AALBORG UNIVERSITY

DENMARK

Personalizing Access to Learning Networks

Dolog, Peter; Simon, Bernd; Nejdl, Wolfgang; Klobucar, Tomaz

Link to publication from Aalborg University

Personalizing Access to Learning Networks

Peter Dolog, Aalborg University

Bernd Simon, Vienna University of Economics and Business Administration

Wolfgang Nejdl, University of Hannover

and

Tomaž Klobučar,Jozef Stefan Institute

In this article, we describe a *Smart Space for Learning^{TM}* (SS4L) framework and infrastructure that enables personalized access to distributed heterogeneous knowledge repositories. Helping a learner to choose an appropriate learning resource or activity is a key problem which we address in this framework, enabling personalized access to federated learning repositories with a vast number of learning offers. Our infrastructure includes personalization strategies both at the query and the query results level. Query rewriting is based on learning and language preferences, rule-based and ranking-based personalization improves these results further. Rule-based reasoning techniques are supported by formal ontologies we have developed based on standard information models for learning domains, ranking based recommendations are supported through ensuring minimal sets of predicates appearing in query results. Our evaluation studies show that the implemented solution enables learners to find relevant learning resources in a distributed environment and through goal-based personalization improves relevancy of results.

Categories and Subject Descriptors: H.3.3 [**Information Search and Retrieval**]: Information filtering, Query formulation, Relevance feedback, Search process, Selection process; H.3.4 [**Systems and Software**]: Distributed systems, Information networks, User profiles and alert services; H.3.5 [**Online Information Services**]: Web-based services, Data sharing; H.5.4 [**Hypertext/Hypermedia**]: Architectures, User issues; I.2.4 [**Knowledge Representation Formalisms and Methods**]: Predicate logic, Semantic networks, Representation languages; K.3.1 [**Computer Uses in Education**]: Distance learning

General Terms: Human Factors, Standardization, Design

Additional Key Words and Phrases: Personalization, Ontologies, Semantic Web, Learning Networks, Personalized Access

---

Peter Dolog, Department of Computer Science, Aalborg University, Fredrik Bajers Vej 7E, DK-9220 Aalborg East, Denmark, `dolog@cs.aau.dk`

Bernd Simon, Vienna University of Economics and Business Administration, Augasse 2-6, A-1090 Vienna, Austria, `bernd.simon@wu-wien.ac.at`

Wolfgang Nejdl, L3S Research Center, University of Hannover, Appelstr. 9a, D-30167 Hannover, Germany, `nejdl@l3s.de`

Tomaž Klobučar, Jozef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia, `klobucar@e5.ijs.si`

1. INTRODUCTION

Recent economical changes require more and more people to acquire new knowledge and skills timely and effectively. Web-based learning solutions allow us to support learning in a more personalized and on-demand manner. In the corporate context, web-based learning has the potential to reduce learning delivery costs, create more effective learning environments, accelerate time-to-competency and increase collaboration. For educational institutions Web-based learning introduces new educational options to improve teaching and learning practices [Harasim et al. 1995].

In this paper we focus on the issue of how to reach a particular learning objective effectively and propose personalized access to learning networks as an important part of the solution. We assume that presenting and consuming the "right" content is of paramount importance in order to create effective learning environments, especially in a cost-driven context such as corporate education and training. We focus on facilitating Learning Networks, as a shared space of interconnected knowledge repositories. These knowledge repositories hold information on learning resources and people, and have the potential to make a vast number of learning resources available. Personalization is needed to support users in making the right choices from an expansive list of options.

Setting up a learning network imposes a significant research challenge: In learning networks several knowledge repositories that use different schemas for describing educational artefacts such as learning resources or learner profiles need to communicate with each other. Although several metadata standards have been proposed, so far no learning network has been built that takes advantage of these specifications for interconnecting learning repositories, personalizing access for the users of the network. As most personalization techniques have only been applied in closed world settings so far, expanding personalized search into a distributed environment was an additional challenge, requiring a common ontology and an effective way of query processing.

In this paper we present a functional framework for learning networks which enables access to learning resources in a personalized and distributed environment. The SS4L framework allows to extend personalization and metadata access functionalities as needed and required by a user. In this framework, personalization is performed in two ways:

—Query — queries can be rewritten to more specific or more general ones

—Results — query results can be differently ranked and annotated

In the first case, personalization happens before a query is submitted to the learning network. In the second case, personalization happens after query results arrive from the learning network.

We also describe a methodology for personalized access where

—Learning resource metadata are seen as constraints on the usage of the learning resources they annotate;

—Learner metadata are used to model learner's abilities and performance;

—Personalization is treated as a set of decisions among variable learning resources

where the decisions are based on the comparison between learner and learning resource metadata.

The design rationale and advantage of the SS4L framework is to provide users and developers with the flexibility to add, change, and extend personalization techniques and to connect them to the ones provided within our framework. Our evaluation shows that such an approach is well accepted by users.

The paper is structured as follows. In Section 2 we present the functional components for our learning networks and describe how to extend the network with personalization and metadata access strategies. Personalization strategies we have implemented and experimented with are described in Section 3. In Section 4 we discuss our solution for knowledge representation in this context, focusing on the personalization aspect. Section 5 provides and discusses our evaluation results. Section 6 contrasts our approach with related work. The paper concludes with a summary and proposals for further work in Section 7.

2. SS4L FUNCTIONAL FRAMEWORK

2.1 Queries and Query Results

Several adaptive Web-based educational systems such as ELM-ART [Weber and Brusilovsky 2001], AHA! [Bra and Calvi 1998], or Interbook [Brusilovsky et al. 1998] have been developed to personalize learner experience. These systems adapt presentation or navigation by showing or hiding text and media fragments [Brusilovsky 2001]. They also generate, hide, or show links which point to additional content or services. The basic assumption behind their personalization functionalities is that the system maintains knowledge about educational artefacts and learners in order to reason about whether a link, text, or page fragment will be shown, hidden or customized.

In learning networks this assumption does not hold any more. Knowledge about learning resources and links to the learning resources as well as knowledge about learners is distributed and has to be collected from various sources. Therefore, the main concern in the learning network is how to handle queries for learning resources and their results. Query rewriting techniques [Gaasterland et al. 1992] are important to personalize the query, content-based [Balabanović 1997] and collaborative filtering algorithms [Billsus and Pazzani 1999] can help at the query results level.

2.1.1 *Scenario.* We will use a simple scenario to exemplify the main concepts of our framework. Let us consider Alice, using our Human Capital Development Suite (HCD) online (depicted in figure 1) to specify her learning goals.[1]. HCD Suite connects to the HCD Online Learning Network via the Simple Query Interface (SQI) [CEN/ISSS 2005]. It employs RDF bindings for the common schema, user queries against the schema are expressed in Datalog [Nejdl et al. 2002], mappings are provided to convert from the common to local schemas [Miklós and Sobernig 2005]. Through HCD Online, the same query is sent to all knowledge repositories, with possibly some notational variations.

