TOWARDS A COMMON PUBLIC SERVICE INFRASTRUCTURE FOR SWISS UNIVERSITIES

Florian Schnabel, Eva Bucherer
Institute of Media and Communication Management, University of St. Gallen
Blumenbergplatz 9, 9000 St. Gallen, Switzerland

Uwe Heck
Institute for Information and Process Management, University of Applied Sciences St. Gallen
Teufener Strasse 2, 9000 St. Gallen, Switzerland

Keywords: Service-oriented architectures, University, Web services, Event-driven architecture, e-Government, Public services.

Abstract: Due to the Bologna Declaration and the according procedures of performance management and output funding universities are undergoing organisational changes both within and across the universities. The need for an appropriate organisational structure and for efficient and effective processes makes the support through a correspondent IT essential. The IT environment of Swiss universities is currently dominated by a high level of decentralisation and a high degree of proprietary solutions. Economies of scale through joint development or shared services remain untapped. Also the increasingly essential integration of applications to support either university-internal or cross-organizational processes is hindered. In this paper we propose an approach for a comprehensive service-oriented architecture for swiss universities to overcome the current situation and to cope with organizational and technical challenges. We further present an application scenario revealing how Swiss universities will benefit of the proposed architecture.

1 INTRODUCTION

Just like companies in the private sector universities are increasingly faced with a highly competitive market. They rival for reputation and funding on national as well as on international level. In response, “excellence” has become the slogan of various initiatives in the university environment. The requirements to achieve excellence in research and teaching are manifold. But not less important is the factor of “operational excellence”. High administrative efforts result from resource and course planning, data administration, accounting of services, and the administration of sponsorships.

The need for efficient and effective processes makes the support through IT, also for the university sector, essential. As a result university IT departments are facing a lot of challenges. The recent organisational change, caused by the Bologna declaration (van der Wende, 2000), requires European universities in addition to adhere progressively to external (system) requirements, such as the transmission of statistical data to public authorities. In addition Bologna increases the transparency and thus the comparability of universities.

This work focuses on Swiss universities and how they could face the organizational and technical challenges. In the Swiss Campus 12 universities, 8 universities of applied sciences and 14 “universities of teacher education” such as further university institutions, supported by the Bund, exist (CRUS, 2008).

As a result of the federal organization, the IT infrastructure and systems of Swiss universities is dominated by a high level of decentralisation. Each university produces and consumes its individual IT services. Economies of scale through joint development or the use of the same standard software product remain untapped. Consequently the potential of cost reduction through the usage of the same standard applications and shared services is high.

Schnabel F., Bucherer E. and Heck U. (2009).
TOWARDS A COMMON PUBLIC SERVICE INFRASTRUCTURE FOR SWISS UNIVERSITIES.
In *Proceedings of the 11th International Conference on Enterprise Information Systems - Databases and Information Systems Integration*, pages 267-273
DOI: 10.5220/0001861802670273
Copyright © SciTePress

Analogous to the experiences and empirical studies of the private sector there is a high potential for universities to decrease IT costs through the systematic use of synergies. Universities commonly agree about the weaknesses of current island solutions and about the opportunities a common service infrastructure will provide, e.g. cost reduction as aforementioned. However the required organizational and technical changes will not occur spontaneously. Universities still hold on to the federate structure, as it grants them certain autonomy. Certain barriers such as lock-in effects through existing structures are almost impossible to be overcome by an individual institution resp. university.

The way for change has to be paved and incentives for universities have to be pointed out. Joint effort is needed to overcome the described barriers. Efforts have to be based on a common architecture and understanding. Such are the declared goals of a consortium of partners from industry and academia that form the research project Swiss Campus Web Services (SCWS) funded by the CTI (Swiss Confederation’s innovation promotion agency)[1]. First results of this project are presented in the paper at hand.

