Dynamic Workflow Management: A Framework for Modeling Workflows

M. Millie Kwan and P. R. Balasubramanian
Boston University School of Management
E-mail: {mmkwan, bala}@bu.edu

Abstract

Current workflow management systems do not provide adequate support for workflow modeling. Real life work processes are much richer in variations and more dynamic than is expressed in a typical workflow model. Users need to be able to adjust workloads and modify workflow models on-the-fly. In addition, data about workflow executions are analyzed with process analysis/simulation tools to evaluate design alternatives, so workflow models and data must be structured to reflect the questions that managers and designers are likely to ask. In this paper, we present Dynamic Workflow Management (DWM), a framework for modeling workflows that aims to satisfy these requirements. DWM provides high level packaged task templates for composing task sequences, a MOP-like structure based on dynamic memory theory to organize the task sequences into flexible workflow models, and four perspectives into a workflow model geared toward different users' interests. We illustrate these features of DWM with a case study of an interlibrary loan process and discuss its application in workflow enactment and simulation.

1. Introduction

As organizations become increasingly process-centered, managers are actively seeking out technologies to support process management and innovation [13]. Existing IT support comes in two categories: process modeling/simulation tools for process design and analysis, and workflow management systems (WFMS) for process implementation. However, the transition from process modeling at the design level to workflow support at the implementation level is not straightforward, as researchers in office systems already learned in the 1980s [8] and many practitioners found out more recently [2]. Real life work processes are much richer in variations and more dynamic than is delineated in a typical process model [31]. This means that users need to be able to adjust workloads and modify workflow models on-the-fly. In addition, data collected by WFMS, mainly with process analysis/ simulation tools, about workflow executions are analyzed to evaluate design alternatives during business process redesign [26]. However, simulations are useful only inasmuch as the models are an accurate representation of reality and contain useful measurements. This means that workflow models and data must be structured to reflect the questions that managers and designers are likely to ask.

The above discussion indicates that support for work-flow modeling poses at least three challenges for developers of WFMS: 1) the structure of the workflow model must be *flexible* enough to describe variety in a process design and accommodate exceptions during enactment, 2) the workflow modeling facility must be *expressive* enough to allow analysts as well as end-users to specify workflow relatively quickly and easily, and 3) the model must be *structured* to facilitate workflow analysis. Recent research shows that modeling capabilities in current WFMS have not yet met these requirements [1]. In this paper, we present **Dynamic Workflow Management** (DWM) a framework for modeling workflows that attempts to address these challenges as follows. First, following a process modeling framework suggested by Curtis et al. [12], DWM describes a workflow specification from four perspectives: functional, behavioral, organizational and informational. These perspectives represent the different areas of concern by workflow users. Second, in the functional perspective, tasks are characterized by proto-typical tasks [4] and organized in a structure similar to that of MOPs in dynamic memory theory and case-based systems [19, 29]. Prototypical tasks are packaged task definitions with predefined attributes and behavior which can be used as templates to define the tasks at hand. They provide a quick and easy way for users to view, create and modify tasks. A MOP-like organization enables easy access as well as insertion or removal of tasks during design and execution.

With its emphasis on structure and variations in workflows, DWM is most appropriate for processes that are strong in those characteristics. In [20], we matched workflow design approaches to process characteristics along two dimensions as shown in Fig. 1. The horizontal axis defines sequential and reciprocal **interdependence**

Proceedings of The Thirtieth Annual Hawwaii International Conference
on System Sciences ISBN 0-8186-7862-3/97 $17.00 © 1997 IEEE

according to Thompson's framework for organizational technology [33]. Processes in which there are successive stages of production exhibit sequential interdependence. Processes in which various techniques and resources are employed in an *ad hoc* manner based on feedback from other tasks exhibit reciprocal interdependence. The vertical axis represents **variety**, a measure of the diversity of inputs to the production process defined by Perrow [27]. The form-based [2], goal-based [10, 11] and communication-based [24] workflow design approaches support the specification of predetermined task sequences. Task variety must be low, otherwise, specifying a large number of task sequences will become unwieldy. Although goal-based designs can derive new task sequences based on rules, the task variety must still be low or there will be too many rules, each covering only a small number of cases. On the other hand, the tool-based approach [7] is suitable for high variety processes because it provides low level task types as building blocks for the end-user to define or modify a workflow in an *ad hoc* manner, rather than specify a design *a priori*. The tedious task to specify all possible cases in advance has been avoided, but the approach is inefficient for sequential workflows with predetermined task sequences.

Variety	Sequential	Reciprocal
High	clinical services for chronic diseases, software support services	software development *tools approach*
Low	Order processing, Claims processing *Goal-based approach, Forms approach*	budgeting, hiring *communication approach*
	Interdependence	

Fig. 1 Workflow system design classification by process characteristics

Our research reveals that there is an unfilled need in workflow modeling capabilities for high variety, sequential workflows. Examples of high variety, sequential processes are software support services and clinical services for chronic diseases. Many production processes, which are of high business value, could also be in this category, but because of the limitations of current modeling capabilities, much of the variety in these workflows have been set aside as exceptions to be dealt with outside the WFMS. As more organizations employ WFMS to support their processes, the ability to handle rich variations in workflows will become a major differentiator among WFMS [2]. High variety, sequential workflows are good candidates for automated support because 1) WFMS can serve as a memory for large sets of complex procedures and 2) WFMS can enhance the performance of structured workflows by managing operational issues such as relieving bottlenecks and tracking hand-offs. The modeling framework, DWM, we propose here is designed to satisfy the modeling requirements of these workflows. DWM extends the tools approach to provide higher level packaged task templates for users to compose task sequences and provides a MOP-like structure to organize the task sequences into flexible workflow models. This flexible organization enables easy model storage, modification and retrieval, providing a foundation for model management support for WFMS [5, 6].