---

[1]HCD Suite Online is one of the two instantiations of SS4L framework, the second one is our Personal Learning Assistant [Dolog et al. 2004]
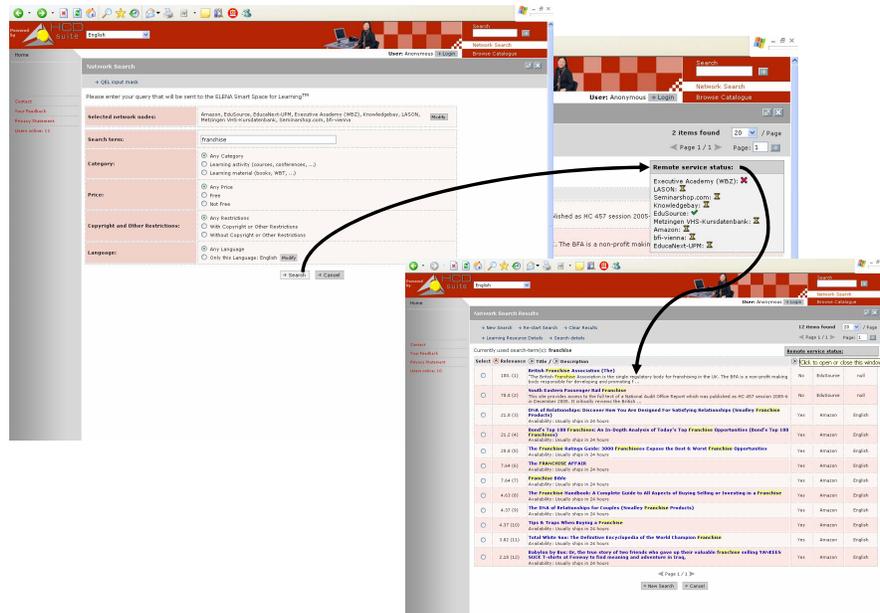
Fig. 1.   Accessing the Learning Network HCD Online

Alice is a computer professional in a small and medium sized enterprise. She has a diploma degree in computer science and has been working mostly as a programmer in the last couple of years. She speaks English and German.

In a few weeks Alice's group will start a 3-month project to build an e-banking application for a local bank. Alice's task is to build a module that will be used for electronically signing orders and contracts between the bank and its customers, and securely archiving the signed contracts.

Since Alice has only basic knowledge on information security, her learning goal is to improve her knowledge on advanced electronic signatures and qualified digital certificates, as well as on legislative requirements imposed on legally valid electronic signatures and long-term secure archives.

Alice uses HCD Suite Online to help finding learning materials and learning activities that contribute most to her learning goals. As Alice's company does not have internal courses on this topic, HCD sends a query to its connected educational network. The query is rewritten in a way that takes into account Alice's knowledge of languages and time constraints for potential learning activities.

The query returns several relevant results from different learning repositories. Before presenting the results to Alice, HCD ranks them displaying the most relevant ones first, taking into account the information on her learning goals and past learning performance. Since Alice's personal training budget for year 2007 has been restricted, prices of learning resources also represent an important factor.
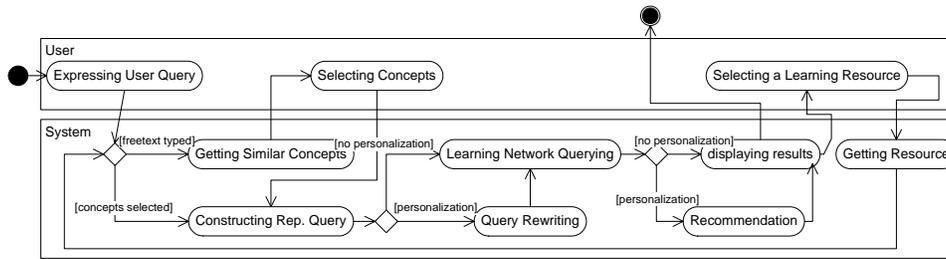
Fig. 2. Personalized search workflow from user's and personalized search system's point of view
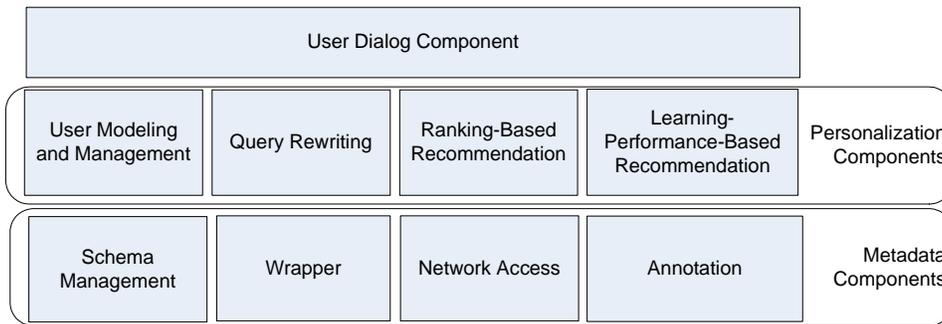

Fig. 3. Functional Components for Personalized Learning Networks

2.1.2 *Scenario Activities.* Figure 2 depicts a basic set of activities for similar scenario as the one described above as a workflow. A user starts with entering a query either as a set of free text keywords or using concepts selected from an ontology. Depending on whether free text or a set of concepts is used, the system matches the free text with the ontological graph and suggests the most similar concepts or the system directly constructs a query from the selected concepts. In the first case, the user gets a list of similar concepts she is supposed to select from. After she selects her concepts as personal learning goals, the system continues with query construction. If the user opts for personalized search, the system looks for additional learning goals, concept, and language preferences to rewrite the initial query, and then sends the query to the learning network. After the results of the query return, the system either displays them or applies additional personalization steps. Depending on the implemented personalization strategy, the results are examined for prerequisites which are compared to the user's learning history such as her learning performance. The ranking-based recommendation utilizes document retrieval and classification techniques when comparing them with learning goals, learner's history and learner's competencies. The user then selects the appropriate learning resource and books it.

2.2 Functional Components

Figure 3 depicts a layered architecture with two groups of components: *Personalization* and *Metadata*. The *User Dialog Component* is used to provide the interface

for user interactions. Personalization and Metadata components are connected in a sequence, initiated by the user dialog.

