The GroupFlow System:  
A Scalable Approach to Workflow Management between Cooperation and Automation  

Ludwig Nastansky - Wolfgang Hilpert  
University of Paderborn

CSDS Project Group (Client Server Distributed Systems)  
The project scope of CSDS encompasses; Research and Prototyping of Business- und Technology Frameworks usable for Office Systems, Workgroup Computing, Workflow Management, Project Management and Connectivity Frameworks in Client-Server architectures  
Paderborn, February 1994
The GroupFlow System:
A Scalable Approach to Based Workflow Management between Cooperation and Automation

Keywords
Workflow management, workgroup computing, business process design, open vs. structured processes, automation vs. cooperation, workflow continuum, graphically workflow modeling, change management, groupware, information management, Lotus Notes

Abstract
In this paper, we will discuss business relevance factors, architectural concepts, tool approaches, and user-interface samples of the GroupFlow environment. GroupFlow offers business process and technology frameworks to set up versatile and flexible workflow systems for distributed information management within organizations and their outside communication partners. We regard the synergetic approaches being used in the GroupFlow architecture, design concepts, and practical implementation - for the frontend client-workplaces of the several user typologies as well as for the distributed backend server components - as innovative. The GroupFlow environment perhaps best can be profiled around integrating concepts that are typically referred to as workgoup computing or Groupware on the one hand, and workflow management or business process design on the other hand.

GroupFlow has been implemented using Lotus Notes as the basic development platform and underlying distributed architecture. The user interfaces on the client sides are either based on Notes-native form and view concepts, or developed using several other graphical frontend tools when appropriate for the respective user tasks to be performed. On the backend server side of GroupFlow, solely Notes technology has been used for data repositories of the actual business information content, for the workflow structure parts, and the various workflow runtime engines supporting processes like messaging, replication, event management or gateway connections. Due to the open architecture of Notes, the backend server side has to be considered completely open for realtime interconnection of data sets being managed in transaction systems, or external processes to be initiated and controlled in an open client server environment.

Table of Contents
1. Introduction ........................................................................................................................................... 1
2. Workflow Management: Automation vs. Cooperation ........................................................................ 2
   (1) Ad hoc Workflows .......................................................................................................................... 4
   (2) Team task: Self-Managing Team Controls Processes .................................................................... 4
   (3) Semi-structured Workflows ......................................................................................................... 5
   (3a) Open Team Task Within Standard Workflows ......................................................................... 5
   (3b) Controlled Team Task Within Standard Workflow ........................................................................ 7
   (3c) Ad hoc Modification and Exception Handling of Predetermined Workflows ............................ 7
   (4) Predetermined Workflows ........................................................................................................... 9
4. Workflow Modeling .............................................................................................................................. 9
5. References and Acknowledgments ...................................................................................................... 12
The GroupFlow System: A Scalable Approach to Workflow Management between Cooperation and Automation
Ludwig Nastansky - Wolfgang Hilpert
University of Paderborn

1. Introduction

In this paper, we will discuss business relevance factors, architectural concepts, tool approaches, and user-interface samples of the GroupFlow environment. GroupFlow offers business process and technology frameworks to set up versatile and flexible workflow systems for distributed information management within organizations and their outside communication partners. We regard the synergetic approaches being used in the GroupFlow architecture, design concepts, and practical implementation - for the frontend client-workplaces of the several user typologies as well as for the distributed backend server components - as innovative. The GroupFlow environment perhaps best can be profiled around integrating concepts that are typically referred to as workgoup computing or Groupware on the one hand, and workflow management or business process design on the other hand.

In the context of this paper, we will focus on the following layers of the whole GroupFlow system:

- The appropriate business process paradigm underlying the actual design and deployment of workflow systems for business and public organization: GroupFlow is modeled around a continuous scale between cooperation and automation (chapter 2).
- The architecture of GroupFlow making various classes of workflows on a continuous scale between flexibility and rigid predefined structures accessible for efficient rapid modeling of real life workflows in organizations (chapter 3).
- Some genuine aspects of the GroupFlow modeler WOMED, like seamless integration of a graphical design frontend in a distributed operative environment, or the clustering approach allowing layered modeling from the business process level down to the workflow runtime parameter settings (chapter 4).

