

A KNOWLEDGE-SHARING FRAMEWORK FOR PUBLIC ADMINISTRATIONS

Olivier Glassey

*Fraunhofer FOKUS, Kaiserin-Augusta-Allee 31, 10589 Berlin,
olivier.glassey@idheap.unil.ch*

Abstract: This paper describes a framework that supports knowledge modeling and sharing within public administrations, and a prototype of such a knowledge-sharing system. We will first give a brief theoretical introduction on processes of knowledge creation, transfer and application, and then we will present our framework and discuss how it relates to these processes. We will furthermore illustrate this with a case study and show the architecture of the prototype we developed in this context.

Key words: knowledge models, process models, public administration, methodology, knowledge sharing, prototype, RSS

1. INTRODUCTION

The goal of this paper is to describe a framework we developed in order to model and share knowledge within public administrations. This framework is based on a formal methodology, on eight different models for graphical representation and on a knowledge-sharing system.

To define our methodology we used the knowledge management life-cycle proposed by Nissen, Kamel & Sengupta (2000). They studied four life-cycle representations and created an amalgamated model consisting of 6 phases: *create*, *organize*, *formalize*, *distribute*, *apply* and *evolve* knowledge. They made a review on what tools, technologies and practices were available for each of these phases. Likewise we propose a set of instruments for each phase, which we will describe in section 2.

The focus of our framework is on graphical representation: Eppler & Burkhard (2004) define knowledge visualization as all graphic means that

can be used to construct and convey complex insights. They propose several visualization types:

- *Heuristic sketches* or ad-hoc drawings
- *conceptual diagrams*: abstract, schematic representations of structural relationships
- *visual metaphors*: used to structure information and convey normative knowledge through the connotations of the metaphor
- *animations*: interactive descriptions of procedural knowledge
- *knowledge maps*: they do not represent knowledge but rather reference it
- *scientific charts*: based on computational algorithms

In their review of conceptual foundations for knowledge management and in the section dedicated to knowledge transfer, Alavi & Leidner (2001) classify knowledge transfer channels as *informal* or *formal*, *personal* or *impersonal*. They mention a few examples: coffee break meetings are typically informal, personnel transfers within departments during a training period are formal and personal, knowledge repositories are formal and impersonal, and so on. For their analysis framework, Alavi & Leidner (2001) rely on the four processes of knowledge creation defined by Nonaka & Takeuchi (1995):

- From implicit to implicit knowledge: *socialization*
- From implicit to explicit knowledge: *externalization*
- From explicit to explicit knowledge: *combination*
- From explicit to implicit knowledge: *internalization*

Using our framework we made a feasibility study for a knowledge management system in a German public administration, which we will briefly explain in section 3. One of the focuses of this study was knowledge sharing amongst the domain workers: was there any, and if so, how did the clerks share knowledge and what were the potential problems? As we found out that there was no systematic and formalized knowledge sharing processes, we proposed an architecture for such a system and developed a small prototype (section 4). This architecture was not only technical, but also organizational, as McLure Wasko & Faraj (2005) explain that the availability of electronic communication technologies is no guarantee that knowledge sharing will actually take place and examine why people voluntarily contribute knowledge. They identified the main problems that sharing knowledge could arise in what they call networks of practice, i.e. loosely knit groups of individuals who are engaged in a shared practice but who do not necessarily know each other:

- Knowledge seekers have no control over the *respondents* and the *quality of the responses*.
- Knowledge contributors have no assurances that those they are helping will ever help them in return (*reciprocity*).

From their detailed survey on knowledge sharing, McLure Wasko & Faraj (2005) conclude that contributors care about their personal/professional reputation within the network of practice and that reputation is a sufficient mean to guarantee the quality of responses in most cases. Furthermore they found out that contributors do not expect direct reciprocity but rather third-party reciprocity, given that there is a critical mass of active participants within the network of practice.

