Preface*

Intelligent Assistant Systems

Information is becoming the raw material of modern society. That “difference that makes a difference” (Bateson, 1979) is the driving force of modern service industry. Our information spaces have been technologized and their size as well as their complexity increased. Access to information spaces and the capability to use them effectively and efficiently has become a key economical success factor. An information dilemma has been diagnosed (Kuhlen, 1999). The alternatives are (1) to search the information spaces oneself and spend an increasing amount of time for satisfying one’s information needs; and (2) to delegate the searching task to an information assistant the performance of which is time-consuming to control and the effectiveness of which is hard to assess. The first choice dominates currently. However, using such an information assistant must be the ultimate goal, as one cannot reasonably expect that technologization will stop or even be undone. In fact technologizing of the world has a long history. It is inextricably connected with technologizing the word, (Ong, 1996, p. 101), as “(s)poken words are always modifications of a total situation which is more than verbal. They never occur alone, in a context simply of words.” Technologization has significantly restructured human consciousness Ong (1996). What we are witnessing right now should be understood as the latest step in the process of technologizing the word. That process began with the invention of script (about 3200-3100 BCE), and was followed by the invention of letters (about 1500 BCE), and print (1540 CE) (Ifrah, 2000, pp. 3-25).¹ The latest step is the automation of uttering and understanding words and text in which we here include calculation and computation. This sums up the capability of verbally controlling processes
of all kinds. Enzensberger (1970) reports the analogy that El Lissitsky drew in 1923 between language use and modes of transportation and, in particular, vehicles. He related articulated language to upright gait; writing to the wheel; and the printing press to animal powered carts. El Lissitzky did not suggest analogies in language use for automobiles and airplanes. One is tempted to suggest that computers correspond to automobiles, and the Web to airplanes. Would it be venturing too far to relate intelligent assistants to space crafts?

With this book, we contribute to the genesis of a particular kind of software systems, that is, intelligent assistant systems. Most of the chapters in this book, in fact, focus on what we call “assistance.” Currently there is no consensus regarding what exactly “assistance” means and what an assistant system is. Actually, we aim at bringing into existence such consensus. Several chapters in this book suggest preliminary answers to the question, “what is assistance?” This book furthermore demonstrates that assistant systems will become reality, as the technology for implementing intelligent assistant systems is available and the problems that require assistance for their solution are “out there.” The chapters in this book are quality assured versions of topics that were presented at a workshop that was held in March 2005. The acceptance rate was below 50%. A variety of issues regarding intelligent assistant systems are addressed such as their scope, purpose, architecture, implementation, deployment, their theoretical background, and use.

The issue underlying this book will not go away soon. Rather, the fact that assistance in the last 30 years has continuously been dealt with seems to suggest that it is used by the various authors because of a need to address a deep-rooted worrying issue. That issue is the role that is played by computers and humans respectively during their interaction. Several authors of the chapters in this book suggest that “intelligent assistant systems” will soon become reality. The technology developed so far and the approaches discussed in this issue show that this suggestion is realistic. A number of advanced general approaches to computer-based technologization of words and text were proposed in the past but were not heavily used in application systems, a point being made in Sowa (2002). We expect that this will not happen to intelligent assistant systems. They will make their way to mass systems, firstly due to aspects of the genesis and evolution of computers, and secondly due to being limited to relatively well-understood domains. The main point regarding the first issue is that since computers were invented in the fifth decade of the 20th century and came into wide-spread use in the sixth and seventh decade of that century they continued to become ever more powerful, so that solutions of practical problems became feasible that were
not feasible only a short time before that. It was as early as at the end of the
eighth decade of the 20th century that Winograd (1979), relying on a study
of the U.S. Department of Defense, wrote that computers “are not primarily
used for solving well-structured mathematical problems or data process-
ing, but instead are components in complex systems. … Many computer
scientists spend the majority of their time dealing with embedded computer
systems such as message systems and text editing and formatting systems.”
By about 1975 the task focus of computers had changed from computation
to communication (Hevner & Berndt, 2000).

