

TechScreen: Mining Competencies in Social Software

Jürgen DORN and Martin HOCHMEISTER

Institute of Software Technology and Interactive Systems, Vienna University of Technology,
Favoritenstrasse 9-11, A-1040 Wien, Austria,
email: juergen.dorn@ec.tuwien.ac.at, martin.hochmeister@gmx.at

ABSTRACT

This paper describes an approach to mine for competencies of users' in a social software system. The system supports knowledge sharing in a structured way supported by social tagging and an ontology. This meta-data is used to find relevant knowledge sources, cluster these resources and to recommend certain resources to users. Our research assumption is, that if a user enters knowledge about a certain domain, s/he is in a certain degree competent in this area. In our experiment we generate competence profiles of users and compare it with a competence profile obtained by a self-assessment of users. In this paper we describe the performed experiments at the university where students and university staff use the system to share knowledge.

Keywords: Competence Management, Social Software, Competence Mining, Semantic, Social Tagging

1. Introduction

Technological knowledge on Internet and related technologies is rapidly growing. New standards by the W3C and other standardization bodies are published regularly, new open source components of frameworks are developed by Apache Software Foundation and others continuously and many software artefacts are provided in repositories such as the Free Software Directory of Free Software Foundation (<http://directory.fsf.org/>) or Sourceforge (<http://sourceforge.net/>). Also new releases of standard software by the large software vendors are offered in short cycles. Many small companies and organizations struggle to be able to use adequately latest technologies demanded by their customers. Open-source content management systems, AJAX (Asynchronous Javascript and XML), Web services or technologies for accessing Web pages by disabled persons are recent challenges for developers. Although there exist considerable knowledge in the Internet, usually there is not enough manpower to evaluate the different technologies and to decide which technique is mature and can be used in a certain environment.

TechScreen is a project that tries to develop a community of users sharing knowledge (especially experience) about Internet technologies. To support this community a social software system is developed. The following aspects are analysed in the project:

1. How is the knowledge to be organized so that members find available knowledge easily?
2. How is the knowledge collected and interrelated?
3. How are users motivated to generate content and to share it with other members?
4. How are companies and organizations motivated to share knowledge?

One application scenario for TechScreen is a university with different faculties and institutes. Usually each institute has a small group of persons (usually technical personal) interested in Internet technologies and trying to build up good information portals for students and staff members. The sharing of knowledge between different institutes will be successful, if the access is easy and contains relevant knowledge. One support to generate a worthy knowledge base is to have many master and Ph.D. students working with these technologies. These students may be obliged to share their knowledge.

Besides documenting knowledge in the system explicitly, the system is used to find other users with certain experience. If a student works on a master thesis where certain technical problems are to be solved, s/he may search for other students having already made this experience. In the following, we focus on the mining of user's competences out of the contributions a user has made in the system.

We present related work and some basic concepts of our approach in the second section. Then we describe the design of our system and the experiments made with the first prototype of the system. Finally we will conclude with a discussion of future extensions.

2. State of the Art

The term social software was created only recently, however, applications that follow this paradigm are much older. Due to different reasons there is some hype about these applications now. Thus new start-up companies offering such information systems achieve a very high financial rating through their large number of users and the large body of information. However, this is only one group of social software that achieves very high volumes of users. Social software is also used to build smaller communities with a restricted access. Thus a company may invite its customers into such a community for online support on products and services of the company. Social software is also used to support knowledge exchange between employees of companies [1].

The success of social software depends on the motivation of its users. Either the number of participants must grow very large or users must be highly motivated to supply information into the system to attract with this information other users. Communities are built around common interests. Often, it is however unclear what the common interests are and whether all community members share the same interests.

This problem is also investigated in knowledge management theory. A company should be interested that information is shared between its employees. Knowledge management systems have to be designed in such a way that users are motivated to share relevant knowledge. Davenport and Prusak [2] describe three motivations that lead to successful knowledge sharing:

reciprocity (if I submit something then also other community members are obliged to share information), reputation (if I submit much information I will be accepted as an expert) and altruism (I want to support this highly relevant community without any immediate benefits). These motivations are also valid for social software applications.

