Distributed and Parallel Databases, 4, 229–247 (1996)

Exotica/FMDC: A Workflow Management System for Mobile and Disconnected Clients

G. ALONSO alonso@inf.ethz.ch
Department of Computer Science, ETH-Zürich, ETH-Zentrum, CH-8092 Zürich, Switzerland

R. GÜNTHÖR rgunthor@heidelbg.ibm.com
IBM European Networking Center, Postfach 10 30 68, 69020 Heidelberg, Germany

M. KAMATH kamath@cs.umass.edu
Department of Computer Science, University of Massachusetts, Amherst, MA 01003, USA

D. AGRAWAL, A. EL ABBADI {agrawal,amr}@cs.ucsb.edu
Department of Computer Science, University of California, Santa Barbara, CA 93106, USA

C. MOHAN mohan@almaden.ibm.com
IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120, USA

Received May 1, 1995; Accepted February 12, 1996

Recommended by: Daniel Barbara, Ravi Jain, Narayanan Krishnakumar

Abstract. Workflow Management Systems (WFMSs) automate the execution of business processes in environments encompassing large numbers of users distributed over a wide geographic area and using heterogeneous resources. Current implementations allow the definition and controlled execution of complex and long lived business processes as the basis for an enterprise-wide collaborative system but, in most cases, the autonomy of the users is greatly restricted due to architectural and design considerations. In particular, existing systems are built around a centralized server. As a result, users need to maintain an uninterrupted connection with the server to perform the different tasks assigned to them. This is a severe restriction, especially when considering the emergence of mobile computing, and the increase in use of laptops and small computers which are connected to the network only occasionally and which will, undoubtedly, be the tool of choice for many users. This paper addresses the problem of supporting disconnected workflow clients in large workflow management systems while still preserving the correctness of the overall execution and allowing coordinated interactions between the different users regardless of their location.

Keywords: Workflow, Disconnected Operation, Mobile Computing

1. Introduction

Workflow Management Systems, WFMSs, are seen as a key tool to improve the efficiency of an organization by automating the execution of its business processes. A WFMS supports the modeling, coordinated execution and monitoring of the activities that take place within an organization. It is up to the user to define such activities and organize them in the most efficient way using the tools provided by business process re-engineering [13], but once the activities and processes of interest have been defined, the workflow management system is used to represent the business processes and to assign the staff and role hierarchies in the organization within which those business processes will be executed. During the execution of the business process, the workflow management system acts as a coordinator: the WFMS delivers the various tasks to each user, collects results, determines the next steps, controls the activities of each user, and detects when the process has successfully terminated. The patterns of collaboration among the users are predefined as dependencies between individual steps within a business process, with each step being assigned to potentially different users. Thus, the synergy between all the steps is provided by the designer of the business process. Note that full automation is not possible, since human intervention is necessary to solve many crucial steps and to determine what to do in case of errors and unpredictable events. However, the use of a workflow management system simplifies to a great extent the task of coordinating large numbers of users working in heterogeneous and distributed environments.

Many existing WFMSs are built based on a client-server architecture due to the simplicity of the design and the synchronization problems posed by other architectures [4, 3]. Such an approach has many advantages and it is useful in many organizations where the server is installed in some central computer and users access the system through terminals, PCs or workstations, installed in their offices. Most clerical work, form processing, accounting activities, and stock management, to name a few applications, is done this way. However, there are many other applications where this may not be the best approach.

Recently, *disconnected operation* has been identified as one of the main ways in which computers will be used in the future [17]. Taking advantage of the arrival of more reliable and powerful portable and home desktop computers, users within an organization can work independently of the main computer facilities: applications and data are loaded in the laptops or desktops by briefly connecting with a server, the connection is broken, and users work locally on those applications and data. After the work has been completed, which may be in a few hours or few days, users reconnect with the server and transfer the results of their work. Disconnected operation offers many obvious advantages but, in many ways, disconnected computing and workflow management systems have contradictory goals. A workflow management system is a tool for cooperation and collaborative work in which users work within a preestablished framework that guarantees progress towards a certain goal, the business process, of which the users may not be aware. This requires constant monitoring and checking of the users' activities. On the other hand, disconnected computing is geared towards supporting users who work in isolation from other users. There is not much room for collaboration in disconnected mode.

