The User Model and Context Ontology GUMO revisited for future Web 2.0 Extensions

Dominik Heckmann, Eric Schwarzkopf, Junichiro Mori, Dietmar Dengler, Alexander Kröner

German Research Center for Artificial Intelligence
D-66041 Saarbrücken, Germany
heckmann@dfki.de

Abstract. We revisit the top-level ontology `Gumo` for the uniform management of user and context models in a semantic web environment. We discuss design decisions, while putting the focus on ontological issues. The structural integration into user model servers, especially into the U2M-UserModel&ContextService, is also presented. We show ubiquitous applications using the user model ontology `Gumo` together with the user model markup language `UserML`. Finally, we ask how data from Web 2.0 and especially from a social tagging application like del.icio.us as a basis for user adaptation and context-awareness could influence the ontology.

Keywords ubiquitous user modeling, semantic web, ontological engineering, web 2.0, user model markup language

1 Motivation and Introduction

A commonly accepted top level ontology for user and context models is of great importance for the user modeling and context research community. This ontology should be represented in a modern semantic web language like OWL and thus be available for all user-adaptive systems at the same time via internet. The major advantage would be the simplification for exchanging user model and context data between different user-adaptive systems.

However, the current trends of web 2.0 and social computing tell us that the users like to create their own tag spaces, naming conventions and taxonomies. The masses of tagging, rating and even blogging define a kind of "wisdom of the crowds". Now the question arises how this new bottom-up approach can be combined with the more top-down approach of ontology engineering. Does a revisiting of a domain ontology like the user model and context ontology GUMO make sense? There are two directions of mutual influence possible. An existing ontology could be used in taxonomy learning of tag spaces in a way of seeding, or the other way round, the taxonomies that are dynamically generated by the tagging behavior of communities can be used to correct or update existing ontologies. Approaches for tag-space mining are presented in [Schmitz et al., 2006], [Heymann and Garcia-Molina, 2006] and [Golder and Huberman, 2006]. And in [Mika, 2005] a first attempt is shown how to learn ontologies from tag-space mining. Please notice that we present in this paper only initial thoughts in the direction of the duality of ontology engineering and tag-space mining. Back to the ontological approach. The problem of syntactical and structural differences between existing user modeling and context systems could be overcome with a commonly accepted taxonomy, specialized for user modeling tasks. Note, that we are talking about a user model ontology rather than a user model*ing* ontology, which would include, the inference techniques, or knowledge about the research area in general. We are analyzing the user's dimensions that are modeled within user-adaptive systems like the *user's heart beat*, the *user's age*, the *user's current position* or the *user's birthplace*.

Ontologies provide a shared and common understanding of a domain that can be communicated between people and heterogeneous and widely spread application systems, as pointed out in [Fensel, 2001]. Since ontologies have been developed and investigated in artificial intelligence to facilitate knowledge sharing and reuse, they should form the central point of interest for the task of exchanging user models. The design choices in our approach are described in the following. The main conceptual idea for the construction of the specialized user model ontology `Gumo` was to divide the descriptions of user model dimensions into three parts: `auxiliary` - `predicate` - `range`. For example if one wants to say *something about the user's interest in football*, one could divide this into the auxiliary part: "interest", the category part "football" and the range part: "low-medium-high". If a system wants to express something like *the user's knowledge about Beethoven's Symphonies*, one could divide this into the triple: "knowledge" - "Beethoven's Symphonies" - "poor-average-good-excellent". As a third example, *the user's hair-color* would lead to: "property" - "hair-color" - "black-red-brown-blonde-white". First of off all, important groups of auxiliaries have to be identified. A list of identified important user model auxiliaries could be { has Property, has Interest, has Believe, has Knowledge, has Preference, has Regularity, has Plan, has Goal, has Location }. This listing is not intended to be complete, but it is a start with which, most of the important user facts can be realized. Then the user model predicates have to be classified and analyzed. But it turned out that actually everything can be a category for the auxiliary "interest" or "knowledge", thus a whole world-ontology would be needed, what leads to a real problem if one does not work modularized. The crucial idea is to leave this part open for existing other ontologies like the general CYC ontology (see [Lenat, 1995] for example), the UbisWorld ontology (see [Stahl and Heckmann, 2004]), or any other. This insight leads to a modular approach which forms a key feature rather than a disadvantage. Nevertheless the problem of finding a commonly accepted, specialized top level ontology for the user modeling research group is moved into the user's property section: Which classes of user dimensions can be identified? In [Jameson, 2001] and in [Kobsa, 2001] rough classifications for such categories can be found. However, no top level user model ontology has been proposed so far.