Many more sides of the whole GroupFlow framework, like analyzing tools, graphical tools for dynamic classification management of flowing job folders, the complete runtime environment, the distributed data structures, or the distributed process and actor architecture are left out here. Architectural aspects like these are being discussed in other papers [Nastansky/Hilpert 1993; Hilpert 1993].

The research and project work described here has been performed with respect to process modeling, system architecture design, and user interaction considerations that are drawn along the lines of studies such as published in Davenport [1993], Hammer/Champy [1993], Ishii/Ohkubo [1991], Marshak [1992], Medina-Mora/Winograd/Flores/Flores [1992]. Basic results of these works are reflected in this paper only to the extent of their impact on designing a pragmatic workflow environment being applicable for deployment within organizations.

GroupFlow has been implemented using Lotus Notes as the basic development platform and underlying distributed architecture. The user interfaces on the client sides are either based on Notes-native FORM and VIEW concepts, or developed using several other graphical frontend tools when appropriate for the respective user tasks to be performed. On the backend server side of GroupFlow, solely Notes technology has been used for data repositories of the actual business information content, for the workflow structure parts, and the various workflow runtime engines supporting processes like messaging, replication, event management or gateway connections. Due to the open architecture of Notes, the backend server side of GroupFlow is to be considered completely open. This inter-connectivity extends from real-time two-way linking of data sets being managed in ‘legacy’ transaction systems, to external processes to be initiated and controlled by GroupFlow around a variety of multi-vendor hardware- and software platforms defining the current infrastructure of an organization’s IS-resources in an open client server environment.
2. Workflow Management: Automation vs. Cooperation

As outlined above we are convinced that workflow management must encompass and support a combination of both:

- a priori defined process structures in the sense of process control ([full] automation), and
- open and flexible processes, whose structures will be determined or refined due to evolving circumstances during task processing (in part automation or autonomous workgroup cooperation).

The first relates to today's existing corporate IS-infrastructures of highly structured large-volume transaction systems. The latter closely refers to the context of office systems based on more flexible IS-paradigms like workgroup computing, CSCW (computer supported cooperative work), or groupware.

Workflow Continuum in the GroupFlow Concept

We identify four different workflow categories whose details and substructures will be outlined below. The combination of these four categories provides a scalable degree of automation for workflow management. We utilize known concepts of information dissemination and messaging, varying and integrating them into a technology-framework from which elements can be derived for maximum synergy depending on the actual requirements. Thus, we do not regard as two distinct or even opposite concepts the more or less rigid structures of workflow automation on the one hand, and the flexible team-driven concepts of workgroup computing on the other hand. Rather, we synthesize both approaches using overlapping workflow (sub-) structures to form a basis for flexible and yet productive IS-design.

The basic patterns and some descriptive annotations describing the GroupFlow architecture from a business process design point of view are summarized in Fig. 1. The various annotations are intended to point out the continuous scale property of organizational workflows, the overlaps in underlying information and communication technologies, and some of the relevance factors defining a relative position in this framework. We will come back to the basic structural elements of the GroupFlow model as referred to in the center of Fig. 1 in chapter 3.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>e-mail, store-and-forward</td>
<td>open team task</td>
<td>a) open team task within standard WF</td>
<td>b) controlled team task within standard WF</td>
<td>c) ad hoc modification of standard WF</td>
</tr>
<tr>
<td>- urgent - short-lived - exceptional - confidential</td>
<td>combination of pre-determined and open tasks within a single workflow</td>
<td>partially unspecified elements within pre-determined workflow</td>
<td>completely pre-determined workflow and exception exception</td>
<td>generally preset next agent, shared DB</td>
</tr>
<tr>
<td>e.g. new type of request</td>
<td>- shared access - common task</td>
<td>- completely open as well as standardized tasks - intersection of both</td>
<td>- pre-set number of group members takes part - regardless of the sequence</td>
<td>- highly recurrent - pre-determined - easy-to-apply ad hoc modification / re-routing</td>
</tr>
<tr>
<td>e.g. co-authoring of publication</td>
<td>e.g. co-editing of annual report</td>
<td>e.g. counter-signature</td>
<td>e.g. consumer credit app. (particular customer request)</td>
<td>e.g. consumer credit application</td>
</tr>
</tbody>
</table>

