



UNITED STATES AIR FORCE RESEARCH LABORATORY

Readiness Assessment and Planning Tool Research (RAPTR)

Allen Batteau

Wayne State University
656 W. Kirby
Detroit MI 48202

Cassie B. Barlow
Air Force Research Laboratory

September 1998

Final Report for the Period May 1996 to September 1998

20000328 038

Approved for public release; distribution is unlimited.

Human Effectiveness Directorate
Deployment and Sustainment Division
Sustainment Logistics Branch
2698 G Street
Wright-Patterson AFB OH 45433-7604

NOTICES

When US Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, as in any manner, licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.

Please do not request copies of this report from the Air Force Research Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield VA 22161

Federal Government agencies and their contractors registered with Defense Technical Information Center should direct requests for copies of this report to:

Defense Technical Information Center
8725 John J. Kingman Rd., STE 0944
Ft Belvoir VA 22060-6218

DISCLAIMER

This Technical Report is published as received and has not been edited by the Air Force Research Laboratory, Human Effectiveness Directorate.

TECHNICAL REVIEW AND APPROVAL

AFRL-HE-WP-TR-1999-0197

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

For C. J. Kidney Lt Col
JAY KIDNEY, Col, USAF, Chief
Deployment and Sustainment Division
Air Force Research Laboratory

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September 1998		3. REPORT TYPE AND DATES COVERED Final - May 1996 - September 1998
4. TITLE AND SUBTITLE Readiness Assessment and Planning Tool Research (RAPTR)			5. FUNDING NUMBERS C - F41624-96-C-5005 PE - 63106F PR - 2940 TA - 02 WU - 04	
6. AUTHOR(S) Allen Batteau; Cassie Barlow				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Wayne State University 656 W. Kirby Detroit MI 48202			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratory, Human Effectiveness Directorate Deployment and Sustainment Division Air Force Materiel Command Sustainment Logistics Branch Wright-Patterson AFB OH 45433-7604			10. SPONSORING/MONITORING AGENCY REPORT NUMBER AFRL-HE-WP-TR-1999-0197	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION AVAILABILITY STATEMENT Approved for public release; distribution unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) Readiness Assessment and Planning Tool Research (RAPTR) was a 6.3 funded advanced research and development contract managed by the Air Force Research Laboratory Sustainment Logistics Branch. It is a server based software application that assists change teams to successfully plan, implement, and manage a process change. RAPTR assesses an organization's culture and technology and then offers suggestions that assist in mitigating potential impediments to change. RAPTR also contains a knowledge base of lessons learned that assist organizations going through process change. RAPTR contains COTS tools that are integrated into a seamless toolkit for change teams. The RAPTR field research took place at WR-ALC in the Reengineering Directorate.				
14. SUBJECT TERMS collaboration, distributed, process modeling, reengineering, process change, change management, culture, culture change			15. NUMBER OF PAGES 116	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

THIS PAGE INTENTIONALLY LEFT BLANK

PREFACE

This report documents the results of a comprehensive study to create a server based software application that assists change teams to successfully plan, implement and manage a process change as part of a logistics research and development program titled Readiness Assessment and Planning Research (RAPTR), (Contract Number F41624-96-C-5005) managed by the Air Force Research Laboratory, Logistics Sustainment Branch (AFRL/HESS), at Wright-Patterson AFB, OH. The primary goal of the project was to assess an organization's culture and technology and then offer suggestions that assist in mitigating potential impediments to change as well as offer planning guidance throughout the change project. The results clearly provide support to organizations that need assistance in managing change within their organization.

Acknowledgement

We gratefully acknowledge the assistance in preparing this report provided by Bill Hetzner, Ron Kohler, and Stan Przybylinski, ERIM; Les Sanders, Wizdom Systems, Inc., and Gina Alexia and Carolyn Psenka, Wayne State University. Final responsibility for its presentation and conclusion rests with the Principal Investigator.

Contents

Acknowledgements	iv
List of exhibits	vi
1. Executive Summary	1
2. Objectives of this Report	2
3. Objectives of the <i>RAPTR</i> project	4
4. State of the Art prior to <i>RAPTR</i>	9
5. <i>RAPTR</i> Development	22
6. Description of the Tool	48
7. <i>RAPTR</i> Field Trials and Field Evaluation.....	66
8. Lessons Learned	87
9. Using <i>RAPTR</i> : Where, How, Why	96
10. Conclusion: Research challenges for a global mission	104

List of Exhibits

Exhibit	page
4.1 Growth in Internet Web Sites August 1995 – September 1998	10
4.2 Planning is one of many possible knowledge engineering applications.....	11
4.3 Search technologies reviewed for RAPTR	15
4.4 Groupware and document management systems reviewed for RAPTR	16
4.5 Workflow framework.....	19
4.6 RAPTR requirements beyond scope of most workflow systems	20
5.1 Initial RAPTR concept	24
5.2 Component template	26
5.3 Operational scenarios.....	27
5.4 Change Management Reference Model	33
5.5 RAPTR navigation paths.....	34
5.6 Microsoft Project integration	35
5.7 RAPTR architecture	36
5.8 RAPTR functionality (software view)	38
5.9 Sociotechnical performance model (early version)	39
5.10 Example of a variable description.....	41
6.1 The RAPTR main screen	48
6.2 The RAPTR Logon dialog box	49
6.3 The RAPTR main screen	50
6.4 Understand process change	51
6.5 Outside resources	51
6.6 Opening a project	53
6.7 The project management screen.....	54
6.8 The Team Self Assessment start screen	54
6.9 The Team Self Assessment question screen	55
6.10 The TSA analysis screen.....	55
6.11 Analysis of the High-Level Assessment.....	56
6.12 Generating a Plan	56
6.13 Scope plan screen	57
6.14 The task management screen	57
6.15 Project repository screen.....	58
6.16 Microsoft Project for Workgroups	59
6.17 Task repository area	60
6.18 Repository upload screen.....	61
6.19 The Browse Area archive screen	62
6.20 Lessons learned screen.....	63
6.21 Lessons learned entry screen	63
6.22 RAPTR search screen	64
7.1 MSD Process map (AS-IS)	73
7.2 Traditional reengineering context.....	83

Exhibit	page
7.3	IMPAC as exemplifying the new reengineering context.....83
7.4	Traditional chain of command management.....84
7.5	New context of change management 85
8.1	RAPTR Assessments.....88
9.1	Contexts in which to use RAPTR – scope and complexity.....97
9.2	Levels of sponsorship 98
9.3	Use of RAPTR functions.....100

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 1: Executive Summary

In 1996 the Logistics Research Division of the Armstrong Laboratory (later incorporated into the Air Force Research Laboratory) commissioned a team led by Wayne State University to develop a change management tool that would address the organizational and cultural issues in change management, especially as they related to streamlining of wholesale logistics support processes. The resulting tool, **RAPTR** (**R**eadiness **A**ssessment and **P**lanning **T**ool **R**esearch) was released in beta form in April, 1998, field-tested in August and September, 1998, and delivered to the Air Force in October, 1998.