This paper addresses the problem of supporting disconnected clients in a large workflow management system. The goal is to give enough autonomy to the clients to allow them to perform work without having to be connected to the rest of the system and still maintain the overall correctness and consistency of the processes being executed. To bridge the gap between disconnection and coordination, we propose a compromise between both worlds. Users must "commit" themselves to perform certain tasks before disconnecting from the system. The workflow management system takes advantage of such commitment to assign tasks to users, allowing them to work on their own while ensuring overall correctness and constant progress towards the goal of the business process. The ideas described in this paper is one of the first studies on the impact of portable computers on collaborative workflow systems [2, 5], and the first to provide a feasible solution where the implementation aspects are discussed within the constraints of a real system. For this purpose, we use FlowMark [18, 19], a workflow product from IBM that uses a client-server architecture, where the ideas described here have been implemented.

The rest of the paper is organized as follows. Section 2 presents the basic ideas behind workflow systems pointing out the architectural and conceptual problems to support disconnected operation. Section 3 describes the functioning of the system when clients are connected to the system at all times. Section 4 discusses possible solutions for supporting disconnected clients in terms of three different phases. Section 5 addresses the problem of locating clients in the system when they are mobile and can connect to different servers. Section 6 discusses some of the issues raised in previous sections and the implemented prototype. Section 7 summarizes related work and Section 8 concludes the paper.

2. Workflow Systems

This section introduces the basic concepts of workflow management according to the definitions provided by the *Workflow Management Coalition*, *WfMC*, in its *Reference Model* [14, 22]. The WfMC is an international organization leading the efforts to standardize workflow management products. Implementation details are discussed based on FlowMark [18, 19], IBM's workflow product.

2.1. Business Processes and Workflow Management Systems

The reference model defines a *business process* as "a procedure where documents, information or tasks are passed between participants according to defined sets of rules to achieve, or contribute to, an overall business goal" [14]. An example of a business process is shown in Figure 1. Some of the steps within a process may also be complex processes themselves. For instance, in Figure 1, *Case study* and *Finalize credit* are shown as nested processes. From this definition, it is easy to see the contradictory goals of automating the execution of business processes and providing support for disconnected operation. The key idea in any workflow system is to pass "documents, information or tasks" around the participants (as *forms* [29], *electronic mail* [12, 23], or transactional activities [11, 27, 30], for instance), while disconnected operation is geared towards user's autonomy.

A business process can be expressed as a schematic representation of the procedural knowledge related to certain activities. This is generally called a *workflow*. Hence, a *workflow management system*, WFMS, is "a system that completely defines, manages and executes workflows through the execution of software whose order of execution is driven by a computer representation of the workflow logic" [14]. Once again, note that a WFMS automates the execution of workflows, which are representations of business processes and, hence, "automation" acquires different meanings depending on the type of processes. For business processes in general, this automation involves performing several tasks: scheduling activities, mapping activities to users currently in the system, tracking the progress of activities, assigning resources to activities, and so forth. In existing systems these tasks are