2.2.1 *Personalization Components.* Personalization components provide reasoning capabilities to recommend or restrict a set of learning resources suitable to reach particular learning goals given a learner's characteristics. The *User Modeling and Management* component is responsible for collecting and identifying features of a learner relevant for personalization and for continuous evolution and update of user profiles based on user interaction through the *User Dialog*. The component can request a query through the Network Access component to collect distributed fragments of learner data. The *Query Rewriting* component is responsible for query refinement or relaxation when a user has initiated a query for learning resources. Learner preferences (language, device, content) are considered to restrict a user query which originally just contains the learning goal. The *Ranking-Based Recommendation* component employs a content-based technique to rank the resources in a result set, e.g. closeness to a user's learning goal, before results are displayed via the User Dialog. Another ranking-based recommendation component takes advantage of collaborative annotations created by learners who used the resource previously. The *Learning-Performance-Based Recommendation* component performs recommendation on the result set. It traverses prerequisites of learning resources and annotates them with recommendations based on the matches between prerequisites, learner's competence and previous learning performance.

We employ a strategy design pattern [Gamma et al. 1995] to connect and implement different personalization strategies in our network. Strategies overload two functions of the generic personalization interface: *QueryRewriting*, and *Recommendation*.

2.2.2 *Metadata Components.* Personalization functions are based on learning resource metadata and learner metadata fragments. All knowledge repositories joining the network must be compatible with the common metadata infrastructure. Metadata components provide access and manipulation capabilities on metadata, towards a common infrastructure. Initiated by the Query Rewriting component, the *Network Access* component provides an interface to query distributed metadata of distributed knowledge repositories as well as learner model fragment servers. It takes advantage of the *Wrapper* component, which provides means for translating between the schema of the Learning Network and the local schemas of the various knowledge repositories. A Wrapper component can be deployed at the search client or at the search target. The *Schema Management* component is capable of registering and identifying schemas and vocabularies used to describe learning resources and learners. It supports the User Dialog as well as the Wrapper service by managing topic ontologies for learning resources or competence ontologies for learner profiles. It also provides functionality to identify an ontology used for values in metadata elements. This is especially useful when it comes to managing topic and competence ontologies. In addition, such a component provides operations to identify subparts of these ontologies based on user requests through an exact query or similarity measures. The *Annotation* component can further extend learning resource metadata if a new schema is registered or a new topic or competence on-

tology is provided. It makes these annotations available in the Learning Network. The component is triggered via the User Dialog or via personalization components.

3. INSTANTIATIONS OF PERSONALIZATION COMPONENTS

In a Learning Network personalization components build a temporal knowledge base of *facts about learning resources and learners* when they are invoked. In our network personalization is implemented as a Web-based service that wraps a personalization strategy or a user modeling strategy. Facts are initiated according to the *input parameters* sent to the Web service. This includes for example learning performance or learning goals as parameters. We provide several techniques to collect information about a user such as previous performance. The information can be derived from questionnaires the user has filled in and evaluations performed within our network. It can also be gathered from other learning providers connected to our network by querying them as for learning resources. Personalization services in our network operate on metadata and extend them with additional facts. This avoids the usual cold start problem if a user is completely new and not known by the system.

3.1   Query Rewriting

Based on an analysis of learner profiles, the Query Rewriting Service extends a user's query by additional restrictions, joins, and variables. These modifications are performed based on predefined heuristic rules and functions [Dolog et al. 2004].

Query Rewriting Services can add additional constraints to user queries based on user preferences and language capabilities. They can also extend a user query based on previous learner performance maintained in the learner profile. They can also rewrite a user query based on information the connected services need, which can be exposed as DAML-S based service profile descriptions.

Inferences take place in the phase of query construction. We have employed a Datalog query model, where the query consists of query literals. Rewriting is based on adding or modifying those literals. Initial query restrictions are constructed either from user selected concepts or from metadata about a presented resource which are passed to the service as input parameters. The result variables are taken from user preferences about the environment. After a formal query is constructed from the concepts initially chosen by a user, additional restrictions are added based on knowledge about the user. This includes user preferences and possibly other information from the user's profile.

The algorithm for query rewriting uses parameterized breadth-first search over language and learning concept preferences. Parameterization sets the distance in the preference tree from the most related concepts to the ones contained in the user query. Selected preferences are transformed to the query restriction literals and added to the original query. These can be added as conjunctions, disjunctions, and consider outer-joins when the resource metadata graph is composed from several nodes.

Figure 4 exemplifies the query construction and rewriting process. Shaded ellipses represent instances of the formal knowledge model and non shaded ellipses represent concepts/classes in our knowledge model. Dashed arrows are added dynamically by query rewriting.
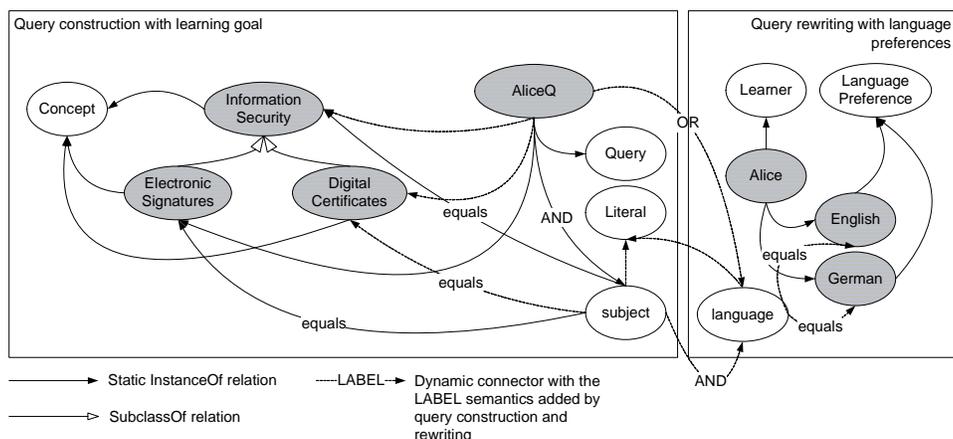
Fig. 4. Example for query construction and rewriting with a learning goals as conceptual graphs and language preferences

Let us recall our initial scenario. Alice formulated her learning goal using *security* as keyword. The keyword has been matched with the *information security* concept and its subconcepts *electronic signatures and digital certificates* (see left part of the figure). These are added as restriction query literals into Alice' query and are added conjunctively.

The Query rewriting algorithm (see right sight of the figure) also checks Alice's language preferences. In our example, we just use a flat model of two language preferences, though these can get more complex especially for preferences. The language preferences are connected disjunctively, and conjuctively to the original query.