For user acceptance of such systems the ease of use is also very important. Technological improvements such as simple interactive Web interfaces accessible with most Internet browsers lower the barrier because users do not have to install software and to learn new interaction techniques.

Providers of social software systems must also get some revenue to provide services. The number of users is most important for financial evaluation of such companies. Google, for example, is the company having one of the largest user groups at the moment. Thus, a provider is motivated to attract as many users as possible with good services. One of the most important services in the Internet seems to be the provision of relevant information. And the easiest way to provide such information is to let the users create the content. Other options are automated creation of text by harvesting the Web with crawlers, search engines and mining techniques.

Social or collaborative tagging is a process where a number of users (the members of a community) assign tags (key words) to certain (Internet) resources. Users are typically free to assign such terms to resources and the set of tags used in a community is called folksonomy. A folksonomy is formally a hypergraph containing three types of nodes related by assignments. A folksonomy $F = (U, T, R, Y)$ is a quadruple where U is the set of users, T the set of tags, R the set of resources and Y the set of assignments where one user relates one resource with a tag. Often tag sets are used to characterize one resource. Although users are free to use such tags, investigations have shown that at least in large communities some convergence in using common tags can be seen [3]. Additional means to improve this convergence are proposed in order to support further reasoning processes. One of this means is to apply ontologies defined by experts. In this case users can either assign terms defined in the ontology to a resource or resources are assigned by means of text mining algorithms to certain concepts of an ontology.

If a certain grade of reuse (using same tags, tagging same resources and users that regularly tag resources), a social network analysis can be used to detect interests of users, clustering of users with similar interests and recommending resources to users [4]. For some investigations (e.g. trends in interests) the time when the tagging has taken place must be considered, too.

Competence management as part of knowledge management is a recent trend in enterprises to organize the development and recruitment of the work staff better. To support competence management, many companies use competence management systems implemented on different abstraction levels. Competence management systems have to fulfil several requirements concerning the content (how detailed the competency model is developed), the technical implementation (which data is kept when and how data is kept up to date) and the organisational implementation (who implements the system, how are people motivated to use the system and to keep data up to date). In [5], an overview of existing competence management systems is given. In the following we sketch only three examples.

Microsoft Corp. implemented a competence management system by defining more than 300 competencies in four categories

(foundation, local and unique, global, and universal skills). Basic, working, leadership, and expert are the terms for different competency levels. Staff members are rated by themselves and by their superiors within an iterative process. Microsoft expects better matching of employees to jobs and work teams. Moreover, they expect that employees will know better what competencies are required and thus are better consumers of educational offers [2].

Before introducing a worldwide competence management system, *Ericsson* had individual solutions in each country, sometimes paper-based or based on databases or spreadsheets. They extended the existing SAP R/3 Personal Management Module and implemented a competency catalogue. For the grades there exists a scale from 0 (not assessed) to 9 (excellent). The values are evaluated by using the period of time a person has used the competency [6].

In [7], a system is described based on an ontology defining competencies, evidence types and jobs in information systems and computer science domain. The system distinguishes competencies in knowledge and an experience aspect. To decide on a competence, evidences such as passed examinations, books read by the person, trainings, assessments, project work, e-learning courses and more are stored. Each competency can be evaluated on three levels: beginner, advanced and expert and on each of these levels a finer scale exists with real values between 0 and 1. The competence of a person is computed by the system at the point of time when it is required and all evidences until this moment are used for the computation.

In this paper, we propose a knowledge-based information system, which measures the competencies of its users in an automated manner. The sources for assessments are the textual contributions written by users. Basically, the idea to quantify competencies is not new. [8] introduces a methodology, where competence identification mainly requires human input that provides interpretation. Recently, systems have been developed, which gain leverage from semantic technologies and machine learning mechanisms and therefore strive an almost entire automation of measurement process.

