

INTERPROM – A COLLABORATIVE FRAMEWORK DRIVEN BY BUSINESS NEEDS

*Service Oriented Inter-Organisational Support for
Business Processes in Collaborative Environments*

Carsten Huth, Norbert Völker

*Department of Computer Science, University of Essex, Colchester CO4 3SQ, United Kingdom
chuth@essex.ac.uk, norbert@essex.ac.uk*

Olaf Hahn, Björn Reinhold

*PAVONE AG, Paderborn, Germany
ohahn@PAVONE.de, breinhold@PAVONE.de*

Keywords: collaborative environments, service oriented architectures, process management, workflow management, project management.

Abstract: There is an increasing demand to support business cooperation between small to medium sized enterprises (SMEs) and major companies. The InterPROM system aims to address this need by providing a service-oriented, J2EE based collaborative platform. The system is build on top of a custom enterprise services bus (ESB) which can be connected across different organisations. InterPROM characteristics include decentralised replication and locking services, organisation directories divided into private and public sections, an elaborate security system and an application manager for the life cycle management of application instances. A unique feature is its integrated approach to project and workflow management.

The InterPROM framework is being developed by a consortium of nine partners and co-founded by the European Union. The commercial lead of the project is with PAVONE AG, a medium sized software vendor specialising in the field of collaborative software environments, knowledge management and CRM solutions. Academic participants are the Universities of Essex (UK), Paderborn (Germany), and Varna (Bulgaria). The European Aeronautic Defence and Space Comp. (EADS) France acts as a pilot user for the project. This paper focuses on architecture and system design aspects of the InterPROM framework. It includes references to underlying technologies such as service oriented architecture (SOA) and J2EE (Java 2 Enterprise Edition) web applications.

1 MOTIVATION AND CONTEXT

Knowledge intensive services are often intangible, highly specific and difficult to standardise. Moreover they tend to be required quickly and need to be tailored to the specific requirements of each application. Because of these characteristics, knowledge intensive services are often outsourced to specialised suppliers. This can lead to complex demand and supply configurations such as the following:

- Construction projects which require the coordination of architects, environmental advisors, building and maintenance companies, as well as auxiliary service suppliers

- Mergers and acquisitions projects, for which buyers and sellers, investment banks, lawyers, chartered accountants, and advising companies have to work together.
- Marketing campaigns that are facilitated by market researchers, as well as advertisement and PR agencies.
- Research and development projects, business development projects, etc.

In addition to the collaboration within projects, SMEs also cooperate with large companies by supplying goods. Driven by the need to customise products quickly in response to customer wishes,

new forms of supply chains are emerging where procurers and suppliers form collaborative networks.

While such cooperation between SMEs and large companies makes good business sense, it can pose difficult management problems. This is due to the heterogeneity of the partners and their IT infrastructure, the application specific interactions, and the requirement to share resources and to coordinate activities while at the same time taking the autonomy of partners into account.

This paper describes a collaborative IT platform aimed at facilitating business processes in inter-organisational networks. The system is developed within the InterPROM research project co-funded by the European Union. The project consortium consists of nine partners. The partners responsible for the major parts of the software development activities are the Universities of Essex (UK), Paderborn (Germany), and Varna (Bulgaria) as well as the PAVONE AG in Paderborn, Germany. The European Aeronautic Defence and Space Comp. (EADS) France is the pilot user in the project.

2 SERVICE ORIENTATION

The Service Oriented Architecture (SOA) (Bieberstein et al, 2005) is a recent approach to the design of distributed systems. Its defining feature is the provision of well defined, independent IT services offered by service providers. Service consumers access and use these services.

A major motivation behind the SOA approach is the prospect for service reuse. The vision is to enable “programming in the large” whereby services can be assembled dynamically (“service orchestration”, “service choreography”) with languages such as WSBPEL (Web Services Business Process Execution Language) (OASIS 2006). The independence of services also aids incremental development and the integration of legacy systems by the use of adaptors. In contrast to earlier approaches like CORBA, SOA relies on loosely coupled and standardized but open protocols independent from particular programming languages. In practice, the invocation of services takes mostly the form of XML based standardized web services.

