

# Collaborative Knowledge Flow – Improving Process-Awareness and Traceability of Work Activities

Schahram Dustdar

Information Systems Institute, Distributed Systems Group,  
Vienna University of Technology,  
Argentinierstrasse 8/184-1, 1040 Wien  
dustdar@infosys.tuwien.ac.at

**Abstract.** Next-generation business solutions in the domain of collaborative knowledge work require the integration of process modeling, groupware, and workflow technologies. This paper aims at discussing the fundamentals required for improving process-awareness for group collaboration. Prospective process-aware collaborative tools are required to record, map and manage processes involved in knowledge work: from creating ideas to their development into profit contributors. Development activities involve distributed teams working across organizations, time, and locations. We present a case study and an implementation of a collaborative knowledge flow solution, enabling the traceability of collaborative work activities for team members. This leads to increased efficiency of New Product Development teams as well as to faster time-to-market of product and services.

## 1 Introduction

Innovation and the development of products and services are the heart of every organization. Companies operating in highly competitive markets have an intensive need for continuous innovation and faster time-to-market for their products and services. Those products and services are the result of complex and expensive business processes. Involved processes and outcomes are highly knowledge intensive and therefore require sophisticated tools for capturing and managing knowledge capital within the enterprise, with partnering organizations, and with customers. Prospective process-aware cooperative tools are required to record, map and manage processes involved in knowledge work: from creating ideas to their development into profit contributors. Development activities involve distributed teams working across organizations, time, and locations. Every day the need for creating and replicating collaborative and innovative processes of the organization rises.

Solutions required for highly efficient and effective knowledge logistics (i.e. who does what, when, how, why, using which resources) [4] require new technological concepts, which go well beyond current software systems such as groupware, workflow [3, 5, 6, 7, 8, 12], project - and knowledge management. The reason is that processes as in product and service development require highly integrated and flexible

systems supporting multiple organizational structures interlinked with business processes and resources [e.g., 11]. A flexible state-of-the-art software architecture for distributed, cross-organizational and multi-disciplinary teams using multiple devices [2], is of paramount importance. To solve this problem future solutions must go further than current systems, which are solely based on documents, shared folders, workspace structures, modeling projects and processes without integration and feedback to their real execution. Some fundamental requirements are to

- provide a chain of linked information, which represents the results (what), who created them, how, when and why they were created (context).
- automatically build “knowledge trails” of activities and their relationships to knowledge processes as well as “Best practice“ libraries.
- provide views on contextual information of work activities, how they relate to the work of others and their relationships to business processes. All relationships, activities, and actors described above may be distributed on the Internet.

Capturing the process by which knowledge is collaboratively developed is as important as documenting the output of group collaboration. In other words, “results” must provide more than data and information - they must also serve as resources for documenting “how we arrived at the outcome“. We propose to streamline the connections between people and processes. People are the greatest factor in an organization’s ability to compete based on knowledge. Successful results depend on knowledge transfer among various business processes. In every kind of business, core knowledge processes are intertwined and interdependent.

The remainder of this paper is structured as follows: Section 1.1 presents related work in this research area as well as states the main contribution of this paper. Section 2 analyses the requirements for collaborative knowledge flow systems based on a New Product Development (NPD) case study, including aspects on organizational modeling, processes and activities, data management and business objects, and the management of knowledge flow itself. Section 3 discusses design- and implementation issues related to the software presented in this paper. Finally, section 4 concludes the paper.

## 1.1 Contribution and Related Work

Collaborative systems can be categorized using multiple criteria and dimensions. Ellis [9], for example, presents a functionally oriented taxonomy of collaborative systems, providing the required understanding for integration of workflow and groupware systems – the fundamental information systems for support of collaborative knowledge flow within and between organizations. Ellis distinguishes between (i) Keepers, (ii) Communicators, (iii) Coordinators, and (iv) Team-agents. Keepers follow a shared workspace metaphor (e.g. a database) and provide access control on artifacts, versioning, and concurrency control. Communicators are messaging oriented and support explicit communications between participants. Coordinators are based on an organizational model and provide support for the ordering and coordination of activities.