Oliveira et al [9] have deployed a knowledge management system to support scientific communication within research centres and universities. An essential part of this approach is a competence-mining module determined to measure competencies out of different publication types. These publication types may be project definitions, blog posts, emails, personal web pages, et cetera. For mining competencies the system uses text-mining algorithms in conjunction with a taxonomy. A knowledge engineer manually maintains the taxonomy. Besides text extraction from documents, the knowledge management system gathers additional information about the interests of users as well. This is achieved through a Web mining facility. Interests may also indicate some degree of competence in a certain environment.

The eCompetence management tool by [10] enables conventional users to co-manage the systems ontology via a graphical user interface. The authors argue this functionality with the fact, that evolution of competencies is faster than its formal codification. The competencies are identified through text extraction methods. Each competence is represented by a collection of skills. Depending on how much skill items are covered by a specific user. This evaluation algorithm is applied recursively throughout the ontology and finally measures a percentage value that represents the degree of a certain competence.

The integration of human resource processes with ontologies is carried out by [11] in a project at DaimlerChrysler AG. The results of process analysis are structured and modelled in a competence catalogue that represents knowledge over all areas of production, management and administration. Actual skill profiles of employees are documented by hand along the same ontology structure. In addition, competencies are automatically measured due to basic inference rules like if someone attends a training course, it is said that s/he has afterwards a competence of a specific degree.

3. Design

The TechScreen social software system is designed with a focus on a flexible integration both from a user/organizational perspective and from a technical perspective. Individual organizations can use the system as a stand-alone service as well as a tool to share knowledge between organizations. From a technological view, we enable the flexible integration of new components and technologies for knowledge management. In the following we first describe the organizational perspective, then the technological and finally the approach to competence management. Fig. 1 gives an overview of TechScreen whose components are described in the following subsections.

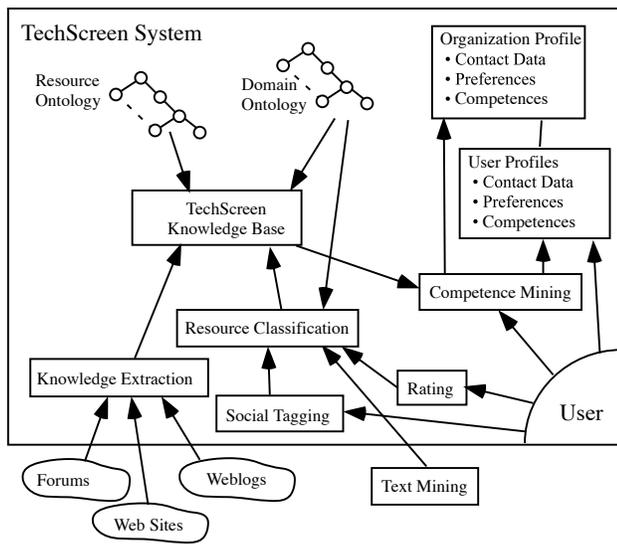


Fig. 1: Architecture of TechScreen

3.1. Management of Users and Organizations

The objective is to attract users to supply information and to maintain their profile. This shall be achieved by providing better services to registered users. Thus registered users have better services than anonymous. For a registration at least a legal email address, a nickname and a password are required. Phone numbers and other contact information is encouraged to support also communication outside of the system. The name of one or more organizations with which a user is somehow associated, can be supplied, but to achieve access to information restricted to a certain organization, the organization’s administrator has to approve this affiliation. A user can specify interests and competence fields. The following services are available for registered users:

1. A user can specify filters in his profile to show in which information s/he is interested and how s/he wants to be alerted on new relevant content.

2. A user can generate a competence profile showing his competencies based on his/her profile specifications, submissions, read articles and taggings. Thus other users may find this user as expert in certain domains.
3. The user may specify to which extent and in which form he is willing to share his knowledge based on his competence profile.

