The Web Service-Based Combination of Data and Logic Integration in Federated ERP Systems

Nico Brehm, Carl von Ossietzky University Oldenburg, Germany; E-mail: nico.brehm@uni-oldenburg.de
Jorge Marx Gómez, Carl von Ossietzky University Oldenburg, Germany; E-mail: jorge.marx.gomez@uni-oldenburg.de

ABSTRACT
ERP systems become more complex and financial expenditures that are connected to the application of such systems increase. ERP systems consist of many software components which provide specific functionality. However, these ERP systems are designed as an all-in-one solution, often implementing functionality not needed. The paper presents the reference architecture of a federated ERP system which allows the distribution of ERP system components on the basis of Web Services. This architecture draws upon a hierarchical standardization model of data and service types. The model advances the reusability of data types and reduces the necessity of data transformation functions in business process descriptions.

Keywords: ERP, FERP, Web Services, Web Service Standardization, WSDL, SOA

1. INTRODUCTION AND MOTIVATION

An *ERP system* is a standard software system which provides functionality to integrate and automate the business practices associated with the operations or production aspects of a company. The integration is based on a common data model for all system components and extents to more than one enterprise sectors [1, 2, 3, 5].

Modern ERP systems consist of many software components which are related to each other. Currently these components are administered on a central application server. In connection to the ERP system complexity several problems appear:

- Not all installed components are needed.
- High-end computer hardware is required.
- Customizing is expensive.

Due to the expensive proceedings of installation and maintenance only large enterprises can afford such complex ERP systems. One solution to face these problems is to develop a distributed ERP system where the system components are reachable over a network (e.g. internet). This component ensemble (federated system) still appears as single ERP system to the user, however it consists of different independent elements which exist on different computers. Based on this construction it is possible for an enterprise to access on-demand functionality (components) as services[1] of other network members over a P2P network. This approach solves the mentioned problems as follows:

- Due to the separation of local and remote functions, no local resources are wasted for unnecessary components.
- Single components are executable on small computers.
- Due to decreasing complexity of the local system also installation and maintenance costs subside.

A *federated ERP system* (FERP system) is an ERP system which consists of system components that are distributed within a computer network. The overall functionality is provided by an ensemble of allied network nodes that all together appear as a single ERP system to the user. Different ERP system components can be developed by different vendors [1, 4].

Figure 1. Architecture of a conventional ERP system

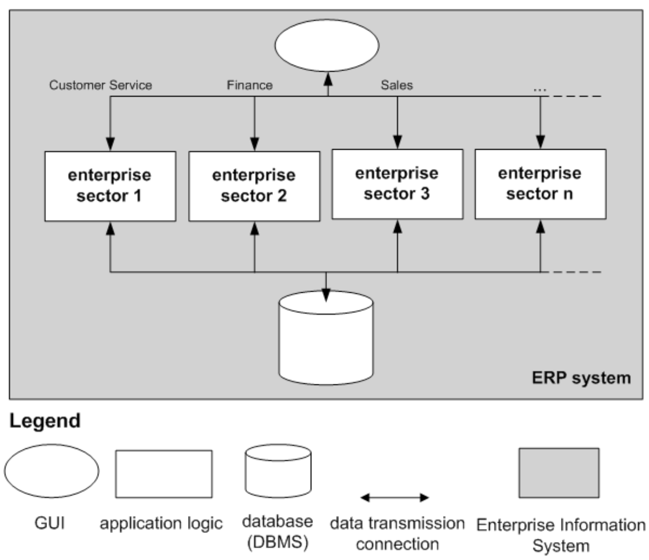
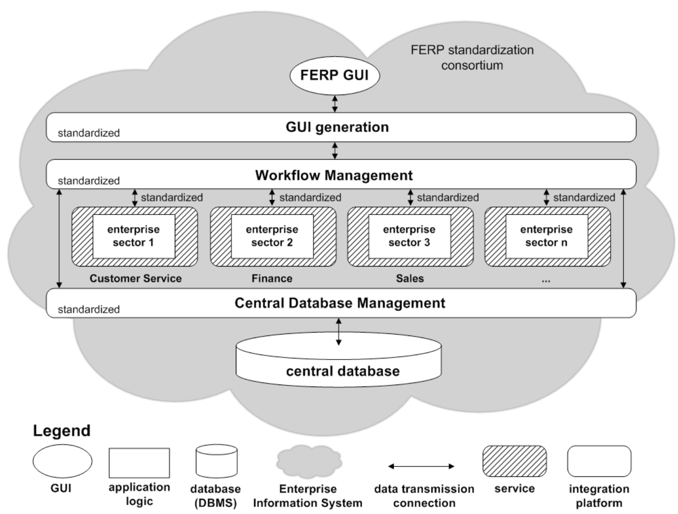


