betrags* von *Warenkörben (Bons)*.

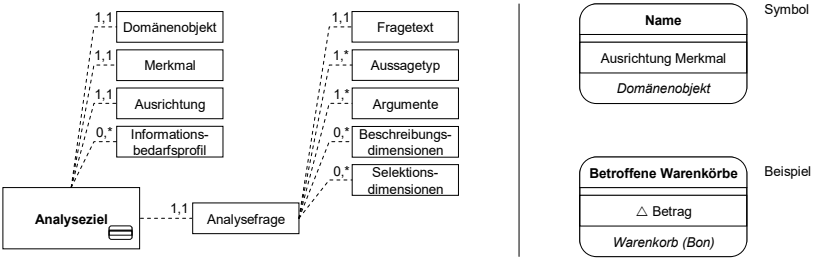


Abbildung 33: Metamodell zur Informationsbedarfssicht (Analyseziele) der Analyseebene mit Symbol und Beispiel (eigene Darstellung)

4.4.2 Informationserzeugungssicht (Problemsicht)

Zur Deckung des im Analyseziel definierten Informationsbedarfs ist eine Beschreibung geeigneter Analysedaten erforderlich, aus denen die gewünschte Information erzeugt werden soll. Diese Beschreibung erfolgt durch Spezifikationselemente, die das Analyseziel zu einem **`Analyseproblem`** erweitern. Dieses Analyseproblem definiert eine auszuführende Datenanalyse und ist als Problem im Sinne einer Barriere bezüglich der Transformation von Daten in Informationen zu verstehen. Die Bestimmung geeigneter Analysedaten erfolgt in Form eines *Analyseobjekts* nach Maßgabe der Analysefrage und wird durch Vorgabe einer *Perspektive* gelenkt, die den Blickwinkel darstellt, aus dem die Untersuchung erfolgen soll.

4.4.2.1 Perspektive

Das mit dem Analyseziel ausgedrückte Interesse an einem Sachverhalt kann grundsätzlich aus mehreren Blickwinkeln befriedigt werden, woraus eine 1:n-Beziehung zwischen Analyseziel und Analyseproblemen resultiert und sich vielfältige Möglichkeiten für die Betrachtung eines Sachverhalts eröffnen. Unterschiedliche Blickwinkel ergeben sich aus der Tatsache, dass die Datenanalyse stets nur ein Abbild des eigentlichen Untersuchungsobjekts bearbeitet. Der Modellcharakter der Daten impliziert, dass Ziele und Unzulänglichkeiten der Modellabbildung (Datenerhebung bzw. -erfassung) den Aussagegehalt der Daten beeinflussen und zu Adäquations- und Datenqualitätsproblemen führen können (vgl. Abschnitt 3.1.2.2). Die Bestimmungsmerkmale der Datenerfassung legen gleichzeitig die `Perspektive` fest, aus der das Untersuchungsobjekt beschrieben wird, denn die Stelle der Datenerhebung (Erhebungseinheit bzw. -objekt) stimmt häufig nicht mit dem Untersuchungsobjekt überein (vgl. [HeMi94, 49]). So wird die Bonität eines Kunden typischerweise nicht bei diesem selbst erhoben, sondern von Auskunfteien wie etwa der SCHUFA bei verschiedenen Unternehmen abgefragt. Auch die Debitorenbuchhaltung verfügt über Daten zur Kundenbonität, die sich jedoch auf Erfahrungen des eigenen Unternehmens beschränken. Die Perspektive determiniert also die sichtbaren Eigenschaften eines Untersuchungsobjekts. Das nahe-

liegende Beschreibungselement für die Perspektive bildet demnach jenes Domänenobjekt, das als *Erhebungsobjekt* für geeignete Daten auftritt. Abbildung 34 illustriert diesen Zusammenhang allgemein und am Beispiel des Untersuchungsobjekts Kunde, zu dem mehrere interne und externe Erhebungsobjekte mit jeweils eigenen Perspektiven verfügbar sind. Die Domänenobjekte sind als Diskurswelt- bzw. Umweltobjekte in einem Interaktionsschema gemäß Semantischem Objektmodell (SOM) [FeSi13, 194ff.] dargestellt.

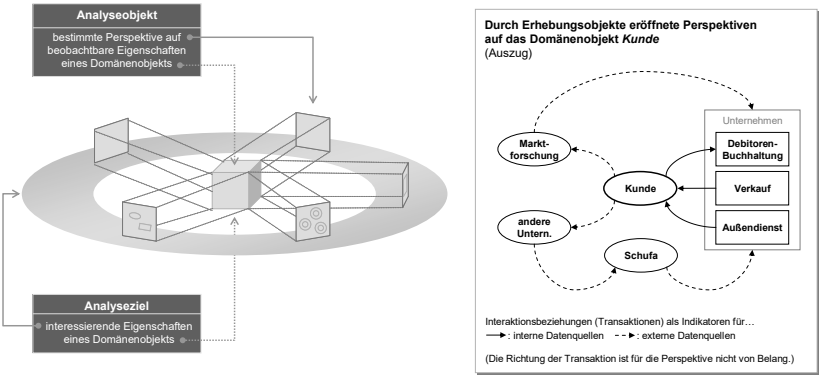


Abbildung 34: Perspektiven auf das Untersuchungsobjekt: Grundprinzip und Beispiel (eigene Darstellung)

Die Erhebungsobjekte verweisen häufig unmittelbar auf zugehörige *Datenquellen*. In Abbildung 34 sind Interaktionen zwischen Untersuchungsobjekt und Diskursweltobjekten als Indikatoren auf interne, solche mit Umweltobjekten auf externe Datenquellen markiert. Interne Datenquellen sind typischerweise spezielle Anwendungssysteme, Datenbanken oder Data Warehouses, externe Datenquellen können z.B. als Web Services oder Webseiten auftreten, sind in vielen Fällen aber nicht näher bestimmt. Werden Daten von der erhebenden Stelle als Dokumente bereitgestellt, können interne Ablagesysteme als Quelle angegeben werden.[75] Die Datenquelle ist dann im Sinne eines Ablageorts zu verstehen, an dem die Daten im Bedarfsfall, etwa für weitere

[75] Dies gilt insbesondere auch für Primärdaten, die häufig in Berichtsform geliefert werden.

Untersuchungen oder zur Überprüfung, zur Verfügung stehen. Die Auswahl konkreter Datenquellen kann durch das *Datenquellenprofil* unterstützt werden, das Eigenschaften einer Quelle benennt (vgl. Abschnitt 4.6.3.2). Das Vorgehen bei der Datenquellenwahl erläutert Abschnitt 5.4.5.2.

Die Spezifikation der `Perspektive` erfolgt als Attribut des Analyseproblems und kann jeweils mehrere `Erhebungsobjekte` und zugehörige `Datenquellen` umfassen. In vielen Fällen ist eine umfassende Perspektive auf das Untersuchungsobjekt nötig, die sich aus einer Reihe von Datenquellen verschiedener Erhebungseinheiten zusammensetzt.[76] Bei internen Erhebungsobjekten kann zusätzlich der `Geschäftsprozess` benannt werden, dessen Durchführung die Perspektive repräsentiert. Erhebungsobjekt, Typ der Datenquelle[77] und Zweck der Datenerfassung bestimmen maßgeblich die repräsentierte Perspektive auf das Untersuchungsobjekt. Zur Identifikation wird der Perspektive ein `Name` zugeschrieben, der sich aus `Erhebungsobjekten` und `Datenquellen` zusammensetzt.[78] Im laufenden Beispiel ist die Perspektive des Verkaufs am Point of Sale (POS) einzunehmen, um die Einkäufe zu untersuchen.

4.4.2.2 Analyseobjekt

Die Spezifikation der Inhalte geeigneter Analysedaten zur Erreichung eines Analyseziels wird als *Analyseobjekt* bezeichnet. Diese griffige Bezeichnung reflektiert die durch dieses Konstrukt repräsentierte Spezifikation eines Modells des Untersuchungsobjekts in der modellgestützten Untersuchungssituation, ist jedoch nicht mit den

---

[76] In der Praxis wird häufig eine „Rundumsicht“ oder „360°-Sicht“ auf ein Untersuchungsobjekt angestrebt.

[77] Das Attibut `Datenquellentyp` wird an der **`Datenquelle`** gepflegt (vgl. Abschnitt 4.6.3.2).

[78] Daten aus mehreren Quellen werden nach ihrer Zusammenführung typischerweise in einem gemeinsamen Datenspeicher abgelegt, so dass eine gemeinsame Datenquelle für mehrere Erhebungsobjekte denkbar ist.

Analysedaten gleichzusetzen. Die Bereitstellung der Daten erfolgt erst im Rahmen der Prozessdurchführung auf Prozessebene.

Das `Analyseobjekt` beschreibt verfügbare oder beschaffbare Daten, die empirische Aussagen über das Untersuchungsobjekt treffen und geeignet sind, die Beantwortung der Analysefrage zu unterstützen. Die Herkunft dieser Daten ist durch die Perspektive hinreichend bestimmt. Ihre Spezifikation erfolgt durch Benennung einer Reihe von Informationsobjekten, die den gewünschten Aussagegehalt liefern. *Informationsobjekte* sind Datenobjekte, die abhängig von ihrem Typ z.B. als Relationen, XML-Dateien oder Dokumente ausgeprägt sind und auf der Ressourcenebene der Datenanalysearchitektur (Abschnitt 4.6.3.1) erläutert werden. Für die Zwecke des vorliegenden Abschnitts werden der Einfachheit halber relationale Informationsobjekte angenommen. Eine Relation eignet sich zur Beantwortung der Analysefrage, wenn ihre Attribute die dort bestimmten empirischen Aussagen enthalten, stützen oder deren Ableitung, z.B. durch Berechnung, erlauben. Gerade bei relationalen Datenquellen sind häufig mehrere Informationsobjekte erforderlich, um den Datenbedarf zu decken bzw. um eine vollständige Beschreibung des Untersuchungsobjekts aus der gewünschten `Perspektive` zu erreichen. Konkrete Hinweise zur Auswahl der Informationsobjekte gibt Abschnitt 5.4.5.2.

Im Falle verfügbarer Daten besteht das `Analyseobjekt` aus einer Aufzählung der betreffenden Informationsobjekte. Im Beispiel des Bonbetrags sind etwa die drei Relationen {POS.BON, POS.BONPOSITION, POS.ARTIKEL}[79] zu wählen, die alle relevanten Angaben zu Einkäufen (Bons) und den betroffenen Artikeln enthalten. Im Falle noch zu erhebender Daten werden passende Informationsobjekte zunächst definiert, indem z.B. neue Relationstypen deklariert oder Fragebogen entworfen werden, die jeweils die gewünschten Merkmale enthalten. Ihre Definition kann sodann als Grundlage für die Datenerhebung dienen.

[79] Die Bezeichnung der Informationsobjekte folgt dem Schema <Datenquelle>.<Informationsobjekt>. Abhängig von der Quelle können weitere Qualifikatoren wie Schema- oder Datenbankname enthalten sein.

4.4.2.3 Metamodell

Das Metamodell zur Informationserzeugungssicht (Abbildung 35) umfasst den Metaobjekttyp **`Analyseproblem`**, der als Spezialisierung des **`Analyseziels`** dessen Attributmenge um Elemente zur Beschreibung zu verwendender Analysedaten erweitert. Dazu wird genau eine (1,1) `Perspektive` definiert, die mindestens ein (1,*) `Erhebungsobjekt` und mindestens eine (1,*) zugehörige `Datenquelle` benennt. Zusätzlich können mehrere (0,*) `Geschäftsprozesse` angegeben werden. Eine griffige, aus `Erhebungsobjekten` und `Datenquellen` zusammengesetzte Kurzbezeichnung der repräsentierten `Perspektive` ergibt das Attribut `Name` (1,1). Jedes **`Analyseproblem`** verfügt über ein `Analyseobjekt`, das auf Grundlage der `Perspektive` spezifiziert wird und aus mindestens einem (1,*) Element besteht.

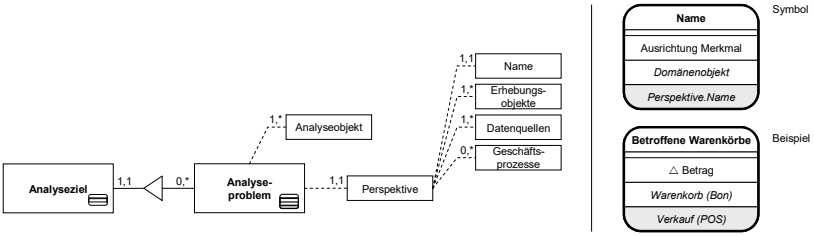


Abbildung 35: Metamodell zur Informationserzeugungssicht (Analyseprobleme) der Analyseebene mit Symbol und Beispiel (eigene Darstellung)

Die grafische Darstellung eines **`Analyseproblems`** erfolgt als abgerundetes Rechteck mit gegenüber dem Analyseziel breiteren Kanten, dessen obere drei Zeilen denen des Analyseziels entsprechen, wobei die Ausprägungen des Analyseproblems verwendet werden. Die vierte, grau schattierte Zeile enthält den `Namen` der `Perspektive`. Diese Elemente gestatten eine prägnante, einheitliche Charakterisierung der auszuführenden Analyse. Das in Abbildung 35 rechts gezeigte Beispiel *Betroffene Warenkörbe* beschreibt ein Analyseproblem zum gleichnamigen Analyseziel als explorative (△) Untersuchung des *Betrags* von *Warenkörben (Bons)* aus der Perspektive *Verkauf (POS)* (d.h., aus Sicht der Datenquelle *POS* des Erhebungsobjekts *Verkauf*).

Die Interpretation des Analyseproblems als Erweiterung des Analyseziels folgt primär Überlegungen zur Praktikabilität und Flexibilität der Modellierung. Einerseits ist ein Analyseproblem dadurch mithilfe eines einzigen Modellierungskonstrukts darstellbar, wodurch die allgemeine Handhabbarkeit steigt und in der nachfolgend beschriebenen Verkettungssicht die Verknüpfung erleichtert wird. Die damit einhergehende Einschränkung eines Analyseobjekts auf ein Analyseziel scheint akzeptabel und aufgrund der hohen Spezifität des Analyseobjekts zweckmäßig, zumal die Nutzung eines Analysedatenbestands zur Verfolgung mehrerer Analyseziele davon unberührt bleibt.[80] Andererseits kann ein Analyseproblem, das gegebenenfalls mehrstufig verfeinert wurde, in dieser Repräsentation durch Abstraktion von seinen Beschreibungselementen jederzeit auf ein Analyseziel reduziert werden, ohne die Ergebnisse der Verfeinerung zu beeinflussen, wodurch Übersichtlichkeit und Flexibilität des Modellsystems steigen.

4.4.3 Verkettungssicht

Analyseziele und -probleme können sequenziell verkettet werden, um den Gang einer aus mehreren Einzelanalysen bestehenden Untersuchung darzustellen und somit eine Analysestrategie zu dokumentieren (Untersuchungsdesign, vgl. Abschnitt 3.1.2.1). Eine **`Verkettung`** wird durch eine gerichtete Kante (Pfeil) symbolisiert und verknüpft jeweils zwei (2,2) **`Analyseziele`**, die auch als **`Analyseproblem`** ausgeprägt sein können. Damit ist eine beliebige Kombination von Analysezielen und -problemen möglich, d.h., die Verknüpfung von Analysevorhaben unterschiedlichen Spezifikationsgrades darstellbar. Ein **`Analyseziel`** bzw. -problem kann an beliebig vielen (0,*) ein- oder ausgehenden Verkettungen beteiligt sein. Ein Schema der Verkettungssicht wird als *Analysekette* bezeichnet. Zur Differenzierung zwischen geplanten und realisierten Analyseketten kann jeder Verkettung ein `Typ` zugewiesen werden: Geplante Verkettungen werden als **`Route`** (unterbrochene Kante), realisierte Verkettungen als **`Pfad`** (durchgängige

[80] Das Analyseobjekt ist nicht mit den Analysedaten identisch, sondern stellt eine Spezifikation des Inhalts von Analysedaten dar. Daten als Modell des Untersuchungsobjekts sind für beliebige Analysen nutzbar.

Kante) repräsentiert. Prinzipiell wird die Ausprägung als Pfad angenommen. An die Verkettung können beliebig viele (0,*) `Bedingungen` geknüpft werden, die zur Verfolgung eines Kettengliedes zu erfüllen sind. Auf diese Weise sind alternative Routen modellierbar. Zu realisierten Ketten lassen sich analog die tatsächlichen `Bedingungen` belegen, die zur Wahl des jeweiligen Pfades geführt haben. Das Metamodell der Verkettungssicht zeigt Abbildung 36.

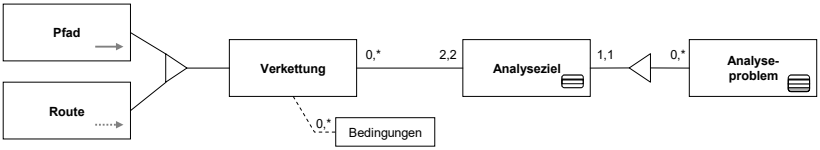


Abbildung 36: Metamodell zur Verkettungssicht der Analyseebene (eigene Darstellung)

Eine mit der Problemkarte aus Abbildung 28 korrespondierende Analysekette zeigt Abbildung 37. Analysen, die dasselbe Domänenobjekt untersuchen, sind vertikal jeweils auf einer Ebene angeordnet. Hierdurch ist auf einen Blick erkennbar, hinsichtlich welcher Merkmale und aus welchen Perspektiven ein Objekt betrachtet wird.

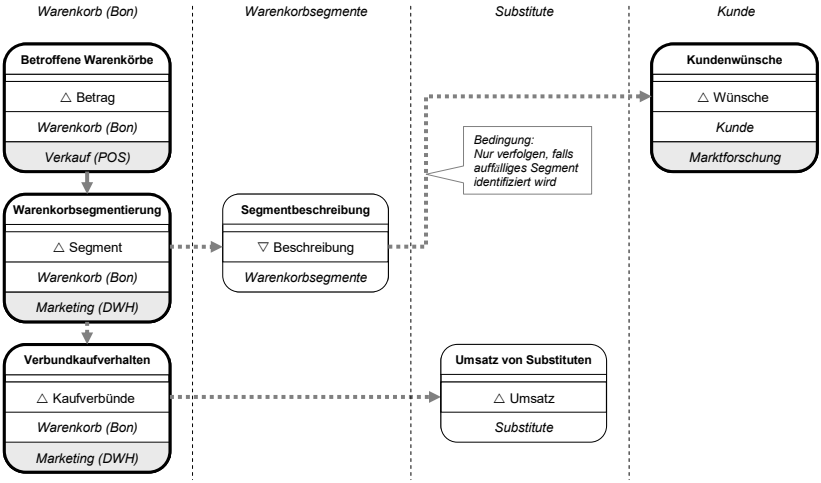


Abbildung 37: Beispielhafte Analysekette mit Analysezielen und -problemen (eigene Darstellung)

Neben Analyseproblemen sind auch zunächst nur als Analyseziele spezifizierte Untersuchungen enthalten. Der resultierende Spielraum verschafft Flexibilität bei der Ausgestaltung der Analysen, ohne die Konzipierung der Kette zu behindern. So ist z.B. die auf Anwendungsebene vorgegebene Umsatzanalyse von Ersatzartikeln noch nicht ausspezifiziert, da während der Verbundkaufanalyse zunächst entsprechende Substitute zu ermitteln sind. Zudem wurde ein Analyseziel zur Verifikation der durch Warenkorbsegmentierung ermittelten Cluster aufgenommen. Zusätzlich wird die Route zur Marktforschung von der Bedingung abhängig gemacht, dass auffällige Segmente identifiziert werden können, die eine teure Befragung lohnenswert erscheinen lassen.

4.4.4 Bibliothekssicht

Auch auf der Analyseebene steht eine *Bibliothekssicht* zur Verfügung, welche die zur Unterstützung des Untersuchungsdesigns zu speichernden Modellierungsartefakte umfasst. Die von dieser Sicht erfassten Metaobjekttypen sind demnach **`Analyseziel`** und **`Analyseproblem`**, aus denen Analyseketten bzw. beliebige Abschnitte von Analyseketten rekonstruierbar sind. Auf die Einzeldarstellung des Metamodells wird wegen dessen einfacher Struktur wiederum verzichtet (vgl. hierzu Abbildung 39).

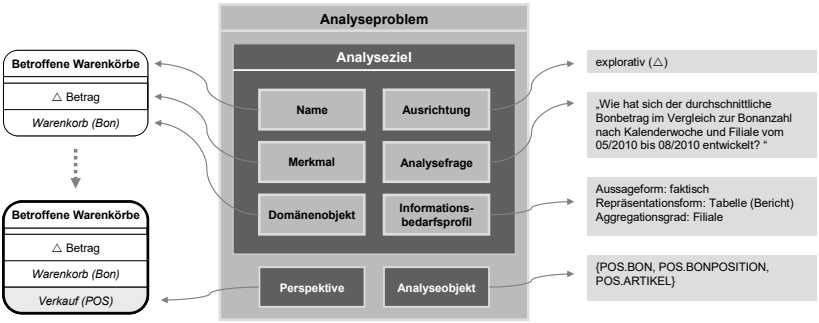


Abbildung 38: Analyseziele und Analyseprobleme am Beispiel Bonbetrag (eigene Darstellung)

Zum Auffinden von Modellierungsartefakten werden inhärente Eigenschaften verwendet, da diese eine eindeutige Kennzeichnung erlauben. Daher wird für Analyseziele das interessierende `Merkmal` des zu untersuchenden `Domänenobjekts` verwendet und für Analyseprobleme der Name der `Perspektive` ergänzt. Das Bezeichnungsschema lautet demnach „<`Merkmal`> von <`Domänenobjekt`> (aus Sicht von <`Perspektive`>)“. Zusätzlich kann ein beliebiger `Name` vergeben werden. Die Abbildung 38 zeigt links erneut die symbolische Repräsentation von Analysezielen und -problemen und rechts die weiteren Beschreibungselemente. Die Analyse-`Ausrichtung` kann mit den in Abschnitt 4.4.1 genannten Dreieckssymbolen signalisiert werden (explorativ △, konfirmatorisch ▽, schließend ▷).

4.4.5 Zusammenfassung zur Analyseebene

Informationsbedarfssicht (Zielsicht), Informationserzeugungssicht (Problemsicht), Verkettungssicht und Bibliothekssicht stellen eine vollständige Beschreibung der Analyseebene der Datenanalysearchitektur dar. Die folgende Abbildung 39 zeigt das integrierte Metamodell. Die Attribute der Metaobjekttypen stehen in allen Sichten zur Verfügung und sind ausführlich in Anhang A4.2 dokumentiert.

Der Analyseebene liegt die Metapher einer *Analysekette* zugrunde, die mehrere Untersuchungen derart verknüpft, dass ihre Realisierung eine umfassende, problemgerechte Beschreibung interessierender Sachverhalte produziert. Die Kette weist den Weg im Sinne einer Tour (Verkettungssicht), bei der konkret bestimmte Informationsbedarfe (Zielsicht) aus gegebenenfalls mehreren Perspektiven (Problemsicht) betrachtet werden.[81] Analyseziele und -probleme können in vielen Fällen nutzbringend konkretisiert werden, um die Untersuchung besser zu fokussieren oder durch Aufnahme weiterer Aspekte zu verbreitern bzw. vertiefen. Kriterien und Vorgehensweise dieser Verfeinerung diskutiert Abschnitt 5.4.5.3. Die resultierende Verfeinerungshierarchie,

[81] Dieser Weg kann im Hinblick auf die Metapher zur Anwendungsebene bildlich als „Besichtigungstour“ innerhalb eines oder mehrerer Orte auf der „mentalen Problemlandkarte“ interpretiert werden.

welche die Genese der Artefakte dokumentiert, kann mithilfe des Attributs `Abstammung` rekonstruiert und separat vom Schema dargestellt werden. Das Attribut enthält das Vaterobjekt und das angewendete Kriterium der Verfeinerung.

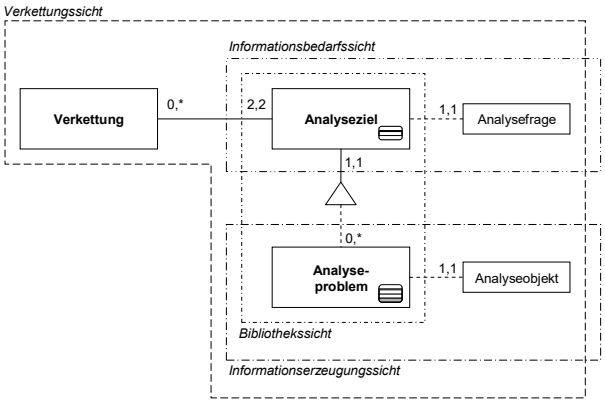


Abbildung 39: Integriertes Metamodell zur Analyseebene (eigene Darstellung)

4.5 Prozessebene: Lösungsverfahren zur Datenanalyse

Zur Lösung eines Analyseproblems der Analyseebene ist eine geeignete Aufgabe zu spezifizieren, die in der Lage ist, aus den ausgewählten Daten die gewünschten Informationen zu produzieren. Diese globale *Analyseaufgabe* umfasst sämtliche hierzu erforderlichen Transformationen und wird daher typischerweise in mehrere Teilaufgaben zerlegt, die gemeinsam den Analyseprozess bilden. Dieser wird auf der Prozessebene beschrieben.

Eine von konkreten Lösungsverfahren und Werkzeugen unabhängige Darstellung von Analyseprozessen erfolgt in der *Aufgabensicht*, die ausschließlich die Außensicht der Aufgaben betrachtet. Durch die Abstraktion von Realisierungsdetails ist diese Sicht zur Repräsentation allgemeingültiger Analyseprozesse geeignet, die (nach entsprechender Konkretisierung) mithilfe verschiedener Werkzeuge und Systeme realisierbar und somit in hohem Maße wiederverwendbar sind. Die *Aktivitätssicht* berücksichtigt zusätzlich die Innensicht der Aufgaben,

indem geeignete, von vorliegenden Werkzeugen bereitgestellte Lösungsverfahren (Operatoren) ausgewählt und mit konkreten Parameterwerten sowie den zu verarbeitenden Daten instanziiert werden. Die in der Aktivitätssicht dargestellten Prozesse sind für das zugehörige Werkzeug ausführbar und korrespondieren mit den typischen Repräsentationsformen gängiger Datenanalyse- und Datentransformationswerkzeuge.

Einzelne Ausführungen der aus Aufgaben- oder Aktivitätssicht spezifizierten Prozesstypen werden in der *Instanzensicht* dargestellt. Sie protokolliert die zur Prozessrealisierung eingesetzten Aufgabenträger, die verarbeiteten Datenausprägungen und den genauen zeitlichen Ablauf. Sie bildet damit zugleich die Grundlage zur späteren Beurteilung und Verbesserung von Analyseprozessen sowie zur Gewinnung von Erfahrungswissen. Zur Wiederverwendung geeignete Modellierungsartefakte werden wiederum in einer *Bibliothekssicht* dargestellt.

4.5.1 Aufgabensicht

Ein Datenanalyseprozess zielt auf die Bereitstellung von Informationen (Output), die durch eine Menge von Transformationen aus bestimmten Analysedaten (Input) abgeleitet werden. Die Beschreibung eines solchen Prozesses besteht in einer gegebenenfalls mehrstufigen Zerlegung der Gesamtaufgabe „Datenanalyse“ in Teilaufgaben, die jeweils spezifische Datentransformationen repräsentieren und über definierte Schnittstellen Daten austauschen (vgl. Abschnitt 2.3). Ein Analyseprozess beschreibt demnach aus Struktursicht eine Menge von Aufgaben, die durch ein Netz von Erzeuger-/Verbraucher-Abhängigkeiten (Austauschbeziehungen) verbunden sind. Diese Abhängigkeiten bestimmen zugleich das mögliche Prozessverhalten, das durch Ereignisse gesteuert wird und durch Vorgabe spezifischer Bedingungen eingeschränkt werden kann.

4.5.1.1 Aufgabe

Eine **Aufgabe** ist aus Außensicht definiert durch Aufgabenobjekt, Aufgabenziele sowie Vor- und Nachereignisse. Aufgabenobjekt (Gegenstand der Aufgabe) der Datenanalyse sind die Analysedaten, die gemäß

den Aufgabenzielen zu transformieren sind. Reihenfolgebeziehungen zwischen Aufgaben werden mithilfe der Ereignisse repräsentiert, die nach Abschluss einer Aufgabe nachfolgende Aufgaben anstoßen. Die Außensicht benennt keinen Aufgabenträger und kein Verfahren zur Aufgabenrealisierung [FeSi13, 98, 203]. Diese Spezifikationstiefe wird als hinreichend erachtet, um die Gliederung einer Datenanalyse in Transformationsschritte und deren Verknüpfung darzustellen, wie sie zur Vorgabe von Untersuchungsdesigns, zur Modellierung von allgemeingültigen Prozessen oder Vorgehensempfehlungen eingesetzt werden. Derart abstrakte Repräsentationen stellen zudem ein geeignetes Hilfsmittel zur Planung konkreter Prozesse dar, indem sie Zwischenstufen der Planung abbilden, die sukzessive verfeinert und in ausführbare Workflows überführt werden können. Aktivitäten als Elemente von Workflows sind stets auch Aufgaben, weshalb alle in diesem Abschnitt getroffenen Aussagen auch auf Aktivitäten zutreffen.

Aufgaben können gemäß ihrer `Verwendung` innerhalb eines Prozesses in **`Analyseaufgaben`** und **`Transformationsaufgaben`** differenziert werden. Erstgenannte beschreiben die globale Analyseaufgabe sowie die eigentlichen Auswertungsschritte innerhalb der Analysephase und verfügen über einen erweiterten Attributsatz, um spezifische Eigenschaften und Zielkriterien abzubilden. Zweitgenannte repräsentieren Aufgaben innerhalb der Datenvorbereitungs- und Ergebnisaufbereitungsphasen, die in der Regel mit weniger Anforderungen beladen sind. Zu den Transformationsaufgaben gehören auch solche Tätigkeiten, die einen Interpretations- oder Entscheidungsspielraum beinhalten und typischerweise interaktiv oder manuell erfolgen (vgl. Abschnitt 2.3.2.2). Die Bezeichnung Transformationsaufgaben dient in erster Linie der Abgrenzung von den Analyseaufgaben.

Die **`Analyseaufgabe`** kann mittels Angabe eines `Analyseproblem`s eine Beziehung zur Analyseebene herstellen. Aus dem Informationsbedarfsprofil des Analyseproblems lassen sich Bewertungsmaßstäbe für Analyseergebnisse ableiten, die in den Attributen `Bewertungskriterien`, `Bewertungsfunktionen` und `Präferenzrelationen` repräsentiert werden. Diese Bewertungsmaßstäbe werden in Abschnitt 7.2.1 im Zusammenhang mit ihrer Verwendung näher erläutert.

`Bewertungskriterien` dienen der Ermittlung von Zielerreichungsgraden, `Bewertungsfunktion`en der monetären Bemessung von Nutzen oder Kosten von Analyseergebnissen, und `Präferenzrelation`en zur Auswahl von Ergebnissen, Objektklassen oder Modellsegmenten. Die **`Transformationsaufgabe`** kommt ohne diese speziellen Bewertungsmaßstäbe aus. Allen Aufgaben können jedoch `Anforderung`en beigefügt werden, die in Form von Deskriptoren repräsentiert werden und Formalziele darstellen, welche die Ausfüllung eventuell bestehender Freiheitsgrade bei der Aufgabendurchführung leiten.

4.5.1.2 Funktion

Das Sachziel der Aufgabe wird durch Zuordnung einer **`Funktion`** einheitlich repräsentiert.[82] Die Funktion beschreibt die Transformation der Eingabedaten in Ausgabedaten, welche durch ihren `Namen` gemäß dem Schema „<`Eingabedatentyp`> <`Transformation`> [in <`Ausgabedatentyp`>]“ dokumentiert wird, wie z.B. „reelle Variable kategorisieren in nominale Variable“ oder „zwei relationale Tabellen verknüpfen“.[83] Die Funktion legt demnach neben der Art der `Transformation` auch die Menge der `Eingabedatentyp`en und `Ausgabedatentyp`en fest. Hierbei wird keine Aussage über die Kardinalität der Ein- oder Ausgabedaten getroffen, d.h., es wird lediglich bestimmt, dass die eine Funktion realisierende Aufgabe prinzipiell in der Lage ist, Daten der angegebenen Typen zu konsumieren bzw. produzieren. So muss z.B. eine Aufgabe die von der Funktion vorgegebenen Ausgabedatentypen nicht zwingend vollständig erzeugen, sondern kann etwa Ergebnisse entweder des einen oder des anderen Typs produzieren.

---

[82] Der Begriff *Funktion* wird zur Abgrenzung von den in einem Prozess enthaltenen (Prozess-) Aufgaben verwendet und ist im Sinne einer funktionalen Beschreibung zu verstehen (vgl. „Funktionalität“ von Aufgaben [EnLS97, 6]; Verhalten eines Lösungsverfahrens als Netz von Aktionen/Funktionen [Fers92, 7]).

[83] Vgl. auch das Bezeichnungsschema <Aufgabenobjekt> <Verrichtung> bei [ThFe06, 207], [FeSi13, 100].

Die Funktionszuordnung folgt Überlegungen aus der Semantischen Prozessmodellierung, bei der Prozesselementen in einer Ontologie formalisierte Begriffe zugewiesen werden, um die mit natürlichsprachigen Beschreibungen einhergehenden sprachlichen Vagheiten (Begriffsdefekte) zu vermeiden und eine maschinell verarbeitbare Semantik von Modellierungsprodukten zu erreichen [ThFe09, 506f.]. Damit sollen die intersubjektive Verständlichkeit und Wiederverwendung von Prozessschemata verbessert sowie Planung, Validierung und Suche innerhalb der Schemata erleichtert werden [ThFe06, 205f.], [ThFe09, 516]. Die Referenzierung von Funktionen, die im Prozessschema nicht in Erscheinung treten, durch Prozesselemente wird auch von BECKER ET AL. zur Unterstützung der Adaption von Referenzprozessmodellen bzw. -bausteinen propagiert. „Das Konstrukt der Funktion wird eingeführt, um semantisch identische Aktivitäten in mehreren Prozessen wiederverwenden zu können, ohne prozessspezifische Abhängigkeiten übernehmen zu müssen“ [BeDK04, 253]. Die Funktion dient dabei gewissermaßen als Vorlage für gleichartige Aufgaben.[84]

4.5.1.3 Flussbeziehung

Die Verknüpfung der Aufgaben erfolgt durch *Flussbeziehungen (Flüsse)*, die Reihenfolge und Kommunikationskanäle zwischen den Aufgaben definieren.[85] Diese sind in Prozessen mit informationeller Wertschöpfung in erster Linie aus Datenaustauschbeziehungen ableitbar

[84] THOMAS & FELLMANN verknüpfen Modellelementinstanzen zur semantischen Annotation mit Instanzen der Ontologieklassen [ThFe06, 213], [ThFe09, 510], während BECKER ET AL. eine eindeutige Zuordnung eines Prozesselements zu einer Funktion über eine referenziert-Beziehung vornehmen [BeDK04, 252f.]. Auch in der vorliegenden Arbeit wird eine Referenz zwischen zwei Metaobjekttypen innerhalb derselben Metaebene verwendet, wodurch die inhaltlich teilweise redundante, doppelte Instanziierung überflüssig wird.

[85] Typischerweise erfolgt eine explizite Differenzierung in Steuer- und Datenflüsse [Reif03, 75], [Jabl05, 204], [RWRW05, 254f.]. Sie unterbleibt an dieser Stelle, da Datenflüsse erst auf Instanzebene definiert sind und alle Beziehungen auf dieser Ebene als Steuerflüsse interpretierbar sind. Die Differenzierung spiegelt sich jedoch implizit in der Unterscheidung zwischen Datenabhängigkeits- und anderen Flüssen.

[RaME95, 468f.], [FeSi13, 62]. Eine *Datenabhängigkeit* [ReDa98, 3] zwischen zwei Aufgaben T_1 und T_2 liegt vor, wenn T_1 Ausgabedaten vom Typ D_a erzeugt und T_2 Eingabedaten vom Typ D_a verbraucht. T_1 und T_2 sind demnach mit einer **`Flussbeziehung`** vom Typ **`Datenabhängigkeit`** sequenziell zu verknüpfen. Datenabhängigkeiten werden mit durchgezogenen, gerichteten Kanten symbolisiert. Ihr `Datentyp` ergibt sich aus den korrespondierenden `Ein-` bzw. `Ausgabedatentypen` der Funktionen der verknüpften Aufgaben. Die Verbindung zweier Aufgaben über eine Datenabhängigkeit ist also nur zulässig, wenn die betroffenen Ein- bzw. Ausgabedatentypen kompatibel sind. Nähere Ausführungen hierzu enthält Abschnitt 5.5.4.3. Jede Flussbeziehung hat einen `Name`n, der an den Kanten notiert wird (Abbildung 40).

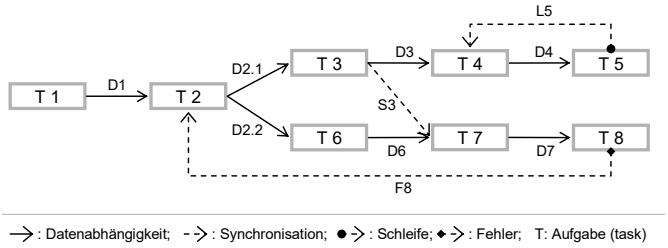


Abbildung 40: Analyseprozess (Aufgabensicht) mit verschiedenen Flussbeziehungen (eigene Darstellung)

Verzweigungen und Vereinigungen von Flussbeziehungen werden nicht durch eigene Modellierungskonstrukte dargestellt, sondern implizit durch mehrere aus- bzw. eingehende Kanten abgebildet (vgl. Aufgabe T_2 in Abbildung 40). Hierbei ist zunächst keine Ausführungssemantik definiert, d.h., über die Ausführbarkeit von Aufgaben in Abhängigkeit von eingehenden Flüssen sowie ihr Verhalten in Bezug auf ausgehende Flüsse wird keine Aussage getroffen. Vielmehr werden Flussbeziehungen auf Aufgabenebene als bloße Kommunikationskanäle interpretiert,[86] über die Ereignisse transportiert werden können. Der Typ der Ereignisse ist durch den Typ der Flussbeziehung festgelegt.

[86] Dies entspricht einem Petri-Netz vom Typ eines Kanal-Instanzen-Netzes [Balz96, 306].

Zur Abbildung spezieller Koordinationsbeziehungen stehen als weitere Flüsse die Typen *Synchronisation*, *Sprung* und *Fehler* zur Verfügung (vgl. [ReDa98, 3f.]). Sie transportieren ausschließlich boolesche Variablen, d.h., sie signalisieren das Zutreffen oder Nichtzutreffen einer Bedingung. Die **`Synchronisation`** unterstützt die Koordination zweier Aufgaben und legt fest, wann eine Aufgabe T_7 in Abhängigkeit des Ausführungszustands einer Aufgabe T_3 durchgeführt werden kann (vgl. Fluss S_3 in Abbildung 40, symbolisiert durch eine gebrochene Kantenlinie).

Flussbeziehungen von Typ **`Sprung`** dienen der Modellierung von *Schleifen* sowie planbaren Sprüngen. Schleifen sind häufig durch geeignete Aufgabenspezifikation vermeidbar,[87] werden zuweilen aber bewusst eingesetzt, um bestimmte Ablaufstrukturen explizit abzubilden, wie etwa die iterative Modellerstellung (vgl. Abschnitt 3.1.2.3) oder die Kreuzvalidierung (vgl. Abschnitt 7.2.1.1) [Bert+09, 30]. Planbare Sprünge können die Auslassung von Prozessabschnitten unter definierten Bedingungen anzeigen, z.B. wenn Maßnahmen der Datenbereinigung angesichts der Datenqualität überflüssig sind (vgl. [RBFD02, 185f.]). Ein Sprung wird durch eine gebrochene Kantenlinie mit gefülltem Kreis am Startpunkt dargestellt und führt von der Aufgabe, welche das Vorliegen der Sprungbedingung überprüft, zu jener Aufgabe, bei der die Prozessausführung fortzusetzen ist (vgl. Schleifenfluss L_5 von T_5 zu T_4 in Abbildung 40). Die Sprungbedingung wird im Attribut `Ausgangsbedingung` des Flusses spezifiziert und kann auf alle innerhalb der Aufgabe verfügbaren Variablen Bezug nehmen (ein- und ausgehende Flüsse oder Parameter des Operators einer Aktivität). Beispielsweise legt die Bedingung `n <= Eingabedaten->size()` fest, dass eine Schleife solange wiederholt wird, bis der Zähler n anzeigt, dass alle Elemente der Eingabedaten verarbeitet wurden.

Ist bekannt, dass während der Durchführung bestimmter Aufgaben häufig *Fehler* auftreten, die typischerweise durch Wiederholung der betroffenen Aufgabe nicht zu beheben sind, weil sie von vorangehenden

[87] Viele Analysewerkzeuge bieten spezielle Operatoren, die mehrfach auszuführende Aufgaben, z.B. zum Iterieren über mehrere Dateien, mit der notwendigen Steuerlogik kapseln [Bert+09, 30], [Rapi10, 31].

Aufgaben verursacht werden, kann dieses Wissen mithilfe von Flussbeziehungen vom Typ **`Fehler`** dokumentiert werden. Sie verbinden Fehlerknoten mit Neustartknoten, bei denen die Bearbeitung im Fehlerfall wieder aufzunehmen ist. Im Gegensatz zu einem einfachen Rückwärtssprung fordert ein Fehlerfluss zur effektiven Fehlerbehebung das Rücksetzen der Wirkungen aller übersprungenen Aufgaben [ReDa98, 4]. Die `Ausgangsbedingung` eines Fehlerflusses kann direkt auf Rückgabeparameter der Operatoren von Aktivitäten Bezug nehmen, falls solche verfügbar sind. Flussbeziehungen vom Typ Fehler werden durch gebrochene Kantenlinien mit gefüllter Raute am Ausgangsknoten dargestellt (vgl. Beispiel F8 in Abbildung 40).

4.5.1.4 Metamodell

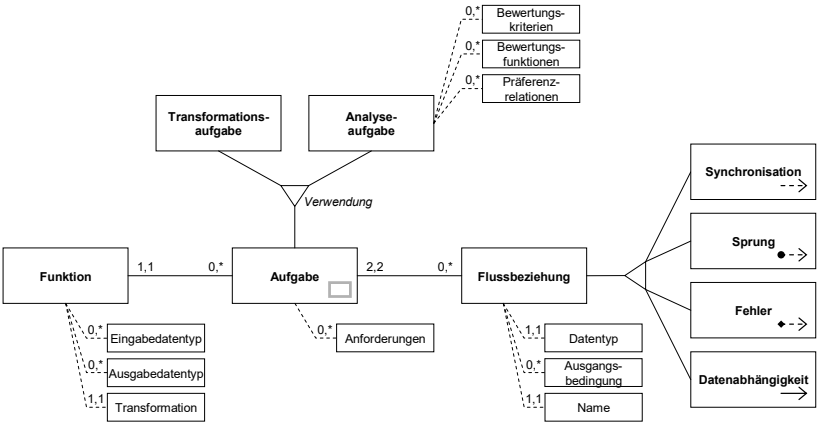


Abbildung 41: Metamodell zur Aufgabensicht der Prozessebene (eigene Darstellung)

Das Metamodell in Abbildung 41 fasst die Beschreibungselemente der Aufgabensicht zusammen. Eine **`Aufgabe`** wird durch ein nicht ausgefülltes Rechteck symbolisiert und kann als **`Analyseaufgabe`** oder **`Transformationsaufgabe`** auftreten. Jede Aufgabe folgt den abstrakten Vorgaben einer (1,1) **`Funktion`**. Diese beschreibt das Sachziel der Aufgabe durch `Ein-` und `Ausgabedatentyp` sowie die zu verrichtende `Transformation`. Funktionen sind in einer Ontologie (Abschnitt

4.7.1) beschrieben und können von beliebig vielen (0,*) Aufgaben referenziert werden. Diese ergänzen die Spezifikation der Funktion um Formalziele, die als `Anforderungen` repräsentiert sind. Jeweils genau zwei (2,2) Aufgaben können durch eine **`Flussbeziehung`** verknüpft sein, die entweder zur Menge der `Eingabeflüsse` oder der `Ausgabeflüsse` der Aufgabe gehört. Eine Aufgabe kann beliebig viele (0,*) ein- und ausgehende Flüsse besitzen. Eine Flussbeziehung kann vom Typ **`Synchronisation`**, **`Sprung`**, **`Fehler`** oder **`Datenabhängigkeit`** sein. Sie transportiert bei Erfüllung ihrer `Ausgangsbedingung` Nachrichten, die einem definierten `Datentyp` folgen. Flussbeziehungen starten und enden stets an einer Aufgabe, die in diesen Fällen als Quelle bzw. Senke von Datenflusskanälen fungiert [Jabl00, 350]. Die erste Aufgabe eines Prozesses beschreibt damit z.B. das Einlesen von Daten aus einer Datenquelle, die letzte Aufgabe das Speichern oder Visualisieren von Ergebnissen.

4.5.2 Aktivitätssicht (Workflow-Sicht)

Ein in der Aufgabensicht spezifizierter Analyseprozess beschreibt, welche Aufgaben in welcher Abfolge für eine Datenanalyse durchzuführen sind, ist jedoch nicht ausführbar. Die Aktivitätssicht verfeinert die Spezifikation der Aufgaben durch Zuordnung von Lösungsverfahren und Aufgabenträgertypen (Beschreibung der Aufgabeninnensicht). Ein vollständig aus Aktivitätssicht spezifizierter Analyseprozess ist demnach grundsätzlich ausführbar. Mithilfe von konkreten Werkzeugen ausführbare Datenanalyseprozesse werden auch als *Datenanalyse-Workflows* bezeichnet (vgl. Abschnitt 2.3.2). Workflows stellen die Verbindung zwischen Aufgaben- und Aufgabenträgerebene her [BBFS11, 8] und sind demnach von der Systemumgebung abhängig, auf deren Ressourcen sie Bezug nehmen.

Ein Workflow beschreibt wörtlich einen Arbeitsfluss, d.h., wie Arbeit über definierte Kanäle durch mehrere Arbeitsschritte fließt, die jeweils einen Beitrag an der zu bewältigenden Gesamtaufgabe leisten [JaBS97, 17], [Jabl00, 349]. Eine vollständige Beschreibung automatisiert ausführbarer Workflows umfasst wenigstens folgende Aspekte [Reif03, 74f.], [Jabl05, 204]: (1) Spezifikation der Arbeitsschritte in Form von

Aktivitäten; (2) Beschreibung der kausalen und temporalen Abhängigkeiten (Reihenfolge) zwischen den Aktivitäten (*Flussbeziehungen*), wie sie u.a. durch die ausgetauschten Daten entstehen; (3) Bestimmung der zur Ausführung der Aktivitäten einzusetzenden *Verfahren und Aufgabenträger*; (4) Zuordnung der für die Realisierung verantwortlichen *Personen*. Diese Aspekte werden im Weiteren erläutert.

4.5.2.1 Aktivität

Eine *Aktivität* ist eine aus Außen- und Innensicht vollständig spezifizierte Aufgabe und stellt einen ausführbaren Arbeitsschritt eines Workflows dar.[88] Aktivitäten werden in Theorie und Praxis der Datenanalyse u.a. auch als Schritte [KlZy96, 575], Operationen [KiVZ00], Operatoren [Rapi10], Simple Tasks [Enge99], Agenten [ZLKO97] oder schlicht als Knoten [Bert+09] bezeichnet. Die **`Aktivität`** erbt als Erweiterung der Aufgabe all deren Eigenschaften und Flussbeziehungen. Zu ihrer Durchführung ist die Zuordnung eines *Lösungsverfahrens* erforderlich, das als `Operator` bezeichnet wird. Die Menge verfügbarer Operatoren ist auf Ressourcenebene beschrieben. Jeder Operator ist eine Implementierung eines konkreten Datenanalyse- oder -transformationsverfahrens durch ein Anwendungssystem oder Werkzeug und steht damit nur gemeinsam mit diesem Aufgabenträgertyp zur Verfügung.

Einer Aktivität kann genau ein Operator zugeordnet werden. Sie stellt demnach eine Aufgabe der niedrigsten Zerlegungsstufe dar, für die eine Verfahrensimplementierung verfügbar ist. Ihre Granularität und Definition werden demnach durch Operatoren bestimmt, die ebenso wie Aufgaben durch Referenz auf eine Funktion beschrieben werden (Abschnitt 4.6.2.1). Mit der Funktionszuordnung wird zugleich die Menge der passenden Operatoren eingeschränkt und die Verfahrensauswahl unterstützt (vgl. [Enge99, 101]). Während Aufgaben mit beliebigem Aggregationsgrad modelliert werden können, bilden Aktivitäten

[88] Sofern nicht zwischen Außen- und Innensicht einer Aufgabe differenziert werden muss, können die Begriffe Aufgabe und Aktivität synonym verwendet werden.

stets die Blattknoten im Aufgabenzerlegungsbaum eines Prozesses [Enge99, 92, 95].

Mit Festlegung des Operators spezifiziert eine Aktivität die Durchführung des zugehörigen Verfahrens auf den Eingabedaten (*Vorgangstyp*) [FeSi13, 99]. Diese Spezifikation ist um `Parameterwerte` zu ergänzen, um den Operator in Einklang mit den Aufgabenzielen geeignet zu instanziieren. Die Menge verfügbarer Parameter wird aus der Operatorbeschreibung (`Operator.Parameter`) übernommen und mit konkreten Einstellungen belegt. Eine Aktivität lässt sich aus dieser Perspektive als Operatoranwendung [KSBF10, 5], [KlZy96, 575] bzw. Operatoraufruf [Enge99, 92] interpretieren.

Jeder Aktivität kann ferner eine `Rolle` zugeordnet werden, welche die Qualifikationen definiert, die ein personeller Aufgabenträger zu ihrer Durchführung erfüllen muss. Die Rollenangabe unterstützt primär die Auswahl von Personen zur Aufgabendurchführung sowie die Kostenabschätzung.

4.5.2.2 Interpretation von Flussbeziehungen

Beziehungen zwischen Vorgangstypen werden über Ereignisse hergestellt und sind als Vorgangsnetze (genauer: Vorgangs-Ereignis-Netze) darstellbar [FeSi13, 99]. Diese Netze beschreiben die durchzuführenden Vorgangstypen, deren Reihenfolge sowie die zum Datenaustausch verwendeten Interaktionskanäle und Flussarten. Typische Repräsentationsformen für Vorgangsnetze sind Petri-Netze oder aus diesen abgeleitete Sprachen [DeOb96, 359]. In Petri-Netzen werden Vorgänge als Übergänge und Interaktionskanäle als Zustände (Stellen) modelliert [FeSi13, 49-51], woraus folgende Interpretation von Flussbeziehungen resultiert (vgl. [FeSi13, 203f.]).

Eine Flussbeziehung repräsentiert einerseits einen Interaktionskanal zwischen Aktivitäten bzw. den durchführenden Aufgabenträgern (Strukturaspekt, Typmerkmal), andererseits einen Zustand, dessen Veränderung durch Ereignisse signalisiert wird (Verhaltensaspekt, Instanzmerkmal). Der Interaktionskanal kann Daten des zugehörigen Datentyps übertragen, welche Bestandteil der Aufgabenobjekte der

verknüpften Aktivitäten sind. Das Vorliegen eingehender Daten versetzt den Kanal in einen Zustand, der eine Vorbedingung der empfangenden Aktivität erfüllt (Vorereignis). Analog repräsentiert die Bereitstellung ausgehender Daten einen Zustand, der bei Erreichen einer Nachbedingung der sendenden Aktivität hergestellt wird (Nachereignis).

4.5.2.3 Zuordnung und Ergänzung von Datenabhängigkeiten

Vor- und Nachbedingungen werden auf Prozessebene nicht explizit modelliert, sondern implizit über Flussbeziehungen dargestellt. Aufgrund der Bindung des Kanals an den Datentyp des Flusses signalisieren Datenabhängigkeiten ausschließlich die Bereitstellung von Daten, die den jeweiligen Bedingungen genügen. Eine Aktivität muss jedoch bezüglich der Datenabhängigkeiten die Anforderungen des zugeordneten Operators erfüllen. Die von diesem benötigten `Eingabedaten` und produzierten `Ausgabedaten` sind in der Beschreibung des Operators auf Ressourcenebene hinterlegt.[89] Zur Prüfung auf Vollständigkeit, zur Dokumentation der Semantik der verarbeiteten Daten sowie zu Zwecken der Verifikation, Validierung und Fehlersuche wird die Zuordnung zwischen eingehenden Flüssen und Eingabedaten sowie ausgehenden Flüssen und Ausgabedaten als `Eingabedatenzuordnung` bzw. `Ausgabedatenzuordnung` in der Aktivität festgehalten.[90] In Form eines Dupels (<`Datenabhängigkeit.Name`>, <`Operator.{Eingabedaten|Ausgabedaten}.Name`>) wird die Rolle beschrieben, die ein Fluss für den Operator bzw. die Aktivität spielt. Beispielsweise besagt die Zuordnung (`Kundenprofil14, Trainingsdaten`), dass der Inhalt des eingehenden Flusses „`Kundenprofil14`“ dem Operator als Trainingsdaten dienen soll.[91] Falls in der zugrundeliegenden Aufgabe nicht

---

[89] Hierbei müssen die Wertebereiche von `Eingabedatentyp` bzw. `Ausgabedatentyp` der Funktion der Aktivität Teilmengen der Wertebereiche der `Eingabedaten` bzw. `Ausgabedaten` des Operators sein.

[90] Diese Zuordnung erfolgt in Datenanalysewerkzeugen typischerweise automatisch, sobald ein Datenabhängigkeitsfluss an eine Ein- oder Ausgabestelle (Port) eines Operators angedockt wird.

[91] Das Werkzeug RAPIDMINER nutzt diese Informationen u.a. im Rahmen der sogenannten Metadaten-Transformation, um die Ausgabe einer Aktivität oder eines

bereits alle vom Operator benötigten Eingabedaten durch eine korrespondierende Datenabhängigkeit abgedeckt sind, müssen fehlende Flüsse an der Aktivität ergänzt werden.

Für die Modellierung von Datenabhängigkeiten ist es unerheblich, ob Operatoren die Daten als Datenstrom (z.B. tupelweise; synchron) einander direkt übertragen oder ob diese über einen gemeinsamen Speicher (z.B. als relationale Tabelle; asynchron) ausgetauscht werden. Ein Datenabhängigkeitsfluss modelliert unabhängig von der physischen Realisierung des Transports die Benachrichtigung über das Vorliegen von Datenpaketen.

4.5.2.4 Erweiterung einer formalen Semantik

Die bisher diskutierte Modellierung von Datenanalysen auf Aktivitätsebene führt zu Workflows, die auf Basis der zugrunde gelegten Systemumgebung wenigstens interaktiv ausführbar sind. Das Schema definiert Regeln und Bedingungen zur Steuerung konkreter Abläufe. Zur vollautomatisierten Durchführung sind weitere formale Beschreibungselemente notwendig, welche die Ablaufsemantik der Prozesse in maschinell interpretierbarer Form definieren [Reif03, 73f.]. Hierfür eignet sich die mathematische Fundierung der Petri-Netze, welche die Verifikation von Syntax und Semantik,[92] die Validierung (Simulation) und die computergestützte Ausführung der Prozesse unterstützt [DeGS95, 464], [LaSW97, 479f.]. Die formale Semantik des Aktivitätsmodells kann durch eine Abbildung auf Prädikat/Transitions-Netze (Pr/T-Netze) hergestellt werden (vgl. [GrKa96, 385]).

Pr/T-Netze sind höhere Petri-Netze und interpretieren Zustände als Prädikate [DeOb96, 363], die Objekte unterschiedlichen Typs enthalten

Prozesses vorauszuberechnen. Hierzu wird eine Simulation anhand der Metadaten der Datenflüsse durchgeführt und die zu erwartende Belegung von Flüssen und Variablen durch alle Aktivitäten des Prozesses propagiert [Rapi10, 63ff.].

[92] Die maschinelle Überprüfbarkeit (z.B. auf syntaktische Korrektheit der Spezifikation oder auf Erreichbarkeit und Ausführbarkeit von Aktivitäten) wird als Grundanforderung an die ordnungsgemäße Modellierung von Prozessen gesehen [LaSW97, 479], [Jabl00, 352].

können. Dadurch werden die von den Flüssen transportierten Daten semantisch unterscheidbar und über Variablen mit dem Namen des Flusses referenzierbar. Übergänge (Transitionen) sind als Funktion f interpretierbar, die Eingaben IN in Ausgaben OUT = f(IN) transformiert [ReDe14, 175]. Die Funktion f wird auch als *Schaltwirkung* bezeichnet. Die Transition findet nur statt, wenn eine *Schaltbedingung* erfüllt ist. Gleichzeitig schaltet der Übergang nur für jene Objekte, die der Schaltbedingung genügen [Balz96, 304].

Schaltbedingung und Schaltwirkung werden im Allgemeinen durch den Operator definiert, der gemäß seiner Implementierung entscheidet, welche der vorliegenden Eingabedatenobjekte er konsumiert bzw. wie viele Ausgabedatenobjekte er erzeugt. Soll eine Verifikation oder Simulation des Aktivitätsschemas erfolgen, muss das Ausführungsverhalten expliziert werden. Dies kann in den Attributen `Schaltbedingung` und `Schaltwirkung` des Operators geschehen, die Bestandteil der Aktivitätsdefinition werden. Die Spezifikationen werden als logische bzw. mathematische Ausdrücke formuliert und können mithilfe der Flussvariablen auf Ein- und Ausgabedaten Bezug nehmen [Balz96, 304].

Die Schaltbedingung des Operators erfasst zunächst nur Datenabhängigkeitsflüsse. Sind weitere Flüsse zu berücksichtigen, wird prinzipiell unterstellt, dass eine Aktivität ausführbar ist, sobald alle eingehenden Kanäle bedient sind [Jabl00, 351], [DeGS95, 462]. Dieses Schaltverhalten lässt sich mithilfe der `Startbedingung` der Aktivitätsdefinition überschreiben. So kann eine Aktivität mit einem eingehenden Fluss IN_1 und einem Fehler F_1 unabhängig vom Zustand von F_1 starten, wenn IN_1 bedient ist (`IN1 AND (F1 OR NOT F1)`). Analog erfordern ausgehende Flüsse vom Typ Sprung und Fehler das Blockieren aller anderen Ausgänge, falls die Sprung- bzw. Fehlerbedingung erfüllt ist. Diese Regeln können als `Endbedingung` der Aktivität formuliert werden. So gilt für eine Schleife wie für einen Fehler X und ausgehende Flüsse OUT: `(X AND NOT OUT) OR (NOT X AND OUT)`. Unabhängig davon kann auch das Verhalten bezüglich Datenabhängigkeiten modifiziert werden. Beispielsweise ist denkbar, dass eine Aktivität mit drei eingehenden Flüssen IN_2, IN_3, IN_4 nur starten soll, wenn entweder

IN_2 und IN_3 oder IN_2 und IN_4 versorgt sind (Startbedingung: `(IN2 AND IN3) OR (IN2 AND IN4)`).

Zur Gewährleistung korrekter Workflows sind symmetrische Steuerstrukturen von Vorteil, da diese die formale Verifikation erleichtern. Prozesssegmente mit Verzweigungen und Sprüngen sind daher als symmetrische Blöcke mit wohl definierten Start- und Endknoten zu spezifizieren. Diese Blöcke können beliebig geschachtelt werden, dürfen jedoch nicht überlappen [ReDa98, 3]. Zu diesem Zweck können jeder Aktivität mehrere `Blockmarkierung`en beigefügt werden, falls sie als Start- oder Endknoten eines Blockes auftritt. Die Markierung benennt die Position der Aktivität im Block und den Blocktyp in der Form {„Start"|„Ende"}.`<Flussbeziehung.Typ>`, also z.B. `Start.Sprung` oder `Ende.Sprung`.[93]

4.5.2.5 Metamodell

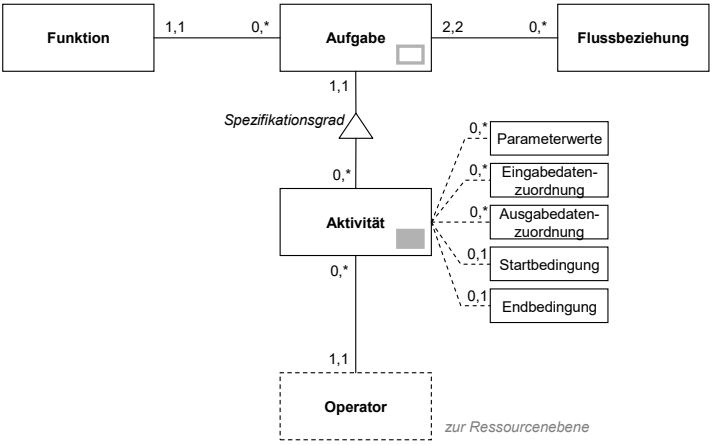


Abbildung 42: Metamodell zur Aktivitätssicht der Prozessebene (eigene Darstellung)

[93] Das Werkzeug KNIME stellt zusätzliche Hilfsknoten zur Verfügung, die Start und Ende zu wiederholender Prozessegmente kennzeichnen und über gemeinsame Variablen eng gekoppelt kommunizieren, um Prozesse in Form azyklischer, gerichteter Graphen zu erhalten [Bert+09, 30].

Das Metamodell der Aktivitätssicht ist in Abbildung 42 dargestellt. Es zeigt die **Aktivität** als Spezialisierung genau einer (1,1) **Aufgabe**. Eine Aktivität wird durch ein ausgefülltes Rechteck repräsentiert, das den gegenüber der Aufgabe höheren Spezifikationsgrad symbolisiert. Eine Aufgabe kann durch beliebig viele (0,*) Aktivitäten konkretisiert werden. Sie erbt die Beziehungen der Aufgabe zu **Funktion** und **Flussbeziehung** und erweitert deren Attributmenge u.a. um `Parameterwerte`, `Ein-` und `Ausgabedatenzuordnung`, `Start-` und `Endbedingung`. Abgebildet ist auch ist die Beziehung zum `Operator` auf Ressourcenebene.

4.5.3 Instanzensicht

Die Prozessbeschreibung in Form von Aufgaben und Aktivitäten erfolgt auf *Typebene*. Zur Abbildung tatsächlich realisierter Abläufe ist die *Instanzebene* zu betrachten. Sie eröffnet eine eigene Metaebene des Modellsystems, die eine Extension der auf Typebene spezifizierten Workflows (Ablauftypen) enthält [FeSi13, 138f.]. Diese Metaebene wird in Form der *Instanzensicht* in die Betrachtung einbezogen, um eine vollständige Beschreibung von Datenanalyseprozessen zu ermöglichen, die deren Spezifikation sowie konkrete Ausprägungen berücksichtigt.

Während der Durchführung eines Prozesses wird ein Workflow-Schema durch ablaufspezifische Daten instanziiert und in Form eines *Instanzmodells* repräsentiert. Dieses hat die Funktion eines Prozessprotokolls über den gesamten Lebenszyklus des Ablaufs [ReBD00, 2]. Es dokumentiert z.B. aktuelle Bearbeitungszustände und -zeiten von Vorgängen, eingesetzte Aufgabenträger, eingetretene Ausnahmesituationen sowie Abweichungen vom Planablauf. Das Instanzmodell bildet zusammen mit dem Workflow-Schema die Grundlage für die Steuerung [Gier00, 290], [Reif03, 83] und Revision von Prozessen.

4.5.3.1 Vorgang

Eine Aktivitätsdurchführung durch einen konkreten Aufgabenträger wird als *Vorgang* bezeichnet [FeSi13, 99]. Ein **Vorgang** referenziert die ihm zugrundeliegende `Aktivität` und speichert individuelle Ausfüh-

rungsdaten [ADH+03, 246f.]. Zunächst werden `Startzeit` und `Endzeit` des Vorgangs protokolliert. Aus ihnen können Ausführungsdauern berechnet und Abschätzungen für Prozesslaufzeiten künftiger Prozesse abgeleitet werden. Der `Zustand` gibt Aufschluss über den aktuellen Ausführungsstatus des Vorgangs [GCC+04, 324f.], wie er durch das Zustandsmodell in Abbildung 43 definiert ist.[94]

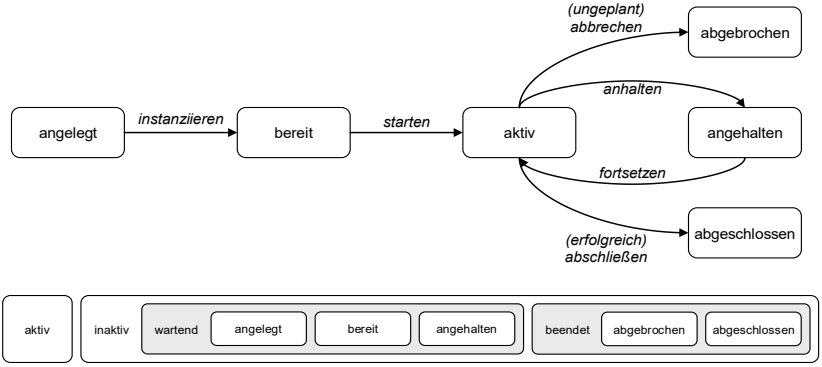


Abbildung 43: Zustandsmodell von Vorgängen und Prozessinstanzen (eigene Darstellung)

Ein Vorgang ist mit Definition der zugehörigen Aktivität im Zustand *angelegt*. Nach Instanziierung mit Verfahrensparametern, Eingabedaten und Aufgabenträgerinstanzen ist er zur Durchführung *bereit*. Er kann nun starten und wird damit *aktiv*. Der Analytiker kann den Vorgang beliebig anhalten (Zustand *angehalten*) und wieder fortsetzen. Kann der Vorgang erfolgreich abschließen, erreicht er den positiven Endzustand *abgeschlossen*. Andernfalls wird der Vorgang ungeplant abbrechen, entweder infolge eines Fehlers oder nutzerinitiiert, und landet im negativen Endzustand *abgebrochen*. Abgeschlossene und abgebrochene Zustände sind im übergeordneten Zustand *beendet*, und angelegte,

[94] Vgl. hierzu das allgemeine Zustandsmodell bei [ADH+03, 248]. Es ist Bestandteil eines allgemeinen Modells für Workflow-Logs, das die Autoren aus der Untersuchung mehrerer Process-Mining-Werkzeuge induzieren. Für weitere Instanzmodelle vgl. z.B. [GCC+04, 331], [Schi04, 268], [AcUA12, 355].

bereite und angehaltene Vorgänge sind zugleich *wartend*. Der Zustand *inaktiv* vereint wiederum wartende und beendete Vorgänge.

Ein Abbruch durch den Nutzer ist von einem verfahrensseitig verursachten Scheitern durch einen Eintrag im Attribut `Fehler` unterscheidbar. Dort können Rückmeldungen des Operators hinterlegt und zur Ursachenanalyse verwendet werden. Eine genauere Untersuchung des Lebenszyklus eines Vorgangs ermöglicht das `Ereignisprotokoll`, das die zeitliche Abfolge von Zustandsänderungen gemäß obigem Zustandsmodell dokumentiert. Zur Kostenberechnung und Leistungsbeurteilung der technischen Infrastruktur dient die Verknüpfung mit der konkret beteiligten Aufgabenträgerinstanz (`Server`) [AcUA12, 355], [GCC+04, 324].

Das Projektmanagement sowie die Erkennung von Schwächen der Prozessgestaltung (vgl. Abschnitt 3.3.3.2; Spiralmodell) unterstützt ein `Zähler`, der die Anzahl der Wiederholungen des Vorgangs verfolgt. Der `Name` des Vorgangs setzt sich aus dem Namen der zugrundeliegenden Aktivität und diesem `Zähler` zusammen (z.B. A-2) und gibt somit direkt über die Anzahl der Wiederholungen Auskunft [ADH+03, 247], [GCC+04, 331]. Wiederholungen von Aufgaben können abhängig von deren jeweiligem Spezifikationsgrad unterschiedlich oft auftreten. So kommen Instanzen abstrakter Aufgaben, wie etwa der Datenvorbereitung, in einem Ablauf zwangsläufig mehrfach vor, wenn mehrere Spezialisierungen dieser Aufgabe (z.B. Datenbereinigung und Translation) im Prozess enthalten sind. Die Wiederholung einer Aktivität liegt nur dann vor, wenn die Aktivitätsinstanzen in ihrer Definition (z.B. bezüglich ihrer Parameterwerte) unverändert sind. Wird eine Aktivität unter Änderungen erneut ausgeführt, handelt es sich um verschiedene Aktivitäten (jedoch gleiche Aufgaben), deren Instanzen demnach keine Wiederholungen darstellen.

4.5.3.2 Analysefall und Prozessinstanz

Mehrere `Vorgänge` bilden eine *Prozessinstanz* (Ablauf), die einen konkreten Analysefall repräsentiert [ADH+03, 241], [AcUA12, 355]. Ein ***Analysefall*** beschreibt eine konkrete Untersuchungssituation der Daten-

analyse und bezeichnet demnach die Durchführung eines Analyseprozesses zur Lösung eines Analyseproblems [Knob03a, 339f.] (Abbildung 44). Der Umfang der **`Prozessinstanz`** ergibt sich aus der Menge der im Rahmen eines Ablaufs durchgeführten Vorgänge. Sie ist durch den die Ausführung initiierenden `Benutzer` [GCC+04, 324] und die `Startzeit` (Zeitstempel) der Ausführung identifizierbar. Sie erhält einen zweiten Zeitstempel der `Endzeit`, wenn der letzte ihr zugehörige Vorgang beendet ist. Ihr aktueller `Zustand` ist aus den Ausführungszuständen aller Vorgänge ableitbar [Reif03, 74], [RWRW05, 255]. Der Analytiker kann zudem eine `Beschreibung` des Ablaufs angeben.

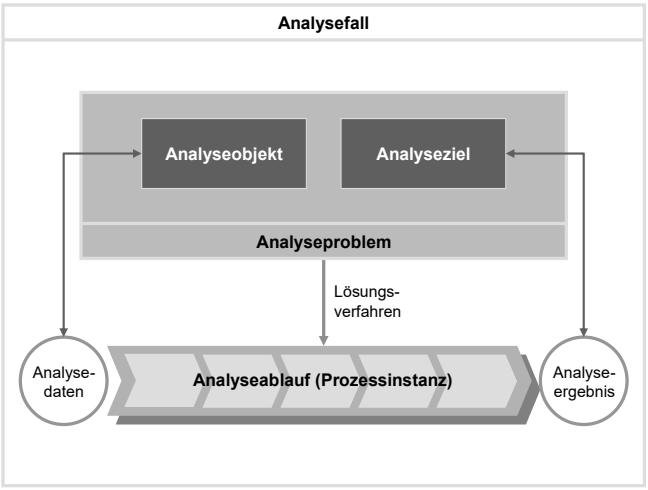


Abbildung 44: Konzept des Analysefalles (eigene Darstellung)

Die explizite Modellierung der Prozessinstanz ist zur redundanzfreien Dokumentation instanzspezifischer Eigenschaften sinnvoll. Dies betrifft auch Änderungen und Abweichungen, die am Instanzschema gegenüber dem Prozesstyp vorgenommen wurden [ReDa98, 16], [RWRW05, 255]. Sie werden als `Änderungen` protokolliert, die neben dem Typ der Änderungsoperation (z.B. Einfügung einer Aktivität), dem betroffenen Änderungsobjekt (neue Aktivität) und typspezifischen Parametern (Position der Einfügung) auch eine Begründung aufnehmen, um die Anpassungen nachvollziehbar zu dokumentieren. Als Repräsentation

eines Analysefalls speichert die Prozessinstanz nicht nur die Ergebnisse der während der Revision erfolgten *Beurteilung des Prozessablaufs* (`Prozessbewertung`), sondern insbesondere auch die Resultate der *Bewertung der Analyseergebnisse* (`Ergebnisbewertung`) (vgl. Abschnitt 7.2). Hierbei können die bei der Analyseaufgabe hinterlegten Bewertungskriterien eingesetzt werden.

4.5.3.3 Datenfluss

Ein- und Ausgabedaten eines Vorgangs werden in Form von *Datenflüssen* modelliert (vgl. [GCC+04, 322], [AcUA12, 355]). Ein **`Datenfluss`** ist eine Instanziierung einer Datenabhängigkeit und enthält eine Referenz auf ein `Informationsobjekt` der Ressourcenebene. Informationsobjekte sind Ausprägungen von semantisch annotierten Datenobjekttypen (Abschnitt 4.6.3.1) und können z.B. als Relationen (Datenbanktabellen), Berichte, Diagramme, Dokumente oder (Prognose-) Modelle in Erscheinung treten. Sie stellen konkrete Inhalte der auf Typebene spezifizierten Flussbeziehungen dar. Als `Name` des Datenflusses wird der Name des Informationsobjekts notiert.

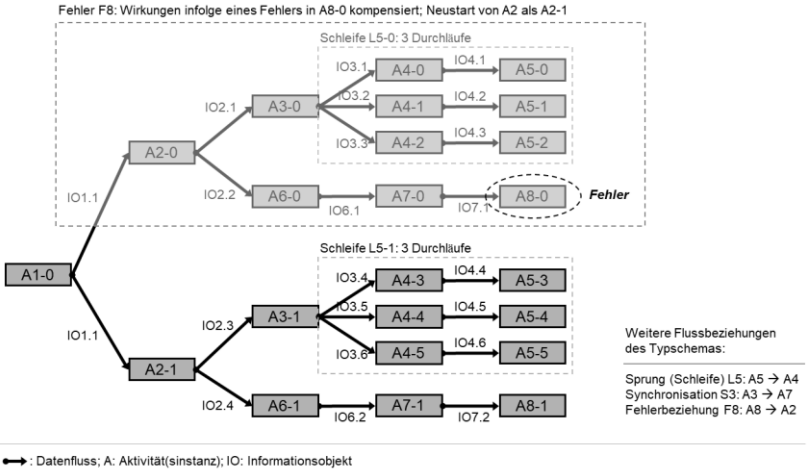


Abbildung 45: Beispiel zur Instanzensicht der Prozessebene (eigene Darstellung)

Die durch Datenflüsse definierte Abfolge der Vorgänge ist entweder sequenziell oder parallel; auf Typebene gegebenenfalls auftretende bedingte Ablaufstrukturen werden auf Instanzebene nach Maßgabe situativer Zustände in lineare Abläufe aufgelöst [Reif03, 23f.]. Flussbeziehungen der Typen Sprung, Synchronisation und Fehler treten auf Instanzebene somit nicht in Erscheinung. Die durch sie spezifizierte Steuerungslogik schlägt sich vielmehr in konkreten Vorgangsfolgen nieder. Das *Ablaufschema* in Abbildung 45 zeigt eine mögliche Realisierung des in Abbildung 40 definierten Prozesses, bei dem die Schleife L5 über die Aktivitäten A4 und A5 insgesamt dreimal durchlaufen wird. Das mit gebrochener Linie dargestellte (untere) Rechteck visualisiert die von der Schleife bedingten Wiederholungen der Aktivitäten, ist jedoch nicht Bestandteil des Schemas. Die Wiederholung von A4 impliziert eine Replikation des Datenflusses D3 zwischen A3 und A4 mit jeweils neuen Instanzen des Informationsobjekttyps IO3. Synchronisationsbeziehungen sind aus dem Vorgangsschema nicht direkt rekonstruierbar. So ist z.B. die Erfüllung der Abhängigkeit S3 der Aktivität A7 von A3 im Schema in Abbildung 45 nur implizit dadurch sichtbar, dass beide Aufgaben durch je einen Vorgang repräsentiert sind und demnach durchgeführt werden konnten.

Die Abbildung zeigt auch die mögliche Auswirkung eines Fehlers: Ein Scheitern von A8-0 (eingekreist) führt gemäß gesetzter Fehlerbeziehung F8 zum Rücksprung zu A2 und zur Kompensation der zwischenzeitlich ausgeführten Aktivitäten (im aufgehellten Rechteck oben). Alle Vorgänge ab A2-0 (einschließlich vorhandener Schleifen) werden demnach wiederholt. Sofern sie nicht vom Fehler betroffen sind, können Informationsobjekte wiederverwendet werden (IO1.1), andernfalls werden sie gelöscht und neue Instanzen erzeugt.

4.5.3.4 Metamodell

Die Modellbausteine der Instanzensicht fasst das Metamodell in Abbildung 46 mit wichtigen Attributen zusammen. Ein **`Vorgang`** ist eine Instanziierung einer **`Aktivität`** und gehört zu genau einer (1,1) **`Prozessinstanz`**, die mindestens einen (1,*) Vorgang umfasst. Ein **`Datenfluss`** verbindet genau zwei (2,2) Vorgänge, die beliebig viele

(0,*) ein- oder ausgehende Datenflüsse besitzen können. Er repräsentiert eine Instanziierung einer **`Datenabhängigkeit`**. Zu einer Aktivität und Datenabhängigkeit können jeweils beliebig viele (0,*) Instanzen existieren; diese folgen jeweils genau einem (1,1) Typ. Vorgänge werden durch ausgefüllte Rechtecke mit Umrandung dargestellt, Datenflüsse als breite Pfeile mit verdickter Basis.

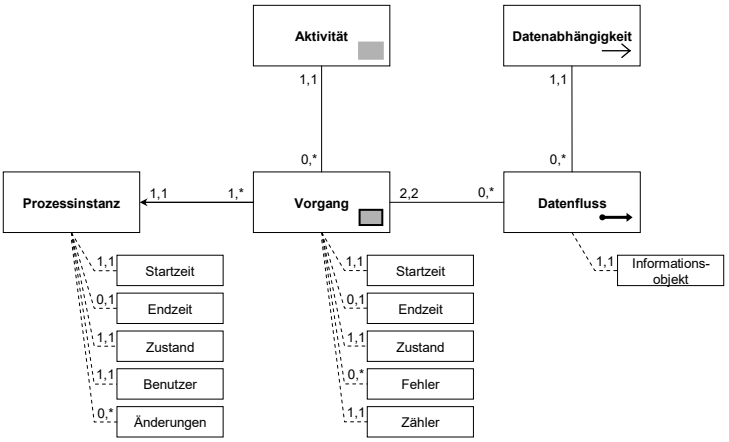


Abbildung 46: Metamodell zur Instanzensicht der Prozessebene (eigene Darstellung)

Das Workflow-Protokoll kann durch Projektion auf relevante Attribute des Vorgangs als sequenzielle Liste von Start- oder Endereignissen dargestellt werden [AcUA12, 355]. Eine typische Repräsentationsform sind XML-Dateien [ADH+03, 246f.]. Die chronologische Abfolge der zur selben Prozessinstanz gehörenden Ereignisse bildet den Pfad der Instanz [Schi04, 268].

Die in den Attributen von Prozessinstanzen, Vorgängen und Datenflüssen erfassten Ablaufdaten bilden die Grundlage für die Prozessrevision. Aus ihnen lassen sich qualitäts-, zeit- und kostenbezogene Kennzahlen berechnen, die in die Beurteilung der Prozessleistung eingehen. Nur wenn die Datenerfassung möglichst einfach und ohne Zusatzaufwand für den Anwender geschieht, ist ihre konsequente Durchführung gewährleistet. Workflow-Management-Systeme erfassen diese Daten vollautomatisiert [Jung02, 78f.].

4.5.4 Bibliothekssicht

Zur Unterstützung der Prozessgestaltung durch Wiederverwendung kommt der Bibliothek von Modellierungsartefakten auf Prozessebene besondere Bedeutung zu. Die Arten zu speichernder Objekte ergeben sich aus den zu unterstützenden Formen der Wiederverwendung und definieren die *Bibliothekssicht*.

Die Wiederverwendung von Analyseprozessen auf Aktivitätsebene ist oft nicht effizient, weil als Vorlage in Frage kommende Workflows oder Prozessausschnitte zu spezifisch sind. Ihr direktes Kopieren in ein neues Projekt ist problematisch, da dessen Rahmenbedingungen mit denen des früheren Projekts nur selten ausreichend übereinstimmen. Wiederverwendung muss deshalb auf verschiedenen Abstraktionsniveaus ansetzen, da die Wiederverwendbarkeit eines Artefakts (einer Komponente) von dessen Allgemeingültigkeit abhängt [WeRü11, 265, 271]. Neben identischer Wiederverwendung einer Komponente C durch vollständige Kopie C' müssen daher mindestens auch spezialisierende und isomorphe Wiederverwendung möglich sein. Eine Spezialisierung von C durch C' liegt vor, wenn Aufgaben und Flüsse von C durch zugehörige Subtypen in C' ersetzt werden. Bei isomorpher Wiederverwendung wird die Struktur von C wiederverwendet, während Aufgaben und Flüsse in C' bei Bedarf durch Abbildungsrelationen verändert werden [Zhug03, 526f.].

In der Bibliothek gespeicherte Artefakte können als Referenzmodelle betrachtet werden, die wichtige Hilfsmittel zur Erreichung von Wiederverwendbarkeit darstellen.[95] *Nichtgenerische Referenzmodelle* werden als Vorlagen bei der Erstellung eines neuen Modellsystems genutzt, gehen dabei aber nicht selbst in dieses ein. Das resultierende Schema ist nicht auf die Referenzmodelle rückführbar. Handelt es sich um nicht vollständige Vorlagen, ist auch der Begriff Strukturmuster gebräuchlich. Ein *Strukturmuster* dokumentiert das Ergebnis eines Entwurfs in einem

[95] Gemäß der wiederverwendungsorientierten Definition von Referenzmodellen genügt die intendierte oder faktische Wiederverwendung eines Modells, um dieses als Referenzmodell aufzufassen. Allgemeingültigkeit oder Einordnung als Best- oder Common-Practice-Modell sind demnach nicht erforderlich [FeBr16].

bestimmten Kontext. Es stellt eine partielle Vorlage dar, die Orientierung bei der Prozesskonstruktion geben kann. *Entwurfsmuster* hingegen dokumentieren punktuelles Wissen über das Vorgehen beim Prozessentwurf in einem bestimmten Kontext.[96] Ist das enthaltene Entwurfsverfahren kontextabhängig parametrisierbar, wird das Muster für eine größere Zahl von Problemen anwendbar. Entwurfsmuster dieser Klasse heißen generisch. Die Generizität eines Musters steigt demnach mit der Zahl der *Kontextparameter*, die nicht durch spezifische Werte vorbelegt sind. *Generische Referenzmodelle* dienen als Ausgangspunkte zur Ableitung neuer Modellsysteme durch Spezialisierung, Detaillierung und Komposition. Das Modellierungsergebnis ist daher auf die Referenzmodelle rückführbar [Sinz97, 14], [HaSW98a, 28f.].

4.5.4.1 Prozessartefakte

Vor diesem Hintergrund werden nach der in Abbildung 47 gezeigten Systematik folgende in der Bibliothek zu speichernde Modellierungsobjekte (*Prozessartefakte*) definiert. ***Prozessbausteine*** stellen tendenziell generische Artefakte dar, die zu Prozessen kombiniert werden können. Elementare Bausteine heißen ***Prozesselemente***, die als *Aktivitäten* (ausführbar) oder *Aufgaben* (nicht ausführbar) ausgeprägt sein können. Nicht elementare Bausteine werden als ***Prozessmodule*** bezeichnet und korrespondieren mit Mustern. Als ***Fragmente*** bestehen sie ausschließlich aus Aktivitäten und sind demnach ausführbar (Strukturmuster). Als ***Schablonen*** können sie auch lediglich aus Außensicht spezifizierte Aufgaben umfassen und sind demnach nicht als Einheit ausführbar. Sie stellen Zwischenstufen der Lösungsentwicklung dar (Entwurfsmuster). Tendenziell nicht generische Artefakte können als ***Prozessvorlagen*** wiederverwendet werden. Sind diese sowohl im Spezifikationsgrad als auch im Umfang vollständig (d.h., sie repräsentieren ein ausführbares, komplettes Prozessschema, bestehend aus Aktivitäten), handelt es sich um ***Workflows***. Diese können zur Anpassung an situative Kontexte modifiziert werden. Nicht vollständige Vorlagen sind wiederum

[96] Struktur- und Entwurfsmuster sind in der Regel Modelle von sehr geringem Umfang [FeBr16]. Sie unterstützen die Zerlegung komplexer Entwurfsprobleme in Teilprobleme, von denen jedes einzelne leichter lösbar ist als das Gesamtproblem [Fers+98, 37].

Prozessmodule, die zur Konstruktion eines vollständigen Workflows um andere Bausteine zu ergänzen sind.[97] Die Artefakte können aus konkreten Prozessausführungen extrahiert oder bewusst als anwendungsunabhängige Module gestaltet werden [EnLS97b, 4f.].

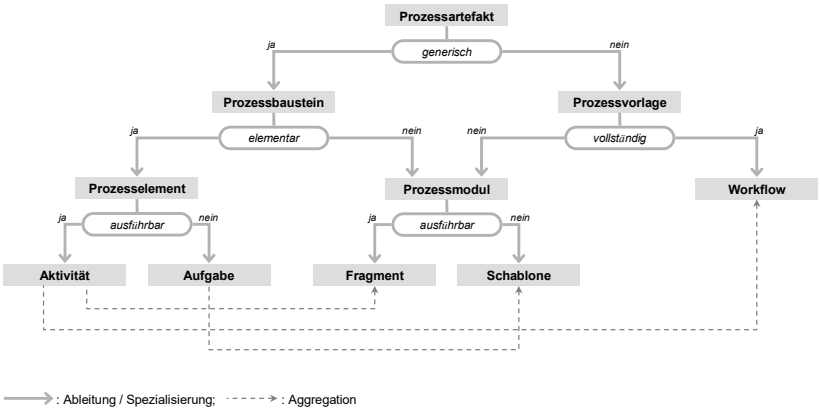


Abbildung 47: Taxonomie von Prozessartefakten (eigene Darstellung)

Prozessbaustein

Ein **`Prozessbaustein`** ist eine Einheit, aus der Prozesse konstruiert werden können. Er kann eine elementare Komponente oder ein Aggregat solch elementarer Komponenten sein. Um die funktionale Eignung eines Bausteins erkennen zu können, ist ihm analog zur Aufgabe genau eine **`Funktion`** zugeordnet. Im Falle eines nicht-elementaren Bausteins erfasst die Funktion das Verhalten aller enthaltenen Komponenten [BeDK04, 254]. Entsprechend verfügt der Baustein wie die Aufgabe über eine Beschreibung seiner `Eingabeflüsse` und `Ausgabeflüsse`, die mit den Ein- und Ausgabedatentypen der Funktion abzustimmen sind. Damit sind die Schnittstellen des

[97] Prozessmodule können im Hinblick auf Generizität unterschiedlich ausgeprägt sein. In ihrer Eigenschaft als Prozessbausteine haben sie eher generischen Charakter. Sie definieren Aufgabenzerlegungen (Konstruktionsverfahren) und sind zu Prozessen kombinierbar. In ihrer Eigenschaft als Prozessvorlagen zeigen sie hingegen auch nicht generischen Charakter. In Form von Fragmenten geben sie Kombination und Konfiguration der enthaltenen Aktivitäten vor (Konstruktionsergebnis).

Bausteins bestimmt, von denen seine Kombinierbarkeit mit anderen Bausteinen abhängt.

Diese Schnittstellen sowie Inhalt und Detailgrad der Funktion definieren die Stellung des Bausteins im Prozessgefüge, die als *Prozesskontext* bezeichnet wird [Enge99, 97]. Weitere *Kontextfaktoren* können im Attribut `Kontext` in Form von Deskriptoren erfasst werden. Diese sind z.B. geeignet, die Einordnung des Bausteins in eine Problemkarte (*Anwendungskontext*), seine Zuordnung zu einem Analyseproblem (*Analysekontext*) und seine Abstimmung auf bestimmte Verfahren oder Daten (*Verfahrens-* bzw. *Datenkontext*) zu beschreiben, und können als Selektionskriterien zur Auswahl passender Bausteine für die Konstruktion eines neuen Prozesses dienen. Sie sind hilfreich, da ein Baustein mit Speicherung in der Bibliothek aus dem Zusammenhang eines Workflows isoliert und Verknüpfungen zu anderen Architekturebenen damit abgetrennt werden. Zur weiteren Unterstützung der Prozessgestaltung können `Kontextregeln` definiert werden, die kontextabhängige Gestaltungshandlungen vorgeben. Beispielsweise kann für einen Baustein festgelegt werden, dass bei Vorliegen eines Kontextfaktors k vor dem Baustein eine Funktion F_k zu realisieren und nach dem Baustein ein Modul M_k einzuplanen ist, um einen effektiven Prozess zu erhalten. Solche *Planungsregeln* sind in Situationen hilfreich, in denen die in den Prozess zu integrierenden Komponenten nicht direkt, sondern über nicht zu bestimmende Zwischenkomponenten mit dem Baustein zu verknüpfen sind.[98] *Konfigurationsregeln* geben Hinweise zur kontextsensitiven Verfahrensparametrisierung von Aktivitäten. Die Modellierung von Kontextfaktoren und Kontextregeln beschreibt Abschnitt 4.7.

Zur Wiederverwendung vorgesehene Artefakte können für den Einsatz in konkreten Situationen unterschiedlich gut geeignet sein. Die Beurteilung dieser Eignung durch den Analytiker ist Voraussetzung für den Abruf möglichst bewährter Bausteine aus der Bibliothek. Die

[98] Die durch Planungsregeln vorgegebenen Bausteine können (bei automatischer Planung) im Sinne einer Agenda noch einzuplanender Elemente verstanden werden (vgl. Abschnitt 5.5.4.1).

Ergebnisse einer detaillierten Bewertung nach einzelnen Revisionskriterien werden am Baustein als `Einsatzbewertung` gespeichert. Eine ganzheitliche Einschätzung der Güte erlaubt die `Einsatznote`. Schlecht benotete Artefakte sollten aus der Bibliothek entfernt oder grundlegend überarbeitet werden. Hinweise hierzu können dem `Einsatzkommentar` entnommen werden. Der `Einsatzzähler` [WRRW05, 9] erlaubt eine rein statistische Messung der Nützlichkeit eines Bausteins anhand der Zahl der Prozessschemata, bei deren Gestaltung er eingesetzt wurde.

Prozesselement

Das Wiederverwendungspotenzial der Prozesselemente liegt in ihrer Parametrisierung. So dokumentieren z.B. Analyseaufgaben, die auf spezielle Analyseprobleme zugeschnitten sind, mit hinterlegten Bewertungsfunktionen, Präferenzrelationen und Anforderungen bewährte Vorgehensweisen, und in der Bibliothek verfügbare Transformationsaufgaben zeigen z.B., welche Maßnahmen der Datenbereinigung beim Zugriff auf spezifische Quellen empfehlenswert sind. Aktivitäten enthalten zusätzlich konkrete Vorschläge zur Verfahrensparametrisierung in dem jeweils abgedeckten Kontext und verbessern die Effizienz der Prozesskonstruktion.

Prozessmodul

Ein Prozessbaustein, der aus einer Menge weiterer Bausteine besteht, wird als **`Prozessmodul`** bezeichnet. Ein Modul stellt allgemein eine abgeschlossene Bau- oder Funktionsgruppe dar, die ausschließlich über festgelegte Schnittstellen mit ihrer Umwelt interagiert [Balz09, 40f.]. Seine interne Struktur repräsentiert einen Teilprozess aus Prozesselementen oder anderen Modulen [BeDK04, 253], der definierte Inputs konsumiert und definierte Outputs produziert. Über die auszutauschenden Ein- und Ausgabeflüsse geht das Modul mit den jeweils beteiligten Elementen seiner Umwelt eine Leistungsvereinbarung ein (Abbildung 48) [ScVr94a, 23], die wiederum durch eine **`Funktion`** beschrieben wird. Das Modul wird bezüglich der Schnittstellendefinition als Black-Box betrachtet und deklariert nur die nach außen sichtbaren Flussbeziehungen [MiMM95, 553] (D1 und D8 im Beispiel in

Abbildung 48); interne Flüsse werden von seinen Komponenten beschrieben. Diese schwache Kopplung mit der Umwelt erlaubt es, ein Modul weitgehend unabhängig zu entwickeln und zu warten. Module gelten daher „im qualitativen und quantitativen Umfang“ als überschaubar und gut verständlich [Balz09, 41].

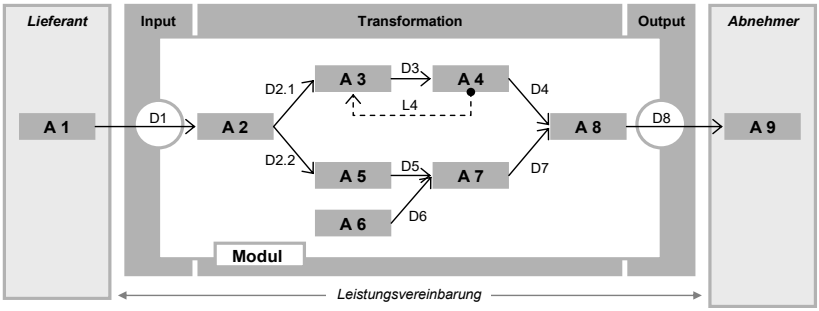


Abbildung 48: Prozessmodul (in Anlehnung an [ScVr94a, 23]), am Beispiel eines Fragments

Abweichend von Funktionsmodulen in der Softwareentwicklung [Balz96, 805f.] wird von Prozessmodulen im hier vorgestellten Sinne nicht gefordert, dass sie ihre interne Struktur nach außen verbergen. Vielmehr sind Module als Muster zu verstehen, die eine Lösung für ein Entwurfsproblem bereitstellen. Sie dienen dazu, eine Aufgabe, deren Funktion mit der Funktion des Moduls übereinstimmt, zu detaillieren, indem die Aufgabe durch den im Modul enthaltenen Teilprozess ersetzt wird. Ein Modul tritt somit im Prozessschema typischerweise nur mittelbar durch das aus seiner Anwendung resultierende Prozesssegment in Erscheinung. Die Entstehung eines Prozessbausteins in einem konkreten Prozessschema aus einem Modul wird im Attribut `Genese` des Bausteins dokumentiert und ist damit nachvollziehbar. Wird ein aus einem Modul hervorgegangener Prozessausschnitt verändert, so betrifft dies nicht die Definition des Moduls, sondern lediglich das Prozessschema, dessen Bestandteil der Ausschnitt ist. Im Rahmen der Revision kann später entschieden werden, ob vorgenommene Änderungen Auswirkungen auf die Modulbibliothek nehmen sollen und ob dies durch Kreieren eines neuen oder durch Modifikation des ursprünglichen Moduls geschieht. Zur Unter-

scheidung verschiedener Fassungen eines Bausteins werden für alle Prozessartefakte `Version` und `Datum` gespeichert.

Fragment

Ein Prozessmodul, das mehrere `Aktivitäten` umfasst, wird als (Prozess-) *Fragment* bezeichnet. Ein **`Fragment`** ist bei Erfüllung seiner Vorbedingungen ausführbar und stellt bezüglich seiner Spezifikationstiefe eine gültige Teillösung für das Prozesskonstruktionsproblem dar. Die vom Fragment gelieferte Gestaltungsvorlage kann beliebig modifiziert werden [AAD+04, 16], ist jedoch mithilfe anderer Prozessmodule nicht weiter verfeinerbar. Fragmente sind ebenso wie die Aktivitäten, auf denen sie beruhen, stets an ein konkretes Datenanalysesystem bzw. den zugehörigen Operatorenpool gebunden. ENGELS extrahiert aus früheren Abläufen Aktivitätsfolgen als Teillösungen für die Datenanalyse im Sinne von Fragmenten (Reusable Process Units), die jedoch nicht anwendungsspezifisch und nur teilweise auf konkrete Datenquellen ausgerichtet sind [Enge99, 94f.].

Schablone

Ein Prozessmodul, das mehrere `Aufgaben` umfasst, heißt (Prozess-) *Schablone*. Eine **`Schablone`** ist eine Vorlage mit abstrakten Elementen, die etwa durch Spezialisierung oder durch Auswahl und Instanziierung von Operatoren konkretisiert werden kann. Sie dient insbesondere der Beschreibung beliebiger Stufen von Aufgabenzerlegungen, dokumentiert also Konstruktionsanweisungen. Sie besitzt hohes Wiederverwendungspotenzial, da ihre abstrakten Komponenten an zahlreiche konkrete Situationen anpassbar sind [HaSW98a, 28f.], [RuPR99, 228]. In ihrer Eigenschaft als Aufgaben können auch Aktivitäten Bestandteil von Schablonen sein. Partiell abstrakte Schablonen erlauben das bewusste Offenhalten von Anpassungsspielräumen, etwa zur Verbreiterung des Einsatzspektrums von Fragmenten oder im Falle von Informationsmangel. Mit Integration einer Aktivität verliert eine Schablone jedoch ihre Aufgabenträgerunabhängigkeit, weshalb diese Mischform zu vermeiden ist. Ausschließlich auf Aufgabenebene beschriebene Schablonen sind in zahlreichen Situationen einsetzbar [RuPR99, 226f.]. Diese Eigenschaft wird durch das Attribut `Reinheit` gekennzeichnet.

Schablonen für die Analyseprozesskonstruktion schlagen auch KIETZ ET AL. [KSBF10] vor. Data Mining Workflow Templates repräsentieren abstrakte Knoten innerhalb einer Aufgabenzerlegung und werden in Anlehnung an die aus der Wissensakquisition bekannten Problem Solving Methods (PSM) definiert.[99] Die von den Schablonen spezifizierten Zerlegungsregeln funktionieren ähnlich wie eine kontextfreie Grammatik, die Aufgaben als nicht-terminale, Operatoren als terminale Symbole nutzt [KSBF10, 6]. Bereits ENGELS verwendet den Ansatz der PSM und erweitert ihn um Fragmente, um abstrakte Elemente auszufüllen [Enge99, 95].

Workflow

Ein vollständiger Datenanalyseprozess auf Aktivitätsebene wird als *Workflow* bezeichnet. Er ist für ein konkretes Analyseproblem und die vorliegende Systemumgebung individualisiert und ausführbar (vgl. [RuPR99, 228]). Prozessschemata, die ein nicht auf Aktivitätsebene spezifiziertes Element enthalten, stellen demnach Schablonen dar. Ein **`Workflow`** ist durch die Menge seiner Aktivitäten und die zugehörigen Flussbeziehungen definiert. Seine explizite Modellierung als Metaobjekttyp ist aus Bibliothekssicht sinnvoll, um ihm bzw. seinen Elementen zugeschriebene Eigenschaften zentral speichern und beim Abruf aus der Bibliothek direkt selektieren zu können. Diese Eigenschaften umfassen neben `Version` und `Datum` die bereits vom Baustein bekannten Beurteilungen (`Einsatzzähler`, `Einsatzbewertung`, `Einsatznote`, `Einsatzkommentar`) und den `Kontext`. Die Kontextdeskriptoren aggregieren alle relevanten Faktoren der enthaltenen Aktivitäten, während die Beurteilung des Workflows in der

[99] Die Wissensakquisition kennt verschiedene Ansätze zur Zerlegung komplexer Expertenaufgaben in weniger komplexe Unteraufgaben. Dabei wird epistemologisches von domänenspezifischem Wissen getrennt und von Aufgabenträgeraspekten abstrahiert. Dem liegt die Annahme zugrunde, dass Problemlösewissen generisch formuliert und auf verschiedene Domänen übertragbar ist [Enge99, 102]. Einen Überblick über entsprechende Ansätze geben z.B. [StBF98].

Regel separat von den Aktivitäten erfolgt.[100] Zur Kennzeichnung des Zwecks des Workflows wird das `Analyseproblem` aus der Analyseaufgabe übernommen. Eine neue `Version` eines Workflow-Schemas entsteht, sobald eine Aktivität oder ein Fluss verändert wird [RWRW05, 255]. Die Historie der `Änderungen` gegenüber der zugrundeliegenden Version wird protokolliert. Die Wiederverwendung vollständiger Workflows wird auch im Ansatz von ENGELS unterstützt, der neben der Komplettlösung auch Problemspezifikation, Aufgabenzerlegung, Datencharakteristika, Kontext und die Ergebnisse der Anwendung (Projektresultate) speichert [Enge99, 94f.].

4.5.4.2 Metamodell

Die Metaobjekttypen der Bibliothekssicht und ihre Beziehungen zeigt Abbildung 49 im Metamodell. Auf die Darstellung von Attributen wird zur besseren Übersichtlichkeit verzichtet. Jeder **`Prozessbaustein`** referenziert genau eine (1,1) **`Funktion`**, die dessen Sachziel beschreibt. Zu einer Funktion können beliebig viele (0,*) Bausteine in der Bibliothek vorliegen. Prozessbausteine können durch beliebig viele (0,*) **`Flussbeziehungen`** verknüpft sein, die den in der Aufgabensicht beschriebenen Typen folgen. Eine Flussbeziehung verbindet jeweils genau zwei (2,2) Prozessbausteine. Ein Prozessbaustein kann als Aufgabe oder Prozessmodul ausgeprägt sein. Ein **`Prozessmodul`** ist eine **`Schablone`**, wenn es mindestens zwei (2,*) **`Aufgaben`** enthält, und es ist ein **`Fragment`**, wenn es mindestens zwei (2,*) **`Aktivitäten`** umfasst. Aufgrund der Spezialisierungsbeziehung zwischen Aufgabe und Aktivität kann eine Schablone auch Aktivitäten enthalten. Eine Aufgabe kann in beliebig vielen (0,*) Schablonen vorkommen. Mehrere (1,*) Aktivitäten bilden einen **`Workflow`**. Eine in der Bibliothek gespeicherte Aktivität kann jeweils Bestandteil beliebig vieler (0,*) Fragmente und Workflows aus der Bibliothek sein. In der Prozessbibliothek werden zunächst alle Workflows mit den zugehörigen Aktivitäten abgelegt. Auf ihrer Basis können Fragmente sowie

[100] Kontextregeln sind am Workflow nicht erforderlich, da diese stets Aktivitäten betreffen und dort hinterlegt sind. Für den Workflow geltende Ablaufvarianten sind als bedingte Verzweigungen zu modellieren.

Schablonen definiert werden. Im Zuge einer regelmäßigen Wartung der Bibliothek werden schlecht bewertete oder selten benutzte Artefakte entfernt. Zugleich ist in vielen Fällen eine Zusammenfassung (Abstraktion und Generalisierung) von Artefakten sinnvoll (vgl. Abschnitt 7.4.2.3). Daher sind einzeln oder als Bestandteil eines Fragments in der Bibliothek auftretende Aktivitäten möglich, während der ursprünglich zugehörige Workflow nicht mehr enthalten ist.[101] Prozessartefakte und Prozesselemente werden nicht explizit, sondern in Form ihrer jeweiligen Subtypen repräsentiert.

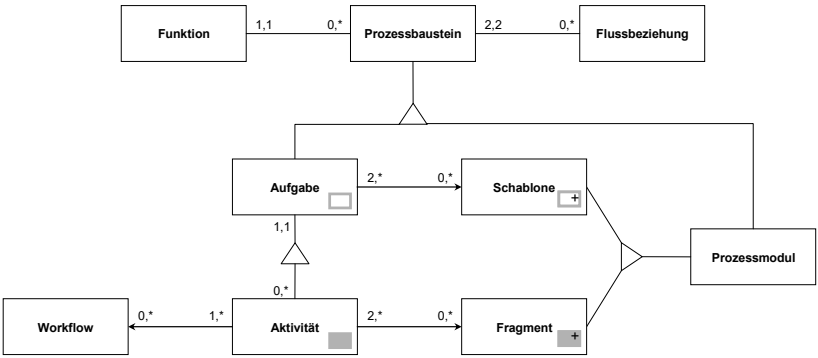


Abbildung 49: Metamodell zur Bibliothekssicht der Prozessebene (eigene Darstellung)

Werden Schablonen und Fragmente als Einheit in ein Prozessschema integriert, werden sie grafisch mit den Symbolen der Aufgabe bzw. Aktivität dargestellt, in die zur Kennzeichnung ihrer Eigenschaft als Aggregat ein Pluszeichen (+) eingefügt ist. Die Integration ganzer Module (aggregierte Operatoren, Meta Nodes, Building Blocks) in Workflows ist bei gängigen Analysewerkzeugen üblich [Bert+09, 29], [Rapi10, 21] und wird daher vom Modellierungsansatz unterstützt. Gleichwohl ist zu empfehlen, die von den Modulen repräsentierten Subprozesse mit ihrer Integration zu expandieren und das Modul gewissermaßen im Schema aufzulösen. Dieses Vorgehen folgt der Sichtweise, nach der ein Prozessschema stets einen Schnappschuss der

[101] Diese Aussage gilt für die Bibliothekssicht. Außerhalb dieser Sicht gehört eine Aktivität stets implizit zu einem Workflow, der jedoch nicht explizit repräsentiert wird.

Prozessentwicklung repräsentiert und demnach entweder abstrakte Elemente oder deren Detaillierung enthält (jedoch keine Aggregate).

4.5.5 Zusammenfassung zur Prozessebene

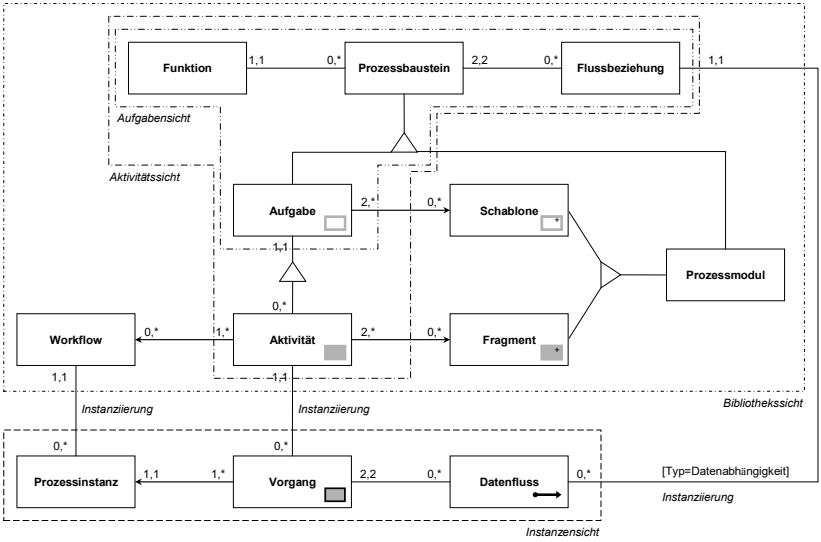


Abbildung 50: Integriertes Metamodell zur Prozessebene (eigene Darstellung)

Die in Abbildung 50 im integrierten Metamodell abgegrenzten Sichten der Prozessebene erlauben eine vollständige Beschreibung eines Analyseprozesses im Hinblick auf Inhalt und Reihenfolge auszuführender Transformationen (Aufgabensicht), Zuordnung und Konfiguration von Operatoren (Aktivitätssicht), Definition und Annotation von Prozessbausteinen (Bibliothekssicht) sowie realisierte Prozessabläufe (Instanzensicht). Die Prozessinstanzen der Instanzensicht dienen nicht der Wiederverwendung und sind daher nicht Bestandteil der Bibliothekssicht. Die Attribute der Metaobjekttypen sind in Anhang A4.3 dokumentiert.

Die Prozessebene folgt der Basismetapher eines ereignisgesteuerten Ablaufs von Aktivitäten, der einerseits auf Typ- und Instanzebene, andererseits sowohl als vollständiger Prozess als auch in Form

wiederverwendbarer Bausteine beschrieben wird. Jeder Ablauf dient der Lösung eines Analyseproblems und bildet zusammen mit diesem jeweils einen Analysefall.

Die Differenzierung zwischen Aufgaben und Aktivitäten erlaubt es, Prozesse unabhängig von Ressourcen zu spezifizieren und in dieser Form als allgemeingültige Empfehlungen zu verwenden. Insbesondere können damit auch verbreitete Prozessmodelle wie etwa CRISP-DM oder abstrakte Prozessmuster, wie sie z.B. das MINING-MART-Projekt für spezielle Aufgabenstellungen (z.B. Kreuzvalidierung [KnSB00, 20-25]) bereitstellt, abgebildet und nahtlos in die Prozessplanung integriert werden.[102] Durch Zuordnung von Ressourcen lassen sich diese Spezifikationen innerhalb desselben Modellierungsansatzes durchgängig zu werkzeugspezifischen Workflows weiterentwickeln. Ebenso können Workflows durch Abstraktion von Ressourcenaspekten auf Aufgabenebene verallgemeinert und in anderen Systemumgebungen weiterverwendet werden.

4.6 Ressourcenebene: Aufgabenträger und Daten zur Analyse

Zur Realisierung von Analyseprozessen bereitstehende Aufgabenträger, zur Analyse genutzte Daten sowie in ihrem Verlauf produzierte Zwischen- und Endergebnisse werden auf Ressourcenebene der Datenanalysearchitektur beschrieben. Mit Ausnahme der Zwischen- und Endergebnisse handelt es sich dabei um Ressourcen, die auch unabhängig von einer Analyse existieren und in mehreren Projekten zum Einsatz gelangen können. Ihre Modellierung verspricht daher projektübergreifenden Nutzen und hilft darüber hinaus, einen Überblick über die Fähigkeiten des verfügbaren Werkzeug-Arsenals und Personals sowie über Struktur und Inhalt existierender Datenquellen zu erlangen.

Die *Datensicht* modelliert passive Ressourcen als Datenobjekttypen. Informationsobjekttypen verfügen zusätzlich über semantische Annota-

[102] Prozessmodelle können im Sinne generischer Initialmodelle (Frameworks) verwendet werden, die mithilfe von Mustern oder durch freie Entwurfsentscheidungen konkretisiert werden können (vgl. [HaSW98a, 29], [JaVW00, 9]).

tionen, um den Inhalt der Daten darzustellen. Die *Aufgabenträgersicht* beleuchtet aktive maschinelle und personelle Ressourcen auf Typebene. Maschinelle Aufgabenträger werden in Form von Software-Produkten (Services) beschrieben, die eine Menge von Operatoren bereitstellen, auf die bei der Workflow-Spezifikation zurückgegriffen werden kann. Zusätzlich können personelle Aufgabenträger als Rollen abgebildet werden, um Fähigkeiten und Kostensätze geeigneter Mitarbeiter darzustellen. Die *Instanzensicht* stellt schließlich konkrete Ausprägungen der Datenobjekttypen und Aufgabenträger in Form von Datenquellen und Informationsobjekten bzw. Software-Installationen und Personen sowie relevante Beziehungen zwischen ihnen dar.

4.6.1 Datensicht

Die Modellierung der Datensicht nimmt für die Datenanalyse naturgemäß eine zentrale Rolle ein. Die Repräsentation sollte einerseits einheitlich und leicht verständlich sein, muss andererseits aber zugleich alle relevanten Datentypen einschließlich komplexer Strukturen und unstrukturierter Formate abbilden können (vgl. [Hein+08, 448]), wie sie als Rohstoff oder als Ergebnis von Datenanalysen auftreten. Darüber hinaus sollte die Repräsentation die (automatisierte) Prozessplanung unterstützen.

4.6.1.1 Datenobjekttyp

Die Einheiten zur Repräsentation bzw. Speicherung von Daten werden auf Typebene als Behälter definiert, die eine Struktur besitzen können und Einschränkungen über die zulässigen Datenwerte treffen. Diese Definition wird als *Datenobjekttyp* bezeichnet und erfolgt anhand der Merkmale Konstruktor und Wertebereich [FeSi13, 324]. Elementare Datenobjekttypen (Skalare) besitzen keine interne Struktur. Sie können nur Einzelwerte speichern und sind ausschließlich über ihre Wertebereiche bestimmt. Beispiele sind die aus Programmiersprachen bekannten Datenobjekttypen `Integer` [-32.768; 32.767] und `Character`

[0; 255].[103] Strukturierte Datenobjekttypen bestehen aus einem Gefüge anderer Datenobjekttypen, das durch den Konstruktor bestimmt ist. Bekannte Konstruktoren sind etwa `Record`, `Array`, `Set` [FeSi13, 324].

Diese Basisdefinition eines Datenobjekttyps wird von HEINRICH ET AL. [Hein+08] erweitert, um eine *Datendeklaration* für Ein- und Ausgabeparameter von Prozessaktivitäten zu erhalten, die ein semantisches Prozessplanungswerkzeug verarbeiten kann. Hierzu werden elementare Datenobjekttypen durch das Tupel (`Name`, `Wertebereich`, `Restriktionen`) bestimmt. Restriktionen erzeugen stets eine nicht-leere Teilmenge des Wertebereichs und dienen zu dessen Einschränkung auf zulässige Werte. Zusammengesetzte Datenobjekttypen können aus elementaren Datenobjekttypen aggregiert werden und erlauben die Abbildung von `Record`-Strukturen. Der Wertebereich wird als primitiver Datentyp oder als ontologische Klasse dargestellt [Hein+08, 451]. Zur Abdeckung beliebiger komplexer Datenobjekttypen wird dieser Ansatz im Folgenden verallgemeinert. Die Spezifikation des Wertebereichs kann nunmehr außer durch Angabe eines primitiven oder bereits existierenden Datenobjekttyps auch unter Verwendung eines *Konstruktors* erfolgen. Der Konstruktor wird notiert mit dem Bezeichner des `Wertebereichs`, gefolgt von Argumenten in geschweiften Klammern. Die Argumente bestimmen jene Datenobjekttypen, aus denen die Struktur zu bilden ist. Er definiert einen neuen **`Datenobjekttyp`** unter dem angegebenen `Namen`, z.B. `(_LabelledTable, _Table{_InputColumns, _TargetColumn}, notEmpty())`. Folgen dem Wertebereichsbezeichner keine Argumente in Klammern, so wird keine neue Struktur definiert, sondern lediglich ein vom Wertebereichstyp abgeleiteter Datenobjekttyp deklariert, z.B. `(_PositiveInteger, _Integer, >=0)`. Zur Kennzeichnung von Datenobjekttypen wird ihrem Namen per Konvention stets ein Unterstrich vorangestellt.

[103] `Character` ist intern als 8-Bit-`Integer` definiert. Die Zahlenwerte werden auf zu wählende Zeichentabellen abgebildet, wie z.B. ASCII oder ISO-8859-1, und können insgesamt 256 Zeichen darstellen.

4.6.1.2 Informationsobjekttyp

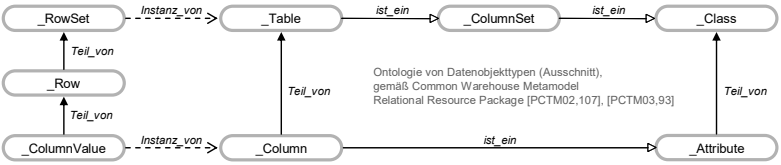
Datenobjekttypen dienen als definierte Behälter für Daten, deren Name typischerweise den Wertebereich (z.B. `_Integer`, `_Character`) oder die interne Struktur (z.B. `_Table`, `_Record`) beschreiben und somit eher syntaktischen Bezug nehmen. Datenobjekttypen, die zusätzlich eine Semantik besitzen, die sich z.B. im Inhalt (z.B. Kundenbeschwerden) oder in ihrer Rolle für eine Prozessaktivität (z.B. Input, Trainingsdaten) ausdrückt, heißen *Informationsobjekttypen*. Sie werden als Spezialisierung (Erweiterung) von Datenobjekttypen modelliert.

Der Begriff Informationsobjekt wird u.a. auch im Prozess- und Workflow-Management sowie im Information Retrieval [ScNZ95, 429], [RaMe95], [InJä05] gebraucht. „In Analogie zu den Werkstücken der Fertigungsprozesse werden in informationellen Prozessen Informationsobjekte ‚produziert'" [RaME95, 469].[104] Ein **`Informations-objekttyp`** kann Eingabedaten, Zwischen- und Endergebnisse, Präsentationen oder Abschlussberichte enthalten. Ihre Semantik vermittelt ein aussagekräftiger `Name`, der im Gegensatz zum Datenobjekttyp ohne Unterstrich notiert wird. Deutlicher wird die konkrete Bedeutung durch Angabe eines `Domänenobjekts`, eines Domänenobjekt-`Merkmals` sowie durch eine `Beschreibung`. Die Zuschreibung zu Domänenobjekten erlaubt z.B. die Auszeichnung von Tabellen oder Dokumenten mit den in ihnen beschriebenen Konzepten, die mithilfe der Merkmale z.B. auf Spaltenebene weitergeführt wird. Weiterhin kann der Informationsobjekttyp als persistent zu speichernde oder nur temporär als Speicherrepräsentation verfügbare Struktur gekennzeichnet werden (`Persistenz`).

Abbildung 51 illustriert die Spezifikation von Daten- und Informationsobjekttypen anhand von Beispielen. Deklaration 1 erzeugt einen Datenobjekttyp `_AktivStatus` vom Typ `_String`, der gültige Ausprägungen enumeriert. Beispiel 2 deklariert einen Datenobjekttyp `_LabelledTable` mithilfe des Konstruktors `_Table` aus zwei Kompo-

[104] Der Begriff erscheint vor dem Hintergrund des Informationsverständnisses aus Abschnitt 2.1.2.2 insbesondere in Bezug auf die Endprodukte eines Analyseprozesses gerechtfertigt.

nenten, einem nicht-leeren `_ColumnSet` für Eingabevariablen und einer `_Column` für die Zielvariable. Die Komponenten können separat oder in die Tabellendeklaration geschachtelt („inline“) deklariert werden; in beiden Fällen sind sie unter den vergebenen Namen `_InputColumns` bzw. `_TargetColumn` referenzierbar. Die Abbildung visualisiert den geschachtelten Fall zur besseren Lesbarkeit durch Ausschwenken der untergeordneten Deklarationen. Beispiel 3 zeigt die Definition eines Informationsobjekttyps `Adresse` als Tabelle mit vier Spalten, die ihrerseits Informationsobjekttypen sind. In Beispiel 4 ist eine typische Deklaration für Eingabedaten einer Aktivität dargestellt. Der Fluss `Input1` erwartet eine beliebige, nicht-leere Relation, ohne deren Struktur weiter einzuschränken. Undefinierte Elemente sind jeweils mit einem Unterstrich („_“) notiert.




```
1.  (_AktivStatus, _String, {aktiv, inaktiv, gekündigt})

2.  (_LabelledTable, _Table{■}, _)
2 a.                          (_InputColumns, _ColumnSet, notEmpty())
2 b.                          (_TargetColumn, _Column,    _)

3.  (Adresse, _Table{■}, _)
3 a.                  (Straße, String,  size()<=30)
3 b.                  (Nr,     Integer, >0)
3 c.                  (PLZ,    String,  size()=5)
3 d.                  (Ort,    String,  size()<=30)

4.  (Input1,  RowSet, notEmpty())
```

Abbildung 51: Ontologie und Beispiele zur Deklaration von Daten- und Informationsobjekttypen (vgl. zur Ontologie [PCTM02,107], [PCTM03,93])

Der Ontologieausschnitt im oberen Teil der Abbildung orientiert sich am COMMON WAREHOUSE METAMODEL (CWM) und verdeutlicht, dass ein Konstruktor stets jene Elemente als Argumente erwartet, die mit

ihm über eine inverse Teil_von-Beziehung verbunden sind, im Falle von `_Table` also `_Columns` (bzw. die inferenziell ableitbaren `_ColumnSets`). Wertebereich und Restriktionen können in Prädikate übersetzt und auf semantische Relationen einer Ontologie abgebildet werden. So ist der Wertebereich implizit stets mit dem Prädikat `ist_ein(Name, Wertebereich)` verbunden, und ein Konstruktor repräsentiert zusätzlich das Prädikat `hat_Teil(Name, {Komponenten})`. Restriktionen implizieren das Prädikat `hat_Restriktion(Name, {Restriktionen})`, das für speziellere Einschränkungen entsprechend detailliert werden kann.

Derart hierarchische Datendeklarationen sind in Datenanalysewerkzeugen durchaus gebräuchlich. So repräsentiert z.B. das System IGOR elementare Variablen durch Frames und setzt strukturierte Datenobjekttypen aus Frame-Hierarchien zusammen [AmCo94b, 46]. Da sich Vor- und Nachbedingungen von Prozessaktivitäten ausschließlich auf Flussbeziehungen richten, können sämtliche Bedingungen mithilfe von Restriktionen der Datenobjekttypen abgebildet werden [Hein+08, 448f.]. Restriktionen eignen sich ebenso zur Spezifikation von Kardinalitäten.

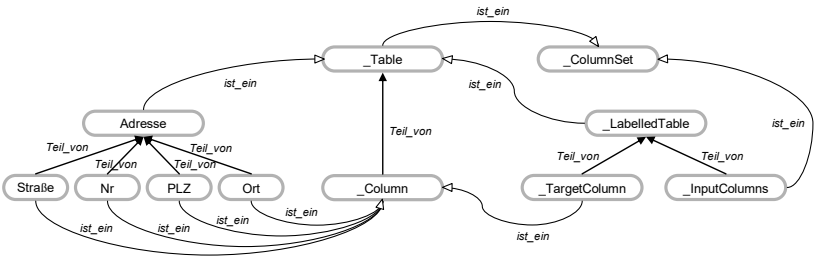


Abbildung 52: Beispiele zur Visualisierung der Beziehungen zwischen Daten- und Informationsobjekttypen (eigene Darstellung)

Für Daten- und Informationsobjekttypen ist keine eigene graphische Darstellung vorgesehen. Vielmehr können die zwischen ihnen bestehenden Ableitungs- und Strukturbeziehungen durch Repräsentation in einer Ontologie visualisiert werden. Abbildung 52 zeigt dies exemplarisch anhand der Beispiele 2 und 3 aus Abbildung 51.

4.6.1.3 Metamodell

Abbildung 53 präsentiert das Metamodell der Datensicht mit vollständigen Attributmengen. Die Deklaration von Daten- und Informationsobjekttypen kann indes nicht die konzeptuelle oder logische Datenmodellierung ersetzen. Entsprechende Schemata sind insbesondere zur Dokumentation der Inhalte von Datenquellen hilfreich und können mit den jeweiligen Metaobjekttypen über das Attribut `Link` verknüpft werden. Datendeklarationen dienen der Festlegung zulässiger Datenausprägungen für Prozessbausteine einschließlich der Möglichkeit, bestimmte Elemente dieser Strukturen über einen Namen referenzierbar zu machen.

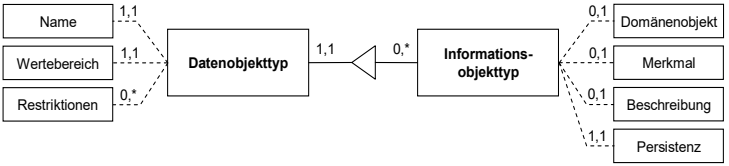


Abbildung 53: Metamodell zur Datensicht der Ressourcenebene (eigene Darstellung)

4.6.2 Aufgabenträgersicht

Die Auswahl maschineller Aufgabenträger für einen Analyseprozess geschieht über die Zuordnung geeigneter Datenanalyse- oder Datentransformationsverfahren zu Aktivitäten (Prozessebene). Das Lösungsverfahren ist Bestandteil der Innensicht einer Aufgabe und nimmt stets Bezug auf einen Aufgabenträgertyp [Fers92, 6f.], [FeSi13, 98]. Somit wird mit Zuordnung des Verfahrens zugleich eine Entscheidung für jene Klasse von Aufgabenträgern getroffen, die das jeweilige Verfahren implementieren.

4.6.2.1 Operator

Lösungsverfahren werden durch *Operatoren* repräsentiert. Ein **`Operator`** wird in der Literatur als einfache Abstraktion eines Verfahrens

(einer Methode)[105] bezeichnet, die auf einem Input eine bestimmte Aktion durchführt (Vorgangstyp) und als Ergebnis einen Output produziert [MWK+06, 2], [HKNW09, 81f.]. Er repräsentiert eine konkrete Implementierung eines Algorithmus und wird unter Vernachlässigung seiner Innensicht nur im Hinblick auf seine Schnittstelle aus Ein- und Ausgabedaten sowie Verfahrensparametern beschrieben. Der Operator ist demnach eine strukturelle Repräsentation eines Lösungsverfahrens aus Außensicht (vgl. [Fers92, 7]).

Das Verhalten des Lösungsverfahrens wird durch eine `Funktion` beschrieben, die dem Operator aus der Prozessebene zugeordnet wird. Die Funktion kennzeichnet die vom Verfahren veranlasste Transformation von Ein- in Ausgabedaten des jeweiligen Typs [AmCo94, 51], [Enge99, 128f.] im Sinne eines Zustandsübergangs des Aufgabenobjekts [Fers92, 7].

Die Verhaltensbeschreibung von Verfahren in Form von Funktionen hat eine Reihe von Implikationen. Erstens unterstützt die Verknüpfung von Funktionen sowohl mit Aufgaben als auch mit Operatoren wirksam die *Verfahrensauswahl*, indem (unter der Annahme einer korrekten und zweckmäßigen Funktionsdefinition und -zuordnung) für eine Aktivität nur jene Operatoren geeignet sind, denen dieselbe Funktion zugeordnet ist wie der Aktivität selbst. Aufgrund der einheitlichen Funktionsbeschreibung können Aufgaben und Verfahren leicht aufeinander abgebildet werden [EnLS97, 164], [GCC+04, 323]. Dadurch ist die Alternativenmenge in realen Situationen in der Regel auf einen oder wenige Operatoren eingeschränkt. Zweitens definieren die verfügbaren Operatoren zugleich die *Granularität der Aktivitäten*, da jede Aktivität durch genau einen Operator realisiert wird (vgl. Abschnitt 4.5.2). Drittens wird deutlich, dass Funktionen idealerweise durch *schrittweise Abstraktion von Operatoren* definiert werden. Durch Fallenlassen von Details der verarbeiteten Daten (Aufgabenobjekt) sowie der auf diesen Daten durchgeführten Transformation (Sachziel) lässt sich leicht eine

[105] Die Begriffe Verfahren, Methode und Algorithmus werden im Weiteren synonym verwendet, soweit sie Lösungsverfahren beschreiben. Eine (Datenanalyse-) Methode ist definiert als Algorithmus zur Durchführung einer (Datenanalyse-) Aufgabe [KlZy96, 575].

Funktionstaxonomie aufbauen (vgl. Abschnitt 7.4.2.3). Abstraktere Funktionen repräsentieren somit eine größere Klasse von Operatoren. Viertens können die Zwischenknoten der *Operatorenbäume*, wie sie in Datenanalysewerkzeugen typischerweise zur Strukturierung des Verfahrenspools genutzt werden, unmittelbar als Funktionen im hier gebrauchten Sinne interpretiert werden. Diese sind jedoch auf Attributebene um die zugehörigen Spezifikationen zu ergänzen.

Die Spezifikation des Operators umfasst die benötigten `Eingabedaten` und produzierten `Ausgabedaten`. Sie korrespondieren mit den Attributen `Eingabedatentyp` bzw. `Ausgabedatentyp` der Funktion derart, dass ihre Wertebereiche Teilmengen jener Wertebereiche der in der Funktion beschriebenen Datenobjekttypen sein müssen. Um aussagefähige Annotationen zu erhalten, sollte die direkt zugewiesene Funktion eine exakte Übereinstimmung aufweisen. Bei der Deklaration der Ausgabedaten können Informationsobjekttypen angegeben werden, die mit den Aussagetypen der Analysefrage korrespondieren [WWSE96, 216] (vgl. Abschnitt 4.4.1). Beispielsweise benötigt der Operator eines überwachten Lernverfahrens zur Erzeugung eines Prognosemodells Eingabedaten vom Typ `_LabelledTable` und produziert ein `_PredictiveModel`. Mithilfe von `Schaltbedingung` und `Schaltwirkung` kann das Verhalten des Verfahrens in Bezug auf die Ein- und Ausgabedaten dargestellt werden. Sie sind als logische Ausdrücke formuliert und dienen, wie in Abschnitt 4.5.2 erläutert, der Spezifikation der formalen Semantik zum Zwecke der Prozessverifikation oder -validierung. Insbesondere können damit obligatorische und optionale Ein- und Ausgabedaten bestimmt werden.[106]

Große Bedeutung zur wirksamen Unterstützung der Prozessgestaltung hat die Beschreibung und Erläuterung der `Parameter` eines Verfahrens, die zur Auslösung des Vorgangs übergeben und nach Beendigung

[106] Ein Operator muss Eingabedaten indes nicht zwingend konsumieren, sondern kann diese auch uninterpretiert an nachfolgende Operatoren weiterreichen. Dieses „Durchschleusen“ wird von einigen Analysewerkzeugen unterstützt, um die Flexibilität der Konstruktion sowie die Übersichtlichkeit der Darstellung von Workflows (durch Vermeidung weiterer/langer Datenabhängigkeitskanten) zu erhöhen. Häufig durchgeschleuste Datenobjekte sind etwa Beispielmengen oder Prognosemodelle [MKFR03].

zurückgeliefert werden [Fers92, 7], [GCC+04, 323]. Algorithmusparameter dienen der Beeinflussung des Methodenverhaltens und spezifizieren die erlaubten Werte [WWSE96, 216]. Die Definition eines Parameters erfolgt analog zum Datenobjekttyp und umfasst zunächst `Name`, `Wertebereich` und `Restriktionen`. Beispielsweise bestimmt das Tupel (`Anzahl Cluster`, `_Integer`, `in(1..100)`), dass der Parameter „Anzahl Cluster“ vom Typ Integer und auf Werte von 1 bis 100 beschränkt ist. Diese Typbeschreibung wird erweitert um das Element `Wert`, das einen Standardwert als Vorgabe enthalten kann. Die `Kardinalität` definiert minimale und maximale Zahl von Ausprägungen und legt damit auch fest, ob der Parameter obligatorisch oder optional ist. Der `Typ` gibt an, ob es sich um einen Eingabe- (`Typ`=in), Ausgabe- (out) oder einen referenzierten Parameter (inout) handelt. Ausgabeparameter übermitteln z.B. Status- oder Fehlermeldungen. Schließlich sollte jedem Parameter eine ausführliche `Beschreibung` und ein `Hilfetext` beigefügt werden, um seine Bedeutung für den Analytiker greifbar zu machen und die Wahl konkreter Werte zu erleichtern.

Die Spezifikation eines Operators wird durch einen `Namen`, der typischerweise auf die spezifische Version oder Implementierung eingeht, und eine optionale inhaltliche `Beschreibung` ergänzt. Zur Auslösung des Verfahrens kann eine `Aufrufnachricht` definiert werden [ZLKO97, 293]. Sie darf Variablen enthalten, die vor dem Absetzen zur Laufzeit mit konkreten Werten (etwa dem URL eines Servers) instanziiert werden. Ihre konkrete Form und Realisierung sind implementierungsabhängig.

In Situationen, in denen mehr als ein geeigneter Operator zur Realisierung einer Aktivität zur Verfügung steht, kann die Auswahl durch weitere Verfahrenseigenschaften gelenkt werden. Diese Eigenschaften sind in anwendungs-, daten- und methodenorientierte Aspekte gegliedert und werden als Deskriptoren formuliert, die gemeinsam das `Verfahrensprofil` eines Operators bilden.[107] Ein Deskriptorenkatalog

[107] Verfahrenseigenschaften können manuell deklariert (z.B. aus der Algorithmusdokumentation entnommen) oder zum Teil automatisiert aus Prozessabläufen extrahiert

ist in Anhang A5.2 aufgeführt, und die Verwendung des Verfahrensprofils zeigt exemplarisch Abschnitt 5.5.3.4. Zum Beispiel lassen sich ausgewählte Verhaltenseigenschaften eines Clustering-Algorithmus mit den Deskriptoren `Autonomie: hoch`, `Sensitivität(Ausreißer): niedrig` und `Suchverhalten: inkrementell` beschreiben. Die vom Operator unterstützte `Ausrichtung` (vgl. Abschnitt 4.4.1) bildet aufgrund ihrer Bedeutung für die Verfahrensauswahl ein eigenständiges Attribut.

Zur Beurteilung durchgeführter Prozesse im Rahmen der Revision werden häufig Leistungsmaße berechnet, die auf verfahrensspezifischen Bewertungsfaktoren beruhen. Diese Größen können dem Operator als `Leistungsfaktoren` beigefügt werden. Ihre Darstellung erfolgt jeweils als Tupel `(Name, Wertgröße, Einheit, Bezugsgröße)`, um z.B. eine spezifische Verarbeitungskapazität von 20.000 Datensätzen je Sekunde auszudrücken `(Durchsatz, 20000, Datensätze, sec)`. Die Beurteilung von Analyseergebnissen erfordert häufig auf die Verfahrensklasse ausgerichtete Kriterien oder Ansätze. Zulässige `Bewertungskriterien` können im gleichnamigen Attribut hinterlegt werden. Die Eignung oder Empfehlung eines Evaluationsansatzes für einen Operator lässt sich durch Verknüpfung zu entsprechenden Funktionen der Prozessebene abbilden. Auf diese Weise können neben einzelnen Evaluationsverfahren (Operatoren) auch Module abgerufen werden, die komplexere Ansätze wie z.B. die mehrfache Kreuzvalidierung durch ein geeignetes Muster beschreiben. Neben der Verknüpfung zur Funktion können im zugehörigen Attribut `Evaluationsansatz` auch Anmerkungen und Empfehlungen zur Entscheidungsunterstützung hinterlegt werden.

Die Prozessgestaltung kann ferner durch `Zusicherungen` (Assertions) unterstützt werden, die bestimmte Eigenschaften der Ausgabedaten eines Operators beschreiben. So kann etwa der Output eines Normalisierungsoperators mit der Zusicherung `normalisiert: wahr` versehen werden. Ebenso ist z.B. nach einer linearen Skalierung bekannt, in welchem Intervall die Werte der betroffenen Variablen

werden (vgl. Abschnitt 7.4.2.2). Darüber hinaus ist auch denkbar, dass ein Werkzeug unbekannte Eigenschaften interaktiv vom Anwender erfragt.

liegen, was mithilfe der Deskriptoren `min` und `max` sowie entsprechenden Methodenparametern formulierbar ist. Diese Angaben können die Prozessverifikation vor der Ausführung vereinfachen oder, wie im zweiten Fall, zur Laufzeit die Durchführung eines berechnungsintensiven Verfahrens ersparen, wenn durch die Zusicherung zuvor bekannt ist, dass die Daten dessen Voraussetzungen nicht erfüllen [MoSE03, 18]. Zusicherungen können dazu in Datencharakteristika von Informationsobjekten überführt werden.

4.6.2.2 Software-Produkt (Service)

Jeder Operator repräsentiert eine konkrete Algorithmusimplementierung durch ein Software-Element und wird durch seine Zuordnung zu einer Funktion beschrieben. Die Menge aller angebotenen Funktionen bestimmt das nach außen verbindlich festgelegte Verhaltensrepertoire der Software und wird allgemein als *Service (Dienst)* bezeichnet. Ein Service ist demnach eine abstrakte Ressource, die eine spezifizierte Funktionalität erbringen kann. Das den Dienst erbringende Software-Element heißt Service-Provider oder kurz *Server*, das den Dienst in Anspruch nehmende Element (Service-) *Client* [LoDi04, 8, 11], [W3C04b, 37].

Gegenüber neueren Interpretationen, die einen Service als Funktionen erbringendes, „eigenständiges und über ein Netzwerk durch nachrichtenbasierte Kommunikation nutzbares Softwareelement“ begreifen, dessen exportierte Schnittstelle durch eindeutige Spezifikation beschrieben ist [Marx12], geht die hier vertretene, allgemeinere Sichtweise lediglich von autonom agierenden, in getrennten Prozessen ablaufenden Client- und Server-Komponenten aus [LoDi04, 14]. Sie deckt somit auch reine Desktop-Werkzeuge ab, wenngleich die meisten modernen Datenanalysesysteme zumindest optional auch als Web-basierte Anwendungsserver betrieben werden können und der neueren Interpretation genügen.

Der Service stellt ein Typmerkmal eines Software-Produkts dar, d.h., alle Installationen einer Software stellen dieselbe Funktionsmenge bereit. Entsprechend wird der *Service* als verhaltensorientierte Sicht auf ein *Software-Produkt* modelliert und im Folgenden synonym verwendet. Ein

Software-Produkt kann sowohl als eigenständiges Werkzeug oder Anwendungssystem (z.B. KDD-Suite, OLAP-Server, Datenbankverwaltungssystem) als auch in Form von Programmbibliotheken auftreten. Beispielsweise sind Analysesysteme wie RAPIDMINER und KNIME mittlerweile um Algorithmusbibliotheken erweiterbar. Zugleich werden viele der Bibliotheken auch in eigenen Werkzeugen angeboten (z.B. WEKA, R). Daneben existieren Programmsysteme, die ausschließlich zur Integration in Anwendungssysteme konzipiert und nicht eigenständig nutzbar sind (z.B. XELOPES) [MiRe11]. Daher wird ein **`Software-Produkt (Service)`** neben seinem `Namen`, einer `Version` und dem `Hersteller` optional auch durch ein `Werkzeug` beschrieben, in dessen Rahmen es zur Verfügung steht. So können z.B. zwei Services mit den Namen `KNIME` und `R (KNIME)` modelliert werden, von denen der zweite durch das `Werkzeug=KNIME` bereitgestellt wird, welches wiederum das gleichnamige Software-Produkt referenziert. Die Funktionen eines Services sind transitiv über die mit ihm verknüpften `Operatoren` ersichtlich. Die Dienstbereitstellung erfolgt über Software-`Installationen` (Aufgabenträgerinstanzen, vgl. Abschnitt 4.6.3).

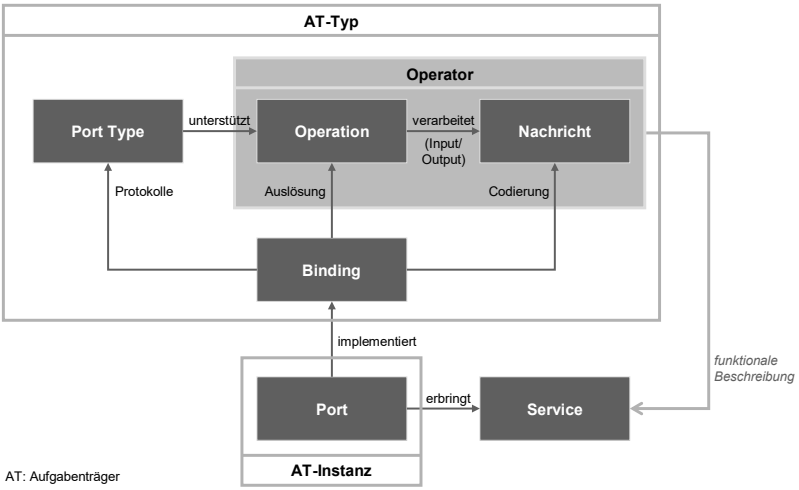


Abbildung 54: Korrespondenz der Repräsentation maschineller Aufgabenträger mit WSDL-Konzepten nach [KoLe04, 121] (eigene Darstellung, angelehnt an die Quelle)

Die dargestellte Sichtweise korrespondiert mit den Beschreibungselementen der Web Service Description Language (WSDL) (Abbildung 54), wonach ein *Port* (Software-Installation) einen *Service* bereitstellt, der durch ein *Binding* implementiert ist. Das Binding definiert für einen Port Type (Software-Produkt/Aufgabenträgertyp) Protokolle und Codierung von *Nachrichten* zum Aufruf von *Operationen* sowie zum Austausch ein- und ausgehender Daten [KoLe04, 121]. Die Protokolle legen die Regeln fest, nach denen Client und Server zur Inanspruchnahme des Dienstes kommunizieren [LoDi04, 14].

4.6.2.3 Rolle

Neben maschinellen Aufgabenträgern sind typischerweise auch Personen an der Durchführung eines Datenanalyseprozesses beteiligt. Sie können auf Typebene durch **`Rollen`** Berücksichtigung finden. Dies ist z.B. in Situationen sinnvoll, in denen der Analytiker ein bestimmtes `Qualifikationsprofil` erfüllen sollte. Dieses kann durch frei definierbare Deskriptoren sowie durch die Verknüpfung mit mehreren `Software-Produkten`, die der Analytiker beherrscht, erstellt werden. Weiterhin ist zur Unterstützung der finanziellen Projektplanung ein `Kostensatz` hinterlegbar, der für Träger dieser Rolle zu erwarten ist.

4.6.2.4 Metamodell

Abbildung 55 zeigt das Metamodell zur Aufgabenträgersicht einschließlich wichtiger Attribute. Ein **`Operator`** repräsentiert die strukturelle Beschreibung eines Verfahrens. Seine funktionale Beschreibung erfolgt durch Zuordnung zu genau einer (1,1) **`Funktion`** der Prozessebene, die wegen ihrer Bedeutung hier (mit gebrochenen Kanten) eingezeichnet ist. Eine Funktion kann durch beliebig viele (0,*) Operatoren realisiert werden. Ein Operator wird von genau einem (1,1) **`Software-Produkt (Service)`** angeboten, das nur mit Bereitstellung von mindestens einem (1,*) Operator in Erscheinung tritt. Personelle Aufgabenträgertypen werden durch das Konzept der **`Rolle`** abgebildet, die zur Beschreibung werkzeugbezogener Anforderungen mit beliebig vielen (0,*) Software-Produkten verbunden sein kann. Umgekehrt darf ein Software-Produkt in beliebig vielen (0,*) Rollen auftreten.

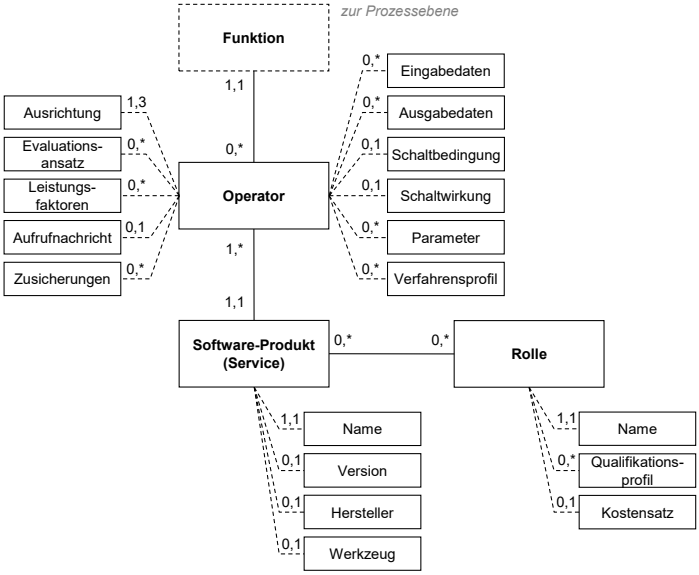


Abbildung 55: Metamodell zur Aufgabenträgersicht der Ressourcenebene (eigene Darstellung)

4.6.3 Instanzensicht

Ausprägungen der Aufgabenträgertypen und Informationsobjekttypen zeigt die Instanzensicht. Sie beschreibt verarbeitete bzw. produzierte Daten sowie eingesetzte bzw. verfügbare maschinelle und personelle Aufgabenträger, wie sie vom Instanzmodell der Prozessebene referenziert werden.

