CIMFLOW: A WORKFLOW MANAGEMENT SYSTEM BASED ON INTEGRATION PLATFORM ENVIRONMENT

Haibin Luo Yushun Fan

State CIMS Engineering and Research Center, Dept. of Automation, Tsinghua University
Beijing, P.R. China. 100084 T. 86-10-62789650 F. 86-10-62770351
E-mail: lhb74@263.net fan@cims.tsinghua.edu.cn

Abstract - *In this paper, a workflow management system called CIMFlow building on the CIMS Application Integration Platform (CIMS AIP) is discussed. The distributed system architecture and the components of CIMFlow are given. Also the primary operation process based on message-driven is presented. CIMFlow provides a solution for modern enterprises to implement business process management and business process control. It makes Business Process Reengineering more feasible so that the enterprises will be more competitive.*

1. INTRODUCTION

Manufacturing industry has passed through the phases of mechanization and automation. In the 1980s, it entered the integration period. CIM concept and the application of CIMS technology have greatly improved the competitive ability of manufacturing enterprises. With the development of network and database technology, Concurrent Engineering, Agile Manufacturing, Virtual Manufacturing and Business Process Reengineering have become the focus of the engineering research. Future manufacturing industry will be greatly influenced by these research achievements [1].

Workflow technology, a key technology for modern enterprises in implementing the management and control of business processes, has provided an integrated framework for the workflow management from design to implementation. In a workflow management system, a set of integrated and interoperable software tools are used to support the whole management process. After a decade of evolution, workflow technology has become matured. A lot of research institutions and organizations including Workflow Management Coalition (WfMC) have done significant work on the theory and method related with workflow [5][7]. Workflow products supplied by a number of software companies also appeared in the market [6][8]. These products are based on different models, different architectures, different infrastructures and can manage different types of business processes in telecommunication, health caring, manufacturing, banking and so on.

In this paper, a workflow management system called CIMFlow building on the CIMS Application Integration Platform (CIMS AIP) is introduced. The following sections of the paper are organized as follows. In section 2 the framework architecture of CIMFlow is presented and the functions of each component constructing the system are discussed. In section 3 the operation mechanism based on the communication of Operation Administration Agents (OAA) of the CIMS AIP are explained and the coordination scenario of all the components in CIMFlow are described. At last a conclusion is given and the future work is presented.

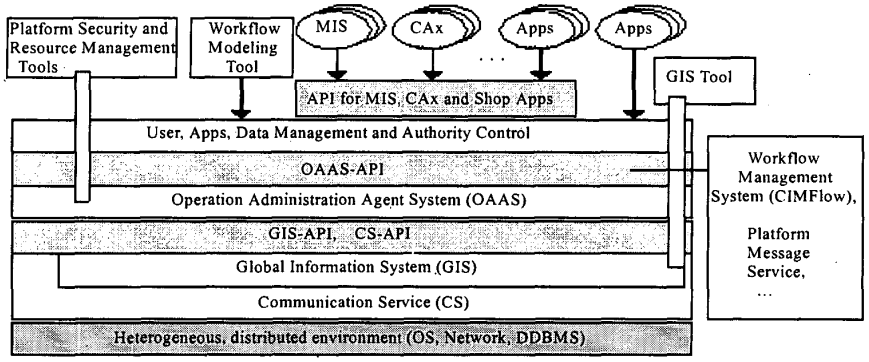
2. THE ARCHITECTURE OF CIMFLOW

CIMS Application Integration Platform is a software platform which is developed for solving the problems in CIMS implementation such as long implementation time, difficulty in the systems integration,

0-7803-5670-5/99/$10.00 ©1999 IEEE

heterogeneous information sharing between different applications and so on [3]. The platform provides application developing tools, application prototype systems, application integration interface, global information definition and maintenance tool, and some integrated applications to support the CIMS implementation in manufacturing enterprises. CIMS AIP constructs a software integration environment to realize the information transparent access and management in the enterprise. It makes the process of CIMS implementation more efficient and more reliable. The architecture of CIMS AIP is presented in figure 1.