RAPTR addresses a challenge that is confronting the Air Force, as well as the other armed services: how to sustain an evolving global mission in an era of constrained resources. These resource constraints are acutely felt in the logistics arena, where more complex systems, accelerated operational tempo, and new business methods (just-in-time, repair-on-demand, lean logistics, agile combat support, and others) require a high level of adaptability from the workforce. Numerous change initiatives have attempted to implement these and related business methods with mixed success, falling short in part due to what was viewed as "cultural problems" within the groups affected by the change.

RAPTR provides the change manager with a set of tools to address these cultural and related problems, through assessment, diagnosis, and the recommendation of both project plans and remedial steps for the specific problems. In doing this it incorporates years of experience with change management projects, fieldwork examining USAF culture, and a distillation of the literature on change management techniques.

This report provides a comprehensive statement of the objectives, methods, results, and lessons learned of the **RAPTR** development. It focuses on a description of the development effort (chapter 5), a description of the tool (chapter 6), and a description of the field trial and evaluation (chapter 7).

Also supplied are a description of the state of the art prior to the **RAPTR** development (chapter 4), lessons learned from the **RAPTR** experience (chapter 9), and guidance on the use of **RAPTR** (chapter 9).

Preparation of this report was a joint effort among the members of the development team: Wayne State University, Wizdom Systems, Inc., and the Center for Electronic Commerce (formerly part of the Industrial Technology Institute (ITI), now a unit of the Environmental Research Institute in Michigan (ERIM)). In the concluding chapter the Principal Investigator of the **RAPTR** project, Dr. Allen W. Batteau, steps back to take a broad view of the research issues involved as the Air Force continues to face the challenge of a global mission with constrained logistics resources.

Chapter 2: Objectives of this Report

The **RAPTR** project had the ambitious goals of packaging expert knowledge about change management into an easily-accessible, PC-based tool, and supporting change management projects with that knowledge. It was developed in response to an understanding that frequent and disruptive change was becoming a way of life for the Air Force, yet as an organization the Air Force needs to improve its tools and its knowledge for managing change. Although there are many knowledgeable and insightful individuals within the Air Force, to date their understanding of the methods and mechanisms of change management has not effectively diffused through the entire Air Force community.

In presenting this final report we propose to describe:

- The objectives of the **RAPTR** project
- How these objectives were addressed
- Our successes and failures in achieving these objectives

And most importantly,

- How AFRL can use **RAPTR** and build on the knowledge it contains

As will be seen from our state-of-the-art presentation (chapter 4), the **RAPTR** research and the resulting tool have made some significant advances in the Air Force's current capabilities for supporting organizational change.

Although conventional understandings of technological development and deployment often assume a linear process (from basic science to applied science to technology development and innovation), in fact the process is linear only in hindsight: in hindsight the myopic concepts, false starts, failed experiments, and strategic blunders are quickly forgotten, and all that is remembered is an unbroken chain of successes.

The **RAPTR** project was fortunate in that it maintained a careful journal of all activities, and as such there is a record of the false starts, failed experiments, etc. We include these here since there is potentially more to be learned from a project's failures than from its successes.

In contrast to the linear assumptions of conventional understandings, the process of technological innovation is a chaotic and complex process, in which the key question is not "Will this feature perform according to specification" (a laboratory question, using explicit knowledge), but rather "Will this complex system fit into and improve the

adaptive potential of an organization” (a field question, using experiential knowledge).¹ Questions of the second sort rely on knowledge that is tacit and context-dependent, difficult to communicate yet critical for a successful innovation.

In this report we thus aim to present not only an unbroken string of successes, in which we take considerable pride, but also the myopic concepts, false starts, failed experiments, and strategic blunders that littered the way. These false starts and strategic blunders will be analyzed as object lessons for those in the Air Force community who are tasked with supporting change management, either through tool support, technical assistance, or leadership.

¹ Cf. Nonaka, Ikujiro, and Hirotaka Takeuchi, The Knowledge-Creating Company. New York. Oxford University Press. 1995. Or Brown, J.S., and P. Dugid, Organizational Learning and Communities-of-Practice. Organization Science 2, #1, 40-57.

Chapter 3: Objectives of the *RAPTR* Project

3.1 Background

Rapid and disruptive change is becoming a way of life in the US Air Force. Declining operational budgets have not been matched by a corresponding ramp-down of mission or readiness requirements. The Air Force is required to do as much, or more, with less.²

Consistent with numerous trends in government and industry (Corporate Information Management, Vice President Gore's National Performance Review, Acquisition Reform, Business Reengineering), the Air Force is meeting this challenge by finding new ways of doing business: New ways of providing and supporting personnel and materiel for the warfighting commands. Streamlined business methods – a reduction in ordering time for repair parts, for example – translate into larger numbers of mission capable aircraft. These initiatives – Integrated Weapon Systems Management, Paperless Acquisition, Lean Logistics, Supply Chain Management – require not only the introduction of new technology, but consistently a cultural change within AFMC: from a process-orientation to a customer-orientation, from fixed to flexible work schedules, from asset-hiding to asset-visibility, from just-in-case to just-in-time.

The *AS-IS* of reengineering and change management scenarios within AFMC is characterized by small teams adapting published methods to local circumstances, *ad hoc* use of tools, and in some locations heavy reliance on consultants to guide the change management process. These teams typically work under aggressive schedules with tight deadlines for deliverables; often they have had little previous experience with reengineering. In addition, the teams at disparate locations seldom share information in any meaningful way.

3.2 Response

The goal of *RAPTR* was to provide a multi-echelon, integrated support environment for those parts of the change management scenario that require experience-based insight into organizational characteristics and change management methods. A key phrase in that statement is "experience-based." The Air Force was spending many millions of dollars on various change efforts that often were unaware of each other, and their respective successes and failures. Many dollars were also being spent on contractors to bring both specific and general skills to bear on the issues surfaced during change efforts. If a tool could be created that would allow organic change agents to capture lessons from previous

² Presentations at the OPTEMPO/PERSTEMPO Conference, Air Force Safety Center, 5-6 August 1997.

projects it would be very useful in the sense of not having to reinvent the wheel (at tremendous cost).

At the same time, we wanted a tool that could serve as an electronic tutor for change agents. A user friendly tool to help them learn about all of the major factors that must be considered *before* attempting a change effort would be not only useful, but cost-effective in the sense that it would allow change teams to stop hiring outside contractors for general change management roles. By using **RAPTR** from the inception of a change project, it would also mitigate the costs of false starts in the processes of introducing changes. Through the Team Self-Assessment, the tool could also point out when a specific skill was needed, and the team could then access that skill for the particular task.