4.6.3.1 Informationsobjekt

Aufgrund seiner Abstammung vom Datenobjekttyp stellt ein **`Informationsobjekt`** stets ein *Datenobjekt* dar, das durch einen `Name`n, seinen aktuellen `Wert` und die vom `Informationsobjekttyp` vorgegebene Struktur definiert ist [FeSi13, 323]. Zulässige Werte sind Subtypen oder Teilmengen der dort spezifizierten Klassen von Objekten bzw. Literalen [ZLKO97, 293]. Die Grundstruktur (`Name`, `Informa-`

`tionsobjekttyp`, `Wert`) ist bewusst in Anlehnung an die Deklaration von Datenobjekttypen gewählt (vgl. Abschnitt 4.6.1 und Abbildung 56).

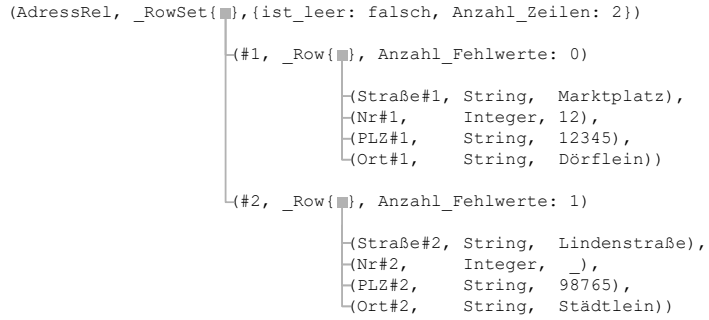


Abbildung 56: Beispiel zu Struktur und Wert eines Informationsobjekts (eigene Darstellung)

Bei elementaren Datenobjekten enthält der Wert unmittelbar die jeweilige Ausprägung, wie etwa „Marktplatz“ für `Straße` oder „12“ für die Haus-`Nr`. der ersten Tabellenzeile in Abbildung 56. Bei komplexen Objekten können stattdessen Deskriptoren angegeben werden, die z.B. *Datencharakteristika* enthalten, welche die Gestaltung oder Steuerung des Prozesses beeinflussen können. So wird für die zweite Tabellenzeile (`_Row` #2) ein fehlender Wert protokolliert, und für die Relation (`_RowSet`) `AdressRel` u.a. die Anzahl der Zeilen angegeben. Das Beispiel illustriert zugleich die aktuelle Struktur des komplexen Objekts `Adresse`. Sie ergibt sich aus den in der Ontologie von Abbildung 51 dargestellten Beziehungen.

Datencharakteristika werden von mehreren Autoren als effektive Hilfsmittel für das Management von Datenanalyseprozessen angesehen. Nähere Ausführungen zu ihrer Verwendung finden sich in Abschnitt 5.5.3.2. Für ihre Ermittlung existieren mehrere Optionen. So berechnen einige Operatoren bzw. Analysewerkzeuge standardmäßig Beschreibungen der verarbeiteten Daten, um sie dem Analytiker zur Information oder zur Entscheidungsunterstützung zu präsentieren. Ist dies nicht der Fall, können explizit spezielle Aktivitäten zur Datenexploration in den Prozess integriert werden, um solche Kenngrößen automatisch zu ermitteln. Auch Zusicherungen der Operatoren (vgl. Abschnitt 4.6.2.1)

können Bestandteil des `Wert`-Attributs von Informationsobjekten sein. Ihre Behandlung analog der Restriktionen von Datenobjekttypen ermöglicht die Verarbeitung durch automatische Planungs- und Workflow-Management-Systeme.

Über die Basisattribute des Datenobjekts hinaus speichert jedes Informationsobjekt ein `Ereignisprotokoll`, in dem die zeitliche Abfolge aller Zustandsänderungen festgehalten wird. Es erfasst die Erzeugung sowie alle Transformationen des Objekts mit Zeitstempel, Operator und Parameterwerten, um eine Abstammungs- und Wirkungsanalyse zu ermöglichen [ABEM15, 265f.]. Die Kenntnis der Herkunft und des Lebenszyklus von Datenobjekten (Data Lineage) wird zur Interpretation und Einschätzung der Glaubwürdigkeit von Analyseergebnissen als zunehmend wichtig erachtet. Die Ereignisse ermöglichen darüber hinaus die Feststellung des Zeitpunkts einer Datenanalyse, d.h. der Aktualität ihrer Ergebnisse [ZLKO97, 292].

4.6.3.2 Datenquelle

Eine Menge inhaltlich zusammengehöriger `Informationsobjekte` bildet eine *Datenquelle*. Sie repräsentiert eine physische oder logische Ressource, wie z.B. eine Datenbank, Organisationseinheit, Institution oder Datenerhebung. Die **`Datenquelle`** ist existenzabhängig von den sie konstituierenden Informationsobjekten und daher ein Element der Instanzebene. So ist etwa eine Datenbank nur solange als Datenquelle nutzbar, wie sie auch Daten enthält, und eine Datenerhebung (Vorgang) kann erst dann als Quelle dienen, wenn ihre Ergebnisse vorliegen. Der `Name` der Datenquelle sollte daher stets auf geeignete Instanzmerkmale eingehen, wie z.B. „Kundenzufriedenheitsumfrage 2016“, „Auswertung Vertriebsbüro Süd, Mai 2015 (Herr Müller)“, etc. Ihre Struktur ergibt sich aus den Typen der enthaltenen Informationsobjekte. Sie wird idealerweise durch ein Datenschema dokumentiert, das z.B. als `Link` mit der Quelle verknüpft ist. Der `Ressourcentyp` gibt Aufschluss über die Art des zu erwartenden Inhalts (z.B. Datenbank, Dokument, Auskunft), der `Datenquellentyp` über die Erhebungsform (Primär-/Sekundärdaten). Die Zuordnung zu einer Organisationseinheit oder Institution erfolgt über `Ansprechpartner`, bei denen entsprechende

Stammdaten hinterlegt sind. Weitere Details können in einer ausführlichen `Beschreibung` dokumentiert werden.

Zur Auswahl von Datenquellen zu einem Analyseproblem wird die Perspektive verwendet, die eine Reihe von Domänenobjekten beschreibt, bei denen die Daten erhoben werden (vgl. Abschnitt 4.4.2.1). Entsprechend ist auch der Datenquelle ein als `Erhebungsobjekt` bezeichnetes Domänenobjekt zugeordnet, das die von der Quelle eingenommene Perspektive repräsentiert. Die Inhalte der Datenquelle werden anhand der Domänenobjektzuordnung der Informationsobjekte hinreichend beschrieben. Weitere formale Eigenschaften sind im `Datenquellenprofil` festgehalten und können mit dem Informationsbedarfsprofil des Analyseproblems abgeglichen werden. In Anlehnung an die dortige Einteilung werden die Eigenschaftsklassen Art, Qualität, Verfügbarkeit und Kosten unterschieden. Eine vollständige Korrespondenz ist jedoch weder in Bezug auf die Klassen noch auf die Einzelkriterien festzustellen, da für Daten andere Eigenschaften gelten als für Informationen. Anhang A5.3 beschreibt einen Katalog möglicher Deskriptoren, und Abschnitt 5.4.5.2 erläutert das Vorgehen bei der Auswahl von Datenquellen. Als Beispiel für das Datenquellenprofil einer operativen Datenbank aus der Perspektive des Verkaufs mögen die Deskriptoren `Aussageform: faktisch`, `Medientyp: Relationen & Tabellen`, `Aggregationsgrad: Artikel` und `Verfügbarkeit: sofort` dienen.

4.6.3.3 Software-Installation (Server)

In ihrer Eigenschaft als abstrakte Ressourcen bedürfen Services zu ihrer Nutzung einer Realisierung als konkrete Ressourcen, die durch einen *Server* erfolgt [W3C04b, 37]. Der Server entsteht mit Installation und Inbetriebnahme von Software-Produkten und stellt eine Instanz eines maschinellen Aufgabenträgers dar. Sein Typ ergibt sich aus der Beziehung zu den installierten `Software-Produkten`. Aufgrund der Möglichkeit zur Einbindung von Bibliotheken kann eine Installation

mehrere Services realisieren.[108] Die **Software-Installation (Server)** erhält einen Namen zur Identifikation und kann durch ihren physischen Standort beschrieben werden. Ist der Server im Netzwerk zugänglich, gibt der URL die Adresse an, unter der er zum Aufruf von Operatoren erreichbar ist. Zusätzlich oder alternativ kann eine Aufrufnachricht angegeben werden, die zur Kommunikation mit der Installation dient. Sie kann in Abhängigkeit von der jeweiligen Systemumgebung z.B. einen fernen Prozeduraufruf, eine HTTP-Anforderung oder eine SOAP-Nachricht darstellen Die konkrete Form und Realisierung sind implementierungsabhängig.

Stehen mehrere gleichartige Server zur Verfügung, kann die Auswahl anhand nicht-funktionaler Kriterien geschehen, die sich auf Qualität, Kosten oder Leistungsfähigkeit der Installation richten. Mögliche Qualitätskriterien sind z.B. Verfügbarkeit, Robustheit gegenüber Störungen und Fehlern, Effizienz, Skalierbarkeit, Sicherheit [LoDi04, 11-13]. Allgemeine Qualitätseigenschaften werden als Dienstmerkmale in Form von Deskriptoren beschrieben, Kostensatz und quantitative Leistungsfaktoren jeweils in Form von Bewertungsfaktoren abgebildet. Beispielsweise werden Kosten in Höhe von 6 EUR je Stunde anhand der Merkmale (Name, Wertgröße, Einheit, Bezugsgröße) durch das Tupel (Betriebskosten, 60, EUR, h) ausgedrückt. Aufgrund der implementierungsunabhängigen Spezifikation von Services können konkrete Installationen ausgetauscht und gegebenenfalls erst zur Laufzeit bestimmt werden [KoLe04, 118], [ElLa04, 104]. Die Nutzung der Dienste ist demnach nicht an einzelne Installationen gebunden. Die zunehmende Verfügbarkeit Cloud-basierter Datenhaltungs- und Analysedienste [BaEc17] erleichtert die kosten- und leistungsorientierte Auswahl sowie die bedarfsgerechte Inanspruchnahme entsprechender Angebote. Zur Abbildung der organisatorischen Verantwortlichkeit für interne und externe Installationen stehen die

[108] Hierbei tritt typischerweise der Fall auf, dass gleiche oder ähnliche Operatoren von mehreren Services angeboten werden und somit innerhalb eines Werkzeugs mehrfach bereitstehen. Wird keine universelle Funktionstaxonomie zur Beschreibung der Operatoren genutzt, wird die Gleichartigkeit der Operatoren nicht direkt sichtbar; die Operatoren hängen an verschiedenen Zweigen des Operatorenpools der Werkzeuge.

Attribute `Zugehörigkeit`, `Organisation`, `Unternehmen` und `Ansprechpartner` zur Verfügung.

4.6.3.4 Person

Personen, die zur Übernahme bestimmter Aufgaben im Rahmen eines Datenanalyseprojekts geeignet sind, können mit einer Reihe von Stammdaten erfasst werden. Neben `Name`, `Adresse` und `Telefon` der **`Person`** interessieren insbesondere die `Zugehörigkeit` (interner/externer Mitarbeiter), der `Kostensatz` bei Beauftragung sowie das `Qualifikationsprofil`. Dieses kann spezifisch hinterlegt oder über eine optionale Rollenzuordnung bestimmt werden. Zusätzlich können Software-Installationen mit der Person verknüpft werden, um sie als verantwortlichen Ansprechpartner für deren Betreuung und Administration zu kennzeichnen.

4.6.3.5 Metamodell

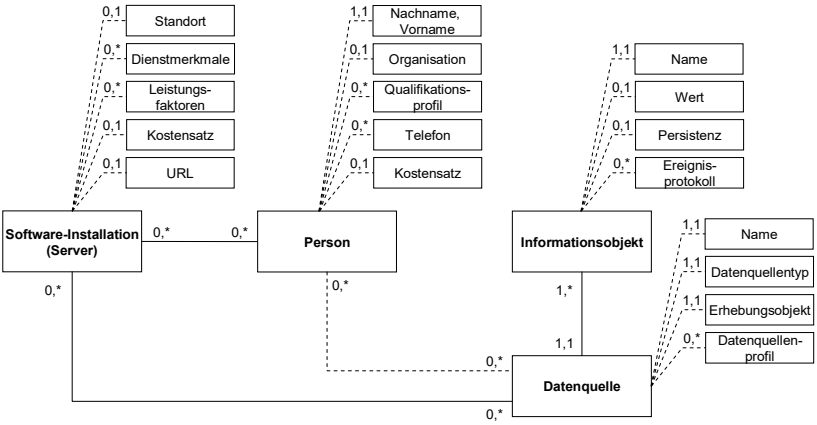


Abbildung 57: Metamodell zur Instanzensicht der Ressourcenebene (eigene Darstellung)

Die Abbildung 57 zeigt das Metamodell der Instanzensicht im Überblick. Mit der Datensicht korrespondierende Instanzen sind das **`Informationsobjekt`**, das jeweils genau einer (1,1) Datenquelle angehört. Eine **`Datenquelle`** entsteht durch Zuordnung einer nicht-

leeren Menge (1,*) von Informationsobjekten. Die weiteren Elemente korrespondieren mit der Aufgabenträgersicht und sind nur durch optionale Beziehungen untereinander und mit den datenorientierten Instanzen verbunden. Eine **`Software-Installation (Server)`** kann (z.B. als Datenserver oder Analysewerkzeug) beliebig viele (0,*) Datenquellen realisieren. Eine solche Datenquelle kann von mehreren (0,*) Servern betrieben werden (replizierte oder verteilte Datenbanken). Für eine Software-Installation oder eine Datenquelle können jeweils beliebig viele **`Personen`** als Ansprechpartner registriert sein. Umgekehrt kann eine Person für keine oder mehrere Server oder Datenbanken verantwortlich zeichnen.

4.6.4 Zusammenfassung zur Ressourcenebene

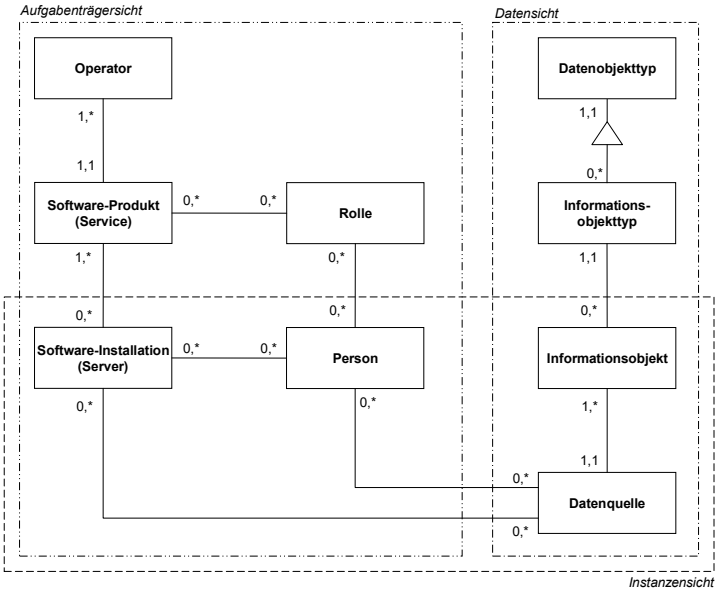


Abbildung 58: Integriertes Metamodell der Ressourcenebene (eigene Darstellung)

Das Zusammenspiel der drei diskutierten Sichten der Ressourcenebene verdeutlicht das integrierte Metamodell in Abbildung 58. Aufgaben-

träger- und Datensicht erstrecken sich hierbei auch auf die zugehörigen Elemente der Instanzensicht. Die Ressourcenebene der Datenanalysearchitektur repräsentiert ein verteiltes System aus maschinellen und personellen Aufgabenträgern, die zur Realisierung von Analyse-Workflows verfügbar sind, sowie die verarbeiteten bzw. vorhandenen Datenobjekte. Die Instanzensicht verzichtet auf die Darstellung der Beziehungen zwischen Software-Installationen bzw. Personen einerseits und den durch diese verarbeiteten Informationsobjekte andererseits. Sie werden auf dieser Ebene als nicht relevant erachtet und sind bei Bedarf unter Einbeziehung der Prozessebene (Instanzensicht) rekonstruierbar. Stattdessen liegt der Fokus auf den Verantwortlichkeiten der Aufgabenträger für Datenquellen.

4.7 Spezielle Sichten auf Datenanalyseprozesse

Die detaillierte Spezifikation der Modellierungsobjekte auf den vier Ebenen der Datenanalysearchitektur lässt sich durch weitere Beschreibungselemente unterstützen. Hierzu erläutert Abschnitt 4.7.1 den Einsatz von Ontologien zur Gewährleistung einer einheitlichen Semantik von Attributwerten. Abschnitt 4.7.2 diskutiert verschiedene Kontexte und Kontextfaktoren, und Abschnitt 4.7.3 erörtert Repräsentationsformen für Restriktionen und Regeln.

4.7.1 Ontologien

Formale Modelle einer Anwendungsdomäne, die den Austausch und die gemeinsame Nutzung von Wissen erleichtern und die Kommunikation zwischen personellen und maschinellen Aufgabenträgern verbessern sollen, werden als ***Ontologien*** bezeichnet [MäSS01, 393]. Eine Ontologie stellt ein abstraktes Modell dar, welches die relevanten Begriffe (Konzepte) eines Phänomens der realen Welt identifiziert. Dabei sind die Begriffstypen und die für ihre Nutzung geltenden Restriktionen explizit definiert und in einer maschinenlesbaren Repräsentation formalisiert.

Das repräsentierte Wissen trifft auf den Konsens einer Gruppe, die es einvernehmlich als gültig akzeptiert [StBF98, 184].[109]

4.7.1.1 Vorgabe und Strukturierung von Vokabularen

Von einer Anwendergruppe akzeptierte Begriffssysteme eignen sich zunächst zur Bereitstellung eines vordefinierten *Vokabulars* für bestimmte Attribute, um die Eingabe von Freitext oder subjektiv gewählten, nicht klar bestimmten Ausprägungen zu vermeiden. Einheitliche Werbebereiche für kategorische Attribute erleichtern die gemeinsame Verwendung von Modellierungsartefakten, indem sie die Vielfalt möglicher Begriffe wirksam einschränken. Vokabulare eignen sich für eine Reihe von Attributen der Metaobjekttypen, die z.B. Typen (Betriebstypen der Maßnahme, Ressourcen- oder Quellentypen der Datenquelle) oder Kategorien darstellen (Branchen der Maßnahme, Ausrichtungen des Analyseziels). Existiert eine Hierarchie der betroffenen Begriffe (z.B. bei den Deskriptoren zur Bildung von Informationsbedarfs-, Verfahrens- oder Datenquellenprofilen), sind ***Taxonomien*** zweckmäßig, die eine hierarchische Ordnung von Begriffen oder Kategorien bilden. Sie werden oft als einfache Form von Ontologien angesehen, die intuitiv und leicht verständlich ist. Begriffe oder Objekte lassen sich durch Navigation in der baumartigen Struktur leicht auffinden [PrKK05, 1312f.].

4.7.1.2 Semantische Annotation von Modellierungsartefakten

Von zentraler Bedeutung für den präsentierten Ansatz ist die Annotation von Prozessbausteinen durch in einer Taxonomie angeordnete Funktionen. Die Ordnung repräsentiert Spezialisierungsbeziehungen (*ist_ein*-Relationen) zwischen den Funktionen. Die ihnen zugeordneten Prozessbausteine und Operatoren stellen Optionen zur Realisierung der Funktionen dar, können mit diesen also durch semantische *realisiert*-Relationen verknüpft werden. Auf diese Weise entsteht eine Ontologie von Prozessbausteinen, die beliebige semantische Beziehungen zwi-

[109] Eine verbreitete Definition beruht auf den Vorschlägen von GRUBER [Grub93] und BORST [Bors97] und versteht unter Ontologie „a formal, explicit specification of a shared conceptualization“ [StBF98, 184].

schen Begriffen (z.B. Zuordnung von Eigenschaften und Restriktionen) abbilden kann. Insbesondere ist auch die Modellierung von Relationen zwischen Klassen und Instanzen möglich. Einem Begriff annotierte Eigenschaften können sodann auf Grundlage von Inferenzregeln auch auf transitiv verbundene Begriffe und Instanzen angewendet werden [PrKK05, 1314]. Diese Möglichkeit erweist sich z.B. für Daten- und Informationsobjekte als hilfreich. Abbildung 51 (Seite 199) zeigt den Ausschnitt aus einer Datenobjektontologie, die über Konstruktoren realisierte, komplexe Datenstrukturen (*Teil_von*-Relationen) sowie *Instanz_von*-Relationen enthält. So ist z.B. ersichtlich, dass eine `_Table` aus `_Columns` besteht und auf Instanzebene als `_RowSet` (Relation) auftritt, das aus `_Rows`, diese wiederum aus `_ColumnValues` bestehen.

Ontologien sind ebenso geeignet, die in einer Domäne, Branche, einem Unternehmen oder Projekt gebräuchliche *Fachterminologie* strukturiert zu dokumentieren und hierbei auch Synonyme und Antonyme durch entsprechende semantische Beziehungen zu verdeutlichen. Derartige Ontologien können einen leicht zugänglichen Katalog von *Domänenobjekten und Domänenobjektmerkmalen* bilden, die z.B. als Beschreibungselemente von Problemdomäne und -inhalt des Problemaspekts, von Untersuchungsobjekt und -ziel des Analyseziels oder von Informationsobjekten Verwendung finden. Ein Objektmodell, das konzeptuelle Domänenobjekte und deren Merkmale sowie mögliche Spezialisierungen (z.B. Kundenauftrag *ist_ein* Auftrag) definiert, unterstützt wirksam die einheitliche Auswahl von Such- und Indextermen für Modellierungsartefakte. Als Grundlage für die Ontologie können bestehende Objekt- oder Klassenmodelle dienen, die gegebenenfalls mit oben genannter Fachterminologie integriert werden. So lässt sich z.B. folgender Sachverhalt auf Basis konzeptueller Domänenobjekte definieren: `Großkunde` *ist_ein* `Kunde`, welcher *hat_Eigenschaft* `Auftragsvolumen > 1000.00 EUR`.

Weitere Einsatzszenarien für Ontologien im Kontext der vorliegenden Arbeit sind etwa Kennzahlensysteme (z.B. DuPont-Schema) zur Bestimmung des Wertbeitrags von Problemaspekten oder Organigramme zur Strukturierung von Organisationseinheiten, Rollen und Personen (vgl. Maßnahmen sowie Ressourcen).

4.7.1.3 Semantisches Prozessmanagement

Abschnitt 4.5.1.2 verweist bereits auf die Potenziale ontologisch annotierter Prozessbausteine für Prozessgestaltung und -wiederverwendung. Die dort vorgestellte Annotation auf Basis der Ansatzes von THOMAS & FELLMANN [ThFe09] ist zur Realisierung der semantischen Prozessplanung nach HEINRICH ET AL. [Hein+08] geeignet und unterstützt die Schemavalidierung sowie die Suche in Prozessbibliotheken [ThFe09, 508]. Neben Prozessbausteinen werden insbesondere auch Informationsobjekte semantisch angereichert, um die Untersuchung der zwischen ihnen definierten Relationen durch Inferenzmechanismen zu erlauben [Hein+08, 448]. Mittels maschineller Inferenz können neue Fakten generiert werden, die nicht explizit im Schema enthalten sind. So kann die Suche nach einem Baustein „Daten bereinigen" z.B. ein Element „Fehlende Werte anreichern" liefern, wenn dieses in der Ontologie als Spezialisierung des ersten hinterlegt ist (vgl. [ThFe09, 509]). Ein Planer ist damit in der Lage, nicht nur unmittelbar passende Bausteine oder Informationsobjekte zur Prozesskonstruktion zu verwenden, sondern kann z.B. auch Subklassen der benötigten Elemente erkennen und einsetzen, wodurch die Flexibilität der Prozessplanung steigt [Hein+08, 446].

Das semantische Prozessmanagement zielt allgemein auf die Reduzierung des Aufwands der Erstellung und Pflege von Prozessschemata. Durch die begriffliche Einordnung sinkt auch bei manueller Prozesskonstruktion der Überprüfungs- und Abstimmungsbedarf. Darüber hinaus sollen Schemata verständlicher sowie leichter zwischen verschiedenen Systemen austauschbar werden [Hein+08, 445-447].

4.7.1.4 Repräsentation

Eine vollständige Ontologie besteht aus einer Menge von Begriffen C, einer Halbordnung auf C (Begriffshierarchie oder -taxonomie), einer Menge von Relationen R, einer Halbordnung auf R (Relations- oder Beziehungshierarchie), einer Abbildung $R \rightarrow C \times C$ (Signatur) sowie einer Menge von Inferenzregeln IR in einer logischen Sprache L [HMSS01]. Das Metamodell in Abbildung 59 zeigt Bausteine zur semi-

formalen Repräsentation von **`Begriffen`**, **`Relationen`** und Begriffstaxonomien. Letztere werden durch Relationen vom Typ **`Typbeziehung`** (*ist_ein*) realisiert. Als häufig verwendete Relationstypen sind **`Aggregation`** (*Teil_von*), **`Instanziierung`** (*Instanz_von*) und **`Realisierung`** (*realisiert*) vordefiniert; weitere Relationen können als **`Interaktion`** modelliert und ihre Semantik über das Attribut `Name` ausgedrückt werden. Relationshierarchien werden im Kontext der vorliegenden Arbeit als verzichtbar erachtet, und Inferenzmechanismen werden nicht weiter betrachtet. Die Signatur bildet das Ergebnis der Modellierung.

Zur formalen Repräsentation bietet sich die standardisierte und weit verbreitete Ontologiesprache OWL [W3C09] an, für deren Erweiterung OWL-DL leistungsfähige Inferenzmaschinen existieren [ThFe09, 509]. Ein Begriff besitzt mehrere `Eigenschaften` [W3C04c], [PrKK05, 1311]. Zusätzlich können `Anmerkungen`, `Restriktionen` und für Relationen auch `Kardinalitäten` definiert werden.

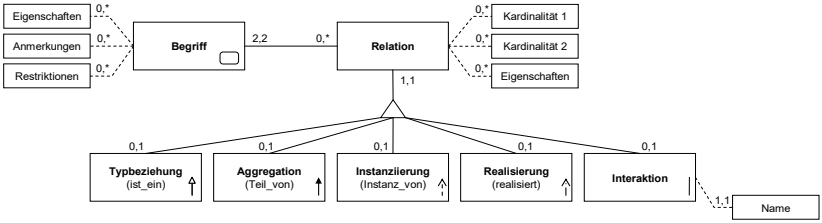


Abbildung 59: Metamodell für Ontologien (eigene Darstellung)

Dem obigem Metamodell folgende Ontologien lassen sich in OWL transformieren. Beispielsweise wird ein Datenobjekttyp als Begriff (Klasse, `owl:Class`) abgebildet und der zugehörige Wertebereich (Eigenschaft) im Falle eines primitiven Datenobjekttyps in eine `owl:DataProperty`, im Falle einer komplexen Struktur in eine mittels Relation verbundene Klasse übersetzt [Hein+08, 451]. Die formale Repräsentation und Transformationsregeln werden nicht weiter betrachtet. Einige Hinweise zur Übersetzung gibt das Attributschema in Anhang A4.5.

Die navigierende Suche und Selektion geeigneter Begriffe wird durch Visualisierung von Ontologien in Form intuitiver Diagramme unter-

stützt. Werden z.B. Domänenobjekte in einem Interaktionsschema gemäß SOM dargestellt, lässt sich ausgehend von einem Orientierungspunkt (etwa dem Umweltobjekt Kunde) leicht jene Transaktion finden, die das Domänenobjekt Kundenauftrag repräsentiert, das durch Markierung im Diagramm ausgewählt und etwa als Suchkriterium für die Prozessbibliothek oder als Indexterm zur Annotation eines Artefakts übernommen werden kann (vgl. das Beispiel zur Wahl der Perspektive in Abschnitt 4.4.2.1). Die Verwendung beliebiger Diagrammtypen ist durch Verknüpfung des zugehörigen Metamodells mit jenem des Ontologiemodells möglich, wie Abbildung 60 zeigt. Intuitiven Zugang zu Begriffssystemen bieten auch Taxonomien, die durch Projektion aus Ontologien ableitbar sind.

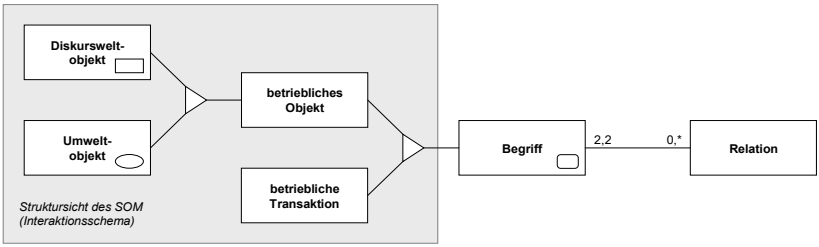


Abbildung 60: Verknüpfung des ontologischen Metamodells mit der Struktursicht des SOM (eigene Darstellung)

Gleichwohl ist das Potenzial der automatischen Suche und der deduktiven Verarbeitung des repräsentierten Wissens nur bei entsprechender Formalisierung nutzbar. Die Notwendigkeit zur manuellen Eingabe der Metadaten stößt jedoch regelmäßig auf sehr geringe Nutzerakzeptanz. Solche Mehrarbeit wird allenfalls dann akzeptiert, wenn sie durch die erreichbaren Vorteile aufgewogen wird [FHTN11, 33]. Da dieser Nutzen sich dem Anwender typischerweise nicht unmittelbar offenbart und die vollautomatische Anreicherung nicht die erforderliche Qualität liefert, ist eine semiautomatische Annotation anzustreben, die dem Benutzer automatisch generierte Metadaten vorschlägt, die er bestätigen oder korrigieren kann [PrKK05, 1310].

4.7.1.5 Ontologien zur Unterstützung der Datenanalyse

Ontologien wurden wiederholt zur Unterstützung des Managements von Datenanalyseprozessen vorgeschlagen.[110] Taxonomien und einfache Ontologien zur Beschreibung von Aufgaben und Operatoren nutzen z.B. die Systeme CAMLET [SuYa98] und IDEA [BePH05] als Basis für automatische Prozessplaner. Im MININGMART-Projekt [MoSc04] wird eine einfache Ontologie zur Indizierung von Vorverarbeitungsprozessen verwendet, um deren Abruf aus einer Fallbasis zur Wiederverwendung zu vereinfachen. Mit Entstehung von Sprachen des Semantic Web wurden fortgeschrittene Ontologien vorgeschlagen. Die Ontologie DAMON [CaCo03] unterstützt die semantische Suche nach Ressourcen für verteilte KDD-Prozesse im Grid, und der GRIDMINER ASSISTANT [BrJT08] ermöglicht die interaktive Prozesskonstruktion, indem automatisch erzeugte, abstrakte Aufgabensequenzen im zweiten Schritt auf Basis von ontologisch definierten Vor- und Nachbedingungen oder von Nutzervorgaben konkretisiert werden. KDDONTO [DiPS09] versorgt einen automatischen Planer mit Operatorbeschreibungen, der auch Spezialisierungs- und Aggregationsrelationen aus der Ontologie sowie Leistungs- und Bewertungsparameter berücksichtigt. Der Prozessplaner der KD ONTOLOGY [ZKZL10] erzeugt abstrakte Prozesse, deren Aufgaben erst nach Instanziierung ausführbar sind. Der bislang umfassendste Ansatz aus dem E-LICO-Projekt erzeugt hierarchische Aufgabennetze durch interaktive, sukzessive Zerlegung einer Initialaufgabe auf Basis der Ontologie DMWF, die über 600 Operatoren der Werkzeuge RAPIDMINER und WEKA aus Außensicht beschreibt [KSBF09], [KSBF10]. Eine zweite Optimierungsontologie DMOP modelliert Implementierungsdetails der Algorithmen und bringt die interaktiv erarbeiteten Prozessvorschläge in eine Rangordnung. Zur weiteren Verbesserung der Vorschläge soll eine Wissensbasis beitragen, die mit Erfahrungen aus früheren Analysefällen gespeist wird und zunächst die Algorithmus- und Modellselektion innerhalb der Analysephase unterstützt [Hila+11].

Nicht unmittelbar der Unterstützung der Prozesskonstruktion, sondern vielmehr der Bereitstellung allgemeingültigen Wissens dient ONTODM

[110] Vgl. hierzu auch die umfassende Aufstellung bei [Hila+11, 277-280].

[PaDS08], [PaSD14]. Sie definiert allgemeine Begriffe der Datenanalyse wie z.B. Aufgabe, Algorithmus, Datenset und Datentyp. Ein ähnliches Vorhaben verfolgt EXPOSÉ [VGBS04], die zum Teil auf ONTODM aufbaut und das Fundament zu einer Experiment Markup Language legen soll. Das Ziel, ein allgemeines Vokabular für Data Mining bereitzustellen, verfolgt indes auch das KNOWLEDGE DISCOVERY METAMODEL der OMG [OMG15], das sich jedoch auf die Entwicklung und Integration von Werkzeugen richtet. Ein standardisiertes Begriffsmodell stellt auch das COMMON WAREHOUSE METAMODEL (CWM) dar, das verschiedene Datenformate und Analyseansätze abdeckt [PCTM02].

Im vorliegenden Kapitel werden Ontologien in erster Linie als Hilfsmittel zur Auswahl von Attributwerten und zur semiformalen Dokumentation von Domänenwissen gesehen. In späteren Kapiteln werden weitere Anwendungsformen im jeweiligen Kontext genannt.

4.7.2 Kontext

Modellierungsobjekte sind nicht ausschließlich durch ihre Repräsentation mittels Metaobjekttyp und Attributmenge bestimmt, sondern häufig von weiteren Faktoren abhängig, die z.B. auf anderen Modellebenen beschrieben oder durch die Anwendungssituation definiert sind. So lässt sich die Geeignetheit einer Gestaltungsoption für die aktuelle Problemstellung etwa besser beurteilen, wenn das verfolgte Analyseziel und die dahinter stehende Anwendung (z.B. Ursachenanalyse eines sinkenden Bonbetrags) bekannt sind. Ebenso können Gestaltungsentscheidungen wie etwa in den Prozess aufzunehmende Aufgaben, die Auswahl von Operatoren oder die Instanziierung von Verfahrensparametern von Faktoren wie z.B. den zu verarbeitenden Daten abhängen. Solche Faktoren werden als *Kontextfaktoren* bezeichnet.

Mit dem Begriff *Kontext* werden im Allgemeinen evolvierende, zweckgerichtet strukturierte und gemeinsam genutzte Informationsräume umschrieben [CCDG05, 49]. Die verbreitete Definition nach DEY & ABOWD sieht Kontext als Menge aller Einflussfaktoren, welche die

Interaktion zwischen Nutzer und Anwendung beeinflussen [DeAb00].[111] Darauf aufbauend setzt ISSELHORST Kontextfaktoren (Kontext bildende Einflussfaktoren) in Bezug zu einer Aufgabendurchführung [Isse07, 111].[112] Für die Zwecke des Managements von Datenanalyseprozessen wird ***Kontext*** verstanden als die Summe aller Einflussfaktoren, die auf die Prozessgestaltung und -lenkung einwirken und nicht Bestandteil der Repräsentation des betroffenen Modellierungsobjekts (z.B. Schema, Aktivität, etc.) sind. Er stellt die Rahmenbedingungen dar, innerhalb derer Entscheidungen des Prozessmanagements zu treffen sind (vgl. [RuPR99, 226], [CCDG05, 50]).

Kontextfaktoren sind konkrete Ausprägungen ausgewählter Merkmale bestimmbarer Entitäten [RuPR99, 230], [Isse07, 141f]. Welche Attribute Kontextfaktoren bilden, ist situationsspezifisch. Grundsätzlich kommen alle definierten, abgeleiteten oder sensorisch ermittelten Merkmale ebenso in Frage wie Beziehungen zu anderen Entitäten [CCDG05, 51]. Ein Kontext aggregiert mehrere logisch zusammengehörige Entitäten[113] und dient als Filter, der eine Menge potenziell relevanter Einflussgrößen selektiert [KZTL08, 466]. Aus dieser Palette kann jeweils der situativ wirksame Satz an Kontextfaktoren zusammengestellt werden.

Für jede Ebene der Datenanalysearchitektur wird mindestens ein Kontext definiert. Die resultierenden Kontexte sind in Tabelle 1 jeweils durch die konstituierenden Metaobjekttypen (vgl. Entitäten als Kontextträger) charakterisiert. Die Kontexte stellen spezielle Sichten (Projektionen) auf die Metamodelle der jeweiligen Ebene dar und dienen der einfacheren Referenz. So kann beispielsweise anstelle der Notation

---

[111] Die Definition entstammt dem Bereich der kontextbewussten Anwendungen [DeAb00] (zitiert bei [HuZi11, 146]).

[112] Stärker vorgangsorientierte Sichtweisen stützen diese Interpretation: So umfasst Kontext nach LIEBERMAN & SELKER [LiSe00] alle Faktoren außer Eingabe und Ausgabe, die Einfluss auf die Lösungsberechnung nehmen [HuZi11, 146]. Ähnlich verstehen WAGNER & FERSTL darunter sekundäre Einflussfaktoren, die auf das Verhalten eines Prozesses einwirken, jedoch nicht primär seine Reaktionen auslösen [WaFe10, 117f.].

[113] Kontexte werden häufig zu Dimensionen zusammengefasst. Gängige Dimensionen sind z.B. Aufgabenkontext, Benutzerkontext, Interaktionskontext, organisatorischer Kontext [Isse07, 98f.], Ressourcenkontext und Umgebungskontext [Geih08, 137f.].

`Aufgabe→Analyseziel→Maßnahme.Branche` kürzer `Anwendungskontext.Branche` geschrieben werden, um Zugriff auf den Wert des Kontextfaktors zu erhalten. Das Metamodell fungiert dabei als generisches Modell von Einflussfaktoren [RuPR99, 226]. Bei Bedarf lassen sich Kontextsichten für einzelne Managementaufgaben durch Projektion weiter einschränken oder um externe Faktoren erweitern.

Anwendungsebene	**Anwendungskontext:** Problemaspekt Maßnahme		
Analyseebene	**Analysekontext:** Analyseziel Analyseproblem		
Prozessebene	**Prozesskontext:** Funktion Aufgabe Aktivität Schablone Fragment Flussbeziehung	**Ablaufkontext:** Prozessinstanz Vorgang Datenfluss	
Ressourcenebene	**Verfahrenskontext:** Operator Software-Produkt	**Ausführungskontext:** Software-Installation Person	**Datenkontext:** Informationsobjekt Datenquelle

Tabelle 1: Modellebenen und zugehörige Kontexte

Um kontextsensitives Prozessmanagement zu ermöglichen, sind die Kontextabhängigkeiten einzelner Gestaltungsoptionen bzw. -entscheidungen explizit zu formulieren. Sie definieren Bedingungen für die Anwendbarkeit der Optionen, indem sie konkrete Ausprägungen von Kontextfaktoren z.B. mit Prozessbausteinen verknüpfen [RuPR99, 227], [HaBR08, 50]. Kontextfaktoren können den Optionen in Form von Deskriptoren annotiert werden oder in Kontextregeln einfließen. Im ersten Fall lassen sich mithilfe der Deskriptoren geeignete Gestaltungs-

artefakte aus einer Prozessbibliothek selektieren (vgl. Abschnitt 4.5.4).[114] Im zweiten Fall können mit logischen Ausdrücken komplexere Bedingungen formuliert werden, um spezielle Handlungen (z.B. Einfügen, Entfernen oder Modifizieren von Prozessbausteinen) auszulösen oder Parameterwerte zu setzen. Ihre Modellierung beschreibt der folgende Abschnitt.

4.7.3 Restriktionen und Regeln

Kontextfaktoren sind zur Beschreibung von Zuständen geeignet, in denen bestimmte Optionen anwendbar sind (Ist-Perspektive). Die Einschränkung der Menge zulässiger Zustände oder Optionen erfolgt mithilfe von Restriktionen oder Regeln (Soll-Perspektive). *Restriktionen* (Constraints) kommen z.B. bei der Definition der Wertebereiche von Datenobjekttypen zum Einsatz. Zu ihrer Repräsentation eignet sich die von der OMG standardisierte OBJECT CONSTRAINT LANGUAGE (OCL) [OMG12]. Ein Ausdruck in OCL gilt stets für eine Instanz eines Objekttyps (Kontext der Restriktion) und nimmt auf ein oder mehrere Attribute Bezug. Um ein Attribut zu referenzieren, wird der Name der Instanz vor dem Attributnamen notiert, z.B.

```
Analyseziel3.Ausrichtung.
```
[115]

Hierbei sind auch Referenzen auf verbundene Objekte und deren Attribute möglich, indem entlang der Meta-Beziehungen durch das Metamodell navigiert wird, wie z.B.

```
Maßnahme8→Problemaspekt.Problemdomäne.
```

Zur Strukturierung langer Pfadausdrücke kann für die Navigation entlang von Beziehungen anstelle des Punktes ein Pfeil → verwendet

---

[114] Sollen nicht nur exakt passende Gestaltungsoptionen erkannt werden, ist eine Möglichkeit zur Ermittlung der Ähnlichkeit einer Option zu einer Anforderung vorzusehen. Dies kann z.B. auf Basis von Beschreibungslogik geschehen, die die Berechnung von Ähnlichkeitsmaßen zwischen Merkmalsvektoren erlaubt [EnLS97b, 5].

[115] Objekt- und Attributnamen werden im Rahmen dieser Arbeit stets gemäß ihrer Bezeichnung notiert, d.h., abweichend von den Konventionen der OCL ist auch Großschreibung erlaubt. Ebenso kann zur besseren Lesbarkeit auf das übliche Schlüsselwort `self` zur Referenzierung der Kontextinstanz verzichtet werden.

werden [RuCz11, 261]. Darüber hinaus steht die volle Ausdrucksmächtigkeit der OCL zur Verfügung, etwa die auf Collection-Attributen (mit Kardinalität _,*) definierten Operatoren. Die oben referenzierte Problemdomäne ist eine Collection (Set) und kann z.B. auf ihren Umfang hin abgefragt werden mit

```
Maßnahme8→Problemaspekt.Problemdomäne->size().
```

Für die referenzierten Variablen können sodann mit mathematischen oder logischen Operatoren Bedingungen formuliert werden, wie z.B.

```
Beispieldaten3.Alter<=3          oder
Beispieldaten3->isEmpty()=false.
```

Auf die Darstellung weiterer Details zur OCL wird verzichtet. Eine Einführung vermittelt z.B. [RiGo02]. Ausdrücke der beschriebenen Form können zur Formulierung von *Regeln* verwendet werden. Regeln sind allgemein Aussagen zur Bestimmung der Struktur oder zur Beeinflussung des Verhaltens eines Systems [RuCz11, 255]. Regeln, die auf Kontextfaktoren zurückgreifen, heißen ***Kontextregeln***. Sie sind geeignet, Erfahrungswissen über die Prozessgestaltung zu manifestieren [RuPR99, 230] und unabhängig vom Wissensträger verfügbar zu machen [RuCz11, 255f.]. Zur automatisierten Auslösung oder Durchführung von Gestaltungsoperationen durch Regeln müssen diese in maschinenlesbarer Sprache formuliert sein und die gerufenen Softwarekomponenten entsprechende Schnittstellen bereitstellen. Im Folgenden werden beispielhafte Funktionen in Pseudo-Code verwendet. Zum Beispiel definiert folgende Regel die Aufnahme einer Evaluierungsaufgabe in das Prozessschema nach der betrachteten Aufgabe (`this`), wenn der gewünschte Aussagetyp des Analyseergebnisses ein Prognosemodell ist:[116]

```
IF Analyseziel.Analysefrage.Aussagetyp=
  "_Prognosemodell"
THEN insert(new(Aufgabe→Funktion="Evaluierung"),
  after,this) ENDIF
```

[116] OCL notiert Regeln mit dem `implies`-Operator oder dem `if-then-else-endif`-Konstrukt.

Mit der folgenden Regel wird innerhalb einer Prozessaktivität der Verfahrensparameter `Support` in Abhängigkeit vom Datenvolumen gesetzt:

```
IF Beispieldaten3.Wert.Anzahl_Beispiele > 500000
THEN set(Support, 0.05)
ELSE set(Support, 0.02) ENDIF
```

Die Prüfung auf Erfüllung der in Restriktionen oder im Regelkopf gesetzten Bedingungen muss ontologische Relationen berücksichtigen, wofür geeignete Inferenzregeln zu definieren sind. Zur transitiven Prüfung von Spezialisierungsbeziehungen stellt OCL den Operator `oclIsKindOf(t)` bereit, der wahr liefert, wenn t direkter Typ oder ein Supertyp des Objekts ist. Demgegenüber prüft `oclIsTypeof(t)` nur auf direkte Typbeziehungen [OMG12, 22f.]. Normative Regeln formulieren Bedingungen an den Zustand eines Schemas oder einzelner Schemaobjekte. Deduktive Regeln dienen der Ableitung neuer Fakten aus der Ontologie. Zur formalen Repräsentation von Regeln als Bestandteil von Ontologien kann z.B. die OWL-Erweiterung SWRL (Semantic Web Rule Language) eingesetzt werden [ThFe09, 511-513].

Regeln sollten keinerlei implizite Annahmen treffen, sondern alle Fakten explizit formulieren [RuCz11, 255]. Sie werden im vorliegenden Ansatz nicht in einer schwer zu wartenden zentralen Regelbasis gespeichert, sondern lokal an den jeweils betroffenen Modellierungsobjekten annotiert. Hierfür steht im Supertyp aller Metaobjekttypen das Attribut `Kontextregeln` zur Verfügung. Lokal hinterlegte Regeln sind auch durch den Anwender leicht und flexibel anpassbar, wenn veränderte Rahmenbedingungen dies nahelegen [GrBC08, 269].

4.8 Zusammenfassung: Modellierung von Datenanalyseprozessen

Die Spezifikation einer Datenanalyse erfolgt grundsätzlich durchgängig über alle vier Ebenen der Analysearchitektur von oben nach unten. Die Betrachtung aller Ebenen ist jedoch nicht zwingend. Die *Anwendungsebene* dient der Einbettung analytischer Projekte in einen fachlich-organisatorischen Rahmen und benennt dabei auch Zielgrößen und Restriktionen aus der Wertsphäre einer Organisation. Erscheint die

Darstellung dieser Rahmenbedingungen sowie eine Rückkopplung mit der Wertsphäre im Rahmen der Evaluation des Analysevorhabens nicht erforderlich, kann auf diese Ebene verzichtet werden. Zugleich unterstützt die Anwendungsebene die Analyse von Problemen und die Gestaltung geeigneter Lösungsansätze auch außerhalb des Einsatzbereichs der Datenanalyse. Die *Analyseebene* dient der detaillierten Beschreibung des Informationsbedarfs (Output) und der zu verwendenden Daten (Input) für eine Analyse sowie der Verkettung mehrerer Analysen. In Fällen, in denen Informationsbedarf und Analysedaten vollständig bekannt oder trivial sind (z.B. bei einfachen Datenbankanfragen oder Datenanalysen innerhalb geschlossener operativer Anwendungssysteme), kann die Modellierung auf der Analyseebene unterbleiben. Die Analyseaufgabe auf *Prozessebene* liefert in solchen Fällen eine ausreichende Spezifikation der auszuführenden Untersuchung. Die *Ressourcenebene* dient der Repräsentation vorhandener Datentransformations- und -analyseverfahren (Operatoren) sowie der Beschreibung von Datenquellen und deren Inhalt. Soll eine allgemeine Handlungsempfehlung für Datenanalysen im Sinne abstrakter Prozessvorlagen erstellt werden, die unabhängig von konkreten Werkzeugen, Anwendungssystemen und Datenquellen gelten sollen, kann die Ressourcenebene prinzipiell ignoriert werden.

Besonderes Merkmal der Datenanalysearchitektur ist die *Differenzierung zwischen Aufgaben- und Aufgabenträgerebene*. Die Aufgabenebene umfasst die Anwendungs-, Analyse- und Prozessebene. Sie stellt eine ganzheitliche Spezifikation der Problemlösungsbeiträge, Leistungen und Schritte einer Datenanalyse bereit (Fachkonzept, vgl. [Sinz97, 5]). Die zu ihrer Durchführung bereitstehenden Ressourcen werden unabhängig davon auf der Aufgabenträgerebene spezifiziert und den Aufgaben flexibel zugewiesen. Die Verbindung stellt die Workflow-Sicht der Prozessebene her. Auf den einzelnen Ebenen lassen sich grobe *Problemlösungspläne* (Anwendungsebene), komplexe *Analysestrategien* (Analyseebene) sowie konkrete *Detailpläne* (Prozessebene) für analytisch gestützte Projekte abbilden, die jeweils von spezialisierten Experten zusammen mit geeigneten fachlichen Ansprechpartnern separat erarbeitet und diskutiert werden können. Die Datenanalysearchitektur unterstützt das Qualitätsmanagement von Datenanalysen, indem sie die

genannten Pläne strukturiert dokumentiert und über die Ebenen hinweg verknüpft. Die Gliederung in Ebenen erleichtert die Wiederverwendbarkeit von Modellierungsartefakten. Insbesondere können die auf Ressourcenebene erstellten Beschreibungen von maschinellen Aufgabenträgern und Datenbeständen projektübergreifend genutzt werden. Die Anpassbarkeit der Artefakte wird durch die Strukturierung in Teilmodellsysteme und die ebenenübergreifende Abstimmung vereinfacht (vgl. [Sinz97, 12-15]). Entsprechende Beziehungsmetamodelle sind in Anhang A4.7 dargestellt.

Mit Ausnahme der Anwendungsebene wird jeweils explizit zwischen Typ- und Instanzebene unterschieden. Dies unterbleibt auf der Anwendungsebene, da Problemaspekte und Maßnahmen aufgrund ihrer hohen Spezifität und ihrer Bindung an konkrete Zeitpunkte oder -räume stets Instanzen darstellen. Die Einführung einer zusätzlichen Typensicht würde die Übersichtlichkeit und Akzeptanz des Ansatzes durch weitere Modellbausteine beeinträchtigen, die als nicht notwendig erachtet werden. Die Übertragbarkeit der Instanzen auf andere Anwendungsfälle (Wiederverwendbarkeit) ist durch Abstraktion von Details problemlos möglich.

Dieses Kapitel präsentiert eine eher statische Sicht auf Datenanalyseprozesse, die durch deren Abbildung mithilfe des vorgestellten Modellierungsansatzes zu einem bestimmten Zeitpunkt entsteht (Schnappschuss). Die Nutzung der Modellsysteme zur Unterstützung der Planung, Steuerung und Revision – und damit auch die Herleitung bestimmter Modellzustände – ist Gegenstand der folgenden Kapitel.

5 Planung von Datenanalyseprozessen

Das vorliegende Kapitel erörtert die methodisch gestützte Planung von Datenanalyseprozessen. Hierzu legt Abschnitt 5.1 die Grundlagen und betrachtet Prozessplanung als Gestaltungsaufgabe. Abschnitt 5.2 leitet eine Planungsstrategie her, und Abschnitt 5.3 zeigt anhand einschlägiger Arbeiten aus der Literatur vier Basisansätze zur Prozessplanung. Das konkrete Vorgehen zur Problemspezifikation und Prozessspezifikation beschreiben die Abschnitte 5.4 und 5.5.

5.1 Prozessplanung als Gestaltungsaufgabe

Die folgenden Abschnitte beleuchten den Begriff der Planung und ihre Relevanz für die Datenanalyse. Anschließend werden Ziele und Anwendungsfälle der Analyseprozessplanung dargestellt sowie Optionen zur Konstruktion flexibler Prozesse erläutert.

5.1.1 Der Planungsbegriff

Der Begriff Planung wird in verschiedenen Disziplinen mit individuellen Bedeutungen ausgefüllt, die sich im Wesentlichen unter *Vorbereitung künftigen Handelns* subsumieren lassen [CoRe08, 3]. Im Folgenden seien stellvertretend zwei Perspektiven auf den Begriff angeführt. In der Betriebswirtschaftslehre steht Planung in der Regel im Kontext der Unternehmensführung und im Zusammenhang mit Entscheidungsprozessen.[117] So versteht SCHNEEWEIß darunter die Gestaltung künftigen Handelns durch gedankliche Vorwegnahme von Ereignissen [Schn91, 1f.], und nach WILD ist Planung ein „systematisch-methodischer Prozess der Erkenntnis und Lösung von Zukunftsproblemen“ [Wild74, 13]. Dieses Verständnis umfasst die Festlegung von Zielen, Maßnahmen und Mitteln (Ressourcen) zur Erreichung der gesetzten Ziele [Wild74, 13], [CoRe08, 3]. Ähnliche Inhalte zeigt die Planungsaufgabe der Künstlichen Intelligenz (KI): RUSSELL & NORVIG definieren Planung als Erzeugung einer Folge von Handlungen (Aktionen), die zur Erreichung eines gegebenen Ziels geeignet ist

[117] Planung ist als Entscheidungsaufgabe einzuordnen [FeLW11, 157].

[RuNo03, 375]. Im Vergleich zum betriebswirtschaftlichen Verständnis sind Ziel und Operatoren hier vorgegeben [Hert89, 15].

Planung zeichnet sich allgemein durch folgende Merkmale aus [Wild74, 13f.], [FeLW11, 155]:

- *Gestaltungscharakter*: Planung zielt auf die Bereitstellung eines Lösungsvorschlags, der zur Handhabung eines definierten Problems dienen soll.
- *Zukunftsbezogenheit*: Planung findet zeitlich vor der Maßnahmenrealisierung statt.
- *Unvollkommene Information*: Zur Entwicklung der Lösung sind Informationen über Ziele, Maßnahmen, deren Wirkungen und Einflussgrößen zu verarbeiten, die zur Planungszeit häufig nur unvollständig verfügbar sind (Unsicherheit) [FeLW11, 158].
- *Methodisches Vorgehen*: Die Konstruktion des Lösungsvorschlags soll durch bewusstes, zielgerichtetes Denken und methodisch-systematisches Vorgehen geprägt sein.[118]
- *Prozessphänomen*: Planung ist in der Regel kein einmaliger Vorgang, sondern ein sich mehrstufig wiederholender Zyklus, der Züge eines Lernprozesses aufweist.

In der vorliegenden Arbeit wird der Planungsbegriff im umfassenden Sinne verstanden und schließt neben der Ableitung und Strukturierung von Handlungsmaßnahmen auch die Spezifikation des zu lösenden Problems und die Entwicklung von Zielen ein. Wird Prozessgestaltung als zielgerichtete Konstruktion aufgefasst (vgl. Abschnitt 2.4.3.1), so beinhaltet diese Aufgabe stets die Vorgabe des gewünschten Systemverhaltens und die Bestimmung einer Systemstruktur, die dieses Verhalten realisieren kann (vgl. [Fers79, 44f.]). Da ein System als Potenzialgefüge jeweils ein gewisses Verhaltensrepertoire ermöglicht [Beck01, 18], ist das zulässige Verhalten zur Gewährleistung der Zielerreichung so präzise wie möglich festzulegen.

[118] Dies ist gemäß klassischem Planungsansatz mit Rationalität gleichzusetzen [Wild74, 13], [FeWL11, 155] .

5.1.2 Relevanz der Planung für die Datenanalyse

Planung gilt als besonders hilfreich in komplexen Problemsituationen. Die vorweggenommene Problemanalyse und -lösung soll künftigen Entscheidungs- und Koordinationsbedarf substituieren, indem als zielführend erachtete Handlungsoptionen antizipativ selektiert und in zulässige Lösungsvorschläge überführt werden [Wild74, 15-17], [Schn00, 491]. Angesichts der oft hohen Komplexität von Datenanalyseprozessen (vgl. Abschnitt 3.2) erscheint deren Planung somit grundsätzlich nützlich. Neben Prozessen und Workflows sind auch Analysestrategien (Analyseketten) unmittelbar als Pläne interpretierbar [AmCo94, 51]. Zugunsten des experimentell-evolutionären Ansatzes wird jedoch häufig auf eine Planung verzichtet (vgl. Abschnitt 3.1.3 und [Baro13, 39f.]). Für Projekte überschaubaren Ausmaßes mag ein ungeplantes Vorgehen fruchtbar sein, mit wachsender Komplexität sinkt jedoch die Erfolgswahrscheinlichkeit [WSG+97, 244]. Besonderen Wert für die *Effektivität* von Datenanalysen besitzt die Problemspezifikation (vgl. R1.1, Abschnitt 3.2.4.2). Erst die Herleitung von Analyseprozessen aus fachlichen Zielen ermöglicht die systematische Erfolgskontrolle und Planüberwachung zur Erkennung erforderlicher Korrekturmaßnahmen.[119] Aus Sicht der *Effizienz* kann Planung zur Vermeidung von Fehlern und Iterationen beitragen, indem sie einen Teil der zu berücksichtigenden Interdependenzen ex ante offenlegt und kalkulierbar macht. Werden gleichartige Analysen nach einem einheitlichen Plan realisiert, erlaubt die damit einhergehende Dokumentation analytischer Expertise gewisse Kosteneinsparungen infolge des verringerten Bedarfs an Spezialisten [KoRS02, 46]. Die dadurch erreichte Verstetigung des Handelns dient der Qualitätssicherung. Der wesentliche Vorteil der Planung gegenüber dem ungeplanten Vorgehen besteht mithin in der frühzeitigen Erkennung und Berücksichtigung von Möglichkeiten und Beschränkungen [Wild74, 16f.] (vgl. v.a. R2.3, R3.1, R3.2, Abschnitt 3.2.4.2).

---

[119] In diesem Zusammenhang bemerkt bereits WILD treffend: „Es ist also nicht nur die (Zweck-) Rationalität der Mittel, sondern auch die der Ziele zu prüfen, um zu verhindern, dass die hohe Rationalität der Mittelverwendung durch eine Irrationalität der Zwecke in Frage gestellt wird. Denn wer ‚falsche' Ziele verfolgt, löst falsche Probleme; wer falsche Maßnahmen ergreift, erreicht seine Ziele nicht" [Wild74, 15].

Ihre wesentliche Einschränkung besteht in der zur Planungszeit herrschenden Unsicherheit (Informationsmangel), die oft nur unvollständige Pläne erlaubt [Schn91, V]. Zudem verbraucht die Planungstätigkeit ihrerseits Zeit und Ressourcen. Der erste Mangel lässt sich teilweise heilen, indem ursprünglich nicht planbare Aspekte durch Formulierung geeigneter Annahmen in planbare Aspekte überführt werden [FeLW11, 163]. Darüber hinaus soll Planung nach MALIK insbesondere die *Flexibilität* des zu planenden Systems sicherstellen [Mali00, 65]. Dies geschieht mithilfe von Handlungsspielräumen [Wild74, 16], die durch Verzicht auf detaillierte Festlegungen oder durch Berücksichtigung mehrerer Handlungsoptionen entstehen und folglich mit weniger Information und geringerem Planungsaufwand auskommen (vgl. B2.3, Abschnitt 3.2.4.3).[120] Planung kann in diesem Sinne geradezu als Instrument zum effizienten Umgang mit Informationsmangel in komplexen Situationen betrachtet werden und erscheint vor diesem Hintergrund gerade für umfangreiche Datenanalysen und explorative Untersuchungen geeignet.

5.1.3 Ziele und Ergebnisse der Analyseprozessplanung

Die Planung von Datenanalyseprozessen gemäß der hier vorgestellten Methodik erfolgt modellbasiert, d.h., sie erzeugt bzw. stützt sich auf die Modellsysteme der Datenanalysearchitektur (vgl. Abschnitt 4.2). Ihr Ergebnis sind demnach die zugehörigen Schemata in Form von Problemkarten, Analyseketten, Prozessen bzw. Workflows (Pläne).[121]

5.1.3.1 Erstellung von Plänen für effektive Datenanalysen

Die vollständige Planung einer Datenanalyse umfasst die ***Problemspezifikation***, in der ein Sachproblem identifiziert (Anwendungsebene) und in Gestalt eines oder mehrerer Analyseprobleme operationalisiert

[120] Vgl. hierzu WALTER GROPIUS: „Planen heißt nicht festlegen, sondern offen halten von Möglichkeiten für die Zukunft“ (zitiert bei [Jeck97, Titelei]).

[121] Pläne stellen das informationelle Resultat der Planung dar. Sie treffen im Allgemeinen Aussagen über Probleme, Ziele, Prämissen, angestrebte Ergebnisse und Wirkungen, Aufgaben, Ressourcen und Termine [Wild74, 14].

wird (Analyseebene), sowie die ***Prozessspezifikation***, deren Resultat eine Aufgabenfolge (bzw. ein Aufgabennetz) zur Lösung jeweils eines Analyseproblems ist (Prozessebene). Problem- und Prozessspezifikation bestimmen gemeinsam Verhalten und Struktur des Analyseprozesses. Die Planung auf Anwendungs- oder Analyseebene kann in bestimmten Fällen unterbleiben (vgl. Abschnitt 4.8), sofern die Zielstellung eindeutig und der Pfad der Untersuchung geradlinig oder einstufig erscheinen. Die Planung auf Prozessebene ist verzichtbar, wenn die Analyse sich z.B. als einfache Datenbankabfrage darstellt oder im agilen, explorativen Umfeld stattfindet (Data Science, vgl. Abschnitt 2.2.2.7). Im Fall explorativer Untersuchungen ist auch bei Verzicht auf die Prozessspezifikation eine Problemspezifikation anzuraten, da sie die Zielerreichung verbessert oder erst ermöglicht (vgl. Abschnitt 5.1.2). Sie dient gerade im Kontext der evidenzbasierten Problemlösung (Business Analytics, vgl. Abschnitt 2.2.2.8) der Effektivität der Untersuchungen. Das *Sachziel* der Analyseprozessplanung besteht nicht in erster Linie in der Produktion von Plänen oder Schemata, sondern vielmehr in der Sicherstellung der *Effektivität* der Analysen im jeweiligen Anwendungs- und Ausführungskontext.

Die Prozessspezifikation erfolgt unter Bezugnahme auf die Ressourcenebene, deren Elemente als gegeben vorausgesetzt werden. Die Gestaltung der Aufgabenträgerkonfiguration und die Entwicklung von Transformationsverfahren sind damit nicht Gegenstand der Analyseprozessplanung. Die Eigenschaften der Ressourcen nehmen jedoch Einfluss auf die Prozessgestaltung, indem sie über die Menge der verfügbaren Operatoren die durchführbaren Aufgaben und deren Verknüpfungsoptionen determinieren.

5.1.3.2 Sicherstellung effizienter und flexibler Datenanalysen

Die *Formalziele* der Analyseprozessplanung lassen sich anhand der Verhaltens- und Strukturdimension sowie der Unterscheidung zwischen Prozessspezifikation und Prozessrealisierung systematisieren.[122] Bezüg-

[122] Vgl. zu dieser Systematisierung [Fers92, 11f.]. Die Zuordnung der Anforderungen zu den genannten Kriterien dient primär ihrer vollständigen Herleitung. Tatsächlich sind

lich des Verhaltens resultieren daraus die Forderungen nach Korrektheit der Prozessspezifikation und nach Effizienz der Prozessrealisierung. Bezüglich der Struktur sind die Integration der Pläne und Flexibilität bei ihrer Realisierung zu fordern. *Korrektheit* bezieht sich auf die Effektivität des Prozesses, d.h. auf seine Fähigkeit, ein zweckdienliches Analyseergebnis zu liefern (Zielerreichung). Dies setzt Vollständigkeit sowie Eindeutigkeit und Widerspruchsfreiheit (Korrektheit i.e.S.) voraus (Abstimmung der Komponenten der Schemata sowie mit Schemata anderer Modellebenen). Die *Effizienz* eines Prozessablaufs ist gegeben, wenn die beabsichtigte Analyse mit geringem Zeitbedarf und zu niedrigen Kosten realisierbar ist (vgl. [KoRS02, 46]). Dieses Ziel kann durch möglichst hohe *Integration* der Komponenten der Schemata befördert werden, die sich strukturorientiert auf Redundanz und Verknüpfung sowie verhaltensorientiert auf Konsistenz und Zielorientierung bezieht (vgl. [Fers92, 12f.]). Die Integration nimmt auch Einfluss auf die Korrektheit. Die *Flexibilität* betrifft die Eigenschaft der Pläne, die Erreichung gesetzter Ziele trotz veränderter Problembedingungen oder auftretender Störeinflüsse zu unterstützen.

Vollständigkeit der Pläne

Die Vollständigkeit eines Plans ist stets vor dem Hintergrund der jeweiligen Planungssituation zu beurteilen. Wie zuvor geschildert, sind nicht zwingend alle Ebenen der Analysearchitektur zu betrachten, und auf Prozessebene muss nicht in jedem Fall ein ausführbarer Plan in Form eines Workflow-Schemas erstellt werden. Vielmehr stellen auch Teil- oder Rahmenpläne zulässige Planungsergebnisse dar, sofern sie einen Beitrag zur Bewältigung der Analysekomplexität leisten. Verbleibende Freiheitsgrade oder Planungslücken sind im Rahmen der Prozesssteuerung situativ auszufüllen.

sie durch Interdependenzen verknüpft. Die Zuordnung der Flexibilität folgt der Erwägung, dass diese Anforderung bei der Prozessrealisierung schwerer wiegt als bei der Prozessspezifikation.

Korrektheit der Prozessschemata

Die Korrektheit i.e.S. eines Prozessschemas dient der Vermeidung von Fehlern während der Prozessrealisierung, die aufwändige Anpassungen und Iterationen auslösen können. Sie ist bei eindeutig definierter Semantik der Prozesselemente (vgl. Abschnitt 4.5.2.4) mittels automatischer Verifikationsverfahren vollständig nachweisbar [Rögl09, 496-500]. Unter dem Strukturaspekt ist die Konformität mit dem jeweiligen Metamodell sowie die syntaktische Übereinstimmung der Schnittstellen verknüpfter Prozesselemente zu prüfen [Reif03, 81]. Sind die Prozesselemente entsprechend ontologisch annotiert, kann zusätzlich auf semantische Kompatibilität der Schnittstellen geprüft werden [FHTN11,26].[123] Unter dem Verhaltensaspekt sind Ausführbarkeit und Terminierbarkeit der Prozesse zu untersuchen, die z.B. durch Verklemmungen, unerwünschte Zyklen oder Konflikte beeinträchtigt werden [DRRA05, 4].[124] Dabei sollten Restriktionen und Kontextregeln Beachtung finden [LaPh94, 26].

Die Schemakorrektheit kann bei entsprechender Werkzeugunterstützung bereits beim Entwurf (einschließlich aller späteren Anpassungen und Abweichungen) überwacht werden [HaBR08, 54f.].[125] Moderne Workflow-Management-Systeme erlauben nur konsistenzerhaltende Gestaltungsoptionen bzw. melden erkannte Modellierungsfehler sofort an den Anwender [GLKK09, 42], [DaRR11, 370-373]. Auch die interaktive Simulation der Abläufe mit grafischer Animation des dynamischen Verhaltens erlaubt die Erkennung von Modellierungsfehlern vor der Prozessausführung (vgl. [CSG+92, 10], [Reif03, 82]).

---

[123] Einen entsprechenden Vorschlag, der auf dem in Abschnitt 4.7.1.3 erwähnten Ansatz zur semantischen Prozessmodellierung [ThFe09] beruht, präsentieren FELLMANN ET AL. [FHTN11].

[124] Ausführbarkeit und Terminierbarkeit sind erreicht, wenn jedes Prozesselement vom Startknoten des Schemas erreichbar ist und einen Pfad zu einem Endknoten besitzt [ReDa98, 7f.]. Konflikte resultieren aus Interaktionen, bei denen eine Aktivität das Ergebnis einer anderen zerstört oder verändert [Hert89, 99-101].

[125] Einen Überblick über Korrektheitskriterien für dynamische Workflow-Änderungen und Realisierungsansätze geben z.B. RINDERLE ET AL. [RiRD04].

5.1.4 Anwendungsfälle der Prozessgestaltung

Die Anwendbarkeit eines Prozessplans auf eine vorliegende Problemsituation kann durch vor oder während der Prozessausführung auftretende Störgrößen massiv beeinträchtigt werden [FeMa95a, 2]. Sie können fachliche und technische Ursachen haben. Fachliche Ursachen sind etwa veränderte Rahmenbedingungen des Anwendungskontexts oder neue Analyseziele. Auch objektive Planungsmängel sind hier einzuordnen. Beispiele für technische Ursachen sind während eines Vorgangs aufgetretene Hardware- oder Software-Fehler, die Nichtverfügbarkeit von Ressourcen oder die falsche Repräsentation von Eingabedaten (vgl. [ReDa98, 1], [Reic00, 20f.], [DeKl00, 51f.]). Bekannten Störgrößen kann bei der Planung durch geeignete Gestaltung des Prozessschemas Rechnung getragen werden [RiDa03, 17].

Zur Entwurfszeit des Prozessschemas unbekannte Störgrößen erfordern gesonderte Behandlung zur Ausführungszeit und bedingen hauptsächlich die eingeschränkte Planbarkeit von Prozessabläufen. Nicht vorhersehbare bzw. nicht berücksichtigte Störereignisse werden auch als *Ausnahmen* bezeichnet. Je nach Verfügbarkeit planungsrelevanter Informationen (Planungsgewissheit) können Prozessausführungen vollständig planbar, teilweise planbar oder nicht planbar sein[126] [Gait83, 178-180], [BeSc95, 282], [Reic00, 17-20]. Abhängig davon sind die folgenden drei ***Ablaufarten*** (Instanzebene) zu unterscheiden:

- **Programmierte Abläufe** sind vollständig planbar. Sie folgen in Struktur und Verhalten genau der im Voraus getroffenen Spezifikation und lassen sich in unmodifizierter Form mehrfach wiederholen. Hierzu sind Workflow-Schemata erforderlich, die allen Aktivitäten geeignete Aufgabenträgertypen zuordnen und relevante Interdependenzen vollständig erfassen. Sie können zulässige

[126] Die Übergänge zwischen diesen Kategorien sind fließend, da sich die Planbarkeit verschiedener Merkmale eines konkreten Prozesses unterschiedlich darstellen kann [Reic00, 17]. Eine strikte Zuordnung von Prozessen in verschiedene Kategorien ist in der Praxis nicht möglich [Reic00, 19] und im Allgemeinen auch nicht notwendig. Letzlich ist während der Prozessausführung stets situativ zu entscheiden, ob der aktuelle Ablauf (weiterhin) einem gegebenen Schema folgen kann oder modifiziert werden muss.

Ablaufvarianten enthalten, um das Prozessverhalten kontextabhängig und deterministisch an zu erwartende Störereignisse anzupassen [RiDa03, 17].

Unterbleibt eine derart detaillierte Spezifikation, so sind die endgültige Ablaufstruktur bzw. Eigenschaften einzelner Prozesselemente während der Ausführung fallweise zu bestimmen [FeSi13, 64]. Entwurfs- und Ausführungszeit solcher nicht vollständig planbaren Abläufe überlappen zumindest teilweise. Sie können wie folgt ausgeprägt sein:

- **Variable Abläufe** sind aufgrund der Unbestimmtheit einzelner Problemmerkmale nur teilweise planbar bzw. werden absichtlich nicht vollständig geplant, um ihr Verhaltensrepertoire durch Ausnutzung von Freiheitsgraden gezielt variieren zu können (vgl. R3.3, Abschnitt 3.2.4.2 sowie [PWFS09, 1f.]). Die zugehörigen Prozessschemata sind soweit differenziert wie zur Entwurfszeit möglich oder sinnvoll und geben zumindest einen Rahmenplan vor. Sie werden unmittelbar vor oder während der Prozessausführung modifiziert, um auf fallspezifische Anforderungen oder Ausnahmen zu reagieren. Infolge dieser situativ getroffenen Entwurfsentscheidungen wiederholen sich variable Abläufe typischerweise nicht mit gleichbleibender Struktur.
- **Ad-hoc-Abläufe** entstehen, wenn noch kein Prozessschema vorliegt, weil eine Planung wegen Informationsmangels unmöglich ist oder aufgrund der Seltenheit des Problemfalls nicht lohnend erscheint [ReSt04, 25], [WRRW05, 8f.]. Ergebnis sind innovative Abläufe, die während ihrer Ausführung situativ konstruiert werden. Sie unterscheiden sich daher meist stark von anderen Prozessabläufen. Sofern Prozessschemata existieren, sind diese meist abstrakt und spezifizieren die Prozessaufgaben nur aus Außensicht, wodurch der Handlungsraum vergleichsweise wenig beschränkt ist.

Gestaltungsaufgaben treten demnach zu verschiedenen Zeitpunkten und gegebenenfalls mehrfach im Prozesslebenszyklus auf. Sie können

die *repetitive* oder *adaptive*[127] *Wiederverwendung* bestehender Prozessschemata und die *innovative Konstruktion* gänzlich neuer Abläufe beinhalten. Durch ihre zeitliche Zuordnung zu den Phasen des Prozessmanagements ergeben sich die in Abbildung 61 dargestellten sechs Anwendungsfälle der Prozessgestaltung, die in Bezug auf einen konkreten Ablauf auch kombiniert auftreten können.

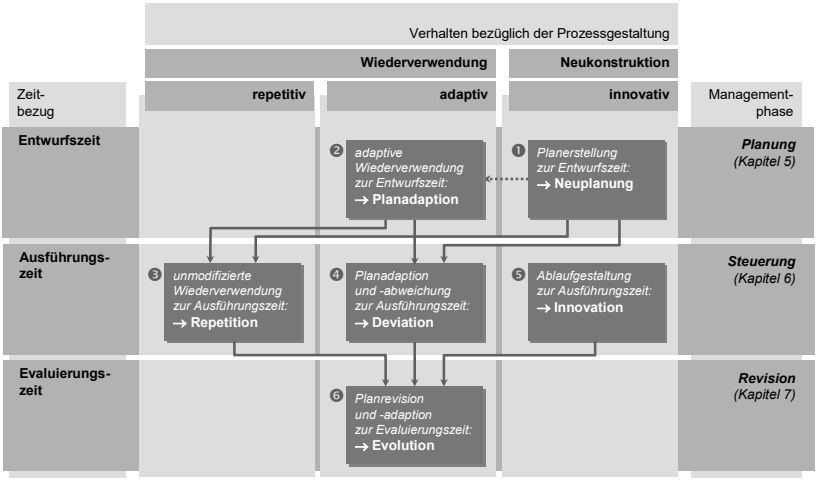


Abbildung 61: Anwendungsfälle der Prozessgestaltung und Zuordnung zu den Prozessmanagementphasen (eigene Darstellung)

Im einfachsten Fall erfolgt die Erstellung eines gänzlich neuen Plans (Schemas) zur Entwurfszeit (*Neuplanung*; 1), der idealerweise zur Ausführungszeit unmodifiziert realisiert wird (*Repetition*; 3).[128] Falls im

[127] Der Begriff adaptiv wird hier im allgemeinen Sinne von auf Anpassung beruhend verwendet und nicht als Eigenschaft des Prozesses, aktiv zur Anpassung fähig zu sein, verstanden.

[128] Da Analyseprozesse höhere Lösungsverfahren darstellen, die sich aus der kombinierten Anwendung mehrerer einfacher Algorithmen zusammensetzen, sind sie durch eine hohe Spezifität gekennzeichnet. Die allgemeine *m:n*-Beziehung zwischen Problemen und Verfahren [Knob01, 65] gilt für sie daher nur mit starken Einschränkungen. Voraussetzung für die mehrfache Eignung eines Analyseprozesses sind typgleiche Analyseprobleme, die sich lediglich bezüglich der Inhalte der Analysedaten (Instanzebene) unterscheiden. Bei übereinstimmendem Datenschema können auf

Voraus bekannt ist, dass ein vorliegender Prozessplan nicht direkt auf die aktuelle Problemsituation anwendbar ist, kann eine entsprechende *Planadaption* zur Entwurfszeit vorgenommen (2) und das modifizierte Schema wiederum direkt ausgeführt werden (3). Zur Ausführungszeit auftretenden Ausnahmen muss durch geeignete Maßnahmen der Planadaption und -abweichung begegnet werden (*Deviation*; 4).[129] Steht kein für das aktuelle Problem geeignetes Prozessschema zur Verfügung, muss durch situative Ablaufgestaltung zur Ausführungszeit ein passender Prozess definiert werden (*Innovation*; 5). Ist zu erwarten, dass ein erfolgreicher Prozessablauf zu einem späteren Zeitpunkt erneut eingesetzt werden kann, lässt sich im Rahmen der Evaluierung eine Revision des zugehörigen Prozessschemas durchführen. Programmierte Abläufe können hierbei weiter verbessert, situative Änderungen an variablen Abläufen in bestehende oder neue Schemata übernommen sowie Ad-hoc-Abläufe als eigene Schemata archiviert werden (*Evolution*, 6) (vgl. [WRWR05b, 5]).

Aus Abbildung 61 ist ersichtlich, dass die Planung nur jene Anwendungsfälle umfasst, die zeitlich vor der Ausführung zur Entwurfszeit stattfinden und sich dementsprechend auf Prozesstypen beziehen: innovative Planerstellung (Neuplanung, 1) und adaptive Wiederverwendung (Planadaption, 2). Sie werden in Abschnitt 5.3 zusammen mit Vorschlägen aus der Literatur diskutiert. Die Fälle 3 bis 5 operieren auf Instanzebene und finden während der Prozessausführung statt. Sie sind Gegenstand der Steuerung von Datenanalyseprozessen in Kapitel 6. Die der Prozessrealisierung nachgelagerte Evolution von Prozessschemata (Fall 6) wird der Revision von Datenanalyseprozessen zugerechnet (Kapitel 7).

Grundlage desselben Prozessschemas typgleiche Analyseergebnisse mit datenspezifischen Aussageinstanzen erzeugt werden.

[129] Zur besseren begrifflichen Unterscheidung zwischen der Anpassung von Prozessschemata zur Entwurfszeit und jener zur Ausführungszeit wird der erste Fall als *Planadaption*, der zweite als *Deviation* bezeichnet.

5.1.5 Planung flexibler Prozesse

Aus den beschriebenen Ablaufarten ergeben sich besondere Anforderungen an die Flexibilität von Analyseprozessen, die umso höher ausfallen, je geringer die Planungsgewissheit ist [Wagn+11, 59]. Einer systemtheoretisch fundierten Analyse des Flexibilitätsbegriffs von WAGNER ET AL. zufolge ist Flexibilität „die Fähigkeit eines Systems, auf System- oder Umweltveränderungen unter Berücksichtigung gegebener Ziele durch Anpassung von Struktur und/oder Verhalten zu reagieren oder sie zu antizipieren“ [WSLF11a, 88]. Dieser Definitionsvorschlag impliziert die Fähigkeit zur autonomen Anpassung (vgl. wandlungsfähige, (selbst-) adaptive Systeme [AnGS05, 65]). Unter der für Analyseprozesse relevanten Annahme, dass die Erkennung des Anpassungsbedarfs und die Realisierung geeigneter Maßnahmen auch systemextern erfolgen können, wird ***Flexibilität*** als zielorientierte Anpassbarkeit von Struktur oder Verhalten an veränderte oder zur Entwurfszeit unbekannte Bedingungen verstanden.[130]

5.1.5.1 Realisierungsoptionen von Prozessflexibilität

Angelehnt an eine Systematisierung nach KANNENGIESSER, die auf einer interdisziplinären Untersuchung bekannter Ansätze aus Entwurfsperspektive basiert [Kann10, 49f.],[131] werden im Folgenden Optionen zur Erreichung von Prozessflexibilität beschrieben und den Anwendungsfällen aus Abschnitt 5.1.4 zugeordnet. Die ersten beiden Optionen erlauben antizipierte (geplante) Anpassung, die letzten beiden nicht-antizipierte (ungeplante) Anpassung (vgl. [Geih08, 135]).

---

[130] Flexibilität ist eng mit der ***Robustheit*** verknüpft, die denselben Zweck über die Unempfindlichkeit gegenüber Veränderungen zu erreichen sucht [Kann10, 47]. Robuste Prozesse sind demnach solche, die von möglichen Störeinflüssen abstrahieren. Robustheit kann a priori stets nur im Hinblick auf antizipierte Störgrößen, bezüglich vorher unbekannter Ausnahmen jedoch nur a posteriori beurteilt werden.

[131] Die Klassifizierung ist vollständig bezüglich der Struktur- und Verhaltensdimension von Prozessen. Sie berücksichtigt die Sichten function, behaviour und structure [Kann10, 45].

Geplante Verhaltensflexibilität

Prozessschemata werden gezielt mit einem Verhaltensrepertoire ausgestattet, das die Anpassung des Ablaufs an zu erwartende Störgrößen erlaubt. Dazu wird zur Ausführungszeit ein situativ passendes Verhalten realisiert, ohne die gegebene Struktur des Prozessschemas zu verändern. Das Verhaltensrepertoire kann durch Variantenbildung oder Parametrisierung aufgebaut werden [WaFe10, 123], [Geih08, 134], [RuNo03, 431]. *Variantenbildung* bezeichnet die explizite Berücksichtigung alternativer Ausführungspfade in der Prozessstruktur. *Parametrisierung* bezieht sich auf einzelne Aufgaben und dient der Beeinflussung der Vorgangsdurchführung. Geplante Verhaltensflexibilität wird im Rahmen der Neuplanung in die Prozessschemata „hineinkonstruiert", kann aber auch während der Planadaption und der Evolution ergänzt werden. Die Nutzung des Flexibilitätspotenzials erfolgt durch dynamische Auswertung situativer Bedingungen (Kontextfaktoren) während der Prozessausführung oder bei der Instanziierung eines Ablaufs. Die Einstellung oder Änderung der Operatorparameter ist ein typisches Beispiel für den zweiten Fall.

Geplante Strukturflexibilität

Prozessschemata werden an definierten Stellen bewusst nicht vollständig bis auf Aktivitätsebene spezifiziert, um Handlungsspielräume offenzuhalten, die erst vor oder während der Prozessausführung ausgefüllt werden und somit die Anpassung an spezifische Bedingungen erlauben (Schablonen, vgl. Abschnitt 4.5.4.1). Die detaillierte Spezifikation von Teilen der Prozessstruktur kann dadurch solange aufgeschoben werden, bis mehr Information über die Problemsituation vorliegt. Die Ausfüllung der Platzhalter kann auf zwei Arten erfolgen. Beim *Late Binding* werden vorspezifizierte Prozessfragmente mit passendem Verhalten aus einer Bibliothek ausgewählt und in das Schema integriert. Diese Option erweitert die Variantenbildung auf Fälle mit sehr vielen Varianten, die aus Komplexitätsgründen nicht

direkt im Schema repräsentiert werden können [Geih08, 135].[132] *Late Modeling* ermöglicht die freie Spezifikation fehlender Strukturdetails, wenn keine geeigneten Fragmente existieren [Deit00, 279f.]. Geplante Strukturflexibilität wird in der Regel während der Neuplanung oder Evolution modelliert und erfordert weitere Gestaltungsentscheidungen im Rahmen der Planadaption oder Innovation.

Flexibilität durch nicht-antizipierte Modifikation

Beim vorgelagerten Schemaentwurf nicht berücksichtigtem Anpassungsbedarf kann vor Prozessausführung durch geeignete Modifikation entsprochen werden. Die Änderungen können beliebige Struktur- und Verhaltensmerkmale einzelner Aufgaben oder des Prozesses im Ganzen betreffen [FeLW11, 153]. Diese Option ist nur während der Planadaption relevant. Die Übernahme von während der Prozessausführung an der Prozessinstanz vorgenommenen Änderungen in das Schema stellt hingegen keine flexibilitätsbedingte Modifikation, sondern eine antizipative Prozessanpassung für künftige Problemfälle dar, die im Rahmen der Evolution erfolgt.

Flexibilität durch Schemaabweichung

Wird ein Anpassungsbedarf erst zur Ausführungszeit des Prozesses erkannt und kann daher weder im Schemaentwurf noch durch spezifische Planadaption berücksichtigt werden, ist Flexibilität nur noch durch situative Abweichung des Ablaufs vom definierten Schema realisierbar. Die hierbei ergriffenen Struktur- oder Verhaltensmodifikationen erfolgen an der Prozessinstanz und führen zu keiner Änderung des Prozessschemas. Die Schemaabweichung entspricht dem Anwendungsfall der Deviation und ist für die Prozessplanung nicht relevant.

Die Planung von Verhaltens- und Strukturflexibilität erfolgt im Idealfall mehrstufig, indem zunächst ein robustes Prozessschema konstruiert

[132] Die späte Einbindung vorspezifizierter Prozessfragmente kann durch Parametrisierung realisiert werden, indem unterschiedliche Prozesskonfigurationen durch bestimmte Parameterwerte identifiziert und die zugehörigen Fragmente integriert werden. Vgl. hierzu auch [Geih08, 135].

und so detailliert wie möglich ausspezifiziert wird. Anschließend werden verhaltens- oder strukturbezogene Flexibilitätsbedarfe ermittelt und die betroffenen Prozesskomponenten identifiziert. Hierzu können Erfahrungen mit Modifikationen und Schemaabweichungen ausgeführter Prozesse im Sinne eines Lernprozesses beitragen. Schließlich werden geeignete Flexibilisierungsoptionen gewählt und angewandt [Kann10, 56]. Strukturflexibilität bietet größeres Anpassungspotenzial als Verhaltensflexibilität [BBFS11, 2].

5.1.5.2 Kontextabhängige Prozessgestaltung

Prozesse, die ihr konkretes Verhalten an situative Gegebenheiten anpassen, heißen *kontextsensitiv* [WaFe10]. Mit geplanter Verhaltensflexibilität ausgestattete Prozesse enthalten etwa an definierten Stellen Kontextparameter, die mit aktuellen Kontextfaktoren instanziiert werden und festlegen, welcher Ablaufpfad zu wählen oder wie eine Aktivitätsspezifikation zu modifizieren ist (vgl. [HaBR08, 55]). Mit Strukturflexibilität geplante Prozesse können mithilfe der Parameterwerte geeignete Prozessfragmente identifizieren und in das Schema einbinden. Die Flexibilität kontextsensitiver Prozesse kann weiter erhöht werden, indem die Werte der Kontextparameter mithilfe von Kontextregeln bestimmt werden (vgl. Abschnitt 4.7.2). Kontextsensitive Prozesse sind im Rahmen geplanter Flexibilitätspotenziale selbst-adaptiv [Geih08, 135].[133]

Die Berücksichtigung der jeweils bekannten Rahmenbedingungen, denen das Prozesssystem situativ unterliegt, ist bei allen Aufgaben der Prozessgestaltung hilfreich. Sie wird als ***kontextabhängige Prozessgestaltung*** bezeichnet. Durch die kontextbasierte Aussonderung aktuell nicht zulässiger Optionen lässt sich die Komplexität des Gestaltungsraums reduzieren und die Effizienz der Planung steigern. Kontext-

[133] Selbst-Adaptivität setzt definierte Variationspunkte und geeignete Anpassungsregeln voraus [Geih08, 135]. Diese Restriktion kann nur durch vollautomatische Erkennung und Behandlung von Anpassungsbedarf umgangen werden. DELLAROCAS & KLEIN [DeKl00] präsentieren ein entsprechendes wissensbasiertes System, das ohne Flexibilität geplante Workflows mit einer Taxonomie bekannter Ablaufmuster vergleicht, denen bekannte Ausnahmen und geeignete Behandlungsstrategien zugeordnet sind.

faktoren machen den kreativen Gestaltungsvorgang, der größtenteils auf dem Verständnis und der Erfahrung des Analytikers beruht, besser nachvollziehbar, indem sie Teile der nicht-funktionalen Gestaltungsentscheidung parametrisieren und in eine funktionale Abbildung überführen (vgl. [WaFe10, 117-119]). Die explizite Formulierung solcher andernfalls nur implizit berücksichtigter Bedingungen kann eine präzisere Anpassung des Prozesses an die Anforderungen der Problemsituation herstellen, die eine höhere Prozesseffektivität erwarten lässt.

Außer durch Deskriptoren und Regeln kann die kontextabhängige Prozessgestaltung auch mithilfe von klassischer Regelungstechnik, Künstlicher Intelligenz, Optimierungsverfahren [Geih08, 138] oder als Empfehlungssystem realisiert werden. Konkrete Anwendungsmöglichkeiten werden im weiteren Verlauf der Arbeit im jeweiligen Zusammenhang genannt.

5.2 Entwurf einer Planungsstrategie

Vor der Diskussion konkreter Planungsansätze erfolgt zunächst die Herleitung einer Planungsstrategie. Hierbei wird die in Abschnitt 3.2.4 erhobene Forderung nach einer Methodik, die auf die modulare Strukturierung des zu planenden Prozesssystems abgestimmt ist, berücksichtigt.

Die Durchführung der Prozessplanungsaufgabe als monolithischer Entscheidungsvorgang, der alle Einflussgrößen simultan berücksichtigt, scheitert an ihrer Komplexität. Die „semantische Lücke“ zwischen dem Sachproblem und den bereitstehenden Datentransformationsoperatoren lässt sich in einem Zug nicht effektiv überwinden. Daher erscheint die Zerlegung der Planungsaufgabe in interdependente Teilentscheidungen, die stufenweise nacheinander bewältigt werden, vorteilhaft.[134] Als Leitidee hierfür dient das *Systems Engineering* [Daen88], das einen allgemeinen Ansatz zur zielgerichteten Gestaltung komplexer Systeme skizziert. Seine Grundprinzipien sind die explizite Problemabgrenzung und Zielformulierung, die Gliederung des betrachteten Systems in

[134] Vgl. hierzu auch die Überlegungen bei [FeMa95a, 5-7] aus dem Kontext der Produktionslenkung.

überschaubare Sub- und Teilsysteme sowie das darauf abgestimmte Vorgehen vom Groben zum Detail. Hierbei werden zunächst alle relevanten Eigenschaften erfasst, die zur Beschreibung des Systems auf einer Betrachtungsstufe notwendig sind. Ist die Einbettung einzelner Komponenten in die Systemstruktur verstanden, können sie auf der nächsten Detaillierungsstufe konkretisiert und weitgehend isoliert behandelt werden. Für jede Stufe wird ein möglichst umfassender Überblick über zulässige Lösungsoptionen angestrebt und anhand der zu erwartenden Wirkung eine Alternative ausgewählt, die auf der nächsten Ebene wiederum eine detailliertere Strukturierung in Komponenten erfährt [Daen88, 27-29]. Die Zerlegung kann solange fortgeführt werden, wie es zweckmäßig erscheint [Daen88, 16].

Die Gliederung in Sub- und Teilsysteme korrespondiert mit der Ebenen- und Sichtenbildung der Datenanalysearchitektur (vgl. Abschnitt 4.2). Ihr hierarchischer Aufbau legt eine verrichtungsorientierte Zerlegung der Planungsaufgabe in mehrere Teilplanungsprobleme unterschiedlichen Ranges nahe. Die Planungsergebnisse einer Ebene *i* setzen dabei die Rahmenbedingungen für die darunter liegende Ebene *i+1* [Schn92, 76]. Die Planung startet auf Anwendungsebene mit der Spezifikation des Sachproblems, gefolgt von der Ableitung von Analyseproblemen auf Analyseebene. Für jedes Analyseproblem wird auf Prozessebene eine Prozessbeschreibung aus Außensicht (Aufgabensicht) entwickelt und um die Innensicht ergänzt (Aktivitätssicht).[135] Dies geschieht unter Nutzung der verfügbaren Operatoren der Ressourcenebene.

Die *Hierarchisierung* kann orthogonal mit *Dekomposition* kombiniert werden, welche eine objektorientierte Zerlegung der Planungsaufgabe vornimmt. Dekomposition ist sinnvoll, wenn ein Problem in mehrere Teilprobleme zerfällt, die zunächst unabhängig voneinander lösbar sind. Abhängigkeiten zwischen Teilproblemen können nachträglich berücksichtigt werden [Schn92, 61f.]. Auf Anwendungsebene führt dieses Prinzip zur Ausgrenzung einzelner Problemaspekte aus einem Sachpro-

[135] Die Mehrstufenplanung mit nicht ausführbaren Prozesselementen, wie sie aus der Differenzierung zwischen Aufgaben und Aktivitäten resultiert, wird in der Künstlichen Intelligenz erfolgreich zur Komplexitätsreduktion durch Partitionierung des Suchraums der möglichen Zerlegungsprodukte eingesetzt [Hert89, 66].

blem, die in separaten Teilprojekten weiter behandelt werden können. Auf Analyseebene lassen sich Untersuchungen häufig in eine Menge verketteter Analyseprobleme zerlegen. Jeder Prozess kann gemäß der generischen Phasen Datenvorbereitung, Datenanalyse und Ergebnisaufbereitung (vgl. Abschnitt 2.3.2.3) in Teilprozesse gegliedert werden. Die Teilplanungsprobleme werden – abhängig von ihrer Stellung in der Datenanalysearchitektur – als Schritte in der Planungsmethodik berücksichtigt (Prozessebene) oder als eigenständige Planungsgegenstände aufgefasst, auf die jeweils ein spezifischer Planungsprozess anzuwenden ist (Anwendungs- und Analyseebene).

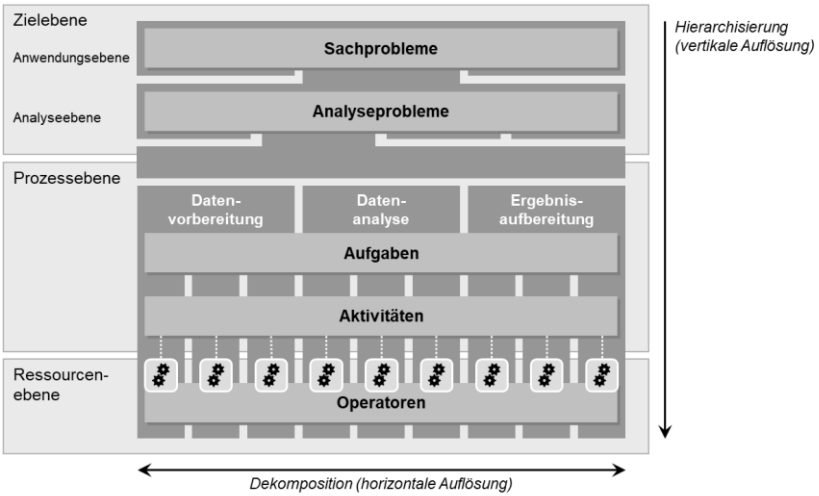


Abbildung 62: Prinzipien der Strukturierung von Prozessplänen in der Analysearchitektur (eigene Darstellung)

Abbildung 62 zeigt die erläuterten Grundprinzipien der Stufenplanung im Kontext der Datenanalysearchitektur. Sie korrespondieren mit der horizontalen und vertikalen Auflösung von Prozessen (vgl. Abschnitt 2.3.1.2). Anwendungs- und Analyseebene werden zur einfacheren Referenz gemeinsam als ***Zielebene*** bezeichnet.[136]

[136] Die Bezeichnung ist der Zielebene des Prozessbegriffs (Abschnitt 2.3.1.4) entlehnt.

5.3 Basisansätze der Analyseprozessplanung

Im Rahmen der skizzierten Planungsstrategie auf Basis einer hierarchischen und objektorientierten Zerlegung der Planungsaufgabe sind verschiedene Basisansätze zur Prozessplanung einsetzbar, die im Folgenden einzeln vorgestellt werden. Hierbei werden einerseits die beiden Anwendungsfälle der Neuplanung und Planadaption aus Abschnitt 5.1.4, andererseits Beiträge aus der Literatur sowie die gängige Analysepraxis berücksichtigt. Diese besteht typischerweise in der Komposition verfügbarer Operatoren zu Workflows auf Aktivitätsebene und verzichtet auf die Zerlegung einer Analyseaufgabe und deren Herleitung aus einem Analyseproblem.

Komplettansätze zur Analyseprozessplanung, die von der Problemspezifikation bis zur Operatorkomposition reichen, sind kaum veröffentlicht. Die wenigen Vorschläge behandeln Data-Mining-Analysen und lassen sich auf eine gemeinsame Wurzel zurückführen: WIRTH ET AL. [WSG+97] entwickeln im CITRUS-Projekt ein KDD-Werkzeug, das eine integrierte Daten- und Prozessverwaltung, ein Prozessausführungssystem und ein Assistenzmodul umfasst. Letzteres führt die „klassische" Operatorkomposition mit der Prozesskonstruktion durch hierarchische Aufgabenzerlegung zusammen und bietet Unterstützung bei der Wiederverwendung früherer Prozesse oder Prozessschablonen. Hierbei wird die zunächst auf die Analysephase beschränkte Idee der hierarchischen Zerlegung von Analyseaufgaben von WIRTH & REINARTZ [WiRe96] aufgegriffen, die später von ENGELS ET AL. [Enge96], [EnLS97], [EnLS97b], [Enge99], [Lind05] zu einem Nutzerführungssystem weiterentwickelt wird.[137] Die Idee von CITRUS erfährt innerhalb des E-LICO-

[137] Einzelne Mitglieder der vorgestellten Projekte waren an der Konzipierung des CRISP-DM-Modells [CCK+00] beteiligt, das ebenfalls eine Spezialisierung generischer Aufgaben durch kontextabhängige Anpassung vorsieht. Die dort behandelten Aufgaben stellen jedoch keine Ausführungselemente eines Prozessschemas dar, sondern beschreiben Projektaufgaben, die bei der Planung und Durchführung einer Analyse zu behandeln sind. Sofern sie die Analysedurchführung betreffen, können die generischen Aufgaben aber als Ausgangspunkt für eine Aufgabenzerlegung herangezogen werden. Vgl. hierzu auch [KSBF10, 10].

Projekts auf Basis eines ontologiebasierten hierarchischen Planers eine Wiederbelebung [KSBF09], [KSBF10].

Ein eigener Vorschlag in Form eines für alle Datenanalyseansätze geeigneten Beschreibungsrahmens [Knob03a] vereint die Anwendungsfälle Neuplanung und Planadaption mit der Aufgabenzerlegung. Prozesskonstruktion soll unter Anwendung ingenieurwissenschaftlicher Prinzipien durch zielorientierte Verknüpfung verfügbarer Prozessbausteine bewältigt werden. Dabei können Fragmente der Zerlegungsstruktur auf beliebiger Aggregationsstufe als funktionale Einheiten betrachtet und in späteren Prozessen wiederverwendet werden. Der Ansatz sieht die Verwendung von Taxonomien vor, in denen die Prozessbausteine in Funktionsklassen organisiert sind und dem Analytiker zur Integration in das Prozessschema übersichtlich angeboten werden.

Die Diskussion offenbart vier Basisansätze der Analyseprozessplanung, die entlang zweier orthogonaler Dimensionen beschrieben werden (Abbildung 63).

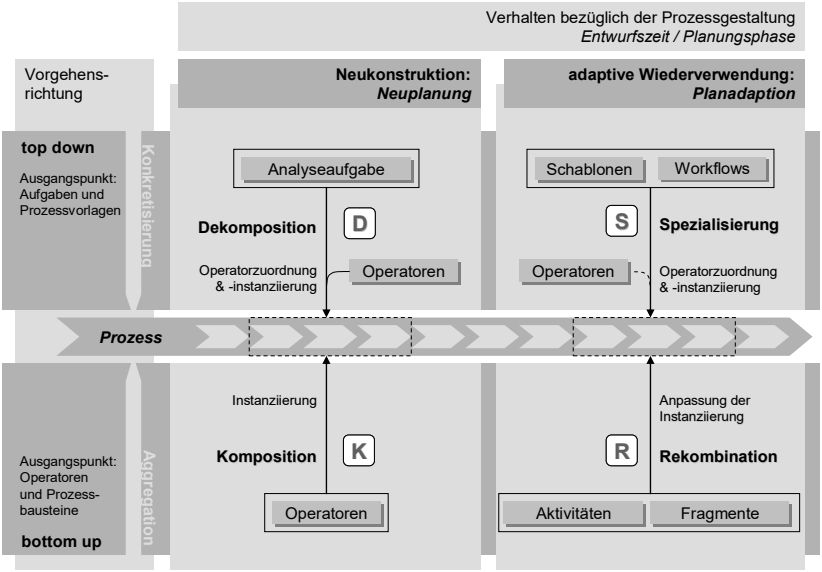


Abbildung 63: Basisansätze der Prozessplanung im Überblick (eigene Darstellung)

Die erste Dimension bilden Neukonstruktion (*Neuplanung*) und adaptive Wiederverwendung (*Planadaption*) als planungsbezogene Anwendungsfälle der Prozessgestaltung. Die zweite Dimension wird durch die Vorgehensrichtung innerhalb der Analysearchitektur aufgespannt. Die absteigende Richtung (*top down*) entspricht einer Konkretisierung, die durch Dekomposition von Aufgaben oder Spezialisierung von Prozessvorlagen erreicht wird. Die aufsteigende Richtung (*bottom up*) beschreibt eine Aggregation von Operatoren bzw. ausführbaren Prozessbausteinen.[138]

Resultat der erfolgreichen Anwendung aller vier Ansätze ist jeweils ein Prozess(ausschnitt) in Form eines Workflows aus Aktivitäten. Sie sind im Einzelnen wie folgt charakterisiert:

- **Dekomposition (D)** konkretisiert die globale Analyseaufgabe durch mehrstufige Zerlegung in Teilaufgaben. Ist ein den jeweiligen Aufgabenzielen entsprechender Operator verfügbar, kann dieser der Teilaufgabe zugeordnet und instanziiert werden (Top-down-Neuplanung).
- **Komposition (K)** aggregiert generische Operatoren nach adäquater Instanziierung in Gestalt zielorientierter Aktivitäten zu einem ausführbaren Workflow (Bottom-up-Neuplanung).
- **Spezialisierung (S)** konkretisiert vorgegebene Prozessvorlagen (Schablonen aus Aufgaben, Workflows aus Aktivitäten) durch Adaption an die aktuelle Problemsituation. Im Falle von Schablonen ist zusätzlich eine Operatorzuordnung und -instanziierung erforderlich (Top-down-Wiederverwendung).
- **Rekombination (R)** aggregiert Prozessfragmente oder Aktivitäten aus der Fallbibliothek nach geeigneter Anpassung zu einem neuen Workflow (Bottom-up-Wiederverwendung).

[138] Die Generalisierung von Prozessbausteinen (Pendant der Aggregation) ist bei der Planung nicht relevant. Sie wird im Rahmen der Revision bei der Herleitung von Prozessmodulen behandelt (Abschnitt 7.4.2.3).

Die Basisansätze können sich wirksam ergänzen. Aus ihrer Kombination entsteht gewissermaßen ein „Baukastensystem“ für Datenanalyseprozesse, das Prozessbausteine unterschiedlicher Art und Granularität enthält. Geeignete Bausteine können ausgewählt, instanziiert bzw. angepasst und in den Prozess integriert werden. Die Modularisierung des Prozesssystems unterstützt die Komplexitätsbewältigung, indem die Bausteine zunächst als Black-Boxes in die Prozessstruktur eingefügt und später bezüglich ihrer inneren Struktur konkretisiert werden [Knob03a, 350f.].

In den folgenden Abschnitten werden die einzelnen Basisansätze näher erläutert. Hierbei beschränkt sich die Betrachtung zunächst auf die Prozessebene und geht von einer gegebenen Analyseaufgabe aus. Deren Herleitung aus einem Sachproblem wird später diskutiert.

5.3.1 Innovative Ablaufgestaltung durch Neuplanung

Die Konstruktion neuer Pläne durch Operatorkomposition sowie Aufgabendekomposition erläutern die folgenden beiden Abschnitte.

5.3.1.1 Operatorkomposition (Bottom-up-Neuplanung)

Die Prozesskonstruktion durch Operatorkomposition entspricht der gängigen Praxis. Datenanalysewerkzeuge bieten hierfür in der Regel grafische Benutzeroberflächen an, auf denen die Prozesse als Graphen modelliert werden können [Gros09b, 6]. Der Analytiker wählt Operatoren aus einer Palette oder einer einfachen Taxonomie aus, zieht sie als Symbolbild auf die Arbeitsfläche und verknüpft die resultierenden Aktivitätsknoten über Datenabhängigkeitsflüsse zu einem Workflow. Ihre Instanziierung erfolgt durch Einstellung der Modusparameter in Dialogfenstern. Diese Realisierungsform geht auf das Werkzeug CLEMENTINE[139] zurück [Enge99, 73].

[139] CLEMENTINE wurde von der Firma Integral Solutions Ltd. entwickelt, die später von SPSS Inc. übernommen wurde. Nach Akquisition von SPSS durch IBM Corp. wird das Werkzeug heute unter dem Namen IBM SPSS MODELER vertrieben.

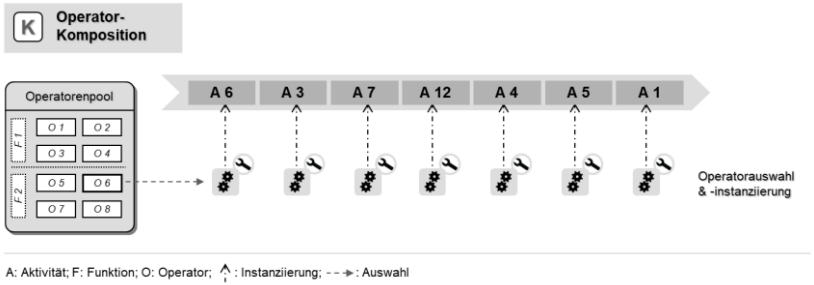


Abbildung 64: Basisansatz Operatorkomposition (K) (eigene Darstellung)

Abbildung 64 zeigt, wie *Operatoren einzeln ausgewählt, instanziiert und als Aktivitäten in einen Prozesszusammenhang gestellt werden*. Die Prozessspezifikation erfolgt somit auf Aufgabenträgerebene. Die Bezeichnung der Aktivitäten ist zunächst nur aus den jeweiligen Operatoren ableitbar, worunter die Nachvollziehbarkeit der Komposition leiden kann. Eine zulässige Reihenfolge muss explizit bestimmt werden. Die Reichweite des Ansatzes ist auf die Aktivitätssicht der Prozessebene beschränkt.

Angesichts von Analysewerkzeugen mit oft Hunderten von Operatoren steht der Analytiker häufig vor einer verwirrenden Zahl von Gestaltungsoptionen [KSBF10, 2]. Damit steigt gleichzeitig das Risiko nicht zielführender Kompositionen, weil die Dokumentation der Algorithmen die Prüfung ihrer Anwendungsvoraussetzungen oft nicht ausreichend unterstützt [Enge99, 75f.]. Eine systematische Prozessherleitung ist bei diesem Vorgehen nur über die Funktionszuordnung der Operatoren möglich, d.h. auf Grundlage der angestrebten Datentransformationen. Die Überbrückung der semantischen Lücke zur globalen Analyseaufgabe erfolgt weitestgehend implizit, da nur die Transformationen innerhalb der Analysephase direkt mit dem Analyseproblem assoziiert sind. Legt dieses beispielsweise die Berechnung von Entscheidungsregeln nahe, kann unmittelbar ein Regelinduktionsverfahren ausgewählt werden.

Die Operatorkomposition lässt sich bei geeigneter Repräsentation der Operatoren direkt als KI-Planungsproblem formulieren und lösen (vgl. [Hert89, 45ff.], [RuNo03, 382ff.]). Hierbei erfolgt die Verknüpfung im

einfachsten Fall allein aufgrund der Ein- und Ausgabedaten der Operatoren durch Rückwärtssuche innerhalb des durch Quelldatenschema (Start) und Ergebnistyp (Ziel) aufgespannten Zustandsraums. Um unsinnige Pläne zu vermeiden, muss die Operatorauswahl zusätzlich durch domänenspezifisches Wissen gelenkt werden [Hert89, 63]. LANSKY & PHILPOT [LaPh94] demonstrieren dies anhand eines Constraint-basierten Planers am Beispiel der Bilddatenanalyse. Zur Begrenzung des Suchraums verwenden sie Lokalisierungsstrategien, die sich an der Datenstruktur orientieren.

Den Inhalt der Daten betreffende Transformationen können jedoch nicht allein aus syntaktischen Diskrepanzen, z.B. zwischen Quelldatenschema und Eingabedaten des Analyseoperators, abgeleitet werden. So ist etwa aus den Datenschemata nicht ersichtlich, ob eine Transformation der Werteverteilung oder eine Stichprobenziehung angezeigt sind. Jüngere Ansätze beschreiben Operatoren in Ontologien, die neben Ein- und Ausgabedaten weitere Merkmale und Anwendungsrestriktionen abbilden und die genannten Einschränkungen zumindest teilweise umgehen können. ŽÁKOVÁ ET AL. [ZPZL09] verwenden Aufgaben- und Datentyp-Ontologien und sind damit in der Lage, auch komplexe oder partielle Datenobjekte zu verarbeiten sowie fortgeschrittene Daten-Restriktionen zu berücksichtigen. BERNSTEIN & PROVOST [BePr01] sowie DIAMANTINI ET AL. [DiPS09] berechnen alle zulässigen Prozesspläne und bieten dem Analytiker die Möglichkeit, diese anhand wählbarer Kriterien bewerten und sortieren zu lassen, um einen passenden Vorschlag zu selektieren.

Die Annotation der Operatoren dient in diesen Fällen zunächst der maschinellen Verarbeitbarkeit. Um auch die manuelle Prozesskonstruktion besser zu unterstützen, sind die semantischen Angaben auch dem Analytiker zugänglich zu machen, etwa indem die Operatoren übersichtlich in einer Funktionstaxonomie präsentiert werden [Knob03a, 343f.].

5.3.1.2 Aufgabendekomposition (Top-down-Neuplanung)

Die Prozessgestaltung durch *Aufgabenzerlegung* greift die Idee des Systems Engineering auf.[140] Die Aufgabenspezifikation erfolgt ausgehend vom Analyseproblem, aus dem unmittelbar die intensionale Definition der globalen Analyseaufgabe resultiert. Die Dekomposition repräsentiert eine extensionale Aufgabendefinition, d.h., eine Option zur Realisierung der zerlegten Aufgabe durch aufgedeckte Teilaufgaben [Berg81, 29]. Sie ist mehrstufig anwendbar, bis eine ausreichend konkrete Spezifikation vorliegt. Dieser Fall ist für jede Teilaufgabe spätestens dann erreicht, wenn ein Operator verfügbar ist, der das jeweilige Aufgabenziel realisiert.[141] Mit Zuordnung und Instanziierung eines Operators entsteht eine ausführbare Prozessaktivität. Dieser Ansatz ermöglicht die bis zum Zeitpunkt der Operatorzuordnung aufgabenträgerunabhängige Prozessspezifikation.

Abbildung 65 illustriert das Vorgehen: Die Wurzel des Aufgabenzerlegungsbaums repräsentiert die globale Analyseaufgabe (T 0), die im Beispiel in drei Teilaufgaben (T 1, T 2, T 3) gegliedert ist. Existiert ein geeigneter Operator, lässt sich eine Aktivität spezifizieren, der die Bezeichnung der Aufgabe übertragen werden kann, wodurch eine semantisch nachvollziehbare Prozessbeschreibung entsteht (z.B. T 2 → A 2). Andernfalls kann die Zerlegung weitergeführt werden (z.B. T 1 in T 1.2 und T 1.3). Die Dekomposition unterstützt zunächst nur die Aufgabenspezifikation. Unterstützung für die Zuordnung von Operatoren ist möglich, wenn diese nach Funktionen gruppiert sind und auf diese Weise eine semantische Verknüpfung zu den Aufgaben hergestellt wird (Operatortaxonomie). Zwar bedingt eine Aufgabenzerlegung zuweilen sachlogische Beziehungen zwischen den Teilaufgaben [Gait83, 58], die jedoch explizit zu spezifizieren sind. Die abschließende Bestimmung von Reihenfolgen und Datenabhängigkeiten zwischen

---

[140] Im Workflow-Management wird die Top-down-Ablaufgestaltung auf KOSIOLS Analyse-Synthese-Konzept der Betriebsorganisation zurückgeführt (vgl. [JaBS97, 8f.], [Reif03, 14]).

[141] Je nach Ziel der Dekomposition kann eine ausreichend konkrete Spezifikation auch unabhängig von der Verfügbarkeit eines Operators erreicht sein (geplante Strukturflexibilität).

Aktivitäten ist nicht allein aus der Aufgabendekomposition herleitbar. Die hier gezeigte Aufgabenzerlegung findet auf Prozessebene statt. Das Grundprinzip der Dekomposition eignet sich jedoch ebenso für die Zielebene (siehe Abschnitte 5.4.4.5 und 5.4.5.3).

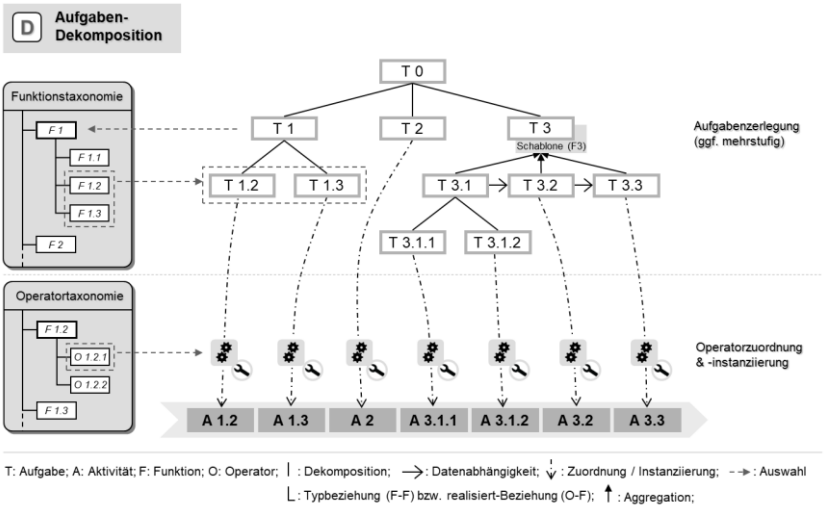


Abbildung 65: Basisansatz Aufgabendekomposition (D) (eigene Darstellung)

Als Zerlegungsprinzipien kommen insbesondere Verrichtungs- und Objektprinzip in Frage, die zur Konkretisierung von Aufgabenziel bzw. Aufgabenobjekt führen [Gait83, 17], [FeSi13, 237]. Die verrichtungsorientierte Zerlegung kann sich von Prozessmodellen leiten lassen. So entsteht etwa der initiale Zerlegungspfad Datenanalyse (global) → Datenvorbereitung → Datenmodifikation → Transformation → Codierung (vgl. Abschnitt 3.1.2.2). Die objektorientierte Zerlegung kann zur besseren Übersichtlichkeit zunächst auf den Eingabedatentyp beschränkt werden. Die Konkretisierung erfolgt durch Typisierung, Teilsystembildung oder Einbeziehung des Ausgabedatentyps. Die erste Option differenziert etwa zwischen relationaler oder dokumentenorientierter Repräsentation oder nach Datenobjekttypen, die zweite Option zerlegt z.B. eine Relation in Tupel oder Spalten (Attribute). Dadurch lässt sich obige verrichtungsorientierte Zerlegung bestätigen

und erweitern. So ist z.B. die Codierung eine attributbezogene Transformation[142] und kann für nominale und numerische Merkmale spezialisiert werden.

Eine widerspruchsfreie Deduktion von Teilaufgaben aus der intensionalen Aufgabendefinition ist indes nicht realistisch. Konfliktäre Aufgabenziele, nicht disjunkte Zerlegungen [FeSi13, 237], Nebeneffekte aufgrund nicht erkannter Interdependenzen zwischen Teilaufgaben sowie nicht durchgängig anwendbare Zerlegungsprinzipien haben zur Folge, dass die Aufgabenzerlegung weniger deduktiv als vielmehr im Rahmen von Selektions- und Bewertungsentscheidungen erfolgt. Diese können auch suboptimal oder fehlerhaft ausfallen. Die Aufgabendekomposition lässt sich daher als eigenes Konstruktionsproblem interpretieren, dessen Lösung in differenzierten Zerlegungshierarchien münden kann [Gait83, 55-58].

Unterstützung bietet z.B. die Künstliche Intelligenz in Gestalt hierarchischer Mehrstufenplaner mit Operatorabstraktion. Diese Hierarchical Task Network (HTN) Planner halten für aus Außensicht spezifizierte Aufgaben („abstrakte Operatoren“) jeweils mehrere Zerlegungsoptionen bereit [Hert89, 149-152], [RuNo03, 422f.] und gestatten auf diese Weise die Einbeziehung heuristischen Wissens über erprobte Zerlegungsstrukturen [Hert89, 162]. Einen solchen Planer verwenden z.B. CHIEN ET AL. [CFM+97] für die Analyse wissenschaftlicher Bilddaten.

Um derartiges Wissen auch für die manuelle Aufgabendekomposition zugänglich zu machen, eignet sich eine Funktionstaxonomie [Knob03a, 348], die komplexe Zerlegungshierarchien abbilden kann. Abbildung 65 zeigt ihren Einsatz am Beispiel der Aufgabe T 1. In der Taxonomie wird die zu T 1 gehörige Funktion F 1 ausgewählt, der auf der nächsten Konkretisierungsstufe drei speziellere Funktionen zugeordnet sind. Nun ist zu entscheiden, welche dieser Funktionen im aktuellen Kontext relevant sind. Im Beispiel werden auf Grundlage fallspezifischer Erwägungen die Funktionen F 1.2 und F 1.3 gewählt und als Teilaufgaben T 1.2 und T 1.3 in die Dekomposition übernommen. Die Auswahl kann durch Kontextfaktoren geleitet werden. Hieraus wird deutlich, dass die

[142] Demgegenüber operiert z.B. Denormalisierung auf Schemaebene.

Aufgabendekomposition primär der sukzessiven Einschränkung der Konkretisierungsoptionen dient (Komplexitätsreduktion). Unter der Prämisse, dass die Taxonomie etabliertes Analysewissen enthält, erlaubt sie eine eingeschränkte Vollständigkeitsprüfung bezüglich der berücksichtigten Teilaufgaben[143] und kann somit zur Qualitätsverbesserung der Prozesse beitragen. Die Bereitstellung einer Funktionstaxonomie verlagert Teile des Dekompositionsproblems auf den Zeitpunkt der Taxonomiekonstruktion. Hinweise zur methodisch gestützten Erstellung solcher Taxonomien diskutiert Kapitel 7.4.2.3.

Vollständige Zerlegungen für einzelne Aufgaben, die wie bei HTN-Planern alle relevanten Teilaufgaben (sowie idealerweise deren Verknüpfung) beschreiben und direkt in die Dekomposition übernommen werden können, lassen sich in Bibliotheken mit vordefinierten Prozessvorlagen bereitstellen. Da sie eine Form der Wiederverwendung darstellen, werden sie in Abschnitt 5.3.2.2 erläutert. In Abbildung 65 ist anhand einer Prozessschablone für Aufgabe T 3 angedeutet, wie solche Vorlagen die Dekomposition vereinfachen können. Eine Kombination beider Top-down-Ansätze ist daher empfehlenswert.

ZHONG ET AL. [ZLKO97] beschreiben einen automatischen Planer für KDD-Prozesse, der den Basisansatz implementiert. Ausgehend von einer abstrakten Initialaufgabe wird das iterative Vorgehen des Analytikers nachgebildet und solange fortgesetzt, bis ein ausführbarer Workflow vorliegt. Hierbei wird jede abstrakte Aufgabe der Dekomposition mithilfe eines nicht-linearen Mechanismus in einen Teilplan expandiert, indem anstelle des kompletten Operatorpools nur eine in jeder Aufgabe hinterlegte Liste zulässiger Teilaufgaben durchsucht wird, um geeignete Zerlegungsoptionen zu finden. Operatoren werden analog wie Aufgaben behandelt.

Die systematische Aufgabenzerlegung für die Analyseprozessplanung wurde erstmals von WIRTH & REINARTZ [WiRe96] auf die Analysephase

[143] Die Vollständigkeitsprüfung ist eingeschränkt in dem Sinne, dass keine objektiv korrekte und vollständige Aufgabenhierarchie deduzierbar ist. Die Vollständigkeitsprüfung erfolgt demnach bezüglich der in der Taxonomie abgebildeten Menge von Konkretisierungsoptionen.

von KDD-Prozessen angewandt. Die in ihrem Beispiel zerlegten Aufgaben sind als Sachprobleme anzusehen und werden bis auf Aktivitätsebene konkretisiert. ENGELS ET AL. [Enge96], [EnLS97] überführen diese Idee in ein Konzept, das Aufgaben primär durch Wiederverwendung früherer Zerlegungsstrukturen verfeinert. Alternativ können Aufgabensequenzen mithilfe eines partiellen hierarchischen Planers konstruiert werden. Die Arbeiten sind Teil des CITRUS-Projekts [WSG+97], in dem ein interaktiver Ansatz verfolgt wird, bei dem der Analytiker den Planungsvorgang lenkt und vom Assistenzsystem Vorschläge zur Plankonkretisierung erhält.[144] Aufgabenzerlegung nutzen auch AMANT & COHEN [AmCo98], [AmCo98b] mit dem speziell auf EDA ausgerichteten System AIDE, das skriptbasiert arbeitet und sich auf die Analysephase beschränkt.

Dass die Lenkung des Zerlegungsvorgangs allein anhand der Ein- und Ausgabedatentypen problematisch sein kann, zeigt sich bei [EnLS97], als die Autoren explizit Rekursion in der Aufgabenzerlegung zulassen und Analyseaufgaben höchster Ebene als eigenständige Prozesse teils in die Datenvorbereitungs-, teils in die Analysephase eines allgemeinen Prozessschemas einordnen und damit den Prozesszusammenhang auftrennen [EnLS97, 4f.]. Dieses Vorgehen mag zulässige und zielkonforme Abläufe hervorbringen; die Gestaltungsentscheidungen sind indessen nicht nachvollziehbar. Die Generierung der Prozessaktivitäten aus den ursprünglichen Aufgaben ist daraus nicht ersichtlich (fehlende Prozessstrukturtransparenz).

Diese Erkenntnis legt die explizite Einbeziehung der Aufgabenziele in den Planungsprozess nahe, um auch semantische Funktionsbe-

[144] Die konkrete Ausgestaltung des Konzepts weist einige Beschränkungen auf. So bilden die Blattknoten der Aufgabendekomposition vordefinierte „Simple Tasks“, die Verfahrensklassen repräsentieren [Enge99, 52]. Ihr niedriger Detaillierungsgrad bedingt, dass sie zur Zuordnung konkreter Verfahren durch Einschränkung von Ein- und Ausgabedatentypen weiter konkretisiert werden müssen [Enge99, 122] und somit keine durchgängige Zerlegung erfolgt. Das präsentierte Anwendungsbeispiel [Enge99, 142ff.] zeigt einen Planungsvorgang in Form einer Rückwärtssuche im Raum der Simple Tasks, die nachträglich KDD-Prozessphasen zugeordnet werden. Dies ist allenfalls als einstufige Zerlegung der globalen Analyseaufgabe interpretierbar.

schreibungen berücksichtigen zu können. Während viele ontologiegestützte Planungsansätze dort nur syntaktische Operatorbeschreibungen ablegen (vgl. Abschnitt 4.7.1.5), stellen KIETZ ET AL. [KSBF09] einen automatischen HTN-Planer vor, der den Operatoren auch Ziele zuordnet, zu deren Erreichung sie beitragen. Die im Rahmen des E-LICO-Projekts durchgeführte Arbeit umfasst einen auf Basis des Open-Source-Werkzeugs RAPIDMINER entwickelten Prototyp, der später zu einem interaktiven System ausgebaut wird, das auch Wiederverwendung unterstützt. Die Ontologie ist auch dem Anwender zugänglich, der das Werkzeug dadurch als Assistenzsystems nutzen kann [KSBF10].

5.3.2 Adaptive Ablaufgestaltung durch Wiederverwendung

Die Wiederverwendung bewährter Ergebnisse früherer Planung zielt im Allgemeinen auf Zeit-, Kosten- und Qualitätsvorteile gegenüber der Neuentwicklung [MiMM95, 528f.], [Reza95, 221f.]. Voraussetzung für die Erreichung dieser Ziele sind Bibliotheken zur Verwaltung der Artefakte (vgl. die Bibliothekssichten in Kapitel 4) [Reza95, 222] und ein Wiederverwendungsmanagement, das die Ermittlung und Verwendung geeigneter Artefakte methodisch lenkt und entsprechende Verantwortlichkeiten definiert [MiMM95, 533-535].

Aufgrund zu erwartender Diskrepanzen zwischen aktueller und früherer Problemsituation, die sich durch Vergleich der Kontextfaktoren des aktuellen Falls mit jenen der Wiederverwendungsobjekte erkennen lassen, sind in aller Regel Anpassungen an den Vorlagen notwendig [MiMM95, 548]. Die unmodifizierte Wiederverwendung von Prozessschemata (Anwendungsfall Repetition) bedarf keiner weiteren Planung und wird im Rahmen der Steuerung behandelt. Die Literatur zur Software- und Modellierungstechnik enthält verschiedene Klassifizierungen für Methoden der Wiederverwendung.[145] Die wiederholte

[145] So unterscheidet die gängige Einteilung in kompositorische („Baustein-Ansatz“) und generative Wiederverwendung (vgl. [MiMM95, 528], [JaVW00, 8]) nach der Art des Wissens, lässt vollständige Schemata nach strenger Lesart aber außen vor. FETTKE & LOOS [FeLo02] charakterisieren Methoden anhand eines umfangreichen Kriterienkatalogs, der u.a. Wiederauffindung, Anpassung, Modellbegriff und Umfang berücksichtigt. Bezüglich der Anpassung ist eine Gliederung in generierende (Konfiguration)

Nutzung von Prozessbausteinen im Sinne von Strukturmustern heißt *kompositorische Wiederverwendung* [JaVW00, 8], die Wiederverwendung des Vorgehens beim Prozessentwurf ist als *generative Wiederverwendung* bekannt und korrespondiert mit der Nutzung von Entwurfsmustern.[146] Die antizipierte Anpassung von Vorlagen an definierten Stellen heißt *Konfiguration* (geplante Verhaltens- und Strukturflexibilität, vgl. Abschnitt 5.1.5.1) [BHB+10, 9], [Drei+05, 692], [BeDK04, 252], die nicht-antizipierte Änderung wird allgemein als *Modifikation* bezeichnet [MiMM95, 555f.]. Im Folgenden wird die Bausteinrekombination als kompositorische Modifikation von der Vorlagenspezialisierung abgegrenzt. Letztere subsumiert antizipierte Anpassungen (Konfiguration) von Prozessvorlagen.

5.3.2.1 Bausteinrekombination (Bottom-up-Wiederverwendung)

Einige Analysewerkzeuge erlauben die Speicherung definierter Workflow-Ausschnitte als Einheit (vgl. Abschnitt 4.5.4.2 sowie [Enge99, 75]). Damit ist die Grundlage für die Wiederverwendung ausführbarer Prozessbausteine im Rahmen neuer Untersuchungen gelegt. Sie geschieht durch Rekombination, d.h., durch *Aggregation einer Menge konkreter Strukturmuster* zu einem neuen Workflow. Abbildung 66 zeigt die Grundidee dieses Basisansatzes. Im Beispiel werden ein Fragment Fr 3.1.3, bestehend aus drei über Datenabhängigkeiten verknüpften Aktivitäten, sowie eine einzelne Aktivität A 5.1.1 aus einer Bibliothek abgerufen, und jeweils nach Anpassung ihrer Parameter in das zu erstellende Prozessschema übernommen. Die modulinterne Struktur kann bei Bedarf beliebig „umgebaut“ werden, etwa durch Austausch von Aktivitäten oder Flussbeziehungen. Die Wiederverwendung einzelner Aktivitäten ist hilfreich, um bewährte Parametereinstellungen aus

---

und nicht-generierende Adaption verbreitet, die auf Spielräume bei der Modifikation fokussiert [BeDK04, 252].

[146] In der Softwaretechnik wird generative Wiederverwendung traditionell mit Anwendungsgeneratoren assoziiert, die auf Grundlage (semi-) formaler Spezifikationen automatisch Programmcode erzeugen [MiMM95, 528], [JaVW00, 9]. Die Automatisierung der Planung unterliegt zahlreichen Beschränkungen und wird in der vorliegenden Arbeit nicht der Wiederverwendung zugerechnet.

früheren Projekten unkompliziert zu übernehmen. In allen Fällen ist das resultierende Schema nicht auf ein einzelnes Ausgangsmodell rückführbar.

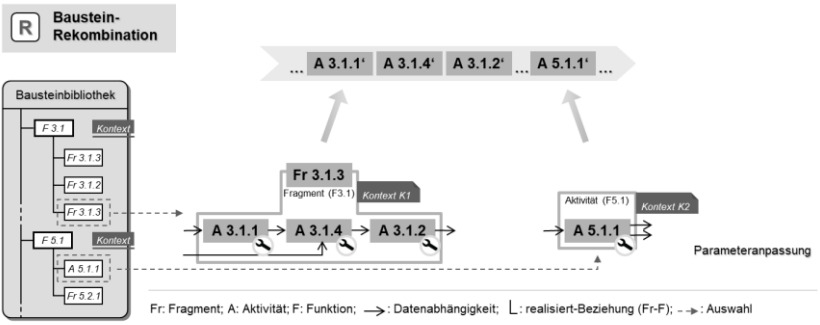


Abbildung 66: Basisansatz Bausteinrekombination (R) (eigene Darstellung)

Rekombination unterstützt die Operatorzuordnung und im Falle von Fragmenten auch die Reihenfolgebestimmung. Bei der Verknüpfung der Bausteine sind analog zur Operatorkomposition (Abschnitt 5.3.1.1) die Ein- und Ausgabedaten zu beachten, die durch andere Bausteine bereitzustellen sind bzw. nach außen exportiert werden. Beispielsweise ist Fragment Fr 3.1.3 mit zwei eingehenden Flüssen zu versorgen und erzeugt eine ausgehende Datenabhängigkeit, Aktivität A 5.1.1 konsumiert einen und produziert zwei Flüsse. Stimmen die Schnittstellen zu verknüpfender Bausteine nicht hinsichtlich Anzahl und Typbindung überein, sind transformierende Aktivitäten zwischenzuschalten oder weitere strukturelle Modifikationen durchzuführen. Die bausteinübergreifende Reihenfolge ist also separat zu bestimmen.

Die Aufgabenspezifikation wird von der Rekombination insoweit unterstützt, wie die Bausteine bei ihrer Spezifikation explizit mit zugehörigen Funktionen annotiert wurden (vgl. Abschnitt 5.3.1.1). In Abbildung 66 ist Fragment Fr 3.1.3 als Option zur Realisierung der Funktion F 3.1 gekennzeichnet, Aktivität A 5.1.1 realisiert Funktion F 5.1. Auf diese Weise entsteht eine Bausteinbibliothek mit funktionalen Äquivalenzklassen (vgl. [MiMM95, 548]). Die Wahl zwischen mehreren Alternativen innerhalb einer solchen Klasse kann durch Kontextfaktoren unter-

stützt werden [RuPR99, 232]. Dazu sind die Bausteine jeweils mit dem Kontext zu annotieren, innerhalb dessen sie anwendbar sind (vgl. Abschnitt 4.5.4.1). Anfragen an die Bibliothek sind ebenso kontextbasiert zu formulieren.

Die Rekombination beschreibt eine kompositorische Wiederverwendung, welche die Artefakte im Rahmen einer nicht-antizipierten strukturellen Modifikation den Anforderungen der aktuellen Situation anpasst. Die Beschränkung auf nicht-antizipierte Modifikationen folgt aus der Betrachtung ausführbarer Prozessbausteine (Fragmente, Aktivitäten), die keine abstrakten Elemente enthalten. Die Parameteranpassung betrifft nicht die Strukturmuster, sondern die geplante Verhaltensflexibilität der Operatoren und wird für alle Formen der Wiederverwendung als nützlich vorausgesetzt. Die erforderlichen Modifikationen können anhand der Unterschiede zwischen den aktuellen Kontextbedingungen und den Kontextfaktoren der Artefakte ermittelt werden [RuPR99, 232]. Ihre Anwendung führt zu neuen Varianten der Ausgangsstruktur, die unabhängig von dieser existieren [BHB+10, 9]. Die strukturellen Änderungen lassen sich durch die Basisoperationen Hinzufügen und Entfernen von Aktivitäten und Flussbeziehungen ausdrücken [ReDa98, 1] und sind prinzipiell nicht eingeschränkt (vgl. freie bzw. analogiebasierte Anpassung als nicht-generierende Adaptionsmechanismen [BeDK04, 252]).

Mit Ausnahme der eher rudimentären Unterstützung durch Analysewerkzeuge sind keine Arbeiten bekannt, welche die systematische Rekombination konkreter Prozessbausteine behandeln. Durch seine Beschränkung auf die Datenvorbereitungsphase im KDD zielt das MININGMART-Projekt [KiZV00], [MoSE03] zwar auf die Wiederverwendung existierender Prozessfragmente, die auf Basis eines Case-based-Reasoning-Systems mit teilautomatisierter Fallanpassung erreicht wird. Während die Operatoren der Aktivitäten beibehalten werden, erfahren die Informationsobjekttypen vor der Speicherung in der Fallbibliothek jedoch eine Abstraktion auf die konzeptuelle Ebene von Domänenkonzepten und müssen vor Anwendung auf einen neuen Fall auf konkrete Datenobjekttypen der aktuellen Datenbasis abgebildet werden. Die Adaption wird durch Multi-Strategy Learning unterstützt,

um fallspezifische Operatorparameter zu ermitteln. Erfolgreiche Prozesse können in ein Web-basiertes Repository eingestellt und mit anderen Anwendern geteilt werden. Auch der Vorschlag von ENGELS [Enge99] nutzt zwar zum Teil Prozessfragmente auf Aktivitätsebene, die jedoch nicht unabhängig existieren, sondern stets Bestandteil abstrakter Prozessschablonen (Aufgabenzerlegungen) sind [Enge99, 95, 123].

Die Nutzung von Strukturmustern ist auch bei der Planung auf Zielebene möglich. Hier wird nicht die Lösung, sondern die Problemspezifikation in Gestalt von Problemkarten oder Analyseketten bzw. deren Elemente (Problemaspekte oder Analyseziele) wiederverwendet. Bei häufig in ähnlicher Form wiederkehrenden Problemsituationen hilft dies, Aufwand der Erfassung und Beschreibung zu sparen (vgl. [MiMM95, 531]) und konsistentes Handeln zu gewährleisten.

Bei unbedachter Wiederverwendung von Prozessfragmenten wird im ungünstigsten Fall eine „fertige Lösung für ein dem Nutzer unbekanntes Problem“ auf ein neues Problem übertragen [HaSW98a, 26]. Bei Kenntnis des alten Problems, wie sie durch Annotation der Bausteine mit Funktionen und Kontextfaktoren erreichbar ist, wird die festgestellte Diskrepanz zum neuen Problem durch analoge Anpassung der Lösung zu kompensieren versucht. Derart experimentelles Vorgehen kann jedoch weder die Effektivität der erzeugten Lösung noch die Erreichung der angestrebten Qualitäts- und Produktivitätsvorteile gewährleisten [MiMM95, 536, 554f.]. Daher erscheint ein Verzicht auf Strukturmodifikationen eines Fragments ratsam (vgl. „Black-Box-Wiederverwendung“ [JaVW00, 8f.]).

5.3.2.2 Vorlagenspezialisierung (Top-down-Wiederverwendung)

Dieser Basisansatz behandelt die bislang noch nicht betrachteten Prozessvorlagen Schablonen und Workflows. Die Spezialisierung von reinen Schablonen[147] stellt eine Wiederverwendung generischer Referenzmodelle dar. Sie ist generativ und konfigurativ. *Schablonen* werden an vorgesehenen Stellen für die aktuelle Problemsituation konkretisiert

[147] Reine Schablonen enthalten ausschließlich aus Außensicht spezifizierte Aufgaben (vgl. Abschnitt 4.5.4.1).

(*antizipierte Anpassung*). Idealerweise beginnt die Prozesskonstruktion mit einer Schablone, die den Prozess aus Aufgabensicht vollständig beschreibt und mithilfe weiterer partieller Schablonen schrittweise bis auf Aktivitätsniveau präzisiert werden kann. Dies schließt die Operatorzuordnung und -instanziierung für jede enthaltene Aufgabe ein. Vollständige *Workflows* sind keine generischen Modelle, werden jedoch unter der Annahme, keine Strukturmodifikationen zu erfahren, in diesen Basisansatz eingeordnet. Werden nur ihre Operatorparameter verändert, handelt es sich um eine antizipierte Verhaltensanpassung, die zugleich die Rückführbarkeit des variierten Workflows auf die Vorlage erlaubt. Die strukturelle Modifikation von Workflows entspricht keinem Basisansatz, sondern stellt eine Mischform dar.

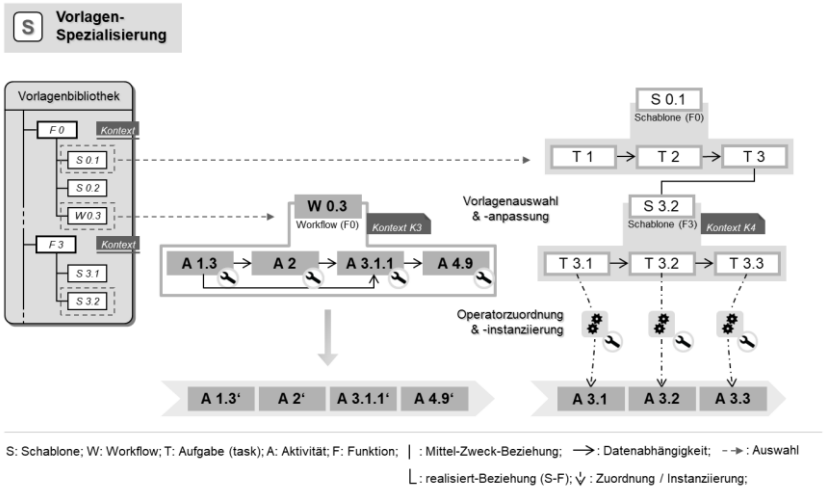


Abbildung 67: Basisansatz Vorlagenspezialisierung (S) (eigene Darstellung)

Die Grundidee der Vorlagenspezialisierung zeigt Abbildung 67, rechts für den Fall einer generischen Schablone, links für den Fall eines vollständigen Workflows. In beiden Fällen ist eine Vorlage zur Realisierung der initialen Analyseaufgabe F 0 gesucht, wofür aus einer entsprechend organisierten Vorlagenbibliothek geeignete Kandidaten abgerufen werden. Die Wahl zwischen mehreren Optionen erfolgt wiederum mithilfe von Kontextfaktoren. Denkbar ist auch die Auswahl

allgemeingültiger Schablonen ohne Kontextbedingungen, wie S 0.1 im Beispiel rechts oben, die etwa die Phasen Datenvorbereitung, Datenanalyse und Ergebnisaufbereitung als Aufgaben repräsentiert. Die Workflow-Anpassung bietet besonders großes Unterstützungspotenzial bei der Prozessgestaltung, da sie Entscheidungen bezüglich der Aufgabenspezifikation, der Reihenfolgebestimmung und der Operatorzuordnung vorwegnimmt. Die Spezialisierung von Schablonen unterstützt nur die ersten beiden Entscheidungen, beschränkt auf die jeweilige Betrachtungsgranularität.

Ein wesentlicher Vorteil dieses Basisansatzes ist seine Unterstützung der kontrollierten Anpassung von Referenzmodellen durch *Konfiguration*. Sie wird von MILI ET AL. ausdrücklich auf Selektion zurückgeführt, d.h., auf die Auswahl einer Option aus einer Menge von Gestaltungsalternativen [MiMM95, 554f.]. Sie erfolgt typischerweise durch Instanziierung von Parametern oder durch Spezialisierung, d.h., durch Substitution abstrakter durch konkretere Elemente [BeDK04, 252]. Abbildung 67 illustriert, wie Aufgabe T 3 aus Schablone S 0.1 mithilfe einer weiteren Schablone S 3.2 in drei Teilaufgaben T 3.1, T 3.2, T 3.3 spezialisiert wird. Die Selektion von S 3.2 erfolgt anhand ihrer Funktion (hier F 3), welche die zu spezialisierende Aufgabe mit passenden Optionen verknüpft, sowie anhand von Kontextfaktoren.

Die Parameterinstanziierung beruht auf der Selektion eines zulässigen Wertes aus einem definierten Wertebereich für einen Parameter. Interpretiert man alle Schablonen der Vorlagenbibliothek analog HAMMEL ET AL. [HaSW98a, 29f.] als integriertes, generisches Referenzmodell, so lässt sich die Konkretisierung abstrakter Elemente vollständig als Selektion und Anwendung von Entwurfsmustern beschreiben. Die Selektion kann u.a. nach Typen (z.B. Funktionszuordnung), durch Auswertung der Attributwerte der Modellelemente oder nach Kontextfaktoren erfolgen [BeDK04, 254-256]. Der Verzicht auf freie Anpassungsoperationen macht die Änderungen reproduzierbar und reduziert das Risiko fehlerhafter Modelle [Drei+05, 692].

Die Wiederverwendung von Vorlagen wirkt komplexitätsreduzierend und verkürzt den Planungsvorgang, da sich die Entwicklung konkreter Lösungsverfahren an wenigstens intensional definierten, partiellen

Lösungsskizzen orientieren kann. Als Initialvorlagen kommen insbesondere auch die in Abschnitt 3.1.1 vorgestellten Prozessmodelle in Frage. ENGELS bemängelt, dass die meisten Analysewerkzeuge solche Modelle allenfalls als Bestandteil der Dokumentation oder als unverbindliche Richtlinien zum Vorgehen ansehen, sie aber nicht explizit zur Planungsunterstützung nutzen [Enge99, 70]. Ihre Aufnahme in die Vorlagenbibliothek ermöglicht eine methodisch fundierte Einbeziehung dieser etablierten Vorschläge in die Prozessspezifikation.