Figure 2. Vision of a Federated ERP system



In this paper we present an FERP system based on Web Services. The main idea follows the multi-layer paradigm of modern information systems which aims at the separation of the application logic from the presentation layer and the database layer. In our approach the application logic of ERP systems is encapsulated in a multiplicity of Web Services which can be provided either locally or remotely. The vision of this approach is to allow the application of business logic components in a distributed manner. In order to facilitate a vendor-independent development and provision of those components the approach considers the standardization of Web Services as well as GUI descriptions and database interactions. The standardization process is supposed to be advanced by a consortium of ERP vendors, utilizing enterprises and scientific institutions (*FERP standardization consortium*). Figure 2 shows the abstract architecture of the presented approach.

2. REFERENCE ARCHITECTURE

Figure 3 gives a survey of the reference architecture of a Web Service-based FERP system. The architecture consists of several subsystems which are interconnected. Because one of the main objective of an FERP system is to integrate business components of different vendors, all components have to comply with standards. In this approach these standards are described as XML schema documents. In order to separate the three different layers of a typical layered architecture of conventional ERP systems each layer is assigned to its own standard.

The subsystems of the proposed architecture are the following:

FERP Workflow System (FWfS)

The FWfS coordinates all business processes which have to be described in an appropriate XML-based workflow language. A workflow in this context is a plan of sequentially or in parallel chained functions as working steps in the meaning of activities which lead to the creation or utilization of business benefits. Workflows implicitly contain the business logic of the overall system. The function types a workflow in FERP systems can consist are the following:

- model based user interface functions, e.g. show, edit, select, control
- database access functions, e.g. read, update
- application tasks which are connected to Web Service calls

FERP User System (FUS)

The FUS is the subsystem which implements functions for the visualization of graphical elements and coordinates interactions with end users. This subsystem is able to generate user screens at runtime. Screen descriptions which have to comply with the *FERP UI standard* are transformed to an end device-readable format, e.g. HTML in case of web browsers.

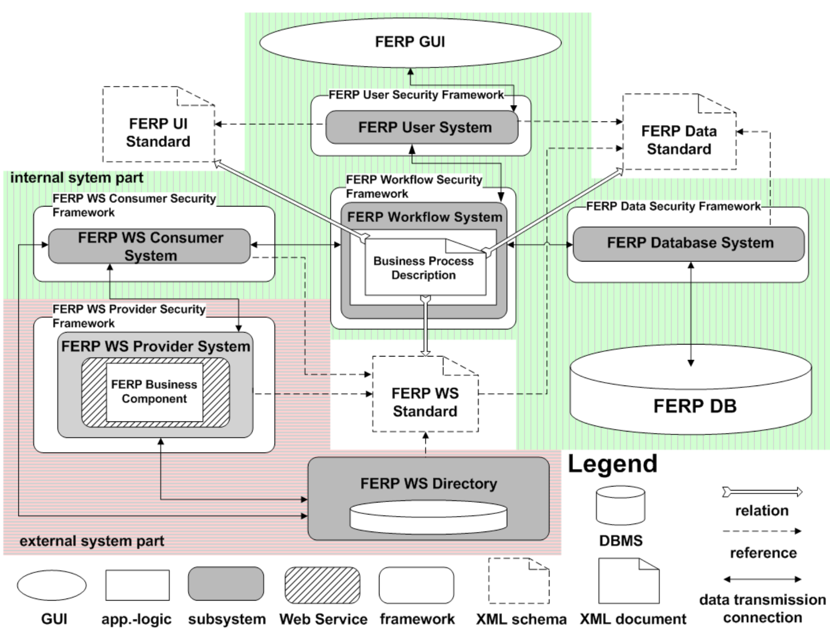
FERP Database System (FDS)

