Semantic-based Groupware System in SAKE for Support of Knowledge and Expert Intensive Public Administration Processes

Peter Butka
Centre for Information Technologies
Technical University of Košice
Boženy Němcovej 3, 040 01 Košice, Slovakia
Peter.Butka@tuke.sk

Gabriel Lukáč
Faculty of Economics
Technical University of Košice
Boženy Němcovej 32, 040 01 Košice, Slovakia
Gabriel.Lukac@tuke.sk

Abstract. *In this article we propose a Semantic-based Groupware System (GWS) for the SAKE project. SAKE (Semantic Agile Knowledge-based E-government) is a STREP Project sponsored by the European Union starting in March 2006. The overall objective of SAKE is to specify, develop and deploy a holistic framework and supporting tools for an agile knowledge-based e-government that will be sufficiently flexible to adapt to changing and diverse environments and needs. All public administration processes are knowledge-intensive and influenced by experts in specific domains. The core of the whole SAKE system is represented by an integrated knowledge space unifying different perspectives and interpretations of knowledge resources. It enables for each knowledge object to be assigned metadata allowing more sophisticated retrieval and use. A goal of our groupware system is to support sharing of their knowledge using collaborative software as well as to help them in managing and creation of experts group, which are frequently used in governmental and self-governmental internal administration processes. One of the main contributions to the semantic enhancement of the system, which will be described in this paper, is based on a method for discussion forums analysis and computing of ranking of users according to the actual annotated discussion topic.*

Keywords. groupware, public administration, discussion analysis, knowledge-based systems

1 Introduction

At present, in many areas of public administration (PA) frequent changes occur and a large amount of new information has to be absorbed in a very short time. The public administration processes are influenced by external changes (mainly political, legislative and economic), as well as internal changes (e. g. organizational).

A change in one activity in an administrative process may require changes in other parts of the same process or system. Therefore, there is a need for resolving changes in a systematic manner, ensuring overall consistency.

Furthermore, these changes impose the need of updating the knowledge needed to perform the administrative process. These changes are more frequent in the case of new member states, since their full integration heavily depends on the possibility to adapt their public administrations to the existing EU regulations in a very short period of time.

SAKE (Semantic Agile Knowledge-based E-government) is a three-year IST Project (STREP) co-funded by the European Union, which started in March 2006. The overall objective of SAKE is to specify, develop and deploy a holistic framework and supporting tools for an agile knowledge-based e-government that will be sufficiently flexible to adapt to changing and diverse environments and needs. Existing approaches for knowledge management in e-government focus mainly on the efficient management of a particular, isolated knowledge resource and on supporting only message-based communication between public administrators. However, the demands for knowledge-based e-government are much higher [1]:

- First, the existing approaches do not take into account the increased granularity of informational resources and the manifold semantic differences in dealing with those resources.

- Second, due to complexity of the decision making processes, effective knowledge management requires the creation of a supportive, collaborative culture while eliminating traditional rivalries.
- Third, the usage of existing knowledge resources is indeed a valid aspiration, but for realizing a learning e-government, the crucial is creation of *new* knowledge.
- Finally, ad hoc management of the changes in e-government systems might work in the short term, but to avoid unnecessary complexity and failures in the long run, management must be done in a systematic way.

Whole SAKE approach will provide tools and methodologies to address these problems. More specifically, SAKE intends to provide [1]:

1. *Integrated knowledge space* instead of a set of isolated and heterogeneous knowledge resources;
2. *Collaborative working environment* instead of a single person decision making process;
3. *Attention (change) management system* instead of ad-hoc management of changes;
4. A platform for *proactive delivery of knowledge* (instead of an one-way knowledge access) that enables creation of an adaptable knowledge sharing environment through learning from the collaboration between public servants and their interaction with the knowledge repository and supporting in that way full empowerment of public servants.

In the aforementioned way, SAKE provides a framework for an agile knowledge-based e-government, which enables efficient satisfaction of "unpredictable" knowledge needs of public administrators in order to ensure high and homogeneous quality of the decision making process, especially in a highly changing environment.