As a part of the CIMS AIP, CIMFlow fulfills the application coordination on the platform. Its communication infrastructure is based on the Operation Administration Agent System (OAAS) of the platform [2]. The distributed agents of OAAS can communicate with each other under the heterogeneous computer environment. CIMFlow consists of a workflow modeling tool, an administration tool, an user interface, a workflow engine and the supporting database. The system architecture is presented in figure 2.



Note: Apps means Applications

Figure 1. The CIMS AIP Architecture

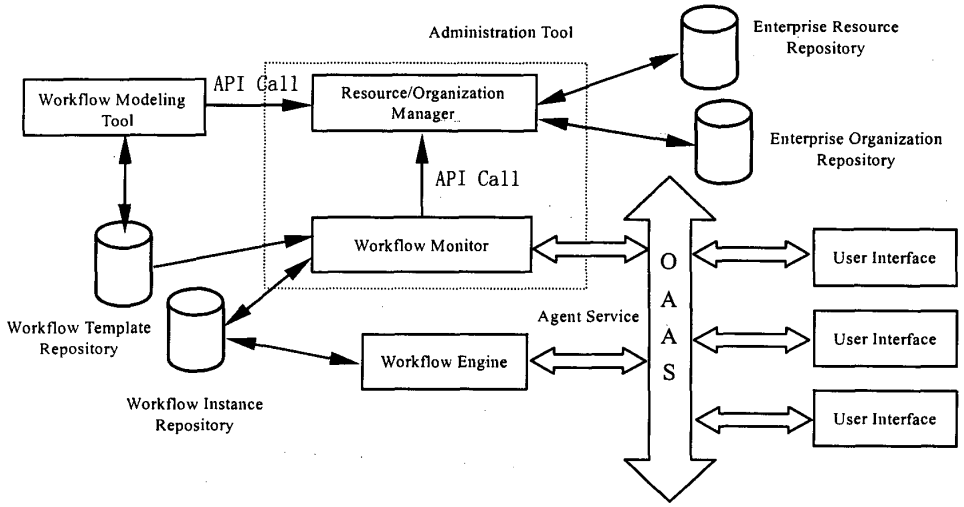


Figure 2. The system architecture of CIMFlow

2.1 Workflow modeling tool

Workflow modeling tool is used to build the workflow model of business processes. By using the graphical interface provided by the modeling tool, user can easily and quickly develop the workflow applications with the point-and-click action. The design of workflow model is the first step to implement the workflow management. User constructs the workflow model with the elements provided by workflow modeling tool and define the detail properties for each element in the tool box. The user built workflow model now describes the structure properties of the process, it is saved as the integrated business process static model. It is necessary to instantiate the static model before it can be executed by workflow engine.

The elements in the workflow modeling tool box can be classified into two categories: node and directional arc. These two kinds of elements represent the basic activity and the relationship between activities in the business process respectively. The node stands for an executable activity which can be run automatically or manually. The directional arc from one node to another represents the execution sequence and relationship of the two activities. To assure the correctness and completeness of the model, we extend the content of the two primary elements. The extended model is depicted in table 1.

Workflow modeling tool also provides an interface between the object-oriented entity operation and the relational database management system. It is used to save and manage the model designed and implemented by object-oriented method in RDBS (Relational Database System) such as SQL Server. The interface not only makes the legacy database applications available in the new management fashion, but also takes advantages of the power of O-O (Object-Oriented) technology [4] and RDBS technology both.

2.2 Administration tool

Administration tool consists of two components: Resource/Organization manager and workflow monitor.