Einen diesem Basisansatz entsprechenden, umfassenden Vorschlag zur kontextbasierten Wiederverwendung von Geschäftsprozessmodellen in Form konkreter Schemata, generischer Schemata sowie generischer Bausteine stellen RUPPRECHT ET AL. [RuPR99] vor. Auswahl und Anpassung der Artefakte geschehen nach Maßgabe von Kontextfaktoren, die mit den Vorlagen ontologisch verknüpft sind. Strukturmuster, deren Kontextfaktoren mit den aktuellen Kontextbedingungen weitgehend übereinstimmen, können kopiert und als Vorlage für die Prozesskonstruktion verwendet werden. Kontextunterschiede determinieren die erforderlichen Adaptionen.

Zur Verwendung von Vorlagen für die Datenanalyse existieren zahlreiche Vorschläge in der Literatur. AMANT & COHEN [AmCo98], [AmCo98b] beschreiben einen KI-basierten Planungsassistenten für die Explorative Datenanalyse, der auf eine Bibliothek wiederverwendbarer Analysestrategien in Form von Skripten zurückgreift. Die Arbeitsweise des Systems gleicht einem hierarchischen partiellen Planer, der abstrakte Pläne im Dialog mit dem Analytiker schrittweise in Subpläne expandiert und dabei Alternativpläne, Rücksprünge und direkte Nutzereingriffe unterstützt. Die Pläne werden als Aktivitätsnetze explizit repräsentiert und dienen gleichzeitig der Dokumentation des zurückgelegten Analysepfads.

Der im CITRUS-Projekt verfolgte Multistrategie-Ansatz stützt sich neben der Operatorkomposition und der KI-basierten Prozessplanung hauptsächlich auf die Wiederverwendung von vollständigen Workflows oder Schablonen [EnLS97, 164], [Enge99, 91f.]. OU & PENG [OuPe06] stellen ein kombiniertes fall- und regelbasiertes System zur Konstruktion und Wiederverwendung von Data-Mining-Workflows vor. Diese

werden auf Basis eines domänenbezogenen Ähnlichkeitsmaßes aus der Fallbasis abgerufen und können direkt oder modifiziert wiederverwendet werden. Ihr Anwendungskontext wird durch Geschäftsprozessmodelle definiert, deren Elemente in Ontologien verwaltet werden, um hierarchische Beziehungen zwischen repräsentierten Konzepten bei der Inferenz berücksichtigen zu können.

KIETZ ET AL. [KSBF10] erweitern den HTN-Planer aus [KSBF09] (vgl. Abschnitt 5.3.1.2) zu einem interaktiven Planungsansatz für Analyseprozesse, der sich auf die Ontologie DMWF stützt, die sowohl maschinell als auch vom Nutzer interpretierbar ist. Das Planungssystem schlägt jeweils zulässige Konkretisierungsoptionen vor, überlässt die Gestaltungsentscheidung aber dem Analytiker. Zwischenplanungsstufen werden durch wiederverwendbare Workflow-Templates abgebildet, die aus Aufgaben und Aktivitäten bestehen können.

WEGENER & RÜPING [WeRü11] definieren Prozess-Patterns zur Wiederverwendung im Data Mining sowie ein Vorgehensmodell zu deren Integration in Geschäftsprozesse. Die Muster beschreiben anwendungsunabhängige Prozesse beliebigen Detaillierungsgrads, können aber spezielle Anforderungen enthalten. Das Pattern höchster Ebene bilden dabei die Aufgaben eines modifizierten CRISP-DM-Modells. Anpassungen erfolgen ausschließlich durch Konkretisierung von Aufgaben in Subaufgaben oder Aktivitäten und resultieren jeweils in einem neuen Pattern [WeRü11, 271f].[148] Der Ansatz wird in [WeAR11] um die Berücksichtigung von Datensemantik und Ontologien erweitert.

[148] Die Ausgestaltung des Ansatzes beruht auf teils fragwürdigen konzeptuellen Annahmen. So wird der vom Muster beschriebene Analyseprozess auf derselben konzeptuellen Ebene wie der Geschäftsprozess modelliert und durch Konnektoren und Datenflüsse direkt in diesen integriert [WeRü11, 269]. Im Hinblick auf die Übereinstimmung von Annahmen und Zielen der Anwendung sowie die Verfügbarkeit und Qualität von Daten werden teils unrealistische Annahmen getroffen, die entweder zu einer identischen Übernahme oder zu einer Zurückweisung des Patterns führen, ohne eine Anpassung zu erlauben. Zudem werden Meta-Aufgaben des Pattern-Managements im Pattern selbst repräsentiert (etwa Prüfung auf Übereinstimmung der Ziele [WeRü11, 272], Neuberechnung von Modellen [WeRü11, 270]), wodurch z.B. die Wiederverwendung zum Gegenstand des Patterns wird. Zur Prozessausführung nicht benötigte Meta-Aufgaben müssen zur leeren Aufgabe „spezialisiert“ werden [WeRü11, 272].

Eine Entscheidungshilfe zur Unterstützung der Lösung von Analyseproblemen stellt PETERSOHN [Pete03], [Pete04] mit der Data-Mining-Anwendungsarchitektur vor. Sie dient der Verwaltung und Konstruktion von Prozessschemata, die über drei Generalisierungsebenen zu erarbeiten sind. Verfahren und Prozesse werden zunächst anwendungsunabhängig auf allgemeingültiger Ebene beschrieben und können für Verfahrensklassen und konkrete Verfahren sowie für Anwendungsbereiche und konkrete Anwendungen spezialisiert werden. Der Ansatz kann als Beitrag zur Systematisierung von Analysewissen angesehen werden. Seine praktische Nutzung wird jedoch durch ein eigenwilliges Begriffssystem und eine wenig intuitive Strukturierung in Dimensionen („Komponenten“) und Ebenen erschwert. Die als Ereignisgesteuerte Prozessketten repräsentierten Prozessschemata sind zur Anwendung in Analysewerkzeugen nicht geeignet. Wie anwendungsspezifische Schemata aus allgemeingültigen Schablonen abzuleiten sind, wird indes nicht beschrieben.

5.3.3 Empfehlungen zur Analyseprozessplanung

Die vier Basisansätze weisen individuelle Stärken und Schwächen auf, die in den vorausgehenden Abschnitten skizziert sind. Als wesentliches Kriterium aus Anwendersicht dient die Frage, inwieweit die drei im Rahmen der Planung zu treffenden Gestaltungsentscheidungen (1) Aufgabenspezifikation und -zerlegung, (2) Zuordnung von Operatoren und (3) Bestimmung der Aufgabenreihenfolge unterstützt werden. Es wird eine Funktionszuordnung der Bausteine und Operatoren gemäß dem Modellierungsansatz aus Kapitel 4 angenommen.

- **Spezialisierung** (S): Im Falle von Workflows werden alle Entscheidungsaufgaben (1), (2) und (3) unterstützt, im Falle von Schablonen nur die Entscheidungen (1) und (3), jeweils beschränkt auf den Umfang des Prozessausschnitts und die jeweilige Zerlegungsgranularität.
- **Rekombination** (R): Der Grad der Unterstützung variiert mit dem Umfang der Bausteine. Fragmente unterstützen für den jeweiligen Prozessausschnitt die Aufgaben (1), (2) und (3). Einzelaktivitäten

bieten nur umfassende Hilfestellung bei Entscheidung (2), während sich die Unterstützung für die Aufgabenspezifikation (1) auf die betreffende Einzelaktivität beschränkt. Die Rekombination liefert als einziger Basisansatz stets Vorschläge zur Instanziierung der Operatorparameter. Inwiefern diese unmodifiziert anwendbar sind, ist kontextabhängig.

- **Dekomposition** (D): Dieser Ansatz bietet uneingeschränkte Unterstützung bei der Aufgabenspezifikation (1). Die Operatorzuordnung (2) wird durch die Funktionszuordnung erleichtert. Die Bestimmung der Ablaufreihenfolge (3) obliegt dem Analytiker.
- **Komposition** (K): Die Operatorkomposition realisiert unmittelbar Entscheidung (2) und liefert mithilfe der Funktionszuordnung implizit die lokale Aufgabenspezifikation (1) für die resultierende Einzelaktivität, ohne einen semantischen Zusammenhang zum Analyseproblem herzustellen. Die Reihenfolgeplanung (3) ist separat zu behandeln.

Basisansatz	Ausprägung von Prozessartefakten	❶ Aufgabenspezifikation		❷ Operatorzuordnung		❸ Reihenfolgebestimmung		Präferenz
S: Spezialisierung	*Workflows*	✓✓		✓✓		✓✓		**1**
	Schablonen	✓	1, 2			✓	1, 2	**2a**
R: Rekombination	*Fragmente*	✓	1, 3	✓✓	1, 4	✓	1	**3**
	Aktivitäten	✓	3, 5	✓✓✓	4			**4**
D: Dekomposition		✓✓		✓	3			**2b**
K: Komposition		✓	3, 5	✓✓				**5**

1: beschränkt auf das jeweilige Prozesssegment; 2: bis zur repräsentierten Granularität; 3: explizit nur bei funktionsorientierter Operator-Kategorisierung; 4: einschließlich Vorschlag zur Parameter-Instanziierung (führt zu Aufwertung); 5: beschränkt auf Spezifikation der betreffenden Einzelaufgabe (führt zu Abwertung)

Abbildung 68: Unterstützung von Gestaltungsentscheidungen durch die Basisansätze der Prozessplanung (eigene Darstellung)

Auf Grundlage der dargestellten Eigenschaften lässt sich eine Bewertung der Basisansätze erstellen[149] (Abbildung 68), aus der eine allgemeine Präferenzordnung sowie Hinweise auf empfehlenswerte

[149] Die zugrundeliegende Nutzenbewertung der Basisansätze erfolgt durch pragmatische Zuteilung von 2 Punkten für jede uneingeschränkt sowie von 1 Punkt für jede eingeschränkt unterstützte Gestaltungsentscheidung. Die Eigenschaften Parameter-Instanziierung und Beschränkung auf Einzelaktivitäten führen wegen ihrer besonderen Bedeutung zu einer zusätzlichen Auf- bzw. Abwertung um jeweils 0,5 Punkte.

Kombinationen ableitbar sind. Bei Einzelbetrachtung ergibt sich die Präferenzordnung $S_{Workflows} > R_{Fragmente} > D = R_{Aktivitäten} > K > S_{Schablonen}$. Hierbei ist zu bedenken, dass Analytiker aufgrund individueller Präferenzen und Erfahrungen zu abweichenden Rangfolgen gelangen oder andere Kriterien anlegen mögen. Diese Bewertung ist daher weniger als Empfehlung für einen Basisansatz zu verstehen, macht aber deutlich, dass die Wiederverwendung vollständiger Workflows und erfolgreich ausgeführter Prozessfragmente besonders hohes Unterstützungspotenzial bieten und alle drei Gestaltungsentscheidungen abdecken. Die anderen Ansätze sind insbesondere bei kombiniertem Einsatz hilfreich.

Die im Rahmen der Schablonenspezialisierung und der Dekomposition zu bearbeitenden Aufgaben erleichtern die Gestaltungsentscheidungen, da jeweils nur nach solchen Optionen zu suchen ist, die eine zulässige Konkretisierung der aktuell betrachteten Aufgabe darstellen. Als Suchkriterium dient die Funktion der Aufgabe, über die sich passende Optionen auffinden lassen. Werden hierbei Optionen aus allen Basisansätzen berücksichtigt, können Schwächen einzelner Basisansätze ausgeglichen werden. Beispielsweise ist das Aufgabenzerlegungsproblem der Dekomposition durch Rückgriff auf Schablonen (Entwurfsmuster) und auf Blattebene des Aufgabenzerlegungsbaums zusätzlich mithilfe von Fragmenten (Strukturmuster) lösbar.[150]

Tatsächlich profitiert die Dekomposition derart von einer Kombination mit der Schablonenspezialisierung, dass ihre eigenständige Anwendung kaum erstrebenswert ist. Die beiden Ansätze können zusammen alle drei Gestaltungsentscheidungen abdecken und werden daher explizit zum gemeinsamen Einsatz empfohlen. Damit ergibt sich die in Abbildung 68 dargestellte Präferenzordnung $S_{Workflows} > (D + S_{Schablonen}) > R_{Fragmente} > R_{Aktivitäten} > K$. Als allgemeine Planungsheuristik lässt sich damit die Empfehlung formulieren, zunächst die Möglichkeit der Wiederverwendung vollständiger Workflows zu prüfen. Scheitert diese

[150] So empfehlen auch HAMMEL ET AL. die Anreicherung generischer Referenzmodelle um nicht-generische Modellbausteine im Sinne von Prozessfragmenten, um in Situationen, in denen generische Lösungen keine Vorteile erzielen, die Entwicklungseffizienz durch solch „einfache Lösungen“ zu steigern [HaSW98a, 31f.].

Option, sollte ausgehend von der globalen Prozessaufgabe mithilfe von Schablonen eine möglichst detaillierte Dekomposition entwickelt werden, die den Rahmen bildet für die Integration wiederverwendbarer Prozessfragmente oder einzelner Aktivitäten. Stehen solche Bausteine nicht in geeigneter Ausprägung zur Verfügung, ist auf die Operatorkomposition zurückzugreifen.

Diese Empfehlung entspricht der intuitiven Vorstellung, jeweils den maximalen Grad der Wiederverwendung anzustreben, lässt jedoch die damit verbundenen Bereitstellungs- und Anpassungskosten außer Acht. Bereitstellungskosten für Prozessartefakte entstehen bei ihrer Beschreibung und Integration in die Fallbibliothek und sind bei Workflows zu vernachlässigen, da diese direkt von bereits geplanten bzw. ausgeführten Prozessen übernommen werden können. Bei Fragmenten und Schablonen sind Aufwände für Identifikation und Isolation geeigneter Prozessausschnitte, bei Schablonen gegebenenfalls auch für deren Abstraktion zu berücksichtigen. Sie amortisieren sich bei mehrfacher Nutzung in der Regel schnell. Wiederverwendung ist im Allgemeinen empfehlenswert, wenn die Anpassungskosten geringer sind als die Kosten einer Neukonstruktion. Dies ist regelmäßig bei konfigurierbaren oder umfangreichen Bausteinen (wegen der Kontrollierbarkeit bzw. des hohen Einsparpotenzials) zu erwarten [MiMM95, 537f.]. Einer Untersuchung realer Data-Mining-Prozesse von WEGENER & RÜPING zufolge betreffen Änderungen der Prozessspezifikation in der Datenvorbereitungsphase zu 50% und in der Analysephase zu 75% manuelle Parameteranpassungen, also die verhaltensbasierte Konfiguration [WeRü11, 265]. Legt man ähnliche Größenordnungen auch bei anderen Analyseansätzen zugrunde, so ist anzunehmen, dass das Kostenkriterium in der Mehrzahl der Fälle erfüllt ist. Es bleiben Fragen der Fallabdeckung, der Identifikation von Prozessbausteinen und der Ermittlung eines geeigneten Abstraktions- bzw. Generalisierungsniveaus zur Ablage in der Bibliothek, für die in Abschnitt 7.4.2.3 Lösungsvorschläge diskutiert werden.

Die in den folgenden Abschnitten präsentierten Handlungsschemata zur Problem- und Prozessspezifikation sind unabhängig von den vier Basisansätzen. In allen Situationen, in denen entsprechende Gestal-

tungsentscheidungen zu treffen sind, können die Basisansätze einzeln oder kombiniert zum Einsatz gelangen. Insbesondere sollte der Einstieg in die Planung über jeden der Basisansätze möglich sein. Konkret bedeutet dies, dass für aktuell zu bearbeitende Modellierungsobjekte (Problemaspekte, Analyseziele, Prozessaufgaben) jeweils geeignete Artefakte aus einer Fallbibliothek angeboten werden, sofern dies die Planung unterstützt. Auf Prozessebene sollten Prozessbausteine gemeinsam mit Operatoren präsentiert werden (vgl. [Enge99, 184]). Alle verfügbaren Gestaltungsoptionen können auf diese Weise nach Kontextfaktoren und weiteren Kriterien bewertet und sortiert werden, um jeweils die zweckdienlichste Alternative zu ermitteln. Zusätzlich ist die Bereitstellung kontextspezifischen Erfahrungswissens hilfreich [Knob03a, 350].

5.4 Problemspezifikation

Der vorliegende und der folgende Abschnitt präsentieren Handlungsschemata zur Planung von Datenanalyseprozessen. Diese sind Bestandteile eines Vorschlags für eine Methodik, wie sie in Abschnitt 3.2.4 (Maßnahme B2) zur Unterstützung des Analytikers gefordert wird. Neben den dort genannten Eigenschaften werden insbesondere eine methodisch gestützte Datenauswahl und -selektion (R1.3) sowie die Wiederverwendung von Prozessartefakten und Erfahrungswissen (R3.1) explizit berücksichtigt. Die Handlungsschemata stützen sich auf den Modellierungsansatz aus Kapitel 4 und konkretisieren das Vorgehensmodell aus Abschnitt 3.3, indem sie dessen Phase zur Planung des Analyseprojekts mit Aufgaben und Empfehlungen ausfüllen.

5.4.1 Aufgaben und Vorgehen bei der Problemspezifikation

Abbildung 69 zeigt die Aufgaben und die idealtypische Reihenfolge für die Problemspezifikation. Sie betreffen die Anwendungs- und Analyse-

ebene der Datenanalysearchitektur, die zur einheitlichen Referenz zur Zielebene Z zusammengefasst werden.[151]

Den inhaltlichen Rahmen der Problemspezifikation bilden die in Abschnitt 3.1.2.1 präsentierten Aufgaben. Sie umfassen die *Identifikation eines Sachproblems (Z1)*, die *Domänenanalyse (Z2)*, die *Spezifikation des Analyseproblems (Z3)*, die Erstellung des *Untersuchungsdesigns (Z4)* und die begleitende *Projektplanung (Z5)*. Sie werden als Schritte der Planungsmethodik übernommen und in den folgenden Abschnitten jeweils einzeln erläutert. Zuvor werden Arbeiten genannt, die zur theoretischen Fundierung der Methodik beigetragen haben.

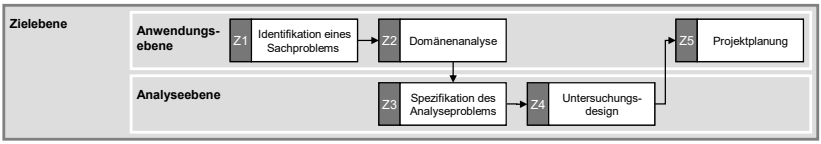


Abbildung 69: Handlungsschema zur Problemspezifikation (eigene Darstellung)

5.4.2 Theoretische Fundierung

5.4.2.1 Verwandte Arbeiten

Trotz der allgemein akzeptierten Sichtweise, dass Datenanalyse zur Lösung von Sachproblemen beitragen soll (vgl. Abschnitt 2.1.1), finden sich in der Literatur nur wenige Beiträge zur systematischen Identifikation und Beschreibung der zu lösenden Probleme. Ein integratives Konzept, das von speziellen Ausprägungen der Datenanalyse abstrahiert und methodisch fundiert vom Sachproblem zum Analyseprozess führt, ist nicht bekannt. Existierende Arbeiten beziehen sich jeweils auf einzelne Analysedisziplinen oder decken nur einzelne der oben genannten fünf Aufgaben ab. Der bereits erwähnte Ansatz der Gruppe um ENGELS [Enge96], [EnLS97], [Enge99], [ThLi98] beinhaltet rudimentäre Hinweise zur Spezifikation von Analysezielen für Data Mining,

[151] Die Bezeichnung Zielebene (Z) wurde primär gewählt, um bei Verwendung von Kurzbezeichnungen eine Verwechslung der äquivalenten Projektebene mit der Prozessebene (P) zu vermeiden.

klammert deren Herleitung jedoch aus. CRISP-DM [CCK+00] enthält eine umfangreiche Aufgabenliste mit Erläuterungen für KDD, bleibt bezüglich der Aufgabenrealisierung aber dennoch vage. Diese Lücke füllt PYLE [Pyle03], der eine Sammlung bewährter Praktiken und konkreter Hinweise zur Identifizierung, Operationalisierung und Beantwortung von Sachfragen beim Data Mining präsentiert, aber eine strukturierte Vorgehensweise schuldig bleibt.[152] Dies gilt analog für den Fragenkatalog zur Problemspezifikation von HANCOCK [Hanc12, 37ff.], der zudem nicht kontextsensitiv ist. CAO ET AL. [CYZZ10] propagieren ein problemorientiertes Paradigma als „Domain-Driven Data Mining."[153] Zur Überwindung der Lücke zwischen Anwendung und Technik soll eine Reihe von Anforderungen, etwa die Einbeziehung von Domänenwissen oder betriebswirtschaftlichen Zielen, beitragen, die jedoch formalisiert zu erfassen und z.B. als Algorithmusparameter zu berücksichtigen sind. Die empirische Forschung in den Sozialwissenschaften verfügt über eine fundierte Methodik des Untersuchungsdesigns und der Operationalisierung von Forschungsfragen (vgl. z.B. [Benn94], [Drei94], [Diek07, 186ff.]), deren Schwerpunkt aber auf Erhebungsdesign und Messtheorie liegt.

Nur wenige Arbeiten befassen sich ausführlicher mit Einzelaspekten der fünf Aufgaben. So präsentieren BRACHMAN ET AL. [Brac+93] ein interaktives Datenanalysesystem, das den Anwendungsbereich mittels eines konzeptuellen Domänenmodells beschreibt. Die Domänenobjekte und Beziehungen dieser Ontologie stellen zugleich eine datenquellenunabhängige Repräsentation verfügbarer Daten dar. Der Nutzer interagiert mit dem System über das Domänenmodell. Dessen Rolle als Kommunikationsbasis und Integrationsmodell erscheint wegweisend. WIRTH & REINARTZ [WiRe96] schlagen die sukzessive Zerlegung von Aufgaben im Sinne von Analyseproblemen vor, bis ihnen konkrete Verfahren zugeordnet werden können. Ein ähnlicher Ansatz von ADOMAVICIUS & TUZHILIN [AdTu97] beruht auf der hierarchischen

[152] Er verknüpft kontextbezogene Empfehlungen von hoher inhaltlicher Relevanz über Querverweise, wodurch keine übergeordnete Systematik erkennbar ist.

[153] Vgl. hierzu auch den Bericht zum ACM SIGKDD Workshop on Domain-Driven Data Mining [Cao+07].

Strukturierung von möglichen Handlungsmaßnahmen (Projektziele). Den Maßnahmen werden jeweils Analyseergebnistypen zugeordnet, um die Handlungsorientierung der Analysen zu unterstützen. Die hierarchische Problemzerlegung erweist sich dabei als geeignetes Hilfsmittel für die Konkretisierung zu untersuchender Sachverhalte und für die Auswahl von Operatoren.

HAND [Hand94] warnt vor Fehlern durch nicht sachgerechte Abbildung des Sachproblems auf analytische Fragestellungen und zeigt anhand einiger Beispiele aus der Statistik, welche Fehlschlüsse und Auswirkungen die Folge sein können. COHEN [Cohe96] und GLYMOUR ET AL. [GMPS97] ergänzen weitere Hinweise zur korrekten Planung statistischer Untersuchungen und zeigen die Bedeutung des Untersuchungsdesigns für die Anwendbarkeit einer Analyse auf. HOGL [Hogl03] untersucht die zweckorientierte Formulierung von Analysezielen in Form von Fragen und gibt Hinweise zur zielorientierten Verfahrensauswahl.

Die Berücksichtigung von Kosten-Nutzen-Aspekten zur Ausrichtung des gesamten Projekts an ökonomischen Kriterien wird unter dem Begriff „Utility-based Data Mining" behandelt [WeZS08]. Der Schwerpunkt vieler Arbeiten in diesem Feld liegt auf der Algorithmenentwicklung, um z.B. die Prognosegenauigkeit an sachlichen Nutzenzielen zu orientieren.[154] ALI & WALLACE [AlWa97] illustrieren in einer früheren Arbeit ähnlicher Prägung, wie betriebswirtschaftliche Ziele auf die Parameter von Analyseverfahren abgebildet werden können. Diese Arbeiten zeigen Wege auf, wie bei der Problemspezifikation berücksichtigte ökonomische Kriterien die Planung der Untersuchung leiten können.

5.4.2.2 Entscheidungstheorie

Die in Abschnitt 4.3.1 eingeführte Sicht auf ein Problem als Wunsch nach Überwindung eines unbefriedigenden Zustands trifft auf zahl-

[154] Die Grenzen zum Domain-Driven Data Mining sind fließend, wenngleich dieses eine Ausrichtung am Anwendungskontext im Allgemeinen anstrebt, während Utility-based Data Mining speziell auf ökonomische Kriterien abstellt.

reiche Situationen in allen Lebensbereichen zu. Die Frage, wie diese Überwindung geschehen soll, ist Gegenstand einer Entscheidung. Bei Zugrundelegung eines weit gefassten Entscheidungsbegriffs lässt sich jede Tätigkeit als Resultat einer bewussten oder unbewussten Entscheidung interpretieren [Hein91, 12]. Es liegt daher nahe zu prüfen, inwiefern die Entscheidungstheorie Hinweise zur Problemspezifikation geben kann.

Die präskriptive Entscheidungstheorie unterscheidet zwischen Entscheidungsfeld und Entscheidungszielen. Das Entscheidungsfeld beschreibt verfügbare Handlungsoptionen (Aktionsraum), nicht beeinflussbare Umweltzustände und die Menge denkbarer Handlungskonsequenzen, die sich aus der Kombination aller Handlungsoptionen mit jedem Umweltzustand ergibt [Hein91, 26f.]. Die Entscheidungsziele bestimmen gemäß verschiedener Kriterien und Regeln, welche Optionen aus dem Aktionsraum zu wählen sind [Hein91, 28f.]. Die deskriptive Entscheidungstheorie ergänzt die präskriptive Theorie, die wegen nicht erfüllbarer Annahmen in der Realität häufig nicht anwendbar ist [Hein91, 44]. Sie beschreibt reale Entscheidungsprozesse anhand der Phasen Willensbildung und Willensdurchsetzung. Letztere betrifft die Anwendung des Wissens im datenanalytischen Vorgehensmodell und kann hier vernachlässigt werden. Die Willensbildung gliedert sich in eine Anregungsphase (Problemerkennung und Ursachenanalyse), eine Suchphase (Festlegung von Auswahlkriterien, Handlungsoptionen und Handlungskonsequenzen) und eine Auswahlphase (Entscheidung) [Hein91, 35f.].

Die von den Entscheidungstheorien behandelten Aspekte sind in Abbildung 70 den Komponenten des Problembegriffs zugeordnet. So fällt die Beschreibung des unbefriedigenden Ausgangszustands in die Anregungsphase. Ein umfassenderes Bild der Ist-Situation entsteht durch Ergänzung der Umweltzustände. Die Ermittlung relevanter Einflussfaktoren ermöglicht eine Ursachenanalyse, die Hinweise auf Handlungsoptionen liefern kann. Handlungsoptionen stellen mögliche Operatoren zur Überwindung der Barriere dar und komplettieren zusammen mit den zu erwartenden Handlungskonsequenzen die Elemente des Entscheidungsfelds. Sie werden gemeinsam mit den

anzustrebenden Zielen in der Suchphase ermittelt. Eine zulässige Handlungsoption muss Konsequenzen erwarten lassen, die mit den gesetzten Zielen im Einklang stehen. Die Ziele sind mit messbaren Erfolgskriterien zu verknüpfen, welche die Beschreibung des Soll-Zustands konkretisieren. Die Entscheidungstheorie erweist sich demnach zur Unterstützung der Problemspezifikation geeignet und fließt in die Herleitung der Methodik ein.

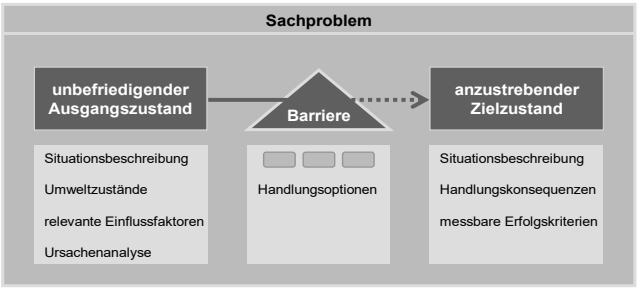


Abbildung 70: Komponenten und assoziierte Aspekte von Sachproblemen (eigene Darstellung)

5.4.3 Identifikation eines Sachproblems (Z1)

Sachprobleme existieren unabhängig von analytischen Erwägungen, können jedoch zum Ausgangspunkt von Datenanalysen werden, sofern diese der Problemlösung dienlich erscheinen. Zur zweckorientierten Planung und Durchführung dieser Analysen ist ein Orientierungsrahmen aufzuspannen, der eine präzise Beschreibung der Sachprobleme und ihrer Domäne enthält, fachliche Annahmen und Anforderungen benennt und die für das zur Problemhandhabung initiierte Projekt geltenden Ressourcenrestriktionen definiert (vgl. [Pyle03, 35]). Dieser Rahmen wird durch die auf Anwendungsebene angesiedelten Aufgaben Z1, Z2, Z5 konstruiert und schließlich durch die Aufgaben der Analyseebene Z3, Z4 ausgefüllt.

Der erste Schritt Z1 zielt auf die Identifikation eines Sachproblems. Sein Ergebnis besteht in einer Problembeschreibung in Form von **`Problem-aspekten`** gemäß Abschnitt 4.3.1.3. Er gliedert sich in die Teilaufgaben

Problemerkennung (Z1.1), *Diskursweltabgrenzung (Z1.2)* und *Problembeschreibung (Z1.3)*.

5.4.3.1 Problemerkennung (Z1.1)

Voraussetzung für die Identifikation eines Sachproblems ist zunächst, dass ein Ist-Zustand als unbefriedigend erkannt wird. Die Problemerkennung wird häufig durch Informationsmangel in Bezug auf die vorliegende Situation oder die anzustrebenden Ziele behindert [FiWo90, 14f., 24]. Die Unkenntnis der aktuellen Lage lässt sich durch Datenanalysen reduzieren, aber bezüglich einzelner Teilzustände niemals vollkommen ausschließen. Hierfür eignen sich z.B. die mit Business Intelligence assoziierten Konzepte Frühwarnung, Früherkennung und Frühaufklärung [Lieb96, 12-18], [Knob02, 339f.], die u.a. durch Standardberichtswesen, Wissensentdeckung und Wirkungsanalysen realisierbar sind (vgl. Abschnitt 2.2.2). Die Unbestimmtheit der Ziele stört die Problemerkennung, da sie die Wahrnehmung von Soll-Ist-Abweichungen vereitelt. Dies kann bereits bei unklar formulierten oder konfliktären Zielen eintreten [Dörn83a, 21-23].

In der Analysepraxis sieht sich der Datenanalytiker zuweilen mit der Aufgabe konfrontiert, ein undefiniertes oder nicht klar artikuliertes Problem zu lösen.[155] Damit ist die Schwierigkeit verbunden, zunächst das dem Auftrag zugrunde liegende Sachproblem zu eruieren. Dies geschieht interaktiv durch Befragung des Auftraggebers oder assoziierter Fachexperten (vgl. [Pyle03, 63f.]). Alternativ liefern explorative Analysen der übergebenen Daten in der Regel Anhaltspunkte (Hypothesen) zu möglichen Problemen. In anderen Fällen werden vom Auftraggeber identifizierte Probleme oft mit nur vager Beschreibung an den Analytiker übergeben. Beispiele sind „mangelnde Produktqualität“ oder „ineffiziente Produktionsprozesse“.

Psychosoziale Merkmale des Entscheidungsträgers beeinflussen die Problemerkennung, indem sie dessen Blick auf die Diskurswelt

[155] Solche Aufträge werden oft mit dem unspezifischen Ziel ausgesprochen, „interessante Informationen“ aus einer Datenquelle zu extrahieren [Pyle03, 63], [Milt10, 3].

bestimmen und durch subjektive Einstellungen, Annahmen, Vorwissen und mentale Kontexte die Wahrnehmung problemrelevanter Aspekte leiten. Demzufolge nimmt auch die Diskursweltabgrenzung unmittelbar Einfluss auf die Fähigkeit zur Erkennung und die Art der Wahrnehmung von Problemen (vgl. [Haus90, 138], [Pyle03, 126]).

5.4.3.2 Diskursweltabgrenzung (Z1.2)

Der Zustand, auf den sich die problematische Soll-Ist-Abweichung bezieht, wird durch ein konzeptuelles Merkmal (Problemmerkmal) eines konzeptuellen Domänenobjekts (Problemobjekt) repräsentiert (vgl. Abschnitt 4.3.1). Dieses *Problemobjekt* bildet den Ausgangspunkt für die Diskursweltabgrenzung. Als Diskurswelt wird der relevante, zu betrachtende Ausschnitt der Realität bezeichnet. Sie umfasst neben den Objekten auch alle relevanten Beziehungen zwischen diesen Objekten [FeSi13, 7]. Hierbei ist zu beachten, dass abhängig von der eingenommenen Sichtweise auch Beziehungen zwischen Domänenobjekten als Problemobjekt auftreten können (z.B. Auftrag als Interaktionsbeziehung zwischen Unternehmen und Kunde).

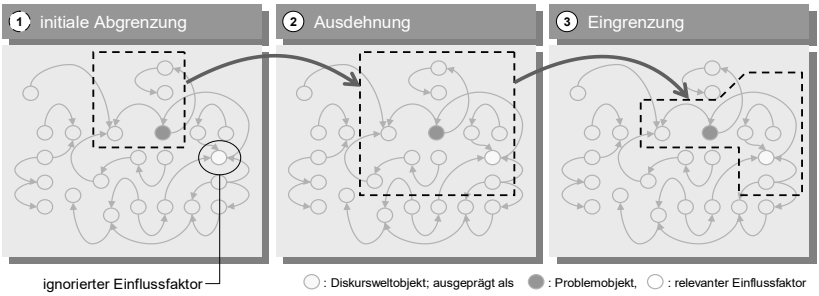


Abbildung 71: Diskursweltabgrenzung: Fokussierung auf die Problemdomäne (eigene Darstellung)

In der Regel ist die Betrachtung des Problemobjekts allein zur vollständigen Erfassung einer Problemsituation nicht ausreichend. Aufgrund der Beziehungen zwischen den Objekten existieren meist zahlreiche Einflussfaktoren, die potenziell auf das Problemmerkmal einwirken. Aus diesem Grund ist eine mehrstufige Vorgehensweise zur

Diskursweltabgrenzung zu empfehlen: Ausgehend vom Problemobjekt und seiner unmittelbaren Umwelt wird die Betrachtung zunächst ausgedehnt, um weitere Einflussfaktoren identifizieren zu können. Anschließend kann der Fokus wieder eingeengt werden. Abbildung 71 zeigt schematisch, wie bei der initialen Abgrenzung (1) ein relevanter Einflussfaktor ignoriert wird. Dieses Objekt gerät erst mit der Ausdehnung (2) ins Blickfeld, welches sodann auf den relevanten Ausschnitt eingegrenzt wird (3).

Es wird deutlich dass die Fähigkeit, zu einer wirkungsvollen Problemlösung zu gelangen, stark von einer adäquaten Diskursweltabgrenzung abhängt, die jedoch nicht einfach zu finden ist. Sie erfolgt typischerweise iterativ und kann auch während späterer Schritte der Problemspezifikation angepasst werden. Das Ergebnis der Diskursweltabgrenzung heißt `Problemdomäne` (vgl. [KlZy96, 576f.]) und ist im gleichnamigen Attribut des Problemaspekts abgelegt. Das Problemmerkmal wird als `Inhalt` des Problemaspekts dokumentiert.

Zur Unterstützung der Diskursweltabgrenzung sowie zur Dokumentation ihrer Ergebnisse ist eine intuitiv verständliche, unkompliziert handhabbare und in Bezug auf Umfang und Detaillierungsgrad leicht anpassbare Repräsentationsform zu empfehlen, wie sie etwa die in Abschnitt 4.7.1.4 angeregte Verknüpfung einer Domänenontologie mit Diagrammen bietet. Auf diese Weise lassen sich die Objekte und Beziehungen, welche die Problemdomäne konstituieren, direkt durch Selektion der Diagrammelemente auswählen.[156]

[156] Um die Akzeptanz des Vorgehens zu erhöhen, ist im Unternehmen etablierten Diagrammtypen der Vorzug zu geben. Da der Modellierungszweck ausschließlich in der Benennung einer Menge ausgewählter Domänenobjekte zur Verortung des Sachproblems im betrieblichen Objektsystem besteht, kann jeder geeignete Ansatz gewählt werden, wie etwa Objekt-/Klassenmodelle. Verhaltensorientierte Geschäftsprozessmodelle sind weniger geeignet, da sie das Zusammenwirken von betrieblichen Aufgaben bzw. Organisationseinheiten jeweils aus der Ablaufsicht einzelner Geschäftsvorfälle beschreiben und zudem häufig Metaobjekttypen umfassen, die von der wesentlichen Interaktionsstruktur ablenken (z.B. Ereignisse).

5.4.3.3 Problembeschreibung (Z1.3)

Die identifizierten und bezüglich ihrer Domäne abgegrenzten Sachprobleme sind schließlich in eine Beschreibung zu überführen, die eine hinreichende Basis für die Entwicklung und Beurteilung von Problemlösungen bildet. Die Formulierung der Problembeschreibung erfolgt typischerweise interaktiv [BrAn96, 48], [Pyle03, 63f.]. Fachexperten verfügen über wichtiges Problemwissen, das bei der Beauftragung einer Datenanalyse oft nicht artikuliert oder als bekannt vorausgesetzt wird. Es ist daher Aufgabe des Analytikers, dieses Wissen zu erheben und das Sachproblem gemeinsam mit dem Auftraggeber zu strukturieren und präzisieren [Milt10, 3-7]. Am Ende steht eine operationale Problembeschreibung, die eindeutige Projektziele, messbare Erfolgsfaktoren und ökonomische Rahmenfaktoren umfasst [HiWi01, 2f.], [DeHa01, 66].

Die Beschreibung erfolgt so detailliert wie nötig und möglich. Eine vollständige Problemdefinition umfasst zwei Problemaspekte (vgl. Abschnitt 4.3.1.3). In einfacheren Fällen kann es genügen, nur Ausgangs- oder Zielzustand zu definieren. So mag z.B. für die Neuentwicklung eines Data Products, etwa eines Fluginformationssystems, der lösungsorientierte Problemaspekt ausreichen, der die „Information über die Ankunftszeit eines gewählten Fluges per Mobiltelefon“ fordert. Abhängig von der Art der Barriere zwischen Ist- und Soll-Zustand sind drei Problemtypen zu unterscheiden [Dörn79, 10f.], die die Definition des Zielzustands behindern können:

- *Probleme mit Interpolationsbarriere*: Ausgangs- und Zielzustand sind bekannt, die Interpolation ist unbekannt. Bekannte Operatoren müssen adäquat kombiniert werden. Dieser Idealfall ist durch Auswahl von Handlungsoptionen relativ leicht lösbar.
- *Probleme mit Synthesebarriere*: Ausgangs- und Zielzustand sind bekannt, die Operatoren sind unbekannt. Geeignete Operatoren sind zu generieren (Synthese). Die Lösung dieses Falls erfordert zusätzlich die Erkundung zulässiger und ausführbarer Handlungsoptionen.

- *Probleme mit dialektischer Barriere*: Der Zielzustand ist mit Ausnahme einiger global gültiger Kriterien unbekannt. Die Problemlösung wird im Rahmen eines dialektischen Prozesses entwickelt. In diesem Fall ist vor der Suche nach Handlungsoptionen das zu erreichende Ziel zu ermitteln.

Im betriebswirtschaftlichen Kontext sind insbesondere die beiden letztgenannten Problemtypen relevant. Sie bedingen häufig eine nicht ausreichend präzise Problembeschreibung, die im Rahmen der anschließenden Domänenanalyse (Z2) zu konkretisieren ist. Dort werden auch geeignete Handlungsoptionen aufgedeckt.

Jeder **`Problemaspekt`** erhält einen `Namen`, eine `Beschreibung` sowie eine Spezifikation des betrachteten Zustands in Form der Merkmale `Zielinhalt`, `Zielausmaß` und `Zeitbezug`. Der situationsbezogene Problemaspekt (`Typ`=„Ist“) dokumentiert den Ausgangszustand und bildet die Grundlage zur Analyse von Ursachen und zur Ableitung von Handlungsoptionen. Der lösungsbezogene Problemaspekt (`Typ`=„Soll“) spezifiziert mit dem Zielzustand die gewünschte Wirkung der Problemlösung. Dem Probleminhalt kann jeweils ein betriebswirtschaftliches Ziel als `Wertbeitrag` zugeordnet werden. Betriebswirtschaftliche Ziele sind z.B. das Streben nach Gewinn, Wirtschaftlichkeit oder Sicherheit. Sie sind typischerweise in ein hierarchisches Zielsystem (Kennzahlensystem) eingebettet und können in konkurrierender, komplementärer oder indifferenter Beziehung zueinander stehen [Hein91, 14, 16-20]. Zielhierarchien lassen sich als Ontologien repräsentieren und erleichtern so die Auswahl geeigneter Elemente (Abbildung 72). Mithilfe des Wertbeitrags können Problemaspekte auf solider Grundlage priorisiert oder verworfen werden, falls ihre Lösung nicht die favorisierten Geschäftsziele unterstützt. Gleichzeitig können mehrdeutige sachliche Ziele konkretisiert werden.

Die Zuordnung einer `Kulisse` erlaubt es, den jeweiligen Zustand in einen Kontext zu stellen und somit eine objektivere Einschätzung der Ausgangslage (Ist) zu erreichen bzw. Rahmenbedingungen für die zu entwickelnde Problemlösung (Soll) festzuhalten. Der Begriff Kulisse bringt zum Ausdruck, dass hier nur unmittelbar erkennbare bzw.

bereits bekannte Sachverhalte beschrieben werden. Die Erfassung weiterer Aussagen, die einen „Blick hinter die Kulissen“ erfordern, ist an dieser Stelle nicht erforderlich; sie können bei Bedarf während der Domänenanalyse ermittelt werden. Die Problembeschreibung dient einzig dem Zweck, ein solides Verständnis des zu handhabenden Problems als Basis für weitere Planungsschritte zu gewinnen.

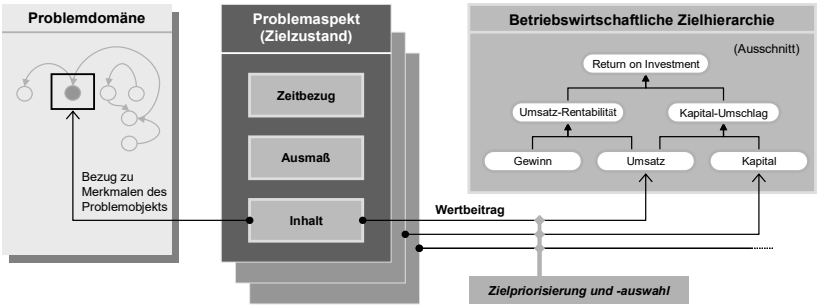


Abbildung 72: Verknüpfung von sachlichen und betriebswirtschaftlichen Zielen (eigene Darstellung)

5.4.3.4 Zusammenfassung: Identifikation eines Sachproblems

Die vorgestellten Teilaufgaben tragen gemeinsam zur Identifikation und Beschreibung eines Sachproblems bei. Sie sind nicht als kaskadische Schrittfolge zu verstehen, sondern sind eng miteinander verwoben. So beeinflusst die Diskursweltabgrenzung die Fähigkeit zur Problemerkennung und muss z.B. bei modifizierter Problembeschreibung angepasst werden. Darüber hinaus bestehen auch Interdependenzen mit der Domänenanalyse, da die dort gewonnenen Erkenntnisse Änderungen an Diskursweltabgrenzung und Problembeschreibung erfordern können. Sie sind daher im Sinne einer Checkliste abzuarbeitender Aufgaben zu interpretieren.

Im Zuge der Problemerkennung auszuführende Datenanalysen sind als **`Informationsmaßnahmen`** mit dem jeweiligen Problemaspekt zu verknüpfen, sofern sie speziell zur Identifizierung des Sachproblems dienen und nicht regelmäßig erfolgen (wie z.B. das Standardberichtswesen).

5.4.4 Domänenanalyse (Z2)

Nach Identifikation eines Sachproblems gilt es, ein tieferes Verständnis für dieses Problem zu erlangen, um effektive Lösungsoptionen entwickeln zu können. Ziel der Domänenanalyse ist somit zunächst die Sammlung und Strukturierung von Domänenwissen (vgl. [FrPM91, 11]). Dieses Wissen wird sodann zur Konkretisierung des Sachproblems in operationale Problemaspekte eingesetzt, deren Beeinflussung durch Handlungsmaßnahmen möglich ist und einen Beitrag zur Problemlösung leistet [Gait83, 66f.]. Ergebnis der Domänenanalyse ist eine *Problemkarte* (vgl. Abschnitt 4.3.1.5).

Das relevante ***Domänenwissen*** umfasst alle zweckdienlichen Aussagen über die Problemdomäne, insbesondere Kenntnisse über Struktur, Verhalten und Einflussbeziehungen der Domänenobjekte, Definitionen wichtiger Begriffe sowie die Sichtweise des Auftraggebers [KlZy96, 578]. Es soll auf Anwendungsebene insbesondere zur Entwicklung, Bewertung und Auswahl von Handlungsoptionen beitragen [GaSc88, 3]. Auf Analyseebene kann es die Formulierung der Analyseziele, die Auswahl adäquater Daten, das Untersuchungsdesign sowie die Ergebnisinterpretation unterstützen [Drei94. 135], [DeHa01, 59]. Seine Einbeziehung in die Planung von Analysevorhaben wird daher allgemein empfohlen (vgl. z.B. [Tuke62, 9], [Vell97, 321f.], [Drei94, 20]).

Die Domänenanalyse erreicht ihre Ziele insbesondere durch eine Problemstrukturierung, die aus einer näheren Beschreibung des Problemobjekts und der anschließenden Ergründung möglicher Einflussfaktoren und Problemursachen resultiert (vgl. [Drei94, 18]). Dabei ergibt sich typischerweise eine Reihe von Fragen, die von Domänenexperten oder durch Datenanalysen beantwortet werden können (vgl. [FrPM91, 12]). Die Problemkarte als Produkt der Domänenanalyse bildet demnach das konzeptuelle Gerüst, in dem Datenanalysen verankert sowie analytisch oder aus der Diskussion mit Experten gewonnene Erkenntnisse eingeordnet werden.

Domänen- und Datenanalyse stehen in einer dialektischen Interaktionsbeziehung, die auf einem fortwährenden Zusammenspiel zwischen Annahmen und Evidenzen beruht (vgl. empirischer Zyklus, Abschnitt

2.3.2.3). Die Domänenanalyse vermittelt erste, häufig noch vage Vorstellungen darüber, welche Aspekte für das vorliegende Problem relevant sind. Diese konzeptuellen Aussagen werden in empirische Aussagen übersetzt, mithilfe der Datenanalyse überprüft und sukzessive weiterentwickelt. Datenanalyseergebnisse können zu bislang nicht berücksichtigten konzeptuellen Aussagen führen, die mit Domänenexperten diskutiert werden und auf Problemaspekte oder zugehörige Lösungsoptionen verweisen können [Ehre76, 429]. Konzeptuelle und empirische Aussagen entwickeln sich im Wechselspiel voran, was in einer Fortschreibung des Domänenwissens resultiert (Abbildung 73)

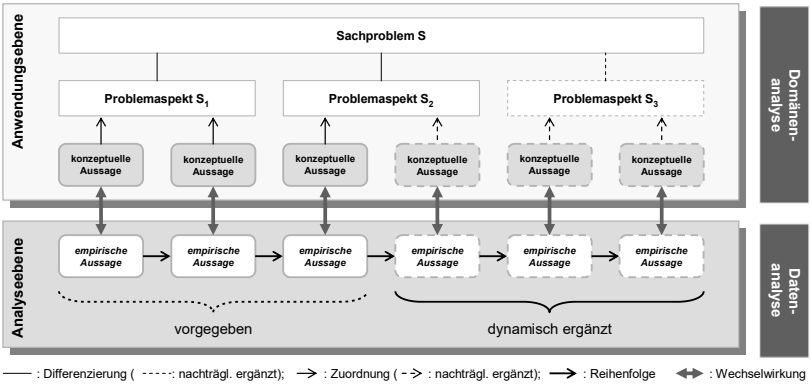


Abbildung 73: Dialektik von Domänen- und Datenanalyse zur Fortschreibung von Domänenwissen (eigene Darstellung)

Die Kombination von Domänen- und Datenanalyse ist geeignet, Problemdomänen umfassend zu erkunden. Bereits die Domänenanalyse per se kann wertvolle Erkenntnisse liefern [Pyle03, 35, 61], indem sie das im Unternehmen über die Domäne vorhandene Wissen aufnimmt und für die Problemlösung nutzbar macht. Die Problemkarte dokumentiert als relevant erachtete **`Problemaspekte`** und zeigt jeweils alle offenen Fragen und zu lösenden Teilprobleme auf [Pyle03, 64], die als **`Informations`**- und **`Handlungsmaßnahmen`** modelliert werden.

Hilfreiche Teilaufgaben zur Systematisierung der Domänenanalyse sind die *Ergründung der Sichtweise des Auftraggebers (Z2.1)*, die *Konkretisierung des Problemobjekts (Z2.2)*, die *Identifikation von Einflussfaktoren (Z2.3)*,

die *Ableitung von Handlungsoptionen (Z2.4)* sowie die *Problemkartierung (Z2.5)*. Sie werden im Folgenden vorgestellt.

5.4.4.1 Ergründung der Sichtweise des Auftraggebers (Z2.1)

Der erste Schritt der Domänenanalyse besteht in der Sammlung des über die Problemdomäne verfügbaren Wissens in Zusammenarbeit mit dem Auftraggeber bzw. mit Domänenexperten. Mit dem Zusammentragen problemrelevanter Information soll zugleich ein einheitliches Verständnis von der Problemdomäne erreicht werden.[157] Da die individuellen Wissensstände, Auffassungen und Vorstellungen der Projektbeteiligten in der Regel divergieren, sollten existierende Positionen aufgeklärt und auf eine gemeinsame Basis gestellt werden. Diese Aufgabe lässt sich am besten in Form von Gesprächen oder Diskussionsrunden bewältigen [Pyle03, 125].

Terminologie

Zur Vermeidung von Kommunikationspannen trägt eine einheitliche Definition der verwendeten Begriffe bei. Zu branchenspezifischen Fachbegriffen sind oft Glossare verfügbar, die in projektbezogene Glossare überführt werden können [CCK+00, 18]. Häufig existieren abweichende unternehmensspezifische Begriffssysteme, die sich oft erst mit ersten Missverständnissen offenbaren. Ebenso kann die Definition scheinbar trivialer Begriffe zuweilen mit erheblichen Schwierigkeiten verbunden sein.[158] Daher sollte jedes Glossar durch den Auftraggeber verifiziert werden. Zur Repräsentation eignen sich u.a. Domänenobjekt-Ontologien (vgl. Abschnitt 4.7.1.2).

---

[157] Dies ist insbesondere bei Beteiligung externer oder fachfremder Mitarbeiter oder Berater von Bedeutung.

[158] Ein Beispiel ist der Begriff *Kunde*. Bei Vertragsbeziehungen (z.B. bei Banken) besteht der Kundenstatus während der Vertragslaufzeit und erlischt mit rechtswirksamer Kündigung. In anderen Fällen ist die Frage, welche Person ein Kunde ist, schwieriger zu beantworten. Im Versandhandel etwa kann ein Kunde, der seit längerer Zeit keine Bestellung mehr getätigt hat, nur vorübergehend inaktiv oder abgewandert sein [NeKn15, 226]. Die Definition solcher Begriffe kann einen eigenen Problemaspekt darstellen.

Vorstellungen und Erwartungen

In der Regel existieren bereits Vorstellungen von möglichen Ursachen eines Sachproblems, und häufig werden Erwartungen geäußert, mit welchen Maßnahmen eine Problemlösung erreichbar ist. Diese Hypothesen stellen eine wichtige Basis für die Problemhandhabung dar, sollten jedoch stets datenanalytisch verifiziert werden. Fehlerhafte und nicht korrigierte Vorstellungen können zum Scheitern des Projekts führen [Pyle03, 9], [Milt10, 21f.].

Dies erweist sich insbesondere dann als problematisch, wenn sie nicht explizit geäußert werden und daher nur sehr schwer erkennbar sind. Implizite Annahmen verhindern den Diskurs über problemrelevante Fragen und werden – häufig unbewusst – genutzt, um Wissenslücken aufzufüllen. Diese Wissenslücken gilt es zu ergründen, da sie auf wichtige Problemaspekte hinweisen können [Milt10, 25f.]. Ansatzpunkte hierzu bietet neben detaillierten Befragungen die systematische Untersuchung der Problemdomäne mithilfe von theoretisch fundierten Modellen oder Heuristiken, wie sie in den folgenden Abschnitten erörtert werden (vgl. [Pyle03, 65f.]).

5.4.4.2 Konkretisierung des Problemobjekts (Z2.2)

Eine genauere Bestimmung der vom Sachproblem betroffenen Objektklasse liefert wichtige Aussagen über Bedingungen der Problementstehung und über Einschränkungen wirksamer Problemlösungen (vgl. [Pyle03, 63f.]). Ziel dieses Schrittes ist daher die Konkretisierung des Problemobjekts, die zu einer Einschränkung, aber auch zu einer Ausweitung der Extension des ursprünglich identifizierten Domänenobjekttyps führen kann.[159]

Da die Art der Problemobjekte (Aufträge, Kunden, Produktionsprozesse, etc.) fallspezifisch variiert, werden Richtlinien zu ihrer Konkretisierung in Form von Heuristiken angegeben. Im Grunde sind Vergleiche

[159] Vgl. hierzu HAND, der die Unterscheidung, ob Aussagen über die Grundgesamtheit oder über eine Teilmenge zu treffen sind, als wichtiges Prinzip der Problemformulierung ansieht [Hand94, 334].

anzustellen, um Unterschiede in der Gesamtpopulation der in Frage kommenden Problemobjekte aufzudecken. Als mächtiges Hilfsmittel erweisen sich dabei W-Fragen, die sich prinzipiell auf alle Situationen anwenden lassen (Abbildung 74). So verweist etwa die Frage „Wer und wo?“ direkt auf die Extension des Problemobjekts:

- ***Wer** ist betroffen? (Welche Objekte sind betroffen?)* Diese Frage richtet sich primär auf Objektmerkmale. Im Falle von Kunden kann z.B. nach Alter, Geschlecht, Bonität oder Wohnort gefragt werden.
- ***Wo** ist das Problem aufgetreten?* Diese Frage zielt auf den Eintrittsort problemrelevanter Ereignisse, wobei der Ortsbegriff sowohl geographisch als auch organisatorisch zu verstehen ist. So kann etwa nach Filialen, Ländern, Produktionsstätten oder Fertigungsstufen differenziert werden. Der Wohnort des Kunden fällt als Objektmerkmal nicht unter diesen Aspekt, die betroffene Vertriebsregion als Ereignismerkmal hingegen schon.

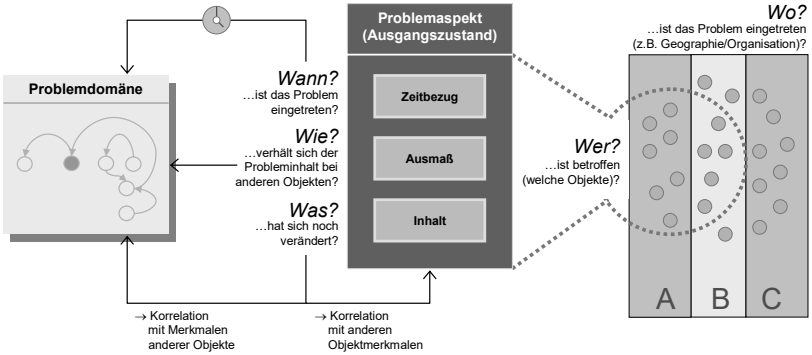


Abbildung 74: Konkretisierung des Problemobjekts (eigene Darstellung)

Weitere Kriterien ergeben sich aus den Beschreibungsdimensionen des Problemzustands, welche die Fragen „Was, wie, wann?“ beantworten und wie folgt erweitert werden können:

- ***Was** hat sich **noch** verändert?* Diese Frage richtet sich auf Korrelationen mit anderen Objektmerkmalen bzw. mit Merkmalen anderer Objekte, mit denen das Problemobjekt interagiert. Werden z.B. mit

einer Kundenabwanderung auch Umsatzrückgänge beobachtet, ergeben sich Hypothesen über Problembedingungen (z.B. Anzeichen einer bevorstehenden Kündigung).

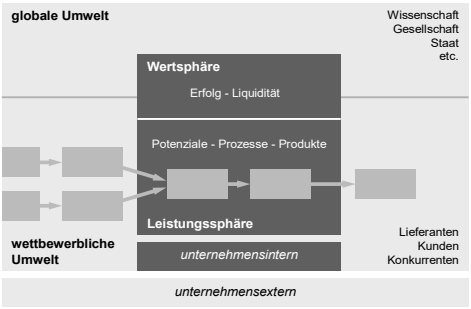
- ***Wie** verhält sich der Probleminhalt **bei anderen** Objekten?* Diese Frage stellt Vergleiche bezüglich des problematischen Objektmerkmals an. Sie können sich auf die Gesamtpopulation (z.B. alle Kunden), bestimmte Teilpopulationen (z.B. Kunden benachbarter Vertriebsregionen) oder vergleichbare andere Objekte (z.B. Kunden im Gesamtmarkt) richten und Hypothesen erhärten, entkräften oder zu neuen Vermutungen Anlass geben.
- ***Wann** ist das Problem eingetreten, und was ist zeitgleich passiert?* Der Zeitbezug ist bei allen Fragen zu berücksichtigen und kann entweder in Vergleiche mit anderen Objekten bzw. Merkmalen innerhalb desselben Zeitraums oder in Vergleiche derselben Objekte bzw. Merkmale mit anderen Zeiträumen münden. Ebenso können Ereignisse außerhalb der Diskurswelt (z.B. Gesetzesänderungen, Naturkatastrophen) oder wichtige Termine (z.B. Beginn der Osterferien) auf potenzielle Einflussfaktoren oder Problemursachen deuten.

Zur Beantwortung solcher Fragen sind in vielen Fällen Datenanalysen geeignet (Informationsmaßnahmen). Die Konkretisierung des Problemobjekts fokussiert die Betrachtung auf jene Objekte, die tatsächlich vom Problem betroffen sind, und trägt somit zur Validität der Ursachenanalyse bei. Die Festlegung des Problemobjekts ist indes nicht endgültig; vielmehr kann sie auf Basis neuer Erkenntnisse modifiziert werden.

5.4.4.3 Identifikation von Einflussfaktoren (Z2.3)

Bestandteil eines umfassenden Problemverständnisses ist insbesondere die Kenntnis wichtiger Einflussfaktoren auf die Problemdomäne, da diese einerseits Ursache des Problemzustands sein können und andererseits Optionen zu dessen zielgerichteter Beeinflussung im Sinne einer Lösungsoption darstellen. Erste Hypothesen hierzu liefert bereits

der vorausgehende Schritt. Dieser Abschnitt zeigt weitere Möglichkeiten zur systematischen Suche nach Einflussgrößen.



Beispielhafte Modelle zur systematischen Identifizierung von Einflussfaktoren:

- Unternehmen als Input-Output-System
- Betriebliche Wertschöpfungskette
- Supply Chains, Liefernetze
- Differenzierung in Unternehmen (interne Faktoren) und Umwelt (externe Faktoren)
- Differenzierung interner Faktoren in Leistungssphäre und Wertsphäre
- Differenzierung externer Faktoren in globale und wettbewerbliche Umwelt

Abbildung 75: Beispielhafte Modelle zur Identifikation von Einflussfaktoren (eigene Darstellung, vgl. [Pyle03, 216ff.], [Beck96, 73f.])

Aufgrund der Vielfalt potenzieller Einflussfaktoren, wie sie die Komplexität der Diskurswelt eröffnet, werden gedankliche Schemata oder Modelle zur Strukturierung der Untersuchung empfohlen [Pyle03, 216ff.]. Einige Beispiele zeigt Abbildung 75. Die Perspektiven der Balanced Scorecard nach KAPLAN & NORTON [KaNo96] spiegeln wichtige Aspekte dieser Gliederungen wider. Der Ansatz stützt sich explizit auf die Untersuchung von Ursache-Wirkungs-Beziehungen, kann um weitere Perspektiven erweitert werden und erscheint aufgrund seiner Verbreitung in der Praxis als geeignete Möglichkeit zur Orientierung bei der Domänenanalyse. Häufig sind domänenspezifische Schemata aufgrund ihrer Fähigkeit zur Abbildung individueller Eigenheiten besser geeignet. Auch Geschäftsprozessmodelle können als Hilfestellung dienen. PYLE schlägt physische Systeme als Gliederungsmetaphern vor [Pyle03, 187f.].

Die genannten Gliederungsprinzipien betreffen zunächst die Struktur der Diskurswelt. Zur Identifikation von Einflussfaktoren sind jeweils die Verhaltensweisen der Objekte innerhalb der resultierenden Teilsysteme zu analysieren. Neben Ereignissen, die sich als Ursache oder Wirkung einer Interaktion äußern, können für Systeme allgemein z.B. Bestände, Flüsse, Kapazitäten, Durchlauf- und Wartezeiten als Ansatzpunkte für die verhaltensbezogene Untersuchung dienen [Pyle03, 176ff., 233,

187ff.]). Zu beachten sind stets auch die implementierten Strategien und Geschäftsregeln, die dem Systemverhalten bewusst Beschränkungen auferlegen, als potenzielle Problemursachen aber leicht übersehen werden.

Erkannte Einflussfaktoren sollten auf adäquate Weise dokumentiert werden. Eine pragmatische Möglichkeit sind so genannte *Mindmaps* [Pyle03, 196f.], die Assoziationen zwischen Konzepten grafisch visualisieren und intuitiv verständlich sind. Aus diesem Grund sind sie gerade für die Ideensammlung in Gruppensitzungen geeignet. Mehr Semantik beinhalten z.B. *Ursache-Wirkungs-Diagramme* nach ISHIKAWA,[160] die einer Wirkung in einer Baumstruktur mehrere Ursachen zuordnen, die ihrerseits durch andere Faktoren verursacht werden. Die Zuordnung weiterer Einflussfaktoren kann fortgesetzt werden, bis die Blattknoten messbare Größen repräsentieren [Pyle03, 248-251]. Diese lassen sich bei Bedarf datenanalytisch untersuchen.

Allerdings sind derartige Diagramme weder geeignet, die Wechselwirkungen der Einflussgrößen abzubilden, noch die Gültigkeit der vermuteten Einflussbeziehungen festzustellen. Hierzu sind fortgeschrittene Methoden erforderlich, wie etwa *System-Dynamics-Modelle* nach FORRESTER [Forr61], die Zustandsänderungen als Ereignisse darstellen, welche in Bezug auf eine Beziehung entweder als Ursache oder als Wirkung auftreten. Eine umfassende Methodik zur Entwicklung und Überprüfung von Hypothesen über kausale Zusammenhänge präsentiert VOIT [Voit10]. Um einen Kausalzusammenhang als gültig zu akzeptieren, sind dessen Chronologie, Gesetzmäßigkeit und Bedingtheit nachzuweisen [Voit10, 100]. Die *Chronologie* setzt die Ursache zeitlich stets vor die Wirkung. Die *Gesetzmäßigkeit* impliziert, dass mit jedem Eintreten des Ursachenereignisses auch die Wirkung eintritt.[161] Die *Bedingtheit* erfordert gewisse Rahmenbedingungen im Sinne einer kausalen Infrastruktur, ohne die ein Ereignis nicht Ursache

[160] ISHIKAWA-Diagramme sind wegen ihrer Form auch als „Fischgrät-Diagramme" bekannt.

[161] In sozialen Systemen ist hauptsächlich von stochastischen Gesetzmäßigkeiten auszugehen, die einer Ursache mit gewisser Wahrscheinlichkeit eine Wirkung folgen lassen [Voit10, 75].

eines anderen Ereignisses sein kann. Diese Infrastruktur besteht aus Objekten, die Gegenstand der betreffenden Zustandsänderungen sind (Ereignisträger), sowie aus Beziehungen, die geeignet sind, die kausal verknüpften Ereignisse zwischen den betroffenen Objekten zu transportieren (vgl. [Voit10, 102f., 112]). Die fundierte Untersuchung von Einflussbeziehungen geschieht typischerweise mithilfe der Datenanalyse. Im Sinne einer konsistenten Modellierung sollte zur Repräsentation der Objekte derselbe Ansatz verwendet werden wie zur Diskursweltabgrenzung (Abschnitt 5.4.3.2).

Das Denken in systemdynamischen Zusammenhängen ermöglicht die Erkennung von Wirkweisen und Erklärungen, die für eine erfolgreiche Problemlösung entscheidend sein können. Die Entdeckung neuer Einflussfaktoren und Kausalzusammenhänge erfordert in der Regel eine Anpassung der Diskursweltabgrenzung.

5.4.4.4 Ableitung von Handlungsoptionen (Z2.4)

Die bisherigen Erkenntnisse legen das Fundament für die Suche nach einer Problemlösung, die von der Entscheidungstheorie (vgl. Abschnitt 5.4.2.2) methodisch geleitet wird: Demnach ist aus einer Menge von Handlungsoptionen eine Alternative auszuwählen, die vor dem Hintergrund nicht beeinflussbarer Umweltzustände zielkonforme Handlungskonsequenzen produziert. Während sich Auswahlkriterien aus der Definition des Zielzustand ergeben und Umweltzustände in den vorhergehenden Schritten identifiziert werden, sind die Ableitung von Handlungsoptionen und die Prognose der Handlungskonsequenzen noch zu bewältigen.

Die Entwicklung geeigneter Lösungsoptionen erfolgt typischerweise im Rahmen eines kreativen, interaktiven Prozesses. Da die Suche nach optimalen Lösungen oft nicht realistisch ist, können heuristische Systematisierungen eingesetzt werden [Milt10, 233-246]. Der folgende Vorschlag orientiert sich an allgemeinen Strategien der Komplexi-

tätsbewältigung[162] und benennt drei allgemeingültige Ansatzpunkte (Abbildung 76): Einerseits kann der Zustand des Problemobjekts direkt manipuliert werden, andererseits können kausale Beziehungen ausgenutzt werden, um diesen Zustand indirekt zu beeinflussen. Dies kann erstens durch Unterdrückung der Ursachen des unerwünschten Ausgangszustands und zweitens durch Begünstigung der Ursachen des angestrebten Zielzustands geschehen.[163] Auch Kombinationen mehrerer Maßnahmen sind möglich. Konkrete Beispiele werden im Rahmen der Fallstudie in Kapitel 8 gezeigt.

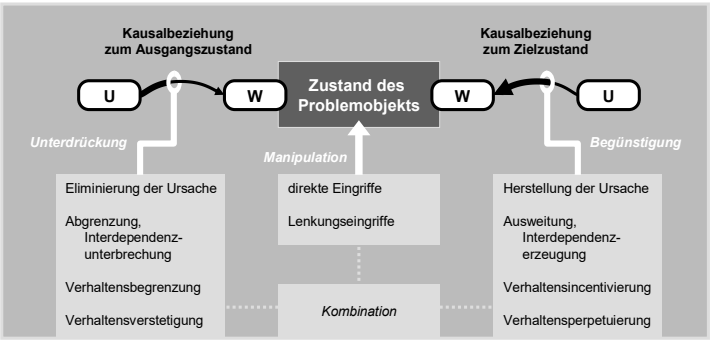


Abbildung 76: Ansatzpunkte zur Ableitung von Handlungsoptionen (eigene Darstellung)

Auch für den Einsatz einer Balanced Scorecard wird empfohlen, Lösungsmaßnahmen strukturiert aus Kausalbeziehungen abzuleiten [KaNo96, 149]. PYLE stellt eine Liste generischer Maßnahmen vor, die auf gängige Stellgrößen des Managements (z.B. Durchlaufzeit, Wartezeit, Flusskapazität, Prozessvariabilität, Prozesseffizienz) Bezug nehmen, und empfiehlt zur Reduzierung der Durchlaufzeit etwa die Streichung von Aufgaben auf dem kritischen Pfad [Pyle03, 234ff.]. Da

[162] Strukturorientiert sind dies insbesondere die Selektivierung (Systemabgrenzung) und Strukturierung (z.B. Interdependenzunterbrechung) [Bron92, 1123f.], verhaltensorientiert sind innerhalb des Unternehmens direkte Lenkungseingriffe [Mali00, 173] und die Verhaltensnormierung [Mali00,182] sowie außerhalb des direkten Einflussbereichs die Verhaltensverstetigung zu nennen [Bron92, 1129].

[163] Hierbei werden Kausalbeziehungen im Sinne von Zweck-Mittel-Beziehungen instrumentalisiert.

die ermittelten Lösungsoptionen häufig noch unspezifisch sind, müssen sie durch Zerlegung in Teil- oder Subalternativen konkretisiert werden [Wild74, 72f.].[164]

Die einzelnen Handlungsoptionen unterscheiden sich typischerweise in ihrem Zielerreichungsgrad, d.h. in der Fähigkeit, das Problem wirksam und rechtzeitig zu lösen. Realisierbarkeit und Effektivität einzelner Maßnahmen sind zudem häufig von bestimmten Bedingungen oder Ereignissen abhängig, über deren Eintreten oft nur unsichere Aussagen möglich sind [Wild74, 70-75]. Schließlich verbraucht ihre Realisierung gewisse Ressourcen und benötigt Zeit, um wirksam zu werden. Zielerreichung, Rechtzeitigkeit, Zeit- und Ressourcenbedarf fließen daher in eine Nutzen-Kosten-Abwägung bezüglich des Zielzustands (lösungsbezogener Problemaspekt) ein, auf deren Basis die Entscheidung für eine Option getroffen wird (Abbildung 77).

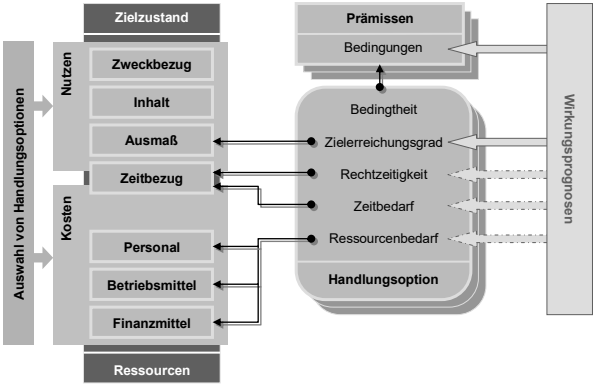


Abbildung 77: Auswahl von Handlungsoptionen (eigene Darstellung)

Der Zielerreichungsgrad und die Entwicklung der Umweltbedingungen sowie weitere Faktoren können Gegenstand von Datenanalysen sein [Wild74, 70], die eine realistischere Einschätzung der zu erwartenden Handlungskonsequenzen sowie der mit einer Maßnahme verbundenen Risiken erlauben. Sie erfolgen häufig in Form modellbasierter Pro-

[164] WILD unterscheidet zwischen sachlicher und zeitlicher Mehrstufigkeit, die zu Alternativenhierarchien bzw. Alternativenfolgen führen [Wild74, 72f.].

gnosen. Gerade über komplexe, nicht-lineare Systeme können verlässliche Aussagen jedoch oft nur simulationsbasiert getroffen werden [Pyle03, 200f.]. Daneben lässt sich die Wirksamkeit einer Maßnahme zuweilen auch experimentell überprüfen [Milt10, 54]. Hierzu wird die Maßnahme auf eine Teilpopulation der betroffenen Domänenobjekte angewandt und anschließend evaluiert. Typische Beispiele hierfür sind Werbeaussendungen an eine kleine Kundenstichprobe oder A/B-Tests für die Neugestaltung von Online-Plattformen.

5.4.4.5 Problemkartierung (Z2.5)

Die Ergebnisse der bisherigen Schritte werden in einer Problemkarte dokumentiert, die während des gesamten Projekts fortgeschrieben wird. Sie zeigt einerseits die Strukturierung des Sachproblems in handhabbare Problemaspekte (Problemstruktursicht), andererseits die zur Lösung einzelner Problemaspekte auszuführenden Maßnahmen (Problemlösungssicht). Alle während der Domänenanalyse identifizierten Fragestellungen werden als Informationsmaßnahmen, die in Schritt Z2.4 ausgewählten Handlungsoptionen als Handlungsmaßnahmen modelliert. Sie sollen in ihrer Gesamtheit zur Lösung des Sachproblems beitragen.

Aufgrund der initialen Vagheit des Sachproblems erfolgt die Problemkartierung nicht logisch-analytisch, sondern argumentativ [Gait83, 70]. Zur ihrer Unterstützung sind folgende heuristische Differenzierungsprinzipien geeignet, die einzeln oder kombiniert zum Einsatz gelangen und das Vorgehen bei der Domänenanalyse leiten können:

- *Barriereprinzip*: Differenzierung eines Problemaspekts anhand der Barriere, die bei der Problemlösung zu überbrücken ist. Sie führt zu je einem situations- bzw. lösungsbezogenen Problemaspekt und ist als erster Differenzierungsschritt allgemein zu empfehlen, eignet sich aber ebenso zur Differenzierung von Teilaspekten.
- *Objektprinzip*: Differenzierung eines Problemobjekts nach Domänenobjekten bzw. -objekttypen. Dieses Prinzip korrespondiert mit der Konkretisierung des Problemobjekts (Abschnitt 5.4.4.2), ist aber ebenso anwendbar auf Domänenobjekte, die nicht unmittelbar

Problemgegenstand sind. Ein Beispiel ist die getrennte Untersuchung des Kundenverhaltens nach bestimmten Produktgruppen (Problemobjekt Kunde, Differenzierungsobjekt Produkt).

- *Systemprinzip*: Differenzierung eines Problemaspekts durch Zerlegung eines Systems in interagierende Komponenten. Dieses Prinzip korrespondiert mit der Identifikation von Einflussfaktoren (Abschnitt 5.4.4.3) und zielt nicht auf die bloße Auflösung eines Objekts in seine Bestandteile, sondern insbesondere auf problemrelevante Interaktionsbeziehungen.
- *Phasenprinzip*: Differenzierung eines Problemaspekts in sequenziell abzuarbeitende Unterprobleme, die den Phasen eines Ablaufschemas entsprechen. Beispiele für allgemeingültige Ablaufschemata sind der Managementzyklus (Planung, Steuerung, Kontrolle), der Entscheidungsprozess (Situationsanalyse, Prognose, Entscheidung) oder die medizinische Behandlung (Untersuchung, Diagnose, Therapie). Ein domänenspezifisches Schema sind z.B. die analytischen Querschnittsaufgaben des Customer Relationship Managements (Kundenbewertung und -profilerstellung, Kundensegmentierung, Marktbearbeitung) [NeKn15, 179f.].

Die Problemkartierung kann von früheren Erfahrungen bei der Lösung ähnlicher Probleme profitieren. Bewährte Strukturmuster oder Problemkarten sind als Vorlage aus der Fallbibliothek abrufbar. Die Suche nach passenden Vorlagen kann insbesondere nach der `Kennzeichnung` des Problemaspekts sowie nach Erfolgs- oder Kontextmerkmalen der Maßnahme (z.B. `Branche`, `Betriebstyp`) erfolgen. Jedem Problemaspekt lassen sich über `Links` Dokumente und Diagramme (z.B. Mindmaps, Kausalmodelle) zuordnen, die bei der Domänenanalyse erstellt wurden.

5.4.4.6 Zusammenfassung: Domänenanalyse

Die Domänenanalyse stellt ein Hilfsmittel zur Entwicklung, Dokumentation und Erklärung des Vorgehens bei der Lösung eines Sachproblems dar (Handlungsmaßnahmen), sowohl gegenüber dem Auftraggeber als

auch als Referenz für spätere Problemfälle. Sie bildet zugleich den Orientierungsrahmen für die Datenanalyse (Informationsmaßnahmen).

Die Problemkartierung Z2.5 als Hauptaufgabe der Domänenanalyse wird durch die Aufgaben Z2.1-Z2.4 methodisch unterstützt. Das Vorgehen ist stark iterativ, da jederzeit Rücksprünge zu bereits bearbeiteten Aufgaben auftreten können, um Ergänzungen und Korrekturen vorzunehmen. Die Domänenanalyse ist auch mit der Identifikation des Sachproblems (Z1) rückgekoppelt, da ihre Ergebnisse die Diskurswelt-abgrenzung und die Problembeschreibung beeinflussen und somit Iterationen der Schritte Z1.2 und Z1.3 erfordern können.

Die Erstellung eines umfassenden Domänenmodells ist ausdrücklich nicht Ziel der Domänenanalyse, da die damit verbundene Komplexität nicht verhältnismäßig erscheint und die Akzeptanz des Ansatzes gefährdet. Vielmehr wird mit der Problemkarte ein projektgebundenes, überschaubares und intuitiv verständliches Schema erstellt, dessen Konstruktion mit Instrumenten erfolgen sollte, die an die Präferenzen des Projektteams angepasst sind. Gleichwohl können über mehrere Projekte inkrementell erarbeitete Domänenontologien entstehen, die im Laufe der Zeit eine gute Abdeckung der Domäne erreichen.

5.4.5 Spezifikation des Analyseproblems (Z3)

Informationsbedarfe, die sich aus einem Problemaspekt ergeben, werden in der Problemkarte jeweils durch Zuordnung einer Informationsmaßnahme modelliert. Für jede Informationsmaßnahme, zu deren Realisierung eine Datenanalyse vorgesehen ist, wird die Planung auf Analyseebene fortgesetzt. Handlungsmaßnahmen sind separat zu behandeln und werden nicht weiter betrachtet. Ziel dieses Planungsschrittes ist die Spezifikation eines oder mehrerer Analyseprobleme für jede Informationsmaßnahme. Ihr Ergebnis ist entsprechend eine ein- oder mehrgliedrige *Analysekette*.

Eine Analysekette bestimmt die im Rahmen einer umfassenderen Untersuchung abzuarbeitenden Analyseschritte im Sinne einer Analysestrategie. Sie wird metaphorisch als Besichtigungstour innerhalb der Problemkarte beschrieben (vgl. Abschnitt 4.4.5), in deren Verlauf

Kenntnisse über das Untersuchungsobjekt erworben werden, die zur Deckung des sachlichen Informationsbedarfs beitragen. Dessen konzeptuelle Aussagen werden in empirische Aussagen transformiert und jeweils in Form eines **`Analyseziels`** festgehalten (empirischer Informationsbedarf). Mit Festlegung geeigneter Analysedaten für ein Analyseziel entsteht jeweils ein **`Analyseproblem`**, das eine auszuführende Datenanalyse im Sinne einer Transformation von Analysedaten in Informationen spezifiziert (vgl. Abschnitt 4.4.2). Analyseziele und -probleme können eine Konkretisierung erfahren. Dies kann eine Strukturierung oder Neuordnung der Analysekette erfordern. Die drei Teilaufgaben der Spezifikation eines Analyseproblems sind demnach die *Formulierung des Analyseziels (Z3.1)*, die *Formulierung des Analyseproblems (Z3.2)* sowie die *Konkretisierung und Strukturierung von Analysezielen (Z3.3)*.

5.4.5.1 Formulierung des Analyseziels (Z3.1)

Das Analyseziel beschreibt, welche Informationen die Datenanalyse als Ergebnis liefern soll. Dieser Informationsbedarf richtet sich auf Eigenschaften eines Domänenobjekts, die zunächst zu bestimmen sind, und muss gewissen Anforderungen genügen, um zur Lösung des Problemaspekts auf Anwendungsebene beitragen zu können.

Bestimmung des Untersuchungsproblems

Der in der Informationsmaßnahme skizzierte Informationsbedarf ist zunächst als *Untersuchungsproblem* auf konzeptueller Ebene zu präzisieren. Es wird als `Merkmal` eines `Domänenobjekts` angegeben.[165] Beispielsweise zielt eine Informationsmaßnahme zur Ermittlung der Kundenzufriedenheit auf das Merkmal Zufriedenheit des Untersuchungsobjekts Kunde. Wird auf die Spezifikation eines Sachproblems verzichtet, stellt das Untersuchungsproblem den Einstieg in die systematische Analyseplanung dar. Da das Untersuchungsziel häufig konzeptuelle Konstrukte beinhaltet, die nicht direkt messbar sind (wie

[165] Das initiale Untersuchungsziel entspricht dem Problemmerkmal des zugehörigen Problemaspekts.

etwa Zufriedenheit), muss es in empirische Aussagen transformiert werden. Dies geschieht durch Operationalisierung.

Operationalisierung

In der empirischen Forschung existiert eine umfassende Theorie zur Operationalisierung, die beschreibt, wie nicht beobachtbaren Begriffen (hypothetischen Konstrukten, latenten Variablen) über Korrespondenzhypothesen empirisch beobachtbare *Indikatoren* (manifeste oder statistische Variable, Items) zugeordnet werden (vgl. z.B. [Drei94, 76ff.], [EnMT95, 5ff.]). Ihr Ergebnis ist ein so genanntes *Messinstrument*, von dem angenommen wird, dass es den zu untersuchenden Sachverhalt adäquat erfasst. Indikatoren werden in drei Klassen gegliedert. Definitorische Indikatoren erlauben eine eindeutige Operationalisierung, da sie den konzeptuellen Begriff determinieren (z.B. definieren Gewinn und Umsatz die Kennzahl Umsatz-Rentabilität). Korrelative Indikatoren stehen in Korrelation mit dem konzeptuellen Begriff, repräsentieren diesen aber in der Regel nicht vollständig (z.B. korreliert die Wiederkaufrate mit Kundenzufriedenheit). Schlussfolgernde Indikatoren lassen auf Aspekte des konzeptuellen Konstrukts schließen (z.B. Weiterempfehlungsbereitschaft auf Kundenzufriedenheit) [Drei94, 75]. Grundsätzlich gibt es mehr als eine mögliche Operationalisierung eines Begriffes, und häufig ist es angebracht, einen Begriff durch mehrere Indikatoren zu operationalisieren, um eine präzise Beschreibung aller Begriffsdimensionen zu erreichen [Benn94, 11], [Drei94, 78].

Als Gütekriterien der Operationalisierung gelten ein adäquates Skalenniveau sowie möglichst hohe Validität (Abwesenheit systematischer Messfehler) und Reliabilität (Abwesenheit unsystematischer Messfehler) [Drei94, 81-88]. Da mit dem *Skalenniveau* der Indikatoren sowohl der Aussagegehalt als auch die Zahl anwendbarer Analyseverfahren steigen, sollte dieses möglichst hoch angelegt sein. Es muss jedoch mindestens dem Begriffstypus der konzeptuellen Variablen entsprechen [Drei94, 54f., 64], [WeKr10, 77].[166] Die *Validität* einer Operationalisierung bezieht

[166] Für qualitative (klassifikatorische und komparative) Begriffe sollte eine Nominal- oder Ordinalskala, für quantitative (metrische) Begriffe eine Intervall- oder Verhältnisskala gewählt werden [Drei94, 47, 63f.]

sich auf die Frage, ob die gewählten Indikatoren den konzeptuellen Begriff tatsächlich repräsentieren. In der Praxis gestaltet sich die valide Operationalisierung schwierig und kann oft erst nachträglich beurteilt werden. Selbst wenn von Validität auszugehen ist,[167] kann die Operationalisierung zu weit oder zu eng gefasst sein. Im ersten Fall misst der Indikator mehr als gewünscht (z.B. allgemeine Zufriedenheit statt Zufriedenheit mit dem Kundendienst), im zweiten Fall misst er zu wenig. Als Richtlinie gilt, dass die Validität einer Operationalisierung umso höher ist, je besser alle Ausprägungen des konzeptuellen Merkmals von den Indikatoren abgedeckt werden, je weniger die empirischen Werte durch externe oder Störfaktoren verfälscht werden, und je mehr Indikatoren parallel eingesetzt werden (konzeptuelle Replikation) [WeKr10, 76]. Als *Reliabilität* (Messstabilität) wird die Reproduzierbarkeit bzw. Konsistenz von Messergebnissen bei wiederholter Anwendung desselben Messinstruments bezeichnet [Drei94, 88], [SePr10, 140].[168] Im Allgemeinen ist die Reliabilität umso höher, je weniger subjektive Einflüsse auf die Messwerte einwirken (*Objektivität*), je homogener die Indikatoren und je gleichmäßiger ihre Werte verteilt sind [WeKr10, 78].

Die Operationalisierung erfolgt prinzipiell ohne Bezugnahme auf verfügbare Datenquellen. Die gewählten Indikatoren stellen vielmehr eine Grundlage für die Auswahl bzw. Erhebung geeigneter Daten dar [Drei94, 77]. Als Indikatoren kommen demnach alle Merkmale in Frage, die *grundsätzlich messbar* oder aus messbaren Merkmalen berechenbar sind. Gleichwohl können Kenntnisse über vorhandene Datenbestände nutzbringend in die Operationalisierung einfließen. Indikatoren sind stets Eigenschafts- oder Beziehungsmerkmale des Untersuchungs-

[167] Theoretisch liegt Validität bei Korrelation der Messwerte mit den korrespondierenden wahren Werten vor (inhaltliche Validität). Da letztere unbekannte konzeptuelle Größen sind, lässt sich Validität nur mithilfe anderer Messungen abschätzen. Vgl. hierzu im Detail z.B. [Drei94, 85-87], [WeKr10, 77f.].

[168] Sie entspricht dem Varianzenverhältnis der unbeobachtbaren wahren Werte und der beobachtbaren Messwerte. Aufgrund der Unbekanntheit der wahren Werte kann auch die Reliabilität nur empirisch geschätzt werden, z.B. mittels mehrerer paralleler Messungen [WeKr10, 78].

objekts (vgl. [Voit10, 121-123]). Eigenschaftsmerkmale kennzeichnen ein Objekt bei isolierter Betrachtung, wie z.B. Alter eines Kunden oder Datum eines Auftrags. Beziehungsmerkmale beschreiben Interaktions- oder Aggregationsbeziehungen eines Objekts zu anderen Objekten, wie etwa die einem Kunden zugeordneten Aufträge oder die zu einem Auftrag gehörenden Auftragspositionen. Insbesondere können neben originären Merkmalen auch abgeleitete Merkmale betrachtet werden, die – gegebenenfalls transitiv – über Beziehungen zu anderen Objekten ermittelbar sind (z.B. können einem Auftrag Alter und Wohnort des zugehörigen Kunden zugeschrieben werden). Zur Orientierung bei der Indikatorwahl kann eine Domänenobjekt-Ontologie eingesetzt werden.

Es empfiehlt sich, für häufig auftretende Untersuchungsziele eine einheitliche Operationalisierung zu entwickeln oder auf etablierte Indikatoren zurückzugreifen [Drei94, 81], um die Konsistenz und Vergleichbarkeit mehrerer Analysen sicherzustellen. Erprobte Vorschläge in Form von Kennzahlen liefert die Controlling-Literatur wie z.B. REICHMANN [Reic01], der zahlreiche Messinstrumente für alle wichtigen betrieblichen Funktionsbereiche vorstellt.

Formulierung der Analysefrage

Das Ergebnis der Operationalisierung wird als `Analysefrage` dokumentiert. Die Dokumentation dient im Rahmen der Planung als Grundlage zur Auswahl von Analysedaten und zur Gestaltung eines Analyseprozesses, sowie im Rahmen der Revision der Unterstützung bei der Interpretation der Analyseergebnisse und im Falle unbefriedigender Ergebnisse bei der Suche nach möglichen Ursachen (vgl. [Benn94, 11f.], [SePr10, 14]). Mehrere Indikatoren, die aus der Operationalisierung resultieren, können entweder gemeinsam in einer oder separat in mehreren Analysefragen münden. Im zweiten Fall entstehen mehrere Analyseziele.

Die Analysefrage wird einerseits in Form eines natürlichsprachigen `Fragetexts`, andererseits in Form eines strukturierten Schemas mit den Elementen `Aussagetyp`, `Argumente`, `Beschreibungsdimensionen` und `Selektionsdimensionen` repräsentiert (Fragestruktur). Die Formulierung als natürlichsprachige Frage erleichtert die Kommuni-

kation und erweist sich einerseits bei der Anforderungsanalyse, andererseits zum Verständnis der von einer Analyse gelieferten Information bei der späteren Anwendung des Wissens hilfreich. Hierzu ist es nicht erforderlich, in den Fragetext alle Einzelmerkmale aufzunehmen; vielmehr ist eine prägnante Formulierung zu favorisieren (z.B. „Umsatz nach Kundenmerkmalen" anstelle der Aufzählung aller in einem umfangreichen Bericht enthaltenen Merkmale). Die vollständige Nennung aller Merkmale erfolgt mit den Argumenten bzw. Dimensionen des Frageschemas. Die Fragestruktur und mögliche Ausprägungen des Aussagetyps sind in Abschnitt 4.4.1.1 ausführlich erläutert.

Ihre Elemente sind im Sinne einer Checkliste zu verstehen, mit deren Hilfe die umfassende Spezifikation des Informationsbedarfs gemäß seiner wesentlichen inhaltlichen Bestimmungsfaktoren unterstützt werden kann. Als Selektionsdimension kann die Objektklasse des Untersuchungsobjekts, als Aussageargument das Untersuchungsziel vorbelegt werden. Dennoch lässt sich ein Informationsbedarf durch mehrere, unterschiedlich formulierte Fragen ausdrücken [Hogl03, 32]. Gerade der Hypothesenbezug einer Frage wird häufig nicht gemäß den statistischen Prinzipien als Wenn-dann- oder Je-desto-Aussagen über Zusammenhänge formuliert (vgl. [Drei94, 20]), sondern vielmehr implizit ausgedrückt.[169] Zur Vermeidung von Unklarheiten wird daher die `Ausrichtung` einer Analyse eigens gekennzeichnet.

Informationsbedarfsprofil

Analysefrage und -ausrichtung beschreiben den Inhalt der gewünschten Information und definieren gewissermaßen das Sachziel der Datenanalyse. Um ihre Eignung zur Lösung des Sachproblems zu gewährleisten, sind weitere Anforderungen zu definieren. Sie können aus Eigenschaften des Informationsbedarfs abgeleitet werden, die sich in die

[169] So werden explorative Analysen oft als (offene) Ergänzungsfragen und konfirmative Analysen als (geschlossene) Entscheidungsfragen formuliert. Zum Beispiel zielt die offene Frage, „Welcher Zusammenhang besteht zwischen Kundenalter und Auftragsvolumen?" auf Aussagen über bislang nicht bekannte, hypothetische Beziehungen, während die geschlossene Frage „Gibt es einen Zusammenhang...?" auf eine Hypothesenprüfung abzielt.

Klassen Art, Menge, Qualität und Nutzen gliedern. Sie spezifizieren Formalziele der Datenanalyse, die bei der Bewertung der Analyseergebnisse berücksichtigt werden. Sie setzen zugleich Rahmenbedingungen für die Auswahl von Analyseverfahren und für die Planung des Analyseprozesses.

Ein Katalog möglicher Deskriptoren mit Erläuterungen findet sich in Anhang A5.1. Er ist im Sinne einer Checkliste zu verstehen, die auf potenziell wichtige Kriterien zur Spezifikation des Informationsbedarfs hinweisen soll. Für eine konkrete Analyse sind jeweils jene Kriterien zu wählen, die situativ von Bedeutung sind. Einige Beispiele sind in Abschnitt 4.4.1.2 genannt.

5.4.5.2 Formulierung des Analyseproblems (Z3.2)

In diesem Schritt werden geeignete Daten zur Beantwortung der Analysefrage spezifiziert. Hierzu sind zunächst geeignete Datenquellen zu identifizieren und auszuwählen. Abschließend sind für die gewählten Quellen jene Informationsobjekte zu bestimmen, welche die in der Analysefrage enthaltenen empirischen Aussagen enthalten. Sie werden dem Analyseziel in Gestalt des Analyseobjekts zugeordnet und erweitern ersteres zu einem Analyseproblem.

Identifikation von Datenquellen

Zunächst ist zu prüfen, ob auf bereits vorhandene Sekundärdaten zurückgegriffen werden kann, oder ob neue Primärdaten zu erheben sind. Primärdaten können zwar genau auf das Analyseziel zugeschnitten werden, ihre Erhebung ist jedoch teuer und zeitaufwändig, was Verzögerungen bei der Informationsbereitstellung herbeiführen kann. Daher wird Sekundärdaten meist der Vorzug gegeben, die jedoch potenziell ein Adäquationsproblem bergen (vgl. Abschnitt 3.1.2.2), z.B. aufgrund ihres Inhalts, Alters, ihrer Repräsentation oder fehlender Historisierung [Pyle03, 223f.].

Die Identifikation und Auswahl von Sekundärdatenquellen wird durch die semantische Annotation der Informationsobjekte mit den repräsentierten Domänenobjekten bzw. -merkmalen (vgl. Abschnitt 4.6.1.2)

methodisch unterstützt. Hierzu werden alle Informationsobjekte, die mit dem `Domänenobjekt` des Analyseziels annotiert sind, ermittelt und nach Datenquellen geordnet. Abbildung 78 zeigt dies anhand des Domänenobjekts Auftrag, das in einem Interaktionsschema zur Suche selektiert wird. Als Ergebnis werden vier relationale Informationsobjekte aus zwei Datenquellen gefunden, die jeweils paarweise gemeinsam den Auftrag repräsentieren.

Auswahl von Datenquellen

Stehen mehrere Datenquellen zur Verfügung, kann die Auswahl anhand der Kriterien Perspektive, Inhalt und Datenquelleneigenschaften erfolgen. Die *Perspektive* beschreibt, aus welchem Blickwinkel eine Datenquelle ein Domänenobjekt beschreibt und ist durch das Merkmal `Erhebungsobjekt` der Datenquelle gekennzeichnet. Sie bestimmt maßgeblich den Aussagegehalt der Daten. Die Reflexion über die Perspektive kann dazu anregen, weitere Daten aus einem bisher nicht repräsentierten Blickwinkel (z.B. externe Daten) zu beschaffen.

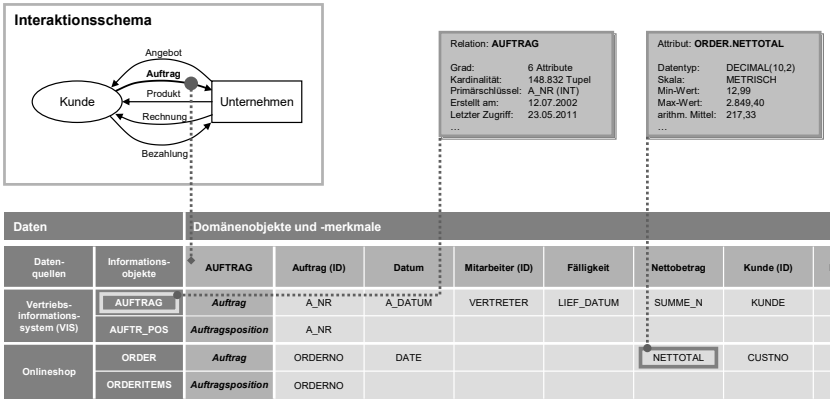


Abbildung 78: Semantische Beschreibung von Datenquellen mithilfe einer Begriffsmatrix (eigene Darstellung)

Nehmen wie im gezeigten Beispiel beide Datenquellen dieselbe Perspektive ein (hier: Verkauf), oder ist eine Entscheidung allein auf dieser Basis nicht möglich, sollte der *Inhalt* der Datenquellen weiter

betrachtet werden. Hierzu können die Metadaten der Informationsobjekte in einer Matrixdarstellung angeordnet werden, um Aggregationsbeziehungen übersichtlich darzustellen. In Abbildung 78 sind Relationen auf der vertikalen und die zugehörigen Attribute auf der horizontalen Achse abgetragen.[170] Um die semantische Äquivalenz von Attributen mit abweichender Bezeichnung in verschiedenen Informationsobjekten abzubilden, sind in der ersten Zeile die konzeptuellen Merkmale des Domänenobjekts aufgeführt, denen die Attribute ontologisch zugeordnet sind. Aus den Kreuzungszellen ist direkt ersichtlich, welche Informationsobjekte welche konzeptuellen Merkmale anhand welcher empirischen Attribute beschreiben.[171] Eine solche Begriffsmatrix ermöglicht einen sofortigen Überblick über den konzeptuellen Inhalt der Datenquellen und unterstützt wirksam deren Auswahl. Weitere Unterstützung ist mithilfe von Datencharakteristika (`Wert`-Attribut des Informationsobjekts) möglich, die z.B. bei Auswahl von Zeilen oder Zellen der Matrix Auskunft über die Anzahl der Zeilen einer Relation oder über statistische Lage- und Streuungsparameter von Attributen geben (in Abbildung 78 oben gezeigt).

Das Datenangebot einer konkreten Quelle wird neben inhaltlichen auch durch formale *Datenquelleneigenschaften* charakterisiert, an denen sich die Auswahl orientieren kann. Dies sind zunächst die Deskriptoren des `Datenquellenprofils`, die z.B. Angaben über Art (z.B. Aussageform, Struktur und Medientyp), Qualität (z.B. Zuverlässigkeit, Detailliertheit, Personenbezogenheit), Verfügbarkeit und Kosten der Daten einer Quelle machen. Einige Beispiele sind in Abschnitt 4.6.3.2 genannt, ein Katalog möglicher Deskriptoren mit Erläuterungen findet sich in Anhang A5.3. Die Merkmale `Ressourcentyp` und `Datenquellentyp`

[170] Das Werkzeug RECON verwendet eine solche Darstellung für Datenbankschemata und ermöglicht die Spezifikation von Datenmodifikationen und Anfragen auf Basis dieser „Query Matrix" [SiLK94, 40].

[171] Als theoretische Grundlage dient ein *formaler Kontext* (G, M, I), der durch eine Menge von Begriffen G, eine Menge von Attributen M und eine Zuordnungsrelation $I \subseteq G \times M$ definiert ist. Dessen Begriffe (A, B) stehen bezüglich ihrer Extension $A \subseteq G$ und Intension $B \subseteq M$ in einer Ordnungsrelation und bilden ein Begriffsgitter, das sich als Diagramm darstellen lässt [StWW98, 452].

geben Auskunft über die Herkunft der Daten (z.B. Datenbank, persönliche Auskunft; Primär-/Sekundärdaten).

Die gewählten Quellen, die zugehörigen Erhebungsobjekte sowie gegebenenfalls repräsentierte Geschäftsprozesse werden im Attribut `Perspektive` des Analyseproblems dokumentiert.

Bestimmung des Analyseobjekts

Nach Festlegung der Datenquellen ist eine Menge von Informationsobjekten zu bestimmen, auf deren Grundlage die Datenanalyse erfolgen soll. Im Falle von Sekundärdaten werden geeignete Informationsobjekte aus den Quellen selektiert. Im Falle von noch zu erhebenden Primärdaten werden neue Informationsobjekttypen deklariert, die den benötigten Aussagegehalt liefern. Die Deklaration erfolgt typischerweise in Form von Tabellen, welche die zu erhebenden Merkmale sowie die zugehörigen Skalen (Wertebereiche, Restriktionen) enthalten und z.B. die Grundlage zur Entwicklung von Fragebögen oder anderen Messinstrumenten bilden (vgl. [Drei94, 81]).

Grundsätzlich sind die Informationsobjekte so zu wählen, dass sämtliche in den Beschreibungselementen der `Analysefrage` enthaltenen Merkmale abgedeckt sind, d.h., dass alle Aussageargumente, Beschreibungsdimensionen und Selektionsdimensionen eine Entsprechung im Analyseobjekt besitzen.[172] Tatsächlich müssen aber nicht alle Elemente direkt im Analyseobjekt repräsentiert sein, da manche Merkmale nicht als Daten vorliegen, aber z.B. im Zuge der Datenanalyse aus anderen Kennzahlen berechnet oder abgeleitet werden können. Die Elemente der Analysefrage geben demnach den Aussagegehalt der Informationsobjekte vor, erfordern aber nicht zwingend deren vollständige Abdeckung. Vielmehr soll der Inhalt des Analyseobjekts insgesamt zur Befriedigung des im Analyseziel formulierten Informationsbedarfs beitragen.

---

[172] Das Untersuchungsobjekt (`Analyseziel.Domänenobjekt`) bestimmt prinzipiell die Objektklasse der Selektionsdimensionen. Dieser Assoziation trägt bereits die beschriebene Vorgehensweise zur Identifikation von Datenquellen nach Maßgabe des Untersuchungsobjekts Rechnung.

Die Bestimmung des Analyseobjekts ist durch drei Eigenschaften gekennzeichnet. Erstens sind in der Regel mehrere Informationsobjekte nötig, um den Datenbedarf zu befriedigen (z.B. wird das Untersuchungsobjekt „Auftrag“ in Abbildung 78 durch jeweils zwei Relationen repräsentiert). Zweitens sind neben dem Untersuchungsobjekt häufig weitere Domänenobjekte zu betrachten. So verweisen Beschreibungsdimensionen oft auf Merkmale anderer Domänenobjekte (z.B. Eigenschaften der im Auftrag bestellten Artikel). Drittens ist häufig eine spätere Anpassung der Spezifikation des Analyseobjekts notwendig, wenn in diesem Schritt nicht alle relevanten Aussagen vollständig bedacht werden können.

5.4.5.3 Konkretisierung und Strukturierung von Analysezielen (Z3.3)

Die konzipierten Analyseziele und -probleme können häufig nutzbringend konkretisiert werden, um die Untersuchung besser auf relevante Aspekte zu fokussieren oder durch Aufnahme weiterer Aspekte zu verbreitern bzw. vertiefen. Als Differenzierungskriterien kommen die zur Beschreibung von Analyseproblemen genutzten Elemente *Merkmal*, *Domänenobjekt* und *Perspektive* in Frage, die verfeinert bzw. (im Falle von Perspektiven) neu definiert werden:

- Zieldifferenzierung (Z): Änderung von `Analyseziel.Merkmal`
- Objektdifferenzierung (O): Änderung von `Analyseziel.Domänenobjekt`
- Perspektivendifferenzierung (P): Änderung von `Analyseproblem.Perspektive`

Es ist zulässig, mehrere Elemente in einem Schritt zu modifizieren. Durch mehrstufige Konkretisierung entstehen Analysezielhierarchien.[173] Abbildung 79 zeigt einige Konkretisierungsoptionen anhand eines Beispiels. Das Analyseziel „Kundenzufriedenheit“ wird zunächst durch Zieldifferenzierung (Z) unterschieden in „Produktzufriedenheit“

[173] Analyseprobleme sind stets auch Analyseziele (vgl. Abschnitt 4.4.2.3), weshalb hier der allgemeinere Begriff verwendet wird.

und „Servicezufriedenheit“, indem das Merkmal (Untersuchungsziel) Zufriedenheit anhand der Bezugsgrößen Produkt und Service konkretisiert wird. Das Ziel „Produktzufriedenheit“ wird sodann mithilfe der Indikatoren Wiederkaufrate und Weiterempfehlungsrate operationalisiert (Zieldifferenzierung Z). Der zweite Indikator wird analog auf das Ziel „Servicezufriedenheit“ angewandt.

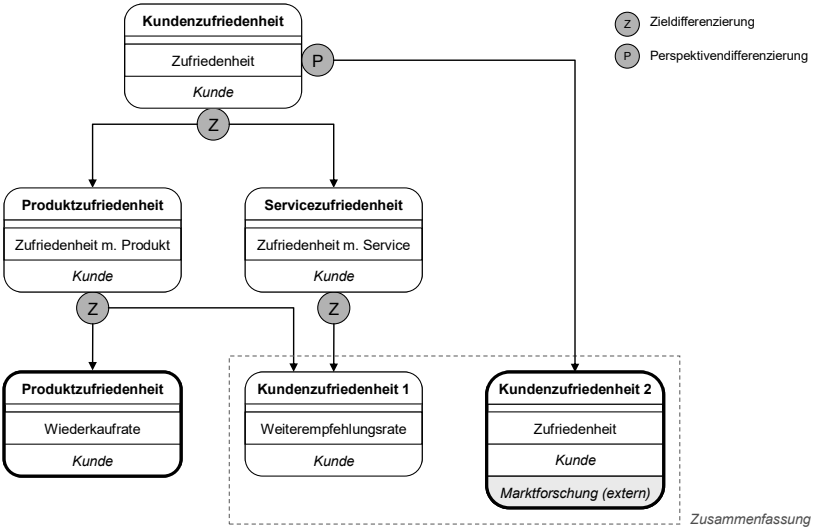


Abbildung 79: Konkretisierung von Analyseproblemen (eigene Darstellung)

Das Unterziel „Kundenzufriedenheit 1“ unterstützt demnach zwei Oberziele. Konvergierende Konkretisierungen auf alternativen Wegen zeigen, dass die Wahl bzw. Reihenfolge der Kriterien nicht entscheidend ist. Diese Tatsache fördert die Akzeptanz des Verfahrens und kann zur besseren inhaltlichen Abdeckung beitragen. So führt eine alternative Verfeinerung der „Kundenzufriedenheit“ anhand der Perspektive (P) zur Zufriedenheitsmessung aus Sicht der Marktforschung. Das entstehende Analyseproblem „Kundenzufriedenheit 2“ subsumiert das Ziel „Kundenzufriedenheit 1“ und kann mit diesem verschmolzen werden. Die nicht fett umrandeten Ziele können verworfen werden.

Die Konkretisierung soll zum Nachdenken über die bereits formulierten Analyseziele anregen und dient dem Zweck, präzise fokussierte Untersuchungen zu konzipieren. Eine vollständige Erfassung aller relevanten Aspekte kann nicht gewährleistet werden, wird durch die Betrachtung von Merkmalen, Domänenobjekten und Perspektiven unter bewusster Inkaufnahme redundanter Überlegungen aber gefördert.

Zu verfolgende Analyseziele sind in einer *Analysekette* zu strukturieren. Analyseketten verknüpfen Analyseziele in sachlicher bzw. zeitlicher Hinsicht und können auch als Netz gestaltet sein, wenn einzelne Untersuchungen parallel erfolgen. Werden Analyseziele, die dasselbe Domänenobjekt untersuchen, vertikal auf einer Ebene angeordnet (vgl. Abschnitt 4.4.3), lässt sich leichter beurteilen, ob die Betrachtung des Untersuchungsobjekts im aktuellen Anwendungskontext hinreichend ist, oder ob weitere Analysen zu ergänzen sind. Im Verlauf des Projekts entstehende neue Analyseziele werden in die Kette eingefügt, sofern sie einem Problemaspekt in der Problemkarte zugeordnet werden können oder einen neuen Aspekt aufdecken, der seinerseits einen übergeordneten Problemaspekt unterstützt. Ist eine solche Zuordnung nicht möglich, ist das Analyseziel für das vorliegende Sachproblem als nicht relevant zu erachten (Zielkontrolle).

5.4.5.4 Zusammenfassung: Spezifikation des Analyseproblems

Analyseprobleme spezifizieren die auszuführenden Datenanalysen und sind demnach auf der Analyseebene angesiedelt. Sie nehmen jeweils auf einen der Problemaspekte Bezug, die aus der Domänenanalyse hervorgehen, und legen die analytisch zu beantwortenden Fragen und geeignete Datenquellen fest. Mehrere Analyseprobleme werden in Form einer Analysekette strukturiert, die den geplanten Ablauf einer Untersuchung vorgibt. Während der Konkretisierung und Strukturierung (Z3.3) können neue Analyseziele und -probleme identifiziert werden, wodurch Iterationen der Schritte Z3.1 bzw. Z3.2 nötig werden. Im Zuge der Konzipierung oder Abarbeitung von Analyseketten sind Wechselwirkungen mit der Domänenanalyse zu erwarten.

5.4.6 Untersuchungsdesign (Z4)

Die bisherigen Überlegungen orientieren sich am zu lösenden Sachproblem. Das Untersuchungsdesign hat die Aufgabe, die sachinhaltlich konzipierten Analyseprobleme unter Einbeziehung statistischer Erwägungen in methodisch fundierte Untersuchungspläne zu übersetzen und bildet zugleich die Brücke zur Prozessplanung. Ergebnis dieses Schrittes ist eine nach methodischen Überlegungen überarbeitete Analysekette sowie konkrete Pläne für jede Einzeluntersuchung.

Das Untersuchungsdesign bestimmt eine Strategie zur Erhebung und Analyse von Daten, um daraus gültige Aussagen über reale Sachverhalte abzuleiten (vgl. [Tuft74, 3], [Hand94, 318]). Nicht fachgerecht konzipierte Analysen können den Erfolg des gesamten Projekts gefährden, weshalb diesem Schritt entsprechende Aufmerksamkeit zu widmen ist [Cohe96, 12]. Ein geeignetes Untersuchungsdesign ist stets im Zusammenhang von Analyseziel, Analysedaten und Analyseverfahren mit Bezug zum jeweiligen Einzelfall zu bestimmen. Es ist weder Ziel, noch liegt es im Rahmen der Möglichkeiten dieser Arbeit, auf diese Aspekte einzugehen. Hierfür ist in großem Umfang einschlägige Literatur vorhanden. Zur Berücksichtigung dieser Belange wird die Teilaufgabe *Methodische Überlegungen zum Untersuchungsgang (Z4.1)* in das Handlungsschema aufgenommen. Auf deren Grundlage können die Aufgaben *Konzipierung des Untersuchungsgangs (Z4.2)* und *Konzipierung von Einzelanalysen (Z4.3)* fundiert ausgeführt werden.

5.4.6.1 Methodische Überlegungen zum Untersuchungsgang (Z4.1)

Die Konzipierung einer problemgerechten Datenanalyse erfordert umfangreiche Kenntnisse und Erfahrungen mit den im jeweiligen Kontext anwendbaren Methoden [Tuke62, 9]. Als prägnante Richtlinie mag PYLES Analogie zum Umgang mit technischen Geräten dienen: „Before assembly, please read all the instructions!“ [Pyle03, 89] Die Forderung, vor Anwendung eines Analyseverfahrens dessen Einsatzvoraussetzungen und inhaltliche Implikationen zu eruieren und die geplante Untersuchung kritisch zu reflektieren erscheint zwar trivial, wird in der Analysepraxis aber aus verschiedenen Gründen allzu oft

ignoriert. Die hierbei anzustellenden Überlegungen dienen letztlich der Vermeidung zweier Facetten des Misserfolgs: einerseits dem Übersehen valider, in den Daten enthaltener Aussagen, andererseits der Ableitung nicht gültiger (irreführender, nicht repräsentativer oder rein zufälliger) Aussagen (vgl. [DeHa01, 229]).

Zwei wichtige Heuristiken, die zur Erreichung dieser Ziele beitragen, sind zum einen die als *Triangulation* bezeichnete Verwendung verschiedener Datenquellen, Analysefragen und Analyseverfahren zur Untersuchung eines Sachverhalts [Pyle03, 267], zum anderen die regelmäßige Überprüfung aller explorativ ermittelten Ergebnisse mithilfe einer *konfirmatorischen Analyse*. Die Triangulation gilt als bewährte Strategie in der Datenanalyse, um Unzulänglichkeiten einzelner Verfahren auszugleichen oder Fehlinterpretationen zu vermeiden (vgl. [GMPS97, 19]). Sie wird vom Konzept des Analyseproblems sowie von der Konkretisierung von Analysezielen (Z3.3) unterstützt. Die Hypothesenprüfung wird u.a. vom idealtypischen Ablauf der Datenanalyse und vom empirischen Zyklus gefordert (vgl. Abschnitte 2.2.1.3 und 2.3.2.3).

5.4.6.2 Konzipierung des Untersuchungsgangs (Z4.2)

Das Untersuchungsdesign umfasst alle Analyseprobleme, die zur Lösung eines Problemaspekts beitragen. In einem ersten Schritt werden daher die erstellten Analyseketten betrachtet und überprüft, ob sie vor dem Hintergrund der methodischen Überlegungen (Z4.1) und der analysestrategischen Gesamtschau zulässig, vollständig und zielführend sind. Die Analyseketten sind in der Regel lediglich punktuell zu modifizieren, indem einzelne Analyseprobleme hinzugefügt, entfernt, inhaltlich variiert oder bezüglich der Verknüpfung angepasst werden. Die getrennte Behandlung der analysemethodischen Aspekte erlaubt die Fokussierung der vorgelagerten Planungsschritte auf rein sachinhaltliche Belange, die durchgängig im Dialog mit dem Auftraggeber erörtert werden können. Darauf aufbauend kann der Analytiker den Untersuchungsgang methodisch fundiert konzipieren.

5.4.6.3 Konzipierung von Einzelanalysen (Z4.3)

Im zweiten Schritt ist für jedes Analyseproblem der Analysekette eine Einzelanalyse zu konzipieren. Im Falle von Sekundärdatenanalysen beschreibt das Untersuchungsdesign, wie die Daten auszuwerten sind. Im Falle von Primärdatenanalyse ist zusätzlich zu bestimmen, wie die Daten zu erheben sind.

Erhebungsdesign (nur Primärdatenanalyse)

Der Datenerhebungsplan legt fest, welche Daten wo (Erhebungsobjekt), wie, wann (wie oft) und in welchem Umfang zu erfassen sind [Drei94, 29f.]. Inhalt und Perspektive sind bereits mit dem Analyseproblem bestimmt, so dass Art, Zeit und Umfang der Erhebung zu definieren bleiben. Grundlegende *Datenerhebungsmethoden* sind in Abbildung 80 dargestellt und nach Charakteristika des untersuchten Verhaltens (Untersuchungsziel) gegliedert.

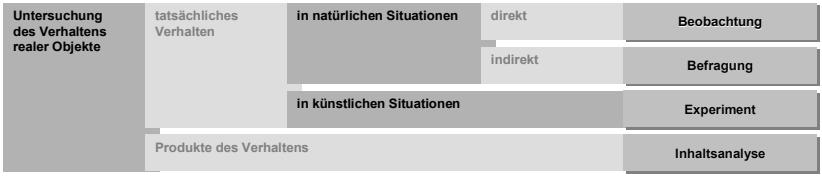


Abbildung 80: Methoden der Datenerhebung (in Anlehnung an [Drei94, 11f.])

Die Methoden zur Erfassung des tatsächlichen Verhaltens sind in Abschnitt 2.2.2.1 skizziert. Nicht-experimentelle Untersuchungen werden auch als Ex-post-facto-Designs bezeichnet („nach den Fakten") [Drei94, 28], die nur eingetretene Ereignisse betrachten, ohne Aussagen über deren Ursachen zu ermöglichen. Letztere Aussagen sind im strengen Sinne nur bei randomisierten Experimenten uneingeschränkt möglich. Für die Durchführung von Experimenten ist mit der statistischen Versuchsplanung eine umfassende Theorie vorhanden [HeMi94, 27]. Die Inhaltsanalyse betrachtet Produkte bestimmten menschlichen Verhaltens, nämlich der expliziten und bewussten Auf-

zeichnung von Wissen i.w.S.[174] Sie zielt somit auf die systematische Auswertung von Dokumenten oder Nachrichten [Diek07, 576, 578] und umfasst u.a. Literaturrecherche, Information und Web Retrieval sowie die Analyse der Kommunikation in Social Media.

Die *zeitliche Dimension* unterscheidet zwischen Querschnitt- bzw. Längsschnittstudien. Querschnittstudien führen eine einmalige Datenerfassung zu einem bestimmten Zeitpunkt durch (Momentaufnahme). Längsschnittstudien sind durch mehrmalige Datenerfassung zu mehreren Zeitpunkten gekennzeichnet und dienen der Untersuchung von Entwicklungen oder Zeitreihen. Nach den untersuchten Einheiten werden zwei Formen unterschieden: Trends erfassen verschiedene Untersuchungsobjekte und ermöglichen somit Zeitreihenvergleiche zwischen unterschiedlichen Stichproben. Panels operieren jeweils auf derselben Stichprobe und erfassen stets die gleichen Merkmale, womit sich Entwicklungsprozesse (individuelle Veränderungen) untersuchen lassen [Drei94, 28f.].

Der *Umfang* der Datenerfassung betrifft die Entscheidung zwischen Vollerhebung (Gesamtpopulation) oder Teilerhebung (Stichprobe) der Untersuchungsobjekte. Im zweiten Fall stellt sich die Frage, nach welcher Methode und in welchem Umfang die Stichprobe gezogen werden soll. Zu näheren Details sei auf die einschlägige Literatur verwiesen (z.B. [Diek07, 373ff.]).

Auswertungsdesign

Im Weiteren wird davon ausgegangen, dass Daten gemäß der Spezifikation des Analyseobjekts erhoben sind, in digitaler Form vorliegen und der Bearbeitung zugänglich sind. Damit sind die Voraussetzungen für die Analyseplanung nach Maßgabe des Analyseproblems erfüllt. Sofern eine Analyse i.e.S. angestrebt wird, entspricht das Auswertungsdesign der *Planung des Analyseprozesses* (Prozessspezi-

[174] Die abweichend vom Verständnis der empirischen Sozialforschung vorgenommene Einschränkung auf die bewusste Aufzeichnung von Wissen ist erforderlich, um die Inhaltsanalyse deutlich von der Sekundärdatenanalyse und von der Auswertung automatisiert durchgeführter Beobachtungen abzugrenzen.

fikation, Abschnitt 5.5). Für einfache Analysen, wie z.B. die manuelle Literaturrecherche oder die Sichtung von Videoaufzeichnungen, kann auf die Prozessplanung verzichtet werden. Auch im Bereich Data Science wird eine Planung häufig bewusst unterlassen, um die experimentelle Natur der Untersuchungen zu unterstreichen.

5.4.6.4 Zusammenfassung: Untersuchungsdesign

Das Untersuchungsdesign überführt die aus Sicht des Sachproblems definierten Analyseprobleme in methodisch fundierte Untersuchungspläne und bildet das Bindeglied zwischen Ziel- und Prozessebene der Datenanalysearchitektur. Hierzu sind grundlegende Erwägungen zur statistischen Strategie anzustellen und die vorgegebenen Analyseketten bei Bedarf angemessen zu modifizieren. Bei Primärdatenanalysen ist ein Plan zur Datenerhebung zu konzipieren.

Angesichts der evolutionären Natur der Datenanalyse liefert auch das Untersuchungsdesign keinen endgültigen Untersuchungsplan. Im Laufe des Untersuchungsgangs auftretende neue Analyseziele erweitern den Plan und erfordern jeweils erneut methodische Überlegungen. Das Untersuchungsdesign sollte stets detailliert dokumentiert werden, um seine Nachvollziehbarkeit, seine Reproduzierbarkeit durch andere Analytiker sowie die Überprüfbarkeit der gewonnenen Erkenntnisse zu ermöglichen [Drei94, 9f.]. Dies ist mithilfe der Analyseketten gewährleistet.

5.4.7 Projektplanung (Z5)

Ziel der Projektplanung ist die Erstellung einer Informationsgrundlage für das Management des gesamten analytisch gestützten Projekts. Sie ist demnach der Anwendungsebene zuzuordnen und kann aus dieser Warte alle relevanten Belange der untergeordneten Ebenen berücksichtigen. Ihr Ergebnis sind Ressourcen-, Zeit-, Budget- und Organisationspläne.

Das analytisch gestützte Projekt dient der Lösung des Sachproblems. Nach diesem Verständnis sind neben der Planung und Durchführung von Datenanalysen auch die Anwendung des Wissens und die

Evaluierung des Projekts Gegenstand der Projektplanung (vgl. Abschnitt 3.3.2). Im weiteren Sinne zählen auch die Schritte Z1-Z4 zur Projektplanung, die das Vorhaben aus leistungsorientierter Sicht ausarbeiten. In diesem Abschnitt wird die Projektplanung im engeren Sinne verstanden und aus administrativer Sicht betrachtet. Ausgangspunkt bilden die Rahmenbedingungen, die dem Projekt etwa in Form von Zeit- und Budgetrestriktionen auferlegt und in den Maßnahmen der Problemkarte dokumentiert sind, sowie die in den Schritten Z1-Z4 getroffenen Gestaltungsentscheidungen. Gleichzeitig begrenzen die Projektrestriktionen die dortigen Gestaltungsspielräume. Die administrative Projektplanung flankiert demnach die anderen Schritte der Problemspezifikation und steht mit ihnen in ständiger Wechselwirkung.

Die inhaltlichen Planungsschritte liefern *Leistungspakete und Meilensteine*[175] als Bezugsobjekte des Projektmanagements. Die Leistungspakete resultieren aus den Informations- und Handlungsmaßnahmen der Problemkarte. Insbesondere definiert jedes Analyseziel der Analysekette ein Leistungspaket (vgl. Abschnitt 3.3.1.4). Als Meilensteine können z.B. die Phasen des Vorgehensmodells auf Projekt- und Prozessebene definiert werden (vgl. Abschnitt 3.3.2).

Die administrative Projektplanung umfasst die Teilaufgaben *Ressourcenplanung (Z5.1)*, *Zeitplanung (Z5.2)*, *Budgetplanung (Z5.3)* und *Organisationsgestaltung (Z5.4)*. Sie werden nur insoweit behandelt, wie es analysespezifische oder methodische Besonderheiten erfordern. Weitere Details des Projektmanagements werden nicht näher betrachtet.

5.4.7.1 Ressourcenplanung (Z5.1)

Auf Basis der Problemkarte und Analyseketten ist der Bedarf des Projekts an personellen und maschinellen Aufgabenträgern zu bestimmen (vgl. [Daen88, 125]). Zur Unterstützung der Personal-

[175] Leistungspakete sind Projektresultate, die an den Auftraggeber übergeben werden (Lieferschritte). Meilensteine markieren den definierten Abschluss von Projektaufgaben und werden bei ihrer Erreichung dokumentiert. Neben Leistungspaketen können auch andere Ereignisse, die der Überprüfung des Projektfortschritts dienen, als Meilensteine definiert werden [Somm01, 89].

planung können die auf Ressourcenebene hinterlegten Rollen, Qualifikationsprofile und Personenzuordnungen beitragen. Darüber hinaus sind verschiedene Rollenmodelle für Datenanalyseprojekte publiziert, die z.B. Projektleiter, Analytiker, Techniker und Domänenexperten fordern (vgl. z.B. [DeHa01, 261]). Deren praktische Relevanz ist jedoch begrenzt. Der Bedarf an maschinellen Aufgabenträgern ergibt sich aus den im Analyseprozess benötigten Verfahren und zu verarbeitenden Daten. Letztere werden in Schritt Z3.2 bestimmt. Für alle nicht datenanalytischen Aufgaben (z.B. Systementwicklung und -wartung, Realisierung der Handlungsmaßnahmen) sind weitere Rollen mit jeweils spezifischem Qualifikationsprofil zu besetzen bzw. erforderliche technische Ressourcen bereitzustellen.

5.4.7.2 Zeitplanung (Z5.2)

Der Zeitbedarf für ein Datenanalyseprojekt kann fallspezifisch stark variieren [DeHa01, 263] und ist anhand allgemeiner Erfahrungen aus vergangenen Projekten nicht zuverlässig abzuschätzen (vgl. [Somm01, 89]). Präzisere Zeitschätzungen liefern die im Rahmen der Revision (Abschnitt 7.4.2.1) auf Basis von Prozessprotokollen ermittelten Durchschnitts-, Mindest- und Höchstdauern einzelner Prozessaktivitäten, die mithilfe von Kontextfaktoren z.B. die Berücksichtigung der eingesetzten Analyseverfahren oder ausgewerteten Datenquellen ermöglichen. Eine sichere Grundlage für die Planung der Gesamtdauer können aber auch sie nicht liefern [Somm01, 90, 95]. Bei der Datenanalyse resultieren Projektverzögerungen insbesondere aus der iterativ-inkrementellen Vorgehensweise sowie aus situativ aufgedeckten Analysezielen, die bei der Zeitplanung zu beachten sind. Zuweilen lässt sich aus Umfang bzw. Detailliertheit von Problemkarte bzw. Analysekette ermessen, inwieweit mit weiterem Zeitbedarf zu rechnen ist.

5.4.7.3 Budgetplanung (Z5.3)

Der Finanzbedarf des Projekts ist auf Basis der Ressourcen- und Zeitpläne zu schätzen [Daen88, 125]. Sie ist mit denselben Unsicherheiten behaftet wie die Zeitplanung und sollte mit ausreichenden

Reserven kalkuliert werden. Bei der Datenanalyse entfällt der größte Anteil in der Regel auf Personalkosten [DeHa01, 264f.]. Auch die Kostenschätzung profitiert von detaillierten Erfahrungswerten aus der Auswertung von Prozessprotokollen.

5.4.7.4 Organisationsgestaltung (Z5.4)

Gegenstand der Organisationsgestaltung sind u.a. die Bildung von Arbeitsgruppen, die Einrichtung von Lenkungsgremien und die Informationsversorgung aller Beteiligten [Daen88, 125-127]. Es empfiehlt sich, das Projekt anhand von Leistungspaketen in überschaubare Abschnitte zu gliedern. Die Differenzierung in situations- und lösungsbezogene Problemaspekte erlaubt eine Gliederung in Teilprojekte, die sich in den Qualifikationsanforderungen des Personals stark unterscheiden können. Eine weitere Gliederung nach Maßnahmen ist oft sinnvoll. Bei regelmäßig auszuführenden Analyseprozessen sind die Wiederholungsrate und die organisatorische Verantwortlichkeit festzulegen. Zur Projektüberwachung sind informelle Gespräche und Sitzungen mit dem Auftraggeber häufig geeigneter als formelle Instrumente, um terminliche oder inhaltliche Probleme frühzeitig zu erkennen (vgl. [Somm01, 85]).

5.4.7.5 Zusammenfassung: Projektplanung

Die Projektplanung begleitet die Planung und Durchführung des Analyseprojekts und dient der Bereitstellung bzw. ständigen Aktualisierung einer Informationsgrundlage für das Projektmanagement. Wegen der Reichweite dieser Aufgabe über alle Projektphasen sind Rückkopplungen zu allen vorgelagerten Aufgaben der Problemspezifikation sowie mit der Prozessplanung erforderlich. Insbesondere sind die Projektpläne bei Aufnahme neuer Elemente in die Problemkarte und in die Analysekette zu aktualisieren.

5.4.8 Zusammenfassung: Problemspezifikation

Die Problemspezifikation erarbeitet Ziele und Rahmenbedingungen für datenanalytisch gestützte Projekte und reicht von der Identifikation und

Strukturierung von Sachproblemen bis zur Bestimmung und Verkettung von Analyseproblemen. In Situationen, in denen die Behandlung der Anwendungsebene (Sachprobleme) verzichtbar erscheint, kann die Problemspezifikation direkt auf der Analyseebene mit Schritt Z3 (Spezifikation von Analyseproblemen) starten.

5.5 Prozessspezifikation

Dieser Abschnitt beschreibt ein Handlungsschema zur Planung von Datenanalysen auf Prozessebene. Die Prozessspezifikation ist für jedes Analyseproblem durchzuführen, für das Schritt Z4.3 im Rahmen des Untersuchungsdesign einen entsprechenden Planungsbedarf identifiziert. Wird auf die Problemspezifikation verzichtet, startet die Planung einer Datenanalyse hier.

5.5.1 Aufgaben und Vorgehen bei der Prozessspezifikation

Die Analyseplanung auf Prozessebene dient der Konstruktion eines Lösungsverfahrens für ein definiertes Analyseproblem in Form eines Analyse-Workflows. Sie wird vom Auftragnehmer der Analyseleistung durchgeführt und erfordert typischerweise keine Interaktion mit dem Auftraggeber. Ihre Ergebnisse gehen in die Zeit- und Kostenplanung der Projektplanung Z5 ein.

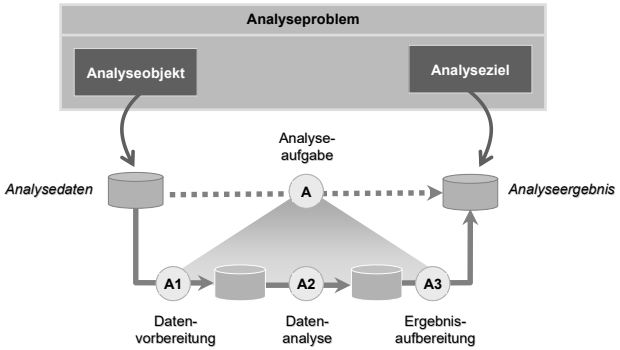


Abbildung 81: Ableitung von Analyseprozessen aus dem Analyseproblem (eigene Darstellung)

Die Prozessspezifikation umfasst die Bestimmung der Prozessaufgaben, die Zuordnung von Operatoren sowie die Bestimmung der Aufgabenreihenfolge und erfolgt strikt zielorientiert. Hierzu wird aus dem Analyseproblem zunächst eine Analyseaufgabe abgeleitet[176] und ein passendes Verfahren zugeordnet. Dieses stellt in aller Regel spezielle Anforderungen an die Datenrepräsentation (Input), woraus eine Datenvorbereitungsaufgabe resultiert, um die gemäß Analyseobjekt bestimmten Daten entsprechend zu transformieren. Entspricht der Output des Verfahrens nicht den im Analyseziel bestimmten Anforderungen an die Analyseergebnisse, ist eine Ergebnisaufbereitungsaufgabe zu ergänzen (vgl. [Knob03a, 346] und Abbildung 81). Die Methodik zur Prozessspezifikation orientiert sich demnach an den drei generischen Phasen der Datenanalyse (vgl. Abschnitt 2.3.2.3), die jeweils eine spezifische Datentransformation leisten.

Die Gliederung der Prozessspezifikation nach Prozessphasen entspricht dem Prinzip der Dekomposition (vgl. Abschnitt 5.2). Es unterstützt die Komplexitätsbewältigung und die Wiederverwendung von Prozessmodulen. Der skizzierte Planungsablauf birgt jedoch eine inhärente Iterationsursache im Hinblick auf die Instanziierung der Operatorparameter. Da die Wahl adäquater Parameterwerte stets auch von den Charakteristika der Eingabedaten abhängt und zu erwarten ist, dass sich diese im Zuge der Datenvorbereitung verändern, ist von der Notwendigkeit einer entsprechenden Parameteranpassung auszugehen. Um diese Ineffizienz zu vermeiden, wird die Instanziierung der Verfahrensparameter als eigener Planungsschritt am Ende des Handlungsschemas eingefügt. Dieses umfasst somit die Aufgaben *Planung der Datenanalysephase (P1)*, *Planung der Datenvorbereitungsphase (P2)*, *Planung der Ergebnisaufbereitungsphase (P3)* sowie *Instanziierung des Analyseverfahrens (P4)* (Abbildung 82).[177]

[176] Im Falle des Verzichts auf die Problemspezifikation ist die Analyseaufgabe eigenständig zu definieren.

[177] Vgl. hierzu auch die Beschreibung des typischen Vorgehens während der Datenanalyse bei [SpNa09], die die Verfahrensauswahl vor der Datenselektion und -vorbereitung, und diese wiederum vor der endgültigen Verfahrensparametrisierung platziert.

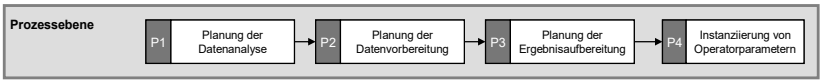


Abbildung 82: Handlungsschema zur Prozessspezifikation (eigene Darstellung)

5.5.2 Theoretische Fundierung

Ausgewählte Forschungsbeiträge zur Prozessplanung sind in Abschnitt 5.3 im Zusammenhang mit den Basisansätzen genannt. Neben den dort vorgestellten Arbeiten im Rahmen der Projekte CITRUS und E-LICO sind für die folgenden Ausführungen insbesondere die Beiträge von HEINRICH ET AL. [Hein+08], [HeKZ11] zum semantischen Prozessmanagement von Interesse, die einen ontologiebasierten Ansatz zur Wahl und Verknüpfung von Prozessaktivitäten sowie zur Aufgabenträgerzuordnung vorstellen. Gute Hinweise für die zielorientierte Auswahl von Verfahren bietet darüber hinaus die Literatur zur Webservice-Komposition (vgl. z.B. [BHMS03], [Ran03], [PrES04], [TGRS04]). Die Arbeiten von LINDNER [LiSt99], [Lind05] ergänzen die Überlegungen von ENGELS um ein fallbasiertes Algorithm Selection Tool (AST), das Datencharakteristika in seine Entscheidungen einbezieht und auf einschlägige frühere Arbeiten aufbaut. Einen ähnlichen Vorschlag zur Modell- und Verfahrensauswahl, der auch die Möglichkeit zur Algorithmusprofilierung sowie Metawissen berücksichtigt, präsentieren HILARIO & KALOUSIS [HiKa01]. Das wissensbasierte System von HOGL [Hogl03] folgt einem mehrstufigen Ansatz zur Wahl konkreter Verfahren, das dem Grundprinzip des Ansatzes von ENGELS [Enge99] gleicht und dem oben skizzierten Vorgehen nahe kommt. Wesentliche Beiträge zur fallbasierten Konzipierung der Datenvorbereitung und der Verfahrensinstanziierung steuert das MININGMART-Projekt [KiVZ00], [MoSE03] bei. Grundlagen zum Planungsablauf liefert die Handlungsplanung der Künstlichen Intelligenz [Hert89, 50], [RuNo03, 388f.].

5.5.3 Planung der Datenanalysephase (P1)

Ziel des ersten Schrittes der Prozessspezifikation ist die Planung der Datenanalysephase, die primär für die Produktion der Analyseergeb-

nisse verantwortlich ist (vgl. [BrAn96, 43]). Sein Ergebnis ist ein geeigneter, aus Aktivitätssicht spezifizierter Prozessausschnitt.

Hierzu erfolgt zunächst eine auf das vorliegende Analyseproblem abgestimmte *Spezifikation der Analyseaufgabe (P1.1)* aus Außensicht. Hierzu zählt auch die *Charakterisierung der Analysedaten (P1.2)*, die aufgrund ihrer Bedeutung als eigene Teilaufgabe behandelt wird. Die Auswahl eines Analyseverfahrens zur Bestimmung der Aufgabeninnensicht wird im Allgemeinen als nicht-triviales Entscheidungsproblem erachtet (vgl. z.B. [Liu99, 357f.], [Lind05, 119]) und erfolgt daher zweistufig: Ausgehend von der Analyseaufgabe wird zuerst die *Bestimmung einer Verfahrensklasse (P1.3)* behandelt, aus der anschließend die *Auswahl eines Analyseverfahrens (P1.4)* erfolgt. Abschließend sind *kontextabhängige Entwurfsentscheidungen (P1.5)* zu treffen.

5.5.3.1 Spezifikation der Analyseaufgabe (P1.1)

Die Beschreibung der **`Analyseaufgabe`** besteht im Wesentlichen aus `Funktion` (Sachziel), `Anforderungen` (Formalziele) sowie `Eingabe-` und `Ausgabeflüssen` (vgl. Abschnitt 4.5.1). Diese Beschreibungselemente können zu großen Teilen aus dem `Analyseproblem` abgeleitet werden. Ist kein Analyseproblem definiert, ist die Beschreibung an dieser Stelle zu entwickeln.

Bestimmung von Funktion und Anforderungen

Die Identifikation der Funktion der Analyseaufgabe erfolgt ergebnisgetrieben, ausgehend von dem in der `Analysefrage` benannten `Aussagetyp`. Beispielsweise zielt die Analyse der Entwicklung des Bonbetrags (vgl. Abschnitt 4.4.1.1) auf eine Veränderung, während ein Analyseproblem zur Zielgruppenbestimmung im Direktmarketing etwa auf einen Einzelwert abzielt (Zielvariable: Antwortwahrscheinlichkeit). Der Aussagetyp determiniert den Typ des als Ergebnis der Analyseaufgabe erwarteten Informationsobjekts und impliziert damit die Transformationsfunktion. Diese wird durch die `Ausrichtung` des Analyseziels näher bestimmt. Abbildung 83 zeigt eine Klassifikation von Analysefunktionen oberster Ebene nach Aussagetyp (Output) und Aus-

richtung.[178] Im ersten Beispielfall ist eine explorative Analyse von Veränderungen (Funktion „Veränderungen & Abweichungen ergründen“, V-E), im zweiten Fall eine schließende Analyse von Einzelwerten (Funktion „Einzelwert deduzieren“, E-D) gefordert.

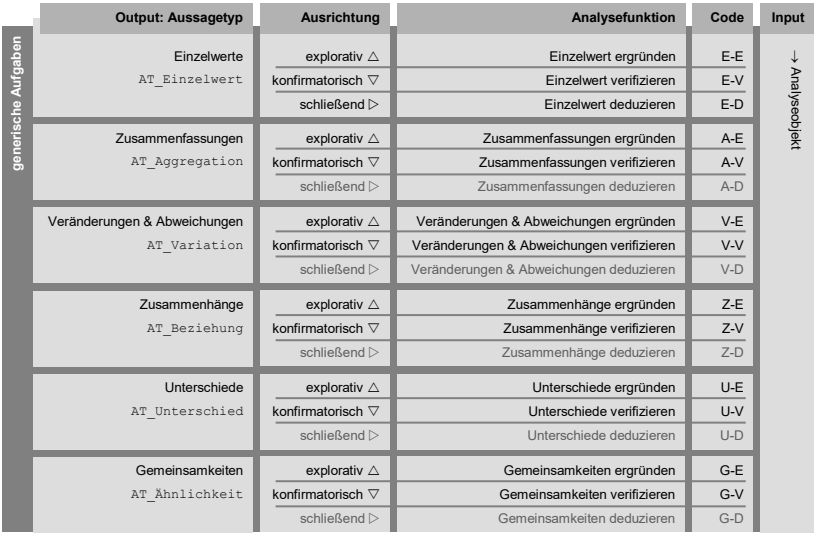


Abbildung 83: Allgemeine Analysefunktionen nach Aussagetyp und Analyseausrichtung (eigene Darstellung)

Die Aufgabenspezifikation wird präzisiert anhand des *Eingabedatentyps* (Input), der sich zunächst aus dem Medientyp der im `Analyseobjekt` benannten Informationsobjekte ergibt. Für das Beispiel wird eine relationale Eingabedatenrepräsentation unterstellt, woraus die Funktion „Einzelwert deduzieren (relational)“ resultiert. Eine weitere Präzisierung ist anhand formaler Kriterien möglich, die sich hauptsächlich aus dem `Informationsbedarfsprofil` des Analyseziels ergeben. Die dort niedergelegten Anforderungen an die Analyseergebnisse können die

[178] Zu den Aussagetypen (Output) sind die korrespondierenden Informationsobjekttypen angegeben. Das Präfix „AT“ steht für Aussagetyp. Schließende Analysen erzeugen im Allgemeinen stets Einzelwerte. Daher sind die verbleibenden schließenden Funktionen ausgegraut, der Vollständigkeit halber aber aufgeführt. Letztlich bleibt es dem Analytiker überlassen, welche taxonomische Ordnung er bevorzugt.

Ausgestaltung der Analyse erheblich beeinflussen. An dieser Stelle im Planungsprozess legt häufig die gewünschte Repräsentationsform der Ergebnisse eine *Konkretisierung des Aussagetyps* nahe. Orientierung bieten Datentyp- oder Funktionstaxonomien, welche die in Abbildung 83 gezeigten Informationsobjekttypen und Funktionen nach verschiedenen Kriterien spezialisieren und auf geeignete, konkretere Analysefunktionen verweisen. Weitere Elemente des Informationsbedarfsprofils können spätere Planungsstufen beeinflussen (z.B. unterstützen Forderungen zur Genauigkeit die Wahl und Instanziierung des Analyseverfahrens, und Forderungen zur Darstellungsform leiten die Gestaltung der Ergebnisaufbereitungsphase). Um zu verhindern, dass wichtige `Anforderungen` übersehen werden, können diese in die Aufgabenbeschreibung einfließen und um weitere Faktoren erweitert werden.

Die Inhalte der Analysefrage sollten genutzt werden, um die Aufgabe durch einen semantisch reichhaltigen `Namen` zu charakterisieren. Es eignet sich das allgemeine Bezeichnungsschema <Ergebnis> <Verrichtung>, das in den oben genannten Beispielen etwa zu den Namen „Veränderung Bonbetrag auswerten“ oder „Antwortverhalten prognostizieren“ führt.

Konkretisierung von Eingabe- und Ausgabeflüssen

Die von der gewählten Funktion vorgegebenen Ein- und Ausgabedatentypen sind für die Aufgabe zu Ein- und Ausgabeflüssen zu konkretisieren. Zunächst können für die Flussbeziehungen inhaltlich sinnvolle Namen gewählt werden, wie z.B. „Kundenprofile“ für den Input, „Kundentabelle“ für den Output in Abbildung 84.[179] Analog können Komponenten der Datenobjekttypen umbenannt und spezialisiert werden. So erhält etwa die Zielvariable des Ausgabeflusses den Namen

[179] Im Beispiel wird unterstellt, dass der zu deduzierende Einzelwert nicht als eigenständiger Output, sondern als angereichtere Spalte an der Input-Tabelle geliefert wird, wie bei gängigen Datenanalysewerkzeugen üblich. Diese Annahme ist realistisch, da Funktionen typischerweise aus Operatoren generalisiert werden. Sie ist inhaltlich sinnvoll, da aus ihrem Zusammenhang gerissene Einzelwerte nicht hilfreich sind.

„Antwortverhalten“ und wird als `_Boolean` mit der Restriktion `notEmpty()` deklariert.[180]

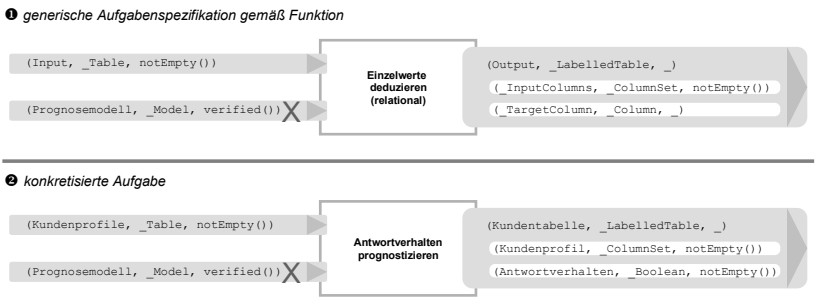


Abbildung 84: Beispiel zur Konkretisierung von Ein- und Ausgabeflüssen der Analyseaufgabe (eigene Darstellung)

Zerlegung von Analyseaufgaben

Zuweilen ist eine Analyse nicht in Gestalt einer Einzelaufgabe ausführbar, sondern bedarf einer Zerlegung in mehrere, über Datenabhängigkeiten gekoppelte Teilaufgaben. Dieser Fall tritt ein, wenn aus der Konkretisierung eine Aufgabenspezifikation resultiert, deren Eingabeflüsse quantitativ oder qualitativ von denen der initialen Analyseaufgabe abweichen. *Quantitative Diskrepanzen* entstehen, wenn die konkrete Aufgabe mehr Eingabeflüsse konsumiert als die ursprüngliche Aufgabe. Dieser Fall liegt im Beispiel aus Abbildung 84 vor, da die gewählte Funktion ein Prognosemodell erfordert, das im Analyseobjekt nicht definiert wurde (X). Daher ist die Analyseaufgabe durch Vorschaltung einer geeigneten Aufgabe „Prognosemodell berechnen“ zu zerlegen, die gegebene Daten von Typ `_LabelledTable` in ein Informationsobjekt des Typs `_Model` transformiert. Das Ergebnis der Zerlegung zeigt Abbildung 85.

