Grant Agreement N° 215483

| | |
|---|---|
| *Title:* | *Virtual Learning Centre* |
| *Authors:* | *UniDue, Tilburg, CITY, INRIA, Lero, Polimi, SZTAKI, TUW, UOC,UPM, USTUTT, VUA* |
| *Editor:* | *Harald Psaier (TUW)* |
| *Reviewers:* | *Philipp Leitner (TUW)* |
| | *Martin Treiber (TUW)* |
| *Identifier:* | *CD-SoE-1.1.6* |
| *Type:* | *Contractual Deliverable* |
| *Version:* | *1.0* |
| *Date:* | *31 May 2011* |
| *Status:* | *Final* |
| *Class:* | *External* |

## Management Summary

This deliverable will establish the virtual learning centre for open learning purposes, including e-learning facilities for use by both academic and industry trainees.

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

## Members of the S-Cube consortium:

| | |
|---|---|
| University of Duisburg-Essen (Coordinator) | Germany |
| Tilburg University | Netherlands |
| City University London | U.K. |
| Consiglio Nazionale delle Ricerche | Italy |
| Center for Scientific and Technological Research | Italy |
| The French National Institute for Research in Computer Science and Control | France |
| Lero - The Irish Software Engineering Research Centre | Ireland |
| Politecnico di Milano | Italy |
| MTA SZTAKI – Computer and Automation Research Institute | Hungary |
| Vienna University of Technology | Austria |
| Université Claude Bernard Lyon | France |
| University of Crete | Greece |
| Universidad Politécnica de Madrid | Spain |
| University of Stuttgart | Germany |
| University of Hamburg | Germany |
| Vrije Universiteit Amsterdam | Netherlands |

## Published S-Cube documents

All public S-Cube deliverables are available from the S-Cube Web Portal at the following URL:

http://www.s-cube-network.eu/results/deliverables/

# The S-Cube Deliverable Series

## Vision and Objectives of S-Cube

The Software Services and Systems Network (S-Cube) will establish a unified, multidisciplinary, vibrant research community which will enable Europe to lead the software-services revolution, helping shape the software-service based Internet which is the backbone of our future interactive society.

By integrating diverse research communities, S-Cube intends to achieve world-wide scientific excellence in a field that is critical for European competitiveness. S-Cube will accomplish its aims by meeting the following objectives:

- Re-aligning, re-shaping and integrating research agendas of key European players from diverse research areas and by synthesizing and integrating diversified knowledge, thereby establishing a long-lasting foundation for steering research and for achieving innovation at the highest level.
- Inaugurating a Europe-wide common program of education and training for researchers and industry thereby creating a common culture that will have a profound impact on the future of the field.
- Establishing a pro-active mobility plan to enable cross-fertilisation and thereby fostering the integration of research communities and the establishment of a common software services research culture.
- Establishing trust relationships with industry via European Technology Platforms (specifically NESSI) to achieve a catalytic effect in shaping European research, strengthening industrial competitiveness and addressing main societal challenges.
- Defining a broader research vision and perspective that will shape the software-service based Internet of the future and will accelerate economic growth and improve the living conditions of European citizens.

S-Cube will produce an integrated research community of international reputation and acclaim that will help define the future shape of the field of software services which is of critical for European competitiveness. S-Cube will provide service engineering methodologies which facilitate the development, deployment and adjustment of sophisticated hybrid service-based systems that cannot be addressed with today's limited software engineering approaches. S-Cube will further introduce an advanced training program for researchers and practitioners. Finally, S-Cube intends to bring strategic added value to European industry by using industry best-practice models and by implementing research results into pilot business cases and prototype systems.

S-Cube materials are available from URL: http://www.s-cube-network.eu/

# Contents

# 1 Introduction

The Virtual Campus, or Virtual Learning Centre (the terms are used interchangably through the remainder of this deliverable), is a part of the integration activity SoE-1.1 of the S-Cube Network of Excellence. The Virtual Campus aims to use a flexible combination of communication and collaboration technologies to encourage and facilitate knowledge exchange, news circulation, debate, feedback to support the interaction and dissemination in the community of Software Services and Systems. The Virtual Campus platform should enhance the cooperation and facilitate the integration of research of geographically distributed organizations of different countries and different research perspectives related to Service-oriented computing (SOC). Besides the integration of research, the Virtual Campus aims to provide a distributed and flexible e-Learning environment supporting the co-joined master and PhD programs of S-Cube as well as to maintain the basis for further cooperation and integration especially in teaching in the SOC Community and related communities. The current deliverable CD-SoE-1.1.6 can be seen as a continuation effort of deliverable CD-SoE-1.1.2 [1]. CD-SoE-1.1.2 presented our initial design of the Virtual Campus platform. Here, we will now describe our first deployed version of this tool. Most of the design statements of CD-SoE-1.1.2 are still valid, however, in detail some deviations from the originally envisioned design have been necessary. The aim of the current deliverable, as part of the spread of excellence activity, is to describe the implementation of the main dissemination platform of the S-Cube project in detail.

Please note that this deliverable describes the Virtual Campus as presented during the Y3 review meeting. The recommendations resulting from the review are mostly not reflected in this document, but can be seen in the final Virtual Campus platform version as delivered at the end of the S-Cube project. Please find Some further information on this topic in Section 3.

## 1.1 Presentation of this Deliverable

In the remainder of this deliverable, we will compare many tools which have been evaluated to implement parts of the Virtual Campus. In general, we will use the following pattern. Firstly, we present the main requirements for each part of the platform. Secondly, we present some alternatives which came into consideration. For simplicity, we will be using the 5 general dimensions `Functionality`, `Accessibility`, `Usability`, `Adaptability` and `Administration` to compare alternatives. We assign numerical values in $[1 : 5]$ to each alternative in each dimension, 1 being best. All of these ratings are to some extend subjective, and should not be understood as absolute values. These comparisons have been mostly translated and adapted from [2]. Finally, we present the selected tool and show what it looks like in the deployed platform.

## 1.2 Relationship to Other Workpackages

The work package SoE-1.1 Virtual Campus and this deliverable in particular are related to following non-research workpackages:

- WP-IA-1.2 establishes the Pan-European Distributed Service Laboratory (EDSL). EDSL will "support the testing, integration, evaluation and benchmark SBA relevant emerging technologies, methodologies and tools"; EDSL will also "support the communication within and across communities by providing a common forum for access and dissemination of resources, including intermediate and final results of the joint research activities, online publications, computing resources and infrastructure." Virtual Campus is related to the EDSL since there are a variety of intersections and commonalities, which will be reflected in future deliverable documents accordingly.

- WP-IA-1.1 aims to provide a comprehensive overview of the "state of the art and an indepth analysis of the research areas relevant to service based applications (SBAs)". WP-IA-1.1 will provide "the initial definition and incremental evolution of the Convergence Knowledge Model". Thus the results of this WP will be adopted by in the Knowledge Base of the Virtual Campus; the Virtual Campus will constantly be updated to include the valid and update version of the Knowledge Model.

- WP-SoE-1.2 "Community Outreach" focuses on the dissemination of research results of S-Cube Consortium i.e. to the wider community. This WP comprises also activities related to electronic dissemination including creation and maintaining of the S-Cube Web portal. The Virtual Campus will maintain links to contents provided on the S-Cube portal in order to avoid redundant copies of information. These relationships to other work packages may be a subject of update during the course of the project and will be reflected in the subsequent deliverables.

# 2 The Virtual Learning Center

The following section describes the current state of the learning center. At the beginning an overview of all modules comprised by the center is given. Details follow in the subsequent subsections and are structured as follows. First, the requirements that motivated the module are stated. Second, the implementation of the module is presented and the combination of the applied tools described. Third, screenshots document the most interesting features of any module.

## 2.1 Introduction to the Domains

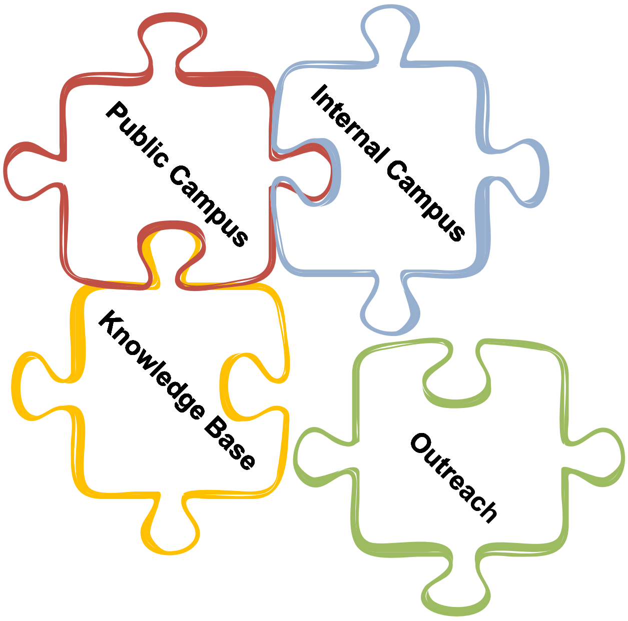


Figure 1: Virtual Campus Domains

Figure 1 lists the three domains of the Virtual Campus that are the focus of the deliverable, namely, *Public Campus*, *Internal Campus*, and *Knowledge Base*. As shown by the figure these modules are already implemented, released, and interconnected (with, e.g., single system registration). However, the fourth domain resulted from recent developments around the Virtual Campus, and in particular with Y3 review recommendations. These consider the

Virtual Campus not only an e-learning platform with forum, course management, and research work management, but **in addition** as a means for the *Outreach* to interested students and researches in the projects results. It is still up to the S-Cube partners to define the content and the glue with the current domains. As this new perspective has established recently and, thus, was not included the DoW for the present deliverable, the authors decided to refer to the requirements only in the concluding section of the document.

The structure of the document is as follows. Next a detailed overview of the three completed domains is given. In Section 2.3 the public part of the Virtual Campus is presented, followed by Section 2.4 outlining the Knowledge Base. Section 2.5 describes the internal campus course systems. The final sub-section describes the relations to joint PhD and Master Programmes. Section 3 concludes the deliverable and in future work first thoughts about the aforementioned *Outreach* domain are presented.