Fig. 1: Workflow Continuum Architecture of GroupFlow Concept

The integration and workflow continuum concept has been the underlying architecture for designing the GroupFlow system. GroupFlow is based on workgroup computing concepts. Its implementation uses...
groupware as a development and deployment environment as well as the operative platform. GroupFlow encompasses the modeling of complete business process procedures on the managerial level as well as the complete set of tasks to develop, deploy and run a workflow system as seen from the technical and operational IS-level.

The business information being managed, communicated, worked upon by actors, automatically processed by software agents, stored, retrieved, or archived with GroupFlow is embedded in [document-] objects. The documents are stored in a distributed database environment (Lotus Notes). In the context of this paper, these documents ought to be regarded as containers being capable of handling such diverse data types as 'classical' formatted data structures (alpha-numeric fields, tables), rich-text data formats, graphics (object graphic [e.g. business graphic, CAD graphics], images e.g. Fax, scan), multimedia objects [e.g. video, speech annotation, sound]. Especially, the documents might as well be containers for workplace-centric application modules ('enduser programs' [e.g. interest calculation in a bank office, customized life insurance plan in an insurance company]) using a variety of object-oriented method embedding concepts in a cross platform environment.

### Business Relevance Factors for the Workflow Continuum Architecture

The GroupFlow workflow continuum architecture not only supports employment of varying automation degrees for new business processes to be computerized. Especially, it also enables a smooth transition for redesigning existing business processes. Thus, it allows adding functionality in direction of more flexible and open workflows to processes, once rigid or completely predetermined.

This flexible architecture seems important to us from a pragmatic point of view as well. It reflects the structure of information management in real life office environments. In the existing world the needs for flexible information flows in organizations have been reflected strenuously by the complex and versatile logistics of paper management for a long time. Many organizations find it difficult to define realistic strategies for productive and cost-efficient change in the current situation characterized by all too many paradigm shifts in information and communication technologies. Some of the patterns defining this change are sketched in Fig. 2. A strategy for change must leverage the investments in mostly standardized business procedures based on transaction systems on the one hand and flexible office systems resolving partially structured or alternating information flows via paper-logistics on the other hand. Companies may utilize the concepts like those embedded in the GroupFlow environment for a smooth transition. This allows for keeping some processes predetermined but integrate them as seamlessly as possible with an open and flexible workflow management infrastructure. In this sense the comprehensive approach of GroupFlow to business process management can be considered as encompassing the alternative concepts of workflow automation underlying large-volume workflow systems or purely cooperative but loosely structured taskflow management typical for office environments. In terms of the change patterns of the IS paradigm shift GroupFlow offers many options for transition paths to the new information system infrastructures as characterized on the right hand side of Fig. 2.

<table>
<thead>
<tr>
<th>Operational and transaction centric systems ('old IS-world'; legacy systems)</th>
<th>Business process and communication oriented systems ('new IS-world'; future systems)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data-centric design</td>
<td>Communication-centric design</td>
</tr>
<tr>
<td>Structured data: transactions, records, numbers</td>
<td>Semi-structured information types: messages, compound documents, rich object types open for multimedia</td>
</tr>
<tr>
<td>Formalized access: SQL, forms</td>
<td>Content-based access: viewer, proximity-search</td>
</tr>
<tr>
<td>Low-level integrity management: 'two-phase commit'</td>
<td>High level synchronization: replication, messaging</td>
</tr>
<tr>
<td>Static and fixed workplace environment</td>
<td>Mobile office</td>
</tr>
<tr>
<td>Internal organization focus</td>
<td>External focus</td>
</tr>
</tbody>
</table>

![Fig. 2: Paradigm-shifts in information system design](image)

