ADAPTATION DRIVEN CHANGE MANAGEMENT

Lai Xu, Paul de Vrieze
SAP Research Switzerland, Blumenbergplatz 9, CH-9000 St.Gallen, Switlzer.and
lai.xu@sap.com, paul.de.vrieze@sap.com

Athman Bouguettaya
CSIRO ICT Centre, Canberra, ACT, Australia
athman.bouguettaya@csiro.au

Jian Yang
Department of Computing, Macquarie University, Sydney NSW 2109, Australia
jian@comp.mq.edu.au

Keywords: Change Management, User Adaptive Systems.

Abstract: In solving the global obesity epidemic the realization has come that it is mainly an issue of behavioral change. Benefits have been shown in personalizing the information that induces these changes. At the same time obesity is a heavily researched field in which new discoveries are made every day. This brings the need for maintainability of these personalized information channels. In this paper we propose architecture for a maintainable system for the provision of tailored documents for obesity patients.

1 INTRODUCTION

Obesity is a global epidemic that has serious impact on the occurrence of non-transmittable chronic diseases. The treatment of obesity is understood to be found in behavioral change. The transfer of information forms a significant part of the efforts to change the dietary behavior and activity levels of obesity patients. It is understood that different obesity patients react differently to different information dependent on their personal properties. Using computers to generate tailored brochures based on user characteristics has been tried by various research groups and found to have a positive health outcome (Brug et al., 1996, Brug et al., 2003, Kreuter et al., 1999).

As obesity is a chronic disease, and behavior change has to be actively monitored and redirected to be retained. As such it is important that the means used to achieve this will remain available through this lifelong process. Obesity however is also an area where a lot of research is being done. This means that the system must be able to evolve with the knowledge it is built upon without loosing the knowledge gained from previous interactions with the patient.

Developing such a system, we face some frequent changes. For example, new or complementing research of obesity treatments becomes available now and then. The personal conditions of each patient may change during the process. Trust in the system or health professionals from each patient will be gained or lost. Personal characteristics are influenced by the experiences of the weight management process. Furthermore, the change of adaptation strategies of the system should also be taken into account. Knowing, adapting to and managing changes are thus important aspects for the design of the system.

There are three types of changes in the weight management system. First, the changes of personal conditions, personal characteristics, and trust in the system or health professionals from the patient are naturally processed by the user adaptive system. Second, new research results on obesity treatment can be seen as new knowledge. In the area of knowledge management the subject of how to detect, handle changes of knowledge and organize information in a better way have been broadly investigated. Third, the changes that cannot be handled by either user adaptive systems or knowledge management systems. These changes, such as adaptation dynamics, changes and integration, are the research focus of this paper. The majority of research in user adaptive system and knowledge management system is focused on construction issues. Coping with the changes and providing maintenance facilities however require a different approach.

Change mining in the context of decision tree classification for real-life application is studies in (Liu et al.,2000). Their primary goal is to know what is changing and how it has changed in order to provide the right products and services to suit the change market needs. In our system, the goal of managing changes of the users' interactions with the systems and changes of knowledge is not only to predict changes but also to improve the efficiency of the system.

Paper (Berendt et al., 2002) discusses the possibilities to combine the two research areas Semantic Web and Web Mining. The idea is to improve the results of Web Mining by exploiting the new semantic structures in the Web, and to make the user of the Web Mining for building up the Semantic Web. The ideas inspire us to define semantics for the log information and using the log information to enhance ontologies and usage of system.

In the areas of knowledge management and ontology evolution, Heflin (Heflin, 2001) argues the needs of evolution of ontologies on the Web. A new formal definition of ontologies is provided for the use in dynamic, distributed environments. The way of performing the change is however not dealt. Ontology evolution can be treated as a part of the ontology versioning mechanism that is analyzed in (Klein and Fensel, 2001). An overview of causes and consequences of the changes in the ontology has been provided. A detailed analysis of the effect of specific changes on the interpretation of data is missed.

We provide an ontology-based approach for adaptation models and discuss issues of adaptation models in section 2. Ontology changes in the system are argued in section 3. Section 4 provides The conceptual architecture of the weight management system. Finally section 5 summarizes concluding remarks.

