CHIME: Service-oriented Framework for Adaptive Web-based Systems

Vadim Chepegin[1], Lora Aroyo[1,2], Paul De Bra[1], Geert-Jan Houben[1]

[1] Department of Mathematics and Computer Science
Eindhoven University of Technology
P.O.Box 513, 5600 MB Eindhoven, The Netherlands

[2] Department of Computer Science
University of Twente
P.O.Box 217, 7500 AE Enschede, The Netherlands

Abstract

In this paper we present our view on how the current development of knowledge engineering in the context of Semantic Web can contribute to the better applicability, reusability and sharability of adaptive web-based systems. We propose a service-oriented framework for adaptive web-based systems, where the main goal is to help the semantic enrichment of the information search and usage process and to allow for adaptive support of user activites. In other words, our aim is to provide flexible information access, presentation and update to a broad range of users (individual and groups) in a personalized way within the context of pursuing a user's goals and performing tasks. We take an ontological approach to enable a shared understanding of concepts throughout the system and to provide semantic relationships between the information resources and the user's knowledge of (or interest in) them. We argue that the future of adaptive web-based systems lies in the modularity of the architecture and the openess to interoperate with other applications or compontens. To achieve this we adopt the concept of the UPML framework for semantic web service integration. Our ideas are illustrated in the context of the Token2000 project for Cultural Heritage in Interactive Multimedia Environments (CHIME) and show how combining elaborate AI strategies with the simplicity of hypermedia interaction can result in more easily applicable knowledge-based systems, or in more reasoning enhanced adaptive hypermedia systems.

1. Motivation

The more the corpus of information accessible through Internet grows, the more crucial it becomes to enhance the ways of finding, accessing and retrieving the right piece of information at the right time as part of our overall problem solving activities. This moves the focus primarily towards the provision of tools to support users to cope with the complexity of the information space and the dynamically changing user demands. Ideally we need a number of independent services which, when combined "on the fly", can support any type of activity of any type of users (on the various levels of granularity of their problem solving activities). Traditionally the field of Artificial Intelligence (AI) implements and successfully applies elaborate modeling approaches within the context of knowledge-based systems (KBS) in order to support users in performance their tasks [1, 9, 15, 25 and so on]. Although lately a lot of research effort is concentrated to decrease the complexity of the KBS and to open them up to various application domains, their design and implementation is still rather application dependent and their maintenance is a sophisticated and time consuming task. This subsequently obstructs their popularity and wide applicability. At the same time, the simple concept of adaptive hypermedia systems wins more and more interest in a short time. Their simple reference architecture [3, 7], aimed at a quick adaptive response, appeared to be very suitable for the web environments. On the other hand they lack the notion of solid knowledge and reasoning, which weakens their position within the context of adaptive systems. The simple modeling approaches appear not to be enough to assess the user's knowledge and to provide accurate adaptation [6]. Their current notion of "user's knowledge" does not cover various knowledge facets, which are important for the assessment of the user's knowledge level.

An ultimate goal within the current ubiquitous software environments, where the users are mobile, mainly web-based and interact simultaneously with various applications, is to allow for reasoning-based adaptation accross applications, see for example [24]. For this we need the simple concept of hypermedia and improve

the adaptation strategies using methods and techniques from AI-based systems. In order to achieve the interoperability across applications we need to offer an open and modularized architecture, which will be able to interact, exchange data and share components. The provision of semantically rich descriptions of the components' functionality and their internal formats is important in order to allow for interoperability among system components. Finally, maintaining a generic sharable (dynamic) user model is needed to serve as a communication point for the different systems [11, 12]. The biggest challenge here lies in the sharing, synchronization and interpretation of the user model. This way the user's behaviour within each system will be permanently evaluated and more detailed and richer user models will be achieved in order to allow for enriched adaptation and personalization of the content.