2. KNOWLEDGE MODELLING

In previous work (Glasse 2005), we developed a framework called MIMIK (Method and Instruments to Model Integrated Knowledge). We will not explain this framework in detail here, although we will provide several examples. MIMIK supports the 6 phases developed by Nissen & al. (2000): *create, organize, formalize, distribute, apply* and *evolve* knowledge. However we considered that creating and evolving knowledge belonged to the same phase: in most cases organizations do not create knowledge ex-nihilo and then evolve it; we would rather consider it as a continuous creation cycle. The goals of MIMIK are to identify:

- Strategic goals of an organization
- Actors and roles
- Knowledge resources
- Processes
- Interactions between these elements.

As the main goal of our work was to represent knowledge graphically, we needed a formalism to do so and we analyzed what was being done in process methodologies. One basic way to represent knowledge in organizations is the use of business rules (Ross 1997). They can be found in all sectors of activity and do not have to be linked to an information system. Some of them are implicit, meaning that they are not written anywhere but they belong nevertheless to the “business culture”. However the basic formalism proposed by Ross (1997) is not sufficient in all cases to model the “know-why”. Indeed, a “knowledge unit” is anything worth storing that may help things to be done better in the future: help, best practices guidelines, examples, stories, lessons learned, troubleshooting advice or training material (Fraser & al. 2003) and business rules cannot model all of these types of knowledge. A different approach is described by authors such as Gamper & al. (1999) and Gruber (1993) that use ontologies (explicit specification of a conceptualization, the latter consisting of identified concepts and relationships assumed to exist and to be relevant) in order to model knowledge. We prefer this method as we previously used RDF to

build a data-model for e-Government (Glassey 2004) and found it more powerful and flexible than classical data models such as Entity-Relationship-Model used in ARIS (Scheer 2001) or than business rules. RDF (Resource Description Framework) is a W3C standard for defining metadata and encoding machine-readable semantics (Noy & al. 2000). It is based on XML and uses graph theory to represent knowledge. It is also a suitable format for specific domain ontology modeling.

However ontologies still cannot represent complex knowledge such as storytelling or human advice. As Samuel Johnson put it in the 18th century already: “Knowledge is of two kinds. We know a subject ourselves or we know where we can find information upon it”. The goal of the component-based architecture we propose is to model “the information upon knowledge” and to describe this knowledge. We identified several attributes that allowed us to do so, beginning with the type of knowledge. Capurro (2004) compares the knowledge typology proposed by Zahn & al. (2000) with the classical Aristotelian one. Here we will only summarize the main points of Capurro’s knowledge typology:

- *Know-how*: knowledge about how to make things (technical knowledge) and knowledge acquired through experience and remembrance (empirical knowledge).
- *Know-why*: logical reasoning (scientific knowledge).
- *Know-what*: knowledge about the best means to achieve given goals, usually a combination of know-how and know-why (practical knowledge).

Capurro (2004) furthermore states that what can be managed is information or explicit knowledge and that implicit knowledge can only be “enabled”. In this context, explicit means that it can be clearly observed and expressed (and also digitalized), as opposed to implicit knowledge that can not be directly formulated (skills, experiences, insight, intuition, judgment, etc.) When knowledge is explicit, it can be represented as declarative or procedural knowledge. We are aware that in the domain of cognitive sciences, the distinction between procedural and declarative models is related to the brain memory system (see for example Ullman, 2001), but here we used these terms here in a limited sense, as defined in computer science:

- *Declarative* knowledge components represent domain knowledge (facts, events, etc.) in terms of concepts and relations.
- *Procedural* knowledge components describe actions to be taken in order to solve a problem step by step.

For cases where knowledge is implicit and cannot be formalized, we introduced the concept of distribution: knowledge can be individual or collective, and in both cases components identify who has this knowledge or where it can be found. Finally we added a set of metadata (know-where,

know-when, know-who, etc.) describing these knowledge-components and making it possible to manage them. Fig. 1 shows the complete component-based architecture under the form of a class diagram, but it can also be formalized in RDF.

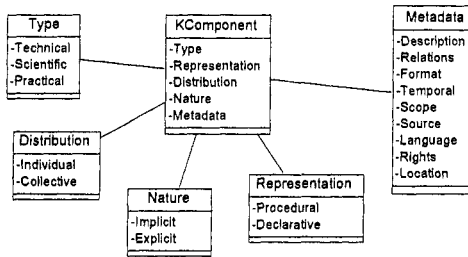


Figure 1. Knowledge component.