In the second section of this article we will shortly describe overall SAKE architecture. Next we will present details regarding our component – the groupware system, especially one of the main contributions to semantic enhancement of system – the method for discussion analysis. This method leads to ranking of users regarding their previous communication activities within forums. It combines user's feedback and filtering of threads (according to annotation of messages in threads) and therefore provides 'argumentation-like' support solution which leads to feedback-based topic-sensitive ranking of discussion forums users. This information can be used by someone who wants to decide for invitation between public administrators (experts) in a new processed case.

2 SAKE overall architecture

Basing on the analysis of the actual state in back-office processes and on the need for applying semantic technologies, we have identified three main individual technological components (Figure 1) [1]:

1. *Semantic-based change (attention) management* ensures high quality of the knowledge update (reediting) process by developing a change management process that enables the consistent propagation of changes to every knowledge stakeholder in order to ensure quality of the decision making process, formal and explicit modeling of changes in public regulations and their relations to depending artifacts in the form of the Change Ontology, which will serve as the backbone of the change management approach, and developing methods and tools for verification of an existing knowledge repository in order to make it easier to understand and cheaper to manage without any loss of information content.
2. *Semantic-based content management system* enables efficient provision of knowledge in the context of a PA process by semi-automatic population of the Information ontology by using Text Mining methods (Ontology Learning). It develops methods and tools for ontology-based tagging, methods and tools for realizing context-aware searching for virtual content, and also methods and tools for editorial process, to satisfy the knowledge items evaluation requirements.
3. *Semantic-based Groupware system* supports more efficient knowledge sharing by developing methods and tools for ontology-based tagging the interaction between public administrators, methods and tools for enabling building community of practice from interaction log and their specific vocabularies by social tagging, methods and tools for collaborative knowledge creation, and methods and tools for pushing of knowledge and for searching for experts.

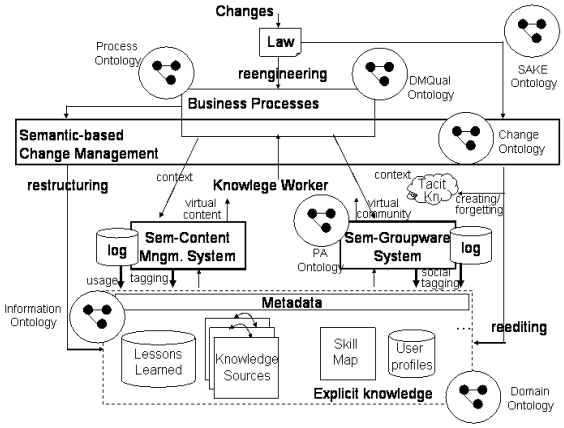

Figure 1. Architecture of SAKE system

SAKE also develops a conceptual framework for a semantic-enabled agile knowledge-based e-

government that will comprise an analysis of the knowledge infrastructure and knowledge sources in e-government. Parts of this conceptual framework are the ontologies. There are several ontologies developed for the SAKE based on purpose of these ontologies. *SAKE Ontology* that provides an overview model of all aspects relevant for achieving agile knowledge-based e-government and serves as the backbone of the approach, *Domain ontology* that models the terminology used in the e-government domain, *Information ontology* that models the different kinds of information sources with their respective structures, access and format properties, *Process Ontology* and *Profile Ontology* that model how an administrative process works and what is it about, respectively, semantic models of users (public administrators), their roles and skills (*PA Ontology*), and Quality model of the decision making process in the public administration (*DMQual Ontology*). Together with them, guidelines for estimating the quality of decision making processes, based on the user and quality model are provided.

Another important subsystem was identified after the first year of the project and it is (from the functional point of view) the core element of the business process management – *Workflow management system* (WfMS), which is fully developed and implemented using jBPM[1]. All other subsystems are tightly integrated using this workflow-based business management, which strongly supports business context management and sharing of the actual context. The subsystems log all of the user-actions semantically to provide a data-source for AMS that performs analyzes on these data and pushes new knowledge back to users.

According to previous information the SAKE solution consists of (at least) three individual components (AMS – Attention (Change) Management System, CMS – Content Management System and GWS – Groupware System). These three are supplemented by a more integration-related, Common Knowledge Space (ontology integration subsystem) and a Workflow Management System (WfMS). The platform is an integrated platform acting as a shell, where different kinds of problems may be introduced. User access to platform will be provided by a single point of entry to the system. Personalized access is required (authentication, authorization – roles). The User Interface (and the offered capabilities) depends on the roles.