2.Background

If we look at the cultural heritage domain and specifically the one of Dutch national museums, we quickly realize that the most artifacts are inaccessible to the general public and experts distributed around the world. Museums own many more artifacts than they can show in their main exhibition at any one time. Large investments are being made to "capture" the artifacts digitally, and projects have been carried out to give broader access to the digitized material. There are two limitations to the current approaches, however. First, they focus only on a single type of user, e.g. novice user in the Rijksmuseum ARIA system or expert user in the Rijksmuseum AdLib database. Second, the system is unidirectional, i.e. "experts" input information into the system and "users" query this information. An important aim of the CHIME project is to offer ways to remove these restrictions and to allow information to be presented to a broad range of users in a suitable way and to allow users to add their own information to the repository, while respecting the integrity of the original historical sources. This allows a decentralized approach to the enrichment of the information in the repository by all its users and to the benefit of all its users. To achieve this we need to focus on providing adaptation to the different users' goals and characteristics, so that we can minimize the time and optimize the efficiency of achieving the goal for each user. Within the scope of CHIME project we focus on (1) tailoring the presentation of cultural information extracted from existing repositories to different types of users; and (2) allowing users not only to query the information database but also to add their own remarks (relevant multimedia data, such as figures, video material, photos, newspaper articles, spoken commentary) to the repository. In other words the central themes of the project are supporting different user performing different tasks, and providing functionality with respect to querying as well as modifying the repository. A central role in this is played by the *Modeling of the User* and the *Modeling of the Content* within a multi-task context (e.g. presentation generation, material searching, etc.). During the past decades we have been observing the success of different types of software systems, which adaptively support users in various activities, e.g. Expert Systems (ES), Intelligent Tutoring Systems (ITS), Information Retrieval Systems (IR), Adaptive Hypermedia Systems (AHS), Web-Based Information Systems (WIS) [4, 5, 14, 20, 21, 26, 27]. The observed problem is that most of the AI systems for user modeling and ITS are built in a very application dependent manner and the process of their developement is time consuming and not oriented towards sharing [17]. On the other hand IR systems propose useful and precise techniques to retrieve data, but they do not consider the application of user features. Finally, AHS and WIS are primarily targeting the adaptation and personalization to the user needs and goals, but they lack the sophisitcation of the the IR and UM techniques and the precision of the user input. Thus, we argue that an interdisciplinary approach would be most beneficial, alllowing us to combine elaborate knowledge acquisition and user modeling techniques from AI and user driven design from HCI and to apply them within the context of adaptive web-based systems.

Currently research in the area of Semantic Web, originating primarily from the knowledge engineering and artificial intelligence fields, with a special focus on ontologies and Web Services, provides a number of standards and accompanying solutions which can be used to achieve the above mentioned requirements. On the one hand we have the notion of ontology, which plays a role in facilitating the sharing of meaning and semantics of information between different software entities. A number of representational formats have been proposed as W3C standards for ontology and metadata representation. The most current advances with OWL exploit the existing web standards (e.g. XML, RDF and RDFS) and add the primitives of description logic as powerful means for reasoning services. The next step in this process of opening the AHS architectures is made by applying a Web Service perspective on the system components. Web Services (also known as software agents) make use of the above mentioned semantics and offer means for flexible composition of services (system components) through automatic selection, interoperation of existing services, verification of service properties and execution monitoring. They appear to be a useful solution for achieving the modularization. We can reach reasonable automatization and dynamic realization of the main aspects of web services (e.g. web service location, composition and mediation) by extending them with rich formal

descriptions of their competence (in standardized languages such as RDF or OWL). This way we can allow adaptive web-based systems to reason about the functionalities provided by different web services, to locate the best ones for solving a particular problem and to automatically compose the relevant web services for dynamic application building.

