# An Infrastructure for Managing Knowledge using Intelligent Workflow

Irma Becerra-Fernandez, Ph.D.*
Karen McCarthy**
Juan Rodriguez**
*College of Business Administration, Decision Sciences and Information Systems
**Knowledge Management Lab
Florida International University
Miami, FL 33199
becferi@fiu.edu

**Abstract**

Today's business environment is characterized as knowledge being the critical resource for any business activity. By automating many of their routine business processes, companies are able to save time and valuable human resources. Workflow management systems can serve as the basis for collaborative computing, as evidenced by their growing popularity. This paper describes current research efforts to develop a collaborative computing environment, Postdoc-Next Generation (NG). The development of Postdoc-NG is based on Postdoc©[1], a document management system developed at NASA-Ames Research Center, and currently used throughout NASA. This effort requires the redesign of the current Postdoc architecture with a focus on knowledge management (KM), including the handling of metadata, supporting workflow application logic, and an object oriented architecture to facilitate user-managed data collection.

## Introduction

Today's business environment is characterized by knowledge being the critical resource for any business activity (Becerra-Fernandez et. al., 2000). The environment is characterized as one with:

1. Increasing complexity: The complexity of the underlying domains (internal, external, competitive, process, technology, etc.) is increasing;
2. Accelerating volatility: The pace of change (volatility) within each domain is increasing;
3. Speed of responsiveness: The time required to take action based upon subtle changes within and across domains is decreasing; and
4. Less experience: Individuals with decision-making authority potentially have less tenure with the organization than ever before, due to such factors as high employee turn over rates.

Traditional information systems are based on a consented interpretation based on the company's business culture and management's needs. Computer generated information typically does not lend itself to interpretation that produces action, and knowledge implies action based on the information. Today's fast paced highly competitive business world forces the need for variety and complexity in the interpretation of information generated by computer systems.

Group-decision making tools can help companies make better decisions by capturing the knowledge from groups of experts. Furthermore, companies that capture their customer's preferences can improve their customer service, which translates to larger profits (Becerra-Fernandez et. al., 1998a). Organizations are now concentrating their efforts not only in gathering documentation, but also in discovering new knowledge, by mining the knowledge and experiences of their employees, customers, and competitors.

In its purest sense, workflow represents the automation of a business process. Workflow Management Systems (WfMS) are tools that support defining, creating, and managing the execution of workflow processes[2]. In other words, they provide a method of capturing the steps that lead to the completion of a project within a fixed time frame. WfMS such as factory assembly lines have been around for some time. By automating many of their routine business processes, companies are able to save time and valuable human resources. WfMS can serve as the basis for collaborative computing, as evidenced by their growing popularity. Workflow systems provide a mechanism for the analysis and optimization of the entire process involved in completing a project. One benefit of using a WfMS is that it

[1] © NASA (1996-1999)

[2] Workflow Management Coalition, Terminology and Glossary (1999)

provides the user with an audit of necessary skills and resources prior to project initiation. Workflow systems also provide a platform for the replication and reuse of stored processes. Finally, WfMS can also serve as training tools since they provide a broad overview with detailed operations of tasks as well as an identification of possible "weak links" in a process.

The Web provides a readily available communications infrastructure for the building of WfMS. WfMS that provide Web interfaces are referred to as web-enabled, while those that use Web technology as their underlying communication infrastructure are considered web-based. For example, Lotus Notes ™ [3], the most widely deployed document management system, was developed as a client-server application. Both the emergence of the Web as the portal of choice for information systems, motivated Lotus to develop Domino ™, it's web-enabled front-end for Lotus Notes.

This paper describes current research efforts to develop the collaborative computing environment Postdoc-Next Generation (NG). Our current research efforts are aimed at re-designing Postdoc into a web-based document management system that takes full advantage of the web by using it as its underlying infrastructure. The workflow component is being designed to take advantage of the communications and interfaces that the web offers. The motivation for this new development is based on the current use of Postdoc by project teams throughout several NASA centers. The addition of the workflow functionality has been requested by many of Postdoc's current users. Therefore, augmenting Postdoc with workflow supports the innovative efforts of these teams working anywhere, and at any time.

