ELSEVIER

Computers in Industry 49 (2002) 267–281

COMPUTERS IN INDUSTRY

www.elsevier.com/locate/compind

Modeling workflow processes with colored Petri nets

Dongsheng Liu[a,*], Jianmin Wang[a], Stephen C.F. Chan[b], Jiaguang Sun[a], Li Zhang[a]

[a]*Department of Computer Science and Technology, Tsinghua University, 100084 Beijing, PR China*
[b]*Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China*

Received 29 March 2001; accepted 6 July 2002

Abstract

The definition and maintenance of workflow processes have become important tasks for enterprises as workflow management systems (WFMS) are systematically applied to critical business processes. In order to simplify the management and usage of workflow processes and to integrate with other applications, a good modeling method is essential. The WFCP-net (workflow-net based on colored Petri net) is an extension of the workflow-net (WF-net) which can be used to model family of workflow processes with similar process routes and logic rules. An expanding suite of tools, which currently includes, can support its application: a process structure graph editor, a WF script (workflow script language for writing business rules) editor and translator (translate WF script into Java classes) and a dynamically loadable workflow engine.

Keywords: Workflow management system; Colored Petri nets; Workflow process specification; Workflow script language

1. Introduction

These years have seen many workflow management systems implemented on a wide range of organizations. From office automation to enterprise application integration, workflow technology is playing a vital role. Workflow management systems are being used to model, execute, monitor, coordinate and improve business process of real or virtual organizations.

A WFMS consists of two basic components: workflow models and workflow enactment engine. Various authors have stressed the need for formal foundations of process specifications techniques [1,8], and many researches focus on conceptual model of workflow [6,7,18]. The Meteor project [12] developed workflow specification language (WSL), a declarative rule-based language, to expression application level interaction with multi-tasks and task specification language (TSL) to depict individual task issues. These languages were used to address the issues of inter-task dependencies, data formatting, data exchange, error handling and recovery. Ellis used information control net (ICN) to represent control flow and data flow [8]. But the ICN definition has only defined individual process model rather than family of models with similar rules and routes. Hofstede use Task Structure to specify workflow processes [9], van der Aalst and coworkers applied Petri net to workflow modeling in a series of papers such as [1–3].

* Corresponding author. Tel.: +86-10-6277-3833; fax: +86-10-6277-2288.
E-mail addresses: liudongsheng98@mails.tsinghua.edu.cn (D. Liu), jimwang@tsinghua.edu.cn (J. Wang), csschan@comp.polyu.edu.hk (S.C.F. Chan), sunjg@tsinghua.edu.cn (J. Sun), lizhang@tsinghua.edu.cn (L. Zhang).

PII: S0166-3615(02)00099-4

While the Petri net variant approach has a solid mathematical basis as well as analysis and verification algorithms and tools, there are still some outstanding issues to be addressed such as the description of family of workflow processes and convenient methods and tools to integrate WFMS with enterprise application software. In seriously competitive worldwide markets, it is vital for enterprises to increasingly improve their business process to meet the demand of consumers, especially for new product development departments. In Collaborative Product Commerce environments [16], there are higher requirements for workflow management process models and execution engines than in rigorous product workflow management system. The workflow management components must have powerful methods to control product design processes and engineering change processes; a case study presented in Section 2 exemplifies these requirements.

The remainder of this paper is organized as follows. Section 2 introduces a case study of workflow processes and puts forward the requirements. Section 3 presents our workflow conceptual architecture. In Section 4, we define WFCP-net by extending WF-net with data type and show that this method is powerful enough to model process families by modeling the case study given in Section 2. Section 5 presents our implementation of workflow engine and other modeling tools.

2. Study case and requirements

Shengyang Auto Manufacture Company (SAMC) is engaging in designing and manufacturing tourist buses in China. The Technical Center, a key research and development department of the company, develops and designs new models and changes old models to meet varied demands of customers (e.g. special interior decoration for specific usage such as blood collection, special technical specifications requirements for special road conditions). The Technique Center, which is managed by a chief designer and a supervisor, is composed of functional groups with responsibilities for designing and changing main assemblies of bus such as bus-body group, chassis group, instrument group, and entire-bus group. Each group is composed of a number of designers and a group head.

We take the approval procedures of engineering documents as a case study for workflow process modeling. The main activities and business rules of the procedure are described in the following sections.

2.1. Design

Designers generate 2D drawings and 3D models with the CAD software AutoCAD, and search and store and retrieve those engineering documents in database using the product data management (PDM) [15] software TSPDM on their Intranet. The engineering documents are classified as Drawing, bill of materials (BOM), and engineering change order (ECO), which are classified as three types: ECO-I, ECO-II, ECO-III according to importance of change, total-list (TL) of drawing, sub-list (SL) of drawing. As soon as a designer completes his document the designer submits it to a checker.

2.2. Check

Using view and markup tools such as AutoVue, the checker examines the engineering documents to find errors or nonstandard notations. If the checker determines that the document is correct the checker submits it to the group head for approval, or returns the document to the designer.

2.3. Group head approval

The group head mainly checks whether the document is coherent with other document designed by same group. If the group head rejects the document he returns it the designer or the process continues. If the type of the document is one of the set {BOM, SL, TL and ECO-I} the approval process terminates.

2.4. Related groups approval

The related groups, which are related to a group in design activities, check whether the approving drawing is consistent with other drawings designed by the related groups. For example, a front-encloser designed by the bus-body group is related to the instrument group; hence, the instrument group should check the document of the front-encloser. The designer determines which groups are selected and should check the document.