Within the context of the CHIME framework we exemplify how the use of ontologies and Semantic Web open standards can be benefitial for the improvement of the adaptation and the interoperability among internal and external system components. We also aim at (1) enhancing the interaction between system agents and providing richer semantics for the adaptive support of various user types, and (2) standardization of user modeling and adaptation in order to enable shareability and interoperability among various adaptive web-based systems. We take an interdisciplinary perspective and show how to enhance existing adaptive hypermedia systems with elaborate AI reasoning methods in order to improve the user's adaptation. This is the first step towards defining a new class of Intelligent Hypermedia Environments (IHE) as a crossing of AHS and ITS.

The remainder of this paper is organized as following. In Section 3 we introduce our architectural considerations for CHIME framework. In Section 4 we position our research in the context of related projects and intiatives, and finally in Section 5 some conclusions and future work are presented.

3. CHIME Architectural Considerations

The CHIME system can be viewed both as an Adaptive System and as a Hypermedia System, which both belong to the broader class of Knowledge-based Systems (KBS). The KBS perspective gives us a basis to develop CHIME as an adaptive (hypermedia) problem-solving environment. In order to achieve this we target a *modular* system architecture of *reusable components*, which supports the *shareability* of CHIME components as well as the use of *third-party components* within the CHIME system. We aim at *standardizing the protocols for message exchange* both between internal and external components. Another important requirement for our architecture is to allow for *scalability* in terms of multi-user support within a multi-task context.

In order to achieve the *modularity* we propose a multi-agent architecture, where both human and software components are considered to be agents. In order to support the *shareability* of CHIME components we follow the current Semantic Web notion of software agents in terms of Semantic Web Services. They allow discovery, configuration and management of agents. Next to this, the ontological engineering offers methods and technologies for adding semantic descriptions to content, functionality and dataflows, in order to allow for the discovery, configuration and management of internal and external agents. The basic idea is that by augmenting encapsulated system modules with rich formal descriptions of their competence we can further improve and also automate many aspects of the system management. Furthermore, by introducing dynamic shareable user model we also enable the inter-system interactions, shareability and reusability of modules. By applying open standards for the realization we secure the interoperability and the wide applicability of the adaptive web-based systems. Exsiting web service frameworks, e.g. the Internet Reasoning Service (IRS-II) introduced by Motta et al. [16], show how we can support the publication, location, composition and execution of heterogeneous semantic rich web services. It uses **UPML** (Unified Problem Solving Method Developments Language) for the specification of reusability in knowledge-based systems by defining how we can build elementary components and how these components can be integrated into one whole system [8]. The IRS-II approach enables us to support *capability-driven service invocation* (e.g. find a service that can solve problem X) because of the explicit separation of *task specifications* (the problems which need to be solved), *method specifications* (the ways to solve problems), and *domain models* (the context in which these problems need to be solved).