3.2   Learning-Performance-Based Recommendation

Learning-Performance-Based Recommendation looks for performance records of a learner. A performance record holds information about previous learning experiences, competencies acquired, certificates gained, metrics about the level of job performance, as well as indicators of the knowledge level in a certain domain. Performance records are matched against prerequisite relations in learning resource metadata. The level of matching between prerequisite and performance records determines the recommendations.

The component checks each learning resource in the query results, and compares its prerequisite relations with the performance records of the user. If a mach has been found, the learning resource is considered for recommendation. The recommendation algorithm is parameterized, and can take the transitive closure of the prerequisite relation into account. Distance from the original resource can serve as a means to optimize the process as performance can degrade significantly if the resource graph induced by the prerequisite relations is large. The recommendation information can be parameterized as well to distinguish between binary recommended/non-recommended information and several levels of recommendations. One possibility is to have a strong restriction on matching and to request all
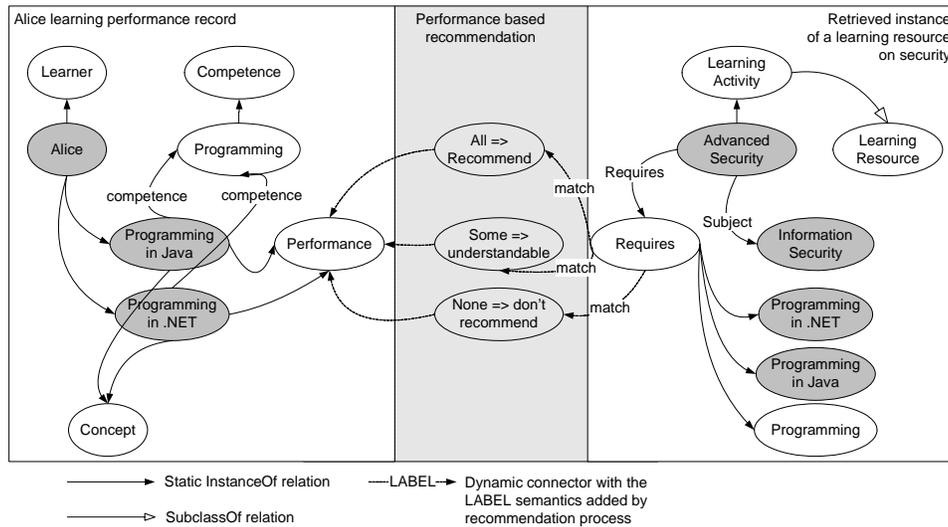
Fig. 5. Example for learning performance based recommendation annotation of retrieved learning resource/learning activity

prerequisites to be matched with the learning performance. Another possibility is to consider the number of occurrences in the user profile.

Figure 5 exemplifies the learning performance based recommendation process on learning resources retrieved as query results. The left side of the figure depicts the conceptual model for Alice's learning performance in terms of programming in Java and .NET courses. These courses extended Alice's knowledge. The corresponding concepts also contribute to the competencies Alice has in programming as such. The right side of the figure depicts a learning resource on advanced security with a subject referring to the selected learning goal in the query. The learning activity is connected to its prerequisites which represent the background knowledge needed to fully understand all topics. The middle part of the figure refers to a partial schematic view on heuristics which have been selected to annotate the learning resources. In this example, all prerequisites of the activity match Alice's performance record, therefore the learning activity will be annotated as recommended.

3.3 Ranking-Based Recommendation

Two kinds of ranking-based recommendations are included in our SS4L framework: Metadata-Based Recommendation and Collaborative Recommendation.

The first one, ***Metadata-based Ranking*** of learning resources, is based on two filters: a text filter and a category filter. We use Lucene [Hatcher and Gospodnetic 2004] as text filter. Learning resource metadata received as results to a query are indexed and learner features retrieved from the learner profile (learning goals, interests, and learning performance) are tokenized. Tokenized terms are then used to query the index and to calculate matching scores.

The Category Filter can be based on the distance between the concept ontology entries selected by a user and those referenced in the retrieved learning re-
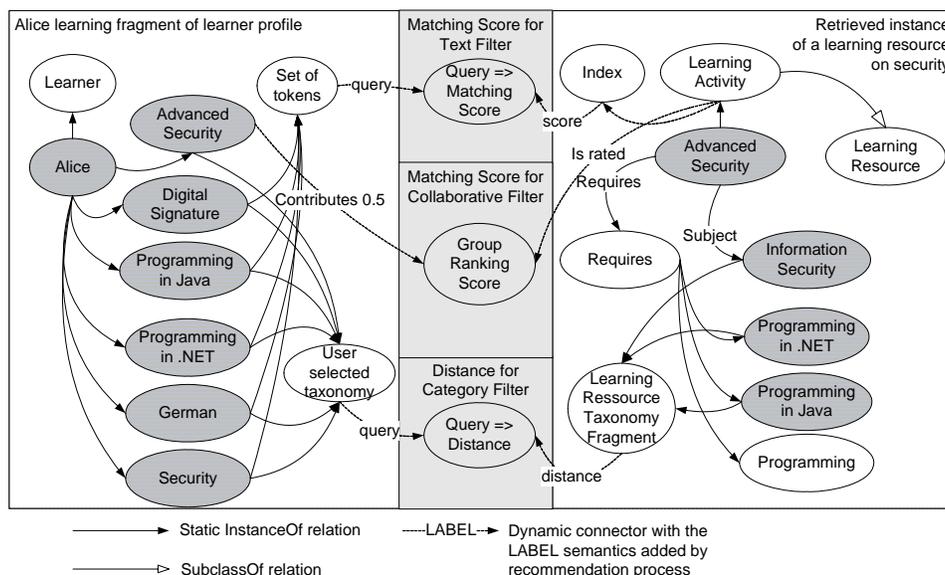
Fig. 6. Example for ranking based recommendation annotation of retrieved learning resource/learning activity

sources. We consider three types of matching: exact match, generalization, and specialization. The category filter is based on distances and weights between user-selected topics and subjects of learning resources. Various techniques from document classification and information retrieval with ontologies and taxonomies can be implemented. Weighting factors from both filters can be used to rank resources separately, or a single weight can be calculated from both.

Collaborative-Filtering-Based Ranking is calculated as an average of user ratings. Usually, we will only consider similar users in this computation [Ardissono et al. 2005].

Figure 6 exemplifies the ranking based recommendation process on learning resources. The left side of the figure depicts a fragment of the conceptual model for Alice's (tokenized) learning profile. The right side of the figure depicts a learning resource on advanced security with a subject referring to the original learning goal. The resource is indexed, the tokens from Alice's learner profile are used as queries over the index. The queries return a matching score for all resources in the query results including the one from figure 6. The score is used to rank the learning resources. Furthermore, the learning resource on security is also indexed by the subject taxonomy, and the learner profile contains pointers to concepts from the same taxonomy. However, the subparts of the taxonomies induced by the learner profile and learning resource differ from one another, so the distance between them serves as another score to order the resources. Collaborative ranking can provide an additional ranking score.