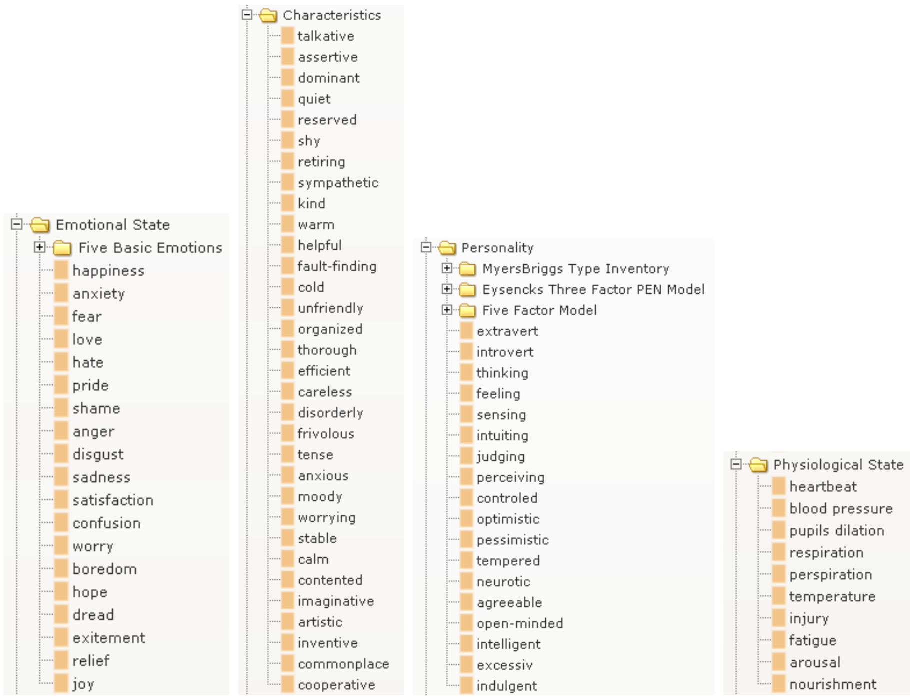

Fig. 1. Several User Model Property Dimensions: Emotional States, Characteristics and Personality with included sub models like the "Five Factor Model"

2 Representation of `Gumo` in OWL

In this section we discuss, why we have chosen the web ontology language OWL. We present three concept definitions, namely the class "Physiological State", the user model dimension "Happiness" and the auxiliary "has Knowledge".

2.1 Three example concept definitions from `Gumo`

Figure 2 presents as a first example the concept of the user model dimension class *Physiological State* which is realized as a `owl:Class`. A class defines a group of individuals that belong together because they share some properties. Classes can be organized in a specialization hierarchy using `subClassOf`. There is a built-in most general class named Thing that is the class of all individuals and a superclass of all OWL classes. The *Physiological State* is defined as subclass of *Basic User Dimensions*.

Every new concept has a unique `rdf:ID`, that can be resolved into a complete URI. Since the handling of these URIs could become very unhandy, a short identification number was introduced, the so called `u2m:identifier`. The identification number in this case is 700016, it has been chosen arbitrarily but seen

```
<owl:Class rdf:ID="PhysiologicalState.700016">
    <rdfs:label> Physiological State </rdfs:label>
    <u2m:identifier> 700016 </u2m:identifier>
    <u2m:lexicon>the state of the body or bodily functions</u2m:lexicon>
    <u2m:website rdf:resource="&UserOL;concept=700016" />
    <rdfs:subClassOf rdf:resource="#BasicUserDimensions.700002" />
</owl:Class>
```

Fig. 2. The OWL class definition of "Physiological State"

under its namespace, it is unique. It has the advantage of freeing the textual part in the `rdf:ID` from the need of being semantically unique. The term *mouse* for example, could be read as the animal mouse, or as the computing device mouse. Apart from solving the problem of conceptual ambiguity, this number facilitates the work within relational databases, which is important from the implementation point of view.

