Integrating Synchronous Multimedia Collaboration into Workflow Management

Michael Weber
Gerhard Partsch
Siegfried Höck
Dept. for Distributed Systems
Oberer Eselsberg
University of Ulm
D-89069 Ulm, Germany
{weber|gpartsch|hoeck}@
informatik.uni-ulm.de

Georg Schneider
German Research Center for
Artificial Intelligence,
Stuhlsatzenhausweg 3
D-66123 Saarbrücken, Germany
georg.schneider@dfki.uni-sb.de

Astrid Scheller-Houy
Jean Schweitzer
Siemens Corporate Research,
Stuhlsatzenhausweg 3
D-66123 Saarbrücken, Germany
{astrid|schweitzer}@dfki.uni-sb.de

ABSTRACT
Two different areas of distributed group work are supported by workflow management systems and real-time collaboration systems. Workflow management systems support work being structured in steps such that each step can be handled with the results of former steps and the expertise of the person working on that step. On the other hand, multimedia collaboration systems are best suited for unstructured group activities. Audiovisual connectivity and shared documents enable flexible group processes. All coordination tasks are left to the conference participants. This paper introduces an integration concept and prototype system which combines the advantages of both types of CSCW-systems.

Keywords
Workflow Management, Teleconferencing, Conference Assistant, Brokerage, Trading

INTRODUCTION AND MOTIVATION

Due to the ongoing globalization many companies are forced to decentralize their organizational structures. This requires an orientation towards business processes derived from an entire view on the enterprise and its relations to the outside world. Optimally tailored working environments are required, which open new application domains for computer-supported cooperative work. One stream being followed is the use of business process engineering succeeded by workflow management to coordinate the processes during their lifetime. A second stream is to provide real-time collaboration tools in order to facilitate conferencing between geographically dispersed teams.

The tasks being assisted by the two approaches diverge significantly. On the one hand a formalized sequence of separate activities is supported. The system coordinates the work of the different workflow participants who work in succession and independent of each other at a point in time. To work this way is also called asynchronous collaboration. On the other hand formally unstructured conferences using appropriate tools are supported. All participants work jointly together at the same time on a task. This style of working is called synchronous collaboration.

Workflow Management Systems

Business processes can be modeled using appropriate tools, e.g. Bonapart [23] or ARIS [17], and subsequently mapped onto workflow management systems [12]. During the flow of business processes the corresponding tasks constitute a workflow which is controlled and coordinated by the workflow management system during run time. When tasks become active a person is assigned to perform the task. This person is provided with computer-based resources, i.e. documents and computer applications, to be able to fulfill the task.

An inherent characteristic of workflows is that flows are defined before they are activated. Workflows can be defined if work can be divided into steps such that each step can be handled with the results of former steps. Workflow management systems also presume that no mistakes have been made earlier in this chain. Unfortunately in daily work there happen many violations on these assumptions.

Multimedia Collaboration Systems

Multimedia collaboration systems (MMC systems) enable users to interact at the same point in time while being spatially dispersed [16, 5]. They provide a facility for audio/video conferencing and add functions to jointly work on shared documents. Thus MMC systems replace face-to-face meeting scenarios through networked computers. Multimedia collaboration systems are mainly used for work which does not follow a routine style of working together

by providing communication channels between persons. They do not have any knowledge about the semantics of the content to be communicated.

Integration of MMC Systems and Workflow Management Systems

An integrated telecooperation platform consisting of MMC and workflow management systems can support problem classes which could not be well supported by each of the isolated systems. Embedding synchronous teamwork as part of the workflow means a change in the workflow paradigm which was limited to "one person - one task - one application" [22]. An activity can incorporate as well more than one person as multiple applications.

In contrast to face to face conferences an MMC system allows to work together in teams on specific topics directly from the desktop without leaving one's place. This reduces media breaks and lengthy interruptions since it allows to contact immediately other workflow participants or even persons outside the workflow management system.

A smooth integration of workflow management and MMC system enables a continuous stream of tasks and activities in which fast, informal, ad-hoc, and direct actions can be taken through conferences within the usual formal workflow. This integration has to go beyond the simple provision of both systems on the same desktop [1]. The systems have to be coupled in order to interoperate between workflow and conferences.

From a technical point of view the integration of MMC systems is rather similar to the interoperability between different workflow management systems. Like in this case it is to deal with the handing-over of documents and the control flow. Furthermore configuration work has to be done, e.g. to provide files in the working directory of the MMC system.

From a conceptual point of view there are two approaches. In the first the MMC system is an enabling technology to impose a virtual conferencing room which eliminates the distance between the participants and makes applications available for a group of users. The "real" activity according to the underlying workflow description is performed by working with these applications. Consequently in a conference a team works on multiple tasks with different tools. To accomplish this at least two systems are in use, i.e. the MMC system plus at least one shared application. This is usually not supported by current workflow management systems. The other approach is to regard an MMC system as a system which coordinates activities. In this case semantics have to be transferred. The workflow management system has to hand over activities and the control over these activities. For this purpose suitable mechanisms and tools have to be chosen.