In Figure 1 we illustrate our idea in the context of high-level services in the CHIME architecture, adapted from (Motta et al., 2003). It alters the well known adaptive hypermedia reference models, e.g. AHAM [3] and the Munich Model [13], by introducing the notion of services and semantic description of functionality in terms of ontologies. The CHIME architecture distinguishes between the following components at the highest level of abstraction:
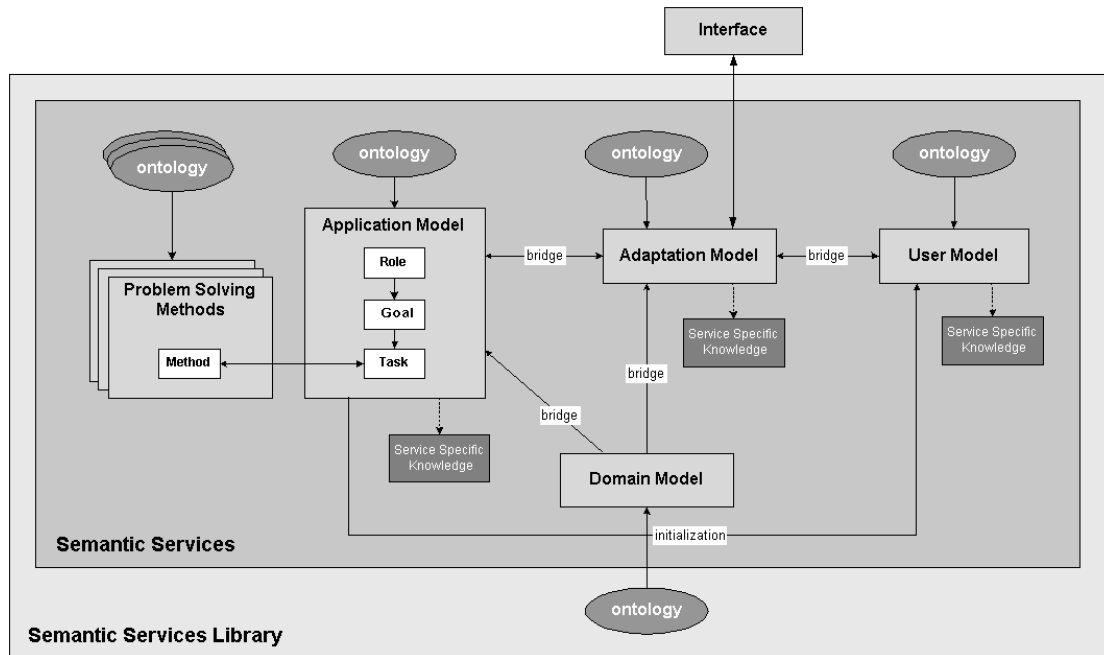
Figure 1: modular CHIME architecture (adapted from Motta et al. 2003)

- **Domain Model Service**, which is responsile for the explicit storage and description of the domain knowledge in terms of concepts of a Domain Ontology;
- **User Model Service**, which is an active agent following the user (inside and possibly also outside the system) in order to collect and further analyze data about the user's activities; this allows for inference of new knowledge about the user;
- **Adaptation Model Service**, which is an agent responsible for the application of rules to plan and perform the adaptation;
- **Application Model Service**, which contains a generic description of the user tasks in the context of Role-Goals-Tasks-Methods (Problem Solving Methods) chains. The Problem Solving Methods (PSMs) provide abstract descriptions of reasoning processes which can be applied to solve tasks in a specific domain. This way a clear distinction is made between tasks and methods. As a result flexible mappings between services and problem specifications can be made. Dynamic, knowledge-based service selection is also enabled in this way [16].
- **Bridges**, in accordance with the UPML framework connector definition [8], specify mappings between the different model services within CHIME framework.
- **Ontologies** play the role of shareable dictionaries in order to define and unify the system's terminology and properties to describe the knowledge of each CHIME service. Each service is specified by means of a corresponding ontology. This way a common ground for knowledge sharing and interoperability between CHIME agents (services) is provided. The choice of using open standards (e.g. XML, RDF, OWL) helps standardizing and formalizing meaning and enables the reuse and interoperability in the context of WWW. Finally, it all leads to very modularized architecture with an enhanced service maintenance.

A central role in CHIME is played by the Application Model Service. In the interaction with the application each user is represented by a particular role (e.g. guest, expert, student, teacher). This role defines for her a corresponding behavior in terms of the goals to achieve. In order to accomplish these goals the user applies appropriate tools (applications or agents), which provide one or several corresponding problem-solving methods (PSMs). Each of these applications maintains additional information about the user-system interactions (e.g. in the form of tables with description of topology and initial probabilities of Belief Networks) in order to be able to monitor and further reason over each step in the entire process. For instance, when the user works with a selected application, which offers a particular PSM, every action she performs through the user interface is communicated to the Adaptation Model Service, which is responsible for selecting the adaptation strategies on the basis of the User Model, the Domain Model and the Application Model. When the decision about the next step is made the Adaptation Model Service sends this information together with the information about the user's actions to the User Model Service. The User Model Service updates the User Model with the new values. The user information is stored and a reasoning engine infers new knowledge from it and makes predictions concerning the user's future behavior. This new knowledge is

sent back to the Adaptation Model Service, which makes a decision about "the best" next information item to be presented to the user. The interface component presents this information to the user and her feedback is translated back to the Adaptation Model Service. This completes the main system loop, which repeats as long as the user interacts with the system.