The FDS is the subsystem which implements functions for the communication with the FERP database. This subsystem is able to interpret XML structures which comply with the *FERP data standard*. The interface differentiates between two kinds of requests. Database update requests contain object oriented representations of business entities as XML trees. Database read requests contain X-Path or X-Query expressions specifying portions of data to be extracted. In both cases the request parameters have to be transformed into different types of request statements that vary depending on the type of database management system (DBMS) which is used. Assumed that a relational DBMS (RDBMS) is used the underlying data model also has to comply with the FERP data standard which means that the corresponding table structure has to reflect the XML-Schema specifications respectively. The java.net project *hyperjaxb2*[2] provides a solution to generate SQL statements on the basis of XML schema definitions. Another solution is the application of native XML databases or XML-enabled RDBMS.

FERP Web Service Consumer System (FWCS)

The business logic of FERP systems is encapsulated in so called FERP business components which are wrapped by a Web Service. The FWCS is the subsystem which provides functions for the invocation of Web Services. All possible types of FERP Web Services are specified by the *FERP WS standard*. This standard contains XML schema definitions which describe Web Service operations as well as input and output messages. A Web Service references these types in its description which is expressed in the Web Service Description Language (WSDL) which is the common standard for the description of Web Services and is already well supported by tools. Furthermore this subsystem is able to search for Web Services which are defined by a unique identifier. By this it is possible that different Web Service providers implement the same business component type as Web Service. Beside the implementation of Web Service invocation and search functions this subsystem is responsible for the interpretation and consideration of non-functional parameters. Examples for those parameters are: security policies, payment polices or Quality of Service (QoS) requirements on the part of Web Service consumers.

Figure 3. Reference architecture of an FERP system



FERP Web Service Provider System (FWPS)

The FWPS is the subsystem which implements functions for the provision of Web Services which comply with the FERP WS Standard. The subsystem includes a Web Server which is responsible for the interpretation of incoming and outgoing HTTP requests which in turn encapsulate SOAP requests. The subsystem provides business components of the FERP system as Web Services. A connection to the *FERP Web Service Directory* allows the publication of Web Services. Furthermore this subsystem is responsible for the negotiation of common communication policies such as e.g. security protocols or usage fees with the requesting client.

FERP Web Service Directory (FWD)

The FWD provides an interface for the publication and the searching of FERP Web Services based on the UDDI standard. The structure of this registry leans on the FERP WS standard. In this standard Web Services are assigned to categories mirroring the predetermined functional organization of enterprises.

3. HIERARCHICAL XML SCHEMA STRUCTURE

The proposed architecture is dependent on the specification of different standards. The next two paragraphs focus on the standardization of FERP data types and Web Service operations in the context of FERP systems. Because of the complexity of enterprise data models and the difficulty to standardize a completed data model we propose a hierarchical standardization model which allows different abstraction levels. This model uses XML namespaces for the representation of hierarchical levels and XML schema documents for the definition of data types and their relationships. The reason for the usage of XML schema documents is

Figure 4. Extract of the hierarchical Easy Access® menu of an SAP/R3® system[3]