## 2.2 Overview

Figure 2 shows the basic structure of the Virtual Campus. The Campus is separated into three logical domains. Orthogonal to the three domains of the Virtual Campus, there are means to provide news feeds relevant to all three domains.

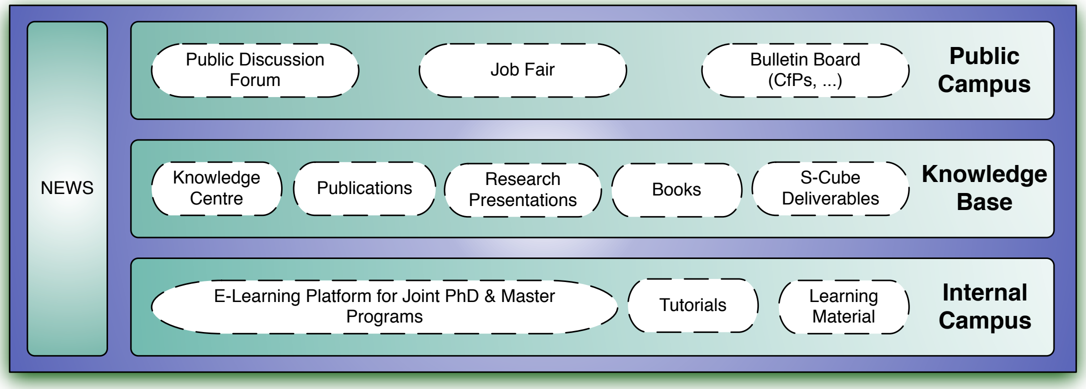


Figure 2: Virtual Campus Overview

1. The *Public Campus* is the "public meeting place" of the Virtual Learning Centre. Content in the Public Campus is accessible for everybody, and everybody is invited to contribute and share his/her ideas. We envision the Public Campus to become the central meeting place for the SOC community in large, students, researchers and practitioners alike. Therefore, the most important component of this domain is a public discussion forum. Furthermore, the Public Campus provides additional community services.

2. The idea of the *Knowledge Base* is to provide a useful interface to disseminate various types of knowledge. Initially, this includes the publications and deliverables produced within the S-Cube project, as well as presentations that have been given (either as keynotes or as research presentations about conference papers). At a later point we envision that the Knowledge Base can also be used to provide an integrated view on other types of publications, such as books written (e.g., the first S-Cube book [3]), tutorials or other supplementary material. Furthermore, we expect that other research projects related to SOC may use the Knowledge Base to disseminate their results. The wider public only has read access to the Knowledge Base, i.e., the possibility to actually

insert contents will initially be restricted to S-Cube-internal content managers. As part of the outreach activities (see Figure 1), it is envisioned to open up the Knowledge Base to external entities, e.g., other EC projects.

3. The *Internal Campus* is a private space in the Virtual Campus reserved for participants in the joint european master's and PhD programmes, as well as participants of internal courses of S-Cube partners handled via the Virtual Campus. The Internal Campus serves as an e-learning platform for those programmes. To that end, it provides means to put online tutorials and learning material, which (for copyright reasons) cannot be made entirely public in the Knowledge Base. Furthermore, all typical e-learning facilities, such as tools to manage student registrations, assignments, schedules and grading, are provided.

The basic framework of the Virtual Campus is formed by a Content Management System (CMS). The CMS is used as a wrapper and "programming environment", on top of which the actual components of the platform will be installed and/or developed.

### 2.2.1 CMS – Challenges and Requirements

We do not require a specifically large set of CMS features for the Virtual Campus. However, given the open nature of the platform we required the following specific features:

1. *Open Source Software.* Evidently, we aim for open source software in order to keep the Virtual Campus solution free from licensing costs.

2. *Easy extensibility.* Given that we did not expect pre-existing components to be available for all required features of the Virtual Campus we needed a CMS that allows us to easily integrate external tools, such as the research dashboard that we implemented for the Knowledge Base domain. Furthermore, we do not yet know for sure if further extensions to the platform will become necessary in the lifetime of the S-Cube project, or beyond.

3. *Based on standard programming languages.* In our experience, maintenance of systems based on well-known languages such as PHP or Java is easier than maintaining systems that need more specialized knowledge, such as knowledge of the typoscript language.

4. *Large existing component library.* The larger the amount of components available, the less components needed to be implemented within S-Cube.

5. *Simple user management.* In order to integrate all tools belonging to the Virtual Campus under a single-sign-on umbrella, it is very helpful if the user management approach used by the selected CMS is reasonably clear and simple, so that it can be integrated with other tools.

### 2.2.2 Implementation and Tools

Even though there are a plethora of CMS systems, both open source and commercial, available, the three which are most commonly named are Typo3[1], Drupal[2] and Joomla![3]. All of those CMS are open source software.

Table 1 compares these three alternative CMS. Based on this comparison, we have chosen to build the Virtual Platform on top of Joomla!. Our decision was mainly based on the simplicity of Joomla!, as compared to the more powerful Typo3. Additionally, Joomla! components and extensions are available for most of the features that we require for the three domains of the Virtual Learning Centre.

[1] http://typo3.com/

[2] http://drupal.org/

[3] http://www.joomla.org/

| Tool | Dimension | Rating | Reason |
|---|---|---|---|
| **Typo3** | | | |
| | Functionality | 1 | Most powerful CMS in the test, custom language typescript is very powerful |
| | Accessibility | 1 | Valid HTML code is generated, which is easy to parse for restricted devices, strong support for important accessibility features, such as access keys |
| | Usability | 3 | Notoriously difficult to “get into”, very text-oriented |
| | Adaptability | 2 | In principle very extensible, but oftentimes quite complex typescript code is necessary to extend Typo3 |
| | Administration | 1 | Installation and administration is straight-forward |
| **Drupal** | | | |
| | Functionality | 2 | Multi-site support, WYSIWYG editor, but less features than Typo3, bad support for SSL and FTP |
| | Accessibility | 2 | Valid HTML code is generated, which is easy to parse for restricted devices |
| | Usability | 2 | Somewhat confusing web interfaces, very text-oriented |
| | Adaptability | 2 | In principle very extensible, oftentimes modules have compatibility problems with newer or older versions of Drupal |
| | Administration | 2 | Installation and administration is straight-forward, but more limited than the Typo3 administration tooling |
| **Joomla!** | | | |
| | Functionality | 2 | Very large feature set, but somewhat less powerful than the best tools in this dimension |
| | Accessibility | 1 | Valid HTML, special template “beez” for restricted devices |
| | Usability | 1 | Easy to use and very well-organized, intuitive |
| | Adaptability | 1 | Very large component library available, which is mostly downward compatible, installation of new modules is straight-forward |
| | Administration | 1 | Very simple installation procedure (wizards) |

Table 1: Comparison of CMS Systems

### 2.2.3 Presentation of Results

The initially deployed version of the Virtual Campus is publicly available[4]. Figure 3 depicts a screen shot of the Joomla!-based platform implementation. As we can see, the layout, colouring and general formatting has been kept as close as possible to the standards set by the official S-Cube web page[5].

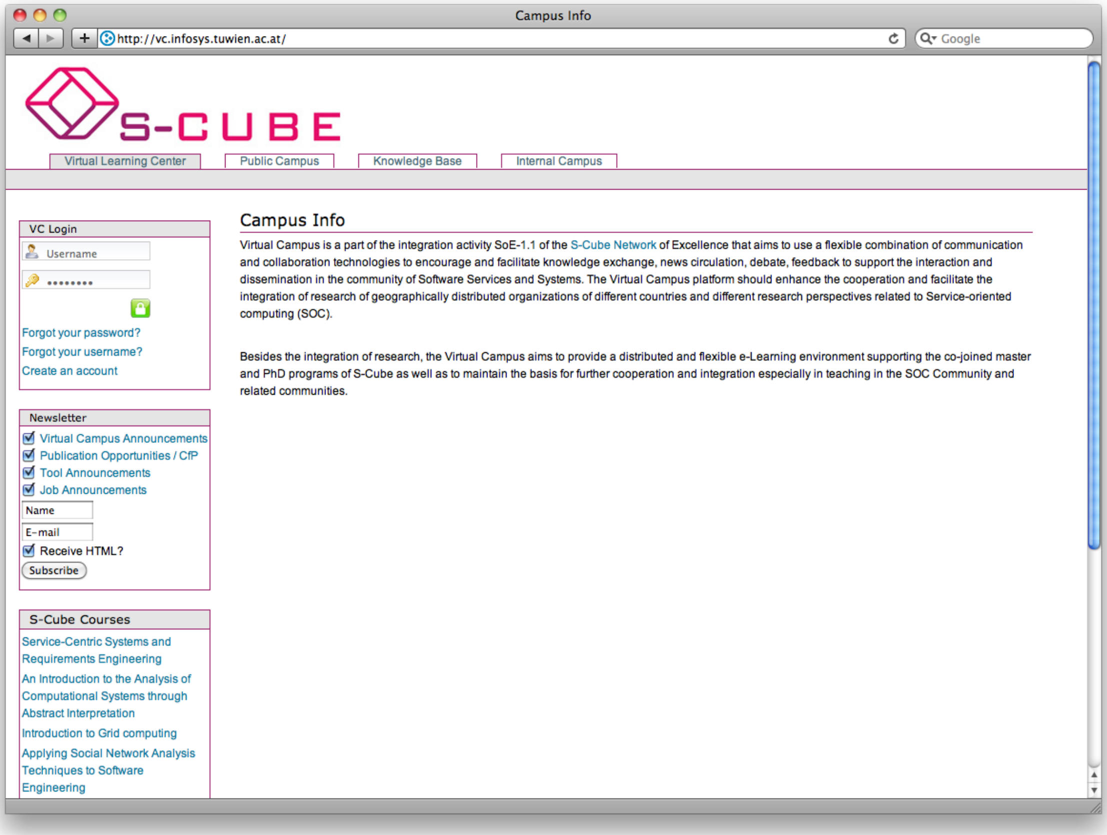
On the left-hand side of the screen we can see a login screen, a possibility to register for the Virtual Campus newsletter, and an overview over the courses currently running in the Private Campus. On the top of the screen, different tabs can be used to access the three domains of the Virtual Campus.