The goal of MIMIK is to provide meta-knowledge on the organization and to model goals, actors, roles, processes and organizational knowledge. We based our work on the model theory approach developed by Wyssusek & al. (2001) to integrate process modeling and knowledge management. They provide an epistemological foundation to justify their approach, but they do not offer any practical methodology or examples. That is why we created a conceptual framework that aimed at the integration of both these approaches.

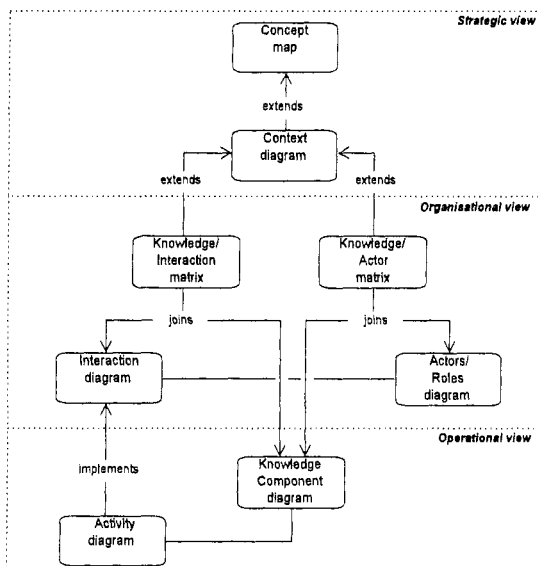


Figure 2. Metamodel showing formal relations between the different types of diagrams proposed by the MIMIK framework.

MIMIK consists of 8 types of diagrams, most of them being inspired or directly taken from existing modeling techniques, mostly UML. As in UML or other modeling languages, it is not necessary to use all of them in order to provide a good representation of reality. Users should rather select the diagrams that suit their needs and goals in terms of modeling. We will provide examples for most of these diagrams, but more explanations can be found in Glassey (2005).

Concept maps (*Cmap*) are the top-level diagrams and show the strategic goals of an organization in terms of functions or processes (Fig. 3). Let us mention that the metamodel of our framework is in itself a concept map. These concept maps can be decomposed in several levels, a terminal node of this type of diagram is implemented by a context diagram (*Cdiag*).

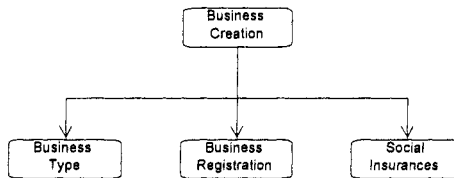


Figure 3. Concept Map.

Context diagrams (Fig. 4) are almost exactly the same as use cases in UML, but we added the concept of knowledge packet. A knowledge packet is an abstract representation of a set of knowledge components. These components encapsulate documents, databases, files, implicit knowledge and so on. They provide metadescriptions for “knowledge units”, thus allowing us to show what type of knowledge is necessary in order to complete a process and which knowledge is relevant in a given context. Context diagrams provide an abstract view of the “know-what”.

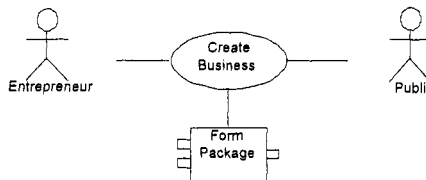


Figure 4. Context Diagrams.

To model organizational structure we used actor-role diagrams, which allowed us to add formal distinction between actors and roles, which does not exist in UML. These actor-role diagrams can be either classical organizational charts or matrices that formally link actors and roles in cases where the organization is too complex to be shown graphically in an

intelligible way. We will not show that here, but let us mention that the abstract actors represented in context diagrams can be linked to these actor-role diagrams. Moreover the actors described in such diagrams are used in the knowledge-actor matrices (see further on).

Fig. 5 shows a knowledge-interaction matrix, formally linking knowledge components to the interactions that implement a use case. In UML an interaction is the specification of how messages are sent between objects or other instances and interaction diagrams (sequence or collaboration diagrams) emphasize object interactions. We comply with this definition and use collaboration or sequence diagrams to specifically describe each interaction shown in this matrix. By matching the “know-why” (knowledge components) and the “know-how” (interaction diagrams), this matrix shows the “know-what” at the operational level.