Resource/Organization manager is responsible for the maintenance of enterprise resources and organizations information which is essential for the operation of workflow instances. The resource and organization views are built with a tree structure. The Resource/ Organization tree contains the categories of available resources and persons (roles)and also the detail item information such as a machine tool or a person. The enterprise user can easily construct his Resource/Organization repository with the provided tool. Resource/Organization manager provides a set of APIs to be called by other parts of CIMFlow including workflow modeling tool and workflow monitor. This set of APIs will be used in application coordination function.

Table 1. The elements in the workflow model

Primary Elements	Extended Elements	Description
Node	Activity	A simple task performed by a person or an application in a continuous period of time. It is the basic unit of the model.
	Sub-process	A compound task which consists of activities and sub-processes. It provides mechanism for building a hierarchical workflow and supports the top-down modeling process.
	Start/End Mark	A mark serves as the only entry/exit point of the model.
	Synchronization Mark	A mark implements the "AND" function. It requires all of its preceding activities to be finished and then activates its post-activities.
Directional Arc	Non-conditional Arc	An arc without any transition conditions. It means the simple sequential relationship.
	Conditional Arc	An arc with a transition function which can return two value: "True" and "False". Only when the function returns "True", the transition is enabled.

In the build-time, workflow modeling tool obtains the resource and role categories which are needed by the workflow modeling tool through calling the APIs. Since many workflow instances will derive from one workflow model in the run-time, it is usually unnecessary to access the detail information of resource items and persons during the build-time.

In the run-time, workflow monitor calls the APIs to support the workflow instances running. The run-time APIs make the details of resource items and persons accessed including the static information such as ID number, name, type and the dynamic information such as the status of entities. Through these APIs called by other applications, Resource/Organization manager accomplishes the schedule of resources and roles and changes each entity status dynamically. The APIs separate the complex function implementation from the simple function call. By this kind of cooperation, the power of distributed applications can be utilized to large extent.

Workflow monitor is used to monitor the state of each running workflow instance and the activities in it. Through the graphical interface, workflow monitor provides the situation of workflow instances for the enterprise administrator so that the business processes will be controlled effectively.

2.3 Workflow engine with OAAS

Workflow engine is the core of CIMFlow in the run-time. It performs such actions as start workflow instances, schedule activities in different workflow models, navigate the right route in a workflow instance, generate worklist for each participant, and maintain the log files, etc. Since CIMFlow is a part of CIMS AIP, we use a centralized workflow engine, that is, on the CIMS AIP there is only one workflow engine.

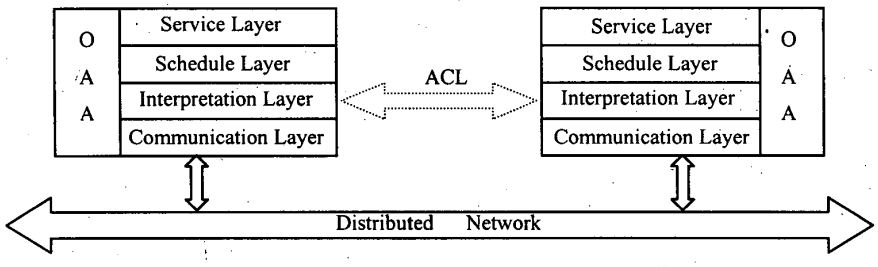
The distributed property of application determines that the workflow engine, whatever centralized or distributed, must lie on a distributed communication system infrastructure. A typical example of the communication infrastructure is CORBA which is brought forward by OMG. The infrastructure of CIMS AIP is composed of a group of Operation Administration Agents (OAA) which have such characteristics as below:

(1) Peer to peer distribution: The OAA not only provides the communication service for local applications, but also cooperates with remote agents to fulfill the requirements of applications.

(2) Client/Server dynamic connection: Different applications and different OAAs are connected together as C/S mode. Each OAA can not only send a request as a client to other OAAs, it can respond to the client OAAs to serve as a server also. The client and server are only connected when the request is sending and the service is responding. This dynamic connection is realized based on socket communication.

