Ontology-based User Modeling for Web-based Information Systems

Anton Andrejko, Michal Barla and Mária Bieliková

Institute of Informatics and Software Engineering,
Faculty of Informatics and Information Technologies,
Slovak university of Technology in Bratislava, Slovakia
{andrejko,barla,bielik}@fiit.stuba.sk

The Web represents an information space where the amount of information grows exponentially. This calls for personalized interaction between users and web-based information systems providing information. Current systems provide a certain level of personalization, which allows the user to set up her preferences manually. Improved efficiency of information acquisition can be achieved by personalization based on a user's particularities used for the adaptation of content or navigation in the information space. A user model that reflects a real user, who requires information provided by an information system, is required for successful personalization. We present an ontology-based approach to user modeling and describe the user model that we designed for a web-based information system aimed at job acquisition. We point out several advantages of the ontology-based approach, namely the sharing of the ontology with other applications and reusability.

1 Introduction

People are often overloaded by information while finding relevant information can be nearly impossible. This problem is yet more exposed in information systems that cover a large information space (e.g., the Web) where we suppose that individual users have different knowledge and information needs. The system's general interface and behavior designed as "one size fits all" is obviously not effective for all categories of users, thus adaptation is desirable.

One approach to solve this problem lies in increasing the efficiency of the user's interaction with the information system by focusing on individual user needs thus introducing personalization as an additional feature of web-based information systems. A user model that reflects the real user who requires information provided by an information system is required for the successful personalization. The aim of this paper is to analyze and discuss possible user model representations in order to show advantages and disadvantages

resulting from an ontology-based approach to user modeling in web-based information systems. We discuss an approach to user modeling where the model is expressed by an ontology and present the approach on an example of the user model being developed in the course of the research project Tools for acquisition, organization and maintenance of knowledge in an environment of heterogeneous information resources [15]. In this project the Web is considered to be a heterogeneous information source and software tools are developed for a web-based information system aimed at job offer acquisition. Since we model a user who is looking for a job, we will use the labor market as the application domain throughout the paper.

2 User characteristics

A user model represents various user characteristics, which can be used to adapt the content, presentation or navigation. The user model is defined as beliefs about the user that include preferences, knowledge and attributes for a particular domain [14] or as an explicit representation of properties of individual users or user classes [4].

Designers describe user models with terms like attributes, features, characteristics or properties. For the purpose of this paper we use the term *characteristic*. Based on the differences that do not lie only in the terminology, it is obvious that the user modeling area needs to be standardized with the first attempt being the User Modeling Meta-Ontology [21].

As an example of using user characteristics in the process of adaptation, let us assume that a characteristic in the user model describes the minimum acceptable salary per month for a specific user. If the system knows that the user is not interested in job offers where the offered salary is lower (or much lower considering the fuzzy nature of the characteristic) than her expected salary, it will not present offers that do not fulfill this condition. The system adapts information in behalf of the user with the help of the user model.

Other occasions where the user model content can be used, is the interpretation of the user's input [12] (which can be ambiguous, incomplete, with errors, etc.) and the personalization of the system's output (sorting of the results, number of results per page, font, colors, etc.). The more relevant characteristics describing the user are included in the user model the more accurate the adaptation provided by the information system can be. Designers exploit invaluable knowledge of specialists who work in the application domain for which the user model is designed to. Their experiences might help to construct the user model reflecting the real user as accurately as possible.

3 User model representations

There are several approaches to representing and storing a user model in a web-based information system. For user modeling it is important to analyze

to what extent is a particular representation flexible for different kinds of user characteristics in a uniform manner together with the possibility of reasoning directed to decisions on information content presented to the user. We do not discuss representations that use proprietary formats as this would almost totally prevent the sharing and reuse of the user model.

3.1 Non-ontological representations

Markedly the most obvious is the use of a relational database to store data about the user since most information systems already use this kind of application data storage. In this case, the user model is represented as a set of database tables and user characteristics are mapped to attributes in the relational data model and store values assigned to individual user characteristics.

