TOWARDS THE NEXT GENERATION OF SERVICE-ORIENTED FLEXIBLE COLLABORATIVE SYSTEMS

A Basic Framework Applied to Medical Research

Jonas Schulte, Thorsten Hampel
Heinz Nixdorf Institute, University of Paderborn, Paderborn, Germany

Konrad Stark, Johann Eder, Erich Schikuta
Department of Knowledge and Business Engineering, University of Vienna, Vienna, Austria

Keywords: SOA, Framework, CSCW, Wasabi, Medical Research Environment.

Abstract: Collaborative systems have to support specific functionalities in order to be useful for special fields of application and to fulfil those requirements. In this paper we introduce the Wasabi framework for collaborative systems, which is characterised by flexibility and adaptability. The framework implements a service oriented architecture and integrates different persistence layers. The requirement analysis for the Wasabi CSCW system is presented in the context of a collaborative environment for medical research which has strict requirements concerning data integrity. This paper shows the results of the requirement analysis and how these are implemented in the Wasabi architecture.

1 INTRODUCTION

In arbitrary fields of application all types of cooperation follow a similar pattern: A group of individuals share data and knowledge within a certain context, add supplementary data, and deduce new knowledge by applying functions. Their results are stored for future collaboration or for other interest groups. Contemporary *Computer Supported Cooperative Work (CSCW)* systems either support only a subset of functionalities useful for the users' daily work or have restricted fields of application. Hence, most systems are inflexible and adapting a system for a new scope of application is a challenging task. Developing an infrastructure for CSCW systems characterised by flexibility and adaptability is desirable, since each scope of application has specific requirements resulting from their ways and methods of collaboration. Specific requirements identified for a field of application can be defined for functionalities of the CSCW system which support users in terms of executing their daily workflows. Furthermore the CSCW system should be able to handle arbitrary data types and data repositories. In particular, handling various persistence layers enables to bind remote repositories of different types or with different characteristics (security, repository management, etc.). The Wasabi infrastructure presented in this paper aims to build a next generation framework for CSCW systems by focussing on flexible service integration – in this context of a *Service Oriented Architecture (SOA)* – and object oriented data integration (integration of persistency layers).

The benefits for such a flexible infrastructure can be obviously seen in connection with a collaborative system for medical research which is used in this paper as an example field of application for our framework. The collaborative environment for medical research is needed in the context of the biobank initiative GATiB (Genome Austria Tissue Bank) which is part of the Austrian Genome Program (GEN-AU, 2007). GATiB aims at the establishment of a tissue bank (BIOBANK, 2007) which builds on a collection of diseased and corresponding normal tissues representing all diseases at the natural frequency of occurrence from a non-selected central European population of more than 700.000 patients. Major emphasis is placed on annotation of archival tissue with comprehensive clinical data including follow-up data (medication, therapy, resection). A more detailed description of the biobank initiative is given in (Asslaber et al., 2007).

In the context of collaborative medical research various conceptual aspects have to be considered: Which individuals are involved in collaboration activ-

Schulte J., Hampel T., Stark K., Eder J. and Schikuta E. (2008).
TOWARDS THE NEXT GENERATION OF SERVICE-ORIENTED FLEXIBLE COLLABORATIVE SYSTEMS - A Basic Framework Applied to Medical Research.
In *Proceedings of the Tenth International Conference on Enterprise Information Systems - DISI*, pages 232-239
DOI: 10.5220/0001696302320239

ities, which sources are accessed for which purpose, and what are the prevailing results? Collaboration in the medical research field is characterised by a high complexity and high variation of collaborative situations. Data is distributed over several institutes and underlays various restrictions in it's accessibility for different persons (roles). Access to patient data is for example highly limited to the context or role of application (e.g. patient context data or disease context data). Data is collected, restructured, analysed, and shared with other persons in different settings.