2 NEEDS OF ONTOLOGY IN USER ADAPTIVE SYSTEMS

In user adaptive systems the system maintains a user model. This user model is updated to reflect actions of the user. When the system then needs knowledge about the user to tailor itself it does so by inferring from the user model the answers needed. What the user model looks like, how the system updates the user model and which inference is used to get the needed answers is described by the adaptation model.

While an adaptation model, or a meta model for a user model is sufficient to describe the behavior of a user adaptive system at a particular point in time, it can not be used to relate with different systems, or even different adaptation models for the same system.

Ontologies allow one to express the relation between two adaptation models. As user models are instantiation of an adaptation model, the ontology can thus allow the transition of a user model for the old adaptation model to a user model for the new adaptation model.

$$diff = am' – am$$

$$um' = apply(diff, out, um)$$

For this and other purposes we need another layer on top of the adaptation model. This ontology layer defines the meanings of the concepts used in the adaptation model. The ontology, for example, describes for the possible user properties what they mean and what type they have.

Ontologies are also essential in supporting change of adaptation models. As adaptation models get updated the user models need to be updated accordingly. As there can be many user models this updating should not need to be manual.

As an example of how ontology can help support an adaptation model *a* containing two properties: *weight* and *BMI* (Body Mass Index). The body mass index is defined as the individual's body weight divided by the square of their height. As this system is used it shows to be inconvenient to update *BMI* as well as *weight*. Given that the height of adults is more or less constant, it would be better to store height instead and calculate the body mass index on demand.

To perform this change we first verify that the ontology contains the concepts *weight*, *height* and *BMI* and their relation. Then we create, based on our original adaptation model *a* a new adaptation model *a'*. In this new adaptation model the *BMI* user property is replaced by a *height* property and a *BMI* function. As the relation between the concepts is defined by the ontology the *BMI* function can be created automatically. Given *a* and *a'* we now need a function $f(u_a)=u_a$ that updates user models for *a* to user models for *a'*. This function again can be determined automatically from the ontology to calculate *height* from *BMI* and *weight* and store this height instead of *BMI*.

Ontologies can further be used for adaptation model composition. With the rise of ubiquitous computing multiple sources of user information become more and more probable. In our weight management system the patient information can be built from different devices such as mobile phones, PDAs and so on. In this respect it is sensible to extend the model to allow multiple applications to share a user model.

3 ONTOLOGY CHANGES

Our weight management information system provides weight control strategies to individual user cases. The system maintains knowledge of diet, exercise and so on. The tailored advice is generated according to different weight control stages and individual conditions. In the context of the weight management information system, the purposes of using ontologies are:

- semantic annotation of knowledge fragments,
- describing relationships between knowledge fragments,
- selecting and assembling demanded knowledge fragments according to the ontology, and
- sketching adaptation model elements.

The first three purposes are normal ontology uses in knowledge management systems. The last use is a specific use for our weight management information system. Contrary to one's intuition in our ontology-based application ontologies need to be timely adapted to changed requirements from environments and users.

3.1 Types of Changes

Ontology change can be classified from two dimensions. One dimension can be viewed as implicit changes and explicit changes whereas another dimension specifies the origin of changes, namely internal or external changes. The external requirements of changes come from the environment, while the internal changes can be required by the system, such as the usage of ontologies in the system. Table 1 summarises all changes expected from different perspectives.

Table 1 – Changes of Environment and System

Changes	Internal / System	External / Environment
Implicit	Concept creep	External sources change usage of ontology.
Explicit	Fixes, turning, refinement	New knowledge

First, we will show an example of internal implicit changes. To show this we distinguish the written ontology from the effective ontology as it is used by the system. The existence of an effective ontology can be shown in analogy with a natural language, such as the living English language, where the dictionary needs to be updated regularly to reflect changes in the meanings of words. As an ontology is used by humans, such subtle change of meaning can also occur in ontologies, changing the effective ontology. Thereby causing the effective ontology to no longer be equal to the written ontology.

Implicit change can also happen externally. If a system incorporates information from external sources, their effective ontologies can change. The ontology then needs to be update to reflect this external change. For example, in a web service environment, suppose one hotel chain decides that the provision of a mini-bar will be standard in its two-star rooms while it is not a standard requirement for a two-star hotel. The two-star hotel can update the room specification independently. As a customer who is using a hotel search system, querying about hotel rooms with a mini-bar. The customer may not be able to find the two-star hotel with a mini-bar, because the ontology in the hotel search system specifies only standard contents of two-star hotels. Being able to overcome such problems, the ontology in the hotel search system should be added a new subconcept or property for the concept of two-star hotel.