Related Work

The Workflow Management Coalition [24] is working towards the interoperability between workflow systems. Interworking of workflow with other forms of collaborative work is not their focus.

Several virtual office environments offer asynchronous and synchronous tools [21, 11]. However, these projects do not involve workflow management, but rather focus on integrating e-mail and collaborative editing tools with audio/video conferencing.

Some systems deal with other CSCW applications in the context of workflow management. Planko [20] presents an event based approach to integrate asynchronous CSCW-applications into workflow management systems. This project focuses on the conference coordination and the development of a homogeneous platform incorporating different cooperation services. Contact [9] supports the project management of reengineering projects. The coordination during this process is the center of interest. Speech acts are used to coordinate the tasks. WAM [14] presents a combined workflow management - MMC application. This system uses a special document organization and a flexible modeling technique based on petri nets.

EMBEDDING MMC CONFERENCES IN WORKFLOWS

Since workflows have a longer duration than conferences, we propose to model conferences into workflows. This means, that synchronous activities involving participants simultaneously have to be incorporated into the formal definitions of an asynchronous workflow at appropriate positions.

Conference Dimensions

In our model we distinguish between process activity and conference activity. This formalism will be used to characterize MMC conferences with regard to their integration into workflow management systems. The term process activity has been defined by the Workflow Management Coalition [24] as follows:

> *"A process activity is a logical step or description of a piece of work that contributes towards the accomplishment of a process. A process activity may be a manual process activity and/or an automated workflow process activity."*

In contrast to this we call an activity which takes place during an MMC conference a conference activity. Here several persons are working simultaneously. If conference and process activities have a 1:1 relationship consequentially the conference's organization is structured because a workflow determines its proceeding (fig. 1, left side). In this case the purpose of an MMC system is only to eliminate the "one person" constraint.

Considering a 1:n relationship a conference looks like a single process activity during which (invisible for the workflow) many conference activities will be accomplished (fig. 1, right side). Here one process activity initiates multiple conference activities. In this case the control of the process activity has to be passed to the MMC system. The relatively inflexible coordination mechanism of a workflow

management system can be replaced by a more flexible mechanism for conferences. We define the term conference activity as follows:

"A conference activity is a logical step or a description of a piece of work that contributes towards the accomplishment of a conference. A conference activity is itself a process activity or it is aggregated with other conference activities to form a process activity".

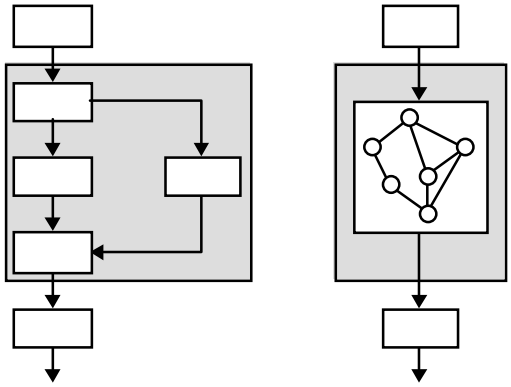

fig. 1: 1:1 versus 1:n relationship of process and conference activities

The above considerations help us to develop a classification of conferences around the dimensions "modeling time" and "conference coordination".

Modeling Time

We distinct two instants in time when a conference is being modeled. Conferences that are already planned at the time of the development of the workflow are called **pre-scheduled conferences**. In contrast to these an **ad-hoc conference** is not foreseeable at the time when the workflow model is specified. The initiation of such a conference depends on the current situation.

Conference Coordination

In a **structured conference** all conference activities are fixed. Within the conference conference and process activities have a 1:1 relationship. The conference activities are fully controllable by the workflow system.

In **unstructured conferences** the control of task execution is passed to the MMC system. The entire conference is regarded as one process activity comprising several conference activities (1:n relationship). The workflow management system cannot influence the course of the conference. It only knows that there is a conference. The participants are free to organize their work themselves. The restrictive conventional workflow mechanism is weakened to support teamwork in a better way. Since conference participants want to react flexible to unexpected situations, their agenda is considered as a dynamic list of conference activities.

Conference Profiles

The space of conference dimensions lead to conference profiles. Combinations of the profiles being described in the following are possible.

Unstructured Pre-scheduled Conference

This conference as such is already part of the workflow at modeling time. The conference activities however will be fixed before or even during the conference (fig. 2, upper left corner).

An example of this type would be a petition arriving at a ministry. In the conference it is discussed which subordinate offices are responsible to work on this petition. After this a suitable subworkflow will be initiated. Because the conference has a brainstorming character a strict ordering of conference activities is not possible. It might often happen that members of subordinate offices are consulted. This procedure can barely be captured in a workflow.