The GUI components are decoupled from the business logic. Thus the AMS, CMS and GWS components need to offer an interface. The Portlets technology is used, which allows components to access the desired Business Logic. Also, the use of Portlets technology allows a better modularization and it facilitates the development of the user interface in a distributed team. Ideally portlets for the CMS, GWS and AMS can be put together in a "plug-and play" fashion because of well defined interfaces of the Portlet specification. The Common Knowledge Space consists of ontologies, which were described below. The Domain ontologies are based on native languages. To avoid problems with integration of multiple components it was decided to use a common development platform, in our case Java platform.

The decision in the SAKE project is to use J2EE/EJB for the implementation of AMS, CMS, GWS business logic (JBOSS is used as an Application Server). Portlet technology (JSR168) is used for the implementation of the User Interfaces provided by AMS, CMS, GWS. The SAKE Main Interface will be the 'Portal' which contains the 'Portlets'. The KAON2 system[2] with its API (wrapped in an EJB) is used for Ontology Management and Reasoning.

3 Architecture of Groupware System

The structure of our groupware system can be viewed in a sense of four basic layers with several components in particular layers (Figure 2) [2]:

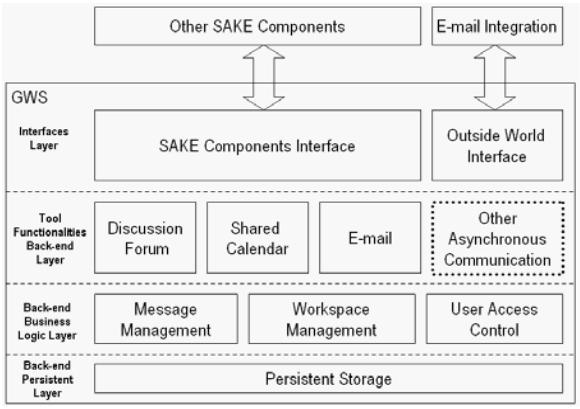

Figure 2. Global view of the GWS architecture

Back-end Persistence Layer – this is the back-end storage necessary for supporting GWS actual work and functionalities in storing different types of data. It is based on the persistent layer of an integrated open-source groupware project. Module named *Persistent storage* has two basic supported functions – persistent layer for data used by the original open-source product and possible place for storing of different types of actual data related to enhanced functionality of the tool. As a persistent middleware layer the Hibernate framework is used.

Back-end Business Logic Layer – the logic behind provided functionalities of the groupware component

1 http://www.jboss.com/products/jbpm

2 http://kaon2.semanticweb.org/

is covered by this layer, mainly management of a shared workspace, user access control to resources inside component and additionally needed management of message-based communication resources (processing of stored forums…). *User Access Control* is responsible for managing and administration of user access to groupware modules and tools. Access rights are then combined using access rights from the main interface and/or A/A (authentication-authorization module) and PA ontology. *Workspace Management* is used in a more intuitive way where the workspace will be the space for collecting people, files, discussions, calendar events important according to defined case (business context) to support their group-work during the business process. *Message Management* is a specialized back-end support component for preprocessing of message-based groupware data sources like discussion forums.

Tool Functionalities Back-end Layer – this is the layer responsible for back-end support of different tool functionalities which are provided to the users. An important fact is that provided functionalities have to cover use cases defined in user requirements and according to this they are required to cover tasks needs related to pilot processes. Different types of communication are provided in this layer like discussion forums, e-mail or other tools for communication as well as the shared calendar or file sharing, provided by corresponding modules.

Interfaces Layer – this layer is mainly responsible for integration with other SAKE components. This means interactions with main integrated User Interface, CMS, AMS and ontology repository as well as integration with external tools e.g. e-mail server. It covers connections to services provided by other components as well as services which are provided by GWS for other parts of the whole integrated system.