Using a relational database is quite straightforward, offers good performance and several other advantages such as security, data recovery etc. that result from good theoretical background of relational calculus and the maturity of its realization by database management systems. However, user models of web-based information systems often contain semi-structured data as they use an *overlay model*, which follows the representation of the information space with various characteristics defined for concepts from the domain model. Relational databases are not primarily designed to express semi-structured data. Moreover, relational databases are not well suited when frequent changes in data structure need to be performed, which is often the case in user modeling.

Another frequently used approach in current web-based adaptive systems is the representation of the user model by an XML based language using the file system, what results in powerful enough expressiveness. An example is the open source general-purpose adaptive web-based system AHA! [7]. The part of the user model which stores information about the user's name is defined in the AHA! as follows:

```
<record>
  <key>personal.name</key>
  <value>John Smith</value>
  <firsttimeupdated>false</firsttimeupdated>
</record>
```

The performance of this solution is limited by the performance of the used file system (it is effective for user models with few instances and rich structure of user characteristics). Reusability and sharing is better than with the database approach, thanks to the platform independence of XML, while using XML has the advantage that it can be used directly in the Web environment. However, XML as a meta-language defines only the general syntax without formally defined semantics, which leads to difficulties when reasoning. Moreover, everyone can invent his own names for tags; somebody stores attributes as tags; somebody uses the attributes of tags defined by XML syntax.

Both of the above mentioned approaches offer only a way of describing user characteristics and do not offer any added value from the user modeling perspective. An ontology-based approach to user modeling offers a way of moving user modeling from the low-level describing of user characteristics to a higher-level with additional possibilities.

3.2 Representing user model by ontology

According to the most cited definition of ontology in the Semantic Web community, an ontology is an explicit specification of the conceptualization of a domain [8]. The term ontology includes a whole range of various models with various semantic richness. In this paper we consider representing the ontology by RDF[1]/OWL[2] formalisms. An approach based on RDF and its extension OWL takes the previously mentioned XML representation (syntax) and eliminates its disadvantage by defining a vocabulary for describing properties and classes. OWL serves as a common language for automated reasoning about the content for the vision of the Semantic Web.

For illustration, bellow is a fragment representing a user's name and working experience that is a part of the ontology-based user model for the job offer acquisition web-based information system:

```
<rdf:Description rdf:about="#name">
  <rdfs:label xml:lang="en">name</rdfs:label>
  <rdf:type rdf:resource= "http://www.w3.org/2002/07/
                                        owl#DatatypeProperty"/>
  <rdfs:domain rdf:resource="#User"/>
  <rdfs:range rdf:resource=
                    "http://www.w3.org/2001/XMLSchema#string"/>
</rdf:Description>
<rdf:Description rdf:about="#hasExperience">
  <rdfs:label xml:lang="en">has working experience</rdfs:label>
  <rdfs:domain rdf:resource="#User"/>
  <rdfs:range rdf:resource="http://www.fiit.sk/
                     classification#ExperienceClassification"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/
                                          owl#ObjectProperty"/>
</rdf:Description>
```

The advantages leading to using ontologies for user modeling come from the fundamentals of this formalism. Ontologies provide a common understanding of the domain to facilitate reuse and harmonization of different terminologies [14]. They support reasoning, which is considered as an important contribution of the ontology-based models. Once user characteristics are in ontological representation, the ontology and its relations, conditions and restrictions provide the basis for inferring additional user characteristics. For

[1] Resource Description Framework, `http://www.w3.org/RDF/`
[2] Web Ontology Language, `http://www.w3.org/2004/OWL/`

example, considering a user who is a programmer and works for a company that develops web-based applications using Java technologies we can infer that she is skillful in Java technologies.

By creating an ontology-based user model and deriving it from the domain ontology, we increase the probability that user characteristics will be shared among a range of systems of the same domain (especially on the Web, where most ontologies are currently represented in OWL). We consider the sharing of user models as one of main advantages of using ontologies for user modeling. One of the most obvious advantages of a shared model is that one system can use the initialized data for personalization from other systems preventing the user from entering the same information into every system (e.g., name, locale settings). However, the key advantage of the shared user model is the availability of user characteristics discovered by other systems since user characteristics acquisition is considered to be the bottleneck of personalization.