The main goal of the project is to develop a service-oriented reference architecture for the administrative area of Swiss universities. This Swiss Campus architecture shall enable the exploitation of existing synergies, while taking into account the specifics of the federal structure of the Swiss Campus. Existing structures shall not be replaced but included into the new architecture. Hence universities will benefit from the harmonization and reuse of internal processes that run similarly in different organisations. Furthermore the complexity of multiple point-to-point connections can be reduced by creating a central infrastructure handling the communication between the universities.

The standardisation of university software and the accrual of a market for university software and IT services shall be promoted, by enabling the IT divisions of universities and IT companies to provide their applications as Web Services.

The SCWS project extends the emerging e-Government initiative in Switzerland. This initiative comprises several governmental domains e.g. the domain for resident registration SEDEX (SEDEX, 2008).

On a technical layer the Swiss eGovernment initiative has jointly specified a secure communication infrastructure based on the concept of an Enterprise Service Bus (ESB), the so called *Event Bus Schweiz* (Bund, 2006). Therefore the envisaged development of a Swiss Campus architecture (see section 3) will be in line with the national initiative *Event Bus Schweiz*.

This paper evaluates how the implementation of a service-oriented concept could promote the use of synergies and support the integration of heterogeneous applications across universities. It is organised as follows. Section 1 introduces the current situation for universities and motivates our work. We give an insight into our research setting. Furthermore this section presents the embedding of our work in the emerging eGovernment landscape in Switzerland. Section 2 then discusses the current IT situation for Swiss universities and gives an overview of the envisioned architecture. The architecture is described in more detail in section 3. Finally an application scenario is introduced in section 4 that will reveal the benefit of our architectural approach.

2 HOW THE SWISS CAMPUS ARCHITECTURE WILL COPE WITH THE CURRENT SITUATION

To underline and clarify the present situation in the Swiss Campus we conducted numerous workshops during the SCWS project involving Swiss universities. The goal of these workshops was to gather information and to derive concrete figures about their current IT infrastructure and systems. Analysing the results of these workshops we figured out both technical and organizational challenges of the existing information technology the SCWS project has to cope with. The following main findings have been derived:

There is a lack of organisational support for tasks and processes spanning different organizational units both within and across universities and their related organizations. From a technical point of view the electronic integration of the correspondent applications to support such processes is still rudimentary.

The current IT infrastructure and IT systems of Swiss universities do not provide the flexibility to accommodate to changing organisational conditions. The objective of SCWS is to provide an IT

[1] See acknowledgements

architecture that is flexible and agile enough allowing for an easy adaptation to changing organisational and legal requirements.

While most of the applications in use are still traditional client/server systems, there is a trend towards the establishment of web-based applications. These web-based applications either replace existing applications or cover functionalities that haven’t been implemented yet.

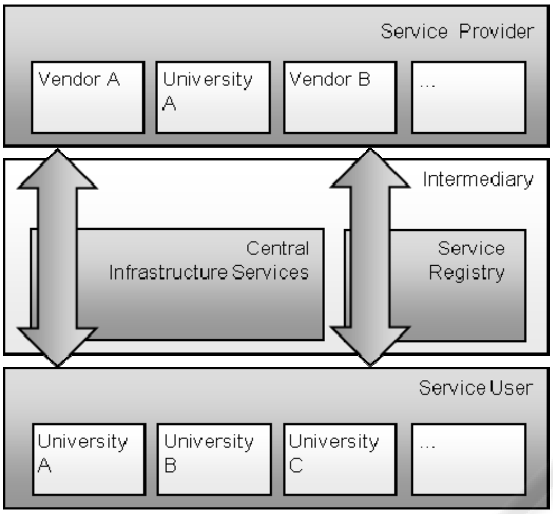

Figure 1: Swiss Campus architecture.

There are only a few commercial software-vendors that dominate the Swiss standard software market for universities. A potential for the Swiss Campus architecture is to define an open infrastructure allowing other software vendors to gain market share, to allow for the modularisation of comprehensive software solutions, and to provide these solutions as a service.

At the moment software is mainly hosted in data centres operated by universities themselves. By leveraging a software market and enabling the easy composition of independent software modules universities will be able to consume these modules as a service. Hence the need to operate an own data centre will become obsolete.