The project then, was to produce a front-end, multi-purpose computer-based tool for integrating cultural, strategic, technology, process, user-readiness issues, and previous project experience into change management scenarios. It would provide assessments of these issues within an Air Logistics Center (ALC) or other aircraft maintenance environments, drawing on a knowledge base of data from previous projects and other sources. We believed that an assessment approach provided the optimum balance between local flexibility (adapting to local situations) and a uniform approach across multiple USAF components. The knowledge generated would be accessible in a useful manner to business reengineering, Lean Logistics, and other change management teams. **RAPTR** would also provide a means to aid virtually co-located teams to maintain clear communications around issues of tasking, and for all to have access to the necessary components of the tool.

In addition, experience, and the literature, had demonstrated that cultural resistance to change is a major factor in the success of reengineering efforts. Yet no tool extant integrates cultural issues with other change management technologies. Such an integration would enable change management teams to anticipate sources of resistance to change, identify and leverage the changes agents within an organization, and enable change management teams to tailor their strategies to that which is feasible within the culture of the organization.

3.3 Users

Users of **RAPTR** were expected to primarily be members of change management teams at both the working and leadership levels. The tool would be designed to produce the high-level views expected by senior officers or civilian managers, and to provide integration between detailed and high-level views. As such it would provide a common environment for executives and business engineers to discuss process and design implementation projects.

3.4 Comprehending culture

A distinctive innovation of the **RAPTR** project was its attempt to create a tool with which project teams could understand and account for cultural issues in making and executing project plans. Cultural issues, such as “resistance to change” or “bureaucratic inertia” were often seen as impediments for accomplishing change objectives such as Lean Logistics or paperless environments.

The **RAPTR** team attempted to strike a balance between the desiderata of (a) a non-intrusive tool that would (b) provide useful information and insight in (c) a wide variety of contexts. Given the impossibility of meeting all three of these objectives, the strategy selected emphasized:

- Focus on ALC and aircraft repair facilities (modifies c)
- A drill-down approach, with a high-level assessment tailoring a more detailed assessment (modifies a)
- Maximize knowledge content and delivery (optimize for b)

Early fieldwork indicated that in popular parlance “culture” was used for all those aspects of an organization that were elusive to managerial rationality. From a managerial perspective the failure to adopt a new technology might appear to be irrational (and hence “cultural”), whereas from another point of view a worker might have any of a number of good reasons for resisting technology: it might “dumb down” his job and give him less opportunity to exercise his skills, it might require a steep learning curve (and thus require a period where his skills appeared inadequate), or the technology itself might actually make his job harder. What might appear “irrational”, up close turns out to be rational indeed.

For this reason the team settled on a definition of culture that placed less emphasis on individual traits and more on shared traits of all members within the organization, traits that were reinforced by organizational structure and history. As viewed here culture is a set of shared sentiments, originating from multiple sources, that guide and influence motivation without actually directing action. When the team modeled culture, it established eleven variables:

- Work group innovativeness
- Internal Status alignment
- Trust
- Commitment to organization
- Commitment to people
- Value given to learning
- Mentoring
- Status conferred by technology

Organizational values
Middle and line management commitment to change
Leadership commitment to change

These variables are defined, and their measures are established in the *RAPTR* variable catalogue.

From this list it is evident that with few exceptions these variables must be seen as attributes of groups, not individuals. Although a variable such as "Value given to learning" might appear to be an individual attribute, as measured in *RAPTR* it consists of resources available for training, organizational responses to efforts to acquire more training, and organizational attitudes toward those who acquire more training.

The definition of culture used here is closest to that of Edgar Schein³, who distinguishes among three levels of culture: Artifacts, espoused values, and basic underlying assumptions. Our departure from Schein takes two forms: First, we recognized that the cultures within an organization could have multiple provenances, and that it was both incongruities within the culture, as well as disconnects between culture and performance objectives that led to suboptimal performance. Second, given our measurement requirements, we attempted to create "distal" measures of the underlying values. For example, most organizations at least in some measure collect certificates as indicators of learning; sometimes they take the form of diplomas on public display, sometimes they are simply notations in personnel records. These are "artifacts." Most organizations are going to give at least lipservice to training and learning; rarely inside an organization does one hear learning derided as such. This constitutes "espoused values." There are, however, some organizations that will offer few training opportunities, make it difficult for employees to adjust their schedules, or give no special recognition to those who complete a training experience. In these organizations we conclude (rhetoric aside) that learning does not have great salience within the organization's hierarchy of values. Measures such as these uncover the "basic underlying assumptions" and establish variable values that, with a well-constructed model, can be used to identify cultural problems.

Our use of culture would be incomplete without reference to the sociotechnical performance model within which it was embedded. This model is described in greater detail in chapter 5 of this report. It modeled an ideal state of alignment among process, technology, culture, communication, personnel practices, organizational characteristics, environmental factors, and an organization's change objectives. Actual values of variables in each of these domains were then measured against the ideal state, and disconnects and mismatches were identified for either further inquiry or remedial action.

³ Schein, Edgar, *Organizational Culture and Leadership*, 2nd edition. San Francisco, Jossey-Bass, 1992.

3.5 System overview

It was determined that the specific uses of the **RAPTR** tool in supporting change management teams would include:

- Initially assessing the situation
- Training and orienting the reengineering team
- Scoping the project
- Managing issues of organizational culture and user readiness
- Learning from previous projects
- Capturing lessons learned from the ongoing project
- Deciding which tools and methods should be used
- Deciding which tasks and deliverables are appropriate given the objective and scope of a change management project
- Designing the *TO-BE* processes and systems
- Serving as an integrating and communication mechanism for the change team

RAPTR would accomplish this support with a unique integration of assessment tools, a knowledge base, communication tools, project management tools, and user-interpreted and prescribed presentations.

3.6 The **RAPTR** Team

The **RAPTR** tool embodies concepts that its developers have been working on for many years prior to **RAPTR**, and continue to develop. At Wayne State University a partnership between the Departments of Anthropology and Industrial Engineering has led to many advances in the understanding of sociotechnical systems and the role of teamwork in effective organizational performance. At the Industrial Technology Institute, the development of tools for organizational change has resulted in several change management tools and publications. At Wizdom Systems, Inc., breakthroughs in process modeling and management are embodied in commercial software. At Warner Robins ALC, the future of Air Force logistics is being created today. All of these continue to represent major resources for effective organizational change within today's Air Force.

Chapter 4: State-of-the-Art Prior to *RAPTR*

Our original concept for *RAPTR*, was that the planning and execution of change efforts would be directed by information about the context of the change. Data collected on culture, strategy, and organizational issues would be used by *RAPTR* to recommend explicit project team actions. Once a plan was created, the system would support the execution of the plan, providing guidance on the methodology, provide access to tools and methods, and support management of the information gathered and created by the project team. Products of and lessons learned from past change efforts would be made available to support the new projects.