We consider the forces of structural change currently taking place in many organizations worldwide as another motivating driver for having modeled the integrative GroupFlow architecture. This change includes...
flatter organizational models, less hierarchical structures, and more team-oriented business control concepts. The lower the degree of predetermined process structures or of office automation, the more care must be taken in assigning and maintaining clear and well communicated personal responsibilities to guarantee accountability for task coordination and performance. An enterprise has to compensate for the abundance of many of the outdated formal restrictions and fixed procedures that were built into most traditional application systems. If it does not do so by establishing sophisticated, reliable, obvious and yet flexible assignments for task responsibilities, it may run into difficulties to manage business process structures in an ever growing complexity of competitive market places. [Scherr 1993, p. 82]

Where traditional system design techniques of data and function modeling were incorporated as the predominant design pattern of application systems (the left hand side of Table 1), the business processes often have come up as rather accidental results of extended software life cycles. When focusing on the business processes as the major target, the system design has to converge on these processes as the dominant factor from the beginning. Like in the GroupFlow concept data and function modeling will just be incorporated as supportive methods, but not as the dominant pattern that preponderate the concern during information and communication system design.

Subsequently, we will cover the various workflow categories and [sub-]structures of the GroupFlow approach in an order from the rather cooperative end towards the automated side.

3. **GroupFlow Architecture Model: Workflow Continuum Structures**

1. **Ad hoc Workflows**

   *Ad hoc workflow* (Fig. 1, col. 1) usually deals with *unique* and rather short-lived processes. These non-recurrent processes vary largely in their degree of complexity. In general, single tasks of this type of workflow cannot be predetermined in advance, and are difficult to structure. Until now, processes of this type have been viewed as not worthwhile to be automated. In many cases these workflows are spontaneous and also urgent or confidential.

   Both initiation and execution of ad hoc workflows usually involve different actors. The various steps of ad hoc workflows are difficult or impossible to define in advance and can hardly be structured or standardized. However, certain parts of ad hoc workflows may be recurrent in some ways, such as automatic mail forwarding or automated posting of reminder messages.

   Generally speaking, the *transport of information* can be achieved in two different ways: (1) All pieces of information can be routed from one person to another (*send model*), or, (2) by giving access to common information bases (*share model*). Both concepts are supported within the GroupFlow framework. In an ad hoc workflow scenario the send model seems to be most appropriate. All information contained in document objects is sent from one actor to another in an email based store-and-forward manner.

   This model is basically applied for rather simple ad hoc routing applications. It is easy to set up a new email based workflow. Usually the addressee does not expect any request beforehand. The focus of the next agent's attention is pushed on incoming documents to be worked on or to be forwarded. In such a mere *point-to-point* routing scenario it may be difficult to track the status of a specific workflow at a given moment when documents are *sent* from one actor to another.

   Typical sample ad hoc workflows can be found for general purposes in office communication environments, in project management of individual tasks, or, customer requests that cannot be matched with any known standard service pattern within the organization.

2. **Team task: Self-Managing Team Controls Processes**

   Workflow applications that are basically performed by *team tasks* of self managing workgroups (Fig. 1, col. 2) have more of a routine typology than email based ad hoc workflows. Even though, it may neither be possible nor preferable to set narrow, predetermined specifications for a task management of this process category. Still, there are some common characteristics among each of these tasks.

   In a workgroup each member has knowledge about the abilities and assigned responsibilities of other team members. Thus, when working cooperatively on a team task all team members of a self managing workgroup can derive from the context of a task which actors have to be involved in its completion.
A team task with these characteristics can best be supported by an information system that offers some type of structured bulletin board or shared workspace. Shared document databases for teams as offered by groupware technology form an isomorphic and efficient foundation for team task support. The so-called share model provides - in a somewhat complementary approach to the send model - a public view on the current jobs all team members are working on. The share model enables group access to shared information databases by viewing a single source of information from different, customized perspectives. The major goal is to organize information for group usage and interaction. Thus it is possible for the team to flexibly react on unexpected changes within the office staff, such as absence from the workplace. Other actors can easily notice pending jobs of absent personnel and react accordingly. In many cases, this rather soft procedure of work assignment and supervision has proven to be much more practical than modeling complicated logical rules for replacement of absent personnel. When considering lean organizational concepts a workflow management environment should at least offer means for decentralized and independent work assignment by the team members themselves.