With the dominant use of computing technology today being for business
application, (Hevner & Berndt, 2000, p. 13), a significant increase took place
in the number of individuals who are directly affected by computers and are
engaged in using them. They also use computers in an increasing number of
domains and tasks. The complexity of problems, for the solution of which
computers are key components, continues to increase. The traditional ap-
proaches of using computers turn out to be ineffective for these problems
of higher complexity. On the one hand, these approaches require the human
problem solver to at the same time maintain a good understanding of the
overall solution architecture as well as the solution detail. Many humans
experience difficulties meeting this requirement. On the other hand, the ap-
proach fails to let computers work out the problems and their solutions, as
humans often under the applying resource restrictions cannot understand and
validate these solutions.

The Oxford English Dictionary Online defines “complex” as “(c)onsisting
of parts or elements not simply co-ordinated, but some of them involved in
various degrees of subordination; complicated, involved, intricate; not eas-
ily analyzed or disentangled.” A related but more operationalized definition
was given by Victor Basili (see Banker, Datar, Kemerer, & Zweig, 1993).
He defined the term “complexity of a system” as the resources needed to
successfully use that system. If we consider a problem as a system the main
components of which are a current state C, a future state F, and an actor A who
wants to transform C into F given the boundary conditions B, then problem
complexity can be understood as a particular instance of system2 complexity,
that is, as the resources required to transform C into F. That definition can be
looked at from a quantitative point of view. Then problem complexity can, for
example, be understood in terms of the number of points being used to score
a problem. This approach is similar to the one taken in the Function Point
methodology (see Garmus & Herron, 1996). Rather than from this standpoint
problem complexity can be considered qualitatively. Then problem complexi-
ity can be understood in terms of the kind of resources needed for solving the problem at hand. New levels of problem complexity then ask for new kinds of resources or for new ways of using known ones. In terms of computer applications we think that new resources and new ways to use old resources can best be understood as what could be called “the interlocutor metaphor” or “role” that a computer system plays in an interaction with a human.

The concept of interlocutor metaphor can be discussed in at least two different ways. Firstly, it can be related to other high-level views of computer applications that make up a conceptual framework for understanding such applications. Secondly, the various interlocutor metaphors can be discussed. We follow a top-down approach and first list those views at computer applications that jointly seem to capture most of their diversity:

- **Functionality**, that is, the operations the application provides that together constitute its behavior.
- **Quantity**, that is, the number of computer applications the computer user has to or may consider using to achieve his/her goals while interacting with an application at hand.
- **Quality**, that is, the computer application’s fitness for stated and implied needs. In particular:
  1. **Behavior metaphor**, that is, a qualification of the application A that makes its behaviour intelligible, such as mechanical (behaviour explanation in terms of the structure of A), intelligent (behaviour explanation in terms of the reasoning capabilities of A), intentional (behaviour explanation in terms of the affection to objects in the environment of A), social (behaviour explanation in terms of A being part of a community of interacting entities), mobile (behaviour explanation in terms of A’s capability to change the location within the space in which it resides), and adaptive (behaviour explanation in terms of A’s capability to adapt to its environment).
  2. **Interlocutor metaphor**, that is, a qualification of the kind of interlocutor as which the application communicates such as tool, collaborator, peer, or master in the 1:1 human-computer-interaction. Also for the communication with software components a qualification of the kind of interlocutor such as client, server, broker, proxy, and so forth. For a more elaborated list of such interlocutor metaphors see, for example, Buschmann, Meunier, Rohnert, Sommerlad, and Stal (1999).
3. **Interface metaphor**, that is, a qualification of the way of using the application’s interface such as *direct manipulation*, *document*, and *dialog*. See, for example, Satzinger, Jackson, and Burd (2004) for an introduction.

4. **Interaction type**, that is, a qualification of the type of human-computer-interaction that the application implements, such as batch, transformational, or reactive.

5. **Interaction sense**, that is, the key human senses that the application appeals to.

The interlocutor metaphors for 1:1 human-computer-interaction were taken from human cooperative labor organization. The generalness of computers enables them to embody many different interlocutor metaphors and thus to appear as qualitatively different resources. Computers can therefore be used for solving problems of quite different complexity. The mentioned interlocutor metaphors for one-on-one human-computer-interaction are discussed in more detail below.

- **master – tool**, that is, in this relationship the master is superior to the tool and the latter has no or only little latitude in performing the tasks allocated to him/her. The success criterion applied to that relationship is that the tool exactly carries out what it is instructed to. Successful use of a tool depends on the master’s capabilities of, first, to structure problem solving procedures into steps that can be carried out by the tools at hand and, second, to handle the tools appropriately. A currently employed version of this relationship is the chain of command in the military all over the world.

