THE S-CUBE KNOWLEDGE MODEL
Experiences in Integrating SSME Research Communities[*]

Vasilios Andrikopoulos, Michael Parkin, Mike P. Papazoglou
European Research Institute in Service Science (ERISS), Tilburg University, Tilburg, The Netherlands

Patricia Lago
Software Engineering Group, Department of Computer Science, VU University Amsterdam, Amsterdam, The Netherlands

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

Keywords: Knowledge model, Knowledge management.

Abstract: The field of Service Science, Management & Engineering (SSME), covers a wide range of research topics and has become fragmented due to the necessary specialization such broad area requires. The European Commission's Network of Excellence in Software Services and Systems (S-Cube) is an attempt to bring scientists together to perform joint research in this field that crosses existing research boundaries and, in the process of doing so, to help establish an enduring European network of researchers practicing SSME.
To assist in the consolidation of research and bridge the gaps between disciplines, the S-Cube Knowledge Model (KM) has been developed to provide a method of capturing, managing and refining the knowledge produced by the network and provide a common understanding of research outputs. This paper describes the motivation, requirements and realization of the S-Cube KM, which allows the collection, analysis and management of research within S-Cube and enables the extraction and combination of the explicit, cross-cutting knowledge embedded in collaborative research.

1 INTRODUCTION

In response to the fragmentation of software service-based systems research, the Software Services and Systems Network of Excellence (S-Cube NoE[2]) was conceived by the European Commission (EC) to establish and develop an integrated, multidisciplinary, vibrant European community in Service Science Management & Engineering (SSME). The S-Cube NoE brings together researchers from many different domains, distributed across 16 academic partner institutions, 6 associated industry partners and 17 associate members to collaborate on research topics defined in the S-Cube research framework. A central goal of the S-Cube NoE is to bring these diverse communities together to ensure their joint research is coherent, interdisciplinary and aligned through the cross-fertilization of knowledge. To help achieve this, the S-Cube NoE has developed the *S-Cube Knowledge Model*, or *KM* for short, to capture, organize and refine the knowledge generated by researchers in the network and provide a common understanding at a terminological level of the wide range of knowledge required for SSME research.

The S-Cube KM is made up of a publicly-accessible technology platform, corpus of information and set of quality assurance procedures. It allows researchers in the network to share, in a standard way, key information about them and their work and to position their research and competencies in relation to case studies and other research and researchers. This information can be used to provide a comprehensive understanding of how the network's research efforts and capabilities fit into the larger body of SSME knowledge and, through its analysis, be used to identify relationships between people, institutions and gaps and overlaps in the research of the network.

This paper is a description of the motivation, methodology and implementation of the S-Cube KM and a presentation of our experiences in capturing, curating, managing and refining the knowledge gath-

[*]The research leading to these results has received funding from the European Community's Seventh Framework Program FP7/2007-2013 under grant agreement 215483 (S-Cube).

[2]http://www.s-cube-network.eu/

Andrikopoulos V., Parkin M., P. Papazoglou M. and Lago P..
THE S-CUBE KNOWLEDGE MODEL - Experiences in Integrating SSME Research Communities.
DOI: 10.5220/0003662102010210
In *Proceedings of the International Conference on Knowledge Management and Information Sharing* (KMIS-2011), pages 201-210
ISBN: 978-989-8425-81-2
Copyright © 2011 SCITEPRESS (Science and Technology Publications, Lda.)

ered from researchers in the S-Cube NoE. The paper's main contribution is to describe how a large, multi-disciplinary research project went about ensuring a common understanding of research, capabilities and outputs by using a holistic and consistent knowledge management framework and a set of procedures for content validation and quality assurance.

The remainder of this paper is organized as follows: Section 2 provides the motivation for the KM through a brief introduction to the organization of S-Cube and related works; Section 3 briefly discusses the methodology followed for the construction of the KM; Section 4 describes how the methodology was used in the context of S-Cube to build the KM; Section 5 reports on the realization of the KM in terms of technologies used and functions offered; Section 6 presents details on how the contents of the KM has evolved; Section 7 provides a discussion of our experiences and findings; and, finally, Section 8 presents a set of conclusions and a description of future work.

SCIENCE AND TECHNOLOGY PUBLICATIONS

2 MOTIVATION

S-Cube is an EC initiative to bring researchers from different research domains together in an interdisciplinary approach to the study, design, and implementation of *service networks*. Research on service networks encompasses a broad range of academic and practical fields in the areas of Information Systems and Computer Science, such as Business Process Management (BPM), Grid computing and Software Engineering. Thus, research in this domain necessarily requires an interdisciplinary effort. S-Cube attempts to bridge these domains and bring together scientists and practitioners from different areas to perform research into service networks that cuts across traditional disciplines.