## 2.3 Public Campus

The public campus is the open part to the campus and provides the visitor with initial information about the e-learning platform. The goal of the Public Campus is to provide all visitors, be it researchers or practitioners, from different domains with an integrated discussion and information area for S-Cube related teaching areas. Through Public Campus they are able to obtain comprehensive and up-to-date information regarding the progress

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

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


Campus Info
http://vc.infosys.tuwien.ac.at/
Google
S-CUBE
Virtual Learning Center
Public Campus
Knowledge Base
Internal Campus
VC Login
Username
••••••••
Forgot your password?
Forgot your username?
Create an account
Newsletter
☑ Virtual Campus Announcements
☑ Publication Opportunities / CfP
☑ Tool Announcements
☑ Job Announcements
Name
E-mail
☑ Receive HTML?
Subscribe
S-Cube Courses
Service-Centric Systems and Requirements Engineering
An Introduction to the Analysis of Computational Systems through Abstract Interpretation
Introduction to Grid computing
Applying Social Network Analysis Techniques to Software Engineering

Campus Info

Virtual Campus is a part of the integration activity SoE-1.1 of the S-Cube Network of Excellence that aims to use a flexible combination of communication and collaboration technologies to encourage and facilitate knowledge exchange, news circulation, debate, feedback to support the interaction and dissemination in the community of Software Services and Systems. The Virtual Campus platform should enhance the cooperation and facilitate the integration of research of geographically distributed organizations of different countries and different research perspectives related to Service-oriented computing (SOC).

Besides the integration of research, the Virtual Campus aims to provide a distributed and flexible e-Learning environment supporting the co-joined master and PhD programs of S-Cube as well as to maintain the basis for further cooperation and integration especially in teaching in the SOC Community and related communities.

Figure 3: Virtual Campus Home Screen

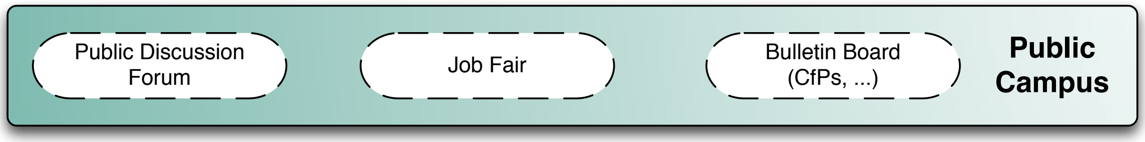


Figure 4: Public Campus Overview

in the corresponding areas; to share ideas, to cooperate, and to interact with each other by means of public forums and announcement mechanisms. Moreover, one of the main goals of the Public Campus is to introduce and present the Virtual Campus as a whole, promoting its mission and facilities to a wide range of young and senior researchers from different countries, institutions, and areas of interest.

In order to accomplish these goals, the Public Campus comprises different information modules and targets various research and application areas. With the Service-Oriented Computing domain atop, the areas are related, however, widespread and individual as the contributing partners are. Thus, they include such SOC domain familiar research fields as Grid and Cloud computing, Business Process Management, Software Engineering, or Human Computer Interaction.

### 2.3.1 Challenges and Requirements

The main challenge of the public campus is to support efficient and simple communication and data sharing between researchers and students. For the public campus, we defined a set of six requirements (see Table 2 PCs).

| ID | Type | Tag | Description |
|---|---|---|---|
| **PC1** | Non Functional | Software | To avoid license costs, we require open source software |
| **PC2** | Functional | Blog and Lists | Members should be able to write blog entries about topics of their interest. It should be also possible to add members to arbitrary mailinglists. |
| **PC3** | Functional | User Management | Forums require member management. Not all members have the same rights and roles should be supported to provide forum moderation. |
| **PC4** | Functional | Edit, Create and Update | To avoid a steep learning curve, content should be entered with a WYSIWIG editor.It should be possible to modify existing Forum structures on demand. Users should be able to upload files and to share data between Forum members. |
| **PC5** | Functional | Search | Users should be able to search in Forum and Blog posts. |
| **PC6** | Functional | Messaging | The system should support the exchange of private messages between virtual campus members. |

Table 2: Requirements of the Public Campus

### 2.3.2 Implementation and Tools

In recent years, a plethora of open source Forum software products were developed. Among the most used are phpBB and Phorum. Both are based on the same software components; a MySQL database for storing the content in combination with a PHP scripting environment for generating Web pages for management and presentation issues. In contrast to these stand alone Forum solutions, we also investigated the Kunena Forum system, which is an extension to the Kunena Content Management System. In addition, we require access to information that cannot efficiently be managed with Forum software. The Forum software of the Public campus is complemented with external pages which are managed by CMS. CMS and their challenges and requirements are discussed in detail in previous Section 2.2.1 and the reader be referred to this section for detailed information. Instead, here we focus our discussion on

**phpBB.** The development of phpBB began in 2000 as open source project. In the meantime, the software reached Version 3.0.6. Generally speaking, phpBB is a very powerful tool to manage Forums. It supports the unlimited creation of Forums and categories. The management console of phpBB provides the means to manage user permissions on individual and group level. Private messages can be sent between Forum members and ”misbehaving” Forum members can be tracked and warned. This is done with the help of a sophisticated Forum rule system that enforces the following of the Forum etiquette. Users can configure Templates arbitrarily, adapting the system to individual needs.

The installation of phpBB requires 9.4 MB of disk space and is guided with an integrated setup assistant. The setup assistant takes care of tasks like creating a database to store the Forum content and supports the modification of the user interface for seeing impaired users. A check with the W3C Markup Validation service revealed the standard conformity of phpBB Web pages.

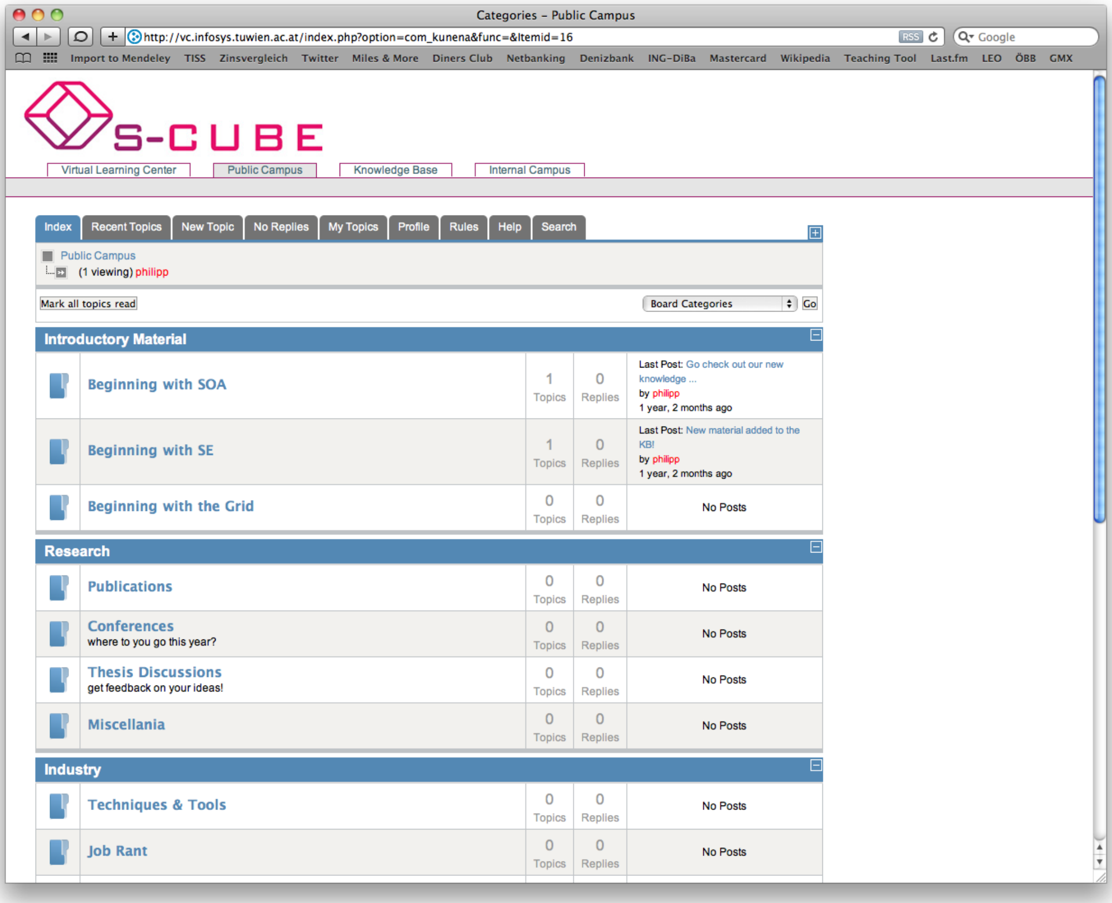
**Phorum.** Like phpBB, Phorum is an open source solution. It has been developed since 1998 and has reached Version 5.2.14. Phorum follows a modular architecture and can be extended, but its functionality is smaller compared to phpBB. Like phpBB, Forum does provide template mechanisms to customize the user interface, but the administration presents itself in a more text-oriented manner: there is no setup assistant that guides the user through the process and the user is required to edit configuration files manually (e.g., to set up the connection to the database).

**Joomla and Kuena.** Unlike phpBB and Phorum, Joomla is a CMS and thus has a different design focus and offers different functionality. However, Joomla’s functionality can be extended with Kunena Forum component. Kunena is fully integrated into the Joomla CMS, providing a seamless synchronization between Kunena and Joomla users. The main benefit of such an integrated solution is the simultaneous access to CMS and Forum functionality, simplifying the use of the software. A feature that is not available in Forum software are newsletters and mailinglists. CMSs like Joomla provide build-in support for these features and the activation does not require an additional setup procedure. However, the default installation, does not support advanced functions like the management of multiple mailinglists. This required theuseof the Acajoom extension which allows for the registration for multiple mailing lists.