Since SOA and business process management initiatives should be combined (Schulte, 2008b) the Swiss Campus architecture has to consider the modelling of processes and tasks as well as to elaborate a concept for the transformation of these models into IT services.

The SCWS project therefore proposes an architectural approach enabling both the SCWS application partners and external organisations to establish a platform where they can bring in and share services.

Figure 1 gives an overview of the envisaged architecture. The architecture includes three main roles of agents; Service users are universities or university institutions. The functionality of a service provider can be taken over by a software provider but as well by a university offering internally developed services to other universities. Service provider and user communicate via a central platform, the intermediary, that provides a service registry and further central infrastructure services (security, authentication, monitoring, etc.).

The Swiss Campus architecture will have to cope with existing, heterogeneous solutions of the universities’ IT-departments. These legacy systems will be integrated into the service-oriented architecture as web services. Another architectural challenge is the implementation of the central infrastructure functionality. Furthermore a goal of the envisioned architecture is to create an environment where software vendors and IT-service providers are leveraged to create services that decrease the universities’ IT and process costs.

3 AN ARCHITECTURE FOR THE SWISS CAMPUS

3.1 Meeting Organisational Challenges

In order to meet the organisational requirements, discussed in the previous section, we propose an architectural approach for the Campus Switzerland that extends the proposed architecture of the Swiss e-Government initiative. This enhanced architecture has to consider the integration of different stakeholders like the universities’ IT departments, markets for commercial software vendors and IT service providers, and peer universities as well as Swiss governmental organisations and initiatives like SEDEX.

The planned architecture is based on the coupling of multi-agent systems that will seamlessly interact with each other. The multi-agent systems represent domains containing central communication and mediation services and standard objects that are exchanged between the so called agents that represent the stakeholders of one domain. Through the lose coupling of these heterogeneous multi-agent systems we will allow each university, institute, and external partner to keep their proprietary applications and infrastructure. Our approach doesn’t necessarily mean to replace existing systems in order to participate. This is one of the critical success factors of our common infrastructure.

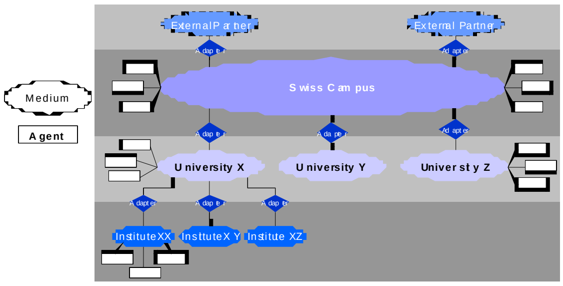

Figure 2: Hierarchy of multi-agent systems in SCWS.

As depicted in Figure 2 the central domain is situated on the second upper horizontal layer and symbolizes the Swiss Campus. The above layer comprises external domains e.g. companies providing insurances for students or Swiss governmental organisations. On the layer below the Swiss Campus layer different universities are placed that interoperate through the Swiss Campus domain. Each university is built up of different institutional domains. The different domains are connected by adapters that allow for the seamless integration into a holistic architecture.

Within the Swiss campus internet services are provided by one central organisation. The SCWS project considers to create one central service hub that will be operated by the organisation currently providing the internet services. You can think about other approaches without a central hub to connect Swiss universities. One example is to organise communication in a peer-to-peer network. However the existence of the internet service organisation and thus the existing expertise in operating a hub and providing services led us to the central hub approach The hub function of the central Swiss Campus domain allows for the loose coupling by providing a central integration function. Furthermore it will guarantee the availability of the message exchange medium and provide security services.

We strongly believe that our envisaged approach is applicable to other countries as well. Through the loose coupling the autonomy of organisational units will be guaranteed. However by simply defining common rules and standards it will be possible to make the coupling tighter.

3.2 Meeting Technical Challenges

The main technical challenge for the Swiss Campus is the seamless integration of existing, heterogeneous applications. We will therefore create a central middleware medium that will allow for the interconnection of these applications. The applications themselves will then be connected either directly, through adapters, or through the coupling of the existing middleware solutions the applications are connected to.