Direct point-to-point connections between service providers and service consumers makes it difficult to enforce binding rules concerning security, Quality of Service (QoS), or the logging and billing of services. Additionally, many applications set further requirements for the IT

infrastructure such as support of transactions, asynchronous communication (messaging) and the dynamic detection of services. Such requirements lead to the introduction of an Enterprise Service Bus (ESB, or short service bus) which meets the above mentioned requirements by providing a standardized medium to which services can be bound in order to be located and executed. The core of an ESB is a messaging system which mediates between the service providers and service consumers. In addition, an ESB often also offers supporting services like the transformation of data formats and the controlling and auditing of the network traffic.

The flexible integration of heterogeneous components is the main reason for choosing a SOA for the InterPROM system. Based on an analysis of typical InterPROM applications, the decision was made to develop an ESB that uses XML web services and J2EE. This service bus implementation largely follows the guidelines of the Java Business Integration (JBI) (JSR 208) standard (Ten-Hove, Walker 2005). The InterPROM ESB offers additional extensions that are indispensable for the intended application domain:

- ESB instances which are in use in different companies (or different departments of one large-scale company) can be connected to each other. Thus, applications of one company (or department) that only have access to their own ESB can transparently use services of another company.
- The visibility of services beyond the border of the ESB of one company can be limited if necessary. Services can be offered locally (within the domain of only the originating company itself), globally, or within a selected number of other companies who take part in collaborative activity.
- The ESB incorporates security mechanisms which perform authentication and authorisation of service requests as well as the encryption of messages.

3 AN ARCHITECTURE FOR THE SUPPORT OF PARTNER NETWORKS

The InterPROM system aims to support the collaboration of partners across organisational boundaries. In the following, we will refer to the group of cooperating partners as a partner network.

Business Context and Key Requirements

At the start of the project, PAVONE AG carried out market research including a survey among its customers. Several key points emerged which have a bearing on the InterPROM architecture:

- It has to be easy to set up partner networks and integrate new partners in an existing partner network.
- The dependencies between partners should be kept as low as possible. Only information relevant for the joint business activities should be shared. It should not be necessary for partners to disclose sensitive information beyond the scope of the cooperation.
- The integration of existing applications at the various partners must be facilitated. In this way, the existence and operation of redundant parallel systems can be avoided.
- In particular, large companies often use enterprise level documents management systems (DMS) or enterprise resource planning (ERP). It would be unfeasible for SMEs to run such systems. Therefore connectors are required which enable the collaboration within a project. In this way, the IT requirements of each partner can be kept to a minimum.

The priorities attached to these requirements varied between different kinds of companies. For SMEs, the protection of autonomy and control of data was very important. This was less of an issue for the larger companies who typically viewed themselves as hosting and controlling the data. On the other hand, it was especially larger companies who emphasised the need for an easy set-up and management of the system, as they regarded themselves as the initiators and to some extent maintainers of the partner networks.

Companies of all sizes viewed the integration with existing applications and legacy systems as a compulsory prerequisite for the introduction of a new collaborative environment. There should be no redundant or parallel processing of tasks and data on existing and new systems. One of the top requirements for the InterPROM platform is therefore the seamless integration with existing systems.

Outline of Architecture

The overall architecture of the InterPROM platform is depicted in Figure 1. As already mentioned in Section 2, the system makes use of a custom ESB that provides the SOA infrastructure. The ESB is also used to plug in *connectors* to legacy