[180] Die Grundidee solch generischer Datenelemente, die vor Aufgabendurchführung mit domänenspezifischen Begriffen zu belegen sind, ist aus den Problem Solving Methods der Wissensakquisition bekannt und wird etwa im CommonKADS-Ansatz eingesetzt, um so genannte Wissensrollen auf konkrete Datenobjekte abzubilden [StBF98, 4, 10f.].

Qualitative Diskrepanzen entstehen bei unterschiedlichen Datendeklarationen der Eingabeflüsse. Diese können alternativ im Rahmen der Datenvorbereitung behandelt werden. Zur Entscheidung, innerhalb welcher Phase entsprechende Aufgaben einzuplanen sind, kann die Heuristik dienen, dass strukturelle Abweichungen (z.B. nominale statt kontinuierliche Variable) in der Datenvorbereitungsphase, inhaltliche Abweichungen in der Analysephase zu behandeln sind. Letztere betreffen z.B. die Aufteilung der Daten einer Relation in mehrere Teilmengen, welche dieselbe Datenvorbereitung zu durchlaufen haben. Im Beispiel kann die Aufgabe „Prognosemodell berechnen", dessen Ergebnis mit der Restriktion `verified()` markiert ist, in die drei Teilaufgaben Modellentwicklung, Modellkalibrierung, Modellevaluierung zerlegt werden (vgl. Abschnitt 3.1.2.3), die jeweils unterschiedliche Qualitätsstufen eines Modells (inhaltlich-qualitativ verschiedene Informationsobjekte) produzieren. Aufgabenzerlegungen können demnach mehrstufig erfolgen. Letztlich bleibt die Platzierung von Aufgaben aber dem Analytiker überlassen.[181]

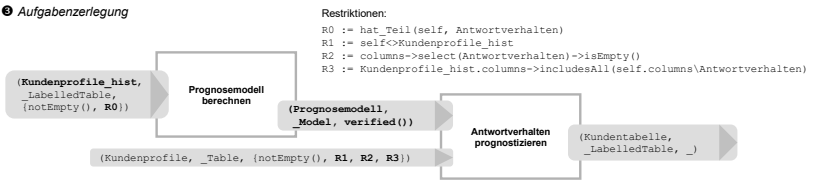


Abbildung 85: Beispiel zur Zerlegung von Analyseaufgaben (eigene Darstellung)

Mit der Zerlegung können sich Einschränkungen in Bezug auf die Ein- und Ausgabeflüsse ergeben, die als Restriktionen innerhalb der Datendeklaration spezifiziert werden können. Beispiele sind in Abbildung 85 angeführt. Für Prognosemodelle muss etwa die Zielvariable in den Trainingsdaten vorhanden sein (Restriktion R0). Weiterhin müssen die beiden Eingaberelationen (Trainings- und Anwendungsdaten) verschieden sein (R1). Zudem kann z.B. festgelegt werden, dass die

[181] So kann z.B. die Extraktion strukturierter Deskriptoren aus Bilddaten sowohl als Bestandteil der Datenanalyse- als auch der Datenvorbereitungsphase angesehen werden.

aktuellen Eingabedaten den Prognosewert nicht enthalten dürfen (R2), sonst aber alle in die Modellberechnung eingeflossenen Attribute aufweisen sollen (R3).

Die beschriebene Vorgehensweise entspricht grundsätzlich der in Abschnitt 5.3.1.2 eingeführten Aufgabendekomposition unter Zuhilfenahme von Funktionstaxonomien.[182] Sie kann durch Wiederverwendung von Aufgabenschablonen unterstützt werden (vgl. Abschnitt 5.3.2.2). Gerade die oben genannten Beispiele zur Zerlegung im Zusammenhang mit der Modellerstellung bieten sich zur Speicherung als Schablonen an. Die erreichte Spezifikation sollte detailliert genug sein, um die anschließende Operatorzuordnung effizient zu bewältigen. Weniger detaillierte Spezifikationen führen dort gegebenenfalls zu einer größeren Menge an Optionen, sind aber ebenso zulässig. Welche Konkretisierungsstufe angebracht ist, hängt nicht zuletzt von der Erfahrung des Analytikers ab. Prinzipiell sind aus einem Analyseproblem mehrere effektive Aufgabenspezifikationen ableitbar.

Laut Metamodell verbindet eine Flussbeziehung jeweils zwei Aufgaben (vgl. Abschnitt 4.5.1.4). Daher sind mit der Definition der Ein- und Ausgabeflüsse an der Analyseaufgabe mindestens zwei Aufgaben in das Prozessschema aufzunehmen, welche diese Flüsse versorgen bzw. konsumieren. Hierzu sind die generischen Aufgaben Datenvorbereitung und Ergebnisaufbereitung geeignet, die zugleich den Rahmen für die weiteren Planungsschritte vorgeben.

5.5.3.2 Charakterisierung der Analysedaten (P1.2)

Die auszuwertenden Daten sind bislang nur bezüglich ihres Medientyps bekannt. Für die Auswahl geeigneter Analyseverfahren und zur Planung der Datenvorbereitung sind weitere Angaben über Verfügbarkeit, Inhalt und Repräsentation der Analysedaten erforderlich. Die Sammlung solcher Datencharakteristika deckt sich mit den Zielen der in Abschnitt 3.1.2.2 erläuterten *Datenexploration*, soweit sie die Prozessplanung

[182] Abweichend von der dortigen Darstellung beginnt die Spezifikation der Analyseaufgabe typischerweise nicht mit den Eingabeflüssen, sondern erfolgt ergebnisgetrieben, d.h., ausgehend von den Ausgabeflüssen.

betreffen, und führt zu einer weiteren Konkretisierung der Aufgabenspezifikation bezüglich der Eingabeflüsse.

Die systematische Nutzung von Datencharakteristika zur Verfahrensauswahl wurde im Rahmen der Entwicklung des Assistenzsystems CONSULTANT für die MACHINE LEARNING TOOLBOX (MLT) [CSG+92] sowie im STATLOG-Projekt [MiST94] vorgeschlagen und von mehreren Autoren aufgegriffen. So wurden etwa im Laufe des METAL-Projekts [Meta01] statistische Maße [KöKe00], Performanzgrößen ausgewählter Algorithmen [PfBG00], [BeGi00] und modellbasierte Charakteristika entwickelt. Im UGM-Projekt lag der Schwerpunkt auf der Unterstützung der Datenvorbereitung, und es kamen informationstheoretische Maße und Charakteristika der Raumkomplexität für Klassifikationsprobleme hinzu [EnTh98], [EnTh98b], [ThLi98]. Im E-LICO-Projekt wurden geometrische Komplexitätsmaße aufgenommen [HND+11]. Auch in der Statistik werden bei der Verfahrens- und Modellauswahl Dateneigenschaften berücksichtigt [Andr+81],[183] [Hand94b]. Beispiele für einfache Datencharakteristika sind Lage-, Streuungs- und Beziehungsmaße der beschreibenden Statistik, wie etwa arithmetisches Mittel, Spannweite, Standardabweichung, Kovarianz, Korrelation oder relative Häufigkeiten [Ehre76, 217ff.], [Drei94, 34], [Beek03, 103ff.]. Darüber hinaus wurden zahlreiche fortgeschrittene Maße vorgeschlagen. Ein Katalog hilfreicher Charakteristika mit Erläuterungen findet sich in Anhang A5.4.

Grundsätzlich sollten Maße, deren Herleitung mit größerem Aufwand verbunden ist, nur bei Bedarf (z.B. abhängig von Skalentyp und Funktion der Analyseaufgabe) berechnet werden. Sie sind nur mithilfe spezifischer Datencharakterisierungstools handhabbar [Lind05, 120]. Solche Einsatzentscheidungen lassen sich z.B. durch Kontextregeln an den Funktionen unterstützen. Einfachere Maße werden häufig von Datenanalysewerkzeugen oder -operatoren standardmäßig berechnet. Ihre explizite Bereitstellung für weitere Planungsschritte (z.B. im `Wert-`

[183] Unter http://www.microsiris.com/Statistical%20Decision%20Tree/ ist eine Implementierung des „Guide for Selecting Statistical Techniques for Analyzing Social Science Data“ [Andr+81] als Entscheidungsbaum online verfügbar (Abruf am 08.04.2017). Er berücksichtigt u.a. Anzahl, Typ und Rolle von Merkmalen.

Attribut von Informationsobjekten, vgl. Abschnitt 4.6.3.1), ist empfehlenswert. Eine Visualisierung der Analysedaten (z.B. durch Streudiagramme) oder berechneter Datencharakteristika wird im Allgemeinen als hilfreich erachtet. Hierzu werden auch eigenständige Datenprofilierungswerkzeuge angeboten [ABEM15, 134, 288].

5.5.3.3 Bestimmung einer Verfahrensklasse (P1.3)

Die Auswahl eines Analyseverfahrens,[184] die für jede der zuvor identifizierten Teilaufgaben durchzuführen ist, erfolgt mehrstufig, indem die Menge anwendbarer Verfahren zunächst anhand zwingend einzuhaltender Restriktionen reduziert und die verbleibenden Optionen anschließend mittels wünschenswerter Präferenzen bewertet werden (vgl. [Schm97, 173]). Am Ende kann das Verfahren mit der höchsten Bewertung übernommen werden. Die wichtigste Restriktion bildet die `Funktion` der Analyseaufgabe. Ihr Abgleich mit verfügbaren *Operatoren* liefert die Klasse der prinzipiell geeigneten Verfahren. Weitere Restriktionen stellen die `Anforderungen` der Aufgabe dar, die entweder bereits in Schritt P1.1 bestimmt oder an dieser Stelle erhoben werden. Beide Gruppen von Kriterien können auch als Präferenzen dienen; die Zuordnung ihrer Rolle obliegt dem Analytiker.

Ein ähnliches Vorgehen wird von ENGELS [Enge99, 124ff.] und HOGL [Hogl03, 95ff.] verfolgt und auch im Kontext der Selektion von Web-Services zur Integration in Workflows propagiert (vgl. z.B. [PrES04], [HeKZ11]). Die beiden erstgenannten Arbeiten nutzen neben vom Analytiker erhobenen Kriterien auch Datencharakteristika, die mit Verfah-

[184] Die Verfahrensauswahl ist nicht mit der in einigen Disziplinen der Datenanalyse gebräuchlichen Modellselektion identisch. Die Statistik versteht unter diesem auch als Modellanpassung bekannten Vorgang die Auswahl eines instanziierten Modells einer Modellfamilie. Im Maschinellen Lernen hingegen wird dieser Vorgang als Modellerstellung (Training, Kalibrierung) bezeichnet und erfolgt mithilfe eines speziellen Lernverfahrens, welches aus der Menge aller verfügbaren Modellerstellungsverfahren (Modellfamilien im Sinne der Statistik) ausgewählt wird (vgl. [HiKa01, 180]). In dieser Arbeit werden alle Verfahren der Datenanalyse – einschließlich jener zur Modellerstellung – als Analyseverfahren bezeichnet. Modelle werden als Ergebnisse modellbezogener Verfahren betrachtet.

renseigenschaften abgeglichen werden. Problematisch erscheint hierbei, dass die rigorose Aussonderung von Verfahrenskandidaten auf Basis nicht erfüllter datenbezogener Anforderungen die Möglichkeit ignoriert, bestimmte Dateneigenschaften durch Datentransformationen herzustellen.

Daher wird im Folgenden eine dreiteilige Unterscheidung eingeführt, die eine differenzierte Behandlung der Auswahlkriterien zulässt. Die Klassifizierung erfolgt anhand der *Strenge* einer Bedingung und kann bei Bedarf verändert werden, um den Auswahlvorgang zu lenken:

- Forderungen vom Typ *Restriktion* müssen zwingend erfüllt sein, damit ein Verfahren in die Auswahl aufgenommen wird. Sie sind als „Muss-Kriterien" zu betrachten.
- Forderungen vom Typ *Präferenz* stellen „Soll-Kriterien" dar, deren Erfüllung wünschenswert, aber nicht notwendig ist. Sie dienen der Bewertung und Reihung der Alternativen.
- Forderungen vom Typ *Transformation* kennzeichnen zunächst nicht erfüllte datenbezogene Bedingungen, die durch adäquate Datenvorbereitungsaufgaben erreichbar und daher als „Kann-Kriterien" zu verstehen sind.

Die um geeignete Anforderungen ergänzte Aufgabenspezifikation wird mit der `Funktion`, den `Eingabe-` und `Ausgabedaten` sowie dem `Verfahrensprofil` der **`Operatoren`** abgeglichen. Zunächst gelten sämtliche Spezifikationselemente als Restriktionen. Bei Diskrepanzen zwischen Eingabeflüssen und Eingabedaten können die zugehörigen „Muss-Kriterien" zu „Kann-Kriterien" gelockert werden. Analog sind Anforderungen auch als Präferenz deklarierbar (Abbildung 86). Es wird deutlich, dass Anforderungen, die zur Verfahrensauswahl beitragen sollen, stets gemäß den Deskriptoren des Verfahrensprofils zu formulieren sind.[185]

[185] Andernfalls sind Abbildungsregeln von Anforderungen auf Verfahrensprofil-Deskriptoren zu definieren. In den Deskriptorenkatalogen in Anhang A5 sind bewusst geeignete Schnittmengen zwischen den Elementen des Informationsbedarfsprofil und des Verfahrensprofils vorgesehen, um die Formulierung zu erleichtern.

Ein exemplarischer Deskriptorenkatalog zur Erstellung eines Verfahrensprofils mit Erläuterungen ist in Anhang A5.2 aufgeführt. Als Ausprägungen der Deskriptoren sind in vielen Fällen qualitative Angaben auf einer Ordinalskala empfehlenswert. HOGL verwendet etwa die Ausprägungen {`gering`, `mittel`, `hoch`} für den Großteil seiner Deskriptoren [Hogl03, 90f.]. Das Assistenzsystem von SPOTT & NAUCK erlaubt dem Analytiker, Anforderungen mithilfe eines Schiebereglers zu bestimmen. Zur Abbildung quantitativer Maße auf qualitative Variablen verwenden sie Fuzzy Logic [SpNa09, 3, 8ff.]. Mit welcher Qualitätsstufe ein Verfahren bezüglich einzelner Maße zu annotieren ist, kann aus den Erfahrungen früherer Anwendungen induktiv abgeleitet werden (vgl. Abschnitt 7.4.2.2).

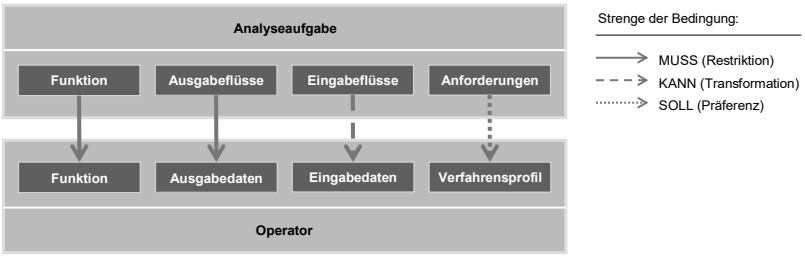


Abbildung 86: Abgleich von Aufgaben- und Operatorspezifikation (eigene Darstellung)

Allgemeingültige Regeln, welche Verfahrenscharakteristika in welcher Weise zur Erfüllung welcher Bedingungen beitragen, existieren nicht.[186] Vielmehr sind verfahrens(klassen)spezifische Überlegungen anzustellen, deren Diskussion die Möglichkeiten der vorliegenden Arbeit übersteigt. Der interessierte Leser sei auf die Literatur zu Analyseverfahren und Algorithmusvergleichsstudien[187] verwiesen. Der hier vorgestellte Ansatz ermöglicht die methodisch gestützte Einschränkung der Alternativenmenge auf Grundlage bekannter Verfahrenseigenschaften. Die Aufstellung von Bedingungen kann durch Regeln unter-

[186] Insbesondere gibt es „keine Methode, die für jedes Problem über alle Anwendungsgebiete die anderen im Ergebnis übertrifft. Der theoretische Beweis ist bekannt als das *No Free Lunch Theorem*“ [Lind05, 119].

[187] Vgl. z.B. [GaBr95], [CaNi04], [CaNi06], [LiLS00], [LeVo10].

stützt werden. Beispielsweise ergänzt die folgende Kontextregel die Analyseaufgabe um eine Anforderung zum Separierungsverhalten in Abhängigkeit vom Datentyp der abhängigen Variablen:[188]

```
IF Zielvariable.Wertebereich = _Real
THEN Anforderungen->add(
    new(Deskriptor(Separierung des Suchraums,
    kontinuierlich))) ENDIF
```

Identifikation einer funktional geeigneten Verfahrensklasse

Die Analyseaufgabe (P1.1) ist im Regelfall direkt mit einer Funktion der Operatortaxonomie verknüpft. Die dieser Funktion zugeordneten Operatoren bilden somit die initiale Klasse geeigneter Verfahren, wobei implizit die Restriktionen bezüglich Funktion und Ausgabeflüssen berücksichtigt werden. Abhängig vom Detailgrad der Aufgabenspezifikation resultiert daraus eine unterschiedlich umfangreiche, typischerweise nicht leere Kandidatenmenge, da in der Hierarchie höherstehende Funktionen sämtliche Operatoren speziellerer Funktionen subsumieren. Ihr Umfang lässt sich durch Variation der Aufgabenspezifikation gezielt erweitern oder einschränken. Wird kein passender Operator gefunden, ist eine allgemeinere Spezifikation in Betracht zu ziehen, sofern dies keine Abweichung vom Analyseziel zur Folge hat. Bleibt die Kandidatenmenge dennoch leer, ist die Aufgabe mithilfe der bereitstehenden Analyseverfahren nicht ausführbar.[189]

Einschränkung der Verfahrensklasse mithilfe von Anforderungen

Bei nicht-leerer Kandidatenmenge können `Anforderungen` der Aufgabe einbezogen werden, um die Verfahrensklasse weiter einzuschränken. Sie werden zunächst als Restriktionen behandelt. Verbleibt nach dieser Verschärfung der Auswahlkriterien kein passendes

---

[188] Vgl. zu diesem Zusammenhang [Pyle99, 109].

[189] Alternativ kann die Suche auch unter Umgehung der Taxonomie durch direkten Abgleich der Ausgabeflüsse mit den Ausgabedaten verfügbarer Operatoren erfolgen. Die taxonomiegestützte Auswahl zieht den Abgleich der Schnittstellen auf den Zeitpunkt der Taxonomieerstellung vor. Erfahrenen Analytikern gelingt es häufig, direkt eine passende Funktion zu identifizieren.

Verfahren, können einzelne Anforderungen selektiv zu Präferenzen abgeschwächt werden, um eine Lösung zu erhalten. Welche Restriktionen hierfür infrage kommen, ist situativ zu entscheiden.

Die Modellierung der Verfahrenseigenschaften als undefinierte Menge von Deskriptoren lässt den Fall zu, dass Operatoren nicht bezüglich aller Anforderungen gekennzeichnet sind (z.B. wenn einzelne Kriterien auf einen Operator nicht zutreffen, oder in Folge lückenhafter Annotation). Solche Anforderungen führen nicht zum Ausschluss des Verfahrens aus der Kandidatenmenge, sondern werden als „unbekannt" markiert. Der Analytiker kann sodann ihm bekannte Eigenschaften ergänzen bzw. markierte Operatoren selektiv eliminieren.

Es empfiehlt sich, sowohl Anforderungen als auch Verfahrenseigenschaften auf eine überschaubare Anzahl relevanter und verständlicher Kriterien zu begrenzen. Verwandte Arbeiten zeigen, dass bereits wenige Kriterien eine starke Reduzierung der Kandidatenmenge ermöglichen [LiSt99, 421]. HILARIO ET AL. repräsentieren Implementierungseigenschaften der Algorithmen in einer Ontologie, die nur automatisiert handhabbar ist. Die aufwändige Recherche dieser Charakteristika stellt selbst nach Einschätzung der Autoren eine langfristige Aufgabe dar [HND+11, 15]. Die Annotation von Operatoren geschieht daher im Idealfall durch den Entwickler. Andernfalls kann der Analytiker Deskriptoren bei Bedarf aus der Literatur oder konkreten Erfahrungen ergänzen, was der Vorstellung eines lernenden Systems folgt.[190]

5.5.3.4 Auswahl eines Analyseverfahrens (P1.4)

Für die endgültige Verfahrensauswahl sind die verbleibenden Elemente der Kandidatenmenge einer *Bewertung* zu unterziehen. Diese Bewertung erfolgt im Hinblick auf die als Präferenz gekennzeichneten Kriterien und lässt sich durch Einbeziehung von Restriktionen erweitern. Damit kann nicht nur berücksichtigt werden, ob, sondern auch in welchem Ausmaß ein Verfahren gesetzte Anforderungen erfüllt. Darüber hinaus

[190] Die Extraktion von Operatorcharakteristika aus Anwendungserfahrungen kann durch Maßnahmen der Erfahrungssicherung unterstützt werden (vgl. hierzu Abschnitt 7.4.2.2).

wird der zu erwartende Aufwand für notwendige Datentransformationen in die Bewertung einbezogen.

Beurteilung der Performanz von Analyseverfahren

Ein wichtiges Bewertungskriterium ist die *Performanz* (Leistungsfähigkeit) eines Verfahrens, die insbesondere durch die Genauigkeit der Ergebnisse und das Laufzeitverhalten bestimmt ist. Problematisch ist hierbei, dass Performanzmaße nur a posteriori, d.h. nach Anwendung des Verfahrens, quantifizierbar sind. Zum Zeitpunkt der Verfahrensauswahl sind allenfalls erfahrungsgeleitete Schätzungen möglich [HiWi01, 66]. Aus diesem Grund geschieht die performanzbasierte Verfahrensauswahl meist experimentell durch Ausprobieren mehrerer Optionen. Für viele Auswahloptionen ist dieses Vorgehen nicht praktikabel [HiKa01, 181], und auch für kleine Kandidatenmengen stehen selten genügend Ressourcen für Vergleichsstudien bereit.[191]

Somit stellt sich die Frage, ob die Resultate solcher Experimente antizipierbar sind [Liu99, 358]. Eine erste Möglichkeit ist die Abschätzung der Performanz durch *Anwendung der Verfahren auf Stichproben* anstelle der vollständigen Datenmenge [Petr00], [BePr01, 6]. Alternativ ist die *Anwendung von Landmarker-Verfahren* denkbar. Landmarker sind besonders einfache Analyseverfahren mit geringer Zeitkomplexität. Die Ausführung eines repräsentativen Landmarkers stellvertretend für jedes Element der Kandidatenmenge ist daher wesentlich effizienter als das direkte Ausprobieren aller Verfahren. Dabei wird unterstellt, dass die realisierte Performanz Rückschlüsse zulässt, welche Verfahrensklasse aufgrund ihrer Eigenschaften (z.B. lineare Separabilität) für die vorliegenden Daten am besten geeignet ist [BeGi00, 325f.].[192]

---

[191] LINDNER berichtet für die von ihm durchgeführten Vergleichsstudien über Laufzeiten zwischen unter einer Sekunde bis zu mehreren Tagen [Lind05, 139].

[192] Beispiele für Landmarker sind Entscheidungsknoten, schlechtester Knoten, zufällig ausgewählter Knoten, Naive Bayes, 1-Nearest-Neighbour, lineare Diskriminante [BeGi00, 326].

Eine zum Entscheidungszeitpunkt effizientere Option bietet *Meta-Learning*.[193] Hierbei werden auf Basis von Performanzwerten früherer Analysefälle sowie unter Berücksichtigung von Verfahrens- und Datencharakteristika Klassifikationsmodelle trainiert, Auswahlregeln induziert oder ähnlichkeitsbasierte Ansätze eingesetzt, um die Operatorauswahl zu bestreiten [BeGi00, 325]. Meta-Learning wird erstmals im STATLOG-Projekt [MiST94] eingesetzt. Die Gewinnung von Performanzmaßen erfolgt dabei durch experimentelle Anwendung gegebener Algorithmen auf eine kleine Zahl von Testdatensätzen. Um die Generalisierbarkeit der induzierten Auswahlregeln durch eine breitere Basis an Referenzanwendungen zu verbessern (vgl. [Lind05, 219]), wird im METAL-Projekt [Meta01] ein Ratgebersystem entwickelt, das aktuelle Performanzschätzungen aus früheren Anwendungsfällen mit ähnlichem Datenkontext ableitet [BSCP03]. LINDNER ET AL. erweitern den dabei verfolgten Case-based-Reasoning-Ansatz um die Einbeziehung von Nutzeranforderungen [LiSt99], [Lind05, 119ff.]. Dabei werden Meta-Learner nicht direkt zur Algorithmusselektion eingesetzt, sondern zur Beurteilung der Relevanz einzelner Daten- und Verfahrenscharakteristika für die Selektionsentscheidung [ThLi98, 5f.]. METAL sowie HILARIO & KALOUSIS [HiKa01] beziehen durch Landmarking ermittelte Kenngrößen beim Meta-Learning ein. Letztere entwickeln ein fallbasiertes Assistenzsystem für die Algorithmenauswahl, das auch verschiedene Konfigurationen der Verfahrensparameter unterscheidet [HiKa01, 190].[194]

Die Einbeziehung von Performanzmaßen in die Verfahrensauswahl lässt eine genauere Beurteilung der zu erwartenden Ergebnisqualität zu. Dies kann einerseits manuell geschehen, indem aus Experimenten gewonnene Schätzungen als Präferenz in die Priorisierung einfließen. Andererseits können Performanzschätzungen aus Empfehlungssystemen der geschilderten Art als Bestandteile des Verfahrensprofils modelliert werden. Die an der **`Prozessinstanz`** gespeicherten An-

[193] Meta-Learning verfolgt die Idee, das Maschinelle Lernen (gewissermaßen rekursiv) für die Modellselektion beim Maschinellen Lernen einzusetzen [BeGi00, 325].

[194] Die Fallbasis ihres Systems umfasst über 1.350 Datenbestände, ca. 37.500 Charakteristika über die Datenbestände und ca. 11.700 Algorithmenanwendungen von 9 Algorithmen [HiKa01, 188].

gaben erlauben die Auswertung des Einflusses von Verfahrens- und Datencharakteristika auf Performanzgrößen (Attribut `Ergebnisbewertung`, vgl. Abschnitt 4.5.3).

Einbeziehung von Datencharakteristika

Die Anwendbarkeit eines Analyseverfahrens sowie dessen Präferabilität bezüglich der zu erwartenden Ergebnisqualität ist typischerweise von Datencharakteristika abhängig, welche die Verfügbarkeit (z.B. fehlende Werte), den Inhalt (z.B. Annahmen zur Werteverteilung oder Unabhängigkeit von Variablen) oder die Repräsentation (z.B. Datentyp) betreffen. Viele Dateneigenschaften lassen sich durch entsprechende Transformation derart verändern, dass sie den Verfahrensanforderungen genügen. Somit hängt die Auswahl eines Operators letztlich davon ab, wie viel Aufwand der Analytiker bereit ist, in die Datenvorbereitung zu investieren [EnTh98, 430]. Hierbei gilt das allgemeine Formalziel, dass der allein zur Herstellung der Verfahrensvoraussetzungen nötige Transformationsaufwand zu minimieren ist [BeLi97, 415]. Er beeinflusst somit die Geeignetheit des Verfahrens.

Zunächst sind auch datenbezogene Anforderungen als Restriktionen gekennzeichnet. Bevor Verfahrenskandidaten, welche diese Bedingungen nicht erfüllen, jedoch aus der Kandidatenmenge entfernt werden, kann der Analytiker für jede betreffende Bedingung einzeln entscheiden, ob diese alternativ als Transformation oder Präferenz zu behandeln ist. Für jene Bedingungen, die durch Datentransformation erfüllt werden können und sollen, sind entsprechende Aufgaben für die Datenvorbereitungsphase vorzumerken. Die Bedingung wird als *Transformation* gekennzeichnet und mit einer Bewertung des zu erwartenden Aufwands versehen, die im nächsten Schritt in die Berechnung der Rangordnung eingeht. Diese Bewertung kann im einfachsten Fall als Mengenzähler pro Transformationsaufgabe ausgeprägt sein. Liegen konkrete Aufwandsschätzungen in Zeit- oder Kosteneinheiten vor, können auch diese verwendet werden.[195] Beschließt der Analytiker, eine

[195] Hierbei ist auf einheitliche Bezugsgrößen sowie auf belastbare Zahlenwerte der Aufwandsschätzung über alle Anwendungsfälle zu achten. Im Idealfall werden die

unerfüllte Bedingung als *Präferenz* zu behandeln, so geht diese negativ in die Berechnung der Rangfolge ein. Die Vormerkung einer Transformationsaufgabe unterbleibt. Dieser Fall ist dann angezeigt, wenn eine Dateneigenschaft die Anwendbarkeit des Verfahrens nicht ausschließt, sondern etwa nur die Effizienz der Verfahrensanwendung beeinträchtigt (z.B. leiden Entscheidungsbaumverfahren unter vielen Merkmalsausprägungen). Zahlreiche Beispiele für Auswirkungen diverser Datencharakteristika auf die Algorithmenanwendung erörtern z.B. [BeLi97, 415ff.] und [Pyle99, 102ff.].

Anforderung	erfüllt	Strenge	Transformation	Aufwand
Anzahl Beispiele: gering	☐	Soll		
Eingabevariablen: normalverteilt	☐	*Kann*	Normalisierung	3
Eingabevariablen: lin. unabhängig	☑	Muss		
Zielvariable.Skala: nominal	☐	*Kann*	Diskretisierung	1

Tabelle 2: Beispiele für datenbezogene Anforderungen eines Operators und zugehörige Transformationsaufgaben

In Tabelle 2 sind beispielhafte datenbezogene Anforderungen eines Operators aufgeführt, von denen zunächst nur die lineare Unabhängigkeit der Eingabevariablen erfüllt ist. Von den nicht erfüllten Anforderungen wird die geringe Zahl der Datensätze als nicht zu behandelnde Präferenz (Soll-Kriterium) markiert, da sie nur das Laufzeitverhalten des Operators beeinflusst. Die geforderte Normalverteilung der Eingabevariablen wird als Kann-Kriterium klassifiziert, das durch Normalisierung erreicht werden kann. Die Transformation wird mit Aufwandsfaktor 3 bewertet. Ebenso wird eine Diskretisierung der nicht nominalskalierten Zielvariablen mit Aufwand 1 vorgemerkt, um die betreffende Voraussetzung herzustellen.

Aufwände pro Datensatz oder Attribut quantifiziert und für den vorliegenden Datenbestand exakt berechnet.

Priorisierung und Auswahl von Operatoren

Alle Restriktionen, Präferenzen und Transformationen können in die Auswahl des letztlich einzusetzenden Analyseverfahrens einfließen. Die Entscheidung, welche Kriterien im konkreten Fall tatsächlich berücksichtigt werden, obliegt dem Analytiker. Hierbei ist zu bedenken, dass die Kriterien einerseits teilweise in konfliktärer Beziehung stehen [HiWi01, 66], andererseits je nach Anwendungskontext unterschiedliche Bedeutung besitzen können. So wiegen z.B. bei der Kreditwürdigkeitsbeurteilung die Kosten der Fehlklassifikation schwerer als die Genauigkeit [Liu99, 358f.]. Derartige Zusammenhänge können in den `Präferenzrelationen` der Analyseaufgabe hinterlegt werden. Eine Unterstützung der Auswahlentscheidung durch Kontextregeln ist möglich. Wird eine Analyseaufgabe aus früheren Analyseprozessen wiederverwendet (vgl. Abschnitt 4.5.4.1), können dort hinterlegte, vormals bewährte Kriterien genutzt werden und zur Qualitätssicherung und Verstetigung von Planungsentscheidungen beitragen.

Operator i		**Kriterium k [Gewicht g_k]**			**Gesamt wertung**	**Rang**
		Interpretierbarkeit [g_1=0,2]	Genauigkeit [g_2=0,6]	Transformation [g_3=0,2]	($\Sigma_{i,k}\ g_i w_{i,k}$) ($\rightarrow$ min)	
⑦	**Decision Tree Learner**	1 = hoch	1= hoch	0	0,8	**1**
⑥	**Support Vector Machine Learner**	2 = mittel	1 = hoch	1	1,2	**2**
②	**Fuzzy Rule Learner**	2 = mittel	2= mittel	1	1,8	**3**

Tabelle 3: Vereinfachtes Beispiel zur Priorisierung von Analyseverfahren einer Kandidatenmenge

Auf Grundlage der Kriterien wird eine Rangliste aller Operatoren aus der Kandidatenmenge erstellt. Hierbei kann die Einflussstärke eines

Kriteriums k mithilfe von Gewichten g_k individuell festgelegt werden. Die Gesamtbewertung eines Verfahrens ist die Summe der gewichteten Einzelbewertungen w_{ik} der Operatoren i. Diese ist zu minimieren,[196] um den geeignetsten Operator zu ermitteln. Tabelle 3 zeigt beispielhaft die Bewertung dreier Verfahren nach gewichteten Kriterien, der zufolge der `Decision Tree Learner` (7) zu favorisieren ist.

Die Auswahlentscheidung lässt sich durch Variation der Kriterien unter alternativen Prämissen fällen. So besteht die Möglichkeit, einzelne Kriterien durch Gewichtung mit dem Faktor $g=0$ (vorübergehend) aus der Bewertung auszuschließen. Ebenso sind Reihungen nach einzelnen Kriterien möglich, etwa nach der Genauigkeit, wonach z.B. Entscheidungsbaum (7) und Stützvektoren (6) dieselbe Bewertung aufweisen. Auf diese Weise können Entscheidungen selektiv revidiert und die Auswirkungen auf die Operatorbewertung im Sinne einer Szenarioanalyse („Was-wäre-wenn-Funktion“) beobachtet werden [CSG+92, 16]. Getroffene Entscheidungen können mithilfe von Anmerkungen nachvollziehbar dokumentiert werden (vgl. [EnLS97b, 6]).

5.5.3.5 Kontextabhängige Entwurfsentscheidungen (P1.5)

Abhängig vom jeweiligen Kontext können weitere Entwurfsentscheidungen hilfreich sein, um die Spezifikation der Analyseaufgabe für die vorliegende Situation zu vervollständigen.

Erweiterung um verfahrensspezifische Vorgaben

Um die zu produzierenden Analyseergebnisse möglichst konkret zu spezifizieren, können weitere `Anforderungen` definiert werden, die sich auf verfahrensspezifische Ergebnisgrößen (z.B. Anzahl erzeugter Regeln oder Cluster bei Regelinduktions- bzw. Segmentierungsverfahren) beziehen und erst mit Zuordnung eines Operators bekannt sind. Zusätzlich können an dieser Stelle `Bewertungskriterien`,

[196] Das Minimierungsziel der Bewertungsfunktion ergibt sich aus der Wahl der Bewertungsmaße, die sich teils auf Aufwände stützen und daher möglichst klein sein sollen. Die Maße könnten z.B. ebenso auf Grundlage eines Zielerreichungsgrades definiert werden, woraus ein Maximierungsziel folgen würde.

`Bewertungsfunktionen` und `Präferenzrelationen` (vgl. hierzu das Beispiel zur Kreditwürdigkeitsprüfung im vorigen Abschnitt) definiert werden. Für einen Operator zulässige Bewertungsoptionen dokumentiert dessen Attribut `Bewertungskriterien`.

Festlegung des Evaluationsansatzes

Modellbasierte Analyseverfahren bedürfen einer Evaluierung des erzeugten Modells (vgl. Abschnitt 3.1.2.3). Im überwachten maschinellen Lernen erfolgt zu diesem Zweck eine Aufteilung der Analysedaten, um das Modell mit je einer Teildatenmenge zu trainieren, kalibrieren und evaluieren. Ein Plan, der das Vorgehen bei der Evaluierung festlegt, wird als Test-Design oder *Evaluationsansatz* bezeichnet. Er legt die Strategie zur Aufteilung der Daten und die anzuwendenden Qualitätsmaße fest [CCK+00, 28]. Hinweise zu ihrem Einsatz enthält Abschnitt 7.2.1.1. Zulässige Optionen sind als `Evaluationsansatz` am gewählten Operator hinterlegt und verweisen jeweils auf **`Funktionen`**. Passende Aufgaben sind kontextspezifisch auszuwählen und zur Integration in den Analyseprozess vorzusehen. Zusammengehörige Datenvorbereitungs- und Ergebnisaufbereitungsaufgaben können in Prozessmodulen gekapselt werden

Kombination mehrerer Analyseverfahren

Ziel der Operatorauswahl ist grundsätzlich die Festlegung auf ein einzelnes Verfahren. Ist dies nicht möglich, etwa weil mehrere Alternativen gleiche oder ähnliche Gesamtbewertungen aufweisen, oder weil ihre individuellen Stärken gleichermaßen für die Aufgabenstellung relevant erscheinen, können mehrere Verfahren zur Realisierung einer Analyseaufgabe kombiniert werden. Bei modellbasierten Verfahren (Inferenzmodellen) ist damit oft eine erhebliche Verbesserung gegenüber Einzelmodellen erreichbar [WiFr00, 250f.]. Wichtige Formen der Modellkombination sind unter den Bezeichnungen Bagging, Boosting und Stacking bekannt und betreffen jeweils die parallele, sequenzielle und hierarchische Verknüpfung mehrerer Verfahren. Eine ausführliche Beschreibung hierzu liefert z.B. [WiFr00, 251-258]. Modellkombinationen sind durch spezielle Aktivitätsfolgen oder vordefinierte

Schablonen im Analyseprozess modellierbar[197] und werden von Analysewerkzeugen zum Teil durch spezielle Strukturkomponenten (z.B. Meta-Modeling-Schemata) unterstützt.

5.5.3.6 Zusammenfassung: Planung der Datenanalysephase

Die Planung der Analysephase dient der Spezifikation der Analyseaufgabe nach Maßgabe des Analyseproblems und der Zuordnung geeigneter Operatoren. Zugleich können Anforderungen, Bewertungskriterien und in späteren Planungsschritten zu berücksichtigende Prozessaufgaben bestimmt werden. Zwischen den beschriebenen Teilaufgaben bestehen Wechselwirkungen, weshalb eine streng sequenzielle Abarbeitung nicht realistisch erscheint (z.B. zieht die Datencharakterisierung eine Anpassung der Aufgabenspezifikation nach sich). Wesentliche Änderungen der Aufgabenspezifikation sind zur Problemspezifikation zu propagieren, wodurch eine Rückkopplung mit der Zielebene entsteht (vgl. [Enge99, 147]).

Aufgrund ihrer strukturellen Ähnlichkeit zu Operatoren können Prozessbausteine gleichberechtigt in die Verfahrensauswahl einfließen. Im Idealfall sind vollständige Workflows für die gegebene Aufgabe verfügbar. Prozessfragmente erlauben – neben den allgemeinen Potenzialen der Wiederverwendung – zudem eine Verbesserung der Alternativenbewertung auf Basis konkreter Prozessausführungen: Performanzmaße müssen nicht geschätzt werden, sondern stehen als erfahrungsbasierte Evidenzen zur Verfügung. Sofern die Module alle vor dem Analyseschritt ergriffenen Datentransformationen umfassen, sind diese Maße überdies uneingeschränkt vergleichbar. Damit wird eine wesentliche Schwäche des Performanzvergleichs behoben, der bezüglich unterschiedlicher Datenbestände nicht sinnvoll ist. Datencharakteristika sind

[197] Werden mehrere Modelle kombiniert, so kann für jedes Modell ein eigener Datenvorbereitungs- und Analyseprozess nötig sein, die in eine gemeinsame Interpretationsphase mit einer Teilaufgabe zur Modellauswahl münden. Fraglich ist hierbei, ob derartige Fälle überhaupt en détail geplant oder vielmehr situativ gehandhabt werden sollten. Wird die Verfahrensauswahl zum Bestandteil des Prozesses, so enthält dieser eine Metaaufgabe. Eine Vermischung von Meta- und Prozessaufgaben ist grundsätzlich zu vermeiden.

aber oft nur für Analyserohdaten, nicht jedoch für transformierte Eingabedaten der Operatoren verfügbar.

Die Einbeziehung von Prozessbausteinen erlaubt zudem die Anwendung der Methodik auf wenig strukturierte Analyseaufgaben, die sich in der Ausführung *eines* Operators (z.B. dem Aufruf eines OLAP-Berichts) erschöpfen. Die Auswahl des Operators ist hier trivial, während die Suche nach einer geeigneten Instanziierung seiner Parameter angesichts der großen Varietät problematisch ist. Der Aufruf parametrisierter Berichtsschablonen wird hierzu als Prozessaktivität repräsentiert.

5.5.4 Planung der Datenvorbereitungsphase (P2)

Die Planung der Datenvorbereitungsphase zielt auf die Konstruktion eines Prozessabschnitts, der die vorliegenden Analysedaten sowohl strukturell in das Eingabedatenformat der Analysephase transformiert, als auch inhaltlich durch geeignete Transformation auf die Erreichung des Analyseziels ausrichtet. Ergebnis dieses Schrittes ist ein Workflow-Ausschnitt, der alle hierfür nötigen Aktivitäten umfasst.

Dieser Planungsschritt gliedert sich in drei Teilaufgaben: Am Anfang steht die *Spezifikation der Transformationsaufgaben (P2.1)*, die danach durch *Zuordnung von Transformationsverfahren (P2.2)* in ausführbare Aktivitäten überführt werden. Am Ende steht die *Reihenfolgeplanung (P2.3)*, um die Aktivitäten in zulässiger und effizienter Weise zu verknüpfen. Das hier verfolgte Prinzip ist dem nicht-linearen Planen der Künstlichen Intelligenz angelehnt, bei dem Reihenfolgebeziehungen erst nachträglich in ungeordnete Pläne eingefügt werden. Die Pläne bestehen aus Prozessbausteinen, Abhängigkeiten zwischen den Bausteinen und noch zu erfüllenden Vorbedingungen [Hert89, 81ff.], [RuNo03, 388f.]. Teilaufgaben P2.1 und P2.2 legen die relevanten Bausteine und zugehörige Abhängigkeiten fest. Teilaufgabe P2.3 bestimmt weitere Abhängigkeiten und ermittelt auf ihrer Grundlage eine zulässige Reihenfolge.

5.5.4.1 Spezifikation der Datentransformationsaufgaben (P2.1)

Die zwischen gegebenen Analysedaten (Startzustand) und den Eingabeflüssen der Analyseaufgabe (Zielzustand) zu überbrückende strukturelle Barriere (vgl. Abschnitt 5.5.1) suggeriert ein triviales Planungsproblem, das mithilfe eines einfachen linearen Planungsalgorithmus lösbar ist: Demnach wäre ein Plan zu finden, der aus einem den Zielzustand produzierenden Operator O und weiteren Operatoren besteht, die rekursiv alle vom Startzustand noch nicht erfüllten Vorbedingungen von O herstellen (vgl. [Hert89, 50], [EnLS97, 165]).[198] Planungsansätze, die allein auf Grundlage der Vor- und Nachbedingungen der Prozessbausteine operieren und die Semantik der Transformationen ignorieren, liefern jedoch häufig ineffiziente[199] und für die Datenanalyse fehlerhafte oder gar sinnlose Prozesse.

So kann ein syntaktisch korrekter Plan nicht gewährleisten, dass er inhaltlich erforderliche oder hilfreiche Aufgaben (z.B. Datenbereinigung; Transformation der Merkmalswerteverteilung) enthält. Weiter rufen semantisch verschiedene Transformationen zuweilen syntaktisch gleiche Effekte hervor. Beispielsweise kann sowohl eine Datentypkonvertierung als auch eine Diskretisierung von Wertintervallen numerische Attributwerte in Zeichenketten überführen, obwohl letztere auch den Inhalt der Daten verändert. Umgekehrt rufen einige inhaltliche Modifikationen, wie etwa die Log-Transformation einer Variablen, keinerlei Änderung des Datenschemas hervor und werden von rein syntaxbasierter Planung nicht erfasst. Zudem ist denkbar, dass ein solcher Planer nicht zielführende Transformationen in den Prozess einfügt, um offene Bedingungen durch syntaktisch passende, aber inhaltlich ungeeignete Operatoren herzustellen.[200] Offensichtlich ist die

---

[198] Vor- und Nachbedingungen von Prozessbausteinen resultieren aus ein- bzw. ausgehenden Flussbeziehungen der Bausteine und sind erfüllt, wenn Daten des vom jeweiligen Fluss erwarteten bzw. produzierten Typs vorliegen (vgl. Abschnitt 4.5.2.2).

[199] So weist z.B. ENGELS auf die Gefahr ineffizienter Pläne als typische Eigenschaft von Means-End-Planern hin und sieht die Notwendigkeit der Einschränkung der Planung durch Kontrollregeln [Enge99, 182f.].

[200] So könnte die Bedingung klassifizierte Daten bereitzustellen, zur Aufnahme eines partitionierenden Verfahrens (z.B. Clustering) führen, das die nötigen Klassenmerk-

Semantik von Datentransformationsaufgaben, wie sie durch die `Funktion` beschrieben wird, zwingend in deren Auswahl einzubeziehen.

Die Bestimmung relevanter Aufgaben kann sich an verschiedenen Überlegungen orientieren. Aufgrund der Präferabilität der Wiederverwendung gegenüber der Neuplanung (vgl. Abschnitt 5.3.3) sollte zunächst der *Einsatz wiederverwendbarer Prozessmodule* geprüft werden. Aus den *Implikationen bereits getroffener Entwurfsentscheidungen* ergeben sich in der Regel wichtige Hinweise auf einzuplanende Aufgaben. Diese können ebenso aus der *Auswertung von Kontextregeln* resultieren. Alle Vorschläge sind vom Analytiker gründlich zu überprüfen, der zusätzliche Maßnahmen durch *situative Erwägungen* identifiziert. Die Definition eines zielführenden Vorbereitungsprozesses erfordert in der Regel die Berücksichtigung aller vier Optionen, die im Folgenden näher erläutert werden.

Einsatz wiederverwendbarer Prozessmodule

Im Idealfall kann zur Funktion Datenvorbereitung ein vollständiges Fragment aus der Bausteinbibliothek gefunden werden, das bezüglich Datenkontext (Datenquelle, Informationsobjekte) und Prozesskontext (Flussbeziehungen zur Analyseaufgabe) zum aktuellen Fall passt. Im Regelfall ist eine solch exakte Passung nicht zu erwarten, so dass partielle Fragmente oder Schablonen zu verwenden sind. Bei der Modulauswahl verdient der Anwendungskontext besondere Beachtung, um nicht irrtümlich wichtige Aussagen in den Daten zu verändern. So ist für eine Kundensegmentierung z.B. grundsätzlich die Beseitigung von Ausreißern angeraten. Richtet sich die Analyse aber z.B. auf die Minimierung von Zahlungsausfällen (Anwendung „Kunde.Zahlungsausfall, –“), so sind Ausreißer als potenzielle Problemfälle von großem Interesse, weshalb ein Prozessmodul in diesem Kontext keine Ausreißerbeseitigung enthalten sollte.

male in den Daten erzeugt, obwohl dieser Schritt dem Analyseziel entgegenläuft. Vgl. zur Motivation dieses Beispiels [KSBF10, 7f.].

Mit der Nutzung von Modulen, welche die Datenvorbereitung nicht vollständig beschreiben, werden Typ- und Aggregationsrelationen zwischen Funktionen und Prozessbausteinen relevant. Ist etwa die Funktion Datenbereinigung (F) zu konkretisieren, so sollten auch Module, welche die Ausreißerbeseitigung als Spezialisierung F^S von F realisieren, berücksichtigt werden. Grundsätzlich sind im Hinblick auf Funktion und Kontextfaktoren exakt passende Artefakte zu bevorzugen. Liegen solche nicht vor, folgt der Abruf von Prozessmodulen folgenden Regeln:

- **Spezialisierungsbeziehungen** (F^S *ist_ein* F): Liegt kein exakt passendes Modul M(F) vor, werden speziellere Bausteine $M(F^S)$ geprüft. Entsprechen sie bezüglich der Kontextfaktoren dem aktuellen Fall, ist ihre Integration auch bei Verfügbarkeit eines funktional passenden Moduls M(F) zu erwägen, da sie Hinweise zu dessen Anpassung geben können.
- **Generalisierungsbeziehungen** (F *ist_ein* F^G): Liegt kein exakt passendes Modul M(F) vor, können allgemeinere Bausteine $M(F^G)$ geprüft werden. Sie können die Substitution oder Anpassung bereits in den Prozess integrierter Bausteine $M(F)_{alt}$ nahelegen. Dies ist dann sinnvoll, wenn $M(F^G)$ mehr Kontextbedingungen erfüllt als $M(F)_{alt}$, also insbesondere dann, wenn $M(F)_{alt}$ kontextunabhängig ausgewählt wurde (etwa als initiales Teilprozessschema).
- **Zerlegungsbeziehungen** ($M^Z(F)$ *Teil_von* M(F)): Bausteine $M^Z(F)$, die Zerlegungsprodukte eines passenden Moduls sind, können die Funktion F nicht vollständig realisieren und werden nur dann geprüft, wenn sie bezüglich der Kontextfaktoren besser zum aktuellen Fall passen als M(F). $M^Z(F)$ kann sodann Vorschläge für Anpassungen an M(F) beisteuern.
- **Aggregationsbeziehungen** (A *Teil_von* A^A) können zwischen Problemaspekten (Anwendungskontext) relevant sein, etwa wenn ein Modul M(A) zur Betrugserkennung gesucht ist, aber nur ein Baustein $M(A^A)$ für den übergeordneten Kontext Betrugsvermeidung existiert.

Weitergehende Typ- und Aggregationsrelationen verlieren mit Umfang und Qualität der Prozessbibliothek tendenziell an Bedeutung, können die Planung aber dennoch bereichern. Da frühere Erfahrungen kaum alle fallspezifischen Belange abdecken können, sind selbst bei Vorliegen exakt passender Module immer auch die folgenden Optionen zu beachten.

Würdigung der Implikationen bereits getroffener Entwurfsentscheidungen

Bisher getroffene Entwurfsentscheidungen können die weitere Prozessplanung durch mit ihnen einhergehende Datenabhängigkeiten beeinflussen. Dies betrifft zum einen die für die Analyseaufgabe ausgewählten Operatoren. Nicht erfüllte Vorbedingungen (Eingabedaten) oder Anforderungen werden in Schritt P1.4 als *Transformationen (Kann-Kriterien)* gekennzeichnet. Diesen können nun geeignete Transformationsaufgaben zugeordnet und in den Prozess integriert werden. Anforderungen beschreiben wünschenswerte Eigenschaften der Analysedaten und gehen über rein syntaktische Aspekte hinaus. Sie stellen somit wertvolles Planungswissen bereit.

Zum anderen sind aus der Fallbibliothek abgerufene Prozessmodule auf unerfüllte Datenabhängigkeiten zu prüfen. Alle vom Modul erwarteten Eingabeflüsse sind adäquat zu versorgen. Dieser Aspekt entspricht dem Vorgehen der linearen Handlungsplanung (vgl. oben und [Hert89, 50], [EnLS97, 165]), bei dem nicht erfüllte Vorbedingungen sukzessive durch Aufnahme weiterer Bausteine hergestellt werden. Entsprechend sind Aufgaben in den Prozess einzuplanen, welche die geforderten Inputs produzieren.

Die Menge offener Vorbedingungen aller Prozessbausteine ist nach jedem Konstruktionsschritt neu zu ermitteln (vgl. [ZLKO97, 294]), da mit Aufnahme oder Streichung eines Bausteins Änderungen dieser Menge möglich sind. Insbesondere können zur Herstellung der Bedingungen eines Bausteins A eingeplante Aufgaben zugleich Vorbedingungen eines Bausteins B erfüllen. Die vollständige Menge der Prozessaufgaben nach Maßgabe dieses Planungsschrittes ist somit erst dann bekannt, wenn die Spezifikation des letzten Prozessabschnitts auf

Aktivitätsebene abgeschlossen ist und dessen Bausteine keine weiteren unerfüllten Vorbedingungen stellen.

Auswertung von Kontextregeln und situative Erwägungen des Analytikers

Die Mehrzahl der Entscheidungen ist vor dem Hintergrund des aktuellen Kontexts bzw. situativ zu treffen. Hierzu zählen auch die Basisansätze zur Neuplanung (Operatorkomposition und Aufgabendekomposition, Abschnitt 5.3.1). Unterstützung hierbei bieten neben allgemeinen Heuristiken insbesondere Kontextregeln. Letztere können in Prozessbausteinen oder Funktionen hinterlegt sein und dienen als *Planungsregeln* (vgl. Abschnitt 4.5.4.1), die bei Vorliegen bestimmter Kontextfaktoren die Aufnahme geeigneter Bausteine in den Prozess anmahnen.

Die Literatur enthält zahlreiche Empfehlungen und Heuristiken zur Planung von Analyseprozessen, die hier nicht wiedergegeben werden können. Vielmehr seien einige allgemeine Hinweise sowie Beispiele angeführt. Sie sind gemäß den drei Teilaufgaben der Datenvorbereitung in Datenselektion, Datenexploration und Datenmodifikation gegliedert (vgl. Abschnitt 3.1.2.2).

Datenselektion

Die Festlegung des `Analyseobjekts` während der Spezifikation des **`Analyseproblems`** (Z3.2, Abschnitt 5.4.5.2) gibt die Datenquelle und den Analysedatenbestand vor. Um diese Daten für den Prozess verfügbar zu machen, ist mindestens eine Aufgabe vorzusehen, welche relevante Fälle und Merkmale aus der Datenquelle extrahiert (Funktion *Datenzugriff*). Die inhaltlichen Kriterien sind direkt der `Analysefrage` zu entnehmen (Z3.1, Abschnitt 5.4.5.1): Die Selektionsdimensionen bilden die Bedingungen für die inhaltliche Auswahl von Fällen (z.B. Datensätzen). Aussageargumente und Beschreibungsdimensionen bestimmen die zwingend erforderlichen Merkmale (z.B. Attribute). Abbildung 87 zeigt für den Fall relationaler Datenquellen, wie die Klauseln einer SQL-Anfrage aus den Strukturelementen der Analysefrage ableitbar sind.

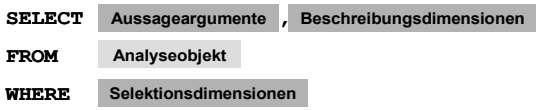


Abbildung 87: Spezifikation der Datenselektionsaufgabe aus der Fragestruktur (Analysefrage) im Falle relationaler Daten (eigene Darstellung)

Hierbei wird zunächst nur die Extraktion der relevanten Daten aus den gewählten Quellen betrachtet; etwaige Berechnungen, z.B. zur Aggregation von Aussageargumenten in Bezug auf Beschreibungsdimensionen, werden für die Datenselektion vernachlässigt.[201] Die Datenextraktion kann auch mehrere Aufgaben erfordern, etwa wenn auf mehrere Datenquellen zuzugreifen ist. In diesem Fall zählt auch die Verknüpfung der Informationsobjekte mehrerer Quellen zur Datenselektion, sofern es die Analyseaufgabe erfordert.

Die Vorgaben des Analyseproblems determinieren die Datenselektion indes nicht abschließend; die Aufnahme weiterer Merkmale oder Fälle ist zu jeder Zeit möglich. Insbesondere ist auf eine repräsentative und ausbalancierte Datenbasis zu achten. Anhaltspunkte für die entsprechenden Entscheidungen liefern Mengencharakteristika der Daten (z.B. Anzahl der Merkmale M, Fälle F, Klassen N, Fälle je Klasse F_K, etc.). Auf ihrer Grundlage lassen sich Kontextregeln spezifizieren, um Selektionsentscheidungen zu lenken. Zur Datenselektion zählt auch die Ziehung von Zufallsstichproben. Eine sehr einfache, als 6MN-Regel bekannte Heuristik ermittelt den minimalen Stichprobenumfang anhand von Datencharakteristika z.B. als $6 \times M \times N$ [DeHa01, 68].

Datenexploration

Die Datenexploration fungiert einerseits als Meta-Aufgabe zur Unterstützung der Prozessplanung, andererseits dient sie als Prozessaufgabe dem Kennenlernen der Daten, um die Bewertung und Interpretation der Analyseergebnisse zu erleichtern (vgl. [Enge99, 149, 159]). Neben der

[201] Derartige Berechnungen sind Gegenstand der Datenmodifikation bzw. -analyse. Sie können im Einzelfall in eine SQL-Anfrage zur Datenextraktion integriert werden, soweit sie an dieser Stelle bereits bekannt sind und falls diesbezügliche Flexibilität in Bezug auf Datenschema und Datencodierung nicht erforderlich ist.

Berechnung von Datencharakteristika in Schritt P1.2 (Abschnitt 5.5.3.2) ist grundsätzlich eine Visualisierung der selektierten Daten bzw. einzelner Merkmale, z.B. als Streudiagramm, Histogramm oder Box Plot, ratsam. Auffälligkeiten legen häufig genauere Betrachtungen nahe [Pyle99, 145]. Sie ergeben sich situativ und sind einer Planung kaum zugänglich.

Datenmodifikation

Entscheidungen über anzuwendende Datenmodifikationen erfordern Methoden- und Erfahrungswissen, das sich gut in Form von Kontextregeln formulieren lässt. Im Folgenden werden zwei Beispiele für Regeln präsentiert, die auf Datencharakteristika beruhende Heuristiken abbilden.

Die erste Regel nimmt eine Aufgabe zur Bereinigung fehlender Werte für das Merkmal (z.B. Tabellenspalte) M in den Prozess auf, wenn die Datencharakteristik `Anzahl_Fehlwerte` (im `Wert`-Attribut des Informationsobjekts) auf deren Existenz hinweist:

```
IF M.Wert.Anzahl_Fehlwerte > 0
THEN insert(new(Aufgabe→Funktion=„Fehlende Werte
bereinigen“,M)) ENDIF
```

Die zweite Regel berechnet situativ das Datencharakteristikum Schiefe (symbolisiert durch den Pseudo-Funktionsaufruf `Schiefe()`) und veranlasst eine Log-Transformation, falls ihr Betrag größer als 0,5 ist. Die damit erreichte gleichmäßigere Werteverteilung verbessert die Interpretierbarkeit statistischer Maße und den Erfüllungsgrad mancher Verfahrensannahmen [Tuft74, 108]:

```
IF M.Schiefe() < -0.5 OR M.Schiefe() > 0.5
THEN T1→Eingabeflüsse->add(
  new(Flussbeziehung(Typ: Synchronisation,
      Startaufgabe:
      new(Aufgabe→Funktion=„Log-Transformation“,M),
      Zielaufgabe: T1))) ENDIF
```

Integration der Aufgaben in den Prozess

Zur Aufnahme in den Prozess identifizierte Aufgaben werden zunächst in einer für die gesamte Datenvorbereitungsphase geltenden Liste

vorgemerkt, die in Anlehnung an die Handlungsplanung der Künstlichen Intelligenz als *Agenda* bezeichnet wird (vgl. [Hert89, 191]). Sie ist sowohl bei manueller als auch bei werkzeuggestützter Planung leicht realisierbar. Dieser Fall ist in der ersten Beispielregel durch den `insert()`-Operator angedeutet. Alternativ kann jede neue Aufgabe einer bestehenden Aufgabe zugeordnet werden, der sie entweder vorausgehen oder nachfolgen soll. Dies lässt sich leicht unter Nutzung der Flussbeziehung vom Typ **`Synchronisation`** abbilden (vgl. Abschnitt 4.5.1.3). Die zugehörigen Kontextregeln fügen in diesem Fall der betreffenden bestehenden Aufgabe einen Synchronisationsfluss zur neuen Aufgabe hinzu. Die zweite Regel oben zeigt diesen Fall anhand einer Aufgabe T1, der eine neue Aufgabe vorangestellt wird. Hierbei wird die neue Aufgabe typischerweise jener Prozessaufgabe zugeordnet, welcher die zugehörige Kontextregel annotiert ist. Hierbei übernehmen Aufgaben alle Regeln ihrer Funktion.

5.5.4.2 Zuordnung von Transformationsverfahren (P2.2)

Für jede in die Datenvorbereitung zu übernehmende Aufgabe ist ein geeignetes Verfahren auszuwählen. Hierbei ist analog zur Auswahl von Analyseverfahren vorzugehen, wie in Abschnitt 5.5.3.3 geschildert: Die Aufgabenspezifikation verweist auf einen Eintrag in der Funktionstaxonomie, und die ihm zugeordneten **`Operatoren`** definieren die Klasse funktional passender Verfahren. Umfasst diese Menge nur ein Element, ist mit dessen Aufnahme die Aktivitätsspezifikation der betrachteten Aufgabe abgeschlossen. Stehen mehrere Operatoren zur Auswahl, ist ein Entscheidungsproblem zu lösen. Häufig ist dies ein Hinweis auf weiteres Konkretisierungspotenzial. Operatoren, die eine Aufgabe nur partiell realisieren, können mit weiteren Operatoren direkt zu einem Aktivitätsnetz verknüpft werden (vgl. Basisansatz Operatorkomposition sowie [HeKZ11, 91]). Liefert der funktionale Abgleich mehrere funktional äquivalente Operatoren, können diese anhand nicht-

funktionaler Kriterien in der Rolle von Restriktionen oder Präferenzen[202] eingeschränkt bzw. priorisiert werden (vgl. Abschnitt 5.5.3.4).

Die nicht-funktionalen Anforderungen richten sich wiederum auf das `Verfahrensprofil`, wobei Typ- und Verhaltenscharakteristika für Transformationsverfahren zu vernachlässigen sind. Von Bedeutung sind hingegen Nutzen- und Kostenmerkmale sowie `Leistungsfaktoren`. Geeignete Qualitätsgrößen für serviceorientierte Systeme diskutieren z.B. [BHMS03], [Ran03], [TGRS04].

Gegenüber der Analysephase umfasst die Datenvorbereitungsphase typischerweise weitaus mehr Aktivitäten. Eine detaillierte Einzelbewertung aller Verfahrensoptionen für jede Aufgabe ist daher oft nicht realistisch. Einen einfachen Ansatz zur Bewertung ganzer innovativ konstruierter Prozessabschnitte durch heuristische Qualitätsgrößen beschreiben BERNSTEIN ET AL. [BePr01], [BeHP02]. Da häufig keine Aktivitätskombination existiert, die hinsichtlich aller Kriterien alle Alternativen dominiert, können Präferenzfunktionen definiert werden, welche die Einzelkriterien gewichtet zu einem Gesamtpräferenzwert verdichten. Die Maximierung der Präferenzfunktion liefert die beste Kombination.[203] Einen vergleichbaren Ansatz zur Auswahl von Web Services im Kontext des semantischen Prozessmanagements erläutern HEINRICH ET AL. [HeKZ11].

5.5.4.3 Reihenfolgeplanung (P2.3)

Die Reihenfolgeplanung fußt auf Abhängigkeiten zwischen Prozessbausteinen, die aus Flussbeziehungen resultieren und zulässige Verknüpfungen einschränken. Die Abhängigkeiten erlegen den Aufgaben eine fachlich-logische und zeitlich-technische Ordnung auf (vgl. [Gait83, 28]). Letztere wird aus **`Synchronisations`**- und **`Datenabhängigkeitsflüssen`** deduziert, wie sie mit der Auswahl von Auf-

[202] Anforderungen vom Typ Transformation sind im Allgemeinen nur für Analyseverfahren von Interesse.

[203] Hierbei ist zu beachten, dass nicht alle Kriterien durch Summation aggregierbar sind. So entspricht die Gesamtausfallwahrscheinlichkeit z.B. der Gegenwahrscheinlichkeit dafür, dass kein Operator ausfällt [HeKZ11, 95].

gaben und Operatoren (Schritte P2.1 und P2.2) bestimmt sind. Erstere ergibt sich aus der Semantik der Aufgaben sowie dem Analyseziel und ist in diesem Schritt festzulegen. Sie wird wiederum auf Synchronisationsflüsse abgebildet.

Festlegung fachlicher Abhängigkeiten

Die Reihenfolge von Aufgaben kann Effektivität und Effizienz eines Prozesses wesentlich beeinflussen. Auf welche Weise die Verknüpfung welcher Aufgaben die Effektivität beeinflusst, ist nicht allgemeingültig zu beantworten. Beispielsweise führt eine Ausreißerbeseitigung vor der Transformation der Werteverteilung zu anderen Ergebnissen als in umgekehrter Folge [AmCo94, 51]. In welche Reihung beide Aufgaben zu bringen sind, hängt wiederum vom Zweck der Datenmodifikation bzw. vom Analyseziel ab. Wird z.B. die Eliminierung seltener, untypischer Werte angestrebt, ist die Ausreißerbeseitigung vor der Variablentransformation (z.B. Standardisierung) sinnvoll. Soll hingegen der Wertebereich auf ein bestimmtes Intervall eingeschränkt werden, sollte die Anordnung gespiegelt werden. Derartige Heuristiken über Prozessstrukturen werden am besten in Form von kontextspezifischen Prozessmodulen codiert.

Einfacher fällt die Formulierung allgemeingültiger Regeln im Hinblick auf die Prozesseffizienz. So lässt sich etwa mithilfe operatorspezifischer `Leistungsfaktoren` eine zeitminimale Reihung festlegen. Hierzu wird eine Aktivität bestimmt, bezüglich der eine Zielgröße zu optimieren ist, etwa die Ausführungszeit einer Aktivität Z, die in Abhängigkeit der Zahl zu verarbeitender Datensätze definiert ist. Existieren nun Aktivitäten V_i, die eine Reduzierung der Datensätze bewirken, können diese vor Aktivität Z platziert werden ($\{V_i\} < Z$), um die Effizienz von Z zu steigern. Innerhalb der V_i ist ein analoges Vorgehen, auch in Bezug auf andere Zielgrößen, möglich. In vielen Fällen, insbesondere bei einmaligen Untersuchungen, genügt häufig die Würdigung qualitativer Geschwindigkeitsmaße mit ordinaler Skala {`schnell, mittel, langsam`}, die als Element des `Verfahrensprofils` annotiert sind. Allein auf ihrer Basis lassen sich z.B. mehrere Aktivitäten mit verschiedenen Effizienzeigenschaften derart verschalten, dass zunächst schnelle Verfahren die Attributmenge so stark ein-

grenzen, sodass zunehmend komplexere und langsamere Verfahren einsetzbar werden und auf diese Weise möglichst viele irrelevante Attribute aussondern [WHKR00, 151]. Dieses Vorgehen folgt der intuitiven Faustregel, stets zuerst effiziente Aktivitäten einzuplanen, bevor ineffizientere Transformationen zum Einsatz kommen.

Fachliche Verknüpfungen sollten nur dann festgelegt werden, wenn tatsächlich ein Einfluss auf Effektivität oder Effizienz des Prozesses zu erwarten ist. Sie werden als Flussbeziehungen vom Typ **`Synchronisation`** modelliert (vgl. Abschnitt 4.5.1.3). Synchronisationen aus Schablonen oder von höheren Abstraktionsebenen einer Aufgabendekomposition werden zunächst an alle Zerlegungsprodukte propagiert und können dort bei Bedarf eliminiert oder durch Umhängen auf andere Prozessbausteine übertragen werden (Abbildung 88).

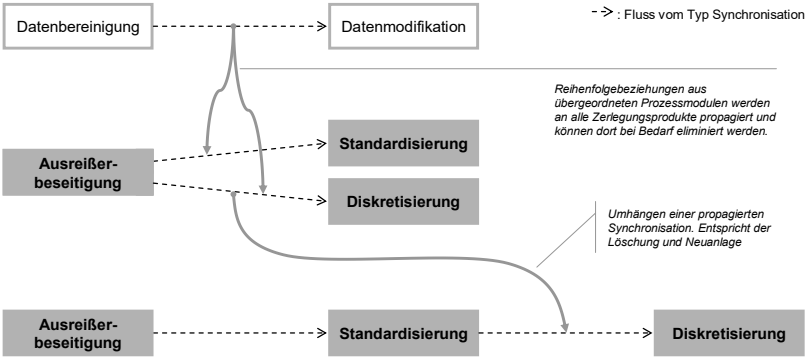


Abbildung 88: Beispiel zur Modellierung von fachlichen Abhängigkeiten (eigene Darstellung)

Überprüfung von Datenabhängigkeiten

Aus Sicht der Datenabhängigkeitsflüsse sind zwei Aufgaben A und B kompatibel, wenn B einen Fluss d konsumiert, der von A erzeugt wird. B ist dann von A abhängig (vgl. [Hert89, 93] und Abschnitt 4.5.1.3). Demnach ist die Kompatibilität von Prozessbausteinen auf Grundlage der Untersuchung von Datenabhängigkeiten feststellbar.

Die Ermittlung der Abhängigkeiten erfolgt nach einem von HEINRICH ET AL. [Hein+08] vorgeschlagenen Ansatz zur semantischen Prozess-

planung. Er ist auf die Schnittstellendeklaration von Prozessbausteinen mithilfe der Datenobjekttypen ihrer Ein- und Ausgabeflüsse in der Form (`Name`, `Wertebereich`, `Restriktionen`) abgestimmt (vgl. Abschnitt 4.6.1.1) und berücksichtigt auch semantische Relationen zwischen Datenobjekttypen, sofern diese ontologisch repräsentiert sind. Damit gelten zwei Bausteine nicht nur bei struktureller Gleichheit ihrer Flüsse als kompatibel, sondern auch dann, wenn diese in bestimmten Spezialisierungs- oder Zerlegungsbeziehungen stehen. Ferner werden auch deklarierte Restriktionen auf semantische Übereinstimmung geprüft. Die ontologische Konzeptualisierung erleichtert die Verarbeitung durch Inferenzmaschinen und ermöglicht somit die Voll- oder Teilautomatisierung der Planung [Hein+08, 446-448.].

Es wird jeweils ein Flusspaar (OUT_A, IN_B) zweier Aufgaben A und B verglichen und auf den Grad seiner Übereinstimmung geprüft. Eine Tabelle mit konkreten Prüfbedingungen enthält Anhang A6. Der Ausgabefluss OUT_A eines Bausteins A stimmt mit dem Eingabefluss IN_B eines Bausteins B überein, wenn zwischen ihnen semantische Relationen ableitbar sind, und wenn die Auswertung der zugehörigen Restriktionen r_{OUT_A} und r_{IN_B} keine disjunkten Mengen liefert. Sind beide Bedingungen erfüllt, ist B von A abhängig und als dessen Nachfolger einsetzbar. Andernfalls ist keine Verknüpfung möglich [Hein+08, 452].

Die Prüfung erfolgt entsprechend in zwei Stufen.[204] In der ersten Stufe werden die als `Datentyp` der Flüsse deklarierten Datenobjekttypen auf semantische Relationen vom Typ Gleichheit, Äquivalenz, Spezialisierung und Aggregation untersucht. Äquivalenz liegt etwa bei Informationsobjekten unterschiedlichen Namens vor, die bezüglich Wertebereich und Restriktionen typgleich sind, z.B. wenn die Tabelle `Kundenprofil` an anderer Stelle als `Trainingsdaten` auftritt. Spezialisierung kommt häufig bei elementaren Datentypen vor. So ist z.B. ein Attribut vom Typ `_Integer` ein zulässiger Input für eine Aufgabe, die ein Attribut vom Typ `_Numeric` erwartet, da der erste Datentyp eine Spezialisierung numerischer Datentypen darstellt. Eine Aggregations-

[204] Die folgende Darstellung ist eng an [Hein+08, 452ff.] angelehnt.

beziehung zwischen OUT_A und IN_B wird angenommen, wenn alle Elemente von OUT_A auch Elemente von IN_B sind. So erfüllt z.B. eine Relation $R_B(A_1, A_2)$ die Bedingungen einer Inputdeklaration, die eine Relation $R_A(A_1)$ erwartet, da die Attributmenge von R_A Teil jener von R_B ist. Vor diesem Hintergrund liegt eine *vollständige Übereinstimmung* eines Flusspaares vor,

- wenn OUT_A gleich oder äquivalent zu IN_B ist ($OUT_A = IN_B \vee OUT_A \equiv IN_B$);
- wenn OUT_A eine Spezialisierung von IN_B ist ($OUT_A \sqsubseteq IN_B$), da jede Ausprägung von OUT_A stets Ausprägung von IN_B ist und somit immer von B verarbeitet werden kann;
- wenn IN_B ein Teil von OUT_A ist ($OUT_A \succ IN_B$).

Vollständig übereinstimmende Flüsse sind uneingeschränkt verknüpfbar. Eine *partielle Übereinstimmung* ist in dieser Stufe gegeben, wenn IN_B eine Spezialisierung von OUT_A ist, da eine Ausprägung von OUT_A nicht zwingend auch Ausprägung von IN_B ist. Solche Flüsse sind nur unter Einschränkungen verknüpfbar (z.B. liefert A ein Attribut vom Typ `_Numeric`, B erwartet den spezielleren Typ `_Integer`). Alle anderen Fälle begründen *keine Übereinstimmung*.

Die zweite Stufe bezieht Restriktionen in die Untersuchung ein. Eine in erster Stufe ermittelte *vollständige Übereinstimmung* wird bestätigt, wenn die Restriktion von OUT_A eine Teilmenge der Restriktion von IN_B liefert. Beispielsweise kann B ein mit `not null` deklariertes Attribut aus A verarbeiten, wenn es diesem Attribut selbst keine oder dieselbe Restriktion auferlegt. Sind hingegen Schnittmenge und Differenzmenge der Restriktionen nicht leer, liegt nur noch *partielle Übereinstimmung* vor. Dies zeigt der umgekehrte Beispielfall, wenn B die `not-null`-Restriktion erhebt, die von A nicht beachtet wird. Eine partielle Übereinstimmung aus erster Stufe wird durch eine nicht-leere Schnittmenge bestätigt. Ist diese Schnittmenge hingegen leer, kann der Output von A nicht als Input von B verarbeitet werden, es wird *keine Übereinstimmung* erreicht. Dies ist häufig bei diskjunkten Werteintervallen gegeben, z.B. [100; 999] bei A und [-1; 1] bei B.

Da jeweils ein Flusspaar betrachtet wird, gilt der ermittelte Übereinstimmungsgrad nur lokal für die geprüften Flüsse, nicht global für alle Flüsse der zugehörigen Bausteine. Die Eingabeflüsse eines Bausteins B können von mehreren Bausteinen A_k gemeinsam versorgt werden (global nicht vollständige Übereinstimmung von Bausteinen, vgl. Abbildung 89 a). Eine Übereinstimmung ist global vollständig, wenn *ein* A alle Inputs von B bereitstellt (Abbildung 89 b) [DiPS09, 290f.]. Der lokale Übereinstimmungsgrad einzelner Flusspaare bleibt von der globalen Übereinstimmung unberührt. Zur besseren Unterscheidung wird der erste Fall als *multiple Abhängigkeit,* der zweite Fall als *totale Abhängigkeit* der Bausteine bezeichnet.

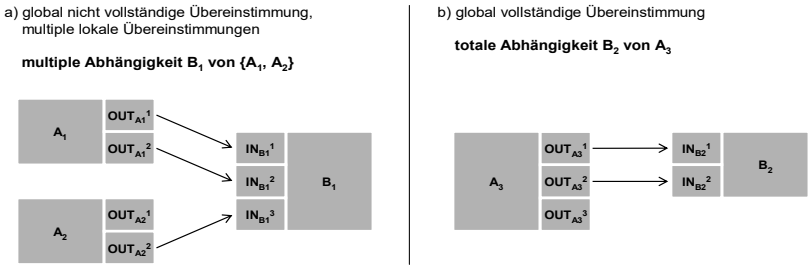


Abbildung 89: Vollständigkeit der Übereinstimmung von Prozessbausteinen, in Anlehnung an [DiPS09, 290f.]

Die Menge der in den Schritten P2.1 und P2.2 eingeplanten Aktivitäten ist nun anhand der Abhängigkeiten daraufhin zu untersuchen, ob ein zulässiger Verknüpfungspfad von den Datenzugriffsaufgaben (Startzustand) zur Analyseaufgabe (Zielzustand) herstellbar ist. HEINRICH ET AL. beschreiben einen Algorithmus zur Ableitung eines bipartiten Abhängigkeitsgraphen, der alle zulässigen Verknüpfungen der Aufgabenmenge darstellt [Hein+08, 453].[205] Bei manueller Prozesskonstruktion werden gegebenenfalls fehlende Aktivitäten nach situativen

[205] Die Überprüfung erfolgt rückwärtsverkettet und startet mit jenen Aktivitäten, die mindestens einen Eingabefluss des Zielzustands (vollständig oder partiell, aber nicht notwendig total) produzieren. Anschließend wird der Graph iterativ um Aktivitäten ergänzt, die Eingabeflüsse der enthaltenen Aktivitäten wiederum mit mindestens einem Fluss versorgen. Können alle Eingabeflüsse von eingeplanten Aktivitäten oder vom Startzustand versorgt werden, ist ein zulässiger Pfad gefunden [Hein+08, 454].

Erwägungen schrittweise ergänzt, um einen zulässigen Pfad herzustellen. Um bei der automatisierten Planung die in Abschnitt 5.5.4.1 erwähnten sinnlosen Prozesspläne zu vermeiden, sollte der Algorithmus nur auf den zuvor eingeplanten Aktivitäten operieren und fehlende Aktivitäten nicht direkt aus dem Operatorenpool oder einer Bausteinbibliothek ergänzen.[206] Vielmehr kann er auf noch einzuplanende Elemente hinweisen und dem Analytiker passende Bausteine vorschlagen.

Fehlende Elemente können zwei Ursachen haben. Erstens können eingeplante Aktivitäten *nicht versorgte Eingabeflüsse* aufweisen. Hier können Aufgaben eingefügt werden, welche passende Ausgabedaten erzeugen. Zweitens können in Folge *partieller Abhängigkeiten* Ausgabeflüsse auftreten, die nicht vollständig von abhängigen Aktivitäten verarbeitet werden können, weil sie eine Obermenge des von diesen erwarteten Inputs darstellen. Hier ist zu entscheiden, ob die nicht verarbeitete Teildatenmenge verworfen oder dennoch berücksichtigt werden soll. Im ersten Fall ist den betroffenen Aktivitäten eine Aufgabe zwischenzuschalten, welche den nicht handhabbaren Anteil ausfiltert. Im oben erwähnten Beispiel einer von IN_B erhobenen und von OUT_A nicht erfüllten `not-null`-Restriktion wäre eine Aufgabe einzufügen, die Datensätze mit `null`-Werten aus dem Datenset entfernt und somit für B zulässige Daten produziert. Im zweiten Fall ist eine Aufgabe zur Aufspaltung der Daten und zur Weiterleitung des von B nicht handhabbaren Anteils an eine komplementäre Aktivität C vorzusehen (vgl. [Hein+08, 452-454]). In beiden Fällen sind die Kriterien der notwendigen Filter unmittelbar aus den partiellen Abhängigkeiten ableitbar. Der Analytiker entscheidet jeweils, wie vorzugehen ist. Die zu fällenden Entscheidungen können zum Teil eigene Subplanungsprobleme darstellen.

Infolge der Abhängigkeitsprüfung neu eingeplante Aufgaben erfordern typischerweise eine Operatorzuordnung und damit die wiederholte Ausführung der Teilaufgabe P2.2.

---

[206] Die Endlichkeit der betrachteten Bausteinmenge und die einmalige Berücksichtigung jedes Bausteins im Abhängigktiesgraph garantieren zugleich die Terminierung des Planungsalgorithmus [Hein+08, 454].

Ableitung einer Aktivitätsfolge

Abschließend sind die eingeplanten Aktivitäten zu einem ausführbaren Prozess zu verknüpfen. Hierzu werden die definierten fachlichen Abhängigkeiten mit den ermittelten Datenabhängigkeiten vereint[207] und genutzt, um einen fachlich sinnvollen und technisch zulässigen Pfad vom Startzustand zum Zielzustand zu generieren. Der Algorithmus von HEINRICH ET AL. leitet aus dem Abhängigkeitsgraphen einen zulässigen Workflow ab [Hein+08, 450, 455]. Die hierbei genutzte Strategie der Vorwärtsverkettung ist in der Lage, aus partiellen Abhängigkeiten Prozessverzweigungen zu deduzieren, um Daten zur fallabhängigen Handhabung aufzuspalten [Hein+08, 454].[208] Darüber hinaus können unabhängige Aktivitäten in parallelen Prozesszweigen angeordnet werden [Hert89, 83f.], [Hein+08, 455]. Dies ist sinnvoll bei längeren Prozessabschnitten sowie bei Aktivitäten mit langen Ausführungszeiten, um die Prozesseffizienz zu verbessern. Die Vorwärtsverkettung erlaubt es zudem, Aktivitäten während der Prozesskonstruktion ausgehend von den Analysedaten bezüglich ihrer Flussbeziehungen zu konkretisieren (vgl. [Enge99, 184]). Hierbei werden für alle Beziehungen A→B jeweils die Eingabeflüsse IN_B mit Name, Wertebereich und Restriktionen von OUT_A belegt, soweit diese zur Planungszeit bekannt sind. Beispielsweise lässt sich IN_B = (`Input, _Numeric, _`) nach Maßgabe von OUT_A zu (`Alter, _Integer, 0..99`) konkretisieren. Diese Angaben können die Verfahrensparametrisierung (P4) wirksam unterstützen.

Ein vergleichbarer Ansatz speziell für die automatische Konstruktion von KDD-Prozessen, der ebenfalls semantische Ähnlichkeiten von Flussbeziehungen berücksichtigt, stammt von DIAMANTINI ET AL. [DiPS09]. Er erzeugt in verschiedenen Verfahrenskontexten wiederverwendbare Prozesse auf Aufgabenebene und zielt bewusst auf Ableitung einer großen Zahl valider Prozessvorschläge, die anhand nutzerdefinierter Kriterien bewertet werden können. Gegenüber der hier

---

[207] Ihre gemeinsame Behandlung durch den Algorithmus zur Erzeugung des Abhängigkeitsgraphen scheitert aufgrund ihres booleschen Datentyps. Da der Algorithmus ausschließlich auf Datendeklarationen abstellt, kann er Synchronisationsflüsse nicht unterscheiden und würde sie beliebig miteinander verknüpfen.

[208] Dieses Vorgehen entspricht dem nichtdeterministischen Planen [Hein+08, 449].

präsentierten Vorgehensweise wird eine weniger detaillierte Deklaration der Analysedaten und eine Zielspezifikation auf Basis vorgegebener Aufgabenklassen verwendet.

5.5.4.4 Zusammenfassung: Planung der Datenvorbereitungsphase

Die Planung der Datenvorbereitungsphase dient der Konstruktion eines abgeschlossenen Workflows, der Rohdaten aus der Datenquelle abruft und in adäquat aufbereiteter Form für die Datenanalysephase bereitstellt. Neben Datenabhängigkeiten werden auch fachliche Abhängigkeiten zur Berücksichtigung von Effektivitäts- und Effizienzaspekten einbezogen. Mit der Prüfung der Datenabhängigkeiten ist eine implizite Korrektheitsprüfung verbunden.

Die Planung sieht explizit die Wiederverwendung von Prozessartefakten vor. Bei der Reihenfolgeplanung sollten Prozessmodule als Einheit behandelt werden, da ihre interne Struktur bereits validiert ist. Eine Auftrennung ist nur dann ratsam, wenn eine Anpassung der Modulstruktur notwendig ist. Die Methodik ist dem typischen Vorgehen bei manueller Prozesskonstruktion angelehnt, das in wesentliche Teilaufgaben strukturiert wird. Sie ist demnach auch manuell anwendbar, wenngleich eine Werkzeugunterstützung der geschilderten Art aus Effizienz- und Komplexitätsgründen von Vorteil ist. Die Entscheidungshoheit obliegt dem Analytiker.

5.5.5 Planung der Ergebnisaufbereitungsphase (P3)

Ziel der Planung der Ergebnisaufbereitung ist die Erstellung eines Prozesses zur Überführung der in der Analysephase erzeugten Aussagen in eine Darstellungsform, die den Anforderungen des Informationsbedarfsprofils genügt und dem Anwendungszweck der Analyseergebnisse dienlich ist. Ihr Ergebnis ist ein Workflow-Ausschnitt mit allen hierfür nötigen Aktivitäten.

Methodisch folgt dieser Schritt dem Vorgehen zur Planung der Datenvorbereitungsphase P2 (Abschnitt 5.5.4). Daher stellen die folgenden Ausführungen hauptsächlich eine Rekapitulation der dort beschriebenen Planungsschritte dar, die – soweit erforderlich – um Hinweise auf

spezielle Aspekte der Ergebnisaufbereitung ergänzt wird. Dies betrifft insbesondere die *Spezifikation der Aufbereitungsaufgaben (P3.1)* in Abschnitt 5.5.5.1. Die Teilaufgaben *Zuordnung von Transformationsverfahren (P3.2)* und *Reihenfolgeplanung (P3.3)* fasst Abschnitt 5.5.5.2 kurz zusammen.

5.5.5.1 Spezifikation der Aufbereitungsaufgaben (P3.1)

Die für die Datenvorbereitungsphase beschriebenen Optionen zur Identifikation notwendiger Aufgaben gelten analog für die Ergebnisaufbereitung. Im Folgenden werden für jede Option exemplarisch wichtige Aspekte aufgezeigt, die für die dritte Prozessphase zu bedenken sind.

Einsatz wiederverwendbarer Prozessmodule

Die Nutzung von Prozessmodulen besitzt in der Ergebnisaufbereitung besonders großes Potenzial. Können in der Datenvorbereitung Analysedaten mit beliebigen Datencharakteristika auftreten, ist hier die Zahl der zu verarbeitenden Aussagetypen vergleichsweise überschaubar. Damit steigt die Wahrscheinlichkeit, in der Bausteinbibliothek passende Artefakte zu finden. Als Suchkriterien sind weniger Ausgabeflüsse als vielmehr Deskriptoren des Anwendungs- und Analysekontext relevant, insbesondere Anforderungen des Informationsbedarfsprofils.

Würdigung der Implikationen bereits getroffener Entwurfsentscheidungen

Entscheidungen früherer Planungsschritte nehmen hauptsächlich in zweierlei Hinsicht Einfluss auf die Ergebnisaufbereitung. Erstens kann es gewünscht sein, zur Datenvorbereitung erfolgte analysespezifische Transformationen nach der Analyse wieder zu kompensieren. Dies ist etwa dann der Fall, wenn intuitiv interpretierbare Daten für den Analysealgorithmus in schwer verständliche Formate übersetzt worden sind (z.B. kategoriale Zeichenketten in numerische Codes für Neuronale Netze) und für den Anwender wieder „lesbar“ gemacht werden sollen. Zweitens produzieren Analyseverfahren häufig Ausgaben, die über den geforderten Aussagetyp hinausgehen. So liefern Verfahren zur Modell-

erstellung in der Regel Fehlermaße, die durch geeignete Aufgaben nutzbringend weiterverarbeitet oder visualisiert werden können.

Auswertung von Kontextregeln und situative Erwägungen des Analytikers

Die in Abschnitt 3.1.2.4 genannten Aufgaben Beurteilung, Vereinfachung, Transformation, Interpretation und Dokumentation der Ergebnisaufbereitung können als Checkliste dienen, um relevante Ergebnisaufbereitungsmaßnahmen zu identifizieren. Beispielhafte Hinweise zu ihrer Planung sind im Folgenden genannt. Sie können in Planungsregeln formuliert oder individuell beachtet werden.

Beurteilung

In vielen Fällen liefern die Analyseverfahren geeignete Bewertungsmaße als Bestandteile der Analyseergebnisse. Ist dies nicht der Fall, sind Aufgaben zur Berechnung solcher Maßzahlen in den Prozess aufzunehmen. Ihre Auswahl kann durch die am **`Operator`** hinterlegbaren, zum Verfahren passenden `Bewertungskriterien` sowie den `Evaluationsansatz` unterstützt werden. Der Evaluationsansatz kann eine Aufteilung der Analysedaten erfordern, die bereits bei der Planung der Datenvorbereitung adäquat zu berücksichtigen ist (vgl. Abschnitt 5.5.3.5).

Vereinfachung, Transformation

Kontextabhängiger Transformationsbedarf entsteht z.B. in Bezug auf den Empfänger oder die geplante Nutzungsform der Analyseergebnisse (vgl. Informationsbedarfsprofil). Sollen diese etwa auf spezielle Medien oder Geräte übertragen werden, kann eine gerätespezifische Anpassung von Formaten oder Darstellungsformen angezeigt sein [KoRS02, 47]. Für diese Nutzungsformen sind gegebenenfalls Transformationsprozesse zu konzipieren, die parallel zum standardmäßigen Bereitstellungsprozess ablaufen sollen (Prozessgabelung).

Interpretation, Dokumentation

Analysewerkzeuge bieten in der Regel für jeden Ergebnistyp geeignete Präsentations- oder Berichtsvarianten an, die interaktiv aufgerufen werden können. Dennoch kann es in manchen Fällen sinnvoll sein, die

Darstellungsform der Ergebnisse situativ abzuändern, die Möglichkeit zur Navigation in den Ergebnissen zu schaffen oder für den Empfänger ansprechender zu gestalten. Hierzu eignet sich z.B. das Laden der Ergebnisse in OLAP-Systeme, um die Mächtigkeit dieser Werkzeuge zu nutzen und neue Empfängerkreise zu erschließen [BöKU03, 176f.]. Diese Option kann umfangreichere Transformationsprozesse erfordern. Wichtig ist stets die problemadäquate Aufbereitung der Analyseergebnisse für den Auftraggeber, etwa mithilfe gängiger Bürosoftware (Präsentationen, Tabellen, Textdokumente). Hierzu kann standardmäßig eine eigene Aufgabe im Prozess vorgesehen werden.

5.5.5.2 Ergänzende Zusammenfassung: Planung der Ergebnisaufbereitungsphase

Die inhaltlich als relevant erachteten Aufbereitungsaufgaben sind schließlich analog zum Vorgehen bei der Datenvorbereitung mit geeigneten Verfahren zu versorgen (P3.2) und in eine effektive und effiziente Ablaufreihenfolge zu bringen (P3.3). Hierzu sei auf die entsprechenden Abschnitte 5.5.4.2 bzw. 5.5.4.3 verwiesen. Die Ergebnisaufbereitung ist typischerweise weitaus weniger komplex und daher einfacher zu konzipieren als die Datenvorbereitung. Darüber hinaus treten ungeeignete Gestaltungsentscheidungen sofort in Erscheinung und lassen sich umgehend korrigieren, während ungeeignete Maßnahmen der Datenvorbereitung häufig unentdeckt bleiben.

5.5.6 Instanziierung von Verfahrensparametern (P4)

Nach Festlegung und Verknüpfung der auszuführenden Aufgaben und der Zuordnung geeigneter Operatoren bleibt die Instanziierung der Verfahrensparameter zu leisten. Ziel des vierten Schritts der Prozessspezifikation ist die Auswahl von zur Erreichung des Analyseziels geeigneten Parameterwerten für alle Prozessaufgaben. Als Ergebnis liegen für alle drei Phasen des Analyseprozesses ausführbare Prozessaktivitäten vor.

Die Parameterinstanziierung umfasst alle verhaltenswirksamen Einstellungen eines **`Operators`**. Hierzu zählt neben der Justierung der

Modus-`Parameter` auch die Belegung der `Eingabedaten` mit konkreten Informationsobjekten aus den Eingabeflüssen der Aktivität. HOGL unterscheidet hierzu zwischen Mikro- und Makroparametrisierung. *Mikroparametrisierung* betrifft die Modusparameter und nimmt Feineinstellungen vor, die das Verhalten eines Algorithmus nicht grundlegend verändern. Beispielsweise beeinflusst der Modusparameter „minimale Konfidenz" bei der Regelinduktion unmittelbar die Anzahl erzeugter Regeln sowie mittelbar ihre Validität und Generalisierbarkeit, nicht jedoch Inhalt und Struktur. Letztere sind von der *Makroparametrisierung* betroffen, die mit der Auswahl konkreter Eingabedaten verschiedene Konfigurationen des Verfahrens erzeugt. So unterscheiden sich die Ergebnisse der Regelinduktion grundlegend danach, welche Informationsobjekte den Operator als unabhängige bzw. abhängige Variablen versorgen [Hogl03, 98f.]. Je nach Belegung dieser Rollen ändern sich Aussageinhalt (welche Elemente sind Antezedenz, welche Konsequenz?) bzw. Struktur der Regeln (unterschiedliche oder gleiche Elemente als Antezedenz und Konsequenz?) [Hogl03, 99].

Bei manueller Planung erfolgt die Parameterinstanziierung typischerweise im Zuge der Verfahrenszuordnung. Da hierbei meist noch nicht alle Kontextfaktoren bekannt sind, werden häufig spätere Anpassungen nötig. Bei automatisierter Planung kann die Makroparametrisierung gegebenenfalls während der Analyse von Datenabhängigkeiten, die Mikroparametrisierung mit der Ableitung einer Aktivitätsfolge geschehen. Die Platzierung dieses Schrittes am Ende des Planungsprozesses soll gewährleisten, dass alle relevanten Kontextfaktoren bekannt sind und Iterationen möglichst vermieden werden. Seine Teilaufgaben *Makroparametrisierung (P4.1)* und *Mikroparametrisierung (P4.2)* werden im Folgenden separat erläutert.

5.5.6.1 Belegung der Eingabedaten (Makroparametrisierung, P4.1)

Jeder Operator benennt eine Menge von `Eingabedaten`, die zur Durchführung des Verfahrens erforderlich sind und in der Regel eine bestimmte semantische Rolle übernehmen (z.B. unabhängige/Eingabevariablen und abhängige/Zielvariablen, etc.). Die jeweilige Rolle ist meist dem Namen der Eingabedaten-Deklaration zu entnehmen. Diesen

Eingabedaten sind passende Informationsobjekte aus eingehenden Datenabhängigkeitsflüssen zuzuordnen. In Bezug auf Flüsse im Ganzen ergibt sich die Zuordnung aus der Überprüfung von Datenabhängigkeiten (P2.3). Beispielsweise wird ein Eingabefluss, der eine Tabelle transportiert, an die passende Eingabedaten-Schnittstelle des Operators angedockt. In der Regel ist zusätzlich eine Zuordnung auf Ebene der Komponenten dieser Flüsse erforderlich, z.B. sind einzelne Spalten der Tabelle einzelnen Eingaberollen zuzuweisen. Das Ergebnis wird als `Eingabedatenzuordnung` an der Aktivität dokumentiert (vgl. Abschnitt 4.5.2.3). Abbildung 90 zeigt diesen Vorgang symbolisch am Beispiel der Analyseaufgabe `Prognosemodell berechnen` (vgl. Abbildung 85, Abschnitt 5.5.3.1/Seite 326). Der ausgewählte Operator `Decision Tree Learner` erwartet ein nominales Zielattribut (`class column`), trifft aber keine Einschränkungen bezüglich der Eingabevariablen.[209] Im konkreten Fall wird das Attribut `Antwort_LK` der Rolle `class_column`, alle verbleibenden Attribute der vorliegenden Analysedaten den `data_columns` zugewiesen.

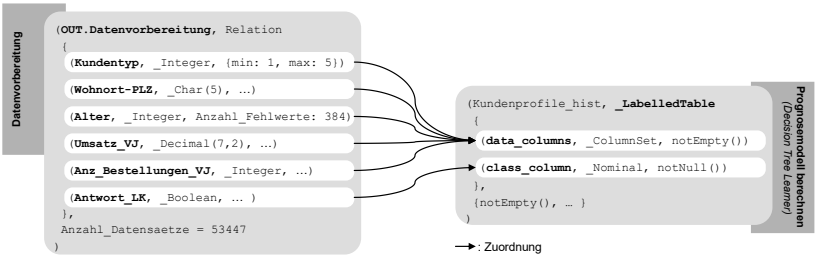


Abbildung 90: Makroparametrisierung am Beispiel der Modellerstellung (eigene Darstellung)

Produziert ein Operator mehrere Ausgabedaten, ist das beschriebene Vorgehen analog auf diese anzuwenden. In jedem Falle wird die Zuordnung von Ausgabedaten und Ausgabeflüssen in der `Ausgabedatenzuordnung` dokumentiert.

[209] Vgl. zur Eingabedaten-Deklaration den Operator `Decision Tree Learner` des Werkzeugs KNIME.

Datentransformationsbedarf entsteht in der Regel im Hinblick auf bestimmte Datenelemente (z.B. Spalten einer Tabelle), die zum Zeitpunkt der Festlegung geeigneter Transformationsaufgaben zwangsläufig bekannt sind. Für Aufgaben der Datenvorbereitung und Ergebnisaufbereitung kann die Makroparametrisierung daher während der Aufgabenspezifikation geschehen.

5.5.6.2 Einstellung von Modusparametern (Mikroparametrisierung, P4.2)

Im diesem Schritt erfolgt die Instanziierung der Modusparameter aller gewählten Verfahren durch konkrete Werte. Da jede Aktivität der Datenvorbereitung die Analysedaten strukturell oder inhaltlich modifiziert und ihnen damit jeweils neue Charakteristika verschafft,[210] erfolgt die Verfahrensinstanziierung schrittweise „vorwärts", ausgehend von den Datenzugriffsaufgaben (Datenselektion) bis zur Analyseaktivität bzw. von letzterer weiter zu allen Aktivitäten der Ergebnisaufbereitung. Auf diese Weise ist es möglich, die Parameterwerte für jede Aktivität auf die tatsächlich zu verarbeitenden Daten abzustimmen (Datenkontext). Hierzu können die jeweils gültigen Datencharakteristika (`Informationsobjekt.Wert`) im Voraus berechnet werden, ohne den Prozess tatsächlich auszuführen.[211]

Ein Operator dokumentiert für jeden verfügbaren `Parameter` eine Beschreibung und einen Hilfetext, die bei der Einstellung konkreter Werte helfen können. Aufgrund der häufig zahlreichen Parameter realisieren Operatoren zuweilen ein sehr umfangreiches Verhaltensrepertoire [Enge99, 127f.]. Die Suche nach passenden Einstellungen für die aktuelle Situation erfolgt daher meist durch Versuch und Irrtum,

---

[210] Die Modifikationen betreffen einerseits das Datenschema, andererseits die Gestalt des Datenkörpers (Werteverteilung, -anzahl und -häufigkeiten). Somit steht erst nach vollständiger Spezifikation des Datenvorbereitungsprozesses fest, auf welchen konkreten Daten der Analysealgorithmus letztlich operiert.

[211] Alternativ können die Datencharakteristika im Zuge der Reihenfolgeplanung (P2.3) durch das Aktivitätsnetz propagiert und für jede Aktivität gemäß geltender `Zusicherungen` aktualisiert werden. Da sie damit an allen Aktivitäten in der jeweils gültigen Fassung bekannt sind, kann die Mikroparametrisierung mit jeder beliebigen Aktivität starten. Dennoch scheint eine systematische Reihenfolge ratsam.

was oft zu suboptimalen Ergebnissen führt und Iterationen im Prozessablauf verursacht [SaBS00, 1]. Zur Handhabung der resultierenden Planungskomplexität wurden daher schon früh Unterstützungsmöglichkeiten, etwa durch Assistenzfunktionen der Analysewerkzeuge, gefordert (vgl. [Enge96, 173], [BrAn96, 53]). Als erster Vorschlag hierzu gilt der MLT CONSULTANT, der Parametereinstellungen empfehlen und durch Auswertung der erreichten Ergebnisse eine Erfolgskontrolle durchführen kann [CSG+92, 10]. CBR-gestützte Empfehlungssysteme werden im MININGMART-Projekt [KiZV00], [MoSE03], von HILARIO & KALOUSIS [HiKa01] sowie von LINDNER [Lind05] vorgeschlagen. SAITTA ET AL. [SaBS00] automatisieren den Versuchs-Irrtums-Prozess mithilfe verfahrensspezifischer Optimierungsmodule.

In dieser Arbeit wird die Mikroparametrisierung mithilfe kontextsensitiver *Konfigurationsregeln* unterstützt (vgl. Abschnitt 4.5.4.1). Diese bieten zugleich eine Basis, auf der weitergehende Ansätze aufbauen können. ENGELS regt an, für jedes Verfahren Initialisierungs- und Adaptionsregeln zu formulieren. Initialisierungsregeln belegen die Parameter anfänglich mit geeigneten Ausprägungen, Adaptionsregeln dienen ihrer Anpassung für den Fall nicht zufriedenstellender Ergebnisse [Enge99, 135]. Relevante Kontextfaktoren ergeben sich zunächst aus dem Informationsbedarfsprofil und den Datencharakteristika. Gütemaße wie z.B. Sicherheit, Genauigkeit und Detailliertheit, auf die häufig im Informationsbedarfsprofil Bezug genommen wird, werden in der Regel stark von der Wahl der Parameterwerte beeinflusst. Einfache Datencharakteristika, wie die Anzahl der verarbeiteten Datensätze, können mitunter direkt zur Berechnung passender Einstellungen genutzt werden, etwa um minimale relative Häufigkeiten (Support) oder Konfidenzniveaus festzulegen. Daneben können Angaben aus dem Anwendungskontext hilfreich sein, denn wie viele Fälle z.B. zur Stützung eines Musters (Support) notwendig sind, ist häufig von Problemdomäne, Branche oder Unternehmen abhängig. Komplexere Datencharakteristika sind in der Lage, auch weniger offenkundige Einflüsse der Daten auf Parametereinstellungen abzuleiten. Beispielsweise geben mittels Diskriminanzanalyse berechnete Eigenwerte und Eigenvektoren Hinweise darauf, wie viele Dimensionen zur Trennung

der Klassen bei Lösung einer Klassifikationsaufgabe erforderlich sind [EnTh98, 432].

Bei vielen Verfahren, insbesondere für die Analysephase, sind die Einflussbeziehungen zwischen einem Parameterwert und den realisierten Gütemaßen allerdings nicht offensichtlich. Selbst qualitative Zusammenhänge sind oft nicht leicht zu ermitteln, da die Interaktionen mehrerer Parameter die Wirkungen von Einzelparametern überlagern können [SaBS00, 1]. ALI & WALLACE fordern daher, dass entsprechende Empfehlungen vom Verfahrensentwickler bereitgestellt werden, und präsentieren hierfür eine rigorose, experimentgestützte Prozedur [AlWa97, 4]. In jedem Falle gestaltet sich die Verknüpfung empfehlenswerter Parameterwerte mit dem Anwendungskontext schwierig (vgl. [AlWa97, 4]). Einen Lösungsansatz bietet z.B. das Process Mining auf Instanzmodellen. Hinweise hierzu gibt Abschnitt 7.4.2. Angesichts des starken Einflusses der Daten auf die Ergebnisse [AlWa97, 12] ist zu empfehlen, Kontextregeln wenigstens in Abhängigkeit von Datencharakteristika aufzustellen.

Allgemeingültige Konfigurationsregeln werden grundsätzlich am Operator annotiert. Die Regeln leiten konkrete Werte bzw. Wertekombinationen für Modusparameter aus verfügbaren Kontextfaktoren sowie aus den `Anforderungen` der zu konfigurierenden Aktivität ab. Eine beispielhafte Konfigurationsregel für die Berechnung eines Prognosemodells ist im Folgenden angegeben.

```
IF   Anwendung = (Kunde.Antwortwahrscheinlichkeit, +,
     Soll)
AND  Genauigkeit = hoch
AND  Interpretierbarkeit >= mittel
THEN ( set(quality_measure, „gain ratio“),
       set(pruning_method, „MDL“),
       set(binary_nominal_splits, true)
) ENDIF
```

Anwendungsspezifische Konfigurationsregeln können alternativ in Prozessbausteinen, die jeweils mit den betreffenden Kontextfaktoren annotiert sind, in der Bausteinbibliothek hinterlegt werden. Sie können im Sinne einer anwendungsbezogenen Regelbasis separat gewartet werden, ohne allgemeingültige Regeln betrachten zu müssen. Zugleich

werden sie nur dann geprüft, wenn der jeweilige Kontext tatsächlich vorliegt. Unabhängig davon kann die Wiederverwendung bewährter Aktivitäten und Fragmente wertvolle Hinweise auf empfehlenswerte Verfahrensinstanziierungen geben und die Planungskomplexität reduzieren (vgl. [EnTh98b, 51]).

5.5.6.3 Zusammenfassung: Instanziierung von Verfahrensparametern

Die Instanziierung von Verfahrensparametern umfasst alle in den Prozess integrierten Aktivitäten und erfolgt ganzheitlich als letzter Schritt der Spezifikation von Analyseprozessen. Als Ergebnis steht ein ausführbarer Analyse-Workflow zur Lösung der Analyseaufgabe bereit.

5.5.7 Zusammenfassung: Methodik zur Prozessplanung

Dieses Kapitel präsentiert eine Methodik zur Planung von Datenanalyseprozessen, die vom Sachproblem über Analyseprobleme bis zu Analyseprozessen eine durchgängige, zweck- und zielgetriebene Konzipierung komplexer Untersuchungsvorhaben unterstützt. Sie berücksichtigt mit der Hierarchisierung, Dekomposition und Modularisierung anerkannte Strategien zur Komplexitätsbewältigung. Neben der innovativen Neuplanung wird explizit auch die Wiederverwendung früherer Pläne und Prozesse einbezogen. Die Methodik stützt sich auf etablierte Konzepte und Theorien, wie z.B. die modellgestützte Untersuchungssituation, die Operationalisierung von Forschungsfragen sowie die Handlungsplanung, und führt verschiedene Vorschläge aus der Literatur zu einem integrierten Ansatz zusammen.

In Abbildung 91 ist der Planungsablauf innerhalb der Ebenen der Analysearchitektur dargestellt. Hierbei müssen nicht alle Ebenen behandelt werden; ein Einstieg ist gleichermaßen auf der Anwendungs-, der Analyse- oder der Prozessebene möglich. Die Verknüpfung zwischen Ziel- und Prozessebene geschieht beim Untersuchungsdesign, das über die Art der auszuführenden Untersuchung und den Bedarf einer detaillierten Prozessplanung entscheidet.

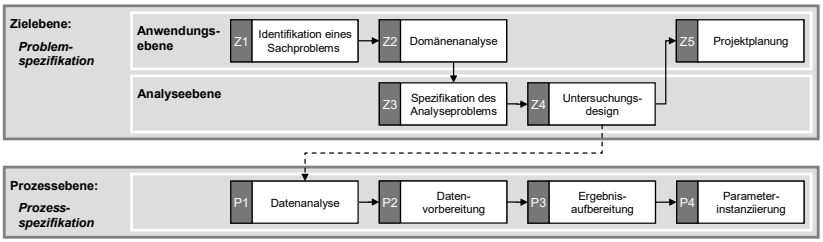


Abbildung 91: Integriertes Handlungsschema zur Planung von Datenanalyseprozessen (eigene Darstellung)

6 Steuerung von Datenanalyseprozessen

Das vorliegende Kapitel beschreibt die methodisch gestützte Steuerung von Datenanalyseprozessen. Hierzu betrachtet Abschnitt 6.1 die Aufgabe der Steuerung und untersucht, inwiefern sie Gestaltungsanteile beinhaltet. Abschnitt 6.2 stellt ihre Teilaufgaben und ein zugehöriges Handlungsschema vor. Für Datenanalyseprozesse geeignete Steuerungsansätze werden in Abschnitt 6.3 betrachtet. Abschnitt 6.4 fasst das Kapitel kurz zusammen.

6.1 Prozesssteuerung als Lenkungs- und Gestaltungsaufgabe

6.1.1 Der Steuerungsbegriff

Die Steuerung wird in Abschnitt 2.4.3.2 der Lenkung von Prozessen zugeordnet, wo sie als Bindeglied zwischen den Tätigkeiten Planung und Kontrolle der Auslösung der Prozessdurchführung dient. Im Allgemeinen wird unter Steuerung die zielgerichtete Beeinflussung des Verhaltens einer Systemkomponente verstanden [FeSi13, 27]. Dieser Steuerungsbegriff kann vor dem Hintergrund der Rolle von Prozessen als Lösungsverfahren zur Bewältigung einer Aufgabe (vgl. Abschnitte 2.3.1.4 sowie 4.5) konkretisiert werden. Besteht die Realisierung eines Lösungsverfahrens in der Durchführung von Operatoren, so dient die Prozesssteuerung der Bestimmung von Art und Reihenfolge sowie der Auslösung geeigneter Aktivitäten derart, dass ein den Sach- und Formalzielen der Aufgabe entsprechendes Prozessverhalten erreicht wird (vgl. [Fers92, 7], [FeSi13, 102]). Aus kybernetischer Sicht ist Steuerung eine antizipative Störungskompensation, die ohne Rückkopplung agiert [Hein91, 60]. Für die Prozesssteuerung ist neben dieser einfachen Kopplung (Steuerkette), die jeweils eine starre Aktivitätsfolge realisiert, auch eine Rückkopplung (Regelkreis) möglich, bei der die Entscheidungen der Steuerung von den Ergebnissen der bisherigen Aktivitätsausführung abhängig gemacht und mit dem Zustand der Regelstrecke abgestimmt werden [Fers92, 7], [FeSi13, 103].

6.1.2 Gestaltungsanteil der Prozesssteuerung

Die Festlegung von Art und Reihenfolge der auszulösenden Aktivitäten beinhaltet Entscheidungen, welche den konkret durchgeführten Ablauf sowie das damit realisierte Verhalten mitunter stark beeinflussen können. Sie betrifft den Gestaltungsanteil der Prozesssteuerung, wie ihn Abschnitt 5.1.4 identifiziert. Dort werden sechs Anwendungsfälle der Prozessgestaltung unterschieden, von denen drei während der Ausführungszeit von Prozessen auftreten und somit der Steuerung zufallen. Sie werden im Folgenden auch als ***Steuerungsmodi*** bezeichnet:

- Der Anwendungsfall ***Repetition*** beschreibt die unmodifizierte Ausführung eines vollständig vorausgeplanten Workflows, dessen Ablauf in Struktur und Verhalten genau der vorgegebenen Spezifikation entspricht. Der Einfluss der Prozesssteuerung beschränkt sich auf die Auflösung von Ablaufvarianten, d.h., auf die Wahl eines von mehreren alternativen Ablaufpfaden. Diese geschieht bei programmierten Abläufen deterministisch durch Auswertung von Kontextfaktoren. In der Regel besteht hierbei keinerlei Entscheidungsspielraum, so dass kein Gestaltungsanteil verbleibt. Dieser Fall betrifft typischerweise Analyseprozesse, die in operative Anwendungssysteme integriert sind (z.B. Empfehlungs- oder Scoring-Systeme).
- Der Anwendungsfall ***Deviation*** beschreibt die Modifikation eines vorgegebenen Prozessschemas zur Behandlung fallspezifischer Anforderungen oder Ausnahmen, die bei der Prozessplanung nicht berücksichtigt wurden. Hierbei werden ausschließlich Änderungen betrachtet, die gänzlich unvorhergesehen sind, also eine Abweichung vom Plan darstellen.[212] Gegenüber dem ursprünglichen Plan entstehen ungeplant variable Abläufe. Die Entscheidungen über die erforderlichen Anpassungen stellen Gestaltungsaufgaben dar. Dieser Fall ist bei Wiederverwendung von Prozessvorlagen als typisch anzunehmen.

[212] Das Ausfüllen von Prozessschablonen betrifft hingegen geplante Anpassungen, die als (gegebenenfalls partielle, auf den jeweiligen Prozessausschnitt bezogene) Innovation einzuordnen sind.

- Der Anwendungsfall ***Innovation*** beschreibt Ad-hoc-Abläufe, für die kein geeignetes vollständiges Prozessschema existiert. Sie sind durch situative Ablaufgestaltung während der Ausführungszeit zu entwickeln. Die Innovation kann sich auf den gesamten Prozess oder auf Prozessausschnitte beziehen, wie etwa bei Vorgabe von Prozessplänen mit abstrakten Anteilen (Schablonen), die zu konkretisieren sind. Innovation bedingt umfangreiche Gestaltungsentscheidungen und ist in der Praxis der Datenanalyse als Regelfall anzusehen.

Die Prozesssteuerung enthält folglich, abhängig vom jeweiligen Anwendungsfall, mehr oder weniger Gestaltungsanteil. In realen Analysefällen stellt die Repetition eher die Ausnahme dar, womit davon auszugehen ist, dass stets ein nicht unerheblicher Anteil der Gestaltungsleistung zur Ausführungszeit während der Steuerung zu erbringen ist.

6.1.3 Gegenstand und Ziele der Prozesssteuerung

Die Planung von Prozessen in Form von Aufgaben bzw. Aktivitäten erfolgt auf Typebene (Prozess- bzw. Ablauftypen). Die Durchführung eines konkreten Ablaufs erfolgt hingegen auf *Instanzebene*, indem geeignete Aufgabenträgerinstanzen (Server und Personen, vgl. Abschnitt 4.6.3) konkrete Vorgänge verrichten [Reif03, 23]. Gegenstand der Prozesssteuerung ist somit eine *Prozessinstanz*, deren jeweiliger Zustand in Form eines *Instanzmodells* (vgl. Abschnitt 4.5.3) erfasst und fortgeschrieben wird [Reif03, 83]. Die Prozessinstanz wird abhängig vom jeweiligen Anwendungsfall aus einem vorgegebenen Prozesstypschema erzeugt oder dynamisch entwickelt.

Ausgangspunkt der Prozesssteuerung bildet im Idealfall ein vollständiger *Prozessplan* aus Problemkarte, Analysekette und Prozesstypschema. Liegt kein Prozessplan in Gestalt eines Workflows vor, der direkt als Folge von Arbeitsanweisungen an die Aufgabenträger übermittelt werden kann (vgl. [FeMa95a, 7], [Knob03a, 342]), so geben Prozessschablonen, Analyseketten und Problemkarten wichtige Rahmenbedingungen vor. Je weniger konkret die Vorgaben auf Prozessebene, desto mehr Bedeutung gewinnen Analyseziele, angestrebte Wirkungen

sowie Zeit- und Budgetrestriktionen der Analyse- und Anwendungsebene.

Sachziel der Prozesssteuerung ist die Auslösung von Aktivitäten derart, dass die gewünschte Prozessleistung mithilfe gegebener Ressourcen realisiert wird (Prozesseffektivität, vgl. [Reif03, 39], [ElLa04, 106]). Bei Verzicht auf Detailpläne ist es in der Regel erforderlich, grobe Vorgaben der Anwendungs- und Analyseebene zu präzisieren (vgl. [FeMa95a, 3]). Formalziel der Prozesssteuerung ist die Effizienz der Prozessausführung [GaSV94, 4] (vgl. Abschnitt 2.4.2).

6.2 Aufgaben und Vorgehen bei der Prozesssteuerung

Aus den bisherigen Ausführungen ergeben sich sogleich wichtige Teilaufgaben, die im Rahmen der Prozesssteuerung zu erfüllen sind. Als erste Aufgabe ist durch *Ablaufinstanziierung (S1)* ein neues Instanzmodell zu erzeugen. In den Anwendungsfällen Deviation und Innovation ist im zweiten Schritt eine mehr oder weniger umfangreiche *Ablaufgestaltung (S2)* erforderlich. Diese wird häufig während des gesamten Ablaufs fortgeführt und kann auch zu einem späteren Zeitpunkt initiiert werden. Sie steht daher mit der dritten Aufgabe, der *Ablaufbegleitung (S3)*, in intensiver Wechselwirkung. Diese Teilaufgabe fasst mit der Vorgangsauslösung, Koordination und Überwachung alle Tätigkeiten zusammen, die den Ablauf über seinen gesamten Lebenszyklus hinweg begleiten und kann als Steuerung i.e.S. betrachtet werden. Abschließend erfolgen *Protokollierung und Dokumentation (S4)*. Auch die Protokollierung erstreckt sich über die gesamte Lebenszeit des Ablaufs.

Daher ist die Abfolge der Teilaufgaben, wie sie das Handlungsschema in Abbildung 92 darstellt, als idealisierende Systematik zu verstehen.[213] Wichtige Ablaufbeziehungen sind als Pfeile dargestellt. Letztlich sind alle Teilaufgaben wichtige Bestandteile einer effektiven Prozesssteue-

[213] Tatsächlich ist nur die Instanziierung zwingend als erster Schritt auszuführen. Die Ablaufgestaltung geschieht im Falle der Innovation wenigstens teilweise vor dem Aufruf des ersten Operators. Die Dokumentation der Ergebnisse ist erst nach Beendigung des Ablaufs möglich und wird als letzter Schritt gezeigt.

rung. Sie erfolgen sämtlich auf Prozessebene, nehmen jedoch zum Teil Bezug auf andere Ebenen der Analysearchitektur. Die Aufgaben werden in den folgenden Abschnitten erläutert.

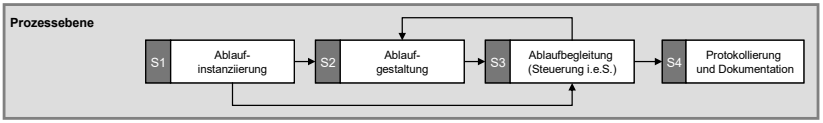


Abbildung 92: Handlungsschema zur Prozesssteuerung (eigene Darstellung)

6.2.1 Ablaufinstanziierung (S1)

Ziel der ersten Aufgabe ist die Erzeugung einer Prozessinstanz. Ihr Ergebnis ist ein *Instanzmodell*. Es entsteht durch Instanziierung eines Prozessschemas mit für den aktuellen Analysefall spezifischen Daten [Gier00, 290] (Abbildung 93).

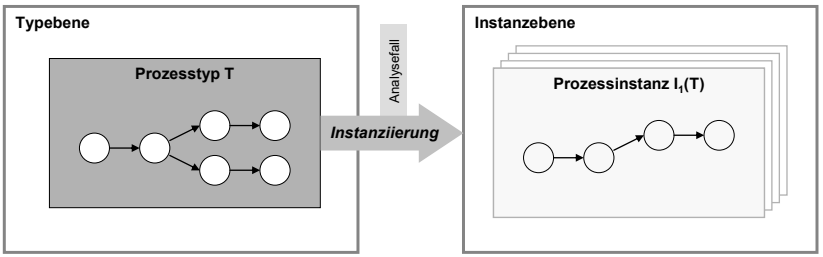


Abbildung 93: Instanziierung von Prozesstypen (in Anlehnung an [Reif03, 24])

Hierzu ist zunächst ein geeignetes Schema auszuwählen, das als Vorlage für den aktuellen Ablauf dienen soll. Die Auswahl erfolgt gemäß der `Funktion` der Analyseaufgabe sowie relevanter Kontextfaktoren, wie z.B. Anwendung und Analyseziel bzw. Analyseproblem. Beispielsweise kann nach einer Vorlage zur Berechnung eines Prognosemodells (Funktion „Einzelwerte deduzieren (relational)“) gesucht werden, welche im Kontext der Anwendung „`(Kunde. Antwortwahrscheinlichkeit, +, Soll)`“ geeignet ist (vgl. hierzu Abschnitt 5.5.3). Im Idealfall lässt sich ein vollständiger Workflow finden. Andernfalls kann eine möglichst passende Schablone eingesetzt

werden. Bei vollständiger Innovation besteht das initiale Schema aus einer einzigen abstrakten Analyseaufgabe.

Das Instanzmodell entspricht der *Instanzensicht* der Datenanalysearchitektur und repräsentiert den jeweils gültigen Zustand der Prozessinstanz samt ihrer Vorgänge und Datenflüsse (vgl. Abschnitt 4.5.3). Hierzu wird das Modell im Sinne eines „Logbuchs" im Rahmen der Dokumentation und Protokollierung (S4) kontinuierlich aktualisiert, indem über den gesamten Lebenszyklus des Ablaufs hinweg alle Zustandsänderungen erfasst werden (vgl. [ReBD00, 2]). Alle bereits definierten Aktivitäten des Prozesses werden als Vorgänge mit Initialzustand *angelegt* in das Instanzmodell übernommen (vgl. Abschnitt 4.5.3.1). Das Instanzmodell bildet zusammen mit dem Prozessschema die Basis für die Steuerung des Ablaufs [Reif03, 83].

6.2.2 Ablaufgestaltung (S2)

Liegt kein vollständiger oder passender Prozessplan für den aktuellen Analysefall vor, sind an dieser Stelle in dem in Abschnitt 6.1.2 geschilderten Ausmaß Gestaltungsentscheidungen zu treffen. Ziel dieser Ablaufgestaltung ist die Festlegung der genauen Ausprägung sowie einer geeigneten Reihenfolge jener Aktivitäten, die zur Realisierung des gewünschten Prozessverhaltens im aktuellen Kontext auszuführen sind (vgl. [Fers92, 7, 13]). Da die Gestaltungsaufgaben der Prozesssteuerung häufig simultan mit der Prozessdurchführung erfolgen, ist ihr Ergebnis nicht zwingend ein neues bzw. modifiziertes Prozesstypschema, sondern zunächst lediglich die Auslösung der betroffenen Aktivitäten, die sich im Prozessinstanzmodell niederschlägt. Die präskriptive Prozessgestaltung im Sinne einer Planerstellung ist Gegenstand der Prozessspezifikation (Abschnitt 5.5).

Die drei Anwendungsfälle der Ablaufgestaltung sind in Abschnitt 6.1.2 charakterisiert. Aus Gestaltungsperspektive sind an dieser Stelle keine näheren Ausführungen erforderlich. Die Repetition bedarf keiner Gestaltungsentscheidungen. Die Gestaltungsaufgaben im Rahmen der Innovation erfolgen analog zur Planung. Für die Anpassung vorgegebener Pläne im Rahmen der Deviation gibt Abschnitt 6.3 konkrete Hinweise.

6.2.3 Ablaufbegleitung (Prozesssteuerung i.e.S.) (S3)

Als Prozesssteuerung i.e.S. werden alle Teilaufgaben verstanden, welche einen Ablauf vom Start des ersten bis zur Beendigung des letzten ihm zugehörigen Vorgangs begleiten. Dies sind erstens die *Vorgangsauslösung (S3.1)*, die für jeden Vorgang durchzuführen ist, zweitens die *Koordination (S3.2)* aller beteiligten Ressourcen, und drittens die *Ablaufüberwachung (S3.3)*, die neben der Überwachung der Aktivitätsdurchführung insbesondere auch die Zielkontrolle leistet. Diese Teilaufgaben sind in der Regel simultan auszuführen und werden zum Teil durch Ablaufsteuerungskomponenten von Analysewerkzeugen bzw. Workflow-Management-Systemen unterstützt (vgl. [Gier00, 58], [ReBD00, 1f.], [Reif03, 2]).

6.2.3.1 Vorgangsauslösung (S3.1)

Die zentrale Aufgabe der Prozesssteuerung besteht in der Auslösung aller eine Prozessinstanz konstituierenden Vorgänge in geeigneter Reihenfolge und erfolgt jeweils durch Benachrichtigung eines geeigneten Aufgabenträgers über einen auszuführenden Bearbeitungsauftrag [ReBD00, 2]. Alle zur Vorgangsauslösung notwendigen Angaben sind in der Aktivitätsspezifikation enthalten [RaSc99, 10]. Voraussetzung für die Auslösung sind einerseits die vollständige Instanziierung durch Modusparameter, Eingabedaten und Zuordnung einer Aufgabenträgerinstanz, andererseits die Verfügbarkeit aller benötigten Ressourcen [Jabl00, 350f.], [Reic00, 17], [GCC+04, 323]. Der Vorgang befindet sich dann im Zustand *bereit* und kann gestartet werden (vgl. Abschnitt 4.5.3.1).

Die Suche nach geeigneten maschinellen Aufgabenträgern wird durch den `Operator` unterstützt, der Bestandteil der Aktivitätsdefinition ist und jeweils durch ein **`Software-Produkt`** implementiert wird, dem wiederum alle vorhandenen **`Software-Installationen`** (Instanzen) annotiert sind. Aus diesen kann anhand von `Dienstmerkmalen`, `Kostensatz` und `Leistungsfaktoren` ein verfügbarer Vertreter ausgewählt werden. Bei personellen Aufgabenträgern kann die Wahl anhand der `Rolle`, die Aktivitäten oder Software-Produkten zuordenbar

ist, erleichtert werden. In der Praxis ist zu erwarten, dass jeweils nur wenige Auswahloptionen bestehen. Sofort verfügbare oder replizierbare Ressourcen wie z.B. Softwareprogramme können direkt gestartet werden [ReBD00, 2]. Bearbeitungsaufträge an Aufgabenträger mit beschränkter Verfügbarkeit oder Kapazität erfordern eine Koordination (Abschnitt 6.2.3.2).

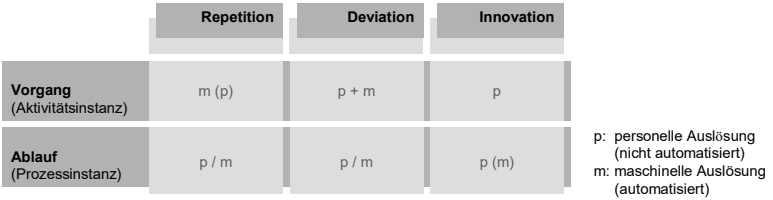


Abbildung 94: Automatisierung der Vorgangs- und Ablaufauslösung (eigene Darstellung)

Konzeptuell ist zwischen der Auslösung des gesamten Ablaufs (Prozessinstanz) und eines einzelnen Vorgangs (Aktivitätsinstanz) zu unterscheiden. Die *Vorgangsauslösung* kann in Abhängigkeit vom gewählten Steuerungsmodus personell (p) oder maschinell (m) erfolgen (Abbildung 94). Eine vollständig automatisierte Prozesssteuerung ist nur bei Repetition möglich, da hier alle anstehenden Aktivitäten aus dem Prozessschema abgelesen und direkt veranlasst werden können. Bei Deviation kann eine automatische Auslösung nur für unveränderte Prozessabschnitte erfolgen; alle neuen bzw. modifizierten Vorgänge werden vom Analytiker manuell veranlasst. Bei Innovation ist eine automatisierte Vorgangsauslösung aufgrund der unbekannten Ablaufstruktur nicht möglich. Eine personelle Vorgangsauslösung ist prinzipiell in allen drei Modi möglich und wird von allen gängigen interaktiven Analysewerkzeugen unterstützt. Sie bietet dem Analytiker Kontrolle über den Fortgang der Untersuchung. Sind Analyseprozesse als Programme kompiliert oder in Skriptsprachen codiert, wie in der Data Science üblich, entfällt diese Option.

Die *Ablaufauslösung* kann in allen Modi personell oder maschinell geschehen. Außer bei vollautomatisierten repetitiven Prozessen ist jedoch von der personellen Auslösung als Normalfall auszugehen. Bei maschineller Ablaufauslösung erkennt das Steuerungssystem den

Eintritt des Vorereignisses der Startaktivität und beginnt mit der Bearbeitung. Dies ist ebenso bei innovativen Abläufen möglich, sofern nicht der gesamte Ablauf, sondern nur Teile situativ gestaltet werden.[214] Die Auslösung der Vorgänge und Abläufe geschieht über nutzer- oder systemgenerierte Ereignisse. Zur Ablaufauslösung kommen zusätzlich zeitorientierte Ereignisse in Betracht, wenn z.B. repetitive Prozesse periodisch zu festgelegten Zeitpunkten starten sollen.

Die *Durchführung* von Vorgängen bzw. Abläufen ist ein Aufgabenträgeraspekt und wird nicht weiter betrachtet.

6.2.3.2 Koordination (S3.2)

Unter Koordination wird im Zusammenhang mit Prozessen die Abstimmung der Prozessaktivitäten bezüglich der Prozessziele und des konkurrierenden Zugriffs auf Ressourcen verstanden [Rühl92, 1165], [Gier00, 22]. Damit betrifft die Koordination auch die Verteilung von Aufträgen und Nachrichten an die zur Aktivitätsdurchführung vorgesehenen Aufgabenträger (vgl. [ReSt04, 29]) und ist zusammen mit der Vorgangsauslösung zu betrachten. Koordinationsbedarf entsteht aufgrund der Interdependenzen zwischen den Aktivitäten [Gier00, 76], [Reif03, 66] und wird daher von der gewählten Prozessstruktur beeinflusst [Gait83, 163]. Mit abnehmendem Planungsgrad ist mit steigendem Koordinationsaufwand zu rechnen, da unbekannte Abhängigkeiten zwischen Prozesselementen zur Ausführungszeit ermittelt und gehandhabt werden müssen.

Die Koordinationstheorie nach MALONE & CROWSTON [MaCr94] unterscheidet vier Grundtypen von Beziehungen zwischen Prozessaktivitäten [Gier00, 76f.] mit jeweils spezifischem Koordinationsbedarf. Der Typ der *zeitlich-technischen Abhängigkeiten (1)* betrifft die zulässigen Verknüpfungen der Aktivitäten (vgl. Abschnitt 5.5.4.3) und wird von Daten-

[214] Eine automatische Ablaufauslösung bei vollständig innovativen Abläufen ist prinzipiell dann denkbar, wenn das System z.B. in der Lage ist, das Auftreten eines definierten Sachproblems (Problemzustand) autonom zu erkennen. Es könnte dazu eine Prozessinstanz mit einer unspezifizierten Startaktivität erzeugen und dem Analytiker zur Bearbeitung vorlegen.

analysewerkzeugen dahingehend überwacht, dass nicht kompatible oder nicht versorgte Ein-/Ausgabegabeflüsse gemeldet bzw. durch entsprechende Ausprägungen des Vorgangszustands signalisiert werden.[215] Dies gilt ebenso für den Typ *ressourcenbezogener Abhängigkeiten (2)* bei maschinellen Aufgabenträgern, deren Nichtverfügbarkeit zu einem nicht bereiten Vorgang führt. Workflow-orientierte Systeme unterstützen häufig das Einreihen von Bearbeitungsaufträgen in eine ressourcenspezifische Warteschlange, aus der die Aufgabenträger bei Verfügbarkeit ein Arbeitselement lesen und ausführen [RaSc99, 10], [GCC+04, 323]. Bei personellen Aufgabenträgern obliegt die Koordination in der Regel der Projektleitung. Auch zur Koordination des konkurrierenden Schreib-/Lesezugriffs auf Datenressourcen [Gier00, 140] bieten sowohl Datenbankverwaltungssysteme als auch Analysesoftware meist Unterstützung.

Der Typ *hierarchischer Beziehungen (3)* ist gemäß dem in dieser Arbeit vertretenen Verständnis von Analyseprozessen nur gemeinsam mit dem Typ *verschiedener Prozesszusammenhänge mit unterschiedlichen Zielen (4)* von Bedeutung.[216] Sie werden im Anschluss an die folgenden Überlegungen erörtert.

Koordinationsgrundlagen

Koordinationsverfahren können nach ihrem Umgang mit Interdependenzen differenziert werden. Bei *expliziter Beachtung* werden bekannte Interdependenzen durch verhaltensverbindliche Vorgaben erfasst und berücksichtigt. Die *implizite Beachtung* von Abhängigkeiten erfolgt während der Prozessrealisierung durch die Aufgabenträger, denen zur Wahrnehmung erforderlicher Abstimmungen ein gewisses Maß an Autonomie zugestanden wird [Gait83, 159]. Die Art der Interdependenzberücksichtigung ist demnach davon abhängig, inwiefern detaillierte Prozessvorlagen verfügbar sind. Vollständige Workflow-Schemata

[215] Die Anzeige des Zustands gemäß Zustandsmodell erfolgt häufig mithilfe der Ampelmetapher, indem nicht bereite Vorgänge etwa mit rot, bereite mit gelb und abgeschlossene mit grün gekennzeichnet sind.

[216] Ausführbar sind jeweils nur einzelne Aktivitäten als Blattknoten einer Aufgabendekomposition, woraus eine nicht-hierarchische Ablaufstruktur resultiert.

treffen präzise Vorgaben und werden für die Repetition vorausgesetzt. Prozessschablonen machen weniger detaillierte Vorgaben und lassen Entscheidungsspielräume. Kontextregeln geben keine Aktivitätsfolgen vor, sondern spezifizieren in einzelnen Situationen zulässige, empfehlenswerte oder unzulässige Verhaltensweisen [Gait83, 177-179]. Sie erleichtern die Suche nach einer Lösung und sind somit, ebenso wie Schablonen, für Deviation und Innovation hilfreich. Vorlagen und Regeln werden aus Prozesszielen abgeleitet [Gait83, 193], deren explizite Kenntnis bei allen Gestaltungsentscheidungen von Vorteil ist. Die erläuterten Vorgaben bilden wichtige Grundlagen der Koordination. Sie dienen als *Koordinationsprotokoll*, dessen Durchsetzung die Ausführbarkeit und Korrektheit des Ablaufs sicherstellen soll [AAD+04, 12], [LoDi04, 14f.].

Koordinationsprinzipien

Als grundlegende Koordinationsprinzipien gelten die hierarchische und die nicht-hierarchische Form [Gait83, 159], [FeSi13, 201]. *Hierarchische Koordination* funktioniert gemäß dem Regelkreisprinzip, indem eine übergeordnete Steuerkomponente Anweisungen an untergeordnete Komponenten gibt. *Nicht-hierarchischer Koordination* liegt das Verhandlungsprinzip zugrunde, wonach gleichberechtigte Komponenten zur Abstimmung ihrer lokalen Teilziele unter Berücksichtigung der gegenseitig beeinflussten Handlungsräume interagieren [FeMa95a, 6].

Das typische Szenario der Datenanalyse ist die hierarchische Koordination, bei der ein einzelner Analytiker einen Analyseablauf vollständig bearbeitet und sich hierzu eines oder mehrerer interaktiver Analysewerkzeuge bedient, über die er Aktivitätsdurchführungen anstößt. Anstelle interaktiver Auslösung ist auch eine teilweise (Skript-basierte Datenanalyse; Data Science) oder vollständige Automatisierung der Koordination möglich (Repetition).

Werden komplexere Untersuchungsvorhaben im Rahmen der Projektplanung (Z5, Abschnitt 5.4.7) in Leistungspakete gegliedert und an mehrere Analytiker oder Teams zur Durchführung übergeben, ist eine Abstimmung zwischen mehreren Aufgabenträgern erforderlich. Sie erfolgt typischerweise nicht-hierarchisch in Selbstabstimmung, da die

menschliche Kommunikation in innovativ-kreativen Situationen den effektivsten Koordinationsmechanismus darstellt (vgl. [Gait83, 212f.], [Reif03, 66]).[217] Die Koordination bezüglich einzelner Leistungspakete geschieht lokal durch eine Abschnittsteuerung, die von einer Zentralsteuerung mit Zielen und Plänen versorgt wird. Die Gliederung ist z.B. nach Analyseproblemen oder Prozessphasen möglich. Letztere gewinnt mit Data Science zunehmend an Bedeutung, die komplexe Datentransformationen leistet und vielfältigen Auswertungs- und Nutzungsformen zuführt.

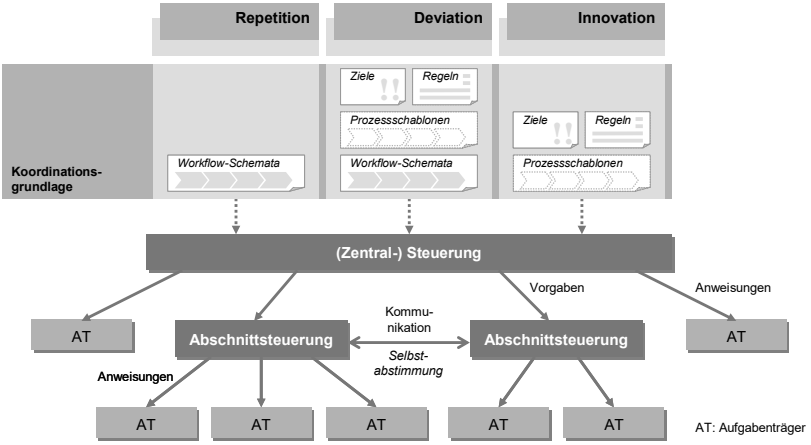


Abbildung 95: Koordinationsmechanismen von Datenanalyseabläufen (eigene Darstellung)

In Abbildung 95 sind die Koordinationsgrundlagen und -prinzipien schematisch dargestellt. Die Repetition ist auf detaillierte Workflow-Vorlagen angewiesen, während Deviation und Innovation von Prozessschemata, Regeln und Zielspezifikationen profitieren können. Grundsätzlich wird ein *hierarchisch-zentrales Koordinationskonzept* realisiert, indem Vorgänge durch direkte Anweisungen an die jeweiligen Aufgabenträger ausgelöst werden. Für komplexere Analysen kann ein

[217] Die nicht-hierarchischen Methoden der Koordinationstheorie (z.B. auf Basis von Verrechnungspreisen und Kosten-Nutzen-Analysen, vgl. [Gait83, 235ff.]) erscheinen im Kontext der Datenanalyse wenig geeignet.

hierarchisch-dezentrales Koordinationskonzept mit kooperierenden Abschnittsteuerungen zum Einsatz kommen. Letztere steuern ihre Prozessabschnitte hierarchisch und kommunizieren zur Selbstabstimmung in dem von der Zentralsteuerung durch Vorgaben gesetzten Rahmen, z.B. um über die Bereitstellung von Daten zu verhandeln. Die Zentralsteuerung hat in Bezug auf diese Abschnitte nur relativ einfache Interdependenzen zwischen Leistungspaketen zu koordinieren.

6.2.3.3 Ablaufüberwachung (S3.3)

Zur Prozesssteuerung gehört schließlich auch die Überwachung des Ablaufs [ReBD00, 2], die auf Grundlage des Instanzmodells realisiert wird. Ziele der Ablaufüberwachung sind (1) die Verfolgung des Prozessfortschritts, (2) die Sicherstellung der Korrektheit durch Erkennung und Behandlung von Fehlern und Ausnahmen, (3) die Gewährleistung der Zielorientierung durch Zielkontrolle sowie (4) die Einhaltung von Finanz- und Zeitrestriktionen [Gier00, 140], [GCC+04, 322], [DRRA05, 7]. Im Gegensatz zur Revision (vgl. Abschnitt 7.2.2), die abgeschlossene Abläufe zum Gegenstand hat, betrachtet die Überwachung laufende Prozessinstanzen [GCC+04, 322]. Die genannten Ziele werden im Folgenden näher erläutert.

Fortschrittsüberwachung

Die Fortschrittsüberwachung beruht, wie auch die Verfolgung aller anderen Überwachungsziele, auf der als *Monitoring* bezeichneten, technischen Ablaufüberwachung, die das zeitliche Ablaufgeschehen und die Prozessparameter (vgl. Abschnitt 2.4.2) beobachtet und daraus den aktuellen Status der Prozessinstanz ableitet [JaBS97, 202f.]. Bei detaillierten Prozessvorlagen ist aus dem Anteil bereits abgeschlossener Vorgänge unmittelbar der Bearbeitungsfortschritt des Ablaufs abzulesen, mit zunehmend abstrakten Schemata fällt dies jedoch schwerer. Sind Analyseketten abzuarbeiten, ist aus der Menge abgeschlossener Einzelabläufe auch der Fortschritt innerhalb der Analysekette ersichtlich. Die Fortschrittsüberwachung ist zugleich Voraussetzung für die korrekte Vorgangsauslösung, da erst mit Signalisierung der Nachereignisse

abgeschlossener Vorgänge die jeweils nachfolgenden Vorgänge initiiert werden können (vgl. [FeSi13, 102f.]).

Korrektheitsüberwachung

Mit zunehmendem Gestaltungsanteil steigen die Anforderungen an die Rückmeldungen über die Aktivitätsausführung, da variable und Ad-hoc-Abläufe ohne detaillierte Angaben über den Erfolg einzelner Vorgänge nicht sinnvoll realisierbar sind. Fehler während eines Vorgangs sowie Störungen im Sinne abweichender Kontextfaktoren sind zu identifizieren und durch geeignete Maßnahmen zu behandeln. Gestaltungseingriffe in die Ablaufstruktur wie im Deviations- und Innovationsmodus können die Realisierung des gewünschten Prozessverhaltens erschweren. Die Ablaufüberwachung muss daher gewährleisten, dass alle vor dem Eingriff gültigen Korrektheitsbedingungen (vgl. Abschnitt 5.1.3.2) auch nachher gelten [ReDa98, 2].

Zielkontrolle

Die konsequente Ausrichtung der Untersuchung auf das gesetzte Analyseziel (*Zielorientierung*) stellt ein mächtiges Instrument zur Reduzierung und Bewältigung der Analysekomplexität dar (R1.1 und B2.1, vgl. Abschnitte 3.2.4.2 bzw. 3.2.4.3) und hat großen Einfluss auf Effektivität und Effizienz der Analyse. Die Instrumentalbeziehung zwischen Analyseziel und Prozesstyp kann im Rahmen der Planung explizit hergestellt werden. Die Aufrechterhaltung dieser Beziehung auf Instanzebene ist von der Ablaufsteuerung durch ***Zielkontrolle*** zu überwachen.[218]

Die Aufgabe der Zielkontrolle besteht in der *Erkennung von Zielabweichungen*, d.h. von Situationen, in denen der Prozessablauf nicht mehr dem gesetzten **`Analyseziel`** folgt, und der *Zielneuausrichtung* im Falle einer Abweichung (Wiederherstellung der Zielorientierung). Zielabweichungen können in folgenden Ausgangssituationen auftreten:

---

[218] So fordert auch JABLONSKI, die Beziehung zwischen Ziel, Prozesstyp und Prozessinstanz stets aufrecht zu erhalten [Jabl00, 348].

- *Am Übergang zu einer neuen Analyse („**Kupplung**“)*: Aus den Erkenntnissen einer abgeschlossenen Analyse hat sich eine neue Fragestellung ergeben, die durch eine Anschlussanalyse beantwortet werden soll, welche nicht bereits in einer geplanten Analysekette enthalten ist.[219] Das neue Analyseziel ist hierbei explizit zu formulieren.
- *Während der Ausführung eines Analyseprozesses („**Weiche**“)*: Im Rahmen der Deviation oder Innovation wird situativ ein neuer Vorgang ausgeführt, der möglicherweise einem abweichenden Ziel folgt. Da diese Abweichung innerhalb eines laufenden Prozesses erfolgt, ist das mit dem neuen Vorgang verknüpfte Analyseziel allenfalls implizit bekannt.