Even if collaboration activities in arbitrary fields of application are very similar concerning the user's activities, the medical research scenario has very strict requirements because of the sensitive data. (Stark et al., 2008) presents several scenarios and a practical example for the usage of the cooperative system in GATiB. For other communities/applications less strict differentiations exist. However, in its characteristic as a collaborative knowledge management system, our Wasabi-CSCW infrastructure applied to GATiB can be easily used as an e-learning platform or for other fields. Finally the flexible configuration of access rights and the architecture's ability to define contextual views on the presented data allows to use it as a powerful collaborative learning platform.

The paper is structured as follows. To describe the background of the requirements for a CSCW system, which is usable for medical research, Section 2 presents example collaboration scenarios and summarises the main requirements for such a system. An overview of the Wasabi-CSCW system applied to GATiB is described in Section 3. Details according to the fulfilment of the requirements are described in Section 4. Our paper ends with an overview of related work in Section 5, followed by some conclusions.

2 EXAMPLE USAGE AND REQUIREMENTS

Collaboration in the GATiB project focuses on both the medical research as well as routine activities of medical scientists and supporting staff. Data locally distributed over institutes and research groups is accessed in manifold ways and for various purposes. Further it is collected, restructured, augmented, analysed, and shared within a certain group of interested researchers. In the following an example scenario is given.

Due to a cooperation between a hospital and a pharmaceutical company a group of suitable human-tissue donors is to be identified to support a medication discovery study. Therefore, pathological diagnoses, survival data, and tissue images of patients that have signed a formal consent are required. After searching and structuring information, confidential patient data has to be protected since an external organisation is involved in the study. Hence, identifying attributes (name, day of birth) are eliminated and quasi-identifying attributes are k-anonymised (P. Samarati, 1998; Stark et al., 2006). Life style data is included by filling out questionnaires. Further, a tissue microarray of the relevant cases is made in order to test candidate tumour markers. Further, the results should be made available to other research groups.

2.0.1 Requirements

The scenario demonstrates the diversity of collaboration types that may occur. A cooperative system in a biomedical research environment requires a high degree of flexibility and extensibility. Distributed data is accessed in different levels of granularity considering data privacy issues, it is annotated and analysed. We use the above-mentioned scenarios to deduce general requirements a CSCW system has to comply with.

- $R(1)$ **Flexible Integration and Composition of Services:** A multitude of data processing and data analysis tools exists in the biomedical context. Some tools act as complementary parts in a chain of processing steps. For example, to detect genes correlated with a disease, gene expression profiles are created by measuring and quantifying gene activities. The resulting gene expression ratios are normalised and candidate genes are preselected. Finally, significance analysis is applied to identify relevant genes (Tusher et al., 2001). Each functionality may be provided by a separate tool (GENESPRING, 2005; Sturn A, 2002). In some cases tools provide equal functionality and may be chosen as alternatives. Through flexible integration of tools as services with standardised input and output interfaces a dynamic composition of tools may be accomplished. From the system's perspective services are technology neutral, loosely coupled, and support location transparency (Papazoglou, 2003). Therewith the execution of services is not limited to proprietary operation systems and any service caller does not know the internal structure of a service. Further, services may be physically distributed over departments and institutes, e.g. image scanning and processing are executed in a specific laboratory where the gene expression slides reside.
- $R(2)$ **Knowledge Creation and Knowledge Processing:** Cooperative medical activities frequently comprise the creation of new knowledge.

Data sources are linked with each other, similarities and differences are detected, and involved factors are identified. Consider for example a set of genes which is assumed to be strongly correlated with the genesis of a specific cancer subtype. If the hypothesis is verified, this information may be reused in subsequent research. Thus, means to formalise knowledge, share it in arbitrary contexts, and deduce new knowledge are required.

