

## Reputation metadata for recommending personalised e-learning resources

Tania Kerkiri, Athanassios Manitsaris, Anastasia Mavridou  
*Dept. of Applied Informatics University of Macedonia*  
*156 Egnatia Street, 54006, Thessaloniki, GREECE*  
{kerkiri@uom.gr, manits@uom.g, mavridan@gmail.com}

### Abstract

*Enhancing the e-Learning systems with reputation, a commonly used notion in e-Commerce systems, adds value to the resource. In this paper a RDF-based e-Learning framework that is based on reputation metadata and educational standards is proposed. Moreover, the model and the retrieval algorithm of a proper recommendation system are described. This system exploits reputation and description metadata to augment personalization in e-Learning systems through recommendation methods.*

### 1. Introduction

Personalisation is an important feature in e-Learning systems due to the differences in background, goals, capabilities and personalities of the adults, the main audience of such systems. Personalisation can be achieved using pre-defined rules that sequentially propose learning resources in a specified learning path [1, 2] subject to the learner's educational progress. It is also achieved by using heuristic rules, user models and recommendation methods [3, 4, 5]. Recommendation is a collaborative filtering algorithms-based method that uses historical information collected from users who have similar profiles to the current one, in order to find out resources that might be suitable for him. Recommender systems provide personalised suggestions about items that users will probably find interesting by using explicit and implicit feedback. Explicit feedback is gathered when a user rates an item as interesting or relevant. On the other hand, implicit feedback is extracted from user actions that indirectly provide some evidence about item quality or relevance, such as link selection, reading time, bookmarking etc [6]. Related work on recommendation as a method to propose personalise learning resources can be found in [7, 8].

A framework that combines both content-based methods, as well as explicit and implicit ranking, is described in [9].

In this paper reputation metadata [10] is proposed as a method to enhance e-Learning systems with recommendation capabilities. Reputation is the cumulative scale of user opinions regarding persons, products, ideas. Reputation is mainly used in e-Commerce systems as a method for building trust [11] and in collaborative networks (such as online social networking-folksonomies and blogs [12]) to provide collaboration.

The paper is structured as follows: in section 2, the proposed framework is presented. In section 3 the model of the proposed system is presented. In section 4 the algorithm that retrieves personalised learning resources using this framework is presented and the system that implements it is evaluated. Our conclusions and future plans are discussed in the final section.

### 2. The proposed framework

Learning resources (LRs) are not only located in libraries but can also be found in different websites, newspapers, magazine articles, conference sites etc, a great amount of which is electronically available. In this aspect the web can be thought as an index to a vast amount of digital information, as well as a content-provider. Not every web-resource is suitable to be used for educational purposes, though. Moreover, searching in the web requires an intelligent recommendation mechanism which retrieves information based on the learner's needs [9]. In this paper a framework is proposed (fig. 1) in which collaborative nodes feed a central data repository with description and reputation metadata about LRs that are now available in local repositories or in various web-sites. Our goal is to create a large number of content metadata for these resources by using the RDF standard [13]. When a resource is described by using RDF it can be retrieved semantically. Moreover, interoperability among applications of the web, through e.g. web services and standard network protocols, is achieved. Finally, the RDF-metadata can be exported to existing ontologies or various Learning Management Systems (LMSs) that

are oriented towards semantic web technologies. The proposed method for describing the LR’s content is the educational standard LOM [14], due to its ability to describe a large range of the attributes of a resource that is suitable for educational purposes. In our system, this kind of information is currently created by domain experts who convert the plain text descriptions about the resources to suitable LOM entries. Proper tools can be used in the RDF-annotator module of the framework, e.g. SHAME (<http://kmr.nada.kth.se/shame/wiki/Overview>), to create the RDF-expressed metadata in accordance to LOM entries.