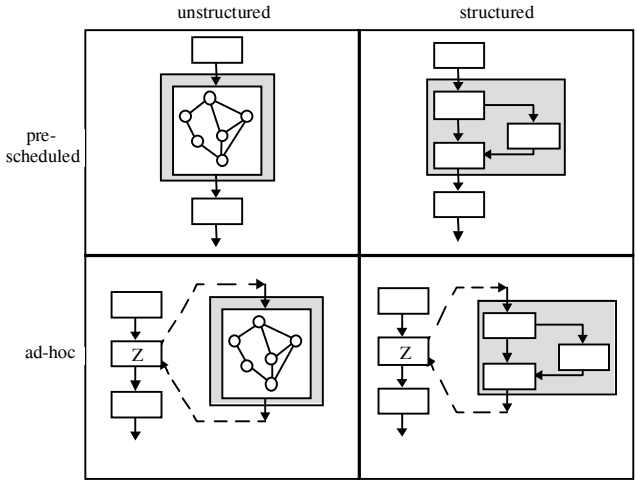

fig. 2: Conference Profiles

Structured Pre-scheduled Conference

Structured pre-scheduled conferences are modeled as part of the workflow (fig. 2, upper right corner). They are workflows themselves. The workflow management system only uses an MMC system as a special resource to support this subworkflow. The course of the conference is coordinated and controlled by the workflow management system.

Unstructured Ad-hoc Conference

Ad-hoc conferences are not foreseen and therefore cannot be part of the workflow from its beginning (fig. 2, lower left). Situations in which this type of conference is required are e.g. a help desk or a check-back with a manager. In such cases a conference will be invoked while the process activity "z" is suspended. After finishing all conference activities the workflow will resume with activity "z".

Structured Ad-hoc Conference

Structured ad-hoc conferences are also not part of the workflow at modeling time. However, they describe a standard situation. When a conference of this type is

invoked the participants have to follow a specific procedure.

An example is a compensation where information already left the scope of the workflow management system [13]. Faulty information arrives, for instance, at a participant due to a mistake done earlier by a colleague. When she wants to correct the mistake she will not have the authorization to do so. In this case she interrupts her task (upper task "z", lower right corner, fig. 2) and starts a conference with the earlier participant. Together the colleagues correct the mistake. Thereafter task "z" is resumed. Such a procedure can be structured as a workflow. When this situation arises the participant will select the suitable (conference-) workflow to invoke and control the conference.

Integration concept

The described conference profiles define the requirements for an integration. Despite the profile being envisaged the same integration phases can be derived (figure 3). In the following the six phases are being described in more detail.

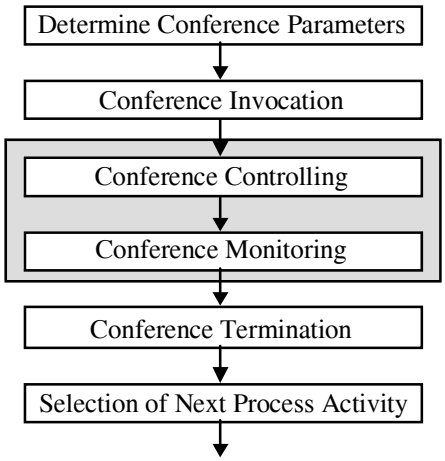

fig. 3: Integration phases of a workflow embedded conference

Phase 1: Determine Conference Parameters

In Phase 1 we have to look at the interface parameters required to start up conferences. Inspecting various MMC systems [4, 6, 10] revealed similar interfaces to operate such systems. The conference parameters mainly contain data which is necessary to establish a conference.

- Conference Title
- Conference Participants
- Conference Leader
- Conference Topic
- Conference Attachments:
  Attachments are documents and applications being used in the conference. Attachments can be known at the beginning of the conference and thus prepared by the workflow management system, or they can be brought into the conference by any participant during run time.

Additional parameters should also pass task information from the workflow to the MMC system.

- Conference Date:
  For ad-hoc conferences this date is usually "now". None of the analyzed MMC systems allows to specify dates for conferences. However, conferences started from a workflow management system should be scheduled such that the mandatory participants are available.
- Conference Agenda:
  The conference agenda could be a document capturing the conference activities.
- Conference Activity Responsible:
  Equivalent to the responsible for a process activity there should be a responsible for each conference activity.

Phase 2: Invoke MMC system

The parameters are used to establish the conference. The system invites the participants and shares the required applications and documents. After this the participants can jointly work on the conference activities.

Phase 3: Conference Controlling

Conference controlling depends on whether the conference type is structured or unstructured. Structured conferences are controlled by the workflow. If a conference activity is completed the next conference activity will be sequenced automatically. Unstructured conferences are controlled by the participants. A conference assistant supports such conferences (see section 2.4).

Phase 4: Conference Monitoring

Controlling the progress of conferences is essential for the integration of conference results into a running workflow. This requires a monitoring function during the conference to record performed activities. Before a conference is closed this record can be evaluated to deduce the conference results. We foresee four levels of monitoring: no monitoring, manual monitoring, semi-automatic monitoring, and automatic monitoring.