- **master – collaborator**, that is, in this relationship the collaborator has a relatively large latitude regarding what tasks to perform and how to do them. However, the master is superior. The success criterion applied to this relationship is that the master achieves the goals. For that, the collaborator appropriately contributes capabilities to the problem the master tries to solve. Successful use of a collaborator depends on capability to learn about and to adapt to the master’s way of conceptualizing and solving problems as well as the master’s ability to use the collaborators capabilities and to delegate tasks, that is, let the collaborator do what the collaborator is good at. A currently employed version of this relationship
is the relationship between an enlightened manager\(^4\) and the manager’s subordinates.

- **peer – peer**, that is, in this relationship partners interact with each other and neither of them has power over the other one or dominates him/her. They cooperate according to their individual goals for tasks and in a way that appears advisable to them. Successfully employing this relationship means perpetuating it and depends on the capability to find peers that are willing to achieve compatible goals and to engage in communication processes that stimulate the ongoing process of goal achievement. A currently employed version of this relationship is the relationship between friends.

These three relationships between interlocutors cause five different roles in 1:1 relationships to exist in which computerized systems can interact with humans. The role of collaborator has two obvious specializations, that is, agent (one might consider the notorious 007 as an archetype of it), and assistant (an archetype of which might be Sherlock Holmes’ well-known companion Dr. Watson). In this book we mainly discuss aspects of the assistant role occupied by a computer in the master—collaborator relationship. We forecast that the trend will continue towards more computers becoming more powerful and used more in everyday life and that more complex applications are dealt with. We therefore anticipate that “computers in the role of collaborator as well as in the role of “peer” will continue to attract attention.

The analysis regarding the increasing complexity of problems that have to be solved is not entirely novel. It is actually quite old. Engelbart (1962) writes, for example, “Man’s population and gross product are increasing at a considerable rate, but the complexity of his problems grows still faster, and the urgency with which solutions must be found becomes steadily greater in response to the increased rate of activity and the increasingly global nature of that activity. Augmenting man’s intellect … would warrant full pursuit by an enlightened society if there could be shown a reasonable approach and some plausible benefits.”\(^5\) Engelbart even used a particular instance of the interlocutor metaphor. He called the resource that he was asking for “clerk.” There are at least two respects, however, in which we differ in our approach from Engelbart’s and comparable ones. We do not aim at creating a particular artefact that would be capable of aiding a human user in all contexts, situations, and conditions. Rather, we aim at domain dependent aids. We furthermore concede that “agent,” “assistant,” and “clerk” are unlikely
to be the only reasonable instantiations of the interlocutor metaphor that can be employed in solving complex problems.

We follow Winograd (1972) in conceiving assistants as systems that understand what they do and that are capable of answering respective questions. In the same spirit, Robertson, Newell, and Ramakrishna (cited after Sondheimer & Relles, 1982, p. 106) write, “One does not wield an intelligent assistant, one tells it what one wants. Intelligent assistants figure out what is necessary and do it themselves, without bothering you about it. They tell you the results and explain to you what you need to know.” From an architectural view that spirit suggests that assistants contain a domain model, a user model, and a situation model that combines the state of affairs of these two models as far as these are relevant for the task at hand. We do not aim at conceiving assistant systems as a particular kind of information system, as the latter from a functional view are understood as “technologically implemented medium for the purpose of recording, storing, and disseminating linguistic expressions as well as for the supporting of inference making” (Hirschheim, Klein, & Lyttinen, 1995, p. 11). In this definition and for this preface, a medium is a device or an individual that intermediates between two parties. For several modern assistant systems that would be a too narrow definition, since it presupposes a human being is one of the communicators and the communication—a verbal one. To cover these modern systems one must conceive assistant systems as embedded systems, that is, as systems that are embedded in other systems and store, record, exchange, and process signals. If one matches the master-assistant communication against Shannon’s communication model (Shannon, 1993) then one finds that certain encoding or decoding steps may be superfluous. Additional to this more technical point one finds that the assumption of a deliberate choice of messages does not necessarily apply to the input that is fed into an assistant system. Rather, such input may result from a master activity that is independent of the assistant. Consider for example a power-steering or a breaking-assistant. The master’s activity (steering or breaking respectively) occurs as a natural problem solving behavior and sensors derive the assistant system’s input from this behavior. Furthermore, the technical problem of Shannon’s communication theory (i.e., the reconstruction of a message that was chosen at a particular point in space at a different point in space at a later time) is not the technical problem of an assistance theory. The technical problem of assistance theory is firstly to identify the master’s state of affairs, intent, and activities; and secondly the identification of those responses to a master’s activity that are likely to aid the master in its current course of action as well as a qualification of the identified responses according to their suitability of being provided to the master.
With respect to those cases in which a human is presupposed to verbally communicate with the assistant, we, however, can stick to understanding the assistant as an information system. The kind of linguistic expressions that the assistant system is then supposed to handle successfully depends on the purpose of that system, so do the particular inferences that could be made. According to our understanding of assistant systems as highly capable, adaptive, cooperative, domain specific problem solving aid—the mentioned linguistic expressions must enable an assistant to autonomously find solutions for problems the master needs a solution for but does not want to or cannot deal with. The assistant’s capabilities must be such that they can be tailored towards an as smooth as possible interaction with the master. A promising approach to that was suggested by Maes (1994). It is based on a three-valued logic (“yes,” “maybe,” and “no” being the truth-values). The assistant would obtain a confidence value with respect to its current suggestion to the master. The assistant would then match this confidence value with one of the truth values and would use for that respective master-definable confidence threshold values. Further action to be taken would then depend in the obvious way on the resulting truth value.