Figure 2 also defines the lexical entry `u2m:lexicon` of the concept of *Physiological State* as "the state of the body or bodily functions", while this textual definition could also be realized through a link to an external lexicon. The attribute `u2m:website` points towards a web site, that has its purpose in presenting this ontology concept, to a human reader. The abbreviation `&UserOL;` is a shortcut for the complete URL to the `Gumo` ontology.

```
<rdf:Description rdf:ID="Happiness.800616">
    <rdfs:label> Happiness </rdfs:label>
    <u2m:identifier> 800616 </u2m:identifier>
    <u2m:durability> Hour.520060 </u2m:durability>
    <u2m:image rdf:resource="http://u2m.org/UbisWorld/img/happiness.gif" />
    <u2m:website rdf:resource="&UserOL;concept=800616" />
    <rdf:type rdf:resource="#EmotionalState.700014" />
    <rdf:type rdf:resource="#FiveBasicEmotions.700015" />
</rdf:Description>
```

Fig. 3. GUMO definition of "Happiness"

Figure 3 defines the user model dimension *Happiness* as an `rdf:Description`. It contains a `rdfs:label`, a `u2m:identifier` and a `u2m:website` attribute. Additionally it provides a default value of the average durability `u2m:durability`. It carries the qualitative time span of how long the statement is expected to be valid (like minutes, hours, days, years). In most cases when user model dimensions or context dimensions are measured, one has a rough idea about the expected durability, for instance, emotional states change normally within hours, however personality traits won‘t change within months. Since this qualitative time span is dependent from every user model dimension, a definition mechanism is prepared within the `Gumo`. Some examples of rough durability-classifications, without any attempt of proven correctness, are:

– physiologicalState.heartbeat - can change within seconds
– mentalState.timePressure - can change within minutes
– emotionalState.happiness - can change within hours
– characteristics.inventive - can change within months
– personality.introvert - can change within years
– demographics.birthplace - can't normally change at all

Another important point that is shown in the definition of *happiness* in figure 3 is the ability in OWL of multiple-inheritance. In detail, happiness is defined as `rdf:type` of the class *Emotional State* as well as `rdf:type` of the class *Five Basic Emotions*. Thus OWL allows to construct complex, graph like hierarchies of user model concepts, which is especially important for ontology integration. Figure 4 defines the auxiliary *has Knowledge* as `rdfs:subPropertyOf` of the

```
<rdf:Property rdf:about="hasKnowledge.600120">
    <rdfs:label> has Knowledge </rdfs:label>
    <u2m:identifier> 600120 </u2m:identifier>
    <u2m:website rdf:resource="&UserOL;concept=600120" />
    <rdfs:domain rdf:resource="#Person.110003" />
    <rdfs:subPropertyOf rdf:resource="#UserModelAuxiliary.600020" />
</rdf:Property>
```

Fig. 4. GUMO Property hasKnowledge as example for general auxiliaries

resource *user model auxiliary* with the `rdf:domain` *#Person*, which is not part of the user model ontology itself, but which is part of the general UbisWorld Ontology, see [Stahl and Heckmann, 2004]. The acronym `u2m` stands for *ubiquitous user modeling* and forms a collection of standards, that are available online at `http://www.u2m.org/`. The new vocabulary for the user model ontology language consists of `u2m:identifier, u2m:durability, u2m:image, u2m:website u2m:lexicon` . The main User Model Dimension that we identified so far are *MentalState PhysicalState, Demographics, ContactInformation, Role, EmotionalState, Personality, Characteristics, Ability, Proficience* and *Motion*.

To support the distributed construction and refinement of the top level user model ontology, we developed a specialized online editor, that helps with introducing new concepts, adding their definitions and transform the information automatically into the required semantic web ontology language. Currently supported are RDF and OWL.