Continuous processes with certain routine characteristics are best modeled in shared document databases: Each person involved knows about the existence of information in the database and the respective tasks to be performed, or responses expected. Thus, all staff members concerned with these workflows consult the database on a regular basis checking for new documents applying a sort of pull approach towards information. They will react by working on existing documents (modify, complete, add data; set status fields), forwarding, copying, archiving documents, creating response documents, or initiating new documents.

The integration of both the send model and the share model as technically supported by the Lotus Notes platform can efficiently result in a variety of combinations over a wide spectrum. This spectrum stretches between email based applications including mechanisms like automatically sending a protocol entry into a shared database for easy tracking of the routing status on the one hand. And, it extends to usual shared databases which employ a push approach, focusing the actor's attention on a newly inserted or modified document in the shared database on the other hand. In our opinion this almost continuous transition and combination of the two models provides a far better flexibility and reflects real business process requirements better than both, mere transaction based process management systems or pure email based routing applications.

(3) Semi-structured Workflows

Next to the two basic workflow categories described previously we now focus on the major novelty being enabled by the GroupFlow approach. When reflecting real business process structures by information technology it is particularly necessary to integrate predetermined workflow structures that have been modeled beforehand as seamlessly as possible with an open and flexible process structure of team tasks. Thus, semi-structured workflows (Fig. 1, cols. 3a - 3c) seem to be the most important category for an appropriate business process support based on state-of-the-art information technology and the most significant innovation within the GroupFlow technology framework. It is particularly important to provide common access mechanisms to diverse information system functionality in an integrated approach instead of forcing the end user to act as the integration component by switching between several applications.

Subsequently, we describe three major types of semi-structured workflows: the intersection of predetermined and open team oriented tasks, the employment of controlled team tasks within a whole business process framework, and the ad hoc modification of predetermined generally well-structured workflows. Any of these semi-structured workflow types may be combined with each other.

(3a) Open Team Task Within Standard Workflows

Within the GroupFlow framework the most genuine combinations of open and structured business process elements can be modeled in a seamless way using two complementary principles: Either predetermined process elements can be viewed as the major structure incorporating open process parts, or, the other way around, open process structures can integrate predefined process elements (Fig. 1, col. 3a).

Predetermined workflows as the major process structure

The integration of an open team task into a predetermined workflow as the major structure implies that within a particular step of the process several members of the workgroup will be engaged in the completion
of a single complex task. These members will be involved without actually predefining the exact order beforehand in which the actors will take part in the team task.

Thus, before a job is entering the team task, or, after leaving it, there may be a predetermined workflow routing path for the job to follow. Particular care must be taken defining the interfacing elements between the predefined and the open workflow elements. This interface is modeled the following way: After a sequence of predetermined tasks have been performed on a job and the open team task is being entered by the job at its entry point, the job will appear on the task list of the group manager who has been assigned as responsible for the open team task.

The group manager is responsible for initiating the cooperative work of the group members during the open team task. This is the typical workflow management paradigm underlying a groupware infrastructure based on shared databases with all its flexibility referred to above [see (2)]. One crucial responsibility of the group manager is to authorize a single member or a particular member subset of the workgroup to finally close out the open team task. The group manager may very well decide to authorize no other member but himself to do so for some particular job.

Once the authorized agent has closed out the open team task and thus referred it to its exit point, the job is put back on the predetermined workflow path and automatically routed to the next agent. The embedding of this open team task concept in the GroupFlow approach is shown in Fig. 3.

![Open team task integrated in a predetermined workflow](image)

**Fig. 3:** Open team task integrated with predetermined workflow

Generally speaking, when lowering the degree of predetermined procedures of the process structure particular attention must be taken to assign clear responsibilities for the coordination of such open tasks within the workgroup. When a (partially) autonomous team takes over control of the workflow there must be a well defined personal responsibility for the task performance. If merely opened process elements are not complemented by reliable methods of accountability the processes may lead to rather unmanageable and chaotic structures.