Team-agents (e.g. applications or user-interface agents) provide domain-specific functionalities, such as meeting scheduling.

The contribution of this paper is the design and implementation of an innovative collaborative knowledge flow system - Caramba [4, 10] - motivated by a case study in the area of New Product Development. The proposed system integrates three of the categories discussed above: Keepers, Communicators, and Coordinators.

## 2 Collaborative Knowledge Flow

Advances in the area of Workflow (WfMS) and Groupware are often seen as substantial for supporting distributed knowledge work. Collaborative knowledge work in teams is increasing and as a consequence the use of collaborative knowledge flow solutions are becoming increasingly pervasive. WfMS have been defined as "technology based systems that define, manage, and execute workflow processes through the execution of software whose order of execution is driven by a computer representation of the workflow process logic" [13].

Groupware systems usually provide very low knowledge context information but enable users to retrieve, share, organize their work in workspaces, and to distribute artifacts. Document Management systems are increasingly integrated with WfMS as recent mergers demonstrate (e.g. Lotus Notes/OneStone). Project management (PM) software is still mostly viewed as software for individuals (i.e. project managers) and rarely offers collaborative or business process-aware solutions. Moreover, in most cases PM software is not integrated with corporate information systems and in fact is only utilized as a graphical modeling tool for outlining tasks. Most Knowledge Management (KM) systems on the market today are workspace-oriented and provide very simple support for modeling organizational structures (e.g. using roles only, but not skills). It is interesting to note that nearly no KM system provides interfaces to business process modeling and enactment systems (the domain of WfMS). Most KM-systems enable users to retrieve knowledge artifacts from repositories, but rarely allow distribution and process-awareness of collaborative work activities.

Knowledge can be viewed as information enriched with context. With context we mean information about the "who, when, how, and why". As an example, consider an "Explorer"-like view on a file system. This view allows the person to see documents (artifacts) stored inside folders. The name of such folders might reflect project names themselves. The mentioned view on these documents does not contain further contextual information on what a person (yourself, or others) actually have to do (did) with it (e.g. create another document, send an e-mail to customer, call partner organization, etc.). For example if the person in the above example needs to see who actually received a document stored in any given (project) folder, he is required to manually retrieve his e-mail box in order to find this information. This simple example shows that links between artifacts, such as documents or database information, and activities performed by persons are usually not stored in information systems such as Groupware, KM or WfMS. However this linkage is of paramount importance for knowledge-intense business processes in order to provide contextual information on knowl-

edge artifacts for processes such as New Product Development, which cannot be modeled using a WfMS. The reason for this is that NPD processes are semi-structured by nature and many exceptions occur during enactment of the processes. This makes it impractical to remodel the process model every time an exception occurs in order to continue the process flow.

## 2.1 Case Study

Consider a New Product Development (NPD) process such as in software development. In our case the team consists of 9 *persons* and has the goal to jointly develop a new product and some product extensions. All team members have a set of *skills* (e.g. Java, Oracle, etc.), which are fundamental for the success of the overall development process. The overall NPD process consists of a set of *subprocesses* (e.g. architectural design, database design etc.). Since team members have different skill-sets, they also have different *roles* (e.g. Product manager) in the NPD team. Based on the roles, team members have different *tasks* (e.g. estimate project costs) to fulfill. The NPD team requires a collaborative knowledge flow system in order to model and to enact their NPD process. Furthermore, it is required to model the organizational structure of the team (persons, roles, skills, tasks etc.). Finally the system should allow integrating NPD artifacts (e.g. source code, management reports etc.) to the appropriate activities the team members are supposed to work on.