With no monitoring neither the course of the conference nor its results are recorded. The workflow management system is not aware of any modifications to documents during the conference. The workflow is resumed independent of the conference activities.

With manual monitoring the conference participants keep minutes of their activities. Evaluation and determination of a follow-up workflow activity is done by them. The conference assistant may support this.

With semi-automatic monitoring the recording process is performed manually but the structure of the conference is predefined. Thus the workflow management system is able to evaluate the results after the conference has been closed.

Automatic monitoring requires a structured predefined list of conference activities in order to coordinate and monitor the entire conference by the workflow management system.

Phase 5: Conference Termination

When the last conference activity is finished the workflow will terminate the conference for structured conferences.

Unstructured conferences are terminated manually by the participants.

Phase 6: Selection of Next Task

Choosing the next task after a conference depends on its result. If all conference activities are completed and results are recorded a decision can be taken. For structured conferences this can be done by the workflow management system. For unstructured conferences the results must be delivered to the workflow management system in an adequate way. For this purpose we also use the conference assistant.

Conference Assistant

The conference assistant. consists of a checklist and an evaluating mechanism. The checklist is an additional document to be shared during a conference. In structured conferences the checklist only serves as an information source to the conference participants, since all conference activities are process activities.

Unstructured conferences need the checklist (figure 4) for conference coordination and monitoring. It describes the conference activities like a to-do list, which can be dynamically updated. Furthermore the checklist is a tool to record the conference results. Thus the checklist is a condensed form of conference minutes to derive a comprehensive conference result. To specify a result we distinguish three completion states of a conference activity: "done", "partially done", and "to do". The latter two cases may happen when not all participants required for an activity were available for the conference, or, when the activity could not be completed due to lack of knowledge or consensus.

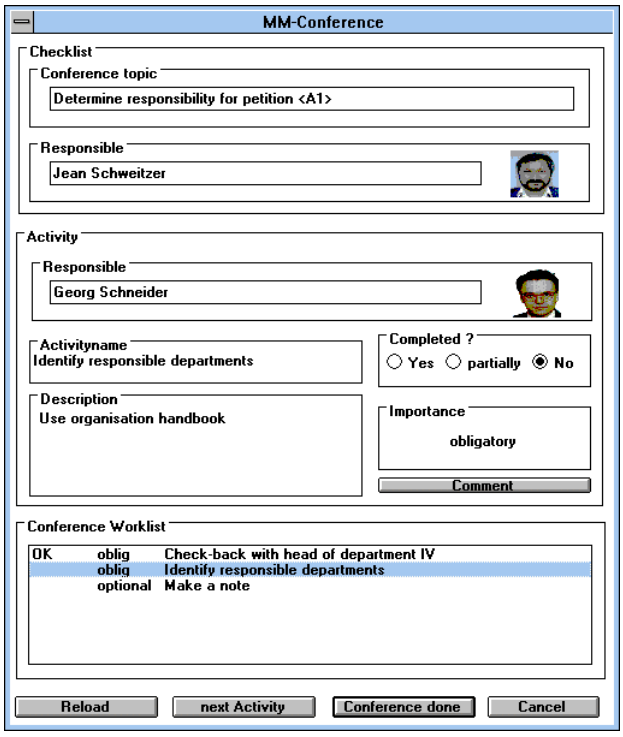

fig. 4: A Conference Assistant's Checklist

Furthermore priorities are assigned to conference activities to support the decision on the conference result. The priorities are: "optional", "deferrable", and "mandatory". Deferrable activities have to be worked on but not completed. They can be resumed in a future conference or as a process activity in the workflow.

Creation of Checklists

Checklists are created depending on the conference profile being used. For all profiles people responsible for the checklist can use a checklist editor. For a structured conference, model checklists can be derived from the workflow automatically by traversing the workflow model. For each encountered conference activity the branches in the workflow graph are inspected for successive conference activities. If found, the list of participants of the two activities are compared. If the sets of participants are disjunct, disjunct conferences have been found and two checklists are created. If the sets are overlapping, the two activities can be performed in one conference and a further activity is added to the already created checklist.

In unstructured conferences the checklist can be extended by additional activities during the conference using the checklist editor.

The responsibility to maintain or evaluate the checklist depends on the type of monitoring being used. The checklist itself can be visible to all participants or it can be a hidden checklist only visible to e.g. the chairperson.

Resuming the Workflow

The final conference result is derived from the different activity completion states. We define three possible results:

- Successful:
  All deferrable and all mandatory conference activities are marked with "done" in the checklist. The workflow can proceed as planned.
- Partially Successful:
  All mandatory activities are marked "done" and at least one deferrable activity is marked "partially done". A further analysis is required to determine how the workflow should proceed and whether an additional conference to finish the remaining activities should be scheduled.
- Not Successful:
  None of the activities is marked "done". The conference should be held again and augmented so that it can be successful. Otherwise the success of the entire workflow is questionable.