To achieve its goals, the S-Cube NoE has a comprehensive research framework that is split into joint research activities and associated tasks to support and promote the integration and dissemination of research (Papazoglou et al., 2010). The scope of S-Cube's Joint Research Activities (JRAs) is designed to cover the areas where future challenges will arise in software service-based systems and applications.

Figure 1 shows the conceptual relationship between the JRAs of S-Cube. Each block represents a JRA and the figure shoes how the research program positions three *service realization mechanisms* as a central 'spine' of research in three functional layers: service infrastructures, service composition and coordination and BPM. Together, these mechanisms cover how service-based applications are built, deployed,

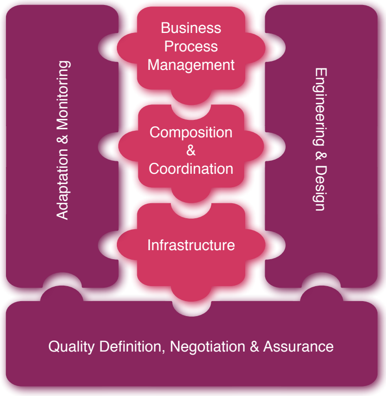

Figure 1: The S-Cube Joint Research Activities (JRAs).

composed and organized before being managed as business processes. The spine of service realization mechanisms is surrounded by three *cross-cutting concerns* applicable to each of the mechanisms, i.e., principles and techniques that should be applied in each of the functional layers. For example, each of the functional layers should be concerned with cross-cutting principles regarding how they are designed and engineered and also methods for defining, negotiating and assuring the quality of that layer. Likewise, each functional layer should also be concerned with how they are monitored and adapted (should the quality-of-service drop below an agreed level).

In addition to the research activities, S-Cube also contains a set of integration workpackages to promote interdisciplinary research across the JRAs, ensure the creation of a network of researchers in the field of software services and systems and assess the state and progress of integration. These are grouped into the areas of: spreading of excellence, integrating communities and integrating knowledge.

The S-Cube KM is part of the 'integrating knowledge' group of integration workpackages and was created to provide a common understanding at a terminological level of the wide range of knowledge required for SSME research. This requirement has come about as the same term is often used in different SSME research areas but with a contextual or domain-specific meaning. As a result, the multiple meanings of terms makes it difficult for researchers to communicate across research boundaries and enhances the fragmentation of research.

As we will describe in Section 4, each term is created as a result of the consolidation and reconciliation of conflicting or overlapping definitions of the vocabulary used in each JRA and the KM provides a method of defining associations between concepts, competencies and methodologies to position knowledge in relation to research domains. As a result, it helps achieve the goals of the integration workpackages by providing data through which the integration of research in S-Cube can be assessed.

Therefore, the KM aims to map, integrate and synthesize the diverse concepts and knowledge from the different JRAs and provide a resource that can be used not only as a reference point for teachers, researchers and practitioners but also as a tool to identify a network member's competencies (for mobility, spread of excellence, contact points and sources of information), to help classify research results (which assists in demonstrating research integration) and to illustrate the use of knowledge through associations with common scenarios and use cases.

2.1 Related Work

Before continuing with the description of the S-Cube KM, it is important to acknowledge methods of capturing the knowledge generated in other EC ICT research projects. For example, the NEXOF Reference Architecture (NEXOF-RA) project[3] has created a glossary containing a list of terms and definitions for Service Oriented computing (Stricker et al., 2009); the Business Experiments in Grid (BEinGrid) project[4] has developed *Gridipedia* ("the knowledge and toolset repository [...] to preserve and make accessible a whole range of resources on Grid computing and related technologies such as cloud computing and virtualization"); and the INTEROP project has developed a *Knowledge Map (KMap)* of competencies required for research into the interoperability of enterprise applications (Velardi et al., 2007).

However, as described in (Andrikopoulos et al., 2008), these previous efforts fail to meet the requirements of the S-Cube NoE. For example, the NEXOF-RA glossary has a domain-specific scope for terms, focusing on architectural and infrastructural knowledge, and follows a "flat" glossary (dictionary) format. Similarly, Gridipedia only focuses on the Grid and Cloud computing communities. In addition to containing knowledge of Grid computing, Gridipedia also contains downloadable Grid software components and solutions for common business problems. In comparison with this previous work, it was a requirement for the S-Cube KM to be applicable to many communities and the hosting of software components was not required. The S-Cube KM was to provide more dynamic, web- and encyclopedia-based approach that is broader in content coverage, focusing on semantic associations between concepts, approaches and methodologies and capturing associated information about the network, such as competencies and illustrations of how knowledge could be used.