- $R(3)$ **User and Role Management:** The CSCW has to be able to cope with the organisational structure available in the institutes and research groups of the hospital. Data protection directives have to fit the system's access right model. Even though, the model has to be flexible to allow the creation of new research teams and information sharing across organisational borders.
- $R(4)$ **Transparency of Physical Storage:** Although data may be stored in distributed locations, data retrieval and data storage depend solely on access rights, irrespective of the physical location. Hence, the complexity of data structures is hidden from the end user's perspective. Appropriate search, addition, and transformation mechanisms have to be offered by the CSCW system.

3 ARCHITECTURE OVERVIEW

The Wasabi framework enabling collaborative work as described in the previous sections is a service-oriented architecture (Schulte et al., 2007). It bases on the JBoss Application Server (AS) in order to fulfil requirements for enterprise solutions and to provide the scalability and performance needed for a CSCW system which is used in a distributed manner like in the GATiB project. In addition, to fulfill the requirements previously presented, service orientation is an important characteristic in the context of flexibility, adaptability, and maintenance. Note that the service orientation of Wasabi can be attained due to the underlying JBoss AS also being service oriented.

Service orientation is generally an essential characteristic for a CSCW system since the data stored might be of arbitrary formats and located in arbitrary repositories. CSCW systems offer a wide field of collaboration activities, supporting its flexible way/adaptability of storing and handling of data, no longer limiting its activities to specific environments or applications. This Wasabi's functionality enables on the one hand to store data in databases or in file systems reachable via the network. On the other hand it can link already existing databases and data sources to provide users access to those data under consideration of their rights. This is particularly beneficial for the GATiB project since the data collected in different hospitals and institutes is thereby available for the whole community without transferring the data for collaboration to a specific repository.

In the following Section 3.1 presents the underlying idea of organising information sharing and co-operative work for the collaborative system. Afterwards Section 3.2 describes the key components of the server architecture.

3.1 Knowledge Spaces

Knowledge spaces are our representation and structure of presenting the different context types of collaboration situations to the users/actors in the collaboration process. The organisation of individuals collaborating in groups has to be visualised in a useful way reflecting the real world. Users need a knowledge space as a virtual environment for their collaboration activities in which they can meet and work. We define a knowledge space as an *actual use case* in a *collaboration context*. For example a knowledge space might be created for an actual clinical study or a collaborative patient analysis.

Although knowledge spaces are separate virtual concepts, data and knowledge exchange between knowledge spaces is encouraged. An important capability is to upload data or link to remote data which is already available. To suffice the different users' needs for their collaboration, they should organise and structure the knowledge space individually. Consequently, users are responsible for the organisation of their knowledge space which implies highest flexibility. However, reorganisation of one specific knowledge space might be restricted to a limited group of users in order to avoid unauthorised modifications. Such self-organisation conforms to the self-organisation forms of knowledge in the Web 2.0 (tagging). Further concepts of the Web 2.0 like annotating available information is important for a useful CSCW system.

3.2 Server Architecture

The server consists of four main components. First, the core of the Wasabi enterprise server architecture implements the framework for a CSCW system and aggregates all services of Wasabi (see Figure 1). Knowledge spaces are realised by rooms which are containers for users and documents (any uploaded content). Furthermore rooms can be connected via exits. To realise the persistence of all objects of the server core in a simple way, they all inherit from the

`WasabiObj` class. To implement a SOA with this underlying core classes, the access to objects of the Wasabi core is realised by the second component of the Wasabi server - the EJB services (see Figure 2). For example the `UserManager` is the responsible service for handling, modifying, and extracting information of `user` objects. Their responsibility classify the EJB services as basis services since they implement basic functionalities on the objects of the Wasabi core. Since their tasks focus on the modification and provisioning of data stored in the Wasabi core objects, these can be further classified as data-centric basic services. EJB services are also used to realise flexible user authentication mechanisms as well as to be adaptable to various content backends/repositories.