The system shall support organizations’ participation. On one side the system may be used as a knowledge management and sharing tool inside of an organization to find experts in certain (technical) domains and to find information about certain topics. Moreover, organizations shall be motivated to share such knowledge with other organizations and their members. Organizations may also pull knowledge out of the system. The following services shall motivate organizations to participate:

1. A member of an organization may find a competent person in a certain competence field outside of the organization.
2. An organization may generate competence profiles generated from submissions of their members and from explicit mentioned competencies.
3. An organization may pose problem statements in TechScreen to be solved by the community.

An organization may be represented with subunits supporting complex management and delegation of privileges. Each organization needs an administrator deciding on whether certain users are members of the organization.

3.2. Knowledge and Information Resources

The system stores explicit information/knowledge in different resource types. Moreover, humans and organizations working with the system are interpreted as resources with implicit knowledge. A search for knowledge may return an explicit resource as well as link to a person/organization that has the required knowledge or expertise. The management of this “implicit” knowledge is described later. The explicit knowledge can take different forms: it may be some explicitly submitted text (a contribution), a document uploaded in the system or a reference to a knowledge artefact outside of the system (a knowledge sharing event, software, or a link to an Internet resource). A document can be a scientific article, a standard, a manual, a tutorial, a white paper or a presentation. Documents are stored in the system in PDF-format.

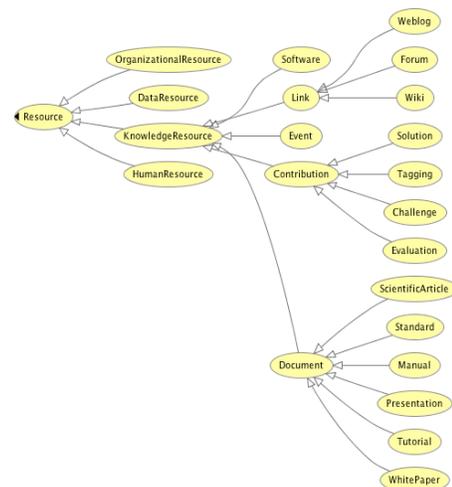


Fig. 2: Resource Ontology of TechScreen

The resource types are stored in different “templates” defined by an ontology. The concept “knowledge resource” has some basic attributes inherited by all children. A knowledge resource has a “created” time attribute and may have a “valid-until” attribute describing the lifetime of a resource. Each knowledge resource has a “creator”; either a user of the system or a software agent that has entered the resource into the system. The creator may define an individual confidentiality level to the object. Per default a resource is visible by all members. If the creator’s organization has set a stronger confidentiality level, then this level is the default. However, the creator may assign an individual level to the object.

One child of a contribution concept is a “challenge” object. An instance of such an object contains individual problem descriptions such as for example “setting up a directory service on a server”. There may be complex challenges composed of smaller problems. Further, we distinguish whether such a challenge needs a new solution or a simple adaptation of an existing solution which is determined by the community. The next child of the root element is a “solution”. Challenges and solutions may be assigned to each other. Thus there may exist different solutions for a problem and a solution may be applied to different problems. A solution can consist of different steps which again can be solutions to smaller problems.

Solutions and solution steps may reference documents. The concept “Software”, with further children, is also a “knowledge resource” having certain attributes and references to download locations. Educational events disseminating knowledge related to domains in the TechScreen system is a further child concept of. The child concept “links” can be used to reference further knowledge outside of the system. Here again different child nodes are defined for blogs, wikis, forums and more.

Besides this knowledge resource type ontology, a second domain ontology is defined. In our first system this ontology describes Internet related concepts such as “Web server”, “XPath” or “DNS”. We could exchange this ontology to share knowledge in a different domain.

3.3. Knowledge Indexing

Each knowledge resource entered into the system obtains meta-data describing the resource. Some descriptions such as the creation time or the creator are assigned automatically. In addition, a user can assign individual terms characterizing the resource. Each of these terms is a quadruple resource / user / tag / time stored. As already mentioned these user supplied tags serve as a secondary meta-data description of the knowledge resource, allowing for a more detailed specification of the stored knowledge. Since the tags of a folksonomy constitute a highly dynamic and user-centred vocabulary, they can succeed in describing complex and rapidly changing domain details (like individual technologies). While they lack explicit relations and semantic expressiveness, they provide users with an emergent vocabulary that is best suited for the users’ needs at a given time. Given a number of such terms we try to conclude whether a submission can be assigned to one or more concepts of our domain ontology.