[3]http://www.nexof-ra.eu

[4]http://www.beingrid.eu/

3 METHODOLOGY

We now describe the theoretical background for the development of the S-Cube KM. The methodology we used closely follows the activities necessary for the construction of a knowledge model described in (Schreiber and Wielinga, 1998), which illustrates how knowledge model development is decomposed to the sequential steps of knowledge *identification*, *specification* and *refinement*.

Figure 2 shows how the three stages proceed from the initiation of the KM and how the final stage of knowledge refinement is an ongoing, iterative task to integrate systematically different bodies of already codified knowledge in a formal, systematic manner. As we will describe in detail in the following sections, the effect of these three stages is to perform the *combination* of explicit knowledge contained in *deliverables*. Deliverables are project documents that either aggregate published/submitted papers and/or include original work, i.e. capture the knowledge produced by the network (see also Section 4). In this sense, deliverables perform the *externalization* of knowledge in Nonaka and Takeuchi's Four Modes of Knowledge Conversion model (Nonaka and Takeuchi, 1995). The *internalization* of explicit knowledge contained in the KM is carried out as the knowledge is internalized into a KM user's tacit knowledge-base, for example through applying the knowledge to their own work. The remainder of this section describes each of the steps from Figure 2 in more detail:

Knowledge Identification. In this phase the *information sources* to be used for knowledge modeling, such as glossaries, summaries and scenarios, are identified. In addition, existing model *components*, e.g., domain or task-related artifacts, are also found and evaluated for reuse during the modeling task.

Knowledge Specification. In the specification phase, a common representation of this information is developed. This is achieved through taking the model components identified in the previous phase and then "filling the holes" between components using the information sources.

Knowledge Refinement. The final phase of KM construction has the purpose of validating the knowledge model, refining the knowledge it contains and completing the knowledge base (to the extent that it is possible). As shown in Figure 2, the knowledge refinement phase is an iterative process. This fits with

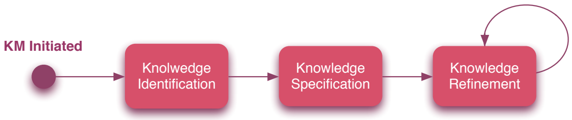

Figure 2: The KM Development Methodology.

previous experiences of architecting knowledge, such as (Boer et al., 2007), which described how architects commonly perform their activities (analysis, synthesis and evaluation) in an iterative fashion.

4 BUILDING THE KM

This section contains a description of how the phases of the general methodology from Section 3 were applied to the construction of the S-Cube KM, with each phase or sub-phase containing a description of the lesson (or lessons) learnt in completing it.

4.1 Knowledge Identification

For the initial stage of the KM development in S-Cube we identified:

Information Sources. In the initial phase of the network, each JRA completed an extensive literature survey that accumulated a significant amount of state-of-the-art knowledge in each area, some of which have subsequently been published as journal articles or book chapters. The results of these surveys acted as the initial, primary information source for the KM. In addition, research expertise of specific people and institutions within the network was identified.

Components. The design of the research framework and the outputs of the integration activities were the basic components for the KM. For example, the model of the structure and relationships between research domains defined in the contractual description of work and by the integration activities, provided a framework for the collection, description and publication of use cases that provides a context for knowledge items.

4.2 Knowledge Specification

For the S-Cube KM, this phase is translated into the following three steps:

4.2.1 Initial Version

The first step in compiling the KM was to ask each JRA to compile a short document describing the research performed in their area. This description had to contain a number of terms (keywords) considered important for the area, together with a separate, self-contained definition for each term. We also asked that when a definition was drawn from an existing source the source to be identified, or, in the case where no widely accepted definition existed, for an explicit original definition. After these terms and definitions were collected, we bootstraped the KM by organizing the information in a simple table format, described in (Andrikopoulos et al., 2008). This initial version of the KM was essentially a type of a dictionary. The competencies of each partner institution were added directly to each term, so as to connect it with one or more experts and/or institutions in the network.

For this first version a commercial spreadsheet software was used to collect and collate the knowledge from the JRAs. After the information was checked and edited where necessary by each JRA leader, it was incorporated in the S-Cube web portal as a set of static web pages. At this stage the KM was only accessible to the network participants and EC project officers responsible for overseeing and reviewing the project.

Lessons Learnt. Creating an initial version of the KM and assessing its strengths and weaknesses allowed us to derive the requirements for the 'final' KM format. It also allowed us to reduce the risk in moving to this format, as it allowed us think about what information each term should contain and refactor the knowledge accordingly (described in later lessons learnt).

