DOI: 10.20378/irb-56288

EDUCATIONAL SYSTEMS

Walter Augsburger
Forschungszentrum
Stiftung Rehabilitation Heidelberg
Postfach 101 409
6900 Heidelberg
West Germany

Abstract

An Educational System (ES) can be said to include several subsystems each of which serves a specific purpose.

Educational systems may thus be subdivided into sections which are dedicated to the actual subject matter, the methods of teaching, the media, the students, the teachers, and to supporting organisations. The interrelations between such subsystems are presented in two-dimensional projects to the extent to which they are significant for Computer Assisted Instruction (CAI).

The CAI-system and the dialogue languages must meet quality standards which permit the realisation of various teaching techniques and different methods of computer assistance.

One such system exists in the "Stiftung Rehabilitation", a foundation at Heidelberg, Germany. The system uses APL and satisfies nearly all requirements which a CAI-system should meet. The realisation of this system is described herein.

1. THE EDUCATIONAL SYSTEM (ES) AS PART OF A SUPRASYSTEM

The term "educational system" comprises all instruments, methods, and persons which participate in the process of formal education and vocational training.

Each ES serves a purpose which in turn governs the processes and objects within the ES. As such processes are running, the structure of the objects or the relations among them or their attributes will change. The ES actually serves the purposes of a superior system (suprasystem) as shown in Figure 1.

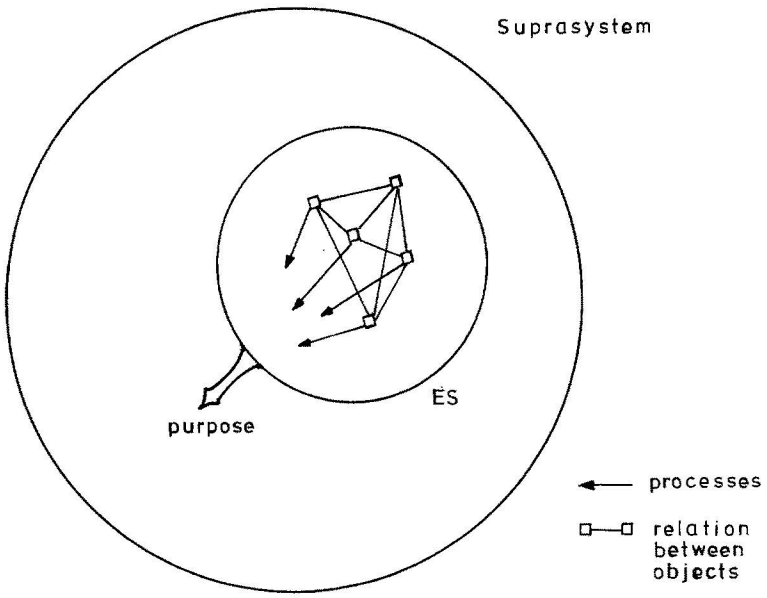


Fig. 1: Educational system, an object of a suprasystem

The general purpose of an ES, described in the most common terms, is to bring about a change in the behavior of the students. It is thus feasible to observe students as well as teachers from the point of view of the ES or the suprasystem.

When analyzing an ES used in vocational rehabilitation, the following hierarchy will be found:

Level I: Society
Level II: Sponsors of rehabilitation institutions
Level III: Rehabilitation institutions
- Vocational institutes
  . Vocational training centers (adults)
  . Vocational schools (adolescents)
- Formal schools (children)
- Rehabilitation hospitals, practising vocational therapy
- Other institutions

Level IV: ...

Figure 2 shows the dependencies of a vocational rehabilitation center (BFW) on other institutions in a schematic representation.

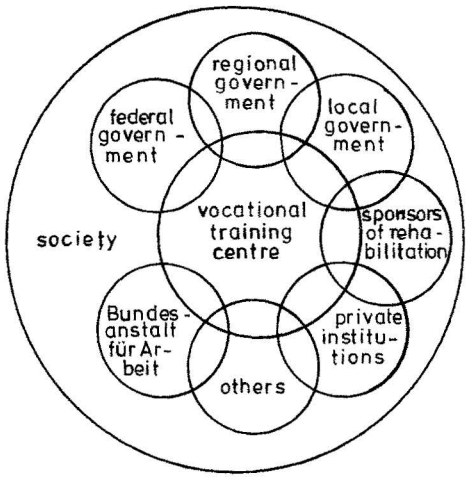


Fig. 2: Vocational rehabilitation center and other institutions

This vocational rehabilitation center may be structured in partial systems, see Figure 3.

2. IMPORTANT SUBSYSTEMS OF AN ES

Let us concentrate on some elements of an ES which are decisive for the use of the medium computer:

- learning goals
- subject matter
- teaching methods
- media
- students
- teachers/instructors
- educational institution.

These elements can be viewed as partial systems which depend on each other. The complex interdependencies may be depicted in a multi-dimensional space but we intend to show only the important ones using two-dimensional projection.

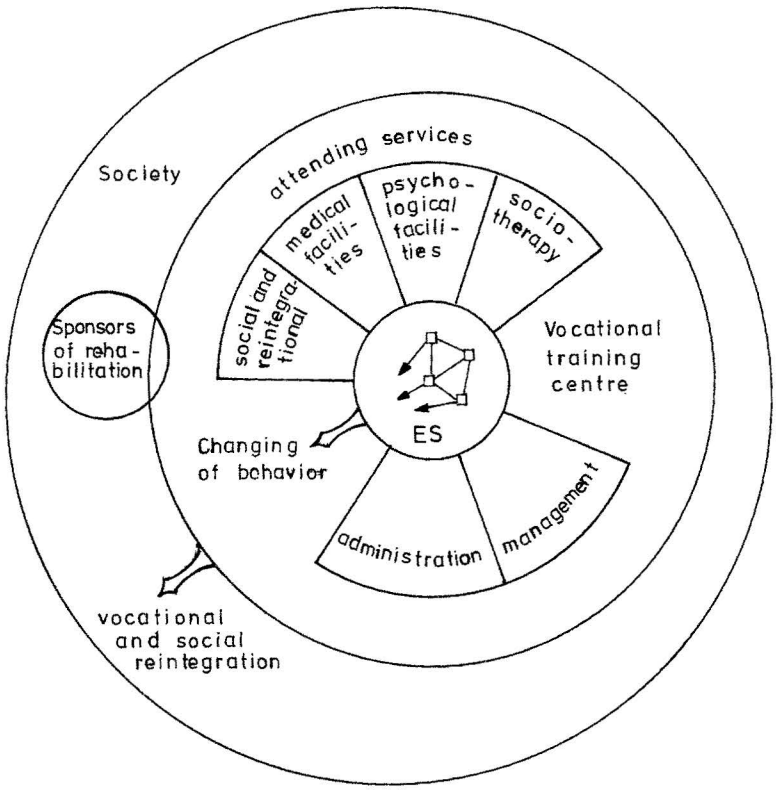


Fig. 3: Partial systems of a vocational rehabilitation center

First, some classification systems for several elements shall be quoted, without explaining the terminology, as there is pertinent literature in abundance [5, 9, 11, 18, 19, 26].

Classification systems quite often tend to be arbitrary, subjective, and depend on the aim of the classification.

2.1 Learning Goals

Besides the actual subject matter the following general goals are to be achieved:

- training the power of concentration and recollection
- increasing the ability to recognize problems and to structure and solve them
- advancing the problem solving behavior and the learning by observation and discovery
- increasing the ability to abstract and to comprehend facts linguistically and formally
- strengthen the ability to think in models and algorithms
- advancing the ability to apply the basic principles behind acquired knowledge to similar facts and to check the validity of such assumptions
- increase the ability to investigate and assess facts and to make decisions
- increase the objective assessment of subjective processes
- advancing social and vocational adaptability by training the intellectual flexibility.

The various processes in an ES are intended to bring about a change in behavior so that this behavior eventually matches the learning goals.