3.4. Competence Evaluation

In another project, we developed an information system to model student competencies supporting the explicit planning of student’s competencies [7]. Within this system a student may compare his/her actual competencies with some goal profile to

determine which lectures, projects or other experience would help to reach a certain goal competency profile. Especially the development of soft skills is considered in a university study today only seldom. We focus on documenting and assessing such competencies in existing courses. Students’ competency profiles can also be used to support the recruiting in companies if these companies have comparable competence management systems [7]. Moreover, such competence management may support the staffing in inter-organizational projects [12].

The competencies are defined in an ontology and for the computer science and the information system study plan we have defined about 140 different competencies in a hierarchic structure. The semantic meaning of this hierarchy is that an upper competency contains the lower competency. Business administration, information systems, computer science and mathematics are competencies defined immediately below the root element. Web Engineering is a competence defined as child concept of information systems and computer science with a knowledge and an experience aspect. Thus a student may achieve a certain degree of knowledge in a lecture and by completing a project in this domain s/he may also achieve a certain grade of experience. However, from this measurement we do not know whether the student has experience in managing a certain application server, for example. Since the content of such university courses changes very frequently and new knowledge and systems are generated continuously, an ontology cannot always represent the latest state of fine-grained competencies.

In the TechScreen system the competence of members shall be measured, too. The vision is that the system will be integrated with the competence management system. However, at the moment only experiments how far the measurement of competencies based on the participation in the system is evaluated. In the existing competence management system, a person may assess itself or a supervisor or peers measure its competencies [13]. With TechScreen we want to measure competencies in technologies that evolve rather fast.

The different contribution types (see Fig. 2) obtain individual weights representing the relevance for the competence calculation process. For example, if a community member has solved a problem it is more likely that this member has a competence than that user who only tags the solution. Basically the calculation is divided in two phases, the knowledge resource identification and classification phase and the competence measurement phase.

In order to identify and classify the content of a knowledge resource, the system analyzes the text corpus of a contribution by text mining, ontological reasoning and social tagging and tries to assign the knowledge resource to one or more concepts of the domain ontology. Such a goal concept is interpreted as a competence area.

A users contribution is supposed to be rated by community members. The rating refers to the integrity and structural quality of an article. The calculation process incorporates the rating information by increasing or decreasing the competence value. If an article has a significant low rating it is excluded from calculation. Otherwise the competences identified are multiplied with the corresponding value.

The system evaluates each submission of a community member. Taggings of the resources are names of potential competencies. If a number of community members have tagged resources with

the term “application server”, then this becomes a certain fine-grained competency. If the same resources were also tagged with the ontology concept “web engineering”, the system assumes that “application server” is a competency that is part of “web engineering”.

A community member may have submitted different resources having assigned such a tag. Then we compute an aggregated value from these resources assuming that the member has competency in “application servers”. Different types of resources have different impact on the computed value and moreover, the evaluation of a resource by another person can change the worth of one’s own submission. For example, someone may have rated an article as very good. Each submission may have influence on different competence areas.

Based on all submissions we may now compute a profile describing the competence in different areas. This profile can be used to find persons having certain competencies.

4. Implementation

The first prototype was implemented with an Apache Web server, MySQL database management system, PHP and Drupal content management system. Further the authorization server of Vienna University of Technology is used for authorization and user management and Protégé with OWL is used for the ontology. We use further the text mining service of Yahoo.



Fig. 3: Recent tag cloud of TechScreen

5. Experiment

In our first experiment we compared students’ competences measured by self assessment with competences mined from submissions. Since all students study at the computer science faculty of the Vienna University of Technology, they are supposed to have at least a basic knowledge in Internet technology. The experiment was conducted within last winter term with 31 participating students.