## 2.2 Organizational Modeling

In order to manage structural information of the work team, we [4] have implemented a component (Caramba Object Center) managing three categories of objects: Organizational, Dynamic, and Business Objects. The *Organizational Objects* category contains the following objects: Persons, Roles, Groups, Skills, Units, Organization, Tasks, and Document-Templates. The *Dynamic Object* category consists of Processes (i.e. templates consisting of linked activities modeled as directed graphs) and Workcases (enacted Processes). The *Business Objects* are a means of integrating corporate databases into our collaborative knowledge flow system. The underlying metamodel enables modification and customization of the attributes of the objects and their relationships (e.g. a Person may have many Roles).

Figure 1 depicts a graphical view on the Object Center objects with all Skills to be found in the NPD team. Figure 2 depicts a matrix view on the Persons and the Skills to be found in the team and their relationships. Additional links (relationships) can be modeled by double-clicking the appropriate cell. Figure 3 provides an example of our case study, where the object *Person* (in this case Everitt Lynn) is associated with *Tasks* from a modeled process, (Fig. 3, left window) and Links (artifacts associated with the Person, e.g. database objects (bug reports) and other documents (e.g. a Word document)).

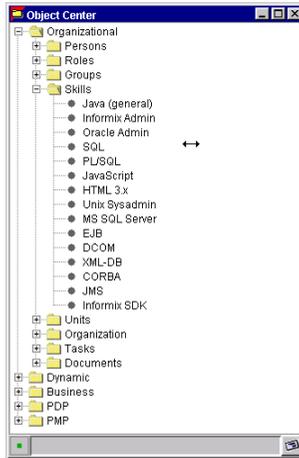


Fig. 1. Object Center: Skills

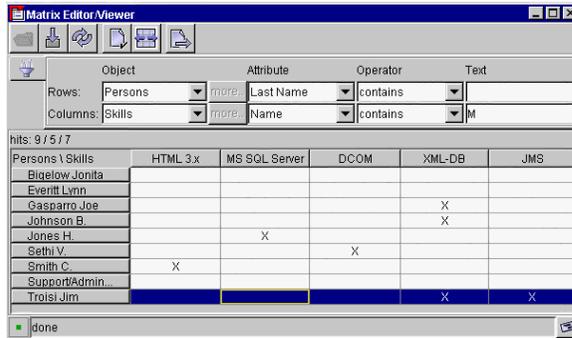


Fig. 2. Matrix Editor: Association of Person and Skills



Fig. 3. Person associations: Tasks and associated artifacts

### 2.3 Processes and Activities

In order to efficiently achieve business goals many organizations model their business processes and enact them. Business Processes can be represented as directed graphs consisting of linked (control flow) activities. An activity constitutes a logical piece of work, which forms a logical step within a process [13]. The literature differentiates between ad-hoc, semi-structured and structured processes [e.g. 8]. In our case study we modeled the NPD process on a high granularity (semi-structured) by modeling six main phases (subprocesses) as depicted in Figure 4. Each activity box in Figure 4 represents a subprocess consisting of other activities modeled as a directed graph. The NPD process shown here is on coarse-grained granularity level and will serve as a template to be enacted. Actors in the process may choose to “coordinate” work item (activities) to other subprocesses or organizational objects (e.g. exceptions).

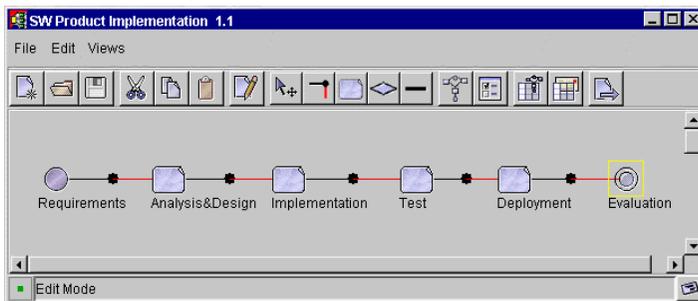


Fig. 4. NPD Process model (build time)