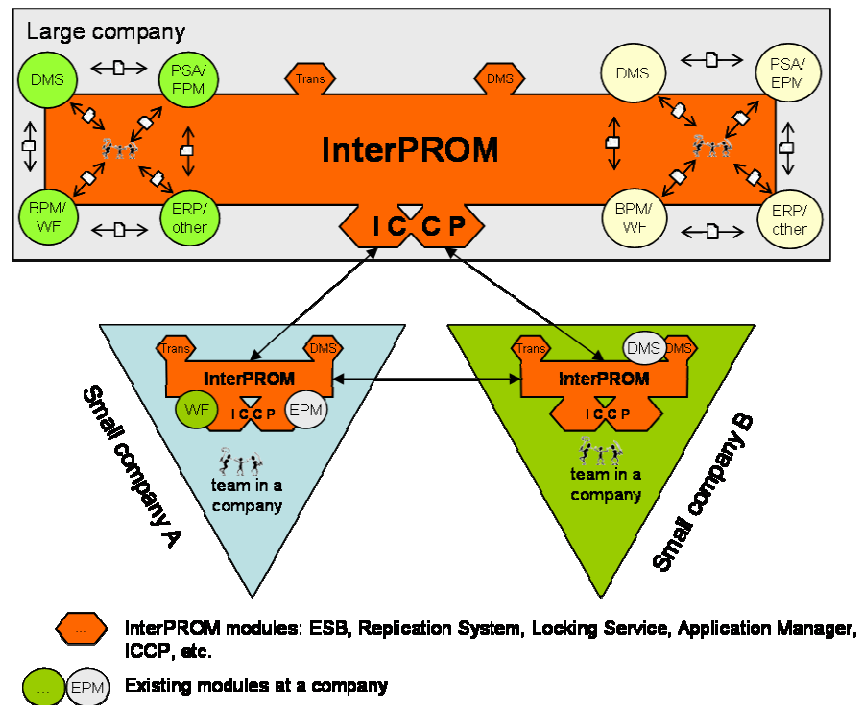


Figure 1: Distributed architecture of the InterPROM system

systems, thus making them available within the InterPROM system.

Replication services allow the synchronisation of data at the different partners. When modifying shared data, write conflicts can be avoided with the help of *locking services*.

A distributed *Organisation Directory* provides the basis for the security system.

The *Application Manager* facilitates the easy set up and maintenance of applications and manages the distribution of applications on the partner network.

The ICCP (InterPROM Collaboration Centric Processes) engine forms an integrated execution environment for projects and predefined workflows.

The InterPROM system provides a web-based user interface for administration of the system as well as for InterPROM applications. All management functions are accessible through a portal environment that makes use of Portlet Specification JSR 168. Assuming end user applications conform to the rules of the Struts framework, they can be converted into portlets with the help of the Apache Struts Bridge (Apache 2005). By combining different portlets into a portal, it becomes easy to construct web interfaces that are tailored to the needs of particular user groups.

The basic paradigm of the InterPROM architecture is decentralisation. A partner network does not contain central servers without which it would break down. Instead, each partner in a network can run his own InterPROM system. The InterPROM instance at a particular partner can be added or removed without effecting the operation of the rest of the partner network.

The local availability of the InterPROM infrastructure at each partner together with the service orientation facilitates the integration of third party and legacy applications. Typically, such additional applications will be locally integrated. Data exchanges between the applications can take place by sending messages across the ESB or by sharing data, for example with the help of the InterPROM replication services.

Replication and Locking Services

The InterPROM platform does not make any assumptions about the structure or the internal workings of applications deployed on a partner network. In particular, there are no binding conventions with respect to the way applications store or retrieve data. Therefore it is impossible to offer a fully automated replication system as it is for example provided in Lotus Notes/Domino (cf.

Kawell et al. 1992) or in relational database systems. For this reason, the lock mechanism and the replication functionality are offered as services which can be used by applications deployed on the InterPROM framework.

In the literature, several approaches to decentralised data storage can be found, cf. (Buretta 1997). For the first version of the InterPROM system, it was decided to employ synchronous replication. With this technique, shared data which is modified at one of the partners is immediately replicated via the ESB to all the other partners within the partner network.

The advantage of synchronous replication is that it guarantees the consistency of data. However, this technique also has significant disadvantages. In particular, it requires high availability rates of servers within the partner network. In addition, high performance data connections might be required if there is a large volume of synchronisation messages. It would be unrealistic to expect a commitment to such high qualities of service for all partner networks. Hence it is planned to provide additional synchronisation strategies in later versions of the InterPROM system.

The locking service offers the possibility to lock data items across servers, and thus allows the exclusive modification of a certain data item. The locking service uses abstract entities, so called "items". These are represented by universal unique identifiers (UUIDs) that are application independent.

It has to be stressed that it is up to the applications deployed in the partner network whether or not to make use of the replication and locking services on offer. Furthermore, while the InterPROM platform provides the infrastructure, it is the applications which are responsible for the sending and the handling of replication and locking messages.

Security System

The two main aspects of the InterPROM security system are authentication and authorisation.

An InterPROM application authenticates a user by calling the login module. This then consults the InterPROM organizational directory or any other LDAP compatible directory in order to verify the user's credentials. The organisational directory uses the JAAS security model (Lai et al, 1999).