In addition to the planning support, our original proposal included the following elements⁴:

- Provide access to a set of **team resources** to support on-going and new reengineering efforts. These resources should document the learning and concepts developed by WR-ALC that led to the creation of their custom reengineering process.
- Create a **designers' notebook** that contains all of the data gathered, analyses performed, and conclusions drawn by the reengineering teams.
- Provide **tool interfaces** to the existing (and eventually new) reengineering tools employed by the reengineering teams. This includes interfaces to desired elements of FRAME/WORK and COSAT (Cross-Organizational STEP Adoption Tool).
- A **Notebook Library** that allows users to draw information from multiple Designers Notebooks for cross-project learning and comparisons.
- A **Reengineering Process Workflow Manager** embedded with the *RAPTR* system that leads the user through the *RAPTR* process, prompting them for data and to execute activities of the process.

The solicitation required that our solution be based on the “current version of Windows”. In addition, we planned to “employ commercial, off-the-shelf (COTS) elements to the greatest extent possible”. As our concept evolved, and the World Wide Web (WWW) exploded around us, we chose to build our technical framework around the Web. Using the terminology that has evolved over the last several years, *RAPTR* became a project-focused extranet (a collaborative system for use across organizational boundaries that is built on Internet standards).

Any Web based solution required the selection of a Hypertext Transfer Protocol (HTTP) server. Exhibit 4.1 shows the growth of HTTP server usage. The requirement for a Windows-based server limited our choices. While the project team used a Netscape

⁴ CABOT Technical Proposal.

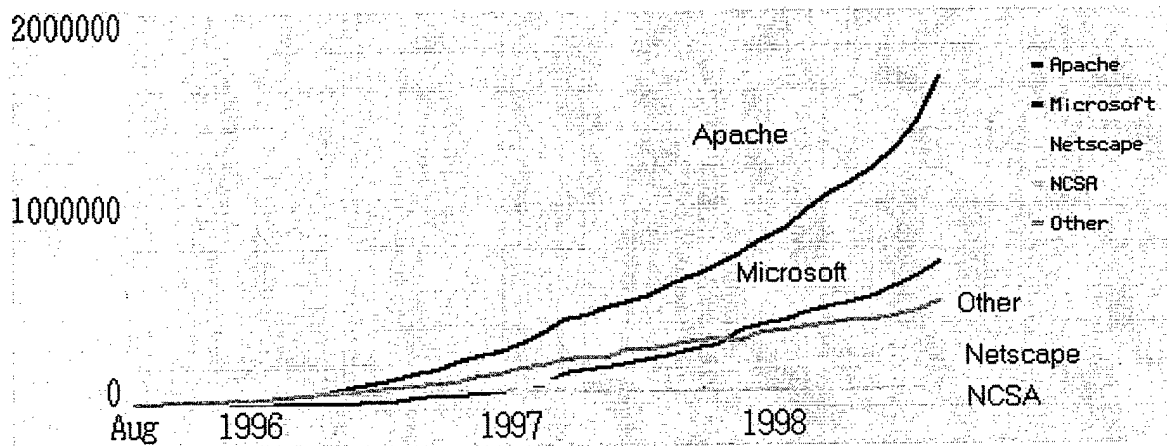
server for the team Web site, we eventually chose Microsoft Internet Information Server (IIS). The graph illustrates that while Netscape was a reasonable choice when we did our research (early 1997), Microsoft has developed a large following.

One area that we proposed was true tool integration: "Provide **tool interfaces** to the existing (and eventually new) reengineering tools employed by the reengineering teams". Our team quickly realized that this goal was beyond our means. A number of other programs, such as I-CASE, spent many times our budget trying to achieve this level of interoperability. Even though true integration was not feasible given our budget and schedule, we felt that we could simplify the experience of our users if we made a single interface, a Web browser, the doorway into **RAPTR**.

Given these original requirements, the **RAPTR** project attempted to mine the state of the practice and push the state of the art in the following areas:

- Knowledge-based planning support
- Knowledge management/document management
- Methodology support
- Workflow management

The following sections describe the state of the practice in these areas prior to the **RAPTR** project. (Of course, for many existing products or technologies, these categories overlap.)



Source: Netcraft September 1998 survey of 3,156,324 sites.
(<http://www.netcraft.com/survey/>)

Exhibit 4.1 -- Growth in Internet Web Sites August 1995 - September 1998

4.1 Knowledge-based Planning Support

Knowledge-based planning systems are one class of expert systems, “man-machine systems with specialized problem-solving expertise⁵.” lists some generic categories of knowledge engineering applications.

Category	Problem Addressed
Interpretation	Inferring situation description from sensor data
Prediction	Inferring likely consequence of given situations
Diagnosis	Inferring system malfunctions from observables
Design	Configuring objects under constraints
Planning	Designing actions
Monitoring	Comparing observation to plan vulnerabilities
Debugging	Prescribing remedies for malfunctions
Repair	Executing a plan to administer a prescribed remedy
Instruction	Diagnosing, debugging, and repairing student behavior
Control	Interpreting, predicting, repairing, and monitoring system behaviors

Source: Hayes-Roth et al, Building Expert Systems, p. 14.

Exhibit 4.2 – Planning is one of many possible knowledge engineering applications

Most planning research in artificial intelligence (AI) focused on developing plans for robots, manufacturing systems, or other hardware “effectors” to achieve a specified goal. In planning systems, problems are characterized by an initial state and a goal state description⁶. For **RAPTR**, the “initial state” is the combination of the characterization of the project goal, the make-up and skills of the project team, and the assessment of the target organization. The “goal state description” is the development of a change project plan that meets the project objectives, subject to the constraints dictated by the project team and the target organization.

⁵ Hayes-Roth, Frederick et al. (1983) Building Expert Systems. Reading, MA: Addison-Wesley Publishing Company, Inc., p. 4.

⁶ Tate, Austin et al. (1990) “A Review of AI Planning Techniques”, in James Allen et al, Readings in Planning, San Mateo: Morgan Kaufmann Publishers, Inc., p. 28.

Historically, planning problems were attacked in two ways:

- *Domain-dependent* approaches that use domain-specific heuristics to control the planner's operation, or
- *Domain-independent* approaches where planning knowledge representation and algorithms are developed to apply to a broad range of application domains⁷.

Most of the planning work in AI has been on domain-independent planning systems. Clearly, the goal for **RAPTR** was to provide a domain-dependent system, where that domain was the planning and management of change projects.

In some respects, planning support in **RAPTR** is more akin to diagnosis, e.g.,

- What tasks should I include in my plan based on my change goals and objectives?
- What tasks should I include in my plan based on characteristics of my team?
- What tasks should I include in my plan based on characteristics of the target organization(s)?

In the terminology of planning systems, these plans are *hierarchical* since they are based on a structured methodology that has internal dependencies, i.e., if detailed process modeling is recommended then some approach to building those models (a lower level in the hierarchy) must also be included.

