

The Emergence of Workflow for Coordinating Business and Software Processes

Keith D Swenson
Fujitsu OSSI

Workflow software is probably the fastest growing software market in the United States. This is caused by a confluence of three trends: first the worldwide recession is causing many companies to look hard at how to be more effective with fewer people; second, the presence of good fast networks enable work in new ways; and third, the management revolution known as business process re-engineering. Software developers benefit two ways from this: 1) use of workflow software can help support software processes for the development of the software, and 2) workflow is an effective tool to integrate existing applications to coordinate larger organizational processes without requiring large custom applications. There are a lot of flavors of workflow. Regatta Technology, from Fujitsu, supports some advanced features of workflow that are critical to dynamic organizations.

Workflow Background

Workflow software draws its roots from the office automation effort of the 70's and early 80's. Evolving from a number of different sources, the single biggest contributors were document imaging companies and their dream of eliminating paper from the workplace. Documents entering the organization were scanned into an image server, and then forwarded to the recipient. Ideally people would be able to share and exchange these document images as easily as the original paper versions. Workflow was invented as a way to route an image through a predetermined path from person to person. This document-centric view of work is limited, but has been the dominant model for workflow until recently.

The availability of fast reliable networks has caused a resurgence in interest in the ways that people can communicate through the computers. This alone does not explain all the interest. At the same time global recession has caused a lot of companies to reexamine how they can be more effective with the resources they have. This in turn helps fuel the interest in the results of the MIT Management in the 90's study that formed the basis for what we call today Business Process Re-engineering. The main point behind BPR is that if we reexamine how work is being done, and completely redesign it around the fundamentally different ways of working offered by information technology we will see improvements in productivity.

Workflow, with its inherent orientation toward processes, becomes an ideal candidate for supporting these new reengineered work processes. At the same time work processes are proving to be more elusive and complex than previously realized, which is causing a proliferation of new approaches to workflow. Categorization of the existing types of workflow software is difficult and even industry analysts disagree on how it should be done. This is a sign that the industry is still young and that we do not yet fully understand the technology. There are still

openings for new and innovative designs to gain general acceptance, and at the same time there is a rapid push to standardization. The best estimates are that by spring 1995 the workflow industry will mature such that the basic kinds of workflow will be clear to a majority of the potential customers and users.

Business Processes

Business processes are already being handled by custom or semi-custom applications. Consider applications such as those used for running a hospital, or an airline, or a newspaper. When such an application needs to involve a number of people to complete a task, the rules and logic for involving these people are coded into the program itself using whatever language the program is written in. This seems natural now, but is quickly becoming outdated.

Workflow will become so common in the next few years that every large company will have a workflow server (engine) of one form or another. The result will be that the rules and interactions between people and tasks will be separated from the applications that the people are using. This is a bold conclusion drawn from some undeniable trends and needs. To best understand it, consider for a moment the introduction of database technology into the corporate world.

In the late 70's and early 80's most large data processing applications were custom or semi-custom applications that stored their data in proprietary formats. An application written in COBOL would store the data in COBOL pictures. An application in C would use C structures. Database technology was then introduced to the application developers and it provided some benefits: first, the most obvious benefit was that it allowed applications to share data without reading proprietary formats. This made it much easier for users and system integrators to piece together systems and, for example, to add better reporting packages. The second benefit was that database developers could optimize their products to be more efficient. The third benefit: since the application defined the view of data it needed it was

independent of the actual underlying database schema. This gave the freedom to upgrade the application, without losing the data; or to upgrade one tool without being forced to upgrade or rewrite the entire suite of tools. The third benefit is actually the most important because it saves companies from being locked into outdated programs.

Workflow has analogous benefits for each of these points. First, by separating the coordination rules from the application, it allows existing applications to be integrated into a single workflow more easily. This is even more important for workflow than for databases because within a process different people are doing different tasks and need different tools for that task. For example a writer will use tools specialized for writers, a lawyer will use specialized legal tools, they would both be severely limited if in order to work together they had to use the exact same application. Second, workflow software can define processes in a more refined and optimized way. Standard procedural languages are not designed specifically for coordination. Thirdly, the separation of application and process rules allows a company to change the process rules without having to rewrite the applications and to upgrade the applications without losing the process definitions. This third benefit is even more critical for workflow because while the database is typically modeling real world physical things which generally do not change much, workflow is modeling tasks and activities. An activity has no existence except as an idea or a shared concept of what is done. We rarely notice it, but task definitions within an organization are constantly changing and evolving to accommodate new people, new technology, new laws, new products, new techniques, new customers, or new competitors. The fluid nature of processes requires a very flexible system.

Software Process

Software developers are effected by workflow in two ways. First is that the workflow engine can be used as part of a enterprise wide system to support processes along with either new or existing applications. From the developer's perspective the workflow engine's Application Programming Interface (API) is most important. Developers who do not want to be tied to a single workflow vendor will demand a common API that will unify the workflow field in a way similar to the way SQL unified the database field.