As an example consider the web-based information system for the job offer acquisition discovering that the user's education is in the domain of information technologies with deep knowledge of the object-oriented paradigm of programming. As the user searches for a job, she visits another adaptive job offer portal. Because it uses the same user modeling server it has access to information about her education and automatically displays offers seeking specialists on object-oriented design at the top of the results list.

Some authors believe that the solution to syntactical and structural differences between user models which interfere with sharing is in a commonly accepted user model ontology [10]. Since we agree that building common vocabularies is important and useful (we remark the role of standards), considering a large distributed information space (e.g., the Web) we need to make a compromise between enabling diversity and looking for mappings between various models. The idea of a single commonly accepted user ontology is simply impossible to reach in such diverse and distributed environment.

Certainly, a unified representation by ontologies can move the personalization on the Web further and give new possibilities of using user characteristics derived by other applications. Considering structural unification a problem arises when applications using the shared user model evaluate some user characteristic differently. This characteristic would constantly change as the user uses various applications, which can lead to unsuitable personalization in all applications using the respective characteristic. One solution to this problem is to keep track of model changes [19]. This would allow each application to use this tracking as an additional information for personalization.

4 User model for the job offer acquisition domain

We developed the user model and software tools for its employing for personalization in the context of research project aimed at the support of acquisition, organization and presentation of information on the Web [15, 16].

The result of the project is a web-based information system in domain of the labor market (both for people who are looking for a job and companies which are looking for employees). The system itself is being developed by means of several cooperating software tools, which support various stages of the data-information-knowledge transformation from raw unknown data from the Web to the information and knowledge related to specific interests of particular users. The interests are stored and maintained in a user model.

The data and knowledge repository is designed as a heterogeneous space [6] where several formalisms are used. Raw data from the Web are stored as files, data extracted from the Web source files are stored in a relational database, the domain model together with the user model both used for personalized presentation of acquired job offers are represented by ontologies.

4.1 User model ontology

We have designed the user model in several iterations according to a user dependency criterion, which divides user characteristics into domain-independent and dependent groups. When shared, the user model consists of one domain-independent part and more domain-dependent parts.

Domain-independent part

The domain-independent part includes user characteristics that describe a user as a person. This part consists of datatype and object properties (see Figure 1) and is linked with all domain-dependent parts (`includes`). Datatype properties are `hasBirthday` and `hasName`. Object properties represent user characteristics (`GenericUserCharacteristic`). Each such characteristic is assigned a level of *confidence* that represents how reliable the characteristic is and *relevance*, which represents the importance of characteristic to the user when reaching a `Goal`. Generic characteristics are mapped only to the domain ontology parts that are independent from our labor market domain. That way a user's gender, education, various skills etc can be expressed.

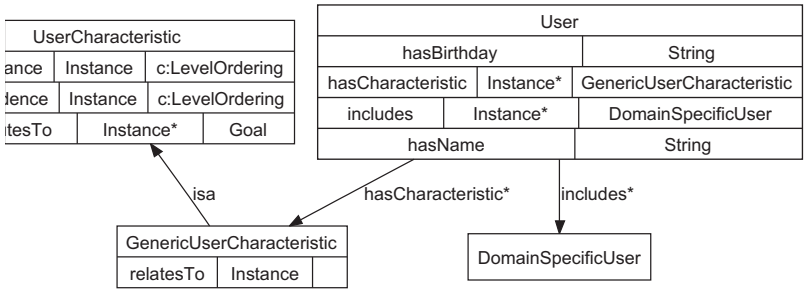

Fig. 1. Domain-independent part of the user model.

One possible scenario is that the user will exploit services provided by some user modeling server (e.g. UMoWS [3]) to store her user model. The user will then populate the domain-independent part of the model and let other applications fill-in appropriate domain-dependent parts.

