A Framework for Document-Driven Workflow Systems

Jianrui Wang and Akhil Kumar

Smeal College of Business,
Pennsylvania State University, University Park, PA 16802, U.S.A.
{JerryWang, AkhilKumar}@psu.edu

Abstract. We propose and demonstrate the feasibility of a framework for document-driven workflow systems that requires no explicit control flow and the execution of the process is driven by input documents. The framework can assist workflow designers to discover the data dependencies between tasks in a process and achieve more efficient control flow design. The framework also provides an architecture to separate the workflow system from application data and facilitate inter-organizational processes. Document-driven workflow systems are more flexible than traditional control flow processes, easier to verify and work better for ad hoc workflows. We also implemented a prototype workflow system using the framework entirely in a RDBMS using Transact-SQL in Microsoft SQL Server 2000. A detailed comparison with control driven workflows has also been done.

1 Introduction

Academic interest in workflow systems has increased considerably in the past decade, especially with the boom in e-business and supply chain management. Workflow is built into most commercial e-business and supply chain management software, and functions as a foundation module to support business process performance and coordination.

ARIS (Architecture of Integrated Information Systems) [17] developed a pioneering approach to model business processes, and also served as a foundation of SAP/R3. ARIS takes five views of business processes: functional, organizational, data, output, and control. The Workflow Management Coalition views workflows as interactions of process, information and resource [9]. Depending on the dimension used for modeling, workflow systems can be viewed from one of the following perspectives:

1. Process based perspective. This perspective tends to emphasize process as the dominant dimension; processes consume, produce or transform information under a set of business rules.
2. Information based architectures. This perspective emphasizes the information dimension, viewing processes as operations that are triggered as a result of information changes.
3. Organization perspective. This perspective views workflow as a mapping of organization structures and focuses on the utilization of organization resource. Unfortunately, although it is well accepted that workflow systems are an integration of data, control, and resource, most workflow modeling languages such as WSBPEL [16] (formerly BPEL4WS) and XPDL [19] focus on control flow, and give less attention to other dimensions. One popular control flow study is the one on workflow patterns by Aalst [1]. There are only a few studies on data flow modelling [4,6,7,11,12]. However, for the most part, data and resource flow research has received little attention compared with control flow [15].

In this paper, we take the information based perspective, and extend the ideas in the WIDE approach [8]. As noted there, workflow systems must be able to respond to data events, temporal events and external events. One logical development of this idea is to consider the possibility of implementing a complete workflow system inside a database using events as the main mechanism to drive the workflow. In our study, we propose a framework and implementation of document-driven workflow systems. This framework is more flexible than control flow oriented workflow systems and works much better for ad hoc workflows. The rest of the paper is organized as follows. In Section 2 we provide a motivation for our approach with a clear example. Then in Section 3, we give a framework and meta-models for document-driven workflow systems. An implementation of this framework is described in Section 4. Here we discuss our SQL-based implementation for a document-driven workflow system. Finally, in section 5 we discuss the advantages and disadvantages of document-driven workflow systems compared with control flow based systems. The paper is concluded in Section 6.

2 Motivation

In this section, we motivate our approach with a detailed example that compares a control flow based workflow with the corresponding data flow based approach. Fig. 1(a) shows an order process using control flow design. In this process, an order is received, and then the customer’s credit rating is checked. Based on the result of the credit check, either the order is cancelled or the steps of warehouse pickup, shipping, invoicing and close order are performed. (To simplify the case, we ignore the exception handling issues.)

The control flow design puts emphasis on the process, that is, the execution sequence of the tasks. It does not explicitly explain why a task should be performed before another. For example, it is not clear why the Warehouse Pickup task is done before Ship (in Fig. 1(a)), or Invoice is done after Ship. In general, control flow diagrams assume that the process designer has the business knowledge to layout the task sequence. Tasks have various kinds of dependencies between them. Zlotkin [20] summarizes three basic types of dependencies: Fit, Flow, and Sharing, as shown in Fig. 2. Using Zlotkin’s dependency theory, we can find that the tasks Warehouse Pickup and Ship have a flow dependency between them, i.e. the output of task Warehouse Pickup is one of the required