3 The U2M-UserModelServer

A user model server manages information about users or individuals in general. The U2M-UserModel&ContextService, see [Heckmann, 2003a] is an applicationindependent server with a distributed approach for accessing and storing user information, while the focus lies on the possibility to exchange and understand the data between different applications, as well as adding privacy and transparency to the statements about the user itself. The semantics for all concepts is mapped to the `Gumo` ontology.

Applications can retrieve or add information to the server by simple HTTP requests, alternatively, by the "UserML" WebService. UserML, see for example [Heckmann and Krüger, 2003], is an XML application which is based on the concept of "situational statements", as introduced in [Heckmann, 2003b]. A request could look like:

```
http://www.u2m.org/UbisWorld/UserModelServer.php?
subject=Joerg.210006&auxiliary=hasProperty&predicate=Age.800302
```

Mentionable is the optional naming convention for disambiguation, like "Joerg.210006" or "Age.800302". These names are unique identifiers for the particular, intended concepts. A general problem when one wants to talk about objects, individuals or concepts is the non-uniqueness of names, as seen before, especially in an open web-based system. In the Semantic Web approach, each resource is mapped to a (hopefully) unique URI. But the URIs have the disadvantage that they are rather long and uneasy to read. The used naming-format "Name.Id" can be seen as a shortcut for such a unique URI. Those unique resource identifiers, for the area of user modeling, are established in the `Gumo`.

The user model server "u2m.org" can be used by every user adaptive system to manage user related data, but also by the modeled user himself. A specialized UserModelEditor is provided which displays the information in a web-browser form that allows the change and privacy control, see `http://www.u2m.org`. The access, the purpose and the retention of every situational statement can be controlled in the "editor view modus". Each statement can contain meta information like creator, method, evidence or confidence. Figure 5 shows the overall architecture of the UserModelServer with its input and output information flows *Query*, *Answer* and *Add* that are represented as arrows. The main block of the illustration contains four piled, dotted rectangles. The lowest one indicates the distributed storage of the so called SituationalStatements, which are explained in detail in [Heckmann and Krüger, 2003]. The second rectangle shows the filter, ranking and conflict resolution strategies that are applied to the set of Situational Statements. The User Model Server itself, which is responsible for communication, handling requests and responses, is based on both introduced rectangles as well as the rectangle on the top for distributed knowledge bases in form of semantic web ontologies. A query or request, that is received in the so called `UserQL` query language will be handled by the user model server in the following way: first all matching situational statements are retrieved, then the filter and resolution strategies are applied and finally the semantics is given by referencing to web ontologies.

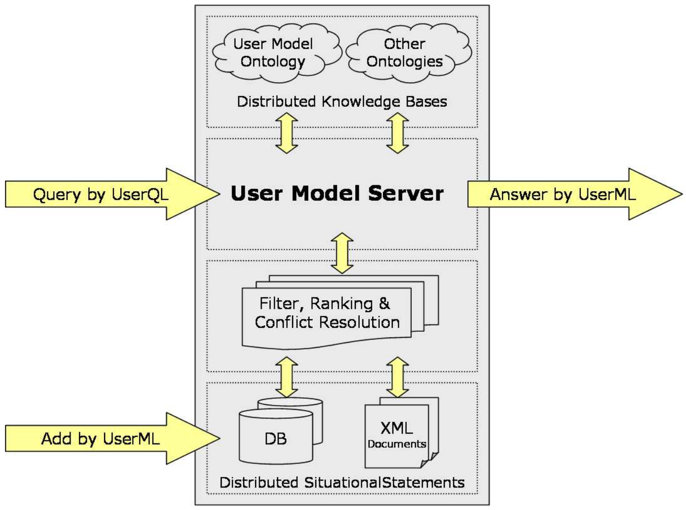

Fig. 5. Architecture of the UserModelServer

4 How to further develop GUMO in the era of Web2.0?

The Semantic Web is based on the content-oriented description of digital documents with standardized vocabularies that provide machine understandable semantics. The result is the transformation from a Web of Links into a Web of Meaning / Semantic Web, (see arrow A in Fig. 6). On the other hand, the traditional Web 1.0 has recently been orthogonally shifted into a Web of People / Web 2.0 where the focus is set on folksonomies, collective intelligence and the wisdom of crowds (see arrow B in Fig. 6). Only the combined muscle of semantic web technologies and broad user participation will ultimately lead to a Web 3.0 with completely new business opportunities in all segments of the ITC market. Without Web 2.0 technologies and without activating the power of community-based semantic tagging, the emerging semantic web cannot be scaled and broadened to the level, that is needed for a complete transformation of the current syntactic web. On the other hand, current Web 2.0 technologies cannot be used for automatic service composition and open domain query answering without adding machine-understandable content descriptions based on semantic web technologies. The ultimate world-wide knowledge infrastructure cannot be produced fully automatically, but needs massive user participation based on open semantic platforms and standards.