**Open team task as the major process structure**

The integration of open and predetermined workflows may function also the other way around.

![Predetermined Workflow within Open Team Task](image)

**Fig. 4:** Predetermined Workflow within Open Team Task
Here, the open team task represents the predominant structural principle of the process. This type of workflow is characterized by a dominating pattern of less structured activities being necessary to complete a rather complex task which can be best performed by an autonomous workgroup. But, it may be desirable to integrate or link into this principally open structure certain predetermined task sequences defined by more routine type business processes. Technically speaking, these predetermined (sub-) workflows are stored in a workflow template library and can be easily chosen from the GroupFlow user interface by each eligible member of the open team task. After terminating the preset sequence of tasks the overall open process structure can be resumed. This architectural concept is shown in Fig. 4.

(3b) Controlled Team Task Within Standard Workflow

This semi-structured workflow is characterized by integrating such types of tasks into predefined workflows that are somewhat more predetermined than completely open process elements. The tasks are still more flexibly performed in the cooperative way of a team task than the tasks that are usually assigned to single users. We call this kind of tasks controlled team tasks (Fig. 1, col. 3b).

The typical example of controlled team tasks are group decision scenarios such as votes or countersignature. In the process of task definition the number of team members out of a group of potential actors participating in task completion are predefined. Later, during the actual task completion at runtime it does not matter which individual actors out of the predefined superset of potential actors participate in task completion. Also, the actors perform the controlled team task regardless of the sequence of their participation.

![Fig. 5: Controlled Team task integrated with predetermined workflow](image)

A more specific subset of this type of controlled team task determines that all members of such a group are required to interact, signing a document electronically etc. This architectural concept is shown in Fig. 4.

(3c) Ad hoc Modification and Exception Handling of Predetermined Workflows

Another basic principle of the GroupFlow architecture is that the predefined workflow is modifiable at runtime by ad hoc modifications and dynamic re-routing of the workflow for special cases and exceptions (Fig. 1, col. 3c). This allows for tied integration of standardized and predefined processes with flexible ad hoc changes to the workflow definitions. Ad hoc reactions may be required by specific circumstances that turn up during everyday work. A workflow system that does not provide the flexibility for the user to respond to this highly probable type of real life necessity forces him or her to leave the context of work within the workflow system - thus possibly causing fatal disruption. A solution to the workflow breakdown could be tried by physically meeting or calling an appropriate person whose interaction is necessary to continue the job. Another way to handle the disruption could be to write a paper based memo or - if possible - use email describing the nature of the problem and asking for solution. The disadvantages of a required synchronous communication in the first alternative are obvious. With the second memo based approach this can be prevented. Still, the effort for explaining the information context of the disrupted job may be immense.
The GroupFlow environment integrates a set of structured mechanisms to handle exception or disruption problems of various kinds in predefined workflows. The possible modification rules applying to given workflows have been modeled after common exception situations in business processes, such as:

- **check-back**,  
- **inquiry**,  
- **task delegation**, or  
- complete **change** of the underlying workflow type.

Some characteristics of these exception and disruption routing routines and resume mechanisms after disruption handling are outlined in Fig. 6. Re-routing can easily be initiated offering similar information as shown in Fig. 6 in the GroupFlow user interface.

<table>
<thead>
<tr>
<th>Check-back with previous agent</th>
<th>Question to anyone else</th>
<th>Detour in routing path</th>
<th>Change the type of workflow</th>
</tr>
</thead>
</table>
| - posing request to previous agent of workflow  
  - e.g. confirmation about a particular fact or item  
  - seamless integration with Email interface  
  - context notification by automatic DocLink |  
  - pausing the routing  
  - pose a question to a person not yet involved in that workflow  
  - e.g. request to supervisor  
  - seamless integration with Email interface  
  - context notification by automatic DocLink |  
  - involving further, previously not involved actor into this workflow by new task assignment  
  - delegate task to a person or role  
  - seamless re-routing of workflow |  
  - exchange of underlying workflow type  
  - continue with appropriate organizational unit in new workflow  
  - complete re-use of previously entered data |

