

RC-20611 (91299) 11/1/96
Computer Science/Mathematics 14 pages

9 6 A 0 0 1 3 7 5

Research Report

On the Integration of Project, Workflow, and Document Management

Melissa J. Buco, Jarir K. Chaar
IBM Research Division
T.J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

LIMITED DISTRIBUTION NOTICE

This report has been submitted for publication outside of IBM and will probably be copyrighted if accepted for publication. It has been issued as a Research Report for early dissemination of its contents. In view of the transfer of copyright to the outside publisher, its distribution outside of IBM prior to publication should be limited to peer communications and specific requests. After outside publication, requests should be filled only by reprints or legally obtained copies of the article (e.g., payment of royalties).

IBM Research Division
Almaden • T.J. Watson • Tokyo • Zurich • Austin



On The Integration of Project, Workflow, and Document Management

Melissa J. Bucu and Jarir K. Chaar
IBM T. J. Watson Research Center
P.O.Box 704
Yorktown Heights, NY 10598
{mjowen, jarir}@watson.ibm.com

I. Introduction

Modern day business practices demand complex procedures and processes that are information intensive. This environment necessitates management by information which consists of collecting and using all data required for effective decision making. Further, today's business environment is becoming more global and competitive. The relentless demand for providing "faster, cheaper, better" products and services is driving businesses at all levels to be more productive, more cost conscious, and constantly seeking improved business practices. The emerging marketplace is beginning to show certain interesting characteristics:

1. The labor resource pool is global. Contract development companies that specialize in various business sectors are available in different parts of the world.
2. Complex projects often involve unique, long running processes, that require participants with specialized talents and skills for the duration of the project.
3. A truly global communication infrastructure for electronic mail, file transfer, remote login, and the Web is emerging as more and more countries join the Internet.
4. Virtual teams composed of members operating in their individual and familiar environments while geographically dispersed and only sporadically connected are replacing more traditional teams.

The above factors, coupled with global economic pressures, make effective project management essential for business in general and large enterprises in particular. Of growing importance is the

role of the project manager who insures that customer requirements are met on schedule and within budget. In addition to creating and maintaining the project plan, the project manager must communicate the plan to the team, drive its execution, track progress, manage deliverables, and facilitate team collaboration. Tools that integrate project, workflow, and document management are crucial to the success of both project manager and team.

Project management tools help define the tasks, their durations, and their assigned resources, establish precedence between tasks, and specify "must" dates. Workflow management technologies are used to ensure the reliable and repeatable execution of business processes and to improve the efficiency of the enterprise by supporting the definition, planning, execution, and analysis of such processes. Document management is concerned with the collection, sorting, routing, display, sharing, archiving, and protection aspects of information handling.

The integration of project, workflow, and document management offers numerous advantages to project managers and teams. It offers project managers the ability to track the execution of a plan, quickly identify any deviations from schedules and/or procedures, and adapt the plan accordingly. Team members are not overwhelmed with the task of collecting support data that project managers require. Such data is automatically collected during the course of execution of the plan. Further, a major benefit of this integration is to eliminate the duplication of effort that goes into managing individual systems.

This paper outlines the design of an integrated project, workflow, and document management system. The implementation of this system in Lotus Notes 4.1 is discussed. This implementation is currently being deployed to various divisions in IBM for managing product (hardware and software) development. It is also receiving wide interest from industry solution units.

