I.J. Education and Management Engineering **2012, 4, 1-8**
Published Online April 2012 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijeme.2012.04.01

Available online at http://www.mecs-press.net/ijeme

A Relational Data-based Lightweight Workflow Engine Model

Li Tian[a], Yonghua Zhu[b], Huaiyang Zhu[c]

School of Computer Engineering and Science, Shanghai University, No.149 Yanchang Road, Shanghai, China, 200072
Computer Center of Shanghai University, Shanghai University, Bldg D, No.99 Shangda Road, Shanghai, China, 200444
Shanghai Rongheng Inforamtion Technology Ltd., Floor 13, Bldg 3, No.28 Round 91 Eshan road, Shanghai, China, 200127

Abstract

With the swift development of workflow technology, workflow management system has been widely used. To date, many models and architectures are efficient in constructing workflow engine which is the core of a workflow management system. This paper presents a relational data-based lightweight workflow engine model which based on the relational database. In this model, the definition of the workflow and the data used in the workflow process are kept in the database. This model contains the necessary functions and characteristics of simple business flow rather than all the business flow. The model is composed of three main function modules. These three modules interact with database through DBMS. Under the guidance of this model, a flexible and low-cost workflow engine can be built efficiently. Now this model has been used by Shanghai Rongheng Inforamtion Technology Ltd.

Index Terms: workflow; workflow model; workflow engine; workflow technology

1. Introduction

Nowadays, the idea of workflow turns out to be very popular. Workflow is a calculation model of a business flow. Workflow technology changes the traditional design which focuses on the module and the function. [1]. In order to realize the business flow automation management, workflow management system (WFMS) comes into being. The main purpose of a workflow management system is to support of the definition, execution, registration and control of processes. These processes can be any kind, but most stimulation comes from its promising usage in managing business processes [2]. How to control and manage a workflow? This is the main content of the conception of Workflow Engine. Workflow engine offers the environment that a workflow

* Corresponding author.
E-mail address: [a]tianli917@hotmail.com; [b]zyh@shu.edu.cn; [c]zhu.william@163.com

instance needs for execution. It can explain the process model, control the process instances, explore the activities of a process, evoke other application and access the workflow data. Apparently, workflow engine is the only way to implement workflow management [3]. A workflow engine with good performance needs a good structure engine kernel that solves simple process running problems, namely execution performs in the order of one node by another. The following questions are main concerned: flow definition, flow dispatching problem, flow execution problem [4].

Most existing workflow products, in varying degrees, are bound to specific applications. These products which rely on specific applications are functional and powerful. But they have remarkable defects such as the weak expansibility, the low-rise dependability and usability of the products, the weak integrating with applied system. They are difficult to be migrated seamlessly. In comparison, the lightweight [5] workflow model (LWWM) is more flexible [5]. The designing principles of LWWM are sufficient, flexible and low-cost. LWWM mainly considers the definition of data model and the coordination between the activities, as well as the distribution of tasks and control.

This paper presents a relational data-based lightweight workflow engine model (RDLW) which based on the relational database. The definition of the workflow and the data used in the workflow process are all saved in the database. This model contains the necessary functions and characteristics of simple business flow rather than all the business flow, so it is flexible and lightweight. Three main functional modules make up the relational data-based lightweight workflow engine model. These closely related modules play different parts in the workflow engine model.

The following part of this paper will be arranged as below. In section 2, this paper compares the advantages of several related workflow engines which are realized in different way. In section 3, the concept of the relational data-based lightweight workflow engine model is presented. In section 4, this paper describes the details of these three main modules in RDLW. Then, the findings and directions of future work are discussed in the last section.

2. Related work

The workflow model is the abstract expression to the workflow or the business process. Generally speaking, most of the workflow models are application-oriented. So different applications may leads to different workflow models, and some models have very good performance. Julian Jang [6] discusses a novel implementation of a workflow engine that supports service-based applications. The applications are defined according to the GAT (Guard-Action-Trigger) model, which is an event-based programming model using conditional guards to determine when both normal and exception-handling activities are to be executed.

Owing to the development of network and B/S structure framework, many researchers attempt to combine their models and architectures with the network technology. Li Wei [7] puts forward a model design and key codes based on the latest Java Web Services technical specifications JAX-WS (Java API for XML Web Services) and the BPEL (Business Workflow Execution Language) specification JBoss (Java Business Workflow Management) BPEL. A web service-enabled workflow system is a flexible tool for accessing distributed scientific data, and executing complex analysis on it. Jie Cheng [8] describes a four-layered architecture of the workflow system, which consists of web interaction layer, workflow engine layer, workflow components layer and resource layer. They also develop an intuitive and easy-to-use web based toolkit and apply it to atmospheric data processing.