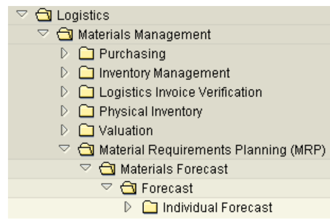
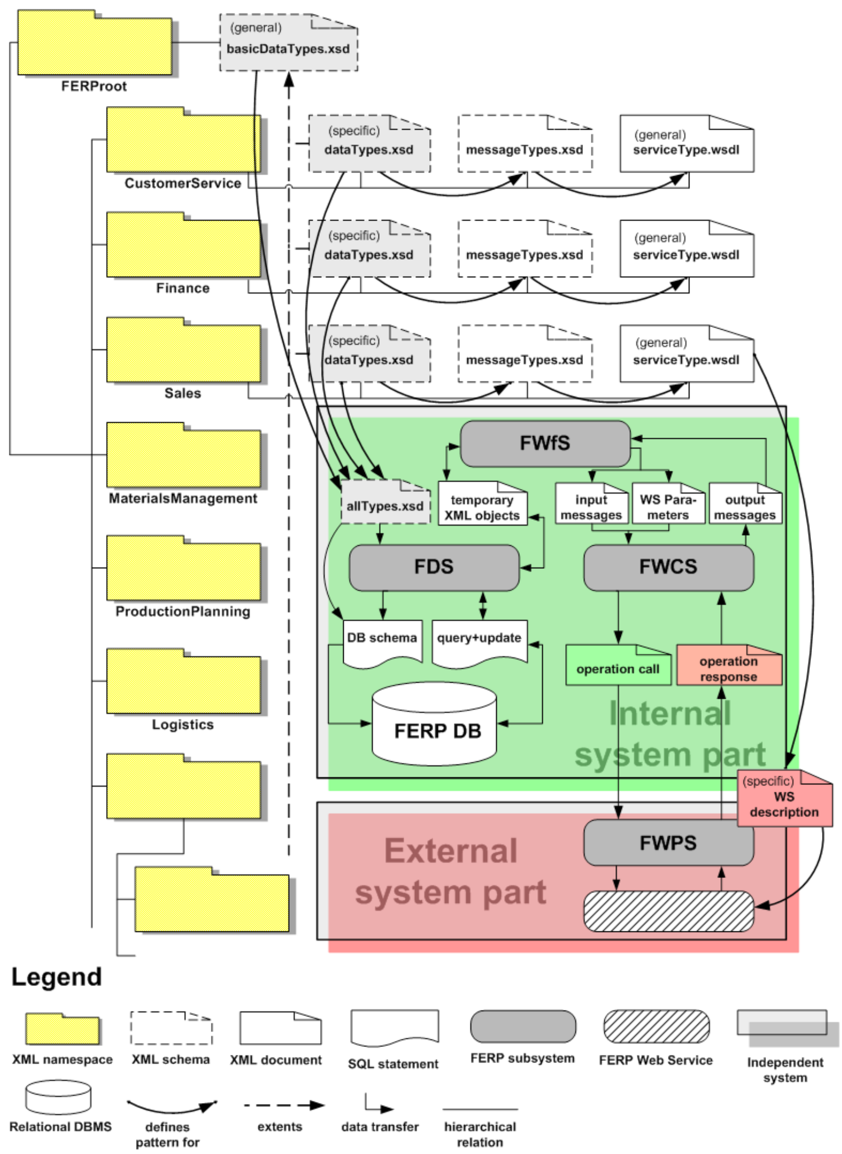


Figure 5. Hierarchical XML schema structure of an FERP system



their compatibility with WSDL. The interoperability between FERP Web Services and the FDS is achieved by a transformation of XML schema-based data model descriptions to SQL-based data model descriptions. Web Service Descriptions in WSDL reference the FERP data standard by including the appropriate XML schema documents of the standard. In order to standardize the input and output messages of FERP Web Services we propose the usage of XML schema documents as well.

Figure 5 shows the hierarchical XML schema structure of an FERP system and shows the influence on the systems activities. The left hand side represents different enterprise sectors which are assigned to XML namespaces. This hierarchy can be compared to the internal structure of the application logic of conventional ERP systems which is often mirrored to the navigation structure of their GUI. Figure 4 shows the function hierarchy of the Easy Access® menu of an SAP®/R3® system.