The basic principles for the Swiss Campus architecture are based on best practices from service-oriented architectures (SOA), as described in (Arsanjani, 2004), and event-driven architectures (EDA), as described in (Chandy, 2006). In order to implement the described functionality of SOA and EDA a middleware infrastructure is required that provides special communication and mediation services (Schulte, 2008a, Schulte, 2008b). For the Swiss Campus we will implement an Enterprise Service Bus (ESB) (Schmidt et al., 2005; Maurizio et al., 2008; Papazoglou and van den Heuvel, 2007). The required communication and mediation services (Keen et al., 2004) will be described in more detail later in this paper.

Furthermore we have to define standard objects for the Swiss Campus that will be exchanged through the event bus. These standard objects follow both syntactical and semantic standards. An example of such a standard object is an object containing a student’s master data. As well we will specify a concept for the connection of legacy applications through adapters.

Communication and Mediation Services. One of the challenges that the SCWS project has to cope with is what central communication and mediation services will be needed and where they should be implemented. Further the service design has to be in line with the service specification of the Event Bus Schweiz initiative (Bund, 2006). Hence we will guarantee that services of the ESB will be able to seamlessly access the services of the Swiss Campus.

The services have to be divided into a group of services that have to be implemented on the central Swiss Campus bus and services that can be implemented on distributed buses. The services running on the Swiss Campus bus comprise a directory service, an event catalogue service, a subscription service, and security services. The services running on the distributed buses cover services for transformation, simulation of events, operational services, tracing, error handling, exception handling, validation, and routing.

Table 1 describes these services in more detail.

Table 1: Communication and mediation services for the Swiss Campus.

Service	Functionality for the Swiss Campus
Services running on the Swiss Campus bus	
Directory service	Provide and make accessible all services provided by Swiss universities, commercial software vendors, and external partners.
Event Catalogue service	Provides a list of all events and their description that can be operated by the Swiss Campus bus.
Sub-scription service	Offers the connected participants to subscribe to the notification or reception of certain events.
Security service	Offers standard security functionality (authentication, authorization, event or data encryption)
Services running on the distributed buses	
Trans-formation service	Transforms messages or events from application-specific formats into the common format of the Swiss Campus and vice versa.
Simulation service	Provides simulation functionality e.g. for sending or receiving events.
Operat-ional services	Provides functionalities for the analysis of log files as well as performance and capacity planning.
Tracing, error/ exception handling, validation service	Allows for the tracing of events and messages. Furthermore error and exception handling as well as validation is covered by this service providing basic functionalities for transaction management.
Routing service	Covers the provision of tables for addressing and routing. The Swiss Campus bus only contains entries for directly connected applications and for the connected buses of other domains.

Standard Objects for the Swiss Campus. Existing applications in Swiss universities and applications from commercial software vendors and service providers have to be integrated into the Swiss Campus architecture. Therefore we will define standard objects that will be exchanged through the Swiss Campus bus. These standard objects are composed of different parts containing information about the Swiss Campus bus and specific application scenarios in the Swiss Campus as well as the message payload.

Adapters. In order to ensure connectivity and interoperability of heterogeneous software systems with the Swiss Campus bus adapter components will be built. These adapters transform messages or events from the external application's proprietary format into the commonly defined Swiss Campus format and vice versa. Furthermore adapters allow for the distributed implementation of services enabling the communication via the Swiss Campus bus. In this case adapters provide functionalities for data transformation.

4 APPLICATION SCENARIO

The Swiss Campus architecture targets among others the integration of existing applications in Swiss universities and the establishment of a new market for university-specific applications. In order to support universities and external organisations in connecting to the Swiss Campus architecture a holistic methodology is developed comprising both organisational and technical challenges. On an organisational layer this methodology describes the proceeding in how to define web services from existing applications, how to model tasks and processes, and how to define information objects that are exchanged. From a technical perspective the methodology will describe how to integrate end systems and what functionalities adapters have to implement.