It is believed that workflow management system has been widely used. This automated way of working has brought convenience to people. So a workflow engine based on a good model is very important. The relational data-based lightweight workflow engine model, in this paper, offers us common method to build a flexible and expansible workflow engine.

3. The concept of RDLW model

The workflow engine plays the most important part in a workflow management system. There are many kinds of methods of constructing the workflow engine. In this paper, a relational data-based lightweight workflow engine model which is based on relational database was put forward. The design of the RDLW model has the following three characteristics.

- WfMC: In this model, the workflow definition is based on the standard of the workflow management coalition (WfMC), so it has good extensibility.
- Lightweight: In RDLW, the enterprise application is abstracted. This model realizes the necessary functions rather than all the functions. So the RDLW model is lightweight and flexible.
- Relational Data-based: The definition of the workflow and the data used in the workflow process are all kept in the database. The workflow data in the database can be divided into two types, some are the definition of the workflow and logical description, and the others are the data used in the workflow process.

The details of this model will be further described in the following.

3.1 The Structure of RDLW

This relational data-based lightweight workflow engine consists of three main modules as shown in Fig. 1. They are the workflow definition module, workflow logic control unit and workflow analysis module. These three modules interact with database through DBMS. The following will describe the specific function of each module.

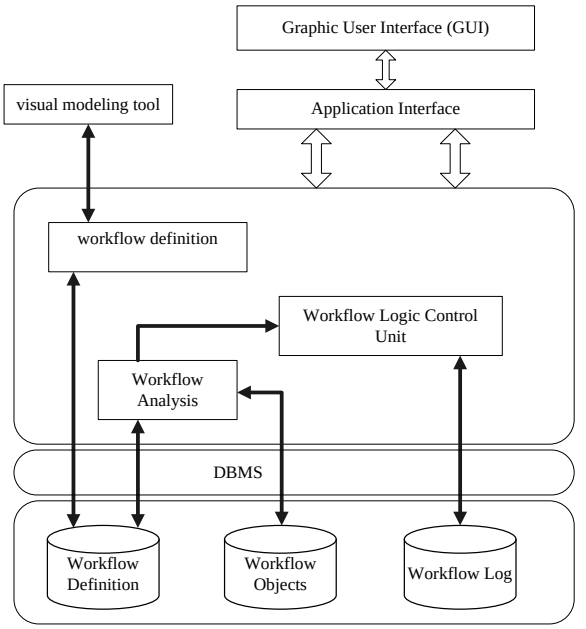

Fig 1. The description of RDLW model

- Workflow definition module: The main function of this module is to generate the definition of the workflow type which is similar to create a class in object-oriented design. With the help of the visual modeling tools, users can make the workflow definition easily. The workflow definition is described in XML Process

Definition Language (XPDL) and saved in the workflow type table. When users need to run a workflow, the workflow engine will create a workflow object according to the workflow type.

- Workflow logic control unit: In this process, the control unit can create a workflow object according to the workflow type. Also, it will generate all the workflow nodes of the workflow object and save the information of the nodes in the relational database. As the workflow object is running, the control unit dominates the workflow process according to the state of the workflow objects and node objects.
- Workflow analysis module: The main function of this module is achieving the Object-relational mapping (ORM). This module realizes the data mapping at program-level and database-level. When the workflow is running, the tables and records at database-level will turn into objects at program-level.

3.2 Design of Database Structure about RDLW

In the model of RDLW, the definition of the workflow, the logic information and the data used in the workflow object are all kept in the database. Therefore, the design of the database structure will be very important. Fig. 2 shows the format of workflow model in database. There are seven tables describing the workflow, and the description can be divided into two categories. The left two tables in Fig. 2 describe the types of the workflow. Other tables in the right part contain the information of the workflow objects. The left part is similar to the concept of class in the object-oriented design. The right part can be seen as objects in the object-oriented design. The conversion is completed through the workflow analysis module.

