



# RFID, RPA & IPS im produktionsnahen ERP-basierten MTO-Auftragsabwicklungsprozess

Thomas Zwiefel

Hochschule für angewandte Wissenschaften Würzburg-Schweinfurt, Konrad-Zuse-Straße 2, 97421 Schweinfurt, thomas.zwiefel@student.fhws.de

1	Einleitung.....	186
2	Geschäftsprozesse.....	187
3	Technologien .....	189
4	Integration der Technologien.....	194
5	Laborbeispiel .....	198
6	Fazit .....	205
7	Literaturverzeichnis .....	205

## **Abstract:**

*Industrie-4.0-Konzepte sind inzwischen in zahlreichen Unternehmen angekommen. Neben den vorrangigen Themen wie Digitalisierung und vernetzten Systemen kommt der Automatisierung nach wie vor eine zentrale Bedeutung zu. Die Einbindung von Softwarerobotern sowie Indoor-Lokalisierungssystemen kann dabei die Effizienz von Geschäftsprozessen massiv beeinflussen. Ziel dieses Beitrags ist die Entwicklung und Umsetzung eines Laborbeispiels, das die weitgehende Automatisierung eines produktionsnahen Make-to-Order Auftragsabwicklungsprozesses auf Basis eines Enterprise-Resource-Planning-Systems durch ausgewählte Industrie 4.0-Komponenten innerhalb einer Laborumgebung beinhaltet.*

*Dazu werden zunächst die theoretischen Grundlagen von Geschäftsprozessen im Allgemeinen sowie des Auftragsabwicklungsprozesses im Besonderen dargestellt. Im nächsten Schritt werden die eingesetzten Technologien separat diskutiert, bevor ein Integrationsansatz aufgezeigt wird. In diesem Zusammenhang werden Radiofrequenz-Identifikation, ein Indoor-Positionierungssystem mittels Ultrabreitband sowie die robotergestützte Prozessautomatisierung evaluiert. Schließlich folgt die Entwicklung eines Laborbeispiels zur Integration der genannten Technologien. Neben dem Verfahren wird auch die konkrete Umsetzung der Technologien durch geeignete Hard- und Software erläutert.*

*JEL Classification: O33, M11, M15*

**Keywords:** Digitalisierung, Prozessautomatisierung, Softwareroboter, Indoor-Lokalisierung, IoT.

## 1 Einleitung

Die vierte industrielle Revolution („Industrie 4.0“) konzentriert sich auf Fokusthemen wie die Digitalisierung oder vernetzte Systeme. Doch auch die Automatisierung ist weiterhin von großer Bedeutung. Im Gegensatz zur dritten industriellen Revolution steht die klassische Automatisierung jedoch zunehmend unter Druck. Während Unternehmen früher vor allem über große Losmengen die Kosten zu reduzieren versuchten, ist die Variantenvielfalt sowie der Wunsch nach mehr Individualisierung und Flexibilität ausgeprägter denn je (Jeschke et al., 2017, S. 8). In Kombination mit einem globalen Wettbewerb und einem damit einhergehenden rauen Preiskampf entsteht dadurch der Drang zu Innovationen. Die Geschäftsprozesse eines Unternehmens bleiben dabei nicht unberührt.

Im Fokus befindet sich nicht zuletzt der Auftragsabwicklungsprozess. Denn schon in den letzten vier Jahrzehnten ist ein Trend zur zunehmenden Dezentralisierung der Fabrikplanung bis hin zur Selbststeuerung und -optimierung der Fabrik erkennbar (Scheer, 2020, S. 37). Hierfür ist neben Know-how auch Hardware für neue Lösungen im Bereich der Supply Chain notwendig. Durch Ansätze wie Radio-Frequency Identification (RFID) oder ein Indoor Positioning System (IPS) können beispielsweise Lager- und Bestandsbuchungen gänzlich ohne menschliche Beteiligung erfasst werden.

Dennoch implizieren neue Systeme sowie eine Dezentralisierung meist einen Mehraufwand für die Unternehmen. Um diesen erhöhten Aufwand vor dem Hintergrund eines steigenden Kostendrucks stemmen zu können, sind Effizienzsteigerungen unumgänglich. Eine steigende Anzahl an Unternehmen verfolgt den Ansatz der Automatisierung durch Roboter. Nachdem Produktionsprozesse bereits vor vielen Jahrzehnten von der „Robotisierung“ betroffen waren, steht die Automatisierung von Geschäftsprozessen erst am Anfang (Siderska, 2020, S. 22). In den „Top Strategic Technology Trends for 2021“ der Beratungsfirma Gartner ist der Einsatz von Softwarerobotern (Robotic Process Automation, RPA) ein wichtiger Schlüssel hin zu mehr Erfolg (Gartner, 2020, S. 12). In Kombination mit Systemen des Enterprise-Resource-Planning (ERP) sind solche Verbundlösungen in der Lage, erhebliche Ressourceneinsparungen in Unternehmen zu erzielen, indem der Mensch schrittweise von zeitraubenden, nicht-wertschöpfenden Tätigkeiten befreit wird (Hierzer, 2020, S. 66–67).

Ziel dieses Beitrags ist die weitgehende Automatisierung eines produktionsnahen Auftragsabwicklungsprozesses auf Basis eines ERP-Systems unter Verwendung ausgewählter Industrie 4.0-Komponenten innerhalb einer Laborumgebung. Dazu werden zunächst die theoretischen Grundlagen von Geschäftsprozessen im Allgemeinen und des Auftragsabwicklungsprozesses im Speziellen erläutert. Im nächsten Schritt werden die eingesetzten Technologien isoliert voneinander betrachtet, bevor ein Integrationsansatz aufgezeigt wird. In diesem Zusammenhang werden Radio-frequency

Identification, ein Ultrabreitband-Indoor-Positionierungssystem sowie die Robotic Process Automation evaluiert. Schließlich folgt die Entwicklung eines Laborbeispiels zur Integration der genannten Technologien. Neben dem Verfahren wird auch die konkrete Umsetzung der Technologien erläutert.

## 2 Geschäftsprozesse

Unter Geschäftsprozessen versteht man eine strukturierte Abfolge von Aktivitäten zur Erreichung eines bestimmten Endzustandes (Kurbel, 2016, S. 11). Dabei hat jede Aktivität Inputs sowie Outputs, die zur Ausführung benötigt werden bzw. durch den Prozessschritt entstehen (Hammer & Champy, 1993, S. 35). Im Rahmen der Anwendung von Geschäftsprozessen kommt der Organisationsgestaltung eine tragende Rolle zu. Durch den Wandel von einer funktionsorientierten hin zu einer prozessorientierten Betriebsorganisation können Unternehmen durch Geschäftsprozesse erhebliche Produktivitätssteigerungen erzielen (vom Brocke & Rosemann, 2014, S. 7). Dies unterstreicht die Notwendigkeit einer systematischen Herangehensweise zur Implementierung und Optimierung von Geschäftsprozessen in Unternehmen. Durch ein „Business Process Management System“ (BPMS) wird dieser Notwendigkeit Rechnung getragen. Das BPMS umfasst dabei neben der Planung auch die Steuerung und Ausführung von Arbeitsabläufen (Gadatsch, 2015, S. 7). Die Beschreibung von Geschäftsprozessen erfolgt meist in der Sprache „Business Process Model and Notation“ (BPMN), da mittels einheitlichem Austauschformat „Extensible Markup Language“ (XML) eine große Kompatibilität gegeben ist (Barton et al., 2017, S. 2). Durch die Aneinanderreihung von Aufgaben, Verzweigungen sowie Zuständen können Geschäftsprozesse leicht verständlich dargestellt werden.

Die Auftragsabwicklung dient als zentraler Geschäftsprozess zur Abwicklung von Kundenaufträgen (Kurbel, 2016, S. 223). Dabei wird der Prozess je nach Modus zur Auftragserzeugung unterschiedlich initiiert. In Abbildung 1 ist ein vereinfachter Auftragsabwicklungsprozess eines Produktionsunternehmens in BPMN dargestellt. Der Auftrag wird dort durch eine Kundenbestellung beim Vertrieb initiiert. Erzeugen und Einplanen des Fertigungsauftrages durch die Produktionsplanung folgen parallel zur Auftragsbestätigung. Nach eingeplantem Fertigungsauftrag werden Bestellungen erzeugt, ausgelöst und schließlich der Produktion bereitgestellt. Die Produktion kann die Fertigung starten, sobald das Material für den jeweiligen Fertigungsauftrag ausgebucht wurde. Die durchgeführten Fertigungsschritte werden über die Auftragsrückmeldung erfasst. Dort wird der Auftrag durch Zeit- und Mengenangaben dokumentiert. Nach erfolgter Produktion kann der Lieferschein erzeugt werden, der typischerweise der Lieferung beigelegt wird. Ist die Ware durch das Materialmanagement verpackt, kann sie an den gewünschten Lieferort versendet werden. Im Nachgang wird durch den Vertrieb noch die Rechnung erstellt und der empfangenden Person zugestellt. Die Zahlungsüberwachung am Ende des Prozesses wird von der Buchhaltung

übernommen. Mit der Buchung des Zahlungseingangs ist der Kundenauftrag abgeschlossen.

