Communication and Scheduling Model Design of Distributed Workflow Management System

Jing Zhao

School of Science

Xijing University

Xi'an, China, 710123

Bin Wei

Key Laboratory of Network & Information Security of APF

Engineering College of APF

Xi'an, China, 710123

weibin82@126.com

Abstract—**The traditional OA system cannot meet the requirements of modern enterprise. Currently the widely used distributed workflow management system (DWFMS) has the problem of load balancing, which makes the enterprise efficiency unimproved. In this paper, on the study of the DWFMS, the model design of common communication transmission and scheduling are analyzed. Then the real-time scheduling strategy model is put forward and discussed, which based on feedback control. By using this model in DWFMS, the task can be controlled effectively, the load balancing performance of the system is improved effectively, and the stability of the system is ensured.**

Keywords—workflow; communication; scheduling model; controller

I. INTRODUCTION

With the complex and frequent requirements of modern enterprise business process, the development of Web system has put forward more severe tests, which make the development of Web system face challenges. The traditional OA system cannot meet the business requirements of the current enterprises. Therefore, the workflow management coalition (WFMC) propose the concept of workflow management system. However, the problem of load balancing affects the system performance seriously, which makes the enterprise's business efficiency unimproved.

Distributed workflow technology is developed on the basis of centralized workflow technology. With the distributed workflow technology, DWFMS has good scalability and continuity, which can meet the requirements of enterprises and users better, and the main features are shown below: 1) Assist in completing the multi-department tasks; 2) As an enterprise application integration platform, it forms a complementary relationship with other business systems effectively; 3) The embedded workflow engine is transparent to the application end user.

The communication transmission between tasks is a key component of DWFMS. The proper communication transmission and task scheduling model can effectively improve the operation efficiency and the resource utilization of the server nodes, thus improving the performance of the distributed system.

II. DISTRIBUTED WORKFLOW MANAGEMENT SYSTEM

The WFMC propose a classic definition of workflow: all or part of it supported or automated by a computer. The workflow management system is to support a batch of professional set workflow, which is defined by the calculated process. It is used to support the definition, management and executive the workflow. It enable the business execution processes to achieve maximum efficiency.

The distributed workflow system is generally divided into 3 parts: the distribution of workflow architecture, the distribution of workflow engine and the distribution of workflow model. The workflow engine distribution is the core of distributed system. The structure diagram of DWFMS is shown in Fig. 1.
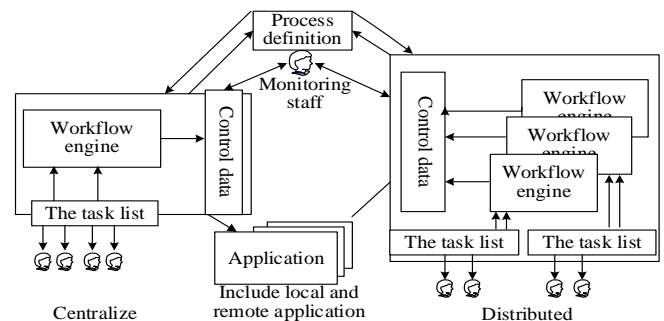

Fig. 1. The structure diagram of DWFMS

III. COMMUNICATION TRANSMISSION MODEL

The main information flows and control flows that need to be transmitted between DWFMS as follow: activity activation, sub process activation, real-time status of active instances, workflow related data transmission, data transmission, process model reads and writes. The WFMC proposed the following two kinds of communication transmission models.

Network model: in network model, a workflow instance within particular workflow service area can be used as a sub process within different region. And the data transmission can be implement with the application functions or the API interface. The network transmission model is shown in Fig. 2.
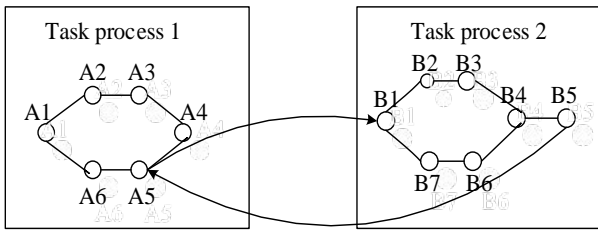
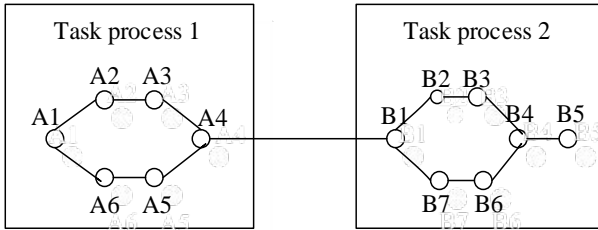Fig. 2. The network transmission model

Fig. 3. The chain transmission model

Chain model: a point A in process 1 can be connected to a point B in process 2, and the B can be any node. This model supports the single task transmission within heterogeneous environment, and then executing by the receiver without further synchronization. In reality, it can realize the data format conversion and the name mapping not only by an application program, but also the way of service to communicate with each other, such as API interface. Chain transmission model is shown in Fig. 3.