(3) Hierarchical structure: In the OAAS, a control agent is setup to manage and coordinate all the other agents. This kind of structure can easily administrate the agents information.

The structure model of OAA is given in figure 3.



Note: ACL means Agent Communication Language

Figure 3. The structure model of OAA

In the whole architecture of CIMS AIP, the OAAS is located in the enable layer which supports the interoperability of platform applications such as CIMFlow. We encapsulate the implementation details of OAA and provide a set of OAA APIs: The distributed applications on the platform can use these APIs to get transparent communicate service. The applications can send parameters and return values without considering the trivial implementation details of the communication system. Currently we provide two versions of OAA APIs which are based on C++ and Java respectively. These APIs are just the implementation fundamental of workflow engine and user interface in CIMFlow.

Workflow engine utilizes the C++ version of OAA APIs to build its message queue and receive the requests of workflow applications such as starting a workflow instance, inquiring a user's worklist, etc. For each message, workflow engine will perform necessary operations to fulfill corresponding functions. Workflow engine also sends requests to Resource/Organization manager to acquire the resource and role information which is essential for instances running. The operation of workflow engine will be described in section 3.

2.4 User interface

User interface is a component of CIMFlow responsible for the interaction between the workflow participators and workflow engine. User interface should be simple, user friendly, and could be distributed across the CIMS AIP. So Java language interface is a desirable solution considering its platform independence characteristic. Figure 4 shows the primary interface.

In figure 4 each button is used to send a message to workflow engine.

"login"---- register to workflow engine in order to get the worklist which will be displayed in the listbox on the left.

"details"----inquire the detail information about some work item in the worklist to workflow engine.

"start"----notify workflow engine that a task has been started.

"finish"----notify workflow engine that a task has ended successfully so that workflow engine will continue the next task in the instance.

"suspend"----notify workflow engine to pause the execution of a task because the task has something wrong.

"refresh"----notify workflow engine to get the updated worklist.

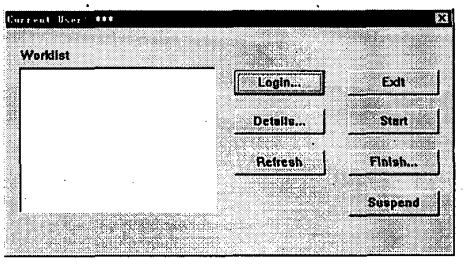


Figure 4. User interface

Workflow engine receives the messages sent by user interface through OAAs and then perform the proper operations to send back the results. From the viewpoint of Client/Server, user interface is a pure client application which always sends requests to server and never receives a request. Thus user interface should be designed portable.

3. CIMFLOW IMPLEMENTATION

After the workflow modeling and the instantiation has been completed, the administration tool can submit the instance to workflow engine, and workflow engine then run the workflow instance automatically. Now the whole workflow management system is in the run-time, in which workflow engine is the kernel of the whole system, and user interface, administration tool and workflow engine cooperate with each other to execute the workflow instance.

3.1 Message-driven execution mode

The execution of a workflow instance is driven by messages. This is determined by the distribution characteristic of CIMFlow. As we have stated before, the transfer of messages is based on the low-level

communication service of OAAS on CIMS AIP. Table 2 gives the list of the messages received and sent by workflow engine.

Table 2. Message list

Message Name	Sender	Receiver	Primary Parameters	Results Returned
Submission of a workflow instance	AT	WE	Workflow instance ID	Success/Fail
User login	UI	WE	User information (ID, password...)	User's worklist
User query	UI	WE	Workitem ID	Workitem information
Activity started	UI	WE	Workitem ID	Success/Fail
Activity finished submission	UI	WE	Workitem ID	Success/Fail
Resource/Organization request	WE	AT	Workitem ID	Success/Waiting/Fail
Resource/Organization allocation	AT	WE	Workitem ID	——
Activity finished notification	WE	AT	Workitem ID	——

Note: AT: Administration Tool WE: Workflow Engine UI: User Interface

It can be seen from the table, a workflow instance is first submitted by administration tool while the progress of the instance is driven by user interface. This means that participators of the workflow control the execution of the activity, in this process, the workflow engine just records some important information related with the activity execution according to the messages sent by the user (e.g., the start and finished time of an activity). This kind of information reflects the users' operation, and can be used for the activity evaluation and business process reengineering.