The adaptation processes may take place on various levels. Different hierarchical classifications may be discerned for the cognitive, affective and psycho-motorical fields of learning:

- Taxonomy for ways of learning according to GAGNE [11]
  - signal learning
  - stimulation - reaction learning
  - formation of chains

  . linguistic associations
  . discriminative learning
  . learning by conceptions
  . rule learning
  . problem solving
- Taxonomy covering the cognitive field of learning and learning goals in the fields of perception, memory and reasoning, according to BLOOM [5]
  . knowledge
  . comprehension
  . application
  . analysis
  . synthesis
  . evaluation
- Taxonomy covering the affective field of learning and learning goals in the fields of instinct, personal engagement, interests and values, according to KRATHWOHL, BLOOM and MASIA [18]
  . reception
  . reaction
  . assessment
  . hierarchical evaluation
  . classification of the personality by a value or a complex of values

2.2 Teaching strategies and media

Learning goals cannot be established solely in conformance with a psychological taxonomy as the only selection criterion. The primary aspect is that of the subject matter. Learning goals and subject matter together will, in conjunction with suitable teaching methods, achieve the desired change of behavior in the student. The organisational aspects are the responsibility of the teacher.

Curriculum development therefore consists not only of planning a number of learning processes, but must also include the media to be used and social and political goals. These pedagogic aspects will be discussed only to the extent to which they relate to the medium computer.

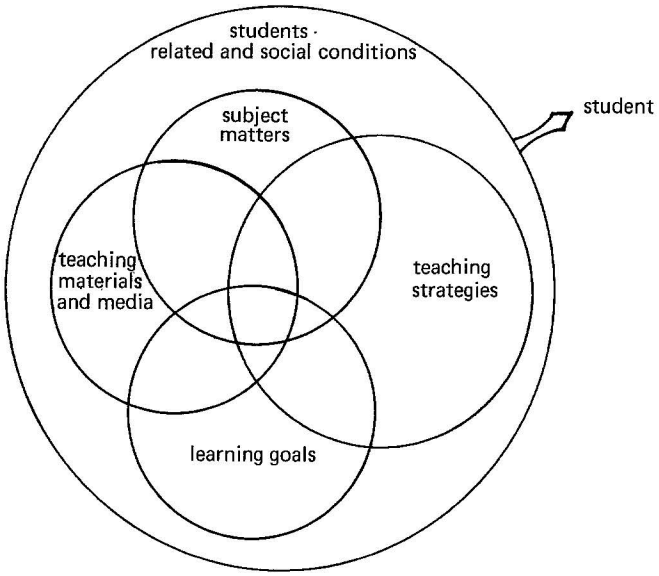


Fig. 4: Factors in the development of curricula

The desired change of behavior may be achieved by different teaching methods:

- working with groups during the lesson
  . small groups
  . large groups
- teaching a single student
- self-study (individual learning).

Learning processes may be computer assisted but should be:

- controlled by a teacher or
- controlled by the student.

Different classification systems may be distinguished, concerning the didactic functions of a learning process, such as:

- motivation
- information
- discussion of problems (social phase)
- training
- application (laboratory)
- self control (tests)

as well as teaching strategies, i.e.:
- drill and practice
- tutorial teaching
- simulation
- gaming
- search for information
- problem solving.

The organisation of lessons is determined by:
- didactic functions
- media used
- type of lessons and
- teaching strategy.

Non-personal media are:
- chalk and blackboard
- books
- projectors (slides, foils, film)
- language laboratory
- television sets
- mechanical models or technical equipment
- computers
- others.

The afore mentioned media allow for an exchange of information or offer the type of information commonly known as "courseware". Courseware includes embossed, imprinted or otherwise fixed information, such as the one on:
- slides
- films
- foils
- video tapes
- tape recordings
- EDP-storage.

Didactics and teaching strategies determine the choice of the medium. For example, television is generally better suited to present the didactic functions of information, provided no interaction between student and medium is expected and information flows in one direction only.

If, however, a dialogue is desired which adapts to the individual needs of the student (within limits), the computer is the only suitable non-personal medium.

2.3 Students, teachers and educational institutions

The purpose of an ES is tightely related to the learning goals and the educational organisation, e.g.
- universities
- graduate schools
- vocational schools
- junior high schools
- elementary schools

as well as the attendees:
- pupil
- adolescent
- adult - rehabilitand
- student
- apprentice.

Because none of the listed institutions or students influence the use of the computer beyond the range described in chapter "Systems approach", the subject is not further pursued.

3. RELATIONS BETWEEN THE ELEMENTS OF AN EDUCATIONAL SYSTEM

3.1 Learning goals - teaching strategies

The association between the cognitive levels of learning goals, according to BLOOM, and general learning goals and teaching strategies of CAI are shown in Figure 5.

The table shows that the medium computer should be used in the ES primarily for the following teaching methods:
- simulation
- gaming
- searching for information and
- problem solving.

strategy / reinforcement increase advancement	drill and practice	tutorial teaching	simulation	gaming	collection of information	problem solving
the power of concentration and recollection	x					
ability to recognize,structure and solve problems			x	x	x	x
problem solving behavior			x	x		x
ability to abstract facts linguistically - formally		x	x	x		x
thinking in models and algorithms			x			x
application and control of principles				x	x	x
discrimination and assessment of facts and making decisions			x		x	x
objective assessment of subjective processes		x	x	x		
social and vocational adaptability training of the intellectual flexibility			x	x		
taxonomy of BLOOM						
knowledge	x	x				
comprehension		x				
application	x		x	x	x	
analysis			x	x	x	x
synthesis			x			x
evaluation						x

Fig. 5: Relation 'general learning goals - teaching strategies'

3.2 Functions and tasks of the computer

The unique capabilities of the computer suggest its use in a variety of ways, such as:

- the computer itself as subject matter (computer science)
- computer as a medium in instruction
  - imparting professional or general learning goals
  - carrying-out and evaluating tests
  - (computer assisted instruction = CAI)
- computer as an aid for teachers, administration and research
  - (e.g. computer managed instruction = CMI).

When so used, the computer executes the functions named in Figure 6.

use of the computer for \ functions of the computer as	adaptive teaching machine	calculating machine and programming device	subject matter	information system	instrument for the planning and organisation of lessons	means of research
training of computer scientists			x			
CAI/CMI	x	x		x	x	
research						x
educational administration				x	x	
methodology	x	x		x		x

Fig. 6: Use of the computer in ES

We distinguish the following fields of activity:

- computer science
  - . programming languages
  - . data structures/data banks
  - . hardware
  - . operating systems
  - . architecture of hard- and software
  - . formal methods

- CAI/CMI
  - . imparting knowledge
  - . training of skills
  - . examination
  - . training the problem solving behavior
  - . search for information
  - . self-teaching
  - . computer managed instruction

- educational administration
  - . establishing:
    schedules for lessons and utilization of class rooms
    statistics and documentation
    certificates, school-reports
    forecasts
  - . carrying-out calculations with regard to planning and decisions
  - . enrolment and disenrolment of the participants of the ES
  - . library management
  - . accounting etc.

- research
  - . pedagogical research
    simulation of learning processes
    development of algorithms for CAI
    evaluation of courseware etc.

- teaching methodology
  - . working techniques
  - . planning and organisation techniques
  - . learning techniques

The above mentioned tasks show that a CAI system may be structured hierarchically, according to these tasks, see Figure 7.

When using the computer in the field of education, it is essential to have the same operating systems and software packages available, as those commonly used in industry administration and science. Obviously all programming languages which are being taught as subject matter are required in addition.