Because of the particular demands of security control in collaborative environments, the InterPROM platform offers its own, fine-grained authorisation. This system is inspired by the Lotus

Privileges/ Access Level	Create Items	Delete Items	Read Items	Write Items	Copy Items	Execute Items	Modify App ACL	Read public Items	Write public Items	Modify Item ACL	Tra- verse
Manager	Y	Opt/Y	Y	Y	Opt/Y	Y	Y	Y	Y	Y	Opt/N
Editor	Y	Opt/Y	Y	Y	Opt/Y	Opt/Y	N	Y	Y	Opt/Y	Opt/N
Author	Opt/Y	Opt/N*	Y	Y*	Opt/Y	Opt/Y	N	Y	Opt/N*	Opt/Y*	Opt/N
Reader	N	N	Y	N	Opt/N	Opt/Y	N	Y	Opt/N	Opt/N	Opt/N
Depositor	Y	N	N	N	Opt/N	N	N	Opt/N	Opt/N	N	Opt/N
No access	N	N	N	N	N	N	N	Opt/N	Opt/N	N	Opt/N

*: There are special rules in place for the access level "Author".
Optional privileges have a default, i.e. "Opt/N" means that the default is "No".

Table 1: Access Levels and Privileges at Application Instance Scope

Notes approach to authorisation (cf. Nielsen et al. 1999). It provides services and functionality to secure access to both applications and their resources. It is up to the applications to utilise these facilities.

Access control makes use of six predefined access levels namely: Manager, Editor, Author, Reader, Depositor, and No Access, see Table 1. Each access level defines a set of privileges. It is not possible to add new types of access levels, but existing ones can be customised within limits, see the "Opt" privileges in Table 1. The limitation of customization is done intentionally in order to preserve the meaning of the various access levels.

Each *application instance* (see Section 5) possesses an access control list (ACL). This ACL associates entities from the organizational directory (such as person, group, organizational unit, etc.) with access levels. In this way, it determines the access rights of users to the application and its resources. Only a user with the access level Manager can modify the application ACL. The application instance ACL is also the place where a customisation of access levels for a particular instance can be performed.

The predefined access levels reduce the effort to specify application instance level access considerably. By associating an organization directory entity with an access level, a well defined value for each of the privileges (i.e. create items, delete items, etc) is ensured.

In addition to the access control on the application instance scope, the access to an entity can be further restricted at the scope of items. An item can for example be a document or a collection of database records. Each item is identified by a unique ID. Similar to an application ACL, an item ACL describes an entity's access to a particular item by assigning/revoking privileges. Forbidding access

to a resource at the application instance scope overrides item scope permission for that entity. Traverse privilege on the application instance scope can be granted for the unusual case that exceptions to this rule are required.

4 ORGANISATION MANAGEMENT IN COOPERATIVE NETWORKS

The organisation directory (OD) plays a major role in the management of partner networks. Since the InterPROM system aims to be completely distributed without any central component, an instance of the organisation directory resides at each of the partners running the InterPROM system. Each instance is divided into a public and a private section, where public means that the information is to be shared within a particular partner network.

The private section of the OD, which is hidden from all other partners, contains the structural organisation of the company to which this InterPROM server belongs. This section has to list all persons, resources, etc. that will be take part in one ore more of the partner networks in which this company is involved. Connection facilities are provided to import organisation information from existing directories and keep them automatically synchronised. The private section of the organisation directory also contains specifications of the *partner connections* between the organisation and its partners within all the different partner networks it is involved in. A partner connection entity comprises information like name, password and network address of a partner server.

The public section of the OD forms the basis for the management of InterPROM applications in a