II. Business Processes and Project Plans

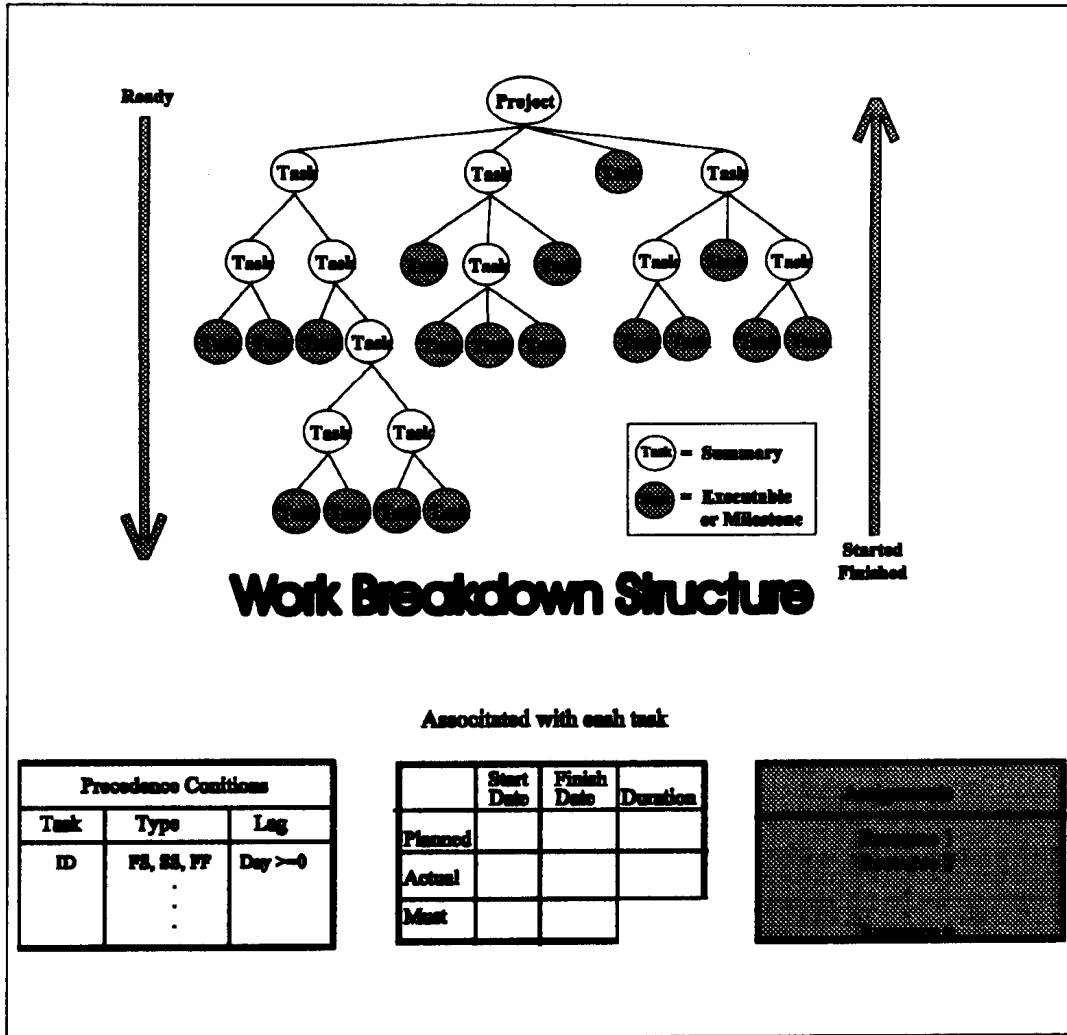
Over the last few years, corporate re-engineering has consistently identified the deployment of a common business process as key to effective process management and improvement. Deploying such processes corporate-wide is a complex endeavor that can benefit from process automation.

Current workflow management systems help automate highly repeatable, short running, well understood departmental processes. As such, they are not adequate for managing corporate-wide business processes that demand flexibility and adaptability.

Corporate-wide business processes specify common procedures as well as standard data sets that form the basis of decision making in the corporation. However, the very size and diversity of a large enterprise requires customizing both procedure and data at almost any level of the hierarchy. Achieving the proper balance between standards and local customization of procedures and data is key to successfully deploying such processes. Workflow management tools that strive to automate corporate-wide processes should allow for such customization.