Both stand-alone Forum systems (phpBB and Phorum) come with the needed functionality to satisfy the core requirements of the Public Campus. A direct comparison between phpBB and Phorum makes phpBB the winner, because of phpBB's simple and efficient installation and management and pre-configured components that help to cope with problems like spam. A closer investigation revealed, that phpBB and Phorum cannot seamlessly integrated with CMS. A possible solution to this was the use of additional Browser windows, but this approach has negative impact on the usability. Consequently, the combined use of a Forum with other Public Campus content such as Information pages, mailinglist or calendars, is best managed by a CMS and dedicated moduls/extensions.

### 2.3.3 Presentation of Results

The look and feel of the Public Campus is aligned to common forums. We think this a feasible solution to attract guest visitors because it presents something most Internet users are familiar with.



Public Campus
(1 viewing) philipp

| | | | |
|---|---|---|---|
| **Introductory Material** | | | |
| Beginning with SOA | 1<br>Topics | 0<br>Replies | Last Post: Go check out our new knowledge ...<br>by philipp<br>1 year, 2 months ago |
| Beginning with SE | 1<br>Topics | 0<br>Replies | Last Post: New material added to the KB!<br>by philipp<br>1 year, 2 months ago |
| Beginning with the Grid | 0<br>Topics | 0<br>Replies | No Posts |
| **Research** | | | |
| Publications | 0<br>Topics | 0<br>Replies | No Posts |
| Conferences<br>where to you go this year? | 0<br>Topics | 0<br>Replies | No Posts |
| Thesis Discussions<br>get feedback on your ideas! | 0<br>Topics | 0<br>Replies | No Posts |
| Miscellania | 0<br>Topics | 0<br>Replies | No Posts |
| **Industry** | | | |
| Techniques & Tools | 0<br>Topics | 0<br>Replies | No Posts |
| Job Rant | 0<br>Topics | 0<br>Replies | No Posts |

Figure 5: Public Campus Forum

Figure 5 shows the following forum functionality:

- *Index.* Gives an overview of the most discussed topics.
- *Recent Topics.* This tab helps to filter and lists only the most recent topics.
- *Rules.* This tab allows administrators to set the forums rules (and users to view these rules)
- *Help.* This tab helps to filter and lists only the most recent topics.
- *Search.* Allows to search the forum.

## 2.4 Knowledge Base

This Section discusses the Knowledge Base of the Virtual Campus. The tasks of the knowledge base are twofold: on the one hand, it organizes and displays consolidated material produced by S-Cube and others to the S-Cube relevant areas; on the other hand, associated (additional) material is stored and presented along with the original materal. The Knowledge Base is made publicly available to participating institutions and the SOC community. We expect that it becomes the first point of access for academics, researchers, industry developers and students to access up-to-date SOC knowledge as produced by the S-Cube project. Figure 6 displays a summary of the content managed in the Knowledge Base.

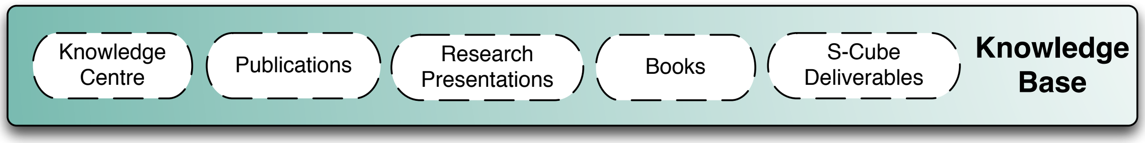


Figure 6: Knowledge Base Overview

In the current, initial version of the tool we use the Knowledge Base to manage the S-Cube publications, presentations and deliverables. In the future, we expect to open up the Knowledge Base for more, and different, material, such as books or publications produced in the scope of other european research projects. The Knowledge Base has been implemented as a plugin-based, Web-based research dashboard.

### 2.4.1 Challenges and Requirements

As one of the core tools of the Virtual Campus, the Knowledge Base serves as a dissemination and education platform for the S-Cube project at large. Table 3 outlines the requirments

### 2.4.2 Implementation and Tools

In the original design of the Virtual Campus [1], a WIKI system such as MediaWIKI[7] in combination with a document management tool such as Remository[8] was envisioned to implement the Knowledge Base. However, using this combination of tools a full implementation of the requirements listed in Section 2.4.1 would have been quite cumbersome. Specifically, the original design envisioned that all data and metadata would be collected in WIKI pages while the document manager holds the actual content. Reading material online would have been hard in such a setting, and one can expect that the WIKI-held metadata would be quite unstructured and, hence, effectively useless for automated searching, grouping and

[7] http://www.mediawiki.org/wiki/MediaWiki

[8] http://remository.com/

| ID | Type | Name | Description |
|---|---|---|---|
| **KB1** | Functionality | Provisioning | The main requirement of the Knowledge Base is to make teaching and research material accessible in a structured way. This does not necessarily mean that all material is physically hosted in the Virtual Campus, as this would be infeasible for copyright reasons. Similarly, not all material is necessarily publicly available. However, even if some material such as copyright-protected research papers is not directly hosted in the Knowledge Base, at least some direct links are maintained (e.g., links to the paper in IEEEXplorer[6]), where the paper can be easily accessed by persons or institutions with sufficient access privileges to do so. |
| **KB2** | Functionality | Links and Metadata | The Knowledge Base should link teaching and research material with potentially available supplementary data or material. For instance, research papers should be presented along with their metadata (authors, keywords, links to the S-Cube Knowledge Model [4]) as well as presentation slides, if slides are available. This allows to search and classify material, and provides added value as compared to e.g., a regular publication database. |
| **KB3** | Functionality | Searching and Browsing | The Knowledge Base should not be only a directory of teaching and research material. Much more, the Knowledge Base was envisioned as an integrated dashboard, which allows to not only find but also consume material. Hence, it should be possible to browse through the collection online, as well as read and print without having to e.g., download papers first. Given sufficient privileges, this should be possible even for material that cannot be hosted directly in the Knowledge Base. |
| **KB4** | Functionality | Flexibility and Extensibility. | Finally, the Knowledge Base should be easy to extend with new features. |

Table 3: Requirements of the Knowledge Base

categorization. Furthermore, these WIKI entries would have to be created and maintained manually.

For these reasons, we have decided to implement a custom-built tool for the Knowledge Base. We refer to this tool as the S-Cube Research Dashboard. The Research Dashboard is a web application developed using Grails[9] technology, with a MySQL[10] powered relational database backend. To ease the development of the tool a number of high-quality plugins have been made use of. These have been summarized in Table 4.

| Plugin | Version | Functionality |
|---|---|---|
| Hibernate | 1.3.1 | Provides Hibernate support for Grails |
| Include | 0.3 | Simple plugin that adds include behaviour to Grails, which is not supported by default |
| Mail | 0.9 | Allows Grails applications to easily send mails |
| Nimble | 0.4 | A powerful plugin which kickstarts Grails applications with often-used functionality. In the Research Dashboard, Nimble was mostly used to implement the user authentication |
| Remote-Pagination | 0.2.3 | Allows AJAX-based pagination of content |
| Richui | 0.8 | Provides a set of nice-looking and powerful AJAX components, for instance drag-and-drop portlets |
| Shiro | 1.0.1 | Apache Shiro (Java security framework) integration for Grails |

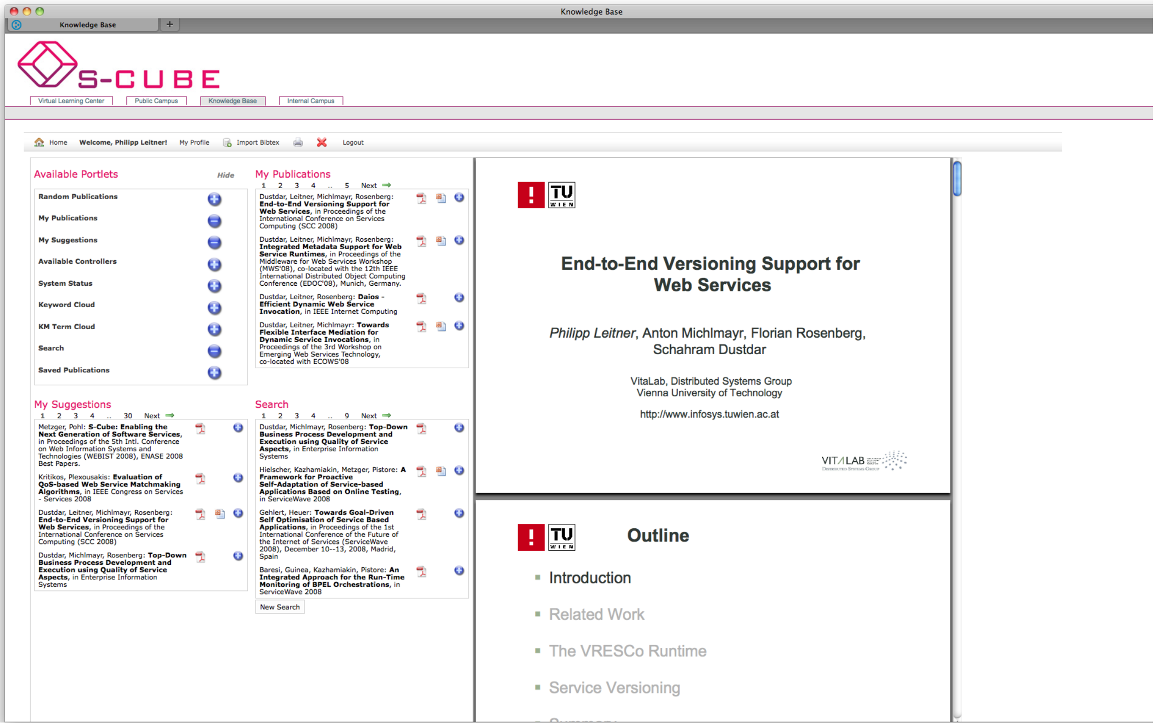
Table 4: Grails Plugins Used in the Research Dashboard

[9] http://www.grails.org/

[10] http://www.mysql.com

### 2.4.3 Presentation of Results

In essence, the Research Dashboard implements two core functionalities, organization and presentation of research material on the one hand, and a primitive reader on the other. Both functionality is implemented as a AJAX-basedWeb application and integrated into the larger Virtual Campus framework. Figure 7 shows the main screen of the dashboard. As we can see, the screen is divided into a portlet area (left-hand side) and an integrated reader (right-hand side).