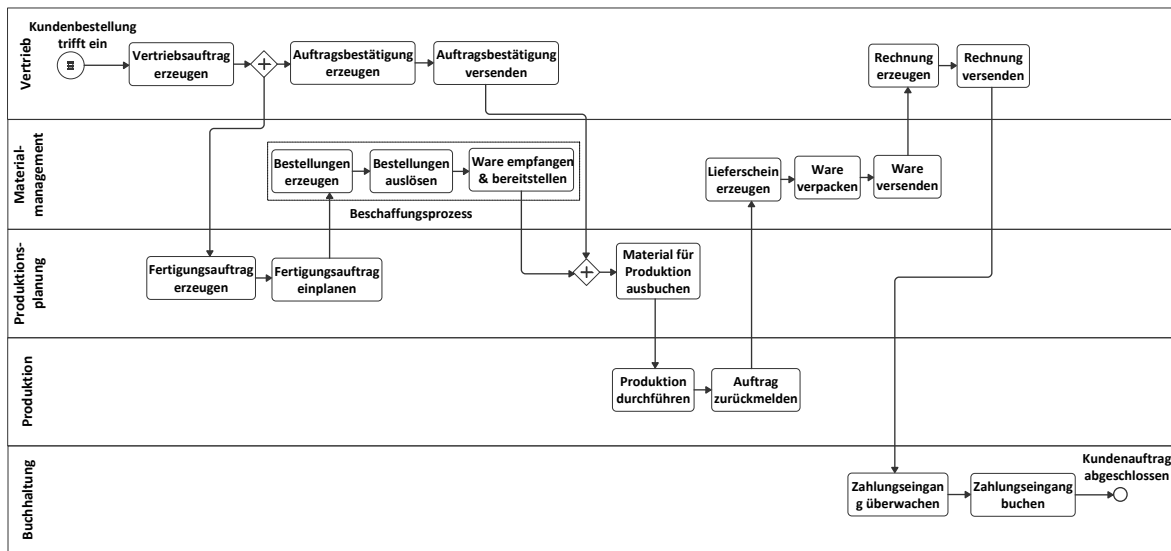


Abbildung 1: Beispielhafter MTO-Auftragsabwicklungsprozess

Quelle: Eigene Darstellung

Die Auftragszerzeugung definiert, wann ein neuer Fertigungsauftrag gestartet wird. Dabei wird zwischen den zwei Grundformen Auftrags- (Make-To-Order, MTO) und Lagerfertigung (Make-To-Stock, MTS) unterschieden. Bei der Auftragsfertigung werden Kundenaufträge unmittelbar an Fertigungsaufträge geknüpft. Auf diese Weise initiieren Kund:innen den Fertigungsauftrag. Die Lagerfertigung entkoppelt diesen Prozess und fertigt Produkte auf Basis einer Absatzprognose. Dadurch werden Kundenaufträge in der Regel unabhängig vom direkten Kundenauftrag direkt aus dem Lager bedient (Wiendahl, 2011, S. 281–283).

Zur Beschreibung der Unternehmensarchitektur im Fokus betrieblicher Informationssysteme unterscheiden Ferstl und Sinz im semantischen Objektmodell drei Modellebenen, wobei von einer Modellebene zur nächsten eine Detaillierung der übergeordneten Stufe stattfindet. Die drei Ebenen werden zunächst in eine Außen- und Innenperspektive aufgeteilt. Die Außenperspektive beinhaltet den Unternehmensplan und damit alle Objekte eines Unternehmens. In der Innenperspektive wird zwischen der Aufgabenebene und der Aufgabenträgerebene differenziert. Die Aufgabenebene enthält dabei das Geschäftsprozessmodell. In diesem Modell sind die Prozesse zur Erreichung des Unternehmensplans abgebildet. Auf der Aufgabenträgerebene befinden sich die Ressourcen für die jeweiligen Geschäftsprozesse. Die Ressourcenebene beinhaltet neben Personal sowie Maschinen und Anlagen auch die Anwendungssysteme (Ferstl & Sinz, 2013, S. 194–197). Zu den Anwendungssystemen der untersten Ebene gehören unter anderem auch das Enterprise Resource Planning (ERP). Durch die Möglichkeit der Steuerung von Informationsflüssen sowie die Integration verschiedenster Geschäftsprozesse in ein System sind diese Art von Anwendungssystemen in

den letzten Jahrzehnten zum unverzichtbaren Standardrepertoire im betrieblichen Umfeld herangewachsen (Caserio & Trucco, 2018, S. 14).

Wichtig für das Verständnis von ERP-Systemen sind die beiden Begriffe Stamm- und Bewegungsdaten. Erstere sind unabhängig von konkreten Aufträgen und werden häufig in vielfältigen Prozessen und Anwendungen verwendet. Bewegungsdaten entstehen direkt aus dem Geschäftsbetrieb heraus (Kurbel, 2016, S. 42). Während sich Bewegungsdaten häufig ändern können, sind Stammdaten meist über längere Zeit konstant. Zur Veranschaulichung soll an dieser Stelle der Auftragsabwicklungsprozess aus Abbildung 1 herangezogen werden. Der gesamte Prozess wird durch ein ERP-System begleitet. Bereits im ersten Prozessschritt, der Kundenauftragserzeugung, werden verschiedene Stammdaten genutzt. Zu erwähnen sind an dieser Stelle die Kundendaten wie Adresse, Zahlungsziel und Ansprechpartner. Bei Bedarf kann auf diese aus verschiedenen Anwendungen heraus referenziert werden. An mehreren Stellen im Prozess entstehen Bewegungsdaten. Ein Beispiel hierfür ist der erstellte Vertriebsauftrag als elektronisches Dokument. Dieser wird auch als Beleg bezeichnet. Viele Geschäftsprozesse in ERP-Systemen basieren auf diesen Belegen (Kurbel, 2016, S. 220).

Zur Beschreibung des Aufbaus von ERP-Systemen wird häufig die dreistufige Client-Server-Architektur herangezogen. Auf der Anwendersicht sind die jeweiligen Clients angesiedelt („Präsentation“). Diese werden auf der zweiten Ebene, der Anwendungsschicht, gebündelt. Diese Schicht verwaltet die Kommunikation zwischen den beiden anderen Ebenen und implementiert die Anwendungslogik („Anwendung“). Auf der dritten Ebene ist eine Datenbank vorzufinden („Datenhaltung“). Dort werden die Daten für das System vorgehalten (Bengel, 2014, S. 12–14).

### **3 Technologien**

Dieses Kapitel beschäftigt sich mit verschiedenen Technologien zur Digitalisierung und Automatisierung von Geschäftsprozessen mit dem Ziel, aktuelle Technologien im Rahmen der Geschäftsprozess-Lehre einzubeziehen sowie Herausforderungen und Potentiale bei der Umsetzung darzustellen. Die Technologien werden hierbei isoliert voneinander betrachtet.

#### **3.1 Radio Frequency Identification**

Bei RFID handelt es sich um eine Technologie zur kontaktlosen Objektidentifikation mit Hilfe von Radiowellen (Kern, 2006, S. 1). Funktionsfähige RFID-Systeme bestehen in der Regel aus mindestens drei Komponenten. Dazu zählen neben Lesegeräten und Transpondern auch die zugehörigen Informationssysteme. Lesegeräte sind meist stationäre Geräte mit externer Stromversorgung. Sie bestehen aus Antennen und Mikrochips, wobei die Antennen je nach genutzter Frequenz unterschiedlich ausgelegt sind (Kern, 2006, S. 33–34). Transponder sind grundsätzlich ähnlich aufgebaut.

In den meisten Fällen erzeugt eine Kombination aus Lesegerät und Low-end-Transponder lediglich eine Seriennummer. Diese Seriennummer oder „unique identifier“ (UID) muss zunächst mittels angeschlossener Informationssysteme verarbeitet werden (Thiesse, 2005, S. 107). Ergänzend zu den drei oben genannten Bestandteilen kann ebenso RFID-Middleware eingesetzt werden. Diese dient der Transformation und Bereitstellung von Daten und wird häufig zwischen RFID-Hardware und Anwendungssysteme geschaltet (Tamm & Tribowski, 2010, S. 20).

RFID-Systeme können anhand mehrerer Kriterien in Kategorien eingeteilt werden. Einen der wichtigsten Einflussfaktoren stellt dabei die Frequenz<sup>37</sup> dar. Hier kann im Wesentlichen zwischen drei verschiedenen Frequenzbereichen unterschieden werden: Die Niedrigfrequenz mit 125-135 Kilohertz („Low Frequency“), die Hochfrequenz mit 13,56 Megahertz („High Frequency“) und die Ultrahochfrequenz mit über 868 Megahertz („Ultra High Frequency“). Die Wahl des Frequenzbereiches ist entscheidend, da die wesentlichen technischen Rahmenbedingungen davon abhängen. So nehmen beispielsweise Energiebedarf, Reichweite und Größe der Antenne mit zunehmender Frequenz ab. Gleichzeitig steigen jedoch auch Reflexionen an Oberflächen und damit die Verluste. Daher existiert keine „one fits all“-Lösung. Stattdessen gilt es, ein an den geplanten Einsatz angepasstes System zu definieren (Kern, 2006, S. 41–43). Eine weitere Kategorisierungsmöglichkeit besteht in der Art der eingesetzten Transponder. Während aktive Transponder eine eigene Energieversorgung besitzen, werden passive Transponder per magnetischem (Niedrig- und Hochfrequenz) bzw. elektromagnetischem (Ultrahochfrequenz) Feld vom Lesegerät energetisch versorgt (Kern, 2006, S. 47). Weiterhin können Systeme anhand des Funktionsumfanges der Transponder unterschieden werden. Hier kann zwischen „Low-end-“ und „High-end“-Systemen unterschieden werden. Low-end-Systeme besitzen entweder keinen oder einen nur sehr schwachen Mikrochip zur Informationsverarbeitung. Mit diesen Systemen kann beispielsweise die Seriennummer ausgegeben werden. Die Vorteile dieser Art von Transpondern sind die kleinen Abmaße sowie die geringen Anschaffungskosten. Bei komplexeren Anwendungen eignen sich Transponder mit größerem Speicher und leistungsfähigeren Mikroprozessoren (Finkenzeller, 2015, S. 27–29). Je nach verwendetem Übertragungsverfahren eröffnet sich eine neue Art der Unterscheidbarkeit. Mit Übertragung ist die Kommunikation zwischen Lesegerät und Transponder gemeint. Dort wird zwischen kapazitiver und induktiver Übertragung sowie dem Backscatter-Verfahren unterschieden. Die kapazitive Übertragung nutzt ein elektrisches Feld, das durch Kondensatoren entsteht. Durch die Änderung des Feldes wird ein Signal erzeugt (Kern, 2006, S. 48–49). Induktive Verfahren nutzen

<sup>37</sup> Die Frequenz bezieht sich grundsätzlich auf die Frequenz des Lesegerätes, die meist mit der Frequenz des Transponders übereinstimmt. Dies ist allerdings nicht zwingend der Fall.

wechselnde magnetische Felder. Dazu wird vom Lesegerät über eine Spule ein magnetisches Feld erzeugt, das sich im Raum verteilt. Dieses Feld durchdringt auch die Antenne des Transponders, woraufhin eine Spannung am Transponder induziert wird. Dem magnetischen Feld wird dadurch Energie entzogen (Kern, 2006, S. 49–50). Die induktive Übertragung ist für kurze Strecken sehr zuverlässig und ist daher die meistgenutzte Form der Übertragung im Nieder- und Hochfrequenzbereich (Lampe et al., 2005, S. 74). Beim Backscatter-Verfahren werden über Dipolantennen elektromagnetische Wellen mit periodisch alternierender Resonanz reflektiert (Kern, 2006, S. 55–57). Dabei sind Distanzen von über einem Meter möglich.

