Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 1996 Proceedings

Americas Conference on Information Systems (AMCIS)

8-16-1996

The Role of Document Management in Workflow Management

Zhiping Doug
University of Rochester, dougz@ssb-edpphd.ssb.rochester.edu

Abraham Seidmann
University of Rochester

Rajiv Dewan
University of Rochester

The Role of Document Management in Workflow Management

Zhiping Dong, Abraham Seidmann and Rajiv Dewan
Simon School of Business Administration
University of Rochester, Rochester, NY 14620
dongz@ssb-edpphd.ssb.rochester.edu

Introduction

The document is an important format of information, which is seen as a corporate asset that can be leveraged and used to stay ahead of competition. Documents serve as information stores, but more importantly, they serve as media for developing and communicating information: A document can be a declaration, a request, a permission, or an authorization. Managing document flow is a big part of business activities in information intensive industries. To a large extend the efficiency of such business processes depends on how documents are manipulated and how information content of those documents is shared or passed around. Document management features, such as the percentage of electronic documents, the number of different types of flowing documents, life cycles of those documents, the nature of different types of document manipulations, and the IT deployed to support document storage and retrieval, are crucial to the efficiency of the underlying business process. To understand such business processes and to evaluate and improve them, one should understand how documents are managed. This calls for a model that studies the role of document management in business processes. To this end, we propose a model, *DFM* (Document Flow Model), which depicts document manipulations and flow in a business process.

There are a number of process models in the literature. Most of them have a clear orientation towards the type of application the model is trying to depict. For example, DFD is primarily useful for modeling general data flow in a transaction processing system. Data are treated as static and structured. They only serve as information stores for processes. Document properties are not modeled. Document particular operations such as copying, printing, merging, reconciliation, validation, verification, annotation are not clearly modeled. As a result, how documents are stored, collected, distributed is not part of the description. DFD also lacks components for trigger modeling or resource management. It neither provides constructs for modeling time and queuing effects. Petri nets are primarily used for modeling state transitions typical of the control aspect of many computer systems. This basic model and various extensions have been widely used in discrete flow control applications such as software and electronic engineering (Peterson 81). Although extensions of Petri net for office workflow representation can be found as early as 1977 (Zisman 77), they are not suitable for many aspects of document flow modeling. Data are still treated as static and the mode of information sharing is not part of the description either. Petri nets are computationally intensive, and thus useful only for systems with small state space. Queuing Networks (QN) (Kleinrock 75) on the other end, are used to model stochastic aspects. They capture queuing delays resulted from random job arrivals and processing rates. However, they do not directly contain document flow features.

The DFM framework presented in this paper combines and extends the key features of the three modeling frameworks discussed above. In particular, DFM provides a generic analysis of the document life cycle from creation to storage. It captures documents as data stores for information sharing and as triggers for communication between different functional units. It models the mode of information sharing and passing in a business process. It deals with organizational capacity constraints and resource contention. As a result, one can capture the coordination of information exchange and actions (Shaw and Fox 93) as well as random routing of jobs, which are typical characteristics of a document workflow.

The purpose of developing a specific process model for document intensive business processes is, firstly to understand the role of document management in a typical information intensive workflow environment, secondly, to provide a basis for evaluating the impact of different document management options on the total performance of a business process, and thirdly, to provide a basis for evaluating different document flow control patterns in a business process.

The Model

DFM not only provides constructs to model tasks in a business process, but also provides constructs to model document manipulation and circulation. A DFM model has three parts: *object modeling*, *process modeling*, and *resource modeling*.

Object modeling identifies different objects in the system and their relationships. A DFM object can be either a *document object*, a *signal object* or a *product object.* A document object can be either *electronic* or *paper*. Electronic documents can be *image*, *plain text* or *formatted documents*, such as *SGML* and *forms*. Relationships between documents are related to how documents are processed. For example, a document can be *a-copy-of*, *an-electronic-version-of*, *merge-of*, other document(s).

The process model combines features of the Petri net and Queuing approaches. It specifies *tasks* and *file stores*. Objects and file stores may serve as *inputs* to and *output* from tasks. Tasks correspond to transitions in Petri net terminology. Instead of places, DFM has queued buffers for objects. A task may have many buffer places with any logical relationships among them. As in a Petri net model, tokens are used to represent arrivals at the buffer places. When input buffers are non-empty, tasks are triggered. A triggered task can only be processed when resources it needs are available. There are four different types of tasks in a process: *negotiation*, *composite, manual*, or *automatic*. *Automatic* tasks model tasks that are accomplished automatically without human intervention. *Manual* tasks need human attention. *Composite* tasks represent activities that are supported by information technology but also require human creativity and problem-solving. *Negotiation* tasks model activities that call upon collaboration and coordination among a group of people. A task also breaks down into three parts: *input*, *process*, and *output*, which models different relationships between tasks and documents. For example, a task can *be-triggered-by, scan, manual enter, retrieve*, *create*, *transform*, *update*, *print*, or *copy* documents. *Creation* operations may have different types. For example, *generation* and *re-conciliation* are two different types of *Creation* operation. *Transformation* operations model the type of operations that only change the states of documents. Examples are *verification, validation, authorization*, *editing*, and *updating.*

A DFM *file store* can be either *electronic* or *paper*. The type of a file store is also identified. For example, an electronic file store *inventory* can be a *relational database file*. A task can either *retrieve* or *update* a file store.

Resources are considered servers in DFM. They are represented by a set of *actors, A.* To represent resource management, actors are classified into different *roles, R*. A mapping : R 2^A is designed to captured the skill sets of actors. Another mapping from R to task set T, defined as : R T, captures responsibilities.

To analyze performance issues such as throughput and utilization, time elements are attached to places, tasks, and servers. Time attached to places represents the delivery time from one task to the next. Time attached to tasks breaks down into three parts: time related to input activities, time related to processing activities, and time related to output activities. Time elements attached to servers identify service rates and priority assignments among tasks.

Analysis of a DFM Representation

DFM clearly identifies document manipulation and circulation in an organization. The analysis of a DFM representation is two-fold: evaluate different document management options and evaluate different workflow design options.

Evaluate document management options

For a given workflow system, let r be the ratio of document processing and distribution time over total task time. The efficiency of document management is hence a decreasing function of r. For document management options A and B, if r(A) > r(B), then B is more efficient than A and vice versa.

Evaluate workflow design options

DFM has an added advantage of offering mathematical analysis for document flow design. It provides indications as to where the delay of a workflow originates, hence provides guideline as to how to remedy the problem. Here are some of the results from the analysis of a DFM representation.

- When two tasks communicate with each other through paper documents, great delay can be incurred due to the possible slow delivery of paper documents. As a result, these tasks are either subject to redesign or to adoption of electronic document management systems or workflow management systems.

- Re-conciliation of paper documents can take a great deal of time, and also are prone to human error. As a result, a reconciliation operation may either be decomposed into multiple reconciliation subtasks or it may be converted to a set of transformation operations.

- A sequence of transformation operations suggests a parallel redesign.

- Sequential tasks that can be processed by personnel with the same skills shall be assigned to the same person.

- When some personnel are skilled in processing different tasks, priority design among those tasks affects the performance of the whole system.

References