Abbildung 96 zeigt den Ablauf der Zielkontrolle. Zunächst ist zu prüfen, ob eine *Zielabweichung* vorliegt, da andernfalls kein Handlungsbedarf besteht und die Analyse fortgeführt werden kann. Der Kupplungssituation ist die Abweichung immanent. Unterscheiden sich ursprüngliches und neues Analyseziel, ist im nächsten Schritt die *Problemrelevanz* des neuen Ziels für die aktuelle Untersuchung zu klären (***qualitative Zielkontrolle***). Dies ist dann der Fall, wenn das neue Analyseziel geeignet ist, einen Beitrag zur Lösung des aktuellen **`Sachproblems`** zu leisten. Ist Problemrelevanz nicht gegeben, ist das neue Ziel zu verwerfen. Ist Problemrelevanz zwar nicht unmittelbar gegeben, in ähnlichen Fällen aber denkbar, so kann das neue Analyseziel als Vorlage für die Konzeption späterer Untersuchungen mit vergleichbarem Sachproblem archiviert werden. In der Situation „Weiche“ ist die Untersuchung jeweils mit dem ursprünglichen Ziel fortzusetzen. In der Situation „Kupplung“ ist die Untersuchung beendet. Kann hingegen unmittelbare Relevanz für das aktuelle Sachproblem festgestellt werden, so sollte das neue Analyseziel in die Untersuchung einbezogen werden. Das weitere Vorgehen ist von dessen relativer Relevanz im Vergleich zum ursprünglichen Ziel abhängig. Wird sie als geringer eingestuft, so sollte das neue Analyseziel suspendiert und dem ursprünglichen Ziel Priorität ein-

[219] Die Einschränkung auf ungeplante Analysen erfolgt unter der Annahme, dass die Planung von Analyseketten bereits zielorientiert geschieht.

geräumt werden. Das neue Ziel kann im Anschluss wieder aufgenommen werden.

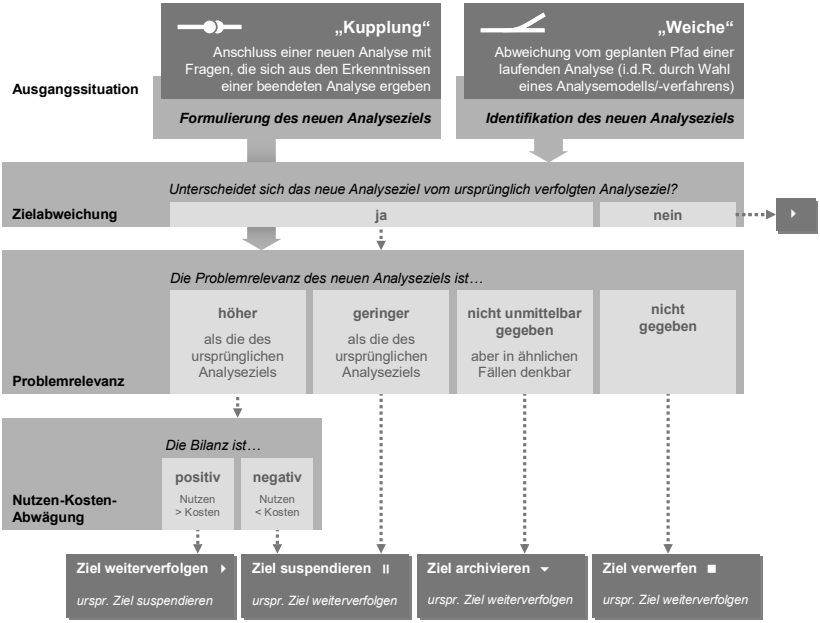


Abbildung 96: Zielkontrolle und Zielneuausrichtung in Datenanalyseprozessen (eigene Darstellung)

Analyseziele mit höherer relativer Relevanz[220] werden einer *Nutzen-Kosten-Abwägung* unterzogen, um den Chancen eines potenziellen Erkenntniszuwachses durch eine weitere bzw. neu ausgerichtete Analyse die damit verbundenen Projektrisiken gegenüberzustellen (vgl. [Bend02, 18f.]). Letztere bestehen in dem zusätzlichen Ressourcenverbrauch, der zu einer Überschreitung des Zeit- oder Kostenbudgets des Projekts führen kann. Sie sind auf Basis einer ***quantitativen Zielkontrolle*** abschätzbar, die letztlich eine Bewertung der bereits ausge-

[220] Falls für die aktuelle Untersuchung bereits suspendierte Analyseziele existieren, werden in die relative Relevanzabschätzung alle unmittelbar relevanten Ziele einbezogen und priorisiert. Im nächsten Schritt wird jeweils nur das Analyseziel mit der höchsten Priorität betrachtet.

führten und noch zu erwartenden Einzelanalysen (Zielebene) der aktuellen Untersuchung mit Zeit- und Kostengrößen beinhaltet (vgl. Abschnitt 3.3.3.2). Übersteigen die zu erwartenden Kosten bzw. Risiken den Nutzen, ist das Analyseziel zu suspendieren und die Untersuchung mit dem ursprünglichen Ziel fortzuführen. Andernfalls ist dem neuen Ziel Vorrang zu geben und das bisherige Analyseziel zu suspendieren.

Suspendierte sowie archivierte Analyseziele werden aus der aktuellen Untersuchung in eigene Prozesse mit differenzierten Zielen abgespalten. Diese ***Prozessabspaltung durch Zieldifferenzierung*** begrenzt wirksam den Analyseraum explorativer Untersuchungen (vgl. R1.2, Abschnitt 3.2.4.2). Sie gewährleistet zugleich, dass stets nur jene Analyse mit der aktuell höchsten Problemrelevanz und niedrigen Projektrisiken betrieben wird, um möglichst frühzeitig aussagefähige Ergebnisse zu erzielen. Weitere relevante Analyseziele mit höheren Risiken können zu einem späteren Zeitpunkt ausgeführt werden, sofern die Budgets noch nicht ausgeschöpft sind.

Die Feststellung der Zielabweichung in der Situation „Weiche“ ist mithilfe der **`Funktion`** des betroffenen Vorgangs bzw. der ihm zugrundeliegenden Aktivität möglich. Ist die Funktion des neuen Vorgangs nicht mit jener der ursprünglich geplanten Aktivität identisch oder über eine Spezialisierungsrelation verknüpft, ist eine Zielabweichung anzunehmen. Dieser Ansatz ist bei Deviation unmittelbar anwendbar. Da bei Innovation keine geplante Aktivität existiert, muss die Funktion des Vorgangs auf Konformität mit dem Analyseziel geprüft werden. Eine Abweichung ist z.B. offensichtlich, wenn innerhalb eines Analyseablaufs zur Prognose des Antwortverhaltens ein Vorgang zur Erkennung von Assoziationsregeln ausgelöst wird. In anderen Situationen mag sich die Feststellung einer Abweichung schwieriger darstellen.[221]

[221] Eine Zielabweichung auf Ablaufebene tritt typischerweise innerhalb der Analysephase auf, da nur deren Aufgaben eine direkte Beziehung zum Analyseziel besitzen.

Überwachung von Finanz- und Zeitrestriktionen

Die Einhaltung von Finanz- und Zeitrestriktionen lässt sich auf Basis des Instanzmodells durch Abgleich der protokollierten Kosten und Zeiten mit den `Budget-` und `Zeitrestriktionen` der **`Informationsmaßnahme`** überwachen. Für Budgetrestriktionen ist hierfür vorauszusetzen, dass allen eingesetzten Ressourcen ein zuverlässiger `Kostensatz` zugewiesen ist. Diese Kosten- und Zeitkonten sowie die im Rahmen der Zielkontrolle erfolgte Nutzen-Kosten-Abschätzung bilden die Grundlage für ein systematisches *Risikomanagement* auf Projektebene. Eine neue Windung des Datenanalyse-Spiralmodells sollte demnach nur betreten werden, wenn die Abwägung der zugehörigen Risiken positiv ausfällt (vgl. Abschnitt 3.3.3.2).

6.2.3.4 Zusammenfassung: Ablaufbegleitung

Die ablaufbegleitenden Aufgaben Vorgangsauslösung, Koordination und Ablaufüberwachung stehen in starker Wechselwirkung und bilden die Hauptinhalte der Prozesssteuerung i.e.S. Sie stützen sich auf das Instanzmodell, das von der nachfolgend beschriebenen Aufgabe S4 gepflegt und stetig fortgeschrieben wird.

6.2.4 Protokollierung und Dokumentation (S4)

Ziel der vierten Aufgabe der Prozesssteuerung ist die Führung eines stets aktuellen Instanzmodells, um die zielkonforme Steuerung von Prozessabläufen zu unterstützen. Die darin abgebildeten Angaben zum Zustand von Prozessinstanz, Vorgängen und Datenflüssen bilden in ihrer Summe den *Ablaufkontext* der Prozessinstanz [Reif03, 83], [Erl04, 96f.]. Der Ablaufkontext schließt Verfahrens-, Ausführungs- und Datenkontext ein, d.h., er erfasst die genutzten Operatoren und Software-Produkte, die konkret eingesetzten Software-Installationen und Personen sowie die bearbeiteten Informationsobjekte und Datenquellen (vgl. Abschnitte 4.5.3 und 4.7.2).

Das Instanzmodell enthält sämtliche Angaben über den Prozesstyp (`Workflow`), alle an der Instanz vorgenommenen `Änderungen` sowie jeweils die aktuellen und protokollierten Ausführungs- und Bearbei-

tungszustände (`Zustand` bzw. `Ereignisprotokoll`) aller Vorgänge und Informationsobjekte [AAD+04, 14], [RWRW05, 255]. Ist kein Prozessschema vorgegeben, ist der Prozesstyp aus dem Instanzmodell zu induzieren. In den Ereignis- und Änderungsprotokollen sind alle Zustandsänderungen und Abweichungen vom Planablauf dokumentiert (vgl. [Gier00, 290]). Neben Art und Kontext der vorgenommenen Änderungen (etwa Operationstyp und Position der betroffenen Aktivität im Ablauf) können auch deren Ursachen festgehalten werden, um sie jederzeit nachvollziehen und bei Bedarf im Rahmen der Prozessrevision in eine neue Schemaversion überführen zu können (Evolution) [WRRW05, 2].

Es wird deutlich, dass das Instanzmodell mehr Informationen aufnehmen kann als die von einem Prozessausführungswerkzeug protokollierten technischen Angaben. Während des Ablaufs vom Analytiker aufgezeichnete Kommentare, z.B. über gewonnene Erfahrungen oder Fehlerursachen [Jung02, 45], können für spätere Projekte mit ähnlicher Problemstellung von großem Wert sein.

Im Instanzmodell (**`Prozessinstanz`**) erfolgt insbesondere auch die Dokumentation der `Ergebnisbewertung` und der `Prozessbewertung`, die erst nach Abschluss des Ablaufs möglich ist und damit die Schnittstelle zur Prozessrevision darstellt. Die Analyseergebnisse selbst sind in Form von Informationsobjekten mit der Instanz verknüpft. Aus den Ergebnissen erstellte Berichte und Dokumente können über `Links` an der Instanz gespeichert werden und sind gemeinsam mit ihr über deren Beschreibungselemente in der Prozessbibliothek auffindbar.

6.2.5 Zusammenfassung: Aufgaben der Prozesssteuerung

Die vorausgehenden Abschnitte beschreiben Aufgaben, die im Rahmen der Steuerung von Datenanalyseprozessen auszuführen sind. Sie sind in ihrer Gesamtheit notwendig, können jedoch nicht in eine eindeutige Reihenfolge geordnet werden. Ihre Realisierung kann teils durch Analyse- oder Prozessausführungswerkzeuge unterstützt werden, in

anderen Fällen (z.B. bei der Koordination personeller Aufgabenträger) obliegt sie dem Projektleiter oder Analytiker.

6.3 Ansätze zur Steuerung von Datenanalyseprozessen

Die drei Anwendungsfälle Repetition, Deviation und Innovation (vgl. Abschnitt 6.1.2) erheben jeweils spezifische Anforderungen an die Prozesssteuerung. Die folgenden Abschnitte stellen geeignete Ansätze (Steuerungsmodi) zur Realisierung dieser Anwendungsfälle vor.

6.3.1 Steuerungsmodus Repetition

Der Steuerungsmodus Repetition eignet sich für programmierte Abläufe, die nach Maßgabe eines Workflow-Schemas unmodifiziert ausgeführt werden können [Gait83, 178]. Gestaltungsentscheidungen sind nicht zu treffen. Der Workflow dient als präskriptives Modell, das alle notwendigen Regeln und Bedingungen zur Ablaufsteuerung enthält [Reif03, 74]. Den Aufgabenträgern können auf dieser Grundlage eindeutige Anweisungen gegeben werden [Gait83, 180]. Dem Analytiker bleibt die Aufgabe, ein geeignetes Prozessschema auszuwählen und gegebenenfalls für den aktuellen Analysefall zu instanziieren (Individualisierung) [RuPR99, 226], z.B. durch Auswahl konkreter Datenbestände oder Festlegung fallspezifischer Analyseparameter (geplante Verhaltensflexibilität, Abschnitt 5.1.5.1). Im Schema auf Typebene enthaltene Ablaufvarianten oder alternative Verzweigungen im Prozessfluss werden auf Instanzebene entsprechend dem Ablaufkontext situativ aufgelöst (vgl. [RuPR99, 228]).

Ein repetitiver Ablauf kann grundsätzlich vollautomatisch gesteuert werden [HiWi01, 86]. Wünscht der Analytiker die Möglichkeit der personellen Ablaufkontrolle, kann er eine manuelle Weiterschaltung zwischen Vorgängen wählen. Eingriffe in die Ablaufsteuerung sind nur erforderlich, wenn über im Schema enthaltene Ablaufvarianten nicht auf Basis von Kontextfaktoren deterministisch entschieden werden kann. Sie sind dem Analytiker von der Steuerung zur Entscheidung vorzulegen.

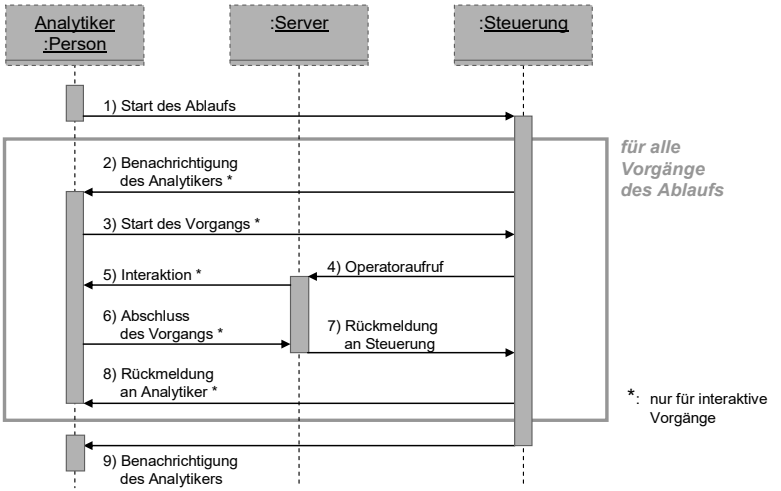


Abbildung 97: Sequenzdiagramm zum Steuerungsmodus Repetition (eigene Darstellung)

Das typische Verhalten bei maschineller Steuerung eines repetitiven Ablaufs, der kooperativ von einem Analytiker und einer oder mehreren Software-Installationen (Servern) durchgeführt wird, zeigt das Sequenzdiagramm in Abbildung 97 (vgl. hierzu [Jabl05, 206-208]). Der Ablauf wird vom Analytiker gestartet[222] (1) und von der Steuerung koordiniert. Für alle vollautomatisierten Vorgänge innerhalb des Ablaufs ruft die Steuerung einen Server auf (4), der den Abschluss seiner Arbeit an die Steuerung zurückmeldet (7). Im Falle teilautomatisierter Vorgänge benachrichtigt die Steuerung den Analytiker über einen anstehenden Arbeitsauftrag (2). Ist der Analytiker bereit, startet er den Vorgang (3), woraufhin die Steuerung den Operator beim jeweiligen Server aufruft (4) und der Vorgang interaktiv ausgeführt wird (5). Der Analytiker beendet den Vorgang, z.B. durch Bestätigung seiner Eingaben (6), woraufhin der Server die Steuerung benachrichtigt (7), die wiederum eine Rückmeldung an den Analytiker übermittelt (8). Nach Abschluss aller Vorgänge benachrichtigt die Steuerung den Anwender über die vollständige Abarbeitung des Prozesses (9).

[222] Es ist grundsätzlich auch eine automatische Prozessauslösung denkbar; vgl. Abschnitt 6.2.3.1.

6.3.2 Steuerungsmodus Innovation

Der Steuerungsmodus Innovation wird gewählt, wenn kein (vollständiges) Workflow-Schema existiert, das ein Handlungsprogramm zur Prozessrealisierung vorgibt [Gait83, 178]. Die zunächst „fehlende Prozessstruktur" [Deit00, 276] ist zur Ausführungszeit situativ zu ergänzen. Die Innovation kann sich auf den Gesamtprozess oder auf einzelne Prozesssegmente beziehen. Beiden Fällen liegt das Grundprinzip der *Ablaufgestaltung nach Bedarf* zugrunde, nach dem ungeplante Aspekte erst dann spezifiziert werden, wenn sie zur Ausführung anstehen (vgl. [WRRW05, 9]). Der zweite Fall ist bei Vorlagen in Form unreiner Prozessschablonen gegeben, die neben Aktivitäten auch lediglich aus Außensicht spezifizierte Aufgaben enthalten (geplante Strukturflexibilität, Abschnitt 5.1.5.1). Reine Schablonen lösen bezüglich der Aufgabeninnensicht Gestaltungsbedarf für den gesamten Prozess aus. Zur Ausfüllung der Gestaltungsspielräume sind alle vier Basisansätze der Prozessplanung aus Abschnitt 5.3 anwendbar (Operatorkomposition, Fragmentrekombination, Aufgabendekomposition oder Vorlagenspezialisierung).

Innovative Ablaufsteuerung ist mithilfe interaktiver Analysewerkzeuge unmittelbar realisierbar, da nicht ausreichend spezifizierte Elemente in der diagrammgestützten Benutzerschnittstelle direkt sichtbar sind und manipuliert werden können. Auf Basis algorithmischer Beschreibung arbeitende Systeme (z.B. Skriptsprachen, klassische Workflow-Management-Systeme) erfordern zum Umgang mit Innovation spezielle Modellierungskonstrukte. Einen geeigneten Vorschlag für Late Modeling[223] in Petrinetz-basierten Systemen unterbreitet DEITERS [Deit00, 279-281], nach dem zur Entwurfszeit nicht oder nicht vollständig beschreibbare Aktivitäten als *Black-Box-Aktivitäten* gekennzeichnet werden. Beim Auftreten definierter modifikationsauslösender Ereignisse wird der Anwender zur Laufzeit um eine Spezifikation dieser Aktivitäten gebeten.

[223] Late Modeling bezeichnet die Ausnutzung geplanter Strukturflexibilität durch Operatorkomposition, Aufgabendekomposition oder Vorlagenspezialisierung; vgl. Abschnitt 5.1.5.1.

Somit kann jede Prozessinstanz verschiedene Spezifikationen der Black-Box-Aktivität enthalten [Deit00, 279]. Innovation einzelner Prozesssegmente generiert ebenso wie Deviation gegenüber der Vorlage variable Abläufe. Während letztere jedoch von der Detailspezifikation einer Vorlage ungeplant abweicht, definiert Innovation nicht ausreichend detaillierte Spezifikationen neu.

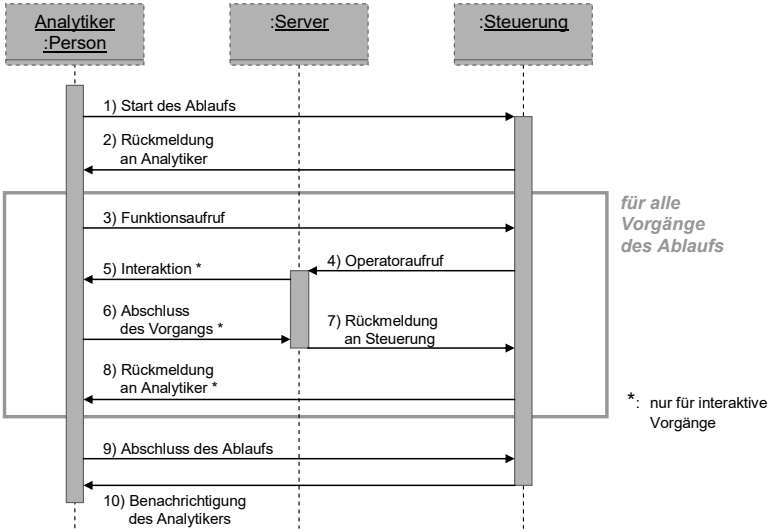


Abbildung 98: Sequenzdiagramm zum Steuerungsmodus Innovation (eigene Darstellung)

Die Ablaufsteuerung ist bei Innovation in Bezug auf die betroffenen Prozessabschnitte nicht voll automatisierbar, sondern erfolgt in Kooperation mit dem Analytiker. Das Sequenzdiagramm in Abbildung 98 stellt das typische Verhalten in diesem Modus dar. Der Start des Ablaufs erfolgt durch den Analytiker (1) und wird von der Steuerung quittiert (2). Jeder Funktionsaufruf des Analytikers konstituiert einen neuen Vorgang (3) und wird von der Steuerung jeweils in einen Operatoraufruf an einen Server übersetzt (4). Interaktive Operatoren initiieren einen Dialog mit dem Analytiker (5), den dieser nach Erledigung der Aufgabe schließt (6). In allen Fällen meldet der Server den Abschluss des Vorgangs an die Steuerung (7), die jeweils eine Rückmeldung an den

Analytiker sendet (8). Erachtet dieser alle notwendigen Schritte als ausgeführt, informiert er die Prozesssteuerung über das Ende des Ablaufs (9) und erhält umgehend eine Rückmeldung über die korrekte Protokollierung der Vorgänge (10).

6.3.3 Steuerungsmodus Deviation

Der Steuerungsmodus Deviation wird eingesetzt, wenn zwar Prozessvorlagen existieren, während der Prozessausführung auftretende *Ausnahmen* (vgl. Abschnitt 5.1.4) aber eine Modifikation dieser Vorlage in Bezug auf die zu realisierenden Aktivitäten oder deren Reihenfolge erfordern (vgl. [Gait83, 178], [RuPR99, 227]). Die vorzunehmenden Anpassungen sind allesamt ungeplant und realisieren Flexibilität durch Schemaabweichung zur Ausführungszeit (vgl. Abschnitt 5.1.5.1). Die Ausnahmebehandlung kann sich auf Verhaltens- und Strukturmerkmale des Ablaufs richten. Verhaltensmodifikationen betreffen die Makro- oder Mikroparametrisierung von Aktivitäten (vgl. Abschnitt 5.5.6). Strukturmodifikationen äußern sich als additive, subtraktive (Hinzufügen bzw. Entfernen von Prozesselementen) und reihenfolgebezogene Maßnahmen (Umordnen von Flussbeziehungen) [ReDa98] sowie im Austausch von Operatoren.

Eine Unterscheidung nach der zeitlichen Wirkung einer Modifikation mündet in die beiden Standardverfahren der Ausnahmebehandlung nach ERFLE & VOGEL [ErVo92], [Gier00, 212f.]:

- *Planänderungen* (*routing exceptions*) betreffen nur künftige, folgende Arbeitsschritte; bereits vollendete Vorgänge müssen nicht zurückgenommen werden. Planänderungen dienen der Anpassung des Ablaufs an aktuelle Kontextbedingungen, bevor ungeeignete Aktivitäten ausgeführt werden. Diese ist wünschenswert, jedoch nicht immer realisierbar.
- *Rücknahmen* (*backtracking exceptions*) setzen den Bearbeitungsstand bereits realisierter Vorgänge zurück, die gegebenenfalls unter veränderten Bedingungen erneut ausgeführt oder durch andere Aktivitäten ersetzt werden sollen. Hierzu muss der Ablaufkontext in einen früheren Zustand überführt werden. Rücknahmen dienen gewisser-

maßen der Stornierung einzelner Schritte (vgl. [AAD+04, 18]) im Zuge der Fehlerbehandlung.

Rücksprünge sind eine spezielle Ausprägung von Rücknahmen, bei der bereits realisierte Vorgänge nicht explizit storniert, sondern die zugehörigen Aufgaben erneut ausgeführt werden (Iterationen). Sie treten in der Datenanalyse regelmäßig auf, da die Kompensation der Wirkungen ausgeführter Vorgänge oft durch Überschreiben der erzeugten Informationsobjekte durch neue Versionen erreichbar ist.[224] Der Ablaufpfad verlängert sich durch die Aufgabenwiederholung. Stornierende Rücknahmen verkürzen den Ablaufpfad auf einen früheren Zustand.

Die Durchführung der Anpassungen erfolgt prinzipiell durch den Analytiker, der jedoch bei der Erkennung und Diagnose von Ausnahmen methodisch-technisch unterstützt werden kann. Die Deviation kann als partielle Neuplanung betrachtet werden und alle zur Planung geeigneten Ansätze nutzen (vgl. [Schm97, 157]). DELLAROCAS & KLEIN [DeKl00] schlagen einen wissensbasierten Ansatz zur Prozesssteuerung vor, nach dem Prozessschemata mit einer Bibliothek von ausnahmeträchtigen Aktivitäten bzw. Funktionen abgeglichen werden. Bekannte Ausnahmetypen lassen sich jeweils mit geeigneten Erkennungs- und Behandlungsprozeduren verknüpfen, die vor oder während der Prozessausführung aktiviert werden und die Vermeidung bzw. Behandlung erkannter Störereignisse unterstützen oder automatisiert bewältigen. Derartiges Wissen lässt sich nach dem in dieser Arbeit vertretenen Ansatz in Form von Kontextregeln modellieren. Alternative Vorschläge stellen konkrete Erfahrungen mit Ausnahmen in ähnlichen Situationen bereit. WEBER ET AL. [WeWB04] präsentieren ein entsprechendes System, das Prozessinstanzänderungen mittels CBR unterstützt. RINDERLE ET AL. [RWRW05] entwickeln diesen Vorschlag für das adaptive Prozessmanagement weiter. Ihr System erfragt Kontext und Gründe der Abweichung vom Anwender und präsentiert Lösungsvor-

[224] Gegenüber operativen (Geschäfts-) Prozessen gestalten sich zugehörige Kompensationsmechanismen für Datenanalyseprozesse einfacher, da erzeugte Informationsobjekte in der Regel nicht von anderen Prozessen gelesen werden und sich nicht fortpflanzen. Zum Rücksetzen von Workflow-Instanzen vgl. [Reic00, 159ff].

schläge mit konkreten Änderungsoperationen. Jeder Fall ist mit jenem Prozessschema verknüpft, auf das er zutrifft.

Das Verhalten der Prozesssteuerung bei Deviation entspricht grundsätzlich jenem der Innovation (vgl. Abbildung 98). Im Zuge des Funktionsaufrufs in Schritt 3 ergreift der Analytiker eine Modifikationsmaßnahme, die entweder explizit vor oder implizit zusammen mit dem Funktionsaufruf realisiert wird. Auf die Darstellung eines Sequenzdiagramms wird daher verzichtet.

6.4 Zusammenfassung: Steuerung von Datenanalyseprozessen

Die Steuerung von Datenanalyseprozessen umfasst eine Reihe eng verzahnter Aufgaben, die größtenteils simultan und begleitend zur Prozessdurchführung zu bearbeiten sind. Die Notwendigkeit, während einer Analyse rasch auf Ausnahmen reagieren und Prozessabläufe entsprechend anpassen zu können, führt zu einem beachtlichen Gestaltungsanteil der Steuerung von Datenanalyseabläufen. Diesem tragen drei Steuerungsmodi Rechnung. Neben der unmodifizierten Durchführung vordefinierter Workflows (Repetition) erfordert die explorativ-evolutionäre Natur der Datenanalyse insbesondere auch die Möglichkeit zur situativen Abweichung von Prozessvorlagen (Deviation) sowie zur Prozessspezifikation zur Ausführungszeit (Innovation). Effektive Datenanalyse kann nur gelingen, wenn die Prozesssteuerung dem Analytiker erlaubt, jederzeit flexibel zwischen den drei Modi zu wechseln.

Diese Flexibilität darf jedoch nicht zulasten der Korrektheit der Prozesse gehen. Auch bei situativen Änderungen müssen Zielorientierung und Effektivität der Abläufe gewahrt bleiben [DaRe04, 5]. Die Berücksichtigung all dieser Formalziele kann in komplexen Steuerungsmechanismen münden, welche gegebenenfalls die Benutzerfreundlichkeit beeinträchtigen (vgl. [DRRA05, 3]). Mit der Entwicklung praxistauglicher Konzepte und Werkzeuge zum adaptiven Prozessmanagement befassen sich mehrere Forschungsgruppen, deren für diese Arbeit relevante Beiträge in die Darstellung eingeflossen sind. Einen Überblick gibt REICHERT [Reic00].

Wurden im Rahmen der Prozesssteuerung Gestaltungsmaßnahmen ergriffen und sollen künftig alle Prozessinstanzen desselben Typs dem modifizierten Ablauf folgen, müssen Modifikationen auf die Typebene propagiert werden [ElKe00, 202], indem eine neue Version des ursprünglichen Prozessschemas erstellt wird. Solche Veränderungen auf Typebene werden als ***Schemaevolution*** bezeichnet [WRWR05b, 4] und im Rahmen der Prozessrevision (Abschnitt 7.4.1.1) behandelt.

7 Revision von Datenanalyseprozessen

Das abschließende Kapitel zur Managementmethodik betrachtet die systematische ganzheitliche Revision von Datenanalyseprozessen. Abschnitt 7.1 diskutiert Gegenstand und Inhalte der Revisionsaufgabe. Im Anschluss wird das hierzu empfohlene Vorgehen erläutert, gegliedert in die Beurteilung durchgeführter Datenanalysen (Abschnitt 7.2), die Evaluierung des zugehörigen Analyseprojekts (Abschnitt 7.3) sowie die Erfahrungssicherung und Prozessverbesserung (Abschnitt 7.4). Es folgt eine kurze Zusammenfassung in Abschnitt 7.5.

7.1 Prozessrevision als Kontroll- und Gestaltungsaufgabe

7.1.1 Der Revisionsbegriff

Gemäß dem Regelkreismodell des Datenanalyseprozessmanagements aus Abschnitt 2.4.4 folgt der Prozesssteuerung und -durchführung eine umfassende Kontrolle, die Analyseergebnisse, Prozessablauf sowie Entscheidungs- und Handlungskonsequenzen einschließt, und aus der Empfehlungen zur Planung künftiger Analysen hervorgehen. Die Kontrolle ist zunächst als Lenkungsaufgabe einzuordnen (vgl. Abschnitt 2.4.3.2). Vor dem Hintergrund des in dieser Arbeit hervorgehobenen Potenzials der Wiederverwendung von Planungsartefakten und Erkenntnissen erlangt die Erfahrungssicherung, die auch über Verbesserungen und Änderungen an den zu speichernden Artefakten und Wissenselementen zu befinden hat, besonderen Stellenwert. Aus diesem Grund wird die Kontrollphase des Prozessmanagements im Folgenden mit dem Begriff ***Revision*** bezeichnet, der neben einer genauen Überprüfung auch die Korrektur und Modifikation des Revisionsgegenstands abdeckt [Kien82, 390], [Dude17d]. Die Prozessrevision hat damit sowohl Kontroll- als auch Gestaltungscharakter. Der nächste Abschnitt (7.1.2) erläutert die aus dieser Überlegung resultierenden Ziele und Kriterien der Revision von Datenanalyseprozessen, bevor Abschnitt 7.1.3 die zugehörigen Aufgaben und die Vorgehensweise darstellt.

7.1.2 Ziele und allgemeine Kriterien der Revision

Die Diskussion zum Revisionsbegriff verdeutlicht, dass die Ziele der Revision einerseits in der *Kontrolle der Datenanalyseprozesse*, andererseits in der *Erfahrungssicherung für künftige Projekte* bestehen. Die Kontrolle im Sinne eines Vergleichs zwischen normativen Vorgaben (Soll) und empirischen Realisationen (Ist) (vgl. [Kuhn90, 55]) dient in erster Linie der *Feststellung des Zielerreichungsgrades* [Wild74, 44]. Abweichungen von den gesetzten Zielen können Anlass zu genaueren Untersuchungen geben, um mögliche Ursachen aufzudecken und gezielte *Modifikationen der Prozesspläne* vorzunehmen, damit in Zukunft verbesserte Prozessvorlagen eingesetzt werden können. Analog ist die Untersuchung erfolgreicher Prozesse geeignet, um Erkenntnisse und Richtlinien zur Analyseplanung und -durchführung zu gewinnen.

Die Revision bedient sich allgemeiner Kriterien, die aus den Zielen des Prozessmanagements resultieren (vgl. Abschnitt 2.4.2). Im Zentrum der Betrachtung stehen die *Analyseergebnisse*. Sie stellen die Leistung des *Prozesses* dar (Ergebnisparameter), der im Hinblick auf Qualitäts-, Zeit- und Kosteneigenschaften (Prozessparameter) zu beurteilen ist. Ergebnis- und Prozessparameter definieren Kontrollstandards (Vergleichsmaßstäbe der Prozesskontrolle), die eine systematische und detaillierte Beurteilung des Zielerreichungsgrads erlauben (vgl. [Wild74, 44]). Sie betreffen die Effektivität und Effizienz des Analyseprozesses. Aus Sicht von Planung und Wiederverwendung ist zusätzlich die Flexibilität der Prozesse zu beurteilen, die sich auf die Anpassbarkeit an veränderte Kontexte und die Anwendbarkeit in ähnlichen Analysefällen richtet.

Die Beurteilung anhand der Ergebnis- und Prozessparameter bezieht sich auf die Konformität des Analyseprozesses mit der Problemspezifikation, d.h. auf die Frage, inwieweit er die in Analyseproblem und Informationsmaßnahme definierten Anforderungen und Restriktionen erfüllt (vgl. Software-Verifikation [Somm01, 427]). Nun ist der Fall denkbar, dass ein Prozess zwar im Einklang mit seiner Spezifikation steht, aber dennoch keinen Nutzen stiftet, weil er keinen *Beitrag zur Lösung des Sachproblems* leistet, in dessen Kontext er zum Einsatz kommt. In diesem Fall sind Mängel in der Problemspezifikation wahrscheinlich.

Daher ist stets auch eine Prüfung auf Übereinstimmung der Prozessleistung mit den Erwartungen des Auftraggebers anzuraten, die unmittelbar mit Bezug auf das Sachproblem erfolgt (vgl. Software-Validierung [Somm01, 427]). Dient die Analyse der Planung oder Durchführung von Handlungsmaßnahmen, sollte im Sinne einer ganzheitlichen Revision geprüft werden, ob der gewünschte Zielzustand des entsprechenden Problemaspekts erreicht wurde, und ob dies effizient, d.h. mit positiver Nutzen-Kosten-Bilanz, geschehen ist. Hierzu sind Evaluationsstudien durchzuführen, die in vielen Fällen Datenanalysen erfordern (vgl. Projektschachtelung, Abschnitt 3.3.3.4).

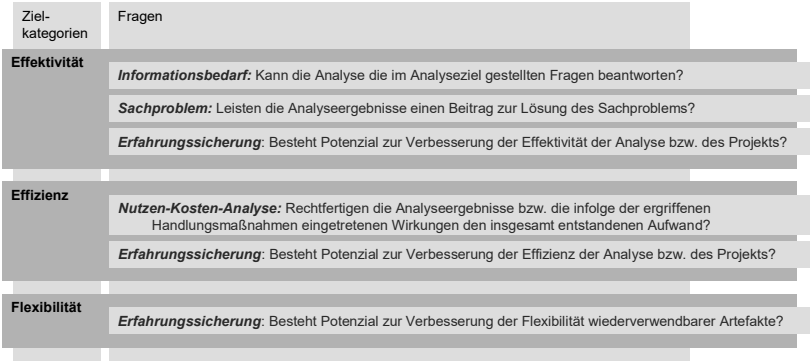


Abbildung 99: Untersuchungsziele der Prozessrevision (eigene Darstellung)

In Abbildung 99 sind wichtige Untersuchungsziele der Prozessrevision als prägnante Fragen formuliert und nach den Zielkategorien des Prozessmanagements gegliedert. Sie sind für jede Revisionsaufgabe sowie für den Einzelfall zu konkretisieren und bei Bedarf zu erweitern.

7.1.3 Aufgaben und Vorgehen bei der Revision von Analyseprozessen

Aus den vorgenannten Zielen lassen sich sechs Aufgaben ableiten, die zur vollständigen Prozessrevision nach dem vorgestellten Verständnis notwendig sind. Aus Effektivitätsperspektive ist die Beurteilung der Analyseergebnisse (Erreichung der Analyseziele) sowie die Evaluation der in ihrer Folge ergriffenen Handlungsmaßnahmen (Erreichung der Anwendungsziele) zu leisten. Die Effizienzperspektive verlangt nach

einer ganzheitlichen ökonomischen Bewertung des Gesamtprojekts, die alle Teilprojekte zur Durchführung von Datenanalysen sowie zur Realisierung der Handlungsmaßnahmen umfasst. Dies bedingt die Ermittlung der Aufwände für alle Analyseprozesse, die jeweils im Rahmen der Beurteilung des Prozessablaufs erfolgt. Diese Aufwände werden um die Kosten der Maßnahmendurchführung ergänzt und in einer globalen Nutzen-Kosten-Analyse den zuvor ermittelten Wirkungen gegenübergestellt. Aus Flexibilitätsperspektive ist der Aspekt der Erfahrungssicherung zu betrachten, der auch im Hinblick auf Effektivität und Effizienz relevant ist. Sie wird durch Extraktion wiederverwendbaren Wissens sowie gezielte Modifikation der Analysepläne realisiert.

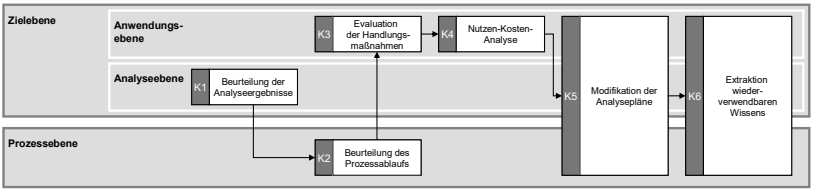


Abbildung 100: Handlungsschema zur Prozessrevision (eigene Darstellung)

Zur Gewährleistung eines möglichst effizienten Ablaufs ist eine Reihung der Aufgaben wie im Handlungsschema in Abbildung 100 empfehlenswert, bei der zunächst einzelne Analysen und anschließend das Analyseprojekt als Ganzes revidiert werden. In allen Schritten stehen jeweils die Ergebnisse vorgelagerter Aufgaben zur Verfügung.

Die Revision startet mit der *Beurteilung der Analyseergebnisse (K1)*, gefolgt von der *Beurteilung des Prozessablaufs (K2)*. Diese beiden Aufgaben sind für jeden Analyseprozess auszuführen und werden als ***Prozessrevision i.e.S.*** bezeichnet. Sie sind auf Analyse- bzw. Prozessebene der Analysearchitektur verortet und können bei Untersuchungen, die nicht im Rahmen eines umfangreichen Sachprojekts stattfinden, hinreichend sein. In der Regel ist die Einbeziehung der Anwendungsebene sinnvoll. Dies geschieht nach Anwendung des Wissens mit der *Evaluation der Handlungsmaßnahmen (K3)* und der *Nutzen-Kosten-Analyse (K4)*. Letztere greift auf die Ergebnisse von K2 und K3 zurück und kann daher nicht isoliert erfolgen. Die Aufgaben K1 bis K4 bilden

zusammen die ***Prozessrevision i.w.S.*** Sie ist hinreichend, falls aus dem Analyseprojekt (außer den Analyseergebnissen sowie subjektiven Erfahrungen der Beteiligten) keine intersubjektiv verwertbaren Erfahrungen gezogen werden sollen.

Da diese Erfahrungssicherung eine systematische Wiederverwendung von Prozessartefakten erst ermöglicht, schließen sich bei der ***lernenden Prozessrevision*** zwei weitere Aufgaben an.[225] Die *Modifikation der Analysepläne (K5)* setzt auf den Erkenntnissen aller vorangehenden Aufgaben auf, um verbesserte Pläne zur Verwendung in künftigen Projekten bereitzustellen. Sie überspannt im Idealfall alle Ebenen der Analysearchitektur und behandelt Analyseprozesse, Analyseketten und Problemkarten. Zuletzt erfolgt die *Extraktion wiederverwendbaren Wissens (K6)*, um aus dem Prozessablauf Hinweise, Regeln und Vorlagen abzuleiten, die bei der Konzipierung und Durchführung von Datenanalysen Unterstützung bieten. Auch sie betrifft alle Architekturebenen.

Die weitere Gliederung dieses Kapitels orientiert sich an der Reichweite der Revision. So behandelt Abschnitt 7.2 die Prozessrevision i.e.S., Abschnitt 7.3 ergänzt die Aspekte der Prozessrevision i.w.S., und Abschnitt 7.4 widmet sich den Aspekten der lernenden Revision.

7.2 Beurteilung der durchgeführten Datenanalyse

Die *Prozessrevision i.e.S.* untersucht im Sinne der Prozessleistungstransparenz und der Prozessstrukturtransparenz des Prozessmanagements Ergebnisse und Konzeption der Analyseprozesse und leistet daher eine umfassende Beurteilung der durchgeführten Datenanalysen. Angesichts der Abhängigkeit der Prozessleistung von der Prozessstruktur sind beide Aspekte im Zusammenhang zu betrachten. Folglich findet zuerst die Bewertung der Analyseergebnisse statt, um die dabei gewonnenen Erkenntnisse danach zur Fokussierung der Prozessbewertung zu nutzen.

[225] Auch die Prozessrevision i.e.S. erlangt durch Hinzunahme der Aufgaben K5 und K6 lernenden Charakter.

7.2.1 Beurteilung der Analyseergebnisse (K1)

Diese Aufgabe findet unmittelbar im Anschluss an die Prozessphase Ergebnisaufbereitung statt und wird zum Teil auch als deren Bestandteil angesehen (vgl. hierzu die Prozessmodelle in Abschnitt 3.1.1). Eine klare Trennung gelingt, indem der Zweck von Analyseprozessen auf die Produktion von Analyseergebnissen beschränkt wird. Entscheidungen über die Qualität und Nutzung der Ergebnisse fallen demnach der Managementaufgabe Revision zu. Aus dieser Sicht ist im Rahmen der Ergebnisbeurteilung (Interpretation) auch über Möglichkeiten und Formen der Anwendung des analytisch erlangten Wissens zur Lösung des Sachproblems zu befinden bzw. eine entsprechende Entscheidung durch den Auftraggeber zu initiieren.

Diese Aufgabe untersucht die Ergebnisparameter des Analyseprozesses mit dem Ziel festzustellen, inwieweit die Analyseergebnisse den im Anwendungs- und Analysekontext entstandenen Informationsbedarf befriedigen. Ihr Ergebnis sind Gütemaße, die in Schritt K2.1 zur Abschätzung des Zielerreichungsgrads der Analyse dienen, sowie inhaltliche Erkenntnisse, die auf Anwendungsebene zur Entwicklung von Lösungsoptionen für das Sachproblem beitragen.

Interessantheit von Analyseergebnissen

Da zahlreiche Kriterien in die Ergebnisbewertung einfließen, wird zunächst ein Orientierungsrahmen präsentiert, der auf dem für das Data Mining entwickelten Konstrukt ***Interessantheit*** beruht. Es umfasst vier allgemeine Gütekriterien, denen alle Analyseergebnisse genügen sollen (vgl. hierzu [FaPS96, 6-9] und [Knob01, 74-76]):

- **Gültigkeit**: Die Aussagen sollen die relevanten Aspekte der untersuchten Datenbasis (beschreibende Analysen) bzw. die zugehörigen Sachverhalte in der Zielpopulation (schließende Analysen) mit ausreichender Genauigkeit und Sicherheit repräsentieren.
- **Neuartigkeit**: Die Aussagen sollen inhaltlich neu sein, da bereits bekannte Sachverhalte keine Information darstellen. Die Interessantheit einer Aussage in Abhängigkeit von ihrer Unerwartetheit

korrespondiert mit dem Informationsgehalt nach SHANNON [Lyre02, 19, 32].

- **Potenzielle Nützlichkeit**: Die Aussagen sollen zur Lösung eines Sachproblems beitragen bzw. in zielführende Handlungsmaßnahmen überführbar sein.
- **Verständlichkeit**: Die Aussagen sollen inhaltlich und formal für den Empfänger verständlich präsentiert werden, um eine sachgerechte Interpretation und Nutzung zu erlauben.

Das Konstrukt Interessantheit ist inhärent subjektiv [SiTu95, 275f.]. So werden verschiedene Personen die Interessantheit derselben Aussage aufgrund ihres individuell unterschiedlichen Vorwissens gänzlich anders einschätzen [Küpp99, 88]. Bezogen auf Einzelkriterien sind objektive Gültigkeitsmaße, wie etwa die geschätzte Vorhersagegenauigkeit eines Prognosemodells, oft relativ einfach zu berechnen. Auch die Nützlichkeit lässt sich häufig monetär quantifizieren, sofern geeignete Bewertungsgrößen verfügbar sind [FaPS96b, 83]. Verständlichkeit und Neuartigkeit sind weitaus stärker subjektiv. Offensichtlich sind Interessantheitsmaße nicht allein auf Grundlage von Analysedaten und -ergebnissen zu bestimmen, weshalb die Implementierung automatisierter Interessantheitsfilter (vgl. [SiTu95, 275f.]) problematisch ist.

Die vier Gütekriterien sind notwendige Bedingungen, damit ein Analyseergebnis für einen Entscheidungsträger in einer Problemsituation Informationscharakter besitzt.[226] In logischer Abfolge organisiert, bilden sie ein Grundschema für die Bewertung von Analyseergebnissen, das im Sinne eines sich stetig verjüngenden Trichters aus der vom Analyseverfahren gelieferten, initialen Ergebnismenge in vier Stufen sukzessive Aussagen ausfiltert, die nicht zur Lösung des Sachproblems beitragen können (Abbildung 101).

[226] Mit Ausnahme der Gültigkeit sind die Kriterien bereits durch den verwendeten Informationsbegriff (Abschnitt 2.1.2.2) abgedeckt, da unverständliche, redundante oder nicht nützliche Nachrichten den Empfänger nicht auf eine höhere semantische Ebene heben und somit keine Information darstellen.

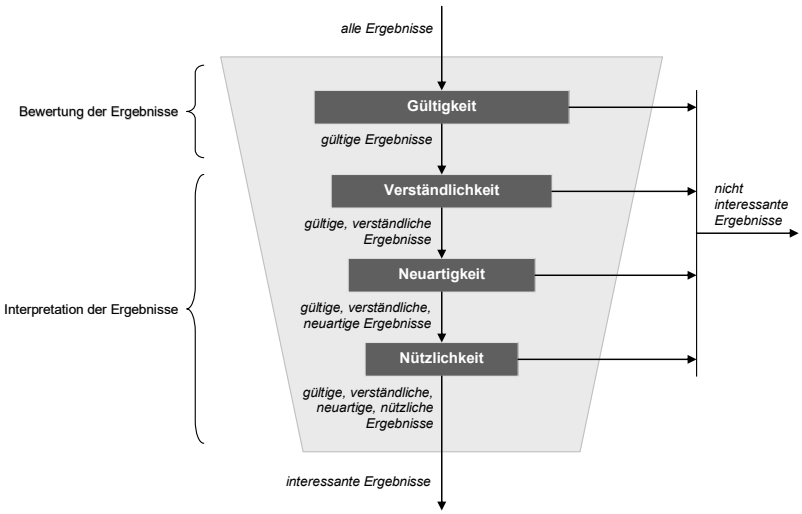


Abbildung 101: Gestufte Filterung von Analyseergebnissen nach Interessantheit (eigene Darstellung)

Auf erster Stufe wird die Gültigkeit der Ergebnisse geprüft, da Entscheidungen auf Grundlage nicht valider Aussagen in jedem Falle zu vermeiden sind. Die *Bewertung der Gültigkeit von Analyseergebnissen (K1.1)* erfolgt in Abhängigkeit vom Basisansatz der Datenanalyse (vgl. Abschnitt 2.2.1) nach unterschiedlichen Gesichtspunkten, die in Abschnitt 7.2.1.1 erläutert werden. Gültige Ergebnisse sind auf zweiter Stufe auf Verständlichkeit zu prüfen. Nur inhaltlich verständliche Aussagen können auf dritter Stufe daraufhin untersucht werden, ob sie die Bedingung der Neuartigkeit erfüllen. Nur gültige, verständliche und neuartige Ergebnisse gelangen schließlich auf die vierte Stufe, in der als letzte Bedingung die Nützlichkeit der Aussage zu überprüfen ist. Die Beurteilung der Stufen zwei bis vier erfordert profunde fachliche Domänenkenntnisse und erfolgt aus diesem Grunde im Rahmen einer integrierten *Interpretation von Analyseergebnissen (K1.2)*, wie sie Abschnitt 7.2.1.2 vorstellt.

7.2.1.1 Bewertung der Gültigkeit von Analyseergebnissen (K1.1)

Die Bewertung der Gültigkeit erfordert eine detaillierte Sichtung der Ergebnisse durch den Analytiker, der hierbei insbesondere analysemethodische Kriterien anlegt, aber auch fachliche Erwägungen trifft (vgl. [CCK+00, 29]). ***Gültigkeit*** eines Analyseergebnisses ist im Allgemeinen dann gegeben, wenn es valide Aussagen über das Untersuchungsobjekt liefert. Dies ist erreicht, wenn folgende Bedingungen erfüllt sind (vgl. Kriterien des Informationsbedarfsprofils, Anhang A5.1 sowie [BeLi97, 97]):

- **Genauigkeit**: Die Aussagen bilden die tatsächlichen Eigenschaften des Untersuchungsobjekts möglichst exakt und korrekt ab.
- **Sicherheit**: Die Aussagen treffen mit großer Sicherheit (z.B. Wahrscheinlichkeit) zu.
- **Zuverlässigkeit**: Die Aussagen sind mittels verlässlicher, verifizierter Methoden aus glaubwürdigen Daten abgeleitet.
- **Überprüfbarkeit**: Die Aussagen sind hinsichtlich ihrer Wahrheit verifizierbar, z.B. anhand der Analysedaten oder anderer Datenbestände.

Zuverlässigkeit und Überprüfbarkeit sind bei fachgerechter Analyseplanung und -ausführung als gegeben zu betrachten und nur qualitativ zu beurteilen. Sie werden bei der Beurteilung des Prozessablaufs (K2, Abschnitt 7.2.2) berücksichtigt. Genauigkeit und Sicherheit lassen sich für die Ergebnismenge konkreter Analysen *quantitativ bewerten*. Welche Gütemaße hierfür geeignet sind, ist vom jeweiligen Kontext abhängig. Methodenspezifische `Bewertungskriterien` sind nur für eine Klasse von Verfahren geeignet und häufig in der Werkzeugdokumentation oder einschlägiger Literatur nachzulesen [HKMW01]. Sie können am Operator hinterlegt und im Rahmen der Spezifikation der Analyseaufgabe (P1.5, Abschnitt 5.5.3.5) für den konkreten Fall ausgewählt werden. Methodenunabhängige Kriterien sind häufig allgemein gefasst und erfordern vor ihrer Anwendung eine Operationalisierung in konkrete Maße [HiWi01, 73].

Im Folgenden werden Hinweise zur Operationalisierung der *Genauigkeit* und *Sicherheit* diskutiert sowie einige Beispiele für konkrete Maßzahlen angegeben. Die Darstellung erfolgt getrennt für beschreibende, konfirmatorische und schließende Analysen. Für die Modellerstellung (Abschnitt 3.1.2.3) können Kriterien aller drei Kategorien hilfreich sein. Insbesondere sind die Kriterien für beschreibende Analysen prinzipiell auch für konfirmatorische Analysen relevant, da letztere stets deskriptive Untersuchungen darstellen (vgl. Abschnitt 2.2.1.3).

Bewertung der Ergebnisse beschreibender Analysen

Bezüglich der Gültigkeit von Beschreibungsmodellen verweist die Literatur meist auf domänenspezifische Abhängigkeiten der Gütekriterien (z.B. bei [BeST99, 225]). Dennoch lassen sich allgemeingültige Richtlinien formulieren. Ein mittels exakter Verfahren produziertes Ergebnis erfüllt die Forderung der **Genauigkeit** genau dann, wenn es nach Maßgabe des zugrunde liegenden analytischen Modells korrekt ist, d.h., wenn z.B. eine Kennzahl im Berichtswesen gemäß der zugehörigen Formel richtig berechnet wird. Die Feststellung der *Korrektheit* geschieht durch Überprüfung anhand der analysierten Daten („Nachrechnen“).[227] Nicht exakte Lösungsverfahren nähern sich der exakten Lösung an, ohne diese tatsächlich zu erreichen. Ist die Distanz zur exakten Lösung bekannt (approximierende Verfahren [FeSi13, 106]), kann die *Approximationsgenauigkeit* berechnet werden, z.B. als Anteil der durch das Modell beschriebenen Merkmalsvarianz an der Gesamtvarianz [HiWi01, 66].[228] Die Approximationsgenauigkeit ist als realisierter Ausschöpfungsgrad des Informationsgehalts der Analysedaten interpretierbar [HiWi01, 76f.]. Anstatt die Genauigkeit zu schätzen, kommen bei Unbekanntheit der exakten Lösung üblicherweise *heuristische Gütemaße* zum Einsatz. Sie beruhen häufig auf Abstandsmaßen, Häufigkeiten oder Wahrscheinlichkeiten. So nutzt die Clusteranalyse

[227] Da beschreibende Analysen auf die Abbildung der analysierten Daten zielen, muss die Überprüfung zwingend auf denselben Daten erfolgen.

[228] So ist etwa das Bestimmtheitsmaß R^2 der linearen Regression unmittelbar als Anpassungsgrad einer Regressionsgeraden an die Punktwolke interpretierbar [Schi03, 111, 136f.].

z.B. Abstands- und Ähnlichkeitsmaße zur Beurteilung der Güte einer Objekteinteilung [Grab01, 318].

Bei Mustersystemen (z.B. Regeln) ist die Abweichung von der exakten Lösung nicht von Interesse, da eine solche für die Gesamtheit aller Aussagen nicht existiert (vgl. [BeLi97, 106]). Hier wird der Ausschöpfungsgrad des Informationsgehalts z.B. durch den Anteil der nicht in die Ergebnismenge eingeflossenen Fälle gemessen. Sein Komplement ergibt ein Maß der *Vollständigkeit*, mit der die in den Daten abgebildeten Fälle im Mustersystem berücksichtigt sind.

Schließlich ist stets auch eine *visuelle Inspektion* der Ergebnisse hilfreich, da sich bestimmte strukturelle Eigenschaften von Modellen oft prägnant in grafischen Darstellungen manifestieren. So wird z.B. in der Zeitreihenanalyse die Ergebnisbewertung wirksam durch Diagramme und Residuenplots unterstützt, um etwa Instationaritätsmuster, Strukturbrüche und andere Phänomene zu identifizieren, welche sich negativ auf die Gültigkeit auswirken [KüBe01, 227, 260].

Die Bewertung der Gültigkeit von Analyseergebnissen schließt stets die Berücksichtigung ihres **Sicherheitsgrades** ein, um beurteilen zu können, wie viel Vertrauen der Analytiker in die Ergebnisse setzen kann [BeLi97, 97], [HiWi01, 82]. Im Anwendungsbereich beschreibender Analysen eignen sich hierfür insbesondere auf *Häufigkeiten* beruhende heuristische Maße. Sie zeigen, wie viele Fälle ein gefundenes Muster stützen, und helfen somit bei der Erkennung von Aussagen, die aufgrund zu geringer empirischer Fundierung als nicht gültig anzusehen sind. Bei vielen beschreibenden Analysen repräsentiert das Analyseergebnis die Gesamtdatenmenge. So werden z.B. Controlling-Berichte höchster Aggregationsebene stets auf Basis aller relevanten Datensätze berechnet. Sobald eine Aussage nur auf einer Teildatenmenge beruht, ist die Kenntnis des *Umfangs dieser Teildatenmenge* in aller Regel von großem Interesse. So wird z.B. ein Umsatzbericht auf Produktlinienebene informativer, wenn die Anzahl der zugehörigen Aufträge bekannt ist. Entscheidungsbäume geben typischerweise für jeden Knoten Anzahl oder Anteil der enthaltenen Fälle an. Bei der Assoziationsanalyse werden spezielle Maße auf Basis relativer bzw. bedingter Häufigkeiten (Support bzw. Confidence) berechnet, um die

Bedeutung einzelner Regeln fassbar zu machen [BeLi97, 131f.], [HeHi01, 445].

Häufig ist nicht der absolute Sicherheitsgrad, sondern die *relative Abweichung von der Erwartung* oder die Verbesserung im Vergleich mit anderen Ergebnissen von Interesse [HiWi01, 82]. Dieses Kriterium wird in seiner Grundform als ***Lift*** bezeichnet[229] und quantifiziert die Abweichung der Häufigkeit eines Merkmalswerts in einem Modell, einem Muster oder einer Stichprobe im Vergleich zur Gesamtpopulation bzw. zum Zufall [BeLi97, 107], [BeST99, 226]. Für Entscheidungsbäume misst der Lift, um welchen Grad sich ein Zielmerkmalswert stärker in einer Partition konzentriert als erwartet und verweist damit auf besonders prägnante Muster [BeST99, 226]. Für Assoziationsregeln der Form A→B beschreibt der Lift, um welchen Faktor Element B in den Transaktionen mit Element A häufiger vorkommt als in der Gesamtheit aller Transaktionen [HeHi01, 447]. Maße dieser Klasse bewerten die Unerwartetheit einer Aussage.

Bewertung der Ergebnisse konfirmatorischer Analysen

Konfirmatorische Datenanalysen zielen auf die Verifikation von Annahmen oder Hypothesen, die als Analysefrage formuliert sind. Grundsätzlich wird die **Gültigkeit** einer Aussage durch Abruf geeigneter Daten über den betroffenen Sachverhalt überprüft („Nachschlagen“ in den Daten). Stützen z.B. die Vertriebsdaten die Vermutung, dass in Vertreterbezirk V seit zwei Monaten kein Großkundenauftrag mehr abgeschlossen wurde, so ist diese als bestätigt zu akzeptieren. Derartige Situationen stellen in der betrieblichen Praxis den häufigsten Anwendungsfall hypothesengetriebener Analysen dar. In vielen Fällen sind Annahmen über Abweichungen, Unterschiede, Veränderungen oder Zusammenhänge zu überprüfen, die sich auf mehr als ein Untersuchungsobjekt richten. Beispielsweise zielt die Analysefrage H_1 „Sind die Umsatzzahlen im Vertreterbezirk V erheblich niedriger als der Mittelwert aller Bezirke?“ auf einen Realnormvergleich hinsichtlich

[229] Lift ist definiert als p(Merkmal$_t$ | Stichprobe) / p(Merkmal$_t$ | Population) [BeLi97, 107].

einer *statistischen Kenngröße* (Mittelwert). Auch hier lässt sich das Zutreffen der Annahme vordergründig durch Datenabruf kontrollieren.

Aufgrund der Tatsache, dass diese Analyse mehrere Untersuchungsobjekte betrifft, können die zugehörigen Daten als Realisation eines Zufallsexperiments interpretiert werden. Daher besteht keine **Sicherheit**, dass die beobachteten Phänomene nicht nur zufällig aufgetreten sind, sondern es sich um tatsächlich relevante Effekte handelt [HiWi01, 73]. In solchen Situationen können probabilistisch begründete Entscheidungen über das Zutreffen von Hypothesen mithilfe der *statistischen Testtheorie* gefällt werden [GoJä09, 191]. Die Logik der statistischen Hypothesentestung entspricht dem Beweis durch Widerspruch: Hierzu wird in der Regel das Komplement der Forschungshypothese H_1 als so genannte Nullhypothese H_0 formuliert [Cohe96, 12f.]. Diese behauptet, dass die betreffende Kenngröße den Wert 0 hat und dass eine eventuelle Abweichung von 0 ausschließlich durch Zufallsschwankungen verursacht wird. Die obige Analysefrage H_1 geht von einem Unterschied zwischen einem Einzel- und einem Mittelwert aus. Eine dazu passende Nullhypothese H_0 lautet: „Die Umsatzzahlen im Vertreterbezirk V sind genauso hoch wie der Mittelwert aller Bezirke“. Anschließend wird ein Hypothesentestverfahren ausgeführt um zu untersuchen, mit welcher Wahrscheinlichkeit p diese empirische Aussage rein zufällig auftreten kann [GoJä09, 191]. Ist dies angesichts zu geringer Wahrscheinlichkeit nicht zu erwarten, kann die in der Analysefrage implizite Forschungshypothese akzeptiert werden [Cohe96, 12f.], [Zöfe03, 90]. Um die Wahrscheinlichkeit einer Fehlentscheidung zu minimieren, ist vom Analytiker ein Signifikanzniveau α festzulegen, mit welchem er die Irrtumswahrscheinlichkeit vorgibt, die er bei Ablehnung von H_0 zu akzeptieren bereit ist. Ein Analyseergebnis ist demnach als *signifikant (statistisch bedeutsam)* zu werten, wenn $p < \alpha$ [Sedl96, 43].[230]

Da die Anwendbarkeit eines Hypothesentests u.a. von Aussagetyp und Eingabedaten abhängt, können geeignete Testverfahren als `Evalua-`

[230] Die Ablehnung oder Akzeptanz von H_0 bzw. H_1 unterliegt in der Praxis weiteren Erwägungen über die Prüfung der Relation von p und α hinaus. Zur Vertiefung vgl. z.B. [Sedl96], [GoJä09, 191f.], [Zöfe03, 95].

`tionsansatz` am Analyseoperator hinterlegt werden. Eine Tabelle ausgewählter Verfahren enthält Anhang A7.1. Alternativ können Kontextregeln zur Auswahl spezifiziert werden. Dies ist insbesondere zur Wahl des Signifikanzniveaus zu empfehlen, das idealerweise in Abhängigkeit von Datencharakteristika bestimmt wird.[231]

Fehlentscheidungen bei Signifikanztests können zweierlei Gestalt annehmen. Wird H_0 irrtümlich abgelehnt, obwohl die empirischen Beobachtungen tatsächlich auf rein zufälligen Artefakten beruhen, liegt ein Fehler 1. Art (α-Fehler) vor. Wird hingehen H_1 irrtümlich abgelehnt, obwohl die empirischen Aussagen tatsächlich auf realen Effekten beruhen, wird ein Fehler 2. Art (β-Fehler) begangen [Sedl96, 43], [GoJä09, 192] (vgl. Anhang A7.1). Abhängig vom jeweiligen Sachproblem kann die eine oder die andere Form riskanter sein [Zöfe03, 99]. Der α-Fehler wird auch als *Konsumentenrisiko* bezeichnet. Nimmt z.B. ein Pharmahersteller irrtümlich die Wirksamkeit eines faktisch wirkungslosen neuen Medikaments an, so geht das Risiko zulasten des Patienten (Konsument). Geht er hingegen irrtümlich davon aus, dass keine Wirkung vorliegt und bringt das tatsächlich heilsame Mittel nicht auf den Markt, so trägt er als Produzent Risiko und Kosten. Entsprechend heißt der β-Fehler auch *Produzentenrisiko* [Zöfe03, 94f.]. Nicht immer ist diese Interpretation auf andere Domänen übertragbar. Dennoch besteht meist eine Präferenz für einen der Fälle, die sich in den `Präferenzrelationen` der Analyseaufgabe dokumentieren lässt und zur korrekten fachlichen Beurteilung der Ergebnisse statistischer Hypothesentests beiträgt.

---

[231] Die Signifikanz einer empirischen Aussage steigt im Allgemeinen mit wachsendem Stichprobenumfang und abnehmender Stichprobenvarianz [Cohe96, 13], [GoJä09, 53]. Somit führt die Ausweitung der Stichprobe bei gleichbleibendem Signifikanzniveau ab einem bestimmten Umfang regelmäßig zu signifikanten Ergebnissen. Zur Vermeidung inkonsistenter Hypothesentests muss also der α-Wert mit wachsender Analysedatenmenge angemessen gesenkt werden [GMPS97, 13]. In der Praxis wird α jedoch meist standardmäßig mit 0,05 oder 0,01 angegeben, obwohl diese Richtwerte auf FISHERs Untersuchungen mit sehr geringen Fallzahlen zurückgehen [GMPS97, 23]. Daher sollten die Anforderungen an die Signifikanz grundsätzlich hoch angesetzt [HiWi01, 73] und gegebenenfalls an das analysierte Datenvolumen angepasst werden. Die korrekte Wahl des Signifikanzniveaus kann mittels BAYES-Faktoren erfolgen [GMPS97, 23f.].

Die Aussagekraft der statistischen Signifikanz wird durch mehrere Unzulänglichkeiten beeinträchtigt und in der Praxis häufig überschätzt. Neben Schwierigkeiten bei der Interpretation der Irrtumswahrscheinlichkeit p, die keine Aussagen zum Wahrheitsgehalt einer Hypothese macht, ist einerseits die Abhängigkeit der Signifikanz vom Umfang der Stichprobe problematisch. Dadurch werden mit wachsenden Stichproben zunehmend kleinere Effekte als signifikant charakterisiert, die aus sachlogischer Sicht oft nicht interessieren [HiWi01, 73]. Andererseits trifft sie keine Aussage zur Stärke eines Effekts (vgl. [Sedl96, 43]). Neben der statistischen Bedeutsamkeit (Signifikanz) sollte daher stets auch die *praktische Bedeutsamkeit* einer empirischen Aussage beurteilt werden [GoJä09, 53f.]. Letztlich ist ein Testergebnis umso sicherer, je stärker der beobachtete Effekt (je größer z.B. eine gemessene Abweichung) ist. Hierzu eigenen sich spezielle Maße der *Effektstärke*. Einen Katalog geeigneter Maße für gängige Hypothesentests präsentiert z.B. COHEN, für den t-Test etwa COHENs *d* [Cohe88]. Die Interpretation der Effektstärke erfolgt stets kontextabhängig. So mag z.B. ein um 1% verbesserter Therapieerfolg einer schweren Krankheit ein bedeutsames Ergebnis darstellen, während eine durch Unternehmensreorganisation erzielte Kosteneinsparung von 1% deren Aufwand möglicherweise nicht rechtfertigt [GoJä09, 54].

Bewertung der Ergebnisse schließender Analysen

Schließende Datenanalysen treffen Aussagen über Objekte oder Ereignisse, die nicht in den ausgewerteten Daten abgebildet sind, indem sie von Stichproben- oder historischen Daten auf die Gesamtpopulation bzw. künftige Sachverhalte schlussfolgern. Typische Beispiele sind Meinungs- oder Kundenumfragen, Klassifikations- oder Schätzmodelle. Sie sind meist als Prognosen interpretierbar (vgl. Abschnitt 2.2.1.2). Werden Prognosemodelle operativ eingesetzt, ist die regelmäßige Kontrolle ihrer Leistungsfähigkeit Bestandteil der Modellbewertung, um festzustellen, wann sie aktualisiert oder ersetzt werden müssen [BeLi00, 186] (*Modellwartung*).

Prognosemodelle, die alle Instanzen fehlerfrei klassifizieren oder schätzen, sind nicht realistisch. Ihre Bewertung richtet sich daher

darauf, wie nahe sie dem Ideal des perfekten Prognosemodells kommen [KrBü11, 41], [BeLi00, 183]. Hierzu werden die Modellvorhersagen mit bekannten Realisationen der Zielmerkmale verglichen [DeHa01, 233] und bestimmt, wie exakt sie zutreffen. Die **Genauigkeit** wird z.B. bei kategorialen Merkmalen durch den Anteil der Fehlklassifikationen, bei metrischen Merkmalen durch die mittlere Abweichung vom wahren Wert gemessen. Die Bewertung ist somit nur retrospektiv möglich, nachdem die betreffenden Ereignisse stattgefunden haben und empirische Daten dazu vorliegen [HiWi01, 66]. Dies ist insofern problematisch, als bei Prognosemodellen nicht die Gültigkeit bezüglich der Trainingsdaten von Interesse ist, sondern die Performanz auf Anwendungsdaten. Die auf Trainingsdaten gemessene Fehlerrate (Resubstitutionsfehler) ist grundsätzlich zu optimistisch, um eine valide Schätzung der Fehlerrate bezüglich neuer Anwendungsdaten abzugeben. Um dennoch vor Einsatz des Modells eine verlässliche Abschätzung der zu erwartenden Genauigkeit zu erhalten, sind *Experimente anhand spezieller Evaluierungsdaten* auszuführen [WiFr00, 121] (vgl. Abschnitt 3.1.2.3).

Derlei Evaluierungsdaten sollten wie die Trainingsdaten repräsentative Stichproben der Problemdomäne darstellen, aber sowohl von diesen als auch von den zur Modelloptimierung verwendeten Kalibrierungsdaten disjunkt sein.[232] Hierin liegt häufig eine Schwierigkeit, denn qualitativ geeignete Daten sind selbst in Anwendungen mit großen Datenmengen oft rar [WiFr00, 119-121]. So sind in der Regel nur wenige Beispieldaten zu Betrugsdelikten verfügbar, aber auch Kreditausfälle oder im Direktmarketing Antwortende sind meist stark unterrepräsentiert. Zur Lösung dieser Schwierigkeiten haben sich verschiedene ***Evaluierungsstrategien*** etabliert. Der einfachste Fall, bei dem ein Teil der Beispieldaten als Evaluierungsset zurückgehalten und der Rest zum Training eingesetzt wird, heißt blindes Testen oder Holdout-Methode. Sind nur wenige Beispieldaten verfügbar, kann die n-fache Kreuzvalidierung zum Einsatz

[232] Die Begriffsverwendung in der Literatur ist nicht eindeutig. Evaluierungsdaten ([BeLi97], [BeLi00], [HiWi01]) werden auch als Testdaten bezeichnet ([WiFr00], [DeHa01]), während andere Autoren Testdaten mit Kalibrierungsdaten („Validierungsdaten“) gleichsetzen ([BeLi97], [WiFr00], [HiWi01]).

gelangen. Sie ermöglicht einen Rollentausch der Datensets unter statistischen Kriterien. Die Beispieldaten werden dazu in n annähernd gleichgroße Partitionen geteilt, und in n Durchgängen der Modellerstellung und -evaluierung jeweils eine der Partitionen als Evaluierungsdaten, der Rest als Trainingsdaten herangezogen. Die Fehlerrate ergibt sich aus dem Durchschnitt der Fehlerraten aller n Durchgänge[233] [WiFr00, 125-127], [DeHa01, 233f.]. Zu weiteren Evaluierungsstrategien vgl. z.B. [WiFr00, 127ff.]. Da die Strategie mit der Aufteilung der Daten auch die Datenvorbereitung betrifft, wird sie während der Prozessspezifikation (P1.5) bestimmt. Zum Teil sind geeignete Ansätze zur Schätzung der Fehlerrate bereits in die Analyseverfahren integriert [BeLi97, 99].

Bei **Klassifikationsaufgaben** ist die Population im Hinblick auf das Zielmerkmal in der Regel nicht ausgewogen, und bestimmte Fehlklassifikationen sind problematischer als andere. Daher ist die Performanz des Gesamtmodells zur Modellbeurteilung oft nicht ausreichend [DeHa01, 234]. Vielmehr sind differenzierte Bewertungen bezüglich einzelner Klassen (Ausprägungen des Zielmerkmals) von Interesse [BeLi97, 98], [LeVo10, 86], die auf Grundlage einer *Klassifikationstabelle* (Confusion Matrix) berechnet werden. Diese Kontingenztabelle stellt die prognostizierten den wahren Klassen gegenüber und weist die Anzahl korrekter und fehlerhafter Vorhersagen aus, wie sie bei Anwendung des Modells auf Evaluationsdaten entstehen [BoAr01, 228-230], [Fawc06, 862]. Aus diesen Häufigkeiten lassen sich z.B. Erfolgsrate und Fehlerrate für das Gesamtmodell sowie Trefferrate (Sensitivität, Recall), Fehlalarmrate (Ausfallrate), Spezifität, Versagensrate, Präzision (Relevanz) oder Trennfähigkeit (Segreganz) für die Bewertung spezifischer Fähigkeiten des Modells ableiten. Eine Klassifikationstabelle mit einer Übersicht über gängige abgeleitete Kenngrößen ist in Anhang A7.2 enthalten.

---

[233] Theoretisch wie praktisch empfehlenswert ist die Variante der 10-fachen geschichteten Kreuzvalidierung [WiFr00, 126f.]. Das blinde Testen ist ein Sonderfall mit n=2 [DeHa01, 234].

Analog zu den Überlegungen zur Präferabilität von α- oder β-Fehler bei konfirmatorischen Analysen ist auch hier kontextabhängig zu entscheiden, welche Art von Fehlklassifikation das größte Risiko birgt bzw. welche Art der korrekten Zuordnung den größten Nutzen verspricht. Beispielsweise wird bei der Betrugserkennung ein minimaler Anteil nicht erkannter Betrugsdelikte (Versagensrate) und ein hoher Anteil korrekt erkannter Betrugsfälle (Trefferrate) angestrebt, während der Verlust durch einige zu Unrecht zurückgewiesene Fälle (Fehlalarmrate) verkraftbar erscheint. Im Direktmarketing hingegen ist die Versagensrate weniger von Bedeutung, da nicht erkannte Interessenten zwar ärgerlich, jedoch nicht riskant sind. Die an der Analyseaufgabe hinterlegten `Präferenzrelationen` könnten daher im ersten Fall lauten „`Versagensrate(min) > Trefferrate(max) > Fehlalarmrate(min)`“, im zweiten Fall wäre „`Trefferrate(max) > Fehlalarmrate(min)`“ angebracht. Die Präferenzen verweisen auf zu berechnende Genauigkeitsmaße und können in die Bewertung der Nützlichkeit (K1.2) eingehen.

Neben diesen methodenunabhängigen Bewertungskriterien existieren zahlreiche weitere Genauigkeitsmaße, die nur für bestimmte Analyseverfahren gültig sind. Ihre korrekte Auslegung erfordert oft profunde Kenntnisse. Für Entscheidungsbäume kann die Güte der Einteilung z.B. mittels Klassenentropie, mittlerem Informationsgewinn oder Gini-Index beurteilt werden, welche die Gleichmäßigkeit der Klasseneinteilung beschreiben [BeLi97, 254f.], [EnTh98, 434].

Bei **Schätzaufgaben** (numerischen Vorhersagen) interessiert nicht allein die Existenz oder Absenz von Fehlern, sondern auch deren Ausmaß. Der *Prognosefehler* für eine Einzelaussage ist im Allgemeinen die Differenz zwischen prognostiziertem und beobachtetem Wert, kann aber auch als Quotient oder prozentuale Abweichung quantifiziert werden [WiFr00, 148], [KüBe01, 289], [HiWi01, 77]. Zur Betrachtung des Gesamtmodells sind die Prognosefehler über alle Einzelaussagen zu aggregieren. Bei der Summation lässt sich die Kompensation positiver und negativer Abweichungen durch Verwendung von Absolutbeträgen oder Fehlerquadraten vermeiden. Letztere nehmen zugleich eine

implizite Höhergewichtung starker Abweichungen vor [HiWi01, 77]. Eine Auswahl gängiger Fehlermaße für Schätzer enthält Anhang A7.2.

Distanzbasierte Maße sind nicht in der Lage, *strukturelle Prognosefehler* zu erkennen. Diese entstehen, wenn ein für die Problemstellung ungeeignetes Prognoseverfahren eingesetzt wird, und äußern sich z.B. dadurch, dass das Modell die Zielwerte grundsätzlich über- oder unterschätzt. Sie sind gut mithilfe grafischer Darstellungen oder statistischer Maße wie dem Korrelationskoeffizienten erkennbar, die den Zusammenhang zwischen prognostizierten und wahren Werten untersuchen [WiFr00, 149]. Für einige Verfahrensklassen sind heuristische Kennzahlen verfügbar (wie z.B. das Tracking Signal zur Erkennung von Über- und Unterschätzungen) [Thon05, 75f.].

Zur Einschätzung der **Sicherheit** eigenen sich wiederum Häufigkeiten, Wahrscheinlichkeiten, heuristische Maße und Signifikanztests. Zusätzlich sind Konfidenzintervalle anwendbar. Die als Support bekannte *relative Häufigkeit* einer Entscheidungs- oder Assoziationsregel kann als Abschätzung der Sicherheit interpretiert werden, mit der nach der Antezedenz auch die Konsequenz der Regel folgt [BeLi97, 106]. Die meisten modellbasierten Prognoseverfahren treffen Wahrscheinlichkeitsprognosen, die bestimmten Ereignissen (z.B. Klassifikationen) eine *Wahrscheinlichkeit* beimessen [KrBü11, 40]. In Anwendungen, in denen eine binäre Klassifikation in positive und negative Fälle nicht angebracht erscheint, sind diese Wahrscheinlichkeiten selbst wichtiger Bestandteil der fachlichen Analyseergebnisse. So ist z.B. die Information, dass Kreditnehmer x seinen Kredit mit Wahrscheinlichkeit p_x% zurückzahlen wird, für die Beurteilung seines Kreditantrags von großer Bedeutung. In allen anderen Anwendungen stellen diese Wahrscheinlichkeiten ein intuitives Sicherheitsmaß dar [WiFr00, 133]. Um die Sicherheit eines Modells in seiner Gesamtheit fassbar zu machen, wurden auf Basis von Wahrscheinlichkeiten verschiedene Bewertungskriterien entwickelt. Sie berechnen *Verlustfunktionen*, die jeder Fehleinschätzung vom vorher-

gesagten oder vom tatsächlichen Wert abhängige Strafkosten zumessen und für das gesamte Modell aggregieren.[234]

Manche Modellierungsverfahren sind in der Lage, anhand der Eigenheiten der ihnen zugrunde liegenden Theorie *spezielle Sicherheitsmaße* zu berechnen. Beispielsweise ist für die Klasse der Nearest-Neighbor-Verfahren bekannt, wie viele Nachbarn dieselbe Vorhersage getroffen haben. Allgemein ist das Vertrauen in eine Prognose umso höher einzustufen, je größer die Übereinstimmung der Einzelvorhersagen ist [BeST99 255]. Die Sicherheit einer Aussage lässt sich auch über *Streubereiche oder Konfidenzintervalle* vermitteln. Streubereiche definieren ein Intervall, in dem ein bestimmter Anteil der wahren Werte liegt. So sind z.B. 68,3% der Werte einer normalverteilten Variablen im Bereich $x \pm s$ loziert.[235] Aussagekräftiger sind Konfidenzintervalle, die jenen Wertebereich angeben, in dem die wahren Werte des Zielmerkmals mit einer als Konfidenzniveau vorgegebenen Sicherheit (üblicherweise 95%) liegen [Zöfe03, 104-107].

Zusammenfassung: Bewertung der Gültigkeit von Analyseergebnissen

Die Prüfung der Gültigkeit umfasst grundsätzlich die Genauigkeit und Sicherheit von Aussagen oder Modellen und geschieht stets in Abhängigkeit vom eingesetzten Analyseverfahren sowie von Anwendungs- und Analysekontext. Die in diesem Abschnitt erörterten Kriterien vermitteln einen Überblick über die Vielfalt der zu treffenden Erwägungen, können diese komplexe Thematik jedoch nicht vollständig erfassen. Empfehlungen zu anwendbaren oder angezeigten Bewertungskriterien liefern jeweils die an den Modellierungsobjekten annotierten Hinweise und Regeln. Die Speicherung anwendungsspezifischer Analyseaufgaben in der Fallbibliothek ermöglicht neben allgemeinen Empfehlungen zusätzlich gezielt auf spezifische Kontexte zugeschnittene Vorgaben.

---

[234] Die weiteste Verbreitung hat die quadratische Verlustfunktion (quadratic loss function) erlangt, ebenfalls große Popularität genießt die Informationsverlustfunktion (informational loss function) [WiFr00, 134-136].

[235] Hierbei sind x: arithmetisches Mittel; s: Standardabweichung.

7.2.1.2 Interpretation von Analyseergebnissen (K1.2)

Nach der Bewertung der Analyseergebnisse, die nach objektiv-quantitativen Maßstäben geschieht, folgt eine inhaltliche Interpretation, die sich stärker am Anwendungskontext orientiert und durch subjektive Erwägungen geprägt ist. Ihr Ziel ist die abschließende Beurteilung der Interessantheit sowie die Erlangung eines tiefen Verständnisses der empirischen Aussagen und der mit ihnen verbundenen Implikationen [Knob01, 99]. Ihr Ergebnis sind Entscheidungen über die Verwendung der Analyseergebnisse und über das weitere Vorgehen.

Die Interpretation verlangt neben analysemethodischen in erster Linie nach profunden Domänenkenntnissen. Idealerweise nimmt ein interdisziplinäres Team ausgewählter Experten die Aufgabe gemeinsam wahr. Auf diese Weise wird sichergestellt, dass die Beurteilung fachgerecht ausfällt und die gewonnenen Erkenntnisse der bestmöglichen Nutzung zugeführt werden. Die Expertise der Domänenexperten kann durch die während der Datenexploration gewonnenen Einblicke in die Datenbasis sinnvoll ergänzt werden. Die Interpretation kann auf die Resultate der Ergebnisaufbereitungsaufgaben des Analyseprozesses zurückgreifen [BeLi97, 72], [FrPM91, 12], [Knob01, 102]. Sie behandelt die in Abschnitt 7.2.1 erörterten Kriterien Verständlichkeit, Neuartigkeit und Nützlichkeit in drei Stufen.

Beurteilung der Verständlichkeit

Die Verständlichkeit von Aussagen lässt sich sowohl aus syntaktischer als auch aus semantischer Perspektive betrachten. Die *syntaktische Verständlichkeit* ist hauptsächlich durch Repräsentationsform und Komplexität der Aussagen geprägt. Entsprechende Vorgaben können im `Informationsbedarfsprofil` des Analyseproblems hinterlegt werden. Die Erfüllung dieser Anforderungen ist kein Selbstzweck, sondern dient der besseren kognitiven Zugänglichkeit für den Informationsempfänger. Bei Bedarf können weitere Transformationen zur Änderung der Repräsentationsform veranlasst oder geeignete Visualisierungswerkzeuge herangezogen werden. Im Hinblick auf die Komplexität ist im Allgemeinen eine möglichst kompakte Beschreibung gewünscht. Gerade explorative Untersuchungen produzieren häufig sehr umfang-

reiche, kaum überschaubare Aussagemengen (z.B. Assoziationsanalysen) [HiWi01, 83].

Ein formales Maß der Kompaktheit einer Beschreibung, das eine Objektivierung der syntaktischen Verständlichkeit anstrebt und einigen Analyseverfahren zur Filterung der Ergebnismenge dient, ist die *Minimum Description Length* (*MDL*).[236] Bei identischem Abdeckungsgrad des Informationsgehalts der Analysedaten gilt jenes Ergebnis als überlegen, das in möglichst wenigen einfachen Ausdrücken (minimaler Beschreibungslänge) formuliert ist [BeLi97, 98], [HiWi01, 76]. Kompakte Ergebnisse führen häufig, jedoch nicht zwangsläufig zu verständlicheren Ergebnissen. So wird eine Entscheidungstabelle, die in allen Zeilen dieselben Attribute aufführt, typischerweise als eingängiger empfunden als ein äquivalenter, aber kompakterer Entscheidungsbaum [Domi99, 418]. Die syntaktische Verständlichkeit muss demnach als nicht vollständig objektivierbar betrachtet werden. In vielen Fällen besteht ein Zielkonflikt zwischen Verständlichkeit und Genauigkeit (vgl. [BeLi97, 94], [Domi99, 421], [WiFr00, 151]). Ob eine Verbesserung der Performanz die damit oft einhergehende Komplexitätssteigerung rechtfertigt, wird letztlich von der `Ausrichtung` der Analyse bestimmt: Ist die Erklärung eines Sachverhalts das Ziel, sollte eine hohe Verständlichkeit angestrebt werden. Ist hingegen eine Prognose zu erstellen, ist die Genauigkeit der eingesetzten Prognosemodelle zu optimieren. Der Zielkonflikt wird somit bereits durch eine angemessene Problemspezifikation aufgelöst.

Die *semantische Verständlichkeit* betrifft die inhaltlich-fachliche, intellektuelle Erfassung und Deutung des von den Analyseergebnissen repräsentierten Sachverhalts sowie der damit verbundenen Implikationen. Kann ein solches Verständnis [Dude16b] nicht erlangt werden,

[236] MDL ist definiert als Anzahl der Bits, die in der Syntax des Verfahrens zur Codierung gefundener Muster sowie aller nicht vom Modell abgedeckten Aussagen nötig sind [BeLi97, 98], [HiWi01, 76]. Ihr liegt das nach dem mittelalterlichen Philosophen WILLIAM OF OCCAM benannte Prinzip „Occam's Razor" zugrunde, demzufolge die beste wissenschaftliche Theorie unter ansonsten gleichen Bedingungen jene ist, die alle Fakten am einfachsten erklärt [WiFr00, 151]. Seine Anwendbarkeit auf die Datenanalyse untersucht DOMINGOS [Domi99].

ist das Ergebnis als nicht interessant zu werten. Die situativen Gründe, weshalb ein Ergebnis nicht nachvollziehbar ist, können vielfältig sein. Im Allgemeinen gilt eine Aussage als umso weniger verständlich, je geringer ihre Anschlussfähigkeit und Konsistenz mit existierendem Vorwissen ist [Domi99, 418], [HiWi01, 83]. Darüber hinaus spielen kognitive Einflüsse und persönliche Eigenschaften des Informationsempfängers eine Rolle [Domi99, 418] (vgl. Abschnitt 5.4.3.1).

Beurteilung der Neuartigkeit

Nach SHANNON entspricht der Informationsgehalt einer Aussage formal dem Logarithmus der Wahrscheinlichkeit ihres Auftretens. Demnach besitzen unwahrscheinliche Aussagen einen hohen Informationsgehalt, während häufig auftretende Aussagen nur wenig Information enthalten. Bereits bekannte, redundante oder triviale Ergebnisse erweisen sich somit als prinzipiell wenig interessant [AdZa96, 19, 116] (vgl. Abschnitt 2.1.2.2).

Demnach ist die Beurteilung der Neuartigkeit eines Ergebnisses stark subjektiv und hängt vom Kenntnisstand des Informationsempfängers ab [Knob01, 100]. In der Mehrzahl der Fälle eignen sich bereits bekannte Aussagen allenfalls zur Bestätigung von Annahmen [KrWZ98, 27]. Dennoch existieren Fälle, in denen gerade bekannte Muster interessante Einblicke erlauben [NeKn15, 264-266]: So kann die Abwesenheit einer erwarteten Regel oder die Abweichung bestimmter Gütemaße vom bisherigen Niveau aufschlussreich sein, z.B. wenn im Einzelhandel die erwartete Konfidenz von 100% bei einem Kaufverbund zwischen Mehrweg-Getränken und dem zugehörigen Pfandbetrag nicht erreicht wird. In die Beurteilung der *Bekanntheit* eines Ergebnisses sollten daher alle verfügbaren Genauigkeits- und Sicherheitsmaße einfließen.

Wird ein Sachverhalt hingegen mehrfach durch inhaltlich kongruente Aussagen beschrieben (*Redundanz*) oder gibt ein Ergebnis logische, kausale oder geschäftspolitische Abhängigkeiten wieder (*Trivialität*), können die betreffenden Aussage in der Regel schadlos als uninteressant eliminiert werden. Beispielsweise ist die Aussage „Im Vertreterbezirk Bamberg wurden Rekordumsätze erzielt“ redundant, wenn zugleich die Aussage „In allen oberfränkischen Vertreterbezirken

wurden Rekordumsätze erzielt“ gilt. Als trivial ist z.B. die Aussage zu kennzeichnen, dass bei Verkauf nach Übersee überdurchschnittlich hohe Frachtkosten entstehen [Knob01, 101].

Beurteilung der Nützlichkeit

Oberstes Kriterium zur Beurteilung von Analyseergebnissen ist die Nützlichkeit für die Lösung des vorliegenden Sachproblems. Diese Handlungsrelevanz berührt den pragmatischen Aspekt des Informationsbegriffs: Die Information soll den Empfänger in die Lage versetzen, konkrete Entscheidungen oder Handlungsmaßnahmen zu treffen.[237] Der aus dem Ergebnis gezogene Nutzen sollte darüber hinaus die entstehenden Kosten übersteigen [GoJä09, 28]. Die Beurteilung der Nützlichkeit bestimmt letztlich über den Fortgang des Analyseprojekts. Nicht handlungsrelevante Ergebnisse führen zur Beendigung bzw. Wiederholung der Untersuchung unter veränderten Vorgaben. Die Ausprägung nützlicher Ergebnisse leitet die Wahl der nächsten Schritte.

Kriterien der inhaltlich-sachlichen Beurteilung

Die *inhaltliche Nützlichkeit* eines Ergebnisses wird anhand der Problemspezifikation geprüft. Zunächst ist die Erreichung des **`Analyseziels`** zu kontrollieren. Diese ist gegeben, wenn das Ergebnis eine befriedigende Antwort auf die `Analysefrage` geben kann, die alle Beschreibungselemente der Analysefrage (Aussagetyp, Argumente, Dimensionen) berücksichtigt. Zusätzlich ist die Erfüllung aller bislang noch nicht beachteten Anforderungen des `Informationsbedarfsprofils` zu kontrollieren, etwa im Hinblick auf Vollständigkeit, Bestimmtheit (Detailliertheit) und Zeitbezug bzw. Aktualität der Aussagen.

Decken die Ergebnisse den Informationsbedarf, ist ihr Beitrag zur Lösung des Sachproblems zu untersuchen, sofern das Analyseproblem einem lösungsbezogenen Problemaspekt oder einer Handlungsmaß-

[237] Dieser pragmatische Aspekt kommt in der häufig von englischsprachigen Autoren erhobenen Forderung nach „actionable information“ zum Ausdruck (vgl. z.B. [BeLi97, 18], [KlPR98b, 311]).

nahme zugeordnet ist.[238] Hierzu werden die Attribute des **Problemaspekts** betrachtet. Konkret ist zu hinterfragen, ob das durch den Zielinhalt bestimmte Domänenobjektmerkmal sowie der Wertbeitrag auf Grundlage der vorliegenden Ergebnisse effektiv in die vom Modifikator anzeigte Richtung beeinflussbar erscheinen (z.B. Steigerung des Bonbetrags zur Realisierung der Umsatzziele). Zur besseren Einschätzung der Realisierbarkeit können Zeitbezug und Ausmaß des Zielinhalts beitragen: Sind die Informationen aktuell, kommen sie rechtzeitig für eine Intervention, ist der gewünschte Soll-Wert damit erreichbar [HsKn95, 156], [BeLi97, 34]?

Kriterien der ökonomischen Beurteilung

Fällt die inhaltliche Einschätzung positiv aus, kann eine Bewertung der *ökonomischen Nützlichkeit* folgen. Hierzu werden die Konsequenzen, die bei Realisierung der jeweiligen Handlungsmaßnahme zu erwarten sind (z.B. der durch eine Werbeaktion initiierte Mehrabsatz eines Produkts), mit monetären Größen bewertet [BeLi97, 110], [WeZS08, 134], [LeVo10, 80]. Geeignete Wertgrößen ergeben sich häufig direkt aus dem Wertbeitrag des Problemaspekts (z.B. Umsatz). Da stets auch die Kosten der Maßnahme berücksichtigt werden sollten, kann die Bewertung auch negative Werte annehmen.[239]

In vielen Fällen ist eine belastbare Abschätzung des Maßnahmenerfolgs nur auf Grundlage differenzierter Einzelbewertungen erreichbar. Für Prognosemodelle, die besonders häufig ihren Niederschlag in Handlungsmaßnahmen finden, wird in Abschnitt 7.2.1.1 auf Nutzen- und Risiko-Präferenzen hingewiesen, die im Hinblick auf spezifische Prognoseeigenschaften eines Modells zu formulieren sind. Sie

[238] Situationsbezogene Problemaspekte bzw. Informationsmaßnahmen spezifizieren einen Informationsbedarf, der vom Analyseziel konkretisiert wird, weshalb in diesen Fällen die Berücksichtigung der Anwendungsebene unterbleiben kann. Wurde kein Sachproblem definiert, ist die Beurteilung dennoch möglich.

[239] Die beschriebene Nutzen-Kosten-Betrachtung stellt eine Ex-ante-Bewertung dar, die sich auf potenzielle, erwartete Wertbeiträge stützt. Sie ist von der in Abschnitt 7.3.2 (K4) diskutierten Ex-post-Bewertung abzugrenzen, bei der tatsächlich angefallene Nutzen- und Kostenwerte angesetzt werden, um retrospektiv die Profitabilität einer Maßnahme zu messen.

resultieren aus der Beobachtung, dass verschiedene Prognosefehler typischerweise unterschiedlich hohe Kosten verursachen (asymmetrische Kostenfunktionen) [BoAr01, 197]. Bei Klassifikatoren sind Fehlalarme in der Regel mit geringerem Risiko und niedrigeren Kosten (z.B. entgangener Gewinn) verbunden als das Versagen bei der Identifikation positiver Fälle (z.B. Verluste infolge eines nicht erkannten Betrugsfalls).

Zur Quantifizierung der bei Einsatz des Modells insgesamt zu erwartenden Kosten sind die Fehlklassifikationen mit spezifischen Kostensätzen zu bewerten [BoAr01, 197], [HiWi01, 77]. Korrekte Zuordnungen können analog mit spezifischen Erträgen oder Nutzenfaktoren belegt werden [WiFr00, 138]. Hierzu ist eine *Nutzen-Kosten-Funktion* zu formulieren (vgl. [BoAr11, 231f.], [KrBü11, 54-57]), die in den `Bewertungsfunktionen` an der Analyseaufgabe dokumentiert wird. Mit bekannten Erträgen E_N durch die korrekte Einordnung von Negativen TN sowie Kosten C_N und C_P für die Fehlklassifikation von Negativen FP bzw. Positiven FN ergibt sich z.B. mit $E_N \times TN - (C_N \times FP + C_P \times FN)$ eine einfache Nutzen-Kosten-Funktion.[240] Sie stellt den Erträgen die Summe der Kosten gegenüber. Sind keine Nutzenfaktoren relevant, kann der erste Term entfallen (vgl. [BeLi97, 259]). Analog lassen sich auch bei Schätzern Fehlerkosten berücksichtigen, indem gemessene Abweichungen bewertet werden [HiWi01, 77].

Bei der monetären Bewertung sollten nach Möglichkeit auch die erwarteten Kosten der Maßnahme berücksichtigt werden. Ein Zahlenbeispiel für die Optimierung einer Direktmarketingkampagne auf Basis des Deckungsbeitrags zeigen NECKEL & KNOBLOCH [NeKn15, 333-335]. Ein Analyseergebnis ist aus ökonomischer Perspektive nützlich, wenn sein erwarteter Nutzen die erwarteten Kosten übersteigt [BeSt99, 227].

Lösungsoptionen zur Auswahl aus mehreren Ergebniskandidaten

Zuweilen verbleiben nach der bisherigen Beurteilung mehrere gültige und potenziell nützliche Ergebnisse. So werden z.B. häufig gezielt

[240] TN (True Negatives), FP (False Positives) und FN (False Negatives) entsprechen den Häufigkeiten aus der Klassifikationstabelle, vgl. Anhang A7.2. Weiterhin liegt die vereinfachende Annahme zugrunde, dass die Faktoren unabhängig von der konkreten Wahrscheinlichkeit einer Instanz sind [KrBü11, 57].

mehrere Prognosemodelle erzeugt, um daraus den im jeweiligen Anwendungskontext nützlichsten Kandidaten auszuwählen. Im Folgenden wird stellvertretend der Einsatz von Halbordnungen und skalarwertigen Bewertungsmaßen als etablierte Hilfsmittel für die Auswahl von Prognosemodellen skizziert.[241]

Halbordnungen stellen eine Reihung mehrerer Modelle nach definierten Kriterien dar. Ein verbreitetes Kriterium ist der für beschreibende Analysen eingeführte *Lift* (Abschnitt 7.2.1.1). Im vorliegenden Kontext bewertet er die durch Modellanwendung erreichte Verbesserung bei Erfüllung einer Prognoseaufgabe gegenüber dem Verzicht auf ein Modell (Trivialprognose) [WiFr00, 139].[242] Auf diese Weise ermöglicht er auch den Vergleich mehrerer (heterogener) Modelle [BeLi97, 107]. Er bezieht sich stets auf eine Teildatenmenge, deren Umfang als Prozentanteil der Gesamtpopulation (Support) angegeben ist [BeST99, 226], [HiWi01, 77].

Das klassische Einsatzgebiet des Lift ist die Zielgruppenoptimierung im Direktmarketing, wo eine Adressliste mit möglichst hohem Anteil an Interessenten (Response-Quote) für eine Kontaktaufnahme gesucht wird. Beträgt z.B. die Response-Quote (Zielmerkmal) in der Population 0,5%, und ein Modell A selektiert eine Teilmenge von 10.000 Adressen mit einer Response von 4%, so beträgt der Lift 4 / 0,5 = 8. Ein Modell B selektiert hingegen 50.000 Kontakte und erreicht eine Response-Quote von 2%, d.h. einen Lift von 4. Je nach Zwecksetzung muss jedoch nicht zwingend das Modell mit dem höchsten Lift das zu präferierende sein [BeLi97, 108]. So kann im Beispiel die von Modell A produzierte Liste zu wenige Adressen für die geplante Marketingaktion enthalten. Um den damit verbundenen Aufwand zu rechtfertigen, mag die von Modell B erzeugte, längere Liste unter Inkaufnahme eines niedrigeren Lifts

[241] Eine ausführliche Diskussion zur Zweckeignung verschiedener Kriterien findet sich bei [KrBü11, 52-54].

[242] Konkret misst der Lift den Grad, in dem die modellgestützte Klassifikation die Konzentration des Zielmerkmalswertes gegenüber dem reinen Zufall verbessert. Für Schätzmodelle quantifiziert er für das Zielmerkmal das Verhältnis des Stichprobenmittelwerts zum Populationsmittelwert [HiWi01, 77].

nützlicher sein. Es ist jeweils abzuwägen, welche Relation der Größen Lift und Support im vorliegenden Kontext zu wählen ist.

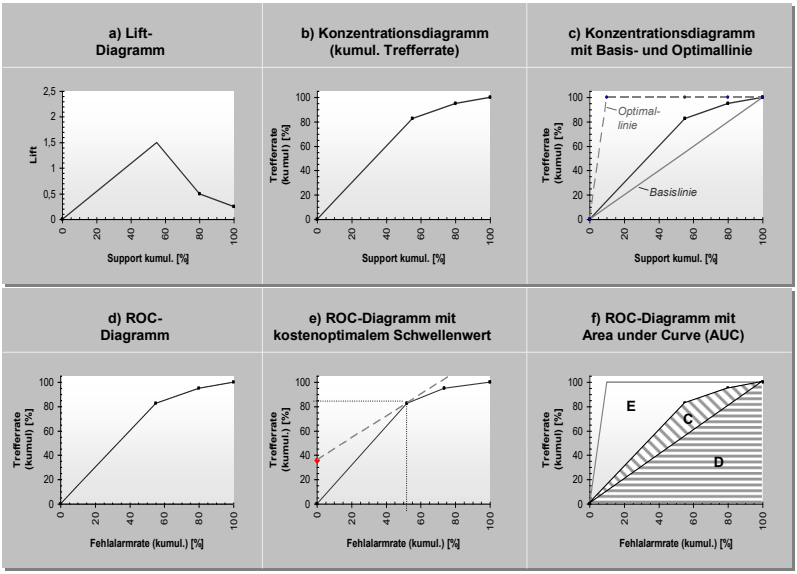


Abbildung 102: Beispielhafte Lift-, Konzentrations- und ROC-Diagramme zur Beurteilung von Klassifikatoren (eigene Darstellung; Daten nach [KrBü11])

Der Zusammenhang zwischen konkurrierenden Zielgrößen lässt sich gut in Diagrammen visualisieren. In einem *Lift-Diagramm* stellt die Abszisse den kumulierten Anteil der bewerteten Instanzen (Support), die Ordinate den Lift dar[243] (Abbildung 102a). Hierzu werden die Instanzen absteigend nach Zielmerkmalswert (Score) sortiert, und für jedes Perzentil bzw. Dezil des Supports der Lift berechnet. Bei Entscheidungsbäumen ist es üblich, für jeden Blattknoten (Partition) einen Datenpunkt einzutragen; der Support bestimmt sich nach dem Anteil der Beispiele in der betrachteten Partition. Die resultierenden Diagramme weisen somit für jeden Blattknoten ein Liniensegment aus [BeLi00, 188].

[243] Alternativ kann auch der kumulierte Lift abgetragen werden.

Konzentrationsdiagramme[244] tragen anstelle des Lifts auf der Ordinate den kumulierten Anteil der Instanzen der Zielklasse ab, wiederum absteigend geordnet nach Zielmerkmalswert. Dieser Wert entspricht der Trefferrate (Sensitivität) (Abbildung 102b). Die konkave Form der Kurve indiziert eine Konzentration der gesuchten (positiven) Zielklasse in den „besten" Partitionen mit den höchsten Zielmerkmalswerten (links im Diagramm) [HiWi01, 78f.]: Die x% der Instanzen mit dem höchsten Score-Wert decken bereits y% der Treffer ab, was einem Lift von y / x entspricht. Diese Verbesserung wird klarer sichtbar, wenn zum Vergleich die Basislinie (Diagonale) eingetragen wird, die die uninformierte Prognose repräsentiert und jedem Support-Anteil den wertgleichen Anteil an der Zielklasse zuordnet (Zufallsauswahl; $x = y$). Weiterhin kann auch die Optimallinie dargestellt werden, die der Partition mit dem höchsten Score ausschließlich alle Positiven der Stichprobe zuweist (Abbildung 102c) [KrBü11, 44-46]. Demnach ist ein Modell umso besser, je näher seine Lorenzkurve der „nordwestlichen Ecke" des Diagramms kommt [BeLi00, 56], [WiFr00, 141].

Eine Abwägung nach anderen Kriterien führt zu spezifischen Trade-off-Diagrammen. In Diagnoseanwendungen sind z.B. *ROC-Diagramme*[245] gebräuchlich (Abbildung 102d), welche die Zielkonkurrenz zwischen Trefferrate und Fehlalarmrate visualisieren[246] [Fawc06, 861]. Sie zeigen damit den Zusammenhang zwischen Nutzen und Kosten auf, die durch korrekte bzw. fehlerhafte Treffer entstehen [Fawc06, 862]. Eine Tabelle mit verbreiteten Diagrammtypen für typische Anwendungsdomänen und spezifischen Eigenschaften enthält Anhang A7.3. Die im jeweiligen Kontext anzuwendenden Kriterien können anhand der `Präferenz-`

[244] Konzentrationsdiagramme (Lorenzkurven, Gini curves, cumulated accuracy profiles) werden häufig irrtümlich als Lift-Diagramme bezeichnet. Vgl. zur Begrifflichkeit die Diskussion bei [BeLi97, 107-109].

[245] Sie entstammen der Signalerkennungstheorie, wo sie die Betriebseigenschaften eines Empfängers (receiver operating characteristics, ROC) bei der Datenübertragung beschreiben [Fawc06, 861].

[246] Lorenz- und ROC-Kurven ähneln sich stark bei sehr kleinem Positiven-Anteil in der Population, durch den der Unterschied zwischen Stichprobengröße und Negativen-Anteil kaum in Erscheinung tritt [WiFr00, 142].

`relation` gewählt werden. Die Relation zur Zielgruppenselektion aus Abschnitt 7.2.1.1 „`Trefferrate(max) > Fehlalarmrate(min)`“ legt z.B. ein Konzentrations- und ein ROC-Diagramm nahe.

Zum Vergleich mehrerer Modelle werden deren Kurven in ein gemeinsames Diagramm eingetragen. Modell A ist besser als Modell B, wenn As Lorenzkurve oberhalb jener von B verläuft [KrBü11, 44]. Für den praktischen Einsatz ist schließlich ein Punkt auf einer Kurve zu selektieren, der eine Teildatenmenge mit hoher Trefferrate bei gleichzeitig hoher Abdeckung der insgesamt in der Population vorhandenen Positiven (Support) repräsentiert [WiFr00, 146]. Häufig ist bezüglich einer der beiden Zielgrößen ein *Schwellenwert* vorgegeben, der nicht über- oder unterschritten werden darf. Beispiele sind der Umfang der Adressliste im Direktmarketing (Mailout, absoluter Support) oder der Score, der für die Akzeptanz eines Kreditantrags im Bankwesen mindestens zu erreichen ist. Solche Schwellenwerte werden in vielen Fällen auf Grundlage monetärer Betrachtungen festgelegt. Durch Bewertung der Zielgrößen mit Kosten oder Gewinnerwartungen entstehen aus Trade-off-Diagrammen z.B. sogenannte Gains Charts bzw. Profit Charts [HiWi01, 78f.]. Abbildung 102e zeigt ein ROC-Diagramm mit einer Tangente, die den kostenoptimalen Schwellenwert markiert. Die Optimierungsfunktionen werden wiederum als `Bewertungsfunktionen` dokumentiert.[247]

In der Regel erstreckt sich die Dominanz eines Modells auf ein bestimmtes Intervall bezüglich der Abszisse [BoAr01, 232f.]. Schneiden sich die Kurven mehrerer Diagramme, oder ist eine globale Bewertung unabhängig von Schwellenwerten gesucht, sind Diagramme zur Modellauswahl nur bedingt geeignet. Für solche Fälle sind *skalarwertige Gütemaße* verfügbar. Diese aggregieren z.B. schwellenwertabhängige Maße an mehreren Punkten [WiFr00, 146], berechnen die Kurvenlänge oder den maximalen vertikalen Abstand zur Basislinie [KrBü11, 48]. In der Praxis verbreitet sind flächenbasierte Maße. Die *Fläche unterhalb der*

[247] Das Nutzenmaximum liegt gemäß der oben diskutierten Nutzen-Kosten-Funktion an jener Stelle, an der die Steigung der ROC-Kurve den Wert $(E_N + C_N) / C_P$ annimmt (vgl. [KrBü11, 54-57]), das Kostenminimum ist durch die Relation $(C_N / C_P) \times (N / P)$ gekennzeichnet [BoAr01, 232].

Kurve (area under curve, AUC) nimmt eine Teilfläche des Einheitsquadrats ein (Fläche C+D in Abbildung 102f) [Fawc06, 868]. Der *Gini-Koeffizient* ist das Verhältnis der Flächen unterhalb der Konzentrationskurven des betrachteten Modells und dem optimalen Klassifikator (Flächen C und C+E in Abbildung 102f). Da beide Maße den Abstand der Diagrammkurve von der Basislinie quantifizieren, ist ein Modell jeweils umso besser, je stärker sich das Maß dem Maximalwert 1 annähert [KrBü11, 48].

Viele gängige Maße führen zu qualitativ identischen Aussagen [CaNi04, 73-75], [KrBü11, 50-52], weshalb die Wahl eines Kriteriums häufig von seiner situativen fachlichen Interpretierbarkeit und von den Präferenzen der Beteiligten geleitet wird. Dennoch sind zur schadlosen Anwendung einiger Maße bestimmte Voraussetzungen zu erfüllen [KrBü11, 53], welche mithilfe geeigneter Kontextregeln überprüft werden können.

Zusammenfassung: Interpretation von Analyseergebnissen

Soweit in den vorherigen Schritten die Interessantheit eines Analyseergebnisses festgestellt wurde, ist nun über dessen weitere Verwendung zu befinden. Im Idealfall kann eine Aussage oder ein Modell direkt in Entscheidungen, konkrete Maßnahmen oder operative Prozesse einfließen (Anwendung des Wissens, vgl. Abschnitt 3.3.3.3). In diesem Zusammenhang entsteht häufig der Bedarf, als Resultat der Interpretation einen neuen Problemaspekt bzw. eine neue Handlungsmaßnahme in die Problemkarte einzufügen (vgl. Domänenanalyse Z2, Abschnitt 5.4.4). Einerseits liefert die Interpretation wichtige Grundlagen für die Ableitung von Handlungsoptionen (Z2.4). Sind die Handlungsoptionen direkt mit den Analyseergebnissen verknüpft (wie beim Einsatz von Prognosemodellen), können die beiden Aufgaben in Bezug auf die Nützlichkeitsbeurteilung zum Teil verschmelzen. Andererseits legt die Interpretation Anschlussuntersuchungen nahe, etwa um analytisch hergeleitete Hypothesen zu verifizieren oder Problemaspekte tiefer zu ergründen.

Die dialektische Beziehung zwischen Domänen- und Datenanalyse bedingt die Fortschreibung der Problemkarte, sobald neue Erkenntnisse erlangt wurden, und ist das zentrale Anliegen der evidenzbasierten

Problemlösung, wie sie in dieser Arbeit vertreten wird. Die Interpretation bildet die Schnittstelle zwischen den beiden Ebenen.

In der Praxis endet die Beurteilung von Datenanalysen üblicherweise mit Aufgabe K1, die sich den Ergebnisparametern widmet. Nicht befriedigende Analyseergebnisse können auf diese Weise nicht auf den sie produzierenden Prozess zurückgeführt bzw. aus dessen Merkmalen erklärt werden. Wiederholte oder künftige Analysen erfolgen demnach zwangsläufig dem Versuchs-Irrtums-Prinzip. Wird eine systematische Prozessverbesserung angestrebt, sind zusätzlich die Prozessparameter zu betrachten. Dies geschieht in der nachfolgenden Aufgabe K2.

7.2.2 Beurteilung des Prozessablaufs (K2)

Das Ziel dieser Aufgabe ist die Prozesskontrolle im Sinne einer Gegenüberstellung der realisierten Leistung mit den Vorgaben sowie die Ermittlung der Ursachen identifizierter Diskrepanzen und die Ableitung geeigneter Verbesserungsmaßnahmen (vgl. [ScSe04, 173]). Ergebnisse der Aufgabe sind der Zielerreichungsgrad des Ablaufs und Vorschläge zur Prozessverbesserung. Darüber hinaus wird der Ressourcenverbrauch des Ablaufs ermittelt, der als Grundlage zur finanziellen Evaluation des Projekts sowie zur Planung künftiger Vorhaben dient. Die Beurteilung stützt sich auf das Instanzmodell (Prozessinstanz), das alle relevanten Angaben detailliert protokolliert.

Offenbart die Ablaufbeurteilung Mängel, die nach einer gezielten Modifikation einzelner Prozesselemente verlangen, so ist diese abhängig vom weiteren Vorgehen vorzunehmen: Wird der Ansatz der lernenden Revision verfolgt, geschieht die Prozessverbesserung im Rahmen der Modifikation der Analysepläne (K5, Abschnitt 7.4.1). Die Änderungen stehen sodann für alle künftigen Analysevorhaben bereit. Andernfalls können die Verbesserungsvorschläge in direkt folgende Prozesse einfließen, die häufig der Wiederholung nicht erfolgreicher Analysen dienen.

Während die Beurteilung der Analyseergebnisse mit den Produkten des Prozesses dessen Sachziel betrachtet, nimmt die Beurteilung des Prozessablaufs vornehmlich auf Formalziele Bezug, die anhand der

Systemmerkmale Verhalten und Struktur eines Prozesses gegliedert werden können [Fers92, 11]. Die weitere Einteilung erfolgt nach den Zielkategorien des Prozessmanagements (vgl. Abschnitt 2.4.2). So wird das Ablaufverhalten separat nach Effektivitäts- und Effizienzgesichtspunkten betrachtet, und die Ablaufstruktur unter Einbeziehung der Flexibilitätsperspektive behandelt. Die Teilaufgaben dieses Schrittes umfassen demnach die *Beurteilung der Effektivität (K2.1)*, die *Beurteilung der Effizienz (K2.2)* sowie die *Beurteilung der Struktur (K2.3)*. Sie werden in den folgenden Abschnitten erörtert. Es schließen sich Lösungsoptionen zur Auswertung von Prozesskennzahlen an (Abschnitt 7.2.2.4).

7.2.2.1 Beurteilung der Effektivität (K2.1)

Kriterium	Bezugsobjekt	Fragestellung
Zielerreichung	Analyseziel	Wurden die im Analyseziel gesetzten Anforderungen erreicht?
Vollständigkeit	Prozessinstanz	Mussten fehlende Funktionen in den Ablauf aufgenommen werden?
Korrektheit	Prozessinstanz	Waren Prozessschema und Prozesselemente eindeutig, widerspruchsfrei und adäquat konzipiert bzw. konfiguriert?
Zuverlässigkeit	Prozessausführung (Aufgabenträger)	Sind während der Prozessdurchführung Fehlfunktionen aufgetreten? Wurden personell Fehler begangen?

Tabelle 4: Kriterien zur Beurteilung der Effektivität des Analyseprozesses

Im Allgemeinen gilt ein Prozess als effektiv, wenn die von ihm erbrachte Leistung die Anforderungen und Erwartungen des Auftraggebers erfüllt [LJWK12, 1039]. Diese Eigenschaft wird häufig mit Prozessqualität gleichgesetzt [ScSe04, 179] (vgl. Abschnitt 2.4.2). Die Beurteilung der Effektivität zielt somit in erster Linie auf die Feststellung des *Zielerreichungsgrades* des Prozessablaufs. Voraussetzung

dafür ist die *Korrektheit* des Prozessverhaltens, welche die Vollständigkeit, Widerspruchsfreiheit und Zuverlässigkeit der Aufgabendefinition und -durchführung impliziert [Fers92, 11]. Tabelle 4 fasst die relevanten Kriterien zusammen.[248]

Zielerreichung

Die Feststellung der Zielerreichung greift auf die Beurteilung der Analyseergebnisse (K1) zurück, ohne erneut auf inhaltliche Details einzugehen. Während dort zusammen mit dem Auftraggeber die fachliche Angemessenheit der Prozessleistung beurteilt wird, ist hier eine technische Prozesskontrolle durch den Analytiker auszuführen. Sie stützt sich auf die Vorgaben von Analysefrage und Informationsbedarfsprofil, die als `Bewertungskriterien` an der **`Analyseaufgabe`** hinterlegt und um weitere Forderungen erweiterbar sind.

Zur Ermittlung des Zielerreichungsgrads des Prozesses werden alle relevanten Kriterien in einem Bewertungsschema angeordnet und die realisierten Ist-Werte mit den gesetzten Ziel-Werten verglichen (Tabelle 5).[249] Jedes Kriterium kann mit Gewichten versehen werden, um ihm größeren oder geringeren Einfluss auf das Gesamturteil zuzubilligen. Unter Würdigung der Relation von Ist- und Ziel-Werten erhält jedes Kriterium einen Zielerreichungsgrad zugeschrieben. Dieser kann im Falle numerischer Werte u.a. als Quotient von Ist- und Ziel-Wert berechnet werden (Kriterium 4), im Falle qualitativer Vorgaben ist er vom Analytiker festzusetzen. Alternativ sind auch Regelsysteme denkbar, die Zielerreichungsgrade für qualitative Kriterien auf Grundlage von Werteintervallen der Ist-Ausprägungen oder Abweichungen

---

[248] In einer Bestandsaufnahme gängiger Kriterien zur Leistungsbewertung von Prozessen verweisen LEY ET AL. auf Defizite bezüglich der Abgrenzung und Klassifikation einzelner Kriterien sowie im Hinblick auf quantifizierbare Effektivitätsmaße. Sie identifizieren die Kategorien Zeit, Kosten, Qualität, Kapazität, Flexibilität, Integration und Komplexität und ordnen jeweils schwerpunktmäßig die Ziele Effektivität und Effizienz zu [LJWK12, 1040]. Sie werden für Aufgabe K2 berücksichtigt, soweit sie jeweils relevant sind.

[249] Vgl. hierzu die Typvereinbarung des Datentyps `Bewertungsergebnis` in Anhang A4.6.

definieren. NAUCK ET AL. präsentieren ein Analysewerkzeug, das in einem ähnlichen Anwendungsfall Fuzzy-Regeln nutzt [NaSA03].

Kriterium		Ziel-Wert	Ist-Wert	Gewicht	Zielerreichung
1	Art.Aussageinhalt. **Analysefrage**	(gemäß Fragetext*)	☑	1	100%
2	Art.Repräsentationsform. **Interpretierbarkeit**	mittel	gering	1	20%
3	Qualität.Gültigkeit. **Sicherheit** → *Response-Quote*	sehr hoch	4%	1	90%
4	Qualität.Gültigkeit. **Genauigkeit** → *Trefferrate*	hoch 1	0,94	1	94%
5	Qualität.Bestimmtheit. **Aktualität**	hoch	2 Wochen	0,5	90%
			Gesamt-Zielerreichungsgrad	4,5	**77,6%**

Legende: ☑: erfüllt / ☒: nicht erfüllt (boolsche Anforderungen) ;
→: operationalisiert zu

*Analysefrage: „Welche Kunden haben die höchste Response-Quote?"

Tabelle 5: Beispielhaftes Bewertungsschema zur Zielerreichung eines Analyseprozesses

Im Beispielschema ist mit der Analysefrage auch ein Kriterium enthalten, das nur im Hinblick auf Erfüllung bzw. Nichterfüllung beurteilt werden soll (boolsche Anforderungen), da ein Vergleich mit Ist-Werten nicht möglich oder nicht sinnvoll erscheint (Kriterium 1). Einige Qualitätskriterien sind abhängig von Analyseaufgabe bzw. -verfahren zu operationalisieren. So werden z.B. die Sicherheit durch die Response-Quote, die Genauigkeit durch die Trefferrate konkretisiert, da es sich um ein Klassifikationsmodell handelt (Kriterien 3 und 4). Abhängig von

der Analyseaktivität können auch verfahrensklassenspezifische Anforderungen als Kriterien auftreten, wie etwa die maximale Anzahl erzeugter Regeln bei Einsatz regelgenerierender Verfahren. Die Auswahl und Gewichtung der letztlich in die Beurteilung eingehenden Kriterien erfolgt entweder im Rahmen der Spezifikation der Analyseaufgabe (P1.1, Abschnitt 5.5.3.1) oder situativ an dieser Stelle. Das Resultat der Beurteilung wird als `Ergebnisbewertung` an der **`Prozessinstanz`** dokumentiert.

Vollständigkeit

Das Verhalten eines Prozesses ist vollständig, wenn er alle zur Produktion der gewünschten Ergebnisse in der gebotenen Qualität erforderlichen Verrichtungen umfasst, ohne dass Nacharbeit, Prozesswiederholungen oder Änderungen am Schema erforderlich sind, bei denen ursprünglich nicht vorgesehene Funktionen realisiert werden. Betrachtungsgegenstand ist die **`Prozessinstanz`**, aus deren Liste der `Änderungen` Abweichungen vom geplanten Ablauf ersichtlich sind, die auf nicht antizipierte Anforderungen hinweisen. Zusätzlich sind fehlende Funktionen zu berücksichtigen, die erst in der Nachbetrachtung als notwendig erkannt werden.

Korrektheit

Die Beurteilung der Korrektheit stützt sich auf den intuitiven Fehlerbegriff (vgl. [Dude16c]): Ein Prozess ist korrekt, wenn er frei von Fehlern ist, d.h., wenn keines seiner Elemente unrichtig, irrtümlich oder nicht den Anforderungen entsprechend konzipiert oder konfiguriert ist.[250] Die Korrektheit kann quantitativ durch Fehlerzahlen gemessen werden,[251]

[250] Das Vorhandensein nicht zielführender Aktivitäten führt zwar zu Ineffizienzen, nicht jedoch zu Fehlern in dem Sinne, dass unbrauchbare Ergebnisse produziert werden.

[251] Im Geschäftsprozessmanagement werden Prozessfehler durch Fehlerquoten in Bezug zur Menge der Prozessergebnisse gemessen [ScSe04, 200f.]. Dies erscheint bei Analyseprozessen jedoch wenig sinnvoll, da als Bezugsgröße keine sinnvolle Outputmenge verfügbar ist. Die Berechnung auf Basis der Anzahl der Prozesselemente oder der insgesamt zu treffenden Gestaltungsentscheidungen erscheint im ersten Fall wenig aussagefähig, im zweiten Fall ist die Ermittlung der Zahl der Entscheidungen nicht praktikabel.

von größerem Interesse sind allerdings Art und Ursache der Fehler. Eine nicht angemessene Konfiguration von Prozesselementen kann im Wesentlichen folgende Gründe haben:

- Vorgänge nutzen Operatoren, deren Anwendungsvoraussetzungen nicht erfüllt sind.
- Vorgänge nutzen Operatoren, die nicht geeignet sind.
- Vorgänge sind bezüglich der Operatorparameter fehlerhaft instanziiert.

Wiederverwendung und Kontextregeln bieten zwar hilfreiche Unterstützungsmöglichkeiten zur Planung korrekter Prozesse. Ihre Nutzung setzt jedoch voraus, dass Erfahrungswissen vorhanden und repräsentiert ist. Solches Wissen kann im Rahmen der Beurteilung gesammelt werden.

Zuverlässigkeit

Während die Korrektheit Gestaltungsfehler fokussiert, betrifft die Zuverlässigkeit Laufzeit- oder Ausführungsfehler, d.h., die Qualität der Vorgangsdurchführung durch die Aufgabenträger. Fehler dieser Art können durch Störungen, Ausfälle oder Überlastungen der technischen Infrastruktur auftreten oder durch Personen begangen werden. Fehlfunktionen bei maschinellen Aufgabenträgern können zur Überprüfung von Verfügbarkeit und Kapazität der gewählten Systemumgebung Anlass geben und z.B. einen Wechsel auf leistungsfähigere Rechner nahelegen. Menschliches Versagen ist meist durch Irrtümer, Informations- oder Kenntnismängel bedingt. Letzteren ist z.B. durch Übertragen der Aufgaben an erfahrenere Experten zu begegnen.

7.2.2.2 Beurteilung der Effizienz (K2.2)

Prozesseffizienz beschreibt das Verhältnis zwischen Leistung und Ressourceneinsatz eines Prozesses [LJWK12, 1039]. Zur Beurteilung des Ressourceneinsatzes werden in diesem Schritt das Zeitverhalten und die bei der Prozessdurchführung angefallenen Kosten betrachtet. Das Zeitverhalten wird anhand der Zeitdauer, der Synchronisation und des Last-

verhaltens näher untersucht. Tabelle 6 zeigt die Kriterien zusammen mit den jeweiligen Untersuchungsfragen. Ausgangspunkt für Effizienzbewertungen bildet stets die Zeitdauer [LJWK12, 1041], da sie als Basis zur Berechnung aller weiteren Bewertungsmaße dient.

Kriterium	Bezugsobjekt	Fragestellung
Zeitdauer	Prozessinstanz; Vorgänge	Wie lange dauerte die Durchführung einzelner Vorgänge bzw. des gesamten Prozessablaufs?
Synchronisation	Vorgänge	Wann und wie oft treten Start- und Endereignisse einzelner Vorgänge ein? (Parallelität, Wiederholungen)
Lastverhalten	Prozessausführung (Aufgabenträger)	Konnten die Aufgabenträger den Last- und Reaktionsanforderungen genügen?
Ablaufkosten	Prozessinstanz; Vorgänge	Wie hoch sind die Kosten für die Durchführung einzelner Aktivitäten bzw. des gesamten Prozesses? Wie hoch sind variable und fixe Kostenanteile?

Tabelle 6: Kriterien zur Beurteilung der Effizienz des Analyseprozesses

Zeitdauer

Die Dauer eines Prozessablaufs (Prozesszeit) setzt sich aus den Prozesszeiten der einzelnen Vorgänge zusammen und wird als Durchlaufzeit oder Zykluszeit gemessen [ScSe04, 188]. Beide sind wichtige Kenngrößen zur Beurteilung eines Prozessablaufs.

- Die *Durchlaufzeit* ist die Zeitspanne vom Auftreten des prozessauslösenden Ereignisses bis zur Übergabe des Ergebnisses an den Leistungsempfänger [GaSV94, 14], [Jung02, 104].
- Die *Zykluszeit* ist die Summe der Prozesszeiten aller Vorgänge, wobei parallel ablaufende Vorgänge einzeln berücksichtigt werden. Sie berichtet über den insgesamt zur Prozessausführung benötigten Zeitaufwand und die Dauer der Ressourcenbindung [ScSe04, 188f.].

Eine genauere Analyse der Prozesszeiten verspricht Erkenntnisse über Ineffizienzen und mögliche Verbesserungspotenziale. Inwiefern diese für Datenanalyseprozesse sinnvoll ist, wird im Folgenden diskutiert. Von besonderem Interesse sind grundsätzlich Zeitanteile, die nicht unmittelbar der Vorgangsdurchführung dienen, sondern als Unterbrechungszeiten unproduktiv sind oder als Rüst-, Transfer- oder Kontrollzeiten auftreten.[252] Bei der Datenanalyse werden Tätigkeiten, die der Datenübertragung, der Interpretation und Kontrolle dienen, in der Regel als Prozessaufgaben modelliert. Der Zeitbedarf der zugehörigen Vorgänge ist somit detailliert messbar. Allerdings beinhalten viele Aufgaben, wie z.B. Operatorinstanziierung und Ergebnisinterpretation, kreative und kommunikative Elemente, für die eine präzise Abgrenzung unproduktiver Zeitanteile weder praktikabel noch sinnvoll erscheint, da sie wesentliche Bestandteile der Analyse sind. Mit Ausnahme repetitiver Analysen enthalten alle Abläufe Gestaltungsanteile (vgl. Abschnitt 6.1.2), die nicht undifferenziert als unproduktiv zu werten sind.

Unter der Annahme einer detaillierten Protokollierung von Start-, End- und Unterbrechungszeiten kann zwischen Brutto- und Netto-Prozesszeit unterschieden werden: Die Brutto-Prozesszeit ist die Zeitpanne vom Start bis zur Beendigung eines Vorgangs. Durch Subtraktion aller Unterbrechungsintervalle ergibt sich die Netto-Prozesszeit (Bearbeitungszeit). Aus dem Vergleich von Brutto- und Netto-Zykluszeit lassen sich für repetitive Abläufe Ansatzpunkte zur Effizienzsteigerung ableiten.[253] Für andere Prozesse ist dies wegen ihres geringen Wiederholungsgrades nur eingeschränkt sinnvoll. Vorgänge mit überdurchschnittlich langen Unterbrechungen können auf schlecht strukturierte Aufgaben verweisen, für die gezielt nach Abhilfe gesucht werden kann.

---

[252] In der Literatur finden sich verschiedene Gliederungsvorschläge. Typischerweise wird zwischen Liege-, Transfer- und Bearbeitungszeit unterschieden [GaSV94, 15]. Je nach Anwendungsbereich können Rüst-, Kontroll-, Reparatur- oder Nachbearbeitungszeiten hinzukommen [Jung02, 75], [LJWK12, 1041]. Zuweilen wird die Bearbeitungszeit weiter differenziert in (operative) Wertschöpfungs- und Unterbrechungszeiten. Liege- und Unterbrechungszeiten werden gelegentlich zur Wartezeit zusammengefasst [Reif03, 173].

[253] Bei vollautomatisierten repetitiven Abläufen sind kaum oder nur kurze Unterbrechungszeiten zu erwarten.

Die exakte Protokollierung der Bearbeitungszeit kreativer Aufgaben ist selbst bei integrierten Analysewerkzeugen nicht realistisch. Zudem ist eine mathematische Berechnung von Unterbrechungszeiten nur in Bezug auf die Zykluszeit problemlos möglich. Die Relation zur Durchlaufzeit ist indes nicht intuitiv zu fassen, wie das Beispiel in Abbildung 103 zeigt. Bei parallelen Vorgängen ist die Durchlaufzeit (hier: 12 Zeiteinheiten, ZE) häufig kürzer als die Brutto-Zykluszeit (14 ZE). Abhängig von der Länge der vermerkten Unterbrechungszeiten (5 ZE) kann die Netto-Zykluszeit kürzer (hier: 9 ZE) oder länger als die Durchlaufzeit ausfallen. Daher lohnt zunächst allenfalls die Betrachtung der nicht mit Prozessarbeit verbrachten Zeiträume (im Beispiel die Intervalle 1, 4, 8 und 12). Diese 4 ZE wurden nicht mit der Analysedurchführung zugebracht, aber möglicherweise für andere analyserelevante Tätigkeiten, etwa auf der Zielebene aufgewendet.

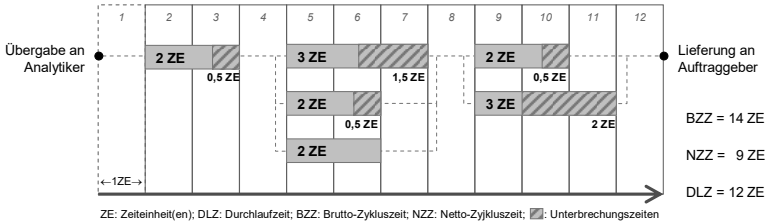


Abbildung 103: Symbolisches Beispiel zur Analyse unproduktiver Zeiten (eigene Darstellung)

Wie sich zeigt, ist zur Beurteilung von Datenanalyseprozessen in erster Linie die Durchlaufzeit von Interesse. Sie bemisst die zur Bereitstellung der Analyseergebnisse insgesamt benötigte Zeit und ist gegen die `Zeitrestriktion` der **`Informationsmaßnahme`** zu prüfen. Die Zykluszeit kann zur Kostenberechnung herangezogen werden, sofern einzelne Ressourcen nach tatsächlich beanspruchten Zeiten verrechnet werden (z.B. Server). In der Regel sind hier die Prozesszeiten einzelner Vorgänge zu berücksichtigen, soweit diese unterschiedliche Ressourcen binden. Personalkosten sollten auf Basis der Durchlaufzeit oder individueller Zeitaufschreibungen kalkuliert werden, falls einzelne Personen nur anteilig am Ablauf beteiligt sind.

Gestalterische Maßnahmen zur Prozessbeschleunigung richten sich meist auf die Struktur, die in Schritt K2.3 untersucht wird. Weitere Maßnahmen können sich aus der Untersuchung von Synchronisation und Lastverhalten ergeben. Nicht unmittelbar der Prozessarbeit dienende Zeiten entstehen zum großen Teil während der Planung und Steuerung der Prozesse, weshalb hier nur die methodische Unterstützung wirksam für Abhilfe sorgen kann.

Synchronisation

Unter Synchronisation wird der Gleichlauf zwischen mehreren Vorgängen bzw. deren zeitliche Abstimmung verstanden [Dude16d]. Im Kontext der Beurteilung von Prozessabläufen sind damit insbesondere der Parallelitätsgrad und der Wiederholungsgrad von Vorgängen gemeint. Sie lassen sich leicht untersuchen, indem Zeitpunkte und Häufigkeiten der Start- und Endereignisse der Vorgänge betrachtet werden. Überlappen die Ausführungszeitintervalle von Vorgängen, laufen diese (teilweise) parallel ab [Gait83, 164]. Aufgrund der damit verbundenen Verkürzung der Durchlaufzeit ist ein möglichst hoher Anteil parallel laufender Vorgänge erstrebenswert.

Vorgänge, die eine Wiederholung unveränderter Aktivitäten darstellen, treten nur bei Schleifen oder Fehlern auf. Sie sind am Wiederholungs-`Zähler`, der Bestandteil ihres Namens ist, erkennbar. Vorgänge, die eine Wiederholung von Aufgaben darstellen, repräsentieren Iterationen, z.B. infolge nicht adäquat instanziierter Operatorparameter (vgl. Abschnitt 4.5.3.1). Sie verweisen auf Planungs- oder Informationsmangel, die es zu untersuchen gilt. Aus dem Blickwinkel der zeitlichen Abstimmung sind Aufgabenwiederholungen daraufhin zu überprüfen, ob sie zu einer Aufgabe verschmolzen werden können, um Iterationen dadurch zu eliminieren.

Lastverhalten

Das Lastverhalten betrachtet die Leistungsfähigkeit der Aufgabenträger mit dem Ziel, die Prozesszeit verlängernde Engpässe zu erkennen. Im Vordergrund steht das Vermögen der Aufgabenträger, den fallspezifischen Mengen- und Reaktionsanforderungen zu genügen. Zur Quanti-

fizierung der Leistungsfähigkeit eignen sich Auslastungs-, Durchsatz- und Zeitmaße. Die *Auslastung* als genutzter Anteil der verfügbaren Kapazität [LJWK12, 1043] gibt Auskunft über die Geeignetheit eines maschinellen Aufgabenträgers (Servers) für die jeweilige Aufgabe. Erreicht die Auslastung die Kapazitätsgrenze, stellt die Ressource einen Engpass dar, der die Prozessausführung potenziell verlangsamt oder die Zuverlässigkeit der Vorgangsausführung beeinträchtigt (vgl. Abschnitt 7.2.2.1). Sie kann etwa als Prozessor- oder Hauptspeicherauslastung gemessen werden. Der *Durchsatz* repräsentiert die Verarbeitungsgeschwindigkeit, z.B. als verarbeitete Datenmenge pro Zeiteinheit, und gilt als primäres Kriterium der Leistungsfähigkeit einer Ressource [LJWK12, 1043]. Im operativen oder im Dialogbetrieb (z.B. bei Online-Scoring-Systemen bzw. interaktiver Visualisierung) kann auch die *Antwortzeit* (Zeitspanne von der Anfrage bis zur Antwort durch den Server) von Bedeutung sein [LJWK12, 1040f.].

Auf Basis der gemessenen Leistungswerte können die zur Auswahl von **`Server`** und **`Operator`** dokumentierten `Leistungsfaktoren` für jeden Prozessablauf fortgeschrieben werden (z.B. als gleitende Mittelwerte). Engpassressourcen sollten durch leistungsfähigere Alternativen ersetzt werden. Alternativ ist auch die Reduzierung der zu verarbeitenden Datenmenge (z.B. durch Nutzung von Stichproben) oder eine parallele Verarbeitung durch mehrere Ressourcen denkbar.

Ablaufkosten

Am Ende der Effizienzbetrachtung steht die monetäre Bewertung des Ressourcenverbrauchs mit Kostensätzen. Ihre Grundlage bilden die zuvor ermittelten Zeitdauern sowie Art und Menge der eingesetzten Ressourcen. Den größten Anteil stellen in der Regel die Personalkosten [DeHa01, 264]. Darüber hinaus fallen typischerweise Hard- und Software-Kosten (in Form von Verrechnungskostensätzen oder als nutzungsabhängige Entgelte) sowie Erhebungs-, Beschaffungs- oder Zugriffskosten für Datenquellen an [BeST99, 327-332], [WiFr00, 138]. Der jeweils geltende `Kostensatz` ist am Aufgabenträger hinterlegt. Als Bezugsgrößen treten meist Zeit- oder Volumeneinheiten auf (vgl. Abschnitt 4.6.3).

Für das beteiligte Personal wird grundsätzlich die tatsächliche Arbeitszeit angesetzt. Nutzungsabhängige Serverkosten können einfacher erfasst werden, da sie direkt von der beanspruchten Netto-Prozesszeit oder der verarbeiteten Datenmenge abhängen. Nicht zeit- oder mengenabhängige Ressourcenkosten (etwa für Software-Lizenzen) können z.B. auf Basis einer Prozesskostenrechnung behandelt werden, die Gemeinkostensätze verrechnet [ScSe04, 203].[254] Allgemein folgt die Kostenermittlung für ein Prozesselement der Formel (vgl. [Reif03, 177])

```
Vorgangskosten = fixe kosten
               + Kostenfaktor x Menge der Bezugsgröße.
```

Nach welchem konkreten Verfahren die Kosten berechnet werden, ist letztlich eine organisationsspezifische Entscheidung. Für den Kostenvergleich über mehrere Prozesse hinweg müssen die Berechnungsverfahren lediglich einheitlich sein.

7.2.2.3 Beurteilung der Struktur (K2.3)

Die Betrachtung der Struktur des Prozessablaufs geschieht anhand von Integrations- und Flexibilitätskriterien [Fers92, 11f.] (Tabelle 7). Die Integration wird mithilfe der Merkmale Redundanz, Verknüpfung, Konsistenz und Zielorientierung untersucht. Die Flexibilität betrifft Prozessvorlagen und wird anhand vorgenommener Abweichungen und Anpassungen beurteilt.

Redundanz

Das mehrfache Vorhandensein von Prozesselementen derart, dass einzelne Exemplare entfernt werden können, ohne die Funktionsfähigkeit des Prozesses zu beeinträchtigen, wird als *Redundanz* bezeichnet [FeSi13, 241]. Sie ist als Effizienzhemmnis grundsätzlich zu vermeiden, solange nicht triftige Gründe ihre Existenz rechtfertigen. Mit

[254] Hier stellt sich das Problem der korrekten Quantifizierung der Ressourcen-Inanspruchnahme. Zu Schwächen und Kritikpunkten an der Prozesskostenrechnung vgl. z.B. [ScSe04, 203].

mehreren Instanzen im Ablauf vertretene Aktivitäten und Aufgaben sind also danach zu untersuchen, ob solche Gründe vorliegen.

Kriterium	Bezugsobjekt	Fragestellung
Integrationskriterien		
Redundanz	Prozessinstanz	Sind Elemente mehrfach im Prozessablauf enthalten?
Verknüpfung	Prozessinstanz	Sind die Vorgänge sachlogisch sinnvoll, effizient und nachvollziehbar verknüpft?
Konsistenz	Prozessinstanz	Sind Vorgänge enthalten oder derart verknüpft, dass die Wirkung anderer Vorgänge kompensiert oder verfälscht wird?
Zielorientierung	Prozessinstanz	Sind überflüssige, nicht zielführende Elemente enthalten?
Flexibilitätskriterien		
Abweichungen	Prozessvorlagen; Prozessinstanz	An welchen Stellen und aus welchen Gründen musste vom geplanten Prozessschema abgewichen werden?
Anpassungen	Prozessvorlagen; Prozessinstanz	An welchen Stellen und aus welchen Gründen mussten die Elemente des Prozessschemas modifiziert werden?

Tabelle 7: Kriterien zur Beurteilung der Struktur des Analyseprozesses

Redundante Aktivitäten sind bezüglich aller Spezifikationselemente (einschließlich Operatorparameter sowie Ein- und Ausgabeflüsse) gleich und nur dann sinnvoll, wenn sie der Prozessbeschleunigung durch Parallelisierung von auf Instanzebene separierten Datenflüssen dienen (vgl. [FeSi13, 242]).[255] Redundante Aktivitäten stellen zugleich *redun-*

[255] Die Nutzung mehrerer redundanter Aufgabenträger zur Vorgangsbeschleunigung wird in der Datenanalyse typischerweise durch spezielle, die Parallelverarbeitung unter-

dante Aufgaben dar. Diese liegen auch dann vor, sobald mehrere unterschiedlich konfigurierte Aktivitäten vom selben Typ auftreten. Handelt es sich dabei um Wiederholungen (Iterationen), werden diese in Schritt K2.2 unter dem Synchronisationsaspekt aufgedeckt. Redundante Aufgaben können bewusst im Ablaufschema auftreten, z.B. wenn mehrere Attribute eine gleichartige Transformation (z.B. Diskretisierung) erfahren, oder mehrere Teildatenmengen (z.B. zur Modellevaluierung) gleichartige Datenvorbereitung erfordern. Im zweiten Fall ist die Beseitigung der Redundanz durch gemeinsame Bearbeitung aller Daten und nachträgliche Aufteilung möglich. Im ersten Fall bedingen die Fähigkeiten verfügbarer Operatoren häufig die Beibehaltung der Redundanz.

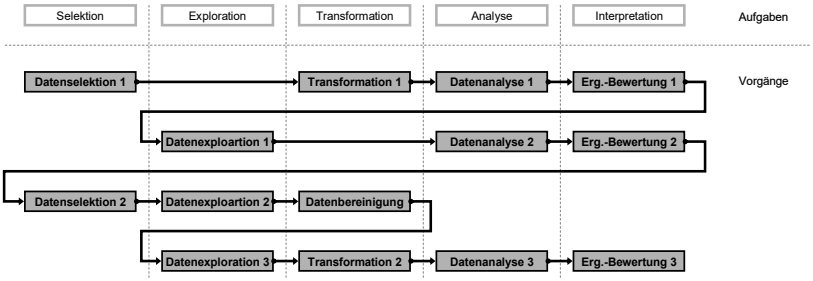


Abbildung 104: Beispiel zur visuellen Analyse von Redundanzen in Prozessabläufen (eigene Darstellung)

Die Untersuchung von Redundanzen lässt sich durch grafische Darstellungen unterstützen, die den Ablauf entlang bestimmter Aufgaben „einfalten“ und auf diese Weise deren mehrfache Existenz unmittelbar visualisieren. Die Vorgänge des Ablaufs können dabei den zugehörigen Aufgaben in vertikalen Spalten zugeordnet werden (Abbildung 104). Die zur Strukturierung dienenden Aufgaben können frei aus allen Ebenen der Funktionstaxonomie gewählt werden; Aufgaben der Blattebene stellen einen geeigneten Einstieg dar. Inwiefern typgleiche Aufgaben tatsächlich redundant sind, ist stets fallspezifisch zu entscheiden.

stützende Operatoren abgebildet, woraus keine redundanten Aktivitäten entstehen. Die Ressourcenebene ist daher nicht zu betrachten.

Verknüpfung

Art und Anzahl der Flussbeziehungen zwischen Prozesselementen können Effektivität, Effizienz und Komplexität des Prozesses beeinflussen. Sie sind dahingehend zu untersuchen, ob sie die Vorgänge sachlogisch sinnvoll, effizient und nachvollziehbar verknüpfen. Verbesserungspotenziale bestehen z.B. in gezielter Parallelisierung, in der Bildung von Schleifenstrukturen um wiederholte Aufgaben oder in der Vermeidung entbehrlicher Datenflussbeziehungen. Häufig werden Aufgabenträger durch unnötige Datentransfers stark ausgelastet, z.B. wenn umfangreiche Datenmengen vollständig durch mehrere Operatoren fließen. Sofern dies zur Bearbeitung nicht erforderlich ist, können Teildatenmengen selektiert oder Flussbeziehungen vermieden werden. Anzustreben ist eine für den Analytiker transparente und kontrollierbare Verknüpfungsstruktur [FeSi13, 242], welche die Interaktion zwischen Prozesselementen minimiert [LJWK12, 1044].

Konsistenz und Zielorientierung

Prozesselemente, die ein unerwünschtes Verhalten des Ablaufs hervorrufen oder nicht zur Zielerreichung der Analyseaufgabe beitragen, sind zu vermeiden bzw. anzupassen. Unter dem Aspekt der *Konsistenz* werden Aktivitäten untersucht, welche die Wirkung anderer Prozesselemente negativ beeinflussen oder kompensieren. Häufig werden dabei die Ursachen von während der Ergebnisinterpretation festgestellten Fehlern aufgedeckt, die etwa eine unbeabsichtigte Verfälschung des Aussagegehalts der Daten hervorrufen. Beispiele sind die Eliminierung wichtiger Detailaussagen durch Aggregation oder Ausreißerbeseitigung, oder die nicht zielführende Bereinigung von Datenfehlern (etwa das Auffüllen fehlender Merkmalswerte).