### 3.2 Indoor Positioning System

Lokalisierungssysteme dienen der Positionsbestimmung von Objekten (Farid et al., 2013, S. 1). Dabei ist ein Indoor Positioning System (IPS) für den Einsatz in Gebäuden vorgesehen. Grundsätzlich kann unabhängig vom Ort des Einsatzes zwischen bekannten und unbekanntem Teilnehmern unterschieden werden. Bei einem bekannten Teilnehmer (Anchor) ist die Position initial bekannt. Im Gegenzug muss bei einem unbekanntem Teilnehmer (Tag) die Position zunächst berechnet werden (Koubâa & Jamâa, 2013, S. 463).

Während sich bei Positionierungssystemen im Außenbereich das Global Positioning System (GPS) als de facto Standard durchgesetzt hat, existieren im Innenbereich verschiedene Ansätze (Hameed & Ahmed, 2018, S. 1). Je nach Anwendungsfall muss dabei unter Berücksichtigung bestimmter Charakteristika das passende System ausgewählt werden. In der Literatur werden mehrere Metriken zur Systemevaluation herangezogen. Allen voran steht die Messgenauigkeit, welche die Abweichung der ermittelten Position von der exakten Position berechnet. Ebenso sind Auflösung, Skalierbarkeit und Reichweite von großem Interesse. Mit Auflösung ist die kleinste Positionsänderung gemeint, die hinreichend exakt ermittelt werden kann. Die Skalierbarkeit beschreibt die Fähigkeit, Systeme an neue Rahmenbedingungen anzupassen. Die Reichweite kann darüber hinaus ein wichtiges Maß sein, um Funktionalität zu gewährleisten. Abseits von technischen Kriterien entscheiden meist die Kosten für Implementierung und Instandhaltung über das gewählte Lokalisierungssystem (Farid et al., 2013, S. 2–3).

Wie zuvor bereits erwähnt muss die Position von Tags zunächst rechnerisch ermittelt werden. Dafür werden je nach verwendeter Methode unterschiedliche Algorithmen eingesetzt. In den meisten Fällen wird jedoch die Triangulation eingesetzt, weshalb der Fokus in diesem Beitrag auf dieser Methode liegt. Innerhalb der Triangulation existieren richtungs- und streckenorientierte Konzepte. Die Richtungsorientierung bezieht sich dabei im Wesentlichen auf den Winkel zwischen Sender und Empfänger zueinander (Angle of Arrival, AOA). Streckenorientierte Konzepte nutzen zur Standortbestimmung entweder die Stärke des zurückgegebenen Signals (Received Signal

Strength, RSS) oder berechnen die Strecke über die Dauer zwischen Senden und Empfangen des Signals (Farid et al., 2013, S. 3–4). Im Laborbeispiel wird die Methode „Time Difference of Arrival“ (TDoA) verwendet. Das TDoA-Prinzip berechnet die Distanz indirekt über den relativen Unterschied zwischen den Transitzeiten der unterschiedlichen Anchor-Tag-Strecken (Bensky, 2016, S. 169).

Die Nutzung von Signalen setzt voraus, dass eine gewisse Infrastruktur zur Signalübertragung existiert. Dabei können für Innenanwendungen zunächst drei verschiedene Technologien voneinander abgegrenzt werden: Infrarotstrahlung, Ultraschall sowie Radiowellen. Bei Infrarotstrahlung handelt es sich um elektromagnetische Wellen, die sich mit Lichtgeschwindigkeit ausbreiten. Die Wellenlänge liegt oberhalb des für den Menschen sichtbaren Lichts. Da Infrarotstrahlung leicht von anderen Objekten beeinflusst wird, muss häufig eine direkte Sichtverbindung bestehen (Hameed & Ahmed, 2018, S. 2). Ultraschall-Systeme kommunizieren im Frequenzbereich zwischen 0,02 und 1000 Megahertz und nutzen den Schall als Medium. Aufgrund der langsameren Geschwindigkeit von Schall gegenüber Licht sind die Messungen mit geringerem technischem Aufwand möglich. Dem entgegen stehen die negativen Reflexionseigenschaften von Ultraschallwellen (Retscher & Kistenich, 2006, S. 26). Die dritte Basistechnologie im Bereich der Signalübertragung sind Radiowellen. Diese haben den Vorteil, auch Wände durchdringen zu können und sind dadurch auch für Innenraumanwendungen mit mehreren abgegrenzten Räumen von Bedeutung. Innerhalb dieser Technologie existieren zahlreiche unterschiedliche Ausprägungen. Die wichtigsten Vertreter sind hier RFID, Bluetooth, Wireless Local Area Network (WLAN) und Ultra-wideband (UWB). Bluetooth ist im Bereich der Kommunikationstechnik ein weit verbreiteter Standard. Dabei werden Radiowellen im Gigahertz-Bereich genutzt, um Daten auszutauschen. Signifikanten Vorteilen wie hohe Sicherheit und niedrigem Stromverbrauch steht die hohe Latenz gegenüber. Wie Bluetooth basiert auch WLAN auf Radiowellen im Gigahertz-Bereich. Oftmals kann WLAN auf einer bereits bestehenden Infrastruktur aufbauen. Hierdurch wird die Bestimmung der Position ohne zusätzliche Hardware und damit ein kosteneffizienter Einsatz des Systems ermöglicht. Für Anwendungen im Bereich der Echtzeitüberwachung eignen sich solche Systeme aufgrund der hohen Latenz jedoch nicht (Farid et al., 2013, S. 5–8). UWB basiert auf dem kurzen Senden von Impulsen nacheinander mit niedrigem Tastgrad<sup>38</sup>. Im Gegensatz zu RFID-Systemen verwendet UWB mehrere Frequenzen gleichzeitig. Der zur Verfügung stehende Frequenzbereich liegt dabei zwischen 3,1 und 10,6 Gigahertz. Die Mindestbandbreite ist auf 500 Megahertz festgelegt (Federal Communications Commission, 2002, S. 15). Aufgrund dieser hohen Bandbreite können Störungen und Interferenzen mit anderen Technologien verhindert werden (Liu

<sup>38</sup> Der Tastgrad als Verhältnis von Impulsdauer zu Periodendauer gibt an, wie lange ein Signal im Vergleich zur Zyklusdauer abgegeben wird.



et al., 2007, S. 1074). Im Vergleich zu anderen Technologien sticht UWB durch hohe Auflösung sowie geringe Störanfälligkeit bei nicht-metallischen Gegenständen heraus. Besonders die guten Eigenschaften bei Anwendungen ohne direkten Sichtkontakt zwischen Sender und Empfänger (Non-Line-Of-Sight) manifestieren das große Potential der UWB-Technologie (Galler et al., 2006, S. 198–203).

### 3.3 Robotic Process Automation

Robotic Process Automation (RPA) ist ein Sammelbegriff für Softwarekomponenten, die auf der Benutzeroberfläche des Computers mit anderen Programmen interagieren. Da RPA typischerweise menschenähnliche Methoden in der Interaktion mit dem Computer verwendet, bleiben die betreffenden Systeme unverändert (van der Aalst et al., 2018, S. 269). Dabei muss zwischen Attended und Unattended RPA unterschieden werden. Bei Attended RPA wird der Softwareroboter direkt auf dem Bildschirm unter Aufsicht des Benutzers ausgeführt. Ähnlich wie Excel-Makros arbeitet der Softwareroboter auf Initiierung des Anwenders seine Tätigkeiten ab. Bei Unattended RPA kommen hingegen spezielle Server zum Einsatz. In diesem Zusammenhang existiert typischerweise keine Interaktion mit Anwendern (Langmann & Turi, 2020, S. 5–6). Für beide Arten von Softwarerobotern sind häufig wiederkehrende Tätigkeiten im Fokus, die eine klare Logik und nur wenige Ausnahmen vorweisen können (Scheer, 2020, S. 117–119). Prozesse mit hoher Variantenvielfalt lassen sich nur schwer oder unter hohen Kosten automatisieren und verbleiben daher meist beim Menschen (van der Aalst et al., 2018, S. 270–271).

Die Beweggründe zum Einsatz von RPA sind vielfältig. Einer der Treiber zum Einsatz von RPA sind die häufig anzutreffenden Systembrüche<sup>39</sup> in Geschäftsprozessen. Fehlende Schnittstellen in den einzelnen Systemen können durch RPA effektiv überbrückt werden, ohne die bestehende IT-Infrastruktur anpassen zu müssen. Durch die hohe Skalierbarkeit sind Softwareroboter bereits weit verbreitet. Bei kurzer Implementierungszeit können auf diese Weise Kosteneinsparungen und Effizienzsteigerungen zeitnah erzielt werden (Langmann & Turi, 2020, S. 8–11). In einer branchenübergreifenden Studie der Unternehmensberatung Capgemini mit über 150 Teilnehmenden vertrat eine große Mehrheit die Meinung, dass RPA Kosten reduzieren, Effizienz steigern und die Qualität verbessern kann (Capgemini, 2016, S. 7). Durch die vordefinierte Logik ist außerdem eine hohe Konformität mit Compliance-Richtlinien erzielbar (Hofmann et al., 2020, S. 101). RPA kann sich zudem durch die Art des Systemzugriffs von anderen Lösungen differenzieren. Während die meisten Business Process Automation-Systeme auf der Schicht der Anwendungslogik agieren, arbeiten die Softwareroboter direkt auf der Präsentationsebene.

