Coordination Support for Scalable Collaborative Work

Henrique João L. Domingos, J. Legatheaux Martins, Nuno M. Preguiça
{ hj, jalm, jalm}@di.fct.unl.pt

Departamento de Informática
Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa
Telef. +351 1 295 44 64 ext. 0727

Abstract

In this paper we discuss the principles, design options and implementation issues to support coordination and awareness services in the context of scalable CSCW sessions.

We analyze essential differences between the coordination support needed by these sessions when compared with the conventional coordination activity within workflow systems. Then we propose a flexible support model, which is more adequate for group-oriented collaborative work tasks developed in the context of a more complex workgroup activity

The support is concerned with a set of adaptable mechanisms as well as system base components and services provided by an integrated, open and extensible support platform. These components implement a set of facilities for coordination and awareness control in the context of collaboration sessions.
.

Introduction

There are different definitions and characterizations that can be used to refer the coordination control activity. Different formalizations of the term "coordination", directly or indirectly applied to CSCW systems, can be found in different authors [Malone90][Holt88],[Wino86]. In the perspective of this paper we define *coordination* as the support for the activity of managing dependencies and possible conflicts between collaborative entities involved in common and inter-related tasks of a collaborative activity.

These entities are different when regarded at different levels of a computational system. At application-level entities relate users participating in a workgroup and interacting in the context of collaborative work sessions, represented by specific applications. At system-level, entities are associated with the different instances of computational processes running in a shared system-level coordination context implementing the notion of a collaboration workspace. The processes are associated with collaboration-aware applications adopted by the users when they interact with that workspace.

In the CSCW literature, the *coordination support* is referred with different perspectives and focus. In the case of specific computational system supports we can distinguish between two fundamental approaches: the approach of the coordination support provided in the context of conventional workflow systems and the approach related with the coordination issues in group-oriented collaboration-aware applications and systems.

In both cases we recognize the base coordination principles as discussed by Malone and Crowstown [Malone94]. However, the essence, focus and emphasis of coordination control activities in both cases have important contextual differences. These differences relate with different requirements and design principles of the coordination support in the two approaches.

In conventional workflow systems, the different tasks (or processes) of a complex and/or longtime activity are modeled by flows determining in a more or less strict way the coordination methodology to be applied. Each task corresponds to a process scheduled by the workflow. The participants in each process, their roles, the resources to be used, the goals of each task, the scheduling information and the relationships between tasks are pre-defined and rigidly assigned in the context of the workflow description. When each task begins, the above data are well-defined for coordination purposes. Thus, the coordination support can be modeled in terms of a rigid aggregation of well-known (possible repetitive) processes implementing "a priori" well-defined work methodologies.

In the case of collaboration activities, the coordination principles are motivated by dynamic facts or work circumstances, not necessarily known in the beginning of the collaboration. The coordination activity itself is event-

driven by nature. This happens because in despite that in certain moments the collaboration goal can be defined, the methodology that grants the best productivity criteria to achieve that goal is unclear. There is place for informal interactions among the participants. Furthermore, collaborative groups need mechanisms for work-facilitation, awareness support and sharing of viewpoints to promote common backgrounds [Dourish 92]. During a collaborative session, different sub-tasks need to be scheduled in a dynamic way, adapting the work circumstances to the manifestations of possible interdependencies and potential conflicts. In short the coordination support must be adaptable and based on tailorable components and services.

Despite the essential differences, in modern organizational structures we recognize possible scenarios in which the ability to switch easily and dynamically between those different coordination contexts is a very important issue. In fact, different organizations have been spurred to implement and validate new collaborative technologies to be used complementarily with conventional workflow management systems. This situation is imposed by several factors. Rising competiveness, distribution of responsibilities and decisions, fragmented markets, specialization levels emphasis on learner and more entrepreneurial behaviors are certainly among the factors requiring the collaboration of workgroups that require the adoption of those collaborative-oriented technologies.

One of the more challenging trends is the understanding about how collaborative systems will deal with new kinds of peer-coordination models and methodologies that change behavior in ways that raise productivity, creativity, motivation and economic efficiency. While existent groupware provides interesting technical capabilities, it is the ability to transform organizational coordination behaviors, promoting common backgrounds and collaboration aware facilities that must deserve the most relevant attention from the organizational perspective.