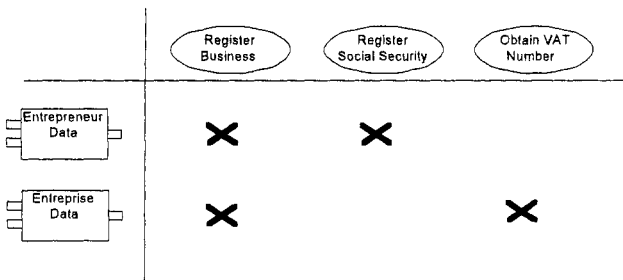


Figure 5. Knowledge-Interaction Matrix.

Exactly as knowledge-interaction matrices link knowledge components and interactions, the concept of knowledge-actor matrices (Fig. 6) create a formal relation between knowledge components and real actors within an organization. They provide an organizational view of the “know-what” or more precisely they show the “who-knows-what”. That proves very useful in order to introduce implicit knowledge in a graphical model: it might not be possible to transform it into explicit knowledge but at least we know who has this knowledge within an organization. Knowledge-interaction matrices can also link actors and interaction diagrams provided a small constraint: within interaction diagrams modelers should only use roles that were defined in actor-role diagrams.

The concept of knowledge matrix (*Kmax*) provides a formal link between the strategic and operational models: it describes the interactions between strategic goals, actors, knowledge and processes such as explained at the beginning of section 2.

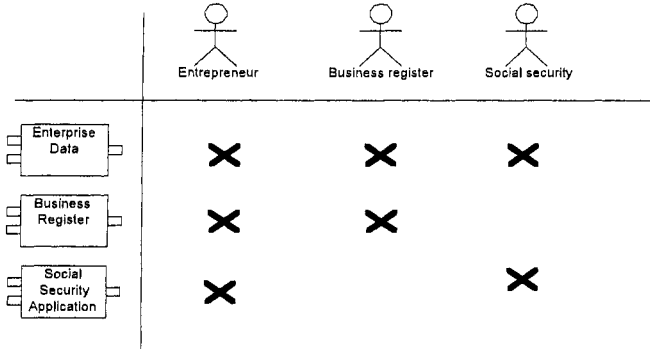


Figure 6. Knowledge-Actor Matrix

At the operational level MIMIK uses standard UML collaboration and sequence diagrams, grouped under the name of interaction diagrams (*Idiag*), as well as UML activity diagrams. The knowledge components (*Kcomp*) are integrated as objects (or instances of the *Kcomp* class) in interaction and activity diagrams.

3. CASE STUDY

Between October 2004 and January 2005 we made a feasibility study in the German County of Herford in Nordrhein-Westfalen in order to assess the potential of a knowledge-based system in the domain of family law (Glassey, Gordon & Pattberg 2005). This system should allow clerks to manage more efficiently the recovering of social benefits paid to elderly persons. In short, the state supports elderly persons that are without financial resources and, under given conditions, it has a right to ask some of this money back from the relatives (mostly the children) of these elderly people. The offices we worked with were to decide whether a child has to support financially his/her parents, and if so, how much money is to be paid. The goals of this feasibility study were:

- To describe the actual work processes
- To evaluate the consistency and the quality of the actual work processes
- To propose optimized work processes
- To develop a prototype for a legal knowledge-based system.

Our team¹ created a model of the relevant laws and regulations used in order to make such parent support decisions. We also led interviews with the

¹ This feasibility study was a collaboration between Fraunhofer FOKUS, Herford County and Fachhochschule für öffentliche Verwaltung Nordrhein-Westfalen (Fhöv). Thomas

clerks in order to identify their work processes and to describe how they acquire and share knowledge, as the regulations in that domain change quite often and the workers have to rely on up-to-date legal sources. Furthermore, two test cases were submitted to ten clerks. Finally we created an online questionnaire in order to find out how they solved the test cases, what information and knowledge they needed and where they found them if they had to specifically research it. This was a very interesting experimentation field as the work processes were weakly formalized and the knowledge was completely decentralized and disseminated, and not directly explicit in many cases. Indeed, the clerks have to interpret regulations in order to make a decision, and formalizing this process can be quite difficult.