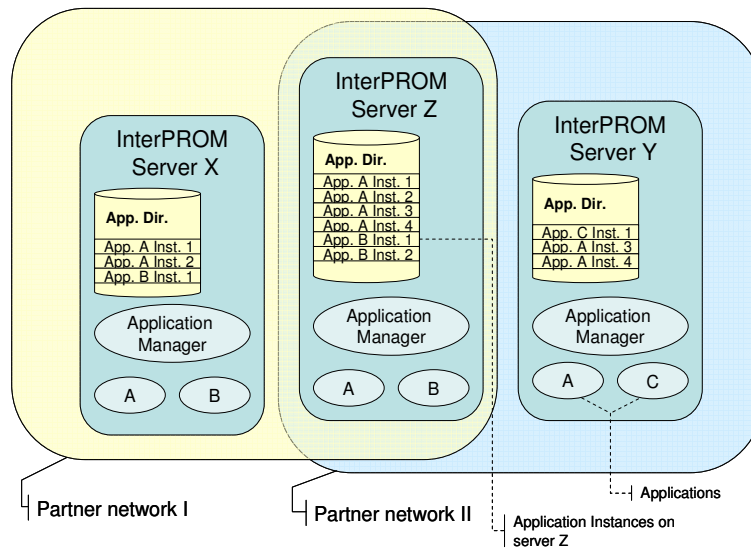


Figure 2: The Application-Manager in a scenario of distributed InterPROM servers

partner network. As such, it needs to specify the employees, resources, roles, etc. that are involved in the collaboration. This is typically done by a list of UUIDs which refer to entities in the private sections of the ODs belonging to the various members within the partner network. It is also possible to create new entities which only exist within the context of a particular partner network, such as network specific groups and roles.

The first step in the construction of a partner network is the creation of a partner network specification. Subsequently members can be added to the network. Before doing so, each new member has to be associated with a partner connection. By adding or removing members from a partner network, the scope of that network can be adapted.

The partner network specification, the membership information as well as the public section of the organisation directory are synchronised among all the members of a partner network.

5 THE APPLICATION MANAGER

In addition to the requirements stated above, the creation and management of applications themselves is of substantial significance for collaborative environments: Companies are typically involved in a number of different projects in which the use of computer supported collaborative environments would be beneficial. However, non-disclosure agreements and information restriction requirements

within these projects make it necessary to keep the information pools of different collaborative projects disjoint.

In today's web based application environments this leads to the repeated installation of collaborative applications for each project. All of these set-ups have to be maintained by specifically skilled technical personnel, i.e. system administrators. Tasks like installing updates, maintenance releases and bug fixes have to be carried out several times for each separate installation of a particular application. This means that the installation and maintenance of applications requires a substantial amount of time and hence operational expenditure, and the set-up of applications is inherently inflexible.

Therefore the InterPROM project pursues a distinctive approach to handle applications: an application is only installed once on each server by the system administrator. In order to make use of the application, a so called *application instance* is created. Each application instance forms a self-contained space to use the application and comes with an access control list (ACL). As explained in Section 3, this ACL associates organisation directory entities with access levels and thus defines the privileges of users and roles for this particular application instance. By applying this concept, several instances of an application can be created without the need of installing and setting up the application over and over again. In addition, the creation of an application instance becomes relatively easy, and can be accomplished by experienced business users rather than system

administrators, because it does not include the technical process of setting up an application on an application server.

The definition of application instances leads to the introduction of the *application manager* as a separate component of the InterPROM system. Apart from controlling the life cycle of application instances, the application manager also determines the distribution of applications on the partner network. The application manager uses an *application directory* to store the necessary information about applications and application instances.

InterPROM applications are generally J2EE compliant applications. In addition, an InterPROM application has to implement a specific interface which contains the functions required by the application manager to control its life cycle, i.e. instance creation, publication, or the deletion of an instance from the partner network. Furthermore, the application manager controls the replica copies of the application instances which reside on different servers of the partner network.

As the InterPROM system is truly distributed and therefore has no central components, the application manager and application directory need to reside on every InterPROM server. In order to maintain an up-to-date application directory, the application manager component of each InterPROM server communicates via the ESB to synchronise the application directories.

6 SYNTHESIS OF PROJECT AND WORKFLOW MANAGEMENT

Traditionally, there is a strict distinction between project and workflow management activities. However, the empirical research that was conducted at the start of the InterPROM project has shown that especially for SMEs, such a strict separation is not always justified. Companies requested combined tools with process support tailored to their individual requirements rather than adhering to the, from their perspective, sometimes artificial separation between project and workflow management. There are similar requests for more flexible process support of agile project management (Augustine et al. 2005).

One of the key objectives of the InterPROM project is therefore to combine the strengths of both kinds of management systems into a new generation of business process support systems. In line with the aims of the InterPROM project, the focus will be on

processes as they occur in collaborative workflow management systems. Highly repetitive, automated production workflow processes will be disregarded here.

The section below describes this integrated approach. The development of corresponding tools is currently in progress. It is based on existing, well established project management and workflow management systems (WFMS) at PAVONE AG. An integrated XML format has been defined that supports the conversion and integration of processes represented as XML documents.

