

Design and implementation of timing tasks based on Activiti5 Workflow Technology

Hu Jieping^{1, a*}, Yang Shulin^{2, b}

¹Department of Basic, Beijing Institute of Graphic Communication, Beijing, China

² College of Information Engineering, Beijing Institute of Graphic Communication, Beijing, China

^ajjieping125@sina.com, ^byyshull@sohu.com

Keywords: Workflow, Activiti5, Timing Task, DWR, Push Technology.

Abstract. This paper introduces and analyzes Activiti5 workflow management platform on the overall design, On the basis of this, study the design and implementation of workflow tasks timing. Using timer boundary event to achieve a timing task and using DWR push technology to achieve a refresh, such scheme in teaching and training system has been applied and received good results.

Introduction

Workflow is a computational model of work flow, is to work out how the front and rear of the logic and rules organized together in a computer in an appropriate model representation and calculating its implementation. The main problem to be solved is the workflow: to achieve a business objective, among a plurality of participants using a computer, the automatic transmission according to some predetermined rule. In recent years, workflow technology has been widely used in a variety of business systems, application of workflow technology improves business processes related fields to achieve a coordinated control of business processes and improve operational efficiency[1]. In practical applications, often have some special needs, for example, certain tasks need to be completed within a specified time, or increase the timing tasks at work or overtime reminder, these requirements relate to the timing and how to remind problems. In this paper, combining with the design practice task workflow, explore solutions to this problem.

Activiti5 technology and the design of timing task workflow

Activiti technology. Workflow is the abstraction, generalization and description of work flow and business rules between the various steps. Its main function is to define through the support of computer technology, information exchange between process execution and management of workflow execution, coordinating workflow between work and community members. Activiti5 is an open source lightweight workflow management platform by developed from jBPM3, jBPM4, which emphasizes embeddable and scalable processes and services, currently consists of jBPM creator Tom Baeyens led the development and maintenance[2]. Activiti provides graphical process designer and process forms designer, which are Web-based and use Activiti Modeler and the Eclipse plugin in the form of SVG implementations of Activiti Designer. Activiti5 architecture shown in Fig. 1.[3]

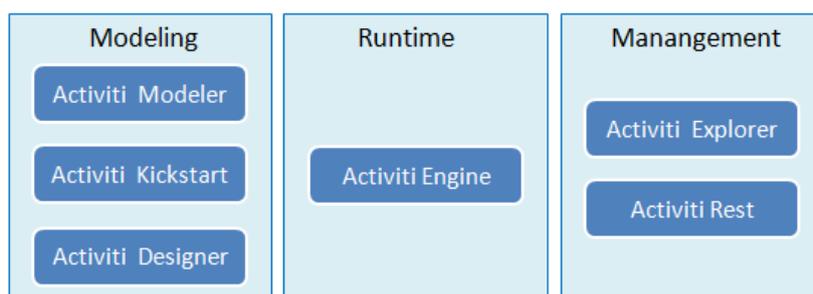


Fig.1 Activiti5 architecture

Activiti Engine is the core of the Activiti projects. It is a process engine written in Java, used to execute the process defined by process modeling language. Activiti Explorer is a Web application for users to access the Activiti Engine. Activiti Modeler provides solutions from the Web to manage Activiti process. Activiti Designer is an Eclipse plug-in, which allows the establishment of a process model within the IDE environment with BPMN2.0. [4] [5]

Design of timing task workflow. To use timer boundary event + Job way to achieve timing task. Timer events are events which are triggered by defined timer. They can be used as start event, intermediate event or boundary event. Timer definition must have exactly one element from the following:

timeDate. This format specifies fixed date in ISO 8601 format, when trigger will be fired.

timeDuration. To specify how long the timer should run before it is fired, a timeDuration can be specified as sub-element of timerEventDefinition. The format used is the ISO 8601 format (as required by the BPMN 2.0 specification).

timeCycle. Specifies repeating interval, which can be useful for starting process periodically, or for sending multiple reminders for overdue user task.

A timer boundary event acts as a stopwatch and alarm clock. When an execution arrives in the activity where the boundary event is attached to, a timer is started. When the timer fires (e.g. after a specified interval), the activity is interrupted and the sequence flow going out of the timer boundary event are followed. A timer boundary event is defined as a regular boundary event. The specific type sub-element is in this case a timerEventDefinition element.

```
<boundaryEvent id="escalationTimer" cancelActivity="true"
  attachedToRef="firstLineSupport">
  <timerEventDefinition>
  <timeDuration>PT4H</timeDuration>
  </timerEventDefinition>
</boundaryEvent>
```

Timing task workflow is shown in Fig.2.

