# Purpose-based User Modelling in Decentralized Agent and Web-Service Based Environments

Xiaolin Niu, Julita Vassileva, Gordon McCalla [1]

Department of Computer Science
University of Saskatchewan
Saskatoon, Saskatchewan, S7N 5A9 Canada
{xin978, mccalla, jiv}@mail.usask.ca

**Abstract.** This paper outlines a new approach for decentralized agent based user modelling using a taxonomy of *purposes* that define a variety of context-dependent user modelling processes rather than creating and maintaining a single centralized user modelling server. This approach can be useful in distributed environments where autonomous agents develop user models independently and do not necessarily adhere to a common representation scheme.

## 1 Decentralized active user modelling

Traditionally user modelling has focused on creating and maintaining a single global description of the user used internally in an application for some purpose defined at design time [1]. Knowledge representation is a key issue in this kind of traditional user modelling. With the emergence of networked applications, user modelling servers have been proposed [2] to store data that can be used to support adaptation in several networked applications. User modelling servers provide a centralized solution: user models are stored in centralized or virtually centralized repository. Even if the user data comes from and serves various applications, the representation of the user model follows a particular centralized schema, which is known in advance to the applications.

However, software systems currently are shifting to web-services, which not only distributed, but also autonomous and often agent-based. The autonomous agents or web services keep user model fragments, which can be used by others only if the services/agents are willing to share the information [4, 8]. These fragments cannot be expected to use the same representation scheme (the same problem arises in distributed databases, see [3]). Even if they wish to do so, the user model fragments come from a range of sources (e.g. raw data, other agents) and are dependent on the context in which they were created, so it would be very hard to ensure consistency in a centralized user model based on these fragments as input. Therefore the focus of user modelling shifts from the collection at one place of as many data about a user as

[1] This work is supported by the Natural Sciences and Engineering Research Council of Canada.

possible to collecting on demand whatever user information is available at the moment from various agents and interpreting it for a particular purpose. This is called *active* user modelling [4].

## 2. Purpose-based user modelling

This paper presents briefly a purpose-based approach to active user modeling [5], which is aimed at defining a taxonomy of purposes and retrieving user information relevant to a particular purpose just in time in order to assemble and integrate fragmented user model information. This purpose-based user modelling is based on procedural representation of purposes. Each purpose has inputs, outputs and functions, which can be used to retrieve and integrate input information in order to generate desired output. Therefore, when an agent or web-service invokes a purpose for a given user modelling task, the purpose shows how to find relevant user model fragments and how to integrate them by executing a modelling function. This modelling process depends on the resource constraints, e.g. which other services or agents are available at the moment to provide information, what kind of user models they can provide and how much time is available for computation.

Purposes can be organized into generalization or aggregation hierarchies. The set of purposes can thus be viewed at many levels, for example, from general to specific. One purpose can aggregate several sub-purposes. The way sub-purposes are aggregated can be defined by the functions of the super purpose. Some sub-purposes can be called sequentially, i.e. when the super purpose is called, the sub-purposes of this super purpose will be called in one after another as long as there are resources available to continue, in an “anytime” fashion.

A library of purposes forms a repository of clichés, which can be adapted to a new situation. For example, a purpose can be generalized into a super purpose, specialized into (a set of) more specific purposes; shared by several super-purposes; or modified for use in a new domain. Purpose re-use is valuable from a software engineering point of view and critical to the active approach. Another aspect to software re-use is model re-use, which means re-using the result of the computation, i.e. the output of a purpose, as input to other purposes.

The purpose hierarchies within this system architecture are maintained by a set of specialized user modelling web-services associated with each purpose. These agents are networked according to the purpose hierarchies. Each user modelling service asks the next (according to the aggregation or to the generalization dimension in the purpose hierarchy) available UM service to continue the computation needed to achieve the appropriate sub-purpose. In this way web-services and agents subcontract user modelling tasks to the specialized user modelling agents for the purpose, which perform computations upon request and return the results to the requesting agent or service without storing any data. In this way, the computation of user models and the storage of user data in this architecture are fully decentralized. Specialized purpose agents can be reused easily.

In a system with many UM services specialized for a wide range of purposes, an agent- or web-service based application can perform just-in-time user modelling by calling the appropriate UM service for its particular purpose. Similarly to developing a full ontology of a domain, envisaging all possible purposes for user modelling in all possible contexts is a hard task. However, the decentralization allows multiple designers to gradually create UM services for various purposes and they will become useful immediately. The effort of the designer should focus at the first place on important, reusable purposes that can be either reused directly, or easily adapted.