The second effect workflow will have on software developers is when workflow supports software development processes. Because software development is a multi-person multi-discipline activity, it is a good candidate for coordination by workflow.

Process support and automation are not the same. Much of the study of software processes has focused on automation of the software development environment. A classic example is the attempt to specify a set of rules such that a programmer's actions will trigger an automatic build and test of the program. While this kind of automation is very

important, workflow might not be the best tool for this. An automated build is essentially a single-user activity, and there exist very good tools for programming and automating these. Where workflow can add a unique benefit is in the support of multi-user non-automated activities such as code review or design approval. "Not-automated" here means that the focus is not to code the skill that the person brings to the activity into the system. Instead of de-skilling the job in this way, workflow treats people as intelligent actors within the system and attempts to provide an environment where people can effortlessly communicate and coordinate their actions.

Workflow Overview

There are three main kinds of workflow. The first is termed "image routing" because it grew out of document image systems. This category of workflow is usually quite simple and often consists merely of a list of people that the document is routed to. These people can either forward the document, or send it back to the beginning. More complicated flow patterns are possible by programming them using a traditional programming language.

Ad-hoc workflow is the second category. Workflow systems in this category give users a choice of basic flow patterns to communicate with. The Coordinator is a good example of this style.

The newest and most interesting category is Object Oriented Workflow. This category is really just a term that tries to encompass all the new workflow paradigms that do not fit into either of the other categories. Typically these systems employ object oriented technology that brings a level of flexibility to the system not previously possible.

All three categories can be called first generation workflow. 1G workflow is concerned with representing the business process accurately and enacting it for the users. The focus is on the process itself. But there is another process to be concerned with, and that is the process of developing process definitions. 1G workflow treats process descriptions as if they were static. Second generation workflow will have features that take into account the fact that processes are changing and being improved, and they will include support to help in the evaluation of processes and in the improvement of them. Second generation workflow will allow continual improvement of processes.

Elements

Workflow systems are usually composed of a number of elements. These are outlined below.

Workflow Engine: The workflow engine provides the logic to enact the processes. The workflow engine is often implemented as a server in a client-server environment, but it might be implemented in a fully distributed manner. In any case, the engine is the logical center of the system, and the other modules interact with it.

Process Definition Tool (Planner): All workflow systems must have some way to define processes. This is done with a process definition tool in a way that is compatible with the model that the workflow system is based upon.

Worklist Tool (Viewer): All workflow systems must have some way for people to find out the tasks that are currently assigned to them. From the worklist, the user selects work items. Each workitem is associated with a set of documents needed to accomplish the task. Ultimately the user will in one way or another "complete" the task, telling the system to remove this item from the list, and possibly adding other items to other people's lists. The worklist tool is the way most people interact with the system.

Invoked Applications: Some steps in the process will be automatic invocations of external applications. These are the tools that are integrated together by the workflow system.

Reporting and Statistics: One of the main benefits of workflow is the greater control over the processes. Most workflow systems have some way to report on process status and to identify all of the process instances that are currently in a particular stage. Another important feature is the collection of statistics about how long certain tasks take to complete, and where the bottle necks are. This information can then be used to improve the process descriptions, or to otherwise change the allocation of tasks within the organization.

Administration: The administration tool is used to set up users, install invoked applications, and a number of other exceptions activities.

Workflow Management Coalition

The workflow management coalition is a coalition of 86 workflow vendors, users and analysts dedicated to developing standards for interoperability of their system. These vendors realize that they are not isolated, and can not count on being able to exclusively service all the workflow needs of a given client. Adoption of electronic mail was hindered by many vendors supporting only their proprietary message exchange standards, and users of a particular email system were not able to send to users of a different email system. Workflow is essentially communications also, and the hope is to avoid the situation that email users were in by working proactively to get workflow systems to interoperate. Work is proceeding along a number of lines.

Terminology: Workflow is such a new field that different vendors are often using completely different terms to describe the same things. This is confusing to the users and potential customers. The first task for the WMC is the development of a standard set of terms to be used to describe workflow. This glossary will be made available to the press and users, as well as the vendors themselves. It is predicted that once a common terminology is available there will be an increase in the awareness and understanding of what workflow can offer to users.

Reference model: The second task of the WMC is to define a standard reference model which defines the common parts of all workflow systems. This reference model is then useful to the vendors as a system to compare to. The elements of workflow mentioned above are part of the standard reference model.

Five key interfaces: From this model, 5 key interfaces have been identified as candidates for standardization. They are: 1) process definition tool interface, 2) worklist manager interface, 3) invoked application interface, 4) server to server interface, 5) administration and monitoring interface.