4. Related Work

In this section we give a brief overview of the related research and studies which served as an inspiration to our approach.

The distributed notion of WWW influences among other areas also the software development process in the direction of intelligent software brokering. A major contribution in this field are the results achieved within the context of the **i-brow**[3] project. It aims at providing intelligent reasoning services on the Web by integrating research on heterogeneous databases, interoperability, ontologies and Web technology within KBS. The objective to increase the level of support on the global information infrastructure and to hide the technological complexity of the underlying system is achieved by the provision of intelligent brokering services. As a result of this, an Internet Reasoning Service (**IRS-II**) has been proposed. It is a Semantic Web Services framework, which allows applications to semantically describe and execute web services [16]. Thus, a framework (**UPML**, similar to the **CML** developed in the CommonKADS project [23]) has been developed to describe modular and reusable architectures and components to facilitate their semi-automatic reuse and adaptation. It partitions the knowledge into ontologies, domain models, task models, and problem solving methods (PSMs) and connects them via bridges. Each of knowledge model types is supported by corresponding ontologies. The work on IRS-II has focused on (1) the integration of the UPML framework with current web service standards, and (2) the enabling of developers to semantically describe code (currently Lisp and Java) of web services. In the context of CHIME, the key design decision we made was to associate each PSM with exactly one web service although a web service may map onto more than one PSM since a single piece of code may serve more than one function. Problem-solving methods provide reusable architectures and components for implementing the reasoning part of knowledge-based systems. We adopt the ideas of i-brow[3] and IRS-II about semantic web services also within CHIME, and thus apply the notion of reasoning services within distributed web-based systems and use the UPML language to specify the components and their relations.

Another approach in this direction is presented in **SOAR**. It offers a general cognitive architecture for developing systems that exhibit intelligent behavior [22]. SOAR defines a single framework for all tasks and subtasks (problem spaces), a mechanism for generating goals (automatic subgoaling) and a learning mechanism (chunking). Next to this a single representation of permanent (productions) and temporary knowledge (objects with attributes and values) is given. All decision computations are made through the combination of relevant knowledge at run-time. A desired state for CHIME is to use all the available knowledge for each task that the system encounters. Unfortunately, because of the complexity of retrieving the relevant knowledge this goal is not our focus, as with the increase of the knowledge body, the tasks become more diverse, and the requirements in system response time become more stringent. The best that can currently be obtained is an approximation of complete rationality and we consider the SOAR design as an investigation of one such approximation.

An important part of the modeling of user tasks, goals, roles and cognitive processes, is played by the learning theories. Two of the most applied ones are **ACT-R** proposed by John Anderson [2] and **Constraint-Based Modeling** (**CBM**) proposed by Stellan Ohlsson [19]. They are both based on the distinction between declarative and procedural knowledge, and the view that learning consists of two main phases. In the first phase the declarative knowledge is encoded and in the second it is turned into more efficient procedural knowledge [20].

Another aspect of modeling cognitive processes is given by existing modeling languages like Hank and UserML. **Hank** is a relatively new cognitive process modeling language, which is designed to be easy to grasp by non-programmers and powerful enough to build models of non-trivial psychological theories [18]. A central role in the modeling is played by the user model, which requires a protocol for encoding the information about the different users, and also makes it possible that any given adaptive system should be able to benefit from others sharing the same user model and that user modeling agents should follow you around [12]. The **User Modeling Mark-up Language (UserML)** offers an XML-based exchange language which is based on an ontology that defines the semantics of the XML vocabulary (UserOL). It provides a

modularized approach for module connections (via identifiers and references to identifiers) which allows for a graph structure representation [10]. It can be used as a protocol language between a User Model service and other services as well as the language for internal representation of user in the User Model.