The paper is organized as follows. In Section 2, we provide a brief review of the workflow modeling literature and describe the elements of workflow employed in DWM. In Section 3, we define the DWM framework. In Section 4, we illustrate the features of DWM using a case study of an interlibrary loan borrowing process. In Section 5, we describe the contexts, i.e., workflow enactment and analysis, in which DWM would be applied. Section 6 discusses related work and future research directions.

2. Workflow modeling

Organizations engage in activities that produce outputs of value to the customer. An organizational process is a set of logically related tasks that transform a set of inputs into outputs of value to the customer. A workflow model is a representation of those aspects of a process that pertain to the coordination of the activities. Because models are designed according to the objectives and the context of the application, different model formulations for workflow have emerged in the literature. Nevertheless, WFMS generally employ a workflow model that is *task-centered*. Detailed descriptions of such models can be found in [15, 17, 28, 34]. The common model elements are:

- **Procedure** - a partial or total ordering of a predefined set of tasks.
- **Task** - a unit of work. In some models, tasks are atomic. In other models, tasks are decomposable.
- **Information object** - data or documents manipulated in a task.
- **Role** - a placeholder for an actor that is associated with the execution of a task.
- **Actor** - an entity (human or computer) that can assume a role. An actor may take on multiple roles and a role may be assigned to multiple actors.

Some models contain other elements that reflect the design objectives of the model. Here is a list of them:

- **Goal** - A task is frequently associated with or identified by a goal. Goals are decomposable just as tasks are decomposable. In some models, goals are used for task planning and exception handling [10, 11], or for completeness and consistency checking [14, 21].

Proceedings of The Thirtieth Annual Hawaii International Conference on System Sciences ISBN 0-8186-7862-3/97 $17.00 © 1997 IEEE

- **Resource** - a tool, information systems or other object that is required to perform a task. In some models, actors are also viewed as a resource [14]. Almost all WFMS provide actor assignment [4] or task assignment (to actors) [28] as a feature. Resources, on the other hand, are generally not modeled in current WFMS, but are extensively modeled in process simulation tools. Even so, it is desirable to give the responsibility of resource management to the WFMS because it can take dynamic conditions into account in assigning resources to tasks.
- **Event** - an occurrence that causes a task to be performed. Events are usually defined to model exceptions and time-driven tasks, but an event-driven model may consider a task or the start and end of a task to be events [18, 28].
- **Responsibility/Authority** - In communication-based models [24, 32], tasks are defined by requests and commitments between the customer and the performer. The focus is on who is responsible for the task and the transfer of responsibility between tasks. Interestingly, the concept of *routes* in form-based models is analogous to transfers of responsibility between actors as the form is routed from one actor to the next.

Rules are frequently named as an element of workflow in the trade press [2, 23]. Rules specify the behavior of the process and are applied during execution. Actor assignment rules specify which actor fills the role for a task. Process execution rules (procedures) schedule tasks to be performed under various constraints and conditions. Resource management rules specify the priority under which tools and objects are made available to actors for performing tasks.

The representation techniques for current workflow models are mostly limited to conventional flow models with hierarchical decomposition. Two notable exceptions are Information Control Net [14] which focuses on information flows and Role Interaction Net [30] which focuses on roles and their interactions in performing tasks. As organizations increasingly deploy WFMS to support more mission-critical processes, the issues of model creation and reuse, i.e., the issues of model management [5, 6], become more important. Although some WFMS support reuse by providing procedure templates and model libraries [28, 32], little has been done to facilitate the storage and retrieval of workflow models. In the following section, we propose a workflow modeling framework, DWM, that will address this issue.

3. DWM: A Workflow Modeling Framework

DWM is designed to address the modeling requirements of high variety, sequential workflows. It models workflows from four perspectives: functional, behavioral, organizational and informational, which address the different areas of concern of different types of WFMS users. The functional structure of DWM consists of building blocks which are high-level packaged task definitions, organized in a *flexible* MOP-like structure in order to accommodate a variety of designs and handle exceptions. We describe the four perspectives of the DWM framework in Sections 3.1-3.4.

3.1. Functional perspective

The functional perspective answers the question: what are the tasks performed and why? It describes the tasks, and the goals that the tasks are intended to achieve. The conventional technique for identifying and organizing tasks in a process is functional decomposition. A top level task is recursively decomposed into subtasks, resulting in a hierarchical structure. Such a structure presumes that there is only one way to achieve any particular goal. However, process goals can often be accomplished in more than one way. For example, the subgoal 'get order data' in an order fulfillment process may, by design, be accomplished in several ways: order data may be obtained by a telephone operator who fills in an order form on computer, or by a clerk who keys in a mail-in order, or it may be automatically generated by a buyer-supplier agreement. Each represents a different set of tasks (i.e. procedure) that accomplishes the same goal. Alternatively, in an *ad hoc* situation, extra tasks may be added to the designed sequence to accommodate a newly hired employee who lacks a certain skill. There are many factors that can cause different task sequences to be employed for accomplishing a goal. Figure 2 lists some typical ones.