Domain-dependent part

The vocabulary of the domain-dependent part of the user model is based on the domain model ontology developed for the project, which represents an explicit conceptualization of job offers. It profits from the advantage of ontology reuse and also uses other ontologies (whose domain is independent from labor market domain) to achieve the desired conceptualization.

The domain model consists of the following ontologies:

- ontology *classification* (prefix "c") – hierarchies for industrial sectors, professions, educational levels, qualifications and various organizations;
- ontology *region* (prefix "r") – domain of regions, countries, languages and currencies that are used in these regions;
- ontology *offer* (prefix "ofr") – general offer domain, which is represented by the `ofr:Offer` class; any offer has a source and a validity interval.

The `JobOffer` class is the key class of the ontology and represents a standalone job offer. `JobOffer` has several object and datatype properties. Some selected object properties of the `JobOffer` class are shown in the Figure 2.

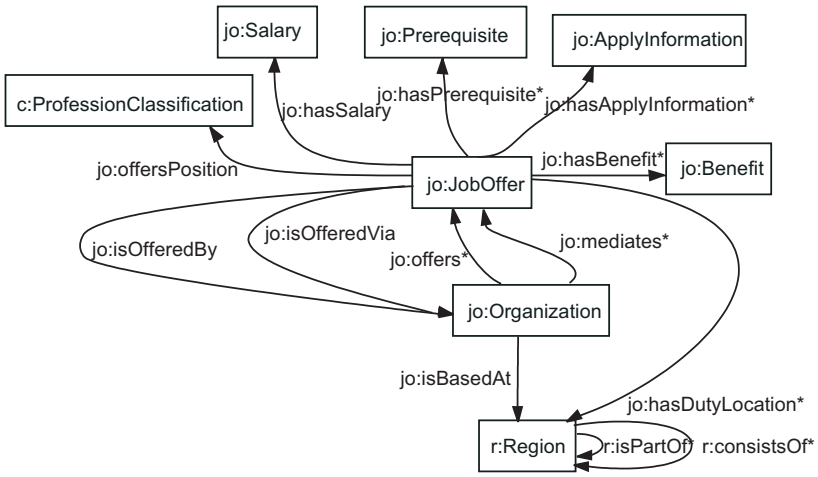

Fig. 2. Selected object properties of the `JobOffer` class.

The aforementioned ontologies provide the base for the user model and a mapping between the domain and user models. Figure 3 depicts the part of the user model created for the domain of labor market. In the case of the job

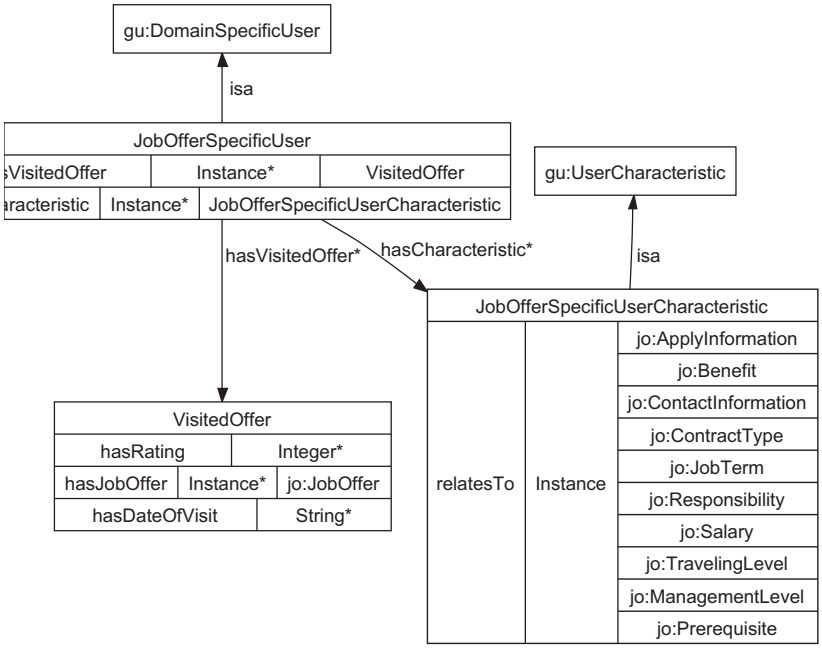

Fig. 3. Domain-dependent part of the user model.

offer domain the model stores information about the user's desirable job offer – user preferences towards a job offer.