## Postdoc-NG System Overview

The Postdoc software required a development effort of five person-years of software coding and testing, beginning with the New Millennium Electronic Documentation Project in the Computational Sciences Division at NASA-Ames Research Center (Becerra-Fernandez et. al., 2000). The New Millennium Electronic Documentation Project was the first NASA web-based Intranet defined for a very large program involving six Deep Space missions, three Earth Orbiting missions and six technology teams in 1995. This effort involved studying how widely dispersed teams interact and what web-based tools are most effective for overcoming their geographical differences and varying computational environments. Postdoc is a multi-user, web-based document management system that is currently used primarily for document storage and retrieval, including word processing documents, spreadsheets, slides, illustrations, images, video, audio, software archives, and others. Currently, Postdoc's architecture is being re-designed in Postdoc-NG, with a focus on KM, to be able to handle metadata and support workflow application logic. An object-oriented (OO) architecture facilitates the objectives of allowing user-managed data collection through object attributes and application extensions (Knight and Aha, 2000). This new architecture allows Postdoc-NG to be able to support the addition of user application tools, or plug-ins, such that users are able to define their own workflow scripts. Adequate access control and authentication is also being re-designed in order to support a wide variety of users. With these changes Postdoc-NG will be much more flexible and be able to support a KM environment.

The new Postdoc-NG incorporates workflow management functionality. The addition of workflow capabilities is important in the context of allowing version control in the collaborative development of documents, such as change requests, document reviews, and proposal writing (Becerra-Fernandez et. al., 2000). The workflow module is able to track changes in the states of documents, proposals drafted, submitted, reviewed, voted on, and tallied in a virtual collaborative environment. Dispersed groups will benefit from a true dynamic, collaborative environment.

Workflow functionality provides the Postdoc-NG user with the following features and capabilities:

1. Graphical Workflow Design: The graphical user interface (GUI) that provides the means for creating workflow processes by defining the flow and the tasks that need to be performed.
2. Rules: The logic that needs to be integrated into the workflow definition.
3. Exception Handling: The system must be able to handle exceptions and errors.
4. Monitoring: The ability to monitor the status of workflow instances.
5. Auditing: Involves being able to log and report time and cost of the workflow process.
6. Notification: The workflow application must be able to inform users of new

[3] http://www.lotusnotes.com

tasks, warn them of late tasks, and reroute tasks to other users.

7. Document attachments: The ability to effectively attach documents to the workflow.

The most important objective for the workflow functionality is that it seamlessly integrates with the PostDoc-NG system. While the features that are to be incorporated into this workflow management system are not vastly different or superior to workflow tools that are currently on the market, the real benefit comes from how the WfMS augments the current capabilities of Postdoc. Other tools, such as Staffware2000, FileNet's Visual Workflo, and IBM's MQ series, offer similar features such as visual workflow design, user defined rules and application logic, web functionality, and role-oriented workflow task assignments. The PostDoc-NG workflow management system, however, is specifically designed to operate within the PostDoc-NG system architecture. This allows the workflow functionality to access and manage all object-oriented features of this system including access control, document handling, document metadata management, and user application components.

## Visual Design Interface

Our objective for the visual design interface is to make it possible for a user to graphically design a workflow on-demand. We are planning to customize an already developed graphical design applet based on the application of Petri Nets for the purpose of creating an adequate graphical visualization tool. A Petri Net visualization tool contains the three fundamental elements needed to graphically design a workflow, the place, the transition, and the arc. Graphic visualization tools can be used to design and model Petri nets using these basic elements. Through an application or applet, a user can graphically add places and transitions, and draw arcs to interconnect them. The essential components of a workflow, particularly the ones used for this tool, can be mapped out to these basic elements. A workflow state can be represented by a place; a condition or task completion event can be represented by a transition; and arcs can be used to interconnect places and transitions and direct the flow of the workflow process. By highlighting a place representing a state, a person can view pertinent information about that state such as descriptive headers, associated documents, participant assignments, and timing information. The graphical design tool can be customized to manipulate the internal state of a workflow thus effectively designing it via easily understandable visual components.