To organize the various task sequences that can be applied toward achieving a goal, we employ a structure similar to that of MOPs in dynamic memory theory [29]. For each goal, a prototypical task sequence is first described. Variations of that sequence are then attached to the same goal, indexed by the situations under which those variations are applicable. To organize tasks of a process in this type of 'hierarchy', we need to group tasks into task sequences with similar subgoals (a bottom-up exercise), and allow goals to be decomposed into multiple hierarchies (a top-down exercise). Techniques for checking completeness, such as identifying missing and implicit goals, can also be appropriately applied here [21]. This flexible model structure enables parts of workflow to be added, changed or replaced without affecting the entire process. To utilize such a model effectively, indices that facilitate the storage and retrieval of the appropriate task sequences must be constructed. The factors shown in Fig. 2 could be the basis for such an index.

Proceedings of The Thirtieth Annual Hawwaii International Conference
on System Sciences ISBN 0-8186-7862-3/97 $17.00 © 1997 IEEE

Skill requirements Example: A worker who can type will prepare his own memos. One who doesn't will write a draft, give it to a typist, receive the typed copy, make revisions and repeat until it is done.
Legal requirements Example: Certain combinations of medical conditions needs to be documented.
Organizational policy Example: Certain items must be purchased from a particular vendor according to a corporate agreement.
Customer/supplier requirements Example: A handicapped customer needs special assistance. A supplier requires prepayment.
Performance requirements Example: A 'rush' case receives higher priorities for resources.
Authority/responsibility constraints Example: If a requisition is > $500, pre-approval by the department head is required.
Production technology constraints Example: A goal may be accomplished by applying either technology A or B. Technology A may be more costly but less available.

Fig. 2 Criteria for selecting different task sequences to accomplish a goal

To facilitate analysis and expression, tasks can be categorized into packaged operations with predefined attributes and behavior. Some workflow modeling tools provide convenient building blocks or templates for this purpose. For example, SPARKS [9] provides different sets of building blocks for different types of business processes. Examples of these building blocks are fax, meet, enter-data, analyze, and sort. Various attributes and measures are defined in these building blocks to facilitate analysis and expression. Although these building blocks are close to reality and easy to use, broader conceptual categories are needed for higher level analyses. Fig. 3 list examples of prototypical tasks and exception handling routines from Barthelmess and Wainer [4] and our work. At the highest level, one can postulate that a process (task) based on a workcase generally consists of four stages: preparation, setup, execute, completion. In the preparation stage, the workcase is run through criteria to determine that it qualifies for the workflow. In the setup stage, the requirements and the conditions, i.e., the plan, for processing the workcase is determined. In the execution stage, the plan is executed. In the completion stage, procedures such as reporting, archiving, and evaluation are performed to wrap up the case.

By including knowledge about the characteristic of the task, and the conditions and rationale under which a task (sequence) should be applied, our workflow model can be more intelligent in delivering work items and managing workflow under different situations.

Prototypical tasks	Exception handling routines
GD: external data gathering	Redo executed tasks
DP: data processing	Attach explanation for exceptional action
NG: negotiation	Skip some of the tasks
DC: decision	Add or change information in the workcase
AU: authorization	Reassign task to another actor
WT: waiting	Forward workcase to another actor
EA: external activity	
HO: hand-off	

Fig. 3 Prototypical tasks and exception handling routines

3.2. Behavioral perspective

The behavioral perspective answers the question: when and how are the tasks performed? It describes the sequencing, control flow and conditions that govern the execution of workflow. Conventional techniques used are state transition diagrams, petri nets and flow charts. In describing the dynamic behavior of workflows, we need to specify when a specific task sequence should be selected to accomplish a goal under a particular situation. The situation is defined by the conditions, and the task sequence is defined by the sequencing and control of the tasks which are performed. Conditions may be static or dynamic or a combination of both. For example, legal requirements are generally static whereas staffing levels and availability of tools can change dynamically. Exceptional events that trigger specific task sequences are also specified as conditions. For example, when a customer calls in to cancel an order, the order status (a condition) will be changed to 'canceling' and a task sequence would be selected to wrap up the aborted workcase.

3.3. Organizational perspective

The organizational perspective answers the question: who, where and with what are the tasks performed? The question 'who' is answered by specifying roles (including customer and supplier) and actors (performers). Roles are assigned responsibility for the tasks. Individual actors may have different skill sets enabling them to play different roles or have roles assigned to them. Actors perform tasks using resources such as tools, services, and databases. Resource requirements for tasks are specified in terms of the resource name, quantity, time schedules and durations.

The organizational perspective also describes how actors and resources are organized. This includes information about their locations, availability schedules, access privileges, cost schedules, priorities and capabilities. This information provides an understanding of the time and spatial constraints as well as cost and capacity considerations in workflow execution.

3.4. Informational perspective

Proceedings of The Thirtieth Annual Hawwaii International Conference
on System Sciences ISBN 0-8186-7862-3/97 $17.00 © 1997 IEEE

The tasks in a process typically involve the creation or processing of objects or artifacts. The informational perspective provides data about these objects necessary for executing the workflow. For example, in an order fulfillment process, orders, invoices and packing slips are objects manipulated in the process. The informational perspective describes, not the business data contained in these objects, but the information about them that pertain to workflow, such as creation date, status, deadlines, and person responsible. This information is often a source of operational constraints, i.e., conditions, that drive the selection of task sequences in workflow execution as described in the behavioral perspective. For example, an imminent payment deadline in an invoice may cause an exceptional task sequence to be executed.