Much of the project management research has focused on network-based project planning. As such, civil engineering and construction have been ripe application areas. The emphasis on project networks and the creation of plans from building blocks that can be directly tied to customer specifications makes this domain a natural for knowledge-based systems. Work goes back to the late 1980's and of late has taken advantage of case-based reasoning technology. Developers who observed skilled planners at work realized that they developed new plans by looking at networks from past projects⁸. In some respects, we used this general approach in **RAPTR** in that we based our activity sequencing and duration estimation on the experience of skilled BPR consultants.

Of course, **RAPTR**'s domain is organizational change. At the start of the **RAPTR** project, the most advanced knowledge-based system for organization design and analysis was the ACTION system, developed at the University of Southern California. ACTION can be used to provide a sociotechnical systems (STS) analysis of either existing or proposed organization or process designs. While ACTION was developed to focus on discrete parts manufacturing, its developers believe that the system has applications in other

⁷ Ibid., p. 26.

⁸ Lee, Kyoung Jun et al. (1998) "Case- and Constraint-Based Project Planning for Apartment Construction", *AI Magazine*, (19)1, p. 14.

domains⁹. In some sense, ACTION would be an ideal tool to use as part of **RAPTR** to analyze the as-is and to-be designs.

RAPTR was originally envisioned as an extension of FRAME/WORK, an earlier project conducted by Wizdom Systems Inc. with Air Force Research Laboratory (AFRL) funds. In that effort, assessment data was processed by a knowledge-based system to provide generalized recommendations to support the adoption of information technology. **RAPTR** was to extend this approach to provide specific recommendations on project activities for inclusion in a change project plan.

Our goal was to mimic a “skilled and conscientious consultant”. The planning system in **RAPTR** should be “skilled” in that it has at its disposal a broad methodological expertise and the ability to accurately diagnose the situation at hand. The planning system should be “conscientious” in that, unlike most consultants who base the magnitude of their approach on the available funds, **RAPTR** would only recommend the minimum number of necessary steps.

To provide this capability, we needed to build **RAPTR**’s knowledge around a broad-based methodology and then to tie the diagnostic model tied to that methodology. An extensive review of the BPR and change management literature provided the raw material to synthesize a comprehensive methodology. As for the diagnostic model, our original intent was to use the organizational literature to capture the causal relationships. While, in general, our literature review was useful, applying individual results in a piecemeal fashion was difficult and did not provide the desired result. Instead, using the resources at our disposal (including a Ph.D. organizational psychologist, a Ph.D. industrial engineer, and a Ph.D. anthropologist, with support from others with related advanced degrees), we developed a causal model (and the necessary instrumentation) from scratch. This approach is consistent with common practice in knowledge-based system development.

4.2 Knowledge Management/Document Management

The capability to manage “all of the data gathered, analyses performed, and conclusions drawn by the reengineering teams” was included in our proposal because of our experience on the COSAT project. While COSAT provided a methodology that many users valued, that first generation tool provided no real means to help users manage the myriad of data that can result from a reengineering project. We referred to this capability as a (Organization) Designers’ Notebook, a metaphor taken from some early hypermedia work at the Baylor College of Medicine¹⁰. They sought to develop “an electronic analog

⁹ Gasser, Les et al (1993) “Organizations as Complex Dynamic Design Problems”, in Miguel Filgueiras and Luis Damas, (eds.) *Progress in Artificial Intelligence*, Lecture Notes in AI#727, Springer Verlag, p. 4.

¹⁰ Shipman, Frank M. et al, *Distributed Hypertext for Collaborative Research: The Virtual Notebook System*, in the proceedings of Hypertext ’89, ACM, p. 129-135.

to the scientist's notebook...(a) repository of data, hypotheses and notes, patient information, and the like...expressly designed to enhance information sharing among members of scientific teams"¹¹. Their Unix-based system, developed prior to the emergence of the WWW, allowed researchers to store and share multimedia information from a variety of medical instrumentation. (A commercial version of their Virtual Notebook System (VNS) died a commercial death just as we began our search for technologies to apply to **RAPTR**.) But their core ideas were a powerful influence: a repository for collaboration, support for heterogeneous, distributed computers, and a basis on de facto standards (in their case, the X Window management system).

Of course, systems that could manage large quantities of user documents have been around for some time. This product category, referred to as "document management systems", typically meant large-scale, expensive systems (over \$100K) that operated in a LAN environment. These systems generally mandate user provision of "metadata", information about the files that can help users locate that file at a later date. This relates to the **RAPTR** Notebook Library, intended to allow "users to draw information from multiple Designers Notebooks for cross-project learning and comparisons". With our emphasis on the Web, this came to mean that **RAPTR** should include some means to index all of the collected information and provide users with capability to quickly and easily search this information space for the desired **RAPTR** asset. To meet this requirement, we investigated many different search technologies, listed in .

Most of these indexing/search technologies only indexed text and HTML documents. Our desire to minimize the "metadata" burden on **RAPTR** users led us to focus those search technologies that could index the contents of application files, not just HTML and text. With this additional requirement, AltaVista was the most obvious search engine choice. Livelink, a groupware product tied to one of the most powerful search engines available, was strongly considered for use in **RAPTR**. Their product engineers provided an in-depth demo to the project team. There were two issues that prevented us from using this product:

- Its high licensing cost (software licenses would have cost over \$100,000) and learning curve.
- While Livelink did allow for the creation of "projects", there was no real support for project management, as in Microsoft Project. With the strong process orientation of our effort, this difference in emphasis could not be ignored.

Company	Product Name	URL
Altavista	Personal Edition	http://altavista.software.digital.com/
CompassWare Development	InfoMagnet	http://www.compassware.com/

¹¹ Ibid., p. 129.

Inc.		
Excite	Excite for Web Servers	http://www.excite.com/
Frontier Technologies	CyberSearch	http://www.frontiertech.com/
Infodata	Virtual File Cabinet for Intranets	http://www.infodata.com/
Inktomi	HotBot	http://www.inktomi.com/
Microsoft	Index Server	http://www.microsoft.com/
Netscape	Catalog Server	http://www.netscape.com/
OpenText	Livelink Search	http://www.opentext.com/livelink/
Symantec	Internet FastFind	http://www.symantec.com/
Verity	Search '97	http://www.verity.com/

Exhibit 4.3 -- Search technologies reviewed for *RAPTR*

As is clear from our discussion of Livelink, a groupware/search/document management product, the lines between document management and groupware are blurry at best. As such, we examined product information on the groupware technologies shown in .

Our focus on wide-area collaboration and the WWW again limited the choices. In early 1997, the only real choices were Livelink, Notes (with their emerging Domino server), and WebShare from RadNet. WebShare built on the learning from Notes but in a totally WWW-based product. At that time RadNet was a start-up company formed by former Lotus executives. We felt that made using their product too risky. Wizdom's software development lead initiated discussions with Lotus on the requirements to become a certified Notes developer. While using a COTS package like Notes or WebShare as a framework for *RAPTR* could have given us a leg up in development, we made the decision to build our system around de facto standards, like HTML and CGI, and Wizdom's existing product family.