While processes provide guidance and direction in building a project plan, it is the project plan that is executed. Processes provide an overall map for the execution of a project. This map captures the many possible paths a project might follow. A plan is specified by following one possible path of this map. Once a path of the map has been fully defined, a schedule can be derived. At any point in time, a plan is fixed. However, plans can and invariably do change. They often deviate in radical ways from the process. While such a change is supported in project management, it creates a conflict with the very principle of workflow management which enforces process execution. A natural mechanism for supporting the above consists of integrating project and workflow management using the project plan to drive the workflow.

A workflow management model defines the tasks and their sequences, the resources or roles, and the deliverables associated with each task. Task sequences include loops and conditional branches. Further, resource assignment can be done during execution. Some elements of the workflow model can not be specified in a project management model. In particular, dynamic elements such as loops, conditional branches, and assignment of resources during execution can not be supported because of the static nature of project plans. Moreover, project management tools do not support associating deliverables with tasks.



The project management model represents a plan as a tree structure known as a work breakdown structure. The tree represents a decomposition of the project into successively smaller more manageable tasks. The width and depth of the tree are determined by the level of granularity at which the project manager chooses to manage the plan.

The root of the tree is the project (task 0). The leaves of the tree represent executable or milestone tasks. Each intermediate node represents a summary of its children tasks. This work breakdown structure defines the structure of the project without regard to precedence conditions.

Associated with each task are optional “must” start and finish dates and flags (on or before, on, on or after) and a list of precedence conditions on other tasks. Together with the work

breakdown structure, these determine the order in which tasks can be executed. Project management offers three types of links:

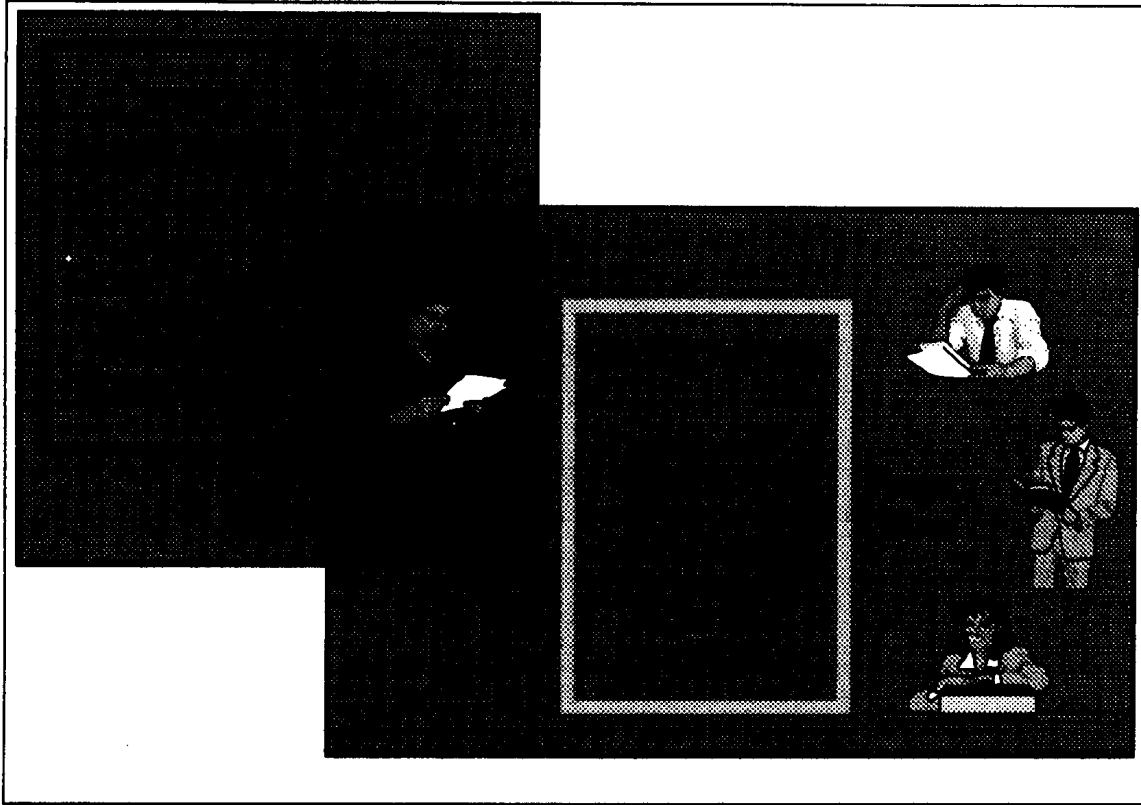
- *FS (finish-to-start)*: task cannot start until the other task finishes.
- *SS (start-to-start)*: task cannot start until the other task starts.
- *FF (finish-to-finish)*: task cannot finish until the other task finishes.