Unter dem Aspekt der *Zielorientierung* ist insbesondere zu überprüfen, ob nicht mit dem **`Analyseziel`** konforme Analyseaufgaben enthalten sind, d.h., ob eine *Zielabweichung* vorliegt (vgl. Zielkontrolle S3.3, Abschnitt 6.2.3.3). Resultiert aus einer beobachteten Zielabweichung vor dem Hintergrund des neuen Analyseziels ein effektiver Ablauf, kann dieser gegebenenfalls mit diesem neuen Ziel annotiert und archiviert werden. Andernfalls ist der Ablauf zu verwerfen.

Abweichungen und Anpassungen gegenüber der Prozessvorlage

Als Flexibilitätskriterien sind im Falle geplanter Prozesse ferner Abweichungen und Anpassungen bezüglich der verwendeten Prozessvorlagen (z.B. Workflow, Schablone) von Interesse. *Abweichungen* modifizieren die Prozessstruktur im Hinblick auf die Anordnung von Aufgaben und Flüssen, Anpassungen betreffen die verhaltenswirksame Parametrisierung (Konfiguration) von Aktivitäten. Sie sind jeweils aus dem Protokoll der `Änderungen` an der **`Prozessinstanz`** ersichtlich. Eine große Zahl solcher Änderungen verweist auf potenzielle Planungsmängel, kann aber auch in situativen Kontextbedingungen des aktuellen Falles begründet sein, die während der Planung nicht antizipierbar waren. Die Gründe der Modifikationen können ebenso dem Protokoll entnommen werden wie ihre Dauerhaftigkeit.[256] Anhand dieser Dokumentation kann entschieden werden, ob Änderungen genauer zu untersuchen sind und ob als dauerhaft markierte Modifikationen auf Typschemaebene zu übernehmen sind.

7.2.2.4 Realisierungsoptionen der Beurteilung des Prozessablaufs

Das Instanzmodell enthält detaillierte Angaben über den Prozessablauf und bildet die Grundlage der Beurteilung (vgl. Abschnitt 4.5.3). Die Struktur des Ablaufs lässt sich gut mithilfe *visueller Darstellungen*, wie in Abschnitt 7.2.2.3 exemplarisch gezeigt, untersuchen. Zur Erlangung eines Verständnisses seiner Verhaltenseigenschaften sind jedoch geeignete Hilfsmittel erforderlich, da die Vielzahl der im Prozessprotokoll enthaltenen Angaben nur schwer fassbar ist.

Eine umfassende Option ist der Aufbau eines *Performance Measurements*, das einen multidimensionalen Ansatz zur Messung und Steuerung der Leistung von Prozessen darstellt [ScSe04, 174].[257] Als Datengrundlage eignet sich ein *Prozess-Data-Warehouse* mit multidimensionaler Datenstruktur [Mueh01], [EdOG02], [GCC+04]. Dadurch können Prozess-

[256] Vgl. hierzu die Typvereinbarung des Datentyps `Änderungsoperation` in Anhang A4.6.

[257] Aus stärker technischer Perspektive sind ähnliche Ansätze auch unter Begriffen wie *Process Intelligence* [GCC+04] und *Process Analytics* [MüSh10] publiziert.

abläufe aus verschiedenen Perspektiven untersucht werden, z.B. im Hinblick auf die durchschnittliche Prozesszeit je Vorgang oder Aufgabe, die Charakteristika erfolgreicher oder gescheiterter Prozesse, die Kausalität beobachteter Verhaltenseigenschaften, sowie die Erkennung von Verbesserungspotenzial in der Prozessgestaltung und Ressourcenzuordnung [GCC+04, 322]. Ein auf den vorgestellten Modellierungsansatz abgestimmter Vorschlag für ein multidimensionales Datenschema mit einigen beispielhaften Analysefragen enthält Anhang A7.4.

Besonderen Wert gewinnen derlei Auswertungen bei Einbeziehung der Analyseergebnisse. So kann die Genauigkeit eines Prognosemodells gezielt auf Veränderungen in Abhängigkeit von Ablaufeigenschaften untersucht werden, z.B. im Hinblick auf Operatorparameterwerte, im Ablauf enthaltene Aufgaben oder genutzte Informationsobjekte. Beispielsweise lassen sich häufig verringerte Durchlaufzeit und höhere Genauigkeit bei Diskretisierung kontinuierlicher Merkmale beobachten. Erst mit der Beherrschung der Einflussfaktoren sind gezielte Prozessverbesserungen und Lerneffekte möglich (Handlungs- und Erfahrungsaspekt der Prozesskontrolle [ScSe04, 215]).

Als einfachere Variante ist ein systematisches *Reporting von Prozesskennzahlen* möglich, das verschiedene Parameter wie etwa mittlere Durchlaufzeiten oder Ressourcenauslastungen darstellt und Vergleiche zwischen Prozessinstanzen und ihren Elementen gestattet [JaBS97, 203ff.].

Eine weitere Option, die sich insbesondere für repetitive oder operative Analyseabläufe eignet, stellen *Prozessaudits* dar. Hierbei handelt es sich um systematische Überprüfungen vorgegebener Qualitätsstandards zum Zweck der Erkennung und Beseitigung von Mängeln oder dem Nachweis gegenüber dem Auftraggeber [ScSe04, 230-233]. Sie können durch nicht beteiligte Analytiker oder externe Experten vorgenommen werden. Unabhängige Audits sind auch in solchen Fällen empfehlenswert, in denen Analyseergebnisse als Grundlage für Entscheidungen dienen, die möglicherweise große Tragweite für ein Unternehmen haben [DeHa01, 233]. Sie sind in der Datenanalyse bislang nicht üblich, erscheinen angesichts der zunehmenden Bedeutung datenanalytisch gestützter Entscheidungen in allen Lebensbereichen aber hilfreich.

7.2.2.5 Zusammenfassung: Beurteilung des Prozessablaufs

Die Beurteilung des Prozessablaufs ist ein Instrument zur systematischen Prozessverbesserung, das drei Kriterienklassen unterscheidet. Die Untersuchung aus Effektivitätssicht richtet sich auf Fehler und Ursachen nicht befriedigender Analyseergebnisse. Sie ist für alle Abläufe ratsam. Die Untersuchung aus Effizienz- und Struktursicht ist insbesondere für Abläufe interessant, die wiederholt durchgeführt werden oder auf einer Prozessvorlage basieren. Für sie lohnt die Suche nach Verbesserungspotenzialen, die unmittelbar die Prozessschemata betreffen.

Die Resultate der Effizienz- und Strukturbeurteilung können analog zur Ergebnisbewertung (K2.1) in einem strukturierten Schema als `Prozessbewertung` an der **`Prozessinstanz`** abgelegt werden. Zusammen mit entsprechenden `Kommentaren` lassen sich auf diese Weise Verbesserungsbedarf und -vorschläge nachvollziehbar am betreffenden Ablauf dokumentieren.

7.3 Ganzheitliche Evaluierung des Analyseprojekts

Da Datenanalysen in der Regel zum Zweck der Unterstützung betrieblicher Entscheidungen oder Problemlösungen erfolgen, ist eine *Prozessrevision i.w.S.* unter Einbeziehung der Anwendungsebene sinnvoll. Dies geschieht nach Realisierung der zugehörigen Handlungsmaßnahmen (Anwendung des Wissens, vgl. Vorgehensmodell in Abschnitt 3.3.3). Die ganzheitliche Beurteilung des analytisch gestützten Projekts zielt auf Beantwortung der Frage, inwiefern die Datenanalysen und die resultierenden Interventionen den gewünschten Beitrag zur Lösung des Sachproblems leisten und ob die erlangten Wirkungen den insgesamt entstandenen Aufwand rechtfertigen.

Die Reichweite der Evaluierung ist abhängig von der Struktur des Sachproblems und seiner Fortschreibung innerhalb der Problemkarte. Zunächst wird jeder Problemaspekt als Einzelprojekt betrachtet, das alle zugehörigen Informations- und Handlungsmaßnahmen umschließt. Danach können alle zum übergeordneten Vater-Problemaspekt gehörigen Kind-Aspekte zu einem Projekt vereinigt werden. Diese Navi-

gation im Problemgefüge wird solange fortgesetzt, bis alle sachlogisch verbundenen Problemaspekte erfasst sind. Letztlich lässt sich jeder Problemaspekt sowohl als eigenständiges Projekt als auch im übergeordneten Kontext betrachten.

Kommen durch Fortschreibung der Problemkarte weitere Problemaspekte hinzu, kann die Beurteilung des Gesamtprojekts auf Grundlage der Evaluierungsergebnisse der neuen Teilprojekte aktualisiert werden. Da die hierbei entstehenden Gesamtprojekte überaus komplexe Strukturen aufweisen und sich über Monate oder Jahre erstrecken können, ist häufig, auch aus sachlogischer Sicht, die Evaluierung abgeschlossener Teil- oder Einzelprojekte sinnvoller.

Die ganzheitliche Beurteilung des analytisch unterstützten Projekts umfasst mehrere Teilaufgaben. Zunächst ist für jede Informationsmaßnahme eine Kostenanalyse zu erstellen. Wurde die Information datenanalytisch erzeugt, geschieht dies im Rahmen der Beurteilung des Analyseablaufs (K2, vgl. Abschnitt 7.2.2). Danach ist die *Evaluation der Handlungsmaßnahmen (K3)* für jede realisierte Lösungsoption durchzuführen. Abschließend erfolgt für jedes (Teil-) Projekt eine *Nutzen-Kosten-Analyse (K4)*. Die Aufgaben K3 und K4 werden im Folgenden erläutert.

7.3.1 Evaluation der Handlungsmaßnahmen (K3)

Ziel dieser Aufgabe ist die Feststellung des Erfolgs der infolge einer Datenanalyse getroffenen Entscheidung oder Handlungsmaßnahme. Ergebnis ist eine Bewertung ihres Zielerreichungsgrades. In der Regel ergeben sich dabei Erkenntnisse über die Angemessenheit der Entscheidungsfindung bzw. Maßnahmenplanung, die zu Nachbesserungen, Wiederholungen oder neuen Maßnahmen führen können (vgl. [Knob03a, 342]). Diese sind in der Problemkarte festzuhalten.

Die Bewertung erfolgt durch Gegenüberstellung der realisierten Effekte mit den gesetzten Zielen [BeLi97, 28f.]. Dazu werden die erreichten Zustände des Problemobjekts betrachtet und mit den Vorgaben aus dem zugehörigen lösungsbezogenen **`Problemaspekt`** (Typ Soll) verglichen,

wie sie in den Attributen `Inhalt`, `Ausmaß`, `Zeitbezug` und `Wertbeitrag` zum Ausdruck kommen.

Die Erfüllung dieser Ziele (z.B. die Steigerung des Bonbetrags auf das vorgegebene Niveau bis Dezember 2010) lässt sich direkt datenanalytisch überprüfen. Hierbei kommt der Zeitrestriktion oft besondere Bedeutung zu: Wird der Sollzustand zwar erreicht, tritt jedoch zu spät ein, wurde das Ziel dennoch verfehlt. Zusätzlich sollte der Wertbeitrag der Maßnahme kontrolliert werden, indem empirische Daten zum definierten Wertkriterium (z.B. Umsatz) analysiert werden (etwa, ob durch die Steigerung des durchschnittlichen Bonbetrags auch insgesamt eine Erhöhung des Umsatzes zu beobachten ist).

Die Evaluation der Handlungsmaßnahmen führt somit im Grunde zu einem neuen Problemaspekt, der durch eine oder häufig auch mehrere Informationsmaßnahmen zu lösen ist. Die Planung der zugehörigen Datenanalysen erfolgt gemäß der in Kapitel 5 vorgestellten Methodik. Die nächsten beiden Abschnitte steuern hierzu Hinweise aus Sicht der Evaluationsforschung bei.

7.3.1.1 Systematische Evaluation

Evaluation bezeichnet allgemein die Bewertung eines Sachverhalts anhand einer Reihe von Merkmalen, ohne eine Vorgehensweise vorzugeben. Insbesondere sind Merkmalsraum und Verknüpfung der Merkmale zur Berechnung einer Gesamtbewertung unbestimmt. Demgegenüber steht die ***Evaluationsforschung*** (systematische Evaluation), die eine theoretisch fundierte Bewertung von Effektivität (Zielerreichungsgrad) und Effizienz eines Sachverhalts unter Verwendung wissenschaftlicher Methoden darstellt. Sie nutzt zuvor bestimmte Bewertungskriterien, die auf definierte Weise zu einer Gesamtbewertung aggregiert werden. Die wissenschaftliche Vorgehensweise soll die Validität der Bewertung gewährleisten [GoJä09, 22f.], [SpGL10, 223f.].

Die vorgestellte Methodik unterstützt die systematische Evaluation, indem die vom Problemaspekt definierten Bewertungskriterien `Inhalt`, `Ausmaß`, `Zeitbezug` und `Wertbeitrag` analog zum Vorgehen in Schritt K2.1 (Abschnitt 7.2.2.1) in einem Bewertungsschema verknüpft

werden, um daraus eine Gesamtbewertung mit Zielerreichungsgrad abzuleiten. Diese primären Kriterien können kontextabhängig durch weitere Kriterien ergänzt werden. Die Evaluationsresultate werden als `Bewertung` an der **`Maßnahme`** dokumentiert. Das binäre Attribut `Erfolg` erlaubt zusätzlich, die Maßnahme in ihrer Gesamtheit als erfolgreich oder als nicht erfolgreich zu kennzeichnen.

Die DEUTSCHE GESELLSCHAFT FÜR EVALUATION (DEGEVAL) hat Evaluationsstandards veröffentlicht, die eine Orientierung für ein Vorgehen nach den aktuellen Regeln der Kunst bieten [DeGE08]. Weitere Ausführungen zur Evaluationsforschung, zu Methodik und Standards enthält die einschlägige Literatur, z.B. in [GoJä09], [Döri10], [SLGR10], [Soel10].

7.3.1.2 Wirksamkeit und Wirkung

Zur Bestätigung der *Wirksamkeit* einer Maßnahme genügt es, wenn die angestrebten Effekte empirisch nachweisbar sind. Dies kann mithilfe einer *Veränderungsevaluation* geschehen, die die Zustände vor und nach der Intervention vergleicht. Der Nachweis, dass diese Veränderung tatsächlich auf die *Wirkung* der Maßnahme zurückgeht, ist auf Grundlage von Ex-post-facto-Ansätzen wie Veränderungsanalysen jedoch nicht valide zu führen. Hierzu ist ein sogenanntes Wirkmodell erforderlich, in dem Kausalhypothesen über die angenommenen Wirkmechanismen formuliert werden [GoJä09, 80-83]. Einen Ansatz zur methodischen Überprüfung von Kausalhypothesen präsentiert VOIT [Voit10] (vgl. hierzu auch Abschnitt 5.4.4.3). Alternativ lässt sich die Wirkung einer Maßnahme durch Vergleich mit einer Kontrollgruppe evaluieren, die unter sonst gleichen Bedingungen nicht der Intervention ausgesetzt war [BeLi97, 111], [GoJä09, 100].

Eine fundierte *Wirksamkeitsevaluation* ist eine konfirmatorische Analyse, die wenigstens die folgenden Fragen beantworten soll [GoJä09, 80]:

- Sind die erwarteten/gewünschten Veränderungen tatsächlich eingetreten?
- Sind die Veränderungen tatsächlich auf die Maßnahme zurückzuführen (Kausalität)?

Abhängig vom Problemkontext können zusätzlich folgende Fragestellungen von Interesse sein:

- Sind die Effekte der Intervention persistent und nachhaltig?
- Sind zusätzlich zur angestrebten Wirkung auch Neben- oder Folgewirkungen aufgetreten?
- Können die beobachteten Effekte auf andere Situationen generalisiert werden (Robustheit)?

Zur Behandlung derartiger Fragen ist die Problemkarte häufig zu erweitern. So kann etwa die Kontrolle der Persistenz mehrere, gegebenenfalls auch regelmäßige Untersuchungen erfordern, die eigene Problemaspekte darstellen. Auch die Problematik der Neben- und Folgewirkungen kann sich komplex gestalten und eine Differenzierung in mehrere Problemaspekte nahelegen.

Tritt eine Wirkung nicht wie gewünscht ein, kann dies verschiedene Ursachen haben, die Anlass zu systematischen Untersuchungen geben können. Grundsätzlich sind Fehler bezüglich grundlegender Annahmen (z.B. langjährige Kunden generieren mehr Umsatz), bezüglich der Kausalhypothesen, auf denen die Wirkmechanismen der Maßnahme beruhen (z.B. ein Programm zur Freundschaftswerbung verlängert die Kundenbeziehung), bezüglich der Konzipierung bzw. Realisierung der Maßnahme sowie bezüglich der Evaluation denkbar [GoJä09, 83f.].

7.3.2 Nutzen-Kosten-Analyse (K4)

Ziel der Nutzen-Kosten-Analyse ist die Beurteilung der Effizienz der ergriffenen Maßnahmen hinsichtlich des Nutzens ihrer Wirkung in Relation zu den entstandenen Kosten [GoJä09, 96]. Die Bewertung berücksichtigt die mögliche Schachtelung von Projekten, indem jeweils zunächst Einzelprojekte evaluiert werden, um die Einzelbewertungen sodann für übergeordnete Projekte zu konsolidieren. Die abschließende Gesamtbewertung steht demnach erst nach Beendigung aller Teilprojekte zur Verfügung [DeHa01, 72]. Das Ergebnis der Nutzen-Kosten-Analyse einzelner Maßnahmen muss im Sinne der systematischen

Evaluation (vgl. Abschnitt 7.3.1.1) als Kriterium der `Bewertung` berücksichtigt werden und beeinflusst somit den Zielerreichungsgrad.

Die Bewertung kann auch ausschließlich anhand der Kosten geschehen, erfolgt nach Möglichkeit aber im Sinne einer Effizienzanalyse unter Berücksichtigung des realisierten Nutzens [GoJä09, 97]. Hierzu ist die folgende Aufgabenreihe auszuführen. Zunächst erfolgt die *Ermittlung der Kosten (K4.1)* für jede Informations- und Handlungsmaßnahme, im Anschluss folgt die *Quantifizierung des Nutzens (K4.2)* für jede Handlungsmaßnahme. Beide Werte werden sodann in der *Effizienzanalyse (K4.3)* gegenübergestellt.

7.3.2.1 Ermittlung der Kosten (K4.1)

Die Ermittlung der Kosten für Datenanalyseabläufe erläutert Abschnitt 7.2.2.2 im Rahmen der Effizienzbeurteilung (K2.2). Sie sind in diesem Schritt als Kosten der Informationsmaßnahme anzusetzen und bei Bedarf um weitere Positionen zu ergänzen. Informationsmaßnahmen anderer Ausprägung (z.B. empirische Studien, Experimente, etc.) sowie Handlungsmaßnahmen erfahren eine Kostenbewertung nach folgenden Grundsätzen.

Bei der Berechnung des Aufwands kommen alle *manifesten Kosten* zum Ansatz, die tatsächlich (z.B. in Form von Zahlungen zwischen Auftraggeber und -nehmer) entstanden sind. Hierzu zählen insbesondere die Kosten für Planung und Durchführung der Maßnahmen, wie etwa Personal-, Material-, Administrations- und Gemeinkosten [WeZS08, 129f.], [GoJä09, 98]. Bei Entscheidungen zwischen alternativen Maßnahmen können auch *Opportunitätskosten* Berücksichtigung finden. Sie quantifizieren die Einbußen die entstehen, wenn man infolge der Entscheidung auf die Vorteile der einen Option verzichtet, andererseits aber die Nachteile der anderen Intervention in Kauf nimmt [GoJä09, 99]. *Latente Kosten*, die meist nicht direkt belegt werden können, sollten angesetzt werden, sofern sie mit ausreichender Sicherheit quantifizierbar sind. Beispiele sind von anderen Kostenträgern übernommene Positionen [GoJä09, 98]. Die Bewertung aller Kostenkategorien sollte stets zu marktüblichen Preisen erfolgen, selbst wenn im Einzelfall niedrigere Beträge anfallen. Auf diese Weise ist die Evaluation bedin-

gungsunabhängig und mit Maßnahmen vergleichbar, in denen diese Sonderbedingungen nicht gelten [GoJä09, 99].

Nach Feststellung der Gesamtkosten ist zu prüfen, ob sie der in der **Maßnahme** gesetzten `Budgetrestriktion` entsprechen. Eine Überschreitung führt entsprechend zu einer negativen Bewertung unter diesem Kriterium.

7.3.2.2 Quantifizierung des Nutzens (K4.2)

Der Nutzen einer Maßnahme lässt sich grundsätzlich durch Bewertung der in Schritt K3 (Abschnitt 7.3.1) gemessenen Wirkung bestimmen [GoJä09, 99]: `Nutzen = Wirkung` × `Wert`. Typischerweise wird der Wert nach realisierten Gewinnen, Umsatzsteigerungen, Kostensenkungen etc. bemessen. Bei präventiven Maßnahmen wie der Betrugserkennung steht der abgewendete Schaden im Vordergrund.[258] Der Wert kann auch abstrakter Natur sein und den subjektiven Nutzen der Wirkung aus Sicht des Unternehmens oder der Betroffenen darstellen, wie z.B. die Erlangung von Wettbewerbsvorteilen oder einer Vorreiterrolle [BeST99, 324-327], [GoJä09, 100], womit eine Quantifizierung weitaus schwerer fällt. Bei Bedarf können mehrere interessengruppenspezifische Nutzenbewertungen durchgeführt und entweder einzeln interpretiert oder zu einer Nutzenfunktion aggregiert werden [GoJä09, 103f.].

Die Bewertung erfolgt prinzipiell durch den Vergleich des `Ausmaßes` des Problemzustands unter der Bedingung, dass die Maßnahme durchgeführt wird mit dem Ausmaß unter der Bedingung, dass die Intervention unterbleibt. Der Gesamtnutzen ergibt sich dann durch Differenzbildung der Zustände unter den beiden Bedingungen [GoJä09, 100]. Demnach beträgt z.B. der Gesamtnutzen eines Systems zur Betrugserkennung, das die Betrugsquote von 15% auf 5% reduziert, bei angenommenen Kosten von 1.000 EUR je Betrugsdelikt und 100 Trans-

[258] In vielen Fällen gibt der `Wertbeitrag` des Problemaspekts Hinweise auf geeignete Bewertungsgrößen. Wie am Beispiel Betrugserkennung deutlich wird, gilt dies jedoch nicht immer unmittelbar.

aktionen pro Tag, (100 × 15% × 1.000 EUR) – (100 × 5% × 1.000 EUR) = 15.000 EUR – 5.000 EUR = 10.000 EUR.

Zuweilen erweist sich die Quantifizierung des Nutzens überaus schwierig, wie etwa die Abschätzung des Wertes eines neu gewonnenen Kunden [BeLi97, 109-111]. Hierzu können eigene Datenanalysen hilfreich sein. Werden Prognosemodelle eingesetzt, sollte die Ex-post-Nutzenbewertung analog zu den in Schritt K1.2 ex ante angestellten Berechnungen erfolgen.

7.3.2.3 Effizienzanalyse (K4.3)

Sind Kosten und Nutzen für jede Maßnahme bekannt, können diese Größen im Rahmen einer *Effizienzanalyse* zusammengeführt werden. Hierzu bestehen folgende Optionen [GoJä09, 104f.]:

- **Nettonutzen = Nutzen – Kosten**: Dieses absolute Nutzenmaß ist zur Bewertung einzelner Maßnahmen geeignet. Zum Vergleich mehrerer Maßnahmen ist es nur dann zu empfehlen, wenn Kosten und Nutzen aller Maßnahmen gleich skaliert und inhaltlich vergleichbar sind.
- **Nutzenquotient = Nutzen / Kosten**: Dieses relative Maß eignet sich für den Vergleich von Maßnahmen auch dann, wenn deren Kosten- und Nutzenbedingungen voneinander abweichen.
- **Profitrate = Nettonutzen / Nutzen**: Dieser Index erlaubt ebenfalls den maßnahmenübergreifenden Vergleich unter abweichenden Bedingungen, ist aber zusätzlich standardisiert.

Am Beispiel einer Direktmarketingkampagne ergibt sich der Nettonutzen aus Gegenüberstellung (a) der Fixkosten für die Konzipierung des Klassifikationsmodells (Datenanalyse) und der Kampagne, (b) der Kosten pro Empfänger der Werbesendung oder des Angebots (einschließlich Rabatte oder Zugaben, z.B. subventionierte Endgeräte wie im Mobilfunkbereich üblich), (c) der Kosten pro positiver Antwort (Kauf) für Auftragsbearbeitung und Warenversand einerseits und (d) des Auftragswerts einer positiven Antwort andererseits [BeLi97, 110].

Ist eine Effizienzanalyse aufgrund nicht quantifizierbarer Nutzenmaße nicht möglich, bietet sich eine *Kosten-Effektivitätsanalyse* an. Hierbei werden die Kosten jeweils für eine von der Wirkung betroffene Bezugseinheit berechnet. Die resultierenden Effektivkosten pro Einheit sind leicht interpretierbar und maßnahmenübergreifend vergleichbar. Als Bezugseinheiten eignen sich im Falle binärer Wirksamkeitskriterien die betroffenen Domänenobjekte. Beispielsweise können die Kosten für jede als Betrug identifizierte Transaktion bestimmt werden. Im Falle metrischer Wirksamkeitskriterien lassen sich die Kosten je Veränderungseinheit berechnen, wie z.B. Kosten pro Euro Mehrumsatz [GoJä09, 105f.].

7.3.3 Zusammenfassung: Ganzheitliche Evaluierung des Analyseprojekts

Die Evaluierung des Analyseprojekts strebt eine ganzheitliche Erfolgsbeurteilung aller im Zusammenhang eines Sachproblems oder Problemaspekts stehenden Maßnahmen an, um die korrekte Zurechnung des gesamten Aufwands (einschließlich Ergründung des Problems sowie Entwicklung und Anwendung von Interventionen) zur realisierten Lösung zu erreichen.

Hierzu sind die Bewertungen aller Einzelmaßnahmen innerhalb der Reichweite des betrachteten Projekts zu einer Gesamtbewertung zu aggregieren.[259] Häufig ist es angebracht, zur Evaluation einen längeren Zeitraum zu beobachten, etwa wenn die Handlungsmaßnahme die Implementierung eines Anwendungssystems oder einer längerfristig angelegten Marketing-Kampagne vorsieht. Ein positiver Nutzen ist hier oft erst nach einer gewissen Amortisationsdauer zu erwarten [AlGr12]. Weiterhin können Maßnahmenkosten einer zeitlichen Dynamik

[259] Wird eine Effektivitätsanalyse durchgeführt, ist der Nutzen des Gesamtprojekts nicht als Differenz quantifizierbar. Die Einbeziehung der Datenanalysekosten in die Effektivkosten ist kaum empfehlenswert, da sie den einzelnen Wirkungseinheiten ähnlich wie bei einer Volkostenrechnung nicht mehr sinnvoll zurechenbar sind [GaFi92, 57]. Die entstehenden Kostensätze sind inhaltlich schwer interpretierbar. Daher scheint es in diesem Fall besser, die Summe der Analysekosten und die Stückkosten je Nutzeneinheit getrennt auszuweisen.

unterliegen, wenn etwa Aktualisierungs- oder Wartungskosten anfallen [GoJä09, 101].

Häufig ist die Berücksichtigung spezifischer Nutzen- und Kostenfaktoren hilfreich. Als Beispiele seien die in Abschnitt 7.2.1.2 für Prognosemodelle beschriebenen klassenspezifischen Nutzen-/Kostenfaktoren, Kosten für Fehlentscheidungen [WeSZ08, 132f.] sowie Kosten von Neben- und Folgewirkungen genannt [GoJä09, 101]. Während der Evaluation gewonnene Erkenntnisse können zur Verbesserung künftiger Vorhaben beitragen und Anlass für weitere Analysen geben.

7.4 Erfahrungssicherung und Prozessverbesserung

Die während der Durchführung und Revision bisher gewonnenen Erkenntnisse über Stärken und Schwächen eines analytisch gestützten Projekts stellen wertvolles Wissen dar, das hilfreiche Beiträge zur Planung und Realisierung künftiger Vorhaben leisten kann und ein lernendes System zur Projekt- und Prozessverbesserung ermöglicht. Die Bereitstellung dieses Wissens zur intersubjektiven und projektübergreifenden Nutzung erfordert die systematische Akquisition, Ordnung und Speicherung, denen zwei Aufgaben im Handlungsschema gewidmet sind.

Im ersten Schritt ist zu entscheiden, inwiefern die dem durchgeführten Ablauf zugrundeliegenden bzw. daraus abgeleiteten Schemata angesichts der Beurteilungsergebnisse vor ihrer Speicherung zu verbessern sind. Diese *Modifikation der Analysepläne (K5)* sichert die unmittelbar aus der Prozessdurchführung resultierenden Erfahrungen. Im zweiten Schritt erfolgt die *Extraktion wiederverwendbaren Wissens (K6)*, das eher mittelbar aus dem Prozessablauf zu ziehen ist und sich in Gestalt von allgemeinen Hinweisen, konkreten Regeln, Prozessbausteinen und Planungsartefakten äußert.

7.4.1 Modifikation der Analysepläne (K5)

Ziel dieser Aufgabe ist die Überarbeitung des durchgeführten Analyseprozesses vor dem Hintergrund zuvor erkannter Mängel und Verbesserungspotenziale. Ihr Ergebnis sind verbesserte Prozesspläne, die zur

Wiederverwendung in der Prozessbibliothek gespeichert werden können.[260] Der Modifikationsbedarf resultiert grundsätzlich aus konkreten Beurteilungskriterien der Revision. Inwiefern ihm entsprochen wird ist abhängig von der Relevanz des Schemas zu entscheiden. Für häufig genutzte Vorlagen lohnt die Behebung von Effizienz- und Flexibilitätsmängeln. In anderen Fällen mag eine Konzentration auf Effektivitätsmängel (Fehler) genügen.

Die ganzheitliche Prozessverbesserung betrachtet alle Ebenen der Datenanalysearchitektur. Dazu werden zwei Teilaufgaben definiert, die zunächst *Modifikationen auf Prozessebene (K5.1)* behandeln, um anschließend zu überprüfen, inwiefern auch Bedarf für *Modifikationen auf Ziel- und Ressourcenebene (K5.2)* besteht.

7.4.1.1 Modifikationen auf Prozessebene (K5.1)

Die Modifikationen werden zwar durch Betrachtung der Instanzebene (Prozessablauf) motiviert, jedoch auf Schemaebene vorgenommen. Wurde der Ablauf ohne Vorlage innovativ gestaltet, ist das zugehörige Workflow-Schema aus dem Instanzmodell ableitbar. Abweichungen und Anpassungen, die während des Ablaufs gegenüber einer Vorlage erfolgt sind und auch für künftige Abläufe als relevant erachtet werden, können auf Schemaebene propagiert werden. Schemaänderungen reflektieren die ***Evolution*** von Prozessen als revisionsbezogener Anwendungsfall der Prozessgestaltung (vgl. Abschnitt 5.1.4). Sie führen jeweils zu einer neuen Schemaversion [WRWR05b, 4].[261] Hierbei kann die alte Version entweder durch die neue Version ersetzt werden (Substitution), oder sie existiert neben der neuen Version weiter (Erweiterung). Die zugehörigen Entscheidungen werden durch die Historie aller an der

[260] Die Aufgabe ist demnach dem „development for reuse“ zuzuordnen.

[261] Im Zusammenhang mit der Schemaevolution diskutiert die Literatur auch die Frage der Anwendung von Schemaänderungen auf laufende Instanzen (Migration, vgl. [DRRA05, 6], [RWRW05, 252], [RiDa03, 17f.]). Dieses Problem ist für die Datenanalyse im Allgemeinen irrelevant, da hier im Gegensatz zu operativen Workflows typischerweise jeweils nur eine Instanz eines Prozesses aktiv ist bzw. die Propagierung der Änderung auf andere Instanzen aufgrund der Varietät der Abläufe nicht angebracht erscheint.

Prozessinstanz vorgenommenen Änderungen unterstützt [Gier00, 213f.], [DRRA05, 7]. Für jede Änderung werden Typ (z.B. Einfügung), Änderungsobjekt (z.B. einzufügende Aktivität) und Parameter (z.B. Position) festgehalten [RWRW05, 255]. Zusätzlich kann eine Begründung hinterlegt werden, die z.B. eingetretene Ausnahmen oder Fehler dokumentiert, auf die mit der Änderung reagiert wurde. Durch das Kennzeichen zur Dauerhaftigkeit können Anpassungen, die nur lokal für die bearbeitete Prozessinstanz gelten sollen, schon während ihrer Vornahme zur Ausführungszeit als solche markiert und für die Evolution ignoriert werden.

Die Schemaevolution startet vorzugsweise mit Strukturmodifikationen. Dadurch können zunächst zur Ausführungszeit erfolgte Deviationen Berücksichtigung finden, die aufgrund ihres direkten Bezugs zur Zielerreichung im Allgemeinen die höchste Priorität besitzen. Sodann werden im Rahmen der Ablaufbeurteilung erkannte strukturelle Schwächen beseitigt. Im Anschluss folgen Verhaltensmodifikationen.

Zur weiteren Unterstützung des Prozessgestalters bestehen mehrere Optionen. Zunächst erscheint eine visuelle Gegenüberstellung der Prozessvorlage bzw. des Ablaufs mit dem zu bearbeitenden Zielschema sinnvoll. Zur Ausführungszeit veränderte Bausteine des Ausgangsschemas können zudem (z.B. farbig) markiert werden, um eventuellen Modifikationsbedarf direkt an den betroffenen Elementen zu verorten. Darüber hinaus ist eine diagrammatische Darstellung der Genese der Bausteine bzw. der Schema-Versionen („Stammbaum“) denkbar, aus der ersichtlich ist, welche weiteren Vorlagen bzw. Versionen möglicherweise ebenfalls von der infrage stehenden Modifikation profitieren, weil das betrachtete Schema aus ihnen hervorgegangen ist.

Im Folgenden wird exemplarisch die grafische Unterstützung struktureller Modifikationen gezeigt. Hierbei wird der in Abschnitt 7.2.2.3 (K2.3) auf Redundanzen untersuchte Ablauf[262] zu einem effizienteren Schema verkürzt (Abbildung 105). Die spaltenorientierte Zuordnung der Vorgänge zu Aufgaben erleichtert die Selektion jener mehrfach ausgeführten Explorations-, Analyse- und Interpretationsaufgaben, die aus

[262] Vgl. hierzu die dortige Abbildung 104 (Seite 352).

dem Ablauf gelöscht werden können (oben, 1). Hierbei ist zu beachten, dass nicht zwingend alle gleichartigen Aktivitäten überflüssig sind. So ist Datenexploration 2 im Beispiel auf die Datenbereinigung gerichtet und somit inhaltlich verschieden von Datenexploration 3, die sich auf Transformation 2 richtet.[263] Die Abbildung zeigt (symbolisiert durch Pfeile) weiterhin die Verschmelzung typgleicher Aktivitäten, die z.B. dann sinnvoll ist, wenn diese verschiedene Informationsobjekte verarbeiten. Die Spezifikation der resultierenden Aktivität ist dabei derart zu modifizieren, dass sie die Ziele aller ursprünglichen Aufgaben vereint.

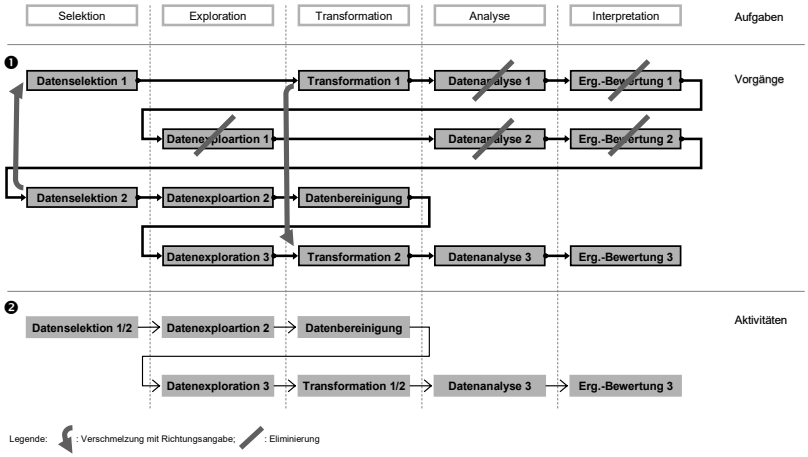


Abbildung 105: Beispiel zur Eliminierung redundanter Aktivitäten aus einem Prozessablauf (eigene Darstellung)

Im Beispiel werden Datenselektion 1 und Datenselektion 2 verschmolzen, indem z.B. die durch Datenselektion 2 aus der Datenquelle extrahierten Attribute in die erste Selektionsaufgabe integriert werden. Analog erfolgt eine Verschmelzung der beiden Transformationen 1 und

[263] Diese Feststellung bedeutet indes nicht, dass der Ablauf im Hinblick auf beide Aktivitäten nicht im Zuge anderer Modifikationen – etwa einer Verschmelzung zu einer verallgemeinerten Explorationsaufgabe, die sich gleichzeitig Datenbereinigung und Transformation widmet – dennoch weiter vereinfacht werden kann.

2 zu einem gemeinsamen Schritt. Die Abbildung zeigt unten (2) das aus Eliminierung und Verschmelzung resultierende, verkürzte Schema.

Neben den gezeigten Modifikationen sind die Ergänzung von Aktivitäten, die Änderung der Operatorzuordnung oder die Umstrukturierung von Flussbeziehungen (z.B. Parallelisierung) häufig auftretende Maßnahmen. Beispielsweise können die Iterationen zwischen Datenanalyse und Ergebnisbewertung in Abbildung 105 durch eine Schleife (Sprung) von Ergebnisbewertung 3 zur Datenanalyse 3 modelliert werden, falls im gegebenen Kontext regelmäßig mit mangelnder Ergebnisqualität zu rechnen ist, die durch Wiederholung der Analyse zu beheben ist. Das resultierende Schema erlangt dadurch höhere Flexibilität und Robustheit gegenüber fallspezifischen Störungen. Verhaltensbezogene Änderungen betreffen die Konfiguration der Operatoren, d.h. die Anpassung der Makro- oder Mikroparametrisierung der Aktivitäten. Sie können großen Einfluss auf die Effektivität und Effizienz des Ablaufs ausüben.

Bei der Prozessverbesserung ist im Allgemeinen mit Zielkonflikten zu rechnen, d.h., dass manche Modifikationen sich zwar positiv auf einen Prozessparameter auswirken, aber zugleich andere Kriterien negativ beeinflussen. So ist die Verringerung der Prozesszeiten durch leistungsfähigere Aufgabenträger (Operatoren) häufig mit Kostensteigerungen verbunden [ScSe04, 180].

7.4.1.2 Modifikationen auf Ziel- und Ressourcenebene (K5.2)

Wie bereits bei der Ablaufbeurteilung muss sich auch die Prozessverbesserung auf die Zielebene der Datenanalysearchitektur fortpflanzen. Zusätzlich wird auch die Ressourcenebene betrachtet. Die Ausdehnung der Betrachtung trägt den starken Abhängigkeiten und Interdependenzen zwischen den Ebenen Rechnung. In vielen Fällen lässt sich ein fehlerhaft oder nicht adäquat konzipiertes Element auf Gestaltungsentscheidungen zurückführen, die auf einer höheren Ebene der Architektur getroffen wurden. Ebenso kann eine Modifikation die Notwendigkeit abgestimmter Anpassungen von Elementen auf anderen Ebenen nach sich ziehen.

Zur Identifizierung von Modifikationsbedarf sind zwei grundlegende Ansätze zu unterscheiden:

- **Rückwärtsverfolgung der Abhängigkeiten**, ausgehend von der *Prozessebene* nach oben in der Analysearchitektur (bottom-up): Dieser Ansatz wird nach Schritt K5.1 gewählt, um Änderungen auf die Zielebene zu propagieren oder um Ursachen festgestellter Mängel auf höheren Architekturebenen zu suchen und zu beheben.
- **Vorwärtsverfolgung der Abhängigkeiten** nach unten in der Analysearchitektur (top-down): Dieser Ansatz kann einerseits von der *Anwendungsebene* ausgehen, wenn die Evaluation der Handlungsmaßnahmen (K3) nicht befriedigend ausfällt und diesbezüglich ergriffene Modifikationen auf tiefer liegende Ebenen zu propagieren sind. Andererseits kann ausgehend von der *Prozessebene* die Anpassung der Ressourcenkonfiguration oder die Fortschreibung der dokumentierten Datenquellen und Informationsobjekte erfolgen.

Die Verfolgung der Abhängigkeiten läuft in beide Richtungen grundsätzlich entlang des Pfades Problemaspekt ↔ Maßnahme ↔ Analyseziel ↔ Analyseproblem ↔ Analyseaufgabe ↔ Aktivität ↔ Operator. Damit lässt sich z.B. prüfen, ob die Auswahl eines nicht geeigneten Operators durch die Spezifikation des Analyseziels bedingt ist, oder inwiefern sich die unklare Bestimmung eines Problemaspekts auf die Spezifikation der Analyseaufgabe auswirkt.

Die konkrete Prüfung der Elemente ist kontextabhängig und erfolgt nach Maßgabe der in den jeweiligen Kapiteln erläuterten Planungs-, Steuerungs- und Revisionsprinzipien. Neben der Gestaltung der Elemente selbst ist insbesondere auch ihre Ableitung aus den jeweils in der Architektur übergeordneten Elementen Gegenstand der Überprüfung. Darüber hinaus sind auch die jeweilige Verknüpfung der Elemente (Problemkarte, Analyseketten) sowie die an den Elementen annotierten Kontextregeln zu prüfen.

Die systematische Beurteilung eines Prozesssystems der Datenanalysearchitektur kann auch mithilfe spezieller programmorientierter Evaluationsmodelle erfolgen. Diese Modelle beleuchten Input, Output,

Prozess und Rahmenbedingungen eines Systems, um seine Funktionsprinzipien zu analysieren und Verbesserungsoptionen zu identifizieren [Soel10, 240f.].[264]

7.4.1.3 Zusammenfassung: Modifikation der Analysepläne

Die Modifikation der Analysepläne erstreckt sich über die gesamte Datenanalysearchitektur, um Verbesserungen konsistent auf allen Beschreibungsebenen vorzunehmen. Diese ganzheitliche Sicht fördert zugleich die Erkennung der Ursachen nicht befriedigender Prozessabläufe, die in früheren Planungsstufen zu suchen sind. Eine automatische Abhängigkeitsverfolgung, die dem Analytiker Änderungsbedarf auf allen Ebenen anzeigt oder selbständig realisiert, scheint erstrebenswert.

Es liegt nahe, bei der systematischen Modifikation von Analyseprozessen eine Optimierung anzustreben, die idealerweise automatisch erfolgt. WIRTH ET AL. beschreiben ein Data-Mining-Werkzeug, das den Analyseprozess als Folge von Datenbankabfragen realisiert und ähnlich eines Anfrageoptimierers relationaler Datenbankverwaltungssysteme effizienter gestaltet [WSG+97, 249f.]. Die Realisierung dieser Idee erscheint jedoch weder für grafische Analysewerkzeuge mit heterogenen, erweiterbaren Operatorenpools noch für skriptorientierte Data Science realistisch, da die hierzu nötige Formalisierung aller Operatoren auf Grundlage einer gemeinsamen Theorie nicht erreichbar ist. Die Optimierung von Analyseprozessen ist daher allenfalls in Bezug auf Teilaspekte, wie etwa die Parameterinstanziierung für einzelne Operatoren, möglich. JABLONSKI bemerkt hierzu, dass selbst im operativen Workflow-Management bestimmte Aspekte nicht auf Basis formaler Theorie berechenbar und daher pragmatisch zu handhaben sind [Jabl00, 355].

---

[264] Ein bekannter Vertreter ist das CIPP-Modell nach STUFFLEBEAM [StSh07]. Die Evaluation orientiert sich am zeitlichen Verlauf des Projekts und untersucht alle Bedingungen, die auf das Gesamtergebnis einwirken. Gestaltungsentscheidungen werden als Weichenstellungen explizit in das Modell aufgenommen.

Aspekte, die einer formalen Beschreibung nicht zugänglich sind, können effektiv mithilfe heuristischer Lösungsansätze behandelt werden. Die unmittelbar nach Abschluss des Ablaufs erfolgende Prozessverbesserung ist in der Lage, in mehreren Zyklen schrittweise eine akzeptable Prozessqualität zu erreichen. Der vorgestellte Ansatz der bedarfsorientierten Modifikation von Prozessvorlagen versucht der meist explorativen Natur der Datenanalyse gerecht zu werden, indem jeweils nur unmittelbar relevante Modifikationen erfolgen, ohne eine vollständige Umgestaltung zu erfordern.

7.4.2 Extraktion wiederverwendbaren Wissens (K6)

Die Datenanalyse und die Lösung von Sachproblemen sind schwach strukturierte Probleme und können stark von Erfahrung und Wissen über ähnliche Fälle profitieren [Tuke62, 63]. Der Datenanalysezyklus (vgl. Abschnitt 2.3.2.3) und der resultierende Verbesserungskreislauf sind zudem zeit- und ressourcenintensiv, weshalb jeder Beitrag zur Senkung von Durchlaufzeit und Aufwand die Effizienz solcher Vorhaben steigern kann [CFM+97, 147]. Ziel dieser Aufgabe ist die methodische Erhebung von Erfahrungswissen zur Unterstützung der Gestaltung und Durchführung von Datenanalyseprojekten, das als Ergebnis in der Fallbibliothek gespeichert wird.

Das relevante Erfahrungswissen umfasst prinzipiell sämtliche Erkenntnisse, die während der Planung, Steuerung, Durchführung und Revision erlangt werden. Seine Akquisition stellt gerade in Domänen mit sich stetig verändernden Prozessen eine Herausforderung für das Prozessmanagement dar [HaHK04, 211]. Zur Systematisierung des Vorgehens werden vier Teilaufgaben definiert, die in den folgenden Abschnitten erläutert werden. Zunächst ist die *Dokumentation von Kommentaren und Bewertungen (K6.1)* zu leisten, anschließend ist die *Ableitung von Kontextregeln (K6.2)* sowie allgemeiner Handlungsempfehlungen ratsam. Zum Aufbau einer Vorlagensammlung ist die *Identifizierung und Speicherung von Prozessartefakten (K6.3)* sowie die regelmäßige *Wartung der Fallbibliothek (K6.4)* notwendig. Abschnitt 7.4.2.5 diskutiert abschließend Realisierungsoptionen des Wissensmanagements.

7.4.2.1 Dokumentation von Kommentaren und Bewertungen (K6.1)

Im Zuge der Planung, Steuerung und Revision von Analyseprozessen sind zahlreiche *Gestaltungs- oder Lenkungsentscheidungen* zu treffen, die oft situativ motiviert sind und nur bei entsprechender Dokumentation nachvollziehbar sind. Beispiele sind die Spezialisierung eines Analyseziels, die Dekomposition einer Aufgabe, die Abweichung vom Prozessschema oder die Anpassung von Operatorparametern. All diesen Entscheidungen lassen sich im betroffenen Modellierungsobjekt `Kommentare` beifügen, um die getroffenen Erwägungen zu begründen. Die lokale Verortung schafft eine direkte Verknüpfung mit den jeweiligen Entscheidungsobjekten und erlaubt zudem ihre situative Erfassung zu dem Zeitpunkt, an dem die Entscheidung getroffen wird. Bei Wiederverwendung des Modellierungsobjekts stehen die Kommentare unmittelbar als Hilfsmittel zur Verfügung. Diese inkrementelle, problemorientierte Wissensakquisition erspart die aufwändige Nacherfassung und fördert die Vollständigkeit der Dokumentation, die bei Ausführung zu einem späteren Zeitpunkt meist leidet. Dennoch können zur Ausführungszeit nicht erfasste Kommentare auch während der Revision ergänzt werden.

Ebenso sieht der Modellierungsansatz vor, *Bewertungsergebnisse* an den Bewertungsobjekten (z.B. Prozessabläufe und Handlungsmaßnahmen) zu hinterlegen. Damit lässt sich der Zusammenhang zwischen den Merkmalen des Objekts (z.B. Parameterwerte einer Aktivität) und den erzielten Ergebnissen (z.B. Genauigkeit, Zielerreichung) herstellen. Die Prozessparameter, die während der Effizienzbeurteilung des Analyseablaufs (K2.2) anfallen, bilden eine empirische Grundlage zur Planung des Zeit- und Ressourcenbedarfs künftiger Projekte. Die Berechnung zugehöriger Lage- und Streuungsmaße für Prozesszeiten und -kosten gestattet z.B. die Abschätzung minimaler, maximaler und typischer Werte, die aus der Erfahrung mit bisherigen Vorhaben zu erwarten sind. Die multidimensionale Ablage dieser Kennzahlen in einem Prozess-Data-Warehouse (vgl. Abschnitt 7.2.2.4) ermöglicht die Analyse dieser Daten in Abhängigkeit von Kontextfaktoren. Werden analog auch Operatorparameter gespeichert, ist eine erfahrungsbasierte Prozessplanung möglich: So können z.B. Parameterwerte für positiv beurteilte

Abläufe ermittelt und bewährte Konfigurationen für beliebige Kontexte reproduziert werden.

7.4.2.2 Ableitung von Kontextregeln (K6.2)

Eine Datenbasis wie das genannte Prozess-Data-Warehouse erlaubt die Herstellung von Korrelationen zwischen Kontextfaktoren, Gestaltungsparametern und Bewertungsergebnissen, die in Kontextegeln oder allgemeinen Empfehlungen münden können (vgl. [GrBC08, 269, 271]). *Kontextregeln* nehmen auf konkrete Kontextfaktoren Bezug und geben eine präzise Spezifikation, wie Prozesselemente unter welchen Bedingungen zu gestalten sind (vgl. Abschnitt 4.5.4.1). *Handlungsempfehlungen* besitzen demgegenüber niedrigeren Detail- und Bestimmtheitsgrad und geben z.B. vor, dass eine bestimmte Algorithmenklasse prinzipiell zu meiden ist, oder dass sich für bestimmte Datentypen eine spezielle Transformation empfiehlt. Sie werden natürlichsprachig dokumentiert. Die Regeln können mit klassischen Mitteln der Wissensakquisition erhoben werden, eine empirisch besser gesicherte und einfacher zu handhabende Option stellt jedoch die Nutzung datenanalytischer Verfahren dar (vgl. [Enge99, 137]), in die Abschnitt 7.4.2.5 kurz einführt. Regeln werden jeweils an dem Modellierungsobjekt hinterlegt, auf das sie zutreffen, z.B. Funktion, Prozessbaustein oder Operator. Hierfür steht das Attribut `Kontextregeln` zur Verfügung. Handlungsempfehlungen werden als `Kommentare` abgelegt.

In der Literatur finden sich einige Arbeiten zu Anwendungsbeispielen und Realisierungsoptionen der regelgestützten Gestaltung von Datenanalyseprozessen. ALI & WALLACE [AlWa97] stellen einen Ansatz zur Verfahrensparametrisierung in Abhängigkeit von der `Anwendung` vor. Hierzu verknüpfen sie zunächst Anwendungen manuell mit Performanzmaßen von Analysealgorithmen und ermitteln danach mithilfe induktiver Lernverfahren Zusammenhänge mit Parameterwerten, die schließlich in Konfigurationsregeln münden.[265] LINDNER [Lind05] zeigt

[265] Beispielsweise korrespondiert die Anwendung (in der Quelle als Geschäftsziel bezeichnet) „Steigerung der Produktionsqualität“ mit der Performanzgröße Genauig-

ein fallbasiertes Werkzeug zur Algorithmusselektion im Data Mining, das aus früheren Analysefällen Regeln zur Verfahrensauswahl erlernt. Nicht speziell auf Datenanalyseprozesse zielt der Ansatz zur robusten Prozesssteuerung von DELLAROCAS & KLEIN [DeKl00], der jedes Prozesselement mit Kontextregeln zur Ausnahmeerkennung und -behandlung annotiert. Diese Regeln können auch im Rahmen der Prozessmodifikation zur Identifizierung fehlerträchtiger Bausteine genutzt werden. Einen ähnlichen Vorschlag unterbreiten GROB ET AL. [GrBC08], die mittels Prognose der zu erwartenden Effektivität geeignete Verbesserungsmaßnahmen ableiten [GrBC08, 277].

Damit die Regeln und Handlungsempfehlungen stets valide und konsistent zur Menge der Analysefälle sind, ist ihre regelmäßige Überprüfung und Anpassung erforderlich [GrBC08, 269]. Einige Arbeiten zur modellgestützten Verfahrenswahl sehen eine Datencharakterisierung oder Algorithmusprofilierung vor (z.B. [CSG+92], [BeGi00], [HiKa01], [LeVo10]), die typischerweise in separaten Projekten geschehen. Die Aktualisierung solcher Modelle kann zum festen Bestandteil dieser Revisionsaufgabe werden. Auf diese Weise können die Eigenheiten jedes neuen Falles sofort in die Regelbasis einfließen und die Regelgüte verbessern. Mehrere Ansätze zur Analyseplanung nutzen in Kontextregeln ordinale Merkmale, wie etwa `{langsam, mittel, schnell}` (vgl. [BePr01], [Hogl03], [NaSA03]). Sie befreien den Analytiker von der Vorgabe konkreter Werte, die oft schwer zu bestimmen sind, weshalb sie auch die vorliegende Arbeit empfiehlt (z.B. im Informationsbedarfs- oder Verfahrensprofil). Zur Festlegung geeigneter Intervallgrenzen eignen sich induktive Ansätze auf Grundlage empirischer Erfahrungswerte, z.B. mittels k-Means-Verfahren [Lind05, 178], sowie Fuzzy Logic [NaSA03].

7.4.2.3 Identifizierung und Speicherung von Prozessartefakten (K6.3)

Nach der Modifikation (K5) liegt ein aus dem Prozessablauf abgeleitetes Prozesstypschema in fehlerfreier, verbesserter Form vor und kann zur

keit eines Klassifikators zur Qualitätsbeurteilung, da die Entscheidung über die Güte eines Werkstücks sehr exakt sein muss [AlWa97, 4].

späteren Wiederverwendung in der Fallbibliothek abgespeichert werden. Der revidierte Workflow wird grundsätzlich vollständig archiviert. Er wird durch die Analyseaufgabe bzw. das Analyseproblem charakterisiert (vgl. Abschnitt 4.5.4).

Darüber hinaus ist zu entscheiden, inwiefern bestimmte Prozessausschnitte, einzelne Elemente oder Abstraktionen davon ebenfalls eine Speicherung lohnen, weil sie geeignet erscheinen, die Gestaltung und Lenkung von Prozessen als wiederverwendbare Artefakte zu unterstützen (vgl. [HwWY04, 362]). Dazu sind im Wesentlichen zwei Fragen zu beantworten:

- Welche Prozessabschnitte sind zu persistieren?
  (*Identifikation von Prozessbausteinen*)
- Auf welchem Abstraktionsniveau sind die Bausteine zu speichern?
  (*Abstraktion von Prozessbausteinen*)

Während über die Speicherung einzelner Prozesselemente (Aufgaben, Aktivitäten) direkt anhand ihrer Definition entschieden werden kann, ist bei mehrelementigen Prozessmodulen zusätzlich die Frage ihrer Ausgrenzung aus dem Prozessgefüge zu stellen. Antworten hierauf steuern Anforderungen (Formalziele) an Prozessmodule bei, die auf Grundsätzen der Prozessmodellierung, der Software-Wiederverwendung sowie auf Integrationszielen [Fers92, 11ff.] beruhen. Demnach sollte ein Prozessmodul aus *Verhaltenssicht* eine klar definierte Funktion realisieren (Zielorientierung), um die Einschätzung seiner Relevanz für spätere Analysefälle zu gestatten (vgl. [Gait83, 75f.]). Das vom Modul als Einheit gezeigte Verhalten muss den Kriterien der Korrektheit genügen [Gait83, 86], [Fers92, 11] und sollte in Bezug auf Ein- und Ausgabeflüsse ein deterministisches Input-Output-System realisieren [Gait83, 82], [JoLe12, 276f.], [RuNo03, 423f.]. Aus *Struktursicht* (Verknüpfung) sind nach außen möglichst unabhängige Bausteine anzustreben, die gleichzeitig schwache Außenbindung (minimale Kopplung) und starke Innenbindung (Kohäsion) besitzen. Dies gewährleistet eine klare Abgrenzung

sowie eindeutige Schnittstellen zu anderen Bausteinen [Gait83, 81], [JoLe12, 273-277].[266]

Aus *Nutzungssicht* sind die pragmatischen Forderungen nach Nützlichkeit und Nutzbarkeit zu beachten [MiMM95, 540]. Ein Baustein ist *nützlich* (relevant), wenn er eine definierte Funktion realisiert, die vorzugsweise häufig in Prozessabläufen auszuführen ist (statistische Relevanz) und einen möglichst großen Teil des Gesamtprozesses abdeckt [Gait83, 75]. Allgemein werden solche Bausteine bevorzugt, die einen erkennbaren Beitrag zur Komplexitätsbewältigung leisten. Ein Baustein ist *nutzbar*, wenn dem Analytiker die Nützlichkeit unmittelbar verständlich ist (vgl. [Gait83, 77]) und seine Handhabung leicht fällt. Die Verständlichkeit wird durch Einordnung in die Funktionstaxonomie befördert. Die Nutzbarkeit ist in der Regel umso höher, je mehr der verhaltens- und strukturbezogenen Anforderungen erfüllt sind.

Die Identifizierung zu speichernder Artefakte geschieht idealerweise grafisch, indem der Analytiker im Prozessdiagramm einen Ausschnitt markiert, seine Auswahl im Lichte der dargelegten Formalziele prüft [Gait83, 86], bei Bedarf revidiert und schließlich durch Zuordnung zu einem passenden Element der Funktionstaxonomie semantisch annotiert. Ergänzende Anmerkungen und Kommentare sind empfehlenswert. Typischerweise kann der Analytiker recht gut einschätzen, welche Ausschnitte eine Speicherung lohnen. Ein derart ausgegrenztes Prozesssegment wird zu einem Prozessmodul aggregiert und ist geeignet, eine der annotierten Funktion entsprechende Aufgabe zu substituieren. Einzelne Prozesselemente werden direkt als Baustein gespeichert.

Heuristische Prinzipien zur Modulausgrenzung

Um die Ausgrenzung von Modulen methodisch zu unterstützen, lassen sich aus den oben erhobenen Anforderungen folgende heuristische Prinzipien formulieren. Sie sind nach ihrer erfahrungsgemäßen

[266] Eine wichtige weitere Forderung nach Minimalität wird bereits durch die Eliminierung unnötiger Aktivitäten und Redundanzen im Rahmen der Modifikation behandelt [Gait83, 86], [JoLe12, 276f.]. Für Prozessschablonen ist Aufgabenträgerunabhänigkeit (vgl. [Fers92, 12-14]) anzustreben, welche durch Bildung reiner Schablonen (vgl. Abschnitt 4.5.4.1) sichergestellt ist.

Präferabilität gereiht und in absteigender Ordnung sequenziell kombinierbar.

- *Leistungsprinzip*: Ein Prozessmodul ist in sich abgeschlossen, wenn es eine definierte Leistung erbringt und anhand des Ausgabeflusses zur Übergabe dieser Leistung von anderen Bausteinen klar abgegrenzt werden kann [Gait83, 65]. Demnach sind Elemente mit mehr als einem Ausgabefluss nicht als Endelement eines Moduls geeignet. Dieses Prinzip unterstützt die Anschlussfähigkeit und führt zu Modulen, die wichtige (Zwischen-) Ergebnisse erstellen.
- *Funktionsprinzip*: Ein Prozessmodul wird ausgehend von einem Eingabefluss eines Prozesselements gebildet (Aufgabenobjekt), der gemäß einer gegebenen Funktion (Sachziel) bearbeitet wird. In Kombination mit dem komplementären Leistungsprinzip entstehen kompakte Module, die einzelne Inputs zielorientiert in definierte Outputs transformieren.
- *Umhüllungsprinzip (Flussprinzip)*: Ausgehend von einem Ausgabe- oder Eingabefluss (Leistungs- bzw. Funktionsprinzip) werden sukzessive alle in das betreffende Element ein- bzw. ausgehenden Flüsse rückwärts bzw. vorwärts verfolgt und die aktuelle Hülle (Modulgrenze) jeweils um ein weiteres Element erweitert. Dies geschieht solange, bis dem abgegrenzten Prozesssegment eine klar definierte Leistung und eine bekannte Funktion attributiert werden können. Dieses Prinzip unterstützt die minimale Kopplung durch schrittweise Integration weiterer Prozessbausteine [JoLe12, 274] und generiert intuitiv verständliche Module.
- *Lenkungsprinzip*: Zur Unterstützung von Kohäsion, schwacher Kopplung und Korrektheit dürfen spezielle Koordinationsstrukturen (z.B. Parallelisierung, Schleifen, Sprünge, etc.) keinesfalls aufgetrennt werden, sondern sind in einem Modul zu kapseln.
- *Verzweigungsprinzip (Objektprinzip)*: Prozesselemente mit mehr als einem Ausgabefluss sind danach zu untersuchen, ob diese Flüsse auf Typebene verschiedene Informationsobjekte transportieren. Ist dies der Fall, entstehen unterschiedliche Aufgabenobjekte (Typebene), die gemäß Funktionsprinzip die Bildung separater Module

nahelegen. Wann die Trennung tatsächlich sinnvoll ist, hängt letztlich von der Relevanz des Funktionsprinzips und von der weiteren Prozessstruktur im jeweiligen Kontext ab. Eine spätere Vereinigung der Flüsse ist ein Indikator gegen eine Separierung. Der Vorrang des Lenkungsprinzips ist zu beachten.

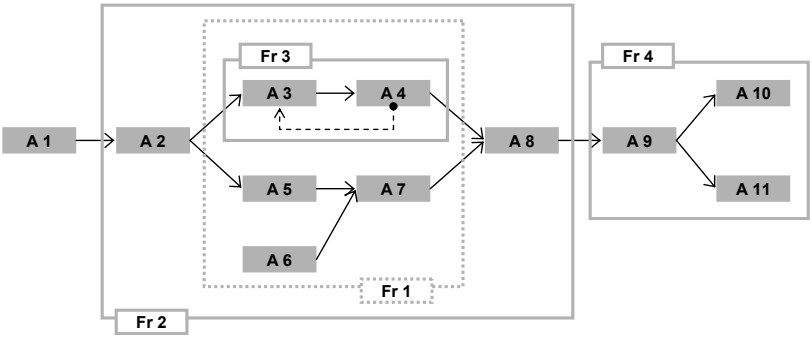


Abbildung 106: Beispiele zur Ausgrenzung von Prozessmodulen (eigene Darstellung)

Abbildung 106 zeigt einen beispielhaften Workflow und mögliche Modulausgrenzungen. Fragment Fr1 weist zwar starke Kohäsion auf, sollte jedoch gemäß Umhüllungsprinzip zu Fr2 erweitert werden, da es mehrere Outputs und Inputs besitzt (keine minimale Kopplung). Durch Ausweitung über A2 (links) wird ein einziger Eingabefluss erreicht (Funktionsprinzip), durch Ausdehnung über A8 (rechts) wird die Leistung des Moduls auf einen Ausgabefluss reduziert (Leistungsprinzip). Zugleich entsteht direkte Anschlussfähigkeit zum Fragment Fr4. Die Schleifenstruktur zwischen A3 und A4 wird gemäß Lenkungsprinzip in Modul Fr3 gekapselt. Jenes kann auch vom Verzweigungsprinzip motiviert sein, sofern A2 typverschiedene Outputs produziert.

Die Identifikation von Prozessmodulen lässt sich auch mithilfe der *Funktionshierarchie* bestreiten. Die semantische Annotation, die jede Aktivität mit Zuordnung zur Funktionstaxonomie (implizit) enthält, beschreibt einen Abstraktionspfad innerhalb der Hierarchie. Abhängig von der Vorgehensweise bei der Gestaltung des Prozessschemas dienen ein Bündelungs- (bei Bottom-up-Planung) bzw. ein Dekompositionsprinzip (bei Top-down-Planung) zur Orientierung.

- *Bündelungsprinzip*: Bei Bottom-up-Planung ist der Abstraktionspfad linear und vereinigt sich mit anderen Pfaden an der Wurzel, von der für jede Aktivität ein Strang ausgeht (Abbildung 107a oben). Ein induktiver Ansatz zur Bildung von Prozessmodulen ermittelt, ausgehend von den Blattknoten, entlang dieses Pfades in der Hierarchie aufsteigend gemeinsame Vorfahren (Funktionen F) und bündelt alle zugehörigen Stränge an dieser Stelle (Abbildung 107a unten). Dabei wird jene Funktion gesucht, die den „kleinsten gemeinsamen Nenner" für die Aktivitätssequenz darstellt. So entsteht z.B. aus Sequenz [A1, A2] durch Bündelung bei F1.3 das Fragment Fr5. Wird eine Aktivitätsfolge hingehen durch Elemente mit Zugehörigkeit zu einer anderen Funktion „unterbrochen" oder liegt eine nicht-sequenzielle Ablaufstruktur vor, so ist zu entscheiden, ob unterhalb der Bündelungsebene eine neue Funktion in die Taxonomie aufgenommen wird, welche die vom Modul repräsentierte, neue Aufgabe treffend beschreibt. Im Beispiel sitzt A4 mit Abstammung F2.6.9 zwischen A3 und A5 mit gemeinsamer Abstammung F2.5.7, weshalb eine Bündelung erst bei F2 möglich wird. Für Fr6 soll die Bündelung jedoch bei einer darunterliegenden, neu spezifizierten Funktion F2.N erfolgen, die der nicht homogenen Abstammung der betroffenen Aktivitäten Rechnung trägt. In jedem Falle ist die sequenzielle Verknüpfung der Aktivitäten bei der Modulbildung zu berücksichtigen.
- *Dekompositionsprinzip*: Sind Prozessbausteine bei Top-down-Planung aus Spezialisierung oder Zerlegung von Aufgaben (T) entstanden, ist die Dekompositionsstruktur aus der `Genese` des Bausteins rekonstruierbar (in Abbildung 107b durch ausgefüllte Kreise symbolisiert). Daraus lassen sich direkt potenzielle Module ableiten. Während einer Traversierung des entstehenden Baumes kann an jeder Verzweigung entschieden werden, ob die dort gewählte Spezialisierung bzw. Zerlegung als Schablone (auf Blattebene als Fragment) zu persistieren ist. Die Traversierung kann abhängig davon, ob nach abstrakten oder eher konkreten Modulen gesucht wird, an der Wurzel bzw. an den Blättern des Baumes beginnen. In Abbildung 107b unten wird z.B. die Zerlegung von T3 als Fragment Fr7 gespeichert, während die Dekomposition von T4.7

unter Vernachlässigung der Operatoren als Schablone S8 archiviert wird. Für die Zerlegung von T4 wird entscheiden, keine Speicherung vorzunehmen.

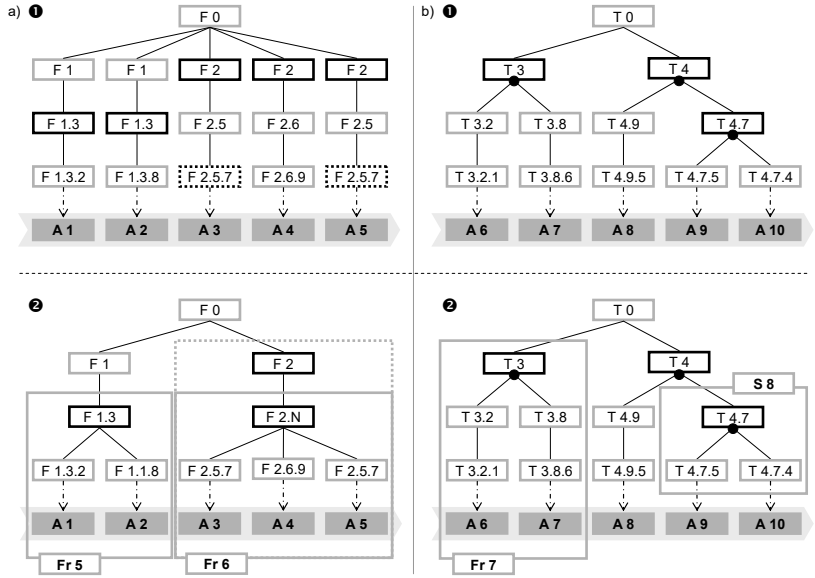


Abbildung 107: Beispiele zur Identifizierung von Prozessmodulen gemäß a) Bündelungsprinzip und b) Dekompositionsprinzip (eigene Darstellung)

Aus dem Beispiel zur Bündelung wird deutlich, dass parallel zur Identifizierung von Prozessmodulen auch die *Pflege von Taxonomien und Ontologien* Gegenstand der Wissensakquisition ist. Mit Bildung jedes Moduls, für das bislang keine passende Funktion existiert, ist eine solche in die Taxonomie aufzunehmen. Analog sind neue Abstraktionen für Datenobjekttypen (Ressourcenebene) bei Bedarf in der Ontologie zu ergänzen.

Speicherung von Artefakten auf Anwendungs- und Analyseebene

Neben Prozessschemata und -segmenten besitzen auch die Elemente der Anwendungs- und Analyseebene Wiederverwendungspotenzial. Sowohl Problemkarten als auch Analyseketten sind aus ihren Elementen

rekonstruierbar, weshalb nur einzelne Problemaspekte bzw. Analyseziele-/probleme persistiert werden (vgl. Abschnitte 4.3.3 und 4.4.4). Hierbei sind stets alle Elemente in die Bibliothek aufzunehmen, denen auf niedrigeren Architekturebenen Elemente zugeordnet sind, um jeweils deren Kontext vollständig beschreiben zu können. Der Analytiker kann gegebenenfalls nicht gespeicherte Elemente eigens zur Speicherung selektieren.

Abstraktion von Prozessartefakten

Da die Entwicklung wiederverwendbarer Prozessbausteine grundsätzlich gewissen Mehraufwand auslöst, ist auch die Kosten-Nutzen-Relation im Blick zu behalten. Kosten entstehen bei Identifizierung und Speicherung, aber auch bei Integration und Anpassung der Artefakte an den neuen Fall.[267] Sie sind der erwarteten Nutzungsintensität gegenüberzustellen [MiMM95, 538]. Daher kann es von Vorteil sein, das Verhaltensrepertoire mehrerer konkreter Ablaufvarianten in einem Prozessmodul zusammenzufassen, um dessen Anwendungspotenzial zu steigern. Dies kann geschehen, indem Elemente auf ein höheres Abstraktionsniveau gehoben werden, oder indem alternative Ablaufpfade, z.B. durch Verzweigungen, in einem Modul integriert werden. Dieser Aspekt betrifft die *Flexibilität* der Vorlagen (vgl. [Gait83, 82]).

Bezüglich des Abstraktionsniveaus bei der Speicherung eines Bausteins bestehen stets mehrere Optionen. Für konkrete Artefakte spricht ihre unmittelbare Ausführbarkeit. Höheres Anwendungspotenzial und mehr Flexibilität zum Einsatz auch in Fällen mit abweichenden Details oder unklaren Vorgaben besitzen hingegen generalisierte, abstrakte Bausteine [BeWi95, 73], [RuNo03, 430]. Aus Komplexitätssicht reduzieren konkrete Module den Anpassungsbedarf, während abstrakte Vorlagen in Situationen, für die keine passende Lösung existiert, Hilfestellung geben und den Alternativenraum wirksam einschränken [BeWi95, 55, 57f., 71].

---

[267] Die Kosten der Integration und Anpassung sollten niedriger sein als die der Neuentwicklung, da Wiederverwendung andernfalls sinnlos ist. Bei ungünstiger Ausgrenzung sind dennoch Fälle denkbar, in denen sie höher liegen.

Daher kann nach der Faustegel „so konkret wie möglich, so abstrakt wie nötig“ ein mehrstufiges Vorgehen gewählt werden: (1) Zunächst sind grundsätzlich immer konkrete Fragmente oder Aktivitäten zu speichern. (2) Zusätzlich wird eine abstrakte Variante immer dann gespeichert, wenn der Analytiker die unmittelbare Relevanz einer Abstraktion auf einer bestimmten Detailstufe erkennt. (3) Weitere Abstraktionen werden erst dann erzeugt, wenn ein Baustein der Bibliothek in einem späteren neuen Fall unter Anpassungen wiederverwendet wurde, die eine Abstraktion nahe legen. Das Modul wird dabei als Kandidat für Abstraktion markiert und kann im Rahmen der Revision des neuen Falls in geeigneter Form abstrahiert werden. Auf diese Weise wird sichergestellt, dass die Bibliothek keine zu hohe Zahl von Artefakten unterschiedlicher (beliebiger) Detailstufe enthält, die letztlich nicht benutzt werden und eine unnötige Komplexität der Bibliothek verursachen. Abstakte Varianten werden erst dann erzeugt, wenn tatsächlich Anhaltspunkte existieren, welche Form der Abstraktion sinnvoll ist.

Den Ablauf der Speicherung und Wiederverwendung abstrakter Lösungen in einem fallbasierten System zeigt Abbildung 108 (vgl. [BeWi95, 57f.]). Aus vergangenen Analysefällen ausgegrenzte konkrete Module werden zunächst abstrahiert, anschließend generalisiert und nach geeigneter Indizierung in der Fallbibliothek abgelegt. Diese stellt für neue Analysefälle passende Vorlagen zum Abruf bereit (Retrieval), die nach Spezialisierung und Konkretisierung als Baustein in das Prozessschema des neuen Falls eingesetzt werden. Die Schritte Retrieval, Spezialisierung und Konkretisierung erfolgen gemäß den Ausführungen zur Prozessplanung in Abschnitt 5.5. Sie orientieren sich ebenso wie Abstraktion, Generalisierung und Indizierung an der Funktionstaxonomie. Die Abstraktion entspricht dem Schritt von der Aktivitäts- auf die Aufgabensicht (Fallenlassen der Aufgaben-Innensicht). Die Generalisierung[268] nimmt Bezug auf ein oder mehrere Merkmale der Aufgaben-Außensicht und geschieht durch Aufsteigen innerhalb der Funktionstaxonomie. Dabei werden Sachziel oder

[268] Die Generalisierung stellt ebenfalls eine Abstraktion dar, die sich auf die Extension der repräsentierten Begriffe auswirkt, d.h., sie erweitert ihren Geltungsbereich auf eine größere Objektmenge [BeWi95, 57].

Aufgabenobjekt (Ein-/Ausgabeflüsse) der Aufgabe verallgemeinert, z.B. indem die Diskretisierung reeller Attribute auf höherer Ebene als Transformation numerischer Attribute charakterisiert wird. Die resultierenden abstrakten und generalisierten Module können auf unterschiedlichen Detailebenen sitzen.

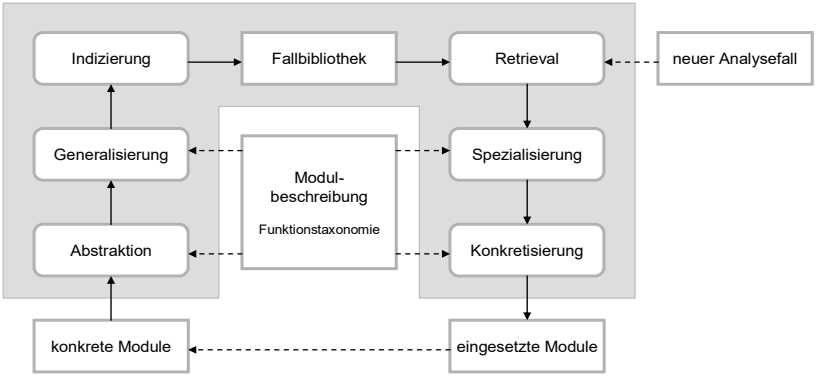


Abbildung 108: Speicherung und Wiederverwendung abstrakter, generalisierter Artefakte in einem fallbasierten System, angelehnt an [BeWi95, 61]

Es lässt sich zeigen, dass abstrakte Fälle auf eine signifikant größere Anzahl von Problemen anwendbar sind als Fälle, die nur einer Generalisierung unterzogen werden [BeWi95, 71]. Die Nützlichkeit der Bausteinbibliothek steigt damit merklich. Dies ist gerade in Domänen wie der Datenanalyse, in der nur eine eher geringe Zahl an Prozessabläufen bereitsteht, mit denen die Bausteinbibliothek gespeist werden kann, von großem Vorteil (vgl. [BeWi95, 73]).

7.4.2.4 Wartung der Fallbibliothek (K6.4)

Die Speicherung von Prozessartefakten in der Fallbibliothek schafft ein Repositorium erfolgreicher Beispiele aus vergangenen Analysevorhaben, das stetig fortzuschreiben und zu pflegen ist. Die Aufnahme neuer Elemente ist eine regelmäßig wiederkehrende Aufgabe [MiMM95, 537], die durch Einbettung in das Handlungsschema zum integralen Bestandteil eines jeden analytisch gestützten Projekts wird. Die Fallbibliothek erfährt nach jedem Prozessablauf eine Erweiterung um den zuge-

hörigen Workflow sowie die assoziierten Analyseziele und Problemaspekte der Analyse- bzw. Anwendungsebene. Zusätzlich werden jene Prozessbausteine aufgenommen, die nach Ermessen des Analytikers eine Speicherung lohnen. Durch diese iterative Bibliothekserweiterung entfällt eine separate Wissensakquisition, die erfahrungsgemäß als unproduktiv empfunden und daher häufig vernachlässigt wird.

Große Bibliotheken sind nicht zwingend vorteilhaft und verursachen unnötige Komplexität. Daher sollten neue Bausteine nur nach fundierter Abwägung aufgenommen werden, und Bausteine mit niedriger Nutzungshäufigkeit sind regelmäßig zu entfernen [MiMM95, 538]. Bei der Wartung der Fallbibliothek kann auf die Beurteilung des Analytikers zurückgegriffen werden, der jeden Baustein mit `Einsatznote`, `Einsatzbewertung` und `Einsatzkommentar` versehen kann (vgl. Abschnitt 4.5.4.1), sowie die Ähnlichkeit zu anderen gespeicherten Elementen berücksichtigt werden (Redundanzvermeidung). Zusätzlich steht ein `Einsatzzähler` zur Verfügung, der die Nutzungshäufigkeit eines Bausteins anzeigt. Daraus lassen sich z.B. Kennzahlen des erfolgreichen Einsatzes ableiten.[269] Unterschreiten diese Maße einen definierten Schwellenwert, kann der Baustein als Kandidat zur Löschung markiert und dem Anwender zur Entscheidung vorgelegt werden.

7.4.2.5 Realisierungsoptionen des Wissensmanagements

Selbst bei Vorgabe von Empfehlungen und Heuristiken beruht die Identifizierung von Prozessartefakten auf der subjektiven Einschätzung des Analytikers und setzt „kreative und konstruktive Akte“ voraus [Gait83, 65]. Die große Zahl möglicher Ausgrenzungen und Abstraktionsniveaus verursacht hohe Komplexität. Gerade die Integration mehrerer Module zu einem nützlichen Baustein ist manuell kaum befriedigend handhabbar. Daher scheint die Unterstützung dieser Aufgabe mit Mitteln der Datenanalyse vielversprechend.

---

[269] Während des Retrievals sollten Module nach der Anzahl ihrer Einsätze insgesamt und der Anzahl bzw. dem Anteil der positiv bewerteten Einsätze in Prozessen priorisiert werden.

Geeignete Ansätze sind unter der Bezeichnung *Process Mining* (auch *Workflow Mining*) bekannt. Process Mining verfolgt das Ziel, aus Ausführungsprotokollen von Prozessabläufen Informationen über die zugrundeliegenden Prozesse zu extrahieren. Als Ergebnisse werden explizite Prozessschemata angestrebt, die mit dem protokollierten Verhalten konsistent sind. Diese Schemaerkennung ist hilfreich, wenn mehrere Abläufe zu einer gemeinsamen Vorlage integriert oder häufig wiederkehrende Prozessfragmente identifiziert werden sollen (vgl. [HwWY04, 345], [WRRW05, 1]). Im Rahmen einer Delta-Analyse lassen sich Deviationen zwischen Plan- und Ist-Abläufen aufdecken, häufige Abweichungen in die Vorlage integrieren, Alternativpfade aufnehmen oder irrelevante Pfade löschen. Zudem können Kontextregeln abgeleitet oder angepasst werden [ADH+03, 239-241], [AaWe04, 232], [AcUA12, 358]. Der Rückgriff auf das gesamte Instanzmodell [GrBC08, 270] gestattet zudem die Analyse von Einflussfaktoren auf Prozessparameter (z.B. Zielerreichung, Durchlaufzeit, Genauigkeit von Analyseergebnissen), auf deren Basis Regeln zur Ausnahmeerkennung, -behandlung und Prozessverbesserung ableitbar sind [GCC+04, 322, 327]. So lassen sich z.B. für definierte semantische Kategorien wie „schlechte", „langsame" oder „zu ungenaue Ergebnisse produzierende" Abläufe Ursachen analysieren (vgl. die Hinweise in Abschnitt 7.4.2.2).

Process Mining scheint somit prinzipiell geeignet, die Identifizierung von Prozessartefakten und die Akquisition von Prozesswissen effektiv zu unterstützen. FRIESEN & RÜPING [FrRü10] wenden diesen Ansatz auf Data-Mining-Workflows an und zeigen seine Geeignetheit zur Ableitung von Prozessfragmenten. Einen Überblick über Verfahren und Probleme vermitteln [AaWe04, 239].[270]

[270] Für die Anwendung auf Datenanalyseprozesse relevante Grenzen dieser Ansätze liegen u.a. in der Erkennung wiederholter Aufgaben [HeKa04, 249] sowie konjunktiv verknüpfter Eingabeflüsse [ADH+03, 242]. Zudem wird die bei Analyseprozessen zu erwartende geringe Zahl verfügbarer Prozessinstanzen problematisch gesehen [ADH+03, 255], diese beeinträchtigt jedoch hauptsächlich die Vollständigkeit der ermittelten Schemata in Bezug auf seltene Ablaufpfade [AaWe04, 233], die wiederum von geringem Interesse ist.

Andere Ansätze zur Unterstützung des Prozess-Wissensmanagements stützen sich mehrheitlich auf *Case-based Reasoning (CBR)*. WEBER ET AL. [WRRW05] präsentieren ein integriertes System zur Analyse und Behandlung von Abweichungen in Prozessinstanzen, das CBR und Process Mining vereint. BARTLMAE & RIEMENSCHNEIDER [BaRi00] nutzen CBR für die projektübergreifende Erfahrungssicherung in KDD-Projekten. Zur Organisation des Wissensmanagements wird der Vorschlag der *Experience Factory* aus der Software-Entwicklung nach BASILI ET AL. [BaCR94] adaptiert, der eine Trennung der Akquisition und Pflege von Erfahrungswissen von seiner Erzeugung und Nutzung in den Projektteams vorsieht. Auf Basis definierter Rollen und Prozesse entsteht ein internes Kompetenzzentrum für KDD, das Projekte beratend unterstützt. Die Wissensverwaltung geschieht mittels CBR. Als Fälle werden verschiedene Typen sogenannter Erfahrungspakete in Form strukturierter Dokumente abgelegt. Sie repräsentieren Erfahrungen mit Dokumenten, Prozessen, Daten, Lösungen, Verfahren, Bewertungskriterien, Software-Produkten sowie Rollen und Personen. Ihre Indizierung erfolgt durch automatische Informationsextraktion aus den Fallbeschreibungen unter Rückgriff auf Ontologien [BaRi00, 2-7]. Auch der Vorschlag von ALTHOFF ET AL. [ABD+02] beruht auf Experience Factory, CBR und Text Mining. Er zielt auf die Erstellung und Pflege von Prozessschemata, die von allen Beteiligten akzeptiert, bei Bedarf an neue Anforderungen angepasst und in der Gruppe diskutiert werden [ABD+02, 54]. Erfahrungswissen soll den Beteiligten über ein Erfahrungsmanagementsystem kontextsensitiv bereitstehen [ABD+02, 55f.].

7.5 Zusammenfassung: Revision

Nach der Durchführung eines Datenanalyseprozesses steht eine Betrachtung aller Elemente der Analysearchitektur im Hinblick auf die Frage, „inwieweit jeder durchgeführte Arbeitsschritt für sich und in der Verzahnung mit den anderen Arbeitsschritten als gelungen bezeichnet werden kann“ [Witt98, 3]. Die ganzheitliche Bewertung analytisch unterstützter Projekte sollte den gesamten Weg von der Formulierung der Problemstellung in der Fachabteilung über die Beauftragung und Durchführung der Datenanalysen und Maßnahmen bis zur Bereit-

stellung der Ergebnisse abdecken, der oft Monate dauern kann [BeST99, 99f]. Dieses Kapitel stellt ein strukturiertes Handlungsschema vor, das bei der Bewältigung dieser Aufgabe unterstützen kann.

Die dargestellte Vorgehensweise zur Revision von Datenanalyseprozessen realisiert eine kontinuierliche Prozessverbesserung, bei der Prozesse inkrementell an neue Anforderungen angepasst [BBFS11, 9] und in die Lage versetzt werden, ein zunehmend umfangreicheres Repertoire an Kontextfaktoren abzudecken. Die Revision übernimmt dabei die Funktion der Kontrolle gemäß dem Regelkreisprinzip, indem durchgeführte Analysen mithilfe definierter Bewertungskriterien im Lichte des Analyseproblems bzw. des übergeordneten Sachproblems ganzheitlich beurteilt und bei Bedarf angepasst werden. Da sich Datenanalysen in weit geringerem Maße wiederholen als Geschäftsprozesse oder operative Workflows, sind Anpassungsmaßnahmen nicht in jedem Falle erforderlich. Vielmehr wird ein Ansatz zur Erfahrungssicherung vorgestellt, der die aus der Prozess- bzw. Analysebeurteilung resultierenden Erkenntnisse auch dann projektübergreifend zur Verfügung stellt, wenn eine direkte Wiederholung des betrachteten Prozesses unwahrscheinlich ist. Somit können die Beurteilungsergebnisse in die Gestaltung neuer oder die Umgestaltung bestehender Prozesse einfließen und zu einem lernenden Datenanalysemanagement beitragen. Folglich unterstützt der Ansatz mit der kontinuierlichen Verbesserung und der Innovation (Reengineering) beide Instrumente des klassischen Prozessmanagements [BBFS11, 9].[271]

[271] Während Geschäftsprozess- und Workflowmanagement die kontinuierliche Verbesserung typischerweise als Regelfall und die Neugestaltung eher als Ausnahme ansehen [GaSV94, 11], stellt sich die Situation in der Datenanalyse tendenziell invers dar.

Teil C:
Evaluation

Die vorausgehenden Teile dieser Arbeit nehmen eine umfangreiche Analyse der Grundlagen und Gestaltungsoptionen von Datenanalysen und Datenanalyseprozessen vor und zeigen die bei ihrem Einsatz zu erwartenden Schwierigkeiten auf. Die hierbei gewonnenen Erkenntnisse dienen als Fundament für den Entwurf einer umfassenden Methodik zum Management solcher Prozesse.

Der dritte Teil dient der Evaluierung und zusammenfassenden Einschätzung der erarbeiteten Konzepte. Hierzu betrachtet Kapitel 8 einen Anwendungsfall eines Konsumgüterherstellers und wendet die Methodik exemplarisch auf die Planungsphase des Managementzyklus an. Die weiteren Phasen werden vor dem Hintergrund der vorausgehenden Kapitel gewürdigt. Kapitel 9 fasst die Ergebnisse der Arbeit zusammen und zeigt anhand ausgewählter aktueller Fragestellungen, wie der präsentierte Vorschlag nutzbringend weiterentwickelt werden kann.

Eine Übersicht über die in den Handlungsschemata der Methodik enthaltenen Aufgaben bzw. Schritte in hierarchischer Gliederung ist in Anhang A8 dargestellt.

8 Fallstudie: Kundenauftragsrückgang in der Konsumgüterbranche

Die Anwendbarkeit der vorgestellten Methodik wird im vorliegenden Kapitel anhand einer Fallstudie überprüft. Hierzu wird ein typisches Szenario aus der Praxis eines Konsumgüterherstellers herausgegriffen und das Handlungsschema der Methodik durchlaufen. Hierbei liegt der Schwerpunkt auf der Planung sowie auf der nicht-automatisierten Anwendung, da eine Werkzeugunterstützung aktuell nicht verfügbar ist. Ziel der Darstellung ist demnach insbesondere die Demonstration der prinzipiellen Anwendbarkeit sowie der Potenziale der Methodik, die sich bei Erweiterung durch ein entsprechendes Software-Tool eröffnen.

Die Steuerung und Revision von Datenanalyseprozessen werden im Anschluss aus allgemeiner Sicht beurteilt und abschließend zu einer ganzheitlichen Einschätzung zusammengeführt.

8.1 Planung von Datenanalyseprozessen im Anwendungsfall

Auf eine Einführung in die Fallstudie wird bewusst verzichtet, da in der Praxis vor Projektbeginn in der Regel keine weiteren Informationen verfügbar sind außer einer allgemeinen Aussage wie jener, dass ein Problem im Hinblick auf einen beobachteten Auftragsrückgang zu lösen ist. Die Erarbeitung der Problembeschreibung wird von der Methodik abgedeckt und erfolgt im Zuge der Identifikation des Sachproblems.

8.1.1 Problemspezifikation

8.1.1.1 Identifikation des Sachproblems (Z1)

Ein Hersteller von Elektrogeräten hat im Rahmen des regulären Berichtswesens sowie durch Aussagen von Vertriebsmitarbeitern seit geraumer Zeit einen anhaltenden Rückgang des Auftragsvolumens in Bezug auf eine Produktgruppe festgestellt (**Problemerkennung Z1.1**). Ziel ist, dieses Problem zu beheben. Ein Analytiker wird beauftragt, hierzu eine Lösung zu entwickeln.

Er beginnt mit der **Diskurswelttabgrenzung (Z1.2)** und identifiziert zunächst als Problemobjekt die Kundenaufträge. Durch gezielte Befragung des Auftraggebers überführt er die allgemeine Problemstellung in eine präzise **Problembeschreibung (Z1.3)**. Gilt das Problem des Kundenauftragsrückgangs bereits als gelöst, wenn ein weiterer Rückgang gestoppt wird, oder soll der Auftragsbestand auf ein früheres Niveau angehoben werden? Oder wird im Grunde gar keine Erhöhung der Auftragsmenge angestrebt, sondern vielmehr eine Umsatzsteigerung? Dieses Ziel ließe sich gegebenenfalls auch bei rückläufigem Auftragseingang erreichen, wenn im Gegenzug die Auftragsvolumina steigen.

Attribut	**Problemaspekt**	
Name	Einbruch des Kundenauftragsvolumens	Erhöhung des Kundenauftragsvolumens
Typ	Ist	Soll
Beschreibung	Das Kundenauftragsvolumen ist im Zeitraum Januar-Dezember 2016 um 45% eingebrochen. Der Gesamtmarkt war im selben Zeitraum stabil (–0,3%).	Steigerung des Kundenauftragsvolumens um 2 Mio. EUR bis zum Juni 2017 zur Erfüllung der Umsatzziele.
Inhalt	Kundenauftrag.Volumen	Kundenauftrag.Volumen
Ausmaß	-45%	+2 Mio EUR
Zeitbezug	01/2016-12/2016	06/2017
Wertbeitrag	Umsatz	Umsatz
Kulisse	Gesamtmarkt stabil (-0,3%). Betroffene Produktlinie: B	
Kennzeichnung (Anwendung)	(Kundenauftrag.Volumen, –, Ist)	(Kundenauftrag.Volumen, +, Soll)

Tabelle 8: Problemaspekte der initialen Problembeschreibung im Anwendungsfall

Das Ergebnis wird in Form von zwei Problemaspekten festgehalten, die in Tabelle 8 gezeigt sind. Als Probleminhalt wird das konzeptuelle Klassenmerkmal „Volumen“ der Kundenaufträge definiert, da tatsächlich der Umsatz und nicht die Absatzmenge zu betrachten ist (Wertbeitrag). Für das Auftragsvolumen werden konkrete Ausprägungen genannt und mit einem Zeitbezug versehen. Als Kulisse des Ist-Zustands werden ein Vergleich mit dem Gesamtmarkt und die betroffene Produktlinie festgehalten.

8.1.1.2 Domänenanalyse (Z2)

Auf Grundlage der Problembeschreibung kann die Domänenanalyse starten, deren Ziel die Erstellung einer Problemkarte ist. Der erste Schritt ist nun die **Ergründung der Sichtweise des Auftraggebers (Z2.1)**, der wichtige Begriffe definiert und seine Vorstellungen und Erwartungen über mögliche Gründe und Lösungsoptionen beisteuert. Diese Einschätzungen bilden die Grundlage für die nachfolgenden Schritte. Zur **Konkretisierung des Problemobjekts (Z2.2)** ist anhand der heuristischen W-Fragen genauer zu bestimmen, welche Kundenaufträge vom Problem betroffen sind. Konkret ist z.B. zu hinterfragen, welche Kundengruppen, welche Produkte der Produktlinie oder welche Vertriebsregionen ein niedrigeres Auftragsvolumen zeigen. Glaubt z.B. ein Mitarbeiter, dass parallel mit dem Auftragsrückgang eine Erhöhung des Alters der Kunden eingetreten ist, so kann dies auf mangelnde Attraktivität der Produktpalette für die ursprüngliche Zielgruppe hindeuten. Auch die Beobachtung eines Kollegen, dass von einem bestimmten Vertriebspartner kaum mehr Aufträge eingehen, verweist auf mögliche Problemursachen, die es näher zu ergründen gilt. Zur Klärung dieser Sachverhalte werden zwei Problemaspekte zur näheren Problemanalyse und zur Ursachenanalyse als Differenzierung des Ist-Problemaspekts in der Problemkarte vermerkt. Zur Problemanalyse werden zwei Kind-Problemaspekte zur Ermittlung der betroffenen Kunden und Produkte sowie zum Branchenvergleich definiert. Der erste soll durch einen umfassenden Bericht aus dem Vertriebscontrolling datenanalytisch gelöst werden, der zweite durch eine Marktstudie. Für sie wird jeweils eine Informationsmaßnahme festgelegt.

Die Ursachenanalyse wird durch die **Identifikation von Einflussfaktoren (Z2.3)** unterstützt. Auch hierzu werden die Meinungen der Mitarbeiter in den Fachabteilungen abgefragt, die z.B. den Verdacht äußern, dass Preis und Qualität der eigenen Produkte im Vergleich zu den Mitbewerbern problematisch sind. Zur genaueren Analyse kann eine Ausweitung der Problemdomäne hilfreich sein, die eine Anpassung der Diskurswelt-abgrenzung nach sich zieht. Relevante weitere Diskursweltobjekte lassen sich mithilfe einer Domänenontologie identifizieren, die im Beispiel durch ein Interaktionsschema gemäß SOM repräsentiert wird (vgl. Abschnitt 4.7.1.4). Das Schema in Abbildung 109 zeigt eine einfache Wertschöpfungskette mit Lieferanten- und Kundenbeziehungen sowie ausgewählte Aspekte der globalen Umwelt. Die Aufnahme von Umweltobjekten, die nicht direkt mit dem Unternehmen interagieren, ermöglicht die Modellierung von Beziehungen zwischen Umweltobjekten, die als Problemursachen infrage kommen (z.B. der Verkauf von Konkurrenzprodukten an potenzielle Kunden durch Mitbewerber oder aus dem sozialen Umfeld des Kunden geäußerte Meinungen und Empfehlungen). Hierbei ist zu beachten, dass auch Transaktionen konzeptuelle Domänenobjekte repräsentieren.

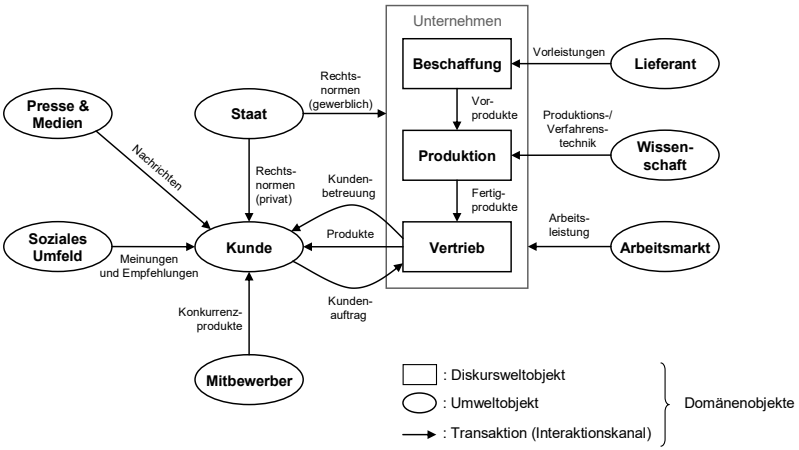


Abbildung 109: Interaktionsschema zur Identifikation von Einflussfaktoren auf den Kundenauftragsrückgang (eigene Darstellung)

Auf dieser Grundlage lassen sich für einzelne Domänenobjekte Ereignisse identifizieren und über zunächst hypothetische Kausalbeziehungen verknüpfen, sofern eine entsprechende (direkte oder transitive) Objektbeziehung existiert. Ereignisse repräsentieren in der Regel Zustandsänderungen eines Domänenobjektmerkmals. Kausalbeziehungen können gleich- oder gegenläufig sein. Im ersten Fall verändern sich die Zustände der referenzierten Objekte in dieselbe Richtung, im zweiten Fall in entgegengesetzte Richtungen (vgl. [Voit10, 126-128]). So steigt mit verbesserter Qualität der Vorleistungen der Lieferanten auch die Qualität der Vorprodukte der Produktion (gleichläufige Beziehung via Beschaffung), während erhöhte Attraktivität von Konkurrenzprodukten zu rückläufigem Volumen der Kundenaufträge bezüglich der eigenen Produkte führt (gegenläufige Beziehung via Kunde).[272] Die Kausalbeziehungen können durch Datenanalysen auf Gültigkeit überprüft werden.

Im Beispiel werden nach ausgiebiger Diskussion mit dem Auftraggeber die Qualität der eigenen Produkte sowie die Attraktivität der Konkurrenzprodukte für den Kunden als relevant erachtet und sollen bezüglich ihres Einflusses auf den Kundenauftragsrückgang hin näher untersucht werden. Hierzu werden zwei Problemaspekte zur Differenzierung der Ursachenanalyse definiert, die eine Überprüfung der Produktqualität sowie die Feststellung der Kundenzufriedenheit zum Gegenstand haben. Als geeignete Informationsmaßnahmen werden eine technische Prüfung und eine Datenanalyse (Produktqualität) sowie eine Kundenbefragung (Kundenzufriedenheit) für hilfreich erachtet und beauftragt.

Die **Ableitung von Handlungsoptionen (Z2.4)** erfolgt, nachdem die Ergebnisse der Informationsmaßnahmen vorliegen. Zur besseren Übersichtlichkeit wird dieser Schritt hier erläutert. Die Domänenanalyse begleitet das gesamte Projekt und schreibt die Problemkarte auf Grundlage gewonnener Erkenntnisse stetig fort. Die Ergebnisse legen nahe, dass einerseits tatsächlich Probleme mit der Qualität einiger Produkte bestehen, die es abzustellen gilt. Entsprechend wird ein neuer Problem-

[272] Die hier genannten Ereignisse beziehen sich jeweils auf Merkmale von Domänenobjekten, die eine Transaktion repräsentieren.

aspekt „Qualitätsprobleme beheben“ unterhalb des Ziel-Problemaspekts eingefügt. Andererseits wurde durch die Kundenbefragung deutlich, dass es eine Gruppe sehr loyaler Kunden gibt, die selbst nach einzelnen negativen Erfahrungen mit Produktmängeln zum Unternehmen stehen und eine große Weiterempfehlungsbereitschaft zeigen. Daher wird beschlossen, ein spezielles Programm zur Freundschaftswerbung zu initiieren, das als weiterer lösungsbezogener Problemaspekt vermerkt wird. Zu seiner Realisierung ist zuerst eine geeignete Kundengruppe auszuwählen, die sowohl loyal ist als auch Interesse zeigt, als Markenbotschafter an einem solchen Programm teilzunehmen. Anschließend ist für die identifizierten Kunden ein Kommunikationskonzept zu erarbeiten, das exakt auf die Merkmale der Kunden zugeschnitten ist. Voraussetzung dazu ist die Behebung der Qualitätsprobleme (sequenzielle Verknüpfung). Die Auswahl der Kundengruppe soll datenanalytisch unterstützt werden, weshalb eine Informationsmaßnahme zur Zielgruppenoptimierung festgehalten wird.

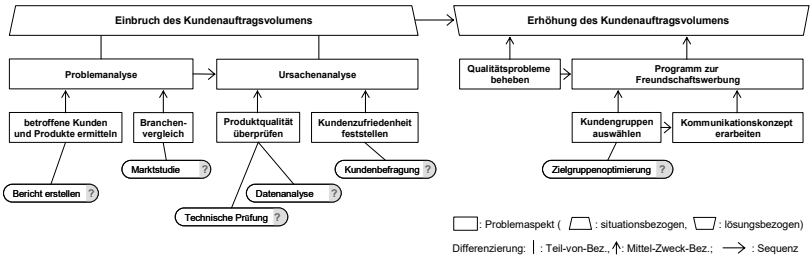


Abbildung 110: Integrierte Problemkarte zum Kundenauftragsrückgang (eigene Darstellung)

Die resultierende Problemkarte zeigt Abbildung 110. Sie kann eine Fortschreibung erfahren, wenn sich im weiteren Verlauf des Projekts neue Problemaspekte ergeben. Die **Problemkartierung (Z2.5)** erfolgt indes simultan mit den zuvor beschrieben Teilaufgaben.

8.1.1.3 Spezifikation des Analyseproblems (Z3)

Für jede Informationsmaßnahme, die datenanalytisch erfolgen soll, sind Analyseprobleme zu spezifizieren. Im Beispiel sind dies (1) der Bericht

zur Bestimmung der vom Auftragsrückgang betroffenen Produkte und Kunden, (2) die Datenanalyse zur Produktqualität, (3) die Kundenbefragung zur Ergründung der Kundenzufriedenheit sowie die (4) Zielgruppenbestimmung für das Programm zur Freundschaftswerbung.

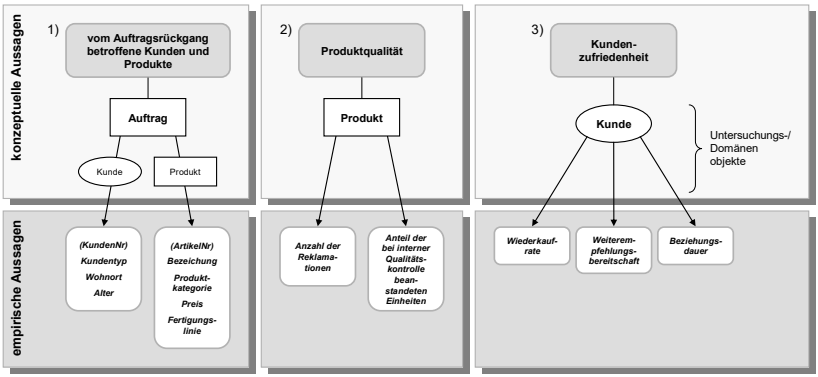


Abbildung 111: Operationalisierung von konzeptuellen in empirische Aussagen für den Kundenauftragsrückgang (eigene Darstellung)

Am Anfang steht die **Formulierung des Analyseziels (Z3.1)**. Hierzu ist jeweils ein konzeptuelles Merkmal eines Domänenobjekts als *Untersuchungsziel* zu spezifizieren und in empirische Indikatoren oder Merkmale zu operationalisieren. In manchen Fällen kann eine solche aus dem zugrundeliegenden Problemaspekt relativ einfach abgeleitet werden. So verweist der Problemaspekt „betroffene Kunden und Produkte ermitteln“ (1) unmittelbar auf die (Beziehungs-) Merkmale Kunde und Produkt des Objekts Kundenauftrag. Aus den Problemaspekten „Produktqualität überprüfen“ (2) und „Kundenzufriedenheit feststellen“ (3) sind hingegen nicht unmittelbar empirisch fassbare Eigenschaften der Produkte bzw. Kunden ableitbar. Ebenso lässt der Problemaspekt „Kundengruppen auswählen“ (4) offen, nach welchen Kriterien die Auswahl zu treffen ist. Abbildung 111 zeigt gültige Operationalisierungen für die Informationsmaßnahmen 1-3. Die betroffenen Kunden- und Produktmerkmale im Fall 1 werden transitiv über Beziehungen des Domänenobjekts Auftrag zu den betreffenden Objekttypen hergeleitet. Die Fälle 2 und 3 zeigen multidimensionale Operationalisierungen, die jeweils mehrere Aspekte der konzeptuellen

Begriffe erfassen. Sie führen jeweils zu mehreren Analysezielen, die sich jedoch später vereinigen lassen. Zunächst werden für die aus der Operationalisierung hervorgehenden Analyseziele geeignete *Analysefragen* formuliert. Abbildung 112 zeigt ausgewählte Ergebnisse. Für Fall 2 ist nur das Analyseziel in Bezug auf die Reklamationen dargestellt, und für Fall 3 ist eine typische Formulierung für eine Kundenbefragung in Bezug auf den Indikator Weiterempfehlungsbereitschaft dargestellt. Zur Zielgruppenbestimmung (4) wird die Antwortwahrscheinlichkeit als Indikator gewählt, die analytisch zu bestimmen ist. Zu jedem Analyseziel kann ein Informationsbedarfsprofil mit Anforderungen an die Analyseergebnisse definiert werden, auf dessen Darstellung an dieser Stelle verzichtet wird.

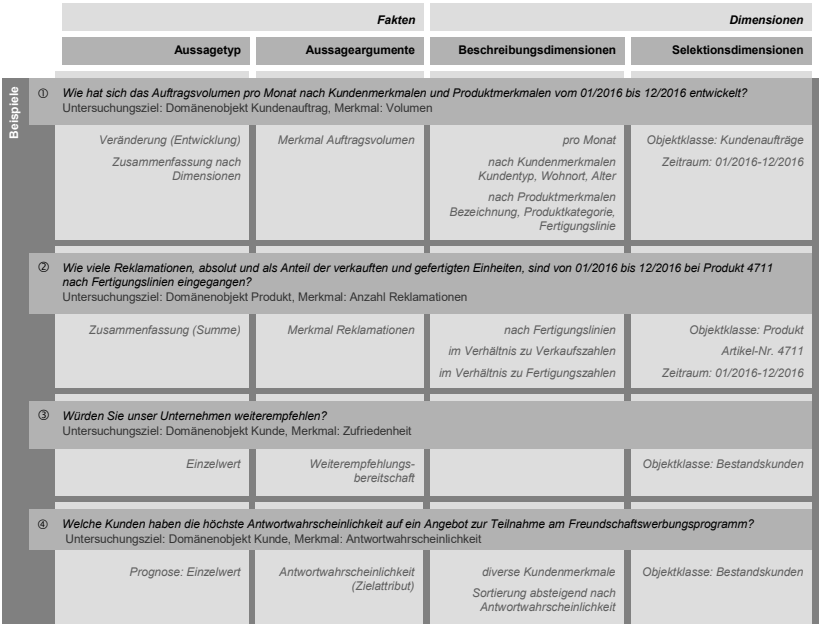


Abbildung 112: Ausgewählte Analysefragen zum Beispiel Kundenauftragsrückgang (eigene Darstellung)

Für jedes Analyseziel können mehrere geeigneter Analysedatenbestände (Informationsobjekt) definiert werden, was im Schritt **Formulierung des Analyseproblems (Z3.2)** erfolgt. Hierzu sind zunächst passende *Daten-*

quellen zu identifizieren. In Fall 3 werden im Rahmen der Kundenbefragung Primärdaten erhoben. In Fall 4 soll ein geeignetes Kundenprofil gewählt werden, welches das Untersuchungsobjekt Kunde durch zahlreiche Eigenschaften umfassend beschreibt. In Fall 2 gehen aus der Operationalisierung bereits die gewünschten Perspektiven hervor: Einerseits soll die Perspektive des Kunden (Anzahl der Reklamationen), andererseits die Perspektive der Produktion (Anzahl der Beanstandungen während der Qualitätskontrolle) betrachtet werden. Durch Sichtung verfügbarer Datenquellen zum Domänenobjekt Produkt wird nun deutlich, dass Reklamationen einerseits in Form von Beschwerden an den Kundenservice, andererseits in Form von Gewährleistungsfällen beim technischen Kundendienst auftreten. Aus dieser Differenzierung ergeben sich zwei verfeinerte Perspektiven, die jeweils zu eigenen Analyseproblemen führen.

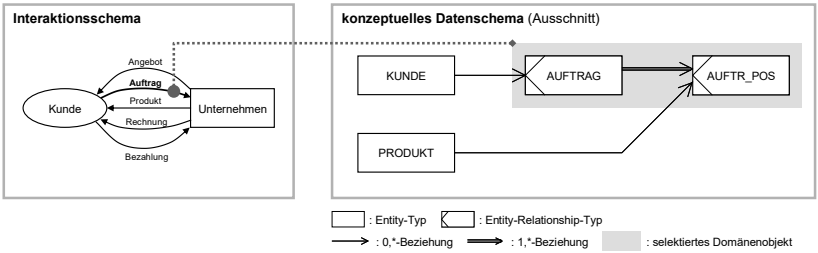


Abbildung 113: Auswahl von Informationsobjekten zur Bestimmung des Analyseobjekts am Beispiel Kundenauftragsrückgang (eigene Darstellung)

Für Fall 1 wird auf operative Daten aus dem Vertriebsinformationssystem (VIS) zurückgegriffen, an dessen Beispiel die Bestimmung des Analyseobjekts demonstriert wird. In der Regel umfasst eine Datenquelle mehrere verknüpfte Informationsobjekte, die durch ein konzeptuelles Datenschema dargestellt werden können. Abbildung 113 zeigt ein SER-Diagramm[273] zum Domänenobjekt Kundenauftrag. Durch Verfolgung der Abhängigkeiten im konzeptuellen Datenschema lassen sich zusätzlich zu den das Untersuchungsobjekt direkt repräsentierenden Relationen leicht weitere relevante Informationsobjekte

[273] Diagramm des Strukturierten Entity-Relationship-Modells (SERM), vgl. [FeSi13, 158ff.].

aufdecken. Im Beispiel sind dies der den Auftrag erteilende Kunde und die in den Auftragspositionen bestellten Produkte. Das zugehörige Analyseobjekt umfasst demnach die Informationsobjekte {VIS.AUFTRAG, VIS.AUFTR_POS, VIS.KUNDE, VIS.PRODUKT}.

Abschließend erfolgt eine **Konkretisierung und Strukturierung von Analysezielen (Z3.3)**, um zu bearbeitende Analyseketten zu bestimmen. Abbildung 114 zeigt links die Konkretisierung des Analyseziels „Produktqualität“ (Fall 2) anhand einer Zieldifferenzierung durch Unterscheidung zwischen Kundenreklamationen und Beanstandungen aus der internen Qualitätssicherung. Zugleich werden mit den Perspektiven „Vertrieb (VIS)“ und „Produktion (PPS)“ geeignete Datenquellen benannt. Aufgrund der identifizierten weiteren Datenquelle „Kundenservice (CRM)“ erfährt das Analyseproblem „Produktqualität 1“ anhand der Perspektive (P) eine erneute Konkretisierung, wobei die Merkmale entsprechend den verfügbaren Daten angepasst werden. Die resultierenden Analyseprobleme sollen von einer gemeinsamen Analyse behandelt werden, weshalb sie zur „Produktqualität 1.3“ vereinigt werden. Rechts in der Abbildung ist eine Analysekette gezeigt, die zur Behandlung des Kundenauftragsrückgangs realisiert werden soll.

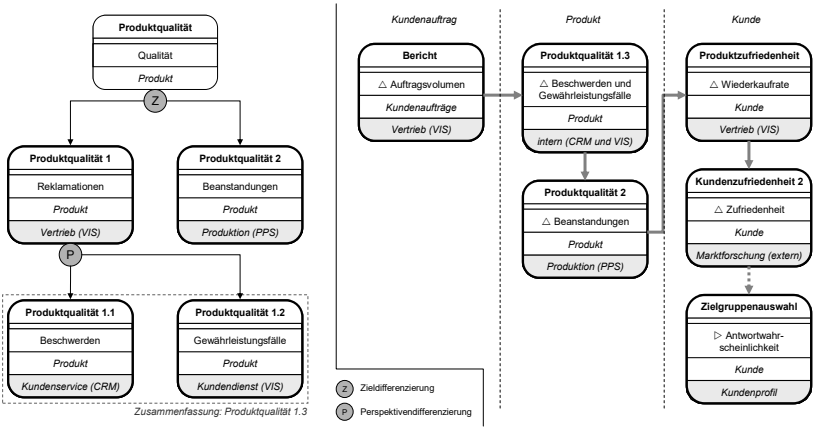


Abbildung 114: Konkretisierung und Strukturierung von Analysezielen zum Kundenauftragsrückgang (eigene Darstellung)

8.1.1.4 Untersuchungsdesign (Z4) und Projektplanung (Z5)

Auf Grundlage der Analysekette ist nun ein Untersuchungsdesign zu erstellen, das einerseits die geplanten Analysen aus methodischer Sicht überprüft, andererseits bei Bedarf eine Prozessspezifikation vornimmt. Die zu realisierenden Informations- und Handlungsmaßnahmen werden von der Projektplanung bearbeitet und in zugehörige Projektpläne transformiert.

8.1.2 Prozessspezifikation

Die Prozessspezifikation wird im Folgenden exemplarisch am Beispiel des Analyseproblems „Zielgruppenauswahl“ demonstriert. Zusätzlich wird auf die Datenselektion für das Analyseproblem „Bericht“ eingegangen.

8.1.2.1 Planung der Datenanalysephase (P1)

Die Planung der Analysephase beginnt mit der **Spezifikation der Analyseaufgabe (P1.1)**. Hierzu ist eine geeignete Funktion zu wählen, die sich aus Aussagetyp und Ausrichtung ergibt. Die Analyse zur Zielgruppenbestimmung zielt auf einen bislang unbekannten Einzelwert, um die Analysefrage zu beantworten, welche Kunden die höchste Antwortwahrscheinlichkeit für das Programm zur Freundschaftswerbung aufweisen. Es ist demnach eine schließende Analyse durchzuführen, welche die Funktion „Einzelwert deduzieren“ realisiert (vgl. Abschnitt 5.5.3.1). Diese Funktion verlangt neben den Eingabedaten auch ein noch nicht existierendes Prognosemodell, weshalb eine Zerlegung der Analyseaufgabe vorzunehmen ist. Das Ergebnis in der Notation des Modellierungsansatzes zeigt Abbildung 115.

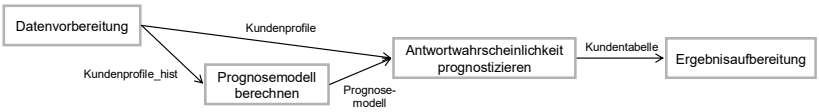


Abbildung 115: Zerlegung der Analyseaufgabe zur Zielgruppenbestimmung (eigene Darstellung)

Anschließend ist eine **Charakterisierung der Analysedaten (P1.2)**, wie sie im Analyseobjekt des Analyseproblems definiert sind, zu empfehlen, um Kenntnis über deren Eigenschaften zu erlangen. Auf dieser Basis kann die **Bestimmung einer Verfahrensklasse (P1.3)** erfolgen. Die wichtigste Restriktion, die ein Analyseverfahren zur Erfüllung einer Aufgabe genügen muss, ist die Funktion. Auf ihrer Basis können die verfügbaren Verfahren auf eine kleine Menge relevanter Kandidaten reduziert werden. Das zur Analysedurchführung gewählte Werkzeug KNIME enthält sieben Operatoren zur Realisierung der benötigten Funktion „Prognosemodell berechnen“[274] (Abbildung 116).

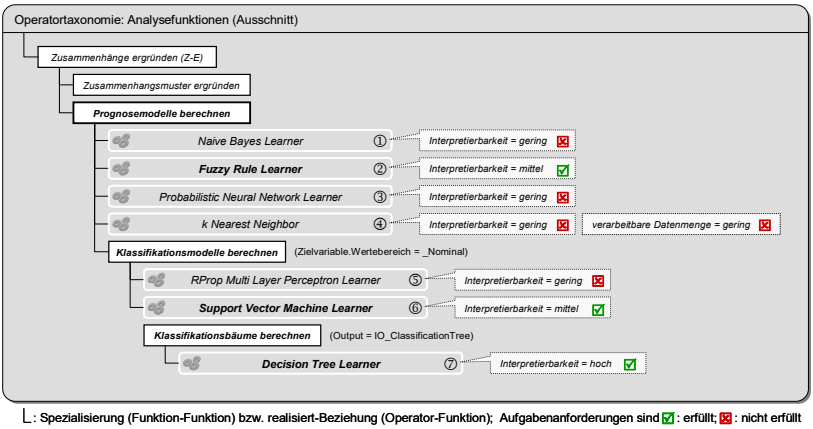


Abbildung 116: Beispiel zur Bestimmung und Einschränkung der funktional geeigneten Verfahrensklasse (eigene Darstellung)

Zur weiteren Reduzierung dieser Kandidatenmenge werden folgende Anforderungen als Restriktionen einbezogen: `Interpretierbarkeit der Ergebnisse >= mittel` und `verarbeitbare Datenmenge = hoch`. Sie können aus dem Informationsbedarfsprofil übernommen oder situativ bestimmt werden. Während Angaben zum zweiten Kriterium nur für das `k-Nearest-Neighbor`-Verfahren (4) bekannt sind und dessen Anwendbarkeit ausschließen, führt die Forderung nach

[274] Die Verfahrensmenge repräsentiert die in KNIME Version 2.9.4 standardmäßig implementierten Operatoren ohne Erweiterungsbibliotheken.

wenigstens mittlerer Interpretierbarkeit zu einer Eliminierung der Optionen 1, 3 und 5. Somit verbleiben der `Fuzzy Rule Learner` (2), der `Support Vector Machine Learner` (6) und der `Decision Tree Learner` (7) in der Kandidatenmenge. Aus ihr ist die **Auswahl eines Analyseverfahrens (P1.4)** zu bestreiten. Gemäß der in Abschnitt 5.5.3.4 (Seite 338) beschriebenen Bewertungsfunktion wird der `Decision Tree Learner` gewählt, der in Bezug auf Interpretierbarkeit, Genauigkeit und Transformationsbedarf die geringste Anzahl an Malus-Punkten erhält. Als **kontextabhängige Entwurfsentscheidung (P1.5)** wird eine Festlegung auf die 10-fache Kreuzvalidierung als Evaluationsansatz getroffen.

8.1.2.2 Planung der Datenvorbereitungsphase (P2)

Die Datenvorbereitungsphase wird durch die **Spezifikation der Datentransformationsaufgaben (P2.1)** funktional festgelegt. Hierzu sind analysemethodische Kenntnisse vor dem Hintergrund der Aufgabe sowie der in Schritt P1.2 erlangten Erkenntnisse über die Eigenschaften der Analysedaten anzuwenden, sofern nicht auf eine Fallbibliothek zurückgegriffen werden kann. Zur Orientierung mag die Gliederung in Datenselektion, Datenexploration und Datenmodifikation dienen (vgl. Abschnitt 5.5.4.1).

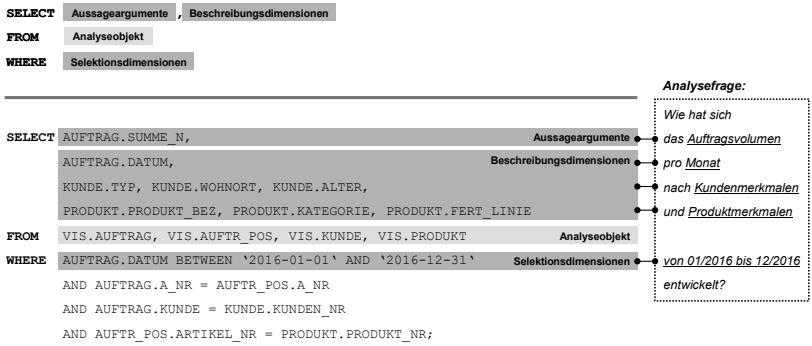


Abbildung 117: Datenselektion mithilfe der Analysefrage zum Kundenauftragsrückgang (eigene Darstellung)

Da zur Zielgruppenbestimmung Kundenprofile verwendet werden sollen (siehe Abschnitt 8.1.1.3) und hierzu keine weiteren Entscheidungen zu treffen sind, wird anhand des Analyseproblems zur Berichterstellung (Fall 1) gezeigt, wie sich zur Analyse der vom Auftragsrückgang betroffenen Kunden und Produkte zu selektierende Daten methodisch bestimmen lassen. Hierzu wird das in Abbildung 117 dargestellte Prinzip der Übertragung von Elementen der Analysefrage in SQL-Klauseln genutzt, nach dem Aussageargumente und Beschreibungsdimensionen als zu selektierende Attribute (Projektion), Selektionsdimensionen als zu wählende Datensätze (Selektion) und die Elemente des Analyseobjekts als Tabellen zu übernehmen sind.

Für die Berechnung des Prognosemodells zur Zielgruppenbestimmung werden zunächst alle verfügbaren Attribute des Kundenprofils selektiert. Als Datenmodifikationen sind Funktionen zur Behandlung fehlender Werte und die Aufteilung der Daten in Trainings-, Kalibrierungs- und Evaluierungsdaten vorzusehen. Diese Aufteilung ist für die in Schritt P1.5 eingeplante Kreuzvalidierung notwendig. Die **Zuordnung von Transformationsverfahren (P2.2)** gestaltet sich einfach, da das verwendete Werkzeug jeweils nur eine Option anbietet. Zur **Reihenfolgeplanung (P2.3)** können fachliche Abhängigkeiten definiert werden, etwa dass die fehlenden Werte vor der Aufteilung der Daten zu behandeln sind. Dies ist mithilfe des Werkzeugs möglich, indem die noch nicht verknüpften Aktivitätsknoten in der gewünschten Folge in Spalten angeordnet werden. Gemäß Modellierungsansatz wird die in Abbildung 115 enthaltene Datenvorbereitung zerlegt und die Zerlegungsprodukte mit Synchronisationskanten verknüpft (Abbildung 118). Die Überprüfung der Datenabhängigkeiten übernimmt das Analysewerkzeug, das nur die Verbindung kompatibler Aktivitäten erlaubt.

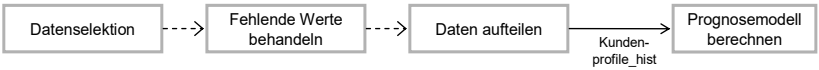


Abbildung 118: Fachliche Reihenfolgebeziehungen zur Berechnung eines Prognosemodells (eigene Darstellung)

8.1.2.3 Planung der Ergebnisaufbereitungsphase (P3)

Die Ergebnisaufbereitung reduziert sich im gegebenen Fall auf eine Funktion, welche die um das Zielattribut angereicherte Kundentabelle in eine Datenbank schreibt.

8.1.2.4 Instanziierung von Verfahrensparametern (P4)

Die Parameterinstanziierung erfolgt für alle Aktivitäten des Workflows gemeinsam. Die **Makroparametrisierung (P4.1)** betrifft die Zuordnung von Eingabeflüssen bzw. von deren Elementen zu Eingabedaten-Deklarationen des Operators und wird in Abschnitt 5.5.6.1 für den Decision Tree Learner illustriert. Die **Mikroparametrisierung (P4.2)** betrifft die Modusparameter der Operatoren. Sie startet bei der Datenselektion und verfolgt alle Datenabhängigkeiten bis zum Ende des Workflows. Sie kann mithilfe von Kontextregeln unterstützt werden, die grundsätzlich auch für die manuelle Planung geeignet sind. Eine Regel zur Auswahl geeigneter Modusparameterwerte für den Decision Tree Learner im hier relevanten Anwendungskontext `(Kunde.Antwortwahrscheinlichkeit, +, Soll)` ist in Abschnitt 5.5.6.2 dargestellt. Der Anwendungskontext wird vom Problemaspekt definiert, dem die Analyseaufgabe transitiv über Analyseproblem und Informationsmaßnahme zugeordnet ist. Die Kontextregel empfiehlt für die Parameter `quality_measure` das Maß „Gain Ratio“, als `pruning_method` die Option „MDL“ sowie die Aktivierung der Option `binary_nominal_splits`.

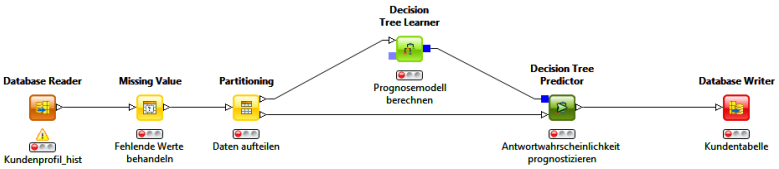


Abbildung 119: Einfacher KNIME-Workflow zur Zielgruppenselektion (Screenshot)

8.1.3 Bewertung: Planung von Datenanalyseprozessen

Das Fallbeispiel zeigt, wie durch Einsatz der Methodik im Rahmen der Problemspezifikation eine wirksame Systematisierung des Vorgehens und eine verständliche Beschreibung auszuführender Analysen erreicht werden kann. Die Methodik wurde vom Verfasser in zahlreichen Praxisprojekten im CRM sowie im Smart Farming und zur Leistungsoptimierung von Cloud-Infrastrukturen genutzt. Seitens der Auftraggeber wird insbesondere der intuitive Ansatz zur Strukturierung von Sachproblemen sowie zur Formulierung von Analysezielen als Analysefragen begrüßt. In der Kommunikation mit Entwicklern und Software-Herstellern wurde wiederholt die klar strukturierte Definition von Analyseproblemen positiv hervorgehoben, die eine einfache und direkte Realisierung der Analysen und Datentransformationen erlaubt. Zur Unterstützung der Problemspezifikation haben sich Dokumentvorlagen (Formulare) bewährt, die auf Grundlage der in Anhang A4 aufgeführten Attributschemata erstellt werden können.

Die vereinfacht geschilderte Vorgehensweise der Prozessspezifikation produziert einen funktionsfähigen Workflow, der über die minimal erforderlichen Funktionen hinaus weitere Aufgaben zur Qualitätssicherung berücksichtigt (Abbildung 119). Dieser Prozess produziert stark verbesserungsfähige Ergebnisse und sollte wenigstens durch eine Normalisierung der Attribute erweitert werden. Derartiges Wissen muss entweder experimentell durch Versuch und Irrtum generiert oder kann durch jene Aspekte der Methodik bereitgestellt werden, die nur mit einer entsprechenden Werkzeugunterstützung effektiv realisierbar sind (Wiederverwendung, komplexe Kontextregeln, semantische Prozessplanung). Gleichwohl zeigt das Anwendungsbeispiel, dass das Handlungsschema zur Prozessspezifikation prinzipiell auch manuell nutzbar und durch die erreichbare Systematisierung des Vorgehens nützlich ist.

8.2 Bewertung: Steuerung von Datenanalyseprozessen

Auf die Darstellung der Steuerung sowie der Revision für den Anwendungsfall wird verzichtet, da das hierfür jeweils notwendige Instanzmodell nur bei Automatisierung verfügbar ist. Zudem stellen die Kapitel 6 und insbesondere 7 die Vorgehensweise bereits ausführlich dar. Daher

erfolgt für diese beiden Managementphasen eine allgemeine Einschätzung.

Aus Sicht der Steuerung von Analyseprozessen, die häufig ungeplant ausgeführt werden bzw. vom gegebenen Plan abweichen, erscheint der Ansatz zur Zielkontrolle nützlich. Der in Abschnitt 6.2.3.3 beschriebene Ablauf zur Überwachung und Behandlung von Zielabweichungen ist prinzipiell manuell ausführbar und definiert klare Kriterien, wann eine Abweichung vorliegt und wie sie im situativen Kontext zu handhaben ist. Insbesondere ist im Falle einer Zielabweichung nicht zwingend das ursprüngliche Ziel weiterzuverfolgen und das neue Ziel zu verwerfen. Nicht weiterzuverfolgende Ziele können archiviert oder zur späteren Abarbeitung vorgemerkt werden. Die konkrete Erkennung von Zielabweichungen ist wiederum nur durch eine Werkzeugunterstützung in letzter Konsequenz realisierbar, weil die Fokussierung des Analytikers auf den Prozess und die Verfahren eine stetige und systematische manuelle Beachtung von Zielabweichungen unrealistisch erscheinen lässt. Zur automatisierten Erkennung von Zieldiskrepanzen ist zusätzlich eine semantische Annotation der Prozesselemente notwendig.

Zur Projektsteuerung wird in Abschnitt 3.3.3 ein Vorgehensmodell präsentiert, das evolutionäre Prinzipien sowie Erkenntnisse aus der Software-Entwicklung auf die Datenanalyse überträgt. Hier bleibt zu überprüfen, inwiefern dieser Vorschlag den Anforderungen an agile Analyseprojekte genügt. Der Ansatz zur Messung des Projektfortschritts (durch Zählung von Inkrementen und Zyklen) ist wiederum nur mithilfe eines zugehörigen Software-Werkzeugs konsequent nutzbar. Die Differenzierung in Projekt- und Analyseebene erlaubt den Austausch oder den Wechsel des Analyseansatzes, ohne eine Änderung des Vorgehens auf Projektebene nach sich zu ziehen, wie dies bei monolithischen bzw. disziplinenspezifischen Modellen oft der Fall ist. Zum anderen erlaubt sie innerhalb eines Projekts mehrere verkettete Analysen durchzuführen, deren Anzahl sich oft erst im Zuge der Untersuchung ergibt. Die infolge der Ebenenstruktur erfolgte Aufspaltung der Problemspezifikation erscheint zunächst diskussionswürdig, wird durch die Handlungsschemata der Methodik jedoch bestätigt, da die Behandlung der Sachprobleme das gesamte analytisch

gestützte Projekt betrifft und demnach auf Projektebene zu verorten ist, während die Lösung von Analyseproblemen durch Analyseprozesse auf Prozessebene zu sehen ist.

8.3 Bewertung: Revision von Datenanalyseprozessen

Die Revision von Analyseprozessen nach der vorgestellten Methodik führt zu einer umfassenden Beurteilung des gesamten analytisch gestützten Projekts und berücksichtigt hierbei insbesondere auch die Schachtelung von Teilprojekten. Da zur Revision weitere Datenanalysen notwendig werden können, ist deren Einbeziehung in den Gesamtzusammenhang vorteilhaft, um die korrekte Zurechnung von Kosten- und Nutzengrößen zu erlauben. Die damit einhergehende Komplexität der Projekte wird durch die Modellierung aller Vorhaben in der Problemkarte besser handhabbar.

Das dreistufige Vorgehen (Revision im engeren, im weiteren bzw. im lernenden Sinne) erlaubt eine kontextabhängige Justierung der durchzuführenden Beurteilungen. Die Ergebnisbewertung wird um eine Prozessbeurteilung ergänzt, die Fehler und Schwachstellen aufdecken kann. Die Evaluation der Handlungsmaßnahmen ist zur Feststellung der Effektivität und des Wertbeitrags der Analysen notwendig und wird im Sinne der wissenschaftlichen, systematischen Evaluation unterstützt. Lerneffekte werden explizit befördert, indem Analyseprozesse gezielt verbessert und in einer Fallbibliothek abgespeichert werden können. Die Wiederverwendung verspricht großes Potenzial zur Bescheunigung und Qualitätssicherung komplexer Vorhaben, ist jedoch nur softwaregestützt sinnvoll nutzbar.

8.4 Zusammenfassende Einschätzung

In der Gesamtschau ist festzuhalten, dass die vorgelegte Methodik zum Management von Datenanalyseprozessen eine breite Reichweite besitzt und in der Lage ist, praxisrelevante Projekte und Prozesse abzubilden und systematisch zu planen, zu steuern und zu kontrollieren. Sie definiert einen Rahmen für die in diesem Umfeld auszuführenden Aufgaben und zeigt mithilfe umfassender Handlungsschemata und

Handlungsempfehlungen, wie diese systematisch realisiert werden können. Die Komplexität des Ansatzes ist in seiner Breite und Tiefe jedoch nur durch Werkzeugunterstützung zu bewältigen.

Für den praktischen Einsatz erscheint sie dennoch auch zur manuellen Nutzung hilfreich, da sie das Vorgehen strukturiert und im Sinne einer Checkliste auf relevante Aspekte verweist. Gerade bei der Problemspezifikation hat sich in realen Szenarien gezeigt, dass sie auch „mit Papier und Bleistift" anwendbar und nützlich ist. Die Problem- und Zielorientierung erscheint nicht zuletzt im Kontext von Big Data und Data Science überaus relevant, wo häufig ein rein datenorientiertes Vorgehen zu beobachten ist, das zunächst ohne konkrete Zwecksetzung und zuweilen ohne Beachtung rechtlicher und ethischer Grundsätze erfolgt.

Die Arbeit mag auch als theoretische Grundlage für weitere Arbeiten dienen.

9 Fazit und Ausblick

Die vorliegende Arbeit präsentiert eine umfassende, interdisziplinäre Methodik zum Management von Datenanalyseprozessen, die im Sinne einer Analysestrategie verstanden werden kann. Sie betrachtet die Planung, Steuerung und Revision dieser Prozesse und bezieht die Problemspezifikation, die Prozessspezifikation und die Ressourcenspezifikation ein. Damit gestattet sie eine in Bezug auf die für Datenanalysevorhaben relevanten Modellierungsobjekte vollständige Repräsentation. Diese Breite ist nicht in allen analytischen Anwendungsfällen erforderlich, erlaubt jedoch die Beschreibung selbst höchst komplexer Untersuchungen. Zugleich bereitet sie die Basis für weitere Arbeiten im Sinne eines umfassenden Bezugsrahmens. Die Methodik beinhaltet Handlungsschemata mit Empfehlungen und Heuristiken für einzelne Schritte, die auch ohne Werkzeugunterstützung anwendbar und nützlich sind.

Die folgenden Abschnitte zeigen konkrete Ergebnisse und Beiträge der Arbeit auf und verweisen auf weiteres Forschungspotenzial.

9.1 Fazit

Der Einsatz der Methodik kann dazu beitragen, Datenanalysen im betrieblichen Umfeld systematisch zu konzipieren sowie analytisch gestützte Projekte ganzheitlich zu evaluieren. Von besonderem Wert für den praktischen Einsatz erweist sich die Unterstützung der Problemspezifikation, die im Dialog mit dem Auftraggeber bzw. der Fachabteilung auch ohne Werkzeugunterstützung unmittelbar realisiert werden kann. Die Strukturierung von Sachproblemen und die Ableitung von Analyseproblemen befördern ein zielorientiertes Vorgehen und leisten einen Beitrag zur Planung und Dokumentation umfangreicher Vorhaben.

Die Orientierung am Handlungsschema zur Prozessspezifikation gestattet ein systematisches Vorgehen, das sich in mehrere Teilaufgaben gliedert, die bei manueller Planung im Sinne einer Checkliste nutzbar sind. Die Potenziale der Methodik, wie sie insbesondere die semantische Annotation und die Wiederverwendung von Prozessbausteinen

bieten, sind jedoch nur bei entsprechender Werkzeugunterstützung vollständig nutzbar. Dies gilt analog für die Revision, die nur dann in vollem Umfang realisierbar ist, wenn das Prozessinstanzmodell automatisiert verwaltet und stets aktuelle Prozessparameter bereitgestellt werden. Gleichwohl zeigt auch hier das Handlungsschema auf, wie eine ganzheitliche, auf sachliche Anforderungen gerichtete Evaluation von Datenanalyseabläufen erreichbar ist.

In der vorliegenden Arbeit wird ein Werkzeug-Konzept ausdrücklich nicht angestrebt. Ihr Beitrag ist aus dieser Perspektive als Rahmenkonzept im Sinne einer Idealvorstellung zu sehen, aus dem funktionale Anforderungen zur Entwicklung geeigneter Werkzeuge ableitbar sind.

Die folgende Aufstellung gibt weitere konkrete Erkenntnisse und Beiträge, gegliedert nach den Hauptkapiteln der Arbeit, wieder.

Datenanalyse und Datenanalyseprozesse:

- Die Datenanalyse lässt sich interdisziplinär als Aufgabe zur Informationserzeugung auf Basis von Datentransformations- und -interpretationsvorgängen definieren, ohne eine Einschränkung auf konkrete Paradigmen, methodische Ansätze oder Verfahren vorzunehmen.
- Datenanalysen können als Prozesse beschrieben werden, die Ziel-, Transformations-, Verkettungs- und Ressourcenaspekte umfassen. Somit lässt sich die Methodik auf höchster Betrachtungsebene unter Rückgriff auf etablierte Ansätze des Prozessmanagements in die Phasen Planung, Steuerung und Revision (Kontrolle) gliedern.

Vorgehen bei der Datenanalyse:

- Der idealtypische Ablauf von Datenanalyseprozessen aller untersuchten Disziplinen kann auf ein allgemeines Prozessmodell zurückgeführt werden, das als Ausgangspunkt für ein Vorgehensmodell sowie für den Entwurf einer interdisziplinären Methodik dienen kann.
- Der tatsächliche Ablauf von Datenanalyseprozessen erfolgt häufig iterativ-inkrementell und führt zu mitunter enormer Handlungs-

komplexität, welcher mithilfe allgemeiner Bewältigungsstrategien aus der Komplexitätstheorie zu begegnen ist.

- Zur Bewältigung der Handlungskomplexität eignet sich ein evolutionäres Vorgehensmodell, das Erfahrungen aus der Softwaretechnik aufgreift, und das die Dynamik der Prozessrealisierung von den eher statischen Aspekten des Anwendungsprojekts durch eine Differenzierung in zwei Ebenen trennt.

Modellierung von Datenanalyseprozessen:

- Das Modellsystem zur Repräsentation von Datenanalyseprozessen wird anhand einer Vier-Ebenen-Architektur gegliedert, die zur Komplexitätsbewältigung beiträgt und zugleich die Grundlage für die Strukturierung der Methodik bildet, deren Handlungsschemata jeweils auf einzelne oder mehrere der Architekturebenen Bezug nehmen.
- Die Repräsentation des Kontexts von Datenanalyseprozessen kann durch Strukturschemata, semi-formale Modelle und Ontologien unterstützt werden. Die im Zuge der Top-down-Planung vorgenommene Aufgabendekomposition liefert den strukturellen Kontext für Prozessaktivitäten und Prozessbausteine. Das Kontextmodell kann als Projektion auf das integrierte Metamodell dynamisch erzeugt werden.

Planung von Datenanalyseprozessen:

- Die Problemspezifikation auf Analyseebene kann durch systematische Eingrenzung und Strukturierung der Problemdomäne auf Anwendungsebene vorbereitet werden, die mithilfe von Strukturmodellen, der Entscheidungstheorie, der statistischen und sozialwissenschaftlichen Methodenlehre sowie mithilfe von Heuristiken unterstützt werden können.
- Die Analyse des fachlichen Informationsbedarfs erfolgt mithilfe natürlichsprachiger Fragen, die durch ein Strukturschema konkretisiert und durch weitere Anforderungen in Form des Informationsbedarfsprofils erweitert werden. Auf dieser Basis wird die zielorien-

tierte und revisionsgerechte Ableitung von Analyseproblemen aus fachlichen Untersuchungsproblemen (Sachproblemen) ermöglicht.

- Die Auswahl geeigneter Analysedatenquellen wird mithilfe des Konzepts der Perspektive unterstützt, welche die vom Erhebungsobjekt (Datenerfassung) eingenommene Sicht auf das Untersuchungsobjekt repräsentiert.
- Die Komplexität der Prozessspezifikation kann durch die Kombination hierarchischer mit dekompositorischen Prinzipien bewältigt werden. Dabei erfolgt die Planung stufenweise von der Analyse- über die Prozess- bis zur Ressourcenebene der Analysearchitektur sowie abschnittsweise bezüglich der Teilprozesse Analyse, Datenvorbereitung und Ergebnisaufbereitung.
- Die Prozessplanung kann durch vier Basisansätze unterstützt werden, die sich aus den Dimensionen Innovationsgrad (Neuplanung/Wiederverwendung) und Vorgehensrichtung (top-down/ bottom-up) ergeben und auch kombiniert zur Anwendung gelangen können. Für die Wahl der Ansätze können effizienzbezogene Kriterien bzw. Prioritäten definiert werden.

Steuerung von Datenanalyseprozessen:

- Aufgrund der eingeschränkten Planbarkeit von Datenanalyseprozessen sind bei ihrer Steuerung in unterschiedlichem Maße Gestaltungsaspekte zu berücksichtigen, die zu den drei Steuerungsmodi Repetition, Deviation und Innovation für repetitive (standardisierte), variable bzw. Ad-hoc-Prozesse führen. Die Prozesssteuerung muss einen flexiblen Wechsel zwischen den Modi zulassen.
- Im Rahmen der Steuerung ist eine systematische Zielkontrolle möglich, welche die Feststellung von Zielabweichungen im aktuellen Analysepfad sowie die relevanzorientierte Neuausrichtung auf das ursprüngliche oder ein neues Analyseziel erlaubt.

Revision:

- Die Beurteilung von Analyseergebnissen und Analyseabläufen erfolgt als Kontrolle der Zielerreichung in Bezug auf definierte

Analyseziele bzw. Sachprobleme (Zielzustände). Die Burteilungskriterien werden systematisch dokumentiert und gemäß der Evaluationsforschung bewertet. Die Rückkopplung mit der Anwendungsebene dient der Feststellung des fachlichen Nutzens und Wertbeitrags des analytisch gestützten Projekts.

- Die während der Revision vorgenommenen Bewertungen und Modifikationen ausgeführter Analyseprozesse etablieren einen Lernprozess, der in seinem Verlauf die Effektivität und Effizienz der Datenanalyse fördert. Die aus realisierten Prozessen gewonnenen Erfahrungen können in künftigen Projekten direkt (Prozessvorlagen oder -bausteine) oder indirekt (Heuristiken und allgemeines Wissen über Datenanalysen) wiederverwendet werden.