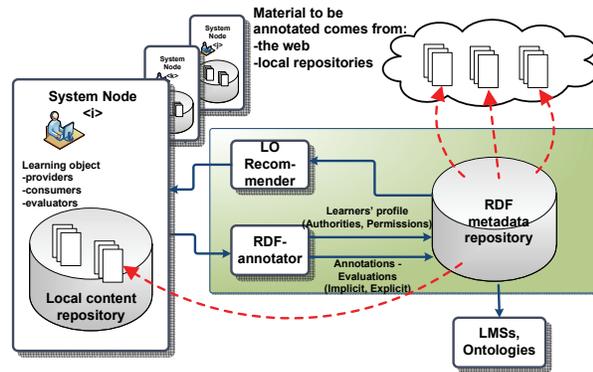


Fig. 1: The framework

Apart the LOM-metadata, the resource is enhanced with a structure that collects reputation metadata for a number of its attributes. As it has been proven, the users of communities like collaborative networks and social networking sites tend to share their tastes about the consumed resources. This kind of explicit reputation is of great importance, because it comes from users that are interested in the knowledge domain the resource belongs to. It also comes from users that have probably consumed it –so they have a personal opinion on it. These reputations are attached to the LR and they can be seen as added value. Suitable criteria for a successful evaluation can be chosen according to the material being evaluated. The criteria of our attention have been chosen for the “Multimedia” course of our institute.

Links to a number of URLs are possible. These links start from this repository and point to different instances of the LR. In this way its availability is augmented. Bi-directional linking between the URL the resource is located and our data repository makes the cumulative reputations of the resource available in public.

The registered learners of this framework have to provide some information about them. The learner profile can be described by using educational standards such as PAPI [15]. PAPI describes the qualifications,

skills, licenses etc the learner has acquired during his life. E.g. the “Certifications” entry of PAPI describes the learner experience in a knowledge domain; the “Preferences” describes his language or document type preferences etc. Having all this information about learners and resources, a recommendation module applies collaborative-filtering methods to recommend personalized LRs.

In the rest of the paper the terms metadata, attributes and properties are used interchangeably, and so are the terms reputation, ranking and evaluation.

### 3. The model

The main concepts considered in the model are: “Learner” (equally, “Consumer”) and “Learning Resource” (LR). The Learners are denoted using the set  $C = \{c_1, c_2, \dots, c_n\}$ . Each  $c_j$  has  $k$  properties  $c_j p = \{c_j p_1, c_j p_2, \dots, c_j p_k\}, \forall j = 1, \dots, n$  that describe his/her profile. Each of them is aligned to entries of the standards PAPI and LOM. For the needs of our system a weight is attached to each of the values of these properties. This weight affects the learner’s evaluation.

The properties in question are presented below, along with their weights:

Property Name	Domain	Allowed Values	Property Weight
cp1=hasCertification	PAPI:Certification	{Vocational Training:0.2, Higher Education:0.4, MSc:0.8, PhD:1, Professional Formation:0.3, Continuous Formation:0.4}	PropertyWeight=0.9
cp2=hasUserRole	LOM:UserRole	{Beginner:0.1, Intermediate:0.4, Advanced:0.9}	Property Weight=0.5
cp3=hasIntendedEndUserRole	LOM:IntendedEndUserRole	{Author:0.8, Lecturer: 0.8, Learner:0.2, Trainer:0.5, Trainee:0.1}	Property Weight=0.9
cp4=hasPerformance	PAPI:Performance.learning_experience_identifier	is calculated	Property Weight=0.7
cp5=hasPreferredFormat	LOM:Format	{Doc, pdf, wav, mp3, txt}	Property Weight=1 if the document has the desired format (this mean it can be proposed), otherwise=0
cp6=hasPreferredLanguage	LOM:Language	{en, el, fr, it, de}	Total Property Weight=1 if the document is in the desired language   otherwise=0
cp7=hasLearningStyle	PAPI:LearningStyle	{active-reflective, sensing-intuitive, visual-verbal, sequential-global}	[16]   Property Weight=1 if the document is in the desired learning style   otherwise=0

The PAPI:Performance entry depicts the resources a learner has consumed (this means: either annotated, or evaluated, or downloaded). The weight of this property is calculated by adding the LRs about a topic a learner

has consumed. The weight of this entry is defined as the accumulated reputation of the resource in the criterion “overall quality”, using the formula (1):

$$W_{iquality} = \sum_{j=1}^t r_{jquality} / r_t \quad (1)$$

where  $r$ =the ranking of the resource  $l_i$  for the criterion “overall quality” and  $t$ =total ranks of the resource  $l_i$  for the same criterion.