3.4 User Modeling and Management

The User Modeling and Management Service stores, retrieves and updates relevant user data in cooperation with the Learning Resource Access Service. This service is derived from an ontology integrating several specifications and standards for learner profiles [Dolog and Schäfer 2005]. The service provides manipulation and retrieval functions at the learner profile fragment level.

The following algorithm is applied when the service searches for a relevant fragment of learner performance:

—Retrieve all instances of the Identification concept for the current user;
—Search instances of the Learner concept on systems referenced in each identification entry;
—If there are further systems referenced in the identification records at the remote systems, reapply this algorithm with the records;
—Retrieve all objects as instances of concepts needed for adaptation (e.g. learner performance);

Learner model retrieval is split into two phases. The first phase is a simple retrieval of learning providers with a learner identification pair. This is applied recursively for all providers identified in the learner profile. The algorithm can be seen as a adapted breadth-first search over a structure induced by the provider neighbour relationships. First, all neighbour provider instances are retrieved at the current provider and a user model query is send to them. Together with a user model query, neighbor provider retrieval is applied at all providers where the user model query was sent to.

In the second phase, the user model query is applied on each registered provider with a partial user model. Depending on the user model component, different algorithms are applied. A breadth first search algorithm is applied for the preference models in a user profile as described in the query rewriting component section. Performance records are retrieved by simple record scan. Additional reasoning may be applied on the competence graphs constructed from the learner performance records and overlaid on the global competence graph.

With an assessment component in our network we try to overcome the cold start problem encountered in pure performance-based recommendation systems, i.e. we try to get at least an initial overview over learner competences when the user enters our network as a novice. Importing a learner's history, e.g. via an ePortfolio record, constitutes another strategy that could be applied.

4. KNOWLEDGE REPRESENTATION IN LEARNING NETWORKS

In a learning network knowledge repositories that use different schemas for describing educational artefacts need to communicate with each other. To ensure correct interpretation, a common (semantic metadata) schema - also referred to as ontology - is required. This ontology specifies the properties of the educational artefacts accessible within the repository [Wiederhold et al. 1992]. Each declaration of a property of an educational artefact constitutes an ontological commitment to use the defined term in interactions. The sum of all ontological commitments is reflected in the common metadata schema.

4.1   Methodology

We have initiated a consensus finding process from two perspectives: *provider*, and *user*. We involved learning resource providers who were interested in joining our Learning Network, and carried out an extensive user study aimed at getting a better understanding of usage scenarios, derived from qualitative interviews. In order to document our findings, activity descriptions were used. An activity was regarded as a sequence of events and actions performed by a person using the Learning Network. It constitutes work, task or process being performed.

We then analyzed the scenarios with respect to the concepts used in the scenario descriptions. We compared identified concepts with concepts from the standards mentioned earlier. Similarly, we analyzed the situation at the provider sites and mapped attributes and concepts from their knowledge repositories and metadata to a common schema.

When instantiating it in the Learning Network, the common schema serves as a point of reference for each node connecting. For each schema joining the Learning Network a wrapper needs to be implemented either at the a) search client, b) the search target (provider), or c) at a mediator. The node(s) responsible for translating between local and common schema require a Schema Management component where new schemas can be registered.

4.2   Standards and Open Specifications

In the last couple of years institutions and associations such as ISO, IEEE, IMS, ADL, CEN/ISSS, or the HR-XML consortium have worked on an increasing number of standards and specifications that can be used as the basis for knowledge representation in Learning Networks. In this subsection we briefly describe and discuss these standards.

For Web-based learning we differentiate between application-oriented and schema-oriented standards and specifications. The former support well-defined application interfaces exposed by systems, in order to support the integration of systems. The schema-oriented strategy support standardized, semantically-rich data schemas shared by various systems. The later are in the focus of the forthcoming subsection.

4.2.1   *Learning Resources and Activities.* Among the most frequently used, the Dublin Core Metadata Schema (DC) [Association] defines a set of 15 core elements from the ISO/IEC 11179 standard, including Title, Description, Identifier, and Language. The Learning Objects Metadata Standard (LOM) [IEEE-LTSC] developed and maintained by IEEE's Learning Technology Standards Committee (LTSC) provides metadata for a learning object, defined as any entity, digital or non-digital that may be used for learning, education or training. The LOM Schema includes 70 elements divided into 9 categories: general, life cycle, meta-metadata, technical, educational, rights, relation, classification, annotation. Application profiles and adaptations of LOM are allowed and necessary [Quemada and Simon 2003]. Open Qualifications (OpenQ) is a metadata specification specifically designed for exchanging course descriptions [Rex and Hettrich 2002] addressing aspects such as pricing, multimedia, and supplier information.

4.2.2 *Learners.* Learner profile specifications have been introduced to support the exchange of learner information between educational systems. IEEE Personal and Private Information for Learner (IEEE PAPI) [IEEE 2000] distinguishes personal, relations, security, preference, performance, and portfolio information about a learner. The IMS Learner Information Package (IMS LIP) [IMS] provides structures for storing identification, qualifications, accessibility, activities, competencies, goals, interests, transcripts, affiliations, security keys, and relationships. Further, the IMS Reusable Definition of Competency and Educational Objective (RD-CEO) [IMS 2002] provides structures for competency models.

4.2.3 *Vocabularies.* We have also employed subject ontologies for referencing particular values in a concept slot. Subject ontologies are formalized vocabularies with a structure of terms which are used as ranges for particular slots of concepts in the learning service ontology. An example of such a subject ontology is the ACM Computing Classification System (CCS) [ACM 2002]. ACM CCS serves as a classification system for computer science literature in the ACM Digital Library. Similar subject ontologies exist for Software Engineering, Mathematics or General Engineering disciplines. Other taxonomies for dates, person roles, department titles, environments, country codes can also be used as vocabularies in a common schema.

4.3 Learning Resource Descriptions based on Standards and Specifications

The primary purpose of the common schema used for describing learning resources is to effectively support searching and selecting learning resources. Hence, a schema of a Learning Network supporting federated search requires: a) elements that can be used to narrow the search, b) elements for the results list that have to be returned in order to serve particular usage scenarios and c) optional elements that can be used to provide a more detailed view on the result list or optional filtering later on.

The schema we propose for corporate learning networks is realized as an application profile which enriches the element set proposed by IEEE LOM by mixing in other metadata standards and specifications such as OpenQ. Other standards and specifications re-used in the common schema are DC, VCard, and iCalendar. The rational behind developing a LOM application profile is to design a schema according to the Learning Network's requirements while preserving LOM compatibility.