Using the MIMIK framework we identified the three strategic missions of “Elternunterhalt” (parent support): *monitoring* the changes in *law and regulations*, *integrating these changes* into the daily work of the organization and *handling the actual parent support cases*.

For the first strategic mission (*monitoring*) we found out that the clerks who answered the online questionnaire used legal texts and databases extensively, including case law:

- The majority of the clerks (75%) use legal texts weekly or several times monthly, the other 25% use them once a month or less.
- They all read various specialized publications in the domain of social welfare. 65% of the respondents read these several times per month.
- They all (except one) take part in training seminars about once a year.
- 40% of their work time is dedicated to doing legal research for particular cases, reading legal journals and publications and calculating the amounts of money the relatives have to pay.

However, each clerk is doing it his/her own way and using different sources (up to 20 different legal sources). Furthermore, there was absolutely *no structured or formalized sharing and integration of this knowledge* (no process corresponding to the second strategic mission). If we refer to the Nonaka & Takeuchi model (1995) presented in the introduction, the only knowledge creation process we identified was *socialization*, and *externalization* and *combination* were almost non-existent. For example, different units at commune and county levels developed their own forms and spreadsheets tables in order to acquire data from the potential social beneficiaries and to calculate the amounts of money their relatives might have to pay. In some cases, clerks would only share knowledge with their colleagues during informal discussions on the phone or at the coffee break and only a handful of them use email to share their specific domain

knowledge. The only “formal” knowledge acquisition activity consists of an annual continuing education seminar.

On the other hand the *actual work processes* describing how the cases are treated and decisions are made were *rather formal, although implicit*. With the help of clerks we were able to define how they conducted interviews and we developed detailed interaction and activity diagrams, which we will not show here. As mentioned in the introduction, one of the goals of this feasibility study was to evaluate the quality and the consistency of the work processes, thus the test cases. We will not present the detailed results here, only a few key points:

- Between 75% and 79% of the respondents found that the cases were clear, well formulated and realistic; they said they understood what they were expected to do and they believed they had all the necessary information in order to solve the cases; finally they agreed that they had enough knowledge and support tools to solve the cases.
- However 44% of them said they had problems to solve the cases; in case 1, 20% of the solutions were not defensible and the respondents needed an average of 152 minutes to solve it; in case 2, 33% of the solutions were not defensible and the respondents needed an average of 162 minutes to solve it.

We expected that these tasks could be optimized with the use of knowledge management technology: a knowledge-sharing system would support the first two strategic missions (monitoring and integrating the changes) and a legal rule-based system should help optimize the consistency and the quality of processes for the third strategic goal. Furthermore the IT infrastructure is already in place: all workers use word processing tools and email, and most of them also have Internet access. Very briefly put, our proposition for process optimization was to select one or several clerks that would have the formal responsibility to monitor the changes in the legal sources and to publish these changes using the prototype that we will present in the next section.

4. PROTOTYPE

The prototype was based on two different tools: a legal rule-based system to support the resolution of the cases and a knowledge-sharing system. The rule-based system will be presented in another publication, but let us mention a few key facts: it is accessible through a simple web-browser, it integrates more than 200 rules and it solved both test cases correctly.

Here we will concentrate on the prototype of a knowledge-sharing system (KnowS). Its architecture is based on RSS (Really Simple Syndication). RSS

is a family of XML file formats for web syndication. The XML files (or RSS feeds) provide “items” containing short descriptions of web content together with a link to the full version of the content. In order to access these feeds, users rely on applications called feed readers that check RSS-enabled Web pages and retrieve any updated content that it finds. Websites featuring RSS feeds include The New York Times, The Wall Street Journal, BBC, news.com, Liberation, etc. RSS is widely implemented in the weblog community in order to share the latest weblog entries. According to a Pew Internet and American Life Project survey (Rainie 2005), there were 8 millions bloggers in the United States at the beginning of 2005 and 27% of Internet users say they read blogs. Furthermore, Gordon (2003) showed that RSS can be used for public participation platforms, for example to facilitate public consultation, deliberation, and participation or “engagement” in policy-making processes such as urban planning. For more on RSS we recommend (Winer 2005) or wikipedia.org.