4.2.2 KM Templating

While compiling the initial version of the KM we became aware of two major problems with the dictionary-style approach. Firstly, the same term may have different definitions in each research domain. For example, the term 'adaptation' has a different meaning in the context of Service Composition & Coordination — it refers to modifying a previously configured service composition — than the one used in the Adaptation & Monitoring JRA, where it applies to service-based applications in general.

Secondly, the dictionary failed to capture if/when a definition was either *domain-specific* or *context-specific*. A domain-specific definition is defined as an item of relevant knowledge that applies across all of the service realization mechanisms or all of the cross-cutting concerns for that term, whilst a context-specific definition applies across two JRAs. In the previous example, 'adaptation' has both a domain-specific definition (in the Adaptation & Monitoring JRA) and a context-specific definition (in the juncture of the Service Composition & Coordination and Adaptation & Monitoring JRAs) which supersedes the former one in the context of service compositions. As the initial version of the KM provided terms as simple documents, it could not express these properties or relationships.

Therefore, to record these different definitions and capture the relationships between domain- and context-specific knowledge we re-thought our approach and designed a grid-like template for terms, shown in Table 1. The grid is arranged with *service realization mechanisms* on the vertical axis and *cross-cutting concerns* on the horizontal axis; the position of a definition within this grid indicates its relationship to each JRA of the research framework. The arrangement of the JRAs like this allows us to capture multiple definitions and both domain- and context-specific information. For example, a definition for a term that is contextual between BPM and Engineering & Design (e.g., a methodology for designing business processes) is recorded in the top left cell of the grid. If a definition is domain-specific to Engineering & Design, it applies across all service realization mechanisms and the definition is placed in the bottom cell of the first column (in the Generic row) to indicate this. The right-most lower cell in the template, where the Generic row and column intersect, is used for terms whose definition is applicable across all JRAs areas; e.g., ‘software service’. In addition, placeholders where competencies, validation scenarios and references to publications/deliverables can be recorded were attached to each term template.

Lessons Learnt. We found that definitions from different research domains differ not necessarily because of ambiguity of meaning but because they may address different contexts. In ontology engineering, differences in definitions usually indicate ambiguities and a lack of cross-fertilization of knowledge and their removal is encouraged (Shvaiko and Euzenat, 2008). In our case we preserve the differences in definitions to capture domain *and* contextual information, and the term definitions are codified across these two dimensions.

4.2.3 Public Release

Once this template had been evaluated and agreed by the members of S-Cube, we proceeded to migrate the set of terms from the initial version of the KM to the template, reviewing and adding definitions to them where appropriate. At the same time, while we were completing this process, each of S-Cube’s JRAs was asked to revisit the state-of-the-art reports already produced to determine further knowledge that could be added to the KM. These reports, together with other first-year project deliverables, were processed by the domain experts within the network, isolating and defining important terms and attaching partner competencies to them. The initial conceptualization of the S-Cube KM was concluded with its release on the network’s web portal and was the first publicly accessible version of the KM. This was done in May 2009, following a successful review of the project by the EC.

Lessons Learnt. Having knowledge sources (i.e., state-of-the-art reports) immediately at-hand accelerated the bootstrapping of the KM, despite the element of duplication that was introduced as knowledge was recorded in the state-of-the-art reports and again in the KM.

4.3 Knowledge Refinement

Since the first public release of the KM, we have performed two major refinement cycles, each resulting in a new version of the KM with one more expected by the end of the project. The motivation for carrying out these cycles is not only to add content to the KM with new knowledge produced by the network, but also to ensure the *quality* of the KM terms. Furthermore, as described earlier, a major goal for the KM is the identification of gaps and overlaps in knowledge produced by different domains, which provides feedback about the status of research in the network and ensures KM *consistency*. Since the KM is an evolving resource, we recognized the need for a clear set of repeatable processes to ensure the continuous growth, quality and consistency of the KM. To this end we defined and implemented three major actions that are carried out at regular intervals:

The KM Update Process. To ensure its quality, each deliverable is reviewed both internally by different project members and externally by reviewers assigned by the EC. An approved/accepted deliverable contains new or updated terms and definitions that need to be recorded in the KM. The same procedure as before is used to extract knowledge from a deliverable and enter it into the KM, with the only difference being that the person contributing to the KM should check for existing terms and definitions before adding them. As each JRA produces deliverables in a different periodic cycle, this complicates the application of this process. For this reason we decoupled it completely from the production of deliverables and the process is triggered at regular intervals independent of the JRAs deliverable cycle. However, this approach introduces an element of reproduction

Table 1: S-Cube Knowledge Model Template.