In the following a small section of our methodology will be presented in more detail. To coherently derive IT services out of business logic is a big challenge both within organisations and across organisations. Different approaches exist, e.g. as described in (Levi, 2002) or (Parnas, 1972).

Table 2 roughly describes how to proceed in order to define appropriate services.

Table 2: Proceeding for the integration of web services.

Integration of Web Services		
Step	Required activity	Proceeding in SCWS
1	Requirements definition for Services	Capture business logic (tasks and processes) of Swiss universities
2	Identification of existing and missing Services	Analysis of IT systems of universities
3	Modelling of missing services	Methodology for the transformation of applications into services according to the business model
4	Derive exact service definition out of service model	Adjust service model to technological requirements, define communication and behavioural patterns
5	Describe and publish services	Description according to the Swiss Campus guidelines, publishing in central Swiss Campus repository

Now we will present an application scenario that provides an example of how the new architecture of the Swiss Campus can be used and what benefit we can expect from it. The goal of the application scenario is the modelling and the implementation of a service-oriented publication platform for scientific results. This publication platform is based on an existing publication platform solution in the area of universities that is limited to university internal usage. Furthermore the goal is to implement personalized services and meta-services as value-added services. An example of such a service is the composition of the information system services with services provided in the web. You can easily link the publication authors with their profiles in Xing or LinkedIn. Furthermore it will be easy to provide services analysing the relations between authors or citation data.

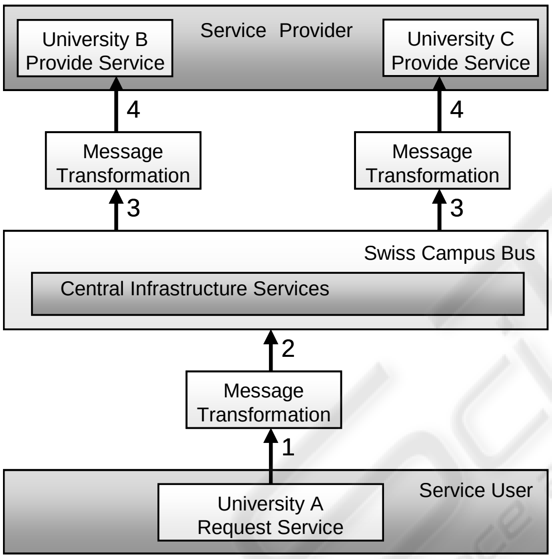

Figure 3: Messages sent via the SCWS architecture.

Figure 3 above depicts an example of how the ESB works in the context of the Swiss Campus. An employee of University A plans to query a specific document provided by another university. The query message is sent from the IT system of University A to the Swiss Campus bus (1). The message in the source format is then transformed into the publicly defined Swiss Campus format and forwarded to the Swiss Campus bus (2). The transformation from the application-specific format into the standard format is done by an adapter. After receiving the message the bus duplicates the query message and transfers it to the two target universities B and C (3). Since these two universities have their own message format the Swiss Campus standard query message has to be transformed again into the required target format and finally forwarded to the university systems (4). As before the transformation and forwarding is done by an adapter.

5 CONCLUSIONS

This paper outlined an envisaged service-oriented architecture for Swiss universities. We proposed an architecture that addresses typical challenges of a cross-organisational setting both on organisational and on technical level. We believe a central hub to be the right architectural design approach since the operation of this hub is guaranteed by the Swiss organisation providing network services for universities. In a peer-to-peer manner there would be more effort needed in order to guarantee availability of and secure communication through the common infrastructure.

Benefits of the SCWS initiative are manifold. The concept of service-oriented architectures (SOA) is evaluated and an architectural model for intra-organisational communication is developed in the area of Swiss universities. Theoretical results are validated by concrete case studies. Furthermore possible business models for SOA providers, software vendors, and universities will be explored.

The SCWS development and application partners gather beneficial experiences in the new SOA environment. As service users they will be able to benefit from resulting economies of scale, leading to lesser costs for implementation, purchase, and operation of software.

The choice of software products and its quality augment. The collaboration between universities will be simplified. In addition universities will be able to act as a service provider themselves.