In general, the chain model can supported independently. There is no need to ensure synchronization, which is more suitable for the development trend of modern workflow system.

IV. THE DESIGN OF SCHEDULING MODEL

In order to achieve the goal of DWFMS load balancing, three factors need to be analyzed.

- Scheduling object: object oriented technology is used to analyze the scheduling objects. In order to view the analysis and communication later, the Unified Modeling Language (UML) modeling be used to graphically display.

- Condition of assignment: after the task queue enters the system, the assignment of tasks is mainly determined by the reference conditions, such as the authorization of the processing nodes at all levels in the audit system.

- The assessment methods of task execution node load: when the task has been assigned, the system may appear uneven load state, at this time for running load nodes or the nodes which in the queue waiting for processing, both need to evaluate the load efficiently, that can provide data to ensure system scheduling strategy initialization.

Suppose the system engine consists of several load task engines, node load information table and the load information

summary table by administrator. Inside of the load task engine and among the load task engine, node load information table and the load information summary table by administrator, the real-time system communication is implemented by messaging. The engine task scheduling process is shown in Fig. 4.
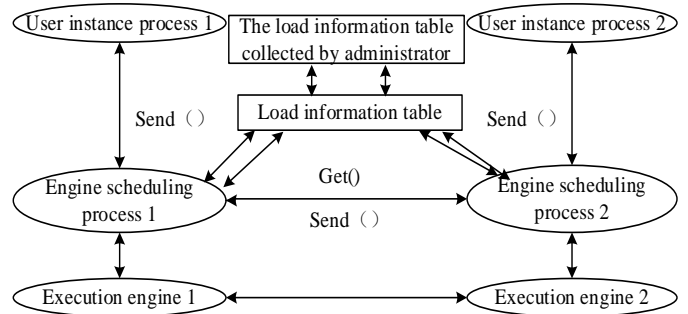

Fig. 4. The engine task scheduling process

V. THE REALIZATION OF SCHEDULING MODEL

A. The scheduling model based on MVC

In this paper, the MVC and the development based on SSH framework are adopted. The MVC and the SSH are briefly introduced as follows. The MVC is the design model of model-view-controller. It can implement the separation of display and data. A control layer is added between the view and the model, and the view can only operate the model by controller. The control layer is the data layer. Then the data update the model notifies the view to update itself. In a desktop program, the user can interact with the view directly. View events can be triggered by the event operations. Through the controller, the purpose of updating the model or data can be achieved. The data exchange model of MVC is shown in Fig. 5.
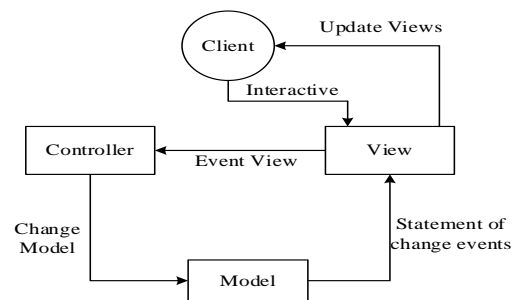

Fig. 5. The data exchange model of MVC

The SSH framework, also known as Strut+ Spring+ Hibernate.

The Strut framework: struts is a large Servlet, called ActionServlet, or a subclass of ActionServlet. which provides the components for the Model, View and Controller.

The Spring framework, spring is a lightweight inversion of control (IoC) and a container framework of aspect oriented

programming (AOP). It is noninvasive, and objects in Spring applications are not dependent on a particular class of Spring.

The Hibernate framework, Hibernate is an open source object-relational mapping framework. The very lightweight object encapsulation for JDBC, which make Java programmers to manipulate the database with object programming thinking. There are five core interfaces: session, session factory, transaction, query and configuration.