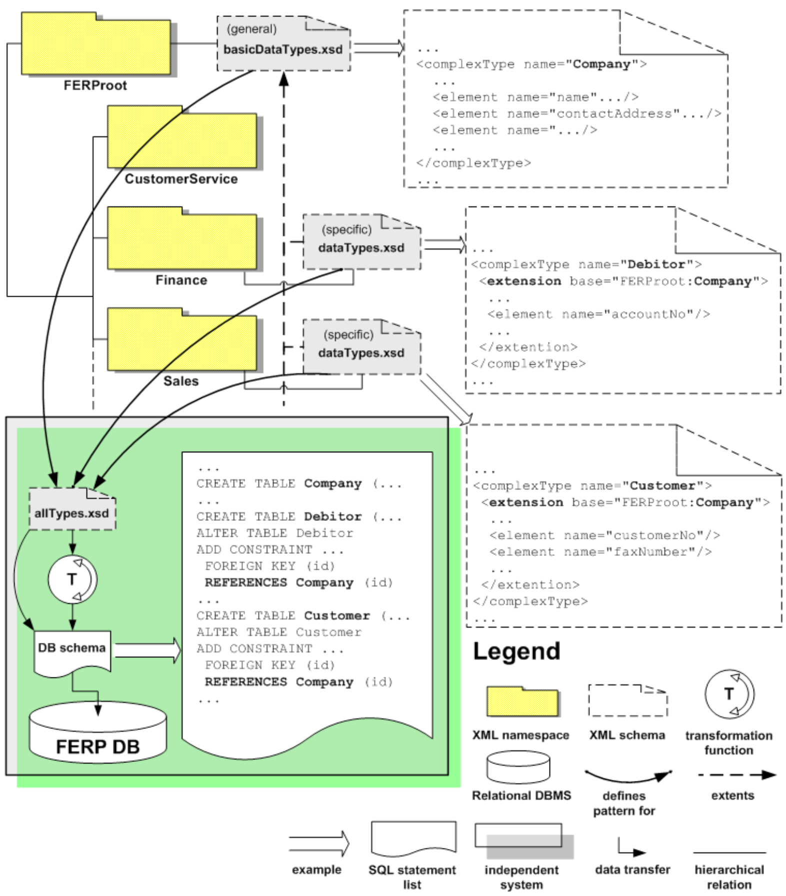
The upper half of figure 5 shows the relationships between XML schema documents and concrete Web Service descriptions. Standardized Web Service input and output messages (defined in *messageTypes.xsd*) build the basis for the standardization of Web Service types (described in *serviceTypes.wsdl*). The lower half of figure 5 shows the interactions between the different subsystems of the FERP system. The system internally creates a new XML schema document (*allTypes.xsd*) which includes a copy of all standardized data types that are used in process definitions. The system has a connection to the server of the FERP standardization consortium and will be notified in the case that the standard changed. Those changes are only allowed in terms of extensions. Thereby old versions will be supported during the whole lifetime of the standard. The hierarchical structure provides a useful foundation for this requirement because it is already field-proved in the context of object oriented programming paradigms like polymorphism, generalization and specialization. The local XML schema representation will be transformed to a relational representation of the data model as SQL statement list (see the example in figure 5 in the next paragraph). In addition to the schema transformation the FDS is able to transform SQL result sets to XML documents that comply with the FERP data standard in the case of *DATABASE_LOAD requests*. On the other hand XML documents will be transformed to SQL INSERT or UPDATE statements in the case of *DATABASE_STORE requests*. Both LOAD and STORE functions are provided by the FDS and can be used by the FWfS.

