INTEGRATING COLORED PETRI NET AND OBJECT ORIENTED THEORY INTO WORKFLOW MODEL

Zhengli Zhai¹,²

¹ Department of Computer Science and Technology, Tongji University, China
zhaizhl@163.com

² Computer Engineering Institute, Qingdao Technological University, China
zhaizhl@163.com

Abstract. Workflow management has always been a hot spot in both business and academic circles. Modeling is the foundation of workflow management; however many of the models today have unavoidable limitations. With its powerful modeling ability, formal description and firm mathemetic foundation, Petri net is a perfect tool to analyze nonlinear adaptive system. Combined with Colored-Petri, here a modeling method based on object oriented theory is put forward and an online shopping model is set up based on Objected Oriented Colored Petri Net.

Keywords: Workflow; Colored Petri net; Object Oriented; Online shopping

Introduction

Since its development from 1990s, workflow has been through rapid growth. It offers enterprise methods of modeling, managing and controlling. Its basic function is to combine the behavior of man and the machine, as a result of which to improve the operational efficiency, to optimize resource allocation and to increase the operational flexibility of the enterprise.

For any research, the backbone is theory, with many suspended problems which are also the relative weak part of workflow. Here we base on the features of Colored Petri Net (CPN for short), describe the resource with colored tokens and analyze the basic properties of Object-Oriented Colored Petri Net (OOCPN for short).

Workflow Overview

For any workflow, it not only contains chains of activities, it also involves the start and end conditions, detailed description of every activities, such as activities executor, specific dealing methods and data needed. This requires a general model to create workflow instance, to realize the integration of different workflow systems and at the same time to manage and maintain it in later period. Workflow Management Coalition (WFMC) [1] published the Workflow reference model in November, 1994. According to this model mutual operation between various workflow systems is available. What’s more, it enables the customer to visit services from different workflow machines. This general model is the basic of workflow system modeling.
Applied in flow management, the workflow has three stages: workflow modeling, workflow simulation and workflow optimization. While workflow modeling means using a clear and visualized way to describe distinct abstract layers, workflow simulation is to discover problems so as to improve the workflow system. And reliable modeling is the basis of the process analysis.

In modeling, the workflow system generally adapts description tools like flow charts and workflow description language which can give a brief introduction. But it contains no uncertainty-handling mechanism. Nor does the workflow management tool. To effectively control every stage in workflow system, we need to combine system modeling, simulation with workflow. However simulation in workflow is a weakness which requires our attention and efforts.

**Colored Petri Net**

High-level Petri net is a further abstraction of Petri net. By classifying and analyzing tokens, high-level Petri net can reduce the number of basic elements in order to decrease scale of the net. With its help, it’s easier (to some extent) to model a complex system [2]. Colored Petri net (CPN for short) is one of a relative mature high-level Petri net.

CPN consists of three parts: first, the net structure it inherited form Petri net which includes place, transition and directed arc; second, the declaration and definition of net. CPN introduce the concept of colored token into Petri net. To increase the model’s reusability, we give different object’s attributes to different colored tokens; third, we adapt Standard Meta Language (SML for short).

Here presents the basic notions of CPN.

**Definition 1** A simple 5-tuple $\Sigma = (S, T; F, W, M)$ is called a colored Petri net (with k kinds of colored token). Where,

1. $(S, T; F)$ is a finite net. The elements of $S$ and $T$ are called places and transitions, respectively.
2. $W : F \rightarrow \{0, 1, 2, ..., \}^k$
3. Map $W$ attaches a non-negative k-dimensional integer vector to each arc of the net.
4. $M : F \rightarrow \{0, 1, 2, ..., \}^k$
5. Map $M$ representing a non-negative k-dimensional integer vector to each place.

For $t \in T$, if $s \in t \rightarrow M(s) \geq W(s, t)$, then transition $t$ has the right to fire.

We say $t$ fires from $M$ to $M'$ and we write $M[t > M']$, where, $M'$ is noted as follows:
From the definition we know that apart from the differences between weighted function $W$ and marking $M$, a simple CPN is just like a weighted Petri net (P\T system with infinite place capacity).

**Object Oriented Theory**

Object-oriented is now the focus of computer industry. The concept of Object-oriented no longer restrained to programming but has expanded to wilder fields, such Database System, Distributed System and Artificial Intelligence, etc.

Object-oriented [3] means that the object world is made up with objects, in other words, everything is an object with its own attributes and actions. Different systems form when different objects interact and influence each other. Clearly, the key in Object-oriented technology is object. It has some features as follow:

1. **Uniqueness**
   Every object possesses unique tokens by which they can be easily distinguished.

2. **Abstract**
   If objects have the same attributes and actions, we can abstract them into a class. When we need to manipulate these attributes and actions, we just need to declare an object to use them; this method turns us into a commander from the executor.

3. **Inheritance and Polymorphism**
   Inheritance is the feature that only OOP possesses. It has greatly enhanced the program’s reusability.
   Polymorphism enables every object to answer the same request with its own methods, which improves both flexibility and scalability.