Thus, a learner’s  $c_i$  weight for the PAPI:Performance is the total weight of each of the resources  $l_i$  the learner has “consumed”:

$$C_{IPAPIPerformance} = \sum_i W_{iquality} * l_i \quad (2)$$

This entry captures the learner’s interests and learning history. It can be used to find out LRs similar to the ones “similar” learners have consumed or to recommend the resources the current learner has consumed to similar learners or to determine a learner’s activity on a specific knowledge topic. The total weight of a learner is calculated by adding all the points he acquired in every one of his/her properties, each multiplied by the “Property Weight”:

$$W_{ci} = \sum_i (W_i * C_{pi}) / n \quad (3)$$

where  $w_i$  is the weight of the property and  $cp_i$  the accumulated weight of the learner in this specific property. This factor differentiates learners according to their relevance to a specific knowledge domain. Using these weights learners are denoted as “similar” when they have the same values in their properties.

The similarity between the learners is calculated for each of their property by using the Pearson’s  $r$  correlation coefficient, as shown in formula (4):

$$f(s) = sim(c_i, c_j) = \frac{\sum_{i,j=1}^k (W_{ci} * W_{cj})}{\sqrt{\sum_{i=1}^k W_{ci}^2 * \sum_{j=1}^k W_{cj}^2}} \quad (4)$$

where  $k$  is the criteria of similarity (the properties of their profile).

In the same way, the Learning Resource is depicted using the set  $L = \{l_1, l_2, \dots, l_m\}$ , each  $l_j$  having  $l$  properties  $l_{jp} = \{l_{jp1}, l_{jp2}, \dots, l_{jpt}\}$ . The domain  $D$  of each property contains the corresponding data elements chosen from LOM.

The properties of our interest are depicted below:

Property	Domain	Allowed Values
hasFormat,	LOM:Format,	{doc,pdf,txt,wav,...,mp3,other}
hasInteractivityLevel,	LOM:InteractivityLevel,	{Very Low, Low, Medium, High, Very High}

hasLanguage,	LOM:Language,	{en, el, de, it, fr}
hasInteractivityType,	LOM:InteractivityType,	{Very Low, Low, Medium, High, Very High}
hasUserRole,	LOM:UserRole,	{Beginner, Intermediate, Advanced}
hasIntendedUserRole,	LOM:IntendedEndUserRole,	{Author, Lecturer, Learner, Trainer, Trainee}
hasDifficulty,	LOM:Difficulty,	{Very Easy, Easy, Medium, Difficult, Very Difficult}
hasLearningResourceContext,	LOM:ResourceContext,	{Very Easy, Easy, Medium, Difficult, Very Difficult}
hasCoverage,	LOM:Coverage,	{0-33, 34-65, 66-100}

Because the LRs’ properties have been chosen from LOM standard their value is textual –and thus not countable. So, to find out the similarity between LRs we have to compare the string values of these properties. For two properties of this kind to be similar they must have exactly the same value. This is expressed by using formula (5):

$$f(l) = sim(l_{ip}, l_{jp}) = \begin{cases} 1, & \text{if } l_{ip} = l_{jp} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

As we can see, an LR is either suitable to the learner’s criteria or absolutely unsuitable. E.g. if a learner asks for LRs having the value “Difficult” in the property “hasDifficulty”, then there is no intermediate state between “Medium” and “Difficult” or “Difficult” and “Very Difficult”.