Research Theme		Technology Principles, Techniques & Methodologies			
		Engineering & Design (ED)	Adaptation & Monitoring (AM)	Quality Definition Negotiation & Assurance (QA)	Generic or Domain-specific
Service Technologies	Business Process Management (BPM)				
	Service Composition & Coordination (SC)				
	Service Infrastructure (SI)				
	Generic or Domain-specific				

as knowledge is first captured in the deliverable and then again in the KM. Unfortunately, this is a direct result of EC reporting requirements that mandate the presentation of knowledge in deliverable (i.e., document) format.

Lessons Learnt. The completion of terms was helped by two factors: when deliverables contained a *glossary of terms*, the entry of definitions was speeded up as it was easy for the person entering the data to copy-and-paste the definitions into the KM; and, as the deliverables used as knowledge sources had already been reviewed both in the internal authoring process and externally by the EC, the quality of the knowledge being added to the KM was already assured, which also helped the person creating or completing a term. Regarding the coordination of who entered which terms and definitions, we found letting people self-organize themselves within the JRAs to process deliverables was preferable to telling them what should be done and when. This 'bottom-up' approach provided a *sense of ownership* of terms, research domains and the KM in general, as contributors essentially became stakeholders.

Quality Assurance (QA) Process – Format. The manual addition of definitions to terms and additional information for associated competencies, validation scenarios and references to publications/deliverables, led to mistakes in the formatting of some terms. We found the most common problems to be:

- Typos, grammatical and expression mistakes.
- Misuse of the term template, such as entering information outside of the grid/template.
- Definitions that were copy-pasted directly from a deliverable sometimes only made sense in the context of the deliverable it was taken from.
- References to deliverables or publications that contain the definitions provided were missing.

We addressed these issues by designing a straightforward QA procedure that was applied by manually checking each page: following the completion of the data-entry stage of the process, each term in the KM was assigned to a contributor (not the person who entered or modified the term during the KM update process) who checked the term for these problems and modified it if necessary. To distribute the terms between S-Cube partners fairly, we developed a simple algorithm that automatically and randomly assigns a proportional number of terms to each partner for checking based on the amount of stated effort the partner is willing to put into the QA process.

Lessons Learnt. Manual addition and editing of the KM content generates mistakes and requires a manual QA process. As this process was carried out after each round of KM update, terms were checked and quality-assured in a regular cycle and formatting mistakes were not long-lasting. Errors due to manual editing were mitigated by iterative QA, as well as by codifying the knowledge structure (in our case through templates). Also, as contributors became familiar with how the template should be used we found that the number of these errors has decreased with each entry cycle.

Quality Assurance Process – Content. The second aspect to QA is checking the accuracy of the

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

knowledge entered in the KM. We observed early in the KM's lifecycle that too many domain-specific terms were being entered at the expense of context-specific terms. This was reflected by a dense concentration of definitions in the shaded area of Table 1. The origin of this problem was found to be a result of the design of the research framework around focused research domains, which meant researchers were unsure of how their contribution applied to other domains. In particular, while questioning contributors to the KM they stated that they were uncomfortable adding a context-specific definition to a term except if they were an expert in both domains. In our opinion, this is a more general problem concerning how people behave in large groups of experts and is a larger issue outside the scope of our analysis.

Therefore, to ensure an even distribution of knowledge in the KM, we designed a process which normalizes and rationalizes the distribution of definitions within the terms, 'landscaping' the recorded knowledge. To do this, we classified each S-Cube partner as specializing in either service realization techniques (rows in Table 1) or cross-cutting concerns (columns in Table 1). We then classified terms as being either domain-specific to service realization techniques or cross-cutting concerns depending on their definitions. Domain-specific terms in service realization techniques were assigned to partners who specialized in cross-cutting concerns and vice-versa to ensure as much as possible that 'new eyes' viewed the term. As in the case of the format-focused QA process above, we assigned the terms to partners based on each partner's intended effort (in person months) to determine the proportion of terms they should 'landscape'. Once a partner received their set of terms to validate, we asked them to determine for each term if each domain-specific definition could be replaced with a context-specific definition. If it could be, we asked them to modify the term accordingly.

Lessons Learnt. After we assigned terms to institutions for checking we found further evidence of self-organization. Terms were swapped between institutions so the right expert could address a particular definition. This also supports our finding that people were uncomfortable editing terms that they did not feel to be an expert in. Furthermore, delegation of responsibility to empowered teams per institution allowed them to autonomously forward KM terms to the 'right' peers. This practice requires that teams know who knows what, a known best practice in knowledge sharing (Clerc et al., 2007).