The students were encouraged to provide problems related to Internet technologies they have faced in everyday life. In most cases, the problems were initiated through challenges the students met during their study. Each problem requires a solution which is either supplied by the author or by an other community member. Besides that, users have tagged problems and solutions, submitted comments to topics they are interested in and have rated the integrity and structural quality of submissions. Out of this captured data, the system calculates individual competence profiles. At the end of the term, each participant was asked to provide feedback by confirming respectively negating the calculated competence values. The overall goal of this first experimental step was to evaluate

components of the calculation process and the readiness of community members to use the system. As a result we expected a first proof of the correctness of the competence mining as well as hints for improving the calculation process and the user interface.

The competence calculation determines one or more competence subareas of Internet technologies (Web engineering) a user may be competent in. If a user shows no competences at all, no competence area will be assigned to the users profile. The abstract competence areas are defined by the idea to find the most vital competences a Web engineer has to hold for professional activities. The competence areas are represented within an ontology. We understand the term competences as a collection of skills. That means a user has to have at least two skills of a competence field to be basically considered competent.

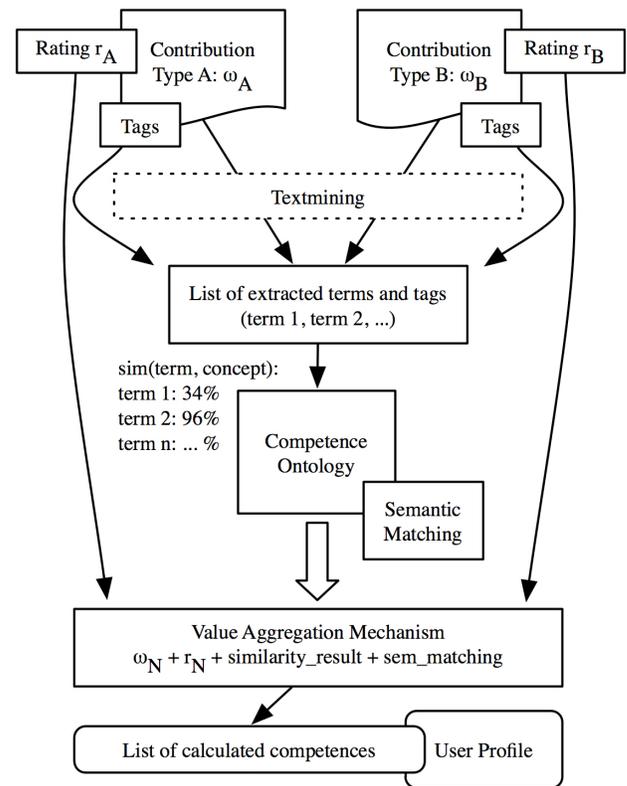


Fig. 4: Calculation of Competences

We started with a base ontology defined in a prior project. By comparing this ontology with used tags and text mining results, we extended the ontology. The term list for every article is processed along the following criteria:

- 1.) Discard all terms that are definitely out of context
- 2.) Discard all terms that are actually Internet related but formulated in a too common sense
- 3.) Group the terms and determine synonyms
- 4.) Categorize the terms with the aid of Wikipedia and expert knowledge
- 5.) Finally integrate the terms with the base ontology

The competence calculation process starts with collecting all contributions of a certain user. After that a term list is created

by the means of text mining services and the tags stored for each article. All the terms are looked up in the ontology and are assigned to a specific competence area (an ontology concept). At the moment we only concentrate to find exact matches. Finally we get an amount of skills for each competence area. The area holding the highest score of skills receives the maximum weight. All the other areas are weighted in relation to this area. The statement of this measure is that it determines the competence area where the user has its core capabilities followed by a gradually list of less marked competences.