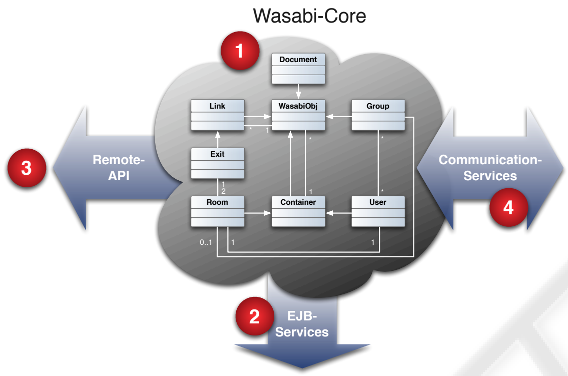

Figure 1: Components of Wasabi Core.

All EJB services handling objects of the Wasabi Core are realised as service beans, i.e. they implement the EJB Management Interface. Service beans have the constraints that only one instance is created on the server. By using service beans no transaction control has to be performed since all users or remote/intern services are interacting with the same instance of the EJB service. For the internal usage of the EJB services local interfaces are implemented which might offer more or different functionalities than accessible from remote services or clients.

Third, the remote API provides an interface for server-client communication (see Figure 3). Therewith a common interface can be used to send requests to different adapted services, if the services expect the same input data. This simplifies the enhancement of Wasabi Beans by adapting new services with little effort. Note that the remote API might differ from the local API of the EJB services. The fourth component is responsible for the message exchange with adapted services. The server core is enhanced by services realising communication features which are essential in a cooperative system (see Figure 4). The specific communication services generate the outgoing messages according to the defined interfaces of the remote webservices and processes and extracts information from incoming messages.

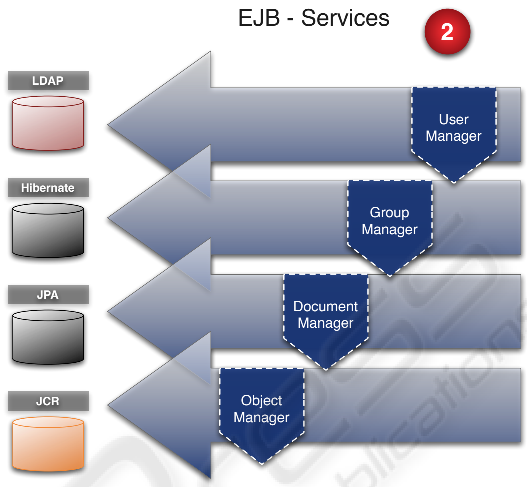

Figure 2: EJB Services Realising Service Orientation.

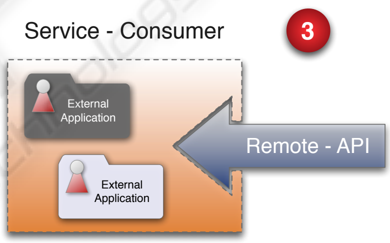

Figure 3: API for User Interaction.

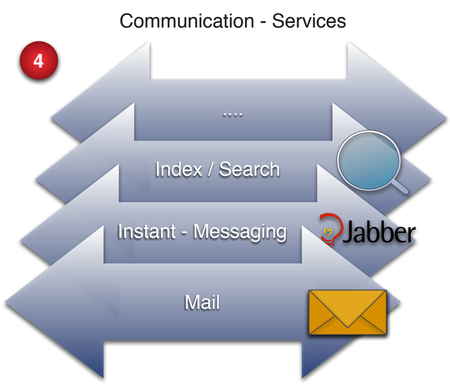

Figure 4: Core-Enhancement for Communication Support.

4 ARCHITECTURE DETAILS

The example application had shown that data used for collaboration has to be differentiated. Furthermore, the environment for a collaboration has to be specified and the users have to be represented in the CSCW system in a suitable way. This section details how the requirements of the Section 2.0.1 are fulfilled in the architecture.