The interesting and urging question that arises is: what happens when the emerging Semantic Web and Web 2.0 meet with their full potential power?

There are no new technologies introduced by Web 2.0, but the role and value of the user has been changed significantly. We focus in this paper on *tagging*.

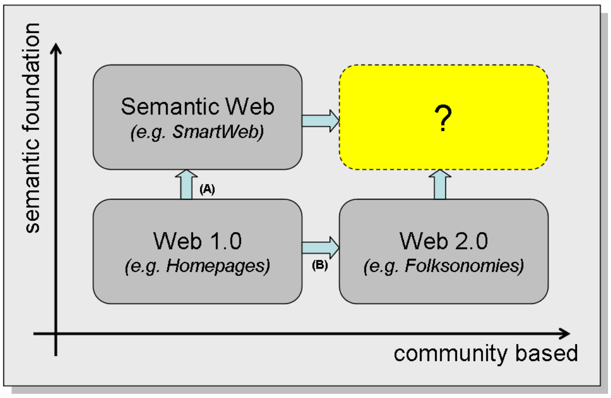

Fig. 6. Joining Semantic Web and Web 2.0

However, a social *rating* system could also be of interest in order to improve the ontologies.

Tag spaces are an obvious source of data for user modeling. The user of a social tagging tool could provide access to his personal tag space to an e-commerce site which could use the data to tailor its structure and presentation to the user. For example, a music store could attempt to assess where a user lives given data from a social bookmarking site. Then, if the user is interested in an album by an artist who will give a concert in the vicinity of the user's home town, the store could offer him tickets for the event. How can we use a tag space and a user's tagging data to create a user model and adapt a system? Furthermore, how can we use the already developed general user model and context Ontology `Gumo` to improve the tagging taxonomy and the generated user model and context rules? Figure 7 shows the possible connection of `Gumo` and Web 2.0.

The approach we are proposing starts with automatically learning a structure of the tag space, then manually defining adaptation rules based on that structure, and finally automatically mapping a user's data into the structure in order to decide what adaptation rules to apply. This implies that the set of possible adaptation rules depends on the learned structure. For instance, creating a rule with a precondition on the home town of a user is sensible only if this information is part of the structure. Not all tag spaces are suitable for this type of user modeling. Because we want to learn something about the user's interests, we require tagging data used by the user for himself (as in del.ico.us) and not for others (as in flickr).

We are aiming for a taxonomy of tags, where subtags of a tag tag (for example, pop-music should be a subtag of music). For the designer of an adaptive system, identifying the semantics of a tag (by using its predecessors and successors its generality (the higher it is in the taxonomy, the more users will Hence, we think a taxonomy is a good underlying structure for the a taxonomy from a