The objects in a process bear specific relationships to each other. For example, in an accounts payable process, a purchase order must precede a receiving document which in turn must precede an invoice. These can also be a source of operational constraint driving the behavioral aspect of workflow. If the relationship given in the example above is violated, an exception would occur.

We have introduced our framework for workflow modeling. The four perspectives, their characteristics and representations are summarized in Fig. 4. In the next section, we present a case study to illustrate the features of DWM.

Perspective	Questions	Workflow elements	Modeling tools
Functional	What, why	tasks	functional decomposition (with variants), prototypical tasks
Behavioral	When, how	process rules	flowcharts, state charts, petri nets
Organizational	Who, with what	actors, roles, resources and resource management rules	organization charts, object hierarchies
Informational	on what	objects	E-R diagrams

Fig. 4 Four perspectives on workflow

4. Case study

In order to test the DWM framework, we have applied it to model the interlibrary loan (ILL) borrowing process at a university. A description of the process is given below. In sections 4.1-4.4, we provide examples from the process to illustrate the four perspectives of the framework.

Interlibrary loan process

Preparation - *When a patron needs an item that is not owned by the library, she requests interlibrary loan service at the reference desk. Her request is screened by a reference librarian who verifies her eligibility for the service and checks that sufficient information about the item is provided in the request form filled out by the patron. The request is passed on to the ILL office where requests are separated into book requests, article requests and dissertation requests. The requests are also screened to remove potential problems, errors, and difficult requests, which will be handled separately.*

Set up - *This involves selecting potential lender(s) and negotiating the loan. Potential lenders can be found by searching a number of bibliographic systems, but mainly on Online Computer Library Center (OCLC), a nation-wide library cataloging and holdings database service. Once a record has been located, the 'best' lenders are selected based on a number of criteria such as cost, contract relationships, and the speed and quality of service. The negotiation is mostly done on OCLC, which has the largest library holdings information in the US and supports the ILL process for its members with a transactions database. Libraries negotiate ILL via a protocol of pre-formatted messages. A library may provide the names of up to 5 potential lenders at a time. When a request is rejected by one candidate lender (the item may be checked out to a patron at that library), the system automatically forwards the request to the next candidate. To initiate a negotiation, an ILL record is created that specifies the terms the library is willing to accept (such as maximum charges), deadline for acceptance, mailing instructions, etc. Replies by candidate lenders are retrieved and processed every day.*

Execution - *Once the terms of the loan have been agreed upon, the lender sends the item to the library. On receipt of the item, an employee checks it in on OCLC, sets up a circulation record, files the item, and notifies the patron to pick up the item. Common circulation events, such as renewal, overdue and loss, may occur after the patron checked out the item. When the patron returns the book, the circulation record is removed. The item is sent back to the lender. ILL fees may be paid on the OCLC system or separately invoiced by the lender.*

Completion - *When the lender acknowledges receipt of the item, the paperwork for the whole workcase is filed.*

4.1. Functional perspective

Fig. 5a shows the top levels of the goal hierarchy for this typical case. The atomic tasks are labeled with prefixes that represent the task types listed in Fig. 3. (Not all the leaf tasks in the diagram are atomic.) One example of a variant from this hierarchy is in the processing of theses (fig. 5b). As the institution that confers the degree is the only possible lender, lender selection is performed in the preparation phase by the Reference librarian, who looks up

Proceedings of The Thirtieth Annual Hawwaii International Conference
on System Sciences ISBN 0-8186-7862-3/97 $17.00 © 1997 IEEE

1060-3425/97 $10.00 (c) 1997 IEEE

the ILL policy of the institution to verify that they lend theses. Otherwise, the loan request is rejected and the librarian may advise the patron to obtain the thesis through other means (such as purchase it from UMI). Other variants can be produced by exceptions. For example, if the patron did not come to pick up the loan item, the task 'circulate item' is skipped and the item is returned to the lender. If no loan has been successfully negotiated before the deadline set by the patron, the workflow is terminated without performing 'execute loan'.

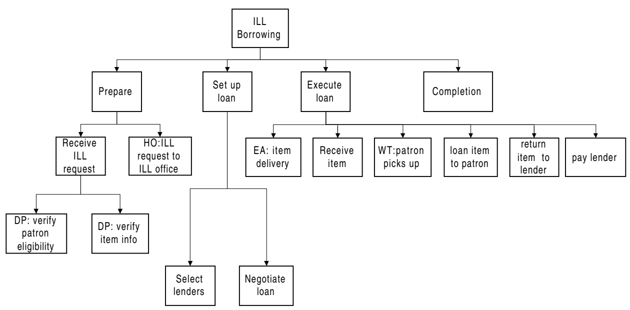

Fig. 5a Interlibrary loan borrowing - functional hierarchy

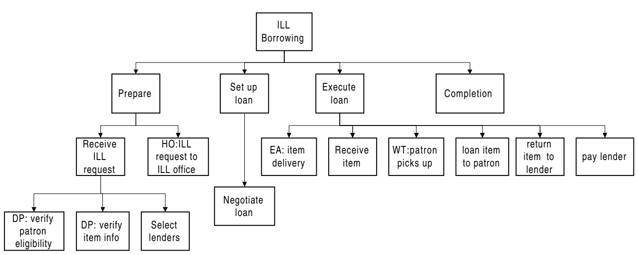

Fig. 5b Interlibrary loan borrowing (theses) - functional hierarchy

Various factors may add to or change the task sequence for achieving a goal. Fig. 6a & b shows extra tasks performed for 'select lenders' as a result of skill requirements, legal requirements, production technology constraints and organizational policy. First, the selection of lenders requires background knowledge about those lenders, such as location, quality of service and charges. Because student workers work only for a few hours each week, they do not accumulate such knowledge from experience. Thus a student worker would locate the proper record on OCLC and print the entire list of potential lenders. A staff member would make the selection from the list. The student then creates the ILL record using that selection. If no student workers are involved (as would be the case during vacation breaks), the staff member would perform the search, select five lenders and create the ILL record. Thus, skill requirements add two hand-offs to the task sequence.