Query-restricting attributes are attributes that each knowledge repository needs to fully understand. Here a semantic agreement of the permissible values (vocabularies) had to be achieved. Mandatory attributes of the results list must be returned for each learning resource that matches a query. Mandatory attributes are attributes all knowledge repositories need to return. The restrictions on permissible values need to be considered when returning values for the attribute list. Optional result attributes are additional attributes a knowledge repository can return.

4.4 Learner Descriptions based on Standards and Specifications

To enable the exchange of learner profile fragments between web-based learning systems, we need to provide explicit information about what is exchanged, which values of the specific subject are considered and how the information is bound to a learner. Learner profile standards and specifications provide us with a represen-
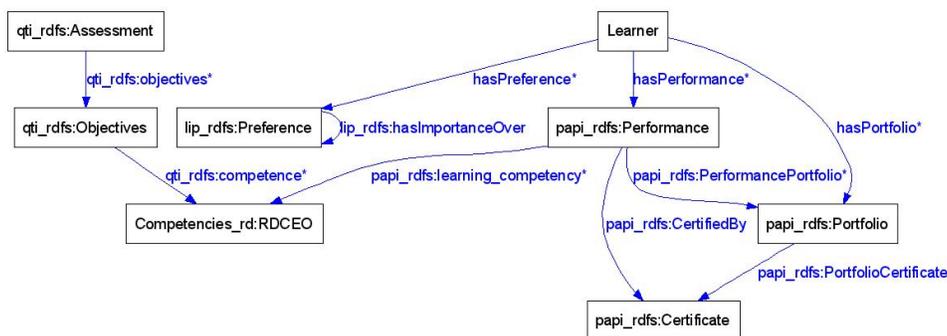
Fig. 7.   An Excerpt of an Ontology for a Learner Profile [Dolog and Schäfer 2005]

tation for subjects of exchange, e.g. learner performance, portfolio, preferences, learning style, certificates, evaluations, and assessment. Domain ontologies provide us with exchangeable/sharable models of domains. Such ontologies can model either the domain which will be overlaid in the learner profile, learner competencies/skills, or can model stereotype structures.

Figure 7 depicts an excerpt of a learner profile ontology configured from fragments based on three specifications. The ontology excerpt describes a situation where learning performance of a student is exchanged as competency records. The competencies are evaluated by learner assessment (e.g. tests) and are derived from learning objectives of tests. Other educational activities, further materials, and projects created within the activities are reported within the portfolio as well. Additional information reported under preferences comprises language, device, resource and learning style preferences.

5.   EVALUATION

For evaluation, the hypotheses we wanted to test were that personalized search for learning resources, implemented as part of HCD Online, enables learners to find relevant resources in the network of knowledge repositories with different local metadata schemas and that information about user goals stored in a learner profile improves ranking of search results.

In the information retrieval literature, numerous evaluation metrics have been proposed [Ribeiro-Neto and Baeza-Yates 1999], the most prominent ones being the precision and recall metrics. Precision is calculated as the fraction of relevant resources among the resources retrieved, while recall measures what percentage of all relevant resources are retrieved. A precision value is easy to calculate, given that a user or an assessor can judge for each result if it is relevant or non-relevant for the query. Calculating recall is more difficult as we need to know for each learning resource in the network whether it is relevant for the query or not. Unfortunately, in a large network, this is virtually impossible.

One of the measures that supplements precision and recall is the ranked half-life (RHL) indicator [Blomgren et al. 2004]. RHL shows the ability of the system to place relevant results high in the ranked list of retrieved results. The lower the RHL value, the higher the relevant LRs are placed in the ranked output. The RHL

indicator thus gives additional information about the degree to which the search engine is capable of ranking its output according to user-related relevance. In order to make the RHL value more comparable, it is normalized into the $RHL\ Index$ for a given cut-off value, i.e. a $RHL\ Index$ value equals the computed RHL value normalized for the corresponding precision value (precision = 1.0).

The RHL value is computed as follows:

$$RHL = L_m + \frac{n/2 - \sum f^2}{F(med)} CI \tag{1}$$

where L_m is lower real limit of the median class, i.e. the rank position of the lowest positional information objects above the median class; n number of observations, i.e. the total sum of the assigned relevance values; $\sum f^2$ cumulative frequency (relevance values) up to and including the class preceding the median class; $F(med)$ the frequency (relevance value) of the median class; and CI a class interval, which is usually equal to 1.

In our evaluation, we investigated how the $RHL\ Index$ value changed after the Ranking-based Recommendation Service was invoked. Ranking was based on information about the learner's goals and learning resources metadata.

5.1　Study

Evaluation tests were performed in spring and autumn 2006. A total of 3 assessors from Austria and Slovenia were involved. They were faculty members of academic institutions and possessed expertise and experience in search engines and Web-based learning.

The users were given a simulated work task situation, which served as the trigger of the user's information need and as the background for the relevance judgments. They were first asked to imagine working in a company in a particular position, and to think of the learning goals they might have in this situation. The users were then required to enter information about their goals, each captioned with a title and elaborated with a free-text description.

Users were required to carry out search tasks to find learning resources that would contribute to each of their self-defined goals. For each search, the user entered one or more search keywords and refined the search by defining the values of several attributes: category (any, learning material (LM) or learning activity (LA)), price (free or not), copyright and other restrictions (with or without), and language (any or a specific one). There was no time constraint imposed on the tasks.

Search queries for learning resources were sent to a network of knowledge repositories with different local metadata schemas. A list of items, each of them consisting of a title and a snippet (up to three lines of text) of an LR description, was returned as a search result. The number of search results from Amazon was limited to 10. Each item was hyperlinked to a site where more metadata of the LR were presented. The users had no access to the complete LRs in the knowledge repositories.

By default, ranking of retrieved results was calculated without taking into account user's personal information. The user looked at the 15 top ranked LRs, and for each of these LRs judged whether it was relevant (1.0), partially relevant (0.5) or not relevant at all (0.0) for the query. Then the user activated the personalization feature and repeated the search and results relevance assessments. For each task,

	default ranking	goal-based personalization
average $precision_{15}$	0.539	0.555
average $RHL\ Index$	16.793	14.610
median $RHL\ Index$	11.951	11.518

Table I.   Statistics for the search task results

two sets of outcomes were obtained: one when no personalization was used and another one when learning goals were considered for ranking-based recommendation.

5.2   Results

The users performed relevance assessments on 36 tasks. The results were used to calculate the $precision_{15}$ value (percentage of relevant results among the top 15 ranked results) and the $RHL\ Index$ value. All values were obtained for non-personalized and personalized search tasks. The results are shown in Table I.