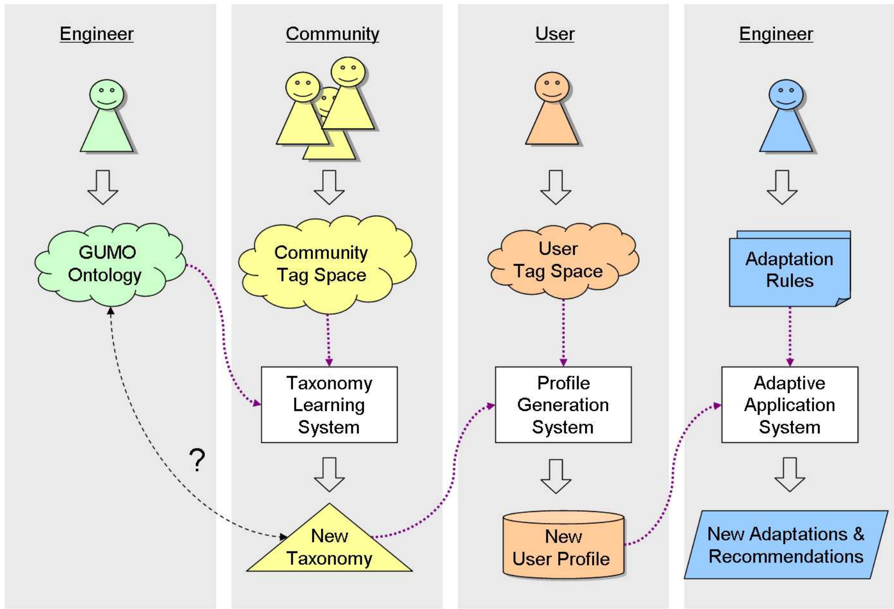

Fig. 7. A possible connection of GUMO and Web 2.0

tag space is the main subject of this paper. See [Schwarzkopf et al., 2007] for a detailed description of this approach.

Summary We have revisited the user model and context ontology `Gumo` in the semantic web ontology language OWL together with the exchange language `UserML` and the U2M UserModel&ContextServer. This work is highly under progress and the future goal is to find out the influence of social computing in Web 2.0 to the so far only semantic web approach in order to determine the possible advantages of combining tag-space mining and ontology engineering.

References

[Fensel, 2001] Fensel, D. (2001). *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*. Springer-Verlag Berlin Heidelberg.

[Golder and Huberman, 2006] Golder, S. A. and Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *J. Inf. Sci.*, 32(2):198–208.

[Heckmann, 2003a] Heckmann, D. (2003a). Integrating Privacy Aspects into Ubiquitous Computing: A Basic User Interface for Personalization. In Krüger, A. and Malaka, R., editors, *Artificial Intelligence in Mobile Systems (AIMS 2003)*, pages 106–110, Seattle, USA. in conjunction with the Fifth International Conference on Ubiquitous Computing.

[Heckmann, 2003b] Heckmann, D. (2003b). Introducing situational statements as an integrating data structure for user modeling, context-awareness and resource-adaptive computing. In *ABIS2003*, pages 283–286, Karlsruhe, Germany.

[Heckmann and Krüger, 2003] Heckmann, D. and Krüger, A. (2003). A user modeling markup language (UserML) for ubiquitous computing. *Lecture Notes in Artificial Intelligence*, 2702:393–397.

[Heymann and Garcia-Molina, 2006] Heymann, P. and Garcia-Molina, H. (2006). Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical Report 2006-10, Stanford University.

[Jameson, 2001] Jameson, A. (2001). *Systems That Adapt to Their Users: An Integrative Perspective*. Habil, Saarbrücken, Germany.

[Kobsa, 2001] Kobsa, A. (2001). Generic user modeling systems. *User Modelling and User-Adapted Interaction Journal*, 11:49–63.

[Lenat, 1995] Lenat, D. B. (1995). CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38.

[Mika, 2005] Mika, P. (2005). Ontologies are us: A unified model of social networks and semantics. *In International Semantic Web Conference, volume 3729 of Lecture Notes in Computer Science*, pages 522–536.

[Schmitz et al., 2006] Schmitz, C., Hotho, A., Jaschke, R., and Stumme, G. (2006). Mining association rules in folksonomies. *In V. Batagelj, H.-H. Bock, A. Ferligoj, and A. iberna, editors, Data Science and Classification, Studies in Classification, Data Analysis, and Knowledge Organization, Berlin, Heidelberg, Springer*, pages 261–270.

[Schwarzkopf et al., 2007] Schwarzkopf, E., Heckmann, D., Dengler, D., and Krner, A. (2007). Mining the structure of tag spaces for user modeling. In *Complete On-Line Proceedings of the Workshop on Data Mining for User Modeling at the 11th International Conference on User Modeling*, pages 63–75, Corfu, Greece.

[Stahl and Heckmann, 2004] Stahl, C. and Heckmann, D. (2004). Using semantic web technology for ubiquitous hybrid location modeling. In *UbiGis*.