5. Further Work and Conclusions

The next step in this research context is to select languages for the agent communication and to specify conventions for agent interaction. This will be considered with other participants of our common project. The further development of the CHIME system will involve application of the principles and techniques of already established examples of multi-agent systems. For example, AgentBuilder offers a good environment and tools for constructing intelligent software agents and agent-based systems. Next to this the Internet Reasoning Service (IRS-II) offers a flexible framework for the integration of web services. We can probably also find a good examples of architectures and infrustructures of web-services in IBROW, UPML, and so on.

Acknowledgements

References

1. Aroyo, L., De Bra, P. (1999). Agent-oriented Architecture for Task-based Information Search System, Proceedings of the "Interdisciplinaire Conferentie Informatiewetenschap", pp. 94-98, Amsterdam, 1999.
2. Anderson, J.R. (2000). *Cognitive Psychology and its Implications.* Fifth Edition. Worth Publishers, NY. - p.241.
3. De Bra, P., Houben, G.-J., Wu, H. (1999) *AHAM: A Dexter-based Reference Model for Adaptive Hypermedia*. In *ACM Conference on Hypertext and Hypermedia*, pp. 139-146, February 1999. URL: http://delivery.acm.org/10.1145/300000/294508/p147-de_bra.pdf
4. De Bra, P., Aerts, A., Smits, D., Stash, N. (2002) *AHA! The Next Generation*, In *ACM Conference on Hypertext and Hypermedia*, pp. 21-22, June 2002. URL: http://delivery.acm.org/10.1145/520000/513347/p21-de_bra.pdf<
5. De Bra P., Aerts, A., Berden, B., de Lange, B., Rousseau, B., Santic, T., Smits, D., Stash, N. (2003) "AHA! The Adaptive Hypermedia Architecture". In *ACM Conference on Hypertext and Hypermedia*, pp. 81-84, August 2003. URL: http://delivery.acm.org/10.1145/910000/900070/p85-eiron.pdf
6. Brusilovsky, P. and Maybury, M. T. (2002). From adaptive hypermedia to adaptive Web. In P. Brusilovsky and M. T. Maybury (eds.), *Communications of the ACM* **45** (5), Special Issue on the Adaptive Web, 31-33.
7. Halasz, F. and Schwartz, M. (1994). The Dexter Hypertext Reference Model. Communications of the ACM, Vol. 37, nr. 2, pp. 30-39, 1994.
8. Fensel, D., Motta, E., Benjamins, V.R., Decker, S., Gaspari, M., Groenboom, R., Grosso, W., Musen, M., Plaza, E., Schreiber, G., Studer, R., and Wielinga B. (1999) "The Unified Problem-solving Method development Language UPML". ESPRIT project number 27169, IBROW3, Deliverable 1.1, Chapter 1., URL: ftp://ftp.aifb.uni-karlsruhe.de/pub/mike/dfe/paper/upml.pdf
9. Haake, J., and B. Wilson (1992). "Supporting Collaborative Writing of Hyperdocuments in SEPIA." *Proceedings of the ACM 1992 Conference on Computer Supported Cooperative* Work, Toronto, Ontario, 1-4 November 1992.
10. Heckmann, D., Kruger, A. (2003) A User Modeling Markup Language (UserML) for Ubiquitous Computing". In *User Modeling 2003 Conference*, pp. 393-397.
11. Kay, J, R.J. Kummerfeld and P Lauder, (2002) Personis: a server for user models, De Bra, P, P Brusilovsky, R Conejo (eds), *Proceedings of AH'2002, Adaptive Hypermedia 2002*, Springer, 203 - 212.
12. A. Kobsa (2001): Generic User Modeling Systems. *User Modeling and User-Adapted Interaction* 11(1-2), 49-63. URL: http://www.ics.uci.edu/~kobsa/papers/2001-UMUAI-kobsa.pdf

13. Koch N., Wirsing, M. (2002). The Munich Reference Model for Adaptive Hypermedia Applications. De Bra P, P. Brusilovsky, R. Conejo (eds), *Proceedings of AH'2002, Adaptive Hypermedia 2002*, Springer, 213 - 223.