Web Service calls are initiated by the FWfS as well (see figure 5). Therefore the FWfS sends a standardized XML representation of the appropriate input message to the FWCS. A second XML document contains configuration parameters which specify the concrete Web Service provider to be chosen by the FWCS. Those parameters include either a URL for a static Web Service call or requirements for a dynamic call like e.g. a maximum price. An alternative way for the specification of requirements for dynamic calls is a centralized mapping between Web Service types and requirements. Once the FWCS chose an appropriate Web Service provider it will repack this message to a *SOAP operation request* which includes the standardized name of the Web Service operation to be invoked. This request will be sent to the FWPS. After having finished the processing of the business logic the FWPS will return a SOAP operation response which includes a standardized response message. Figure 5 shows how this response message is going to be sent back to the FWfS that primarily initiated the Web Service call.

4. DATA INTEGRATION OF DIFFERENT ENTERPRISE SECTORS

One of the main reasons why ERP systems have been achieving success in the past is their data-driven view towards the integration of different enterprise sectors. As defined above "*this integration is based on a common data model for all system*

Figure 6. Example for the hierarchical enterprise data integration



components and extents to more than one enterprise sectors" (see paragraph 1). The presented architecture of a Web Service-based FERP system reaches this integration by the hierarchical XML schema structure which is mirrored to the relational data model. Figure 6 shows an example for this hierarchical XML schema structure. In this example the general type *Company* has two specialized representations, *Debitor* and *Customer* which define additional elements. These elements can be seen as individual properties of the assigned enterprise sector. Because the FERP DB uses separate database tables for the management of these records, updates will also influence related enterprise sectors which share the same table for all general elements. We tested this functionality by the application of the *hyperjaxb2*[4] framework which draws upon *JAXB*[5] and *Hibernate*[6].

5. RELATED WORKS

The intra-enterprise integration of business application systems is commonly abbreviated as EAI[7]. According to the 3-tier architecture of business application systems today's EAI platforms support the integration over all three tiers. Enterprise portals mainly provide a basis of the consolidation of existing software systems on the user interface level which means that portals feature a user-centric orientation [8]. The Web Services paradigm implements a Service-oriented architecture which presupposes a middleware for the management of services. Search and publication requests are processed by this middleware. Business Process Management (BPM) platforms support the orchestration of such services. Thus it is possible to centralize Web Service accesses in business process definitions. In some cases EAI platforms support both a portal functionality and a BPM platform in combination with each other.

Disadvantages of domain-independent BPM platforms as foundation for the implementation of Web Service-based ERP systems are the following:

- XML object representations have to be transformed to different data models (XML schemas) when independent Web Services are used in one process because in most cases no common standard[8] is referenced in independent Web Service descriptions.
- XML object representations of Web Service return values have to be transformed to SQL statements[9] in the case that return values have to be stored in an external database. This transformation has to be part of the business process definition. Because of this problem Web Services often are assigned to their own database which is directly accessed by the business logic of a Web Service. The problem of this solution is that Web Services cannot be exchanged if they are provided externally[10] because the connected database will be not available anymore. Another problem is that such a solution would not comply with the definition of an ERP system (see paragraph 1) where the integration of different enterprise sectors is achieved by the usage of a common data model. Therefore conventional ERP systems use a central DBMS whereby also the management of database transactions is simplified.
- Input values for Web Services which have been extracted from the enterprise database have to be transformed from a database result set representation to an XML object representation which complies with the respective Web Service description.

The presented approach of a Web Service-based FERP system offers the following advantages that all together address actual challenges of business process modelling approaches[11]:

- Output values of Web Services can be directly used as input values for other Web Services because all Web Service definitions reference the same standard.
- Output values of Web Services can be transmitted to the FDS directly because both implement the same data model.
- XML object representations which have been extracted from the FDS can be used straightforward as input values of Web Services because both, the data model of the FDS and the parameter description of each Web Service comply with the same standard.