Explicit internally driven change can be seen in the maintenance of the system and ontology, such as fixing, turning and refining ontology. For instance, from user feedback of a hotel search system, the system administrator knows that customers complain about the inability to check whether ``Heineken" beer is served by the hotel. Therefore, subconcept or property ``Heineken" of concept ``Beer" should be added. In our weight management system, when a new adaptation strategy is found, deploying a new adaptation strategy is also explicit internal change.

Finally, explicit, externally driven change is for example caused by the need to add new knowledge to the system which needs new concepts added to the ontology. For example, introducing a new concept, such as eight-star hotel rooms in a hotel search system.

3.2 Needs of Ontology Changes

While it is interesting to see where change originates, one should not neglect looking at the need for change. If one understands the needs for change, one knows what to look for so that the needed changes are made. If the underlying ontology is not up-to-date, annotated and assembled knowledge fragments can be inconsistent, redundant or incomplete. The reliability, accuracy, effectiveness and adaptivity of the weight management system thus decrease significantly.

Like ontology change, the need for ontology change can be regarded from the two axes of implicit versus explicit and internal versus external. There is no one to one mapping between ontology change and need for ontology change though. The summary of needs for ontology changes is presented in Table 2.

Table 2: Needs for Ontology Changes

Needs for ontology changes	Internal	External
Implicit	Lack of query focus, precision or recall; Unused concepts for the current ontology.	Abandonment of concepts etc.; Inexistent concepts for the current ontology
Explicit	Incompatibility of new concepts with ontology; Inefficient treatments.	Out-of-date information; Adopting new user models.

To address the needs for ontology change, assessment of an ontology-based system is needed. From an information retrieval perspective, the criteria are the improvement of precision and recall of the system. From adaptive system, adaptivity is another standard.

An implicit, internally motivated need for change could for example be that the system has a bad precision (on query), or that a concept is never used. It requires one to actively look at what is wrong.

An implicit external need could be the abandonment of concepts by an external source, or access to new knowledge was the ontology to be updated. For this reason it is useful to monitor the actual usage of term and identifies new terms such as what is ``eight-star hotel room".

An explicit internally motivated need for change would for example be to updated the ontology to be compatible with new knowledge. Like a new treatment does not comply with the used ontology to annotate.

Finally, an example of an explicit external need for change would be the change of a law that applies to the system. An example of a change of a law is updating four-digit zip codes into five digits.

4 ARCHITECTURE

The management of an ontology-based user adaptive system is complex and essential for a sustainable and extensible information system dealing with changes in user's need as well as environment. We present architecture (in Figure 1) of a user adaptive system, which can handle changes in the adaptation model.

In order to efficiently retrieve advice, the content of the knowledge resource (authorised document, cf. 1 in Figure 1) is annotated in terms of the underlying domain ontology (10). The annotation process can be supported by an annotation tool (2). Annotations can be explicitly added to the knowledge resources. The annotated knowledge sources are stored in a knowledge repository (3).

The knowledge is used in the response handler. The document composition engine (4) is responsible for assembling the needed document based. The response planner (5) determines which document the document composition engine should compose.

The adaptation component is the part of the system that is responsible for modeling the user, and making this knowledge available to the rest of the system (de Vrieze, 2006). An appropriate way to look at the adaptation component would be as a virtual copy of the user that does not mind answering lots of questions. As such the question handler (6) takes the questions from the rest of the system, and based on the user model (7) determines the best answer. The user model stores the knowledge on the user, and is maintained by the user behavior analyser (8).

The event dispatcher (9) is responsible for feeding the user actions, as events, to the various interested parts of the system. Events are sent to the response planner such that the system may respond. Events sent to the user behavior analyser allow it to take the events into account in the user model. Events sent to the log allow for later, off-line, analysis of usage of the system that forms an important basis for the recognition of the need for evolution of the system.