Second, journal articles are handled differently from books. The library observes a copyright guideline that poses a limit of 5 photocopies each year of articles from the 5 most recent volumes of a journal title. As a result, a library maintains a count of photocopies for each journal title and checks the count each time request for photocopies are made. When the limit is exceeded, copyright clearance, i.e., payment of royalties, or permission of publisher, is required. In this situation, copies are obtained through clearinghouses such as the Copyright Clearance Center, and royalty fees may be charged. Thus, legal requirements add an extra task to the workflow.

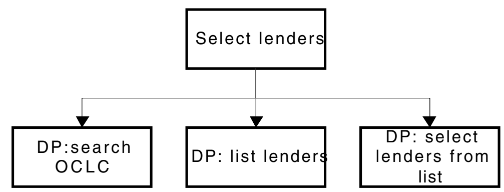

Fig. 6a Select lenders

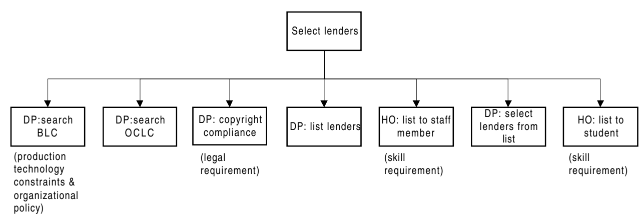

Fig. 6b Additional tasks due to different conditions

Third, requests for journal articles are first searched on the union catalog of the Consortium, of which the library is a member. This additional step is performed for two reasons: 1) the Consortium catalog has more complete and up-to-date holdings information of journal issues than OCLC, and 2) Consortium members have agreed to waive ILL charges against each other. Thus, production technology (Consortium catalog vs. OCLC) and corporate policy (the consortium agreement) both contributed to the addition of this extra step.

4.2. Behavioral perspective

We specify when and how the tasks identified above are performed, i.e., in what sequence, under what conditions, timing constraints, and so on. For each task, we specify, the enabling conditions, the action, possible outcomes. The enabling conditions typically include the existence of inputs, the availability of resources required and actors who can take up the role of performing the task. The action describes what is done. The outcomes describe the outputs and any conditions that may have been set by the action. These specifications represent the execution rules of the

Proceedings of The Thirtieth Annual Hawwaii International Conference on System Sciences ISBN 0-8186-7862-3/97 $17.00 © 1997 IEEE

1060-3425/97 $10.00 (c) 1997 IEEE

workflow. Fig. 7a shows the behavioral perspective of task 'search OCLC'. Prototypical tasks have standard behaviors. For example Wait typically involves setting a reminder schedule to flag the wait condition if the anticipated event has not realized. Fig. 7b shows the task 'wait for item delivery' in the ILL process.

Task: Search OCLC
Enabling conditions: request in search queue, OCLC terminal available, OCLC system is up, searcher is ready
Action: search for the OCLC record for the requested item, print record if found
Outcome1: request and OCLC record printout placed in 'select lenders' queue
Outcome2: request placed in 'problems with search' queue

Fig. 7a Behavior of task 'search OCLC'

Task: Wait for item delivery
Type: Wait
Enabling conditions: lender notified shipped date
Reminder schedule: every T = five days
Reminder action: RA = track item
Action: while not terminate condition wait T (set by reminder schedule) perform RA.
Terminate condition: item received, maximum wait period or deadline expired
Outcome1: item received Outcome2: item lost Outcome3: item status unknown

Fig. 7b Behavior of task 'wait for item delivery'

4.3. Organizational perspective

The *roles* in this workflow are ILL supervisor, ILL staff, students, patron, Reference librarian, and lender. *Responsibility* for tasks are specified at all levels of the hierarchy and all its variants. Thus, the task 'select lenders' is the responsibility of ILL staff. When this task is performed as in Fig. 6b, all the subtasks except 'select lenders from list' have been delegated to students. Individual students have different skill sets. Some are trained to search OCLC and/or Consortium catalog. Others perform tasks such as 'loan item to patron' or filing.

The *resources* in this workflow are the support tools and services, including OCLC, Consortium catalog, Copyright Clearance Center, ILL policy handbook (which contains ILL policy of institutions that participate in ILL), mail services, accounting services (for ILL charges), as well as equipment such as PCs, fax, printers and scanners, data stores such as copyright compliance file, circulation records, and patron requests. The same PCs are used to access OCLC, Consortium catalog and Copyright Clearance Center, so that multiple resources must be simultaneously available to support a task. In addition, the same tools are used in different tasks, e.g., OCLC is used for searching new requests, negotiating loans, receiving loan items and notifying lenders when the items are returned. Because of limited resources and uncertain work schedules of students, it is essential to include knowledge about operating schedules for resources and actors, as well as task priorities in a model of this workflow.