B. The realization of scheduling strategy model

The feedback control real-time scheduling model presented in this paper is shown in Fig. 6. The scheduling model consists of feedback controller, task dispatcher, task scheduler and PID scheduler.
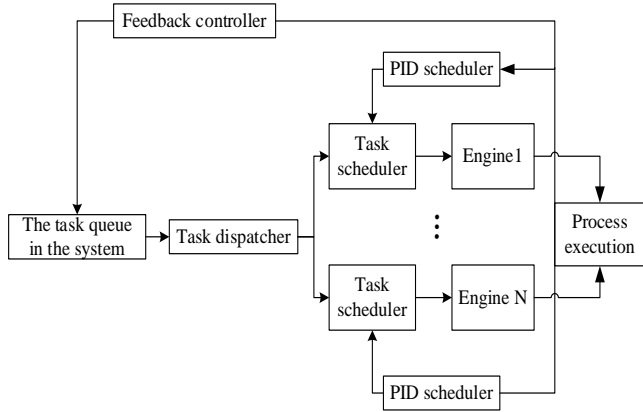

Fig. 6. The adaptive task scheduling model

In the scheduling model diagram, the feedback controller control the total processes. By adjusting the task flow, the engine load in the distributed workflow system is controlled and the problem of instantaneous overload is prevented. According to the load balancing of the process engine, the feedback controller real-time calculate the load state of each node and system, then controlling the system task according to the calculated results.

The PID controller uses the feedback control method to calculate and evaluate the parameters of the tasks which already in the system. Then the variable range of the system load is evaluated, which provides a reliable reference value for the next load change of the system, and the system load can keep at a high stable value.

The system feedback control strategy consists of feedback controller and PID controller. It is mainly used to predict and control the load evaluation and task execution time, then the stability of system load is realized. The main function of task allocator is to allocate the tasks that already entered system to the proper engine. In this way, the goal of the maximum efficiency of utilizing system resources is achieved and the efficiency of the mission is improved.

The main role of the task scheduler is to determine the priority level of task execution in the process engine based on the load priority table. It uses PID controller to calculate the node load state. Aim at the node current state, the system

acceptance of load tasks is adjusted by setting the effective threshold. The implementation of scheduling strategy is shown in Fig. 7.
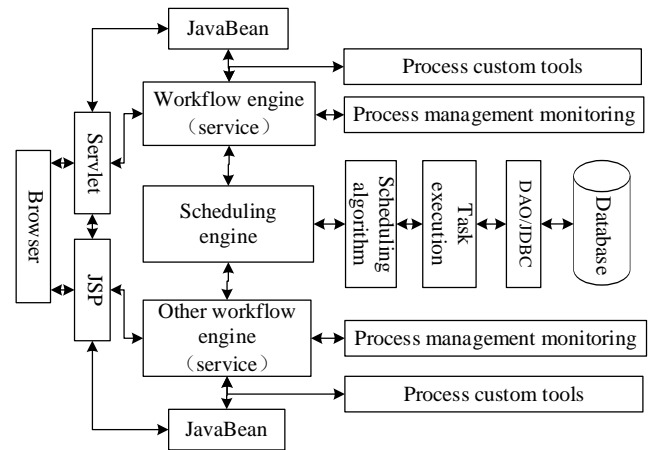

Fig. 7. The implementation of scheduling strategy

VI. CONCLUSION

Based on the in-depth study of DWFMS, this paper discusses the problems of workflow technology in practical application. Based on the common model analysis of communication transmission and scheduling, the real-time scheduling strategy model based on feedback control is proposed and discussed. By using this model in DWMS, the task can be controlled effectively, the load balancing performance of the system is improved effectively, and the stability of the system is ensured.

REFERENCES

[1] Tan, Y. Y., Wang, R., Fan Y. S., et al. Adaptive scheduling method for real-time tasks in distributed workflow [J]. Computer Integrated Manufacturing System, 2010,16(9):1887-1895.

[2] Zhong, X. X. Research on the event processing mechanism of distributed workflow [D]. Lanzhou: Lanzhou university of technology, 2010.

[3] Zhao, Z. G. Research and implementation of distributed workflow management system [D]. Harbin: Harbin Engineering University, 2010.

[4] Boukerche, A., De Grande R. E. Dynamic Load Balancing Using Grid Services for HLA-Based Simulations on Large-Scale Distributed Systems[C]. Distributed Simulation and Real Time Applications, Singapore: IEEE, 2009: 175-183.

[5] Liu, J., Li, X. Z. Research and implementation of a dynamic load balancing mechanism [J]. Computer Engineering and Applications, 2006(2): 142-144.

[6] Weber, B., Reichert, M., Rinderle, S., etal. Providing integrated life cycle support in process-aware information systems[J].International Journal of Cooperative Information System,2009,18(1):115-165.

[7] Gong, X. Q., Zhang, Y. J., Hao, K. G. Overview of technologies applied to implement the distributed workflow management system [J]. Jounal of Xi'an Institute of Technology, 2003, 23(1): 87-94.