My Publications

Dustdar, Leitner, Michlmayr, Rosenberg: **End-to-End Versioning Support for Web Services**, in Proceedings of the International Conference on Services Computing (SCC 2008)

Dustdar, Leitner, Michlmayr, Rosenberg: **Integrated Metadata Support for Web Service Runtimes**, in Proceedings of the Middleware for Web Services Workshop (MWS'08), co-located with the 12th IEEE International Distributed Object Computing Conference (EDOC'08), Munich, Germany.

Dustdar, Leitner, Rosenberg: **Daios - Efficient Dynamic Web Service Invocation**, in IEEE Internet Computing

Dustdar, Leitner, Michlmayr: **Towards Flexible Interface Mediation for Dynamic Service Invocations**, in Proceedings of the 3rd Workshop on Emerging Web Services Technology, co-located with ECOWS'08

My Suggestions

Metzger, Pohl: **S-Cube: Enabling the Next Generation of Software Services**, in Proceedings of the 5th Intl. Conference on Web Information Systems and Technologies (WEBIST 2008), ENASE 2008 Best Papers.

Kritikos, Plexousakis: **Evaluation of QoS-based Web Service Matchmaking Algorithms**, in IEEE Congress on Services - Services 2008

Dustdar, Leitner, Michlmayr, Rosenberg: **End-to-End Versioning Support for Web Services**, in Proceedings of the International Conference on Services Computing (SCC 2008)

Dustdar, Michlmayr, Rosenberg: **Top-Down Business Process Development and Execution using Quality of Service Aspects**, in Enterprise Information Systems

Search

Dustdar, Michlmayr, Rosenberg: **Top-Down Business Process Development and Execution using Quality of Service Aspects**, in Enterprise Information Systems

Hielscher, Kazhamiakin, Metzger, Pistore: **A Framework for Proactive Self-Adaptation of Service-based Applications Based on Online Testing**, in ServiceWave 2008

Gehlert, Heuer: **Towards Goal-Driven Self Optimisation of Service Based Applications**, in Proceedings of the 1st International Conference of the Future of the Internet of Services (ServiceWave 2008), December 10--13, 2008, Madrid, Spain

Baresi, Guinea, Kazhamiakin, Pistore: **An Integrated Approach for the Run-Time Monitoring of BPEL Orchestrations**, in ServiceWave 2008

TU WIEN

**End-to-End Versioning Support for Web Services**

*Philipp Leitner*, Anton Michlmayr, Florian Rosenberg, Schahram Dustdar

VitaLab, Distributed Systems Group
Vienna University of Technology

http://www.infosys.tuwien.ac.at

TU WIEN **Outline**

- Introduction
- Related Work
- The VRESCo Runtime
- Service Versioning

Figure 7: Knowledge Base Main Screen

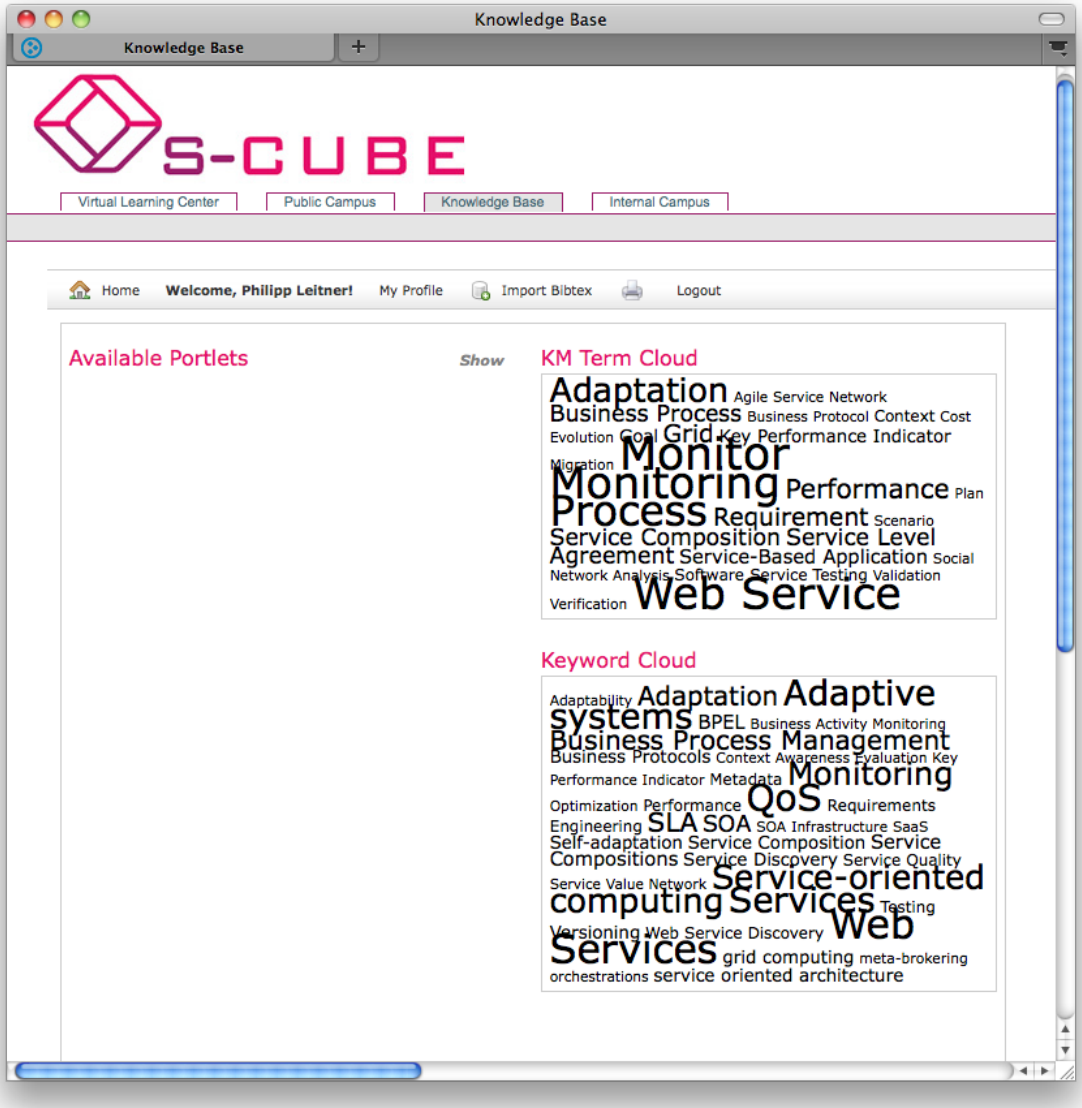
Technically, the reader is nothing else than an integrated iframe[11] which displays whatever media the user selects in the portlet area. As such, the capabilities of the reader are naturally limited to the feature set that the browser plugin displaying the content provides. The portlet area on the left-hand side is more interesting. The portlet area is a free space that can be filled with portlets, small applets with specific functionality. At the moment we provide the following portlets:

- *My Publications* displays all material that the currently logged in user has (co-)authored.
- *Random Publications* displays a small number of random entries from the database.
- *My Suggestions* displays some suggested papers based on interests that the logged in user has specified in her profile. Users can specify interest in S-Cube Knowledge Model terms or keywords.
- *Search* allows to issue searches on authors, title and abstract, and displays the results of these searches.
- *Knowledge Model Tag Cloud* visualizes the usage of Knowledge Model terms as tag cloud. A term is displayed bigger if it is associated with more material in the database.

[11] http://www.w3schools.com/tags/tag_iframe.asp

- *Keyword Tag Cloud* is the analog to the Knowledge Model Tag Cloud for keywords.
- *Saved Publications* storesand displays material that the logged in user has marked as "saved".

The tag clouds used in the Knowledge Model and Keyword Tag Cloud portlets are depicted in the screenshot in Figure8. If one is to click on one of the terms, all material associated with this term or keyword is depicted in a special list view (see Figure9). In this list view the portlets are hidden, instead the full left-hand area of the screen is used to display e.g., the material associated with a term. The reader stays unchanged in this view.