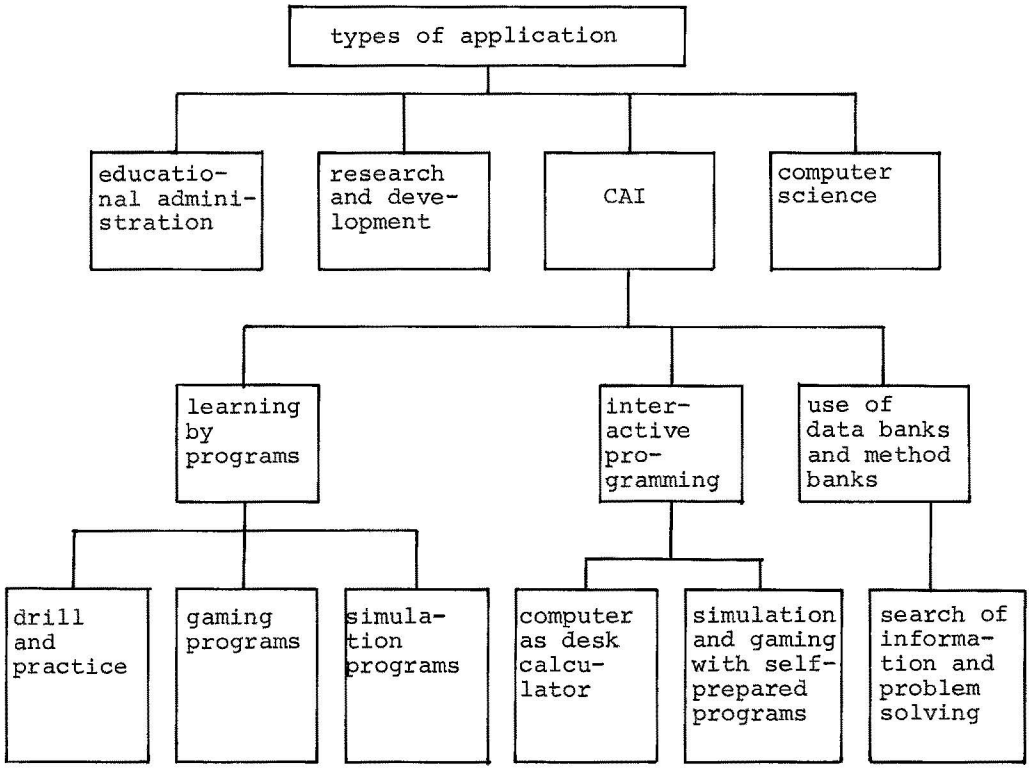


Fig. 7: Ways of applying computers in education

3.3 Teaching strategies and software instruments - types of application and language levels

In realizing the teaching strategies of CAI we distinguish four application methods, see Figure 8.

Simulation and gaming via dialogue have been added to the original methods (drill/practice, tutorial teaching) only in recent years.

In vocational training the newly developed strategies are characterized by [9].

- realistic training situation
- manipulation of time in slow motion/fast motion which allows for detailed analysis and/or exercise
- users selection of controlling factors which facilitates understanding

- the fact that no raw material and no special machinery is required for simulation
- saving and analyzing the input data and resulting processes enable the individualisation of training
- these methods enable the individual to discover contexts by himself.

application methods \ teaching strategies	drill and practice	tutorial teaching	simulation and gaming	search for information	problem solving
interactive programming					x
use of program packages			x		
learning and practising in dialogue by programs	x	x	x		
use of data bank systems and method banks	x			x	x

Fig. 8: Ways of applying CAI - teaching strategies

The instruments available to the user are classified in Figure 9.

application methods \ software instruments	dialogue language	program libraries	programs or functions in a support-language and subsystems	data manipulation and query languages (DML)
interactive programming	x			(x)
use of program packages	(x)	x	x	
learning and practising in dialogue by programs	(x)	(x)	x	
use of data bank systems and method banks		x	x	x

Fig. 9: Relationship between application methods and software instruments

The software instruments are applied to CAI via command languages at various levels:

Language level 1: System commands on the operating system level to log on, log off and to call-in host-languages and programs.

Language level 2: Commands on the dialogue supportlanguage level to call-in programs or functions.

Language level 3.1: Program statements in the supportlanguage.

Language level 3.2: Learner's dialogue embedded in the function flow of subsystems.

Language level 3.3: Commands in the DML or query language of a subsystem which can be activated by a command in the learner's dialogue (DML = Data Manipulation Language).

Some language levels may be skipped automatically by system service routines.

The implementation of the most important CAI methods should include at least the user applications shown in Figure 8.

If so, the learner would then employ the software instruments shown in Figure 9 by using the different command languages.

A variety of teaching strategies requires different software instruments. The learner must know the applicable commands to make use of these instruments, see Figure 10.

software instruments / application methods	dialogue language	program libraries	programs or functions in a support-language and sub-systems	data manipulation language and query language
interactive programming	2,3			
use of program packages		1	2	
learning and practising in dialogue by programs			3.2	
use of data bank systems and method banks		2	3.2	3.3

Fig. 10: Application methods versus software instruments and associated language levels

4. HARDWARE AND SOFTWARE REQUIREMENTS

Employing CAI in a flexible way so as to satisfy the needs of the learners requires language levels with various degrees of complexity. In addition a suitable input/output device, henceforth termed the "communication area", is required.

Presently dialogues are conducted via typewriter terminals and data displays with means to connect voice or image equipment in the future.

Future communication areas should possess:
- a defined but alterable character set,
- colour,
- brightness,
- flashing,
- touch-sensitivity,
- lock-out.

The teachers demand that the dialogue language and the operating system meet the following requirements:
- easy to learn and not ambiguous,
- suited for several levels of skill,
- separate definition and execution modus,
- transparency between language levels and easy switching from one to another,
- programming language and problem definition language should be identical,
- simple means to realize different teaching methods,
- simple means of error correction,
- must allow development of data banks,
- must include access methods for data banks,
- open-ended language design,
- provide means for generative program development,
- fast response time,
- high availability/reliability,
- means for communication among the users via the system.

The educational institutions would expect the following qualities:
- favorable cost/profit relation,
- system transferable to new-generation hardware and software,

- open-ended for capacity increases,
- distributed intelligence,
- simultaneous use of different language systems,
- step-by-step expansion,
- decentralized usage.

The computer scientists expect to find of the commonly required technical properties, such as: modularity, well defined interfaces, a small number of independent basic structures in the language (orthogonal design), field-reconditioning or up-grading possibilities.

It is obvious that none of the classical author languages meet the afore mentioned requirements, except in cases of extremely limited or specialized user demands. For the latter, smaller systems may be sufficient.

BRUNNSTEIN [6] published a survey of small computer systems for the use in schools with details about:
- programming languages,
- operating systems,
- file systems and program libraries,
- basic configurations and their cost
- possibilities for extension
- operating requirements

An assessment of different author languages and their application range is to be found at KAIER [15]. He employs the classification of languages for CAI by ZINN [26] which contains:

(1) description of successive frames or items,
(2) provision for conversation within a limited context,
(3) description of a procedure by which material (a curriculum film) is presented,
(4) specification of an environment for programming and problem solving.

His evaluation results in:

Type of language by ZINN	Decisions of the author		
	contents	algorithm	commands of the learner
(1) and (2) author languages	limited	restricted	no
(3) didactical pattern	limited	no	no
(4) general dialogue languages	yes	yes	yes

The application range of some CAI-languages is described by KAIER [17], see Figure 11.