Intelligent assistant systems are knowledge media, which, according to Tanaka (2003, p. 11), are defined as media to “externalize some of our knowledge as intellectual resources and to distribute them among people.” Until now the archetype of knowledge media is the book. Tanaka, (2003, pp. 30–31) provides interesting details regarding the history of books. While books are very effective in disseminating knowledge, they are less effective with respect to operationalizing (i.e., putting knowledge to action) and reusing it, as the book is a passive medium. Tanaka points out that each knowledge medium additionally to the knowledge represented in it is equipped with an access method. The computer is an active knowledge medium. The access methods that can be used with respect to the knowledge stored in a computer can thus be more effective and efficient than the ones that can be used with respect to a book. Also, parts of the knowledge stored in a computer can be stored in “active form,” that is, as a program that actually can be executed. Consequently the computer has the potential to substitute the book as the number one knowledge medium. For that to be achieved, however, the computer’s powers must be harnessed and suitable end-user interfaces provided. These interfaces would enable modification and reuse of the stored knowledge with high-level end-user-proof operations. This is what Tanaka and his collaborators have aimed at and achieved with the implementation of the meme media. These media are highly relevant for assistance theory, as they seem to have the potential for effective reuse of existing knowledge.
Assuming two systems interacting with each other, Liebermann and Selker (2000, p. 618) define the context of an interaction step as any aspect of that system interaction which is not explicit in that step. With this definition, obviously adaptive knowledge media such as assistant systems have to be context sensitive because adaptation means changes carried out though no respective explicit command; only an incomplete or syntactically invalid one was issued. As Liebermann and Selker (2000, p. 623) note, computer applications always include an application model, a user model, and a task model. Traditionally these models have been only implicit in the application’s code. Liebermann and Selker (2003, p. 623) argue that better suited “for contextual computing are systems that represent a system model explicitly and try to detect and correct differences between the user’s system model and what the system actually can do.” Winograd’s 1972 request that intelligent assistant systems should understand what they are doing seems to suggest that an intelligent assistant has an explicit and dynamic model of itself.

If one tries to identify computerized systems that implement the interlocutor metaphors mentioned above then one easily identifies tools, such as case-tools, text-processors, compilers, and so forth, as implementing the master-slave relationship. Regarding assistant systems the situation appears to be more difficult, however. The references (Boy, 1991; Burke, Hammond, & Young, 1997; Kaiser, Feller, & Popovich, 1988; Maes, 1994; Marcus, 1982; O’Connor, 2000; Winograd, 1972) show that assistants were constructed as an example for programming/software development and software project planning; e-mail processing; meeting scheduling; retrieving bibliographic information, piloting aircraft, and browsing (selecting new cars, choosing a rental video, finding an apartment, selecting a restaurant, and configuring home audio systems). There are several assistant systems in modern cars, for example, for steering, breaking, and accelerating. As an aid in the rather well understood area of software component installation, intelligent assistants have found a quite widespread use. In this book we bring some new examples. Of course there are not only success stories. Realizing intelligent automated assistance is quite difficult for domains that are not well understood such as everyday office work. To illustrate this Liebermann and Selker (2000) use the example of Microsoft’s Word that has the capability of after a “.” replace the first nonblank character by its capital-case version (i.e., replace “r” by “R”). Word often, even after a user has undone that correction, recorrects, and uses the capital character. It would not be too hard to stop Word from doing that. However, the vendor would have to see this as a sensible feature. Microsoft Inc., for example, has acknowledged difficulties with their office assistant by making it easier to switch on or off. They also have changed
the default for this assistant from switched-on to switched-off in the more recent versions of Word.