4.1 $R(1)$: Flexible Integration and Composition of Services

The requested flexibility is achieved by the service orientation of the Wasabi-CSCW infrastructure. To access data from internal objects, appropriate remote and local services exist. Intermediate services are used to invoke remote webservices and to handle information from those. Due to the service orientation, providing the user a composition of services requires little effort for internal services as well as a mixture of internal and remote services (or service compositions).

Arbitrary services can be linked to the CSCW system. Intermediate services can use either the local or remote API of the server's EJB services. These are responsible for generating appropriate messages for remote services and to handle incoming messages. The handling of incoming messages consists of calling an adequate service and method of the Wasabi server and to transform the data in the requested format. Hence, a flexible integration or composition of services is possible.

4.2 $R(2)$: Knowledge Creation and Knowledge Processing

A knowledge object is the result of a non-empty sequence of functions applied to at least one data object. Generally, each knowledge object enhances information currently available in data objects. We distinguish various types of knowledge objects: graphical objects (plots, diagrams, etc.), tabular objects presenting summarised information (e.g. the results aggregation operations), and dependency structures (e.g. ontologies). Semantic structures may be used to formalise knowledge. Relationships between objects or groups of objects are stored in a processable way in order to build a knowledge repository. This repository integrates conclusions from various knowledge spaces allowing to explore consolidated knowledge and deduce new knowledge. Making knowledge persistent and processable allows us to fulfil requirement $R(2)$. The post-processing of accessible data objects is a specific but also important criterion for a CSCW system supporting medical research. Since no limitations concerning the data stored and accessed should exist, the methods of post-processing should also not be limited. The CSCW system has to support all required post-processing steps and must be flexible to integrate new functionalities with little effort.

The post-processing of data can be performed in various ways and with different intentions and analysis targets. The methods of post-processing significantly depend on the users as well as the underlying data objects. Hence, support a flexible framework for integrating new post-processing methods and new knowledge objects, essential for the collaborative system. The service oriented architecture supports an easy linking of services which is more flexible than integrating specific functionalities into the collaborative system itself. The linking is performed by dedicated services. The Wasabi core has data objects and knowledge objects to fulfil the distinction as requested.

For GATiB the CSCW system should provide different views on the same data, for example for the staff working directly with a patient have full access to all her details, whereas researchers only see statistics about multiple patients. Showing statistics implies that a post-processing – evaluating the accessible data and generating the statistics – is performed automatically by the collaborative system. (Stark et al., 2008) presents several scenarios for the usage of the cooperative system in GATiB. The Wasabi framework interacts therefore with a specific service which generates the statistics according to the content requested by the user. In order to ensure the data privacy, the data is read out by the Wasabi's service which transfers it as a SOAP message to the service responsible for post-processing. The output is then stored by the Wasabi server as a knowledge object itself before presenting it to the user. The user can then decide to store the knowledge object which enables her to add annotations. Multiple users can generate different statistics on the same data which results in the generation of multiple knowledge objects.

4.3 $R(3)$: User and Role Management

An individual is a person participating in collaborative acts. Individuals may be for instance researchers from the medical or biological domain, medical students, or project managers. Individuals are categorised into internal and external persons in order to differentiate between access to sensitive patient-related and research-related data as well as access to anonymised and summarised data.

The CSCW system ensures that individuals are distinguished in order to reflect the activities of one individual in the real world as a one-to-one mapping in the virtual world. This is important in the context of collaboration (discussions, annotations, etc.) as well as for the data access (authentication and authorisation). The organisational mapping accomplishes the user and role management specified in $R(3)$.

The Wasabi-CSCW infrastructure creates a user object for each CSCW system user, who are distinguished internally when checking access rights and service invocation.