The key class is `JobOfferSpecificUser` a subclass of `DomainSpecificUser` from the domain-independent part of the model. This enables the inclusion of this model into the overall user model. We introduced a class for representing domain-dependent characteristics `JobOfferSpecificUserCharacteristic` as a subclass of `UserCharacteristic` (that means it has confidence and relevance as mentioned in the domain-independent part). Instances of this class are bound to the attributes of the job offer class and represent user preferences. For example, the user model can contain information about the preferred contract type, expected salary or prerequisites.

A specific part of the user model can define additional information necessary for personalization. Such an information in the domain of job offers can be a list of already visited job offers, which is modeled by the `VisitedOffer` class. The user model stores information about the offer itself (`hasJobOffer`), the evaluation assigned to it by the user (`hasRating`) as well as the date of the visit (`hasDateOfVisit`).

4.2 Tools for user modeling support

We employ the described user model to provide personalization [1]. To fill the user model with data we observe user actions within a web-based system

combined with explicit input from the user. However, we focus on automated acquisition of the user characteristics. We collect as much data as possible about the user's actions by employing standard server side logging mechanisms as well as a special client side JavaScript logging tool called *Click*. Click records actions, which are not visible on the server (e.g., reload of a page stored in browser's cache, hover on page elements or using the back button, which is important for discovering user interests on the portal).

Afterwards, the collected data are analyzed to estimate selected user characteristics. We analyze user navigation and implicit user feedback derived from the time the user spent by viewing job offers [2]. The estimation uses heuristics and predefined patterns of navigation on the site. Some of these heuristics need to compare two domain concepts to find out their common and different aspects. Once an instance of the model exists in a system, it can be used by other components of the system to perform the adaptation itself, which can be of various types – annotation of displayed content, its sorting etc. Therefore, the comparison between the user ontology and the domain ontology instances is necessary albeit not straightforward.

In contrast to the common attribute-value models, the ontology provides structured data and a one-on-one comparison does not provide satisfactory results because two individuals usually may provide semantically similar information even though they are not on the same level in the hierarchical structure of the ontology. Since a part of the structure of the ontology is known (property `subClassOf`) we use a recursive algorithm to traverse the hierarchy. Because we consider attributes that have identical parent nodes to be closer, we take into account both a straight path between attributes in the hierarchy and also what branch of the hierarchy tree they belong to.

The user characteristics are used by several software tools aimed at further refinement of user interests. The *Top-k aggregator* tool retrieves the most relevant job offers with respect to user preferences (e.g., salary, education requirements, place) based on ordered lists of user preferences [9]. The *Aspect* tool searches for similar documents (job offers) based on a probabilistic model for soft clustering [17]. Using the described approach for the comparison of domain and user ontology instances the devised clusters are presented the user according to her characteristics.

5 Related works

Although several possible user model representations are currently used, the user modeling community has changed focus recently to ontology based approaches, which bring several advantages as discussed above. Several projects, which either concentrate on building reusable user model ontologies or employ the user model ontology as a part of an adaptive web-based system exist.

UserML – the RDF-based user model exchange language [11] extends the XML structure to be able represent graph structure by means of two cooper-

ative levels. The first one defines a simple XML structure for the user model entries and the second one are categories defined in the ontology. The advantage of this approach is that different ontologies can be used with the same UserML tools. UserML served as a base for the reusable user model ontology GUMO – General User Model Ontology [10] represented in OWL. GUMO provides a collection of the user's dimensions (e.g., user's heart beat, age, position, etc.) that might be helpful for several information systems intending to provide personalization based on the user model. These characteristics can be shared also with our user ontology when the web-based information system realizes adaptation according such personal characteristics.