Each condition may also include a lead/lag time. There can be any number of precedence conditions associated with a task however these conditions cannot violate the work breakdown structure or create a loop. As a matter of style, some project management tools allow precedence conditions to be specified only for the executable and milestone tasks. The same results can be achieved by using additional links and/or milestone tasks.

Associated with each executable or milestone task is a list of one or more resources assigned to the task. For executable tasks, this is the list of resources that are assigned to perform the task. For milestone tasks (tasks with a duration of 0) this is the list of resources to be notified when a milestone is accomplished.

Each task has associated with it a state. Possible states in order of their progression are: *Not Ready*, *Ready*, *Started*, and *Finished*. The state change to *Ready* is one which flows down the tree from the root due to the implicit link between a task and its parent (i.e. a task does not become ready until its parent is ready). The change in state to *Ready* is also based on the FS and SS links and on the must start date. The state changes to *Started* and *Finished* are initiated by the task assignees and propagate up the tree from the leaves. A task becomes *Started* when at least one child task or assignment is *Started*. A task becomes *Finished* when all children tasks and assignments are *Finished* and all FF links are satisfied.

The integrated project, workflow, and document management model requires the functions provided by standard project management and workflow management models. In addition, features such as communication, collaboration, and document handling are key. Support for disconnected and mobile users is also desirable. The architecture of a system that supports the integrated model is discussed next.



III. Integrated Architecture

The integrated architecture is based on enhancing the architecture of project management tools. Project management tools tend to be single-user tools and do not address the needs for communication, collaboration, and document handling. Such needs are the focus of groupware technologies. The term *groupware* is much used, little understood, and frequently the source of confusion and skepticism. This paper views groupware technologies as work group enablers through communication, collaboration, and coordination. Where as electronic messaging enables communication and computer conferencing facilitates collaboration, workflow adds the structure of business processes to communication and collaboration so as to implement the policies of an enterprise.