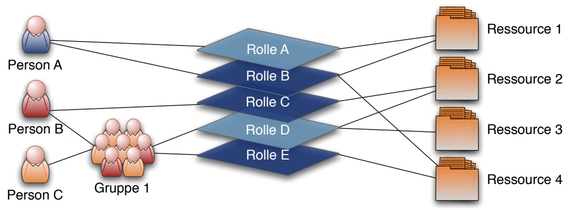

Figure 5: Role Assignment.

Individuals having the same access rights and using the same set of functions may be integrated to groups and certain roles may be assigned to users/groups. Roles are used to adjust the access of individuals and groups to resources of the CSCW system. As shown in (see Figure 5), a role can be used to bundle access parameters. The Wasabi-CSCW infrastructure supports a fine-grained definition of roles and groups for users. Users can be assigned to an arbitrary number of groups which can also be related to other groups. Hence, individuals can be part of several groups and a group can consist of subgroups. For example all users working in the same hospital are in one group, denoted *hospital X*. The "regular" medical staff is allowed to read and write all data which is presented patient-centered. The medical students associated to the hospital are not allowed to see all sensitive data. Accordingly, the medical staff is a subgroup of hospital X. The EJB services `GroupManager` and `UserManager` (see Figure 2) are responsible for the user management. They ensure that logins are unique within a domain or within the whole system. Furthermore, rights are defined when generating a user and additional access rights can be set by assigning users to groups.

Virtual knowledge spaces (rooms) can have links to other rooms and links to data stored in any linked repository. A room supports several perspectives which enables users in the same room to see the data in a different way and with a different level of details shown. The perspectives for a user can be defined by roles. In particular, the various level of details is a strong requirement in the context of the GATiB project since the data available is often sensitive and data privacy has to be supported by the system. In addition to the perspectives which might vary from user to user, the Wasabi framework also provides a detailed authorisation model. For each user the access rights can be defined, and before granting the user write/read access, these are checked by a specific service of the framework. Since data stored not directly in the Wasabi server has to be prevented from unauthorised access, security issues have to be considered not only in the context of accessing data through the collaborative system's user interface. Data stored in a databases is protected from unauthorised access by requesting user and password during the connection establishment with the database. Data stored in a filesystem requires also a similar access validation. This requirement was considered in the Wasabi framework by using an LDAP server. The `UserManager` is responsible for the user management within the CSCW system and consequently it interacts with the LDAP server. This solution has the advantage that access rights are checked within the collaborative system and in addition also in the repository. Furthermore, interacting with an LDAP server has the advantage that often already established repositories work with LDAP servers and consequently no additional effort is necessary for transferring the right specification into the Wasabi framework. Note that because of the service orientation of Wasabi, other authorisation services providing similar functionalities like LDAP servers can be integrated.

4.4 $R(4)$: Transparency of Physical Storage

A flexible and adaptable CSCW system for various fields of application must not have limitations concerning the data stored and accessed. Since arbitrary data types have to be supported, applications used to visualise or modify the data should be made available within the CSCW system. Content made available in the CSCW system is handled as a document object which has no limitations concerning data types, repository, or size. Access rights are checked internally before sending requests to the repository. We propose flexible levels of data granularity to satisfy requirement $R(4)$.

The data used for analyses in the GATiB project is primarily stored in databases. The relational concept of data objects has to be transferred into an object oriented concept presented in virtual knowledge spaces. Each tuple of information stored in one row of a database table has to be mapped into an object. Most developments using the Java Persistence API start with the definition of classes from which then the databases are automatically generated. Since the data used in the GATiB project is already stored in different databases basing on various relational models, the object relational mapping has to be "manually" performed in the Wasabi framework by an appropriate service. Therefore each data object inherits from a `WasabiObj` class defined in Wasabi (see Figure 1). Internally core objects like individuals, groups, and rooms are persistently stored in the central database. Since the related classes are subclasses of `WasabiObj`, it is necessary to assign to each object a unique identifier (UUID). Storing all `WasabiObj` in a central database, support to generate a unique identified for a `WasabiObj`, i.e. no room or individual can have the same UUID.