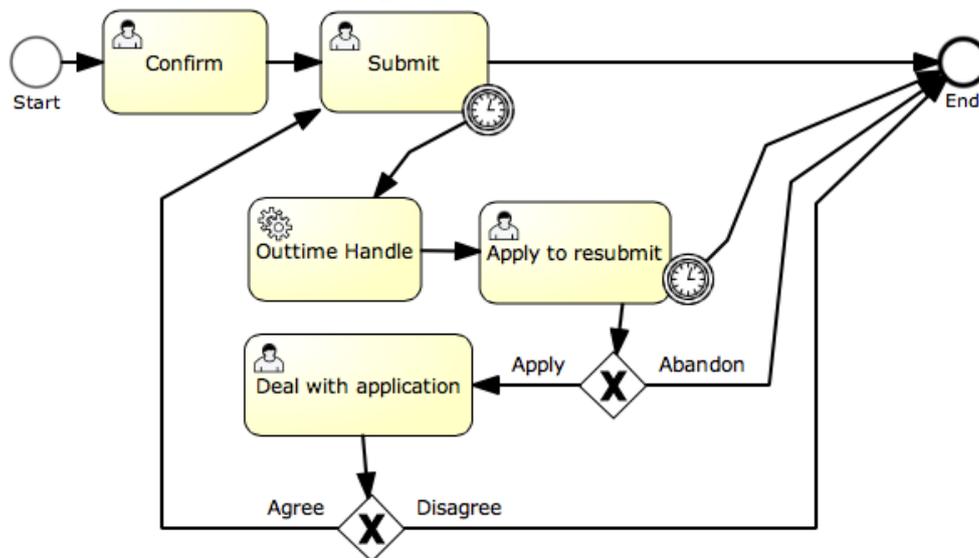


Fig.2 Timing task workflow

Confirm. To confirm task show that students have accepted the task, while preserving the relevant information on the business data tables.

Submit. Submit the completed results. In order to achieve timing, added A timer boundary event in this task. Part of the code is shown as follows:

```
<userTask id="submit" name="Submit" activiti:assignee="{assignee}"/>
<boundaryEvent id="boundarytimer1" attachedToRef="submit" cancelActivity="false">
```

```

<timerEventDefinition>
<timeDate>${time1}</timeDate>
</timerEventDefinition>
</boundaryEvent>
<sequenceFlow id="sequenceFlow-21898ea5-4157-4282-a077-7156c7ca5e36"
    sourceRef="boundarytimer1" targetRef="outtimedeal"/>

```

Outtime Handle. If the mandate holder do not submit their result, will be called by serviceTask handler automatically record overtime log. If not timely submit results, will be called by serviceTask processing program, recording the time-out log automatically, and remind the user's application to re submit. If submitted on time to submit the result ,will end the process. Used variables as the time parameters, using the following format time in Java code:

```

SimpleDateFormat sd1 = new SimpleDateFormat("yyyy-MM-dd");
SimpleDateFormat sd2 = new SimpleDateFormat("HH:mm:ss");
String time = sd1.format(task.getEndDateTime()) + "T" + sd2.format(task.getEndDateTime());

```

Apply to resubmit. If the mandate holder want to apply to resubmit, he need to fill out the reason. it is the user can give.and The task also is added A timer boundary event

Deal with application. Managers deal with the mandate holder's application. May agree to resubmit, or do not agree resubmit.

To realize the function of reminding and refresh on time.

The boundary timer events can only be triggered when open the job executor (i.e., need in configuration file will be jobExecutorActivate set to true, because the default is disabled actuators). Design of timing task workflow. In the Web terminal by the JavaScript script display countdown time, so it will be easier to implement. But how to in time, submitted to the server, in order to go to the next process. It is difficult to realize. Here the use of DWR (Direct Web Remoting) long connection based on push technology. Advantages: good real-time (message delay small); good performance (can support a large number of users). DWR is is a Ajax open source framework for the remote server to improve the web page and Java interaction, can help developers website which include AJAX technology.

At the Web end, for submission and refresh the JavaScript code as shown below:

```

function submitAndRefresh(id) {
    $.post(
        ctx+'/SxTaskAction!completePrev.action',
        {id: id},
        function(result) {
            window.location.reload();
        }, 'json');
}

```

DWR automatically push the code shown below:

```

public class DWRHelper {
    public static void send(final String receiverid, final String msg) {
        Browser.withAllSessionsFiltered(new ScriptSessionFilter() {
            @Override
            public boolean match(ScriptSession session) {
                if (session.getAttribute("userId") == null) {
                    return false;
                } else {
                    return (session.getAttribute("userId")).equals(receiverid);
                }
            }
        })
    }
}

```

```

    }, new Runnable() {
        @Override
        public void run() {
            ScriptBuffer script = new ScriptBuffer();
            script.appendScript("submitAndRefresh(")
                .appendScript(msg)
                .appendScript(");");
            Collection<ScriptSession> colls = Browser.getTargetSessions();
            for (ScriptSession scriptSession : colls) {
                scriptSession.addScript(script);
            }
        }
    });
}
}
}

```

Summary

Activiti5 workflow model based on BPMN2 like branches, processes in the previous workflow system in parallel, process more difficult to achieve functions such as merger, through a simple graphical operation can be realized. Compared with the traditional development mode, Activiti5 in the hard-coded workflow system eliminates the past business analysts and software developers, the problem of confusion, its business designers can independently process design features, the developers from tedious business process design and coding, greatly improve the corresponding velocity of business process requirements change. This paper study the design and implementation of workflow tasks timing, The proposed scheme applied in teaching and training system has been applied and received good results.

Acknowledgement

Fund: Beijing Institute of Graphic Communication the university generally project funded project (E-b-2014-20).

References

- [1] Xu Yinan, Ge Zhihui, Pan Haiyuan. Applications of Activiti5 in OA system. Popular science & Technology, 2014, (1):5-7.
- [2] Mauricio Salatino, Esteban Aliverti, Mauricio Salatino, Esteban Aliverti. *JBPM5 Developer Guide[M]*. Packt Publishing, 2012.
- [3] Activiti. <http://www.activiti.org/>.
- [4] Li Haibo, Zhan Dechen, Xu Xiaofei. Component composition based on workflow engine system structure [J]. Journal of software, 2006.
- [5] Jin Zhengye, Cai Yueru, Ma Weiqin, Liu Xiling. Computer engineering and design of component based development of [J]. workflow system, 2006.
- [6] Yang Enji. Workflow notes: development of workflow application based on Activiti, Publishing company of electronics industry, 2014.