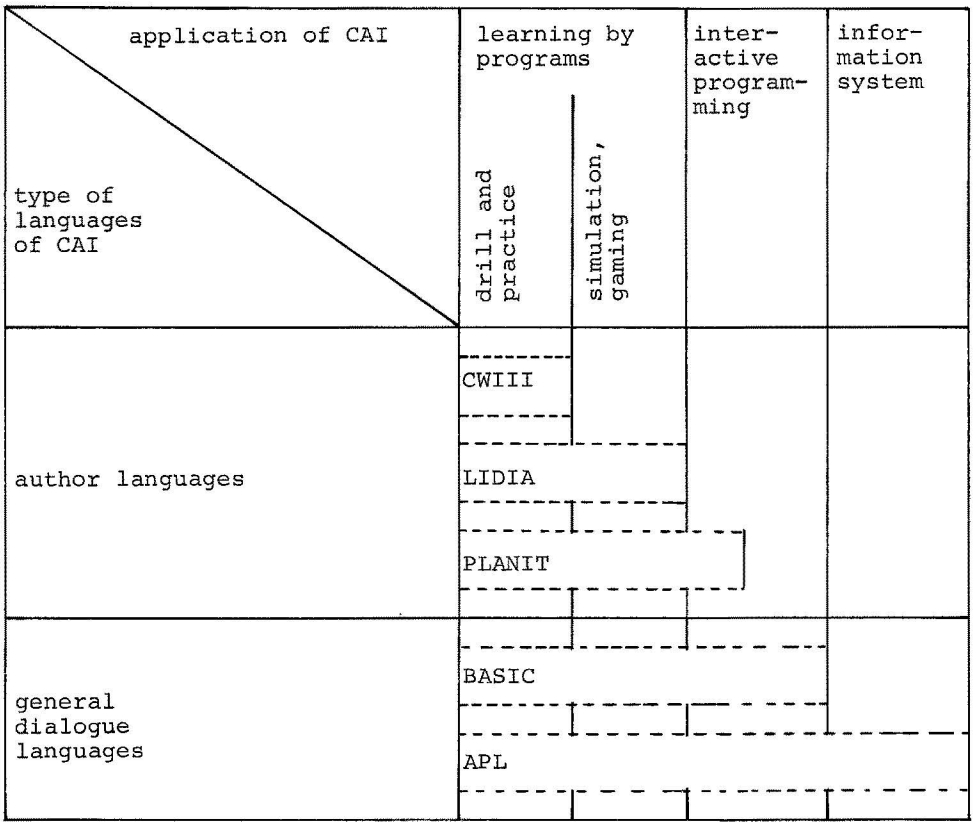


Fig. 11: Extention of the application of CAI-languages

5. SYSTEMS APPROACH IN HEIDELBERG

5.1 Environmental Aspects

The foundation "Stiftung Rehabilitation" has been engaged in the introduction of CAI for vocational training purposes since 1968.

As of 1970 CAI has become the daily routine at the rehabilitation center. The ever increasing experience has meanwhile led to a continuous improvement of CAI at the Heidelberg institution. As improvements are still being made, any system description can only give a momentary picture of the present situation. We believe, however, that the processes of continuous adaption to new didactic as well as technical requirements are essential for any efficient CAI system. Many attempts by others have failed in the past because they stuck to rigid conceptions and underestimated detail problems.

Factors which are often underestimated are:
- the reliability of hardware and software,
- the convenience in handling the CAI system,
- adequate complexity of courseware,
- definition and checking of cognitive input requirements.

Instead of emphasizing the above listed factors, quite often attention was given only to didactic instruments and methods.

In Heidelberg the integration and intensification of CAI is considered most important, that means:
- information,
- counseling,
- training,
- service and maintenance,
- supply of organizational, technical auxiliary materials and accompanying lessens.

To reach these goals a section 'media technology' has been established as a service group which supervises the continuous integration process.

The necessity for adaptation does not only result from experience gained with the CAI system itself, but is also caused by:
- the change of psychological principles in education and curriculum development,
- fast technological development,
- the change of educational objectives, and
- adjusting the priorities of public goals to the general economic situation.

When selecting a hardware/software conception, considerable attention must be given to an open design which allows for the introduction of new teaching methods as well as new hardware and software. Technical improvements in a CAI system must be installed without disturbing the user. Typical examples for such changes are the switching of SHARP-APL [22] users from OS/MVT to OS/VS2, or the connection of new terminals to the remote data processing network. The stability of the running system must not be impaired by such improvements.

5.2 Subjects Taught

5.2.1 Learning goals and subject matter

The Heidelberg vocational education center offers the broad spectrum of courses laid down in the long range plans for adult vocational training with emphasis on future needs.

As we cannot list them all, we shall list only those subjects, where CAI is used daily:

- accounting, auditing,
- business calculation,
- applied mathematics,
- general economics,
- book-keeping,
- mathematics for electronic engineers,
- mechanical engineering,
- mathematics for skilled labourers,
- physics for technicians and skilled labourers,
- NC-programming,
- basic electrical engineering,
- reading of blue prints,
- principles of measuring,
- training for medical technical assistants (MTA).

For CAI those subjects are preferred which may be easily structured and at the same time give consideration to general learning goals.

Learning goals are given on all levels of the taxonomy of BLOOM.

5.2.2 Teaching strategies and media

A big part of the development of teachware hitherto concentrated on the development of adaptive training programs with tutorial components which are designed to eliminate specific gaps in knowledge.

Besides the mere development of teachware, software instruments were developed which made it possible to efficiently develop programs and to assess their effectivity.

At present we intend to employ the newer teaching methods
- simulation,
- gaming,
- problem solving.

We further intend to place more emphasis on the tuning of technical systems.

The medium computer is used in Heidelberg as:
- subject matter in computer science itself,
- aid for administration, teachers and research,
- instrument for instruction (CAI)
  . in multi media systems
  . in traditional group teaching
  . for individualized learning.

In the multi media approach media are dispatched to learning phases according to didactical functions.

CAI is included organizationally and didactically into the process of training.

The following systems are used for CAI:

IBM/370-155 II
IBM/370-145 (to be replaced by model 148 in 1977)
SIEMENS/4004-151.

A total of more than 300 terminals, some of them via remote connection, are available daily from 7.00 a.m. until 9.00 p.m. and 8.00 a.m. to 12.00 a.m. on Saturdays.

5.2.3 Students, teachers and educational institutions

The Heidelberg CAI system is used by several vocational rehabilitation centers (Heidelberg, Frankfort, Linz, Dortmund, Munich), as well as the "Südwestdeutsches Rehabilitationszentrum" for children and adolescents in Neckargemünd and for vocational therapy in the rehabilitation hospital Karlsbad-Langensteinbach. The age structure ranges from pupils to adults. The use of CAI by instructors does not require special knowledge in data processing apart from the handling of terminals and the use of the necessary command language.

5.3 Realisation

5.3.1 General aspects

The following statements apply only to CAI and the description of software.

The course of decisions leading to the realisation of the systems running today cannot be traced back as it was a continuous process over some years. This process was certainly influenced by exchanging know-how with international experts of CAI. The steps taken to realize the CAI system in Heidelberg differed from those taken by other CAI teams in Germany, because the latter developed ideas different from ours, especially with regard to the importance attached to the language systems used.

The success of the Heidelberg CAI system is in my opinion based on several facts:

- the selection of an expedient dialogue system and the avoidance of author languages,
- pragmatic approach in the initial phase without letting ourselves be impeded by desirable but not realizable didactic prerequisites,
- gradual adaptation of the running system to didact needs and development of specific curricula,
- development of suitable subsystems in the dialogue supportlanguage and turning towards new teaching strategies,
- extention of the repertoire,
- continuous effort in development and integration of CAI and the set-up of the technical section 'media technology',

- feedback regarding the quality of the developed courseware and teaching strategies through the daily use at the vocational rehabilitation center Heidelberg,
- involving the instructors in all phases of planning, development and integration of CAI,
- selection of suitable learning goals and strategies with gradually increasing degree of difficulty,
- shifting the development from the research team to a large group of teachers,
- giving highes priority to the availability and reliability of functions in the CAI system currently used,
- the vital interest of the educational institution in CAI.

5.3.2 Hostlanguage APL

APL in Heidelberg is used as general purpose language for CAI. Besides APL the author languages CWIII and PLANIT have been tested.

APL was used as follows:

1970:	APL / 360	on IBM/360-50
1973:	APL * PLUS	on IBM/370-155
Since 1974:	APL \ 4004	on SIEMENS/ 4004-151
Since 1975:	SHARP-APL	on IBM/370-155.

The reasons for using APL as CAI language were manifold, and I shall not repeat them here. See [2,21].

In Heidelberg APL is used for:
- drill and practice,
- simulation,
- gaming,
- searching information and use of method banks,
- problem solving.

In addition, different subsystems have been developed, which are used by the learner through appropriate command languages. We distinguish three types of application [15], see Figure 12.

CAI application	activity of user
interactive programming	solving of problems by self-prepared programs
learning by programs	working on given programs
use of data bank systems and method banks	formulation of inquiries to a central source of information

Fig. 12: CAI application - activity of user

Consequently different language levels and types of dialogue are used, see [15] and Figure 13.