<sup>39</sup> Mit Systembruch ist die Verwendung unterschiedlicher Medien und Systeme innerhalb von zusammenhängenden Prozessen gemeint.

Grundsätzlich arbeiten Softwareroboter ähnlich wie Menschen mit Tastatur- und Mauseingaben. Eine häufig genutzte Form des Anwendungszugriffs ist daher die Automatisierung über die Benutzeroberfläche. Für umfangreichere Systeme wie SAP existieren darüber hinaus eigene Programmierschnittstellen. Auch der direkte Zugriff auf zugrundeliegende Datenbanken aus RPA heraus ist realisierbar (Smeets et al., 2019, S. 16). Vor diesem Hintergrund kommen mehrere bereits etablierte Methoden zum Einsatz. Dies führt zu einer hohen Produktreife bei einfachen RPA-Systemen. Einfache RPA-Systeme sind Softwareroboter, die ohne größere Anwendung von künstlicher Intelligenz auskommen. Wird die einfache Robotic Process Automation um Methoden der künstlichen Intelligenz erweitert, spricht man von Intelligent Process Automation (IPA). Dabei liegt der Fokus auf der Automatisierung von höherwertigen und komplexeren Aufgaben. Beispiele hierfür sind natürlichsprachliche Dialoge im Kundenservice oder eigenständige statistische Auswertungen durch Roboter (Scheer, 2020, S. 126). Durch die Möglichkeiten der künstlichen Intelligenz, wie beispielsweise der Einsatz von künstlichen neuronalen Netzen, kann IPA auch unstrukturierte Daten<sup>40</sup> verarbeiten. Dadurch können Systeme eigenständig Muster in Daten erkennen und auch komplexere Entscheidungen lernen (Taulli, 2020, S. 38–43).

## 4 Integration der Technologien

Die Integration der einzelnen Technologien zu einem Gesamtsystem stellt eine wichtige und gleichzeitig komplexe Aufgabe dar. Damit die verschiedenartigen Systeme reibungslos zusammenarbeiten, sollten diese zu Beginn schematisch dargestellt werden. Dabei ist zudem auf den Informationsaustausch zwischen den Systemen zu achten. Ohne diese Kommunikation ist eine Verknüpfung und damit eine durchgängige Automatisierung nicht möglich. Nachfolgend wird zunächst auf Webservices als zentrale Kommunikationsschnittstelle zwischen Maschinen eingegangen. Im Anschluss folgt die Erläuterung einer Referenzarchitektur für Internet of Things (IoT) Anwendungen.

### 4.1 Webservices

Unter Webservices versteht man Softwarefunktionalitäten in der Kommunikation zwischen Maschinen (Machine-to-Machine). Über eine eindeutige Adresse (Uniform Resource Identifier) werden dabei mittels XML-Nachrichten über unterschiedliche Internetprotokolle ausgetauscht (W3C, 2017).

Im „Open Systems Interconnection“ (OSI) Referenzmodell wird ein universeller Ansatz zur Entwicklung von Kommunikationssystemen definiert. Dabei wird die Gesamtaufgabe in sieben aufeinander aufbauende Schichten unterteilt. Jede Schicht

<sup>40</sup> Unstrukturierte Daten haben keine vordefinierte Struktur. Beispiele hierfür sind Bilder, Videos und Tonaufnahmen.

übernimmt im Kontext von Rechnernetzen eine exklusive Aufgabe und nutzt jeweils die Dienste der nächstniedrigeren Ebene (Ferstl & Sinz, 2013, S. 417). Damit die Kommunikation zuverlässig funktioniert, müssen sich sowohl Sender als auch Empfänger am Modell orientieren. Zusätzlich finden zwischen Sender und Empfänger verschiedene Protokolle Anwendung. Protokolle stellen für die Kommunikation wichtige Regeln bereit. Hierzu werden Formate und Bedeutungen unterschiedlicher Begriffe festgelegt (Baun, 2020, S. 35).

Analog zum OSI-Referenzmodell werden beim Webservice Protocol Stack mehrere aufeinander aufbauende Schichten modelliert. Dabei kommen in der Regel vier Schichten zum Einsatz, die in der Literatur jedoch nicht einheitlich benannt werden. In Abbildung 2 ist eine Architekturvariante dargestellt. Beginnend auf der untersten Abstraktionsebene definiert die Transport Layer die Art der Nachrichtenübertragung durch das zu verwendende Protokoll. Bei Webservices wird häufig das Hypertext Transfer Protocol (HTTP) verwendet (Finger & Zeppenfeld, 2009, S. 56). In der Messaging Layer wird das Kommunikationsformat festgelegt. Hierzu wird häufig das Simple Object Access Protocol (SOAP) eingesetzt.

Bei SOAP handelt es sich um ein XML-basiertes Nachrichtenprotokoll. Eine SOAP-Nachricht besteht typischerweise aus den drei Komponenten Envelope, Header und Body. Der Envelope enthält als Wurzelement die Version des SOAP-Protokolls sowie gekapselt die beiden untergeordneten Elemente Header und Body. Der optionale SOAP Header enthält allgemeine Meta-Informationen wie beispielsweise Sicherheitsinformationen der Nachricht. Der Body ist hingegen verpflichtend, da hier die Nutzdaten verankert sind. Der eigentliche Nachrichteninhalte ist somit ausschließlich im Body zu finden (Melzer, 2010, S. 87–91). Der Aufbau von SOAP-Nachrichten kann ebenfalls Abbildung 2 entnommen werden.

In der Description Layer werden die notwendigen Informationen zur Ausführung von Webservices beschrieben. Hierfür wird der Standard Web Service Description Language (WSDL) genutzt. Dieser basiert ebenso wie SOAP auf XML (Melzer, 2010, S. 116). Die oberste Schicht wird als Discovery Layer bezeichnet. In dieser kann ein Webservice registriert und durch Dritte gefunden werden. Der standardisierte Verzeichnisdienst wird auch als Universal Description Discovery and Integration (UDDI) bezeichnet. Dieser speichert spezifische Daten des Webservices in einem WSDL-Dokument ab (Vaddi & Mohanty, 2019, S. 26).

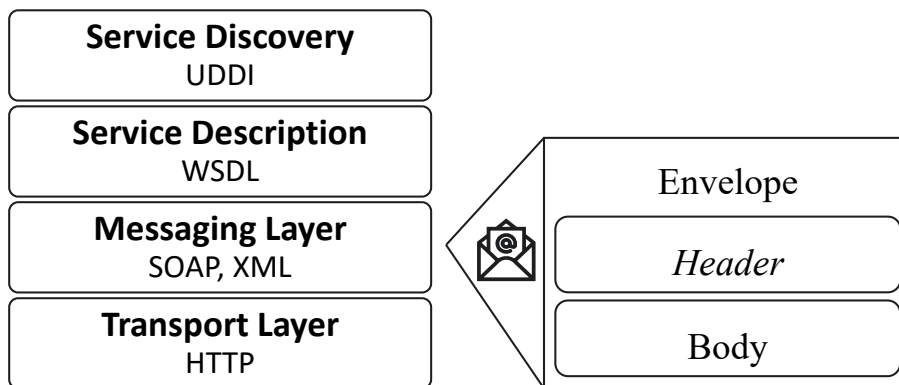


Abbildung 2: Webservice Protocol Stack

Quelle: Eigene Darstellung, in Anlehnung an Melzer, 2010, S. 87;

S.K.V. Jayakumar et al., 2014, S. 10

Allgemein können durch Webservices Daten in ERP-Systemen bzw. in den darauf aufbauenden Datenbanken über das Internet erstellt, gelesen oder geändert werden. Dies ist in vielen Fällen eine wichtige Voraussetzung zur Implementierung von IoT-Anwendungen.

## 4.2 Referenzarchitektur

Im IoT sind Objekte über ein Netzwerk miteinander verbunden. Über dieses Netzwerk können Geräte ohne menschliches Mitwirken untereinander kommunizieren und interagieren (Jeschke et al., 2017, S. 3). Durch die Verknüpfung mehrerer Geräte sollen technische Umgebungen intelligenter werden. Über meist durchgängig verfügbare drahtlose Verbindungstechnologien ist die Grundlage für smarte Umgebungen vorhanden (Zaheeruddin & Gupta, 2020, S. 6–8). Die Anzahl der Geräte sowie die Anbieter auf dem Markt sind in den letzten Jahren durchgängig gewachsen. Ein methodisches Vorgehen zur Integration der unterschiedlichen Objekte erscheint daher unerlässlich.

In der Literatur hat sich kein Konsens für eine einheitliche Referenzarchitektur durchgesetzt. Die meisten Modelle beinhalten drei oder fünf Schichten. Das Drei-Ebenen-Modell besteht aus den Schichten Perception, Network sowie Application Layer. Dieses Modell wird auch als konventionelle IoT-Architektur bezeichnet (Jabraeil Jamali et al., 2020, S. 3–4). Die unterste Schicht, die Perception Layer, besteht als physische Ebene aus Sensoren und Aktuatoren. Auf dieser Ebene werden Informationen wie beispielsweise Positionskoordinaten oder Temperatur erfasst. In der Network Layer werden Daten übertragen. Die Schicht ist demnach für die Kommunikation mit anderen Objekten und Servern zuständig. Dafür sind eindeutige Adressierung und Weiterleitung notwendig. Die Wahl des Netzwerks ist dabei unerheblich. Auf der obersten Ebene, der Application Layer, ist die Verbindung zum eigentlichen Nutzer gegeben. Dabei werden die Daten aus der Perception Layer zur Verfügung gestellt (Jabraeil Jamali et al., 2020, S. 3–13).