To find out LRs having an intermediate value in their properties, a number of countable “reputation” properties are proposed. These reputation properties are used to rank the value of the properties of the LR. Let  $R = \{r_1, r_2, \dots, r_t\}$  be the set of reputation-criteria. For the educational subject “Multimedia”, those have been chosen to be: {Coverage, EducationalContext, InteractivityLevel, InteractivityType, Semantic Density, Difficulty, Originality, Usability, Overall Quality}. Some of them correspond to the description properties of the LR. The last three have been chosen according to the scope of the lesson. An LR with these specific characteristics is more likely to attract the learner’s interest, and the objects having these properties will be preferred by the trainers. Using this set of criteria, reputations are explicitly collected in  $ER_{ij} = \{(c_j, l_i, r_1), (c_j, l_i, r_2), \dots, (c_j, l_i, r_t)\}$ , where  $c_j$  is the learner’s URI,  $l_i$  the LR’s URI and  $r_k$   $k=1, \dots, t$  is the reputation for a specific criterion. As the elements of this set are triples, they can also be described by using RDF. Using this set, learners are defined as “similar” when they have provided similar reputations for a set of common consumed LRs.

Their similarity for a specific criterion is given by the formula (6):

$$f(er) = \frac{\sum_{q \in ER} \sum_{m \in LR \cap I_j} (er_{i,q} - \overline{er_{i,q}}) * (er_{j,q} - \overline{er_{j,q}})}{\sqrt{\sum_{q \in ER} \sum_{m \in LR \cap I_j} (er_{i,q} - \overline{er_{i,q}})^2 * \sum_{q \in ER} \sum_{m \in LR \cap I_j} (er_{j,q} - \overline{er_{j,q}})^2}} \quad (6)$$

where  $m$  is the LR that has been evaluated from both learners ( $c_i, c_j$ ),  $er_{i,q}$  is the current rate given by  $c_i$  for the resource  $l_i$ , using the criterion  $q$ ,  $\overline{er_{i,q}}$  is the average reputations of  $c_i$  for the specific criterion  $q$  for the set of the  $m$  LR and similarly,  $r_{j,q}$  is the rating of  $c_j$ .

We also assume that if a learner asks for an LR by using some of its description metadata and then downloads it, they accept that this resource has the desired value in this metadata. In this way, they provide implicit metadata about the resource and the description metadata are turned into reputation metadata. To collect this kind of implicit reputations, the set IE is defined. In this set, the current average value of the resource's explicit reputation is inserted,  $IE = \{c_j, l_i, r_{pcurrent}\}$ . Because they have been collected implicitly they don't have the same weight in the system. The formula (7) is used to calculate the similarity with the aid of the implicit reputations of the set IR:

$$f(ir) = \frac{\sum_{q \in IR} \sum_{m \in LR \cap I_j} (ir_{i,q} - \overline{ir_{i,q}}) * (ir_{j,q} - \overline{ir_{j,q}})}{\sqrt{\sum_{q \in IR} \sum_{m \in LR \cap I_j} (ir_{i,q} - \overline{ir_{i,q}})^2 * \sum_{q \in IR} \sum_{m \in LR \cap I_j} (ir_{j,q} - \overline{ir_{j,q}})^2}} \quad (7)$$

We assume that similar learners prefer similar LR. Thus, combining the formulas (4), (5), (6), (7) we can find out for similar learners  $f(s)$  those LR which have the desired value  $v$  in their attributes ( $f(l)$ ), or those LR that the learners said that have the desired value  $v$  in their attributes ( $f(er)$ ) or, finally, those LR that the system assumed that they have the desired value ( $f(ir)$ ). So the collected reputation metadata is used to recommend personalised resources to learners.