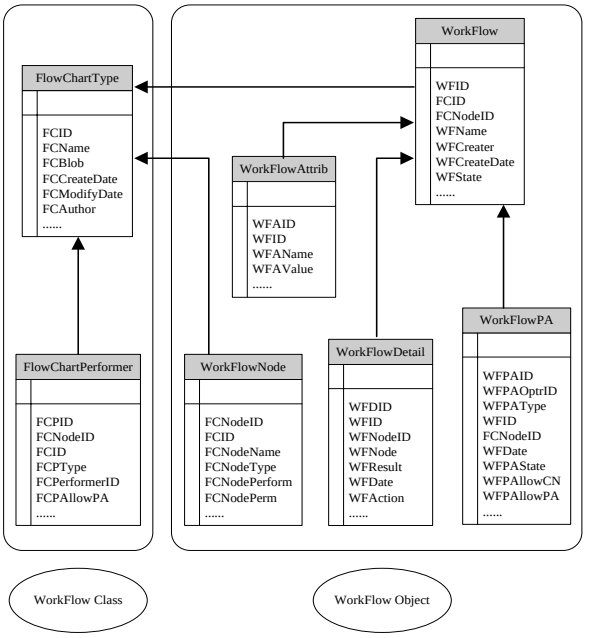

Fig 2. The E-R chart of the workflow model

In this database structure, FlowChartType table and WorkFlow table are the most important tables. Different workflow definitions will be different records in FlowChartType table, and these records can be seen as the templates of the workflows. WorkFlow table stores the real workflow objects created by the templates. The workflow engine relies on the workflow objects to processes the work.

4. Details of the modules in RDLW

As described in the last section, these three modules play different parts in the RDLW model. They all need the support of relational database, and they interact with each other through the relational database. In the following part, some details of these three modules will be described.

4.1 Workflow Definition Module

As Fig. 1 shows, there are two parts that interact with the RDLW model. One is the visual modeling tool, with the help of which users can generate the definition of the workflow easily. The other is the extensible part called application interface, through which other designers can use the workflow engine seamlessly.

The workflow definition module collects the information of the workflow from the visual modeling tool and generates the definition of the workflow in XPDL. Fig. 3 shows the description of the definition in XPDL.

When users create the workflow by the visual modeling tool, the workflow definition module will generate the definition of the workflow at the same time and save the definition in database. In this definition, the "Activity" element represents the workflow node object. And the "Transition" element represents the relationship between workflow nodes. This is a kind of general description of workflow definition.

The visual modeling tool which cooperates with workflow definition module in RDLW will fulfill the workflow definition.

```
<?xml version="1.0" encoding="utf-8"?>
<WorkflowProcess Id="1" Name="WorkFlow1" Created="20010-04-11">
    <Activities>
        <Activity Id="1" Name="Start" Type="Start"/>
        <Activity Id="2" Name="Task One" Type="Automatic">
            <Performer>Per1</Performer>
        </Activity>
        <Activity Id="3" Name="Task Two" Type="Artificial">
            <Performer>Per2</Performer>
        </Activity>
        <Activity Id="4" Name="Task Three" Type="Automatic">
            <Performer>Per3</Performer>
        </Activity>
        <Activity Id="5" Name="End" Type="End"/>
    </Activities>
    <Transitions>
        <Transition Id="1" From="1" To="2">
            <Condition>Con1</Condition>
        </Transition>
        <Transition Id="2" From="2" To="3">
            <Condition>Con2</Condition>
        </Transition>
        <Transition Id="3" From="3" To="4">
            <Condition>Con3</Condition>
        </Transition>
        <Transition Id="4" From="4" To="5">
            <Condition>Con4</Condition>
        </Transition>
    </Transitions>
</WorkflowProcess>
```

Fig 3. Description of workflow definition in XPDL

4.2 Workflow Logic Control Unit

The workflow logic control unit processes the work by executing each node within the workflow according to information in it. First of all, the workflow logic control unit gets the workflow object from the database and

finds the current node in the workflow. Then the control unit will do the logic work under the guidance of the information that saved in the workflow node. When the logic work of the current node is fulfilled, the control unit will get down to the next node.

The abstract application procedure of workflow logic control unit based on RDLW is described in Fig. 4. This flow chart focuses on describing the work process of this workflow engine. In this procedure, users can run a exist workflow or create a new workflow through visual modeling tool. When the workflow instance is running, the workflow logic control unit will execute the workflow according to the information of workflow objects and node objects that kept in the database.

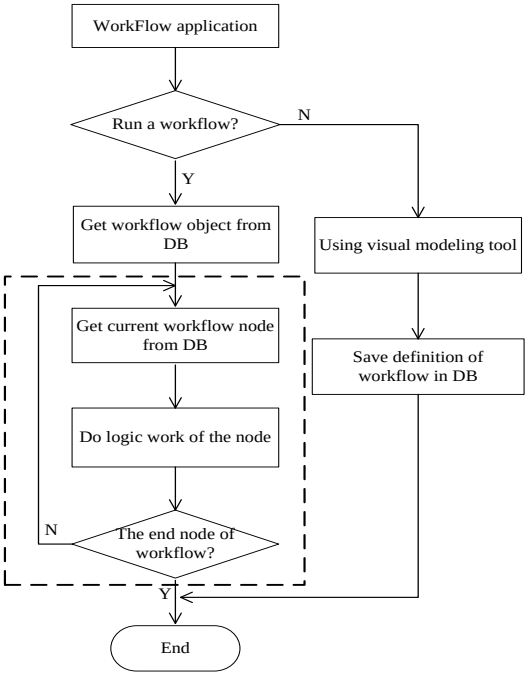

Fig 4. Procedure of the workflow engine

4.3 Workflow Analysis Module

In the RDLW model, the database plays an important part, and all the data are kept in it. So this model is also a database-driven model. As described in Fig. 5, the data in RDLW model can be divided into two levels: the program-level and the database-level. The data have different performances at the two levels.