4.4 Informational perspective

The informational entities are the patron's request, ILL record on OCLC, loan item, terms of loan, and invoices for ILL charges. Examples of informational elements that pertain to workflow can be found in these entities. In the patron's request, the item type (book, journal article or thesis), the request date and the deadline after which the item is no longer useful. In the OCLC record, the loan status is the entry or exit condition of many tasks, e.g., when the loan status is 'lender shipped on xx date', the task 'wait for item delivery' is started. The terms of loan on the OCLC record determines which task sequences should be applied to achieve certain goals. For example, method of delivery (specific mail courier) may be specified, or item may be 'library use only' which means that the item cannot be circulated but the patron can use it in the library.

The relationships between these entities also drives behavior. For example, the loan item (such as a book) must match the item specified in the patron's request and in the OCLC record. Otherwise, an exception would occur.

5. Applying DWM to organizational processes

DWM was constructed with the goal of supporting workflow enactment and analysis. We envision DWM to be employed by workflow analysts in modeling complex organizational workflows. These workflow models will be maintained by a model management system in the WFMS. During enactment, the workflow engine presents work items to users according to the model specifications. When an exception occurs, the system will suggest relevant models from the modelbase to the user who may select an appropriate model for the situation and make modifications as necessary. During analysis, a manager may make certain changes to the conditions of the workflow to simulate certain scenarios that are of interest. For example, the impact of newly trained workers on the total workflow performance may be simulated by incorporating rework cycles in a portion of the workcases. In process redesign projects, an analyst may create alternative workflow models with new task sequences, resources and conditions and apply simulation studies with these models to find out the performance of those new designs. In sections 5.1 and 5.2,

Proceedings of The Thirtieth Annual Hawwaii International Conference
on System Sciences ISBN 0-8186-7862-3/97 $17.00 © 1997 IEEE

1060-3425/97 $10.00 (c) 1997 IEEE

we discuss how DWM can be applied in workflow enactment and workflow analysis respectively to support these uses.

5.1. Workflow Enactment

In workflows of high task variety, the number of possible task sequences can quickly overwhelm the human memory. In these workflows, WFMS can be useful as a memory support system for organizational procedures. The success of a memory support system relies on the efficacy of its storage and retrieval mechanism. The flexible structure of DWM enables parts of workflow to be added, changed or replaced. We have also proposed two ways of indexing submodels (task sequences). One way is to use typical factors such as those listed in Fig. 2 to index different task sequences that can be employed for accomplishing goals. A second way is to use prototypical tasks such as those in Fig. 3 to group similar tasks into categories. By including knowledge about the characteristic of the task, and the conditions under which a task (sequence) should be applied, this model structure can be used by a case-based system to provide intelligent storage and retrieval of workflow models [19]. In WFMS where workflow templates can be stored and manipulated in an *ad hoc* fashion by end-users [32], a case-based system built on DWM models can provide the necessary storage and retrieval capability for flexible support of workflows of high task variety.

5.2. Workflow Analysis

Workflow analysis is needed for day-to-day operations, as well as for workflow planning and design purposes. Process managers at various levels need workflow data for monitoring and short-term planning. Together with process analysts, they are also involved in workflow modeling, design and simulation to achieve process improvement and innovation. DWM facilitates analysis by structuring workflow models into four perspectives, i.e., functional, behavioral, organizational and informational. We posit that questions that are of interest to managers and analysts can be expressed using the relationships defined in these perspectives. Some questions may be answered using the models themselves. Static measures such as 'how many hand-offs are required on a workcase with these characteristics', or 'how many actors are employed in this workflow' can be answered by analyzing the models from the organizational perspective. These high level measures are generally more useful for process design purposes.

Most questions, however, have to be answered with dynamic measures that require simulations with workflow transaction data. Fig. 8 lists examples of questions concerning workflow by users with different responsibilities. Question 1, for example, can be answered by running a simulation of the workflow with the desired priority assigned to T in the organizational model. The effectiveness of that change can be measured by grouping transaction records of task T with the wait-status for R at regular time intervals and averaging the results. The side-effects of that change to the performance of the workflow can also be analyzed, e.g., from the organizational perspective for resource utilization, or from the informational perspective for throughput.

Unit supervisor
1.There is a long queue for resource R that task T needs, should we change T's priority so that it can be completed in two hours? 2.There is a bottleneck at processing center C. How can we alleviate it?
Departmental manager
3.What were the percentages of the different types of workcases handled in the last month? How well did the procedures for each type of workcase perform? 4.We expect a surge in volume in the next two months. Do we need to hire some temporary workers?
General manager
5.If we automate this task T, what will be the change in the performance of all associated processes? 6.If we change job designs and start recruiting employees with different skill sets, what would be the impact on the performance of the workflow?