Another neighbouring working area is represented by Federated Database Systems [9] which are a type of Meta-DBMS. Those systems integrate multiple autonomous DBMS to a single system. This integration is achieved by schema mapping techniques. One solution for the data integration in Web Service-based ERP systems could be the utilization of a Federated Database System in order to consolidate independent DBMS which are directly assigned to Web Services. The Federated Database System would represent a central entry point to a decentralized DBMS structure which in turn would comply with the definition of an ERP system. This solution has the following disadvantages in comparison to the presented approach:

- Enterprises are dependent to Web Service providers who also provide one part of the database federation. In the case that different providers offer the same Web Service type a migration from one provider to another implicitly necessitates data migration.
- A global schema[12] indeed can define a normalized data model but redundancies in the overall network of independent DBMS are possible anyway. Because Web Services would directly access their local DBMS duplicate entries in the DBMS federation could lead to complications when a process or another Web Service accesses the central Federated Database System.

In comparison this approach, an FERP-system has the following advantages:

- A migration from one Web Service provider to another does not influence the data view because all data is stored in a central database which can only be accessed by a local[13] process. Web Services have no direct database connection.
- Because the hierarchical FERP standard considers the combination of Web Services' duties and affected data each level in the hierarchy is assigned to unique operation and data types. Furthermore the inheritance support allows a reutilization of general data types. Thereby redundancies in the database can be avoided because on the one hand existing data type definitions can be reused for new Web Service definitions and on the other hand standardized Web Services which use existing data types will create redundancy-free[14] data.

6. CONCLUSIONS AND OUTLOOK

Comparing distributed ERP systems and ERP systems running on only one computer, the distributed systems offer a lot of advantages. Particularly small- and medium sized Enterprises (SMB) benefit from using shared resources. However, the design of distributed system architectures is subject to a number of problems. The paper addresses the problem of redundant data in business application systems of independent vendors presents a basis for the standardization of ERP system components that are provided as Web Services. A standardized data model builds the basis for message and service standardization. The hierarchical structure of the presented standard advances the reuse of existing data types. Furthermore we presented a reference architecture of FERP systems which reduces the necessity of data transformation functions in business process descriptions.

The standardization of the syntactic level is only the first step. Behaviour, synchronization and quality of Web Services must flow into the definition of an overall ERP system standard. The future work must pick up these problems to realize the vision of a loosely coupled ERP system which allows the dynamic outsourcing of applications [5, 7] and the combination of software components of different providers.

ACKNOWLEDGMENTS

This paper has drawn upon opinions and views of many colleagues. In particular we acknowledge the review of Lars Gersmann who defined his practical position as an experienced software engineer and highlighted practical advantages of our approach which even influenced the title of this paper. Furthermore we thank Inty Saez for his review which gave us more ideas for the prototypical implementation of the presented reference architecture which is still in progress. Last but not least, also a special thanks in this order to the students of the University of Oldenburg who jointly performed lots of test implementations which delivered the foundation for many architectural design decisions. At the moment we are testing YAWL[15] as basis for the prototypical implementation of an FWfS.

REFERENCES

1. ABELS, S., BREHM, N., HAHN, A. & MARX GÓMEZ, J. (2006) Change management issues in federated ERP-systems – An approach for identifying requirements and possible solutions. *International Journal of Information Systems and Change Management (IJISCM),* 1, 318-335.

2. BREHM, N. & MARX GÓMEZ, J. (2005) Secure Web Service-based resource sharing in ERP networks. *International Journal on Information Privacy and Security (JIPS),* 1, 29-48.
3. BREHM, N. & MARX GÓMEZ, J. (2005) Standardization approach for Federated ERP systems based on Web Services. *1st International Workshop on Engineering Service Compositions.* Amsterdam.
4. BREHM, N. & MARX GÓMEZ, J. (2007) Federated ERP-Systems on the basis of Web Services and P2P networks. *International Journal of Information Technology and Management (IJITM)*.
5. BREHM, N., MARX GÓMEZ, J. & RAUTENSTRAUCH, C. (2005) An ERP solution based on web services and peer-to-peer networks for small and medium enterprises. *International Journal of Information Systems and Change Management (IJISCM),* 1, 99-111.
6. CUOMO, G. (2005) IBM SOA "on the Edge". *ACM SIGMOD international-conference on Management of data.* Baltimore, Maryland, ACM Press.
7. DAN, A., DAVIS, D., KEARNEY, R., KELLER, A., KING, R., KUEBLER, D., LUDWIG, H., POLAN, M., SPEITZER, M. & YOUSSEF, A. (2004) Web services on demand: WSLA-driven automated management. *IBM SYSTEMS JOURNAL,* 43, 136-158.
8. SHETH, A. P. & LARSON, J. A. (1990) Federated database systems for managing distributed, heterogeneous, and autonomous databases. ACM Computing Surveys (CSUR), 22, 183 - 236.
9. SMITH, M. A. & (2004) Portals: toward an application framework for interoperability. Communications of the ACM, 47, 93 - 97.
10. VERNER, L. (2004) BPM: The Promise and the Challenge. *QUEUE (ACM Press),* 2**,** 82-91.