This situation motivates the research of coordination control mechanisms promoting points of convergence between conventional workflow coordination control and the coordination needs provided by collaborative-oriented system platforms.

This convergence can be approached in two possible ways. One direction is to analyze the coordination support of future workflow management systems as a group-oriented collaborative activity itself, in a globally, scalable and decentralized perspective. This approach has been developed in recent projects in the field of workflow systems (e.g., [Santanu 97] [Sheth 97]). Another vision (perhaps not so ambitious) is to provide effective interoperability mechanisms to switch between a workflow task and generic coordination support mechanisms managing collaborative sub-tasks in the context of coordinated collaborative workgroup sessions. In this paper we take this second direction.

Background and Motivations

During the last two years, we have been working on the specification, design and implementation of a generic, extensible and integrated platform[1] supporting CSCW/Groupware for large scale distributed settings. The platform is based on a flexible object-oriented JAVA framework. It supports the requirements for the development and operation of groupware in large-scale collaborative work settings. With this framework we have implemented several applications and tools as an initial effort to validate the base support.

The platform is organized as a set of middleware components and base support services, structured as an extensive multi-layered architecture, providing the following functionality:

- **Communication services** - support for group-oriented multi-synchronous communication (providing a flexible support for synchronous as well as asynchronous and disconnected group-oriented interaction modes). Flexibility is achieved by synchronous and asynchronous settings and semantics covered by different protocols.

- **Collaborative object-group replication support** - support for collaboration-oriented workspace services namely: object sharing based on flexible concurrency control models and consistency management. This flexibility provides the complementary reuse of the base object-replication abstractions in the context of peer-to-peer synchronous closely-coupled collaboration models (for synchronous highly-interactive groupware) and asynchronous loosely-coupled collaborative environments (for asynchronous groupware and disconnected collaborative work).

---

[1] The platform called DÁgora has been developed in the computer science department at the FCT-UNL. References about the system and demos are available (http://dagora.di.fct.unl.pt).

- **Data repository service** - a repository of objects, structured as a weakly replicated set of servers, allowing client caching to cope with disconnected operation. The repository is based on a read-any/write-any model of object replication. An object framework allowing flexible type specific update conflict detection and resolution completes the object-storage service.

- **Coordination support** - a set of reusable and tailorable components, anticipating different coordination behaviors and providing facilities for adaptation to specific coordination needs of different collaborative tasks. The coordination support model implements the notion of coordinated collaborative sessions as self-contained coordination units. The users participating in session workgroups use different collaboration-aware tools and applications in a complementary way. The coordination support is also related with the awareness support and interoperable facilities between each session and orthogonal standard services (ex., E-MAIL, WWW and SMS mobile channels).

The different support layers co-exist in an integrated and flexible way, but the framework preserves open and extensible characteristics, maintaining adequate levels of orthogonality. The details of the different support layers above are explained in several publications [Dom 97],[Simão 97],[DAgora 98] [Preguiça 98].

Implementing the coordination support

In this section we propose and discuss the mechanisms and services to support coordination in an integrated way. These mechanisms and services relate with the coordination support level of the DAgora platform. The main issues that we will describe are: (1) the definition of a collaborative-session model, (2) the components and services implementing the system coordination abstractions.

The model for collaborative sessions

Logically, a *session* embraces all tasks, procedures, data and associated users, applications and dependency relations established in the process of reaching a common goal - the session's purpose (or the session collaborative goal). Coordination among users involved in a session is used to guarantee that each participant's efforts are positive contributions to the common goal. Different types of sessions (for instance, with different objectives and different number of participants) require different coordination degrees.

Moreover, in the same session, different subtasks also need different types of coordination. Consequently, flexibility and tailorability of coordination control is required. There are *open sessions* and *closed sessions* depending when the participation is free (users just binds to the session without previous authentication) or when requires the previous registration of participants (users must authenticate when join to the session).

The session represents a *coordination unity* in which all the activities inherent to each collaborative process take place.

In the framework, several properties are associated with this notion of a session: *general information* - comprising a set of attributes such as the session title, description of the session goal and associated notification services; a *workgroup of participants* - including a registry service with individual users information (as names and roles in session's context if there are roles assigned); *tasks and interdependencies* - including the information and bindings related with the tasks that will be performed (dynamically) in the context of the session.