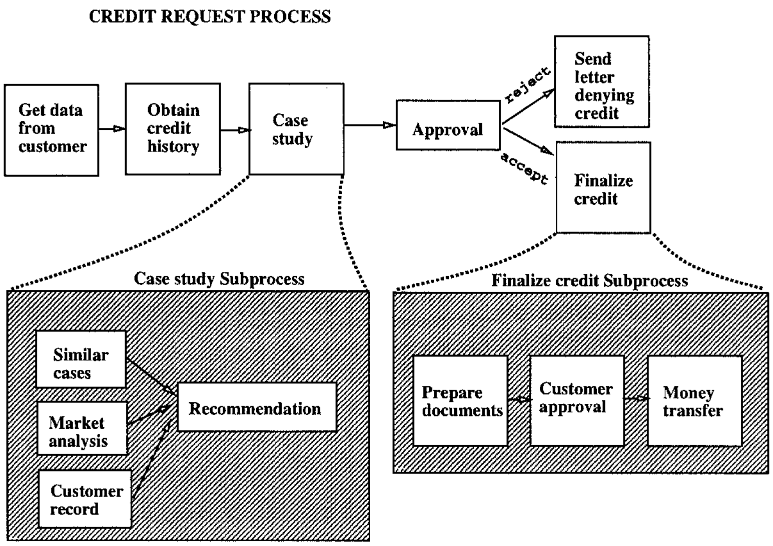

Figure 1. A Loan Request as an Example of a Business Process

all performed by the server, hence, to accommodate disconnected clients, some modifications to the overall system design will be necessary to guarantee progress without clients having to access the server at each step.

2.2. Workflow Model

The Workflow Management Coalition defines a *process* as a coordinated set of activities connected in order to achieve a common goal [22]. The automated components of such a process forms a *workflow process*. For simplicity, in what follows we will use the term process to refer to a workflow process, i.e., we will assume a process can be automated in its entirety. A *process* consists of *activities* and *relevant data*. Activities are the different steps of the process and associated with each of them is an *invoked application*, the application to execute, and a *role*, the set of users responsible for the execution of the activity. The *invoked application* can be a computer program or a human activity. The *relevant data* is defined as the data being transferred among activities. The flow of control within a process, i.e., what to execute next, is determined by *transition conditions*, usually boolean expressions based on relevant data. Users are represented in terms of *roles*, and activities are generally associated to roles instead of individual users. This allows the system to determine who is currently available and assign the activity to that user instead of having to wait until an individual user logs on. The invoked applications can be almost anything as long as there is a way to communicate their result to the WFMS. In the case of applications being programs, *Application Program Interface*, API, calls are used to access and return data.

One of the most relevant aspects of a WFMS is the use of *worklists*. Worklists can be seen as the interface of the WFMS to the end user. A worklist is a list of workitems associated with a particular user, and each user has one. Each workitem is an activity that belongs to a process being executed and that has been assigned to this user, and possibly also to others with the same role, for its completion. Hence, the same activity may appear in several worklists at the same time. In this case, when an activity is selected in one worklist, it will be deleted from all other worklists. The mapping between roles, users, activities and worklists is automatically performed by the system. All this functionality must be maintained during disconnected operation.

2.3. *Workflow System Architectures*

WFMSs have various possible architectures [3], but for the purposes of this paper they all have similar characteristics. In general, the functionality of a workflow system is distributed among three different components: *runtime control*, *runtime interactions* and *buildtime* [14]. Runtime control has two aspects to it: *persistent storage* and *process navigation*. Persistent storage allows the system to recover from failures without losing data and also provides the means to maintain an audit trail of the execution of processes. The navigational logic controls the execution of processes. Thus, we consider two components within runtime control, the *storage server* and the *navigation server*.These are referred to as the Workflow control data and the Workflow Engine in the reference model. Similarly, runtime interactions are of two types, with the users and with invoked applications. The former is the interface with the end users and consist mainly of the worklist assigned to a given user. The latter is the interface to the applications being executed as part of a workflow. We consider them as separate components, the *User Interface* and the *Application Interface*. These appear in the reference model as *Worklist* and *Invoked Applications*. We will not discuss the details of buildtime operations since they are somewhat orthogonal to the actual execution of processes.

2.4. *FlowMark's Model and Architecture*

The reference model does not provide any guidance in terms of implementation and lacks a detailed description of the interactions between the different components. For this purpose we use FlowMark, an IBM product which closely follows the reference model. Except for some minor implementation differences, the ideas described in the rest of the paper should apply to any WFMS that follows the functional specifications of the Workflow Coalition reference model.

Business processes are modeled in FlowMark as acyclic directed graphs in which nodes represent steps of execution and edges represent the flow of control and data [18, 19]. The main components of FlowMark's *workflow model* are: *processes*, *activities*, *control connectors*, *data connectors*, and *conditions*. A process is a description of the sequence of steps involved in accomplishing a given goal and it is represented as a graph. Activities are the nodes in the process graph and represent the steps to be completed. Control connectors

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**