CAI application	language level	type of dialogue
interactive programming	APL as general dialogue language	user-controlled
learning by programs	commands within the range of the languages of the subsystems	program-controlled (adaptive to user)
use of data bank systems and method banks	commands of a query language	user- or program controlled

Fig. 13: CAI applications - language levels - types of dialogue

5.3.3 Subsystems MOPS and TIPS

In Heidelberg several existing subsystems were either adopted without change, enhanced, or newly developed. Figure 14 gives a survey of the most important subsystems.

Subsystem	Remarks
MOPS	modular package system; contains functions to support the generation and maintenance of CAI programs
TIPS	test item package system; set-up and administration of test item banks (on-line), usable on- or off-line
NOTENBANK	system to store student grades in a data bank
EXIS	experimenting INF-DOC-SYSTEM; set-up and use of computer administrated "encyclopedic books", allows recording of hierarchical and general structures of data

Subsystem	Remarks
AUTOGUT	programs for the computer assisted set-up of psychological reports
EDIT	editing texts in APL, wording, changing formating and printing of texts; overlaying of text portions possible [8]
TCS	terminal communication system

Fig. 14: Subsystems

The subsystems were written in the APL philosophy so that they may be combined as function packages by using the appropriate interface definitions.

5.3.3.1 MOPS

The development of MOPS has been sponsored by the federal minister of research and technology (Project DV-BV 320, 1974-75) and is described comprehensively in [21].

MOPS, Modular Package System, was developed to set up and maintain programs for the CAI type of application 'learning by programs', see Figures 7 and 8.

Figure 15 shows the relation of MOPS to Figure 8 'ways of application - teaching strategies'.

application methods	drill and practice	tutorial teaching	simulation and gaming	search for information	problem solving
interactive programming					
use of program packages					
learning and practising in dialogue by programs	MOPS TIPS	(MOPS)	APL MOPS	(MOPS) (TIPS)	
use of data banks and method banks				EXIS	

Fig. 15: Use of MOPS

APL and MOPS avoid the disadvantages of author languages which are manifest in the fact that:

- the author (teacher or instructor) inconsciously adapts to the prescribed language structures without giving much thought to didactic principles or better suited structures,
- teaching and learning strategies, such as simulation, problem solving, writing programs, etc., could either not be realized at all or only with great effort or very little effect,
- the student had no access to the computer for purposes other than those permitted by the author language (e.g. impossible to use ASSEMBLER).

With MOPS and APL a dialogue system is available which may be used efficiently for all teaching strategies. MOPS was set up as a teaching system of the class III type according to KLOTZ [18], see Figure 16.

An instruction system of the class III type is a technical means which:

- stores a prepared program,
- generates a tutorial program from the prepared program according to conditions found during the running of an interactive process,
- presents (puts out) the program in a way that can be described exactly,
- requests an answer from the learner (student) as reaction to the didactical unit presented,
- registers learners reactions (within a given choice of answers),
- checks the answers,
- confirms or rejects the answers,
- speeds up the program depending on learners progress,
- starts and finishes the presentation of the program on the individual learner's request.

Fig. 16: Teaching system of a calss III type, according to KLOTZ

MOPS is characterized by:

- modular concept (kit),
- easy revision,
- flexibility (adapts to needs of student and teacher).

- flexibility (adapts to teachers and students needs),
- allowing extension (inclusion of new hard- and software features),
- economical standard routines (which can be embedded anywhere because of time and space saving optimal structure),
- easy handling of CAI programs,
- versatile possibilities of communication (variable sizes, approximation to natural dialogues),
- standardized formats for users (prescriptions for the definition of names of functions, variables and learner commands),
- adaptable to various timing, routing and teachers requirements,
- reliability of soft- and teachware,
- collection and saving of learners data for
  - control of learning process
  - statistics
  - research,
- allowing individualized work on programs,
- random data generation (with automatic cancellation in case of recurrence,
- fetching data sets from the information documentation system via dialogue,
- transition to programming status while working on a problem solution within a lesson,
- free choice of teaching methods,
- re-entry ports.

The sequence in the above, certainly not complete, list is arbitrary.

MOPS ensures that the author is not unconsciously fixed on a pre-defined didactic structure such as the one shown in Figure 17.

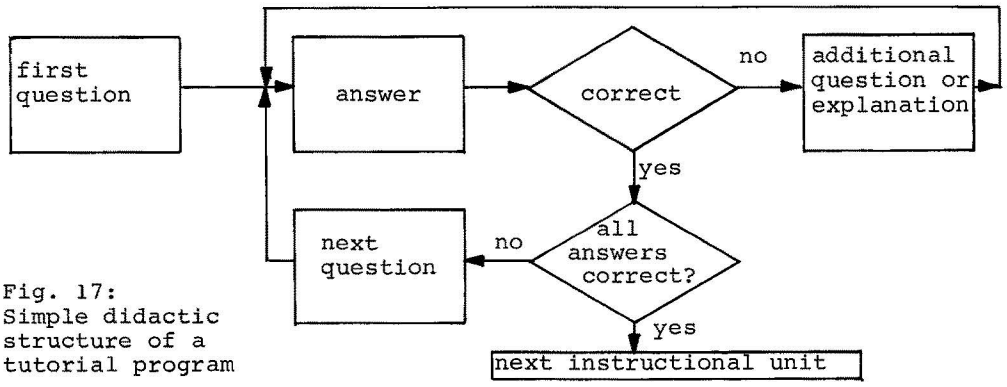


Fig. 17:
Simple didactic structure of a tutorial program

Independent of teaching strategies the learner may enter commands in language level 3.2, see Figure 10, at any program step and have them executed. In this way he may control the execution sequence. With MOPS the learner has control commands and auxiliary commands available to change the course of the dialogue.

Control Commands	Effect
ENDE	completes program execution
ERKL	an explanation is requested
STOP	the program is interrupted
WAHL	the user is allowed to select program sequences
Auxiliary Commands	
INFO	access is given to data bank systems
KRIT	criticism is stored
PAUSE	timer is stopped
PROT	protocol of session is issued
RECH	system is put in desk calculator mode
TEXT	list of latest data or problems is issued

MOPS distinguishes various sections which consist of a package of APL functions.

In the tutorial program generation section, we distinguish three types of functions:

Types of Functions	Meaning
closed functions	cannot be changed by the author
open functions	may be changed by the author
empty functions	must be filled by the author with data (for instance text)

MOPS furthermore offers service functions for testing and automatic program documentation. MOPS also has proved useful in the revision of previous programs. With the aid of MOPS it was possible to audit all drill and practice programs and to raise them to a homogeneous maximum quality standard. Presently about 500 different tutorial programs are available, all controlled by the learner through commands. The programs are used by the training groups shown in Fig. 18.

5.3.3.2 TIPS

TIPS = Test-Item Package System is an auxiliary instrument for building and using data banks by

- instructors who:
  . set-up task lists and detail data for later input by data typists
  . prepare tests and evaluate statistics,
- learners who:
  . prepare for written examination
  . test their own knowledge
  . repeat lessons.

Lists may be printed on terminals and/or high-speed printers. The learner works on problems in dialogue fashion or off-line, if necessary.

5.3.4 Statistics

During running operations, statistical data are recorded which give information (see Figure 19) about:

- the number of users of the CAI system,
- the degree of loading of the system IBM/370-155,
- the terminal usage (Figure 20),
- CPU and connection time per user.

The system WWW [24] automatically records who fetched which program at what time from public libraries. Private libraries are excluded.