Session's tasks have two different types: *intermediate and final*. Intermediate tasks are defined through the definition of a new *collaborative sub-session* (the session is therefore a recursive concept). Final tasks clearly define their goals and procedures to achieve them. There are *synchronized and non-synchronized* tasks. Associated with each synchronized task it is defined (besides general task description) a set of data: its participants and associated roles if assigned (correspondent to minimal and maximal subsets of sessions participants); scheduling time; applications to be used in each task (if applicable); a reference to data produced (filled in the end of the task, by its coordinator or by some participant). Associated with each non-synchronized task it is defined a deadline for its execution and to obtain the respective data, participants and associated permissions (for manipulating the data).

The *collaborative session manager application* - CSm - controls the coordination information associated with a *collaborative session*, including all tasks executed in its context. Each participant may *jump* to some task through simple interface mechanisms provided by the session manager - applications associated with the task are started and the binding process is automatically executed (if any - used primarily in synchronoized tasks).
Although the same concepts are apparently used in both WFm (workflow management programs) and CSm (collaborative session management programs), fundamental differences exist. A workflow is used as a systematic description of well known (possible repetitive)

processes to reach some goal, while the CSm is used to create and evolve dynamically and collaboratively possible unknown processes to reach the session goal.

For the above reason, flexibility and tailorability are key-properties in the context of the CSm. Not only new tasks can be created within the session, but also new types of tasks. For instance, new synchronized tasks may be defined with new different associated tools. Moreover, sessions coordinators must have the ability to modify dynamically some previously task, define it as concluded, or trace dependencies among them. These activities can be done starting from previous negotiation processes, established by previous tasks.

Although, the process to reach the session goal can be unknown "a priori", some fundamental tasks may have been identified and reused later in the context of the current or new collaborative sessions. Thus, templates for sessions exist, allowing the definition of some initial tasks. However, the overall process is unknown, and must be defined as a result of session interaction - therefore, we are not in presence of a traditional workflow. This ability to evolve session templates for collaborations done with success, reusable in the context of future activities seems to be a fundamental aspect of capturing the memory of collaborative methodologies, contributing for the widely acceptance of CSCW systems in the perspective of organizational structures.

Summarizing, the tailorability support at coordination level relates with the anticipation of coordination behaviors and the functionality aggregated in the CSm. In fact, the CSm acts as a tailoring tool to compose coordination components: the combination (binding facilities) for simple tools (Java classes implementing collaborative tasks) or complex multi-synchronous collaboration-aware applications; facilities to dynamically change the session customizations, dynamic configurations of existent tools and applications; and ways to control session-awareness notifications. All these facilities are provided as end-user level adaptations in the perspective of individual users, workgroups and organizational coordination. The system also provides facilities for reusing coordination contexts applicable to other future similar activities.

Session coordination components and services

The session services implementing the coordination support level are: the session binding service, the user's registration service, the tasks scheduler and its multi-tool binding components and the session notification service.

All these services are accessible from the *CSm* environment conceived an application supported in base components interacting with all the above services.

We will describe briefly each one of those services explaining in more detail the implementation of the awareness support provided by the session notification service.

Session binding. The session binding service is responsible by providing browsing and binding facilities to all the sessions supported in each moment in the data repository. Each session has an unique name (similar to an usual web URL address, ex:
 http://dagora.di.fct.unl.pt/SessionID).
The sessionID is an unique identifier. Each session maps on a volume as explained above when we described the data-storage component. Sessions can be registered in a session name service, with the set of coordination attributes. A user binds to a session just by downloading initially a CSm which provides the adequate functions to access to the coordination context of the session (coordination attributes related with the session model explained above and facilities for all the other session services).

User's registration service. In the case of closed sessions, when the session is created and configured, the users that will participate in the tasks of the session must be registered. This is done by means of the user's registration service. To each user is assigned a session role: coordinator, participant or observer. These roles are used by the session manager to provide access control rights in the context of the session. Coordinators can modify the coordination attributes in the context of a session (modifying for instance the roles assigned to the users) and can also schedule new tasks choosing the adequate tools (using the scheduler component of the CSm). Participants act as users without privileges to modify the coordination context but they can use the different tools related with the tasks scheduled and announced in the session manager. With these tools they collaborate to achieve the different goals subjacent to the scheduled tasks. When using the different tools they can handle the different object-types managed in the application-specific context. Finally, observers only use the session manager to bind to tasks results (they cannot bind to the applications concerned with those different tasks). A result is a static information accessible by a link managed in the context of the session manager and represents a snapshot or annotation of a coordinator user (reporting the state or the result of a previous task).