Implementation of the Groupware System (GWS) in SAKE is based on the open source solution named Coefficient[3]. This system has been selected after a detailed analysis as the best candidate. One of the problems of selection process was that we have not found any other proper solution supporting all of required functionalities. The most notable example is the shared calendar, which is now implemented with the usage of an external open-source server solution; concretely it is based on the Jakarta Slide WebDAV server[4] with iCal4j[5] Java library supporting the iCal[6] standard. A slightly complicated situation is solved by creation of the 'GWS Adaptor' which should be an abstraction layer for a combination of systems e.g. Coefficient, calendar server, mail server and so on. Through an adaptor the GWS component will be able to communicate with one or more systems realizing required functionalities defined by the adaptor interface. One of the benefits is that SAKE GWS system features (and whole SAKE, because CMS is using the same adaptor mechanism) could be provided also to an environment with existing groupware system. This will help in adopting it in different public administration sites.

Coefficient provides several collaboration functionalities. Basically it is a scalable project collaboration platform that can run in J2EE and web application containers. Coefficient uses the term *projects,* but in other contexts, projects are also called *groups, workspaces,* or *share spaces.* Coefficient functionality is driven by modules. Some modules, such as the project module are core of the system. Others augment the behavior of the system. *Project Module* serves as a workspace where individuals can collaborate. The project can be adorned with modules that enhance the functionality of a project. Coefficient currently deploys with the several add-on modules. *Mail Forum Module* implements a discussion forum based on the Dithaka[7] framework. The GWS component in SAKE extends these functionalities with an integrated approach including semantic logging, which provides semantic information to pieces of content; and integration with all other components of the system. Specific GWS functionalities are extended and enhanced regarding user, implementation, semantic and integration requirements to provide relevant functions in the scope of SAKE.

The portal-side shared calendar component was implemented from scratch as a part of the GWS component. To enable access to the shared calendar functionalities by external clients, Jakarta Slide WebDAV server was used providing a secured access to published iCal calendar files and the iCal4j - a Java library providing the methods for a simple iCal standard calendar files generation. The main advantage of using the iCal protocol is the wide support of external calendar clients (eg. Mozilla Sunbird, Mozilla Lightning, Microsoft Outlook 2007 and so on).

4 Main semantically enhanced features of Groupware System

According to usage of Groupware System (GWS) in context of SAKE we have identified several aspects for which semantics and text-mining methods are interesting (within scope of internal public administration processes which are mainly reflected in SAKE). All of them are now implemented; our

3 http://coefficient.sourceforge.net/
4 http://jakarta.apache.org/slide/
5 http://ical4j.sourceforge.net/
6 http://tools.ietf.org/html/rfc2445
7 http://dithaka.sourceforge.net/

GWS is fully integrated with other subsystems and is in testing phase.

The role of GWS in SAKE can be (more practically) viewed in [5]:

- Management of shared workspace of users in specific process context - forums, shared calendar, mail, ...
- Providing GWS-related services to other components
- Providing support for GWS-related activities in the workflow
- Providing semantically-enhanced features in order to achieve re-use of knowledge
- Capture the users behavior

Several of these aspects are interconnected directly through usage of sematic technologies, e.g. capture of user's behaviour in a semantic way leads to semantically enhanced feature known as semantic log. Then semantically enhanced features within GWS are [4]:

1. *Full support/usage of semantic context within GWS.* It is one of the basic assumptions for semantic-based support in the whole integrated SAKE system. Context is defined according to current user (precisely identified user with defined role, e.g. in case of small local authority Mayor can be role, also Expert role is important for experts' discussions, …) and actual state of the current process instance (precisely defined workflow-based process with several activities and tasks, current context knows actual state in one particular process for every moment, e.g. process is 'preparing of a new law about dogs/pets in local authority', current context is 'preparing of a draft version of law'). This context is then important for:
    a. Preparing of information which is shown to the user – the current context (user, process) is important for access rights to GWS functionalities and data
    b. Logging of context-specific information into integrated system (see next feature)
2. *Semantic log of user's activities within GWS.* All activities of users within GWS are semantically logged into ontologies. It means that all operations like creation, editing, deleting, accessing of discussion forums, discussion threads, messages, calendar events, and so on, are logged with their current context into the Log Ontology (which is part of Information Ontology). This information is used by AMS for querying and reasoning in order to achieve proactive delivery of knowledge extracted from the log. This is one way how to fulfill the goal of agile e-government knowledge-based system.
3. *Annotation of discussion messages*. Contributions of users in discussion forums (it means discussion messages) are annotated in two different ways:
    a. A set of keywords is attached as metadata to every contribution – there is one annotation to one message, keywords mean free text separated by commas. If the application domain has a corresponding fully understandable ontology with a good covering concepts' vocabulary, then also domain concepts could be used for the annotation. In our first testing pilot (internal process for preparing of local laws for local authority) we cannot have only specific domain ontology (law can be about everything, we can have only some top-like ontology about law-like facts, structure of process, …), free-text keyword-based annotation seems to be reasonable approach.
    b. Relevance feedback by users to discussion messages. This means that every user can add a feedback to a message representing user's own opinion of message's relevance to the current problem. It is based on a simple scaled annotation from negative to positive feedback. This information can then be used in other features like extraction of potential experts and search for relevant information in discussion forums.
4. *Metadata/Semantic search*. In order to achieve benefits from information provision to user metadata search for every subcomponent (functionality) is provided. It means that search is realized as combination of metadata searches for different information resources like documents, discussion forums, threads, and messages, calendar events, and so on. Semantic extension of this metadata search is achieved by an internal semantic expansion of a query with a textual description of concepts which are similar to query keywords.
5. *Extraction of potential experts for new case by ranking of users according to previous discussions.* Previously presented statistical method for discussion forums analysis (introduced in [3]) has been improved by user's feedback (scale of feedback from negative to positive) and annotations of messages. Combination of user's feedback and filtering of threads (according to annotation of messages in threads) provides 'argumentation'-like support solution which leads to feedback-based topic-sensitive ranking of discussion forums users. This information can be used by someone who wants to invite new members to process (e.g. expert) to help distinguish between several users. This improved method will be more deeply described in following chapter as our main goal for semantic enhancement of groupware system in

order to achieve collaboration-based support for building of 'Communities of practice' for specific cases.

5 Discussion forum analysis and ranking of users (experts)

Ranking of users within the GWS can be viewed as a voting procedure in the following way: Initialising a new discussion thread represents a desire of an author to increase his/her authority and to strengthen the author's position within the community discussion space. Responding to a contribution of another author represents voting of a respondent for the author of the contribution and increasing the authority of the author (the contribution of the author is worth for the respondent to react). First, statistical approach is based on application of previous discussion analysis algorithm, detailed description and experiments can be found in [3] (For completeness, short description is provided in section 5.1). In the next subsection of this chapter improved version of this algorithm is presented in order to use information like annotations of messages and relevance feedback.

5.1 A statistical approach to discussion threads evaluation

A discussion thread consisting of one or more contributions is a basic structural unit of a discussion group. The thread represents a particular view on the process of introducing a topic by presenting an opinion of one person and developing the topic by adding different views of different people.

The authorities of group members change as they participate in discussions within discussion threads. The threads are ordered using time (publication time of root contributions is considered) as an ordering criterion. Particular threads are processed sequentially and the authorities of authors are updated based on their participation within each given thread.

From the point of representation, a discussion thread is a tree-like structure nodes of which represent particular contributions while each arc represents a relationship among two contributions, one playing the role of an initiator and the second one the role of a respondent. If a contribution responses to another contribution, it is a respondent. If it attracts at least one response, then it is an initiator. It means that the root node of a tree (consisting of more than one node) representing a thread is an initiator, leaf nodes are respondents and intermediate nodes play the both roles – they are initiators and respondents at the same time.

In order to process a discussion thread and to update weights of participants, a two iteration procedure is involved:

- Calculating popularity of contributions
- Updating weights of contributors

The first iteration ensures that different contributions in the thread are treated differently. The difference is given not by the content of the contributions but by the response the contributions were able to attract. Therefore two contributions are different if they are responded by different numbers of responses or the responses were produced by authors having different authorities.

In order to calculate popularity of all contributions forming a thread, a bottom-up approach is employed. Popularity of pure respondents (contributions representing leaf nodes of the tree) is set to 1. Popularity of each initiator is based on popularity of its respondents and authority of authors of these responding contributions. It is calculated according to the following formula

$$y_j = \sum_{i=1}^{k} w_i y_i \qquad (1)$$

where k represents the number of contributions reacting to the contribution in question, y_i represents popularity of the i-th contribution, and w_i represents the weight of the i-th contribution's author. First, popularity of leaf nodes is calculated, popularity of the thread root is determined at the end.