According to the Oxford English Dictionary Online, an assistant is “one who is present, a bystander; one who takes part in an assembly. Usually in pl.,” or “one who gives help to a person, or aids in the execution of a purpose; a helper, an auxiliary; a promoter; also, a means of help, an aid.” This book is supposed to promote the genesis of assistant systems as everyday-life gear. According to our analysis so far, assistant systems are interactive, user-adaptive problem solving aids that understand what they do, accept goals being set as input rather than instructions or deduce such goals, and, once these goals are identified, aim at solving them independently from their user. Assistant systems that interact with humans will often incorporate a question answering component as well as an explanation component. We assume that intelligence can improve effectiveness and efficiency of assistants. We thus anticipate assistants to be intelligent. For us that means that available information is exploited as effectively as possible for aiding the master in achieving the goals. It is characteristic for assistance (of information systems) (Boy, 1991) that (1) the responsibility remains with the master; (2) the master may ask or cause the assistant to execute a task whenever the master wishes so or it becomes necessary; and (3) the assistant may be proactive such that it is ready for upcoming tasks if the respective preparation does not interfere with the current activities.

Agent systems are similar to assistant systems. We distinguish these from each other in such way that agents are systems that have an operational mode in which they are not necessarily interactive, adaptive, or even accessible. Again querying the Oxford English Dictionary Online results in the following the word “agent” may be applied for referring to persons or things. The most important of the replied explanations are (1) “One who (or that which) acts or exerts power, as distinguished from the patient, and also from the instrument”; (2) “He who operates in a particular direction, who produces an effect. … The efficient cause”; (3) “Any natural force acting upon matter, any substance the presence of which produces phenomena, whether physical as electricity, chemical as actinism, oxygen, medicinal as chloroform, and so forth”; (4) “One who does the actual work of anything, as distinguished from the instigator or employer; hence, one who acts for another, a deputy, steward, factor, substitute, representative, or emissary. (In this sense the word has numerous specific applications in Commerce, Politics, Law, and so forth, flowing directly from the general meaning.)” These explanations seem to be consistent with our belief that agents are different from tools as
agents act on their own in achieving complex user goals while tools require successive instruction to carry out relatively simple tasks. It also appears to be justified to distinguish agents from assistants. Agents often do their job without much or any interaction with the user. Assistants typically would do their job in close interaction with the user, observe what he or she is doing and propose aid that is likely to be acceptable for the user.

**Book Outline**

The topics in this book are grouped into three sections. Within each section, the material is listed in the alphabetical order with respect to the author names. The topics in the first section deal with various kinds of foundational problems that are relevant for intelligent assistant systems. The topics in the second section are devoted to the exploration of the use of meme media technologies for intelligent assistant systems. Finally, the topics in the third section focus on using intelligent assistant systems.

Foundational problems that are relevant for intelligent assistant systems seem quite naturally to be connected to the information that such an assistant needs to acquire and use for being an effective and efficient problem solving aid to its master. These processes of information acquisition and utilization seem quite naturally to be related to reasoning about subjects that are relevant for intelligent assistant systems. As intelligent assistant systems are supposed to learn and draw inferences from available data, respective contributions need to explore processes of information acquisition and utilization as well as the structure of the involved subjects. Formalization of these processes and structures is the final aim, as that is the foundation on which intelligent assistant systems effectively can be developed. The Foundations chapter has six topics which in the sequel are briefly abstracted.