## Workflow Tool Design Concepts

The workflow tool makes use of the object-oriented design features of PostDoc-NG to accomplish its goals. An important design issue for this project was to develop a tool that would be useful across all possible circumstances. A user "sandbox" programming environment was envisioned as a way to resolve this. This environment was developed as a set of generic components that could be used for the design of any workflow process. The workflow components are made up of GUI interfaces that are used to specify workflow attributes and processes. These components allow the specification of workflow participants, relevant documents, timing information, and step metadata. The workflow makes use of a critical design feature of the PostDoc-NG system that is capable of generalizing across all possible workflow circumstances and creating a true "sandbox" programming environment, thus providing the ability to specify and execute programming functions on objects within the OO database. With this feature, a user or workflow designer can specify what event happens at the completion of a step or task, what action to take in the case of an unexpected occurrence, or what path to programmatically select from within the workflow. The tool provides an environment by which the generic attributes of a workflow process can be designed. The specifics are left up to the user.

An important component of the workflow tool is the Event Manager. Basically, it provides a way to execute an object method at a specified time. This component is important because it adds a crucial timing element to the workflow tool. It is used primarily to set deadlines for workflow tasks and to periodically check for updated documents. The workflow designer uses the interfaces described above to set the timing and function information for an event.

## Application Logic

Application logic in this context refers to how the workflow tool makes decisions on what action to take according to a specified circumstance. Because the tool is designed to be usable in all possible situations, the description of this logic will be confined to the underlying system events that are used to make decisions rather than actual

workflow circumstances. However, an example of the application of these underlying system principles to the processing of real world events will be described.

The workflow itself is modeled after a finite state machine (FSM). Like an FSM, it consists of states and transitions. In the design of this tool, each workflow task is considered a state. Transitions are used to organize the flow of the tool from one state to another. Each state represents a task that, when completed, triggers a workflow event and makes a transition to the next appropriate state. A workflow event is a function scripted by the workflow designer that executes a specific set of programming commands.

The application logic consists of three areas. These are the acknowledgement that a task has been completed, the selection of a state from a range of possible states in the case of branching workflow processes, and the implementation of an exception handling mechanism for unexpected occurrences in the workflow. All require the event manager and the processing of some internal workflow condition. A task is considered complete when a user-specified condition is satisfied. The workflow event for that task is then triggered and the transition to the next step is made. A branching workflow contains at least one state that has transitions to more than one other state. In this case, a workflow-defined variable determines the selected transition. Exception handling in this tool allows the user to specify what action to take in case of an unexpected occurrence. An unexpected occurrence can be a workflow exception such as a missed deadline or a system exception such as a server error. These exceptions are handled through the sending of an email notification to someone who has been appointed as a workflow overseer. Further development in this area can include more robust exception handling designed for specific types of exceptions.

A condition is used within a workflow to specify what is required for a task to be completed. The user chooses one or more conditions for each workflow state at design time from a finite set of pre-defined conditions. These conditions are the passing of a deadline, the update of a document or folder, and user notification that a task has been completed. In the case of the passing of a deadline, the event manager is used to set an event that triggers the transition to the next state. If there are several possible states to choose from then an internal variable is used to select from one of the possible states. In the case of the update of a document or folder, the modification time of these objects are checked periodically for any changes, if so then the condition is satisfied. As for the selection of the user-notification condition, the task participant is asked to fill out a web-page form that specifies that a particular task completed. The event manager is again used to periodically determine if this form has been filled out. When it is, a transition is triggered. Through the function interface, the workflow designer is allowed to script a function that occurs when the condition has been satisfied. What we usually intend to happen is that participants that are assigned to the next task are notified by email with related documents or links and a description of what they are expected to do.