Metadata of document objects is also stored in the central database by using the persistence capabilities of entities. In the central database a reference to the repository is held in the associated file, data or knowledge object is stored. Through the service orientation of the Wasabi framework, for each database having a different relational model a specific service can be integrated. Before accessing data of a repository, the Wasabi's `DocumentManager` selects the correct service according to configuration options for the repository to which the related entry in the central database refer. Afterwards the `DocumentManager` engaged this service to perform the object relational mapping. The usage of a repository-specific service enables information stored in databases to be handled with different underlying models in a flexible way. Note that repositories with the same relational model can use the same Wasabi service since the connection parameters are read out dynamically.

The support of different databases can be realised through using different services. For cooperative systems in general a mime-type specific repository selection is realised in the first Wasabi prototype having several advantages. First of all, specific content types might ask for specific repository features. For example a video can be stored on a video server providing features a "normal" repository does not provide, like video-streaming. Furthermore, small objects like annotations can be stored within a database. However storing text-documents in a database is not very efficient since databases transform each object to a specific data type; accordingly these will be stored as character- or binary large objects (CLOB/LOB). The type transformation implies that for writing content into the database a serialisation is necessary; for reading content, the objects have to be de-serialised. This effort is time-intensive for large files and consequently in this case the usage of file systems as repositories should be preferred. The configuration which mime-type should be stored in which repository (any file system or database) is specified within an XML file.

When using file systems two different approaches for the data organisation can be followed. In one solution, the hierarchy of the knowledge spaces can be reflected in the data organisation in the filesystem, e.g. files are named as within the CSCW system and the folder organisation conforms to the room name and hierarchy. Reflecting the hierarchy in the filesystem organisation has the advantage that the filesystem could be accessed via WebDAV or FTP from individuals not registered within the CSCW system who are interested in the content. Of course, such an external data access has to be protected from unauthorised individuals in order to ensure data privacy. If the data stored within the filesystem should be not available apart from the CSCW system, the files can be stored in an arbitrary hierarchy and the UUIDs assigned to the `WasabiObj` can be used as filenames. This alternative solution has the advantage that an additional security mechanism is integrated. Even if an unauthorised individual can see the data structure, they can only guess what is in the files, as the file type is not reflected within the name.

5 RELATED WORK

There exist a strong demand for architectures allowing the developers to encapsulate information tools as services so that clients can access without knowledge of or control over their internal workings. These requirements are implemented by service-oriented architectures (SOAs) since these are suitable for complex environments with changing demands (Foster, 2005).

Today's CSCW systems, for instance Moodle (Rice, 2006) or Open-sTeam(Bopp et al., 2006) are mainly focused on using a single server, for this reason all collaborative services are provided by one system. Examples for CSCW systems that involve flexible service oriented architectures to interact with other existing applications are rare.

A number of international projects have focussed on the development of collaborative systems, not only to support collaborative work, but also to support the needs of medical research. As an example, based on web services the myGrid project (Stevens et al., 2003) in the UK uses bioinformatics as their major focus. Furthermore, the Japanese BioGrid (Nakamura et al., 2004) project provides a Grid infrastructure to fulfil the needs of computational power in life science by making available sufficient power to complete their complex analyses. In particular, these Grid developments differ from our Wasabi infrastructure in the intention of usage. Wasabi is a framework for a CSCW system, whereas myGrid and BioGrid are middlewares for running computational jobs (medical analyses) and for enabling remote data access in a Grid infrastructure. Since Wasabi is a SOA and medical analyses of data stored in the CSCW system are performed by using remote web services, even Grid services can be invoked from a CSCW system building on Wasabi. Consequently, services providing the computational power requested for medical analyses can be linked. Hence, such extensions can be implemented in Wasabi with little effort because of its service orientation.