ENDNOTES

1 In this term, a service is a software component that encapsulates one or more functions, has a well defined interface that includes a set of messages that the service receives and sends, and a set of named operations [6].

2 Hyperjaxb2 – relational persistence for JAXB objects: https://hyperjaxb2.dev.java.net/ (last visit October 2006)

3 Easy Access® and SAP/R3® are trademarks or registered trademarks of SAP AG

4 We tested the inheritance support of *hyperjaxb2* in version 0.4.

5 Java Architecture for XML Binding (JAXB): http://java.sun.com/webservices/jaxb/ (last visit: October 2006)

6 Hibernate, Relational Persistence for Java and .NET: http://www.hibernate.org/ (last visit: October 2006)

7 Enterprise Application Integration (EAI)

8 No common standard means that the BPM platform is domain-independent and transformations have to be included in the process definition.

9 In case of using an RDBMS

10 Externally means that Web Services can be provided outside the enterprise's intranet by independent software vendors.

11 Currently business processes include both abstract business logic and technical constructs in an unstructured manner which complicates traceability [10].

12 In the Local as View (LaV) mapping direction the local schemas of independent DBMS are defined in terms of the global schema. In the Global as View (GaV) mapping direction the global schema is defined in terms of the underlying schemas.

13 Local in this context means that the FWfS is directly connected to the central database.

14 New Objects update old objects of the same type and the same identity because a central DBMS is used.

15 Yet Another Workflow Language (YAWL): http://sourceforge.net/projects/yawl/ (last visit: October 2006), YAWL was released under the GNU Library or Lesser General Public License (LGPL)

Related Content

Rough Set Based Similarity Measures for Data Analytics in Spatial Epidemiology
Sharmila Banu K.and B.K. Tripathy (2016). *International Journal of Rough Sets and Data Analysis (pp. 114-123).*
www.irma-international.org/article/rough-set-based-similarity-measures-for-data-analytics-in-spatial-epidemiology/144709

Vague Correlation Coefficient of Interval Vague Sets and its Applications to Topsis in MADM Problems
John Robinson P.and Henry Amirtharaj E. C. (2014). *Contemporary Advancements in Information Technology Development in Dynamic Environments (pp. 140-173).*
www.irma-international.org/chapter/vague-correlation-coefficient-of-interval-vague-sets-and-its-applications-to-topsis-in-madm-problems/111609

Integrated Design of Building Environment Based on Image Segmentation and Retrieval Technology
Zhou Liand Hanan Aljuaid (2024). *International Journal of Information Technologies and Systems Approach (pp. 1-14).*
www.irma-international.org/article/integrated-design-of-building-environment-based-on-image-segmentation-and-retrieval-technology/340774

Design of Graphic Design Assistant System Based on Artificial Intelligence
Yanqi Liu (2023). *International Journal of Information Technologies and Systems Approach (pp. 1-13).*
www.irma-international.org/article/design-of-graphic-design-assistant-system-based-on-artificial-intelligence/324761

UX Quality with Online Learning Systems and Course Materials
Elizabeth Sucupira Furtado (2015). *Encyclopedia of Information Science and Technology, Third Edition (pp. 7557-7563).*
www.irma-international.org/chapter/ux-quality-with-online-learning-systems-and-course-materials/112457