The OpenWater Project – A substrate for process knowledge management tools

Authors: Keith Whittingham, Markus Stolze, Heiko Ludwig
{wgk, mrs, hlu @zurich.ibm.com}
Date: 7th October 1999

Abstract

Process support tools, such as workflow management systems, need to be 'industrial strength' before they can be used to support real work. This poses a problem to researchers wanting to build complementary tools or experiment with process data as building a framework of this quality is expensive and time consuming. We propose a framework that provides such a substrate supporting semi-structured organizational processes with hooks to plug in process support components. The platform offers the opportunity for modules such as artificial intelligence tools to use, and be used in, real-world process instances thereby contributing to knowledge management for organizational processes.

1. Introduction

Today's computer systems may still be some way off replacing humans in the capacity of performers of some business tasks but, particularly when networked, they are very good in a supporting role. An opportunity exists to use existing and new developments in the knowledge management and artificial intelligence fields to enhance the existing work in the business process management arena.

Workflow management systems, as tools to enact business processes, perform four principle functions:

1. Control the order of activities that make up a business process
2. Control the allocation of activities to the eventual performers of the activity
3. Control the flow of data between the performers
4. Integrate the various tools that are used to perform the activities.

Being so central to the business process they are prime candidates as a starting point for new work but there are drawbacks.

Building a WfMS is a serious undertaking. WfMSs require a buildtime environment to generate process models and a runtime environment to enact them. The underlying data model is complex. Additionally if the system is to be used to support real workflow then persistence, data sharing, transactional processing and very high availability must also be addressed.

The implementation of workflows is also burdensome. WfMSs use a pre-defined process model as a basis for the process enactment. Designing process models is an arduous task; generally a 'clean-case' workflow is defined and then handling for exceptional cases is added. In most business processes there is no shortage of these, so-called, "expected exceptions". Building a model calls for a high level of expertise in process modeling as well as in the process domain. Once working correctly, the WfMS becomes critical to the business and failures are hard felt.

We propose a workflow support system that has several unique features that make it particularly useful as a basis for research and experimentation. We are currently using the system in in-house as well as external pilots.

2. Architecture

OpenWater is the name of a project that started towards the end of 1998 in the IBM Research Laboratory in Zurich. The principle goal of the project was to address the problem of capturing the business process in a computerized model by providing a system that discovers the business process. Early on in the project the opportunity to use the tool as the basis for knowledge management and process mining became very apparent. For this reason an open architecture has been adopted in the hope that this will foster other projects in this area.

The key features are:

1. **Openness**: A mechanism is provided to allow plug-in modules to interact with the workflows and workflow data. Plugins are easy and quick to build and run in a sand-box so that the integrity of the business process is not compromised
2. **Semi-structured messaging**: Workflow traffic is passed between the participants in the business process using a novel mechanism allowing relationships between the participants to be established in a dynamic and pragmatic way
3. **Immediate**: There is no need to pre-define workflow models.

4. **Process traces**: Workflows leave traces in an 'organizational memory', a valuable resource to be used as the basis for knowledge management and process mining

2.1 Overview

Figure 1 below shows a sketch of the architecture. Electronic Circulation Folders are used to represent process instances. An ECF (A) is constructed from an ECF template in the ECF factory. The ECF is sent to other participants in the system (B, C, D and E) as required, each user adding to or modifying the process data as necessary. Important events are reported to the Organization Memory, where they are reconstructed into cases and by the case interpreter. The information gained is made available to future instances of ECFs.

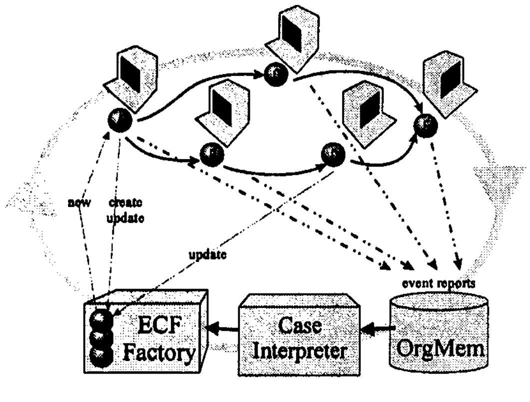

Figure 1. Architectural overview

2.2 Philosophy

We distinguish between a Workflow Management System (WfMS) and a Workflow Support System (WfSS) as follows. Classical WfMSs conforming to the WfMC's reference model (WfMC 1990) such as IBM's MQSeries Workflow™ (IBM 1999) *manage* or control the flow of work between participants in the workflow. The workflow process must be captured and modeled in a process template before it can be enacted by the system. In contrast, the OpenWater project simply *supports* the flow of work between the participants but leaves the addressing of the work to the discretion of the users.