THE WOTEL PROTOTYPE SYSTEM

In our prototype system WoTel (Workflow and Telecooperation) neither a workflow management system nor a MMC system are modified. A separate conference broker serves as a mediator between the two systems. The broker takes conference descriptions from the workflow management system and conveys the parameters to the

MMC system (figure 5). The conference descriptions are assembled in an incremental way. If conferences follow each other, the broker extracts the difference of the two and forwards only changes in the parameters to the MMC system. This method allows dynamic adaptation to the set of participants or to conference documents while pertaining the session over various activities. Without this a conference would be closed and potentially reopened again with only slightly modified parameters. We call this method Δ-conferencing.

Since the broker initiates conferences, it also has to be notified about conference states. Thus, the broker takes active part in conferences as a virtual participant. This virtual participant is visible to the human participants to keep them aware of being in a workflow embedded conference.

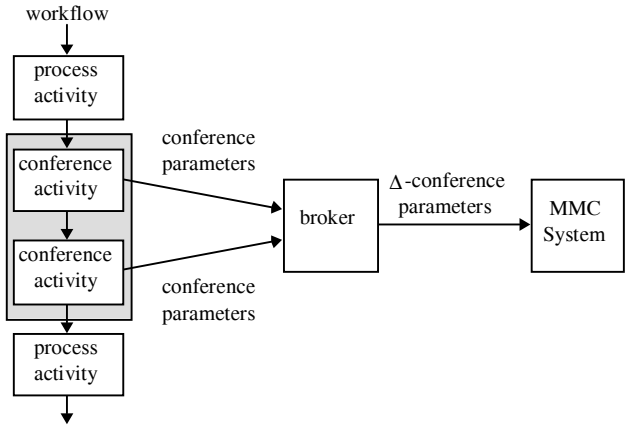

fig. 5: Conference Broker

The simplest way to implement the conference broker is through a central component. However, to support heterogeneous distributed systems our conference broker has been divided into a workflow module and a conference module (fig. 6). The conference module is specific to each MMC system, because it has to support the respective system calls. The workflow module is generic to different workflow management systems due to the ability of exchanging required information via simple protocols.

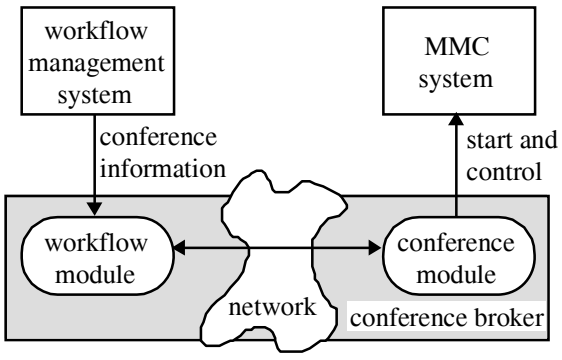

fig. 6: Distributed Conference Broker

Initiating Conferences

The conference information, i.e. the conference parameters, being collected in phase 1 have to be conveyed to the workflow module of the broker, which in turn should initiate the conference using its conference module.

We use the knowledge query and manipulation language KQML [8] being developed for agent communication to transfer the respective data. The workflow management system subscribes at the broker to the newest information on the set of conference participants and on the conference results. If intermediate conference results arrive the workflow management system can monitor the course of the conference and might interact when structured conferences are performed. The following KQML performative shows how conference information is "told" to the broker. The parameters of `content` reflect the conference parameters. `Language` and `ontology` specify the syntactical scope being used. `Reply-with`, `sender` and `receiver` are used for communication sequencing and addressing.

```
(tell
   :content (
      (title test-conf)
      (participants
         (Michael Sigi Gerhard))
      (chair Michael)
      (agenda test.doc)
      (conference-system GroupX)
      (media (sharing audio video)))
   :language     WOTEL-Language
   :ontology     WOTEL
   :reply-with   id1
   :sender       WFMS
   :receiver     WFMS_Module )
```

Then the workflow management system subscribes to actual information such as changes in participants chair media or conference-result.

```
(subscribe
   :content (
      (participants chair media
          conference-result)  )
   :language     WOTEL-Language
   :ontology     WOTEL
   :reply-with   id1
   :sender       WFMS
   :receiver     WFMS_Module)
```

With the above information the broker is able to initiate a conference using its conference module. Despite the separation and distribution of these broker modules, we still have a fixed coupling of a workflow management system and an MMC system. However, since many new MMC systems appear on the market, it should be possible to have a dynamic choice of which system to use during run time. The trading concepts as defined in [3, 7, 15] promise to fulfill these requirements. A trader comprises an exporter part representing the service and an importer part requesting a service. The trader is able to perform a late

(i.e. run time) selection and binding of the respective interfaces.

Trading of MMC Systems

We include importer functionality into the workflow module and exporter functionality into the conference module (figure 7). The trader is then able to mediate between the exported conference service offers and the importer's service request. The trader itself is implemented as a distributed service with peer traders distributed over the internet at participant domains.