Tasks scheduling and multi-tool binding. This service provides the way to schedule and announce new tasks in the context of a session. Each announcement has all the necessary binding information to users automatically bind to each task. When a task is created (by an user acting as coordinator) it is chosen the tool that will be used to work cooperatively to develop that task.

Awareness support. The awareness support is provided by a service called Session Events Notification Service. This service provides a generic way to disseminate and notify events occurring in the context of all the tasks developed within a session. We explain the architecture and functionality inherent to this service in the next section.

6. Conclusions

In the paper we analyze the essential differences between the coordination support provided by conventional workflow management systems and the case of the coordination requirements and support for collaborative sessions. The main motivation for this analysis is based on the background and experience in materializing a generic, flexible and integrated CSCW platform - the DAgora system [DAgora 98][Dom 97] - an object-oriented groupware platform implementing a flexible framework for the requirements of groupware running in large scale settings.

In such a platform, the main support components are structured as an extensible middleware architecture providing base abstractions and support services at different levels, namely: group-oriented communication services, collaboration support, data-repository service and the coordination support.

The coordination support relates with the specification of a coordination model for collaborative sessions and a set of coordination components and services organized in the context of a fundamental notion of collaborative session. We describe briefly the coordination model for collaborative sessions that we propose in the DAgora system. Finally we describe some main support components that can be reused by different groupware applications as base system building blocks providing coordination concepts and abstractions.

Bibliography

[DAgora 98]   References and demos about the DÁgora project: http://dagora.di.fct.unl.pt

[Dom 97]   Henrique J. Domingos, J. Legatheaux Martins, Nuno M. Preguiça and Jorge F.Simão, "Support for Coordination and Flexible Sunchronicity in Large Scale CSCW" , in Proc. of the CRIWG'97 - 3rd CYTED RITOS International Workshop on Groupware, pp 81-90, October 1997

[Dourish 92]   Paul Dourish, Victoria Belloti, "Awareness and Coordination in Shared Workspaces", in Proc. of the ACM CSCW 92, pp 107-114, November 1992

[Holt 88]   A. Holt, "Diplans: A New Langiage for the Study and Implementation of Coordination", ACM Transactions on Office Information Systems, n. 6, vol.2, pp 109-125, 1988

[Malone 90]   T. W. Malone, K. Crostown, "What is Coordination Theory and How Can It Helps Design Coooperative Systems", ACM CSCW 90, in Readings in "Groupware and Computer-Supported Cooperative Work", Writen and Edited by Ronald Baecker, pp. 375-388, Morgan Kaufmann Publishers, 1993

[Malone 94]   T. W. Malone, K. Crostown, "The interdisciplinary study of coordination", ACM Computing Surbeys, 26, pp 87-119, Mar 1994

[Preguiça 98]   N. Preguiça, J. Legatheaux Martins, H. Domingos, J. Simão, "System Support for Large-Scale Collaborative Applications" Technical Report, TR-01-98 DI-FCT-UNL, Dep. of Computer Science, FCT - New University of Lisbon

[Santanu 97]   Paul Santana, Edwin Park and Jarir Chaar, "RainMan: A Workflow System for the Internet", in Proc. of the USENIX Symposium on Internet Technologies and Systems, pp. 159-179, December 1997

[Sheth 97]   Amit Shet and K. Kochut, "Workflow Applications to Research Agenda: Scalable and Dynamic Work Coordination and Collaboration Systems", TR

[Simão 97]   Jorge F. Simão, N. Preguiça, J. Legatheaux Martins and Henrique J. Domingos, "DÁgora: An Object-Oriented Groupware Platform", Workshop on Groupware Platforms, ECSCW 97,

[Wino 86]   T. Winograd, F. Flores, "Understanding Computers and Cognition: A New Foundation for Design", Norwood, NJ Ablex, 1986