The use of the Swiss Campus architecture allows universities to be part of the common architecture while still keeping their autonomy. Nonetheless the success of the initiative is dependent on personal and organisational commitment. The needs for change are present, but awareness has to be created in all ranks.

Another challenge concerns data protection. Currently universities prefer to store and maintain data in local databases for security issues. We have to provide a convincing security concept in order to bring universities to accept data storage and maintenance on external systems.

A comparison to architectural approaches for universities in other countries would be helpful. We envision to discuss our approach in scientific communities as well in order to evaluate and further improve the Swiss Campus architecture.

ACKNOWLEDGEMENTS

The research presented in this paper results from the project Swiss Campus Web Services. The project is funded by the CTI under the project number: 9545.1; 2 PFES-ES. The project duration is 05/2008 to 10/2009. The following industry and university partners are involved in the project: Fachhochschule St.Gallen, HTW Chur, Universität St.Gallen, Abraxas AG, Crealogix AG, SWITCH, Ecademy, Fachhochschule Nordwestschweiz, Haute école de gestion de Genève. More information on the project can be found at: http://www.scws.ch

REFERENCES

Arsanjani, A. (2004) Service-oriented modeling and architecture - How to identify, specify, and realize services for your SOA, Series, IBM, http://www-106.ibm.com/developerworks/webservices/library/ws-soa-design1/

Bund, I. (2006) *Event Bus Schweiz, Konzept und Architektur,* ISB.

Bundesverwaltung (2008) Elektronische Projekte, Series, Bundesverwaltung Schweiz http://www.bfs.admin.ch/bfs/portal/de/index/news/veranstaltungen/blank/blank/itstat/03.html

Chandy, K. M. (2006) Event-Driven Applications: Costs, Benefits and Design Approaches, Series, California Institute of Technology http://complexevents.com/wp-content/uploads/2006/07/ManiChandy-Gartner_20060620.ppt

CRUS (2008) Studieren in der Schwei, Series, CRUS, http://www.crus.ch/information-programmes/etudier-en-suisse.html?L=1%3Fref%3DAhuKozmetik.com

Keen, M., Acharya, A., Bishop, S., Hopkins, A., Milinski, S., Nott, C., Robinson, R., Adams, J. and Verschueren, P. (2004) *Patterns: Implementing an SOA Using an Enterprise Service Bus,* IBM.

Levi, K. and Arsanjani, A. (2002) A Goal-Driven Approach to Enterprise Component Identification And Specification, *Communications of the ACM,* 45**,** 45-52.

Maurizio, A., Sager, J., Corbitt, G. and Girolami, L. (2008) Service Oriented Architecture: Challenges for Business and Academia, Hawaii International Conference on System Sciences, Waikoloa.

Papazoglou, M. P. and van den Heuvel, W.-J. (2007) Service oriented architectures: approaches, technologies and research issues, *The VLDB Journal,* 16**,** 389-415.

Parnas, D. L. (1972) On the criteria to be used in decomposing systems into modules, *Communications of the ACM,* 15**,** 1053-1058 http://www.cs.umd.edu/class/spring2003/cmsc838p/Design/criteria.pdf.

Schmidt, M.-T., Hutchison, B., Lambros, P. and Phippen, R. (2005) The enterprise service bus: making service-oriented architecture real *IBM Syst. J.* , 44 781-797.

Schulte, R. W. (2008a) *SOA Applications Should Mix Client/Server, EDA, And Conversational Patterns,* Gartner.

Schulte, R. W. (2008b) *Using SOA to achieve four key design goals,* Gartner.

SEDEX (2008) Sedex - Secure Data Exchange, Series, Bundesamt für Statistik, http://www.bfs.admin.ch/bfs/portal/de/index/news/00/00/sedex.html

van der Wende, M. C. (2000) The Bologna Declaration: Enhancing the Transparency and Competitiveness of European Higher Education, *Higher Education in Europe,* 25**,** 305 - 310 http://www.informaworld.com/10.1080/713669277.

SciTePress Science and Technology Publications