As the second step, weights of authors contributing to the given thread are updated. The weight modification depends on how many contributions they have authored, how popular are their contributions as well as how many contributors were attracted by the given thread. The weight of an author is updated for each his/her contribution in the thread according to the following formula

$$w_k^{n+1} = w_k^n + \frac{y_i}{y_R}\frac{n}{N} \qquad (2)$$

where w_k is the weight of the k-th author, y_R is popularity of the root contribution of the thread, n is the number of different authors participating in the thread and N is the number of all group members. Since discussion threads have different sizes and the number of group members is also not constant, two normalisations are included in the used formula – popularity of the given contribution is related to the population of the root contribution and the number of involved authors is related to the number of all group members.

Finally after the thread evaluation process, every forum contributor is equipped by a weight value that represents his authority within the discussion forums. Having the weight value available for every forum user, it is possible to sort contributors by their weight value and to find those with the highest authority.

5.2 Semantic extension to discussion threads evaluation

When using the previous method for discussion analysis, the relevance of reactions is not guaranteed.

According to this disadvantage it was required to extend the method with a possibility to enable discussion forums users to enter a feedback on messages manually to inform the system about the message quality, to express their positive or negative reaction [4].

5.2.1 User feedback on messages

The purpose of the user feedback on messages is to provide a solution of disadvantages emerging from the previously mentioned method. Concretely it is a solution of the fact that authority values of forum participants contributing with low quality messages accumulate as they are being frequently commented by messages whose authors posses a rank of – in the context of a discussion forum – a higher value. This accumulation of high popularity values of messages with obvious low content quality after such statistical computation we may interpret as very competent discussion contributions.

With having these problems in mind we have extended the *statistical approach to discussion threads evaluation* algorithm with a third step, so the consequential algorithm is the following:

- Calculating popularity of contributions for a given discussion thread
- Calculating popularity of contribution for a given discussion thread based on received user feedback values
- Updating weights of contributors

The user feedback is collected for every message manually in the manner, that every discussion forum user posses a possibility of one-time evaluation of any message by selecting a value from a list of predefined choices (*0 – Bad, 1 – Rather Bad, 2 – Neutral, 3 – Rather Good, 4 - Good*) that represent his opinion about the concrete message.

On the basis of collected user feedbacks for particular messages, new message popularity values are computed:

$$y_j^{'} = 0{,}25\overline{f_j}\,y_j \qquad (3)$$

where $y_j^{'}$ is a new popularity value of the contribution *j*, $\overline{f_j}$ is an average value of the user feedback for the contribution *j*, y_j is a popularity value of an original contribution *j*.

By using this new proposed approach the original popularity value y_j of the message *j* is transformed to a new value $y_j^{'}$ from the interval <0, y_j> according to the enriched user feedback value of the original post. If there is no user feedback value for contribution *j*, the average value $\overline{f_j}$ is set to a neutral value of 2 and with this new setting is the new popularity computed. Omitting this step all of the contributions without the user feedback are perceived as very competent contributions, but it might not be true.

5.2.2 Discussion threads filtering

According to a need of expert selection for particular internal public administration processes (e.g. expert group creation important in the process of preparing of new local law) it makes sense to analyze only those discussion threads, which concern about topics related to the topic of current process. This chapter describes the design of a method for filtering topic-related discussion threads that are used as an input for *extended statistical approach to discussion threads evaluation* algorithm described in previous chapter.

Every user when creating a message contribution is necessarily requested to create an annotation by specifying keywords describing the content of the message. The message content together with its annotation is saved into the *Lucene*[8] index. The filtering process is based on searching for particular keywords in the *Lucene* index, especially in message annotation fields. Discussion threads, which were found by performing such search, are extracted into a new thread collection and used as an input for a context-specific discussion threads evaluation.

6 SAKE pilots and testing

The SAKE project evaluation will be performed in three public administration authorities of different size and area of influence [6].

6.1 Pilot sites and selected processes

The Hungarian pilot carried out by the Ministry of Education with a support of Corvinus University in Budapest deals with receiving and processing the changes in legislature in e-gov system and recommending actions resulting from these changes, especially for the preparing of the strategy for higher education in Hungary – selected process is "Higher Education Portfolio Alignment with World of Labour Needs".