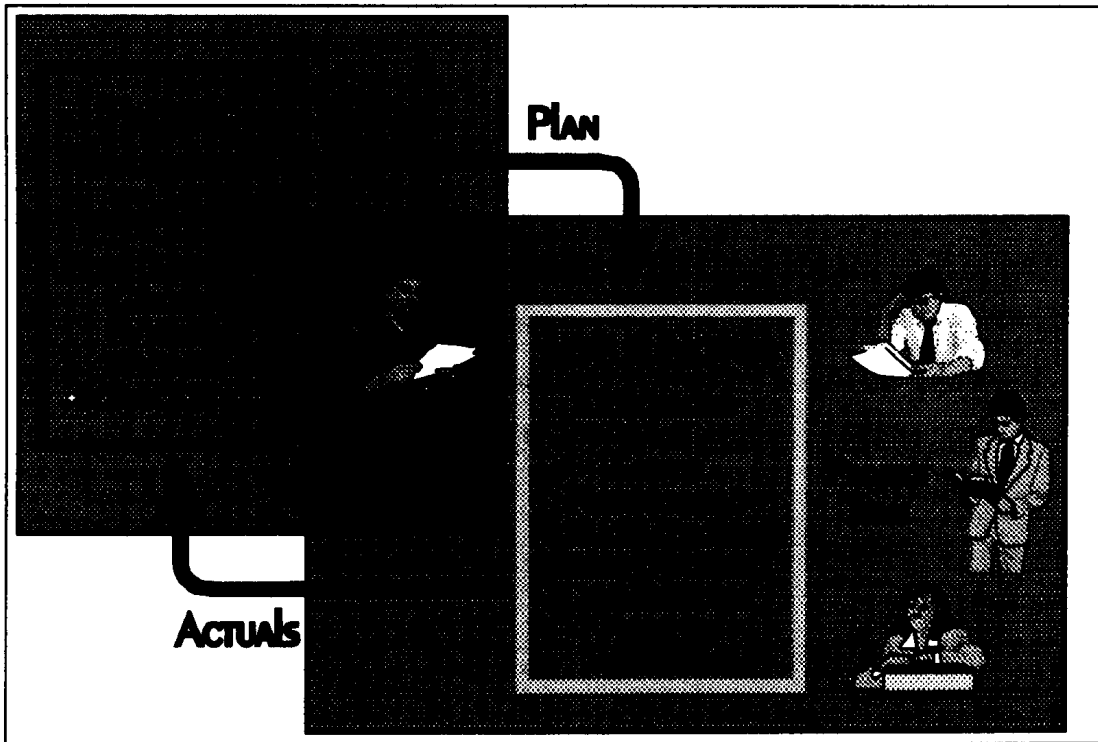
In the integrated architecture, the assignee to the project manager role is the only member of the team who requires access to the project management function. This function creates, updates, and stores the plan. All scheduling, standard project management analysis techniques such as critical

path method (CPM), and views such as GANTT and PERT charts are handled in the project management function. Standard functions are extended to specify an association between tasks and deliverables. Moreover, ownership and edit access are also specified in the project management function.

All team members require access to the groupware functions. Such functions:

- *provide access to the project plan*: Team members have read-only access to all views of a project plan as offered by the project management function.
- *compose and manage the project deliverables*: When a deliverable is specified in the project management function, it is automatically composed based on a predefined form in the groupware function. Ownership of and access to the deliverable is determined by the associations made in the project management function. Deliverables:
 - supports documentation requirements for ISO 9000 compliance,
 - maintain an audit trail,
 - provide for collaboration among the team,
 - provide for archiving,
 - provide for cross project roll up,
 - can be fixed or data driven.
- *associate tasks and deliverables*: Each task can have zero or more deliverables. The coupling between a task and deliverables is a loose one. This implies that the state of a task is not explicitly linked to the state of any deliverable. Hence, project plan and deliverables are managed independently.
- *maintain role assignments*: Roles are specified as the resources in the project management function and are resolved to team members in the groupware function.
- *advance workflow*: The sequence and constraints specified by the project management function are used by the groupware function to determine when tasks are ready to start or milestones are completed. The groupware function then notifies (via mail or to do list) the assignees of tasks which can be started and of milestones that have been completed. A pointer to the assignment is sent to the assignee. In turn, the assignment contains pointers to the deliverables, if any, associated with the task at hand. This enables the assignee to access both the assignment and the deliverables from mail or to do list.

- *populate the audit trail*: The assignee signals the start and finish of an assigned task.
- *monitor progress*: Optionally, notifications can be automatically sent for assignments which are nearly due or which have missed scheduled dates.



IV. Lotus Notes Implementation

The implementation of the system includes two components: a standard project management tool and Lotus Notes Release 4.1 as the groupware tool. The philosophy of the system is to use each tool for the functions for which it was designed and to avoid replicating function in either tool. The system relies on the project management tool as the engine to build and maintain the project plan and perform standard project management analysis.