Ontologies (10) store ontologies used for annotating knowledge and ontologies used for changing or integrating adaptation models. Ontology usage is monitored in the monitoring component (13). It results in a set of recommendations for changes to the annotations and/or domain ontologies. These recommendation sets are processed in the ontology evolution component (12). The recommendations can still be accepted or rejected by the knowledge engineer. The ontology evolution component needs to ensure the consistency of the system (incl. ontology and all dependent artifacts) after resolving changes.

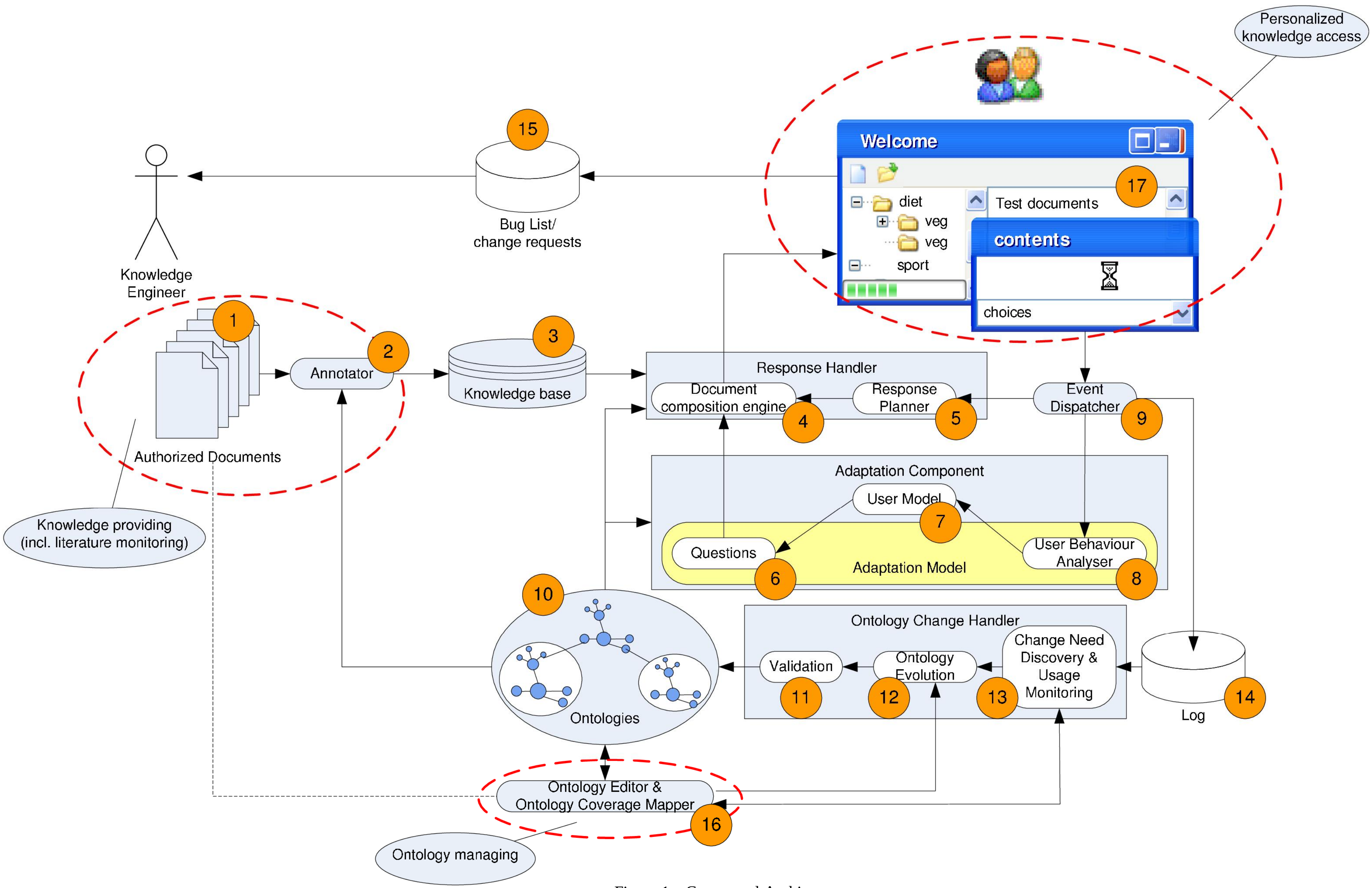

Figure 1 – Conceptual Architecture

Information whether and how the knowledge repository reflects the needs of end-users can be obtained by analyzing the user's interaction with the system. For example which subjects are looked at, how many results are delivered, which documents are found and so on. This data is captured in the log file (14). The log file is also processed in the ``Change need discovery & usage monitoring'' component (13). In contrast to the analysis of annotation, the log-file analysis might result in the requirements to update the knowledge repository. The bugs and suggestions reported by the final users of the system are stored in the bug list (15).