### 2.4 Data Management and Business Objects

Caramba utilizes data managers, which enable organizations to integrate corporate database tables (e.g. bug report tables) into the Caramba collaborative work management system. In order to being able to access those tables, Caramba provides a data manager containing information such as database name, user, etc. for each integrated table. Each data manager may be responsible for many tables or views. Attributes such as PROTOCOL, PORT, DATABASE are utilized to address a connect string for database access using JDBC. For example a connect string might be:

```
jdbc:oracle:thin:@dbserver:1844:Caramba,PROTOCOL=jdbc.ora
cle.thin,SERVER=dbserver, PORT=1844, DATABASE=Caramba
```

Integrated database tables are called Caramba Business Objects (CBOs). Utilizing the Caramba metamodel the administrator may configure attributes of the table to be displayed inside Caramba. CBO-records may be “attached” during the coordination activities of team members, similar to file-attachments. In our case study CBOs are used for tables such as source code, bug reports, management reports, review reports etc.

When a Caramba object is accessed the appropriate Caramba data manager is loaded and access to the database table is provided. Figure 5 shows the database model of Caramba data manager and its relation to the Caramba tables.

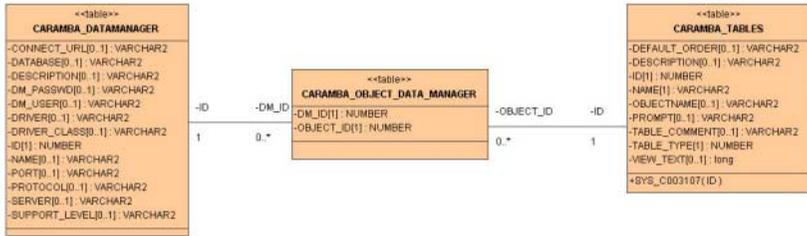


Fig. 5. Caramba Data Manager

### 2.5 Knowledge Flow Management

The main end user component (Caramba Activity Center) provides the management hub for managing personal worklists consisting of *work items*, personal *actions* performed on work items, and personal *scheduling* information on work items. The Activity Center is based on a messaging metaphor and aims at integrating the first three collaborative systems paradigms discussed by Ellis [9]. The Object Center provides the features required by *Keepers*. The Activity Center implements the *Communicator* features, and the *Coordinator* is realized by the Caramba Process Editor.

Caramba users (in our case study software engineers working on joint projects) may access work items being routed to them (coordination) and retrieve a knowledge trail of all “ coordinations ” based on any work item they are associated with. This allows each team member to monitor the progress of the overall project and increases the process-awareness of personal activities. Figure 6 depicts the Caramba Activity Center with windows for personal actions, time/cost related information, personal scheduling information, and a view on the overall progress for a selected work item.