The decentralization of the user modelling process and focussing it on the particular adaptation purpose at hand is orthogonal to the problem of commonly understandable representation language for user modelling information. There are two approaches to ensure such common understanding: using a centralized ontology for user modelling and mapping different user model representations. The second approach is more interesting since it is also decentralized, and it is a currently active area of research [3, 9].

Our work [5, 6] has demonstrated the use of purposes in a multi-agent portfolio management domain [7]. There are two kinds of agents in the system: personal agents (PA) represent investors who need advice, and expert agents (EA) who provide that advice. There are many purposes for adaptation in this domain. One of the main adaptation purposes for a PA is to find an appropriate EA for a given investor, which is a kind of personalized recommendation. This purpose requires models of the investor and the EA as well as eventually models of other investors and their EAs.

## 3 Experiments and Results

The goal of the experiments is to demonstrate that this purpose-based user modelling is *feasible* and that there is a methodology of systematic development of purposes through organizing them in hierarchies that allow for reuse. The system was evaluated with personal agents representing simulated investors to show that is provides adapted functionality according to pre-set investors' preferences. The experiments [6] demonstrated the anytime aspect of purposes and also showed that the quality of the decisions improves when more resources and agents are available. These results help to establish a performance baseline for the system and to discover the influence of environment factors on the system and possible measures (e.g. reordering the purposes to take into account such factors as the expected level of deception).

We only ran 40 personal agents in our experiments. The further challenge is scaling up to a larger set of agents. There will be a lot of overheads associated with the multi-agent system, such as finding the agents and communication etc. We expect that the quality of predictions should improve and the speed not worsen as the number of agents grows. This is the case because the more people use the system; the greater the chances are of finding close matches for any particular investor (i.e. neighbours). One potential limit to growth will be communication (influencing network traffic, performance, response time), if the system grows very large. There will always be a trade-off between resource constraints and quality of performance. The anytime algorithms allow an easier solution to this: if time resources are extremely limited, and there is sufficient number of investors, a reasonable performance can be achieved while executing only one or two simple sub-purposes which are much cheaper and require less communication.

## 4 Contributions

Our research makes a number of contributions in both active user modelling and software engineering.

- Designing and developing an agent environment for studying decentralized, active user modelling issues.
- Demonstrating how a purpose-based approach can implement the decentralized active user modelling paradigm.
- Providing an extendible approach to active user modelling to allow re-use and adaptation of purposes and anytime algorithms to handle varying modelling constraints.
- Showing an example of how simulation can be used to evaluate an adaptive system.

The purpose based approach and its evaluation is described in detail in [5, 6].

## References

1. Browne, D., Totterdell, P., Norman, M.: Adaptive user interfaces. Academic Press Ltd., London, UK, 1990
2. Fink, J., and Kobsa, A.: A Review and Analysis of Commercial User Modeling Servers for Personalization on the World Wide Web. User Modeling and User-Adapted Interaction, 10 (3-4) (2000), 209-249
3. Giunchiglia, F., Zaihrayeu, I.: Making peer databases interact - a vision for an architecture supporting data coordination. Technical Report # DIT-02-0012. Also to appear in Proc. Cooperative Information Agents (CIA 2002), Madrid, September 2002
4. McCalla, G., Vassileva, J., Greer, J. and Bull, S.: Active Learner Modeling. Proc. ITS2000, Springer LNCS 1839, (2000) 53-62
5. Niu, X., McCalla, G., Vassileva, J., (2003) Purpose-based user modelling in a Multi-agent portfolio management system. Proceedings User Modelling UM03, Johnstown, PA, June 22-26, 2003. Springer Verlag LNCS 2702, 398-402
6. Niu, X., McCalla, G., Vassileva, J.: Purpose-based expert finding in a portfolio management system. Computational Intelligence Journal, Vol. 20, No. 4, (2004) 548-561.
7. Tang, T., Winoto, P. & Niu, X.: Who Can I Trust? Investigating Trust between Users and Agents in a Multi-agent Portfolio Management System. In AAAI-2002 Workshop on Autonomy, Delegation, and Control: From Inter-agent to Groups. Edmonton, Canada, July 28, 2002
8. Vassileva, J., McCalla, G., Greer, J.: Multi-Agent Multi-User Modelling. To appear in User Modelling and User-Adapted Interaction, (2003), ***13**:(1), 179-210.*
9. Hopkins, I., Vassileva J. (to appear) Beyond Keywords and Hierarchies. *Journal of Data and Information Management*, special issue on Distributed Data Management. Available on line at: http://bistrica.usask.ca/madmuc/tagging.htm