Anwendung:

- Die Methodik ist modular strukturiert und enthält „Sollbruchstellen“, um eine auf den situativen Kontext abgestimmte Vorgehensweise zu erlauben. Die Betrachtung aller vier Ebenen der Analysearchitektur ist nicht zwingend; vielmehr können alle Ebenen prinzipiell auch eigenständig bearbeitet werden.
- Die Anwendung der Kriterien des Prozessmanagements auf die Datenanalyse ist möglich, jedoch nicht in jedem Falle sinnvoll. Insbesondere im Rahmen der Revision ist die umfassende Begutachtung des Analyseablaufs häufig nur für repetitive oder operative Analyseprozesse erforderlich, und die lernende Revision erscheint hauptsächlich in größeren Organisationen von Vorteil, wo mehrere Analytiker von geteilten Erfahrungen profitieren.

9.2 Ausblick

Die Ergebnisse der Arbeit bieten zahlreiche Ansatzpunkte für weitere Forschungs- oder Entwicklungsvorhaben. Im Folgenden werden einige Aspekte herausgegriffen, die als besonders relevant erachtet werden.

Das Konzept der Problemspezifikation auf Analyseebene, das eine Informationsbedarfsanalyse auf der Grundlage intuitiv formulierter Analysefragen vorsieht, stellt eine flexible, informale und natürliche

Form der Kommunikation mit dem Auftraggeber dar. Sie hat sich in mehreren Praxisprojekten des Verfassers sowohl im Bereich Data Mining als auch im Reporting bewährt, um auch „Ad-hoc-Probleme“, wie von HUBER [Hube97, 189] (vgl. Abschnitt 1.1) beschrieben, unkompliziert zu erfassen. Es liegt nahe, sie vor dem Hintergrund der zunehmenden Bedeutung von *agilen Methoden* bei der Entwicklung von Business-Intelligence-Systemen zu betrachten [ZKTG12]. Diese Methoden sollen der hohen Änderungsdynamik bei der Erstellung von Software-Systemen Rechnung tragen und erscheinen gerade im explorativ-experimentellen Umfeld von Vorteil, weshalb sie bei der Mehrheit der Industrieunternehmen bereits zum Management datenanalytisch gestützter Projekte im Einsatz sind [LPDK16, 9]. Die Analyseziele der vorgestellten Methodik spezifizieren einen klar definierten Informationsbedarf aus Anwendersicht und sind mit den „User Stories“ von Scrum, dem populärsten Vertreter agiler Methoden [Wilm12, 16], vergleichbar. Ähnlich können noch nicht realisierte Analyseketten als „Product Backlog“ interpretiert werden. Es bietet sich eine Untersuchung an, inwiefern der hier vertretene Ansatz, auch im Zusammenspiel mit dem evolutionären Vorgehensmodell aus Abschnitt 3.3, mit agilen Methoden kompatibel ist bzw. diese für Datenanalyseprojekte ersetzen kann.

Die Methodik ist explizit für alle Disziplinen der Datenanalyse konzipiert. Auf Prozessebene geht sie jedoch von einer Gliederung in definierte Aufgaben aus, die als Schritte eines Analyseablaufs von einem Analysewerkzeug ausgeführt werden. Diese klare Struktur ist in der programm- bzw. *skriptbasierten Data Science* typischerweise nicht zu beobachten (vgl. Abschnitt 2.2.2.7). Zwar unterstützen gängige Datenanalyse- und -transformationswerkzeuge die Kapselung von Code- oder Skriptfragmenten in Aktivitäten, dies ist für umfangreichere Programme jedoch nicht üblich. Zugleich führt die Branche aktuell erste Diskussionen, welche die Effizienz des unstrukturierten Vorgehens infrage stellen, weil Code nicht systematisch wiederverwendet und die Konsistenz des von mehreren Analytikern erstellten Codes nicht gewährleistet werden können. Lösungsvorschläge beinhalten Versionskontrolle, Dokumentation und Modularität der erstellten Programme [Pyth17]. In diesem Zusammenhang wäre zu untersuchen, inwiefern

die vorgestellte Methodik einen Lösungsbeitrag liefern kann bzw. ob sie zur Unterstützung skriptbasierter Datenanalysen zu erweitern ist.

Die Wiederverwendung von Prozessartefakten und die Extraktion von Erfahrungswissen versprechen wirksame Unterstützung bei der Prozessgestaltung, setzen jedoch eine gewisse Anzahl an Workflows bzw. Analyseabläufen voraus, aus denen dieses Wissen zu ziehen ist. Hierfür bietet sich die *Kollaboration* mehrerer Analytiker bzw. Organisationen an, die ihr Wissen teilen und auf diese Weise insgesamt auf einen größeren Erfahrungsschatz zurückgreifen können. Derartige Ansätze sind unter dem Begriff *Networked Science* bekannt und richten sich in erster Linie an Wissenschaftler, da effektive industrielle Analyseprozesse einen Wettbewerbsvorteil darstellen und demnach nicht geteilt werden. Dennoch scheint es lohnend, diese Form der Kollaboration daraufhin zu untersuchen, ob sie helfen kann, die Potenziale der Wiederverwendung besser auszuschöpfen. Für das Maschinelle Lernen hat sich z.B. mit OpenML eine offene Plattform etabliert, die von zahlreichen Analysewerkzeugen mit entsprechenden Plug-Ins zum direkten Datenaustausch unterstützt wird [VRBT13]. Die Abstraktion konkreter Workflows zu Schablonen könnte ein Lösungsansatz sein, um fallspezifische Details von der Veröffentlichung auszuschließen. Alternativ sind geschlossene Gruppen oder von Unternehmen ausschließlich für den internen Gebrauch betriebene Plattformen denkbar.

Schließlich ist die *Entwicklung eines Software-Werkzeugs* hilfreich, das einzelne Aspekte oder die gesamte Methodik zum Management von Datenanalyseprozessen abbildet. Damit ließen sich auch ihre bislang nur theoretisch untersuchten Potenziale realisieren. Der Modellierungsansatz sieht eine Werkzeugunterstützung bereits vor, indem er einerseits zur Repräsentation aller als bedeutsam erkannten Kontextfaktoren zugehörige Attribute an den Metaobjekttypen enthält. Andererseits sind zur Automatisierung relevante Aspekte, wie z.B. die formale Semantik von Workflows (vgl. Abschnitt 4.5.2.4) und insbesondere die semantische Prozessplanung nach dem Ansatz von HEINRICH ET AL. [Hein+08] explizit berücksichtigt. Die für diesen Planungsansatz notwendigen Parameterdeklarationen lassen sich unmittelbar durch Projektion auf

das Attributschema des Metaobjekttyps Datenobjekttyp generieren (vgl. Abschnitt 4.6.1.1).

Die genannten Erweiterungsmöglichkeiten seien als Anregung für Forschungsthemen empfohlen. Es ist künftig von weiterem Bedeutungszuwachs der Datenanalyse auszugehen. Zur Rechtfertigung und Priorisierung analytischer Untersuchungen sowie zur Qualitätssicherung ihrer Ergebnisse erscheint methodische Unterstützung, wie sie diese Arbeit vorschlägt, überaus hilfreich. Mit zunehmender Verfügbarkeit und Benutzerfreundlichkeit analytischer Funktionen gewinnt nicht zuletzt die Vermeidung des „Fehler der dritten Art“ an Gewicht, damit wir die richtigen, relevanten Fragen an Analysesysteme richten.

Anhang

A1 Überblick über gängige Datenanalysemethoden

Die Vielzahl der verfügbaren Datenanalysemethoden ist selbst für erfahrene Analytiker unübersichtlich. Die stetige Proliferation von Analyseverfahren erfährt mittlerweile zunehmend Kritik in der Literatur. So bemängelt etwa HAND, dass die Entwicklung immer neuer Methoden ohne theoretische Fundierung, ohne kritische Beurteilung und Vergleiche mit existierenden Verfahren zu nur bescheidenem Fortschritt geführt hat und häufig losgelöst von einem konkreten Anwendungsproblem erfolgt. In der Praxis werden letztlich die in Analysewerkzeugen implementierten Methoden genutzt, während von der Mehrheit der neuen Vorschläge keine Kenntnis genommen wird [Hand99, 7].

Zur Datenanalyse eignen sich allgemein analytische, konnektionistische und wissensbasierte Methoden [Zimm95, 8]. Zudem ist im Rahmen der Interpretation stets die menschliche Intuition und Urteilsfähigkeit erforderlich. Die meisten Methoden können als Erweiterung oder Kombination einiger grundlegender Verfahren angesehen werden [FaPS96, 12], deren Ursprünge in der Statistik, der Künstlichen Intelligenz, der traditionellen Mustererkennung, der Datenbanktechnik, der Computerlinguistik, dem Information Retrieval und der Computergrafik liegen [Küst01, 95f.].

Die folgende Tabelle 9 zeigt eine einfache Systematik bedeutender Analysemethoden, ohne Anspruch auf Vollständigkeit zu erheben. Die einzelnen Verfahrensklassen sind anhand einer Kurzbeschreibung und einiger typischer Vertreter charakterisiert und nach ihrem Anwendungsschwerpunkt (Basisansatz) geordnet. Die Wahl der Klassen orientiert sich an gängigen Einteilungen der Datenanalyseliteratur. Da einige Verfahrensklassen für mehrere Zwecke geeignet sind, existieren Querbezüge. So ist die Klasse der künstlichen Neuronalen Netze z.B. für die Cluster-, Diskriminanz-, Regressionsanalyse und Prognose einsetzbar. Eine detaillierte Beschreibung einzelner Methoden würde den Rahmen dieser Arbeit sprengen und ist für deren Zielsetzung nicht erforderlich. Hierfür sei auf die umfangreiche Literatur verwiesen. Ausführliche Darstellungen wichtiger Analysemethoden bieten z.B. [Küst01], [KüKa01], [Pete03], [BEPR11], [Cues13], [ClLä14], [EMC15].

Bedeutende Datenanalysemethoden	Ausrichtung		
Verfahrensklassen Beschreibung → *exemplarische Vertreter*	explorativ	konfirmatorisch	schließend
Kenn- und Maßzahlen	☑	☑	☐
Berechnung deskriptiver statistischer Größen zur Charakterisierung der Gesamtpopulation → *Lagemaße (Mittelwerte, Mediane, etc.), Streuungsmaße (Varianz, Standardabweichung, Konfidenzintervalle, etc.), Verhältniszahlen (Gliederungs-, Beziehungs-, Indexzahlen)*			
Berichte	☑	☑	☐
Bereitstellung themen- oder bereichsbezogener empirischer Größen (Kennzahlen) → *klassisches Reporting, OLAP-Berichte*			
Visualisierungen	☑	☐	☐
grafische Darstellung statistischer Größen → *Box-Plots / Box-and-Whisker-Plots, Punktwolken (Scatter-, Linien-, Coplots etc.), Stem-and-Leaf-Plots, Histogramme, dynamische Grafiken, Berechnung von Kernschätzern (Dichtefunktionen), multidim. Skalierung, Beziehungsgraphen und -netze*			
Abweichungsanalyse	☑	☐	☐
Erkennung und Diagnose von Abweichungen, Ausreißern und unerwarteten Ereignissen → *Ausreißeranalyseverfahren in Regressionsmodellen, exponentiellen Glättungsmodellen und ARIMAX-Modellen; Frühwarnsysteme, Heuristiken*			

(Fortsetzung auf nächster Seite)

Assoziationsanalyse	☑	☐	☐
Ermittlung struktureller Zusammenhänge oder Beziehungen zwischen Objekten → *boolsche Assoziationsverfahren (z.B. Apriori-Familie, SETM, AIS)*			
Clusteranalyse und Segmentierung	☑	☐	☐
explorative Konstruktion von Objektklassen, die in sich möglichst homogen, untereinander aber möglichst heterogen sind → *multivariate statistische Clusteranalyse: agglomerativ-hierarchische Verfahren (z.B. Nearest Neighbor, Average Linkage, Centroid, Median, Ward), partitionierende Verfahren (z.B. k-Means, demographische Clusteranalyse); KNN-basierte Verfahren: Kohonen-Netze (z.B. Self-Organizing Maps/SOM, Competitive Nets, LVQ), ART-Netze (Adaptive Resonance Theory)*			
Kreuztabellen, Kontingenztafelanalyse	☑	☑	☐
Darstellung statistischer Größen in Abhängigkeit anderer (nominalskalierter) Variablen in Form von Kreuztabellen und deren Analyse mithilfe statistischer Maße und Tests → *Kontingenzkoeffizienten, χ2-Test auf Signifikanz, Analyse mit loglinearen Modellen*			
On-Line Analytical Processing (OLAP)	☑	☑	☐
interaktive, dynamische Analysesysteme zur Navigation in multidimensionalen Datenräumen; umfasst zahlreiche weitere Verfahrensklassen (vgl. Kenn- und Maßzahlen, Berichte, Kreuztabellen, Zeitreihenanalyse sowie Visualisierung, Abweichungserkennung und Datenabruf) → *OLAP-Operatoren und assoziierte Verfahren*			
Dimensionsreduzierende Verfahren	☑	☑	☐
Reduzierung der Anzahl der zu betrachtenden Variablen → *Indexbildung, Hauptkomponentenanalyse, Faktorenanalyse, latente Klassenmodelle*			

(Fortsetzung auf nächster Seite)

Datenabruf, Data Access, Information Retrieval	☐	☑	☐
gezielter Abruf spezifizierter Daten oder Informationen zur Überprüfung von Hypothesen oder zur Beantwortung fachl. Fragen → *Verfahren der Datenbanktechnik und des Information Retrievals*			
Diskriminanzanalyse und Klassifikation	☑	☐	☑
Bestimmung der Klassenzugehörigkeit eines Objekts aufgrund einer Menge charakterisierender Merkmale, Ableitung von Klassifikationsmodellen → *multivariate statistische Diskriminanzanalyse, Regelinduktion (z.B. AQ15, CN2, LCNR), Klassifikationsbäume (z.B. ID3, C4.5, CHAID, CART), KNN (Perceptrons, Backpropagation-Netze), Support Vector Machines*			
Zeitreihenanalyse und -prognose	☑	☐	☑
Beschreibung der Entwicklung einer Variablen im Zeitablauf (Zeitreihe); Ableitung von Prognosemodellen → *Trendextrapolationen, exponentielle Glättungsmodelle, Box-und-Jenkins-Modelle (ARMA/ARIMA), ARCH/GARCH, KNN (Backpropagation-Netze, Jordan-, Elman- und RBF-Netze), dynamische Regressionsmodelle, Regressionsbäume (z.B. CART, M5, SRT), sequenzielle Assoziationsverfahren*			
Regressions- und Varianzanalyse	☑	☑	☑
Beschreibung und Erklärung von Beziehungen und Abhängigkeiten zwischen statistischen Größen, Ableitung von Vorhersagemodellen → *Korrelationsanalyse (Korrelationskoeffizienten), bivariate und multiple Regression, Kleinste-Quadrate-Schätzer, nichtlineare Regression, Varianz- und Kovarianzanalyse, Logit- und Probit-Modelle, Modell- und Residuendiagnostik, Regelinduktion und Regressionsbäume (z.B. CART), Inferenz auf Basis von Regressionsmodellen*			

(Fortsetzung auf nächster Seite)

Künstliche Neuronale Netze (KNN)	☑	☑	☑
flexible Funktionsapproximatoren, die für verschiedene Anwendungen der Regression, Klassifikation, Segmentierung und Prognose geeignet sind → *überwacht lernende Netze (Strukturabbildung): z.B. Perceptrons, RBF-Netze, LVQ-Netze, Probabilistische Neuronale Netze, Counter-Propagation-Netze; unüberwacht lernende Netze (Strukturentdeckung): z.B. Self-Organizing Maps (SOM/Kohonen), ART-Netze (Adaptive Resonance Theory), Deep Networks*			
Weitere Verfahren zur Unterstützung der Datenanalyse			
Text Mining	☑	☐	☐
Extraktion von Schlagworten aus unstrukturierten Daten (Dokumenten) zum Zwecke der Ordnung (Segmentierung, Klassifikation, Gruppierung) der Dokumente			
Evolutionäre Algorithmen	☑	☑	☐
allgemein anwendbare stochastische Optimierungsverfahren; einsetzbar etwa in der Cluster-, Diskriminanz- oder Zeitreihenanalyse			
Bayes-Netze	☐	☑	☑
probabilistische Varianten regelbasierter Expertensysteme (Wissensverarbeitungsmethoden); einsetzbar zur Hypothesenüberprüfung oder Inferenz			
Intelligente Agenten	☑	☑	☑
autonome Verfahren zur Ausführung spezifischer Aufgaben in verteilten Umgebungen; einsetzbar etwa zur Informationssuche und -sammlung, zur Vorverarbeitung der Analysedaten oder zur Unterstützung des Analytikers			

(Fortsetzung auf nächster Seite)

Anwendungsspezifische Verfahren oder Heuristiken	☑	☑	☑
speziell für ein Anwendungsproblem entwickelte analytische oder heuristische Analyseverfahren; z.B. zur Abweichungserkennung und -diagnose oder zur ABC-Analyse			

Erläuterung der Abkürzungen:
KNN: Künstliche Neuronale Netze

Tabelle 9: Bedeutende Klassen von Datenanalyseverfahren

A2 Maßnahmen zur Bewältigung der Analysekomplexität

Die Prozesskomplexität ergibt sich aus Struktur und Verhalten der Elemente des Prozesssystems. Der Strukturaspekt betrifft die Komponenten und Beziehungen sowie deren Kombinationsmöglichkeiten. Der Verhaltensaspekt betrifft die Inhalte der Interdependenzen, die sich aus der Funktionsweise der Komponenten ergeben, und die dynamische Veränderung der Systemzustände. Die Analysekomplexität ergibt sich als Problemkomplexität während der Konzeption und Ausführung einer Analyse durch den Informationsmangel bzw. die resultierende Überforderung durch die Vielzahl zu berücksichtigender Einflüsse (vgl. Abschnitt 3.2.2).

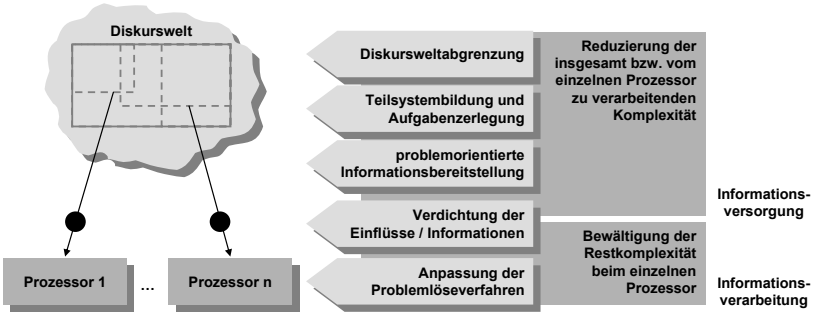


Abbildung 120: Basisstrategien zur Beherrschung der Problemkomplexität (eigene Darstellung)

Aus einer umfassenden Literaturrecherche resultieren die in Abbildung 120 dargestellten Basisstrategien zur Beherrschung der Problemkomplexität [DeFr80, 13, 39], [Luhm80, 1066-1068], [GeSc84, 8f.], [FiWo90, 13f.], [Kore90, 290], [Wers96, 15], [Mali00, 178, 239f., 255, 290, 310]. Ihre Erläuterung und Herleitung gehen über das Ziel dieser Arbeit weit hinaus; sie werden im Folgenden verwendet, um die in Abschnitt 3.2.4 diskutierten Maßnahmen zur Komplexitätshandhabung zu klassifizieren. Das Ergebnis zeigt Tabelle 10.

Maßnahmen zur Bewältigung der Analysekomplexität	betroffene Strategien zur Komplexitätsbewältigung				
	Diskurswelteingrenzung	Aufgabenzerlegung, Strukturierung	Informationsbereitstellung	Selektion, Filterung, Abstraktion	spezielle Lösungsverfahren
U: Umgehung von Analysekomplexität					
U1: Verzicht auf Datenanalysen					
U2: Rückgriff auf vorhandene Informationen bzw. Analyseergebnisse					
→ Datenanalyse i.w.S., daher nur komplexitätsreduzierende Wirkung					
U3: Vollautomatisierung von Datenanalysen					
→ derzeit nicht realistisch					
R: Reduzierung von Analysekomplexität					
R1: Begrenzung des Analyseraums bzw. Verkürzung des Analysepfads	☑	☐	☐	☐	☐
R1.1: strenge Zielorientierung durch klare Problemspezifikation	☑	☐	☐	☑	☑
R1.2: Prozessabspaltung durch Zieldifferenzierung	☑	☑	☐	☐	☐
R1.3: methodisch-systematische Datenauswahl und -selektion	☐	☐	☐	☑	☑
R2: Prozessverkürzung	☑	☑	☐	☐	☐
R2.1: Abspaltung analyseunabhängiger Aktivitäten	☑	☑	☐	☐	☐
R2.2: Zugriff auf spezielle Analysedatenbanken	☑	☐	☐	☑	☐
R2.3: Vermeidung unnötiger Aktivitäten und Iterationen	☐	☑	☑	☐	☑
R2.4: Wiederverwendung von Zwischenergebnissen	☐	☑	☑	☐	☐
R2.5: Unterstützung durch integriertes Analysewerkzeug	☐	☐	☐	☐	☑

(Fortsetzung auf nächster Seite)

R3: Reduzierung von Planungsaufwand	☐	☑	☑	☐	☐
R3.1: Wiederverwendung von Prozessplänen, Wiederholung von Prozessabläufen	☐	☐	☑	☐	☐
R3.2: Vereinheitlichung und Standardisierung von Prozessen	☐	☑	☑	☑	☐
R3.3: Bereitstellung vereinheitlichter oder abstrakter Prozesspläne (Schablonen)	☐	☑	☑	☑	☑
R4: (Teil-) Automatisierung der Prozessplanung bzw. -ausführung	☐	☑	☐	☐	☑
B: Bewältigung von Analysekomplexität					
B1: Ordnung des Prozesssystems	☐	☑	☐	☐	☐
B1.1: Strukturierung und Modularisierung	☐	☑	☐	☐	☐
B1.2: Ebenen- und Sichtenbildung, Hierarchisierung	☐	☑	☐	☐	☐
B1.3: Taxonomien von Aufgaben und Verfahren	☐	☑	☑	☐	☐
B2: methodische und wissensbasierte Unterstützung des Analytikers	☐	☐	☐	☐	☑
B2.1: Problemspezifikation	☑	☑	☐	☐	☑
B2.2: Vorgabe von Prozessschemata	☐	☑	☑	☐	☑
B2.3: Flexibilisierung des Vorgehens (evolutionäre, inkrementelle Lösungsansätze)	☐	☐	☐	☐	☑
B2.4: Bereitstellung von Erfahrungs- und Domänenwissen	☐	☐	☑	☐	☑
B2.5: Bereitstellung von Handlungsempfehlungen und Best Practices	☐	☐	☑	☐	☐
B2.6: Zugriff auf kontextabhängiges Wissen	☐	☐	☑	☐	☐

(Fortsetzung auf nächster Seite)

B3: werkzeugbasierte Unterstützung des Analytikers	☐	☐	☐	☐	☑
B3.1: Werkzeugunterstützung der Problemlösemethodik	☐	☐	☐	☐	☑
B3.2: Automatisierung von Gestaltung-, Lenkungs- und Managementaufgaben	☐	☑	☐	☐	☑
B3.3: Assistenz- und Beratungsfunktion	☐	☐	☑	☐	☑

Tabelle 10: Gliederung von Maßnahmen zur Bewältigung von Analysekomplexität

A3 Phasen und Aufgaben des Vorgehensmodells zur Datenanalyse

Die folgende Aufstellung zeigt die hierarchische Gliederung der Phasen und Aufgaben des Vorgehensmodells aus Abschnitt 3.3.3. Zur einfacheren Referenz ist ihrer Nummerierung ein V (für Vorgehensmodell) bzw. A (für die auszuführenden Prozessaufgaben) vorangestellt.

V1: **Planung des Analyseprojekts**

- V1.1: Identifikation des Sachproblems
- V1.2: Domänenanalyse
- V1.3: Spezifikation von Analyseproblemen und Analyseketten
- V1.4: Untersuchungsdesign
- V1.5: Projektplanung

V2: **Durchführung der Analyse** (A)

mehrere Iterationen gemäß Datenanalyse-Spiralmodell:

- A1: Planung der Analyse
  - A1.1: Spezifikation und Präzisierung des Analyseproblems
  - A1.2: Prozessplanung
- A2: Datenvorbereitung
  - A2.1: Datenselektion
  - A2.2: Datenexploration
  - A2.3: Datenmodifikation
    - A2.3.1: Anreicherung
    - A2.3.2: Bereinigung
    - A2.3.3: Konsolidierung
    - A2.3.4: Transformation i.e.S. (Codierung)

A3: Datenanalyse

A3.1: Modellspezifikation

A3.2: Modellentwicklung

A3.3: Modellkalibrierung

A3.4: Modellevaluierung

A4: Ergebnisaufbereitung

A4.1: Ergebnisbeurteilung

A4.2: Vereinfachung

A4.3: Transformation

A4.4: Interpretation

A4.5: Dokumentation

V3: **Anwendung des Wissens**

V3.1: Identifikation von Einsatzpotenzialen

V3.2: Maßnahmenplanung

V3.3: Maßnahmendurchführung

V3.4: Abschlussbericht

V4: **Evaluierung des Analyseprojekts**

V4.1: Beurteilung der Analyseergebnisse

V4.2: Beurteilung des Prozessablaufs

V4.3: Evaluation der Handlungsmaßnahmen

V4.4: Nutzen-Kosten-Analyse (Bewertung des ökonomischen Erfolgs)

V4.5: Erfahrungssicherung

A4 Attributschemata zum Modellierungsansatz

Dieser Anhang listet Attributschemata für alle Metaobjekttypen des Modellierungsansatzes, geordnet nach den Architekturebenen, aus Kapitel 4 in tabellarischer Form auf. Zusätzlich werden in Abschnitt A4.5 die Schemata für die Ontologie (Abschnitt 4.7.1) dargestellt.

Der Wertebereich (Domäne) spezifiziert den Datentyp des Attributs. Dieser kann auch ein strukturierter Datentyp oder eine Referenz auf einen anderen Metaobjekttyp sein. Beziehungsmerkmale, d.h. Attribute, die Referenzen zu anderen Metaobjekttypen gemäß Metamodell beschreiben [Voit10, 124f.], sind *kursiv* gesetzt. Für strukturierte Datentypen ist in Abschnitt A4.6 jeweils eine Typvereinbarung definiert. Die Kardinalitäten (Kard.) sind in (min, max)-Notation angegeben und beziehen sich im Falle von Collections auf die Anzahl der enthaltenen Elemente. Die Beschreibung erläutert die Semantik des Attributs und kann zudem eine Menge gültiger Ausprägungen aus dem Wertebereich angeben. Die verwendeten elementaren Datentypen orientieren sich am Object Data Standard der ODMG [Catt00].

Alle Metaobjekttypen sind Spezialisierungen des abstrakten Super-Objekttyps aus Tabelle 11 und erben entsprechend dessen Attribute. Sofern diese für die Definition eines Metaobjekttyps von besonderer Relevanz sind, können sie dort redundant angeführt werden; in diesem Fall ist ihnen jeweils ein *V* (für Vererbung) nachgestellt.

Metaobjekttyp	**Objekttyp** *generischer Typ, von dem alle anderen Metaobjekttypen abgeleitet sind*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name	string	1,1	Name/Identifikator des Objekttyps
Beschreibung	string	0,1	Beschreibung
Kommentare	set<Kommentar>	0,*	Menge von Textkommentaren und Anmerkungen
Schlüsselworte	set<string>	0,*	Menge von Schlüsselworten (Tags), die dem Objekt zu Suchzwecken zugeordnet sind
Kontext	set<Deskriptor>	0,*	Menge von Kontextfaktoren, die für das Objekt gelten
Kontextregeln	set<string>	0,*	Menge von Kontextregeln, die für das Objekt gelten
Links	set<Link>	0,*	Menge von Verknüpfungen zu Ressourcen, die mittels URL erreichbar sind

Tabelle 11: Typvereinbarung (Attributschema) des abstrakten Metaobjekttyps *Objekttyp*

A4.1 Attributschemata der Anwendungsebene

Metaobjekttyp	**Problemaspekt**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung des Problemaspekts (seiner Semantik)
Beschreibung (V)	string	0,1	Beschreibung des Problemaspekts; Erläuterung des problematischen Ist- bzw. des anzustrebenden Soll-Zustands
Typ	string	0,1	Typ bzw. Zustandsversion des Problemaspekts; {situationsbezogen | lösungsbezogen} = {Ist | Soll}
Vateraspekt	Verknüpfung	0,1	Verknüpfung zum übergeordneten Problemaspekt, aus dem dieser Problemaspekt hervorgeht (Verknüpfungstyp: eingehende Teil_von-Beziehung)
Kindaspekte	set<Verknüpfung>	0,*	Menge von Verknüpfungen zu untergeordneten Problemaspekten, die als Differenzierungsprodukte aus diesem Problemaspekt hervorgehen (Verknüpfungstyp: ausgehende Differenzierung)
Vorgänger	set<Verknüpfung>	0,*	Menge von Verknüpfungen zu Problemaspekten, die in der Problemkarte diesem Aspekt vorausgehen (Verknüpfungstyp: eingehende Sequenz)

(Fortsetzung auf nächster Seite)

Nachfolger	set<Verknüpfung>	0,*	Menge von Verknüpfungen zu Problemaspekten, die in der Problemkarte diesem Aspekt nachfolgen (Verknüpfungstyp: ausgehende Sequenz)
Maßnahmen	set<Maßnahme>	0,*	Menge aller Maßnahmen, die zur Lösung des Problemaspekts geeignet erscheinen
Problemdomäne	Domänenobjekt → Ontologie	1,*	Menge von konzeptuellen Domänenobjekten (Objekte oder Beziehungen), die zum Verständnis des Problems erforderlich sind
Inhalt	Domänenobjektmerkmal → Ontologie	1,1	Zustandsinhalt als Merkmal eines Domänenobjekts, das untersucht bzw. verändert werden soll (`Problemobjekt.Problemmerkmal`; vgl. Untersuchungsobjekt und -ziel)
Ausmaß	literal	0,1	Zustandsausmaß als beobachtete bzw. angestrebte Ausprägung des Zustandsinhalts. Wertebereich ist abhängig von `Inhalt.Datentyp`.
Zeitbezug	list<date>	0,2	Zeitpunkt (Einzelwert) oder Zeitraum (Doppelwert), zu bzw. in dem das Zustandsausmaß beobachtet wurde bzw. zu erreichen ist
Wertbeitrag	Begriff → Ontologie	0,1	betriebswirtschaftliches Zielkriterium, das vom Zustandsinhalt beeinflusst wird

(Fortsetzung auf nächster Seite)

Kulisse	string	0,1	Beschreibung von Vergleichsgrößen zur besseren Einordnung des betrachteten Zustands
Modifikator	Modifikator	1,1	Kennzeichnung des bezüglich des Zustandsausmaßes beobachteten bzw. angestrebten Veränderungsereignisses {+ | - | = | o | *}
Kennzeichnung (Anwendung)	list<Anwendung>	1,*	einheitliche Kennzeichnung des Problemaspekts; Rückgriff auf Merkmale `Inhalt`, `Modifikator` und `Typ` (Zustandsversion)

Tabelle 12: Typvereinbarung (Attributschema) des Metaobjekttyps *Problemaspekt*

Metaobjekttyp	**Maßnahme**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung der Maßnahme (ihrer Semantik)
Beschreibung (V)	string	0,1	Beschreibung/Erläuterung der Maßnahme
Typ	string	1,1	Typ der Maßnahme; {Informationsmaßnahme | Handlungs-maßnahme}
Problemaspekt	Problemaspekt	1,1	Problemaspekt, dem die Maßnahme zugeordnet ist
Zeitrestriktion	list<date>	0,2	Zeitpunkt (Einzelwert) oder Zeitraum (Doppelwert), bis zu bzw. in dem die Maßnahme vollendet sein soll

(Fortsetzung auf nächster Seite)

Budget-restriktion	float	0,1	Wert des für die Maßnahme bereitstehenden finanziellen Budgets
Organisation	Begriff → Ontologie	0,1	Organisationseinheit, die zur Maßnahmendurchführung bestimmt ist
Ansprechpartner	Person	0,1	Verantwortliche Person für die Maßnahme
Projekt	Begriff → Ontologie	0,1	Projekt, das die Durchführung der Maßnahme realisiert
Unternehmen	Begriff → Ontologie	0,1	Unternehmen, für das die Maßnahme konzipiert wurde
Branche	Begriff → Ontologie	0,1	Branche, für die die Maßnahme konzipiert wurde
Betriebstyp	Begriff → Ontologie	0,1	Typ des Betriebs, für den die Maßnahme konzipiert wurde
Bewertung	set<Bewertungs-ergebnis>	0,*	Menge von Bewertungen bezüglich definierter Erfolgskriterien bei Realisierung der Maßnahme
Erfolg	boolean	0,1	Gesamturteil des Maßnahmenerfolgs, abgeleitet aus Bewertung bzw. Zielzustand des Vateraspekts; {0 | 1} = {nicht erfolgreich | erfolgreich}

Tabelle 13: Typvereinbarung (Attributschema) des Metaobjekttyps *Maßnahme*

Metaobjekttyp	**Verknüpfung**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	0,1	Bezeichnung der Verknüpfung (ihrer Semantik)
Beschreibung (V)	string	0,1	Motivation/Begründung für die Verknüpfung
Typ	string	0,1	Typ der Verknüpfung; {Sequenz | Teil-von-Beziehung | Mittel-Zweck-Beziehung}
Startobjekt	Problemaspekt	1,1	Problemaspekt, von dem die Verknüpfung ausgeht
Zielobjekt	Problemaspekt	1,1	Problemaspekt, in den die Verknüpfung mündet

Tabelle 14: Typvereinbarung (Attributschema) des Metaobjekttyps *Verknüpfung*

A4.2 Attributschemata der Analyseebene

Metaobjekttyp	**Analyseziel**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung des Analyseziels (seiner Semantik)
Beschreibung (V)	string	0,1	Beschreibung/Erläuterung des Analyseziels
Informations-maßnahme	Maßnahme	0,1	Informationsmaßnahme der Anwendungsebene, der das Analyseziel zugeordnet ist

(Fortsetzung auf nächster Seite)

Domänenobjekt	Domänenobjekt → Ontologie	1,1	konzeptuelles Domänenobjekt als Untersuchungsobjekt der Analyse
Merkmal	Domänenobjekt-merkmal → Ontologie	1,1	primäres konzeptuelles Merkmal des Domänenobjekts, das untersucht werden soll (Untersuchungsziel)
Ausrichtung	string	1,1	Analyseausrichtung {explorativ | konfirmatorisch | schließend}
Analysefrage	Analysefrage	1,1	Beschreibung des Inhalts des Analyseergebnisses in natürlich-sprachiger und strukturierter Form
Informations-bedarfsprofil	set<Deskriptor>	0,*	Menge von formalen Anforderungen an das Analyseergebnis
Abstammung	set<Ab-stammung>	0,*	Menge von Analysezielen, aus denen das betrachtete Analyseziel anhand eines definierten Kriteriums abgeleitet wurde
Vorgänger	set<Verkettung>	0,*	Menge von Verkettungen zu Analyse-zielen, die in der Analysekette dem betrachteten Analyseziel vorausgehen (eingehende Verkettung)
Nachfolger	set<Verkettung>	0,*	Menge von Verkettungen zu Analyse-zielen, die in der Analysekette dem betrachteten Analyseziel nachfolgen (ausgehende Verkettung)

Tabelle 15: Typvereinbarung (Attributschema) des Metaobjekttyps A*nalyseziel*

Metaobjekttyp	**Analyseproblem** *ist_ein Analyseziel*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
	+ alle Attribute des Analyseziels		
Perspektive	Perspektive	1,1	Beschreibung der Sichtweise auf das Untersuchungsobjekt, welche die Analysedaten repräsentieren sollen
Analyseobjekt	set<Informations-objekt>	1,*	Menge von Informationsobjekten, welche in ihrer Gesamtheit die Basis für die Analysedaten bilden

Tabelle 16: Typvereinbarung (Attributschema) des Metaobjekttyps A*nalyseproblem*

Metaobjekttyp	**Verkettung**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Startobjekt	*Analyseziel*	1,1	Analyseziel oder -problem, von dem die Verkettung ausgeht
Zielobjekt	*Analyseziel*	1,1	Analyseziel oder -problem, in das die Verkettung mündet
Typ	string	1,1	Typ der Verkettung; {Route | Pfad}; (Standardwert = Pfad)
Bedingungen	set<string>	0,*	Beschreibung der Bedingungen, unter denen der Verkettungsschritt ausgeführt werden soll bzw. darf

Tabelle 17: Typvereinbarung (Attributschema) des Metaobjekttyps *Verkettung*

A4.3 Attributschemata der Prozessebene

Die Reihenfolge der dargestellten Metaobjekttypen orientiert sich zunächst an der Taxonomie der Bibliothekssicht aus Abschnitt 4.5.4 und ergänzt diese davor und danach um die verbleibenden Objekttypen, wobei jeweils Referenzbeziehungen berücksichtigt werden.

Metaobjekttyp	**Funktion** → Ontologie		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung der Funktion (ihrer Semantik), formuliert in Bezug auf Eingabedatentyp, Transformation und ggf. Ausgabedatentyp
Beschreibung (V)	string	0,1	Beschreibung/Erläuterung der Funktion
Eingabe- datentyp	set<Daten- objekttyp>	0,*	Menge der Eingabedatentypen
Ausgabe- datentyp	set<Daten- objekttyp>	0,*	Menge der Ausgabedatentypen
Transformation	string	1,1	Bezeichnung der Verrichtung (Transformation der Eingabedatentypen in Ausgabedatentypen)

Tabelle 18: Typvereinbarung (Attributschema) des Metaobjekttyps *Funktion*

Metaobjekttyp	**Prozessbaustein**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung des Bausteins (seiner Semantik)
Beschreibung (V)	string	0,1	Beschreibung/Erläuterung des Bausteins
Kontext (V)	set<Deskriptor>	0,*	Menge von Kontextfaktoren zur Beschreibung der Einordnung eines Bausteins in die Analysearchitektur sowie zur kontextabhängigen Selektion
Kontextregeln (V)	set<string>	0,*	Menge von Regeln oder Hinweisen zur kontextabhängigen Anpassung des Bausteins an die aktuelle Situation
Version	unsigned short	1,1	Version des Prozessbausteins
Datum	timestamp	1,1	Datum/Zeitstempel der aktuellen Version
Typ	string	1,1	Typ des Prozessbausteins; {Aktivität | Aufgabe | Fragment | Schablone}
Funktion	Funktion	1,1	Funktion, welche der Baustein realisiert (funktionale Beschreibung des Aufgabensachziels des Bausteins)
Eingabeflüsse	Set<Fluss-beziehung>	0,*	Menge der Eingabeflüsse
Ausgabeflüsse	set<Fluss-beziehung>	0,*	Menge der Ausgabeflüsse

(Fortsetzung auf nächster Seite)

Genese	set<Ab-stammung>	0,*	Menge von Bausteinen, aus denen der betrachtete Baustein anhand eines definierten Kriteriums abgeleitet wurde
Einsatzzähler	unsigned short	1,1	Anzahl der Einsätze des Bausteins in Prozessschemata seit seiner Speicherung in der Bibliothek
Einsatznote	unsigned short	0,1	Gesamtbewertung des Bausteins durch den Anwender beim Einsatz in Prozessschemata Schulnoten: {1..6}
Einsatz-bewertung	set<Bewertungs-ergebnis>	0,*	Einzelbewertung des Bausteins nach den Kriterien der Revision (Qualitäts-, Zeit- und Kostenbewertung)
Einsatz-kommentar	set<Kommentar>	0,*	Kommentare und Anmerkungen des Anwenders, die sich speziell auf die Wiederverwendung des Bausteins beziehen
Ursprung	string	0,1	Kennzeichnung des Entstehungstyps des Bausteins; manuell durch den Analytiker oder automatisiert durch Analysetechniken (z.B. Process Mining); {Nutzer | System}
Änderungen	list<Änderungs-operation>	0,*	Liste von Änderungen, die am Baustein gegenüber der vorhergehenden Version vorgenommen wurden

Tabelle 19: Typvereinbarung (Attributschema) des Metaobjekttyps *Prozessbaustein*

Das Attribut `Ursprung` eines zur Wiederverwendung zu speichernden Artefakts wird gepflegt, um spezifisch nach manuell oder automatisch erzeugten Artefakten suchen zu können.

Metaobjekttyp	**Prozessmodul** *ist_ein Prozessbaustein*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
	+ alle Attribute des Prozessbausteins		
Typ (V+)	string	1,1	Typ des Prozessmoduls; {Fragment | Schablone}

Tabelle 20: Typvereinbarung (Attributschema) des Metaobjekttyps *Prozessmodul*

Das Attributschema des Prozessmoduls entspricht jenem des Prozessbausteins. Die Domäne des Attributs `Prozessmodul.Typ` ist jedoch gegenüber dem Prozessbaustein eingeschränkt derart, dass im Modul nur die Werte {Fragment | Schablone} zulässig sind.

Metaobjekttyp	**Fragment** *ist_ein Prozessmodul*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
	+ alle Attribute des Prozessmoduls		
Aktivitäten	set<Aktivität>	2,*	Menge von Aktivitäten, die das Fragment als Einheit repräsentiert (d.h., die als Kind-Komponenten in ihm enthalten sind)

Tabelle 21: Typvereinbarung (Attributschema) des Metaobjekttyps *Fragment*

Metaobjekttyp	**Schablone** *ist_ein Prozessmodul*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
	+ alle Attribute des Prozessmoduls		
Aufgaben	set<Aufgabe>	2,*	Menge von Aufgaben oder Aktivitäten, die die Schablone als Einheit repräsentiert (d.h., die als Kind-Komponenten in ihr enthalten sind)
Reinheit	boolean	1,1	Kennzeichen, ob nur Aufgaben enthalten sind (1), oder ob die Reinheit der Schablone durch einzelne oder mehrere Aktivitäten getrübt wird (0)

Tabelle 22: Typvereinbarung (Attributschema) des Metaobjekttyps *Schablone*

Metaobjekttyp	**Aufgabe** *ist_ein Prozessbaustein*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
	+ alle Attribute des Prozessbausteins		
Anforderungen	set<Deskriptor>	0,*	Menge von Anforderungen als Formalziele der Aufgabendurchführung
Verwendung	string	1,1	Typ der Aufgabe gemäß ihrer Verwendung {Analyseaufgabe | Transformationsaufgabe}

Tabelle 23: Typvereinbarung (Attributschema) des Metaobjekttyps *Aufgabe*

Aufgaben des Verwendungstyps Transformationsaufgabe verfügen über dasselbe Attributschema wie die Aufgabe, daher wird im Folgenden nur die Spezifikation der Analyseaufgabe gezeigt.

Metaobjekttyp	**Analyseaufgabe** *ist_ein Aufgabe*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
	+ alle Attribute der Aufgabe		
Analyseproblem	Analyseproblem	0,1	Analyseproblem, das durch die Aufgabe gelöst werden soll
Bewertungs-kriterien	set<Bewertungs-kriterium>	0,*	Kriterien zur Bewertung von Analyseergebnissen
Bewertungs-funktionen	set<string>	0,*	Funktion zur monetären Bewertung von Nutzen oder Kosten von Analyse-ergebnissen oder Modellen
Präferenz-relationen	set<string>	0,*	Kriterien oder Rangfolge von Kriterien, nach denen Ergebnisse beurteilt oder ausgewählt werden sollen

Tabelle 24: Typvereinbarung (Attributschema) des Metaobjekttyps *Analyseaufgabe*

Metaobjekttyp	**Aktivität** *ist_ein Aufgabe*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
	+ alle Attribute der Aufgabe		
Operator	Operator	1,1	Zuordnung eines Operators
Parameterwerte	set<Parameter>	0,*	Menge von Parametereinstellungen zur Instanziierung des Operators

(Fortsetzung auf nächster Seite)

Eingabedaten-zuordnung	set<Rollen-zuordnung>	0,*	Menge von Zuordnungsbeziehungen zwischen einem Element von `Eingabeflüsse` auf ein Element von `Operator.Eingabedaten`
Ausgabedaten-zuordnung	set<Rollen-zuordnung>	0,*	Menge von Zuordnungsbeziehungen zwischen einem Element von `Ausgabeflüsse` auf ein Element von `Operator.Ausgabedaten`
Startbedingung	string	0,1	Bedingung in den Variablen der eingehenden Flüsse, die zur Ausführung der Aktivität erfüllt sein muss; Standardverhalten: konjunktive Verknüpfung aller Inputs
Endbedingung	string	0,1	Bedingung in den Variablen der ein- und ausgehenden Flüsse, die beschreibt, welche der ausgehenden Flüsse wann bedient werden; Standardverhalten: konjunktive Verknüpfung aller Outputs
Block-markierungen	set<string>	0,*	Aufzählung von Rollen der Aktivität im Steuerfluss zur Markierung von Blockstrukturen
Rolle	Rolle	0,1	Rolle der Ressourcenebene zur Bekanntgabe von Anforderungen an geeignete Analytiker
Vorgänge	set<Vorgang>	0,*	Menge der Vorgänge, die von dieser Aktivität ausgeführt wurden

Tabelle 25: Typvereinbarung (Attributschema) des Metaobjekttyps *Aktivität*

Metaobjekttyp	**Workflow**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung des Workflows (seiner Semantik)
Beschreibung (V)	string	0,1	Beschreibung/Erläuterung des Workflows
Version	unsigned short	1,1	Version des Workflows
Datum	timestamp	1,1	Datum/Zeitstempel der aktuellen Version
Analyseproblem	Analyseproblem	1,1	Analyseproblem, das durch den Workflow gelöst werden soll (= `Analyseaufgabe.Analyseproblem`)
Kontext (V)	set<Deskriptor>	0,*	Menge von Kontextfaktoren zur Beschreibung der Einordnung des Workflows in die Analysearchitektur sowie zur kontextabhängigen Selektion
Einsatzzähler	unsigned short	1,1	Anzahl der Einsätze des Workflows seit seiner Speicherung in der Bibliothek
Einsatznote	unsigned short	0,1	Gesamtbewertung des Workflows durch den Anwender beim Einsatz Schulnoten: {1..6}
Einsatz-bewertung	set<Bewertungs-ergebnis>	0,*	Einzelbewertung des Workflows nach den Kriterien der Revision (Qualitäts-, Zeit- und Kostenbewertung)

(Fortsetzung auf nächster Seite)

Einsatz-kommentar	set<Kommentar>	0,*	Kommentare und Anmerkungen des Anwenders, die sich speziell auf die Wiederverwendung des Workflows beziehen
Prozessinstanzen	set<Prozess-instanz>	0,*	Menge der Instanzen, die von diesem Workflow abgeleitet wurden
Änderungen	list<Änderungs-operation>	0,*	Liste von Änderungen, die am Workflow gegenüber der vorhergehenden Version vorgenommen wurden
Aktivitäten	set<Aktivität>	1,*	Menge von Aktivitäten, die der Workflow in der Bibliothek als Einheit repräsentiert (d.h., die als Kind-Komponenten in ihm enthalten sind)

Tabelle 26: Typvereinbarung (Attributschema) des Metaobjekttyps *Workflow*

Metaobjekttyp	**Flussbeziehung**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Name/Identifikator der Flussbeziehung
Beschreibung (V)	string	0,1	Beschreibung oder Begründung der Flussbeziehung, z.B. zur Erläuterung von Maßnahmen zur Fehlerbeseitigung (Typ = Fehler)
Typ	string	1,1	Typ (Semantik) der Flussbeziehung; {Datenabhängigkeit | Synchronisation | Sprung | Fehler}
Datentyp	Datenobjekttyp → Ontologie	1,1	Datentyp der Flussbeziehung (für Synchronisation, Sprung und Fehler stets boolean)

(Fortsetzung auf nächster Seite)

Startaufgabe	Aufgabe	1,1	Aufgabe, von welcher die Flussbeziehung ausgeht
Zielaufgabe	Aufgabe	1,1	Aufgabe, in welche die Flussbeziehung mündet
Ausgangs-bedingung	set<string>	0,*	Menge konjunktiv verknüpfter Ausdrücke in den Variablen der Aufgabe, die den von der Flussbeziehung übertragenen Wert bestimmen

Tabelle 27: Typvereinbarung (Attributschema) des Metaobjekttyps *Flussbeziehung*

Metaobjekttyp	**Vorgang**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Name des Vorgangs, gebildet aus `Aktivität.Name + Zähler`
Beschreibung (V)	string	0,1	Erläuterung/Beschreibung des Vorgangs
Aktivität	Aktivität	1,1	Aktivität, die der Vorgang realisiert
Startzeit	timestamp	1,1	Zeitstempel des Starts des Vorgangs
Endzeit	timestamp	0,1	Zeitstempel der Beendigung des Vorgangs, unabhängig von Art/Grund der Beendigung
Zustand	Instanzzustand	1,1	aktueller Zustand des Vorgangs
Ereignisprotokoll	list<Ereignis>	1,*	geordnete Liste der Veränderungen von `Zustand` mit Zeitstempel und Ereignistyp
Fehler	list<string>	0,*	Fehlerprotokoll (aus Rückgabeparametern des Operators)

(Fortsetzung auf nächster Seite)

Zähler	unsigned short	1,1	Anzahl von Vorgangswiederholungen
Server	Software-Installation (Server)	1,1	zur Realisierung eingesetzte Instanz eines Software-Elements
Eingabedaten	set<Datenfluss>	0,*	Menge der eingehenden Datenflüsse
Ausgabedaten	set<Datenfluss>	0,*	Menge der ausgehenden Datenflüsse

Tabelle 28: Typvereinbarung (Attributschema) des Metaobjekttyps *Vorgang*

Metaobjekttyp	**Datenfluss**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Flussbeziehung	Datenabhängigkeit	1,1	Flussbeziehung vom Typ Datenabhängigkeit, die der Datenfluss realisiert
Informationsobjekt	Informationsobjekt	1,1	Informationsobjekt, das die konkret übertragenen/verarbeiteten Daten repräsentiert
Startvorgang	Vorgang	1,1	Vorgang, von dem der Datenfluss ausgeht
Zielvorgang	Vorgang	1,1	Vorgang, in den der Datenfluss mündet

Tabelle 29: Typvereinbarung (Attributschema) des Metaobjekttyps *Datenfluss*

Metaobjekttyp	**Prozessinstanz** *Ausprägung eines Workflow-Ablaufs, repräsentiert einen Analysefall*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Beschreibung (V)	string	0,1	Erläuterung/Beschreibung der Instanz durch den Analytiker
Benutzer	Person	1,1	Benutzer/Analytiker, der die Prozessausführung initiiert hat
Startzeit	timestamp	1,1	Zeitstempel des Starts der Prozessausführung
Endzeit	timestamp	0,1	Zeitstempel der Beendigung der Prozessausführung, unabhängig von Art/Grund der Beendigung
Zustand	Instanzzustand	1,1	aktueller Zustand der Prozessinstanz
Änderungen	list<Änderungs-operation>	0,*	Liste von Änderungen, die an der Instanz gegenüber dem zugrundeliegenden Workflow vorgenommen wurden
Ergebnis-bewertung	set<Bewertungs-ergebnis>	0,*	Menge von Bewertungen bezüglich definierter Erfolgskriterien im Hinblick auf die erzielten Analyseergebnisse
Prozess-bewertung	set<Bewertungs-ergebnis>	0,*	Menge von Bewertungen bezüglich definierter Erfolgskriterien im Hinblick auf den Prozessablauf
Workflow	Workflow	1,1	Workflow als Prozesstyp, von dem die Instanz abgeleitet ist
Vorgänge	set<Vorgang>	1,*	Menge von Vorgängen, welche die Prozessinstanz konstituieren

Tabelle 30: Typvereinbarung (Attributschema) des Metaobjekttyps *Prozessinstanz*

A4.4 Attributschemata der Ressourcenebene

Die Reihenfolge der Darstellung auf Ressourcenebene orientiert sich an der Gliederung in Daten-, Aufgabenträger- und Instanzensicht.

Metaobjekttyp	**Datenobjekttyp**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung/Identifikator des Datenobjekttyps
Wertebereich	Begriff → Ontologie	1,1	Benennung eines elementaren, bereits existierenden Datenobjekttyps oder eines Konstruktors zur Definition eines neuen Typs
Restriktionen	set<string>	0,*	Menge von Ausdrücken zur Einschränkung von `Wertebereich`

Tabelle 31: Typvereinbarung (Attributschema) des Metaobjekttyps *Datenobjekttyp*

Metaobjekttyp	**Informationsobjekttyp** *ist_ein Datenobjekttyp*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
	+ alle Attribute des Datenobjekttyps		
Domänenobjekt	Domänenobjekt → Ontologie	0,1	konzeptuelles Domänenobjekt, das durch den Informationsobjekttyp beschrieben wird
Merkmal	Domänenobjekt-merkmal → Ontologie	0,1	konzeptuelles Merkmal des Domänenobjekts, das durch den Informationsobjekttyp beschrieben wird

(Fortsetzung auf nächster Seite)

Beschreibung (V)	string	0,1	inhaltliche Beschreibung des Informationsobjekttyps (seiner Semantik)
Persistenz	boolean	1,1	Kennzeichen, ob das Informationsobjekt persistent ist oder nur als temporäre Speicherrepräsentation existiert (transient) {1: persistent | 0: transient}
Informations-objekte	set<Informations-objekt>	0,*	Menge der aktuell verfügbaren Instanzen des Informationsobjekttyps

Tabelle 32: Typvereinbarung (Attributschema) des Metaobjekttyps *Informationsobjekt-typ*

Metaobjekttyp	**Operator**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung/Identifikator des Operators
Beschreibung (V)	string	0,1	Beschreibung des Operators
Funktion	Funktion	1,1	Funktion, welche der Operator realisiert (funktionale Beschreibung des intendierten Zustandsübergangs der Eingabedaten)
Eingabedaten	set<Datenobjekt-typ>	0,*	Menge der Eingabedatentypen
Ausgabedaten	set<Datenobjekt-typ>	0,*	Menge der Ausgabedatentypen

(Fortsetzung auf nächster Seite)

Schaltbedingung	string	0,1	Bedingung in den Variablen der `Eingabedaten`, die zur Ausführung des Verfahrens erfüllt sein muss; Standardverhalten: konjunktive Verknüpfung aller Inputs
Schaltwirkung	string	0,1	Bedingung in den Variablen der `Eingabedaten` und `Ausgabedaten`, die beschreibt, welche der Outputs wann produziert werden; Standardwert: konjunktive Verknüpfung aller Outputs
Parameter	set<Parameter>	0,*	Menge von Parametern zur Instanziierung des Operators
Verfahrensprofil	set<Deskriptor>	0,*	Menge von Eigenschaften des Verfahrens
Leistungs-faktoren	set<Bewertungs-faktor>	0,*	Menge von Bewertungsfaktoren, die zur Beurteilung der Verfahrensdurchführung beitragen können
Evaluations-ansatz	list<Funktions-empfehlung>	0,*	Liste von Empfehlungen für geeignete Evaluationsfunktionen mit Anmerkungen, geordnet nach `Funktionsempfehlung.Rang`
Zusicherungen	set<Deskriptor>	0,*	Menge von Zusicherungen, die den Ausgabedaten attributiert werden können
Bewertungs-kriterien	set<Bewertungs-kriterium>	0,*	verfügbare bzw. zulässige Kriterien zur Bewertung von Analyseergebnissen
Ausrichtung	string	1,3	Menge unterstützter Analyseausrichtungen {explorativ | konfirmatorisch | schließend}

(Fortsetzung auf nächster Seite)

Aufrufnachricht	string	0,1	parametrisierbare Nachricht zur Auslösung des Operators, wie sie an den Server zu versenden ist
Software-Produkt	Software-Produkt	1,1	Software-Produkt, das den Operator implementiert

Tabelle 33: Typvereinbarung (Attributschema) des Metaobjekttyps *Operator*

Metaobjekttyp	**Software-Produkt (Service)**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung/Identifikator des Software-Produkts
Beschreibung (V)	string	0,1	Beschreibung des Software-Produkts
Operatoren	set<Operator>	1,*	Menge der Operatoren, die das Software-Produkt implementiert
Version	string	0,1	Versionsnummer des Software-Produkts
Hersteller	string	0,1	Name des Herstellers
Werkzeug	Software-Produkt	0,1	Werkzeug, in dem der Service bereitgestellt wird; referenziert wiederum Software-Produkt
Installationen	set<Software-Installation>	0,*	konkret verfügbare Instanzen des Software-Produkts in Form von Software-Installationen
Rollen	set<Rolle>	0,*	Menge von Rollenbeschreibungen, die diesem Software-Produkt zugeordnet sind

Tabelle 34: Typvereinbarung (Attributschema) des Metaobjekttyps *Software-Produkt (Service)*

Metaobjekttyp	**Rolle**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung/Identifikator der Rolle
Beschreibung (V)	string	0,1	Beschreibung der Rolle
Qualifikations-profil	set<Deskriptor>	0,*	Menge konkreter Anforderungen, die ein Träger der Rolle erfüllen muss
Kostensatz	Bewertungsfaktor	0,1	Angabe eines Kostensatzes (z.B. je Stunde) für einen typischen Träger dieser Rolle
Personen	set<Person>	0,*	Menge der Personen, die aufgrund ihrer Qualifikationen als Träger der Rolle in Frage kommen
Software-Produkte	set<Software-Produkt>	0,*	Menge der Software-Produkte, die ein Träger der Rolle als Bestandteil des Qualifikationsprofils beherrschen muss

Tabelle 35: Typvereinbarung (Attributschema) des Metaobjekttyps *Rolle*

Metaobjekttyp	**Informationsobjekt**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Name/Identifikator des Informationsobjekts
Informations-objekttyp	Informations-objekttyp	1,1	Informationsobjekttyp als Daten-objekttyp, vom dem das Infor-mationsobjekt eine Instanz repräsentiert

(Fortsetzung auf nächster Seite)

Wert	literal	0,1	Menge von Ausdrücken zur Beschreibung des aktuellen Werts bzw. Zustands des Informationsobjekts
Ereignisprotokoll	list<Ereignis>	0,*	geordnete Liste der Veränderungen des Informationsobjekts mit Zeitstempel und Ereignistyp

Tabelle 36: Typvereinbarung (Attributschema) des Metaobjekttyps *Informationsobjekt*

Im `Wert`-Attribut können einem Informationsobjekt beliebige Metadaten beigefügt werden. Für Dokumente existiert hierzu mit dem DUBLIN-CORE-Metadatenschema ein anwendungsunabhängiger Standard. Er gibt eine Menge von Deskriptoren für Web-Ressourcen vor (z.B. ID, Format, Type, Language, Title, Subject, Description, Creator, Source, Date) und ist vor allem im Bereich Digitaler Bibliotheken verbreitet [PrKK05, 1311].

Metaobjekttyp	**Datenquelle**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung/Identifikator der Datenquelle
Beschreibung (V)	string	0,1	inhaltliche Beschreibung der Datenquelle (ihrer Semantik)
Ansprechpartner	set<Person>	0,*	Menge von Personen, welche die Datenquelle betreuen
Ressourcentyp	string	1,1	Organisationsform (Art) der Datenquelle; {Datenbank | Web | Dokument | Institution | Erhebung | Auskunft}

(Fortsetzung auf nächster Seite)

Datenquellentyp	string	1,1	Typ der Datenquelle; {primär | sekundär.operativ | sekundär.analytisch}
Erhebungsobjekt	Domänenobjekt → Ontologie	1,1	Domänenobjekt, dem die Datenquelle konzeptuell zugeordnet ist (vgl. Perspektive)
Datenquellen-profil	set<Deskriptor>	0,*	Menge von formalen Eigenschaften zur Beschreibung der Datenquelle
Informations-objekte	set<Informations-objekt>	0,*	Menge der aktuell in der Datenquelle enthaltenen Informationsobjekte
Server	set<Software-Installation>	0,1	Server, von dem eine elektronische Datenquelle bereitgestellt wird

Tabelle 37: Typvereinbarung (Attributschema) des Metaobjekttyps *Datenquelle*

Metaobjekttyp	**Software-Installation (Server)**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung/Identifikator der Software-Installation
Beschreibung (V)	string	0,1	Beschreibung der Software-Installation
Software-Produkt	set<Software-Produkt>	1,*	Menge der Software-Produkte, welche die Installation umfasst (Werkzeuge, Anwendungssysteme, Programmbibliotheken)
Standort	string	0,1	Beschreibung des physischen Standorts der Software-Installation

(Fortsetzung auf nächster Seite)

Zugehörigkeit	string	1,1	Kennzeichnung der Zugehörigkeit zum Unternehmen (intern | extern)
Organisation	Begriff → Ontologie	0,1	Organisationseinheit, welche die Installation betreut
Unternehmen	Begriff → Ontologie	0,1	Unternehmen, welches die Installation betreut
Ansprechpartner	set<Person>	0,*	Menge von Personen, welche die Installation betreuen
Dienstmerkmale	set<Deskriptor>	0,*	Menge nicht-funktionaler Eigenschaften der Diensterbringung, welche die Dienstqualität betreffen
Kostensatz	Bewertungsfaktor	0,1	Angabe eines Kostensatzes (z.B. je Stunde) für die Inanspruchnahme der Dienste dieses Servers
Leistungs-faktoren	set<Bewertungs-faktor>	0,*	Menge von Bewertungsfaktoren, welche die Auswahl des Servers unterstützen
URL	string	0,1	Adresse (Uniform Resource Locator), unter der die Installation im Netzwerk erreichbar ist
Aufrufnachricht	string	0,1	parametrisierbare Nachricht zur Auslösung eines Operators der Software-Installation
Typ	string	1,1	Typ der Software-Installation; {Datenserver | Analyseserver | Analyse-werkzeug | ... }
Datenquelle	set<Datenquelle>	0,*	Menge von Datenquellen, die ein Server vom Typ=Datenserver verwaltet

Tabelle 38: Typvereinbarung (Attributschema) des Metaobjekttyps *Software-Installation (Server)*

Metaobjekttyp	**Person**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Nachname	string	1,1	Nachname der Person
Vorname	string	1,1	Vorname(n) der Person
Titel	string	0,1	Titel der Person
Beschreibung (V)	string	0,1	ergänzende Angaben zur Person
Stelle	string	0,1	Funktion/Aufgabenbereich der Person
Zugehörigkeit	string	1,1	Kennzeichnung der Zugehörigkeit zum Unternehmen (intern | extern)
Organisation	Begriff → Ontologie	0,1	Organisationseinheit, der die Person angehört
Unternehmen	Begriff → Ontologie	0,1	Unternehmen, dem die Person angehört
Adresse	set<string>	0,*	Adresse
Telefon	set<string>	0,*	Telefonnummer
Rollen	set<Rolle>	0,*	Menge der Rollen, die die Person übernehmen kann
Qualifikations-profil	set<Deskriptor>	0,*	Menge konkreter Qualifikationen, über welche die Person verfügt
Kostensatz	Bewertungsfaktor	0,1	Angabe des Kostensatzes (z.B. je Stunde) für die Beauftragung der Person

(Fortsetzung auf nächster Seite)

Software-Installation	set<Software-Installation>	0,*	Menge von Servern, welche von der Person betreut werden
Datenquellen	set<Datenquelle>	0,*	Menge von Datenquellen, welche von der Person betreut werden

Tabelle 39: Typvereinbarung (Attributschema) des Metaobjekttyps *Person*

A4.5 Attributschemata für Ontologien

Attribute anderer Attributschemata, die mit dem Hinweis → Ontologie gekennzeichnet sind, profitieren davon, wenn sie als Begriffe einer Ontologie repräsentiert werden. Andernfalls sind die Attribute jeweils mit Datentyp `string` abzubilden und entsprechende Ausprägungen zu wählen.

In den Tabellen sind Hinweise zur Transformation der Attribute in OWL enthalten, die sich am Standard orientieren [W3C09]. Transformationsregeln sind nicht Gegenstand dieser Arbeit.

Metaobjekttyp	**Begriff**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung des Begriffs
Typ	string	1,1	Typ des Begriffs; {`owl:Class` | `owl:NamedIndividual`}
Anmerkungen	set<string>	0,*	Menge von Annotationen zum Begriff; (vgl. `owl:AnnotationProperty`)
Eigenschaften	set<literal>	0,*	Menge von Eigenschaften des Begriffs; (vgl. `owl:DataProperty`)
Restriktionen	set<string>	0,*	Menge von Restriktionen zum Begriff; (vgl. `owl:ClassExpression`)
Relationen	set<Relation>	0,*	Menge von Relationen zu anderen Begriffen; (vgl. `owl:ObjectProperty`)

Tabelle 40: Typvereinbarung (Attributschema) des Metaobjekttyps *Begriff*

Relationen finden abhängig von ihrem Typ unterschiedlichen Niederschlag in OWL. So kann eine Typbeziehung z.B. als `owl:ClassExpression`, eine Interaktion als `owl:ObjectProperty` abgebildet werden.

Metaobjekttyp	**Relation**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Bezeichnung der Relation (ihrer Semantik); Werte abhängig von `Typ`: {ist_ein | Teil_von | Instanz_von | <definiertbar>}
Typ	string	1,1	Typ der Relation; {Typbeziehung | Aggregation | Instanziierung | Interaktion}
Startbegriff	Begriff	1,1	Begriff, von dem die Relation ausgeht
Zielbegriff	Begriff	1,1	Begriff, in den die Relation mündet
Anmerkungen	set<string>	0,*	Menge von Annotationen zur Relation; (vgl. `owl:AnnotationProperty`)
Eigenschaften	set<literal>	0,*	Menge von Eigenschaften der Relation; (vgl. `owl:DataProperty`)
Restriktionen	set<string>	0,*	Menge von Restriktionen zur Relation; (vgl. `owl:ClassExpression`)
Kardinalität 1	unsigned short	0,1	Anzahl minimal erforderlicher Objekte von `Zielbegriff`; (vgl. `owl:ObjectMinCardinality`)
Kardinalität 2	unsigned short	0,1	Anzahl maximal erlaubter Objekte von `Zielbegriff`; (vgl. `owl:ObjectMaxCardinality`)

Tabelle 41: Typvereinbarung (Attributschema) des Metaobjekttyps *Relation*

A4.6 Verwendete abgeleitete und strukturierte Datentypen

Im Folgenden sind von den oben deklarierten Attributschemata verwendete Datentypen in alphabetischer Reihenfolge aufgeführt.

A

Datentyp	**Abstammung** *Beschreibung der Herleitung eines Objekts aus einem Vaterobjekt zusammen mit dem angewendeten Kriterium*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Vaterobjekt	Objekttyp	1,1	Bezeichnung des Objekttyps, von dem die Herleitung ausgeht
Kriterium	Begriff → Ontologie	1,1	Kriterium, nach dem die Herleitung erfolgt ist (auch als Ableitungsoperator definierbar)
Argumente	set<string>	0,*	Menge von Argumenten zur näheren Beschreibung bzw. Instanziierung von `Kriterium`

Tabelle 42: Typvereinbarung (Attributschema) des strukturierten Datentyps *Abstammung*

Datentyp	**Analysefrage** *natürlichsprachige und strukturierte inhaltliche Spezifikation des Informationsbedarfs*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Fragetext	string	1,1	natürlichsprachige Formulierung der Analysefrage zur inhaltlichen Darstellung des Informationsbedarfs

(Fortsetzung auf nächster Seite)

Aussagetyp	set<Begriff>	1,*	Typ der Aussage, die in Bezug auf die `Argumente` als Information gewünscht wird
Argumente	set<string>	1,*	Merkmale oder empirische Begriffe zur Instanziierung bzw. näheren Bestimmung von `Aussagetyp`
Beschreibungs-dimensionen	set<string>	0,*	Bezugsobjekte, Vergleichs-/ Verhältnisgrößen oder Ordnungs-/ Sortierkriterien zur genaueren Beschreibung der Aussage
Selektions-dimensionen	set<string>	0,*	Objekte/Objektklassen sowie sachliche, zeitliche oder räumliche Kriterien zur Einschränkung der Aussage

Tabelle 43: Typvereinbarung (Attributschema) des strukturierten Datentyps *Analysefrage*

Datentyp	**Analyseobjekt** *inhaltliche Spezifikation des Datenbedarfs*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
–	set<Informations-objekt>	1,*	Menge von Informationsobjekten

Tabelle 44: Typvereinbarung (Attributschema) des abgeleiteten Datentyps *Analyseobjekt*

Datentyp	**Anwendung** *Kennzeichnung eines Problemaspekts* *in Gestalt eines beobachteten oder anzustrebenden Ereignisses*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Problem-merkmal	Domänenobjekt-merkmal → Ontologie	1,1	Name des zu betrachtenden Domänenobjektmerkmals; Schema: Problemobjekt.Problemmerkmal
Modifikator	Modifikator	1,1	Typ des relevanten Veränderungsereignisses
Zustandsversion	string	1,1	Version des zu betrachtenden Zustands, wie er durch Problemobjekt.Problem-merkmal charakterisiert ist; {Ist | Soll}

Tabelle 45: Typvereinbarung (Attributschema) des strukturierten Datentyps *Anwendung*

Datentyp	**Änderungsoperation** *Kennzeichnung einer Operation* *zur Veränderung eines Prozessschemas auf Typ- oder Instanzebene*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Typ	Begriff → Ontologie	1,1	Typ der Änderungsoperation
Änderungsobjekt	Prozessbaustein	1,1	Gegenstand der Änderung (z.B. einzufügende Aktivität)
Parameter	set<Begriff>	0,*	Liste typspezifischer Parameter

(Fortsetzung auf nächster Seite)

Begründung	string	0,1	Erläuterung/Begründung der vorgenommenen Änderungen
Dauerhaftigkeit	boolean	1,1	Kennzeichen, ob eine Änderung dauerhaft protokolliert werden soll (1), oder ob sie nur lokal für die konkrete Instanz gilt (0)

Tabelle 46: Typvereinbarung (Attributschema) des strukturierten Datentyps *Änderungsoperation*

Eine Änderungsoperation *op: = (opType, s, paramList)* bestimmt sich durch den Operationstyp *opType* (z.B. Einfügeoperation), den Änderungsgegenstand *s* (z.B. die einzufügende Aktivität) und operationsspezifische Parameter *paramList* (z.B. die Position der neuen Aktivität im Ablauf) [RWRW05, 255]. Durch das Kennzeichen zur `Dauerhaftigkeit`, das bei Vornahme der Änderung zu setzen ist, können temporäre Anpassungen, die nur lokal für die betrachtete Prozessinstanz gelten sollen, im Rahmen der Revision leicht als solche erkannt werden.

B

Datentyp	**Bewertungsergebnis** *ist_ein Bewertungskriterium* *Zielerreichungsgrad der realisierten Ausprägung* *eines Bewertungskriteriums als Resultat eines Bewertungsvorgangs*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
	+ alle Attribute des Bewertungskriteriums		
Istwert	literal	1,1	realisierte Ausprägung von `Bewertungskriterium.Merkmal`. Wertebereich abhängig von `Bewertungskriterium.Datentyp`.

(Fortsetzung auf nächster Seite)

Zielerreichung	unsigned short	0,1	Grad der Zielerreichung, den der `Istwert` im Hinblick auf `Bewertungskriterium.Zielwert` repräsentiert, ausgedrückt in Prozent.

Tabelle 47: Typvereinbarung (Attributschema) des strukturierten Datentyps *Bewertungsergebnis*

Datentyp	**Bewertungsfaktor** *Kosten-, Leistungs- oder Kapazitätsfaktor als Berechnungsgrundlage*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name	string	1,1	Bezeichnung des Bewertungsfaktors (z.B. Personalkosten)
Wertgröße	float	1,1	numerischer Wert, der je Einheit der `Bezugsgröße` anzusetzen ist (z.B. **60** EUR)
Einheit	Begriff → Ontologie	1,1	Einheit zur `Wertgröße` (z.B. 60 **EUR**)
Bezugsgröße	Begriff → Ontologie	1,1	Einheit einer Bezugsgröße, die als Multiplikator in die Berechnung eingeht (z.B. h für den Bewertungsfaktor 60 EUR/h)

Tabelle 48: Typvereinbarung (Attributschema) des strukturierten Datentyps *Bewertungsfaktor*

Datentyp	**Bewertungskriterium** *angestrebter Zielwert eines Merkmals* *als Grundlage für Bewertungsvorgänge*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Merkmal	Begriff → Ontologie	1,1	Merkmal, nach dem die Bewertung erfolgt
Datentyp	Datenobjekttyp → Ontologie	1,1	Datentyp von `Merkmal`
Zielwert	literal	1,1	angestrebte Ausprägung von `Merkmal`. Wertebereich ist abhängig von `Datentyp`.
Gewicht	short	1,1	Wichtungsfaktor zur Berücksichtigung der Bedeutung von `Merkmal`; (Standardwert = 1)

Tabelle 49: Typvereinbarung (Attributschema) des strukturierten Datentyps *Bewertungskriterium*

Ein Kriterium ist gemäß [Dude17c] ein „unterscheidendes Merkmal als Bedingung für einen Sachverhalt, ein Urteil, eine Entscheidung". Demnach umfasst ein operationales Bewertungskriterium neben dem `Merkmal`, nach dem die Beurteilung stattfinden soll, insbesondere auch eine konkrete Ausprägung dieses Merkmals, die als Bedingung für die Entscheidung über die Erfüllung des Kriteriums fungiert. In komplexen Beurteilungssituationen können einzelne Kriterien zudem unterschiedliche Bedeutung besitzen, was durch Wichtungsfaktoren berücksichtigt werden kann (vgl. auch Nutzwert-Analysen [KaRe91, 943]). Gesamtbewertungen als Aggregation der Bewertungen einzelner Kriterien können aus diesen berechnet werden. Das `Gewicht` der Gesamtbewertung entspricht der Summe der Gewichte der Einzelbewertungen, woraus sich im `Bewertungsergebnis` leicht die Gesamt-`Zielerreichung` errechnen lässt. Daher wird ein Bewertungskriterium als Tupel aus den Elementen `Merkmal`, `Datentyp` des Merkmals, angestrebter `Zielwert` des Merkmals und `Gewicht` verstanden.

D

Datentyp	**Deskriptor** *Schlüssel/Wert-Paar zur Beschreibung von Objekten oder Konzepten (vgl. ODMG-Typ `dictionary`)*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Schlüssel	Begriff → Ontologie	1,1	Name/Identifikator des Deskriptors
Wert	literal	1,1	Ausprägung des Deskriptors

Tabelle 50: Typvereinbarung (Attributschema) des strukturierten Datentyps *Deskriptor*

Metaobjekttyp	**Domänenobjekt** *Objekttyp oder Konzept der Anwendungsdomäne* → **Ontologie**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Name/Identifikator des Domänenobjekts
Beschreibung (V)	string	0,1	Beschreibung der Bedeutung/ Definition des Objekttyps
Merkmale	set<Domänen-objektmerkmal>	1,*	Menge von konzeptuellen Merkmalen, die das Domänenobjekt charakterisieren

Tabelle 51: Typvereinbarung (Attributschema) des Metaobjekttyps *Domänenobjekt*

Metaobjekttyp	**Domänenobjektmerkmal** *Merkmal eines Objekttyps oder Konzepts der Anwendungsdomäne* → **Ontologie**		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Name/Identifikator des Domänenobjektmerkmals
Beschreibung (V)	string	0,1	Beschreibung der Bedeutung/ Definition des Merkmals
Datentyp	Datenobjekttyp → Ontologie	1,1	Datentyp des Merkmals
Restriktionen	set<string>	0,*	Menge von Ausdrücken zur Einschränkung des Merkmals
Domänenobjekt	Domänenobjekt	1,1	Domänenobjekt, welches das Merkmal beschreibt

Tabelle 52: Typvereinbarung (Attributschema) des Metaobjekttyps *Domänenobjektmerkmal*

E

Datentyp	**Ereignis** *Zustandsänderung mit Zeitstempel*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Zeit	timestamp	1,1	Zeitstempel des Ereigniseintritts
Typ	Begriff → Ontologie	1,1	Typ des Ereignisses bzw. der Zustandsänderung

Tabelle 53: Typvereinbarung (Attributschema) des strukturierten Datentyps *Ereignis*

F

Datentyp	**Funktionsempfehlung** *kommentierte Verknüpfung mit einer Funktion*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Titel	string	1,1	Titel der Funktionsempfehlung
Funktion	Funktion	1,1	Funktion der Prozessebene als Verknüpfungsziel
Kommentare	set<Kommentar>	0,*	Menge von Textkommentaren und Anmerkungen zur Funktionsempfehlung
Rang	unsigned short	1,1	Rang der Empfehlung in einer Liste (Präferenz oder Geeignetheit des Eintrags gegenüber anderen Einträgen)

Tabelle 54: Typvereinbarung (Attributschema) des strukturierten Datentyps *Funktionsempfehlung*

I

Datentyp	**Instanzzustand** *Typ des aktuellen Zustands einer Prozess- oder Aktivitätsinstanz (Vorgang)*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
–	string	1,1	Kennzeichnung des Zustandstyps; zulässige Ausprägungen: {inaktiv | wartend | angelegt | bereit | aktiv | beendet | abgeschlossen | abgebrochen}

Tabelle 55: Typvereinbarung (Attributschema) des abgeleiteten Datentyps *Instanzzustand*

Zu den Ausprägungen siehe ausführlicher das Zustandsmodell in Abschnitt 4.5.3.1.

K

Datentyp	**Kommentar** *benannter Kommentar mit Informationen zu Autor und Erstellungszeit*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Titel	string	1,1	Titel des Kommentars
Inhalt	string	1,1	Text/Inhalt des Kommentars
Autor	Person	0,1	Verfasser des Kommentars
Erstellung	timestamp	0,1	Zeitstempel der Erstellung des Kommentars

Tabelle 56: Typvereinbarung (Attributschema) des strukturierten Datentyps *Kommentar*

L

Datentyp	**Link** *benannte Verknüpfung zu beliebigen Ressourcen*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Titel	string	1,1	Titel der Verknüpfung
URL	string	1,1	Universal Resource Locator (URL), über den die Ressource auffindbar ist
MIME-Typ	Begriff → Ontologie	1,1	Medientyp gemäß MIME zur Klassifikation und korrekten Handhabung der verknüpften Ressource
Autor	Person	0,1	Verfasser des Links

(Fortsetzung auf nächster Seite)

Erstellung	timestamp	0,1	Zeitstempel der Erstellung des Links
Kommentare	set<Kommentar>	0,*	Menge von Textkommentaren und Anmerkungen zum Link

Tabelle 57: Typvereinbarung (Attributschema) des strukturierten Datentyps *Link*

M

Datentyp	**Modifikator** *Typ eines Veränderungsereignisses für Domänenmerkmale*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
–	string	1,1	Kennzeichnung des Veränderungsereignistyps; zulässige Ausprägungen: + : Erhöhung, Verbesserung – : Reduzierung, Verschlechterung = : Stabilisierung, Stagnation o : Eliminierung, Wegfall * : Herstellung, Auftreten

Tabelle 58: Typvereinbarung (Attributschema) des abgeleiteten Datentyps *Modifikator*

P

Datentyp	**Parameter** *ist_ein Datenobjekttyp* *Spezifikation von Parametern mit Erläuterungen und Ausprägungen*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
	+ alle Attribute des Datenobjekttyps		
Wert	literal	0,1	Ausprägung des Parameters (kann auch Standardwert dokumentieren)
Beschreibung (V)	string	0,1	Ausführliche Beschreibung der Bedeutung des Parameters
Hilfetext	string	0,1	Anleitung zur Unterstützung des Analytikers bei der Wahl von Ausprägungen (`Wert`)
Kardinalität	unsigned short	1,1	Anzahl der Ausprägungen des Parameters, die anzugeben sind; {0: fakultativ | 1: obligatorisch | >1: multipel}
Typ	string	1,1	Typ des Parameters als Angabe der Rolle, die er für den Operator spielt (Richtung); {in: Eingabe | out: Ausgabe | inout: Ein- und Ausgabe}

Tabelle 59: Typvereinbarung (Attributschema) des strukturierten Datentyps *Parameter*

Datentyp	**Perspektive** *Beschreibung einer Sichtweise auf das Untersuchungsobjekt*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Name (V)	string	1,1	Name der Perspektive Schema:`Erhebungsobjekte.Datenquelle`
Erhebungs-objekte	list<Domänen-objekt>	1,*	Menge von Domänenobjekten, bei denen die Daten über das Untersuchungsobjekt erhoben werden
Datenquellen	list<Datenquelle>	1,*	Menge von Datenquellen, welche mit jeweils einem Erhebungsobjekt korrespondieren
Geschäfts-prozesse	set<Begriff>	0,*	Menge von Geschäftsprozessen, welche die Perspektive repräsentiert

Tabelle 60: Typvereinbarung (Attributschema) des strukturierten Datentyps *Perspektive*

R

Datentyp	**Rollenzuordnung** *Abbildung von Datenabhängigkeiten (Flüssen) auf Ein- oder Ausgabedaten eines Operators*		
Attributname	*Wertebereich*	*Kard.*	*Beschreibung*
Fluss	string	1,1	Bezeichnung des Flusses als Referenz auf `Datenabhängigkeit.Name`
Rolle	string	1,1	Bezeichnung der Ein- oder Ausgabedaten als Referenz auf `Operator.Eingabedaten.Name` bzw. `Operator.Ausgabedaten.Name`

Tabelle 61: Typvereinbarung (Attributschema) des strukturierten Datentyps *Rollenzuordnung*

A4.7 Beziehungsmetamodell

Die ebenenübergreifenden Beziehungen zwischen Metaobjekttypen sind im folgenden integrierten Beziehungsmetamodell dargestellt (Abbildung 121).

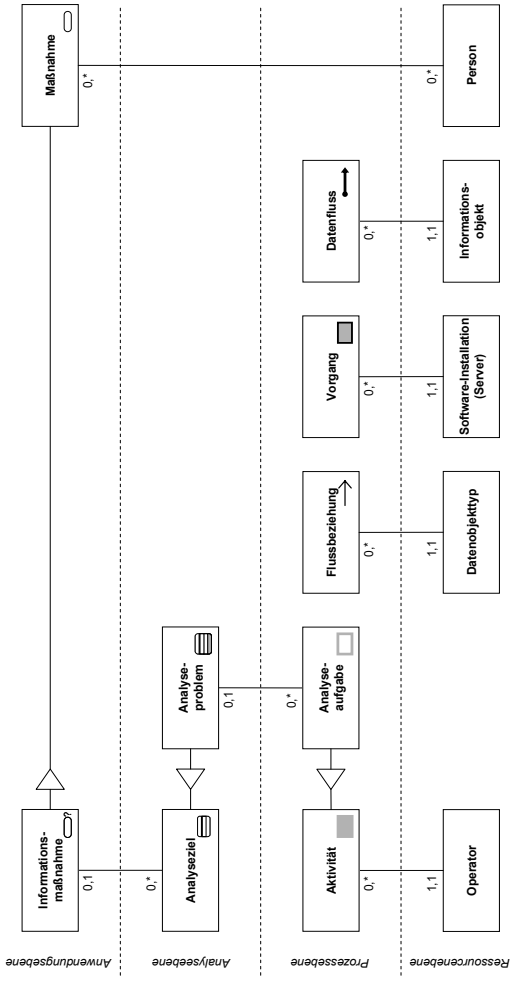


Abbildung 121: Integriertes Beziehungsmetamodell zum Modellierungsansatz (eigene Darstellung)

Eine `Informationsmaßnahme` der *Anwendungsebene* kann durch beliebig viele (0,*) `Analyseziele` der Analyseebene konkretisiert werden. Umgekehrt verweist ein Analyseziel typischerweise auf exakt eine Informationsmaßnahme. Da die Modellierung der Anwendungsebene nicht zwingend ist, ist die Kardinalität als 0,1 dargestellt. Einer `Maßnahme` können beliebig viele `Personen` als Verantwortliche zugeordnet werden. Eine Person kann in mehreren, muss aber in keiner Maßnahme referenziert werden.

Einem `Analyseproblem` der *Analyseebene* können beliebig viele (0,*) `Analyseaufgaben` eines oder mehrerer Prozesse zugewiesen sein, die ein Lösungsverfahren zum Analyseproblem beschreiben. Da in bestimmten Fällen auch auf die Modellierung auf Analyseebene verzichtet werden kann, ist auch hier die inverse Beziehung optional (0,1). Um die Vorteile der durchgängigen Methodik nutzen zu können ist jedoch zu empfehlen, für jeden Prozess ein Analyseproblem zu spezifizieren.

Eine `Aktivität` auf *Prozessebene* ist über die Zuordnung genau eines (1,1) `Operators` der *Ressourcenebene* zu einer Aufgabe definiert. Ein Operator kann in beliebig vielen (0,*) Aktivitäten zum Einsatz gelangen. Eine `Flussbeziehung` referenziert genau einen (1,1) `Datenobjekttyp`, der beliebig viele (0,*) Datenflüsse typisieren kann. Innerhalb der Instanzensicht bestehen Zuordnungsbeziehungen zwischen `Vorgang` und `Datenfluss` einerseits und jeweils genau einem Element (1,1) von `Software-Installation (Server)` bzw. `Informationsobjekt` andererseits. Die inversen Beziehungen sind beliebig (0,*).

A5 Kataloge von Deskriptoren

Dieser Abschnitt zeigt exemplarisch Vorschläge für Deskriptoren, um verschiedene Anforderungen oder Kontextfaktoren zu modellieren. Die Kataloge sind aus der Literatur recherchiert und in Kategorien gegliedert, stellen jedoch weder eine vollständige Aufstellung zur unmittelbaren Nutzung in automatisierten Systemen dar, noch enthalten sie Wertebereiche. Sie seien vielmehr als Anregung zum Aufbau konkreter Deskriptorenlisten verstanden.

A5.1 Informationsbedarfsprofil

Zur Spezifikation des Informationsbedarfsprofils, das formale Anforderungen an die Analyseergebnisse erhebt, stehen die in den nachfolgenden Tabellen gezeigten Klassen von Deskriptoren zur Verfügung. Sie werden im Anschluss kurz erläutert. Sie resultieren aus der Definition von ***Informationsbedarf*** als *Art*, *Menge* und *Qualität* von Informationsgütern, die in einem gegebenen Kontext zur Durchführung einer Aufgabe benötigt werden [Ball00, 29f.]. Zusätzlich wird eine Klasse mit anwendungsspezifischen *Nutzenkriterien* ergänzt. Dieser Katalog ist als Vorschlag zu verstehen, der verändert und ergänzt werden kann. Ausprägungen sind kriterienspezifisch zu bestimmen; in vielen Fällen sind ordinale Werte wie {niedrig, mittel, hoch} ausreichend.

Die **Art** der Information (Tabelle 62) umschließt neben ihrem Inhalt, der durch die Analysefrage hinreichend bestimmt ist, ihre Aussage- und Repräsentationsform [Bert75, 33, 39]. Die *Aussageform* ist davon abhängig, ob reale Sachverhalte, gedankliche Vorstellungen oder die Information selbst beschrieben werden sollen. Aussagen über reale Sachverhalte können faktischer, prognostischer oder explanatorischer Natur sein und sind aufgrund ihrer empirischen Überprüfbarkeit primärer Gegenstand der Datenanalyse. Faktische Aussagen beschreiben vergangene oder gegenwärtige Tatsachen (Ist-Aussagen). Prognostische Aussagen machen Angaben über die Zukunft (Wird-Aussagen) und sind grundsätzlich mit Unsicherheit behaftet. Explanatorische Aussagen erklären Tatsachen mithilfe von Kausalhypothesen (Warum-Aussagen). Nicht empirisch überprüfbare Aussagen über

gedankliche Vorstellungen können als Vergleichs- oder Bezugsgrößen von Interesse sein. Hierzu zählen konjunktive Aussagen, die denkbare Möglichkeiten ausdrücken (Kann-Aussagen), sowie normative Informationen über Ziele und Werturteile (Soll-Aussagen). Metasprachliche Aussagen über Informationen können logisch, explikativ oder instrumental ausgeprägt sein. Während logische Schlussfolgerungen (Muss-Aussagen) wie Deduktionen und Berechnungen im Rahmen der Datenanalyse verbreitet sind, besitzen explikative und instrumentale Informationen über Definitionen und Sprachregelungen bzw. über Methoden und Instrumente des Denkens eher Bedeutung für die Domänenanalyse (vgl. [Küpp05, 157f.]).

Art		
Aussageinhalt	***Analysefrage***	
	Analyseausrichtung	{explorativ, konfirmatorisch, schließend}
Aussageform	*Reale Sachverhalte*	{faktisch, prognostisch, explanatorisch}
	Gedankliche Vorstellungen	{konjunktiv, normativ}
	Metasprachliche Aussagen	{logisch, explikativ, instrumental}
Repräsentationsform	→ *Empfänger* → *Verwendungszweck*	***Interpretierbarkeit***
		Darstellungsform
		Komplexität → *Spezifität*

Tabelle 62: Artbezogene Aspekte zur Charakterisierung des Informationsbedarfs

Die *Repräsentationsform* ist im Hinblick auf Empfänger und Verwendungszweck der Analyseergebnisse zu wählen und wird durch die Aspekte Interpretierbarkeit, Darstellungsform und Komplexität charakterisiert. Die *Interpretierbarkeit* definiert eine Anforderung, welche die Wahl der Darstellungsform und der Komplexität der Ergebnisse

beeinflusst. Sie ist im Sinne der Verständlichkeit, Nachvollziehbarkeit und Erklärungsmächtigkeit zu verstehen. Zielt die Analyse auf die Beschreibung eines Sachverhalts (situationsbezogener Problemaspekt), so ist die Interpretierbarkeit der Aussagen für den Informationsempfänger von größter Bedeutung, um ein besseres Verständnis der Problemsituation zu ermöglichen. Wird hingegen ein lösungsbezogener Problemaspekt behandelt, tritt die Interpretierbarkeit häufig zugunsten der Genauigkeit der Ergebnisse in den Hintergrund, z.B. bei der Anwendung eines Prognose- oder Klassifikationsmodells. Da Interpretierbarkeit und Genauigkeit oft in konfliktärer Beziehung stehen, muss vor dem Hintergrund des zugehörigen Problemaspekts abgewogen werden, welches Formalziel zu priorisieren ist. Als *Darstellungsform* kommen z.B. grafische Darstellungen, Produktionsregeln, Tabellen, mathematische Gleichungen oder Modelle infrage, die jeweils unterschiedlich interpretierbar sind. Während Grafiken z.B. sehr gut verständlich sind, lassen sich künstliche Neuronale Netze gar nicht interpretieren (Black-Box-Modelle). Ebenfalls Auswirkungen auf die Interpretierbarkeit nimmt die *Komplexität* der Ergebnisse. Sie beschreibt die Übersichtlichkeit der Aussagen, die sich z.B. aus der Anzahl und Anordnung der Elemente von Berichten, aus der Anzahl erzeugter Regeln oder der in einzelnen Regeln enthaltenen Variablen ergibt. Tendenziell ist mit steigender Komplexität der Ergebnisse eine zunehmende Genauigkeit sowie abnehmende Verständlichkeit zu erwarten.

Die **Qualität** der Information (Tabelle 63) umfasst die Dimensionen Vollständigkeit, Gültigkeit und Bestimmtheit (vgl. [Bert75, 39f., 43]). Die *Vollständigkeit* nimmt Bezug auf Art- (Inhalt und Aussageform) und Mengeneigenschaften der Information (Extension der Domänenobjekte), die vor dem Hintergrund des zugehörigen Sachproblemaspekts entsprechend zu wählen sind. Die *Gültigkeit* der Information wird anhand der Kriterien Sicherheit, Genauigkeit, Zuverlässigkeit und Überprüfbarkeit bestimmt. Die *Sicherheit* entspricht der Wahrscheinlichkeit einer Aussage wahr zu sein, und beschreibt die Stärke einer Regel oder die Reichweite eines Musters. Sie ist nicht zwingend im Sinne des statistischen Wahrscheinlichkeitsbegriffs zu interpretieren, lässt sich aber häufig anhand statistischer Maße wie z.B. Support (relative

Häufigkeit) oder Konfidenz abschätzen. Abhängig vom Analysezweck kann es durchaus von Interesse sein, seltene Phänomene mit geringer Wahrscheinlichkeit zu untersuchen, etwa bei der Missbrauchs- oder Betrugserkennung. Im Regelfall sind starke Muster zu präferieren.

Qualität		
Vollständigkeit	→ *Sachproblemaspekt*	
Gültigkeit	***Sicherheit***	z.B. Support, Konfidenz (Reichweite, Stärke)
	Genauigkeit	▪ Messgenauigkeit ▪ Treffgenauigkeit ▪ Anteil unentscheidbarer Fälle
	Zuverlässigkeit	→ *Glaubwürdigkeit und Objektivität der Datenquelle*
	Überprüfbarkeit *(empirisch / logisch)*	
Bestimmtheit	***Detailliertheit***	▪ Skalenniveau ▪ Aggregationsgrad
	Zeit	▪ Zeitbezug (Zeitpunkt/Historie) ▪ Aktualität (Alter)

Tabelle 63: Qualitätsaspekte zur Charakterisierung des Informationsbedarfs

Die *Genauigkeit* einer Aussage äußert sich in zwei Facetten. Zum einen beschreibt die Messgenauigkeit die Streuung der Messergebnisse. Sie wird einerseits von der Reliabilität der Operationalisierung, andererseits von der Spezifität der Aussagen beeinflusst. Letztere gibt Auskunft über die von der Aussage abgedeckten Fälle und ist positiv mit deren Komplexität und negativ mit der Sicherheit korreliert. Da spezifische Muster weniger Fälle repräsentieren, weisen sie tendenziell eine höhere Varianz der Mess- bzw. Prognosewerte auf, verfügen also über geringere Messgenauigkeit (vgl. [AlWa97, 5]). Zum zweiten gibt die Treffgenauigkeit die Abweichung der Messwerte vom jeweils wahren Wert an

[Küpp05, 158]. Insbesondere bei Behandlung lösungsbezogener Problemaspekte ist die Treffgenauigkeit ein dominierendes Kriterium, das bei Inferenzen häufig als Vorhersagegenauigkeit oder Fehlklassifikationsrate quantifiziert wird. Die Genauigkeit einer Inferenz kann auch durch einen zu hohen Anteil unentscheidbarer Fälle beeinträchtigt werden, d.h. durch Objekte, für die keine Klassifikation oder Prognose möglich ist. Da in diesen Fällen manuelle Eingriffe erforderlich sind, kann eine Minimierung dieses Anteils ein wichtiges Formalziel der Analyse darstellen. Die *Zuverlässigkeit* (Bestätigungsgrad) ist eine generische Anforderung, die Information immer erfüllen sollte. Bei der Auswahl von Datenquellen ist jedoch stets auf dieses Kriterium zu achten, da deren Glaubwürdigkeit und Objektivität nicht vorausgesetzt werden können. Die *Überprüfbarkeit* ist bei logischen Aussagen am größten, die jedoch nur Schlussfolgerungen, aber keine direkten Aussagen über die Realität treffen. In der Klasse der empirischen Aussagen sind faktische Informationen am besten überprüfbar (vgl. [Küpp05, 158]).

Die *Bestimmtheit* von Informationen wird durch Detailliertheit und Zeit definiert. Die *Detailliertheit* wird vom Skalenniveau und vom Aggregationsgrad der Informationen bestimmt. Das Skalenniveau schränkt die Präzision der Werte ein, die klassifikatorische, komparative oder metrische Begriffe darstellen können. Der Aggregationsgrad gibt vor, auf welcher Verdichtungsstufe der Objekte bzw. Merkmalswerte Aussagen getroffen werden sollen. So können etwa Kundentransaktionen, Kunden oder Kundengruppen betrachtet sowie Zeit- und Datumswerte z.B. auf Tages-, Wochen-, Monats- oder Jahresebene aggregiert werden. Die Bestimmtheit der Ergebnisse bezüglich der *Zeit* äußert sich im Zeitbezug und in der Aktualität. Der inhaltliche Zeitbezug wird zwar bereits mit dem Aussageinhalt spezifiziert, für Vergleiche im Zeitverlauf sind aber historisierte Daten erforderlich, die nicht in allen Datenquellen verfügbar sind. Daher wird mit der zeitlichen Bestimmtheit zusätzlich festgelegt, ob ein Zeitpunkt oder eine Historie betrachtet wird. Die Aktualität definiert Anforderungen an das Alter der Informationen.

Die Charakterisierung der **Menge** des Informationsbedarfs (Tabelle 64 oben) ist einerseits inhaltlich in Bezug auf die *Extension (Instanzen) der beschriebenen Domänenobjekte*, andererseits im Hinblick auf die *Häufigkeit der Informationsbereitstellung* zu verstehen. Der erste Aspekt betrifft die Frage, ob die Gesamtpopulation oder eine Stichprobe der Untersuchungsobjekte zu beschreiben ist. Bezüglich des zweiten Aspekts ist zwischen einmaligem und periodischem Informationsbedarf zu differenzieren. Falls eine periodische Bereitstellung gewünscht wird, ist die Frequenz bzw. das Zeitintervall zu bestimmen, nach denen die Information jeweils erzeugt werden soll (z.B. täglich, wöchentlich, monatlich).

Menge	
Extension (Objektinstanzen)	{Gesamtpopulation, Stichprobe}
Häufigkeit der Informationsbereitstellung	{einmalig, periodisch}
Nutzen	
Betriebswirtschaftliche Ziele	→ *Wertbeitrag des Sachproblems; z.B. Rentabilität, Gewinn*
Domänenspezifische Kriterien	z.B. Antwortrate, Risiko, Deckungsbeitrag

Tabelle 64: Mengen- und Nutzenaspekte zur Charakterisierung des Informationsbedarfs

Die genannten Kriterien können nicht alle Eventualitäten der betrieblichen Praxis umfassend abdecken. So sind etwa abhängig vom Medientyp weitere Qualitätskriterien denkbar, wie z.B. die Auflösung bei Bilddaten. Zwar könnte dieses Merkmal der Detailliertheit zugeordnet werden, es scheint aber dennoch angebracht, die Spezifikation weiterer Kriterien zu erlauben, da sich für jeden Problemfall zahlreiche *betriebswirtschaftliche* (z.B. Rentabilität, Gewinn, niedriges Risiko) oder *domänenspezifische Nutzenmaße* (z.B. hohe Antwortrate im Direktmarketing) ergeben. So bildet der Wertbeitrag des Sachproblems ein wichtiges Formalziel, das direkt als betriebswirtschaftliches Kriterium übernommen werden kann (Tabelle 64 unten).

WEISS ET AL. definieren **Nutzen** als Totalmaß für die Zufriedenheit mit dem gesamten Analyseprozess und benennen die Kosten der Datenbeschaffung, die Kosten der Analyse und die Kosten-Nutzen-Bilanz der Ergebnisanwendung als ökonomische Einflussfaktoren auf den Gesamtnutzen einer Untersuchung [WeZS08, 129f.]. Diese können in vielen Fällen als Funktion messbarer Kriterien ausgedrückt werden, wie etwa der Datenmenge, des Zeitbedarfs für die Algorithmenanwendung oder der Bestimmtheit der erzeugten Information. Gleichwohl existieren gerade bezüglich der Ergebnisanwendung stets Formalziele, die von solchen Kriterien unabhängig sind. Beispiele sind die Maximierung des Deckungsbeitrags oder des Gewinns der im Rahmen einer Werbekampagne verkauften Produkte oder eine hohe Wirksamkeit einer medizinischen Therapie mit möglichst geringen Nebenwirkungen (vgl. [ShYZ02]). Diese Ziele können nur durch spezifische Analyseverfahren, durch entsprechende Datenvorbereitung unter Anreicherung entsprechender Daten (vgl. [YaHB04]) oder durch geeignete Merkmalsauswahl erreicht werden, sofern die Ziele explizit spezifiziert sind.

A5.2 Verfahrensprofil

Das Verfahrensprofil besteht aus einer Menge von Deskriptoren, die bestimmte Eigenschaften eines Verfahrens beschreiben. Welche Deskriptoren einen konkreten Operator kennzeichnen ist einerseits verfahrensspezifisch, hängt andererseits aber auch davon ab, inwieweit entsprechende Informationen überhaupt bekannt oder dokumentiert sind. In der Literatur finden sich zahlreiche Aufstellungen von Verfahrenscharakteristika,[275] aus denen der in den nachfolgenden Tabellen dargestellte Katalog resultiert, der auch die Ein- und Ausgabedaten-Deklarationen aufführt (vgl. Kennzeichnung mit „INPUT“ bzw. „OUTPUT“). Er erhebt keinen Anspruch auf Vollständigkeit, verdeutlicht aber die wichtigsten Eigenschaftsklassen. Sie beschreiben anwendungs-, daten- und methodenorientierte Aspekte (vgl. [Küpp99, 87ff.]), die im Folgenden kurz erläutert werden.

[275] Vgl. für die hier gebrachte Aufstellung insbesondere [Küpp99, 87ff.], [BeLi97,422ff.], [Hogl03, 90f.], [WiFr00], [GiPo03], [HiKa01, 185ff.] und [Pyle99].

Ergebnisse			**OUTPUT**
Aussageinhalt	→ *Ausgabedaten-Deklaration*	***Aussagetyp*** *(Analysefrage)*	
Repräsentationsform	→ *Informations-bedarfsprofil*	***Interpretierbarkeit der Ergebnisse***	
		Darstellungsform	
		Komplexität	
Nutzen			
Nützlichkeit	***Charakterisierung von Unsicherheit***		
	Berücksichtigung externer Kostenwerte		
	Erfolgshistorie bei Anwendung auf ähnliche Datenbestände		
Benutzbarkeit	***Dokumentation***		
	Schwierigkeitsgrad		
	Nachvollziehbarkeit der Ergebnisberechnung		
	Verfügbarkeit	z.B. Ausfallwahrscheinlichkeit	
Geschwindigkeit	***Zeitbedarf für Ergebnisberechnung*** → *Leistungsfaktoren*		
Kosten			
Nutzungskosten	***Zugangskosten***	→ *Kostensatz des Servers, z.B. Lizenzkosten, Leistungsentgelt*	
	Personalkosten	→ *Zeitbedarf*	
Transformations-kosten	→ *absehbarer Aufwand für Datenvorbereitung*		

Tabelle 65: Anwendungsaspekte zur Charakterisierung von Operatoren

Anwendungsaspekte betreffen die produzierten Ergebnisse sowie Nutzen und Kosten eines Verfahrens (Tabelle 65). Der *Aussageinhalt* der **Ergebnisse** entspricht dem Aussagetyp der Analysefrage, wie in der Deklaration der Ausgabedaten des Operators beschrieben. Die *Repräsentationsform* umfasst Interpretierbarkeit, Darstellungsform und Komplexität der Ergebnisse und wird gegen korrespondierende Elemente des Informationsbedarfsprofils aus den Anforderungen der Aufgabe geprüft. **Nutzenaspekte** dienen der Bewertung der Tauglichkeit eines Verfahrens unabhängig vom produzierten Ergebnis. Als *Nützlichkeit* wird seine Fähigkeit zur Charakterisierung von Unsicherheit und zur Einbeziehung externer Kostenwerte dargestellt. Zusätzlich können Anwendungserfahrungen aus früheren Projekten die Beurteilung der Erfolgsaussichten erleichtern. Sie sind jedoch nur sinnvoll, sofern die betrachteten Erfahrungen sich auf ähnliche Analysedaten beziehen. Die *Benutzbarkeit* bestimmt sich durch die verfügbare Dokumentation und den Schwierigkeitsgrad, die Nachvollziehbarkeit der Ergebnisberechnung sowie die technische Verfügbarkeit, etwa im Hinblick auf die Ausfallwahrscheinlichkeit der Implementierung. Letztere kann z.B. vom zugehörigen Server abgeleitet werden. Für eine zeitnahe Bereitstellung der Analyseergebnisse kann die Geschwindigkeit des Verfahrens im Sinne des Zeitbedarfs für die Ergebnisberechnung (z.B. Trainings-, Test- und Anwendungslaufzeit eines Modells; vgl. Leistungsfaktoren) von Bedeutung sein. **Kosten** verursacht ein Analyseverfahren einerseits im Zuge seiner *Nutzung* durch Mitarbeiter (Personalkosten, abhängig vom erwarteten Zeitbedarf) und für den Zugang. Diese Zugriffskosten äußern sich, je nach Abrechnungsmodell, etwa als Lizenz- oder Leistungskosten und sind aus der Server-Beschreibung herzuleiten. Andererseits können *Transformationskosten* anfallen, wenn die Analysedaten erst in ein spezifisches Format zu überführen sind.

Datenaspekte nehmen Bezug auf Struktur und Medientyp sowie Datenmenge der zu verarbeitenden Daten (Tabelle 66). Datentyp, Medientyp und Restriktionen zu Verteilungsannahmen der Merkmalswerte sind in der Eingabedatendeklaration des Operators beschrieben. Zur Charakterisierung der *Datenmenge* dient zunächst das verarbeitbare Datenvolumen, das hier anders als in den Leistungsfaktoren des Operators ordinalskaliert angegeben ist. Hohe Werte verweisen auf gute

Skalierbarkeit und gutes Laufzeitverhalten des Verfahrens. Weiterhin kann die Variablenzahl explizit angegeben werden, die auch aus der Datendeklaration hervorgeht. Von besonderem Interesse ist hierbei die Anzahl abhängiger Merkmale, da nicht alle Verfahren imstande sind, mehr als eine Zielvariable zu verarbeiten. Häufig ist auch die Anzahl der Variablenwerte relevant, z.B. wenn Klassifikationsverfahren nur binäre Klassen (Zielvariablen) verarbeiten können. Mengenbezogene Angaben können gegen Datencharakteristika geprüft werden.

Daten			**INPUT**
Struktur und Medientyp	→ *Eingabedaten-Deklaration*	***Datentyp***	
		Medientyp	
		Verteilungsannahmen der Datenwerte	
Datenmenge → *Daten-charakteristika*	***verarbeitbares Datenvolumen***		
	Zahl verarbeitbarer Variablen	{univariates Verfahren, bivariates Verfahren, multivariates Verfahren}	
	Zahl verarbeitbarer Variablenwerte		

Tabelle 66: Datenaspekte zur Charakterisierung von Operatoren

Methodenaspekte beschreiben Typ, Verhalten sowie verfahrensklassenspezifische Eigenschaften der Operatoren. **Typbezogene Eigenschaften** (Tabelle 67) beziehen sich einerseits auf das *Wesen* des Verfahrens (z.B. die Herkunft aus einer bestimmten Disziplin), insbesondere auf dessen Fähigkeit zur Realisierung explorativer, konfirmatorischer oder schließender Analysen (Analyseausrichtung). Die Dimensionen des *Verfahrenstypus* geben weitere Hinweise auf die Anwendbarkeit eines Operators. Modellbezogene Verfahren sind typischerweise für alle Ausrichtungen geeignet (Training, Evaluierung, Anwendung), jedoch besonders sensitiv gegenüber Ausreißern und Datenmängeln. Nichtparametrische Methoden hingegen sind wegen ihrer starken Abhängigkeit von lokalen Strukturen nur für niedrigdimensionale Daten geeignet

(Merkmal Modellbezogenheit).[276] Die Beschreibungsform beeinflusst die erreichbare Trennschärfe bei Separierung des Datenraumes. Modelltyp, Zielerreichungsgrad (vgl. [FeSi13, 104-106]) und Optimierung drücken aus, ob ein Verfahren exakte und optimale Ergebnisse oder nur Näherungslösungen erzeugt.

Typ		
Wesen	***Ursprung und Herkunft***	z.B. Statistik, Maschinelles Lernen, etc.
	Analyseausrichtung	{explorativ, konfirmatorisch, schließend}
Verfahrenstypus	***Modellbezogenheit***	{parametrisch (modellbezogen), nicht-parametrisch (modellfrei)}
	Beschreibungsform	{achsenorthogonal separierend, linear, nicht-linear, nicht-funktional}
	Modelltyp	{analytisch, wissensbasiert, konnektionistisch}
	Zielerreichungsgrad	{exakt, approximierend, heuristisch, lernend}
	Optimierung	{optimierend, nicht optimierend}

Tabelle 67: Methodentypaspekte zur Charakterisierung von Operatoren

Verhaltenscharakteristika (Tabelle 68) geben Auskunft zum Vorgehen bei der Lösungsberechnung. Die *Autonomie* bestimmt sich aus der Überwachtheit, der Notwendigkeit zur Nutzerinteraktion (z.B. navigierende Verfahren, OLAP) und der Autarkie (Abhängigkeit von nutzerdefinierten Verfahrensparametern, z.B. Baumtiefe und Knotenumfang bei

[276] Nicht-parametrische (modellfreie) Verfahren ersetzen die Daten nicht durch ein parametrisiertes Modell, sondern nehmen zur Ergebnisberechnung direkt auf den ähnlichsten oder „nächstgelegenen" Datenpunkt Bezug (z.B. Nearest-Neighbor-Verfahren). Hochdimensionale Datenräume sind jedoch so dünn besetzt, dass die lokalen Nachbarschaften häufig leer sind [ElPr96, 101f.].

Entscheidungsbaumverfahren) und ist negativ mit dem Schwierigkeitsgrad korreliert. Überwachte Verfahren benötigen Beispieldaten, um Regelmäßigkeiten zu erlernen, unüberwachte Verfahren funktionieren auch ohne ein solches Training.

Verhalten		
Autonomie	***Überwachtheit***	
	Abhängigkeit von Nutzerinteraktion	
	Autarkie *(Notwendigkeit der Vorgabe von Verfahrensparametern)*	
Sensitivität	***Verletzung von Modellannahmen***	
	Datenmängel	
	Datendynamik → *Robustheit der Ergebnisse*	
Sucherverhalten	***Inkrementalität***	
	Konstruktivität	{selektiv, konstruktiv}
	Separierung des Suchraums	{diskret, kontinuierlich}
Lernverhalten	***Lernansatz***	{logisch, kompetitiv, schwellenbasiert}
	Lernstrategie	{lazy, eager}
	Variablen-handhabung	{sequenziell, parallel}
Performanz	→ *qualitative Schätzung*	

Tabelle 68: Methodenverhaltensaspekte zur Charakterisierung von Operatoren

Sensitivitätsmaße geben an, wie stark das Verfahren auf Verletzungen von Modell- oder Verfahrensannahmen, auf Datenmängel (z.B. fehlende Werte, Ausreißer) und Datendynamik reagiert. Letztgenannter Aspekt gibt Hinweise, wie häufig Ergebnisse bei hoher Datenänderungsrate

neu zu berechnen sind. Deskriptoren zum *Such- und Lernverhalten* erlauben z.B. Rückschlüsse auf Laufzeitverhalten und Ergebnisgenauigkeit (Inkrementalität, Konstruktivität, Lernansatz und -strategie, Variablenhandhabung) sowie die Geeignetheit für diskrete und kontinuierliche Variablen (Separierung des Suchraums [Pyle99, 109]). Zum Vergleich mehrerer Alternativen innerhalb einer Verfahrensklasse sind **klassenspezifische Charakteristika** hilfreich, etwa die Pruning-Strenge für Entscheidungsbäume oder Anzahl und Überlappung der Zentren für Radial-Base-Function-Netze [HiKa01, 184].

Große Bedeutung bei Inferenzmodellen hat typischerweise die **Performanz**, die jedoch aufgrund ihrer starken Abhängigkeit von den konkreten Analysedaten a priori nicht sinnvoll quantitativ, etwa in Gestalt von Genauigkeitsmaßen, abgeschätzt werden kann.

A5.3 Datenquellenprofil

Zur Spezifikation des Datenquellenprofils, das unabhängig von ihrem Inhalt (Aussagegehalt der Informationsobjekte) Eigenschaften einer Datenquelle beschreibt, stehen die in den nachfolgenden Tabellen gezeigten Deskriptoren zur Verfügung, die in die Klassen *Art*, *Qualität*, *Verfügbarkeit* und *Kosten* der Quelle bzw. der gespeicherten Daten eingeteilt sind. Sie werden im Anschluss kurz erläutert.

Zur **Art** der Daten (Tabelle 69) zählen neben dem Aussageinhalt, der sich aus der semantischen Annotation der Informationsobjekte und der vom Erhebungsobjekt eingenommenen Perspektive ergibt, auch die *Aussageform* sowie *Struktur und Medientyp*. Die Ausprägungen der Aussageform sind mit jenen des Informationsbedarfs identisch, wobei Analysedaten meist faktische, prognostische und normative, seltener auch explanatorische Aussagen[277] treffen, während die anderen Aussageformen häufig erst analytisch erzeugt werden. *Struktur und Medientyp* beeinflussen die Nützlichkeit der Daten ebenso wie die Wahl und Anwendbarkeit von Analyseverfahren. Strukturierte Daten sind

[277] Explanatorische Aussagen liegen oft in Form von Dokumenten vor (z.B. als Konzepte oder Beschlussprotokolle), während faktische, prognostische und normative Aussagen in der Regel als strukturierte Daten in Erscheinung treten.

Datenbankrelationen, Berichtstabellen, multidimensionale Datenstrukturen und objektorientierte Repräsentationen, unstrukturierte Daten sind alle Ausprägungen von Multimediadokumenten. Als *Übermittlungsmedium* sind elektronische Formen zu bevorzugen, da Daten auf physischen Medien (z.B. Fragebogen in Papierform) erst nach digitaler Erfassung der rechnergestützten Analyse zugänglich sind.

Art		
Aussageinhalt	***Merkmale des Untersuchungsobjekts***	Informationsobjekte
	Perspektive auf Untersuchungsobjekt	Erhebungsobjekt
Aussageform	*Reale Sachverhalte*	{faktisch, prognostisch, explanatorisch}
	Gedankliche Vorstellungen	{konjunktiv, normativ}
	Metasprachliche Aussagen	{logisch, explikativ, instrumental]
Struktur und Medientyp	***Medientyp***	Strukturierte Daten, z.B. Relationen & Tabellen, Hyperwürfel, Objekte Unstrukturierte Daten, z.B. Text/Dokumente, Grafiken, Foto, Audio, Video
	Übermittlungsmedium	{elektronisch, physisch}

Tabelle 69: Artbezogene Aspekte zur Charakterisierung von Datenquellen

Die **Qualität** des Datenangebots (Tabelle 70) wird anhand der Vollständigkeit, Gültigkeit und Bestimmtheit beurteilt. Die *Vollständigkeit*

bezieht sich auf die sachliche, räumliche und zeitliche Abdeckung aller relevanten Fälle zu untersuchender Sachverhalte. Während zahlreiche stichprobenbasierte Datenanalysen auf sachgerecht selektierten Teilpopulationen operieren, stellt eine darüber hinausgehende inhaltliche Unvollständigkeit einen Mangel dar, der die Erzeugung belastbarer Analyseergebnisse verhindern oder stark einschränken kann. Die *Gültigkeit* der Daten wird zunächst von der sachlich-inhaltlichen Korrektheit der Datenwerte bestimmt (*inhärente Datenqualität*, vgl. 3.1.2.2), die a priori aber mitunter schwierig zu prüfen ist. Gute Anhaltspunkte liefert die *Kontrolliertheit* der Daten, die sich durch den Grad der Nutzung und Interaktion darstellt. Die Nutzungshäufigkeit ist ein guter Indikator für die Datenqualität, da Fehler umso eher entdeckt und korrigiert werden, je intensiver mit den Daten gearbeitet wird (Datenqualitätsregelkreis, vgl. [Orr98]). Hier spielt auch die Art der Nutzung eine Rolle. So ist etwa zu erwarten, dass Kundendaten aus einem operativen CRM-System mit Schreibzugriff für den Kundenbetreuer von besserer Qualität sind als solche aus dem Marketing, auf die nur lesend zugegriffen wird. In diesem Zusammenhang sind auch mit anderen Daten verknüpfte Datenbestände positiver zu bewerten als isoliert gehaltene, da Fehler in der Interaktion mit anderen Prozessen tendenziell schneller zutage treten. Dieser Umstand wird als Integriertheit bezeichnet. Ebenso ist die *Zuverlässigkeit* der Datenquelle im Ganzen zu berücksichtigen, die von der Art der Datenerfassung sowie von der Objektivität und Glaubwürdigkeit geprägt wird. Datenerfassung mit personeller Beteiligung, etwa bei manueller Auftragsannahme oder an der Scannerkasse im Supermarkt, ist typischerweise weitaus fehlerträchtiger als die automatisierte Datenerhebung (z.B. bei Bestelldaten aus dem Onlineshop oder Telefonverbindungsdaten). Die Objektivität (Intersubjektivität) leidet z.B. bei Kommunikaten in Social Media oder bei Erhebung individueller Einschätzungen wie etwa der Kundencharakterisierung durch Außendienstmitarbeiter, und die Glaubwürdigkeit kann z.B. bei externen Daten von Interessenverbänden beeinträchtigt sein.

Die *Bestimmtheit* von Daten kann im Hinblick auf Zweckbezogenheit, Detailliertheit, Zeit und Personenbezogenheit beurteilt werden. Die

Zweckbezogenheit betrifft die unterschiedliche Adäquatheit von Primär- und Sekundärdaten und wird separat als Datenquellentyp dokumentiert.

Qualität		
Vollständigkeit	***Fallabdeckung***	▪ sachlich ▪ räumlich ▪ zeitlich
Gültigkeit	***Korrektheit*** *(inhärente Datenqualität)*	
	Kontrolliertheit	▪ Nutzungshäufigkeit ▪ Nutzungsart ▪ Integriertheit: {isoliert, integriert}
	Zuverlässigkeit	▪ Datenerfassung: {manuell, automatisiert} ▪ Objektivität ▪ Glaubwürdigkeit
Bestimmtheit	***Zweckbezogenheit*** *(Adäquatheit)*	{Primärdaten, Sekundärdaten} → *Datenquellentyp*
	Detailliertheit	▪ Skalenniveau ▪ Aggregationsgrad
	Zeit	▪ Aktualisierung (Frequenz, Datum) ▪ Auftreten: {diskret, kontinuierlich}
	Personen-bezogenheit	{gegeben, pseudonymisiert, anonymisiert, anonym, nicht gegeben}

Tabelle 70: Qualitätsaspekte zur Charakterisierung von Datenquellen

Die *Detailliertheit* ergibt sich wie beim Informationsbedarf aus dem Skalenniveau und dem Aggregationsgrad, wobei sich detaillierte Werte jederzeit verallgemeinern lassen, grobe Angaben aber kaum zu verfeinern sind. Die Bestimmtheit nach der *Zeit* entsteht zum einen durch Frequenz und Datum der letzten Aktualisierung, woraus die zeitliche Distanz der Daten zur Realität erkennbar ist. Zum anderen entsteht sie

durch das zeitliche Auftreten, das diskret (als jederzeit abrufbares, persistentes Abbild der Realität) oder kontinuierlich (als transienter Datenstrom) ausgeprägt sein kann und im zweiten Fall besondere Anforderungen an die Analyseverfahren stellt. Die *Personenbezogenheit* der Analysedaten ist von besonderer Bedeutung für die Einhaltung von Datenschutzgesetzen, da diese die Zulässigkeit der Auswertung personenbezogener Daten an bestimmte Voraussetzungen knüpfen. Personenbezogene Daten sind Einzelangaben über Merkmale einer bestimmten oder bestimmbaren Person, wozu auch Angaben über Personengruppen zählen, sofern dadurch Rückschlüsse auf Merkmale der zugehörigen Einzelpersonen möglich sind [NeKn15, 139]. Die Personenbezogenheit lässt sich nach den Ausprägungen gegeben, pseudonymisiert (Identifikationsmerkmale durch Kennzeichen ersetzt), anonymisiert (Identifikationsmerkmale eliminiert), anonym (z.B. Transaktionen über nicht identifizierbare Supermarktkunden) und nicht gegeben (z.B. Artikelstammdaten) charakterisieren.

Verfügbarkeit	
Beschaffbarkeit	***Zugänglichkeit***
	zeitliche Verfügbarkeit
Zulässigkeit	***Datenschutz-Freigabe*** → *Personenbezogenheit*
Öffentlichkeit	→ *Exklusivität der Daten*
Kosten	
Zugangskosten	z.B. für Zugriff, Erhebung, Erfassung
Beschaffungskosten	z.B. für Extraktion, Übertragung
Transformations-kosten	→ *absehbarer Aufwand für Datenvorbereitung*

Tabelle 71: Verfügbarkeits- und Kostenaspekte zur Charakterisierung von Datenquellen

Die Beurteilung der **Verfügbarkeit** von Daten (Tabelle 71) geschieht anhand der Kriterien Beschaffbarkeit, Zulässigkeit und Öffentlichkeit. Die *Beschaffbarkeit* kann durch Schwierigkeiten bezüglich der faktischen Zugänglichkeit und der zeitlichen Verfügbarkeit beeinträchtigt werden, wenn Daten nicht bzw. nicht rechtzeitig erhältlich sind. Die *Zulässigkeit* der Datennutzung für analytische Zwecke richtet sich nach der Personenbezogenheit. Die *Öffentlichkeit* der Daten bezieht sich auf die Zugänglichkeit für andere Interessenten. Grundsätzlich sind exklusive Daten gegenüber öffentlich verfügbaren vorzuziehen, da sie Wettbewerbsvorteile bringen können [Pyle03, 222f.].

Kosten der Datennutzung vor der eigentlichen Analyse (Tabelle 71) können für Zugang, Beschaffung und Transformation anfallen. *Zugangskosten* sind für Zugriff (z.B. Datenbankzugangsgebühren für Auskunfteien), Erhebung und Erfassung (z.B. bei eigens durchgeführten Befragungen) anzusetzende Entgelte. *Beschaffungskosten* umfassen die Aufwendungen für Extraktion, Transport und Übermittlung. Da der Großteil der Daten vor der Analyse einer Vorbereitung zu unterziehen ist, fallen meist *Transformationskosten* an. Für die Beurteilung von Datenquellen sollten hier nur jene Transformationen berücksichtigt werden, die spezifisch für einen Datenbestand sind, da die Einbeziehung analysespezifischer Datenmodifikationen nicht den Datenquellen anzulasten ist.

A5.4 Datencharakteristika

Typischerweise werden die eingesetzten Datencharakteristika nach ihrer methodischen Herkunft gegliedert, etwa in statistische Maße für numerische und informationstheoretische Maße für symbolische Attribute (vgl. z.B. [ThLi98, 2f.]). Aus Anwendungssicht erscheint jedoch eine Klassifizierung nach den beschriebenen Eigenschaften der Daten sinnvoller, die im Folgenden entwickelt wird. Die folgenden Ausführungen stützen sich auf die in Abschnitt 5.5.3.2 genannte Literatur sowie auf [Pyle99, 127ff], [Enge99, 150ff], [Lind05, 121ff]. Optionen zur Berechnung der Eigenschaften werden exemplarisch genannt. Sofern nicht anders angegeben, beziehen sich die Charakteristika jeweils auf Attribute.

Schema- und Rollencharakteristika der Datenelemente lassen sich größtenteils aus dem Data Dictionary entnehmen. Außer dem *Datentyp* ist hierbei insbesondere das realisierte *Skalenniveau* von Bedeutung, da etwa der Datentyp `Integer` gleichermaßen nominale (z.B. Identifikationsmerkmal oder Klassenzugehörigkeit), ordinale (z.B. Bewertung oder Rangfolge) und metrische Skalen (z.B. Mengenangaben) repräsentieren kann. Für die Modellerstellung ist die Auszeichnung von Zielvariablen relevant, um die *Rolle* einzelner Attribute bei der Analyse zu markieren. Die Kenntnis der *Defaultwerte* kann verhindern, dass häufig auftretende Vorgabewerte als Substitute fehlender Attributwerte irrtümlich als Muster erkannt werden [Pyle99, 134].

Mengencharakteristika beschreiben die *Anzahl der Attribute* des Datenbestands, gegliedert in die Gesamtzahl und die *Anzahl numerischer* bzw. *symbolischer Attribute*, sowie die *Anzahl der Datensätze*. In Bezug auf die Attributrollen sind die *Existenz von Zielvariablen* (boolsches Merkmal) sowie die *Anzahl der (unabhängigen) Eingabevariablen* und die *Anzahl der (abhängigen) Zielvariablen* von Interesse. Für Zielvariablen sind Informationen über die *Anzahl der Klassen* und die *Anzahl der Beispiele (Datensätze) für jede Klasse* zur Beurteilung der Komplexität der Modellerstellung bzw. der Balanciertheit der Daten hilfreich. Weiterhin kann die *Abdeckung der Default-Klasse* berechnet werden, die den Anteil der Datensätze in der häufigsten Klasse beziffert.[278] Hinsichtlich der Merkmalswerte sind Informationen zur *Existenz fehlender Werte* (boolesches Merkmal) für jedes Attribut wichtig, um die Datenqualität abzuschätzen. Sie ist durch die *Anzahl fehlender Werte* und den *Anteil fehlender Werte* an der Gesamtzahl der Datensätze zu ergänzen. Für symbolische Attribute erlaubt die *Anzahl verschiedener Werte* durch Vergleich mit dem Wertebereich Rückschlüsse auf Datenmängel [Pyle99, 142].[279] Für

[278] Sie wird auch als Klassifikationsgenauigkeit des „Default-Algorithmus“ interpretiert. Hierunter wird die triviale Zuordnung aller Beispieldatensätze zur Default-Klasse verstanden [Lind05, 171].

[279] Datenmängel sind zu erwarten, wenn die *Anzahl verschiedener Werte* > n +1, wenn n die Anzahl zulässiger (definierter) Werte ausdrückt und eine Ausprägung für fehlende Werte berücksichtigt wird [Pyle99, 142].

unstrukturierte Medientypen können bei Bedarf andere Maße Anwendung finden, z.B. die Wortanzahl für Textdokumente.

Verteilungscharakteristika bilden in erster Linie Lage- und Streuungsmaße. Lagemaße für metrische Skalen sind *Minimal-* und *Maximalwerte* (Wertebereich), *arithmetisches Mittel*, *Median* und *empirische Quantile*. Streuungsmaße umfassen *Standardabweichung*, *Medianabweichung* und *Quartilsabstand*. Zur Erkennung von Extremwerten ist ein Vergleich zwischen hierfür anfälligen Maßen mit robusten Kenngrößen (z.B. Median, Quartilsabstand) geeignet. Beispielsweise verweisen große Unterschiede zwischen arithmetischem Mittel und Median auf mögliche Ausreißer [EnTh98, 433]. Auf ihrer Basis sind heuristische *Extremwertindikatoren* berechenbar. Beispiele hierfür finden sich bei [EnTh98b, 45f] und [Enge99, 166]. Weitere Hinweise auf Ausreißer können *Schiefe* und *Wölbung* vermitteln, die über die Gleichmäßigkeit der Merkmalsverteilung Aufschluss geben. Sie werden häufig als Indikatoren für das Vorliegen einer *Normalverteilung* verwendet, zuverlässiger und zudem auch für multivariate Verteilungen geeignet sind aber häufig spezifische Testgrößen [Enge99, 155f.]. Verteilungseigenschaften metrischer Merkmale lassen sich aus Boxplots ablesen. Der *Modus* als Lagemaß für kategoriale Skalen sollte durch eine Häufigkeitsverteilung ersetzt werden. Weitere Angaben zur Streuung liefert die Entropie. Die *Attributentropie* misst den Informationsgehalt eines Attributs und ist umso kleiner, je zufälliger dessen Werte auftreten [Enge99, 160f.]. Bezogen auf Zielvariablen beschreibt sie als *Klassenentropie* die Anzahl binärer Fragen, die zur Trennung der Klassen erforderlich sind [Enge99, 161].

Zusammenhangscharakteristika dienen zur Beurteilung von Beziehungen, Einflussstärke und Redundanz von Variablen. Bei metrisch skalierten Attributen eignet sich hierfür der *Korrelationskoeffizient*, der jeweils für ein Merkmalspaar zu bestimmen ist (Korrelationsmatrix) [EnTh98b, 46]. Mehr Aussagegehalt birgt häufig der *multiple Korrelationskoeffizient*, der den Zusammenhang einer Einzelvariablen mit der restlichen Variablenmenge charakterisiert und im Falle vollständig linearer Abhängigkeit die Redundanz des betreffenden Attributs signalisiert [EnTh98, 431, 433]. Berechnet für die Zielvariable bemessen die

Koeffizienten das Vorhersagepotenzial der referenzierten Eingabevariablen. Der kanonische Korrelationskoeffizient misst Abhängigkeiten zwischen zwei Variablenmengen. Analog kann für kategoriale Attribute paarweise die *Synentropie* (Transinformation, mutual information) und für eine Attributmenge in Bezug auf eine Zielvariable die *durchschnittliche Synentropie* (average mutual information) berechnet werden [Enge99, 161f.]. Alternativen zu diesen Maßen sind z.B. Gini-Index, Information Gain, Relevanzmaß und g-Funktion, die in Studien stets zu ähnlichen Aussagen führten [EnTh98, 431], [Enge99, 163]. Ebenso ist auch die Induktion von Assoziationsregeln denkbar, um Abhängigkeiten zwischen symbolischen Attributen aufzudecken [EnTh98b, 46]. Um das Attributgefüge im Ganzen zu charakterisieren eignen sich aggregierte Kenngrößen, etwa der *durchschnittliche (multiple) Korrelationskoeffizient* (vgl. [HND+11, 11]), die *durchschnittliche Attributentropie* und die *durchschnittliche Synentropie* (jeweils als arithmetisches Mittel der Basismaße) [Enge99, 162].

Weitere **spezifische Charakteristika** können in Abhängigkeit von der Analyseaufgabe oder von zu testenden Verfahrensannahmen berechnet werden. ENGELS & THEUSINGER [EnTh98], [EnTh98b] untersuchen z.B. die Anwendung der Diskriminanzanalyse zur Bewertung der *Komplexität des numerischen Datenraums* für Klassifikationsaufgaben. In ähnlicher Weise nutzen HILARIO ET AL. [HDN+11, 11f.] geometrische Maße zur Charakterisierung der Raumkomplexität. Ein Beispiel für die Prüfung von Verfahrensannahmen ist der Boxsche M-Test, der die Klassen (Zielvariable) auf homogene Kovarianzen bezüglich der Eingabevariablen untersucht [EnTh98, 432]. PYLE [Pyle99, 72] verwendet Linearitätsmaß und Ankunftsrate, um Variablen auf Monotonie zu prüfen. LINDNER [Lind05, 131f.] berechnet auf Basis der Entropie einen Quotienten zur Charakterisierung der äquivalenten Attributanzahl und einen Rauschfaktor (Signal-Rausch-Abstand), die jeweils auf irrelevante Attribute in den Eingabedaten hinweisen.

A6 Prüfung von Abhängigkeiten zwischen Prozessbausteinen

Tabelle 72 zeigt Bedingungen für die Übereinstimmung von Flussbeziehungen zwischen Prozessbausteinen nach [Hein+08, 452ff.]. Sie dienen zur Reihenfolgeplanung im Rahmen von Schritt P2.3 (Abschnitt 5.5.4.3) und werden in zwei Stufen I und II geprüft. Sie betreffen Ein- und Ausgabeflüsse OUT_A und IN_B von Prozessbausteinen A, B sowie zugehörige Restriktionen r.

I. Analysiere semantische Relationen	**II. Analysiere Restriktionen**	**Übereinstimmung**
[1] $OUT_A = IN_B \vee OUT_A \equiv IN_B \vee$ $OUT_A \sqsubseteq IN_B \vee OUT_A \succ IN_B$	[1.1] $r_{OUT_A} \subseteq r_{IN_B}$	vollständig
	[1.2] $r_{OUT_A} \cap r_{IN_B} \neq \emptyset \wedge$ $r_{OUT_A} \setminus r_{IN_B} \neq \emptyset$	partiell
	[1.3] $r_{OUT_A} \cap r_{IN_B} = \emptyset$	keine
[2] $OUT_A \sqsupseteq IN_B$	[2.1] $r_{OUT_A} \cap r_{IN_B} \neq \emptyset$	partiell
	[2.2] $r_{OUT_A} \cap r_{IN_B} = \emptyset$	keine
[3] $OUT_A \neq IN_B \wedge OUT_A \not\equiv IN_B \wedge$ $OUT_A \prec IN_B \wedge OUT_A \sqsubseteq IN_B \wedge OUT_A \sqsupseteq IN_B$	–	keine
Legende der semantischen Relationen: $=$ Gleichheit, $\neq$ Ungleichheit; $\equiv$ Äquivalenz, $\not\equiv$ Nicht-Äquivalenz; $\sqsubseteq$ Spezialisierung, $\sqsupseteq$ Generalisierung; $\succ$ Aggregation, $\prec$ Zerlegung		

Tabelle 72.: Bedingungen für die Übereinstimmung des Flusspaars (OUT_A, IN_B) nach [Hein+08, 453]

A7 Spezifische Kriterien zur Beurteilung von Analyseergebnissen

Zum besseren Verständnis sowie zur Vertiefung sind in diesem Abschnitt einige Kriterien zur Beurteilung von Analyseergebnissen dargestellt, die in Abschnitt 7.2.1 referenziert werden. Zusätzlich wird der Vorschlag eines multidimensionalen Datenschemas zum Aufbau eines Performance Measurements für Datenanalyseprozesse gezeigt (vgl. Abschnitt 7.2.2.4).

A7.1 Bewertungskriterien für Ergebnisse konfirmatorischer Analysen

Auswahl parametrischer Hypothesentestverfahren (metrische Merkmale)

Kenngrößen	Abweichung von einer Konstanten	Unterschied zwischen Bedingungen/Stufen	intraindividuelle Veränderung
Einzelwerte	z-Test	--	kritische Differenz (D_k)
Mittelwerte	Ein-Gruppen-t-Test	t-Test für unabhängige Stichproben (bei k = 2 Stufen); einfaktorielle Varianzanalyse (bei k ≥ 2 Stufen)	t-Test für abhängige Stichproben (bei k = 2 Messzeitpunkten); Messwiederholte Varianzanalyse (bei k ≥ 2 Messzeitpunkten)
Häufigkeiten	Binomialtest	χ2-Test	McNemar-Test

Tabelle 73: Klassifikation parametrischer Hypothesentestverfahren in Abhängigkeit (a) von der statistischen Kenngröße und (b) von der Fragestellung [GoJä09, 193]

Fehler 1. und 2. Art bei der Entscheidung über die Nullhypothese

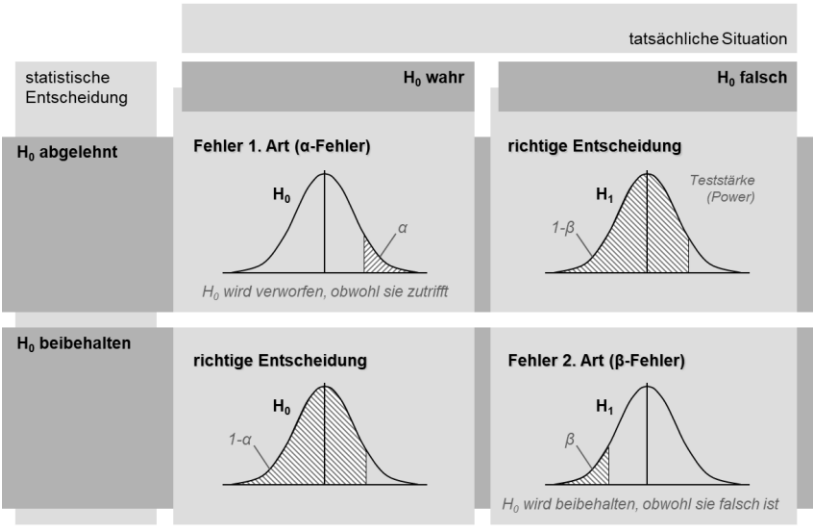


Abbildung 122: Fehler 1. und 2. Art (konfirmatorische Analysen) (vgl. [GoJä09, 198], [Zöfe03, 95])

A7.2 Bewertungskriterien für Ergebnisse schließender Analysen

Klassifikatoren

Die *Klassifikationstabelle* (Confusion Matrix) ist eine Kontingenztabelle, welche die prognostizierten Klassen den wahren Klassen gegenüberstellt und die Anzahl korrekter und fehlerhafter Vorhersagen ausweist, wie sie bei Anwendung des Modells auf Evaluationsdaten mit bekannten Zielmerkmalswerten ermittelt werden. Die Werte auf der Hauptdiagonalen zeigen die richtigen, jene auf der anderen Diagonale die falschen Zuordnungen [BoAr01, 228-230], [Fawc06, 862].

Im Zwei-Klassen-Fall können vier mögliche Ergebnisse beobachtet werden (Abbildung 123), deren Bezeichnungen die Perspektive des Modells wiedergeben: *True Negatives (TN)* und *True Positives (TP)* kennzeichnen korrekte Klassifikationen. *False Negatives (FN)* sind tatsächlich positive Objekte, die irrtümlich als negativ eingeordnet wurden, und

False Positives (FP) entsprechend tatsächlich negative Objekte, die fehlerhaft als positiv markiert sind[280] [WiFr00, 138]. Die Zeilensummen geben mit *Predicted Negatives (PN)* die insgesamt vom Modell als negativ und mit *Predicted Positives (PP)* die positiv klassifizierten Fälle an, während die Spaltensummen die tatsächlichen *Negatives (N)* und die tatsächlichen *Positives (P)* ausweisen. *K* bezeichnet die Zahl der Datensätze im Evaluierungsset. Gute Modelle sind im Allgemeinen durch hohe Werte auf der Hauptdiagonalen charakterisiert [WiFr00, 138].

	tatsächliche Situation		
Vorhersage des Modells	**0 / negativ** **NEIN**	**1 / positiv** **JA**	*Summen*
0 / negativ **NEIN**	**True Negatives** **TN**	False Negatives FN	*Predicted Negatives* *PN*
1 / positiv **JA**	False Positives FP	**True Positives** **TP**	*Predicted Positives* *PP*
Summen	*Negatives* *N*	*Positives* *P*	*Anzahl Datensätze* *K*

Abbildung 123: Klassifikationstabelle (eigene Darstellung; vgl. [WiFr00, 138], [BoAr01, 229], [Fawc06, 862])

[280] Welche Ausprägung des Klassenmerkmals als positiv bzw. negativ gilt, hängt von der Anwendung ab; typischerweise werden Positive als „Treffer" interpretiert. Im Beispiel wurden Positive mit 1 („Ja"), Negative mit 0 („Nein") gekennzeichnet. Dies entspricht z.B. der Sprechweise bei medizinischen Diagnosen, wo ein positives Ergebnis auf eine Erkrankung hinweist. Auch bei der Kreditwürdigkeitsprüfung ist ein Treffer regelmäßig ein Kreditausfall; hier ist positiv also nicht mit „gut" gleichzusetzen. Im Direktmarketing hingegen ist ein Treffer ein potenziell positiv reagierender Kunde, was als „gut" anzusehen ist.

In Tabelle 74 auf der folgenden Seite sind gängige Kenngrößen mit ihrer Definition aufgeführt, die aus der Klassifikationstabelle direkt ableitbar sind. Die für das Gesamtmodell gültigen Größen sind die bereits erläuterte *Erfolgsrate* und ihr Komplement, die *Fehlerrate*. Die globale Erfolgsrate wird zuweilen auch als Trefferrate bezeichnet (vgl. z.B. [BoAr01, 230]), typischerweise steht dieser Begriff aber für die *True-Positive-Rate*, die auch als Sensitivität und im Information Retrieval als Recall bekannt ist. Sie ist als Wahrscheinlichkeit einer korrekten Zuordnung der Positiven [GoJä09, 115] die häufig dominante Zielgröße (klassenspezifisches Maß). Die Fehlalarm-Rate ist der Anteil der irrtümlich positiv Klassifizierten aller tatsächlich Negativen (*False-Positive-Rate*). Sie stellt das Komplement der auch als Spezifität bekannten *True-Negative-Rate dar*. Die dadurch repräsentierte Quote der korrekt erkannten Negativen steht in Konkurrenz zur Sensitivität [BoAr01, 230]. Die *False-Negative-Rate* zeigt die Versagensquote bei der Erkennung Positiver an und ist in bestimmten Anwendungen wie etwa der Betrugserkennung von großer Bedeutung. Sie wird häufig direkt als Komplement der Trefferrate („1 – Sensitivität“) notiert. Als *Precision* wird die Treffgenauigkeit gemessen, mit der ein positiv Markierter tatsächlich positiv ist. Der Begriff stammt aus dem Information Retrieval und drückt insbesondere die Relevanz eines abgerufenen Dokuments aus. Im Direktmarketing entspricht sie der zu optimierenden Responsequote [BoAr01, 230]. Ihr steht mit der *Segreganz* die Trennfähigkeit gegenüber, mit der ein Negativer auch als solcher erkannt wird.

Da oft mehrere Kriterien in Konkurrenz stehen oder gleichzeitig zu optimieren sind, stehen auch Maße zur Verfügung, die mehrere Einzelgrößen kombinieren. Ein Beispiel aus dem Information Retrieval ist das so genannte *F-Maß*, das Präzision und Sensitivität über ein gewichtetes harmonisches Mittel vereint [CaNi04, 70], [LeVo10, 87].