The chapter of Gunter Grieser and Steffen Lange deals with the question of how the interaction between humans (in the role of master) and computers (in the role of assistant) may be designed such that both partners are enabled to bring in their respective strengths. In the context of Information Extraction from semistructured documents, several scenarios of interaction are identified and analyzed with respect to their adequacy for cooperative problem solving. The theoretical considerations are illustrated by a particular interaction approach called consistency queries.

Learning aptitude is a must for intelligent assistant systems. The chapter by Klaus Jantke and Nataliya Lamonova focuses on this capability and consid-
ers a specific form of learning, that is, planning in complex and dynamic environments. Plans are hypotheses about how to solve a problem. Inductive learning in this topic is identified as a crucial task of an intelligent assistant system. In the area of therapy plan generation, inductive learning plays a particularly important role. Therapy planning must include reasoning about the conditions of plan-executability in the future. Estimates of several future parameter values are driving the inductive planning process. Obviously, aiding problem solving in other domains may benefit from this work.

The chapter by Klaus Jantke and Carsten Müller addresses the human use of search engines and in particular their still often unsatisfactory quality and the resulting negative usage experience. Currently, search engines are tools that are not easy to wield and that are far from assisting their human users. Wrappers are extraction procedures for extracting information on demand. Typed wrappers employing in a sophisticated way information extraction technology have characteristics of intelligent assistants. The chapter stresses the dichotomy of internal mechanisms and external assistance behavior. Jantke and Müller contribute to the discussion about the functionality and quality of a computerized system that establishes intelligent assistance. They also discuss the potentials and risks of a long-term trend of substituting tools by assistants.

Roland H. Kaschek, in his chapter considers the problem of assistance such that it can be separated into two parts, that is the assistant’s generation of best guesses of what to propose to the master, and the assessment of these best guesses according to their suitability of being proposed to the master. He proposes using a calculus of confidence in judgments for reasoning about models of the master. Models in his approach are understood as sets of judgments. While he initially considers judgments only individually, he later on provides concepts for combining judgments (i.e., conjunction of judgments and conditional judgment). The calculus proposed allows then to calculate the confidence in combined judgments. The chapter’s recommendation is then finally to propose to the master those best-guesses that score highest with respect to confidence.

Son Bao Pham and Achim Hoffman consider the problem of assisting human experts in developing knowledge bases for a number of natural language processing tasks. The techniques they propose are embedded into the knowledge acquisition framework KAFTIE. It has been designed earlier for building knowledge bases for natural language processing. KAFTIE includes an intelligent assistant, the rule suggestion module. This module assists the expert (in the role of the master) by suggesting new rules in order to address
incorrect behavior of the knowledge base at hand. Initial experiments with the new rule suggestion module are reported. They suggest that the cooperation between the expert and the assistant may lead to the development time for the knowledge being reduced and the knowledge base to be more compact.

In their chapter, Klaus-Dieter Schewe, Bernhard Thalheim, and Alexei Tretiakov aim firstly at formalizing user preferences, obligations and rights in the context of Web information systems and secondly at indicating how this formalization can be used to reason about a specification of a Web information system. Storyboarding is used for specifying Web information systems and the resulting storyboards are represented by an algebraic expression in a Kleene algebra with tests. User preferences are then formalized using the equational theory of these algebras. This enables sophisticated propositional reasoning that can be applied to WIS personalization. Obligations and rights give rise to a propositional deontic logic that, for example, could be applied with respect to problems of security or safety.

The second section is concerned with employing Tanaka’s meme media technology for assistant systems. This technology is not genuinely associated with assistance. However, its potential of significantly increasing effectiveness and efficiency of reusing knowledge artifacts together with computers being active media may actually prove to be key parts in assistant creation, maintenance, and use.

Kimihito Ito introduces a software architecture for building intelligent multimodal assistants. This architecture includes three basic components: a meme media system, an inference system, and an interface assistant system. This embodied assistant makes multimodal presentations available to users. The author’s experimental implementation of the architecture is reported. This experimental implementation shows how character agents are defined in a simple declarative manner using logic programming on meme media objects.

Nataliya Lamonova, Kimihito Ito, and Yuzuru Tanaka introduce an approach for creating Web-application capable of operating in complex environments. The approach shows how Meme Media technologies combined with other technologies can be used for solving several kinds of problems related to Therapy Planning in clinical trials. The combined us of logic programming and fuzzy logic for creating Web applications is also introduced in this chapter.