The changes derived from the user's behavior are application-oriented and, after applying these changes, the system should improve its performances. The task of the validation component (11) is to detect changes that decrease system performance and to react appropriately. E.g. by rolling back the changes involved, restoring the original state of the whole system (the ontology, annotations and the knowledge repository) in the case of decreasing performance.

The ontology editor and ontology coverage mapper (16) is an interface for ontology management within the system. For example for the coverage mapper it integrates with the knowledge as stored in the system.

5 CONCLUSIONS

We derived a novel approach for dealing with weight management. Specifically the dynamic nature of the user behavior and the obesity related research. The approach is based on a user-adaptive system. In particular, we consider changes from both user behavior perspectives as well as changes of the (knowledge) environment. In order to allow the system to handle the evolution during long-term usage, the design of the system takes into account the sustainable management of change.

The benefits of the proposed approach are manifold: (i) the system can handle changing user behavior and build up a long term trust relationship, The system further allows the integration of different adaptation models. (ii) The system will handle the change of the environment. The evolution process enables continuous system improvement by semi-automatic discovery of changes. (iii) The system provides a sustainable information system. The validation component helps the maintainer in better understanding of effects of each change providing detailed insight into each change being performed.

ACKNOWLEDGEMENTS

Part of work from Lai Xu and Paul de Vrieze has been performed at the CSIRO ICT Centre, Australia.

REFERENCES

Moore, R., Lopes, J., 1999. Paper templates. In *TEMPLATE'06, 1st International Conference on Template Production*. INSTICC Press.

Smith, J., 1998. *The book*, The publishing company. London, 2nd edition.

Berendt, B., Hotho, A., and Stumme, G. 2002. Towards Semantic Web Mining. In *Proceedings of the First international Semantic Web Conference on the Semantic Web* (June 09 - 12, 2002). I. Horrocks and J. A. Hendler, Eds. Lecture Notes In Computer Science, vol. 2342. Springer-Verlag, London, 264-278.

Heflin, J. 2001. Towards the Semantic Web: Knowledge Representation in a Dynamic, Distributed Environment. PhD thesis, University of Maryland, College Park.

Klein, M. and Fensel, D. 2001. Ontology Versioning on the Semantic Web. Proceedings of the International Semantic Web Working Symposium (SWWS), pages 75–91.

Lerner, B. S. and Habermann A. N. 1990. Beyond schema evolution to database reorganization. In OOPSLA/ECOOP '90: Proceedings of the European conference on object-oriented programming on Object-oriented programming systems, languages, and applications, pages 67–76, New York, NY, USA.

Liu, B. Hsu, W. Han, H.-S. and Xia, Y. 2000 . Mining changes for real-life applications. In DaWaK 2000: Proceedings of the Second International Conference on Data Warehousing and Knowledge Discovery, pages 337–346, London, UK. Springer-Verlag.

Mann, W. and Thompson, S. 1987. Rhetorical Structure Theory: A Framework for the Analysis of Texts. University of Southern California, Information Sciences Institute.

National institutes of Health, National Heart, Lung, and Blood Institute, NHLBI Obesity Education Initiative, and North American Association for the Study of Obesity. 2008. The practical guide: Identification, evaluation, and treatment of overweight and obesity in adults. http://www.nhlbi.nih.gov/guidelines/obesity/prctgd_c.pdf. Last accessed Feb 4, 2008.

NHS. Clinical knowledge summaries. http://www.cks.library.nhs.uk/.

de Vrieze, P. 2006 *Fundaments of adaptive personalisation*. PhD thesis, Radboud University Nijmegen. ISBN-13: 978-90-9021113-8.

Wing, R. and Phelan, S. 2005. Long-term weight loss maintenance. American Journal of Clinical Nutrition, 82(1 (S)):222–225.