OntobUM (Ontology based User Model) is a generic ontology-based user modeling architecture developed for a knowledge management system [18]. The user model consists of an implicit and an explicit part. While the explicit part contains characteristics such as identity, preferences, the implicit part is related to experiences related to system usage. Our approach of considering domain dependence of user model ontology extends this classification.

Among the projects, which use an ontology based user model representation we mention ADAPT^2 – Advanced Distributed Architecture for Personalized Teaching & Training [5]. It stands for a general framework for distributed education that employs an Ontology Server to user model exchange.

The idea of a shared user model is also elaborated in [13]. Here, the Personis server that uses a proprietary representation of a model based on triplets *component–evidence–source* is described. There is no explicit definition of the triplets' semantics and each application can define its own triplets not regarding the others, which limits its reusability. Another project, UMoWS [3] uses OWL representation of a model. Because the same knowledge can be represented by different ontologies on different levels of abstraction UMoWS supports the representation in multiple ontologies and can provide the mapping between them to applications, so they can share a common model.

6 Conclusions

The main contribution of this paper lies in describing advantages of an ontology-based representation of user models aimed at providing personalization in web-based information systems. We concentrated on comparison with other currently used approaches for user model representation. We consider the simplification of exchanging user model data between different applications as the major advantage of using ontologies. The presented ontology developed in the course of a research project aimed at the job offer acquisition application domain contributes to the state of the art by separation of domain-independent and domain-dependent parts of the user model. Separating the domain-independent part of user characteristics allows us to build a general user model. This kind of the user model can be used in a wide-range of applications while only adding parts, which differ from one application to

another. We also use the presented ontology as the user model in a research project aimed at developing a recommendation layer for digital libraries that serves for personalization of its services.

An ontology is not the only representation that is advantageous for user modeling in web-information systems. Systems that build user models based on user monitoring represent logs that are also considered to be parts of the user model using simpler data structures as ontologies (often XML files are sufficient). Another example of non-ontological parts of the user model are statistics related to user behavior. Ontological representation is advantageous for those parts of the user model that are related to user characteristics where some reasoning is useful, i.e. checking the consistency of values.

Developed software tools mentioned in the paper can also be used for other domains as that of labor market. The tools are designed to be domain independent realizing their methods with an optional domain-dependent layer. Navigation in the domain is done through a faceted semantic browser [20], which is designed for the use in various application domains.

Acknowledgement. This work was partially supported by the Slovak Research and Development Agency under the contract No. APVT-20-007104 and the State programme of research and development "Establishing of Information Society" under the contract No. 1025/04.

References

1. A. Andrejko, M. Barla, M. Bieliková, and M. Tvarožek. Tools for user characteristics acquisition. In *Datakon'06*, 2006. Accepted.
2. M. Barla. Interception of user's interests on the web. In V. et al. Wade, editor, *4th Int. Conf. on Adaptive Hypermedia and Adaptive Web-Based Systems, AH'06*, pages 435–439, Dublin, Ireland, 2006. Springer, LNCS 4018.
3. M. Bieliková and J. Kuruc. Sharing user models for adaptive hypermedia applications. In *5th Int. Conf. on Intelligent Systems Design and Applications, ISDA'05*, pages 506–511, Wroclaw, Poland, 2005. ACM Press.
4. P. Brusilovsky, A. Corbett, and F. de Rosis. User modeling 2003: Preface. In *9th Int. Conf. on User Modelling*, Johnstown, USA, 2003. Springer, LNCS 2702.
5. P. Brusilovsky, S. Sosnovsky, and M. Yudelson. Ontology-based framework for user model interoperability in distributed learning environments. In G. Richards, editor, *E-Learn 2005*, pages 2851–2855, Vancouver, Canada, 2005. AACE.
6. M. Ciglan, M. Babik, M. Laclavik, I. Budinska, and L. Hluchy. Corporate memory: A framework for supporting tools for acquisition, organization and maintenance of information and knowledge. In *9th Int. Conf. on Inf. Systems Implementation and Modelling, ISIM'06*, Prerov, Czech Republic, 2006.
7. P. de Bra, A. Aerts, B. de Lange, B. Rousseau, T. Santic, D. Smith, and N. Stash. AHA! the adaptive hypermedia architecture. In *ACM Conf. on Hypertext and Hypermedia*, pages 81–84, Nottingham, UK, 2003.