Using blog platforms, end users can publish new knowledge via a Web interface, a simple email sent to a special address or a dedicated feed publishing client. This requires no specific knowledge (other than being able to send an email, at the most basic level of use), the input text is automatically transformed in an RSS feed by the system. A “moderator” should validate this new content before it is available to anyone, but it is not required. In our Herford example, clerks would be notified when colleagues have found new pieces of knowledge in legal databases, online law commentaries, or when they have themselves implemented a new form or a new calculation formula. A moderator could then validate or complete this knowledge published by a clerk. All clerks could then rely on this knowledge in their daily work, as it would have been validated by a “domain expert”. Furthermore, specific thematic RSS feeds can be defined: users can then choose precisely what knowledge they want to receive.

Once new knowledge has been published, it can be used in very flexible ways. Users can simply visit the Web page of the blog, but they can also use Web aggregators such as Bloglines.com, their own email client or a specialized feed-reader that provides more advanced functionalities. However, the Internet and American Life Project survey on blogs (Rainie 2005) stated that only 5% of Internet users rely on dedicated aggregators to get RSS feeds. RSS aggregators and, to some extent, email clients offer powerful content management capabilities, such as filters to limit access to only relevant content: a user can for example subscribe only to feeds that aggregate content on social welfare issues and limit this to parent support, they can furthermore implement filters stating that all feed elements not concerning the Land of Nordrhein-Westfalen are to be deleted. This is very useful to avoid information overflow, that is to limit the risk that the users

will not read the feeds anymore because they receive to much irrelevant information.

RSS feeds support “enclosures”, which allow the addition of any type of multimedia files, similar to an attachment in an email. Thus we added this functionality to our prototype: it can be used to share automatically new documents, files or any piece of digitalized information. With an advanced RSS reader, it becomes possible to check periodically (once a day, every week, etc.) selected feeds and to download relevant documents automatically.

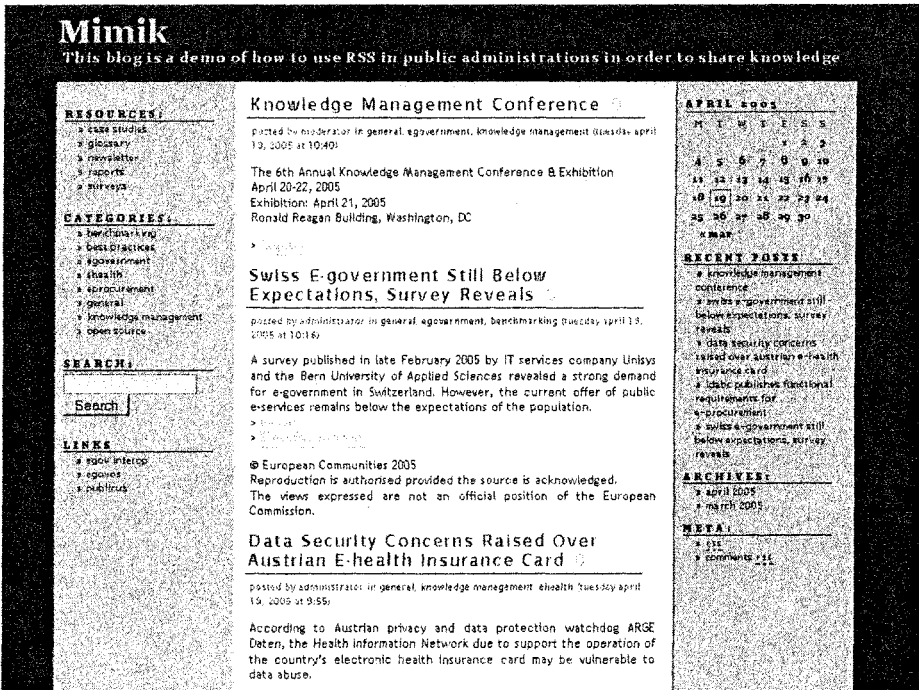


Figure 7. Web interface to the Mimik demo blog.

