Available online at www.sciencedirect.com
ScienceDirect
Computers in Industry 59 (2008) 193–209
COMPUTERS IN INDUSTRY
www.elsevier.com/locate/compind

A timed colored Petri nets modeling for dynamic workflow in product development process

Seungchul Ha, Hyo-Won Suh *

Department of Industrial Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Republic of Korea

Available online 23 August 2007

Abstract

Product development process (PDP) is one of the most important business processes for enterprises but it has difficulty in workflow management because of the uncertain and dynamic characteristics. Thus, even though there have been many workflow modeling and management methods, they have limitations to deal with the special characteristics of PDP.

We define PWF-nets based on timed colored Petri nets for the workflow management of PDP and propose a method of organizing PWF-nets that are composed of workflow patterns. Considering the characteristics of PDP, PWF-nets are somewhat complex unlike existing workflow modeling because they manage necessary information related to dynamic workflow execution. Thus, we define transition behaviors and function primitives to support the construction of workflow patterns for PWF-nets. These workflow patterns can be microscopically handled and can be adapted for the dynamic and uncertain environment related to data and time. Next, we discuss the soundness and properties of the PWF-net and introduce an example of implementation. Additionally, we can check errors or abnormal executions of workflow and evaluate the lead time of PDP through our program for the PWF-net.

Keywords: Workflow modeling; Workflow patterns; Timed colored Petri net; Product development process (PDP); PWF-net

1. Introduction

The companies have numerous business processes and a key success factor in competition is the effective management of core business processes. In this context workflow technology has been introduced for systematic business process management that supports a consistent information flow between agents and the adequate sharing the data and information. There have been many works related to the workflow applications. Most of works focused on general business processes which are relatively simple and repetitive so many workflow management systems (WFMS) adequately support when the business process to be executed is well-structured. In these cases, we can predefine all reachable paths in advance. However, PDP is hard to be planned completely a priori because the detail route of workflow can be generally determined at run-time. There are several studies focused on the PDP with the special characteristics such as *uncertain*, *evolutionary*, *iterative*, and *coupled*.

Petri net allows a smooth transition from the conceptual level of process model to an implementation level [1] and it is a state-based description so it has been broadly used for workflow modeling. van der Aalst [2] explained the reason of using a state-based description that can keep distinct between the enabling of an activity and the execution of an activity. There are various types of Petri net related to workflow modeling. Basic Petri nets describing a real system tends to be complex and extremely large so that extended Petri nets with time or color has been used to model a system [3]. Extended Petri nets include several high-level Petri nets such as a timed Petri net, a colored Petri net, and a hierarchical Petri net. High-level Petri nets are suitable to workflow modeling of PDP because they can manage various attributes like due date and whether some predecessor activities are executed or not. A timed Petri net generally associates transitions with deterministic or stochastic firing times so that we can analyze the temporal behavior of PDP using them. In a colored Petri net, attributes are associated with tokens and each activity is assigned to the tokens of own types. Especially, within a workflow modeling, the color or the value of a token contains data about the case such as routing parameter, due date, responsible or preferred resource [2].

* Corresponding author. Tel.: +82 42 869 3123; fax: +82 42 869 3110.
E-mail address: hw_suh@kaist.ac.kr (H.-W. Suh).

0166-3615/$ – see front matter
doi:10.1016/j.compind.2007.06.016

These data are necessary to PDP management and attribute of them can be updated according to the progress of workflow. Lastly, a hierarchical Petri net allows for hierarchical modeling so it is possible to decompose a complex system into simpler subsystems. There have been many Petri net-based workflow modeling methods but they mainly focused on the quite simple and repetitive workflow. Therefore they have difficulty in supporting the characteristics of PDP and require the additional consideration.