Ein weiteres Modell für IoT-Architekturen ist das Fünf-Schichten-Modell. Es erweitert das bestehende Modell um die beiden Ebenen Middleware Layer und Business Layer. Die Middleware Layer ist zwischen Network Layer und Application Layer angeordnet. Dies führt dazu, dass die Informationen aus der Network Layer zunächst in dieser Ebene verarbeitet werden. Durch beispielsweise universelle Berechnungen und vorgelagerte Logik-Entscheidungen werden Informationen vorverarbeitet. Eine Speicherung der Daten in Datenbanken ist ebenfalls möglich. Die oberste Ebene im Fünf-Schichten-Modell ist die Business Layer, die sich damit oberhalb der Application Layer befindet. Die Business Layer ist für die Koordination der einzelnen IoT-Geräte verantwortlich. Ebenso findet die Einbindung der Daten in die Geschäftslogik an dieser Stelle statt. Nur durch die sinnvolle Verknüpfung von individuellen Objekten zu smarten Systemen innerhalb eines durchdachten Geschäftsmodells kann IoT sein gesamtes Potenzial entfalten (Khan et al., 2012, S. 258–259).

Ähnlich wie das OSI-Referenzmodell durchlaufen die Informationen die jeweiligen Schichten sequenziell. Durch die Orientierung an dieser Architektur kann die Komplexität bei der Implementierung von IoT-Systemen bewältigt werden. Nachfolgend ist die geplante IoT-Architektur dargestellt. Dafür werden die in Kapitel 3 erläuterten Technologien und Methoden in ein gemeinsames Modell integriert. Als Grundmodell wird die fünfschichtige IoT-Referenzarchitektur verwendet.

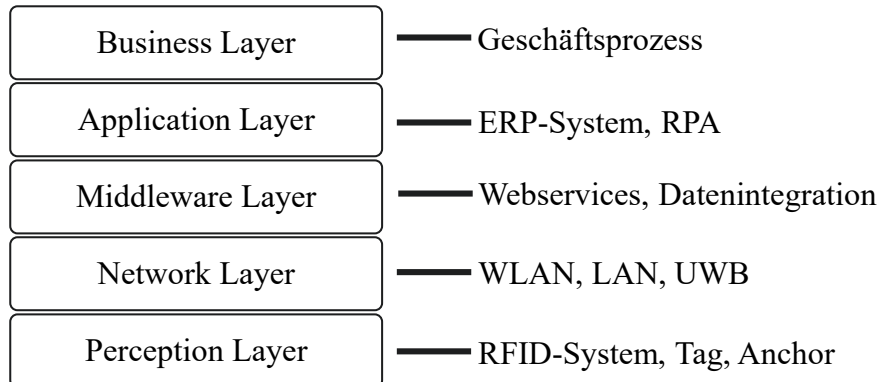


Abbildung 3: IoT-Referenzarchitektur

Quelle: Eigene Darstellung, in Anlehnung an Khan et al., 2012, S. 259

In der geplanten IoT-Architektur erfassen RFID-Systeme sowie die Tags und Anchors des IPS die Informationen. Über verschiedene Netzwerkkomponenten werden die Informationen dann kabelgebunden oder drahtlos (WLAN und UWB) weitergegeben. Über Webservices können die Informationen vorverarbeitet werden. Es folgen weitere Transformationen in der Middleware Layer. In der Application Layer werden schließlich die Informationen aus den zugrundeliegenden Ebenen verwertet. Hierbei werden RPA und ERP-Systeme genutzt. Auf der obersten Ebene sind schließlich die Geschäftsprozesse definiert.

## 5 Laborbeispiel

### 5.1 Vorstellung des Unternehmens

Das fiktive Unternehmen „ERP-Bar GmbH“ ist eine kleine Bar, welche die beiden Getränke Gin Tonic und Hugo vertreibt. Zur Veranschaulichung wurde für das Unternehmen eine eigene Webseite erstellt. Dort kann das gewünschte Getränk gewählt werden. Nach Getränkewahl gelangen die Kund:innen auf eine Unterwebseite, auf der nähere Anweisungen zum weiteren Vorgehen gegeben werden. Das gewünschte Produkt wird eigenständig nach Anleitung produziert. Da jedes Getränk durch den Bestellenden hergestellt wird, liegt die Losgröße grundsätzlich bei eins. Das Unternehmen arbeitet daher innerhalb einer MTO-Auftragsabwicklung. Auf diese Weise wird die Produktion eines neuen Endproduktes erst gestartet, wenn eine Bestellung aufgegeben wurde. Der Bestand an Endprodukten liegt dabei typischerweise bei null.

Nach Bestelleingang wird im Vertrieb ein Vertriebsauftrag erzeugt. Die Fertigungsplanung erstellt aus dem Vertriebsauftrag heraus einen Fertigungsauftrag. Der Auftrag wird dadurch in die Produktionsplanung überführt. Gleichzeitig werden etwaige Bestellungen erzeugt und der aufkommende Materialbedarf reserviert. Das Auslösen der Bestellungen ist die Aufgabe des Materialmanagements. Diese tragen ebenso Verantwortung für den Wareneingang sowie die Bereitstellung der bestellten Materialien für die Produktion. Die Herstellung der Endprodukte beginnt mit der Entnahme des jeweiligen Glases. Danach müssen die Zutaten geschnitten und in das Glas gegeben werden. Die Getränke werden dann im dritten Schritt gemäß Stückliste eingefüllt. Im letzten Arbeitsgang werden alle Zutaten miteinander vermischt. Das Endprodukt ist nun produziert. Durch das Materialmanagement wird der Auftrag versendet. Der physische Objektfluss ist an dieser Stelle beendet. Nachdem der Vertrieb die Rechnung erzeugt und versendet hat, überwacht die Buchhaltung den Zahlungseingang. Wenn der fällige Rechnungsbetrag als Zahlungseingang gebucht wird, ist der Auftrag vollständig beendet. Der gesamte Auftragsabwicklungsprozess ist mittels BPMN in Abbildung 4 dargestellt. Grau hinterlegte Felder sind im Laborbeispiel automatisiert worden.

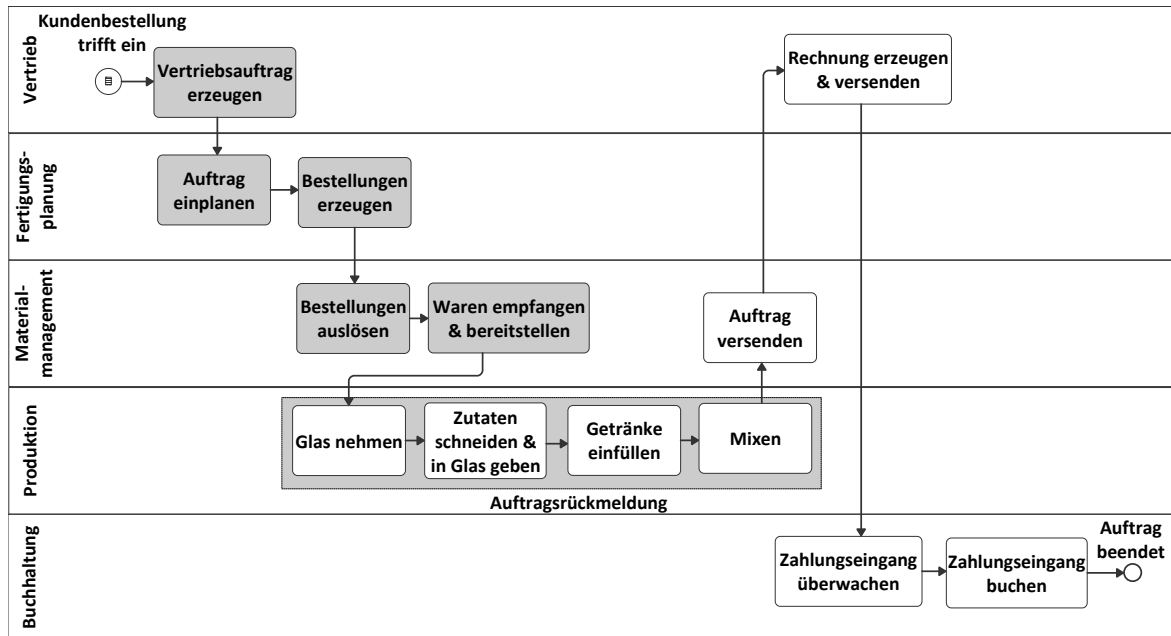


Abbildung 4: Auftragsabwicklungsprozess im Laborbeispiel  
Quelle: Eigene Darstellung

## 5.2 Verwendete Hard- und Software

Gemäß der Einteilung in die Fünf-Schichten-Architektur im Abschnitt 4.2 ist in Abbildung 5 die realisierte IoT-Architektur mit den verwendeten Hard- und Softwarelösungen dargestellt. Im darauffolgenden Absatz werden zudem die einzelnen Systeme sowie deren Funktionalität kurz erläutert.

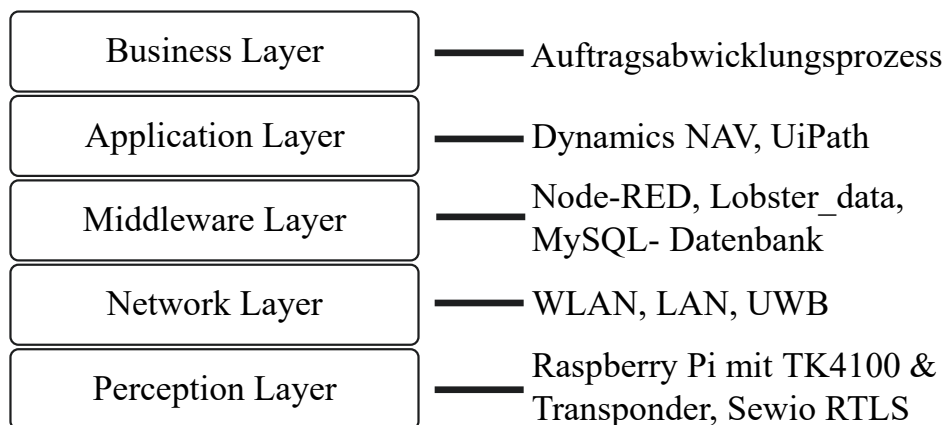


Abbildung 5: Realisierte IoT-Architektur im Laborbeispiel  
Quelle: Eigene Darstellung, in Anlehnung an Khan et al., 2012, S. 259

Microsoft Dynamics NAV ist eine ERP-Systemlösung, die sich vor allem an kleine und mittlere Unternehmen richtet. Im Laborbeispiel wird die Version Dynamics NAV 2018 genutzt. Als zentrales ERP-System dient Dynamics NAV der Unterstützung sämtlicher Geschäftsprozesse. Auf diese Weise können alle Aufgaben aus dem Auftragsabwicklungsprozess durch das ERP-System begleitet werden.