The KnowS prototype comes in three “flavors”, that is three technical architectures corresponding to different needs. The first one is based on existing services: Blogger.com, a weblog platform owned by Google, and Feedburner.com, a free post-processing service that allows publishers to enhance their feeds. It allowed us to create a simple proof of concept² for what was presented in the previous paragraphs, without any material acquisition or software installation. However Blogger does not support categories for blog entries: we believed that this could prove very useful to

² It can be seen at mimik-demo.blogspot.com

organize and personalize knowledge. Indeed by matching categories and interactions such as defined in the knowledge-interaction matrices, we could implement a system corresponding to the knowledge models created with MIMIK. Moreover Blogger's user rights management is very limited. For these reasons, we built a second KnowS prototype (Fig. 7) operating on a dedicated blogging platform: WordPress³ offers good categories' management tools and 10 levels of users' rights (read, write, edit, validate, publish...). We consider this type of blog publishing platform, powerful and simple to use, would be sufficient in most cases in order to implement a knowledge-sharing system for public administrations. In some complex cases however, a complete content management system supporting RSS could be necessary. In order to illustrate such a technical architecture, we created a third prototype based on Agora⁴, a specialized Web content platform developed on top of the SPIP⁵ system at the initiative of the French government. To summarize the advantages of this system, it allowed us to integrate categories and keywords, thus to have a finer knowledge organization, and to assign the control over the content of single categories and keywords to given users. With these functionalities we were able to match not only the knowledge-interaction matrices in the system, but also the knowledge-actor matrices. Finally, let us mention that the two more complex versions⁶ of KnowS integrate directly most metadata defined for the knowledge components. RSS provides a *description* for content and integrates various tags (author/*source* of an RSS element, *publication date*, *language*). The KnowS prototype supports several *temporal markers* (published, updated, validity), it has advanced *rights* management functions and supports embedded files with MIME type description (*format*). Moreover it is possible to use permalinks (a type of URL designed to refer to a specific information item and to remain unchanged permanently) to identify the *location* of a knowledge element and to show *relations* between these knowledge elements with track-backs (system allowing bloggers to see who has written another entry concerning a given post).

5. CONCLUSIONS

As we mentioned in the introduction, the MIMIK framework is focused on knowledge visualization and in this paper we have shown several types of

³ WordPress is based on PHP/MySQL, available under GPL licence at www.wordpress.org

⁴ www.agora.gouv.fr

⁵ Spip is a popular Web publishing system written in PHP/MySQL, available under GPL licence at www.spip.net

⁶ Both run on a local server and cannot be accessed because they are behind a firewall.

visualization: concept maps (*Cmap*) can be used to create heuristic sketches; context diagrams (*Cdiag*), interaction diagrams (*Idiag*) and knowledge components diagrams (*Kcomp*) are typical conceptual diagrams; knowledge matrices (*Kmax*) provide basic knowledge maps. In the introduction we also defined fundamental concepts of knowledge creation and transfer: Fig. 8 shows how we think the different MIMIK models and tools support the knowledge creation life-cycle and what main knowledge creation processes they facilitate.

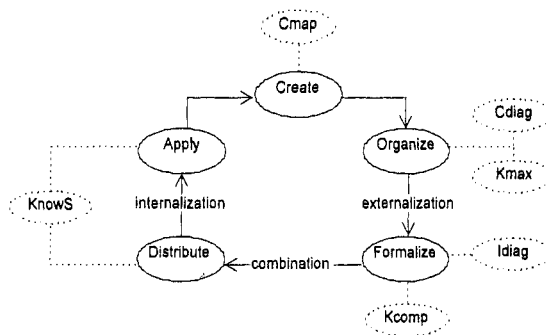


Figure 8. Knowledge creation cycle and MIMIK models.

Finally we showed with the knowledge-sharing system (*KnowS*) prototype that it was possible to define a technical architecture supporting knowledge creation and transfer. On the other hand we have not been able to test it in a real situation and thus could not investigate the issues of quality of responses, reciprocity and reputation linked with knowledge sharing over networks of practices.