The formulas (6) and (7) calculate the relative reputations of the resource. The absolute total reputation of a resource is calculated by using formula (8):

$$f(er_p) = Avg \left( Avg (w_{c_jz} * er_p) \right) \quad (8)$$

$j \in \{1, \dots, n\} \quad z \in \{1, \dots, k\}$

where,  $f(er_p)$  the average value of the reputations  $er_p$  accrued for a specific reputation metadata  $r_p$ ,  $p$  is a reputation criterion and  $w_{c_jz}$  is the weight of the Learner  $j$  for his property  $c_j p_z$ . Using (8) for a specific  $l_i$  the following can be derived:

- for given  $p$ , the ranking of a given reputation criterion,
- for given  $z$ , the reputation provided from a specific kind of Consumer. In this case the LR acquires different reputation, whether we emphasise on the

property “PAPI:Certification” or “IntendedEnd-UserRole”,

- for given  $p$  and  $z$ , the reputation of an LR's property from a specific kind of a Consumer.

The absolute reputations can be used to change the value of the description metadata. Different approaches can be followed on how to use these accumulated reputations; the description metadata of a resource can be changed according to the total average mean of the reputations provided by the majority of the learners. A second approach is to adopt the reputations of the “most relevant” learners, according to the PAPI:Performance or PAPI:Certification property. It is predicted that if the average reputation of all learners is adopted, the learner's satisfaction from the retrieved LR will be augmented as the system proposes the LR that were positively evaluated, and the description metadata will fit better with the content of the resource according to the opinion of the majority.

Heuristic rules have been embedded in our framework. These rules intend to improve the satisfaction of the learner of the proposed LR in the specific topic “Multimedia”. They are in accordance to the criteria being chosen.

They combine properties of the learners to those of the LR, as shown below:

Learner properties	Learning Resource properties
UserRole	→ InteractivityLevel, SemanticDensity, Difficulty, LearningResourceType
IntendedEndUserRole	→ Interactivity Level, Interactivity Type, Format, LearningResourceType
LearningStyle	→ InteractivityLevel, InteractivityType, Format

Numerous rules can be created by combining these criteria. Some examples are:

*Rule 1:* If the learner is “Beginner” (hasUserRole = “Beginner”) then propose LR that have “InteractivityLevel” >= “High” or “SemanticDensity” <= “Medium” or “Difficulty” <= “Low” or LearningResourceType = “Tutorial” or LearningResourceType = “Simulation”

This rule is scalable according the learner's level (Intermediate, Advanced) and the properties of the LR e.g. the Difficulty of the LR (from Medium to Very Difficult).

*Rule 2:* If the learner hasIntendedEndUserRole = “Trainee” then recommend LR that have LearningResourceType = “Simulation” OR Format = “Video”.

*Rule 3:* If the learner has LearningStyle = “Active” then recommend LR that have LearningResourceType = {Exercise, Experiment} OR Format = “avi”.

Reputations may also become part of these rules, as follows:

*Rule:* Find LRs according to the learners' properties or according to the properties that the learners said that the resources have. E.g:

Find LRs that have InteractivityLevel="High" OR LRs that their average value of the analogous reputation metadata property is high (e.g hasReputation-ForTheInteractivityLevel $\geq$  0.66)

#### 4. Recommending

The recommender-module of the proposed framework exploits both description and reputation metadata to propose personalized learning resources. It functions according to the following algorithm:

*Step 1)* search method: using description metadata (formulas 4 and 5)

If  $c_i$  wants  $l_j$  having  $sim(v, l_j) = 1$  in the property  $p$ , then find for him every  $l_k$  having  $sim(v, l_k) = 1$

*Step 2)* recommendation method using description metadata: apart the resources the system have found using the previous step, recommend "similar" LRs to "similar" learners, based on the attributes the learners have (formula 4) and their previous preferences,

If  $f(s) = sim(c_i, c_j) = 1$  and  $c_i$  wants  $l_j$  having  $sim(v, l_j) = 1$ , then propose to  $c_j$  the same set of  $l_j$ .

*Step 3)* recommendation method using reputation metadata: Find out LRs according to what similar learners have said about "similar" LRs (formulas 4, 6).