6 CONCLUSIONS

Since collaboration activities are very similar in various fields of application, a CSCW system should be flexible and adaptable for the usage in different scopes. This paper determines the requirements for such a CSCW system exemplarily in the GATiB project which has many restrictions because of the usage of highly sensitive data. To fulfil its requirements (see Section 2.0.1), the presented Wasabi CSCW infrastructure is a service oriented architecture and is able to handle arbitrary data types and repositories. The flexible object model of the Wasabi core supports an adjustment for specific fields of application and is therewith not limited to the usage in GATiB.

Requirements in the context of GATiB are fulfilled through the SOA, persistence layers, as well as the flexible Wasabi core: The user and role management is realised through the Wasabi core as well as the support of cooperative functions, knowledge presentation, and knowledge processing. The service orientation enables the transparency of physical storage as well as flexible integration and composition of services.

The requirements identified for GATiB can be found in arbitrary fields of application. However, in most scopes the requirements are less strict. Hence, the Wasabi CSCW infrastructure applied to GATiB can be adapted for other fields of collaborative work.

REFERENCES

Asslaber, M., Abuja, P., Stark, K., Eder, J., Gottweis, H., Trauner, M., Samonigg, H., Mischinger, H., Schippinger, W., Berghold, A., Denk, H., and Zatloukal, K. (2007). The genome austria tissue bank (gatib). In *Pathobiology*, volume 74, pages 251–258.

BIOBANK (2007). A biobank for the advancement of medicine. http://www.bioresource-med.com.

Bopp, T., Hinn, R., and Hampel, T. (2006). A service-oriented infrastructure for collaborative learning in virtual knowledge spaces. In *Education for the 21st Century*, pages 35–44.

Foster, I. (2005). Service-oriented science. *Science*, 308(5723):814–817.

GEN-AU (2007). Genome research in austria. http://www.gen-au.at/english/content.jsp.

GENESPRING (2005). Genespring, cutting-edge tools for expression analysis. http://www.silicongenetics.com.

Nakamura, H., Date, S., Matsuda, H., and Shimojo, S. (2004). A challenge towards next-generation research infrastructure for advanced life science. *New Gen. Comput.*, 22(2):157–166.

P. Samarati, L. S. (1998). Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. In *Proceedings of the IEEE Symposium on Research in Security and Privacy*.

Papazoglou, M. P. (2003). Service-oriented computing: Concepts, characteristics and directions. In *Web Information Systems Engineering (WISE 2003)*, volume 00, page 3, Los Alamitos, CA, USA. IEEE Computer Society.

Rice, W. (2006). *Moodle E-Learning Course Development*. Packt Publishing.

Schulte, J., Hampel, T., Bopp, T., and Hinn, R. (2007). Wasabi framework – an open service infrastructure for collaborative work. *skg*, 0:242–247.

Stark, K., Eder, J., and Zatloukal, K. (2006). Priority-based k-anonymity accomplished by weighted generalisation structures. In *DaWaK 2006: Proc. of the 8th International Conference on Data Warehousing and Knowledge Discovery*, Lecture Notes in Computer Science, Volume 4081. Springer Verlag.

Stark, K., Schulte, J., Hampel, T., Schikuta, E., Zatloukal, K., and Eder, J. (2008). Gatib-cscw, medical research supported by a service-oriented collaborative system. In *The 20th International Conference on Advanced Information Systems Engineering (CAiSE'08)*.

Stevens, R. D., Robinson, A. J., and Goble, C. A. (2003). mygrid: personalised bioinformatics on the information grid. *Bioinformatics*, 19 Suppl 1.

Sturn A, Quackenbush J, T. Z. (2002). Genesis: Cluster analysis of microarray data. In *BIOINFORMATICS -OXFORD-*, volume 18(1), pages 207–208.

Tusher, V., Tibshirani, R., and Chu, G. (2001). Significance analysis of microarrays applied to the ionizing radiation response. In *Proceedings of the National Academy of Science*, volume 98, pages 5116–5121.