**Check-back** | **Inquiry** | **Re-route** | **Change Workflow**

*Fig. 6: Ad hoc change and exception handling of predetermined workflows in the GroupFlow system*

Exception management necessitates rather complex considerations in the information management model underlying the GroupFlow approach. One example is handling delegation of tasks within a predefined workflow. Managing a delegation requires authorization granting, collecting of the task description and pending job context by the current agent and routing to the new delegated handling agent including automatic email notification, setting up the resume entry points, and more. Another example is workflow change. Causes for complete change of an underlying workflow type are manifold in real life business processes. GroupFlow allows for a job, having been routed part way through one process, to seamlessly swap to a different process, transferring the relevant information, and initiating the second process at the appropriated workflow step.

As with all other activities within a workflow system, the exception handling to the standard workflow specifications must be thoroughly recorded. The audit trails can be found as entries in the workflow protocol. If for instance a certain inquiry pattern is regularly started at a specific task within a given
workflow over and over again the reason has to be investigated. This may, after all, result in changes to the regular workflow type definition.

(4) Predetermined Workflows

Standard processes usually consist of highly recurrent structures (Fig. 1, col. 4). These workflows pass through the same predetermined order of steps over and over again. Very often they consist of routine activities.

A one-time investment in task analysis and the development of automated applications seems to be profitable for high volume processes. It should be considered as well for typical sequential processing patterns being followed at several occasions within an organization and thus being re-usable within many workflows. Pre-designed process models determine the complete procedures with their activities, agents, and routing paths including possible alternatives in advance. The involved actors of the processes with their position inside the organizational structure and their roles have to be included.

Within the GroupFlow approach predetermined workflows are the partially underlying structures to some of the workflow cases discussed above. Many of the aspects around defining and structuring the rigid logic, parameter settings and routing sequence of tasks in the GroupFlow system environment are discussed in the following chapter.

4. Workflow Modeling

WOMED Workflow Modeling Editor

The motivation for developing the WOMED graphical modeler with its graphical workflow representation is twofold within the GroupFlow framework. One reason is to enable an intuitive and straightforward modeling during business process planning cycles depending basically on the skill set of typical management organizers. The second reason is to support the GroupFlow workflow designers and administrators being responsible for the GroupFlow runtime environment WOREN by isomorphically reflecting the workflow structure repository in a graphical user interface.

All data of WOMED are being held in the very workflow structure repository databases (Lotus Notes server database files) defining and monitoring the GroupFlow runtime systems. Thus, WOMED can be used as input tool for workflow structure definition as well as an output / presentation tool for workflows alternatively being edited in the Lotus Notes native outlining and form fashion. Fig. 7 gives an idea of this concept by representing the ‘INCOMING MAIL’ workflow in both isomorphic ways. For an ex ante analysis the modeled workflow may be simulated in the WOMED simulation mode before actually using it in the runtime environment. [Ott 1994]

Fig. 7: Transform Graphical Workflow Model into Repository Entries
The WOREN operative system uses the entries in the workflow structure repository to actually route the document objects from one actor to another. This is depicted in Fig. 8 presenting one workflow step of the above 'INCOMING MAIL' example as seen from the users' frontends during actual workflow operations at their client workstations. We have tried to design the GroupFlow user interfaces around suggestive metaphors especially to model workflow and routing initiating actions like 'transfer to next agent'. Thus, in this example the user has to push an action button in order to transfer a document object to the next agent according to the workflow definition as specified by the WOMED workflow modeler.

![Routing information in forms and views of the GroupFlow runtime system user interfaces](image)

**Fig. 8**: Routing information in forms and views of the GroupFlow runtime system user interfaces

**Workflow Clustering**

For large processes encompassing many tasks it may become difficult to comprehend the graphical network of the numerous tasks and their temporal constraints as expressed by the edges between task nodes. In order to provide the means for sufficient understanding of complex business processes, diverse aggregation levels of the workflow representation are required. Thus, the workflow structure will be easier to overview and to manage as compared to a flat-model approach.