Zur Automatisierung werden im Laborbeispiel zwei verschiedene Webservices verwendet. Diese wurden mithilfe des „Microsoft SQL-Server Management Studios“ (SSMS) sowie innerhalb des „Microsoft Dynamics NAV Development Environment“ entwickelt. Das SSMS ist ein integratives Tool für SQL-Server und kann relationale Datenbanken verwalten und konfigurieren (Microsoft, 2019). Das Development Environment dient der Erstellung eigener Anwendungen. Innerhalb der Entwicklungsumgebung können verschiedene Arten von Objekten erstellt werden. An dieser Stelle sollen lediglich Objekte der Art Codeunit genannt werden. Codeunits sind Container für Programmcode in der Programmiersprache C/AL. Diese können im Anschluss von mehreren Applikationen genutzt werden.

Der Raspberry Pi ist ein Einplatinencomputer mit zahlreichen Anschlüssen (Follmann, 2018, S. 1). Über umfangreiche Softwarebibliotheken und vorgefertigte Programme kann der Raspberry Pi zügig in bestehende Architekturen eingebunden werden. Daher eignet sich diese Art von Systemen vor allem für Anwendungen im IoT-Umfeld (Molloy, 2016, S. 5–6). In diesem Laborbeispiel wird das Betriebssystem Raspberry Pi OS verwendet. Dieses basiert auf Debian. Zusätzlich wird der Raspberry Pi über WLAN mit dem Netzwerk sowie über USB mit dem RFID-Lesegerät verbunden. Als RFID-Lesegerät wird das Modell „TK4100“ eingesetzt. Dieses kann lediglich die eindeutige Seriennummer (UID) des Transponders auslesen und funkt auf einer Frequenz von 125 Kilohertz. Die Energiezufuhr des Transponders erfolgt passiv über das Lesegerät. Zum Einsatz kommt dabei das induktive Verfahren über Spulen. Es handelt sich somit um ein passives Low-End-System im Niedrigfrequenzbereich mit induktivem Übertragungsverfahren (siehe Abschnitt 3.1). Als zusätzliche Software wird für den Raspberry Pi die Programmierumgebung „Node-RED“ installiert. Dabei sendet der Raspberry Pi die UID des Transponders per Websocket an Windows. Websocket ist eine bidirektionale Verbindung im TCP-Standard. Dabei wird zu Beginn eine Verbindung eröffnet (Handshake), die danach immer wieder verwendet werden kann. Dadurch kann eine Nachricht gesendet werden, sobald sie verfügbar ist. Dies verbessert die Latenz signifikant. Im Gegenzug dazu muss bei konventionellen HTML-Anwendungen die Verbindung stets neu aufgebaut werden (Wang et al., 2013, S. 7–8).

Node-RED ist eine auf JavaScript-basierende graphische Programmierumgebung. Die zugrundeliegende Plattform ist die Node.js-Plattform. Durch diese erhält Node-RED einen browserbasierten Editor zum Erstellen und Verwalten von Flows. Innerhalb der Flows besteht die Auswahl aus verschiedenen Knotenpunkten (Nodes), die Funktionen darstellen. Die Nodes werden per Mausklick miteinander verbunden. Auf diese Weise können umfangreiche Transformationen und Verbindungen zu anderen Systemen zügig realisiert werden (Lekic & Gardasevic, 2018, S. 1–2). Im Laborbeispiel sind zwei Node-RED-Instanzen notwendig. Der erste Flow läuft auf dem Raspberry Pi, der zweite Flow unter Windows. Dort werden zunächst die ankommenden



Informationen des Raspberry Pi empfangen. Nach einem klaren Algorithmus und mehreren Verarbeitungsstufen sendet Node-RED eine SOAP-Nachricht per HTTP-POST an den Webservice des ERP-Systems. Damit wird der Webservice „CreateSalesOrder“ in Anspruch genommen. Dieser erzeugt im ERP-System einen Verkaufsauftrag und gibt die Auftragsnummer als Rückgabewert aus. Die Auftragsnummer wird anschließend extrahiert und in eine Textdatei geschrieben, die RPA später auslesen kann. Unmittelbar danach wird RPA über das Systemkommando gestartet.

Als RPA-Lösung wird im Laborbeispiel die Studio-Version von UiPath in der Community Edition verwendet. Der erstellte Verkaufsauftrag des Webservices wird über mehrere Maus- und Tastatureingaben in die Produktion eingeplant. Dies entspricht den Arbeitsgängen „Auftrag einplanen“ bis „Ware empfangen & bereitstellen“ aus Abbildung 4. Die restlichen Schritte werden über das IPS in Verbindung mit einem Webservice realisiert.

Das Unternehmen Sewio Networks bietet Systemlösungen für IPS an. In diesem Kontext kommt das „Real-Time Location System“ (RTLS) zum Einsatz. RTLS ist eine auf UWB-basierende Plattform zur vorwiegenden Lokalisierung von Objekten im Innenraum. Dabei wird das TDoA-Schema zur Berechnung der Distanz genutzt. Das System besteht aus Anchors, Tags sowie einer Software (Sewio Networks, o. J.). Der Tag sendet dabei regelmäßige Signale an die Anchors. Fünf Anchor sind fest an der Wand montiert und über Ethernet-Kabel miteinander verbunden. Das Layout der Produktion ist in Abbildung 6 zu sehen. Da mehrere Anchors zu unterschiedlichen Zeiten die Signale empfangen, kann die Position berechnet werden. Die Systemkonfiguration wird über die mitgelieferte Software „RTLS Studio“ durchgeführt. Die wichtigsten Module innerhalb der Software sind der „RTLS Manager“ sowie die durch die Hochschule bereitgestellte MySQL-Datenbank. Über den RTLS Manager können Anchors, Tags und die Zonen eingerichtet werden. Zudem wird dort das Layout des Innenbereichs hochgeladen. Durch den hinterlegten Maßstab kann das RTLS die Übergänge zwischen den Zonen präzise erkennen. Die angeschlossene MySQL-Datenbank dient als Grundlage für die Nutzung des Webservices zur automatischen Auftragsrückmeldung unter Zuhilfenahme des Tools Lobster data.

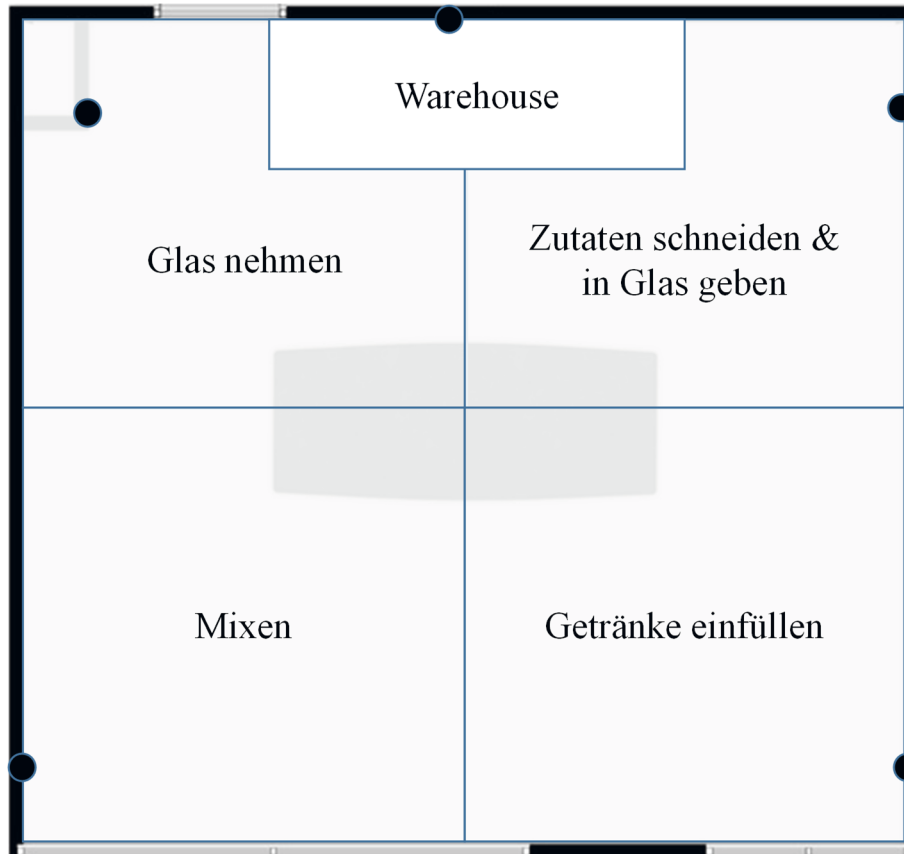


Abbildung 6: Layout IPS mit Positionen der Anchor (schwarze Punkte)

Quelle: Eigene Darstellung

Lobster data ist ein Tool zur Datenintegration. Über mehrere Verarbeitungsstufen kann ein elektronischer Datenaustausch zwischen verschiedenen Quell- und Anwendungssystemen realisiert werden. Der browserbasierte Client ist dabei hardwareunabhängig erreichbar. Das System arbeitet als Middleware in Profilen. Profile sind bei Lobster data mit ausgeführten Prozessen gleichzusetzen. Notwendige Definitionen sind ebenfalls dort deklariert (Lobster DATA, 2018, S. 2–6). Im Laborbeispiel werden zwei Profile benötigt. Im ersten Profil wird eine erste Relation im Intervall von drei Sekunden abgefragt. Bei einem Zonenwechsel wird ein Eintrag in einer zweiten Relation erzeugt. Innerhalb des zweiten Profils wird im gleichen Intervall eine SQL-Abfrage auf die erste Relation durchgeführt. Ändert sich der Status eines Tags erneut aufgrund eines Zonenwechsels, kann die Dauer eines Arbeitsgangs durch die Zeitdifferenz zwischen den beiden Relationen berechnet werden. Danach werden die Daten an die Zielstruktur angepasst und als SOAP-Nachricht formatiert. Nach Übermittlung der Daten an den Webservice wird die zweite Relation wieder geleert und die beiden Profile können von vorne beginnen.