14. Koedinger, K. R., Aleven, V., Heffernan, N. T. (2003). Toward a Rapid Development Environment for Cognitive Tutors. 12th Annual Conference on Behavior Representation in Modeling and Simulation. Simulation Interoperability Standards Organization.
URL: http://nth.wpi.edu/pubs_and_grants/03-BRIMS-063.doc

15. Leake, D., Hammond, K., Birnbaum, L., Marlow, C., and Yang, H. (1999). Task-based knowledge management. In *Exploring Synergies of Knowledge Management and Case-Based Reasoning, Proceedings of The American Association of Artificial Intelligence (AAAI-99) Workshop*. Orlando, Florida: AAAI Press. - Pp. 35-39.

16. Motta, E., Domingue, J., Cabral, L., and Gaspari, M. (2003) "IRS-II: A framework and Infrastructure for Semantic Web Services". In *2nd International Semantic Web Conference 2003 (ISWC 2003)*, 20-23 October 2003, Sundial Resort, Sanibel Island, Florida, USA - in press.
URL: http://www.cs.unibo.it/~gaspari/www/iswc03.pdf

17. Murray, T. (1999). Authoring intelligent tutoring systems: an analysis of the state of the art, *International Journal of Artificial Intelligence in Education*, 10, 98-129.

18. Mulholland, P. and Watt, S. N. K. (2000). Learning by building: A visual modelling language for psychology students. *Journal of Visual Languages and Computing*, **11** (5), 481-504.

19. Ohlsson, S. (1994) "Constraint-Based Student Modelling. *Student Modelling: The Key to Individualized Knowledge-Based Instruction*". pp 167-189, Springer-Verlag, 1994.

20. Mitrovic, A., Koedinger, K.R., Martin, B. (2003) "A Comparative Analysis of Cognitive Tutoring and Constraint-Based Modeling". In Proceedings of 9th International Conference on User Modelling 2003, USA, pp.313-322.

21. Page, L., Brin, S. (1998). The anatomy of a large-scale hypertextual web search engine. *Proceedings of the 7th Intl. WWW Conf.*, 107-117, 1998.

22. Rosenbloom, P.S., Laird, J.E., Newell, A. (editors.) *The Soar Papers: Research on Integrated Intelligence*. MIT Press, 1993.

23. A. Th. Schreiber, B. J. Wielinga, J. M. Akkermans, W. Van de Velde, and R. de Hoog. (1994) CommonKADS: A comprehensive methodology for KBS development. *IEEE Expert*, 9(6):28–37, December 1994.
URL: http://www.cs.vu.nl/~guus/papers/Schreiber94f.html

24. Slaney, M., Subrahmonia, J., Maglio, P. (2003). Modeling Multitasking Users. In *P. Brusilovsky et al. (Eds.): UM 2003, LNAI 2702*, pp. 188.197.

25. Sycara, K. and Zeng, D. (1995). Task-Based Multi-Agent Coordination for Information Gathering. In Working Notes of the AAAI Spring Symposium Series on Information Gathering from Distributed, Heterogeneous Environments. Stanford, CA, 1995.

26. Szolovits, P., Long, W.J. (1982). The development of clinical expertise in the computer. In P. Szolovits, editor, *Artificial Intelligence in Medicine*, pages 79-117, Westview Press, Boulder, Colorado, 1982.

27. Weber, G. and Brusilovsky, P. (2001) ELM-ART: An adaptive versatile system for Web-based instruction. *International Journal of Artificial Intelligence in Education* **12** (4), Special Issue on Adaptive and Intelligent Web-based Educational Systems, 351-384.