The WOMED workflow modeling editor seamlessly supports simultaneous top-down and bottom-up planning cycles for business processes. On a rather high aggregation level, an organizer may want to first design complex tasks which have to be decomposed into several subtasks successively later on. On this macro level a workflow model representing real business process infrastructures will usually consist of interconnected macro workflows and not operative tasks. During the same planning cycle, he or she may want to draft as well on specific (sub-) workflows on a much lower or even operative level aggregating them into reusable workflow objects to be included at several macro steps of other workflows.

WOMED uses a clustering mechanism to manage this type of complexity on several abstraction levels, eventually leading down to the operative tasks on the very bottom of the layered workflow graph. Each cluster can be understood as a sub-workflow representing a group of tasks. More precisely, as clustering works recursively, a cluster can contain (final) tasks as well as other clusters, again representing sub-workflows (Fig. 9). On the next higher level a cluster will be denoted as an aggregated (sub-) workflow object.
As shown in Fig. 10 a cluster is represented by a special icon [A]. A particular child window offers a detailed view of the several objects (tasks or sub-workflows) comprising the cluster [A']. The variety of task classes as described above in chapter 3 is iconized. As an example, [B] denotes an open team task. The open team task [B] appears as editable task on the main workflow level. It is also mapped down as one exit point to the sub-workflow level [A']; there it is - naturally - not editable (denoted by the bounding rectangle). For modification of an existing workflow a user may want to add a new task. This is handled by a drag and drop metaphor: [C1.] select and drag a new task node icon over the drawing pad, [C2.] indicate inclusion possibilities by temporarily changing the icon layout over possible inclusion points, and [C3.] finally include the new node by dropping the icon over an edge between two existing task nodes where the new task is to be inserted.

Fig. 10: Workflow Modeling Editor: Example for Clusters and 'Include New Task' Metaphor

The rather general approach taking into account by developing the clustering mechanism within the WOMED workflow editor enables consistent modeling of strictly business relevant layers: In a scenario, where information bound to document objects flows completely electronically between independent business units, particular care must be taken for the design of the common electronic interface between such enterprises. First of all, efficiency requires to exchange exactly the necessary type of information for a wide area workflow that runs across enterprise boundaries. Also, security requires a tight management of the information to be exchanged or given external access to (It is definitely not a simple endeavor to remodel the complexity, flexibility, and security patterns underlying the logistics of bulky paper based
corporate information exchange with the typical electronic means like, say, access control lists, user-ids and passwords, or RSA encryption, taking into account that everything, of course, has to be done for [real] people as well as roles. Further, transmission costs between the business units have to be considered.

Particularly in terms of the first two aspects, the workflow clustering technique may be useful: The organizations that are to model a common workflow exchange the process model on an aggregated level. They may only know a rather general description of the tasks their partner or branch organization is aiming at. Each of the participating branches can decompose its own task description, but none of the external tasks or clusters. These are regarded as confidential and encapsulated black boxes of which only specific interface information will be available to the external partner organization.

5. References and Acknowledgments

References

[Davenport 1993]

[Hammer/Champy 1993]

[Hilpert 1993]

[Ishii/Ohkubo 1991]

[Marshak 1992]

[Medina-Mora/Winograd/Flores/Flores 1992]

[Nastansky/Hilpert 1993]

[Ott 1994]

[Scherr 1993]
In: IBM Systems Journal, 32, 1, 1993, pp. 80-98.

Acknowledgments

GroupFlow has been developed within the CSDS project group (client server distributed systems) at the University of Paderborn in the Department of Business Computing over the past two years. Part of the basic research projects around modeling the business process side as well as researching some of the architectural principles were supported by Deutsche Forschungsgemeinschaft (German National Research Council, DFG). A great deal of resources for enabling the prototyping and implementation of the runtime modules have been brought together by partnerships with external organizations. The development could not have been possible without the keen engagement of many students who were bearing the arduous burdens of code development and testing, to mention last but not least Frank Brockmeyer, Stephan Gieffers, Ralf Heindörfer, Oliver Heinz and Marcus Ott.