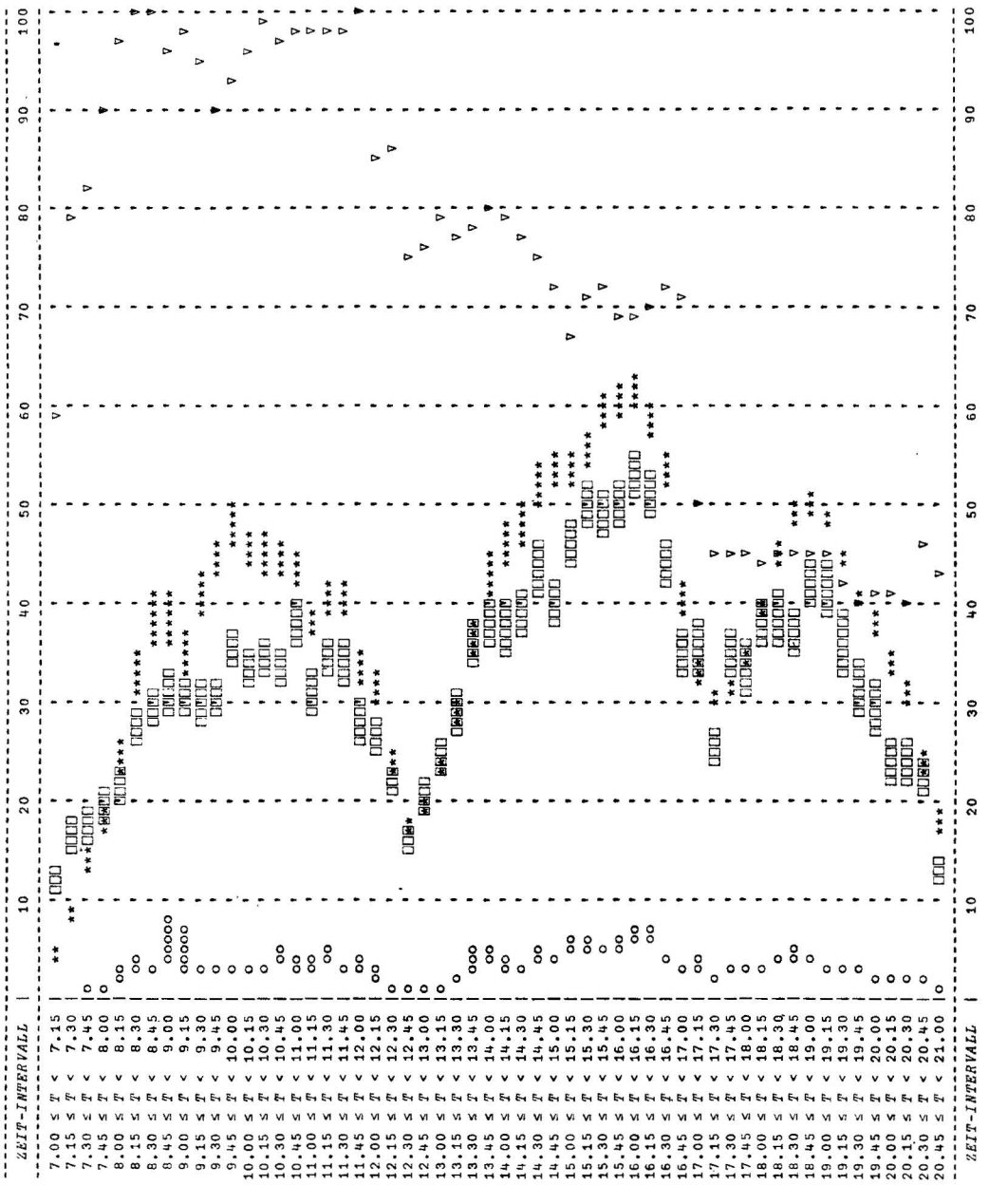
With the assistance of WWW monthly statistics are set up which inform about the use of programs and their users. Figure 21 shows such a WWW report.

As the system has been in use since 1974, the first statistic data on the behavior of participants could be issued as soon as 1975 [23]:

- mean waiting period for APL users: 1.1 sec
- number of monitored workspaces of public libraries: 298
- number of participating persons: 3257
- number of WWW requests: 78743

SECTION	Training Group	Duration of training (months)	Usage of terminals (hrs) total	relat.	absol.	down
commercial professions	industrial purchaser	10	23.00			0.17
	"	7	10.09			0.33
	office clerk	15	12.00			2.03
	"	7	9.10			
	"	7	9.90			
	clerical employee	15	8.23			
	"	4	5.50			1.00
	"	1	3.00			
	bank employee	15	8.80			
	"	4	13.10	102.81	12.7	
EDP-professions	data typist	3	13.50			1.50
	"	9	53.50	67.00	8.3	4.90
machinists, tool and die makers, technical professions	precision tool maker	9	42.45			3.67
	"	4	30.62			0.27
	"	1	5.95			
	quality inspector	17	37.35			3.87
	"	4	10.00			
	"	9	45.45			3.58
	draftsman	13	15.00			3.42
	design engineer	9	8.00			
	design technician	13	48.58			0.90
	"	1	9.60			1.20
	quality inspector	13	99.15			5.88
	"	1	11.55			2.00
	NC technician	16	63.15	426.85	52.6	6.50
electro-technical professions	telephone technician	13	10.87			1.10
	"	1	18.67			1.17
	electronic technician	17	11.00			
	"	11	17.83			
	"	13	22.17			1.27
	"	9	20.20			3.10
	"	1	2.50			
	electrical technician	13	6.50			
		1	6.50	116.24	14.3	
graduated school	management consultant	26	59.55			0.50
	"	8	4.50			
	electrical engineer	26	14.34			
	mechanical engineer	26	9.90			
	"	8	4.25	92.54	11.4	
professions in the field of construction and civil engineering	draftsman in the field of construction	3	5.75	5.75	0.7	

Fig. 18: Terminal usage from 7.30 a.m. to 3.00 p.m. by training groups



SYSTEM-MITTELWERTE IM ZEITRAUM VOM 1.5.1976 BIS 31.5.1976

ooo----→ WARTEZEIT [100 PROZENT = 10 SEC]
***----→ TERMINALZAHL [STUECK]
□□□----→ MASCHINENAUSLASTUNG [PROZENT]
∇∇∇----→ VERFUEGBARKEIT [PROZENT]

Fig. 19 Statistical Data of the CAI-System

I. Times

month	days of education	hours of group session	hours of voluntary session	total time
5	22	135.0	88.0	223.0

II. Trouble Report

	Wieblingen		Pfaffengrund	
	month	to date	month	to date
number of failures	2.00	48.00	1.00	27.00
number of troubled days	2.00	34.00	1.00	20.00
failure time (hours	0.48	34.00	0.17	28.49
relative failure (with regard total time)	0.22	3.24	0.06	2.70

III. Usage Report

1. Terminal assignment for group sessions

	Wieblingen		Pfaffengrund	
	month	to date	month	to date
possible time (hours)	135.0	630.00	135.00	630.00
actually used (hours)	40.55	178.04	53.17	151.97
relative	30.04	28.26	39.39	24.12

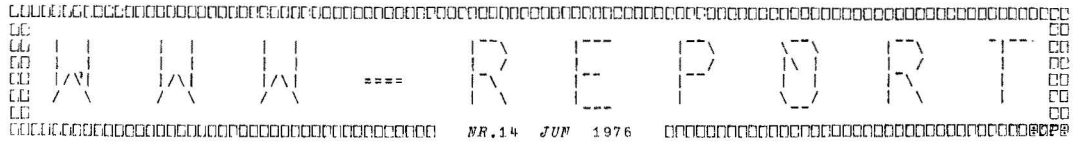
2. Average usage of terminals during group sessions (lines)

	number of terminals	average per month	absolute to date	average per month	relative to date
Wieblingen	34	11.34	9.68	33.36	28.47
Pfaffengrund	30	6.45	5.61	21.51	18.69

3. Average usage of terminals during voluntary sessions

	number of terminals	average per month	absolute to date	average per month	relative to date
Wieblingen	34	11.97	9.38	35.22	27.58
Pfaffengrund	30	7.19	5.94	23.97	19.81

Fig. 20: Terminal usage report, May 1976 Rehabilitation Center Heidelberg



LAUFZEIT DES APL-SYSTEMS: 234 STUNDEN (AN 24 LAUF-TAGEN)
ZAHL DER APL-SYTEMSTARTS: 29 (1.2 STARTS / TAG)