We initially applied the QA processes for formatting and content to all the terms in the KM. However, as the number of terms in the KM grew, the workload on members of S-Cube carrying out the QA processes increased. To reduce the effort required we prioritized the checking of terms modified since they were last subjected to QA (i.e., terms not modified since the last round of QA were not re-checked) and those which had only domain-specific definitions. We determined which terms to check and which to leave out of the process by developing a set of tools to 'crawl' the KM, find simple formatting errors and determine the distribution of definitions within the term. These tools, and the implementation of the KM, are described as part of the next section.

5 IMPLEMENTATION

5.1 Platform

For the implementation and publication of the S-Cube KM we used the Plone content management system (CMS) (McKay, 2009). Plone is a versatile free and open source CMS oriented towards web page publishing but also supporting document publishing and groupware applications. Four reasons led us to using Plone as the platform for the KM:

1. There was a heavy investment in implementing the project's web portal using Plone. In contrast, there was no provision in the budget of the project for separate infrastructure for the KM.
2. Other knowledge management solutions, such as the Simple Knowledge Organization System (SKOS), were immature at the time we were seeking a KM platform. A separate platform would also require integration with the main project portal, which would require additional effort.
3. Plone offers wiki-like capabilities like collaborative editing and versioning that were deemed to fulfill our purposes.
4. The use of a CMS like Plone for the KM means that data input and editing is done through the browser. No familiarization effort with a particular tool (further of the one for learning the template) is therefore necessary and the learning curve is shallow and short.

Lessons Learnt. Using an out-of-the-box CMS system was cheaper in terms of time and money than developing a new application with database schema, persistence methods, middleware and presentation layer. As the KM was implemented in the same CMS as the network's web portal, there was no overhead for integrating the KM with a primary dissemination channel of S-Cube. This increased the *visibil-*

ity of the KM and in time the KM became synonymous with the S-Cube NoE. Regarding the adoption of the platform, partners within the network were more than happy to contribute their knowledge using Plone and this ensured the steady accumulation of knowledge and growth in the total number of items. We feel that this success was mainly a result of the KM being implemented using a standard, what-you-see-is-what-you-get CMS that had the same 'look-and-feel' as the project's web portal.

5.2 Functions

The implementation of the KM based on Plone provides to the KM users and contributors with the following standard functions, which are accessible in a tab-based view:

View displays the actual content, along with page creation information. Also allows for direct editing of the term by double-clicking on the page.

Edit allows the editing of the term.

Sharing provides the option to make the term visible to and editable by particular users.

History shows a description of changes to the term per user and date, allowing for differencing between versions and offering roll-back to a previous version.

These functions are available only to registered users of the S-Cube Web portal, allowing us with a better control over the KM modifications. Unregistered users of the KM Web page are offered only the view function.

5.3 Applying the KM Template

In order to apply the KM Template discussed in the previous section, a standard page was created containing an empty KM term template. Each new KM term was (and still is) created by copying this template page, completing the term-specific details and saving as a new page. Each term is represented in a single web page and has a unique URL, allowing for indexing, bookmarking and sharing of the URL. A subdomain in the web portal was created to contain these pages, publicly accessible at http://www.s-cube-network.eu/km.

Lessons Learnt. The copy-and-paste of text from various sources to the CMS behaved differently depending on which browser is used by inserting hidden characters in the pasted text. These characters are not directly a problem since they are simply not rendered to KM viewers. However, they become important when we use automated tools (discussed below) to extract the information they contain for data analysis and reporting.

5.4 Tool Support

As part of the KM development and QA processes, we created a set of command-line scripts to check the format and content of the KM and ensure its general consistency. As the KM is deployed within the S-Cube web portal and access to the main Plone database is restricted, the tools were developed to 'crawl' the KM by following the hyperlinks found in the KM term index and retrieve each web page representing a term, parse the HTML and create a local in-memory representation. Using this representation allows the formatting and content analysis of the KM terms and the automatic report generation that support the management of knowledge in the KM. They have also allowed us to perform the transformation of the KM into different formats, e.g., XML dialects such as GraphML, which was used in an attempted visualization of the relationships between concepts.

Lessons Learnt. Although 'screen-scraping' is often seen as an archaic method of gathering data, this approach has worked well for us: building tools in a programming language known for its speed of development (i.e., Ruby (Flanagan and Matsumoto, 2008)) allowed us to spend a minimum amount of time on developing supporting tools for the KM. The standard structure of each term meant that once the main body of code for retrieving and processing web pages was written it only needed to be extended to generate new reports.