KM Term Cloud

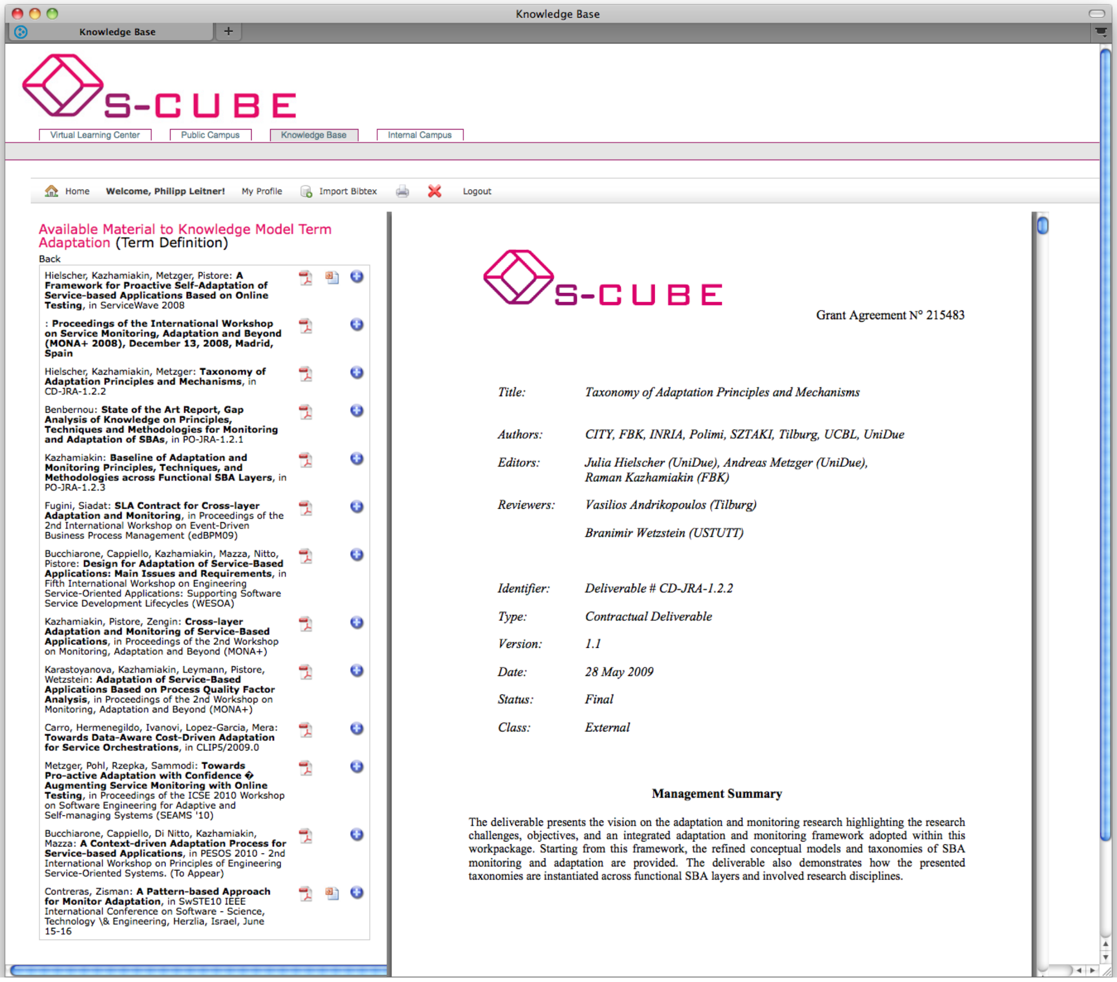
Adaptation Agile Service Network Business Process Business Protocol Context Cost Evolution Goal Grid Key Performance Indicator Migration Monitor Monitoring Performance Plan Process Requirement Scenario Service Composition Service Level Agreement Service-Based Application Social Network Analysis Software Service Testing Validation Verification Web Service

Keyword Cloud

Adaptability Adaptation Adaptive systems BPEL Business Activity Monitoring Business Process Management Business Protocols Context Awareness Evaluation Key Performance Indicator Metadata Monitoring Optimization Performance QoS Requirements Engineering SLA SOA SOA Infrastructure SaaS Self-adaptation Service Composition Service Compositions Service Discovery Service Quality Service Value Network Service-oriented computing Services Testing Versioning Web Service Discovery Web Services grid computing meta-brokering orchestrations service oriented architecture

Figure 8: Knowledge Base Tag Cloud Visualization

Furthermore, the list of available portlets is realized as a portlet itself. The logged in user can customize her view freely by selecting and deselecting portlets, and re-arranging them via drag-and-drop. Additionally, the size of the portlet area can be customized in the profile (the number of horizontal and vertical portlet slots can be selected). This is useful to customize the Research Dashboard to the user screen resolution. New portlets can be added easily. Authentication for the Research Dashboard works via the general Virtual Campus, i.e., there is no separate login necessary.



Available Material to Knowledge Model Term Adaptation (Term Definition)

Back

Hielscher, Kazhamiakin, Metzger, Pistore: **A Framework for Proactive Self-Adaptation of Service-based Applications Based on Online Testing**, in ServiceWave 2008

: **Proceedings of the International Workshop on Service Monitoring, Adaptation and Beyond (MONA+ 2008), December 13, 2008, Madrid, Spain**

Hielscher, Kazhamiakin, Metzger: **Taxonomy of Adaptation Principles and Mechanisms**, in CD-JRA-1.2.2

Benbernou: **State of the Art Report, Gap Analysis of Knowledge on Principles, Techniques and Methodologies for Monitoring and Adaptation of SBAs**, in PO-JRA-1.2.1

Kazhamiakin: **Baseline of Adaptation and Monitoring Principles, Techniques, and Methodologies across Functional SBA Layers**, in PO-JRA-1.2.3

Fugini, Siadat: **SLA Contract for Cross-layer Adaptation and Monitoring**, in Proceedings of the 2nd International Workshop on Event-Driven Business Process Management (edBPM09)

Bucchiarone, Cappiello, Kazhamiakin, Mazza, Nitto, Pistore: **Design for Adaptation of Service-Based Applications: Main Issues and Requirements**, in Fifth International Workshop on Engineering Service-Oriented Applications: Supporting Software Service Development Lifecycles (WESOA)

Kazhamiakin, Pistore, Zengin: **Cross-layer Adaptation and Monitoring of Service-Based Applications**, in Proceedings of the 2nd Workshop on Monitoring, Adaptation and Beyond (MONA+)

Karastoyanova, Kazhamiakin, Leymann, Pistore, Wetzstein: **Adaptation of Service-Based Applications Based on Process Quality Factor Analysis**, in Proceedings of the 2nd Workshop on Monitoring, Adaptation and Beyond (MONA+)

Carro, Hermenegildo, Ivanovi, Lopez-Garcia, Mera: **Towards Data-Aware Cost-Driven Adaptation for Service Orchestrations**, in CLIP5/2009.0

Metzger, Pohl, Rzepka, Sammodi: **Towards Pro-active Adaptation with Confidence � Augmenting Service Monitoring with Online Testing**, in Proceedings of the ICSE 2010 Workshop on Software Engineering for Adaptive and Self-managing Systems (SEAMS '10)

Bucchiarone, Cappiello, Di Nitto, Kazhamiakin, Mazza: **A Context-driven Adaptation Process for Service-based Applications**, in PESOS 2010 - 2nd International Workshop on Principles of Engineering Service-Oriented Systems. (To Appear)

Contreras, Zisman: **A Pattern-based Approach for Monitor Adaptation**, in SwSTE10 IEEE International Conference on Software - Science, Technology \& Engineering, Herzlia, Israel, June 15-16

S-CUBE

Grant Agreement N° 215483

| | |
|---|---|
| *Title:* | *Taxonomy of Adaptation Principles and Mechanisms* |
| *Authors:* | *CITY, FBK, INRIA, Polimi, SZTAKI, Tilburg, UCBL, UniDue* |
| *Editors:* | *Julia Hielscher (UniDue), Andreas Metzger (UniDue),*<br>*Raman Kazhamiakin (FBK)* |
| *Reviewers:* | *Vasilios Andrikopoulos (Tilburg)*<br>*Branimir Wetzstein (USTUTT)* |
| *Identifier:* | *Deliverable # CD-JRA-1.2.2* |
| *Type:* | *Contractual Deliverable* |
| *Version:* | *1.1* |
| *Date:* | *28 May 2009* |
| *Status:* | *Final* |
| *Class:* | *External* |

**Management Summary**

The deliverable presents the vision on the adaptation and monitoring research highlighting the research challenges, objectives, and an integrated adaptation and monitoring framework adopted within this workpackage. Starting from this framework, the refined conceptual models and taxonomies of SBA monitoring and adaptation are provided. The deliverable also demonstrates how the presented taxonomies are instantiated across functional SBA layers and involved research disciplines.

Figure 9: Knowledge Base Material to Keyword

## 2.5 Internal Campus

This section describes the S-Cube Virtual Campus internal module, the Internal Campus. The idea of the Internal Campus is to provide an E-Learning platform based on a course management system for the students in S-Cube's joint Master & PhD Program. On release the module provides a fully functional campus that offers across an E-Learning Platform various courses. Students can register with the provided courses. Teaching personal is capable of offering selective material for their provided courses. Furthermore, teachers are enabled to interact with their students using the platform tools. Thus in particular, the tools of the Internal Campus must provide means of creating new, updating, and managing courses. Once created the application must allow to link related learning material to the courses. Additionally, tools must be provided to enhance the courses with tests and examination material, as well as, feedback questionnaires. Registered course members must be able to communicate among each other. This relates to the course participants, but participants must also be able to ask questions across the platform directed to the course leader. In addition to those features administration of the Internal Campus must, in detail, focus on access, create, and update rights management. Overall, Figure 10 presents the central blocks of the Internal Campus.

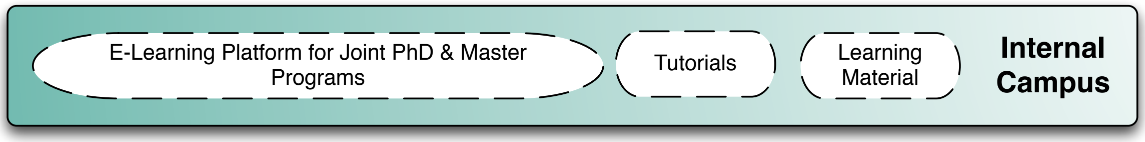


Figure 10: Internal Campus Overview

In S-Cube the Internal Campus is used to provided course and teaching material of the different involved partners. Related to the S-Cube partners collaboration, the Internal Campus will provide a jointly maintained repository for the collection of various material in different categories reflecting the partners' individual knowledge on the field of SOA. As such a complete collection it is a important support for young researchers and Master/PhD Students to immerse themselves in the wide area. Also, teaching personnel can include the material of the partners in their own courses for an extension or create completely new courses from scratch with the collected material. Furthermore, with the collection courses as a base also new courses can be developed jointly with the contribution of a team of teachers. Next, challenges and requirements are detailed.