A) Intertwining of Projects and Workflows

A pure workflow-based approach to business process management still fails frequently because of the need to predefine the structure of the whole processes at an unnecessarily detailed level, thus leading to excessively rigid models (Aversano and Canfora 2002). In practice, business processes involve both highly structured parts which follow a workflow pattern as well as less structured parts that are more suited to project management.

For instance, a marketing campaign might start with a creative phase, and once an agreement on an idea is achieved and a first draft of the marketing material has been created, the process might follow the structure of an established workflow including steps like revision and finally printing, distribution etc.

On the other hand, a process which generally follows the pattern of a highly structured workflow instance might contain parts that are more of a project nature. As an example, consider the processing of software problem reports by a software provider. While a majority of such problem can be answered and solved in a structured way, i.e. as a workflow instance, some of them might require a more thorough investigation in order to be solved or responded to. Therefore the initiation of a project would be an appropriate action to take at such a point.

In the InterPROM approach, two kinds of intertwining of projects and workflow instances will be supported. Type A represents a workflow instance or a project forming a sub-structure of the respective other type, i.e. a workflow instance is executed as a sub structure of a project task or vice versa. Type B represents the case where a process changes its type permanently, i.e. a project turns at some point into a workflow instance, or a process that starts off as a workflow is turned into a project and completed as such. This intertwining of

workflow instances and projects requires an integrated user interface for both process types and a certain integration of the underlying workflow engines and project management tools

B) Guided Process Type Conversions

Normally it is assumed that projects are unique and only executed once. In fact, this is one of the characteristics that distinguish projects from workflows. In spite of this, sometimes new workflows in organisations emerge from project executions. A process that starts as a project in its first execution may become a successful reference example for the future. Especially for knowledge based services, best practises developed in a project might become more established. Hence, as a next step, it might be useful to formalize and automate that same process in form of a workflow.

In the InterPROM approach, this conversion will be supported by tools that transform a project model into a rudimentary workflow definition. While it is expected that there will be less business need for a conversion in the other direction, it is also planned to provide tools that transform a workflow protocol, i.e. the sequence of steps in a particular workflow instance, into a project template.

Despite the tool support, the result of the conversion can in both cases only serve as a draft for the new process type. For example in the case of the project-to-workflow conversion, the user will have to abstract concrete persons or resources into roles or resource types. Furthermore, it is the user's job to endow the project tasks with conditionals and other control structures. The conversion in the opposite direction requires the opposite of these steps, i.e. instantiations of roles and resource types and a linearization of workflow control structures. Additionally, task completion times and costs have to be allocated.

C) Extensions of Integrated Process Support

Craven and Mahling (1995) suggest building a new type of process management system integrating project and workflow management around a comprehensive shared task notion that combines aspects of workflow and project tasks. The InterPROM approach keeps the distinction between workflow instances and projects. However, the functionality of the corresponding management systems is extended in two ways.

Firstly, the management of resource utilisation is to be added to the WFMS. Facilities for the

posterior analysis of workflow instance resource utilisation, costs and duration will be provided. Averaging these values for a particular workflow type can help predicting the values for future executions of that type.

Secondly, a function for the prediction of project completion time and expenditure will be added to the project management component. For this purpose, project tasks can be assigned with conditions that have to be met for their completion, and expected probabilities for the failure of these conditions. A continuously updated prediction of the project completion time can then be made based on these probabilities, which then allows the calculation of delay times of tasks in conjunction with resource utilisation ratios. This prediction can be used to take corrective action in a timely manner, e.g. assign further resources or change the project plan. The corrective actions again lead to updated project completion time calculations.

By utilizing the extended functionality outlined above, it is expected that a seamless and more comprehensive resource planning support can be provided, including a better utilisation of human resources employed in workflows and projects (cf. Bahrami 2005).

Overall, the better integration of project and workflow management described in this section should help to extend the applicability domain of business process support systems for collaborative environments.

7 CONCLUSIONS

The InterPROM system provides a J2EE based collaborative platform for the support of inter-organisational networks. The decentralised, service-oriented architecture makes it easy to integrate third-party applications and to add or remove partners in a partner network. The security model allows fine-grained access control for applications and resources. An application manager facilitates the life cycle management and distribution of application instances.