6 EVOLUTION

In principle, the evolution of the KM is driven by the progress of the network, i.e., the more knowledge produced by the partners, the more content (terms, definitions, competencies and references, etc.) is added to the KM. The development of the KM content over time is summarized by Figure 3. Measurements are shown at 7 different points in the network's lifespan that coincide with deliverable releases, project reviews and contractual milestones, where Month 1 of the network corresponds to March 2008. There are two major growth periods in Figure 3, around month 13 and month 33, marking the production of two major versions of the KM (the first public release and the *consolidated* version, respectively).

The latest version of the KM (month 39 of the project) contained 688 definitions across 419 terms with 215 recorded competencies. Due to the nature of

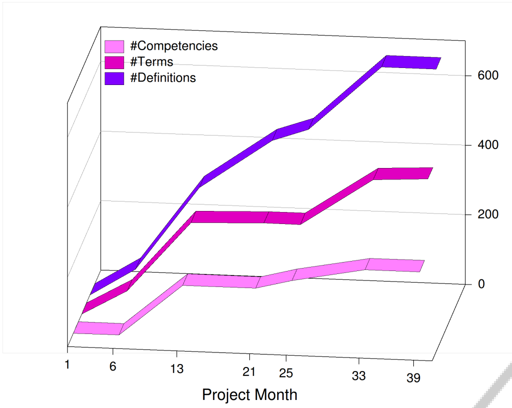

Figure 3: Evolution of the KM.

the processes discussed in Section 4, there were very few removals of terms from the KM, with the emphasis being on reconciliating and landscaping problematic terms — which also partially contributes to the continuous growth of the KM shown in Figure 3. Overall, Figure 3 reflects the periodical nature of the activity in the KM on behalf of its contributors where short periods of intense activity and growth are followed by longer periods of smaller activity and corrective actions. This is to be expected since the evolution of the KM is aligned with the network's timeline, defined by the deliverable schedule and important milestones at specific intervals (e.g., annual reviews).

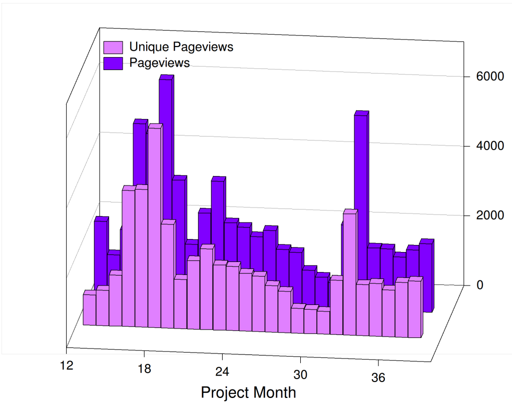

Figure 4: KM Visits – Source: Google Analytics^{TM}

In terms of public visibility of the KM, Figure 4 shows the number of visits to the KM subdomain of the S-Cube Web portal as reported by Google Analytics. The KM receives on average approximately 2500 page views and 2000 unique page views (i.e., number of visits during which one or more of the KM pages were viewed) per month since going public. The KM is visited from users in more than 100 countries, with USA, India, Philippines and Canada in the top 10. Note that the support tools discussed in Section 5 access the KM via an external HTTP connection and do not effect the data gathered by the Google Analytics service used to provide metrics of the KMs use and popularity.

Lessons Learnt. In most months, the difference between page views and unique page views is relatively small, meaning that KM visitors view just a few pages per visit, as shown in Figure 4. This can be explained by the lack of *direct links* between terms in the KM, which discourages viewers from navigating between them, as one would when browsing Wikipedia. We are planning to address this deficiency of the KM by adding (hyper)links between terms as part of the final version.

7 FURTHER LESSONS LEARNT

Generally, we have found the development of the S-Cube KM to be remarkably smooth: conflicts between researchers, such as those regularly observed between Wikipedia authors fighting to have 'their' knowledge accepted, were minimized due to several factors (listed in no particular order):

- The grid structure of each term allowed multiple definitions to co-exist within the same term.
- The production of the state-of-the-art reports in the initial phase of the network gave S-Cube researchers the opportunity to understand their partners research strengths, weaknesses and perspectives and use them in later stages of the project.
- Most discussions over the finer points of terminology had already taken place as the deliverables were being produced so by the time they entered the KM most ambiguities, discrepancies and controversies had already been removed.
- The network's emphasis on regular face-to-face meetings to encourage joint research allowed researchers to start 'speaking the same language' more quickly than if their relationships had been remote. In terms of impact to the KM, the shared understanding of the objectives of S-Cube created by these meetings facilitated the KM construction.

We feel the success of the KM can be also demonstrated in examples of how the KM has been and is currently being used as:

A Point of Reference. For example, the terms and definitions from the S-Cube KM were used as a commonly accepted glossary in (Dustdar and Li, 2011).