BIBLIOTHEK NR.	1	555	700	701	702	703	799	800	899	900	901	902	999	*SUM PROZ*		(*)	*HAEUFIGSTES PROG.*
BAD VILBEL		4	151		2	12		69	623	1	162			1024	12	155	899 *GE160*
ESSEN	32	270	13				154	96	213		509		1	1288	16	38	555 *LUNA*
HAMBURG	9	133	226	1	2	1	131	7	3	1	339	11	2	866	10	179	901 *RECH32*
LANGENSTEINBACH	2	12	17			2	32	3	21		44	9	3	145	2	14	901 *RECH4*
LINZ		68	30					18		2	27			145	2	16	555 *WORTSPIEL*
NECKARGEMUEND		85	75		3			26	3		122			314	4	54	901 *RECH36*
PFAFFENGRUND	26	217	155			2	8	9	18	4	613	177	2	1231	15	164	901 *RECH50*
WIEBLINGEN	488	359	480	16	55	243	249	229	371	208	407	25	146	3276	40	388	1 *NOTENBANK*
SUMMEN	557	1148	1147	17	62	260	574	457	1252	216	2223	222	154	8289	100	1008	
PROZENT	7	14	14		1	3	7	6	15	3	27	3	2	100			
(*)	134	219	239	10	36	72	161	174	206	46	463	54	30				

(*) *ANZAHL DER TEILNEHMER-NUMMERN, VON DENEN DIE ENTSPRECHENDEN WWW-AUFRUFE GENERIERT WURDEN.*

```
  1 (  5 WS ): NOTENBANK, WSDOC, ZNREF, EDIT
555 ( 19 WS ): SPIELE
700 ( 49 WS ): WIRTSCHAFTSARITHMETIK
701 ( 22 WS ): ANGEWANDTE WIRTSCHAFTSARITHMETIK
702 ( 22 WS ): VOLKSWIRTSCHAFTSLEHRE
703 ( 82 WS ): RECHNUNGSWESEN
799 (  8 WS ): UEBERGANGSBIBLIOTHEK ZU 700
800 ( 55 WS ): ELEKTROTECHNIK
899 ( 36 WS ): UEBERGANGSBIBLIOTHEK ZU 800
900 ( 29 WS ): MASCHINENBAU (FACHHOCHSCHULE)
901 ( 54 WS ): MASCHINENBAU (FACHARBEITER)
902 ( 57 WS ): MATHEMATIK FUER NC-TECHNIKER
999 ( 47 WS ): UEBERGANGSBIBLIOTHEK ZU 900
    (485 WS ): ANZAHL ALLER UEBERWACHTER WS
```

F A V O R I T E N D E S M O N A T S

```
 1. (295 AUFRUFE):   555 LUNA
 2. (282 AUFRUFE):     1 NOTENBANK
 3. (197 AUFRUFE):   700 PRAXIS3
 4. (194 AUFRUFE):   700 PRAXIS1
 5. (175 AUFRUFE):   700 PRAXIS2
 6. (162 AUFRUFE):   555 WORTSPIEL
 7. (160 AUFRUFE):   901 RECH1
 8. (150 AUFRUFE):     1 WSDOC
 9. (141 AUFRUFE):   999 PLO
10. (140 AUFRUFE):   700 PRAXIS2740
11. (136 AUFRUFE):   555 PFERDE
12. (128 AUFRUFE):   899 GE100
13. (123 AUFRUFE):   899 GE140
14. (120 AUFRUFE):   900 APLTEST
15. (115 AUFRUFE):   899 GE160
16. (107 AUFRUFE):   901 RECH32
```

FAVORITEN-SUMME = 2625 = 31.7 PROZENT DER AUFRUFE

Fig. 21 WWW-report

Figure 22 shows the allocation of workspace to user groups.

Number	%	Registered Users	Workspace used total	%	Workspace used by each user
101	3.1	authors, operators, assistants, producing instructors	2439 (*)	42.3	24.1
312	9.4	instructors	709	12.3	2.3
2212	66.9	students	2089	36.2	0.9
679	20.6	others	530	9.2	0.8
3304	100.0		5767	100.0	
(*) 1841 private workspace 598 public workspace					

5.3.5 Primitive data bank systems in APL

Primitive data banks are in our opinion data banks which display the following characteristics:

- direct addressing of data blocks via tables,
- no explicite external references, that is, only such external references which are part of the data anyway and are defined by the users interpretation,
- not suitable for repetitive processing of large volumes of data because the individual accesses cause excessive loading of the system,
- little program independence.

The primitive data banks of the Heidelberg CAI system contain:

- documentation of the function package MOPS, DMOPS (see 5.3.3.1),
- terminal communication system, TCS,
  - for message exchange between the users of APL and authors/instructors,
  - messages are stored for 30 days,
- telephone register,
- test item bank system, TIPS (see 5.3.3.2)
- externally stored data, such as
  - explanations
  - functions
  - graphics
  - tables,

- Autogut [4]
  assists in the composition of psychological reports by offering text portions with deliberate gaps, thus saving time,
- Notenbanksystem [23] (grading bank)
  stores students grades and comments; contains appropriate access methods for statistical evaluation.

The Notenbanksystem contains the annual report cards - about 90000 in total. This system meets the following requirements:

- Each instructor has a private 'grades book' at his free disposition, he may enter grades, ratings, and comments for each test and change them. Further, he may call off a list containing all single grades as well as the most important statistical values. Finally he may get a graphic survey of the distribution of grades.
- Higher management (if entitled) may at any time obtain information about the achievments of groups or single participants. The information is, however, limited to average values, single grades or instructors comments remain in privacy.
- The grades gathered are stored in a space saving way and kept ready for later statistical evaluations or the issue of report cards.
- The Notenbanksystem is easy to handle and requires no knowledge of EDP. The instructors who are familiar with APL are however allowed to manipulate their own grades freely, without being able to endanger other data or break any security barriers.

The Notenbanksystem's place in the training center is shown by the following statistical data:

- number of instructors,	203
of these privileged	99
- number of training groups	96
- number of students	2619
- number of subjects	820
- number of examinations	2693
- number of grades given	47366
- number of pseudo-grades	18075
- space required for the data (bytes)	542232

The Heidelberg data bank system EXIS [25] (a function package in APL) does not belong to the primitive data bank systems.

EXIS is very comfortable because it offers direct access to data files via descriptors, key words or synonyms. General data structures may be generated. In EXIS the user has various service functions at his disposition. The system is extremely easy to use. In Heidelberg EXIS is actually used as an instrument for developing data banks.

For special tasks, the Heidelberg CAI system has further subsystems, see Figure 23, which contain:
- functions for the editing of answers,
- functions for the analysis of free answers,
- functions for the desk calculator mode.

Many programs for the simulation of business cycles, currency and monetary cycles are available in extra workspace [16].

6. OUTLOOK

To increase the efficiency of CAI systems the following objectives must be met:

- We need computer assisted systems which automatically generate programs for special purposes. The idea is to avoid the dual effort of writing lessons then programming them. Actually the author should write course material directly at a terminal and thereby create the appropriate program.

- Hardware and software should not only be dedicated to CAI but also be utilized for other purposes so that wider usage lowers the cost.

- New strategies which encourage the student to use the computer on his own instead of being guided step-by-step.

- Stick to the principle of modularity by developing program packages which are dedicated to specific tasks.

- Continuous efforts to stabilize and tune the system.