### 5.3 Ablauf

Zwischen den einzelnen Systemen werden unterschiedliche Informationen ausgetauscht. In diesem Abschnitt soll der Ablauf des Laborbeispiels nochmals kurz umrissen werden.

Im ersten Schritt halten die Kunden den RFID-Transponder gemäß Getränkewahl an das Lesegerät. Das Lesegerät „TK4100“ empfängt dabei induktiv die UID des RFID-Transponders. Diese wird in Node-RED als Tastatureingabe erfasst und als Klartext weitergegeben. Über Websocket erfolgt die Datenübertragung zum zweiten Node-RED-Flow. Dieser erzeugt eine SOAP-Nachricht. Je nach Transponder wird Gin Tonic oder Hugo bestellt. Über einen Webservice wird per HTTP-POST-Methode in Microsoft Dynamics NAV ein Verkaufsauftrag angelegt. Nach erfolgreicher Erstellung wird der Verkaufsauftrag durch das RPA-Tool UiPath Studio in die Fertigung eingeplant. Danach kommt für die Auftragsrückmeldung das RTLS-System zum Einsatz. Die Tags des RTLS senden per UWB Impulse an die Anchors. Diese leiten die Daten per Ethernet-Kabel an die Software weiter. Mit TDoA wird dabei an zentraler Stelle auf Basis der Transitzeiten die Position des Tags berechnet und in eine Relation der MySQL-Datenbank geschrieben. Auf diese Datenbank greift das Middleware-Tool Lobster data zu. Mittels mehrerer SQL-Befehle können auf diese Weise ebenfalls Daten per SOAP-Nachricht an den Webservice des ERP-Systems gesendet werden. Die Funktionsweise der SQL-Befehle ist in Abschnitt 5.2 beschrieben. Auf der Applikationsebene folgt das ERP-System Dynamics NAV den Richtlinien der Business Layer. Innerhalb dieser ist der Prozess der Auftragsabwicklung definiert. In Abbildung 7 sind die einzelnen Schritte in Form eines BPMN zusammengefasst. Die Swimlanes richten sich nach der Einteilung in die IoT-Architektur gemäß Abbildung 5.

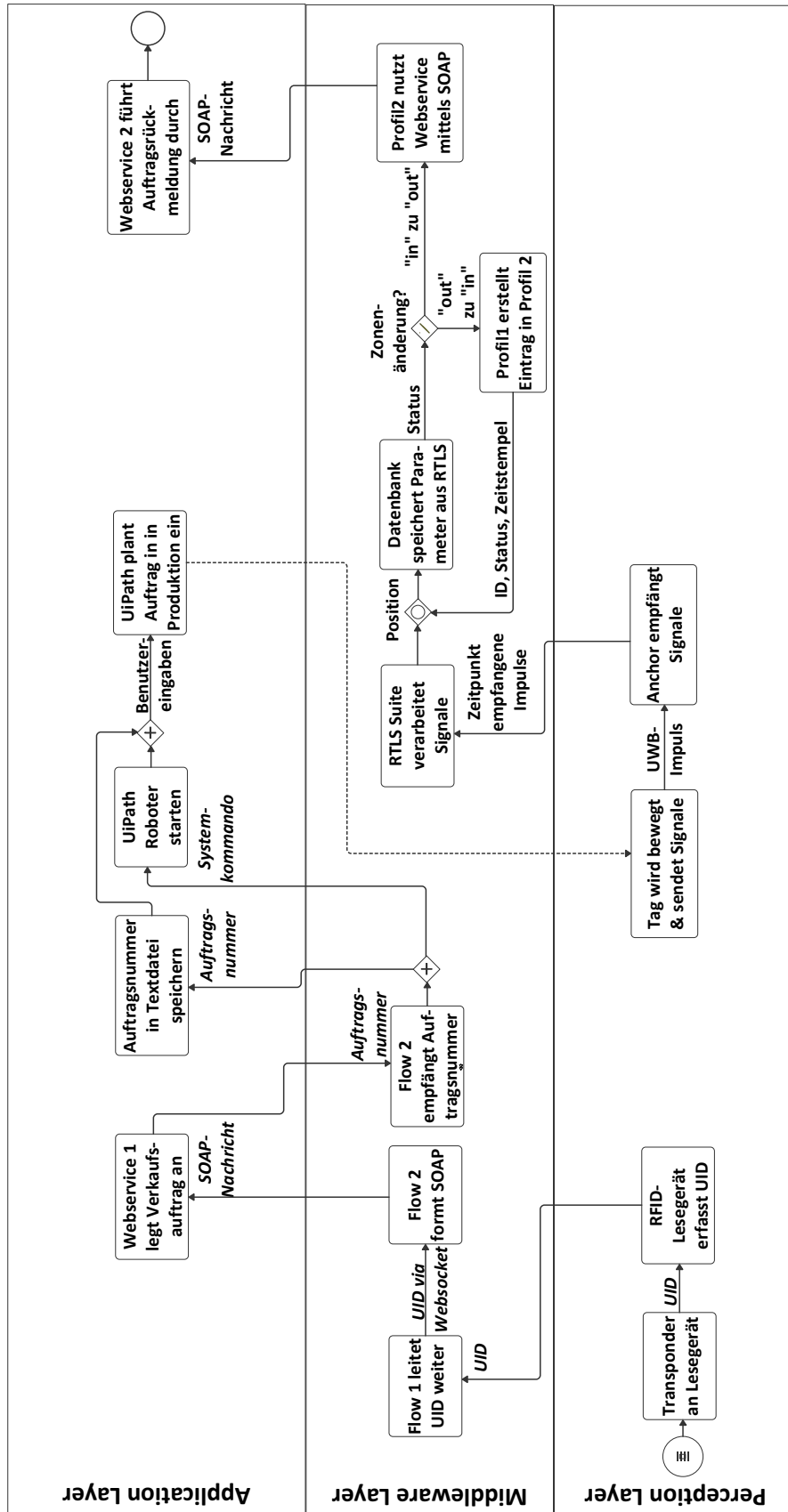


Abbildung 7: Auftragsabwicklungsprozess im Laborbeispiel

Quelle: Eigene Darstellung

## 6 Fazit

Im Zuge dieses Beitrags wurde untersucht, wie ein Auftragsabwicklungsprozess durch verschiedene Technologien digitalisiert und weitestgehend automatisiert werden kann. Hierfür kamen die Technologien RFID, IPS und RPA zum Einsatz. Durch das Laborbeispiel wurde die Umsetzung innerhalb einer vereinfachten Umgebung mit ausgewählter Hard- und Software realisiert. Durch die verschiedenen Technologien konnten die manuellen Systemeingaben deutlich reduziert werden, indem neun von 17 Arbeitsschritten automatisiert wurden. Ebenso konnte die Durchlaufzeit von Prozessen reduziert werden. Durch die größtenteils abteilungsübergreifenden Tätigkeiten ist erfahrungsgemäß mit einer prozessualen Durchlaufzeit von über einer Stunde zu rechnen. Durch eine konsequente Automatisierung kann diese bei Vernachlässigung der Fertigungszeit auf unter zwei Minuten verringert werden. Gleichzeitig wird die Datenkonsistenz gesteigert. Die jeweiligen Programme folgen dem Algorithmus und verarbeiten die Daten stets gleich. Dennoch ergeben sich auch entscheidende Nachteile. Der Einsatz mehrerer verschiedener Technologien benötigt ein hohes Maß an Systemverständnis aufgrund der zahlreichen Schnittstellen. Im Laborbeispiel folgen mehr als zehn verschiedene Schnittstellenaufrufe. Dies führt zu einer erhöhten Komplexität, zum anderen zu vermehrten Portfreigaben. Damit muss das System zwingend auch vor dem Hintergrund der Informationssicherheit betrachtet werden. Ein weiterer Kritikpunkt ist die teilweise mangelnde Flexibilität der verwendeten Systeme. Besonders im Bereich von RPA können Änderungen der Benutzeroberfläche kritisch sein. Auch eine Änderung der Bildschirmauflösung muss bei der Bilderkennung berücksichtigt werden. Beim IPS sei auf die Stammdatenkonsistenz hingewiesen. Ändert sich beispielsweise der Arbeitsplan eines Produktes, muss dies ebenfalls im RTLS konfiguriert werden. Auch die verwendete Hardware muss intakt sein. An dieser Stelle sei besonders die Funktionsfähigkeit des Raspberry Pi und des IPS genannt. Im Laborbeispiel wurden limitierende Rahmenbedingungen gesetzt. In Zukunft sollte daher ein besonderer Fokus auf der Weiterentwicklung des Systems hin zu mehr Flexibilität sein. Auch die Einbindung künstlicher Intelligenz im Rahmen der Prozessautomatisierung (IPA) sollte weiter erforscht werden.

## 7 Literaturverzeichnis

- Barton, T., Müller, C. & Seel, C. (2017): Geschäftsprozesse. Springer Fachmedien Wiesbaden.
- Baun, C. (2020): Computernetze kompakt: Eine an der Praxis orientierte Einführung für Studium und Berufspraxis. Springer Vieweg, Berlin, Heidelberg.
- Bengel, G. (2014): Grundkurs Verteilte Systeme: Grundlagen und Praxis des Client-Server und Distributed Computing. Springer Vieweg, Wiesbaden.