It is useful and significant to mapping the data at different levels. The mapping work of data in different level is fulfilled by the workflow analysis module. This will simplify the design of the workflow engine, and the workflow engine may work more efficiently.

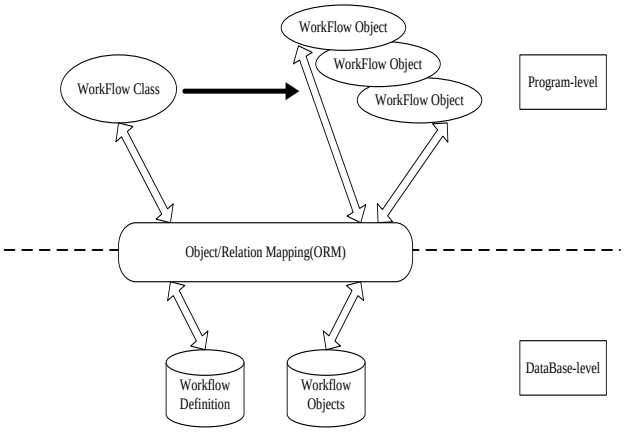

Fig 5. The data mapping of the RDLW

5. Conclusions and future work

In conclusion, it is wise to apply workflow technology to management systems, so as to optimize the business processes. The relational data-based lightweight workflow model (RDLW) only implements the necessary functions and characteristics of the workflow, so it improves extensibility and flexibility of workflows. Further more, the difficulty of the workflow implementation is largely decreased by this relational data-based lightweight workflow model.

This model has been applied in a project of Shanghai Rongheng Information Technology Ltd and has been proved to improve the efficiency of the development of enterprise application.

However, our RDLW model is still open to improvement and we will also attempt to apply the RDLW model to the Web-based application. In the future we will also focus on combining network and SOA architecture to upgrade our model for Web-based use.

Acknowledgment

This work is supported by Shanghai Rongheng Inforamtion Technology Ltd. Special thanks should go to Mr. Zhu, Huaiyang and Mr. Wu, Hui for their technique support. Also we would like to thank Mr. Chi, Qiang and Mr. Wu, Junjie for their help in case design and Zhang, Qian and Wang, Lu for their coding work. Finally, we would like to thank Shanghai Rongheng Inforamtion Technology Ltd. and Computer Center of Shanghai University for their resource supplements.

References

[1] Li Xin-ke, Guo Bin, “The Research of Composing Semantic Workflow Engine Based on Web Service”, Computer Engineering and Applications (ICCEA), pages 262-265, 2010.

[2] Yuhong Yan, Bejan, A., “Modeling workflow within distributed systems”, Computer Supported Cooperative Work in Design, pages 433 – 439, 2001.

[3] Jin Xin, Xu Jing and Li Xuemeng, “The Design and Implementation of XML-based Workflow Engine”, Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2007., pages 137-142, 2007.
[4] Meng You-xin, Huang Shuai, “Research and Application of Lightweight Workflow Engine Kernel”, Computational Intelligence and Software Engineering, 2009. CiSE 2009., pages 1-4,2009.
[5] K.M.Anderson, Aaron Andersen, Neet Wadhwani, “Metis: lightweight, flexible, and Web-based workflow services for digital libraries”, Digital Libraries, 2003. Proceedings. 2003., pages 98-109, 2003.
[6] Julian Jang, Alan Fekete, Paul Greenfield and Surya Nepal, “An Event-Driven Workflow Engine for Service-based Business Systems”, Enterprise Distributed Object Computing Conference, 2006., pages 233-242, 2006.
[7] Li Wei, Lu Hui and Li Jing, “Research on Web Services Oriented Power Marketing Workflow Model”, Information Technology and Applications, pages 7-10, 2009.
[8] Jie Cheng, Xiaoguang Lin, Yuanchun Zhou and Jianhui Li, “A Web Based Workflow System for Distributed Atmospheric Data Processing”, Parallel and Distributed Processing with Applications, pages 584-588, 2009.