For much complex information involved, it’s impossible to meets the need of every parts in only one WFMS. What we need is a distributed, non-platform related WFMS, which is also capable of intelligently dealing data information. Combining the OO and Petri net (OOPN), we can create different objects for different parts in WFMS. CPN’s powerful logical describing and analyzing power assures a proper workflow using various activities in the system.

**Process Modeling**

Workflow model is the abstraction of the actual workflow, to put it in another way the abstraction of practical business processes. First we must understand what a process is. Usually a process requires many departments to participate in; each of them is responsible for different
parts and work separately. A process is finished when all parts involved is well interacted and cooperated.

Petri net theory is a perfect tool to explain the execution of process in workflow models. Here we introduce 4 structured-blocks [4], which are AND-split, AND-join, OR-split and OR-join. Fig.1 shows these four blocks.
(1) Sequence: it means the two actions are executed in serial order. Action B is executed after action A is done.
(2) Parallel: it means action B and C can be executed simultaneously with random order.
(3) Select: it means when action A is finished, action B or C is selected to be executed.
(4) Repeat: it means one action can be executed many times, for example, action B and C in Fig.1(d).

In Fig.1, transitions represent actions while places mean the relations between various actions and the tokens contain the information of workflow and resources related to actions. Fired transition means the action is being carried out and directed arc shows the direction of workflow. When CPN is dealing with complex problems, tokens in places no longer simply mean action being executed, but have been given to specific semantic.

![Fig. 1 Petri net implementation of four structured blocks](image)

**Model of Online Shopping**
Now we briefly describe a process model of online shopping.
First the customer wants to purchase a certain commodity, then apply to online mall, pay for the commodity via online banking which will give both the success or failure information to online mall. After verifying the online mall then it will deal with the successful orders.

However how can we attach object-oriented features into CPN? To solve this we must observe a problem with object oriented thoughts. Step one, we must identify every objects in the system; step two, build up the relations(connections) between each object; step three, confirm the initial state of the system, put tokens into relative places.

In OOCPN, transitions stand for the activities of customers and online mall, for example apply to purchase; places stand for the state of customers and online mall, for example the confirmation of an order; the arcs between transitions and places stand for the causal relationship. If a place fits the firing condition and fire a transition, there would be an arc pointing from the place to the transition and vice versa.

The OOCPN model of order applying and conforming of online shopping is given in Fig. 2 bellow.

Fig. 2 Model of order applying and conforming of online shopping

P1: means the customer is prepared to filling in the order;
P2: means the completed order;
P3: means the customer is waiting for the return of the information of the order;
P4: means the customer has received the information of the order;
P5: means online mall prepares to accept the order;
P6: means online mall receives the order;
P7: means online mall is preparing to send the information of the order back;
P8: means online mall completed sending the information of the order back;
t1: means the customer is filling the order;
t2: means sending the order to the online mall;
t3: means the customer has received the denial order from the online mall;
t4: means the customer has received the confirmed order from the online mall; 
t5: means online mall is accepting the order from the customer; 
t6: means online mall is dealing with the order, to judge whether it is successful or not; 
t7: means online mall is sending the customer the information that the order is failed; 
t8: means online mall is sending the customer the information that the order is successful; 
MP1: means having received the order from the customer and send it to the online mall; 
MP2: means finish sending the customer back the information about the order.

As we can discover above, that the working process of MP1 and MP2 are quite similar, so we 
can use colored token which was encapsulated in one place to distinguish the two actions. In this 
way the number of places and transitions is reducing which results in easier reading of the chart 
or figure, as shown in Fig.3.

![Fig. 3 Simplified model of order applying and conforming of online shopping](image)

In CPN, we assume place means activity while transition means the transform of different 
activities, colored token means different attributes of the activities [5]. If we use different 
colors to present MP1 and MP2 and package the two colors in one place to do the similar work, then we 
get our Fig. 3.

We mark a two-dimensional vector [0, 1] in MP, the first component is red and the second one is 
yellow. Fig.3 shows that if the vector is [0, 1] then MP receives the information from P2 and 
deliver it to the online mall to confirm. Else if the vector changes into [1, 0], MP receives the 
order-confirmed information from online mall and deliver it to the customer.

OOCPN can effectively decrease the complexity of process modeling in system. With its 
powerful operational functions, it’s a further development of OO applied in Petri net and it has a 
superior status in modeling the complex business processes.
Conclusions

First thing we need to know when modeling by Petri net is how to describe an abstract workflow model using Petri net theory. To study modeling complex workflow, here we introduce OOCPN, which offers better methods than PN when modeling too complex model. The simplification of model effectively reduce the complexity of algorithm, object-oriented is an easy-understanding theory and fits human’s habitual thinking which has become one of the hottest spots of workflow modeling using Petri net.

Acknowledgement

This work has been funded by The National Basic Research Program of China (973 Program) under Grant No. 2010CB328101, National Nature Science Foundation under Grant No. 61173042, Shanghai Nature Science Foundation under Grant No. 13ZR1443100, Excellent Youth Scientist Research Reward Fund of Shandong Province under Grant No.BS2010DX009, Qingdao Science and Technology Program under Grant No.12-1-4-4-(7)-JCH, Qingdao Technological University’s Research Fund under Grant No.C2007017.

References