As can be seen from the results the average and median values of a $RHL\ Index$ improved after personal goals were taken into account when calculating results' rankings. Out of 36 search tasks, ranking improved 14 times and became worse 4 times. We observed several general factors that influence the rankings of the retrieved learning resources:

(1) Rankings improve if a user provides a general search keyword and the goal description significantly narrows down the result. For example, if the user wants to learn about risk management and searches for LRs about management, after personalization LRs about risk management are ranked higher.

(2) The results can become worse when a user has several heterogeneous goals, and information about other goals affects rankings for a particular goal. This is mainly a user interface issue of our current implementation, as the user does not have an option to specify to which goals he would like to apply personalization.

In the future, we want to further investigate in which situations personalization is useful, and what kind of information (content, form) is helpful in our learner profiles. To allow assessing the effectiveness of the chosen algorithm in comparison to other existing algorithms (especially non-semantic retrieval algorithms), it might be useful to create a marked-up test corpus including pre-formulated queries and corresponding relevance judgements. The currently offered collections in TREC (Text REtrieval Conference) and CLEF (Cross-Language Evaluation Forum) do not have the semantic mark-up needed to evaluate the strength of the common schema and the connected retrieval algorithms.

6.   RELATED WORK

Adaptive and personalized Web-based learning applications have been studied primarily for closed environments. First steps towards open adaptive Web-based learning solutions have been investigated in [Henze and Nejdl 2002; Dolog et al. 2003]. In this paper we extend this work by moving towards more decentralized solutions of learning networks where both resources and computation can be distributed. Besides personalization services we introduce metadata services which are important to realize personalized access in distributed learning networks.

Similar to our approach, [Dagger et al. 2003] builds on separating learning resources from sequencing logic and additional models for adaptivity: Adaptivity blocks in the learning object metadata and in various other models like the narrative model, candidate groups, etc. define the kind of adaptivity realizable with the current piece of learning content. The main driving force in these models are the candidate groups that define how to teach a certain learning concept. A rule engine selects the best candidates for each specific user in a given context. A shortcoming of the approach is that adaptivity requirements are considered only in the adaptivity blocks, while our approach considers all metadata as useful for adaptation. In addition, we consider a combination of rule-based and ranking-based recommendation techniques.

A similar framework is proposed in [Lee et al.], where probabilistic semantic inference for query keywords, LOM-based user preference logging, and other users feedback are used to retrieve suitable learning objects from a single repository. Though the functional architecture is similar, we employ different algorithms. We use a different strategy to reason about users' intentions. We use an n-gram algorithm [Kim and Shawe-Taylor 1994] to find similar concepts in a domain ontology, and the user then selects from recommended concepts, to representuser intention. Furthermore, we consider preferences already in the query expansion process. For our recommendations we utilize the user's learning performance represented as a set of achieved competencies. In addition, we consider similarity functions between query results and learning goals, preferences and history.

An early approach for defining an architecture for personalization and adaptivity in the Semantic Web has been proposed in [Aragao et al. 2001]. This approach is characterized by the transfer of ownership of Semantic Web resources to the user, and therefore done on the client side. The architecture introduced considers Semantic Web applications as isolated islands, while our approach extends it with additional services needed to achieve personalization functionality in distributed environment such as the Schema Management Service, the User Modeling and Management Service and the Network Access Service. Those services ensure that we can retrieve relevant learning resources and learner fragments from distributed metadata bases and process them in connection with personalization services.

Comparing our work with standard models for adaptive hypermedia systems such as the one used in AHA! [Bra et al. 1999], we observe that they define several models like conceptual, navigational, adaptational, teacher and learner models. Compared to our approach, these models either correspond to ontologies / taxonomies, to different schemas describing teacher and learner profile, and to schemas describing the navigational structure of a course. We express adaptation functionalities as encapsulated and reusable Datalog-based rules, while the adaptation model in AHA uses a rule based language encoded into XML. At the level of concept or information items AHA! provides functionalities to describe requirements [Bra et al. 2002] for the resource, which declare the requirements on the user to visit that information.

This paper builds on our previous partial results described in [Dolog et al. 2004; Simon et al. 2004; Simon et al. 2006; Nejdl et al. 2002; Dolog and Schäfer 2005]. It provides a unified view on these results and a new functional framework which integrates rule-based personalization techniques on query and results with ranking

based recommendation techniques on query results, not published before. Furthermore, the paper provides an evaluation of our approach and shows that personalization in distributed learning network can be realized with promising results.

7. CONCLUSIONS AND FURTHER WORK

We have presented a new and effective approach for personalized access to learning resources and activities in distributed heterogeneous learning networks. We have described the SS4L framework and a corresponding infrastructure which enables flexible chaining of services as needed according to the requirements of particular communities, users and customers. Personalization functions are modularized and provided in two groups: query rewriting and personalization on query results. Rule-based techniques are supported by formal ontologies for learning resources, vocabularies and learners we have developed. Ranking is supported based on identifying a minimal set of predicates in query results. Our evaluation showed the approach to be feasible, and able to improve search quality through personalization.

Distributed environments and their heterogeneity need more computational power and additional services with some communication overhead. This is balanced, however, by the added value of these services providing access to a large pool of learning resources. Personalization services enable the user to choose proper learning resources more easily in these environments.

In addition to the user services introduced in this article, additional services such as learning path generation, course guidance or collaborative learning support are possible, providing even more interactive learning resources. The learning resource ontology we have developed is prepared for that but new services have to be implemented to fully enable these functionalities. This includes new recommendation algorithms on more complex structures and additional algorithms for chaining, composing and configuring services for particular learning tasks.

8. ACKNOWLEDGEMENTS

REFERENCES

ACM. 2002. The acm computer classification system. http://www.acm.org/class/1998/.

ARAGAO, V., FERNANDES, A., AND C.A.GOBLE. 2001. Towards an architecture for personalization and adaptivity in the semantic web. In *Proc. 3rd. Intl. Conf. on Information Integration and Web-Based Applications and Service, iiWAS 2001*. Linz, Austria.

ARDISSONO, L., GOY, A., PETRONE, G., AND SEGNAN, M. 2005. A multi-agent infrastructure for developing personalized web-based systems. *ACM Trans. Inter. Tech. 5,* 1, 47–69.

ASSOCIATION, D. C. The dublin core metadata initiative. http://dublincore.org/.

BALABANOVIĆ, M. 1997. An adaptive web page recomendation service. In *Proc. of the First International Conference on Autonomous Agents*. Marina del Rey, USA.

BILLSUS, D. AND PAZZANI, M. 1999. A personal news agent that talks, learns and explains. In *Proceedings of the 3rd International Conference on Autonomous Agent (Agents 99)*. Seattle, Wash., 268–275.