### 2.5.1 Challenges and Requirements

Here we present the challenges and requirements of the Internal Campus. The next Table 5 provides a list of the most important requirements.

| ID | Type | Name | Description |
|---|---|---|---|
| **IC1** | Functionality | Management | Internal Campus must provides user-friendly means to manage the collection of material. Means include create, update, and role management functions |
| **IC2** | Functionality | User Interface | Internal Campus must be accessible with current standard web applications. Thus, for the client side a standard web browser is the best choice to represent the campus interface |
| **IC3** | Functionality | Communication | Internal Campus must provides means of communication between the members. Not only must members in the same role be allowed to send messages to each other, but also, across the roles. For example, a student must have the possibility to ask questions to a course leader |
| **IC4** | Functionality | Collaboration | Internal Campus must allow members to collaborate. With the help of the Internal Campus students must be enable to collaborate on courses by cooperating in assignments. Also, students give feedback on courses and collaborate in quality management. Teacher must be supported in creating, and managing offered courses together. |
| **IC5** | Functionality | Repository | Internal Campus must allow members to manage their learning/teaching material. With the help of the Internal Campus teachers can create a repository of learning material for their students that are than able to enable to learn from the provided sources. |

Table 5: Requirements of the Internal Campus

The listed requirements are all equally important as they reflect some of the main ideas of the S-Cube project. Apart from satisfying all stated requirements, the challenge in composing the Internal Campus to a modern, fully functional, interactive E-Learning platform is to find existing tools that are ready to use, find the missing parts, and integrate the parts to a final complete application.

### 2.5.2 Implementation and Tools

The implementation of the Internal Campus is included in the Virtual Campus spanning CMS with all the aforementioned features and building blocks. In detail, it combines a CMS with a Joomla! wrapper. At the beginning two CMSs where considered and analyzed focusing on their usability and possible integration difficulties. The features and selection of Joomla! have already been motivated in detail in Section 2.2.2. Next, a better insight into the two considered CMS is given.

**Moodle**[12] is the abbreviation for *Modular Object-Oriented Dynamic Learning Environment*. Moodle is a software package for producing Internet-based courses and web sites. It is a global development project designed to support a social constructionist framework of education. Moodle is provided freely as Open Source software (under the GNU Public License). The Moodle application is based on PHP, and requires a database, for example MySQL. It can be run on any modern operating system. Moodle was build for education. The focus is to provide a tool that joins both student or teacher studying or teaching in an online course. Statistics show a high interest in the applications. Up to the recent version 1.9.7 45.733 installations were registered.

A Moodle platform is aware of different access roles. Thus, depending on the user's role different start configurations are available. Here we detail as an example the initial requirements to a course creator role. S/He is presented initially with different possibilities to manage the courses. There is, however, two main activities such a user needs to complete first. When creating a new course the courses resources and learning material must be provided. Next, the learn activities must be specified.

Currently Moodle recognizes the following types of course materials:

- *textpage.* These are simple plain text notes for a quick publication.
- *webpage.* Is common HTML based and formated information.
- *link to resources.* This includes in particular the possibility to link to on-line content, such as, webpages and files for download, or whole directories containing course materials.
- *IMS-content-package.* This is a XML based type specification that allows to create standardized and system independent content-packages.
- *label.* A label structures the courses various contents under one unique tag.

Furthermore, Moodle comprises the following modules that support the creation of learning content:

- *assignments.* These allow teaching personnel to collect work of students via an upload feature and later to give an evaluation and feedback.
- *charts.* Provide the means of on-line discussions with synchronous messaging.
- *database.* Allows to create table based content with individual fields including teaching material essential figures, references, and related data.
- *forum.* Offers a discussion platform related to the created course and allows to relate the presented topics to those in the forum.
- *glossary.* Provides the possibility to publish a list of definitions related to the course.
- *lesson.* Is HTML-page collection prototype that allows to create an on-line version of a whole lesson.

[12] `http://moodle.org/`

- *quiz and choice.* Allows to create more general tests including, in particular, multiple choice tests.
- *SCORM.* SCORM is the *Sharable Content Object Reference Model* and allows to interconnect different learning units with similar content.
- *survey.* Provides the course participants with the feature of giving feedback to the teacher.
- *wiki.* Finally, the modules also include the possibility to set up a wiki collaboration platform for the course.

Generally, the Moodle application allows to provide course material and learning activities. Further, once created a courses access right can be managed as necessary. Also, courses provide modules, such as, forums, test forms, or charts. The design and development of Moodle is guided by a "social constructionist pedagogy". This concept can be explained by four main, related attributes: *constructivism, constructionism, social constructivism, and connected and separate.* From a constructivist point of view, users can actively construct new knowledge as they interact with their environments. Constructionism asserts that learning is particularly effective when constructing something for others to experience. This can be anything from a spoken sentence or an internet posting, to more complex artifacts like a painting, a house or a software package. Social constructivism extends constructivism into social settings, wherein groups construct knowledge for one another, collaboratively creating a small culture of shared artifacts with shared meanings. When one is immersed within a culture like this, one is learning all the time about how to be a part of that culture, on many levels. Finally, connected and separate means that there is two types of behavior. Separate behaviour is when someone tries to remain 'objective' and 'factual', and tends to defend their own ideas using logic to find holes in their opponent's ideas. Connected behavior is a more empathic approach that accepts subjectivity, trying to listen and ask questions in an effort to understand the other point of view. Moodle application tries to support both and advocates a healthy combination of both approaches.

**ILIAS**[13] is a German CMS product developed by the University of Cologne. It has similar features to Moodle. However, one particular feature of the application is a Personal Desktop. Similar to an individual user account it is the main point of reference for an user and provides all important information. A calendar allows subscription to appointments and a task management. Search features allow to search through documents. The shown comments for a user can be organized public and private. The desktop also includes the standard mail and repository features with ftp/http access.

For the Virtual Campus, the combination Moodle and Joomla! proofed adequate and integrated smoothly. All of the requirements listed in the previous section could be included. The currently used components are listed in the following Table 6:

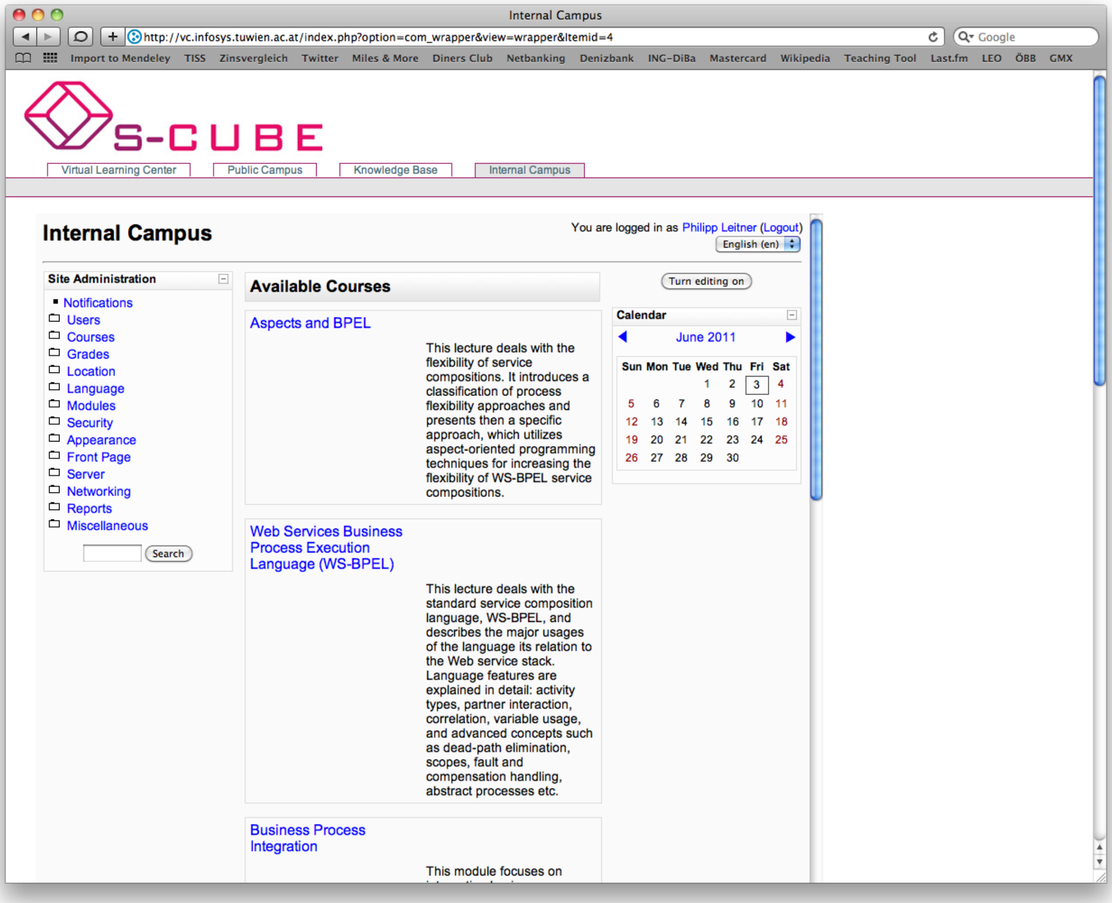
| Component | Version | Functionality |
|---|---|---|
| Moodle | 1.9.7 | Provides the Course Management System (CMS) |
| Apache | 2.2.8 | Provides the web server environment |
| PHP | 5.2.4 | Provides the required HTML-embedded scripting language. |
| MySQL | 5.0.51a | Provides the database backend |

Table 6: Moodle Components

[13] `http://www.ilias.de`

### 2.5.3 Presentation of Results

From its portal the Internal Campus presents itself as an announcement of all available courses in the community 11. To the left is the administration panel of the campus with all required functionality shown in folder style. Thus, the Internal Campus provides all requirements stated and allows for a flawless teacher – student interaction.



