Multi - server Workflow Management System:
Business Process Automation and Task-tracking

3 H — 1

Shunsuke Akifuji*, Takashi Saito*, Yuichi Mori*, Hiroshi Majima** and Hiroshi Tsuji*

*Systems Development Lab. Hitachi, Ltd.   **Software Development Center, Hitachi, Ltd.

1. Introduction

Workflow management in client-server type groupware systems is a vital area of research. An enterprise which constructs a multi-server system will inevitably treats thousands of users working with client PCs. Here, we discuss two major problems associated with multi-server workflow management systems, and propose solutions to these problems. Our main concerns are the scalability and robustness of the system.

2. Multi-server workflow management system

The goal of any workflow management system is to achieve office automation and improve productivity for group work, such as loan processing, claims processing, and purchase requisitions [1]. Therefore, the functions of a workflow management system should include 1) routing electronic documents to the appropriate users described in a process flow definition, 2) recording the status of each process flow, for example, to monitor where the documents are at any given time, and 3) reporting the work load of each user in order to pinpoint bottlenecks in the defined process flow.

Most workflow management systems have a visual editor which describes work sequences and also a report generator which monitors the status of both the process flow and each user. The server dedicated to the distribution of electronic documents according to the stored process-flow definition. However, from the viewpoint of scalability and robustness, the workflow management system should also be capable of operating under multi-server environments. The major problems concerning this are as follows:

1) The way in which the flow definition is to be distributed among the servers.

The update of the flow definitions should be in accordance with the growth of that organization. A workflow system initially implemented as a single server system may later accommodate multiple servers. This could create the problem where one department may be responsible for one part of the process-flow definition while other departments may be responsible for the remaining parts. Therefore, it is necessary to design a means of distributing the flow definition among the servers.

2) The way in which electronic documents are to be tracked.

Under a multi-server environment, some of the servers may be out of service. Even in such a case, users may like to track the status of their workflow data in the deactivated servers, and to know in which servers their workflow data resides. Therefore, the way in which workflow data logs are to be stored and documents, tracked is problematic.

As a solution to these problems, we propose 1) a method defining the connection between two servers via a process-calling mechanism, and 2) a tracking agent which replicates log data and stores the replicated log data in server computers.

3. A method defining the connection between two servers

Each process-flow definition is described for a server computer. Then, the calling mechanism,(i.e., a process-flow definition executed by one server calling a sub-process flow definition executed by another) , prevents local changes in the *called* process-flow definition from affecting the *caller* process-flow definition. The mechanism is similar to a program function call.
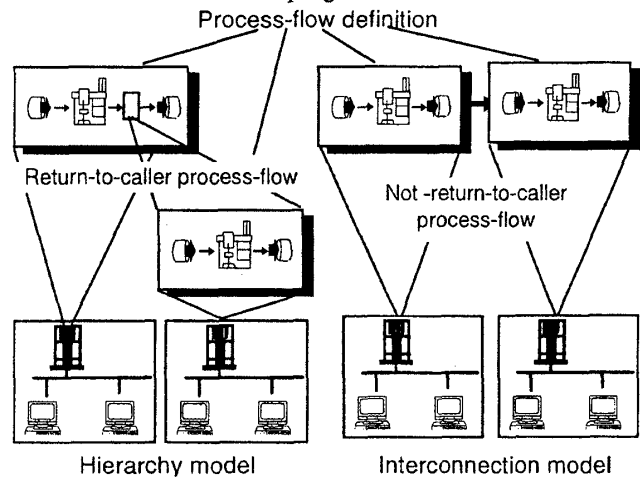

Fig. 1 Hierarchy and Interconnection models

Our calling mechanism allows two different models to be specified in the organization : a hierarchy model and an interconnection model (Fig. 1)[2]. The former provides a framework for describing the relationship between a section and a sub-section, and the latter allows the relationship between two sections to be described. We

call the flow definition the "*caller* process-flow definition" for the sender and the "*called* process-flow definition" for the receiver.

● Hierarchy model.

This is comparable with a sub program-main programs relationship. The called definition is similar to a sub-program whereas the caller definition is similar to the main program. In this model, there is sub-main relationship between the caller and the called definitions.