The user feedback showed us an average match of 53 % without eliminating potential outliers. We think that this low percentage will be considerably increased through the introduction of similarity measures, the growth of ontology in quantity and quality aspects, the use of semantic matching heuristics and last but not least through incorporation of contribution weights and user ratings. Problems in data interpretation may also be caused through the relatively short length of many text corpora. The method of evaluation through self-assessment seems generally appropriate, since the direct involvement of users leads to an increased motivation to participate actively. The following table summarizes the experiment:

Table 1: Experimental result

Participants	31
Terms in Base Ontology	588 (including synonyms)
Contributions submitted	92 Problems
	101 Solutions
	65 Comments
	453 Tags
	160 Ratings
Feedbacks submitted	23
Average Matching Percentage	53,44 %

6. Discussion

We have described a social software system supporting the exchange of knowledge between community members. We distinguish knowledge that can explicitly stored in such a system and tacit knowledge existing only implicitly in human resources or in the organizational memory. The knowledge is indexed by automated techniques, user-initiated tagging and by assignment to concepts of an ontology. These indexing techniques support also the finding of knowledge resources, i.e. human experts, organizations or documents relevant to certain problems.

The indexing of explicit knowledge is more or less standard in many community systems. However, the indexing of implicit knowledge and competencies is new. We believe that this increases the motivation of users to supply knowledge because as a consequence other users may recognize their status as experts in a certain competence area.

We have performed first experiments with the system showing that we can mine competencies from the submissions, but it is not yet proved that the motivation has risen. A next step is semantic matching of in hierarchies of competence areas. In the moment we mine only technical competences, but in the long run we envision also the mining of social competences that can be derived from users' behaviours in the system.

7. References

- [1] Wenger, E. (2004). *Communities of practice: learning, meaning, and identity*. Cambridge et al.: Cambridge University Press.
- [2] Davenport, Th. H. and Prusak, L. (1998) *Working Knowledge – How Organizations Manage What They Know*, Harvard Business School Press.
- [3] Golder, S. and Huberman, B. A. (2006) Usage Pattern of Collaborative Tagging Systems, *Journal of Information Systems* 32(2) 198-208
- [4] Schmitz, Ch., Grahl, M., Hotho, A., Stumme, G. Cattuto, C., Baldassarri, A., Loreto, V. and Servidio, V. (2007) Network Properties of Folksonomies, *Proceedings of the WWW 2007, Banff, Canada*.
- [5] Draganidis and Mentzas (2006) Competency based management: a review of systems and approaches in *Information Management & Computer Security*.
- [6] Hustad and Munkvold (2005) IT-Supported Competence Management: A Case Study at Ericsson in *Journal of Information Systems Management*.
- [7] Dorn, J. and Pichlmair, M. (2007) A Competence Management system for Universities, in *Proceedings of European Conference on Information Systems, St. Gallen, 759-770*.
- [8] Ley, T. and Albert, D. (2003) Identifying Employee Competencies in Dynamic Work Domains: Methodological Considerations and a Case Study. *Journal of Universal Computer Science*, 9, 1500-1518
- [9] Oliveira, J., de Souza, J. M., Miranda, R., Rodrigues, S., Kawamura, V., Martino, R. N. D., Mello, C., Krejci, D., Barbosa, C. E., Maia, L. Zhou, X., Li, J., Shen, H. T., Kitsuregawa, M. and Zhang, Y. (2006) GCC: A Knowledge Management Environment for Research Centers and Universities. *APWeb, Springer, 3841, 652-667*
- [10] Pernici, B., Locatelli, P. and Marinoni, C. (2006) The eCCO System: An eCompetence Management Tool Based on Semantic Networks. *OTM Workshops (2), Springer, 4278, 1088-1099*
- [11] Biesalski, E. and Abecker, A. (2005) Human Resource Management with Ontologies. *Springer Postproceedings: Workshop on IT Tools for Knowledge Management Systems: Applicability, Usability, and Benefits (KMTTOOLS), 499-507*
- [12] Dorn, J. (2006) Competence Management in Virtual Enterprises, in *Proceedings of XML-Tage, Berlin*.
- [13] Dorn, J., Pichlmair, M., Schimper, K. and H. Tellioglu (2008) Supporting Competence Management in Software Projects, in *Proceedings of International Conference on Concurrent Enterprising, Lisboa, pp. 451 - 458*

The TechScreen system can be accesses at <https://techscreen.tuwien.ac.at>