BLOMGREN, L., VALLO, H., AND BYSTRM, K. 2004. Evaluation of an information system in an information seeking process. In *Proceedings of ECDL 2004*. 57–68.

BRA, P. D., AERTS, A., SMITS, D., AND STASH, N. 2002. AHA! version 2.0. In *Proceedings of the AACE ELearn'2002 conference*. 240–246.

BRA, P. D. AND CALVI, L. 1998. AHA!: An open adaptive hypermedia architecture. *The New Review of Hypermedia and Multimedia 4*, 115–139.

BRA, P. D., HOUBEN, G.-J., AND WU, H. 1999. AHAM: A dexter-based reference model for adaptive hypermedia. In *Proc. of ACM Conference on Hypertext and Hypermedia*, K. Tochtermann, J. Westbomke, U. Wiil, and J. Leggett, Eds. Darmstadt, Germany, 147–156.

BRUSILOVSKY, P. 2001. Adaptive hypermedia. *User Modeling and User-Adapted Interaction 11,* 1-2, 87–100.

BRUSILOVSKY, P., EKLUND, J., AND SCHWARZ, E. 1998. Web-based education for all: A tool for developing adaptive courseware. *Computer Networks and ISDN Systems (Proceedings of Seventh International World Wide Web Conference) 30,* 1–7 (Apr.), 291–300.

CEN/ISSS. 2005. *A Simple Query Interface Specification for Learning Repositories (CEN Workshop Agreement #15454)*. ftp://ftp.cenorm.be/PUBLIC/CWAs/e-Europe/WS- LT/cwa15454-00-2005-Nov.pdf, Brussels, Belgium.

DAGGER, D., WADE, V., AND CONLAN, O. 2003. Towards "anytime, anywhere" learning: The role and realization of dynamic terminal personalization in adaptive elearning. In *Ed-Media 2003, World Conference on Educational Multimedia, Hypermedia and Telecommunications*. Hawaii.

DOLOG, P., GAVRILOAIE, R., NEJDL, W., AND BRASE, J. 2003. Integrating adaptive hypermedia techniques and open rdf-based environments. In *Proc. of 12th International World Wide Web Conference*. Budapest, Hungary.

DOLOG, P., HENZE, N., NEJDL, W., AND SINTEK, M. 2004. Personalization in distributed e-learning environments. In *Proc. of WWW2004 — The Thirteen International World Wide Web Conference*. ACM Press, New York.

DOLOG, P. AND SCHÄFER, M. 2005. A framework for browsing, manipulating and maintaining interoperable learner profiles. In *Proc. User Modeling 2005: 10th International Conference, UM 2005*, L. Ardissono, P. Brna, and A. Mitrović, Eds. LNAI, vol. 3538. Springer, Edinburgh, Scotland, UK.

GAASTERLAND, T., GODFREY, P., AND MINKER, J. 1992. An overview of cooperative answering. *Journal of Intelligent Information Systems 1,* 2, 123–157.

GAMMA, E., HELM, R., JOHNSON, R., AND VLISSIDES, J. 1995. *Design Patterns. Elements for Reusable Object-Oriented Software*. Addison Wesley.

HARASIM, L., HILTZ, S. R., TELES, L., AND TUROFF, M. 1995. *Learning Networks*. The MIT Press, Cambridge, London.

HATCHER, E. AND GOSPODNETIC, O. 2004. *Lucene in Action*. Manning.

HENZE, N. AND NEJDL, W. 2002. Knowledge modeling for open adaptive hypermedia. In *Proccedings of the 2nd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH 2002)*. Malaga, Spain.

IEEE. 2000. IEEE P1484.2/D7, 2000-11-28. draft standard for learning technology. public and private information (papi) for learners (papi learner). Available at: `http://ltsc.ieee.org/archive/harvested-2003-10/working\_groups/wg2.zip`. Accessed on December 20, 2003.

IEEE-LTSC. Ieee lom working draft 6.1. Available at: http://ltsc.ieee.org/wg12/index.html. Accessed on October 25, 2002.

IMS. IMS learner information package specification. Available at: http://www.imsproject.org/profiles/index.cfm. Accessed on May 25, 2006.

IMS. 2002. Ims reusable definition of competency or educational objective - information model. http://www.imsglobal.org/competencies/rdceov1p0/imsrdceo _infov1p0.html.

KIM, J. Y. AND SHAWE-TAYLOR, J. 1994. Fast string matching using an n-gram algorithm. *Software: Practice and Experience 24,* 1, 79–88.

Lee, M.-C., Tsai, K. H., Ye, D. Y., and Wang, T. I. A service-based framework for personalized learning objects retrieval and recommendation. In *Advances in Web Based Learning - ICWL 2006, 5th International Conference, Penang, Malaysia, July 19-21, 2006*.

Miklós, Z. and Sobernig, S. 2005. Query translation between rdf and xml - a case study in the educational domain. In *WWW Workshop on Interoperability of Web-Based Educational Systems*, D. Olmedilla, N. Saito, and B. Simon, Eds. CEUR, Chiba, Japan, 27–35.

Nejdl, W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., Nilsson, M., Palmér, M., and Risch, T. 2002. EDUTELLA: a P2P Networking Infrastructure based on RDF. In *In Proc. of 11th World Wide Web Conference*. Hawaii, USA.

Quemada, J. and Simon, B. 2003. A use-case based model for learning resources in educational mediators. *Educational Technology & Society 6,* 4, 149–163.

Rex, S. and Hettrich, A. 2002. Spezifikation open-qcat. version 0.9.5 - public draft.

Ribeiro-Neto, B. and Baeza-Yates, R. 1999. *Modern Information Retrieval*. NY: ACM Press.

Simon, B., Dolog, P., Miklós, Z., Olmedilla, D., and Sintek, M. 2004. Conceptualising smart spaces for learning. *Journal of Interactive Media in Education. Special Issue on Educational Semantic Web* 9 (May), 221–239.

Simon, B., Sobernig, S., Wild, F., Aguirre, S., Brantner, S., Dolog, P., Neumann, G., Huber, G., Klobucar, T., Markus, S., Miklós, Z., Nejdl, W., Olmedilla, D., Salvachua, J., Sintek, M., and Zillinger, T. 2006. Building blocks for a smart space for learning. In *ICALT'2006: The 6th IEEE International Conference on Advanced Learning Technologies*. IEEE Press, Kerkrade, The Netherlands, 309–313.

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, 351–384.

Wiederhold, G., Wegner, P., and Ceri, S. 1992. Toward megaprogramming. *Communications of the ACM 35,* 11, 89–99.

...