The system asks: "Do you want LRs that other learners said that are similar to those you are looking for?"  
if "yes" then:  
if  $c_i$  wants  $l_i$  having value  $v$  of a property  $l_{ip}$  then find out all  $l_i$  having  $|v - f(er)| = 0$

*Step 4)* Find out LRs, according to what the system predicts about similar learners (using formula 3).

Ask the learner "Do you want LRs that are quite suitable to your criteria?" If yes then:  
if  $c_i$  wants  $l_i$  having value  $v$  of a

property  $l_{ip}$  then find out all  $l_i$  having  $|v - f(ir)| = 0$

*Step 5)* If neither of the previous steps have retrieved enough resources then find out LRs that have an explicit reputation close-enough to the desired value, using a gradually augmented error  $e$ .

1.  $|e| = 0$
2. Ask the user: Do you want LRs that perhaps are suitable to your criteria?
3. If yes then
4.  $e = e + e_{step}$
5. if  $|v - f(er)| = |e|$   
return  $l_i$
6. Ask: More?
7. If yes then go to 4

After each run of the loop collect new reputations. Use this feedback to find out if the choices of the system were successful. If the reputations of the learner turn out to be negative, then stop retrieving new learning resources. While the LRs are evaluated positively, then turn the approximate values into implicit reputations and the implicit reputations into explicit ones. This step gives a chance to the rarely ranked resources to collect reputations.

*Step 6)* Use predefined rules to recommend LRs.

Ask the learner "Do you want LRs that the system predicts that will satisfy you?"

As we can see the reputation metadata offers a second chance to the LR to be found –it creates a set of meta-metadata upon the LRs. Moreover, in case that no sufficient description for the LR has been given, the explicit and implicit reputation metadata may aid to its retrieval.

A system has been developed to implement this framework. The rules have been defined by using FLORA [17], an F-Logic version, a knowledge representation language. The Java 3-tier application that was developed is based on KAON –the Karlsruhe Ontology and Semantic Web Framework– data structure. The mediators implemented to compute the accumulated formulas are in accordance with KAON's database structure. The system was evaluated according to its capability to propose relevant LOs to the learner's profiles, by using Accuracy/Recall metrics [18].

1.175 LRs were inserted to evaluate the system. These LRs accepted 4.520 RDF-descriptions. 13 learners were registered into the system. The results of the evaluation can be found in fig 2.

In X-axis of this figure the 13 users are presented. In Y-axis their evaluation about the system's Accuracy is presented. The 1st-column depicts the average mean of each user evaluation concerning the relevance of the documents returned by the system in the case where only description metadata was used. The 2nd-column shows their evaluations about the relevance of the retrieved learning objects using reputation metadata as well.

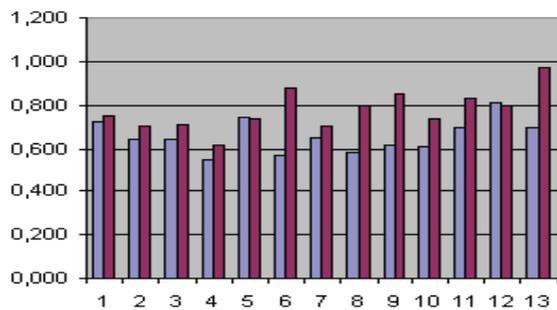


Fig. 2: Evaluation of the system

## 5. Conclusion-Future work

In this paper reputation is used in semantic web-based e-Learning systems as a mechanism that recommends personalized learning resources. The learners' feedback is used to describe more suitable the LR's metadata by recording the opinions of the majority of the learners for a learning resource. The experimental implementation of the proposed recommendation system revealed a number of benefits regarding the improvement of the personalization of learning resources. It was proved that the use of reputation metadata augmented the learner's satisfaction from the result sets; using evaluation-criteria, the system retrieves those LR's that were evaluated positively. The reputation set of metadata matches the learning resources to the searching-criteria more accurately. A system based on such metadata finds LR's that content-based systems perhaps would fail to retrieve.