● Interconnection model.

This is comparable with the "goto" statement in a programming language. In this model, the called definition is connected to the caller definition.

The hierarchy model is suitable intra-organization use because the data or documents are processed in the subsection and returned to the section; the result is then checked. The interconnection model is suitable for inter-organization business flows among sections.

4. Tracking Agents

Each server includes a tracking agent. Each tracking agent communicates with other agents to gather and replicate workflow log data. Despite the communication policies which regulates replicated-data transfer between servers, the tracking agent can always access any workflow log data(Fig. 2).
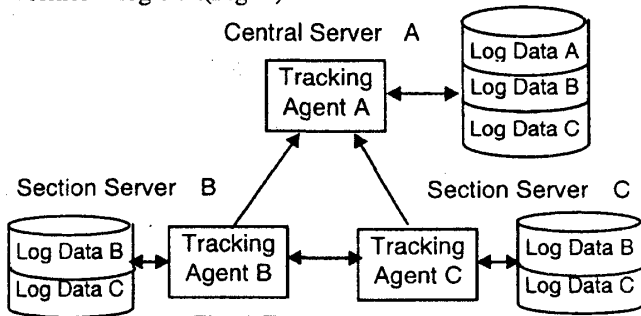

Fig. 2 Tracking Agents

Our tracking agents have alternative policies for the selection and exchange of log data. The sending policy is determined by the scale of the system and the purpose of tracking. If the purpose is to maintain consistent workflow status among the servers, all the log data are gathered in one central server. To respond quickly to a status query, each tracking agent periodically exchanges log data with other agents and stores the replicated log data in all servers. To reduce data traffic, other tracking agents are confined to sending only a part of the log data. Thus, our sending policy allows workflow system designers to customize response times and the scope of

tracking. Therefore, the tracking agents allow system scalability by allowing change in policy in accordance with the growth of the system.

5. Implementation

The system has a workflow manager engine, a multi-server transfer agent, and a tracking agent in the server. By referring to the process flow definition, the workflow manager engine delivers electronic documents to the appropriate users. The multi-server transfer agent sends and receives data among servers. Both electronic mail (e-mail) and reliable messaging software are used to transport data. The tracking agent communicates with the multi-server transfer agent under the replication mechanism supported by a database management system for sending and receiving workflow log data(Fig. 3).
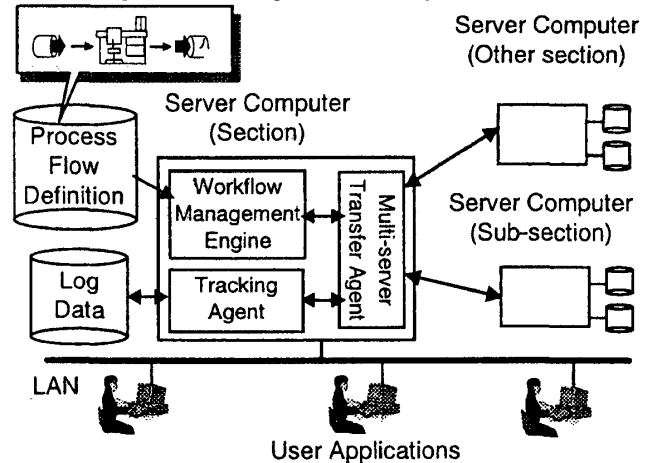

Fig. 3 Functional Component of the Multi-server Workflow System

6. Conclusion

We discussed the major problems with multi-server workflow management systems and proposed methods to solve them. To enable the flow definition to be shared among the servers, we proposed a method which defines the connection of two servers via a process-calling mechanism. We also proposed status monitoring which caters to both the purpose of tracking and the scale of the system. This facility is provided by tracking agents that can replicate log data and store them in all servers.

References

1) S. Akifuji, et al:A Survey of Workflow Management Systems, System Control and Information (in Japanese), Vol. 40, No. 5, pp. 203 - 208(1996).

2) T. Saito, et al:Workflow Management System "Flowmate" (5) -Multi-server- (in Japanese),Proc. 51st Annual Convention IPS Japan, Vol. 6, pp. 177-178(1995).