Besides the submission of a workflow instance, the administration tool also responds to the "Resource/Organization application" message send by the workflow engine. It allocates the right resource items and persons to the activity, then lock them as "busy" status in case there are other activities which also want to hold them at the same time. For those items and persons that are busy now, the administration tool will return the "Waiting" message to workflow engine, and set up a waiting queue. Once the items or the persons are released, the administration tool will inform workflow engine and allocate the resources or persons to the first activity in the waiting queue.

3.2 The transition of activity states

Each activity instance has several states according to different phases it experiences during the execution. Clarifying these different states and making them available to the enterprise administrators is important. It can ensure that every workflow instance is under effective control.

We define five states for the run-time lifecycle of an activity instance:

- "Initial": the activity has been instantiated, but has not been run yet;
- "Active": the activity is being executed by some person or an application;
- "Done": the activity has been finished successfully;
- "Exceptional": there are some errors during the activity execution, e.g., failure of machines;
- "Terminated": the activity is terminated because some errors have happened.

The transition diagram of these five states is shown in Figure 5:

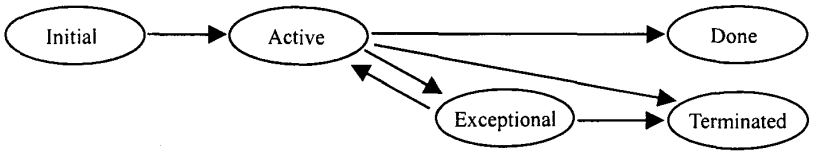


Figure 5. The state transition diagram

Table 3. List of messages driven the state transitions

Transition of the states	Driven messages (events)
Initial→Active	The performer sends "Activity started" message from user interface.
Active→Done	The performer sends "Activity finished submission" message from user interface and fill in the feedback information form of the activity.
Active→Exceptional	A. The performer sends "Error" message from user interface; B. The administrator sends "Error" message from administration tool.
Active→Terminated	The administrator sends "Terminate" message from administration tool.
Exceptional→Active	The administrator sends "Restart" message.
Exceptional→Terminated	The administrator sends "Terminate" message from administration tool.

State transition is triggered by the event (message), the messages and behind those messages (events) reflect participators' actions. Table 3 is the list of the messages that cause the state transitions.

3.3 The lifecycle of a workflow instance

For each workflow instance submitted, it will occupy one of the threads of workflow engine at one moment and last some time. Each state transition for every activity in the workflow instance will repeat this procedure. The entire workflow engine's threads and all the time occupied by one workflow instance makes up its lifecycle.

Once a workflow instance is submitted (i.e. workflow engine receives "Submission" message from administration tool), workflow engine will create a new thread and act as follows:

(1) Open the related database table according to the message parameters, change the state of the workflow instance to "Active" and record the startup information (time, person, etc.);

(2) Open the database table and find the start activity of this workflow instance, i.e., get the entry of the instance;

(3) Find all the subsequent activities according to the information stored in database;

(4) Call the corresponding function for each subsequent node. Below is the procedure of running the basic activity. First, send the "Resource Request" and "Person Request" messages to administration tool.

- If successful, the workflow engine sends a short words to the performers (just like an E-mail, supported by OAAS of CIMS AIP), informing them to check their worklists, and this thread is ended.
- If failed, the workflow engine calls the error handling function.
- If the resource items or the persons are busy and the activity has to wait, the workflow engine ends the thread. This process will go on once administration tool sends the "Resource /Person Allocation Success" message back.