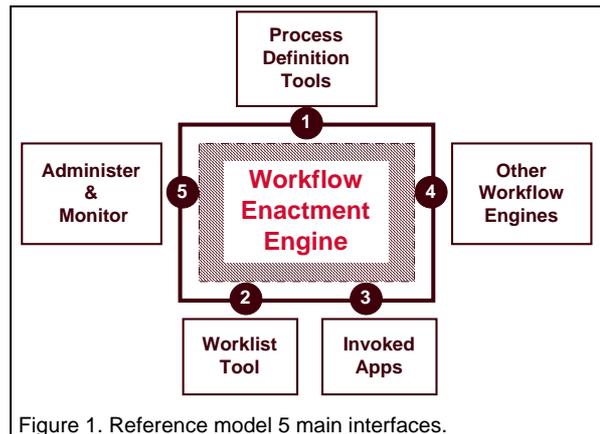


Figure 1. Reference model 5 main interfaces.

Regatta Technology

Regatta technology, developed by Fujitsu, is a good example of an advanced workflow system that can demonstrate the advantages of using workflow.

Regatta models the communications required to coordinate processes. The "activities" in a Regatta plan actually represents requests from one person to another about the activity. Because both the originator and the assignee of a request are known, a number of important social behaviors are allowed which are not possible in other systems.

Requests can be created on-the-fly, or automatically as part of a process plan. Requests function much like electronic mail, with the added benefit that all correspondence may share information within the context of the request. The recipient may accept or decline the responsibility. Regatta supports the human process of negotiation to determine exactly what task is to be performed, and when. If accepted, then that person is expected to complete the task at some later time.

Recipients have full control over how a request is to be satisfied. They may make requests of other people, set up a request contingent upon the completion of an outstanding request, or invoke a template for a process that has been prepared beforehand for handling such requests. The template might be an organizational standard, or an individual variant. By leaving users in control, Regatta empowers them to do their work, without restricting their options.

The Visual Process Language

A key component of the system is a graphical tool that makes it easy to diagram processes. These diagrams are composed of stages (requests) which are represented by ellipses. Every option is represented by an arrow coming out of the stage and pointing to a stage that will be subsequently activated. The graphical tool requires no more technical sophistication than a spreadsheet.

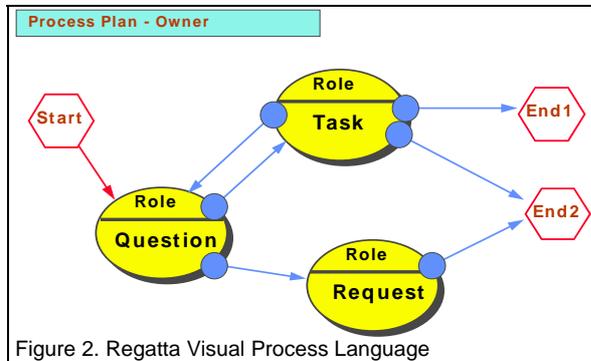


Figure 2. Regatta Visual Process Language

The visual language is effective in helping users to understand the process and the current status. A picture of the process is far easier to grasp than a binder full of textual descriptions. Understanding the current process is the most important factor to be able to improve the process. As a process is actualized, the process status is displayed with colors indicating whether the stage is completed, in process, or not yet started.

Regatta facilitates the discovery or "capture" of organizational exception handling processes.

3. The creation of process policies is a collaborative activity. Different parts of the organization will automate those parts that are important to them, in the way that best fits their needs.

Regatta is radically different from other workflow tools because the process does not need to be pre-defined by someone outside the process, and then introduced to the entire organization en-mass. By making it easy to create and try different process

descriptions, users are encouraged to refine and maintain their own segment of the organization.

Regatta allows groups to decide for themselves whether to support a process or to operate manually. Users define only those processes that will benefit.

This sounds obvious, but when required to define the process completely before using it, it is not always clear which parts should be automated, and which are a waste of time. Work processes often involve tacit knowledge: tasks learned by trial and error that have never been expressed externally in a written or verbal form. It is unreasonable to expect workers (who are not experts on group processes) to be able to describe tasks -- even those tasks done routinely every day -- accurately enough for an independent process programmer to implement. Regatta's design-as-you-go approach allows users to make requests as they need them. Then, after completion, one can go back and see what was really done, how effective it was, and how to make it better next time. This is process improvement.

Regatta Technology is "organization aware". The policies that you use depend upon the policies you have made for yourself, or upon policies for your group, and finally upon policies for the entire organization. This allows a new user to get started using the "default" policies of the organization, but then to customize those policies as needed to match personal style, or the way the local group works.

Summary

Workflow is rapidly becoming a major software category. Workflow engines will provide useful services to applications developers, as well as supporting the software development itself. Regatta Technology from Fujitsu is an example of second generation workflow technology which is concerned not only with processes, but also with how those processes are discovered, developed, and maintained.