Kenngröße	Synonyme	Definition	Beschreibung
Erfolgsrate	Genauigkeit (Accuracy) [c] (zuweilen auch: *Trefferrate* [b])	(TP + TN) / K [a] 1 – Fehlerrate [a]	Anteil insgesamt korrekt Klassifizierter
Fehlerrate		(FN + FP) / K [b] 1 – Erfolgsrate	Anteil insgesamt falsch Klassifizierter
True-Positive-Rate	TP-Rate; Trefferrate (Hit Rate); [c] **Sensitivität**; Recall	TP / P [c]	Anteil korrekt (positiv) Klassifizierter an allen Positiven; Empfindlichkeit (Vollständigkeit) bzgl. der Positiven
False-Positive-Rate	FP-Rate; Fehlalarm-Rate; Ausfallrate (Fallout); **1 – Spezifität**	FP / N [c] 1 – Spezifität [b]	Anteil falsch (positiv) Klassifizierter an allen Negativen
True-Negative-Rate	TN-Rate; **Spezifität**	TN / N 1 – FP-Rate [c]	Anteil korrekt (negativ) Klassifizierter an allen Negativen; Empfindlichkeit (Vollständigkeit) bzgl. der Negativen
False-Negative-Rate	FN-Rate; Versagensrate (Miss Rate); **1 – Sensitivität**	FN / P 1 – Sensitivität [b]	Anteil falsch (negativ) Klassifizierter an allen Positiven

(Fortsetzung auf nächster Seite)

Präzision (Precision)	Positive Predictive Value; Relevanz; Selektionsfähigkeit	TP / PP TP / (TP + FP) [c]	Anteil korrekt (positiv) Klassifizierter an allen positiv Klassifizierten; Akzeptanz-/ Treffgenauigkeit
Segreganz	Negative Predictive Value; Trennfähigkeit	TN / PN	Anteil korrekt (negativ) Klassifizierter an allen negativ Klassifizierten; Zurückweisungs-genauigkeit
F-Maß	F-Score	2 / (1/Precision + 1/Recall) [c] 2 TP / (2 TP + P)	gewichtetes harmonisches Mittel aus Präzision und Trefferquote (Recall)
	Quellen: a [WiFr00, 147] b [BoAr01, 230] c [Fawc06, 862]		

Tabelle 74: Ausgewählte Kenngrößen für Klassifikationsmodelle (Quellen: [WiFr00, 147], [BoAr01, 230], [Fawc06, 862], [NeKn15, 331])

Schätzer

Gängige Fehlermaße zur Beurteilung der Genauigkeit von Schätzern zeigt Tabelle 75. Der *Mean Error* als Trivialmaß ist problematisch, wenn sich die Vorhersagegenauigkeit in verschiedenen Regionen des Datenraums unterschiedlich verhält[281] oder wenn Gefahr besteht, dass sich negative und positive Abweichungen betragsmäßig ausgleichen [BeLi97, 99f.]. Er ist daher nur sehr eingeschränkt zu empfehlen. Seine Mängel umgeht der häufig eingesetzte *Mean Squared Error (MSE)*, der die durchschnittliche quadratische Abweichung bemisst. Er kann leicht durch Wurzelziehung in ein skalenerhaltendes Maß transformiert werden (*Root Mean Squared Error, RMSE*). MSE und RMSE sind üblich für Kausalprognosen und finden z.B. für Nachfrageprognosen Ver-

[281] Beispielsweise werden Umsatzprognosen in Abhängigkeit vom Preis mit steigender Preiselastizität zunehmend fehlerhaft [BeLi97, 99].

wendung [Thon05, 46]. Die *Mean Absolute Deviation (MAD)* nutzt absolute Abweichungen und eignet sich besonders in Fällen, in denen sich die Kosten von Fehlprognosen proportional zum Prognosefehler verhalten [Thon05, 47]. Zum Vergleich der Güte von Modellen mit verschiedenen Skalen ist eine Normierung des Prognosefehlers an den Werten erforderlich, wie sie der *Mean Absolute Percentage Error (MAPE)* vornimmt. Er beschreibt die mittlere prozentuale Abweichung des Prognosewerte von den Ist-Werten [KüBe01, 289], [Thon05, 47].

Fehlermaß	**Berechnung**	**Eigenschaften**	**Anwendung**
Prognosefehler e_k	Prognosewert f_k – tatsächlicher Wert y_k		Einzelwert *k*
			Gesamtmodell ↓
Mean Error ME	$\Sigma_{k=1..K}\ e_k$ / K	skalenerhaltend; Ausgleich pos./neg. Fehler	nur in besonderen Fällen empfehlenswert
Mean Squared Error MSE	$\Sigma_{k=1..K}\ e_k^2$ / K	nicht skalenerhaltend; starke Wichtung großer Abweichungen	meistverbreitetes Maß
Root Mean Squared Error RMSE	√ MSE	skalenerhaltend; starke Wichtung großer Abweichungen	
Mean Absolute Deviation MAD	$\Sigma_{k=1..K}\ \lvert e_k\rvert$ / K	skalenerhaltend; intuitiv verständlich; gleiche Wichtung aller Abweichungen	Kosten proportional zur Höhe des Prognosefehlers
Mean Absolute Percentage Error MAPE	$\Sigma_{k=1..K}\ \lvert e_k / y_k\rvert$ / K * 100	relative prozentuale Abweichung vom tatsächlichen Wert	Vergleich mehrerer Modelle mit versch. Skalen

Tabelle 75: Ausgewählte Fehlermaße für Schätzmodelle (Quellen: [WiFr00, 147f.], [KüBe01, 289f.], [Thon05, 72ff.])

A7.3 Diagrammtypen zur Auswahl von Prognosemodellen bei Zielkonflikt

Diagrammtyp	Achsen	Domäne / Anwendung	Eigenschaften
Lift-Diagramm	y: Lift x: Support	allgemein anwendbar	direkte Visualisierung des Lift im Diagramm, keine Darstellung eines Zielkonflikts mit einer konkurrierenden Größe
Konzentrations-diagramm (Lorenz-/Gini-Kurve)	y: Trefferrate x: Support	Auswahlprobleme; z.B. Direkt-marketing	invariant ggü. monotonen Transformationen der Klassenwahrscheinlichkeiten, unabhängig von Kalibrierung [KrBü11, 46]
ROC-Kurve	y: Trefferrate x: Fehlalarmrate	v.a. Diagnose-probleme	robust ggü. Änderungen von Klassenverteilung und Fehlerkosten [Fawc06, 865f.]; geeignet für Anwendungen mit schiefen Klassenverteilungen und ungleichen Klassifikationsfehlerkosten [Fawc06, 861]
Recall-Precision-Kurve	y: Trefferrate (Recall) x: Präzision	Information Retrieval	sensitiv ggü. Änderungen der Klassenverteilung [Fawc06, 865f.]

Tabelle 76: Gängige Diagrammtypen zur Evaluierung von Klassifikatoren sowie zur Berücksichtigung konkurrierender Zielgrößen (vgl. [WiFr00, 146])

A7.4 Dimensionsschema zur Auswertung von Prozessparametern

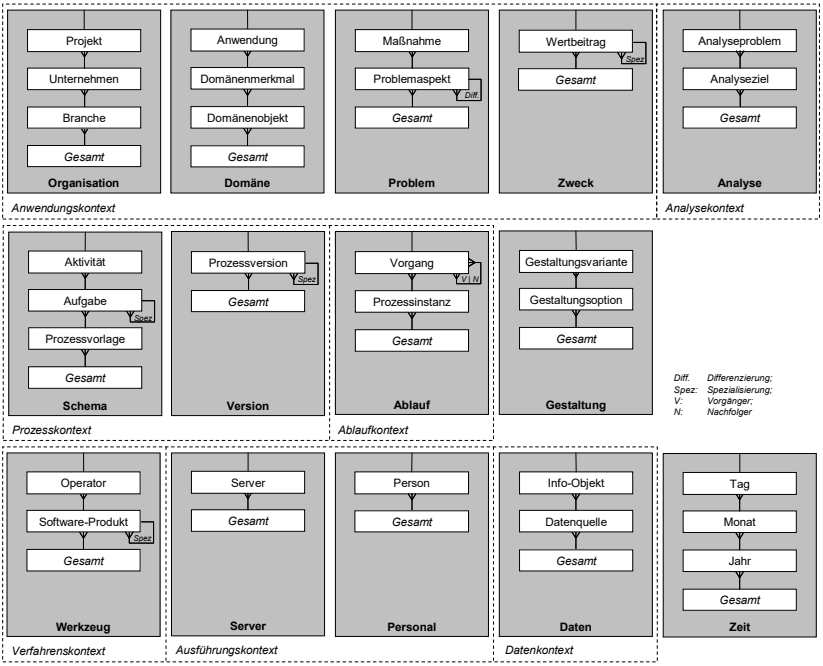


Abbildung 124: Dimensionssicht einer multidimensionalen Datenstruktur zur Analyse von Prozesskennzahlen (eigene Darstellung)

Die Dimensionen zur Aufschlüsselung der Prozessparameter im Zuge der Beurteilung des Prozessablaufs (K2, Abschnitt 7.2.2) sollten die wesentlichen Bestimmungsfaktoren des Analyseprozesses bzw. seiner Elemente umfassen. Sie können aus dem Metamodell abgeleitet und um weitere Aspekte ergänzt werden [EdOG02, 6], [Benk16]. Wichtige Dimensionen neben der Zeit resultieren aus Anwendungs-, Analyse-, Prozess-, Ablauf-, Verfahrens-, Ausführungs- und Datenkontext (vgl. [Mueh01]), die sich unmittelbar aus der Datenanalysearchitektur ergeben. Der Anwendungskontext enthält auch Organisationsmerkmale zu Projekt und Unternehmen. Die Differenzierung zwischen Prozesskontext (Typisierung und Zerlegung von Prozesselementen) und Ablaufkontext (Vorgänger-Nachfolger-Beziehungen auf Instanzebene) erlaubt

die detaillierte Analyse der Abhängigkeiten der Leistungsparameter von der Ausgestaltung des Prozessgefüges [Benk16, 227f.]. Zusätzlich ist die Prozessversion auswertbar. Flankierend werden Informationen zur Gestaltungsoption mitgeführt, d.h., ob der Prozess durch Innovation oder durch Wiederverwendung entstandenen ist und welche Variante (top-down/bottom-up) dabei gewählt wurde. Die Ressourcenebene ist durch Verfahrenskontext (Werkzeug), Ausführungskontext (Server und Personal als Aufgabenträgerinstanzen) und Datenkontext repräsentiert. Abbildung 124 zeigt einen Vorschlag für die Dimensionssicht einer multidimensionalen Datenstruktur in Notation des Semantischen Data-Warehouse-Modells (SDWM) [Böhn01, 257ff.].

Mögliche, über die einfache Abfrage von Prozesskennzahlen hinausgehende Analysefragen auf Grundlage dieser Datenbasis umfassen unter anderem die folgenden Aspekte:

- Wie unterscheiden sich Prozesskennzahlen in Abhängigkeit von variierenden Ausprägungen bestimmter Kontextfaktoren?
- Welche Aktivitäten führen (regelmäßig) zu Problemen?
- Welche Probleme treten im Zusammenhang mit welchen Kontextfaktoren auf?
- Wie unterscheiden sich Prozesskennzahlen eines Vorgangs in Abhängigkeit von den vor- oder nachgelagerten Vorgängen im Prozessablauf (Instanzebene)?
- Welche Ressourcen/Personen sind an problematischen Abläufen beteiligt?
- Die Beteiligung welcher Personen führt zu überdurchschnittlich hohen unproduktiven Zeitanteilen?
- Welche Methodenparameter haben zu zufriedenstellenden, welche zu nicht akzeptablen Ergebnissen (Ergebnis-Gütemaße) geführt?
- Welche Leistungsverbesserungen wurden durch modifizierte Prozesse gegenüber dem Ursprungsprozess erreicht?
- Welche Version eines Prozesses führt zu den besten Prozesskennzahlen?

A8 Aufgaben des Handlungsschemas der Methodik

Die folgende Aufstellung zeigt die hierarchische Gliederung der Aufgaben der vereinten Handlungsschemata der Managementmethodik für Datenanalyseprozesse gemäß Kapitel 5-7.

Sie sind mit jenen des Vorgehensmodells (vgl. Anhang A3) kompatibel, jedoch weder identisch noch vollständig deren Aufgaben eindeutig zuordenbar. So erfolgt die Prozessspezifikation P je nach gewähltem Steuerungsmodus entweder im Rahmen der präskriptiven Planung (Phase V1 des Vorgehensmodells) oder simultan zur Ablaufgestaltung S2 (V2). Für die Anwendung des Wissens (V3) sieht das Handlungsschema keine Detaillierung vor; die Maßnahmenplanung aus dem Vorgehensmodell (V3.2) kann jedoch mit der Ableitung von Handlungsoptionen (Z2.4) überlappen. Das Handlungsschema enthält nur Aufgaben, die unmittelbar die Datenanalyse bzw. ihre Ergebnisse und Konsequenzen betreffen, das Vorgehensmodell benennt zusätzlich rein maßnahmenbezogene Aufgaben.

Planung von Datenanalyseprozessen

vgl. „V1: Planung des Analyseprojekts" des Vorgehensmodells

Z: Problemspezifikation

- Z1. Identifikation des Sachproblems
  - Z1.1: Problemerkennung
  - Z1.2: Diskurswelttabgrenzung
  - Z1.3: Problembeschreibung
- Z2. Domänenanalyse
  - Z2.1: Ergründung der Sichtweise des Auftraggebers
  - Z2.2: Konkretisierung des Problemobjekts
  - Z2.3: Identifikation von Einflussfaktoren

Z2.4: Ableitung von Handlungsoptionen

Z2.5: Problemkartierung

Z3: Spezifikation des Analyseproblems

Z3.1: Formulierung des Analyseziels

Z3.2: Formulierung des Analyseproblems

Z3.3: Konkretisierung und Strukturierung von Analysezielen

Z4: Untersuchungsdesign

Z4.1: Methodische Überlegungen zum Untersuchungsgang

Z4.2: Konzipierung des Untersuchungsgangs

Z4.3: Konzipierung von Einzelanalysen

→ *für jeden Analyseprozess: Prozessspezifikation P*

Z5: Projektplanung

→ *begleitend zu allen anderen Aufgaben der Problemspezifikation*

Z5.1: Ressourcenplanung

Z5.2: Zeitplanung

Z5.3: Budgetplanung

Z5.4: Organisationsgestaltung

P: Prozessspezifikation

P1: Planung der Datenanalysephase

P1.1: Spezifikation der Analyseaufgabe

P1.2: Charakterisierung der Analysedaten

P1.3: Bestimmung einer Verfahrensklasse

P1.4: Auswahl eines Analyseverfahrens

P1.5: Kontextabhängige Entwurfsentscheidungen

P2: Planung der Datenvorbereitungsphase

P2.1: Spezifikation der Datentransformationsaufgaben

P2.2: Zuordnung von Transformationsverfahren

P2.3: Reihenfolgeplanung

P3: Planung der Ergebnisaufbereitungsphase

P3.1: Spezifikation der Ergebnisaufbereitungsaufgaben

P3.2: Zuordnung von Transformationsverfahren

P3.3: Reihenfolgeplanung

P4: Instanziierung von Verfahrensparametern

P4.1: Makroparametrisierung

P4.2: Mikroparametrisierung

S: Steuerung von Datenanalyseprozessen

während „V2: Durchführung der Analyse" des Vorgehensmodells

S1: Ablaufinstanziierung

S2: Ablaufgestaltung

→ *gemäß den Steuerungsmodi Repetition, Deviation, Innovation*

S3: Ablaufbegleitung (Prozesssteuerung i.e.S.)

S3.1: Vorgangsauslösung

S3.2: Koordination

S3.3: Ablaufüberwachung

S4: Protokollierung und Dokumentation

K: Revision von Datenanalyseprozessen

vgl. „V4: Evaluierung des Analyseprojekts" des Vorgehensmodells

K1: Beurteilung der Analyseergebnisse

K1.1: Bewertung der Gültigkeit von Analyseergebnissen

K1.2: Interpretation von Analyseergebnissen

nach den Kriterien Verständlichkeit, Neuartigkeit, Nützlichkeit

K2: Beurteilung des Prozessablaufs

K2.1: Beurteilung der Effektivität

K2.2: Beurteilung der Effizienz

K2.3: Beurteilung der Struktur

K3: Evaluation der Handlungsmaßnahmen

→ *Evaluationsstudien, ggf. unter Einsatz von Datenanalysen*

K4: Nutzen-Kosten-Analyse

K4.1: Ermittlung der Kosten

K4.2: Quantifizierung des Nutzens

K4.3: Effizienzanalyse

K5: Modifikation der Analysepläne

K5.1: Modifikationen auf Prozessebene

K5.2: Modifikationen auf Ziel- und Ressourcenebene

K6: Extraktion wiederverwendbaren Wissens

K6.1: Dokumentation von Kommentaren und Bewertungen

K6.2: Ableitung von Kontextregeln

K6.3: Identifizierung und Speicherung von Prozessartefakten

K6.4: Wartung der Fallbibliothek

Literaturverzeichnis

AAD+04 Acker, H.; Atkinson, C.; Dadam, P.; Rinderle, S.; Reichert, M.: **Aspekte der komponentenorientierten Entwicklung adaptiver prozessorientierter Unternehmenssoftware**. In: Turowski, K. (Hrsg.): Architekturen, Komponenten, Anwendungen. Proc. 1. Verbundtagung AKA 2004, Augsburg, Dezember 2004. LNI P-57, 2004, S. 7-24. URL: http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/AADR04.pdf (Abruf am 28.01.2006).

AaDO00 van der Aalst, W.; Desel, J.; Oberweis, A. (Hrsg.): **Business Process Management. Models, Techniques, and Empirical Studies**. Berlin (Springer) 2000.

AaWe04 van der Aalst, W.M.P.; Weijters, A.J.M.M.: **Process mining: a research agenda**. In: Computers in Industry 53 (2004) 3, S. 231-244.

ABD+02 Althoff, K.-D.; Becker-Kornstaedt, U.; Decker, B.; Klotz, A.; Leopold, E.; Rech, J.; Voss, A.: **The indiGo Project: Enhancement of Experience Management and Process Learning with Moderated Discourses**. In: [Pern02], S. 53-79.

ABEM15 Apel, D.; Behme, W.; Eberlein, R.; Merighi, C.: **Datenqualität erfolgreich steuern. Praxislösungen für Business-Intelligence-Projekte**. 3. Aufl., Heidelberg (dpunkt) 2015.

AcUA12 Accorsi, R.; Ullrich, M.; van der Aalst, W.M.P.: **Process Mining**. Aktuelles Schlagwort. In: Informatik-Spektrum 35 (2012) 5, S. 354-359.

ADH+03 van der Aalst, W.M.P.; van Dongen, B.F.; Herbst, J.; Maruster, L.; Schimm, G.; Weijters, A.J.M.M.: **Workflow mining: A survey of issues and approaches**. In: Data & Knowledge Engineering 47 (2003), S. 237-267.

AdTu97 Adomavicius, G.; Tuzhilin, A.: **Discovery of Actionable Patterns in Databases: The Action Hierarchy Approach**. In: [HMPU97], S. 111-114.

AdZa96 Adriaans, P.; Zantinge, D.: **Data Mining**. Harlow (Addison-Wesley) 1996.

AlGR12 Allen, H.; Gearan, P.; Rexer, K.: **5th Annual Data Miner Survey. 2011 Survey Summary Report**. Winchester, USA (Rexer Analytics) 2012. Zusatzmaterial verfügbar unter http://www.rexeranalytics.com/DMSurvey2011_MeasuringSuccess (Abruf am 14.09. 2016).

AlWa97 Ali, F.Ö.G.; Wallace, W.A.: **Bridging the gap between business objectives and parameters of data mining algorithms**. In: Decision Support Systems 21 (1997), S. 3-15.

AmCo94 Amant, R.S.; Cohen, P.R.: **Toward the Integration of Exploration and Modeling in a Planning Framework**. In: [FaUt94], S. 49-59.

AmCo94b Amant, R.S.; Cohen, P.R.: **A Planning Representation for Automated Exploratory Data Analysis**. In: Buntine, W.; Fisher, D.H. (Hrsg.): Proc. Knowledge-Based Artificial Intelligence Systems in Aerospace and Industry, 5.-6. April 1994, Orlando (Florida). Bellingham (SPIE – The International Society for Optical Engineering) 1994, S. 44-52.

AmCo98 Amant, R.S.; Cohen, P.R.: **Interaction with a Mixed-Initiative System for Exploratory Data Analysis**. In: Knowledge-Based Systems 10 (1998) 5, S. 265-273.

AmCo98b Amant, R.S.; Cohen, P.R.: **Intelligent Support for Exploratory Data Analysis**. In: The Journal of Computational and Graphical Statistics 7 (1998), S. 545-558.

Andr+81 Andrews, F.M.; Klem, L.; Davidson, T.N.; O'Malley, P.M.; Rodgers, W.L.: **A Guide for Selecting Statistical Techniques for Analyzing Social Science Data**. Second Edition. Survey Research Center, Institute for Social Research, University of Michigan. Ann Arbor (University of Michigan) 1981. Online-Version: http://www.microsiris.com/Statistical%20Decision%20Tree/ (Abruf am 08.04.2017).

AnGS05 Andresen, K.; Gronau, N.; Schmid, S.: **Ableitung von IT-Strategien durch Bestimmung der notwendigen Wandlungsfähigkeit von Informationssystemarchitekturen**. In: [FSEI05], S. 63-82.

AnHe85 Andrews, D.F.; Herzberg, A.M.: **Data. A Collection of Problems from Many Fields for the Student and Research Worker**. New York u.a. (Springer) 1985.

ARSB09 Adams, N.; Robardet, C.; Siebes, A.; Boulicaut, J.-F. (Hrsg.): **Advances in Intelligent Data Analysis VIII**. Proc. 8th International Symposium on Intelligent Data Analysis (IDA 2009), Lyon (Frankreich), August/September 2009. LNCS 5772. Berlin (Springer) 2009.

AyCG14 Ayankoya, K.; Calitz, A.; Greyling, J.: **Intrinsic Relations between Data Science, Big Data, Business Analytics and Datafication**. In: Proc. SAICSIT 2014, Centurion, South Africa, 29. September-1. Oktober 2014. New York (ACM) 2014, S. 192-198.

Baar+14 Baars, H.; Felden, C.; Gluchowski, P.; Hilbert, A.; Kemper, H.-G.; Olbrich, S.: **Gestaltung der nächsten Inkarnation von Business Intelligence. Flexibel gesteuerte Capability-Netzwerke zur Informationsintegration und -analyse**. In: Wirtschaftsinformatik 56 (2014) 1, S. 13-19.

BAB+01 Baragoin, C.; Andersen, C.M.; Bayerl, S.; Bent, G.; Lee, J.; Schommer, C.: **Mining Your Own Business in Telecoms – Using DB2 Intelligent Miner for Data**. Redbook, IBM International Technical Support Organization SG24-6273-00. San Jose (IBM Corp.) 2001.

BaCR94 Basili, V.R.; Caldiera, G.; Rombach, H.D.: **Experience Factory**. In: Marciniak, J. (Hrsg.): Encyclopedia of Software Engineering, Vol 1. New York (Wiley) 1994.

BaEc17 Bange, C.; Eckerson, W.: **BI und Datenmanagement in der Cloud. Treiber, Nutzen und Herausforderungen**. BARC-Anwenderstudie, Würzburg (Business Application Research Center – BARC GmbH) 2017.

BaGü04 Bauer, A.; Günzel, H. (Hrsg.): **Data-Warehouse-Systeme. Architektur, Entwicklung, Anwendung**. 2. Aufl., Heidelberg (dpunkt) 2004.

BaGü13 Bauer, A.; Günzel, H. (Hrsg.): **Data-Warehouse-Systeme. Architektur, Entwicklung, Anwendung**. 4. Aufl., Heidelberg (dpunkt) 2013.

Ball00 Ballensiefen, K.: **Informationsplanung im Rahmen der Konzeption von Executive Information Systems (EIS). Theoretische Analyse, empirische Untersuchungen und Entwicklung von Lösungsansätzen**. Lohmar (Josef Eul) 2000. Zugl.: Köln, Univ., Diss., 1999.

Balz09 Balzert, H.: **Lehrbuch der Software-Technik: Basiskonzepte und Requirements Engineering**. 3. Aufl., Heidelberg (Spektrum) 2009.

Balz96 Balzert, H.: **Lehrbuch der Software-Technik: Software-Entwicklung**. Heidelberg (Spektrum) 1996.

BaRi00 Bartlmae, K.; Riemenschneider, M.: **Case Based Reasoning for Knowledge Management in KDD-Projects. Concepts, Organizational Setting, Categorization into KM and Application in the case of Knowledge Discovery in Databases**. In: Riemer, U. (Hrsg.): Proc. Third Intl. Conference on Practical Aspects of Knowledge Management (PAKM2000), Basel, Schweiz, 30.-31. Oktober 2000. URL: http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-34/bartlmae_riemenschneider.pdf (Abruf am 14.10.2016).

BaRi11 Baeza-Yates, R.; Ribeiro-Neto, B. (2011): **Modern Information Retrieval. The Concepts and Technology behind Search**. 2. Aufl., New York (Addison Wesley) 2011.

Baro13 Baron, P.: **Big Data für IT-Entscheider. Riesige Datenmengen und moderne Technologien gewinnbringend nutzen**. München (Hanser) 2013.

BBFS11 Bartmann, D.; Bodendorf, F.; Ferstl, O.K.; Sinz, E.J.: **Merkmale, Systemarchitekturen und Management hochflexibler Geschäftsprozesse**. In: [SBBF11], S. 1-13.

Beck01 Becker, W.: **Planung, Entscheidung und Kontrolle**. Unter Mitarbeit von S. Seedorf. Bamberger Betriebswirtschaftliche Beiträge, Edition Unternehmensführung & Controlling (Lehrmaterialien). 2. Aufl., Bamberg (Otto-Friedrich-Universität) 2001.

Beck95 Becker, W.: **Planung, Entscheidung und Kontrolle**. Bamberger Betriebswirtschaftliche Beiträge, Edition Unternehmensführung & Controlling, Bamberg (Otto-Friedrich-Universität) 1995.

Beck96 Becker, W.: **Strategisches Management**. Bamberger Betriebswirtschaftliche Beiträge, Edition Unternehmensführung & Controlling. 3. Aufl., Bamberg (Otto-Friedrich-Universität) 1996.

BeDK04 Becker, J.; Delfmann, P.; Knackstedt, R.: **Konstruktion von Referenzmodellierungssprachen. Ein Ordnungsrahmen zur Spezifikation von Adaptionsmechanismen für Informationsmodelle**. In: Wirtschaftsinformatik 46 (2004) 4, S. 251-264.

BeEE09 Berekoven, L.; Eckert, W.; Ellenrieder, P.: **Marktforschung. Methodische Grundlagen und praktische Anwendung**. 12. Aufl., Wiesbaden (Gabler) 2009.

Beek03 Beekmann, F.: **Stichprobenbasierte Assoziationsanalyse im Rahmen des Knowledge Discovery in Databases**. Wiesbaden (DUV) 2003. Zugl.: Diss., Univ. Duisburg-Essen 2003.

BeGi00 Bensusan, H.; Giraud-Carrier, C.: **Discovering Task Neighbourhoods through Landmark Learning Performances**. In: [ZiKZ00], S. 325-330.

BeHa99 Berthold, M.; Hand, D.J. (Hrsg.): **Intelligent Data Analysis. An Introduction**. Berlin u.a. (Springer) 1999.

BeHP02 Bernstein, A.; Hill, S.; Provost, F.: **Intelligent Assistance for the Data Mining Process: An Ontology-based Approach**. Working Paper IS-02-02, Center for Digital Economy Research (CeDER), Leonard Stern School of Business, New York University. New York (New York University) 2002. URL: http://citeseer.ist.psu.edu/611926.html (Abruf am 23.02.2007).

BeKR00 Becker, J.; Kugeler, M.; Rosemann, M.: **Prozessmanagement. Ein Leitfaden zur prozessorientierten Organisationsgestaltung**. Berlin (Springer) 2000.

BeLi00 Berry, M.J.A.; Linoff, G.S.: **Mastering Data Mining. The Art and Science of Customer Relationship Management**. New York (Wiley Computer Publishing) 2000.

BeLi97 Berry, M.J.A.; Linoff, G.: **Data Mining Techniques. For Marketing, Sales, and Customer Support**. New York (Wiley) 1997.

Bend02 Bendoly, E.: **Theory and Support for Process Frameworks of Knowledge Discovery and Data Mining from ERP Systems**. In: Information & Management 40 (August 2002) 7, URL: http://www.fc.bus.emory.edu/~elliot_bendoly/S04A5A819.-1/ERP_I&M.pdf, 2002 (Abruf am 21.02.2006).

Benk16 Benker, T.: **A Generic Process Data Warehouse Schema for BPMN Workflows**. In: Abramowicz, W.; Alt, R.; Franczyk, B. (Hrsg.): Business Information Systems. Proc. 19th International Conference BIS 2016, 6.-8. Juli 2016, Leipzig. LNBIP Band 255, o.O (Springer International Publishing Switzerland) 2016, S. 222-234.

Benn94 Benninghaus, H.: **Einführung in die sozialwissenschaftliche Datenanalyse**. 3. Aufl., mit Diskette. München, Wien (Oldenbourg) 1994.

Benz73 Benzécri, J.-P. (Hrsg.): **L'analyse des données. Leçons sur l'analyse factorielle et la reconnaissance des formes et travaux du Laboratoire de Statistique de l'Université de Paris**. 2 Bände. Paris (Dunod) 1973.

BePH05 Bernstein, A.; Provost, F.; Hill, S.: **Towards Intelligent Assistance for a Data Mining Process: An Ontology-based Approach for Cost-sensitive Classification**. In: IEEE Transactions on Knowledge and Data Engineering 17 (2005) 4, S. 503-518.

BePr01 Bernstein, A.; Provost, F.: **An Intelligent Assistant for the Knowledge Discovery Process**. Working Paper IS-01-01, Center for Digital Economy Research (CeDER), Leonard Stern School of Business, New York University, New York (New York University) 2001. URL: http://citeseer.ist.psu.edu/bernstein01intelligent.html (Abruf am 23.02.2007).

BEPR11 Backhaus, K.; Erichson, B.; Plinke, W.; Weiber, R.: **Multivariate Analysemethoden. Eine anwendungsorientierte Einführung**. 13. Aufl., Heidelberg (Springer) 2011.

Berg81 Berg, C.C.: **Organisationsgestaltung**. Stuttgart 1981.

Bert+09 Berthold, M.R.; Cebron, N.; Dill, F.; Gabriel, T.R.; Kötter, T.; Meinl, T.; Ohl, P.; Theil, K.; Wiswedel, B.: **KNIME. The Konstanz Information Miner. Version 2.0 and Beyond**. In: SIGKDD Explorations 11 (2009) 1, S. 26-31. URL: http://www.kdd.org/ exploration_files/ p4V11n1.pdf (Abruf am 04.11.2016).

Bert75 Berthel, J.: **Betriebliche Informationssysteme**. Stuttgart (Poeschel) 1975.

BeSc11 Bernstein, A.; Schwabe, G. (Hrsg.): **Proc. 10th International Conference on Wirtschaftsinformatik**, Zürich (Schweiz), Februar 2011. URL: https://files.ifi.uzh.ch/WI2011/ (Abruf am 08.04.2011).

BeSc95 Bea, F.X.; Schnaitmann, H.: **Begriff und Struktur betriebswirtschaftlicher Prozesse**. In: Wirtschaftswissenschaftliches Studium (WiSt) 24 (1995) 6, S. 278-282.

BeST99 Berson, A.; Smith, S.; Thearling, K.: **Building Data Mining Applications for CRM**. New York (McGraw-Hill) 1999.

BeUB16 Becker, W.; Ulrich, P.; Botzkowski, T.: **Data Analytics im Mittelstand**. Management und Controlling im Mittelstand. Wiesbaden (Springer Gabler) 2016.

BeWi95 Bergmann, R.; Wilke, W.: **Learning Abstract Planning Cases**. In: Lavrac, N.; Wrobel, S. (Hsg.): Machine Learning: ECML-95. Proc. 8th European Conference on Machine Learning, Heraclion (Crete, Greece), April 1995. Berlin (Springer) 1995, S. 55-76.

BGKS04 Bouzeghoub, M.; Goble, C.; Kashyap, V.; Spaccapietra, S. (Hrsg.): **Semantics of a Networked World. Semantics for Grid Databases**. Proc. ICSNW 2004, Paris, 17.-19. Juni 2004. LNCS 3226, Heidelberg (Springer) 2004.

BHB+10 Balko, S.; ter Hofstede, A.H.M.; Barros, A.; La Rosa, M.; Adams, M.: **Business Process Extensibility**. In: Enterprise Modelling and Information System Architectures 5 (2010) 3, S. 4-23.

BHMS03 Berbner, R.; Heckmann, O.; Mauthe, A.; Steinmetz, R.: **Eine Dienstgüte unterstützende Web-Service-Architektur für flexible Geschäftsprozesse**. In: Wirtschaftsinformatik 47 (2003) 4, S. 268-277.

Bigu96 Bigus, J.P.: **Data Mining with Neural Networks. Solving Business Problems from Application Development to Decision Support**. New York (McGraw-Hill) 1996.

BiHA16 Bichler, M.; Heinzl, A.; van der Aalst, W.M.P.: **Business Analytics and Data Science: Once Again?** Editorial. In: Bus Inf Syst Eng. Online First, 20.12.2016. DOI 10.1007/s12599-016-0461-1. URL: http://link.springer.com/article/10.1007/s12599-016-0461-1 (Abruf am 03.01. 2017).

Biss01 Bissantz, N.: **DeltaMiner**. In: Wirtschaftsinformatik 43 (2001) 1, S. 77-80.

Biss96 Bissantz, N.: **CLUSMIN. Ein Beitrag zur Analyse von Daten des Ergebniscontrollings mit Datenmustererkennung (Data Mining)**. Arbeitsberichte des Instituts für mathematische Maschinen und Datenverarbeitung 29 (1996) 7, Erlangen 1996. Dissertation, Universität Erlangen-Nürnberg 1996.

BlGG00 Bleymüller, J.; Gehlert, G.; Gülicher, H.: **Statistik für Wirtschaftswissenschaftler**, 12. Aufl., München (Vahlen) 2000.

BoAr01 Bonne, T.; Arminger, G.: **Diskriminanzanalyse**. In: [HKMW01], S. 193-239.

Boeh88 Boehm, B.W.: **A Spiral Model of Software Development and Enhancement**. In: IEEE Computer 21 (1988) 5, S. 61-72.

Böhn01 Böhnlein, M.: **Konstruktion semantischer Data-Warehouse-Schemata**. Wiesbaden (DUV) 2001. Zugl.: Dissertation, Universität Bamberg, 2001.

BöKU03 Böhnlein, M.; Knobloch, B.; Ulbrich-vom Ende, A.: **Synergieeffekte zwischen Data Warehousing, OLAP und Data Mining – Eine Bestandsaufnahme**. In: [MaWi03], S. 167-193.

Boll96 Bollinger, T.: **Assoziationsregeln. Analyse eines Data Mining Verfahrens**. In: Informatik-Spektrum 19 (1996) 5, S. 257-261.

Bors97 Borst, W.N.: **Construction of Engineering Ontologies**. PhD Thesis, Enschede (Universität Twente) 1997.

BöUl00 Böhnlein, M.; Ulbrich-vom Ende, A.: **Grundlagen des Data Warehousing. Modellierung und Architektur**. Bamberger Beiträge zur Wirtschaftsinformatik 55, Bamberg (Otto-Friedrich-Universität) 2000.

Brac+93 Brachman, R.J.; Halper, F.; Selfridge, P.G.; Kirk, T.; Terveen, L.G.; Lazar, A.; Altman, B.; McGuinness, D.L.; Borgida, A.; Alperin Resnick, L.: **Integrated Support for Data Archaeology**. In: Proc. KDD-93 Workshop, Menlo Park (AAAI Press) 1993.

BrAn96 Brachman, R.J.; Anand, T.: **The Process of Knowledge Discovery in Databases: A Human-Centered Approach**. In: [FPSU96], S. 37-58.

BrFa00 Bruha, I.; Famili, A.: **Postprocessing in Machine Learning and Data Mining**. In: SIGKDD Explorations 2 (2000) 2. URL: http://www.kdd.org/exploration_files/KDD2000PostWkshp.pdf, 2000 (Abruf am 02.03.2017).

BrFM97 Brillinger, D.R.; Fernholz, L.T.; Morgenthaler, S. (Hrsg.): **The Practice of Data Analysis**. Essays in Honor of John W. Tukey. Princeton (Princeton University Press) 1997.

BrJT08 Brezany, P.; Janciak, I.; Tjoa, A.M.: **Ontology-based Construction of Grid Data Mining Workflows**. In: Nigro, H.O.; Gonzales Cisaro, S.E.; Xodo, D.H. (Hrsg.): Data Mining with Ontlogies: Implementations, Findings and Frameworks. O.O. (IGI Global) 2008.

BrMi92 Bramer, M.A.; Milne, R.W. (Hrsg.): **Research and Development in Expert Systems IX**. Proc. Expert Systems 92, the Twelfth Annual Technical Conference of the British Computer Society Specialist Group on Expert Systems, London, December 1992, Cambridge (Cambridge University Press) 1992.

Bron92 Bronner, R.: **Komplexität**. In: Frese, E. (Hrsg.): Handwörterbuch der Organisation (Enzyklopädie der Betriebswirtschaftslehre, Band II). 3. Aufl. Stuttgart (Poeschel) 1992, S.1122-1129.

BrSW97 Breitner, C.; Schlösser, J.; Wirth, R.: **Process-Based Database Support for the Early Indicator Method**. In: [HMPU97], S. 131-134.

BSCP03 Brazdil, P.; Soares, C.; Costa, J.; Pinto, D.: **Ranking Learning Algorithms: Using IBL and Meta-Learning on Accuracy and Time Results**. In: Machine Learning 50 (2003) 3, S. 251-277.

CaCo03 Cannataro, M.; Comito, C.: **A Data Mining Ontology for Grid Programming**. In: Proc. 1st Int. Workshop on Semantics in Peer-to-Peer and Grid Computing, in conjunction with WWW 2003, Budapest (Hungary) 2003, S. 113–134.

CaFa16 Cao, L.; Fayyad, U.: **Data Science: Challenges and Directions**. Eingereicht für Communications of the ACM, 2016. URL: http://www-staff.it.uts.edu.au/~lbcao/publication/ds-cacm16.final.pdf (Abruf am 09.12.2016).

CaNi04 Caruana, R.; Niculescu-Mizil, A.: **Data Mining in Metric Space: An Empirical Analysis of Supervised Learning Performance Criteria**. In: Kim, W.; Kohavi, R.; Gehrke, J.; Du-Mouchel, W. (Hrsg): Proc. 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, USA. New York (ACM Press) 2004, S. 69-78.

CaNi06 Caruana, R.; Niculescu-Mizil, A.: **An Empirical Comparison of Supervised Learning Algorithms**. In: Cohen, W.W.; Moore, A. (Hrsg): Proc. ICML '06, The 23rd International Conference on Machine Learning, Pittsburg, USA. New York (ACM) 2006.

Cao+07 Cao, L.; Yu, P.S.; Zhang, C.; Zhao, Y.; Williams, G.: **DDDM2007: Domain Driven Data Mining**. Review of the ACM SIGKDD Workshop on Domain Driven Data Mining (DDDM2007). In: SIGKDD Explorations 9 (2007) 2, S. 84-86.

Cao16 Cao, L.: **Data Science: A Comprehensive Overview**. Eingereicht für ACM Computing Surveys. URL: https://www-staff.it.uts.edu.au/~lbcao/publication/dsa-cusr.draft.pdf (Abruf am 09.12.2016).

Capl71 Caplow, T: **Elementary Sociology**. Englewood Cliffs (Prentice Hall) 1971.

Catt00 Cattell, R.G.G.; Barry, D.; Berler, M.; Eastman, J.; Jordan, D.; Russell, C.; Schadow, O.; Stanienda, T.; Velez, F.: **The Object Data Standard: ODMG 3.0**. San Francisco (Morgan Kaufmann) 2000.

CCDG05 Coutaz, J.; Crowley, J.L.; Dobson, S.; Garlan, D.: **Context is Key**. In: Communications of the ACM 48 (2005) 3, S. 49-53.

CCK+00 Chapman, P.; Clinton, J.; Kerber, R.; Khabaza, T.; Reinartz, T.; Shearer, C.; Wirth, R.: **CRISP-DM 1.0 – Step-by-Step Data Mining Guide**, http://www.crisp-dm.org/CRISPWP-0800.pdf, 2000 (Abruf am 28.10.2000).

CFM+97 Chien, S.; Fisher, F.; Mortensen, H.; Lo, E.; Greeley, R.: **Using Artificial Intelligence Planning to Automate Science Data Analysis for Large Image Databases**. In: [HMPU97], S. 147-150.

Cham98 Chamoni, P.: **Entwicklungslinien und Architekturkonzepte des On-Line Analytical Processing**. In: [ChGl98a], S. 231-250.

ChDü98 Chamoni, P.; Düsing, R. (Hrsg.): **Workshop „Data Mining: Grundlagen, Verfahren und Anwendungen der Datenanalyse"**. Arbeitsberichte des Fachgebietes Wirtschaftsinformatik und Operations Research Nr. 2, Gerhard-Mercator-Universität – Gesamthochschule Duisburg 1998.

Chen01 Chen, Z.: **Data Mining and Uncertain Reasoning. An Integrated Approach**. New York (Wiley) 2001.

ChGl98a Chamoni, P.; Gluchowski, P. (Hrsg.): **Analytische Informationssysteme. Data Warehouse, On-Line Analytical Processing, Data Mining**. Berlin (Springer) 1998.

ChGl99b Chamoni, P.; Gluchowski, P.: **Entwicklungslinien und Architekturkonzepte des On-Line Analytical Processing**. In: Chamoni, P.; Gluchowski, P. (Hrsg.): Analytische Informationssysteme. Data Warehouse, On-Line Analytical Processing, Data Mining. 2. Aufl., Berlin (Springer) 1999, S. 261-280.

CHS+97 Cabena, P.; Hadjinian, P.; Stadler, R.; Verhees, J.; Zanasi, A.: **Discovering Data Mining. From Concept to Implementation**, Upper Saddle River (Prentice Hall) 1997.

ClLä14 Cleve, J.; Lämmel, U.: **Data Mining**. München (Oldenbourg) 2014.

Cohe+09 Cohen, J.; Dolan, B.; Dunlap, M.; Hellerstein, J.M.; Welton, C.: **MAD Skills: New Analysis Practices for Big Data**. In: Proc. VLDB Endowment (PVLDB) 2 (2009) 2, S. 1481-1492. URL: http://www.vldb.org/pvldb/2/vldb09-219.pdf (Abruf am 03.01.2017).

Cohe88 Cohen, J.: **Statistical Power Analysis for the Behavioral Sciences**. 2. Aufl., New York (Academic Press) 1988.

Cohe96 Cohen, P.R.: **Getting what you deserve from data**. In: IEEE Expert 11 (October 1996) 5, S. 12-14.

Coom67 Coombs, C.H.: **A Theory of Data**. 2. Aufl., New York (Wiley) 1967.

CoRe08 Corsten, H.; Reiß, M. (Hrsg.): **Betriebswirtschaftslehre**, Band 2. 4. Aufl., München (Oldenbourg) 2008.

Cros79 Crosby, P.B.: **Quality is Free**. New York (Penguin Group) 1979.

CSG+92 Craw, S.; Sleeman, D.; Graner, N.; Rissakis, M.; Sharma, S.: **CONSULTANT: Providing Advice for the Machine Learning Toolbox**. In: [BrMi92], S. 5-23.

Cues13 Cuesta, H.: **Practical Data Analysis. Transform, model, and visualize your data through hands-on projects, developed in open source tools**. Birmingham (Packt Publishing) 2013.

CYZZ10 Cao, L.; Yu, P.S.; Zhang, C.; Zhao, Y.: **Domain Driven Data Mining**. New York (Springer) 2010.

Daen88 Daenzer, W.F. (Hrsg.): **Systems Engineering. Leitfaden zur methodischen Durchführung umfangreicher Planungsvorhaben**. 6. Aufl., Zürich (Verlag Industrielle Organisation) 1988.

DaHa07 Davenport, T.H.; Harris, J.G.: **Competing on Analytics. The New Science of Winning**. Boston (Harvard Business School Press) 2006.

DaHM10 Davenport, T.H.; Harris, J.G.; Morison, R.: **Analytics at Work: Smarter Decisions, Better Results**. O.O. (Harvard Business Review Press) 2010.

DaRe04 Dadam, P.; Reichert, M.: **ADEPT – Prozess-Management-Technologie der nächsten Generation**. In: Spath, D.; Haasis, K. (Hrsg.): Aktuelle Trends in der Softwareforschung. Tagungsband zum doIT Software-Forschungstag 2003, IRB Verlag Stuttgart 2004, S. 27-43. URL: http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/DaRe04.pdf (Abruf am 28.01.2006).

DaRR11 Dadam, P.; Reichert, M.; Rinderle-Ma, S.: **Prozessmanagementsysteme**. In: Informatik-Spektrum 34 (2011) 4, S. 364-376.

Date95 Date, C.J.: **An Introduction to Database Systems**. 6. Aufl., Reading (Addison-Wesley) 1995.

Dave06 Davenport, T.H.: **Competing on Analytics**. In: Harvard Business Review (Januar 2006), S. 99-107.

Dave93 Davenport, T.H.: **Process Innovation. Reengineering Work through Information Technology**. Boston (Harvard Business School Press) 1993.

DeAb00 Dey, A.K.; Abowd, G.K.: **Towards a better understanding of context and context-awareness**. In: Proc. CHI 2000 Workshop on the What, Who, Where, When, and How of Context-Awareness, Vol. 4. The Hague (ACM Press), S. 1-6.

DeFr80 Deutsch, K.W.; Fritsch, B.: **Zur Theorie der Vereinfachung: Reduktion von Komplexität in der Datenverarbeitung für Weltmodelle**. Königstein/Ts. (Athenäum) 1980.

DeGE08 Deutsche Gesellschaft für Evaluation (DeGEval) e.V. (Hrsg.): **Standards für Evaluation**. 4. Aufl., Mainz 2008. Kurzfassung verfügbar unter http://www.degeval.de/fileadmin/user_upload/Sonstiges/STANDARDS_2008-12_kurz.pdf (Abruf am 2016-10-07).

DeGS95 Deiters, W.; Gruhn, V.; Striemer, R.: **Der FUNSOFT-Ansatz zum integrierten Geschäftsprozeßmanagement**. In: Wirtschaftsinformatik 37 (1995) 5, S. 459-466.

DeHa01 Delmater, R.; Hancock, M.: **Data Mining Explained: A Manager's Guide to Customer-Centric Business Intelligence**. Boston (Digital Press) 2001.

Deit00 Deiters, W.: **Information Gathering and Process Modeling in a Petri Net Based Approach**. In: [AaDO00], S. 274-288.

DeKl00 Dellarocas, C.; Klein, M.: **A Knowledge-Based Approach for Designing Robust Business Processes**. In: [AaDO00], S. 50-65.

DeOb96 Desel, J.; Oberweis, A.: **Petri-Netze in der Angewandten Informatik**. In: Wirtschaftsinformatik 38 (1996) 4, S. 359-366.

Devl97 Devlin, B.: **Data Warehouse. From Architecture to Implementation**. Reading (Addison-Wesley) 1997.

Dhar13 Dhar, V.: **Data Science and Prediction**. In: Communications of the ACM 56 (Dezember 2013) 12, S. 64-73.

DhSt97 Dhar, V.; Stein, R.: **Seven Methods for Transforming Corporate Data into Business Intelligence**. Upper Saddle River (Prentice-Hall) 1997.

Diek07 Diekmann, A.: **Empirische Sozialforschung. Grundlagen, Methoden, Anwendungen**. 18. Aufl., Reinbek b. Hamburg (Rowohlt) 2007.

DIN00 DIN Deutsches Institut für Normung e.V. (Hrsg.): **Geschäftsprozessgestaltung: Typisierung und Modellierung**. Berlin (Beuth) 2000.

DiPS09 Diamantini, C.; Potena, D.; Storti, E.: **Ontology-Driven KDD Process Composition**. In: [ARSB09], S. 285-296.

DKRS83 Dörner, D.; Kreuzig, H.W.; Reither, F.; Stäudel, T. (Hrsg.): **Lohhausen. Vom Umgang mit Unbestimmtheit und Komplexität**. Bern (Huber) 1983.

Domi99 Domingos, P.: **The Role of Occam's Razor in Knowledge Discovery**. In: Data Mining and Knowledge Discovery 3 (1999), S. 409-425.

Döri10 Döring, N.: **Planung und Durchführung von Evaluationsstudien**. In: [HoSc10], S. 261-272.

Dörn79 Dörner, D.: **Problemlösen als Informationsverarbeitung**, 2. Aufl., Stuttgart (Kohlhammer) 1979.

Dörn83 Dörner, D.: **Die Anforderungen komplexer und unbestimmter Probleme**. In: [DKRS83], S. 19-104

Dörn83a Dörner, D.: **Handeln in Komplexität und Unbestimmtheit**. In: [DKRS83], S. 19-26.

Dree01 Dreesmann, J.: **Datenanalyse, computergestützte**. In: [Mert01], S. 134-135.

Drei+05 Dreiling, A.; Rosemann, M.; van der Aalst, W.M.P.; Sadiq, W.; Khan, S.: **Model-Driven Process Configuration of Enterprise Systems**. In: [FSEI05], S. 687-706.

Drei94 Dreier, V.: **Datenanalyse für Sozialwissenschaftler**. München (Oldenbourg) 1994.

DRRA05 Dadam, P.; Reichert, M.; Rinderle, S.; Atkinson, C.: **Auf dem Weg zu prozessorientierten Informationssystemen der nächsten Generation: Herausforderungen und Lösungskonzepte**. In: Spath, D.; Haasis, K.; Klumpp, D. (Hrsg.): Aktuelle Trends in der Softwareforschung. Tagungsband zum doIT Software-Forschungstag 2005, Karlsruhe, Juni 2005. MFG Stiftung 2005, S. 47-67. URL: http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/DRRA05.pdf (Abruf am 28.01.2006).

Dude16b Bibliographisches Institut GmbH, Dudenverlag (Hrsg.): **Duden: Verständnis.** Rechtschreibung, Bedeutung, Defintion, Herkunft. URL: http://www.duden.de/rechtschreibung/Verstaendnis (Abruf am 07.09.2016).

Dude16c Bibliographisches Institut GmbH, Dudenverlag (Hrsg.): **Duden: Fehler.** Rechtschreibung, Bedeutung, Defintion, Herkunft. URL: http://www.duden.de/rechtschreibung/Fehler (Abruf am 17.09. 2016).

Dude16d Bibliographisches Institut GmbH, Dudenverlag (Hrsg.): **Duden: synchronisieren.** Rechtschreibung, Bedeutung, Defintion, Herkunft. URL: http://www.duden.de/rechtschreibung/synchronisieren (Abruf am 17.09.2016).

Dude17b Bibliographisches Institut GmbH, Dudenverlag (Hrsg.): **Duden: inkrementell.** Rechtschreibung, Bedeutung, Defintion, Herkunft. URL: http://www.duden.de/rechtschreibung/inkrementell (Abruf am 02.03.2017).

Dude17c Bibliographisches Institut GmbH, Dudenverlag (Hrsg.): **Duden: Kriterium.** Rechtschreibung, Bedeutung, Defintion, Herkunft. URL: http://www.duden.de/rechtschreibung/Kriterium (Abruf am 04.03. 2017).

Dude17d Bibliographisches Institut GmbH, Dudenverlag (Hrsg.): **Duden: Revision.** Rechtschreibung, Bedeutung, Defintion, Herkunft. URL: http://www.duden.de/rechtschreibung/Revision (Abruf am 20.04. 2017).

Ecke04b Eckerson, W.: **Understanding Business Intelligence**. In: What Works 16 (November 2003). Abdruck im TDWI Germany E-Mail Letter 7/2004. URL: http://www.sigs.de/newsletter/tdwi/2004/07/Art_05.pdf (Abruf am13.08.2004).

EdOG02 Eder, J.; Olivotto, G.; Gruber, W.: **A Data Warehouse for Workflow Logs**. In: Han, Y.; Tai, S.; Wikarski, D. (Hrsg.): Proc. International Conference on Engineering and Deployment of Cooperative Information Systems (EDCIS 2002), Beijing, China, 17-20 September 2002. LNCS 2480, Berlin (Springer) 2002, S. 1-15.

Ehre76 Ehrenberg, A.S.C.: **Das Reduzieren der Zahlen. Statistische Analyse und Interpretation**. Köln (Bund-Verlag) 1976.

ElKe00 Ellis, C.A.; Keddara, K.: **A Workflow Change Is a Workflow**. In: [AaDO00], S. 201-217.

ElLa04 Elfatatry, A.; Layzell, P.: **Negotiating in Service-Oriented Environments**. In: Communications of the ACM 47 (August 2004) 8, S. 103-108.

ElPr96 Elder IV, J.F.; Pregibon, D.: **A Statistical Perspective on Knowledge Discovery in Databases**. In: [FPSU96], S. 83-113.

EMC15 EMC Education Services (Hrsg.): **Data Science and Big Data Analytics. Discovering, Analyzing, Visualizing and Presenting Data**. Indianapolis (Wiley) 2015.

EmMD16 Emmert-Streib, F.; Moutari, S.; Dehmer, M.: **The Process of Analyzing Data is the Emergent Feature of Data Science**. In: Frontiers in Genetics 7 (February 2016) 7. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4746257/pdf/fgene-07-00012.pdf (Abruf am 09. 12.2016).

Enge96 Engels, R.: **Planning Tasks for Knowledge Discovery in Databases. Performing Task-Oriented User-Guidance**. In: [SiHF96], S. 170-175.

Enge99 Engels, R.: **Component-Based User Guidance in Knowledge Discovery and Data Mining**. Sankt Augustin (infix) 1999. Zugl.: Karlsruhe, Univ., Diss., 1999.

Engl99 English, L.P.: **Improving Data Warehouse and Business Information Quality: Methods for Reducing Costs and Increasing Profits**. New York u.a. (Wiley) 1999.

EnLS97 Engels, R.; Lindner, G.; Studer, R.: **A Guided Tour through the Data Mining Jungle**. In: [HMPU97], S. 163-166.

EnLS97b Engels, R.; Lindner, G.; Studer, R.: **A Methodology for Providing User Support for Developing Knowledge Discovery Applications**. Working Paper, AIFB, Universität Karlsruhe 1997. URL: http://citeseer.ist.psu.edu/578421.html (Abruf am 23.02.2007).

EnMT95 Engel, A.; Möhring, M.; Troitzsch, K.G.: **Sozialwissenschaftliche Datenanalyse**. Mannheim (BI-Wissenschafts-Verlag) 1995.

EnTh98 Engels, R.; Theusinger, C.: **Using a Data Metric for Preprocessing Advice for Data Mining Applications**. In: Prade, H. (Hrsg.): ECAI 98. Proc. 13th European Conference on Artificial Intelligence (August 23-28, 1998, Brighton, UK). Chichester (Wiley) 1998, S. 430-434.

EnTh98b Engels, R.; Theusinger, C.: **Support for data transformation in machine learning applications**. In: Giraud-Carrier, C.; Hilario, M. (Hrsg.): Proc. 10th ECML Workshop on Upgrading Learning to the Meta-Level: Model Selection and Data Transformation, Vol. CSR-98-02, Technische Universität Chemnitz, Chemnitz 1998, S. 43-53.

ErVo92 Erfle, R.; Vogel, P.: **Backtracking Office Procedures**. In: Proc. DEXA 92 – International Conference on Database and Expert System Applications, Valencia, Spanien 1992, S. 506ff.

FaBi12 Fan, W.; Bifet, A.: **Mining Big Data: Current Status, and Forecast to the Future**. In: SIGKDD Explorations 14 (2012) 2, S. 1-5. URL: http://www.kdd.org/ exploration_files/V14-02-01-Fan.pdf (Abruf am 04.11.2016).

FaPS96 Fayyad, U.; Piatetsky-Shapiro, G.; Smyth, P.: **From Data Mining to Knowledge Discovery: An Overview**. In: [FPSU96], S. 1-34.

FaPS96b Fayyad, U.; Piatetsky-Shapiro, G.; Smyth, P.: **Knowledge Discovery and Data Mining: Towards a Unifying Framework**. In: [SiHF96], S. 82-88.

FaUt02 Fayyad, U.; Uthurusamy, R.: **Evolving Data Mining into Solutions for Insights**. In: Communications of the ACM 45 (2002) 8, S. 28-31.

FaUt94 Fayyad, U.; Uthurusamy, R. (Hrsg.): **Knowledge Discovery in Databases**. Papers from the 1994 AAAI Workshop. August 2, Seattle, Washington. Technical Report WS-94-03. Menlo Park (AAAI Press) 1994.

FaUt95 Fayyad, U.; Uthurusamy, R. (Hrsg.): **Proc. The First International Conference on Knowledge Discovery & Data Mining**, Montréal, August 1995. Menlo Park (AAAI Press) 1995.

Fawc06 Fawcett, T.: **An introduction to ROC analysis**. In: Pattern Recognition Letters 27 (2006) 8, S. 861-874.

FeBr16 Fettke, P.; vom Brocke, J.: **Referenzmodell**. In: Gronau, N.; Becker, J.; Sinz, E.J.; Suhl, L.; Leimeister, J.M. (Hrsg.): Enzyklopädie der Wirtschaftsinformatik, Online-Lexikon. URL: http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/is-management/Systementwicklung/Softwarearchitektur/Wiederverwendung-von-Softwarebausteinen/Referenzmodell, letzte Änderung: 26.09.2016 (Abruf am 20.01.2017).

FeHa94 Ferstl, O.K.; Hagemann, U.: **Simulation hierarchischer objekt- und transaktionsorientierter Modelle**. Bamberger Beiträge zur Wirtschaftsinformatik Nr. 23, Bamberg 1994.

FeLo02 Fettke, P.; Loos, P.: **Methoden zur Wiederverwendung von Referenzmodellen: Übersicht und Taxonomie**. In: Becker, J.; Knackstedt, R. (Hrsg.): Referenzmodellierung 2002. Modelle, Methoden, Erfahrungen. Arbeitsberichte des Instituts für Wirtschaftsinformatik Nr. 90, Münster (Westfälische Wilhelms-Universität) 2002, S. 9-33.

FeLW11 Ferstl, O.K.; Leunig, B.; Wagner, D.: **Analyse und Handhabung unvollständig planbarer Geschäftsprozesse**. In: [SBBF11], S. 151-171.

FeMa95 Ferstl, O.K.; Mannmeusel, T.: **Gestaltung industrieller Geschäftsprozesse**. Bamberger Beiträge zur Wirtschaftsinformatik Nr. 31, Bamberg 1995. Auch erschienen in: Wirtschaftsinformatik 37 (1995) 5, S. 446-458.

FeMa95a Ferstl, O.K.; Mannmeusel, T.: **Dezentrale Produktionslenkung**. Bamberger Beiträge zur Wirtschaftsinformatik Nr. 27, Bamberg 1995.

Fers+98 Ferstl, O.K.; Sinz, E.J.; Hammel, C.; Schlitt, M.; Wolf, S.; Popp, K.; Kehlenbeck, R.; Pfister, A.; Kniep, H.; Nielsen, N.; Seitz, A.: **WEGA: Wiederverwendbare und erweiterbare Geschäftsprozeß- und Anwendungssystemarchitekturen**. Abschlussbericht. Bamberg, Walldorf, Frankfurt (Universität Bamberg, SAP AG, KPMG Unternehmensberatung GmbH) 1998.

Fers79 Ferstl, O.K.: **Konstruktion und Analyse von Simulationsmodellen**. Königstein/Ts. (Hain) 1979.

Fers92 Ferstl, O.K.: **Integrationskonzepte betrieblicher Anwendungssysteme**. Fachberichte Informatik 1/92, Koblenz (Universität Koblenz-Landau) 1992.

FeSi01 Ferstl, O.K.; Sinz, E.J.: **Grundlagen der Wirtschaftsinformatik**, Band 1. 4. Aufl., München (Oldenbourg) 2001.

FeSi13 Ferst, O.K.; Sinz, E.J.: **Grundlagen der Wirtschaftsinformatik**, 7. Aufl., München (Oldenbourg) 2013.

FHTN11 Fellmann, M.; Hogrebe, F.; Thomas, O.; Nüttgens, M.: **Checking the Semantic Correctness of Process Models. An Ontology-driven Approach Using Domain Knowledge and Rules**. In: Enterprise Modelling and Information Systems Architectures 6 (2011) 3, S. 25-35.

FiBo90 Fisch, R.; Boos, M. (Hrsg.): **Vom Umgang mit Komplexität in Organisationen: Konzepte, Fallbeispiele, Strategien.** Konstanz (Universitäts-Verlag) 1990.

Fisc93 Fischer, T.: **Computergestützte Warenkorbanalyse – dargestellt auf der Grundlage von Scanningdaten des Lebensmitteleinzelhandels unter besonderer Berücksichtigung einer selbsterstellten Analysesoftware**. Schriften zu Distribution und Handel, Band 11. Frankfurt/Main (Lang) 1993. Zugl.: Diss., Münster, Univ., 1992.

FiWo90 Fisch, R.; Wolf, M.F.: **Die Handhabung von Komplexität beim Problemlösen und Entscheiden**. In: [FiBo90], S. 11-39.

Flic02 Flick, U.: **Qualitative Sozialforschung. Eine Einführung**, 6. Aufl., Reinbek (Rowohlt Taschenbuch) 2002.

FlZü97 Floyd, C.; Züllighofen, H.: **Softwaretechnik**. In: Rechenberg, P.; Pomberger, G. (Hrsg.): Informatik-Handbuch. München (Hanser) 1997, S. 641–667.

Forr61 Forrester, J.W.: **Industrial Dynamics**. Waltham (Pegasus Communications) 1961.

FPSU96 Fayyad, U.; Piatetsky-Shapiro, G.; Smyth, P.; Uthurusamy, R. (Hrsg.): **Advances in Knowledge Discovery and Data Mining**. Menlo Park (AAAI Press) 1996.

Fres92 Frese, E. (Hrsg.): **Handwörterbuch der Organisation** (Enzyklopädie der Betriebswirtschaftslehre, Band II). 3. Aufl., Stuttgart (Poeschel) 1992.

FrPM91 Frawley, W.J.; Piatetsky-Shapiro, G.; Matheus, C.J.: **Knowledge Discovery in Databases: An Overview**. In: Piatetsky-Shapiro, G.; Frawley, W.J. (Hrsg.): Knowledge Discovery in Databases. Menlo Park (AAAI Press) 1991, S. 1-27.

FrRü10 Friesen, N.; Rüping, S.: **Workflow Analysis using Graph Kernels**. In: [HiLK10], S. 13-24.

FSEI05 Ferstl, O.K.; Sinz, E.J.; Eckert, S.; Isselhorst, T. (Hrsg.): Proc. **Wirtschaftsinformatik 2005**. eEconomy, eGovernment, eSociety. Heidelberg (Physica) 2005.

FSWS97 Famili, A.; Shen, W.-M.; Weber, R.; Simoudis, E.: **Data Preprocessing and Intelligent Data Analysis**. In: Intelligent Data Analysis 1 (1997) 1, S. 3-23.

GaBr95 Gama, J.; Brazdil, P.: **Characterization of Classification Algorithms**. In: Proc. 7th Portuguese Conference on Artificial Intelligence, EPIA '95, S. 189-200. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.25.202&rep=rep1&type=pdf (Abruf am 01.09.2011).

GaFi92 Gabele, E.; Fischer, P.: **Kosten- und Erlösrechnung**. München (Vahlen) 1992.

Gait83 Gaitanides, M.: **Prozeßorganisation. Entwicklung, Ansätze und Programme prozeßorientierter Organisationsgestaltung**. München (Vahlen) 1983.

Galt67 Galtung, J.: **Theory and Method of Social Research**. New York (Columbia University Press) 1967.

GaSc88 Gaul, W.; Schader, M.: **Characterization of Research Papers by Data Analysis Techniques**. In: Gaul, W.; Schader, M. (Hrsg.): Data, Expert Knowledge and Decisions. Heidelberg (Springer) 1988, S.3-9.

GaSc89 Gaul, W.; Schader, M.: **Data Analysis and Decision Support**. In: Applied Stochastic Models and Data Analysis 5 (1989) 11, S.341-356.

GaSc94 Gaul, W.; Schader, M. (Hrsg.): **Wissensbasierte Marketing-Datenanalyse. Das WIMDAS-Projekt**. Frankfurt/M. (Lang) 1994.

GaSV94 Gaitanides, M.; Scholz, R.; Vrohlings, A.: **Prozeßmanagement: Grundlagen und Zielsetzungen**. In: [GSVR94], S. 1-19.

GCC+04 Grigori, D.; Casati, F.; Castellanos, M.; Dayal, U.; Sayal, M.; Shan, M.-C.: **Business Process Intelligence**. In: Computers in Industry 53 (2004) 3, S. 321-343.

Geih08 Geihs, K.: **Selbst-adaptive Software**. In: Informatik-Spektrum 31 (2008) 2, S. 133-145.

Geis97 Geisler, M.: **Client/Server-basiertes Geschäftsprozeßmanagement**. o.O. (Verlag Managementwissen Zukunft) 1997.

GeSc84 Gediga, G.; Schöttke, H.: **Problemlösen und Komplexität**. Forschungsberichte aus dem Fachbereich Psychologie der Universität Osnabrück Nr. 40, Universität Osnabrück 1984.

Gier00 Gierhake, O.: **Integriertes Geschäftsprozessmanagement. Effektive Organisationsgestaltung mit Workflow-, Workgroup- und Dokumentenmanagement-Systemen**. 3. Aufl., Braunschweig (Vieweg Gabler) 2000.

GiPo03 Girault-Carrier, C.; Povel, O.: **Characterising Data Mining Software**. In: Intelligent Data Analysis 7 (2003), S. 181-192.

Gira05 Giraud-Carrier, C.: **The Data Mining Advisor: Meta-Learning in the Service of Practitioners**. In: Proc. 4th International Conference on Machine Learning and Applications 2005, S. 113-119.

GlCh16 Gluchowski, P.; Chamoni, P. (Hrsg.): **Analytische Informationssysteme. Business Intelligence-Technologien und -Anwendungen**. 5. Aufl., Berlin (Springer Gabler) 2016.

GlGD08 Gluchowski, P.; Gabriel, R.; Dittmar, C.: **Management-Support-Systeme und Business Intelligence. Computergestützte Informationssysteme für Fach- und Führungskräfte**. 2. Aufl., Berlin (Springer) 2008.

GLKK09 Gruhn, V.; Laue, R.; Kühne, S.; Kern, H.: **A Business Process Modelling Tool with Continuous Validation Support**. In: Enterprise Modelling and Information Systems Architectures 4 (2009) 2, S. 37-51.

GMPS97 Glymour, C.; Madigan, D.; Pregibon, D.; Smyth, P.: **Statistical Themes and Lessons for Data Mining**. In: Data Mining and Knowledge Discovery 1 (1997), 11-28.

GoJä09 Gollwitzer, M.; Jäger, R.S.: **Evaluation kompakt**. Programm PVU (Psychologie Verlags Union), Weinheim (Beltz Verlag) 2009.

GoRi09 Golfarelli, M.; Rizzi, S.: **Data Warehouse Design. Modern Principles and Methodologies**. New York (McGraw Hill) 2009.

GöRS00 Görz, G.; Rollinger, C.-R.; Schneeberger, J. (Hrsg.): **Handbuch der Künstlichen Intelligenz**. 3. Aufl., München (Oldenbourg) 2000.

Grab01 Grabmeier, J.: **Segmentierende und clusterbildende Methoden**. In: [HKMW01], S. 299-361.

GrBC08 Grob, H.L.; Bensberg, F.; Coners, A.: **Regelbasiere Steuerung von Geschäftsprozessen. Konzeption eines Ansatzes auf Basis von Process Mining**. In: Wirtschaftsinformatik 50 (2008) 4, S. 268-281.

GrGe00 Grothe, M.; Gentsch, P.: **Business Intelligence. Aus Informationen Wettbewerbsvorteile gewinnen**. München (Addison-Wesley) 2000.

Gros09b Grossman, R.L.: **What is Analytic Infrastructure and Why Should We Care?** In: SIGKDD Explorations 11 (2009) 1, S. 5-9. URL: http://www.kdd.org/exploration_files/p1V11n1.pdf (Abruf am 04. 11.2016).

GRSS95 Gaul, W.; Radermacher, F.J.; Schader, M.; Solte, D.: **Data, Expert Knowledge, and Decisions: An Introduction to the Volume**. In: Annals of Operations Research 55 (1995), S. 1-7.

Grub93 Gruber, T.R.: **A translation approach to portable ontology specifications**. In: Knowledge Acquisition 5 (1993) 2, S. 199-221.

GSVR94 Gaitanides, M.; Scholz, R.; Vrohlings, A.; Raster, M.: **Prozeßmanagement. Konzepte, Umsetzungen und Erfahrungen des Reengineering**. München (Hanser) 1994.

HaBR08 Hallerbach, A.; Bauer, T.; Reichert, M.: **Anforderungen an die Modellierung und Ausführung von Prozessvarianten**. In: Datenbank-Spektrum 8 (2008) 24, S. 48-58.

HaCC98 Han, J.; Chee, S.H.S.; Chiang, J.Y.: **Issues for On-Line Analytical Mining of Data Warehouses** (Extended Abstract). In: Proc. 1998 SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery (DMKD '98, Seattle, USA, Juni 1998), 1998, S. 2:1-2:5.

Hage96 Hagedorn, J.: **Die automatische Filterung von Controlling-Daten unter besonderer Berücksichtigung der Top-Down-Navigation (BETREX II)**. Arbeitsberichte des Instituts für mathematische Maschinen und Datenverarbeitung 29 (1996) 7, Erlangen 1996. Diss., Universität Erlangen-Nürnberg, 1996.

HaHK04 Hammori, M.; Herbst, J.; Kleiner, N.: **Interactive Workflow Mining**. In: Desel, J.; Pernici, B.; Weske, M. (Hrsg.): Business Process Management. Proc. Second International Conference, BPM 2004, Potsdam, June 17-18, 2004. Berlin (Springer) 2004, S. 211-226.

HäLe05 Härder, T.; Lehner, W. (Hrsg.): **Data Management in a Connected World.** Essays Dedicated to Hartmut Wedekind on the Occasion of His 70th Birthday (Festschrift). Berlin (Springer) 2005.

Hanc12 Hancock, M.F.: **Practical Data Mining**. Boca Raton (CRC Press) 2012.

Hand15 Hand, D.J.: **Statistics and computing: the genesis of data science**. In: Statistics and Computing 25 (2015), S. 705–711. URL: https://link.springer.com/article/10.1007/s11222-015-9565-6 (Abruf am 09. 12.2016).

Hand94 Hand, D.J.: **Deconstructing Statistical Questions**. In: Journal of the Royal Statistical Society, Series A, 157 (1994) 3, S. 317-356.

Hand94b Hand, D.J.: **Statistical Strategy: Step 1**. In: Cheeseman, P.; Olford, R.W. (Eds.): Selecting Models from Data: Artificial Intelligence and Statistics IV. Lecture Notes in Statistics, Vol 89. Berlin (Springer) 1994, S. 3-9.

Hand99 Hand, D.J.: **Introduction**. In: [BeHa99], S. 1-14.

HaSt98 Hagemeyer, J.; Striemer, R.: **Anforderungen an die Erweiterung von Metamodellen für die Geschäftsprozessmodellierung und das Workflow Management**. In: [HeSW98], S. 161-180.

HaSW98a Hammel, C.; Schlitt, M.; Wolf, S.: **Pattern-basierte Konstruktion von Unternehmensmodellen**. In: Informationssystem-Architekturen 5 (1998) 1, S. 22-37.

Haus90 Hauschildt, J.: **Komplexität, Zielbildung und Effizienz von Entscheidungen in Organisationen**. In: [FiBo90], S. 131-147.

Haux86 Haux, R. (Hrsg.): **Expert Systems in Statistics**, Stuttgart (Fischer) 1986.

HeHi01 Hettich, S.; Hippner, H.: **Assoziationsanalyse**. In: [HKMW01], S. 427-463.

Hein+08 Heinrich, B.; Bewernik, M.-A.; Henneberger, M.; Krammer, A.: **SEMPA – Ein Ansatz des Semantischen Prozessmanagements zur Planung von Prozessmodellen**. In: Wirtschaftsinformatik 50 (2008) 6, S. 445-460.

Hein91 Heinen, E.: **Industriebetriebslehre als entscheidungsorientierte Unternehmensführung**. In: Heinen, E. (Hrsg.): Industriebetriebslehre. Entscheidungen im Industriebetrieb. 9. Aufl., Wiesbaden (Gabler) 1991, S. 1-71.

HeKa04 Herbst, J.; Karagiannis, D.: **Workflow Mining with InWoLvE**. In: Computers in Industry 53 (2004) 3, S. 245-264.

HeKZ11 Heinrich, B.; Klier, M.; Zimmermann. S.: **Automatisierte Modellierung, Umsetzung und Ausführung von Prozessen. Ein Web Service-basiertes Konzept**. In: Bernstein, A.; Schwabe, G. (Hrsg.): Proc. 10th International Conference on Wirtschaftsinformatik, 16.-18. Februar 2011, Zürich. Volume 1. URL: https://files.ifi.uzh.ch/WI2011/Volume1_WI2011_Proceedings.pdf (Abruf am 08.04.2011).

HeMi94 Heiler, S.; Michels, P.: **Deskriptive und Explorative Datenanalyse** (Lehr- und Handbücher der Statistik). München (Oldenbourg) 1994.

Henn34 Henning, K.W.: **Einführung in die betriebswirtschaftliche Organisationslehre**, Berlin 1934.

Hert89 Hertzberg, J.: **Planen. Einführung in die Planerstellungsmethoden der künstlichen Intelligenz**. Mannheim (BI-Wissenschaftsverlag) 1989.

HeSW98 Herrmann, T.; Scheer, A.-W.; Weber, H. (Hrsg.): **Verbesserung von Geschäftsprozessen mit flexiblen Workflow-Management-Systemen 1: Von der Erhebung zum Sollkonzept.** Veröffentlichungen des Forschungsprojekts MOVE. Heidelberg (Physica) 1998.

HiKa01 Hilario, M.; Kalousis, A.: **Fusion of Meta-knowledge and Meta-data for Case-Based Model Selection**. In: [RaSi01], S. 180-191.

Hila+11 Hilario, M.; Nguyen, P.; Do, H.; Woznica, A.; Kalousis, A.: **Ontology-Based Meta-Mining of Knowledge Discovery Workflows**. In: [JaDG11], 273-315.

HiLK10 Hilario, M.; Lavrač, N.; Kok, J.N. (Hrsg.): **Proc. Third International Workshop on Third-Generation Data Mining: Towards Service-Oriented Knowledge Discovery (SoKD-10)**. Workshop im Rahmen der European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2010), Barcelona (Spanien), 24. September 2010. URL: http://cui.unige.ch/~hilario/sokd10/sokd10-proceedings.pdf (Abruf am12.10.2011).

HiWi01 Hippner, H.; Wilde, K.D.: **Der Prozess des Data Mining im Marketing**. In: [HKMW01], S.21-91.

HKMW01 Hippner, H.; Küsters, U.; Meyer, M.; Wilde K.D. (Hrsg.): **Handbuch Data Mining im Marketing. Knowledge Discovery in Marketing Databases**. Braunschweig (Vieweg Gabler) 2001.

HKNW09 Hilario, M.; Kalousis, A.; Nguyen, P.; Woznica, A.: **A Data Mining Ontology for Algorithm Selection and Meta-Mining**. In: [PLKB09], S. 76-87.

HMPU97 Heckerman, D.; Mannila, H.; Pregibon, D.; Uthurusamy, R. (Hrsg.): **Proc. Third International Conference on Knowledge Discovery and Data Mining** (Newport Beach, California, USA, 14.-17. August 1997). Menlo Park (AAAI Press) 1997.

HMSS01 Hotho, A.; Mädche, A.; Staab, S.; Studer, R.: **Seal-II. The soft spot between richly structured and unstructured knowledge**. In: Journal of Universal Computer Science 7 (2001) 7, S. 566-590.

HND+11 Hilario, M.; Nguyen, P.; Do, H.; Woznica, A.; Kalousis A.: **Ontology-Based Meta-Mining of Knowledge Discovery Workflows**. In: Jankowski, N.; Duch, W.; Grabczewski, K. (Hrsg.): Meta-Learning in Computational Intelligence, Berlin (Springer) 2011, S. 273-315. URL: http://www.dmo-foundry.org/sites/default/ files/sky/hilario-etal-11.pdf (Abruf am 17.04.2012).

Hogl03 Hogl, O.M.J.: **Eine wissensbasierte Benutzerschnittstelle für das Invisible Data Mining**. Dissertation, Univ. Erlangen-Nürnberg 2003.

HoKl91 Hoschka, P.; Klösgen, W.: **A Support System For Interpreting Statistical Data**. In: Piatetsky-Shapiro, G.; Frawley, W.J. (Hrsg.): Knowledge Discovery in Databases. Menlo Park (AAAI Press) 1991, S. 326-345.

HoLP14 Holsapple, C.; Lee-Post, A.; Pakath, R.: **A unified foundation for business analytics**. In: Decision Support Systems 64 (2014), S. 130-141.

HoMP01 Hopfenbeck, W.; Müller, M.; Peisl, T.: **Wissensbasiertes Management. Ansätze und Strategien zur Unternehmensführung in der Internet-Ökonomie**. Landsberg/Lech (Moderne Industrie) 2001.

HoSc10 Holling, H.; Schmitz, B. (Hrsg.): **Handbuch Statistik, Methoden und Evaluation**. Handbuch der Psychologie, Band 13. Göttingen (Hogrefe) 2010.

HsKn95 Hsu, C.-N.; Knoblock, C.A.: **Estimating the Robustness of Discovered Knowledge**. In: [FaUt95], S. 156-161.

Hube11 Huber, P.J.: **Data Analysis. What Can Be Learned from the Past 50 Years**. Hoboken (Wiley) 2011.

Hube97 Huber, P.J.: **Speculations on the Path of Statistics**. In: [BrFM97], S. 175-191.

Hump86 Humphreys, P.: **Intelligence in decision support**. In: Bremer, B.; Jungermann, H.; Lourens, P.; Sevón, G. (Hrsg.): New directions in research on decision making. Amsterdam (North Holland) 1986, S. 333-361.

HuZi11 Hussein, T.; Ziegler, J.: **Situationsgerechtes Recommending**. In: Informatik-Spektrum 34 (2011) 2, S. 143-152.

HwWY04 Hwang, S.-Y.; Wei, C.-P.; Yang, W.-S.: **Discovery of Temporal Patterns from Process Instances**. In: Computers in Industry 53 (2004) 3, S. 345-364.

ImMa96 Imielinski, T.; Mannila, H.: **A Database Perspective on Knowledge Discovery**. In: Communications of the ACM 39 (1996) 11, S. 58-64.

InJä05 Ingwersen, P.; Järvelin, K.: **The Turn: Integration of Information Seeking and Retrieval in Context**. Berlin (Springer) 2005.

Isse07 Isselhorst, T.: **Modellierung von Kontext für Führungsinformationssysteme**. Duisburg (WiKu-Verlag) 2007. Zugl.: Diss., Univ. Bamberg 2006.

Jabl00 Jablonski, S.: **Workflow Management Between Formal Theory and Pragmatic Approaches**. In: [AaDO00], S. 345-358.

Jabl05 Jablonski, S.: **Processes, Workflows, Web Service Flows: A Reconstruction**. In: [HäLe05], S. 201-213.

JaBS97 Jablonski, S.; Böhm, M.; Schulze, W. (Hrsg.): **Workflow-Management. Entwicklung von Anwendungen und Systemen. Facetten einer neuen Technologie**. Heidelberg (dpunkt) 1997.

Jaco91 Jacoby, W.G.: **Data Theory and Dimensional Analysis**. Sage University Paper Series on Quanitative Applications in the Social Sciences, 07-078. Newbury Park (Sage Publications) 1991.

JaDG11 Jankowski, N.; Duch, W.; Grabczewski, K. (Hrsg.): **Meta-Learning in Computational Intelligence**. Studies in Computational Intelligence, Vol. 358. Berlin (Springer) 2011.

JaVW00 Janssens, G.K.; Verelst, J.; Weyn, B.: **Techniques for Modelling Workflows and Their Support of Reuse**. In: [AaDO00], S. 1-15.

Jeck97 Jeck, S.: **Planen und das Lösen von Alltagsproblemen**. Lengerich (Pabst Science Publishers) 1997. Zugl.: Mainz, Univ., Diss., 1997.

John97 John, G.: **Enhancements to the Data Mining Process**. Doctoral Dissertation, Department of Computer Science, University of Stanford 1997.

JoLe12 Johannsen, F.; Leist, S.: **Das Dekompositionsmodell nach Wand und Weber im Kontext der Prozessmodellierung**. In: Wirtschaftsinformatik 54 (2012) 5, S. 263-280.

Jung02 Jung, B.: **Prozessmanagement in der Praxis. Vorgehensweisen, Methoden, Erfahrungen**. Köln (TÜV-Verlag) 2002.

JuWi00 Jung, R.; Winter, R. (Hrsg.): **Data Warehousing 2000. Methoden, Anwendungen, Strategien** (Proc. Data Warehousing 2000, Friedrichshafen, 14./15. November 2000). Heidelberg (Physica) 2000.

Kafk99 Kafka, C.: **Konzeption und Umsetzung eines Leitfadens zum industriellen Einsatz von Data-Mining**. Diss., Univ. Karlsruhe 1999.

Kann10 Kannegiesser, U.: **Towards a Methodology for Flexible Process Specification**. In: Enterprise Modelling and Information Systems Architectures 5 (2010) 3, S. 44-63.

KaNo96 Kaplan, R.S.; Norton, D.P.: **The Balanced Scorecard. Translating Strategy into Action**. Boston (Harvard Business School Press) 1996.

KaRe91 Kappler, E.; Rehkugler, H.: **Kapitalwirtschaft**. In: Heinen, E. (Hrsg.): Industriebetriebslehre. Entscheidungen im Industriebetrieb. 9. Aufl., Wiesbaden (Gabler) 1991, S. 897-1068.

KCC+02 Kim, W.; Chae, K.; Cho, D.; Choi, B.; Jeong, A.; Kim, M.; Lee, K.; Lee, M.; Lee, S.; Park, S.; Yong, H.; Kim, H.; Lee, J.; Lee, W.: **The Chamois Reconfigurable Data-Mining Architecture**, URL: http://www.jot.fm/issues/issue_2002_07/column2, 2002 (Abruf am 05.08.2003).

KCHK+03 Kim, W.; Choi, B.-J.; Hong, E.-K.; Kim, S.-K.; Lee, D.: **A Taxonomy of Dirty Data**. In: Data Mining and Knowledge Discovery (2003) 7, S. 81-99.

KDnu14 o.V.: KDnuggets Polls: **What main methodology are you using for your analytics, data mining, or data science projects?** URL: http://www.kdnuggets.com/polls/2014/analytics-data-mining-data-science-methodology.html, 2014 (Abruf am 08.01.2015).

KDnu15 o.V.: KDnuggets Polls: **When will most expert-level Predictive Analytics/Data Science tasks be automated?** URL: http://www.kdnuggets.com/polls/2015/analytics-data-science-automation-future.html, 2015 (Abruf am 03.03.2017).

KeBa06 Kemper, H.-G.; Baars, H.: **Business Intelligence und Competitive Intelligence. IT-basierte Managementunterstützung und markt-/wettbewerbsorientierte Anwendungen**. In: HMD – Praxis der Wirtschaftsinformatik (2006) 247, S. 7-20.

KeFi98 Kemper, H.-G.; Finger, R.: **Datentransformation im Data Warehouse. Konzeptionelle Überlegungen zur Filterung, Harmonisierung, Verdichtung und Anreicherung operativer Datenbestände**. In: [ChGl98a], S. 61-77.

Kien82 von Kienle, R: **Fremdwörter-Lexikon**. Hamburg (Xenos) 1982.

KiZV00 Kietz, J.U.; Zücker, R.; Vaduva, A.: **MINING MART: Combining Case-based Reasoning and Multistrategy Learning into a Framework for Reusing KDD Applications**. In: Michalski, R.S.; Brazdil, P. (Hrsg.): Proc. 5th Intl. Workshop on Multistrategy Learning (MSL 2000), Guimaraes, Portugal, June 2000, S. 151-163.

KlPR98b Kleinberg. J.; Papadimitriou, C.; Raghavan, P.: **A Microeconomic View on Data Mining**. In: Data Mining and Knowledge Discovery (1998) 2, S. 311-324.

KlRo88 Klatt, E.; Roy, D.: **Langenscheidts Taschenwörterbuch Englisch**. Erster Teil. Englisch-Deutsch. 18. Aufl., Berlin (Langenscheidt) 1988.

KlTH13 Klein, D.; Tran-Gia, P.; Hartmann, M.: **Big Data**. Aktuelles Schlagwort. In: Informatik-Spektrum 36 (2013) 3, S. 319-323.

KlZy96 Klösgen, W.; Žytkow, J.M.: **Knowledge Discovery in Databases Terminology**. In: [FPSU96], S. 573-592.

Knob01 Knobloch, B.: **Der Data-Mining-Ansatz zur Analyse betriebswirtschaftlicher Daten**. In: Informationssystem-Architekturen 8 (2001) 1, S. 59-116.

Knob02 Knobloch, B.: **Ein Bezugsrahmen für integrierte Managementunterstützungssysteme. Einordnung und funktionale Anforderungen an Business-Intelligence-Systeme aus managementtheoretischer Sicht**. In: [MaWi02], S. 335-355.

Knob03a Knobloch, B.: **A Framework for Organizational Data Analysis and Organizational Data Mining. From Business Objectives to Steps of Action**. In: Nemati, H.; Barko, C. (Hrsg.): Organizational Data Mining. Leveraging Enterprise Data Resources for Optimal Performance. Hershey (IDEA Group Publ.) 2003, S. 334-356.

Knob07 Knobloch, B.: **Prozessmodelle der Datenanalyse**. Diskussionspapier, Systemberatung für Wirtschaftsinformatik, Ködnitz 2007. URL: http://www.bernd-knobloch.de/doc/publ/5-prozessmodelle-2007 (Abruf am 28.01.2015).

KnSB00 Knobbe, A.; Schipper, A.; Brockhausen, P.: **Domain Knowledge and Data Mining Process Decisions**. Mining Mart Project (IST-1999-11993), Deliverable No. D 5, o.O., June 23, 2000. URL: http://www-ai.cs.uni-dortmund.de/dokumente/knobbe_etal_2000b.pdf (Abruf am 22.05.2004).

KnWe00 Knobloch, B.; Weidner, J.: **Eine kritische Betrachtung von Data Mining-Prozessen. Ablauf, Effizienz und Unterstützungspotenziale**. In: [JuWi00], S. 345-365.

Koch00 Koch, K.-R.: **Einführung in die Bayes-Statistik**. Berlin (Springer) 2000.

KöKe00 Köpf, C.; Keller, J.: **Meta-Analysis: From Data Characterization for Meta-Learning to Meta-Regression**. In: PKDD-2000 Workshop on Data Mining, Decision Support, Meta-Learning and ILP, 2000.

KoLe04 Kossmann, D.; Leymann, F.: **Web Services**. In: Informatik-Spektrum 27 (2004) 2, S. 117-128.

Kore90 Koreimann, D.S.: **Strategien zur Komplexitätsreduzierung**. In: [FiBo90], S. 283-297.

KoRS02 Kohavi, R.; Rothleder, N.J.; Simoudis, E.: **Emerging Trends in Business Analytics**. In: Communications of the ACM 45 (2002) 8, S. 45-48.

KoZy97 Komorowski, H.J.; Żytkow, J.M. (Hrsg.): **Proc. First European Symposium on Principles of Data Mining and Knowledge Discovery** (June 24-27, 1997). LNCS, Band 1263. London (Springer) 1997.

KrBü11 Krämer, W.; Bücker, M.: **Probleme des Qualitätsvergleichs von Kreditausfallprognosen**. In: AStA Wirtschafts- und Sozialstatistisches Archiv (2011) 5, S. 39-58.

Krie+07 Kriegel, H.-P.; Borgwardt, K.M.; Kröger, P.; Pryakhin, A.; Schubert, M.; Zimek, A.: **Future Trends in Data Mining**. In: Data Min Knowl Disc 15 (2007), S. 87-97.

Krue+10 Krueger, J.; Grund, M.; Tinnefeld, C.; Eckart, B.; Zeier, A.; Plattner, H.: **Hauptspeicherdatenbanken für Unternehmensanwendungen. Datenmanagement für Unternehmensanwendungen im Kontext heutiger Anforderungen und Trends**. In: Datenbank-Spektrum 10 (2010) 3, S. 143-158.

KrWZ98 Krahl, D.; Windheuser, U.; Zick, F.-K.: **Data Mining: Einsatz in der Praxis**. Bonn (Addison-Wesley) 1998.

KSBF09 Kietz, J.-U.; Serban, F.; Bernstein, A.; Fischer, S.: **Towards Cooperative Planning of Data Mining Workflows**. In: [PLKB09], 1-12.

KSBF10 Kietz, J.-U.; Serban, F.; Bernstein, A.; Fischer, S.: **Data Mining Workflow Templates for Intelligent Discovery Assistance and Auto-Experimentation**. In: [HiLK10], S. 1-12.

KüBe01 Küsters, U.; Bell, M.: **Zeitreihenanalyse und Prognoseverfahren: Ein methodischer Überblick über klassische Ansätze**. In: [HKMW01], S. 255-297.

Kuhn90 Kuhn, A.: **Unternehmensführung**. 2. Aufl., München (Vahlen) 1990.

KüKa01 Küsters, U.; Kalinowski, C.: **Traditionelle Verfahren der multivariaten Statistik**. In: [HKMW01], S.131-192.

KuKi99 Kulkarni, J.; King, R.: **Business Intelligence-Systeme und Data Mining: Grundlage für strategische Entscheidung**. SAS Institute White Paper. o.O. (SAS Institut Deutschland) 1999.

Küpp05 Küpper, H.-U.: **Controlling. Konzeption, Aufgaben, Instrumente**. 4. Aufl., Stuttgart (Schäffer-Poeschel) 2005.

Küpp99 Küppers, B.: **Data Mining in der Praxis. Ein Ansatz zur Nutzung der Potentiale von Data Mining im betrieblichen Umfeld**. Frankfurt/M. (Lang) 1999.

Küst01 Küsters, U.: **Data Mining Methoden: Einordnung und Überblick**. In: [HKMW01], S. 95-130.

Kuts03 Kutschker, M.: **Prozessmanagement von Kooperationen**. Diskussionsbeiträge der Wirtschaftswissenschaftlichen Fakultät Ingolstadt Nr. 165, Ingolstadt 2003.

KZTL08 Kunze, C.; Zaplata, S.; Turjalei, M.; Lamersdorf, W.: **Enabling Context-Based Cooperation: A Generic Context Model and Management System**. In: Abramowicz, W.; Fensel, D. (Hrsg.): Proc. 11th International Conference on Business Information Systems (BIS 2008), Innsbruck, 5.-7. Mai 20078. LNBIP 7, Berlin (Springer) 2008, S. 459-470.

LaPh94 Lansky, A.L.; Philpot, A.G.: **AI-Based Planning for Data Analysis**. In: IEEE Expert 9 (Februar 1994) 1, S. 21-27.

LaRe08 Lang, A.; Reinwald, B.: **Nutzung unstrukturierter Daten für Business Intelligence**. In: Datenbank-Spektrum 8 (2010) 25, S. 12-22.

LaSi82 Launer, R.L.; Siegel, A.F. (Hrsg.): **Modern Data Analysis**. New York (Academic Press) 1982.

LaSW97 Langner, P.; Schneider, C.; Wehler, J.: **Prozeßmodellierung mit ereignisgesteuerten Prozeßketten (EPKs) und Petri-Netzen**. In: Wirtschaftsinformatik 39 (1997) 5, S. 479-489.

LeHM95 Lehner, F.; Hildebrand, K.; Maier, R.: **Wirtschaftsinformatik: Theoretische Grundlagen**. München (Oldenbourg) 1995.

LeVo10 Lessmann, S.; Voß, S.: **Unterstützung kundenbezogener Entscheidungsprobleme. Eine Analyse zum Potenzial moderner Klassifikationsverfahren**. In: Wirtschaftsinformatik (2010) 2, S. 79-93.

LiBa90 Liebenau, J.; Backhouse, J.: **Understanding Information. An Introduction**. Houndmills (Macmillan) 1990.

LiBe11 Linoff, G.S.; Berry, M.J.A.: **Data Mining Techniques. For Marketing, Sales, and Customer Relationship Management**. 3. Aufl., Indianapolis (Wiley) 2011.

Lieb96 Liebl, F.: **Strategische Frühaufklärung: Trends, Issues, Stakeholders**. München (Oldenbourg) 1996.

LiLS00 Lim, T.-S.; Loh, W.-Y.; Shih, Y.-S.: **A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-Three Old and New Classification Algorithms**. In: Machine Learning 40 (2000) 3, S. 203-228. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.44.7213&rep=rep1&type=pdf (Abruf am 01.09.2011).

Lind05 Lindner, G.: **Algorithmenauswahl im KDD-Prozess**. Diss., Karlsruhe, Univ., 2005. URL: http://www.ubka.uni-karlsruhe.de/vvv/2005/wiwi/6/6.pdf (Abruf am 27.09.2005).

LiSe00 Lieberman, H.; Selker, T.: **Out of Context: Computer Systems that Adapt to, and Learn from, Context**. IBM Systems Journal 39 (2000) 3-4, S. 617-632.

LiSt99 Lindner, G.; Studer, R.: **AST: Support for Algorithm Selection with a CBR Approach**. In: [ZyRa99], S. 418-423.

Liu99 Liu, X.: **Systems and Applications**. In: [BeHa99], S. 351-364.

LJWK12 Ley, T.; Jurisch, M.; Wolf, P.; Krcmar, H.: **Kriterien zur Leistungsbeurteilung von Prozessen: Ein State-of-the-Art**. In: Mattfeld, D.C.; Robra-Bissantz, S. (Hrsg.): Multikonferenz Wirtschaftsinformatik 2012. Tagungsband der MKWI 2012. Berlin (GITO Verlag) 2012.

LoDi04 Lockemann, P.C.; Dittrich, K.R.: **Architektur von Datenbanksystemen**. Heidelberg (dpunkt) 2004.

LPDK16 Lueth, K.L.; Patsioura, C.; Diaz Williams, Z.; Kermani, Z.Z.: **Industrial Analytics 2016/2017. The current state of data analytics usage in industrial companies**. Sponsored Report, December 2016. o.O. (IoT Analytics GmbH) 2016.

Luhm80 Luhmann, N.: **Komplexität**. In: Grochla, E. (Hrsg.): Handwörterbuch der Organisation (Enzyklopädie der Betriebswirtschaftslehre, Band II). 2. Aufl., Stuttgart (Poeschel) 1980, S. 1064-1070.

Lyre02 Lyre, H.: **Informationstheorie. Eine philosophisch-naturwissenschaftliche Einführung**. München (Fink) 2002.

MaCr94 Malone, T.W.; Crowston, K.: **The Interdisciplinary Study of Coordination**. In: ACM Computing Surveys (1994) 3, S. 87-119.

Mali00 Malik, F.: **Strategie des Managements komplexer Systeme. Ein Beitrag zur Management-Kybernetik evolutionärer Systeme**. 6. Aufl., Bern (Haupt) 2000. Zugl.: St. Gallen, Univ., Habil.-Schrift, 1977.

Mart98 Martin, W.: **Data Warehouse, Data Mining und OLAP: Von der Datenquelle zum Informationsverbraucher**. In: [Mart98a], S. 19-37.

Mart98a Martin, W. (Hrsg.): **Data Warehousing, Data Mining, OLAP**. Bonn (Addison-Wesley) 1998.

Marx12 Marx Gómez, J.C.: **Serviceorientierte Architektur**. In: Gronau, N.; Becker, J.; Sinz, E.J.; Suhl, L.; Leimeister, J.M. (Hrsg.): Enzyklopädie der Wirtschaftsinformatik, Online-Lexikon. URL: http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/is-management/Systementwicklung/Softwarearchitektur/Architekturparadigmen/Serviceorientierte-Architektur/index.html, letzte Änderung: 23.08.2012 (Abruf am 30.01.2017).

MäSS01 Mädche, A.; Staab, S.; Studer, R.: **Ontologien**. WI-Schlagwort. In: Wirtschaftsinformatik 43 (2001) 4, S. 393-395.

MaWi02 von Maur, E.; Winter, R. (Hrsg.): **Vom Data Warehouse zum Corporate Knowledge Center**. Proc. Data Warehousing 2002. Heidelberg (Physica) 2002.

MaWi03 von Maur, E.; Winter, R. (Hrsg.): **Data Warehouse Management. Das St. Galler Konzept zur ganzheitlichen Gestaltung der Informationslogistik**. Berlin (Springer) 2003.

MeGr00 Mertens, P., Griese, J.: **Integrierte Informationsverarbeitung, Band 2: Planungs- und Kontrollsysteme in der Industrie**. 8. Aufl., Wiesbaden (Gabler) 2000.

Mert01 Mertens, P. (Hrsg.): **Lexikon der Wirtschaftsinformatik**. 4. Aufl., Berlin (Springer) 2001.

Meta01 o.V.: **MetaL: A Meta-Learning Assistant for Providing User Support in Machine Learning and Data Mining**. ESPRIT METAL Project (26.357), Dec. 1998-Nov. 2001. http://www.metal-kdd.org/ (Abruf am 06.02.2002).

MiCD13 Minelli, M.; Chambers, M.; Dhiraj, A.: **Big Data, Big Analytics. Emerging Business Intelligence and Analytic Trends for Today's Businesses**. Hoboken (Wiley) 2013.

Milt10 Milton, M.: **Datenanalyse von Kopf bis Fuß**. Köln (O'Reilly) 2010.

MiMM95 Mili, H.; Mili, F.; Mili, A.: **Reusing Software. Issues and research directions**. In: IEEE Transactions on Software Engineering 21 (1995) 6, S. 528-561.

Mint79 Mintzberg, H.: **The Structuring of Organizations**, Englewood Cliffs 1979.

MiRe11 Mikut, R.; Reischl, M.: **Data Mining Tools. Advanced Review**. In: WIREs Data Mining and Knowledge Discovery 1 (September/Oktober 2011), S. 431-443.

MiST94 Michie, D.; Spiegelhalter, D.J.; Taylor, C.C. (Hrsg.): **Machine Learning, Neural and Statistical Classification**. Boston (Ellis Horwood) 1994.

MKFR03 Mierswa, I.; Klinkenberg, R.; Fischer, S.; Ritthoff, O.: **A Flexible Platform for Knowledge Discovery Experiments: YALE – Yet Another Learning Environment**. In: LLWA 03 – Tagungsband der GI-Workshop-Woche Lernen, Lehren, Wissen, Adaptivität. Dortmund 2003. URL: http://rapid-i.com/component/option,com_docman/task,doc_download/gid,9/lang,en/ (Abruf am 29.08.2006).

MNL+80 Mills, H.D.; O'Neill, D.; Linger, R.C.; Dyer, M.; Quinnan, R.E.: **The management of software engineering**. In: IBM Systems Journal 19 (1980) 4, S. 414-477.

Morr38 Morris, C.: **Foundations of the Theory of Science**. In: Neurath, O.; Carnap, R.; Morris, C. (Hrsg.): International Encyclopedia of Unified Science. Chicago (The University of Chicago Press) 1938.

MoSc04 Morik, K.; Scholz, M.: **The MiningMart Approach to Knowledge Discovery in Databases**. In: Zhong, N.; Liu, J. (Hrsg.): Intelligent Technologies for Information Analysis. Berlin (Springer) 2004, S. 47-65.

MoSE03 Morik, K.; Scholz, M.; Euler, T.: **MiningMart: Final Report**. Mining Mart Project (IST-1999-11993), Deliverable No. D 20.4. April 28, Dortmund 2003. URL: http://www-ai.cs.uni-dortmund.de/dokumente/morik_etal_2003a.pdf (Abruf am 22.05.2004).

MSMF09 Marbán, O.; Segovia, J.; Menasalvas, E.; Fernández-Baizán, C.: **Toward data mining engineering: A software engineering approach**. In: Information Systems 34 (2009), S. 87-107.

Mueh01 zur Muehlen, M.: **Process-driven Management Information Systems. Combining Data Warehouses and Workflow Technology**. In: Gavin, B. (Hrsg.): Proc. Fourth International Conference on Electronic Commerce Research (ICECR-4), Dallas 2001, S. 550-566.

Müll00 Müller, J.: **Transformation operativer Daten zur Nutzung im Data Warehouse**. Wiesbaden (DUV) 2000. Zugl.: Bochum, Univ., Diss., 1999.

Muns11 Munson, M.A.: **A Study on the Importance of and Time Spent on Different Modeling Steps**. In: SIGKDD Explorations 13 (2011) 2, S. 65-71.

MWK+06 Mierswa, I.; Wurst, M.; Klinkenberg, R.; Scholz, M.; Euler, T.: **YALE: Rapid Prototyping for Complex Data Mining Tasks**. In: Proc. 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2006). Philadelphia (ACM Press) 2006, S. 935-940. URL: http://www-ai.cs.uni-dortmund.de/dokumente/mierswa_etal_2006a.pdf (Abruf am 18.11.2010).

NaSA03 Nauck, D.; Spott, M.; Azvine, B.: **SPIDA: a novel data analysis tool**. In: BT Technology Journal 21 (October 2003) 4, S. 104-112.

Neck02 Neckel, P.: **Konzeption, Erstellung und Nutzung von Kundenprofilen aus Transaktionsdaten am Beispiel des Lebensmitteleinzelhandels**. Diplomarbeit, Fakultät Wirtschaftsinformatik und Angewandte Informatik, Otto-Friedrich-Universität Bamberg 2002.

Neck07 Neckel, P.: **Self-Acting Data Mining. Das neue Paradigma der Datenanalyse**. Whitepaper, Berlin (mayato GmbH) 2007. URL: https://www.mayato.com/wp-content/uploads/2015/03/mayato_Whitepaper_S-ADM_11.07.pdf (Abruf am 03.03. 2017).

NeKn06 Neckel, P.; Knobloch, B.: **Systematisches Customer Relationship Analytics im Einzelhandel**. In: HMD – Praxis der Wirtschaftsinformatik (Februar 2006) 247 (Schwerpunktheft „Business & Competitive Intelligence"), S. 94-104.

NeKn15 Neckel, P.; Knobloch, B.: **Customer Relationship Analytics. Praktische Anwendung des Data Mining im CRM**. 2. Aufl., Heidelberg (dpunkt) 2015.

Newe82 Newell, A.: **The Knowledge Level**. In: Artificial Intelligence 18 (1992), S. 87-127.

NiPi95 Nippa, M.; Picot, A. (Hrsg.): **Prozeßmanagement und Reengineering. Die Praxis im deutschsprachigen Raum**. Frankfurt (Campus) 1995.

Nipp95 Nippa, M.: **Anforderungen an das Management prozeßorientierter Unternehmen**. In: [NiPi95], S. 39-58.

NiSV14 Nica, A.; Suchanek, F.M.; Varde, A.S.: **New Research Directions in Knowledge Discovery and Allied Spheres**. In: SIGKDD Explorations 16 (2014) 2, S. 46-49. URL: http://www.kdd.org/exploration_files/Volume16-Issue2.pdf (Abruf am 04.11.2016).

Nord31 Nordsieck, F.: **Grundprobleme und Grundprinzipien der Organisation des Betriebsaufbaus**. In: Die Betriebswirtschaft 24 (1931), S. 158-162.

Nord34 Nordsieck, F.: **Grundlagen der Organisationslehre**, Stuttgart 1934.

OMG15 OMG Inc. (Hrsg.): **Knowledge Discovery Metamodel (KDM).** URL: http://www.omg.org/technology/kdm/index.htm. Zuletzt aktualisiert: 27.05.2015. (Abruf am 07.02.2017).

Oppe95 Oppelt, R.U.G.: **Computerunterstützung für das Management. Neue Möglichkeiten der computerbasierten Informationsunterstützung oberster Führungskräfte auf dem Weg von MIS zu EIS?** München (Oldenbourg) 1995.

OpSc84 Opitz, O.; Schader, M.: **Zur Entwicklung der qualitativen Datenanalyse**. Arbeitspapiere zur mathematischen Wirtschaftsforschung, Heft 73/1984. Augsburg (Institut für Statistik und Mathematische Wirtschaftstheorie der Universität Augsburg) 1984.

Orr98 Orr, K.: **Data Quality and Systems Theory**. **One Certain Way to Improve the Quality of Data: Improve its Use!** In: Communications of the ACM 41 (1998) 2, S. 66-71

Öste+10 Österle, H.; Becker, J.; Frank, U.; Hess, T.; Karagiannis, D.; Krcmar, H.; Loos, P.; Mertens, P.; Oberweis, A.; Sinz, E.J.: **Memorandum zur gestaltungsorientierten Wirtschaftsinformatik**. In: [ÖsWB10], S. 1-6.

ÖsWB10 Österle, H.; Winter, R.; Brenner, W. (Hrsg.): **Gestaltungsorientierte Wirtschaftsinformatik. Ein Plädoyer für Rigor und Relevanz**. St. Gallen (infowerk AG) 2010.

OuPe06 Ou, L.; Peng, H.: **XML and Knowledge Based Process Model Reuse and Management in Business Intelligence System**. In: [SLL+06], S. 117-121.

Oxfo17 Oxford University Press (Hrsg.): **English Oxford Living Dictionaries. Definition of analytics in English**. URL: https://en.oxforddictionaries.com/definition/analytics (Abruf am 18.01.2017).

Oxfo17b Oxford University Press (Hrsg.): **English Oxford Living Dictionaries. Definition of process in English**. URL: https://en.oxforddictionaries.com/definition/process (Abruf am 18.01.2017).

PaRa14 Patzak, G.; Rattay, G.: **Projektmanagement. Projekte, Projektportfolios, Programme und projektorientierte Unternehmen**. 6. Aufl., Wien (Linde) 2014.

PCTM02 Poole, J.; Chang, D.; Tolbert, D.; Mellor, D.: **Common Warehouse Metamodel**. New York (Wiley) 2002.

PCTM03 Poole, J.; Chang, D.; Tolbert, D.; Mellor, D.: **Common Warehouse Metamodel Developer's Guide**. Indianapolis (Wiley) 2003.

Pern02 Perner, P. (Hrsg.): **Advances in Data Mining. Applications in E-Commerce, Medicine, and Knowledge Management**. Lecture Notes in Artificial Intelligence 2394, Berlin (Springer) 2002.

Pete03 Petersohn, H.: **Data Mining. Verfahren, Prozeß, Anwendungsarchitektur**. Habilitationsschrift, Univ. Leipzig 2003.

Pete04 Petersohn, H.: **Data-Mining-Anwendungsarchitektur**. In: Wirtschaftsinformatik 46 (2004) 1, S. 15–21

Pete95 Peters, G.: **Entwicklung von Verfahren zur Intelligenten Datenanalyse und ihre Anwendung auf die Prognose ökonomischer Daten**. Diss., RWTH Aachen, Aachen 1995.

Petr00 Petrak, J.: **Fast Subsampling Performance Estimates for Classification Algorithm Selection**. Technical Report TR-2000-07. Wien (Austrian Research Institute for Artificial Intelligence) 2000.

PfBG00 Pfahringer, B.; Bensusan, H.; Giraud-Carrier, C.: **Meta-Learning by Landmarking various Learning Algorithms**. In: Proc. Seventeenth International Conference on Machine Learning, ICML'2000, San Francisco (California), June 2000, New York (Morgan Kaufmann) 2000, S. 743-750.

PiFr95 Picot, A.; Franck, E.: **Prozeßorganisation. Eine Bewertung der neuen Ansätze aus Sicht der Organisationslehre**. In: [NiPi95], S. 13-38.

PiRe84; Picot, A.; Reichwald, R.: **Bürokommunikation: Leitsätze für den Anwender**. 2. Aufl., München (CW-Publikationen) 1984.

PiRe91 Picot, A.; Reichwald, R.: **Informationswirtschaft**. In: Heinen, E. (Hrsg.): Industriebetriebslehre. Entscheidungen im Industriebetrieb. 9. Aufl., Wiesbaden (Gabler) 1991, S. 241-393.

PLKB09 Podpečan, V.; Lavrač, N.; Kok, J.N.; de Bruin, J. (Hrsg.): **Proc. International Workshop on Third-Generation Data Mining: Towards Service-Oriented Knowledge Discovery (SoKD-09)**. Workshop im Rahmen der European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2009), Bled (Slowenien), 7. September 2009. URL: http://www.ecmlpkdd2009.net/wp-content/uploads/2008/09/service-oriented-knowledge-discovery_2.pdf (Abruf am 12.10.2011).

Popp82 Popper, K.R.: **Logik der Forschung**. 7. Aufl., Tübingen (Mohr) 1982.

Port85 Porter, M.E.: **Competitive Advantage. Creating and Sustaining Superior Performance**. New York 1985.

PrES04 Preuner, G.; Eichinger, C.; Schrefl, M.: **Static-Dynamic Integration of External Services into Generic Business Processes**. In: [BGKS04], S. 263-277.

PrFa13 Provost, F.; Fawcett, T: **Data Science for Business**. Sebastopol (O'Reilly) 2013. URL: http://proquestcombo.safaribooksonline.com/9781449374273 (Abruf am 04.01.2017).

PrKK05 Priebe, T.; Kolter, J.; Kiss, C.: **Semiautomatische Annotation von Textdokumenten mit semantischen Metadaten**. In: [FSEI05], S. 1309-1328.

PüSi10 Pütz, C.; Sinz, E.J.: **Modellgetriebene Ableitung von BPMN-Workflowschemata aus SOM-Geschäftsprozessmodellen**. In: Engels, G.; Karagiannis, D.; Mayr, H.C. (Hrsg.): Proc. Modellierung 2010. 24.-26. März, Klagenfurt, Österreich. LNI 161, Bonn (Gesellschaft für Informatik) 2010, S. 253-268.

PWFS09 Pütz, C.; Wagner, D.; Ferstl, O.K.; Sinz, E.J.: **Geschäftsprozesse in Medizinischen Versorgungszentren und ihre Flexibilitätsanforderungen – ein fallstudienbasiertes Szenario**. Bericht 2009-001. Bayerischer Forschungsverbund forFLEX – Dienstorientierte IT-Systeme für hochflexible Geschäftsprozesse. Bamberg, Erlangen-Nürnberg, Regensburg 2009.

Pyle03 Pyle, D.: **Business Modeling and Data Mining**. San Francisco (Morgan Kaufmann) 2003.

Pyle04a Pyle, D.: **This Way Failure Lies. Nine simple rules you won't want to follow**. In: DB2 Magazine 9 (2004) 1. URL: http://www.db2mag.com/story/showArticle.jhtml?articleID=17602328 (Abruf am 28.04.2004).

Pyle99 Pyle, D.: **Data Preparation for Data Mining**. San Francisco (Morgan Kaufman) 1999.

Pyth17 o.V.: **Building a Mature Analytics Workflow: The dbt Viewpoint**. Dokumentation. Python Software Foundation, o.O., o.J. URL: https://dbt.readthedocs.io/ en/master/about/viewpoint/ (Abruf am 08.04.2017).

Quin91 Quinlan, J.R.: **Foreword**. In: Piatetsky-Shapiro, G.; Frawley, W.J. (Hrsg.): Knowledge Discovery in Databases. Menlo Park (AAAI Press) 1991, S. ix-xii.

RaME95 Raufer, H.; Morschheuser, S.; Enders, W.: **Ein Werkzeug zur Analyse und Modellierung von Geschäftsprozessen als Voraussetzung für effizientes Workflow-Management**. In: Wirtschaftsinformatik 37 (1995) 5, S. 467-479.

Ran03 Ran, S.: **A Model for Web Services Discovery with QoS**. In: ACM SIGecom Exchanges 4 (2003) 1, S. 1-10.

Rapi10 Rapid-I GmbH (Hrsg.): **RapidMiner 5.0 Benutzerhandbuch**. Dortmund (Rapid-I GmbH) 2010. URL: http://tenet.dl.sourceforge.net/project/rapidminer/1.%20RapidMiner/5.0/rapidminer-5.0-manual-german_v1.1.pdf (Abruf am 24.01.2012).

RaSc99 Rantzau, R.; Schwarz, H.: **A Multi-Tier Architecture for High-Performance Data Mining**. In: Buchmann, A. (Hrsg.): Datenbanksysteme in Büro, Technik und Wissenschaft, GI-Fachtagung BTW 99, Freiburg, März 1999. URL: http://elib.uni-stuttgart.de/opus/volltexte/1999/ 524/ (Abruf am 27.09.2005).

RaSi01 de Raedt, L.; Siebes, A. (Hrsg.): **Proc. 5th European Conference on Principles of Data Mining and Knowledge Discovery** (September 03-05, 2001, Freiburg). LNCS, Band 2168. London (Springer) 2001.

ReBD00 Reichert, M.; Bauer, T.; Dadam, P.: **ADEPT – Realisierung flexibler und zuverlässiger unternehmensweiter Workflow-Anwendungen**. In: Proc. KnowTech 2000, Leipzig, September 2000. URL: http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/RBD00.pdf (Abruf am 28.01.2006).

ReDa98 Reichert, M.; Dadam, P.: **ADEPT$_{flex}$ – Supporting Dynamic Changes of Workflows Without Losing Control**. In: Journal of Intelligent Information Systems 10 (March 1998) 2, S. 93-129.

Reic00 Reichert, M.: **Dynamische Ablaufänderungen in Workflow-Management-Systemen**. Diss., Fakultät für Informatik, Univ. Ulm 2000.

Reic01 Reichmann, T.: **Controlling mit Kennzahlen und Managementberichten**. 6. Aufl., München (Vahlen) 2001.

Reif03 Reif, M.: **Erweitertes Workflow-Management. Ein Ansatz zur Unterstützung des Prozessmanagements in Workflow-Anwendungen**. Berlin (Logos) 2003. Zugl.: Dissertation, Univ. Bamberg, 2003.

Reim91 Reimer, U.: **Einführung in die Wissensrepräsentation. Netzartige und schema-basierte Repräsentationsformate**. Stuttgart (Teubner) 1991.

ReSt04 Reichert, M.; Stoll, D.: **Komposition, Choreographie und Orchestrierung von Web Services: Ein Überblick**. In: EMISA FORUM, Mitteilungen der GI-FG Entwicklungsmethoden für Informationssysteme und deren Anwendung, 24 (2004) 2, S. 21-32. URL: http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/ReSt04.pdf (Abruf am 28.01.2006).

Reza95 Rezagholi, M.: **Management der Wiederverwendung in der Softwareentwicklung**. In: Wirtschaftsinformatik 37 (1995) 2, S. 221-230.

RiDa03 Rinderle, S.; Dadam, P: **Schemaevolution in Workflow-Management-Systemen.** Aktuelles Schlagwort. In: Informatik-Spektrum 26 (Februar 2003) 1, S. 17-19.

RiGo02 Richters, M.; Gogolla, M.: **OCL: Syntax, Semantics, and Tools**. In: Clark, T.; Warmer, J. (Hrsg.): Object Modeling with the OCL. The Rationale behind the Object Constraint Language. LNCS 2263, Berlin (Springer) 2002, S. 42-68.

RiGr89 Ritter, J.; Gründer, K.: **Historisches Wörterbuch der Philosophie**, Basel (Schwabe & Co) 1989.

RiHS05 Richter, J.-P.; Haller, H.; Schrey, P.: **Serviceorientierte Architektur**. In: Informatik Spektrum 28 (Oktober 2005) 5, S. 413-416.

RiRD04 Rinderle, S.; Reichert, M.; Dadam, P.: **Correctness criteria for dynamic changes in workflow systems – a survey**. In: Data & Knowledge Engineering 50 (2004), S. 9-34. URL: http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/RRD04a.pdf (Abruf am 28.01. 2006).

Rögl09 Röglinger, M.: **Verifikation von Webservicekompositionen. Eine Konkretisierung des Korrektheitsbegriffs und ein Anforderungsframework für serviceorientierte Modellierungsansätze**. In: Wirtschaftsinformatik 51 (2009) 6, S. 496-505.

Rohm98 Rohm, C.: **Prozeßmanagement als Fokus im Unternehmenswandel: Ein ganzheitlicher Ansatz zur strategieorientierten Identifikation, Analyse und Gestaltung von Unternehmensprozessen**. Schriftenreihe des Instituts für Unternehmensplanung, Band 24. Gießen (Ferber'sche Universitäts-Buchhandlung) 1998.

Rose96 Rosemann, M.: **Komplexitätsmanagement in Prozessmodellen. Methodenspezifische Gestaltungsempfehlungen für die Informationsmodellierung**, Wiesbaden (Gabler) 1996. Zugl.: Münster, Univ., Diss., 1995.

RuCz11 Rupp, C.; Cziharz, T.: **Mit Regeln zu einer besseren Spezifikation**. In: Informatik-Spektrum 34 (2011) 3, S. 255-264.

Rühl92 Rühli, E.: **Koordination**. In: [Fres92], S. 1165.

Runk00 Runkler, T. A.: **Information Mining. Methoden, Algorithmen und Anwendungen intelligenter Datenanalyse**. Braunschweig (Vieweg) 2000.

RuNo03 Russel, S.J.; Norvig, P.: **Artificial Intelligence. A Modern Approach**. Second Edition. Upper Saddle River (Pearson Education/Prentice Hall) 2003.

RuPR99 Rupprecht, C.; Peter, G.; Rose, T.: **Ein modellgestützter Ansatz zur kontextspezifischen Individualisierung von Prozessmodellen**. In: Wirtschaftsinformatik 41 (1999) 3, S. 226-237.

RWRW05 Rinderle, S.; Weber, B.; Reichert, M.; Wild, W: **Integrating Process Learning and Process Evolution: A Semantics Based Approach**. In: Proc. Int'l Conf. on Business Process Management, BPM 2005, Nancy, Frankreich, September 2005, LNCS 3649, S. 252-267. URL: http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/RWRW05.pdf (Abruf am 28.01.2006).

SaBS00 Saitta, L.; Beccari, G.; Serra, A.: **Informed Parameter Setting**. Mining Mart Project (IST-1999-11993), Deliverable No. D 4.1. o.O., December 22, 2000. URL: http://www-ai.cs.uni-dortmund.de/forschung/projekte/miningmart/deliverables/d4/ del-4.1.pdf (Abruf am 16.01.2002).

Säub00 Säuberlich, F.: **KDD und Data Mining als Hilfsmittel der Entscheidungsunterstützung**. Frankfurt am Main (Lang) 2000.

SBBF11 Sinz, E.J.; Bartmann, D.; Bodendorf, F.; Ferstl, O.K. (Hrsg.): **Dienstorientierte IT-Systeme für hochflexible Geschäftsprozesse**. Schriften aus der Fakultät Wirtschaftsinformatik und Angewandte Informatik der Otto-Friedrich-Universität Bamberg, Band 9. Bamberg (University of Bamberg Press) 2011.

Sche04 Schennach, R.G.: **Konzeption und Anwendung eines Vorgehensmodells zur Nutzung von Knowledge Discovery in Databases am Beispiel eines Automobilkonzerns**. Diplomarbeit, Lehrstuhl für Wirtschaftsinformatik, insbes. Systementwicklung und Datenbankanwendung, Otto-Friedrich-Universität, Bamberg 2004.

Sche96 Schelle, H.: **Projekte zum Erfolg führen**. München (Beck) 1996.

Sche98 Scheer, A.-W.: **ARIS: Vom Geschäftsprozeß zum Anwendungssystem**. 3. Aufl., Berlin (Springer) 1998.

Schi01 Schimm, G.: **Process Mining elektronischer Geschäftsprozesse**. In: Proc. Elektronische Geschäftsprozesse, 2001.

Schi04 Schimm, G.: **Mining Exact Models of Concurrent Workflows**. In: Computers in Industry 53 (2004) 3, S. 265-281.

Schm97 Schmidt, G.: **Prozeßmanagement. Modelle und Methoden**. Berlin (Springer) 1997.

Schn00 Schneeberger, J.: **Planen**. In: [GöRS00], S. 491-515.

Schn91 Schneeweiß, C.: **Planung. Band 1: Systemanalytische und entscheidungstheoretische Grundlagen**. Berlin (Springer) 1991.

Schn92 Schneeweiß, C.: **Planung. Band 2: Konzepte der Prozeß- und Modellgestaltung**. Berlin (Springer) 1992.

Schw06 Schwalbe, K.: **Information Technology Project Management**. Boston (Thomson) 2006.

ScNZ95 Scheer, A.-W.; Nüttgens, M.; Zimmermann, V.: **Rahmenkonzept für ein integriertes Geschäftsprozeßmanagement**. In: Wirtschaftsinformatik 37 (1995) 5, S. 426-434.

ScSe04 Schmelzer, H.J.; Sesselmann, W.: **Geschäftsprozessmanagement in der Praxis. Produktivität steigern, Wert erhöhen, Kunden zufrieden stellen**. 4. Aufl., München (Hanser) 2004.

ScVr94a Scholz, R.; Vrohlings, A.: **Realisierung von Prozeßmanagement**. In: [GSVR94], S. 21-36.

Sedl96 Sedlmeier, P.: **Jenseits des Signifikanztest-Rituals: Ergänzungen und Alternativen**. In: Methods of Psychological Research Online 1 (1996) 4, S. 41-63.

SePr10 Seidel, T.; Prenzel, M.: **Beobachtungsverfahren: Vom Datenmaterial zur Datenanalyse**. In: [HoSc10], S. 139-152.

Shan48 Shannon, C.E.: **Mathematical Theory of Communication**. In: Bell Systems Technical Journal 27 (1948), 379-423.

ShDT14 Sharda, R.; Delen, D.; Turban, E.: **Business Intelligence and Analytics. Systems for Decision Support**. 14. Aufl., Boston (Pearson) 2014.

Shei13 Sheik, N.: **Implementing Analytics. A Blueprint for Design, Development, and Adoption**. Waltham (Morgan Kaufmann) 2013.

ShWe49 Shannon, C.E.; Weaver, W.: **The Mathematical Theory of Communication**. Urbana (University of Illinois Press) 1949.

ShYZ02 Shen, Y.-D.; Yang, Q.; Zhang, Z.: **Objective-Oriented Utility-Based Association Mining**. In: Proc. 2002 IEEE International Conference on Data Mining (ICDM), Maebashi City, Japan, S. 426-433.

SiHF96 Simoudis, E.; Han, J.; Fayyad, U.: **Proc. The Second International Conference on Knowledge Discovery & Data Mining (KDD-96)**. August 2-4, 1996, Portland, Oregon, USA. Menlo Park (AAAI Press) 1996.

SiLK94 Simoudis, E.; Livezey, B.; Kerber, R.: **Integrating Inductive and Deductive Reasoning for Database Mining**. In: [FaUt94], S. 37-48.

Sinz10 Sinz, E.J.: **Konstruktionsforschung in der Wirtschaftsinformatik: Was sind die Erkenntnisziele gestaltungsorientierter Wirtschaftsinformatik-Forschung?** In: [ÖsWB10], S.27-33.

Sinz94 Sinz, E.J.: **Geschäftsprozeßmodellierung als Grundlage für den Einsatz von Workflow-Management-Systemen**. In: Hasenkamp, U. (Hrsg.): Einführung von CSCW-Systemen in Organisationen. Tagungsband der D-CSCW ´94. Braunschweig (Vieweg) 1994, S. 219-224.

Sinz95 Sinz, E.J.: **Ansätze zur fachlichen Modellierung betrieblicher Informationssysteme. Entwicklung, aktueller Stand und Trends**. Bamberger Beiträge zur Wirtschaftsinformatik Nr. 34, Bamberg 1995. Erschienen in: Heilmann H., Heinrich L.J., Roithmayr F. (Hrsg.): Information Engineering. München (Oldenbourg) 1996, S. 123-143.

Sinz97 Sinz, E.J.: **Architektur betrieblicher Informationssysteme**. Bamberger Beiträge zur Wirtschaftsinformatik Nr. 40, Bamberg 1997.

SiTu95 Silberschatz, A.; Tuzhilin, A.: **On Subjective Measures of Interestingness in Knowledge Discovery**. In: [FaUt95], S. 275-281.

SLGR10 Spiel, C.; Lüftenegger, M.; Gradinger, P.; Reimann, R.: **Zielexplikation und Standards in der Evaluationsforschung**. In: [HoSc10], S. 252-260.

SLL+06 Shen, H.T.; Li, J.; Li, M.; Ni, J.; Wang, W. (Hrsg.): **Advanced Web and Network Technologies, and Applications. Proc. APWeb 2006 International Workshops: XRA, IWSN, MEGA, and ICSE. Harbin, China, January 16-18, 2006.** LNCS 3842, Berlin (Springer) 2006.

Soel10 Soellner, R.: **Modelle der Evaluation**. In: [HoSc10], S. 233-243.

Somm01 Sommerville, I.: **Software Engineering**, 6. Aufl., München (Pearson) 2001.

SpGL10 Spiel, C.; Gradinger, P.; Lüftenegger, M.: **Grundlagen der Evaluationsforschung**. In: [HoSc10], S. 223-232.

SpNa09 Spott, M.; Nauck, D.: **Automatic Intelligent Data Analysis**. In: Wang, H.-F. (Hrsg.): Intelligent Data Analysis. Developing New Methodologies Through Pattern Discovery and Recovery. Hershey (Information Science Reference) 2009, S. 1-17.

StBF98 Studer, R.; Benjamin, V.R.; Fensel, D.: **Knowledge Engineering: Principles and Methods**. In: Data & Knowledge Engineering 25 (1998), S. 161-197.

StGR98 Stickel, E.; Groffmann, H.-D.; Rau K.-H. (Hrsg.): **Gabler Wirtschaftsinformatik-Lexikon**, Wiesbaden (Gabler) 1998.

StSh07 Stufflebeam, D.L.; Shinkfield, A.J.: **Evaluation Theory, Models, and Applications**. San Francisco (Jossey-Bass) 2007.

StSt14 Stockinger, K.; Stadelmann, T.: **Data Science für Lehre, Forschung und Praxis**. In: HMD 51 (2014), S. 469-479.

StWW98 Stumme, G.; Wille, R.; Wille, U.: **Conceptual Knowledge Discovery in Databases Using Formal Concept Analysis Methods**. In: [ZyQu98], S. 450-458.

SuYa98 Suyama, A.; Yamaguchi, T.: **Specifying and learning inductive learning systems using ontologies**. In: Working Notes from the 1998 AAAI Workshop on the Methodology of Applying Machine Learning: Problem Definition, Task Decomposition and Technique Selection, 1998.

TaBa98 Tayi, G.K.; Ballou, D.P.: **Examining Data Quality**. In: Communications of the ACM 41 (1998) 2, S. 54-57.

TGRS04 Tian, M.; Gramm, A.; Ritter, H.; Schiller, J.: **Efficient Selection and Monitoring of QoS-Aware Web Services with the WS-QoS Framework**. In: Proc. ACM/IEEE/WIC International Conference on Web Intelligence (Peking, China, 2004), S. 152-158.

Thei99 Theis, H.-J.: **Handels-Marketing. Analyse- und Planungskonzepte für den Einzelhandel**. Frankfurt/Main (Deutscher Fachverlag) 1999.

ThFe06 Thomas, O.; Fellmann, M.: **Semantische Ereignisgesteuerte Prozessketten**. In: Schelp, J.; Winter, R.; Frank, U.; Rieger, B.; Turowski, K. (Hrsg.): Integration, Informationslogistik und Architektur. Proc. DW2006, 21.-22. September 2006, Friedrichshafen. GI-Edition Lecture Notes in Informatics, Vol. P-90. Bonn (Gesellschaft für Informatik) 2006.

ThFe09 Thomas, O.; Fellmann, M.: **Semantische Prozessmodellierung: Konzeption und informationstechnische Unterstützung einer ontologiebasierten Repräsentation von Geschäftsprozessen**. In: Wirtschaftsinformatik 51 (2009) 6, S. 506-518.

ThLi98 Theusinger, C.; Lindner, G.: **Benutzerunterstützung eines KDD-Prozesses anhand von Datencharakteristiken**. In: Wysotzki, F.; Geibel, P.; Schädler, K. (Hrsg.): Beiträge zum Treffen der GI-Fachgruppe 1.1.3 Maschinelles Lernen (FGML-98), Technical Report Band 98/11, TU Berlin, Berlin 1998. URL: http://citeseer.ist.psu.edu/theusinger98benutzeruntersttzung.html (Abruf am 23.02.2007).

Thon05 Thonemann, U.: **Operations Management. Konzepte, Methoden und Anwendungen**. München (Pearson Studium) 2005.

TsHe04 Tschenlin, D.K.; Helmig, B.: **Datenanalyse**. In: Bruhn, M.; Homburg, C. (Hrsg.): Gabler Lexikon Marketing, 2. Aufl., Wiesbaden (Gabler) 2004, S. 148.

Tuft74 Tufte, E.R.: **Data Analysis for Politics and Policy**. Englewood Cliffs (Prentice-Hall) 1974.

Tuke62 Tukey, J.W.: **The Future of Data Analysis**. In: The Annals of Mathematical Statistics 33 (1962), S. 1-67.

Tuke77 Tukey, J. W.: **Exploratory Data Analysis**. Reading (Addison-Wesley) 1977.

Uthu96 Uthurusamy, R.: **From Data Mining to Knowledge Discovery: Current Challenges and Future Directions**. In: [FPSU96], S. 561-569.

VaBl09 Vanschoren, J.; Blockeel, H.: **Stand on the Shoulders of Giants: Towards a Portal for Collaborative Experimentation in Data Mining**. In: [PLKB09], S. 88-99.

Vard09 Varde, A.S.: **Challenging Research Issues in Data Mining, Databases and Information Retrieval**. In: SIGKDD Explorations 11 (2009) 1, S. 49-52. URL: http://www.kdd.org/exploration_files/s7V11n1.pdf (Abruf am 04.11.2016).

Vell97 Velleman, P.F.: **The Philosophical Past and the Digital Future of Data Analysis: 375 Years of Philosophical Guidance for Software Design on the Occasion of John W. Tukey's 80th Birthday**. In: [BrFM97], S. 317-337.

VGBS04 Vilalta, R.; Giraud-Carrier, C.; Brazdil, P.; Soares, C.: **Using Meta-Learning to support Data Mining**. In: International Journal of Computer Science and Applications 1 (2004) 1, S. 31-45.

Voge91 Vogel, F.: **Beschreibende und schließende Statistik. Formeln, Definitionen, Erläuterungen, Stichwörter und Tabellen**. 6. Aufl., München (Oldenbourg) 1991.

Voit10 Voit, T.: **Entwicklung und Überprüfung von Kausalhypothesen. Gestaltungsoptionen für einen Analyseprozess zur Fundierung betrieblicher Ziel- und Kennzahlensysteme durch Kausalhypothesen am Beispiel des Performance-Managements.** Diss., Univ. Bamberg. Bamberg (University of Bamberg Press) 2010.

VRBT13 Vanschoren, J.; van Rijn, J.N.; Bischl, B.; Torgo, L.: **OpenML: Networked Science in Machine Learning**. In: SIGKDD Explorations 15 (2013) 2, S. 49-60. URL: http://www.kdd.org/exploration_files/15-2-2013-12.pdf (Abruf am 04.11.2016).

W3C04b W3C Web Services Architecture Working Group: **Web Services Architecture**. W3C Working Group Note 11 February 2004, o.O. (W3C) 2004. URL: http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/ (Abruf am 05.06.2004).

W3C04c W3C RDF Working Group: **Resource Description Framework (RDF): Concepts and Abstract Syntax**. W3C Recommendation 10 February 2004, o.O. (W3C) 2004. URL: http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/ (Abruf am 20.01.2012).

W3C09 W3C OWL Working Group: **OWL 2 Web Ontology Language Document Overview**. W3C Recommendation 27 October 2009, o.O. (W3C) 2009. URL: http://www.w3.org/TR/2009/REC-owl2-overview-20091027/ (Abruf am 20.01.2012).

WaFe10 Wagner, D.; Ferstl, O.K.: **Erhöhte Abbildungstreue von Geschäftsprozessmodellen durch Kontextsensitivität**. In: Engels, G.; Karagiannis, D.; Mayr, H.C. (Hrsg.): Proc. Modellierung 2010, 24.-26. März, Klagenfurt, Österreich. LNI P-161 GI 2010, Bonn (Köllen) 2010, S. 117-132.

Wagn+11 Wagner, D.; Leunig, B.; Suchan, C.; Frank, J.; Ferstl, O.K.: **Klassifikation von Geschäftsprozessen anhand ihres Flexibilitätsbedarfs**. In: [SBBF11], S. 53-78.

WeAR11 Wegener, D.; Anguita, A.; Rüping, S.: **Enabling the reuse of data mining processes in healthcare by integrating data semantics**. In: Proc. 3rd International Workshop on Knowledge Representation for Health Care, KR4HC '11. Bled, Slovenia, 2011, S. 222-235. URL: http://publica.fraunhofer.de/eprints/urn:nbn:de:0011-n-1780925.pdf (Abruf am 12.03.2012).

WeKr10 Westermann, R.; Krohn, J.: **Gütekriterien**. In: [HoSc10], S. 71-86.

Wers96 Wersig, G.: **Die Komplexität der Informationsgesellschaft**. Schriften zur Informationswissenschaft, Band 26. Konstanz (Universitätsverlag Konstanz) 1996.

WeRü11 Wegener, D.; Rüping, S.: **On Reusing Data Mining in Business Processes: A Pattern-Based Approach**. In: zur Muehlen, M.; Su, J. (Hrsg..): BPM 2010 International Workshops and Education Track. Revised Selected Papers, Hoboken, NJ, USA, September 13-15, 2010. LNBIP 66, Berlin (Springer) 2011, S. 264-276

WeWB04 Weber, B.; Wild, W.; Breu, R.: **CBRFlow: Enabling Adaptive Workflow Management through Conversational Case-based Reasoning**. In: Proc. European Conference on Case-based Reasoning (ECCBR '04), Madrid 2004, S. 434-448.

WeZS08 Weiss, G.M.; Zadrozny, B.; Saar-Tsechansky, M.: **Guest Editorial: Special issue on utility-based data mining**. In: Data Mining and Knowledge Discovery 17 (2008), S. 129-135.

WfMC99 Workflow Management Coalition (WfMC) (Hrsg.): **Terminology & Glossary**. Document Number WFMC-TC-1011, Issue 3.0, 1999.

WHKR00 Wittmann, T.; Hunscher, M.; Kischka, P.; Ruhland, J.: **Data Mining. Entwicklung und Einsatz robuster Verfahren für betriebswirtschaftliche Anwendungen**. Frankfurt/M. (Lang) 2000.

Wiem73 Wiemann, H.G.: **Untersuchungen zur Frage der optimalen Informationsbeschaffung. Eine literaturkritische Analyse zur Problematik der betriebswirtschaftlichen Informationstheorie**. Frankfurt/M. (Harri Deutsch) 1973.

WiFr00 Witten, I.H.; Frank, E.: **Data Mining. Practical Machine Learning Tools and Techniques with Java Implementations**. San Francisco (Morgan Kaufman) 2000.

WiHu96 Williams, G.J.; Huang, Z: **Modelling the KDD Process. A Four Stage Process and Four Element Model**. URL: http://citeseer.ist.psu.edu/cache/papers/cs/13922/modelling-the-kdd-process.pdf, 1996 (Abruf am 09.02.2006).

Wild01 Wilde, K.D.: **Data Warehouse, OLAP und Data Mining im Marketing: Moderne Informationstechnologien im Zusammenspiel**. In: [HKMW01], S. 1-19.

Wild74 Wild, J.: **Grundlagen der Unternehmungsplanung**. Reinbek bei Hamburg (Rowohlt) 1974.

Wilm12 Wilms, J.: **Auf Anforderungen flexibel reagieren und Rollen festlegen. Effektivität in BI-Projekten**. In: BI-Spektrum (2012) 05, S. 16-19.

Wint16 Winter, R.: **Analytische Informationssysteme aus Managementsicht: lokale Entscheidungsunterstützung vs. unternehmensweite Informations-Infrastruktur**. In: [GlCh16], S. 67-96.

WiRe96 Wirth, R.; Reinartz, T.P.: **Detecting Early Indicator Cars in an Automotive Database: A Multi-Strategy Approach**. In: [SiHF96], S. 76-80.

Witt59 Wittmann, W.: **Unternehmung und unvollkommene Information. Unternehmerische Voraussicht – Ungewissheit und Planung**. Köln, Opladen (Westdeutscher Verlag) 1959.

Witt98 Wittenberg, R.: **Grundlagen computerunterstützter Datenanalyse**. 2. Aufl., Stuttgart (Lucius & Lucius) 1998.

Wrob+15 Wrobel, S.; Voss, H.; Köhler, J.; Beyer, U.; Auer, S.: **Big Data, Big Opportunities. Anwendungssituation und Forschungsbedarf des Themas Big Data in Deutschland**. In: Informatik-Spektrum 38 (2015) 5, S. 370-378.

WRRW05 Weber, B.; Reichert, M.; Rinderle, S.; Wild, W: **Towards a Framework for the Agile Mining of Business Processes**. In: Bussler, C.J.; Haller, A. (Hrsg.): Business Process Management Workshops. BPM 2005 International Workshops, BPI, BPD, ENEI, BPRM, WSCOBPM, BPS, Nancy, Frankreich, September 5, 2005. Revised Selected Papers, S. 191-202. URL: http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/WRRW05.pdf (Abruf am 28.01.2006).

WRWR05b Weber, B.; Reichert, M.; Wild, W.; Rinderle, S.: **Balancing Flexibility and Security in Adaptive Process Management Systems**. In: Proc. 13th Int'l Conf. on Cooperative Information Systems (CoopIS'05), Agia Napa, Zypern, November 2005. URL: http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/WRWR05a.pdf (Abruf am 28.01.2006).

WSG+97 Wirth, R.; Shearer, C.; Grimmer, U.; Reinartz, T.P.; Schlösser, J.; Breitner, C.; Engels, R.; Lindner, G.: **Towards Process-Oriented Tool Support for Knowledge Discovery in Databases**. In: [KoZy97], S. 243-253.

WSLF11 Wagner, D.; Suchan, C.; Leunig, B.; Frank, J.: **Towards the Analysis of Information Systems Flexibility: Proposition of a Method**. In: [BeSc11], S. 808-817.

WWSE96 Wrobel, S.; Wettschereck, D.; Sommer, E.; Emde, W.: **Extensibility in Data Mining Systems**. In: [SiHF96], S. 214-219.

YaHB04 Yao, H.; Hamilton, H.J.; Butz, C.J.: **A Foundational Approach to Mining Itemset Utilities from Databases**. In: Berry, M.W.; Dayal, U.; Kamath, C.; Skillicorn, D. (Hrsg.): Proc. Fourth SIAM International Conference on Data Mining, Lake Buena Vista, Florida (USA) 2004, S. 482-486.

YaWu06 Yang, Q.; Wu, X.: **10 Challenging Problems in Data Mining Research**. In: International Journal of Information Technology & Decision Making 5 (2006) 4, S. 597-604.

Zhug03 Zhuge, H.: **Component-based workflow systems development**. In: Decision Support Systems 35 (2003), S. 517-536.

ZiKZ00 Zighed, D.A.; Komorowski, J.; Zytkow, J. (Hrsg.): **Principles of Data Mining and Knowledge Discovery**. Proc. 4th European Conference, PKDD 2000, Lyon (France), September 2000. Berlin (Springer) 2000.

Zimm14 Zimmermann, A.: **The Data Problem in Data Mining**. In: SIGKDD Explorations 16 (2014) 2, S. 38-45. URL: http://www.kdd.org/exploration_files/Volume16-Issue2.pdf (Abruf am 04.11.2016).

Zimm95 Zimmermann, H.-J. (Hrsg.): **Datenanalyse. Anwendung von Data-Engine mit Fuzzy Technologien und Neuronalen Netzen**. Düsseldorf (VDI-Verlag) 1995.

ZKTG12 Zimmer, M.; Krawatzeck, R.; Trahasch, S.; Gansor, T.: **Standards für Agile BI in der BI-Community durch den TDWI. Definition und Herausforderung**. In: BI-Spektrum (2012) 05, S. 12-15.

ZKZL10 Žáková, M.; Kremen, P.; Železný, F.; Lavrač, N.: **Automating Knowledge Discovery Workflow Composition through Ontology-based Planning**. In: IEEE Transactions on Automation Science and Engineering, 2010.

ZLKO97 Zhong, N.; Liu, C.; Kakemoto, Y.; Ohsuga, S.: **KDD Process Planning**. In: [HMPU97], S. 291-294.

Zöfe03 Zöfel, P.: **Statistik für Wirtschaftswissenschaftler** im Klartext. München (Pearson Studium) 2003.

ZPZL09 Žáková, M.; Podpečan, V.; Železný, F.; Lavrač, N.: **Advancing Data Mining Workflow Construction: A Framework and Case using the Orange Toolkit**. In: [PLKB09], S. 39-51.

ZyQu98 Žytkow, J.M.; Quafafou, M. (Hrsg.): **Proc. Second European Symposium on Principles of Data Mining and Knowledge Discovery** (September 23-26, 1998), Nantes. LNCS, Band 1510. Berlin (Springer) 1998.

ZyRa99 Žytkow, J.M.; Rauch, J. (Hrsg.): **Proc. Third European Conference on Principles of Data Mining and Knowledge Discovery** (September 15-18, 1999), Prag. LNCS, Band 1704. Berlin (Springer) 1999.

University
of Bamberg
Press

Die fortschreitende Digitalisierung, die ubiquitäre Verfügbarkeit computergesteuerter Systeme sowie die Tendenz zu nutzererzeugten Inhalten führen zu einem stetigen Anwachsen betrieblicher Datenbestände, deren Auswertung große Potenziale birgt. Zur Durchführung erfolgreicher Datenanalysen in der Praxis sind zahlreiche Aspekte zu berücksichtigen, die unter anderem folgende Fragestellungen umfassen:

- Wie kann die methodische Ableitung einer analytischen Fragestellung aus dem Anwendungsproblem gelingen?
- Wie kann die Zielorientierung komplexer Analysevorhaben sichergestellt werden?
- Welche Ansätze eignen sich zur Planung von Datenanalyseprozessen?
- Nach welchen Kriterien kann eine Beurteilung durchgeführter Datenanalysen erfolgen?

Die vorliegende Arbeit präsentiert eine interdisziplinäre Methodik zum Management von Datenanalyseprozessen zur Unterstützung von Business Analytics. Die Methodik umfasst einen Modellierungsansatz, ein Architekturmodell sowie ein Vorgehensmodell.

Die wesentlichen Beiträge der Methodik liegen in ihrer Breite, Höhe und Tiefe: Zum Ersten ist sie in ihrer Breite für verschiedene datenanalytische Disziplinen (z.B. Statistik, Reporting, Data Science) geeignet, wie sie in der Praxis zur umfassenden empirischen Untersuchung eines Sachverhalts – häufig kombiniert – zum Einsatz gelangen. Zum Zweiten begleitet sie Datenanalysen über die Höhe von vier Architekturebenen, welche die Strukturierung des Sachproblems, die Formulierung empirischer Fragestellungen, die Konzipierung zugehöriger Handlungspläne sowie die Beschreibung verfügbarer Ressourcen behandeln. Zum Dritten geht sie in ihrer Tiefe detailliert auf einzelne Aspekte der Planung, Steuerung und Revision von Datenanalyseprozessen ein und gibt dem Analytiker konkrete Empfehlungen an die Hand. Sie gestattet damit eine vollständige Repräsentation von Datenanalysevorhaben. Zugleich liefert sie einen Bezugsrahmen zur theoretischen Untersuchung der betrieblichen Datenanalyse.

ISBN 978-3-86309-566-6
9 783863 095666
www.uni-bamberg.de/ubp