After the four steps are finished, the thread is also finished. This is the first process for the execution of the workflow instance.

When the performer receives the short words from workflow engine, he can use user interface to login into

the workflow management system and interact with workflow engine. The user can get his worklist from workflow engine, and check the detail information about one work item. This kind of actions won't change the state of the activity and we call them "Read Out" actions. The user can submit the "Start", "suspend" and "Finished" messages to workflow engine. This kind of actions will change the state of the activity and the changed results will be recorded in the log file. We call them "Write In" actions.

Here we focus on workflow engine's actions which respond to the "Activity Finished" message sent by user interface, because these actions not only change the state of the activity, but also invoke the navigation function of workflow engine, i.e. they will influence the execution of subsequent activities. When workflow engine gets "Activity Finished" message, it will first change the state of this activity to "successful", and record the maintenance information, then workflow engine will send a "Activity finished notification" message to administration tool, this action will cause administration tool to free all the resource items and roles which the activity has held. Next, workflow engine will open the database, find the subsequent activities and follow the step 4 we have stated above.

The coordination and interaction between workflow engine, user interface, and administration tool enables the workflow instance progress step by step. Workflow engine keeps a record for the running process of the instance, while administration tool records the information about the allocation/release of the resource items and roles. This information is very important for the enterprise to evaluate the business process and carry out business process reengineering.

4. CONCLUSIONS

CIMFlow gives a typical example for the application of workflow technology in the CIMS environment. All its components and the corresponding supporting systems of CIMFlow provide the enterprise an feasible solution for the distributed business environment. The workflow management system makes different tasks cooperate with each other effectively. The administrator can control the enterprise performance in a global, transparent style. These are essential for the enterprise to survive in the competitive environment and fulfill the goal of business process reengineering. For it is not designed for a specific kind of manufacturing system, CIMFlow can be adoptable for any industry. The main adoptative procedure is as follows:

(1) Build the information infrastructure which includes installing the OAAS;

(2) Deploy the components of CIMFlow throughout the enterprise;

(3) Build the workflow model of the business process by using the workflow modeling tool of CIMFlow;

(4) After testing, submit the workflow for running.

For the future work, we consider the following aspects are necessary and important:

(1) Improve the workflow modeling tool to support the dynamic, adaptive workflow;

(2) Implement distributed workflow engines by coordination mechanism;

(3) Make the workflow management system robust to different kinds of exceptions;

(4) Provide a set of simulation and analysis tools to support business process reengineering more effectively.

REFERENCES

[1] Shi Wei, Wu Cheng, Fan Yushun. The Research of Workflow Technology in CIMS Application Integration Platform, *Journal of Tsinghua University*, 38(8): pp.125-128, 1998.(In Chinese)

[2] Xiong Rui, Fan Yushun, Wu Cheng. The Research of Operation Administration System Based on Agents in CIMS Application Integration Platform. *Information and Control*, 27 (5): pp.351-358 , 1998.(In Chinese)

[3] Fan Yushun, Wu Cheng, Shi Wei. The Development and Trend of the Application Integration Platform. *CIMS*, 8 (5): pp.3-8, 1997.(In Chinese)

[4] Jacobson I. , *The Object Advantage: Business Process Reengineering with Object Technology.* Addison Wesley, 1995.

[5] Workflow Management Coalition, *The Workflow Reference Model,* [WfMC1003] WFMC TC00-1003, 1994.

[6] Thomas M. , *The Evolution and Future of Workflow*, http://www.delphigroup.com/pubs , Delphi Consulting Group, 1996.

[7] Georgakopoulos D. , Hornick M. , and Sheth A. , An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure. *Distributed and Parallel Databases*, 3 (2): pp.119-154, 1995.

[8] Leymann F. , Roller D. , Workflow-based Applications, *IBM System Journal*, 36 (1): pp.114-120, 1997.