Fig. 8 Different users have different questions about workflow

Fig. 9 shows the architecture of a workflow simulation system which is similar to that of Cassandra+, a system that integrates a simulation system with a temporal database management system [3]. A workflow design interface provides a user-friendly language and tools for inputting workflow specifications. The specifications are translated by a workflow language interpreter into an internal format which are then stored in a workflow model base maintained by a model management system. The model base contains model data that are structured according to our framework. A workflow query interface accepts queries from a user in the form of parameters specified within the model framework and allows users to specify the workflow models to which those parameters apply. The parameters include input specifications (e.g. distributions of different kinds of workcases over time), instantiations of workflow elements and specific measures to track. A controller manages the query activities among the query interpreter, model management system, the simulator, the workflow transaction manger and the workflow analyzer. The query interpreter translates the query into requests for the model management system to instantiate the appropriate workflow models and pass them on to the simulator. The interpreter also passes the input specifications and measures to the simulator. The simulator creates workflow transactions based on the input specifications and workflow models and passes them to the workflow transactions manager which

Proceedings of The Thirtieth Annual Hawwaii International Conference
on System Sciences ISBN 0-8186-7862-3/97 $17.00 © 1997 IEEE

maintains them in a workflow transactions database. The workflow analyzer provides tools for analyzing workflow models as well as workflow transactions.

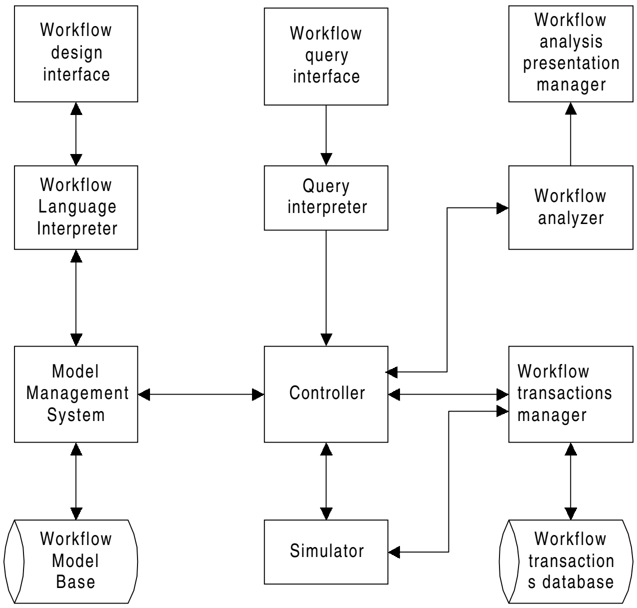

Fig. 9 Architecture of a simulator for a WFMS

DWM provides the basic framework for the structure of the modelbase as well as the workflow transactions database in this simulation system. To build such a simulation system, we will need to develop a specification language or map an existing one to this model structure. The specification language will also be the basis for developing a query language for running simulations.

6. Related work and future research

We have described DWM, a framework for modeling workflows that provides the flexibility, expressivity and structure necessary for high variety, sequential workflows. Our work has drawn from workflow modeling theory and techniques proposed by other researchers. In this section, we discuss their contribution and how DWM differs from their work.

Our approach to organizing the task hierarchy in the functional perspective is similar to the object-oriented approach in the Process Handbook project [22]. In addition to applying specialization for constructing a goal hierarchy, they have also suggested that certain characteristics of the process may be used to 'bundle', i.e., categorize, alternatives. The project has categorized various types of dependencies between tasks as a means of generating alternative coordinating mechanisms for a process. In this paper, we have identified a list of factors that may be used to index the alternative task sequences for achieving a goal. We have also included prototypical tasks as another means of further structuring of the model.

Our approach draws from the goal-based approach where tasks are organized as frames with the following attributes: goal, preconditions, effects, decomposition, control and plan rationale, agents, constraints [11]. In the goal-based approach, inference rules are applied to goals for planning task sequences. Our approach differs from that approach by providing a framework that distinguishes the four perspectives of tasks in a workflow and organizes task sequences into flexible goal hierarchies for indexed retrieval.

Our concept of task sequences draws from the model of organizational work proposed by Gasser [16]. In Gasser's model, task chains describes the production sequence for an object or event. Task chains intersect and must be coordinated for the production of work. Due to its identification with event outcome and the requirement for coordination, task chains are a natural unit for describing workflows. Gasser's study also showed that task sequences may be part of multiple processes. By organizing workflow models with task sequences as building blocks, DWM implements Gasser's view of task sequences to facilitate workflow model management and reuse. This organization also enables workflow simulations to take into account the effects of multiple process goals of each task sequence.

We have presented an overview of DWM, a framework for modeling workflows. DWM aims to satisfy the requirements of high variety, sequential workflows by providing high level packaged task templates for composing task sequences, a MOP-like structure to organize the task sequences into flexible workflow models and four perspectives into a workflow model geared toward different users' interests. We have illustrated these features using a case study of an interlibrary loan process. Future research will focus on further development of the framework, including identification of prototypical tasks, construction of indices for model retrieval, and development of a model representation, a modeling language for specification and query as well as a methodology for employing the framework in organizations. To support simulation, a set of workflow measurements will also be identified.

Acknowledgments

The authors would like to thank Rhoda Bilansky for providing the data for the interlibrary loan case study. This research has been sponsored by the Systems Research Center at Boston University.

Bibliography

[1] K.R. Abbott and S. K. Sarin. Experiences with workflow management: issues for the next generation. In *CSCW '94 Proceedings*, Chapel Hill, NC, October 1994, 113-120.

[2] D. Amrhein. The workflow software market: Choosing the right tool for the job. *Imaging Business*, Aug. 1995, 27-34.

[3] P. Balasubramanian and A. Tuzhilin. Using query-driven simulations for querying outcomes of business processes. *Decision Support Systems*, 16, 1996, 275-295.