While our focus was on COTS, we also investigated a government off-the-shelf (GOTS) technology that met many of our project requirements. The Knowledge Worker System (KWS), developed by the U.S. Army Construction Engineering Research Laboratory (CERL), is a project-oriented groupware system that met most of our requirements for knowledge management. (For more information, please see <http://www.cecer.army.mil/>.) The KWS is a LAN-based system that, during the time of our discussions, was planning to move to the Web in a subsequent release. Initially we thought that we could employ KWS directly but our insistence on a Web-based solution and the timing of their product evolution schedule did not match. We had hoped to take advantage of the data

and process modeling that went into the design and development of KWS but we could not reach a mutually beneficial agreement with CERL.

Company	Product Name	URL
Amdahl	IntraNet Architecture, etc.	http://orpheus.amdahl.com/
Documentum	DocPage Server	http://www.documentum.com/
Fujitsu Software Corporation	TeamWare Flow 1.0	http://www.teamware.us.com
Hitachi	AdaptFile/VisiFlow	http://www.ais-hitachi.com/
Interleaf	BusinessWeb	http://www.ileaf.com/
Lotus	Notes	http://www.lotus.com/
Microsoft	Exchange, etc.	http://www.microsoft.com/
Netscape	Communicator	http://home.netscape.com/
Novell	GroupWise 5.0	http://www.novell.com/
OpenText	Livelink 7.0	http://www.opentext.com/livelink/
Radnet	WebShare 1.2	http://www.radnet.com/
SoftScape	Explorer Plus	http://www.softscape.com/
Thuridon	Crew	http://www.thuridon.com
WebFlow Corp	SamePage Suite	http://www.webflow.com/

Exhibit 4.4 -- Groupware and document management systems reviewed for *RAPTR*

4.3 Methodology Support

The simplest form of methodology support has been around for hundreds if not thousands of years: books. In *RAPTR*, we wanted to build on our experience with the COSAT project to provide a richer information source for change teams. Textual descriptions of each step of the methodology were a given. As in COSAT and other projects worked on by the team, we also wanted to provide links to tools and templates for use in some

project activities. Wizdom System's Document Manager product is an example of this approach. They provide a hierarchical outline of their Minerva BPR methodology, complete with templates for background investigations, data collection instruments, and samples different types of project reports.

In this time period, some organizations had already begun to move their paper-based assets to electronic form. The Air Force Materiel Command (AFMC) licensed Texas Instruments' (TI) Business Process Engineering (BPE) methodology for use within the command. The complete methodology, including all of the tools and templates, were provided in Microsoft Word format. Many other BPR consultants took much the same approach. During **RAPTR** we also interacted with OASD/C3I, an office that provided significant support to on-going DoD BPR-related efforts. They funded the development of a large reengineering methodology, The Framework for Managing Process Improvement, which was subsequently put up on the Web in hypertext format. Unfortunately the appendices that provided the details on tools, methods, and related topics necessary to apply this methodology were not made available on the Web.

Prior to **RAPTR**, some organizations built tools in the same vein. To support their BPR consulting work, TI also developed the Business Design Facility (BDF)¹², a toolset that built on their I-CASE experience. The goals of BDF sound exactly like those of **RAPTR**:

- "...the tools should be easily usable by BPR specialists"
- "...it should enable the visualization of the business processes in a format acceptable to business management"
- "...it should support the four main phases of BPR"
- "...it should be of open design, supporting any BPR methodology and interfacing to other tools within standard operating system services."¹³

BDF was released in 1993 and by the time of our investigations was no longer sold as a commercial product. The cited article on BDF also mentions a tool called ARRAE developed by Price Waterhouse (now PricewaterhouseCoopers) that they called the "most powerful software available in this area to date"¹⁴.

¹² Mills, Michael and Clive Mabey (1993), "Automating Business Process Re-Engineering with the Business Design Facility", in Kathy Spurr et al (eds.), Software Assistance for Business Re-Engineering, New York, NY: John Wiley & Sons, Chapter 10, p. 153-176.

¹³ Ibid., p. 155.

¹⁴ Ibid., p. 173.

4.4 Workflow Management

The Workflow Management Coalition, an industry group that defines standards in this area, provides the following definition:

Workflow Management System - A system that completely defines, manages, and executes "workflows" (the computerised facilitation or automation of a business process in whole or part) through the execution of software whose order of execution is driven by a computer representation of the workflow logic¹⁵.

In March 1997, we witnessed demonstrations of many workflow systems at the Business Process and Workflow conference conducted by the Giga Information Group. At that meeting Connie Moore and Derek Miers presented a very useful framework for discussing workflow systems¹⁶. This framework has three elements:

- The users' ability to modify the process - Process adaptability ranges from pre-structured processes, as in back office work in banks or order processing, to support for common practices that could result in the creation of personal objects, like custom reports or models. It is this type of task that would most likely be performed as part of a change effort.
- How work is managed and coordinated - The types of work to be supported range from a set of standard processes, again back offices with shared queues are used as examples, to knowledge management tasks, with shared documents created using standard input skills. Again, it is this latter type of tasks that would most often occur in change efforts.
- How the process interacts with information and applications - Finally, different systems can support the use of different types of documents. Simpler systems act as "electronic filing cabinets", simply moving documents between roles. As the systems get more complex, electronic forms may allow the movement of tasks and forms between users. Shared data spaces, with applications, data, and documents, are the next higher category. At its most powerful, a system may provide an integrated information repository that understands document structure and controls access to documents independently of process.

Exhibit 4.5 illustrates this three dimensional framework.

¹⁵ The Workflow Management Coalition, The Workflow Reference Model, Document Number TC00-1003, Issue 1.1, November 29, 1994, p. 5.

¹⁶ Moore, Connie and Derek Miers, "A New Future For Workflow", Business Process & Workflow Conference, March 3, 1997.