- Bensky, A. (2016): *Wireless positioning technologies and applications*. Artech House, Boston, Mass.
- Capgemini (2016): *Robotic Process Automation - Robots conquer business processes in back offices* [Online]. Abgerufen am 28.09.2021 von <https://www.capgemini.com/consulting-de/wp-content/uploads/sites/32/2017/08/robotic-process-automation-study.pdf>.
- Caserio, C. & Trucco, S. (2018): *Enterprise Resource Planning and Business Intelligence Systems for Information Quality: An Empirical Analysis in the Italian Setting*. Springer, Cham.
- Farid, Z., Nordin, R. & Ismail, M. (2013): „Recent Advances in Wireless Indoor Localization Techniques and System“. In: *Journal of Computer Networks and Communications*. Vol. 2013, S. 1–12.
- Federal Communications Commission (2002): *FCC 02-48: Revision of Part 15 of the Commission’s Rules Regarding Ultra-Wideband Transmission Systems* [Online]. Abgerufen am 28.09.2021 von [https://transition.fcc.gov/Bureaus/Engineering\\_Technology/Orders/2002/fcc02048.pdf](https://transition.fcc.gov/Bureaus/Engineering_Technology/Orders/2002/fcc02048.pdf).
- Ferstl, O. K. & Sinz, E. J. (2013): *Grundlagen der Wirtschaftsinformatik*. 7. Aufl., Oldenbourg, München.
- Finger, P. & Zeppenfeld, K. (2009): *SOA und WebServices*. Springer-Verlag, Berlin.
- Finkenzeller, K. (2015): *RFID-Handbuch: Grundlagen und praktische Anwendungen von Transpondern, kontaktlosen Chipkarten und NFC*. 7. Aufl., Hanser, München.
- Follmann, R. (2018): *Das Raspberry Pi Kompendium*. Springer-Verlag, Berlin, Heidelberg.
- Gadatsch, A. (2015): *Geschäftsprozesse analysieren und optimieren: Praxistools zur Analyse, Optimierung und Controlling von Arbeitsabläufen*. Springer Vieweg, Wiesbaden.
- Galler, S., Schroeder, J., Rahmatollahi, G., Kyamakya, K. & Jobmann, K. (2006): „Analysis and Practical Comparison of Wireless LAN and Ultra-Wideband Technologies for Advanced Localization“. In: *2006 IEEE/ION Position, Location, And Navigation Symposium*. Coronado, CA, April 25-27, 2006, IEEE, S. 198–203.
- Gartner (2020): *Gartner Top Strategic Technology Trends for 2021* [Online]. Abgerufen am 28.09.2021 von <https://www.gartner.com/smarterwithgartner/gartner-top-strategic-technology-trends-for-2021>.

- Hameed, A. & Ahmed, H. A. (2018): „Survey on indoor positioning applications based on different technologies“. In: 2018 12<sup>th</sup> MACS. Karachi, Pakistan, 11/24/2018 - 11/25/2018. Piscataway, NJ, IEEE, S. 1–5.
- Hammer, M. & Champy, J. (1993): Reengineering the corporation: A manifesto for business revolution. HarperBusiness, New York, NY.
- Hierzer, R. (2020): Prozessoptimierung 4.0: Den digitalen Wandel als Chance nutzen. 2. Aufl., Haufe Group, Freiburg im Breisgau.
- Hofmann, P., Samp, C. & Urbach, N. (2020): „Robotic process automation“. In: Electronic Markets. Vol. 30, No. 1, S. 99–106 [Online]. doi: 10.1007/s12525-019-00365-8.
- Jabraeil Jamali, M. A., Bahrami, B., Heidari, A., Allahverdizadeh, P. & Norouzi, F. (2020): Towards the Internet of Things. Springer International Publishing, Cham.
- Jeschke, S., Brecher, C., Meisen, T., Özdemir, D. & Eschert, T. (2017): „Industrial Internet of Things and Cyber Manufacturing Systems“. In: Jeschke, S., Brecher, C., Song, H. & Rawat, D. B. (Hg.), Industrial Internet of Things. Springer International Publishing, Cham, S. 3–19.
- Kern, C. (2006): Anwendung von RFID-Systemen. Springer, Berlin.
- Khan, R., Khan, S. U., Zaheer, R. & Khan, S. (2012): „Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges“. In: 2012 10<sup>th</sup> FIT. Islamabad, Pakistan, 12/17/2012 - 12/19/2012, IEEE, S. 257–260.
- Koubâa, A. & Jamâa, M. B. (2013): „Taxonomy of Fundamental Concepts of Localization in Cyber-Physical and Sensor Networks“. In: Wireless Personal Communications. Vol. 72, No. 1, S. 461–507.
- Kurbel, K. (2016): Enterprise resource planning and supply chain management in der Industrie: Von MRP bis Industrie 4.0. 8. Aufl., Walter de Gruyter GmbH, Berlin, Boston.
- Lampe, M., Flörkemeier, C. & Haller, S. (2005): „Einführung in die RFID-Technologie“. In: Fleisch, E. & Mattern, F. (Hg.), Das Internet der Dinge: Ubiquitous Computing und RFID in der Praxis: Visionen, Technologien, Anwendungen, Handlungsanleitungen. Springer-Verlag, Berlin, Heidelberg, S. 69–86.
- Langmann, C. & Turi, D. (2020): Robotic Process Automation (RPA) - Digitalisierung und Automatisierung von Prozessen. Springer Fachmedien Wiesbaden, Wiesbaden.
- Lekic, M. & Gardasevic, G. (2018): „IoT sensor integration to Node-RED platform“. In: 2018 17<sup>th</sup> INFOTEH. East Sarajevo, 3/21/2018 - 3/23/2018, IEEE, S. 1–5.

- Liu, H., Darabi, H., Banerjee, P. & Liu, J. (2007) „Survey of Wireless Indoor Positioning Techniques and Systems“. In: IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews), Vol. 37, No. 6, S. 1067–1080.
- Lobster DATA (2018): Lobster\_data Handbuch für Einsteiger [Online]. Abgerufen am 28.09.2021 von [https://www.lobster-world.com/wp-content/uploads/2018/10/Lobster\\_data\\_Einsteiger\\_Handbuch\\_2018.pdf](https://www.lobster-world.com/wp-content/uploads/2018/10/Lobster_data_Einsteiger_Handbuch_2018.pdf).
- Melzer, I. (2010): Service-orientierte Architekturen mit Web Services: Konzepte – Standards – Praxis. Spektrum Akademischer Verlag, Heidelberg.
- Microsoft (2019): SQL Server Management Studio (SSMS) [Online]. Abgerufen am 22.07.2021 von <https://docs.microsoft.com/de-de/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver15>.
- Molloy, D. (2016): Exploring Raspberry Pi: Interfacing to the real world with embedded Linux. John Wiley & Sons Inc, Indianapolis, IN.
- Retscher, G. & Kistenich, M. (2006): „Vergleich von Systemen zur Positionsbestimmung und Navigation in Gebäuden“. In: ZfV – Zeitschrift für Geodäsie, Geoinformation und Landmanagement. Vol. 131, No. 1, S. 25–35.
- S.K.V. Jayakumar, Singh, J. & Joseph, K. S. (2014): „Suitable QoS Parameters Survey for Standard Web Services & Web Applications to Understand their Cloud Deployability“. In: International Journal of Computational Intelligence and Informatics. Vol. 4, S. 1–18.
- Scheer, A.-W. (2020): Unternehmung 4.0. Springer Fachmedien Wiesbaden, Wiesbaden.
- Sewio Networks (o. J.): Real-Time Location System (RTLS) on Ultra-wideband | Sewio RTLS [Online]. Abgerufen am 28.09.2021 von <https://www.sewio.net/real-time-location-system-rtls-on-uwband>.
- Siderska, J. (2020): „Robotic Process Automation — a driver of digital transformation?“. In: Engineering Management in Production and Services. Vol. 12, No. 2, S. 21–31.
- Smeets, M., Erhard, R. & Kaußler, T. (2019): Robotic Process Automation (RPA) in der Finanzwirtschaft. Springer Fachmedien Wiesbaden, Wiesbaden.
- Tamm, G. & Tribowski, C. (2010): RFID. Springer-Verlag Berlin Heidelberg, Berlin.
- Taulli, T. (2020): The Robotic Process Automation Handbook. Apress, Berkeley, CA.
- Thiesse, F. (2005): „Architektur und Integration von RFID-Systemen“. In: Fleisch, E. & Mattern, F. (Hg.), Das Internet der Dinge: Ubiquitous Computing und



RFID in der Praxis: Visionen, Technologien, Anwendungen, Handlungsanleitungen. Springer-Verlag, Berlin, Heidelberg, S. 101–117.

- Vaddi, S. & Mohanty, H. (2019): „Webservice Specification and Discovery“. In: Mohanty, H. & Pattnaik, P. K. (Hg.), Webservices. Springer Singapore, Singapore, S. 25–51.
- van der Aalst, W. M. P., Bichler, M. & Heinzl, A. (2018): „Robotic Process Automation“. In: Business & Information Systems Engineering. Vol. 60, No. 4, S. 269–272.
- vom Brocke, J. & Rosemann, M. (2014): Handbook on Business Process Management 1: Introduction, Methods, and Information Systems. Springer-Verlag, Berlin, Heidelberg.
- W3C (2017): Web Services Architecture Requirements [Online]. Abgerufen am 28.09.2021 von <https://www.w3.org/TR/wsa-reqs/>.
- Wang, V., Salim, F. & Moskovits, P. (2013): The Definitive Guide to HTML5 WebSocket. Apress, Berkeley, CA.
- Wiendahl, H.-H. (2011): Auftragsmanagement der industriellen Produktion. Springer-Verlag, Berlin, Heidelberg.
- Zaheeruddin & Gupta, H. (2020): „Foundation of IoT: An Overview“. In: Alam, M., Shakil, K. A. & Khan, S. (Hg.), Internet of Things (IoT). Springer International Publishing, Cham, S. 3–24.