[4] P. Barthelmess and J. Wainer. Workflow systems: a few definitions and a few suggestions. *Proc. of the COOCS'95*, Milpitas, CA, November 1995, 138-147.

[5] R. W. Blanning et al. Model management systems. In *Information Systems and Decision Processes*, E. A. Stohr and B. Konsynski (eds.), IEEE Computer Society Press, Los Alamitos, CA, 1992, 205-252.

[6] A. Bharadwaj, J. Choobineh, A. Lo and B. Shetty. Model management systems: a survey. *Annals of Operations Research*, 38, 1992, 17-67.

[7] D.P. Bogia and S. M. Kaplan. Flexibility and control for dynamic workflows in the wOrlds environment. In *Proc. of the COOCS'95*, Milpitas, CA, November 1995.

[8] G. Bracchi and B. Pernici. The design requirements of office systems. *ACM TOIS*, 2, 1984, 151-170.

[9] Coopers and Lybrand International Center of Excellence for Business Modeling and Analysis. *What is SPARKS?*

[10] N. Craven and D. Mahling. Goals and process: A task basis for projects and workflows. *Proc. of the COOCS'95*, Milpitas, CA, November 1995, 237-248.

[11] B. Croft and L.S. Lefkowitz. Knowledge-based support of cooperative activities. In *Readings in Distributed Artificial Intelligence*, A.H. Bond and L. Gasser (eds.), Morgan Kaufman, San Francisco, 1988, 599-605.

[12] B. Curtis, M. I. Kellner and J. Over. Process Modeling. *CACM*, 35:9, 1992, 75-90.

[13] T. H. Davenport. *Process Innovation : Reengineering Work Through Information Technology*. Boston, Mass. : Harvard Business School Press, 1993.

[14] C. Ellis and G. Nutt. *The Modelling and Analysis of Coordination Systems*. University of Colorado at Boulder Department of Computer Science Technical Report CU-CS-639-93, January 1993.

[15] C. Ellis and K. Keddara. *Dynamic Change Within Workflow Systems*. University of Colorado at Boulder Department of Computer Science Tech. Rep. CU-CS-667-93, August 1993.

[16] L. Gasser. The Integration of Computing and Routine Work. *ACM TOIS*, 4, 1986, 205-225.

[17] D. Georgakopoulos, M. Hornick and A. Sheth. An overview of workflow management: from process modeling to workflow automation infrastructure. *Distributed and Parallel Databases*, 3, 1995, 119-153.

[18] S. Joosten. Trigger modelling for workflow analysis. *Proc. CON'94: Workflow Management, Challenges, Paradigms and Products*, Oldenbourg, Vienna, 1994, 236-247.

[19] J. Kolodner. *Case-based Reasoning*. San Mateo, CA: Morgan Kaufmann Publishers, 1993.

[20] M.M. Kwan and P. Balasubramanian. *Workflow Support for Organizational Processes: A Framework and Research Agenda*. Boston University School of Management Working Paper 96-24, May 1996.

[21] J. Lee. Goal-based process analysis: A method for systematic process redesign. *COOCS'93*, 1993, 196-201.

[22] T.W. Malone, K. Crowston, J. Lee and B. Pentland. Tools for inventing organizations: toward a handbook of organizational processes. In *Proc. of the 2nd IEEE Workshop on Enabling Technologies Infrastructure for Collaborative Enterprises*, Morgantown WV, April, 1993.

[23] R.T. Marshak. *Selecting and Implementing Workflow: A Roadmap for Successful Business Process Design and Workflow Applications*. Boston, MA: Patricia Seybold Group Inc. April, 1995.

[24] R. Medina-Mora, T. Winograd, R. Flores and F. Flores. The action workflow approach to workflow management technology. In *CSCW '92 Proceedings*, Nov. 1992, 281-288.

[25] P. Mi and W. Scacchi. *A Meta-model for Formulating Knowledge-based Models of Software Development*. University of Southern California IOM0 Department Technical Report 94-16, August, 1994.

[26] N. Palmer. Business process simulation and modeling: An introduction and survey of tools. *Enterprise Reengineering*, Jan/Feb 1996, 25-26.

[27] C. Perrow. A framework for the comparative analysis of organizations. *Amer. Soc. Review*, 32, 1967, 194-208.

[28] S.K. Sarin, K. R. Abbott and D. R. McCarthy. A process model and system for supporting collaborative work. In *Proc. Conference on Organizational Computing Systems* (COOCS '91), November 1991, 213-224.

[29] R. Schank. *Dynamic Memory: A Theory of Learning in Computers and People*. New York: Cambridge Univ. Press, 1982.

[30] B. Singh and G. L. Rein. *Role interaction nets (RINs): a process description formalism.* Tech. Rep. CT-083-92. Microelectronics and Computer Technology Corp., Austin, Texas, 1992.

[31] L. A. Suchman. Office procedure as practical action: models of work and system design. *ACM TOIS*, 1, 1983, 320-328.

[32] K.D. Swenson. Visual support for reengineering work processes. *COOCS'93*, 1993, 130-141.

[33] J. Thompson. *Organizations in Action*. New York: McGraw-Hill, 1967.

[34] Workflow Management Coalition. *Glossary - A Workflow Management Coalition Specification*. [web page] November, 1994; http://www.aiai.ed.ac.uk/WfMC/. [Accessed April 10, 1996].

Proceedings of The Thirtieth Annual Hawwaii International Conference on System Sciences ISBN 0-8186-7862-3/97 $17.00 © 1997 IEEE