Internal Campus

You are logged in as Philipp Leitner (Logout)

Site Administration

- Notifications
- Users
- Courses
- Grades
- Location
- Language
- Modules
- Security
- Appearance
- Front Page
- Server
- Networking
- Reports
- Miscellaneous

Available Courses

Aspects and BPEL

This lecture deals with the flexibility of service compositions. It introduces a classification of process flexibility approaches and presents then a specific approach, which utilizes aspect-oriented programming techniques for increasing the flexibility of WS-BPEL service compositions.

Web Services Business Process Execution Language (WS-BPEL)

This lecture deals with the standard service composition language, WS-BPEL, and describes the major usages of the language its relation to the Web service stack. Language features are explained in detail: activity types, partner interaction, correlation, variable usage, and advanced concepts such as dead-path elimination, scopes, fault and compensation handling, abstract processes etc.

Business Process Integration

This module focuses on

Calendar

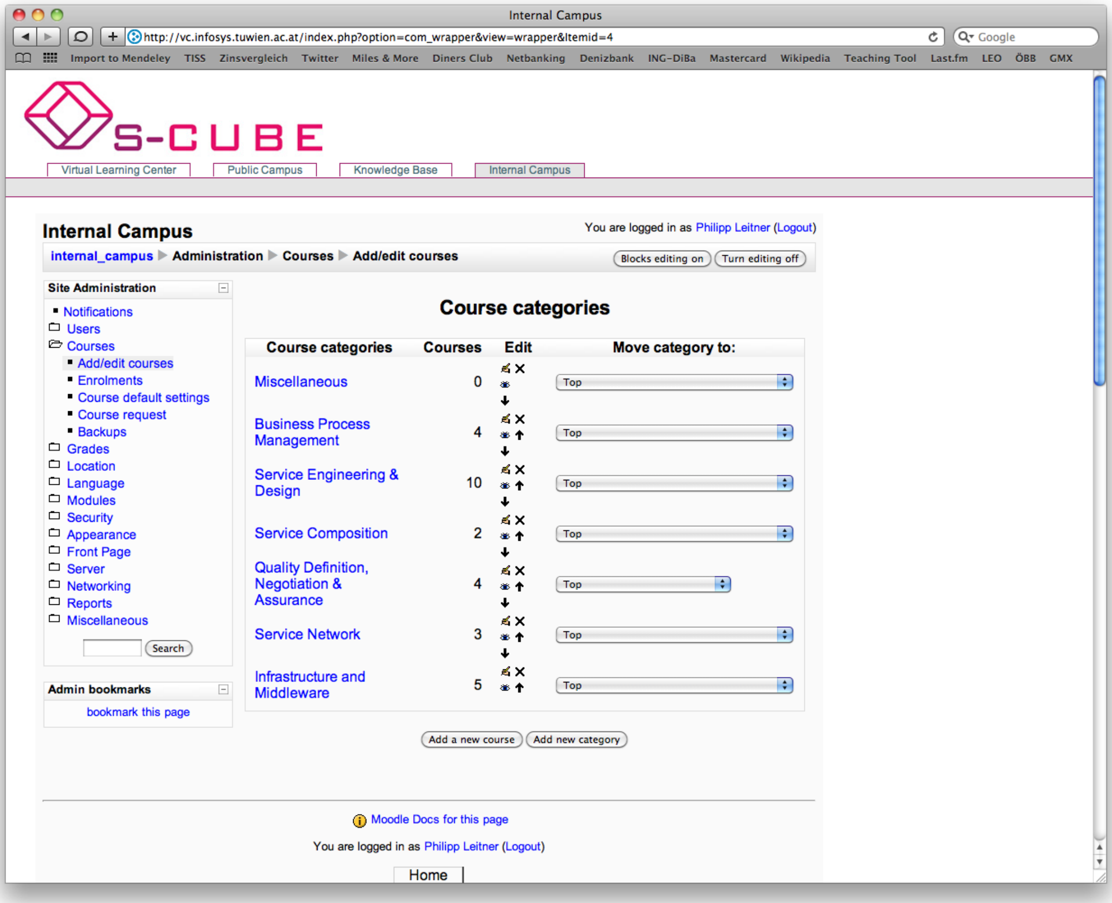
June 2011

| Sun | Mon | Tue | Wed | Thu | Fri | Sat |
|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 26 | 27 | 28 | 29 | 30 | | |

Figure 11: Internal Campus Management

The panel covers the requirement with the following range of functions. A selection is detailed next:

- *Notifications* provides a notification feature. Here upcoming exams and deadlines can be monitored.
- *Courses.* This area provides all functions for the course management including adding and editing courses, enrollments, and also requests from students for new courses (c.f, Figure 12.).
- *Grades* Allows to setup the grading system. This is in particular comfortable in a international course system as the S-Cube community because a mappings between the grade systems can be jointly discussed and defined.
- *Reports.* Any large management system requires reporting to present KPIs and statistics at the end of a period. This is also integrated in the Virtual Campus's course system.



Internal Campus

You are logged in as Philipp Leitner (Logout)

internal_campus ▶ Administration ▶ Courses ▶ Add/edit courses

Course categories

| Course categories | Courses | Edit | Move category to: |
|---|---|---|---|
| Miscellaneous | 0 | ✕ ↓ | Top |
| Business Process Management | 4 | ✕ ↑ ↓ | Top |
| Service Engineering & Design | 10 | ✕ ↑ ↓ | Top |
| Service Composition | 2 | ✕ ↑ ↓ | Top |
| Quality Definition, Negotiation & Assurance | 4 | ✕ ↑ ↓ | Top |
| Service Network | 3 | ✕ ↑ ↓ | Top |
| Infrastructure and Middleware | 5 | ✕ ↑ | Top |

You are logged in as Philipp Leitner (Logout)

Figure 12: Public Campus Course Management

## 2.6 Relationship to Joint Master and PhD Programmes

The finished Virtual Campus is considered a major milestone in the support of both programmes. The described infrastructure is considered a all-round supporting tool for the students and teacher in S-Cube. This is not only true for the joint master students. Their on-line courses provide them with a plethora of different courses of all S-Cube partners. At the more, the joint PhD students and their research activities are, in particular, supported by the Knowledge Base module. This allows them to get insight into their partners work including not only the research work itself, but also the presentation style of the others or the current most relevant topics of interest in SOA research. However, not only the work of other PhDs is at the core of this module of the Virtual Campus. In fact, it is even more important the S-Cube PhDs can represent themselves on such a project platform and present their contributions and, finally, find their place in the community. From this point of view, the whole Virtual Campus is based on the concept of mutual visibility and, thus, in future (even after the project ends) with its supports different as well as common topics and interests can be extracted and lead to new pan-European collaborations. Last but not least, also the teaching personal of S-Cube is able to structure an enhance their current learning materials and offer it to the large community of students in the project. Additionally, with the transparency given also here common learning methods can be found or defined. Overall, the finalized Virtual Campus will be a major contribution to the S-Cube project and its joint programmes.

# 3 Conclusion and Future Work

The focus of Deliverable CD-SoE-1.1.6 was to present the current status of the Virtual Campus platform. The platform is a on-line portal and devoted to students and researches of the S-Cube project. The work presents the Virtual Campus in all its technical details. It demonstrates how the basic framework builds on a modern Content Management System (CMS). Thereafter, all three domains *Public Campus*, *Internal Campus*, and *Knowledge Base* are described and, in particular, decisions made explained. Screenshots of the UI give an impression of the fully developed portal. A final chapter outlines the relationship to the joint S-Cube programmes. However, as the considered use cases evolve and the all-day use of the finished platform increases requests for updates and changes may arise. The leader of the Virtual Campus are open to any change requests by the partners that meets eventually arising requirements. Nevertheless, requirements can only be considered as long as the planned effort for those remains in the boundaries of the remaining package effort.

## 3.1 Outreach

The Y3 review has already risen a major request for extension of the Virtual Campus. One of the recommendations of the Y3 review considers the Virtual Campus as the appropriate tool to present the S-Cube results to a broader audience. At the core the recommendations envision a condense presentation of each S-Cube partner's key results summarized in packages. These packages must be presented by the right tooling in order to allow visitors and interested students to have a quick overview and learn about the research and knowledge build and comprised in the S-Cube project. At the more, the Virtual Campus is considered one of the major project outcomes and a plan for its sustainability must be presented.

During the writing of this document we have realized that the current tooling can not provided the requested extensions towards a condense and preprocessed presentations of the project's results. All current efforts in the package will be spend on planning how to provide a structured cluster of the research fields, just as recommended, and present those in a interactive and attractive manner. In our selection for a tool we consider the following requirements:

- *Structured and Condense.* All partners need to agree on a sound overview structure that introduces the third parties into the S-Cube project. In addition, all need to identify the key results and how they are integrated into the structure.
- *Sustainability.* As a major joint result of S-Cube accessible from the web and directed in particular to future students and researches in *service science* a plan to provide to platform beyond the projects life time must be consolidated with the partners.
- *Attractive and User friendly.* The tool must present itself with the newest available technology for content presentation and allow a user friendly browsing, searching, and gathering of the content.

The initial ideas and input provided try to cover the requirements by starting from the S-Cube Book's [3] structure and identify if the structure still fits all partners and can be put in a representative manner. A tool that can be considered for the presentation of the results is the web-based **Prezi**[14]. The tool allows to jointly create interactive presentations on various subjects in a hierarchical manner. It bases on the common Flash technology and thus is compatible with all browsers capable of presenting Flash data.

[14] `http://prezi.com/`

# References

[1] O. Danylevych, "CD-SoE-1.1.2: Organizational Structure for Virtual Campus," Tech. Rep., 2009.

[2] K. Zettler, "Planung, Konzeption und Implementierung eines Virtual Campus im Rahmen von S-Cube," Master's thesis, Distributed Systems Group, Vienna University of Technology, 2010, (in german).

[3] M. Papazoglou, K. Pohl, M. Parkin, and A. Metzger, *Service Research Challenges and Solutions for the Future Internet: Towards Mechanisms and Methods for Engineering, Managing, and Adapting Service-Based Systems.* Springer, 2010.

[4] V. Andrikpolous and M. Parkin, "CD-SoE-1.1.3: Integrated Knowledge Model," Tech. Rep., 2009.