A unique feature of InterPROM is a more integrated approach to project and workflow management. Projects and workflow instances can be intertwined and there are conversion tools for turning a project model into a rudimentary workflow as well as turning a workflow instance into a project template. Projects tasks can be labelled with conditions in order to express the dependency of

task completion on various factors. Together with the statistical analysis of workflow instance data, this should help to better predict process duration, costs and resource utilisation.

The system will be evaluated in a pilot study at EADS as well as in other end-user projects. The first concrete applications, which are currently being developed on top of the InterPROM platform, are project, risk and supply chain management solutions. The results from these case studies will inform future work such as the implementation of further replication services, or enhanced support for service orchestration and choreography (Zimmermann et al. 2005).

REFERENCES

- Apache Software Foundation 2005: Apache Portals Project. See <http://portals.apache.org>
- Augustine, S.; Payne, B.; Sencindiver, F.; Woodcock, S. 2005: Agile project management: steering from the edges, Communications of the ACM, Volume 48 Issue 12, ACM Press
- Aversano, L.; Canfora, G. 2002: Process and Workflow Management: Introducing eServices in Business Process Models, Proceedings of the 14th International Conference on Software Engineering and Knowledge Engineering SEKE '02, ACM Press
- Bahrami, A. 2005: Integrated Process Management: From Planning to Work Execution, Proceedings of the IEEE EEE05 International Workshop on Business Services Networks BSN '05, IEEE Press
- Bieberstein, N.; Shah, R.; Bose, S.; Fiammante, M.; Jones, K. 2005: Service-Oriented Architecture Compass: Business Value, Planning, and Enterprise Roadmap, ISBN: 0-13187-002-5
- Bullinger, H.-J.; Murmann, H. 1999: Dienstleistungen - der dynamische Sektor; Universum Verlag, Wiesbaden, 1. Auflage.
- Buretta, M. 1997: Data Replication: Tools and Techniques for Managing Distributed Information, Wiley & Sons Inc., New York.
- Cramer, J. 2004: Management wissensintensiver Dienstleistungen; in Hermann, Sibylle (eds.): Ressourcen strategisch nutzen; Fraunhofer Irb Verlag, Stuttgart, 1. Auflage, pp. 180-206.
- Craven, N.; Mahling D. 1995: Goals and Processes: A Task Basis for Projects and Workflows, Proceedings of Conference on Organizational Computing Systems, ACM Press
- Nielsen, S. P.; Dahm, F.; Lüscher, M.; Yamamoto, H.; Collins, F.; Denholm, B.; Kumar, S.; Softley, J. 1999: Lotus Notes and Domino R5.0 Security Infrastructure Revealed, IBM International Technical Support Organization. See <http://www.redbooks.ibm.com>
- Kawell, L.; Beckhardt, S.; Halvorsen, T.; Ozzie, R.; Greif, I. 1992: Replicated Document Management in a Group Communication System, in: Marca, D.; Bock, G. (eds.): Groupware: Software for Computer Supported Cooperative Work, IEEE Computer Society Press, pp. 226-235.
- Lai, C.; Gong, L.; Koved, L.; Nadalin, A.; Schemers, R. 1999: User Authentication and Authorization in the Java™ platform. Proceedings of the 15th Annual Computer Security Applications Conference, Phoenix, AZ, 1999. See also <http://java.sun.com/security/jaas>
- OASIS (Organization for the Advancement of Structured Information Standards) 2006: Web Services Business Process Execution Language (WSBPPEL). See <http://www.oasis-open.org>
- Ten-Hove, R.; Walker, P.; (eds), 2005: JSR-208 Java Business Integration Evaluation. See <http://www.jcp.org>
- Voigt, K.-I.; Thiel, M. 2003: Beschaffung wissensintensiver Dienstleistungen - Net Sourcing als alternative Bezugsform; in Bruhn, M.; Stauss, B. (Ed.): Dienstleistungsmanagement Jahrbuch 2003 - Dienstleistungsnetzwerke; Gabler Verlag, Wiesbaden, 1. Auflage, pp. 287-318
- Zimmermann, O.; Doubrovski, V.; Grundler, J.; Hogg, K. 2005: Service-Oriented Architecture and Business Process Choreography in an Order Management Scenario: Rationale, Concepts, Lessons Learned. OOPSLA'05, Companion, pp. 301-312 ISBN:1-59593-193-7