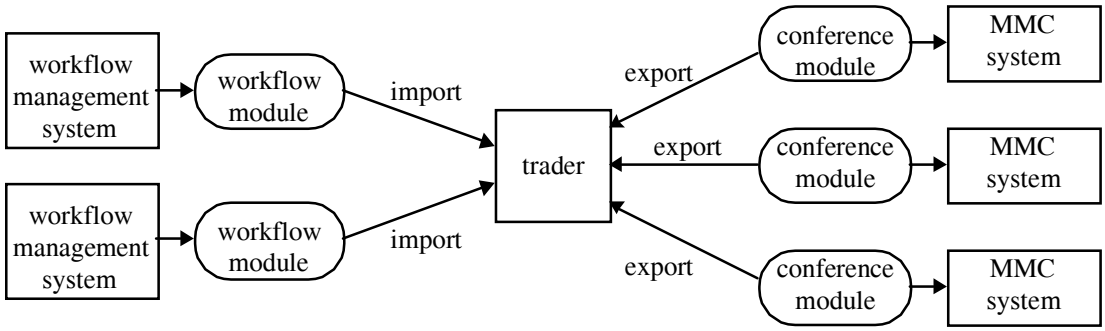

fig. 7: Trader and Broker Modules

Workflow Module

The workflow module imports the service `conference` or any of its subtypes after the workflow management system passed over KQML performatives containing information about the desired conference. This information describes the users to be invited, the documents to be shared, and audio, video requirements and other general conference parameters. The workflow module then contacts the trader core to receive binding information on the desired conference module. This method of flexible matching of MMC systems has several advantages.

First, it is possible to see by the availability of a service offer whether and which MMC system of a person is up and running. Second, it is possible to specify a person by its role and the matching of the trader resolves the role to a real person. Third, it is possible to choose the most suited MMC system for the current set of participants.

Conference Module

The conference module acts as an exporter. It knows its MMC system type and the system's parameters. Starting the MMC system the conference module will be initialized and the service offer will be exported to the trader. For efficiency reasons the service property `users` is a dynamic parameter which will be retrieved at the time of an import by a workflow module. The conference module also measures the actual bandwidth and returns the currently possible quality of service. When the MMC system is shutdown the conference module withdraws the service offer.

Service Types for MMC Systems

In order to achieve flexible trading in heterogeneous distributed systems we need an appropriate conference service type. This must contain all possible properties which MMC systems provide. To avoid the problem of a large and non-manageable service type subtyping is used. The main service type is called `conference`:

```
TYPE conference BEGIN
   conferencing-system :
        conferencing-system-type;
   users/d : list of user-type;
   annotation : string;
ENDTYPE
```

It has the service properties, `conferencing-system`, `users/d` and `annotation`. The property `conferencing-system` is used to store the type of the exporting MMC system. The property `users/d` holds the list of users of the MMC system. The property `annotation` holds a textual description. It is not useful to hold all user entries in the repository of the trader although it is necessary for an importer to know the available users. Therefore this parameter is a dynamic property ("/d"). Normally a user has a role within his or her company. With the following definition of `user-type`, it is possible to search a user by name or by role.

```
TYPE user-type BEGIN
   username : string;
   userrole : list of role-type;
ENDTYPE
```

The next step is to extend the capabilities of the service type `conference` by creating the subtype `tele-conference`. This type provides the audio, video and sharing properties of the MMC system. For example the audio capabilities of an MMC system are represented as a list of supported audio formats. The actual quality of the audio or video connection is catched by dynamic properties. A definition of the type `teleconference` is:

```
TYPE teleconference
   SUBTYPE OF conference BEGIN
      audio: list of audio-type;
      video: list of video-type;
      application-sharing:
           list of sharing-protocol;
      quality-of-audio/d: quality-type;
      quality-of-video/d: quality-type;
ENDTYPE
```

The above service types can be further refined to accommodate additional or future properties of MMC systems.

APPLICATION EXAMPLE

The following example illustrates the benefits of an integrated MMC conference, the procedure and the application of the concepts described so far.

The workflow in figure 8 describes the process to order a piece of hardware. An employee fills in an electronic form which is forwarded to a technical controller. If from a technical point of view the order is correct a procedure for financial evaluation gets selected depending on the cost of the ordered material.

This test guarantees that the available budget will not be overdrawn. The responsible manager has to agree in such a case. If everything is correct, the order will be executed. If the order is rejected, the applicant has the possibility to make changes or to correct errors in the form.

In figure 8 the tasks where conferences are intended to take place are marked black. In the following we focus on the task "financial control with check-back" also shown in figure 9 as a sub-workflow without using a conference. Obviously the workflow is clearer, shorter and faster when an MMC conference is used. This MMC conference could be regarded as a structured pre-scheduled conference.