In this paper, we define a PWF-net that is based on a timed colored Petri net for the workflow management of PDP and additionally propose a method of composing a PWF-net by workflow patterns. First of all, we introduce Petri net-based workflow patterns for a PWF-net. These patterns represent the typical execution types of PDP and can support a less rigid workflow specification for PDP which contains the workflow of fixed route and ad hoc route as well. The fixed part of workflow is specified at build-time but the ad hoc part of workflow cannot be wholly specified at build-time, and it can be completely determined during run-time. Workflow patterns are composed of six types such as Sequential, Overlapped, Iterative, Coupled, Split, and Join pattern. Considering the characteristics of PDP, PWF-nets are somewhat complex unlike existing workflow modeling because has to manage necessary information that is related to the execution of dynamic PDP workflow. Thus, we define four transition behaviors and eight function primitives to support the construction of workflow patterns for PWF-nets. Transition behaviors are related to the cases that the color of token are changed and function primitives represent the types that are several typical structure included by workflow patterns. Workflow patterns formed by function primitives and transition behaviors are microscopically handled and can be adapted for the dynamic and uncertain environment related to data and time. Next, we discuss the soundness and the properties of PWF-net and introduce an example of implementation. Additionally, we can check errors or abnormal executions of workflow and evaluate the lead time of PDP through PWF-net executor.

The rest of this paper is organized as follows. In the following chapter, related works are introduced. In Section 3, modeling patterns for workflow of PDP are introduced. Petri nets for workflow model are introduced and Petri net for PDP workflow (PWF-net) are suggested in Section 4. Analysis of PWF-net and related program and implementation are presented in Section 5. Finally, conclusion is given in Section 6.

2. Related works

Workflow model is used to capture a process abstraction and includes an activity structure and information exchange among activities in a workflow, exception handing, activity duration, and priority attributes [4] but there is no standard of the workflow modeling. Describing the uncertain and dynamic characteristics of PDP requires dynamic workflow management. The works for dynamic workflow are classified into the evolution of defined workflow specification [5,6] and the data flow-based workflow management [7,8]. The latter includes the specification of a flexible execution behavior through less restrictive behaviors to express an uncertain or dynamic behavior in advance. In this paper, we design and use the process templates for dynamic workflow management. The process templates have a semi-structured process definition [9–11].

Practically, a lot of works have used the Petri nets representation for workflow modeling. The reasons for this are summarized as five elements such as formal semantics, graphical nature, expressiveness, analysis method, and vendor independent [12]. The previous research using a Petri net is various from a basic Petri net to an extended Petri net as a high-level Petri net. A basic Petri net describing real systems tend to be complex [13]. In order to solve this problem, some researchers extend a basic Petri net with a timed Petri net, a colored Petri net, or a hierarchical Petri net.

Adam et al. [1] presented a framework for workflow modeling and analysis using a Petri net. They classified activity relations as control flow, value, and external (temporal) dependencies and identified several structural properties of a Petri net related to the inconsistent dependency specifications among activities. van der Aalst et al. [3] composed a Petri net-based workflow model in which the key concept of workflow management is an activity and a procedure. The modeling is a partially ordered set of control activities, pairs of activities, a set of resource classes, and (sub) procedures. Moreover, van der Aalst [12] proposed the formal Workflow-net (WF-net) that is the method for workflow modeling based on a basic Petri net and showed the mapping procedure workflow management concepts onto Petri nets. Especially, it turned out that a WF-net is sound if and only if a slightly modified version of WF-net is live and bounded. Based on the WF-net, Ling and Schmidt [14] proposed the Timed Workflow net (TWF-net) that is basically extended with time from the WF-net for the purpose of modeling and analyzing workflow system with time constraints. They showed the verification of behavioral properties such as the liveness and the soundness considering time constraints. Liu et al. [15] suggested the Workflow net based on a colored Petri net (WFCP-net) that is extended with color from the WF-net as well so it can be used to model a family of workflow processes with similar process routes and logic rules. They allowed users to write business rules and to integrate with the applications through the workflow script language in editor. Fig. 1 shows the relationship among these Petri nets.

Gou et al. [13] proposed the Interval Timed Colored Petri net (ITCPN) for description of business processes. ITCPN is a Petri net extended with time and color and a timestamp is attached to each token indicating the time when the token becomes

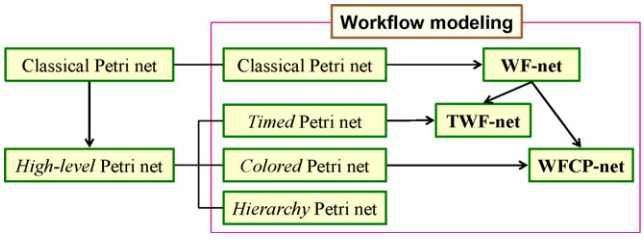

Fig. 1. Workflow modeling based on Petri nets.