In our future plans the implementation of a reward method that encourages the learners to provide reputations is being worked out. The automatic extraction of RDF-expressions from the textual annotations is a great challenge.

## 6. References

[1] P. Dolog, N. Hense, W. Nejdl, M. Sintek "Personalization in Distributed e-Learning Environments", www2004, New York, USA, ACM 1-58113-912/04/0005, 2004

[2] Yang Jin-Tan D, Chiu C-H, Tsai C., Wu T-H., "Visualized Online Simple Sequencing Authoring Tool for SCORM-compliant Content Package", Proceedings of the IEEE International Conference on Advanced Learning Technologies, 2004

[3] P. Resnick, H. Varian, "Recommender systems". Communications of the ACM 40, p. 56-58, 1997

[4] Kun Hua Tsai, Ti Kai Chiu, Ming Che Lee, Tzone I Wang: "A Learning Objects Recommendation Model based on the Preference and Ontological Approaches", Proceedings of the Sixth International Conference on Advanced Learning Technologies, IEEE, 2006

[5] X. Ochoa, E. Duval, "Use of Contextualized Attention Metadata for Ranking and Recommending Learning Objects", CAMA'06, ACM 1-59593-524-X/06/0011, 2006

[6] M. Claypool, P. Le, M. Waseda, D. Brown, "Implicit interest indicators", In Proceedings of ACM Intelligent User Interfaces, 2001

[7] A. Bouzeghoub, B. Defude, S. Ammour, J.F. Duitama, C. Lecocq, "A RDF Description Model for Manipulating Learning Objects", IEEE International Conference on Advanced Learning Technologies, Finland, 2004

[8] R. Farzan, P. Brusilovsky, "Social Navigation Support in a Course Recommendation System", International Workshop on Applications of Semantic Web Technologies for E-Learning (SW-EL), In conjunction with AH'06 2006

[9] S.F. Pan, J.H. Lee, "eDAADe: An Adaptive Recommendation System for Comparison and Analysis of Architectural Precedents", V. Wade, H. Ashman, and B. Smyth (Eds.): AH 2006, LNCS 4018, 2006, pp. 370 - 373

[10] T. Kerkiri, A. Manitsaris, I. Mavridis, "E-Reputation Ontology for Adaptation in E-Learning", International Workshop on Applications of Semantic Web Technologies for E-Learning (SW-EL), In conjunction with AH'06, 2006

[11] C. Dellarocas, P. Resnick, "Online Reputation Mechanisms - A Roadmap for Future Research", Summary Report of the First Interdisciplinary Symposium on Online Reputation Mechanisms, April 26-27, 2003, Cambridge, MA, 2003

[12] S. Bateman, C. Brooks, G. McCalla, "Collaborative Tagging Approaches for Ontological Metadata in Adaptive E-Learning Systems", International Workshop on Applications of Semantic Web Technologies for E-Learning (SW-EL), 2006

[13] O. Lassila, R. Swick, W3C Resource Description framework (RDF) Model & Syntax Specification, <http://www.w3.org/TR/REC-rdf-syntax>, 1999

[14] IEEE Learning Technology Standards Committee. IEEE LOM Working Draft 6.1. <http://ltsc.ieee.org/wg12/index.html>, 2001

[15] IEEE P1484.2.1/D8, 2001-11-25 Draft Standard for Learning Technology Sponsored by the Learning Technology Standards Committee of the IEEE Computer Society

[16] R.Felder, L.Silverman, "Learning and Teaching Styles In Engineering Education" Journal of Engineering Education, Vol. 78, No.7, 1988, pp. 674-681

[17] [http://www.ontoprise.de/documents/tutorial\\_flogic.pdf](http://www.ontoprise.de/documents/tutorial_flogic.pdf)

[18] G. Salton, M. E. Lesk, "Computer evaluation of indexing and text processing", Journal of the ACM, 15(1): 8-36, 1968