6. REFERENCES

- Alavi, M. & Leidner, D. E. (2001). Review: Knowledge management and knowledge management systems: Conceptual foundations and research issues. *MIS Quarterly* 25(1), 107-136.
- Capurro, R. (2004). Skeptical Knowledge Management. In H. C. Hobohm (Ed.), *Knowledge Management: Libraries and Librarians Taking Up The Challenge* (pp. 47-57). Munich: Saur.
- Eppler, M. J. & Burkhard, R. A. (2004). Knowledge Visualization. Retrieved 21.04.2005, from <http://www.knowledgemedia.org/modules/pub/view.php/knowledgemedia-67>.
- Fraser, J., Adams, N., Macintosh, A., McKay-Hubbard, A., Pariente Lobo, T. & Fernandez Pardo, P., et al. (2003,). Knowledge Management Applied to E-government Services: The Use of an Ontology. *Proceedings 4th International Working Conference KMGov 2003*, Heidelberg: Springer Verlag.

- Gamper, J., Nejd, W. & Wolpers, M. (1999). Combining Ontologies and Terminologies in Information Systems. Paper presented at the *5th International Congress on Terminology and Knowledge Engineering*, Innsbruck, Austria.
- Glassey, O. (2004). Developing a one-stop government data model. *Government Information Quarterly* 21(2), 156-169.
- Glassey, O. (2005). Knowledge Component-based Architecture for Process Modelling. Paper presented at the eGOV INTEROP'05 Conference, Geneva, Switzerland, February 23-24, 2005.
- Glassey, O., Gordon, T. & Pattberg, J. (2005). Machbarkeitsstudie eines wissensbasierten Rechtsberatungssystems im Kreis Herford. In E. Schweighofer, Liebwald, D., Menzel, T. & S. Augeneder (Eds.), *Aktuelle Fragen der Rechtsinformatik 2005*, Boorberg Verlag.
- Gordon, T. (2003). An Open, Scalable and Distributed Platform for Public Discourse, *Informatik 2003*, Vol. 2, 232-234.
- Gruber, T. (1993). Towards principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies* 43(5-6), 907-928.
- McLure Wasko, M. & Faraj, S. (2005). Why should I share? Examining Social Capital and Knowledge Contribution in Electronic Networks of Practice. *MIS Quarterly* 29(1), 35-57.
- Nissen, M. E., Kamel, M. N. & Sengupta, K. C. (2000). A Framework for Integrating Knowledge Process and System Design. *Information Strategy: The Executive's Journal* 16(4), 17-26.
- Nonaka, I. & Takeuchi, H. (1995). *The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Evolution*. Oxford University Press.
- Noy, F., Ferguson, R.W. & Musen, M. (2000). The Knowledge Model of Protégé-2000: combining interoperability and flexibility. Retrieved 12.01.2003, from <http://www.smi.stanford.edu/projects/protége/>.
- Rainie, L. (2005). The State of Blogging, Pew Internet & American Life Project, February 2, 2005. Retrieved 10.03.2005, from <http://www.pewinternet.org/>.
- Ross, R. (1997). *The Business Rule Book: Classifying, Defining and Modelling Rules, Version 4.0*. Houston, TX: Business Rules Solutions.
- Scheer, A. W. (2001). *ARIS - Modellierungsmethoden, Metamodelle, Anwendungen*. Berlin Heidelberg: Springer-Verlag.
- Ullman, M. T. (2001). A Neurocognitive Perspective on Language: The Declarative/Procedural Model. *Nature Reviews - Neuroscience* 2(10), 717-726.
- Winer, D. (2005). RSS 2.0 Specification, RSS at Harvard Law. Retrieved March 10, 2005, from <http://blogs.law.harvard.edu/tech/rss>.
- Zahn, E., Foschiani, S. & Tilebein, M. (2000). Nachhaltige Wettbewerbsvorteile durch Wissensmanagement. In: *Wettbewerbsvorteile durch Wissensmanagement* (pp. 239-270). Stuttgart: Methodik und Anwendungen des Knowledge Management.