Whilst this promises to provide better support for ad hoc workflows and exception handling there are some occasions where it is necessary to enforce some level of compliance with business rules. To this end, business rules or *policies* can be defined and applied to reusable fields. This enables the definition of constructs such as 'a signature field can only be signed by a manager and the signature locks the quantity, part number and cost fields'.

2.3 Components

2.3.1 Resources

The resource model allows all entities that contribute to the workflow to enter the system.

A resource is either a person or a computer program capable of performing some part of a business process. The notion of group is also supported: groups can contain participants or other groups, allowing hierarchies to be constructed. Groups, persons and programs are considered types of resources.

The notion of relationships between two resources is also supported, this is useful when defining policies.

2.3.2 ECF Templates

In the OpenWater project, work items are represented as 'Electronic Circulation Folders' similar to those used in the PoliTEAM project (Karbe, Ramsperger and Weiss 1990). ECFs are sent between participants of workflow process to inform the receiver that some service is expected and to pass the process data. ECFs are created from a reusable ECF template that contains a pointer to process documentation and a definition of the process data. A different ECF template is used for each process type: a user might have access to templates for travel requests, equipment ordering, travel expense processing and so on.

2.3.3 Fields

The process data in an ECF consists of a structured list of fields. Each field may either be a data field or a structure itself containing more fields. Fields may either be placed directly in the ECF (one-off), in the ECF template (common to all instances of a business process) or they may be named and used as reusable components in several ECF templates. Named fields can be used in conjunction with policies.

2.3.4 Policies

Certain processes require some level of compliance with business rules; the notion of policies, similar to that proposed by Bussler (Bussler 1995) is introduced to support this. An example of the use of a policy is where a purchase order must be signed. The signature, a field, signs a number of other fields such as part number, quantity, total cost, etc. A policy specifies the relation between the different fields involved, defines the desired value of the field and controls the access and visibility of the fields.

2.3.5 The user interface

The user interface is relatively simple consisting of a single window and a few dialog boxes. The main window has three trays: the in-tray contains incoming ECFs, the working-tray contains ECFs currently being worked on by the user and the pending-tray contains a list of active ECFs currently of interest to the user.

2.4 The plugin framework

In order to ease the development process, a framework that allows plugins is used. This allows the workflow support to be extended by third parties under the control of the user. Viewers or editors can be provided to support the user in any number of different situations, such as:

- Provide a process data editing tool for a particular task
- Provide context dependent, just-in-time help such as selected examples of previous instances of the process at hand
- Attempt to suggest routing for ECFs based on assessment of previous process instances in order to optimize workflow performance
- Automatically handle easy workflow cases on behalf of the user

A plugin is generally a visual component that 'fits on top of' one of the in, working or pending trays. In general: in-tray plugins work with workitems that are in the user's to-do list; working-tray plugins provide enhanced views, help or editing tools for particular tasks; pending-tray plugins watch or act on workflows that the user has some interest in.

The interface between the workflow support system and its plugins is limited to a need-to-know basis. This effectively builds a sandbox around the plugin increasing overall system reliability. The plugin can specify which trays and which ECF types it can work with. Each user can further refine these options to tailor the system as required.

Plugins are written in Java™. The plugin is installed by copying a JAR file into a predefined directory on the client machine and is loaded automatically the next time the client starts. There are no limitations regarding access to other system resources such as JNI™ or JDBC™.

2.5 Technical

The software is written entirely in the Java™ programming language making extensive use of JDK™ 1.2 and the Swing™ graphical user interface. Data is stored centrally in an IBM DB2™ database using a proprietary library to store Java™ objects in the database.

3. Conclusion

The possibility to use business process management tools in conjunction with knowledge management techniques and artificial intelligence is potentially very rewarding. Basing research projects on classical workflow systems is difficult. Implementing a workflow system as part of the project is an even more daunting task.

We attempt to resolve these problems by presenting an open and extendable framework to build process knowledge management tools. The substrate provides the basic workflow plumbing allowing researchers and developers to concentrate on more interesting issues. Additionally and in contrast to other workflow systems, OpenWater does not dictate the flow of work to the participants of the workflow presenting a unique workflow watching opportunity.

4. References

Bussler C. J. 1995. Policy Resolution in Workflow Management Systems, DEC Technical Journal, volume 6 number 4

International Business Machines Corporation 1999. IBM MQSeries Workflow™, Concepts and Architecture, Version 2 Release 3

Karbe B., Ramsperger N., Weiss P 1990. Support of Cooperative Work by Electronic Circulation Folders. OIS90, Coordination Technology p 109

Workflow Management Coalition Reference Model 1990. The Workflow Reference Model Workflow Management Coalition : WfMC-TC-1003, 1.1