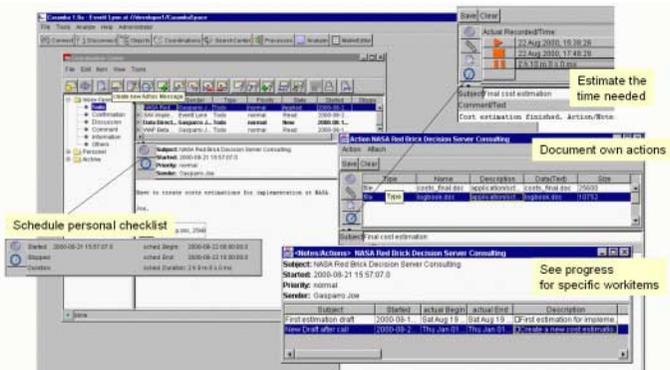


Fig. 6. Caramba Activity Center

### 3 Design and Implementation

The Caramba software architecture [4, 10] is composed of multiple layers: middleware, client suite, and a persistence store. Objects and services are accessed through the Transparent Access Layer (TAL) from the CarambaSpace platform (middleware). Depending on access mechanisms and the requested services (e.g. via Java client with RMI protocol or via Web browser with http), Caramba provides a unique way to handle requests using a metamodel framework to describe content and separating presentation, logic and data. This model permits high flexibility, enables customization, and extensions as well as the adoption of new devices or technologies. The goal of this layer is to offer transparent access to a CarambaSpace. The TAL utilizes various services to transform, describe, manipulate and observe objects. All objects managed through a CarambaSpace are well described using a metamodel description framework. Objects can be customized in their structure (e.g. adding columns to tables, adding relations to objects) and their presentation by adopting their metamodel description. Any changes are dynamically reflected by client components. Based on the metamodel description framework Caramba enables various options to customize data and content and to integrate data from different resources (e.g. corporate databases). This layer also provides facilities for fine-grained object notification services and the implementation of customized services based on object observers. The middleware does not manage states and persistence of objects. Objects are stored, manipulated, and retrieved via the Persistence Layer (PEL). Caramba middleware and client are written in Java and leverage standard Java based technologies (e.g. JDBC, JNDI, HTTP, etc.) and Internet standards (e.g. SMTP) to access and integrate corporate information systems.

### 4 Conclusion

This paper contributes to the goal of building process-aware collaborative work management systems by designing and implementing a collaborative knowledge flow solution. It enables *links* between artifacts, business processes, and resources (e.g. persons, skills). We have presented a case study for motivating the need and have discussed the architecture and some implementation issues of a system (Caramba) aiming at improving process-awareness and traceability of collaborative work activities. Next-generation business solutions clearly have to provide an integrated environment, supporting knowledge workers to monitor and coordinate work activities in the context of the overall business process. Future work includes providing collaborative knowledge flow solutions for web services [1].

## References

1. Baina, K., Dustdar, S. (2002). Web-Services Coordination Model. International Workshop on Cooperative Internet Computing (CIC 2002), in conjunction with VLDB conference, August, Hong Kong, Kluwer Academic Press.
2. Bolcer, G.A. "Magi: An Architecture for mobile and disconnected Workflow", *IEEE Internet Computing*, May and June 2000, pp. 46 – 54.
3. Bussler, C. "Enterprise-wide Workflow Management", *IEEE Concurrency*, 7(3), pp. 32-43.
4. Caramba Labs Software AG (2002) <http://www.CarambaLabs.com>
5. Casati, F. et al., "Developing e-Services for composing e-services", *Proceedings CaiSE 2001*, Computer Science Lecture Notes, Springer Verlag, 2001, pp. 171-186.
6. Chen, Q. et al., "Peer-to-Peer Collaborative Internet Business Servers", HP-Labs Technical Working Paper HPL-2001-14.
7. Craven, N. and Mahling, D.E. "Goals and Processes: A Task Basis for Projects and Workflows", *Proceedings COOCS International Conference*, Milpitas, CA, USA, 1995.
8. Dayal, U. et al., "Business Process Coordination: State of the Art, Trends, and Open Issues", *Proceedings of the 27<sup>th</sup> VLDB Conference, Roma, Italy*, 2001.
9. Ellis, C. (Skip), A Framework and Mathematical Model for Collaboration Technology, in. *Coordination Technology for Collaborative Applications – Organizations, Processes, and Agents* (Conen and Neumann, Eds.), Springer Verlag, 1998, pp.121-144.
10. Hausleitner, A. and Dustdar, S. "Caramba - Ein Java basiertes Multimedia Koordinations-system", In *Erfahrungen mit Java. Projekte aus Industrie und Hochschule*. Silvano Maffei, et al. (Eds.), dPunkt-Verlag, Heidelberg 1999.
11. J. Puustjärvi, and H. Laine, "Supporting cooperative inter-organizational business transactions", *Proceedings DEXA 2001*, Computer Science Lecture Notes, Springer Verlag, 2001, pp. 836-845.
12. Schal, T., *Workflow Management Systems for Process Organizations*. New York: Springer 1996.
13. Workflow Management Coalition (WfMC), *Workflow Management Specification Glossary*, <http://www.wfmc.org>