8. T.R. Gruber. Towards principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Deventer, The Netherlands, 1993. Kluwer Academic Publishers.
9. P. Gurský, R. Lencses, and P. Vojtáš. Algorithms for user dependent integration of ranked distributed information. In M. Böhlen et al., editor, *TCGOV 2005 Poster Proceedings*, Bozen-Bolzano, Italy, 2005.
10. D. Heckmann, T. Schwartz, B. Brandherm, M. Schmitz, and M. von Wilamowitz-Moellendorff. Gumo – the general user model ontology. In L. et al. Ardissono, editor, *10th Int. Conf. on User Modeling, UM'05*, pages 428–432, Edinburgh, Scotland, UK, 2005. Springer, LNCS 3538.
11. Dominik Heckmann and Antonio Krueger. A user modeling markup language (UserML) for ubiquitous computing. In P. et al. Brusilovsky, editor, *9th Int. Conf. on User Modelling, UM'03*, pages 393–397, Johnstown, USA, 2003. Springer, LNCS 2702.
12. J. Kay. User modeling for adaptation. In C. Stephanidis, editor, *User Interfaces for All*, Human Factors Series, pages 271–294, Florence, Italy, 2000.
13. J. Kay, B. Kummerfeld, and P. Lauder. Personis: A server for user models. In P. et al. de Bra, editor, *2nd Int. Conf. on Adaptive Hypermedia and Adaptive Web-Based Systems, AH'02*, pages 29–31, Malaga, Spain, 2002. Springer, LNCS 2347.
14. J. Kay and A. Lum. Ontology-based user modeling for the semantic web. In *10th Int. Conf. on User Modeling, UM'05, Workshop 8*, pages 11–19, Edinburgh, Scotland, UK, 2005.
15. P. Návrat, M. Bieliková, and V. Rozinajová. Methods and tools for acquiring and presenting information and knowledge in the web. In *Int. Conf. on Computer Systems and Technologies, CompSysTech 2005*, Varna, Bulgaria, 2005.
16. G. Nguyen, M. Laclavik, Z. Balogh, E. Gatial, M. Ciglan, M. Babik, I. Budinska, and L. Hluchy. Data and knowledge acquisition with ontology background. In W. Abramowicz, editor, *Business Information Systems, BIS'06*, Poznan, Poland, 2006.
17. G. Polčicová and P. Tiňo. Making sense of sparse rating data in collaborative filtering via topographic organisation of user preference patterns. *Neural Networks*, 17:1183–1199, 2004.
18. L. Razmerita, A. Angehrn, and A. Maedche. Ontology-based user modeling for knowledge management systems. In *9th Int. Conf. on User Modeling, UM'03*, pages 213–217, Johnstown, PA, USA, 2003. Springer, LNCS 2702.
19. M. Tury and M. Bieliková. An approach to detection ontology changes. In *1st Int. Workshop on Adaptation and Evolution in Web Systems Engineering (AEWSE 06) at ICWE 2006*, Palo Alto, California, 2006. ACM Press. Accepted.
20. M. Tvarožek. Personalized navigation in the semantic web. In V. et al. Wade, editor, *4th Int. Conf. on Adaptive Hypermedia and Adaptive Web-Based Systems, AH'06*, pages 467–471, Dublin, Ireland, 2006. Springer, LNCS 4018.
21. M. Yudelson, T. Gavrilova, and P. Brusilovsky. Towards user modeling meta-ontology. In L. et al. Ardissono, editor, *10th Int. Conf. on User Modeling, UM'05*, pages 448–452, Edinburgh, Scotland, UK, 2005. Springer, LNCS 3538.