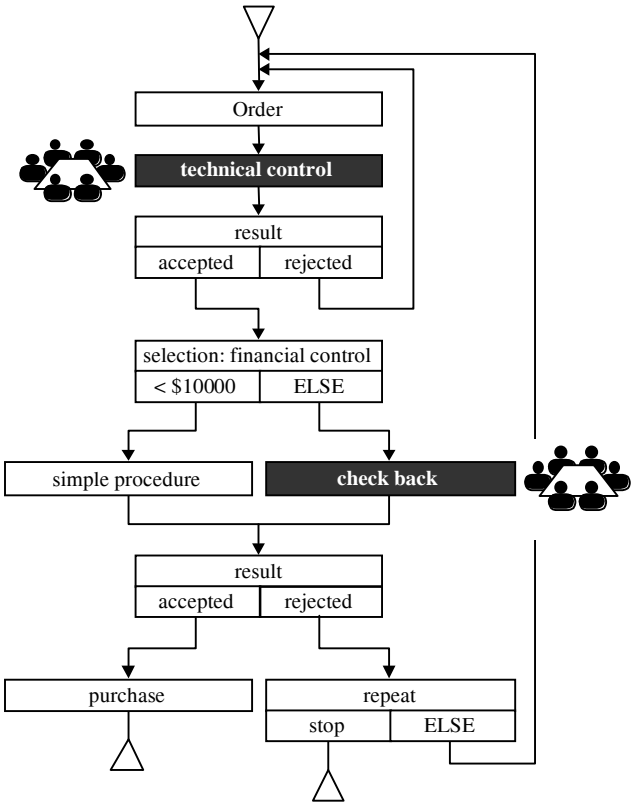

fig. 8: Sample workflow with integrated MMC conference

The above task could also be performed as an unstructured pre-scheduled conference. At the point where the next step will be a "financial control with check-back" the workflow is forwarded to the financial controller to prepare the conference.

Her duty is to determine the conference parameters as far as they are not already determined at the modeling time of the workflow. The conference title would be "financial control with check-back order #xyz". The participants are the financial controller and the responsible manager. The conference topic will be a brief description about the ordered hardware and its costs. The order form to be discussed will serve as a conference attachment. The conference date can be either "immediately" or an appointment between the conference participants has to be made.

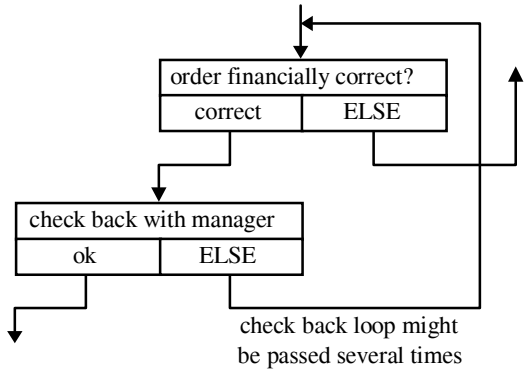

fig. 9: Subworkflow "financial control with check-back" without using an MMC conference

For structured conferences the conference agenda can be generated automatically parsing the specific "conference part" of the workflow (which will look similar to fig. 9). As it concerns now an unstructured pre-scheduled conferences the agenda was already prepared at modeling time capturing the tasks "financial control" and "authorization from manager".

At the specified date the MMC system is invoked using these conference parameters. to establish a conference between the financial controller and the responsible manager. The conference assistant's checklist and the order form are added as shared documents. Furthermore the checklist is used in the conference to capture the results of the conference activities. A result can be a simple o.k. or a modification request. Using these results a conference evaluation can automatically be executed and the workflow will continue. The next task can be either the execution of the order, or the order will be returned to the applicant for changes.

CONCLUSION AND FUTURE WORK

The WoTel approach to integrating workflow management systems with real-time conference support has been introduced. A set of conference profiles has been used to incorporate synchronous collaboration into asynchronous workflows. Coordination of conferences is supported through checklists provided by a conference assistant. A broker service is used to mediate between workflow and MMC tools. Our current prototype links the workflow management systems WorkParty, LinkWorks and FlowMark with the MMC tools GroupX, ProShare and Netmeeting. It has been demonstrated and evaluated using real life scenarios taken from a civil construction authority. The workflows cover several authority locations and departments.

In the future we want to support MMC conferences further by using more knowledge available from the workflow model or from the workflow execution. The conference preparation is a crucial point for the success of a MMC conference [2]. Schneider [19] points out that a comprehensive instruction improves the result of a group activity. For this reason we require at least an agenda for all conferences profiles. We will also include short abstracts about the work to do, so the participants will join the conference well prepared.

Another topic is resource planning for MMC conferences. Here resources are human resources and time. A schedule must be compiled to fix the time of the conference to assure that all participants are available. Since the participants of a conference are specified by their role and not by their name, not only time is a variable here.

Furthermore there will be an integration of MMC systems also into heterogeneous workflow environments where an interface between the workflow management systems has to transport not only workflow relevant data but conference relevant data too [18]. This allows to get information about workflows over company borders. Finally a distributed version will be evaluated to draw conclusions on scalability, performance and failure resilience.