Yuzuru Tanaka focuses on reusing of resources found in the Web. At least two parameters of such reuse task can be identified, that is the scope of search and the required interoperability of the resources. Both of these parameters may
change during a reuse attempt in an unpredictable way. Intelligent assistant systems for such reuse tasks must aid the master in performing ad hoc federation of resources. This chapter deals with using meme media technologies for ad hoc federation of intelligent resources over the Web. It proposes the Wiki piazza architecture that works as a repository and lookup service, and combines this service with a location reference service to propose a way of restricting the scope of discovery using location-dependent contexts.

The third section deals with human learning. Learning is an activity undertaken for being prepared to master anticipated tasks. If carried out successfully it thus is highly beneficial for the involved individuals as well as their social groups. Learning itself, however, is a time-consuming and difficult process. Improving effectiveness and efficiency of learning processes is therefore a challenge. A drawback of many approaches that aim at improving learning based on computer technology is that they do not provide an effective learning environment but deteriorate and more or less restrict themselves to content delivery. This certainly ignores most of the complexities in technology supported learning. The topics in this chapter are briefly summarized in what follows.

Klaus Jantke, Christoph Igel, and Roberta Sturm set out with the observation that humans need assistance in learning. Learning belongs to the more complex activities undertaken by humans, as it deeply connected to the complexities of the learner. These authors argue that tools that support human learning tend to be complex themselves and be limited in effect. They propose steps toward assistance in e-learning and systematically illustrate these steps by means of the authors’ e-learning projects and systems eBuT and DaMiT. These steps are summarized in a process model proposed to the e-learning community.

Sabina Jeschke and Thomas Richter are concerned with the problem of the increasing speed with which new scientific knowledge is created. They argue that the traditional “learning on supply” does no longer apply and that learners should be guided towards efficient self-controlled learning. They recommend applying new media and technology for achieving a respective turning point in the educational system and discuss four areas of the application of these media and technologies: the presentation of mathematical content; intelligent lexicon toolkits that learn from natural language texts; homework training courses that are able to break up assignments into elementary subproblems as needed by the learner; and virtual laboratories that are able to provide courses that adapt to the errors of the learner and are rich enough to be used in research problems.
Alexander Krumpholz focuses on a particular aspect of the important problem of acquiring physical capabilities (rather than mental ones). He describes the virtual trainer that was developed for CSIRO’s temporal bone dissection simulator. This simulation software runs on an immersive haptic virtual reality environment. The prototype system uses a task model based on a finite state machine to describe the procedure and interactive landmarks to trace the user’s action in relation to vital structures. This gives the user an activity related feedback. A virtual trainer of product quality needs to take note of the research on intelligent tutoring systems for tailoring feedback for the students and maximize their knowledge and skill acquisition. Various features for such a system are described in the chapter.

Roland H. Kaschek
Massey University, New Zealand

References


---

**Endnotes**

1 See, furthermore http://www.gutenberg.de/erfindu2.htm (In German) for different figures that seem to be suggested by latest research.

2 For an accessible introduction to systems theory see Chapter 4 of Ahituv and Neumann (1990).
The metaphors “mechanical” and “intentional” are adapted versions of Daniel Dennett’s “mechanistic” and “intentional” stances. See, for more detail, for example, Haase (1997).

The rate of this kind of manager in the set of all managers is apparently rather small.

It may be instructive to also take note of Engelbart and English (1968).

A very similar view is taken in Liebermann and Selker (2000, p. 620).

Please note that I use the term “knowledge representation” as a metaphorical expression. I do not really believe that knowledge can be represented. Rather I think that Shannon’s communication model applies to human communication. According to this model humans select messages but send and receive only signals. Human communication has a chance of working out fine if these signals then can be decoded and a suitable message be re-constructed. In my view the term “knowledge” refers to a reification of the capability of systems to respond to external stimuli differently when they occur at different times or repeatedly.

Obviously the “.” is taken as an indicator for the end of sentences. That this is not the only use of the symbol causes the problem.

Note that the term (software) agent is used in a number of different ways. Tanaka (2003) discusses three meanings of the word. O’Connor (2000, p. 38) says that among “the agent community there is no commonly agreed definition” of the term.

I am thankful for the suggestions made by Barry Jackson, Klaus P. Jantke, Nadav Katz, and Bernhard Thalheim.