The system extends the project management tool to associate deliverables with tasks. Because most project management tools provide similar function, any of a number of project management tools can be used. The system relies on Lotus Notes for communication among the team, driving

the workflow, collecting actual (start and finish dates), and the handling of the deliverable documents.

The system flow begins with the project manager creating a project plan in the project management tool. This plan can be built from scratch or from an existing project or template. The project manager then creates a Lotus Notes database from a template database for the system. The database template contains forms which are used to create the deliverables and to represent the plan. The template also contains views of the documents which are useful for the team (e.g. the project schedule, role assignments, my tasks, my deliverables, etc.). Further, the database template provides agents which manage the workflow, maintain role assignments, and manage the exchange of data with the project management tool.

Within the Lotus Notes database created for the project, the project manager creates a project document. On this project document, the project manager specifies the project management tool and project file to be used for the project. An action button on the project document allows the project manager to initiate the import of the plan data from the project management tool. The bulk of the import processing is done in a Lotus Notes agent which invokes the API of the project management tool only to export the plan data to flat files in Comma Separated Value (CSV) format.

The imported project plan is stored in a Lotus Notes database as a set of documents:

- *a project document* which contains all project level information and serves as a point of control for the project manager. This document contains action buttons which allow the project manager to exchange information with the project management tool and to control access to the plan in Lotus Notes.
- *a task document for each executable, summary, or milestone task* which contains information specific to task (e.g. task name, task ID, scheduled start/finish, actual start/finish, state, work breakdown structure code, links, etc.)
- *an assignment document for each assignment* which contains information specific to an assignment to a task (e.g. resource name, assignee, pointers to deliverables, task name, task

ID, scheduled start/finish, actual start/finish, state, etc.) as well as action buttons to “Start” and “Finish”. Much of the information from the task document is repeated in the assign document for convenience.

- *a role document for each resource* which contains the resource name, the userid and mail address of the team member assignee and an indicator of whether this individual has restricted or unrestricted access to the project deliverables.

The project, task, and assignment documents each have a state. The possible states of the documents in order of progression are *Not Ready*, *Ready*, *Started*, and *Finished*. An agent, running on the Lotus Notes server, drives the workflow by advancing the states from *Not Ready* to *Ready* based on the work breakdown structure, precedence conditions, and must start conditions imported from the project management tool. The agent notifies assignees when assignments are *Ready*. The notification for an assignment contains a pointers to a view of the project schedule and to the assignment document. The assignment document contains a description of the task to be performed and pointers to the deliverables, if any. On the assignment document are action buttons with which the assignee indicates that the assignment is *Started* and *Finished*. An agent running on the server detects when an assignee has started or finished an assignment and determines if the workflow can advance as a result of this change in state.

The actual start and finish dates are recorded in the project, task, and assignment documents for the project. These documents serve as an audit trail of the project execution. Once a project has started, the project document provides an action button which allows the project manager to initiate the export of the actuals (start and finish dates) to the project management tool. A flat file (CSV format) is created and the project management API is invoked to import the data into the project plan.

At any point, the project manager can import a modified project plan. The import of a plan is, in effect, a replacement of the existing plan. However, all information which originates in the Lotus Notes database (e.g. actual start and finish dates, role assignments, and certain project control information) is not replaced by an import. The import of an “active” (*Ready* or *Started*) plan can

result in an “active” assignment being changed or even deleted. In such cases, the assignees are notified of the change or deletion.

Additional agents run on the server to:

- detect assignments which become ready based on calendar date,
- notify assignees of assignments which are nearly due or have missed scheduled dates,
- handle changes to role assignments.

The deliverables of the plan are also stored in the database as documents. The agent which performs the import of a plan determines which deliverables referenced in the plan do not exist and composes them using forms which must be predefined in the database. Further, the agent updates edit authority to the deliverables insuring that only those roles which have assignments which reference the document have update authority to it and updates read authority for restricted documents.