REFERENCES

1. K. R. Abbott, S. K. Sarin: Experiences with Workflow Management: Issues for the Next Generation. *Conference on Computer-supported Cooperative Work*, 1994.
2. V. Barent, H. Krcmar, H. Lewe, G. Schwabe: Improving Continuous Improvement with CATeam: Lessons from a longitudinal case study. *28th Hawaii International Conference on System Sciences*, 1995.
3. M. Bearman: Trading in Open Distributed Environments. *Proceedings of the International Conference on Open Distributed Processing*, 1995.
4. S. Cronjäger, W. Reinhard, J. Schweitzer: Functional Components for Multimedia Services. *International Conference on Communications*, 1993.
5. T. Crowley, P. Milazzo, E. Baker, H. Forsdick, R. Tomlinson: MMConf: An Infrastructure for Building Shared Multimedia Applications. *Conference on Computer-supported Cooperative Work*, 1990.
6. G. Dermler, T. Gutekunst, B. Plattner, E. Ostrowski, F. Ruge, M. Weber: Constructing a Distributed Multimedia Joint Viewing and Tele-Operation Service for Heterogeneous Workstation Environments. *4th Workshop on Future Trends in Distributed Computing Systems*, 1993.
7. J.-P. Deschreval: The ANSA Model for Trading and Federation. Architecture Report of ANSA Phase III, APM.1005.01, 1993.
8. T. Finin, J. Weber, et al.: Specification of the KQML Agent-Communication Language. DARPA Knowledge Sharing Initiative, 1993.
9. J. Galler, J. Hagemeyer, A-W. Scheer: ContAct: Ein Koordinationssystem für verteilte Modellierungsaktivitäten. in: W. Augsburger, H. Ludwig, K. Schwab (eds). *Koordinationsmethoden und -werkzeuge bei der*

computergestützten kooperativen Arbeit. Bamberger Beiträge zur Wirtschaftsinformatik Nr. 30/1995, 1995.

10. Intel: Intel Proshare Developers Kit. 1995.
11. H. Ishii, M. Kobayashi, J. Grudin: Integration of Interpersonal Space and Shared Workspace: Clearboard Design and Experiments. *ACM Transactions on Office Information Systems*, October 1993.
12. S. Jablonski: MOBILE: A Modular Workflow Model and Architecture. *4th International Conference on Dynamic Modelling and Information Systems*, 1994.
13. F. Leymann: The Workflow-Based/Application Paradigm. in: J. Becker, M. Roseman (eds). *Arbeitsbericht 47 Workflowmanagement - State-of-the-Art aus Sicht von Theorie und Praxis,* Universität Münster, 1996.
14. B. Messer, G. Faustmann: Effiziente Videokonferenzen durch Workflow-Management-Systeme. in: K. Hammer, D. Schmolke, F. Stuchlik (eds). *Synergie durch Netze*, Otto-von-Guericke-Universität Magdeburg, 1995.
15. ODP Trading Function. Final Draft - ISO/IEC DIS 13235, ITU-T Rec.9tr, ISO/IEC JTC1/SC21, 1995.
16. W. Reinhard, J. Schweitzer, G. Vˆlksen, M. Weber: CSCW Tools: Concepts and Architectures. *IEEE Computer*, May 1994.
17. A.-W. Scheer: Wirtschaftsinformatik - Referenz-modelle für industrielle Geschäftsprozesse. Springer, 6. Aufl., 1995.
18. G. Schneider, H. Maus, C. Dietel, A. Scheller-Houy, J. Schweitzer: Concepts for a flexibilisation of workflow management systems with respect to task adaptable solutions. *AAAI Workshop: AI in business: AI in Electronic Commerce and Reengineering*, 1996.
19. H. D. Schneider: Kleingruppenforschung. In: Studienskripten zur Soziologie, Teubner, Stuttgart, 1985.
20. K. Schwab, H. Ludwig: Ein ereignisbasierter Ansatz zur Integration von Workflow-Management-Systemen mit Groupware-Werkzeugen. in: W. Augsburger, H. Ludwig, K. Schwab (eds). *Koordinationsmethoden und -werkzeuge bei der computergestützten kooperativen Arbeit.* Bamberger Beiträge zur Wirtschaftsinformatik Nr. 30/1995, 1995.
21. M. Sohlenkamp, G. Chwelos: Integrating Communication, Cooperation, and Awareness: The DIVA Virtual Office Environment. *Conference on Computer-supported Cooperative Work*, 1994.
22. H. Stark, L. Lachal: Ovum Evaluates Workflow. Ovum Ltd, London, 1995.
23. UBIS GmbH: Bonapart, Version 1.0, User Manual. Berlin, 1992.
24. Workflow Management Coalition: Reference Model. http://www.aiai.ed.ac.uk/WfMC/, 1994.