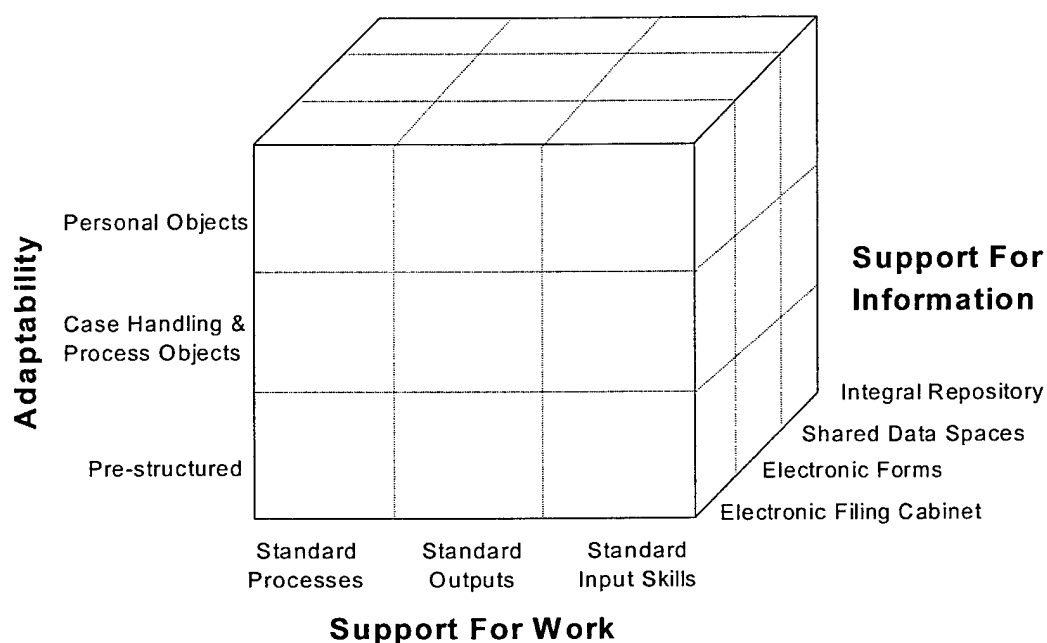


Exhibit 4.5 -- Workflow framework

At the 1997 conference, this framework was systematically applied to many of the major commercial workflow systems. (Many of the systems reviewed were also demonstrated in sessions at this same meeting.) While there were some minor differences, most of the systems best supported standard processes and “filed” pre-structured objects. Thus they would fit in the lower left-hand corner of this framework. If we consider the task that **RAPTR** was created to do, i.e., provide an integral repository for the management of, mostly, personal objects (application files), that are the result of context-specific processes. graphically illustrates that the **RAPTR** requirements are diametrically opposed to the state of the practice in workflow systems, at least at the time we made our design decisions.

The commercial system shown would also inhibit our ability to provide a browser-based interface to all of **RAPTR**’s functionality. Most of these systems separate their process definition environment, usually a special client program, from their process execution environment. These process definition clients ran on a stand-alone machine. As promoted at that conference, some of the process execution environments were moving to the Web as Java applets. Thus, the technology available at that time would not allow us to provide a single, browser-based interface to **RAPTR**.

The team’s experience using Microsoft products and knowledge of their planned evolution provided another avenue to meet **RAPTR**’s workflow needs. The Microsoft Exchange server, originally an email server, was to evolve beyond just email support.

Microsoft Outlook, a new product tied to Exchange, was introduced in June 1996¹⁷. In addition to email and contact management, Outlook provides for task management. Users can assign tasks to themselves or other Exchange/Outlook users. This delegation capability was then linked to the resource management functionality in Microsoft Project. As part of a project plan, managers can identify responsible parties and include their email address. As tasks are to begin or end, Project uses Exchange/Outlook to delegate tasks, providing some simple workflow functionality. Project's hierarchical view of a process does not provide for the same decision modeling and branching capabilities as most workflow systems, but the integration with Exchange, Outlook, and the Internet met our needs.

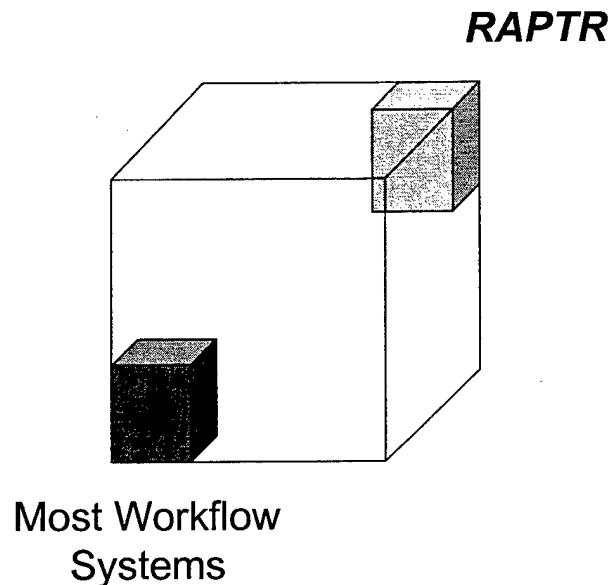


Exhibit 4.6 -- *RAPTR* requirements beyond scope of most workflow systems

A system much like **RAPTR** was cobbled together using Microsoft products by the Hydro-Electric Corporation in New Zealand¹⁸. Their focus was on project management, particularly to support change efforts. Their lengthy methodology, 197 pages with 220+ pages of supporting procedures and forms, had to be disseminated and consistently applied throughout the organization. They saw workflow technology as the answer and made the decision to develop a custom solution. Their table-driven system was built around Microsoft Project and provided access to methodology support and limited document management capabilities in a LAN-based environment. It provides:

- Context-sensitive guidance on phases, activities, tasks, and detailed procedures

¹⁷ <http://www.microsoft.com/presspass/press/1996/jun96/offc97pr.htm>, accessed September 16, 1998.

¹⁸ Chu, Johnny. (1997) "Hydro-Electric Corporation's PMLink: A case study of re-engineering through workflow computing", *Business Process Management Journal*, 3(2), p. 162-172.

- Access to input and output documents; and
- Reports on the status of project documentation across a project¹⁹.

In conclusion, the goals for *RAPTR* included knowledge-based planning support, knowledge management/document management, methodology support, and workflow management capabilities. Our investigations uncovered existing systems or commercial products that individually fulfilled many of these requirements. However, none of these systems met all of them, resulting in the need for some custom development to link together COTS components.

¹⁹ Ibid., p. 168.

Chapter 5: Narrative of *RAPTR* development

On September 5, 1995, a team led by Wayne State University, and consisting of the Industrial Technology Institute, and Wizdom Systems, Inc., proposed to the Logistics Research Division of the Armstrong Laboratory to create a tool that would “integrate the cultural, strategic, and user readiness factors” of an organization “into the business reengineering scenario.” The development program proposed had four objectives, which are quoted here:

- 1) Package a deductive cultural/strategic/change management model in a manner that makes it easily useful at the ALC.
- 2) Create and integrate context-based assessment methods, incorporating self-guided procedures, for implementing the deductive model at an ALC.
- 3) Incorporate the results into a working prototype, front-end Business Engineering context assessment and project management tool, supporting object-oriented business analysis, and other methods and tools.
- 4) Use multiple iterations and successive versions of these methods, procedures, models, and tools, to support Depot Reengineering at Warner Robins ALC.

We are pleased to report, that with technical modifications to objective #3, away from using object technology, all of these objectives were accomplished by the *RAPTR* program.

5.1 Phase I: From Vision to Concept

The *RAPTR* project, started on April 10, 1996. An management orientation was held among the technical partners (Wayne State, ITI, Wizdom) to resolve contractual issues and establish project roles. The launch of the *RAPTR* project among the technical partners was on April 18, with an all-day meeting to review concepts and technical priorities, and to plan a kickoff meeting at AFRL for May 6, 1996.