User definable fields in the project management tool are used to associate deliverables with specific assignments. Five parameters (separated by commas) uniquely identify a deliverable. The first parameter is the name of the Lotus Notes form to be used for composing the document. The remaining four parameters are optional and allow for multiple instances of a single form. An asterisk “*” preceding the five parameters indicates that the assignee is the owner of the document. An exclamation point “!” preceding the form name indicates that the document has restricted reader access.

Each deliverable document has a state. Possible states (e.g. Draft, Revision, Final) are defined by the project manager. Each deliverable document has a team member as an owner, responsible for the document. An assignment can reference multiple deliverable documents and a deliverable document can be referenced by multiple assignments. By design, there is no explicit link between the state of a document and that of a task. The ability to modify a deliverable is not restricted to the duration of a task.

The deliverables can be considered applets. Two types of deliverable documents, fixed format and data driven, are supported. The fixed format documents can contain any type of information (numeric, text, etc.) and require a unique form for each. The data driven forms allow up to twenty essay type questions (rich text) and provide for inheritance of answers. The advantages of the data driven forms are that only one form is needed thereby reducing maintenance and new deliverables can be created by adding data to a question database thereby reducing the need for expertise in form creation.

All the deliverable forms (fixed and data driven) are designed with sections to track status, maintain an audit trail, and initiate and track collaboration. Actions on each document provide the document owner the ability to update the status of a document and maintain a status history, request input from other team member on the content of the document, and to record significant changes effecting the document .

The project manager has the added capability of archiving a subset or the full set of deliverable documents on demand. Archived documents are stored in a separate database. They offer snapshots of the various states of a deliverable document and can be used by the project manager in tracking and evaluating progress during the course of a project. Further , archiving may be used to link changes to the original project plan when caused by lack of progress on deliverable documents. The project manager creates such a link by archiving the documents in question when a plan changes.

V. Conclusion

Over the last few years, corporate re-engineering has consistently identified the deployment of a common business process as key to effective process management and improvement. Deploying such processes corporate-wide is a complex endeavor that can benefit from process automation. Current workflow management systems help automate highly repeatable, short running, well understood departmental processes. As such, they are not adequate for managing corporate-wide business processes that demand flexibility and adaptability. This paper describes an integrated

project, workflow, and document management environment that allows the specification, execution, and tracking of such processes.

Project, workflow, and document management integration offers numerous advantages to project managers and teams. It offers project managers the ability to track the execution of a plan, quickly identify any deviations from schedules and/or procedures, and adapt the plan accordingly. Team members are not overwhelmed with the task of collecting support data that project managers require. Such data is automatically collected during the course of execution of the plan. Further, a major benefit of this integration is to eliminate the duplication of effort that goes into managing individual systems.

The current implementation is a key component of IBM Re-engineering Project Manager's Workbench. It will be used by product development teams within all of the divisions to manage by information their product development life cycles from concept through end-of-life. Further, other industry solution units and IBM customers are interested in the integrated aspects offered by this implementation. They perceive it as an important step toward fully integrating the automated aspects of their business processes. Hence, many enhancements are already planned for the very near future and will be reported in due time.

Future plans include support for inter-project dependencies and subprojects. Inter-project dependencies capture the precedence conditions to the tasks of other projects. Subprojects deal with complex tasks that are project aggregates themselves allowing for very large projects to be more efficiently managed. Furthermore, a data warehouse for aggregating the data collected by individual projects will be implemented. Such data is key to improving business processes, benchmarking various projects, and managing by information.

References

L. Fischer (editor). *New Tools for New Times: The Workflow Paradigm - The Impact of Information Technology on Business Process Re-engineering*. Second Edition. Future Strategies 1995.

S. Khoshafian and M. Buckiewicz. *Introduction to Groupware, Workflow, and Workgroup Computing*. Wiley 1995.

Lotus/IBM. *Developing Applications with Lotus Notes Release 4*. First Edition. January 1996.