Similar philosophy has the Polish pilot (performed by the Town of Czestochowy Town Hall supported by The Association of Town on Internet). The application is focused on processing the changes in legislature and the consecutive identifying of fields influenced by the change according to material resources for education institutions and their needs for repairs and reconstructions – selected process is "Management of educational institutions' material resources".

8 http://lucene.apache.org/java/docs/index.html

The Slovak pilot application (performed by the Košice – City Ward Sídlisko Ťahanovce supported by the Technical University of Košice) is focused on process of adopting of global national or European law into local authority environment – selected process is "The making of local legal regulations".

6.2 Testing of Groupware System

Implementation and testing follows three-phase iterative process. First, the basic functionality prototype has been tested. This version had only basic functionality without semantic enhancement. Also, basic functionality prototype was used for the first testing on pilot sites in order to update user's requirements as well as some implementation details and suggestions for next steps. Second, First iteration of semantically enhanced version has been developed also with re-use of first version of discussion forums analysis algorithm (as described in chapter 5.1). According to first evaluation of Basic functionality prototype and experiences with first prototype, we have finalized development of GWS by implementation of Second prototype. Now we have all expected enhancements and we are now fully integrated with other subsystems of SAKE. Our internal testing shows full potential of the system. In the following months after all of the pilot preparation work is finished, we are starting the evaluation of our knowledge-based system in real environments.

7 Conclusions

In this article we have presented our groupware component within the knowledge-based supporting system (in project SAKE) for public administration and their (mostly) internal processes. The core of the system is represented by an integrated knowledge space unifying different perspectives and interpretations of knowledge resources. It enables for each knowledge object to be assigned metadata allowing more sophisticated retrieval and use. A goal of our groupware system is to support sharing of the knowledge using collaborative software as well as to help user in managing and creation of experts group, which are used in governmental and self-governmental internal administration processes. We have identified and described several implemented enhancements of standard groupware. One of the main contributions to semantic enhancement of the system is based on method for discussion forums analysis. Combination of user's feedback and filtering of threads (according to annotation of messages in threads) provides 'argumentation'-like support solution which leads to a feedback-based topic-sensitive ranking of discussion forums users. This information can be used by someone who wants to invite new members into process (e.g. expert) and help him to distinguish between several users in order to achieve collaboration-based support for building of 'Communities of practice' for specific cases.

8 Acknowledgments

The work presented in the paper is supported by the EC within the FP6 IST 027128 project "SAKE – Semantic-enabled Agile Knowledge-based E-government", by the Slovak Grant Agency of Ministry of Education and Academy of Science of the Slovak Republic within the project No. 1/4074/07 "Methods for annotation, search, creation, and accessing knowledge employing metadata for semantic description of knowledge", and by the Slovak Research and Development Agency under the contract No. RPEU-0011-06 (project PoZnaŤ).

References

[1] Stojanovic N, Mentzas G, Apostolou D: **Semantic-enabled Agile Knowledge-based e-government**, Semantic Web meets eGovernment 2006, AAAI Spring Symposium Series, Stanford University, California, USA, 2006.

[2] Butka P, Hreňo J: **Architecture of Groupware System in SAKE**, In: Electronic Government – 6th International EGOV Conference: Proceedings of ongoing research, project contributions and workshops, DEXA conferences, September 2007, Regensburg, Germany, pp. 301-308.

[3] Mach M, Lukáč G: **A Dedicated Information Collection as an Interface to Newsgroup Discussions**, Proceeding of 18th International Conference on Information and Intelligent Systems, IIS 2007, September 2007, Varaždin, Croatia, 2007, pp. 163-169.

[4] SAKE Consortium: **D21 – 2nd Iteration Prototype of Semantic-based Groupware System**, April 2008, SAKE project deliverable.

[5] SAKE Consortium: **D24 – Integration Plan**, March 2008, SAKE project deliverable.

[6] Želinský T, Sabol T, Hreňo J, Butka P: **SAKE Project: Agile Change Management in Knoweldge-based eGovernment Back-office**, In Proceedings of National and Regional Economics VI, Herľany, Slovakia, October 2006, pp. 471-475.