A Teaching Aid. The KM has been used in the S-Cube Virtual Campus[5] to tag course material so they can be re-used by lecturers and students across teaching modules.

An Accepted Knowledge Source. As discussed in Section 6, the KM has been accessed by researchers from many countries seeking an accepted definition for a term. In this sense, the KM has provided publicly-available reference material for researchers in SSME and helped to align research across domains.

A Hub for Other EC Knowledge-related Activities. The S-Cube KM has been adopted by other EC-funded projects, e.g., the Hola! co-ordination activity[6] intends to build a repository of structured knowledge using KM terms from projects in the SSME area.

8 CONCLUSIONS & FUTURE WORK

This paper presents our experiences in developing a Knowledge Model (KM) for S-Cube, a large, pan-European research network that brings together scientists and practitioners from different areas to carry out fundamental interdisciplinary research in SSME. The KM aims to map, integrate and synthesize various concepts from different research areas, facilitates research by consolidating and reconciling overlapping definitions used by each research area and provides a resource that can be used as a reference point, teaching aid and a hub for project activity.

The variety of communities involved and the differences in how they use terminology led us to design a template to capture knowledge that allows the positioning of knowledge within a domain and context. We implemented this template using the content management system (CMS) used to provide the network's web portal and developed an iterative methodology to accumulate knowledge captured in project deliverables (documents that aggregate existing knowledge and/or contain existing work) in the template. To address issues with mistakes due to manual editing and uneven distribution of knowledge across the KM we developed and applied appropriate QA processes at regular intervals.

By observing the KM's evolution over many months we can conclude that network members were happy to contribute their knowledge to the KM, resulting in a successful product. The number of accesses to the (publicly available) S-Cube KM from all over the world is evidence of this success. During the different phases of the KM construction we had to take a number of design and management decisions. It is our intuition that some of the lessons learnt in these decisions are more applicable/useful for large communities (like S-Cube) and distributed/virtual ones (as in the case of Wikipedia) than others (e.g., enterprises). Further investigation of our findings are necessary to provide empirical evidence of this intuition. In addition, we plan to investigate visualization techniques for the KM that will make it more accessible to users.

[5] http://vc.infosys.tuwien.ac.at/

[6] http://best149.best-center.external.hp.com/eu/node/12

REFERENCES

Andrikopoulos, V., van den Heuvel, W.-J., and Papazoglou, M. P. E., editors (2008). *CD-IA-1.1.1 - Comprehensive Overview of the State of the Art in Service-Based Systems*. S-Cube NoE.

Boer, R., Farenhorst, R., Lago, P., van Vliet, H., Clerc, V., and Jansen, A. (2007). Architectural Knowledge: Getting to the Core. In *Software Architectures, Components, and Applications*, volume 4880 of *LNCS*, pages 197–214.

Clerc, V., Lago, P., and van Vliet, H. (2007). Global software development: Are architectural rules the answer? In *ICGSE 2007 Proceedings*, pages 225–234.

Dustdar, S. and Li, F., editors (2011). *Service Engineering : European Research Results*. Springer-Verlag, Vienna.

Flanagan, D. and Matsumoto, Y. (2008). *The Ruby Programming Language*. O'Reilly Media.

McKay, A. (2009). *The Definitive Guide to Plone*. Expert's Voice in Open Source. APress.

Nonaka, I. and Takeuchi, H. (1995). *The Knowledge Creating Company*. Oxford University Press.

Papazoglou, M., Pohl, K., Parkin, M., and Metzger, A., editors (2010). *Service Research Challenges and Solutions for the Future Internet: S-Cube - Towards Engineering, Managing and Adapting Service-Based Systems*, volume 6500 of *LNCS*. Springer-Verlag.

Schreiber, A. and Wielinga, B. (1998). Knowledge Model Construction. In *Proceedings of the 11th Workshop on Knowledge Acquisition, Modeling and Management*, Banff, Canada.

Shvaiko, P. and Euzenat, J. (2008). Ten Challenges for Ontology Matching. In *On the Move to Meaningful Internet Systems*, volume 5332/2008 of *LNCS*, pages 1164–1182. Springer.

Stricker, V., Heuer, A., Zaha, J. M., Pohl, K., and Panfilis, S. (2009). Agreeing Upon SOA Terminology — Lessons Learned. In *Towards the Future Internet - A European Research Perspective*, pages 345–354. IOS Press.

Velardi, P., Navigli, R., and Petit, M. (2007). Semantic Indexing of a Competance Map to support Scientific Collaboration in a Research Community. In *IJCAI'07 Proceedings*, pages 2897–2902.