An example of a process where this workflow tool would be useful is a proposal review process. A preliminary step in the design of this workflow is to identify the participants and documents to be involved, what tasks are to be completed, and what sequence these steps should be in. In this example, the participants would be the person(s) submitting the proposal and the review panel. Additional participants can include a workflow manager or overseer that is notified if there is any unexpected occurrence. The documents can be identified as a preliminary proposal document and a panel review document. The tasks are the initial submission of the proposal document, the review of this document by the proposal committee, and the results of the proposal committee review. This process can be translated for implementation within the workflow tool as follows:

First a general folder can be added to the system environment that will be used to handle proposal documents. The condition for the first step can be considered satisfied when a document has been uploaded into this folder.

The condition for the proposal review can be specified as a timed deadline condition. The committee has a set amount of time to complete the proposal review document.

When this deadline has passed this document can be sent via email to the person(s) submitting the proposal.

Additionally, an exception can be specified in the case of the proposal document not being completed or modified by enough reviewers by the task deadline. As a result of the use of this tool, this process is automated and completed much more efficiently.

## Future Plans

Postdoc-NG offers many possibilities to improve collaborative environments, for example, the

proposal review process. Currently, Postdoc notifies users of any changes in the document. In Postdoc-NG, notifications will be set for specific changes that modify particular states of the document (Becerra-Fernandez et. al., 2000).

Another possible consideration is the provision of an active lessons-learned module. Users want to be able to benefit from knowledge gained in previous projects that are contained within archived documents. For example, e-mails can be a rich source of lessons learned. Users can extract solutions and best practices from past experiences to avoid repeating costly mistakes. Currently an application of the use of case based reasoning technology is being explored to accomplish this purpose (Knight and Aha, 2000).

The new augmented features of Postdoc-NG are expected to prove invaluable to other key processes in NASA, for example in the Payload Processing Customer Support Process. The augmented Postdoc-NG environment with workflow, document management, and groupware capabilities would be useful to develop a prototype for collaborative computing ideal for the NASA-KSC Payload Processing Directorate. This application will improve the Payload Processing Customer Support process--by combining a central repository of information and a workflow system through a common web-interface. This application will improve and streamline business processes and the sharing of organizational knowledge (Becerra-Fernandez et. al., 1998b). With the integration of workflow, document management, and groupware, Postdoc-NG will provide a mechanism for virtual team collaborations inside as well as outside NASA.

**Acknowledgements** The authors wish to acknowledge NASA-ARC and NASA-Headquarters, for financial support to this project, grant number NAG2-1384, our technical monitor Helen Stewart, Martha Del Alto and Chris Knight, collaborators in this project. The authors also acknowledge the contributions of the students who work in the FIU Knowledge Management Lab, particularly Winston Spratt, and Lenny Ardiles.

**References**

Becerra-Fernandez, I. (1998a). Center for Innovation and Knowledge Management. Association for Computer Machinery SIGGROUP Bulletin, special Issue on Knowledge Management "Knowledge Management at Work", 19 (1), 46-51.

Becerra-Fernandez, I. (1998b) NASA Project Corporate Memory–Final Report, grant No. NAG10-0232.

Becerra-Fernandez I., Stewart H., Del Alto M., Knight C (2000). Developing an Advanced Environment for Collaborative Computing, Proceedings of the Twelfth Annual International Florida Artificial Intelligence Research Symposium (FLAIRS), Orlando, Florida.

Knight, C., Aha, D. (2000). A Common Knowledge Framework and Lessons Learned. Proceedings of the AAAI Workshop on Intelligent Lessons Learned Systems, Austin, Texas.