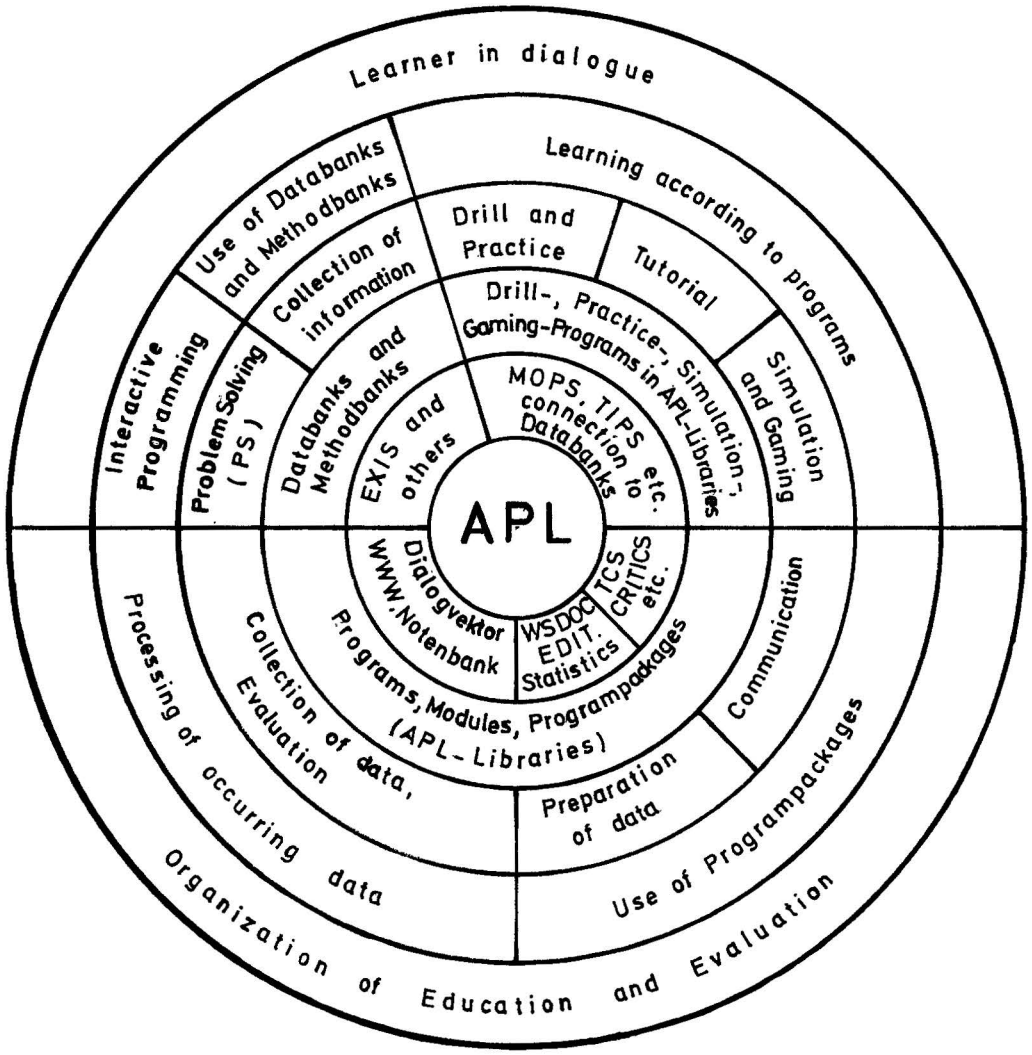


Fig. 23 CAI-System realized in APL

- Develop concepts for a partial decentralisation of processes so that the switching from one to another is not noticed by the user. Access to data banks and data bank maintenance must nevertheless remain a centralized function.

- Continuous improvements in CAI software through participation of all institutions involved.

- Ensure portability of the newly developed software.

The economic recession in recent years has slowed down CAI expansion. It is, however, safe to say that this is a passing symptom evidenced by the following expected trend:

- CAI will incur ever decreasing costs in years to come because new teachware can use already existing modules and the need for new development itself will decrease.

- In the near future CAI will change more likely due to technological progress rather than due to new education objectives.

- The computer will continue to take on new tasks of information processing and in the long run will become the daily tool of man in his life-long effort of learning and education. The daily use of the computer as a service tool, as communication system, as information processing machine, and as aid for decision making will make it necessary to teach computer usage in schools in the future.

In the field of vocational rehabilitation CAI is a daily used instrument already today.

Literature

[1] Augsburger, W. and Lebong, K.-H., "APL - Autorensprache der Zukunft". In: APL - Zwischenbilanz der Praxis, IBM - Beiträge zur Datenverarbeitung - Anwendungen, IBM - Deutschland, IBM-Form F12-2504.1.74 (1974)

[2] Augsburger, W. and Berger, H., "APL und CUU". Stiftung Rehabilitation Heidelberg, 1974

[3] Augsburger, W., "Thesen zur computer-unterstützten Ausbildung in der beruflichen Erwachsenenbildung". In: Arlt, W., "DV im Bildungswesen". Berlin, 1976

[4] Augsburger, W., "Rehabilitation und Informatik". In: Rehabilitation und Prävention, Volume 2, Stiftung Rehabilitation Heidelberg, Springer, Heidelberg, publication in autumn 1976

[5] Bloom, B.S., "Taxonomy of Educational Objectives", Handbook I, Cognitive Domain, New York, 1956

[6] Brunnstein, B., "Rechner für den Schuleinsatz - ein Schulrechnerkatalog". In: FEoLL, Paderborn, 1975

[7] Fendt, W.E., "Bausteine in APL". Stiftung Rehabilitation Heidelberg, 1974

[8] Fendt, W.E., "EDIT Textverarbeitung in APL". Stiftung Rehabilitation Heidelberg, 1976

[9] FEoLL, "Zur Analyse und Bewertung heute verfügbarer Autoren- und Dialogsprachen". FEoLL, Paderborn, 1974

[10] Freibichler, H., "Chancen des computer-unterstützten Unterrichts in der betrieblichen Aus- und Weiterbildung". In: Lindner, H.(ed.), Lehrsysteme 73 Paderborn, GPI, Berlin, 1973

[11] Gagnê, R.M., "The Analysis of Instructional Objectives of the Design of Instruction". In: Glaser, R. (ed.), Teaching Machines and Programmed Learning, II, Data and Directions, Washington, 1965

[12] Gagné, R.M., "Die Bedingungen des menschlichen Lernens". Beiträge zu einer neuen Didaktik. Hannover, 1969

[13] Heyderhoff, P., "Konzepte für zukünftige Kommunikationsgeräte". GMD, Bonn-Birlinghoven, 1973

[14] Jochum, I., "Benutzung der Datenbanken im Fachbereich Psychologie des RZN". Stiftung Rehabilitation Heidelberg, 1975

[15] Kaier, E., "Zur Organisation des computer-unterstützten Unterrichts". Stiftung Rehabilitation Heidelberg, 1974

[16] Kaier, E., "Experimentelle Behandlung makroökonomischer Kreislaufmodelle im computer-unterstützten Unterricht (CUU)". Dissertation, Universität Karlsruhe, 1975

[17] Kaier, E., "Was ist CUU?". Elektronische Rechenanlagen 18, 1976

[18] Klotz, G.R., "Beiträge zu einer analytischen Didaktik". Mainz/ Nieder-Olm, ZeF, 1972

[19] Krathwohl, D.R., Bloom, B.S. and Masia, B.B., "Taxonomy of Educational Objectives", Handbook II, Affective Domain, New York, 1964

[20] Kupka, I., "Zur Charakterisierung von Dialogsprachen". In: Goos,G. and Hartmanis, J., GI - 3. Jahrestagung, Springer, Berlin-Heidelberg-New York, 1973

[21] Lampl, G.R. and Schell-Haungs, I., "Bausteine in APL - Part 1, Part 2, Part 3". Stiftung Rehabilitation Heidelberg, 1975

[22] Sharp, I.P., "An Introduction to SHARP APL". SHARP Associates Limited, Toronto, 1975

[23] Specht, J. and Stengele, R., "Notenbank". Stiftung Rehabilitation Heidelberg, 1974

[24] Specht, J., "Heidelberger DOC-System". Stiftung Rehabilitation Heidelberg, 1975

[25] Steinhauer, H., "Structure of an Inf.-Doc.-System and its Realization as an Experimental System in APL-PLUS". In: APL-75, Proceedings of the APL-Congresse in Pisa, ACM, New York, 1975

[26] Zinn, K.L.,(ed.), "Computer Learning Under Evaluation". CLUE Report, Ann Arbor, 1971