The agenda for the April 18 meeting was:

- Welcome, introductions, organizational presentations (all)
- The USAF CABE program (Presentation by AFRL/HESS)
- RAPTR* scope review
- Phase I schedule
- Initial task assignments

- Task management procedures
- Communications
- Subcontracts and administrative procedures (Presentation by Dan Graf)
- April 25 technical orientation
- Planning for kickoff meeting at WPAFB

After the kickoff, the three partners held a technical orientation (April 25, 1996), at which lead concepts for **RAPTR** development were reviewed, and each partner's capabilities and contribution were discussed.

The next milestone was a research planning meeting on April 25, 1996. At this meeting we started an issues log that tracked open, conceptual issues. Typically these issues were sufficiently high level that closing them out required months; some remained open for the duration of the project (for example, issues of usability); others, such as the use of Web media, were resolved fairly quickly.

The actual kickoff of the **RAPTR** project was held at Armstrong Laboratory on May 6. The agenda for the kickoff was as follows:

- Introductions
- Technical Approach
- Technical Risks
- Travel
- Major unresolved issues
- Benefits
- Programmatic issues

Two elements of the kickoff are worth elaborating here, because from the hindsight of 30 months, they proved to be quite prescient in terms of the opportunities and challenges faced by the **RAPTR** development.

Half of the kickoff was spent on the technical approach. We presented a view of **RAPTR**, shown here in exhibit 5.1 on the next page, that was modified as the project evolved. The major variance in this evolution was in the middle term, the "Assessment tools and benchmarking": As we developed the models and tools that these were based on, we determined that the results would be less linear than the notion of an optimizer (with feedback for successive iterations) would suggest. The collection, management, and user-accessible presentation of non-linear phenomena was one of the major challenges of the **RAPTR** research.

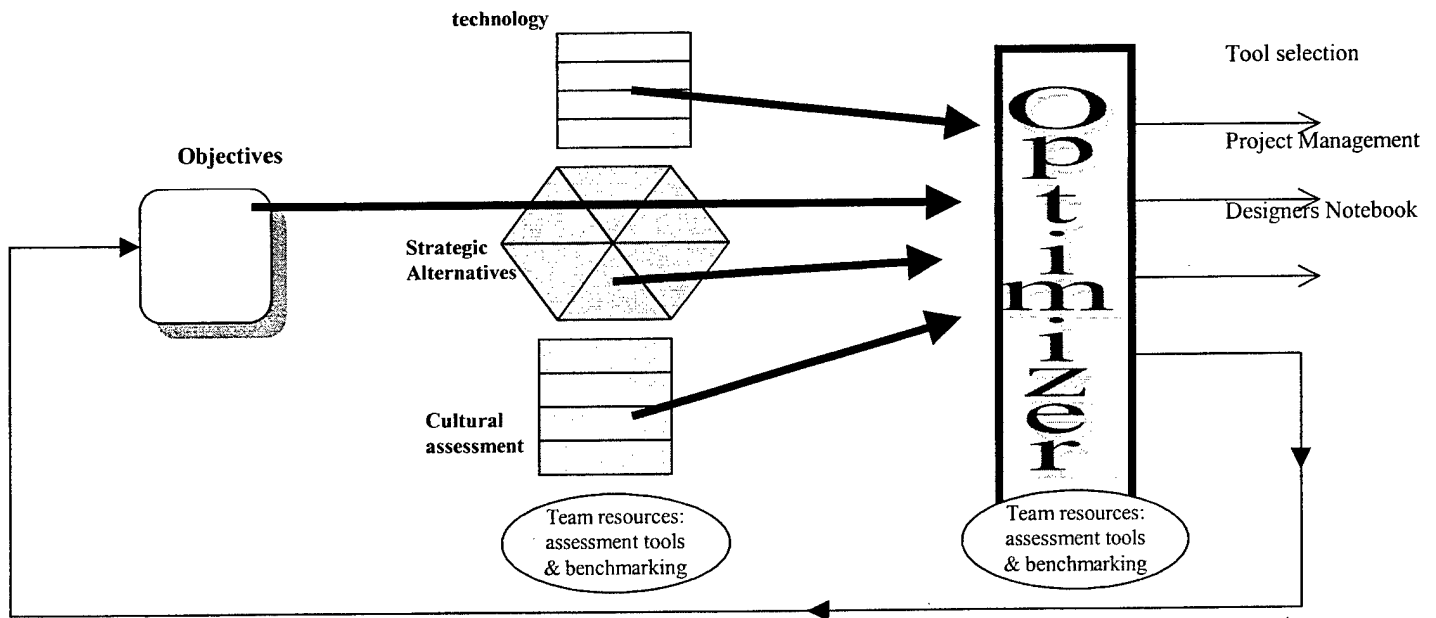


Exhibit 5.1 -- Initial *RAPTR* concept

In the discussion of technical risk, we identified eight risk factors, which are enumerated here:

1. Achieving a neutral view of Business Process Reengineering Methods, particularly given the fact that project teams and experts make significant investments in these methods.
2. Finding the proper scope of Business Process Reengineering issues: How much depth and completeness are desirable?
3. Presenting a user scenario with a "look and feel" that would engage the users.
4. Finding representative case materials with which to populate the notebook library, particularly given the fact that most projects do not leave good project archives.
5. How to present assessment results: as descriptions, directives, or advice?
6. Integration of *RAPTR* with the University of Arizona DOME effort. Initially *RAPTR* intended to integrate organizational attributes with IDEF0 and IDEF1x models; this was set aside as it became clear that DOME would be working on this same issue.
7. Seamless integration of *RAPTR* components.

8. Adapting to an evolving technological environment.

Management of these risks received ongoing attention through the remainder of the project.

The first task in the project was to define a conceptual architecture for **RAPTR** which would identify designable components. Setting aside issues of computability, and levels of automation, the team identified twelve components within **RAPTR**. These are, with description:

Reference model of reengineering: The “backbone” of **RAPTR**, a compilation of standard reengineering tasks as derived from the literature and experience. The Reference Model was also referred to as the “gene pool” of change management, inasmuch as any specific project would draw on some but not all of its elements.

Process modeling and characterization: The ability to create or import process models and add performance attributes such as throughput or process stability.

Goals and objectives: A description and characterization of an organization’s objectives in a reengineering scenario.

Characterization of the organization: Basic organizational data including size, complexity, hierarchy.

Technology assessment: A characterization of the AS-IS technology of the reengineering target.

Communication assessment: A characterization of the communication media and effectiveness within the organization.

Cultural assessment: Identifying those aspects of an organization’s culture, such as value given to learning, that promote either acceptance of or resistance to change.

Project management / workflow manager: A tool that would identify necessary tasks in a reengineering project, and manage the flow of documents through those tasks.

TO-BE process design: A tool that would provide advice for TO-BE process alternatives, based on a characterization of the AS-IS process.

Team resources: Methodological tips, templates, guides, and software tools for executing the tasks in the Reference Model.

Designers notebook: An evolving project document which assembles both active and completed project documents.

Notebook library: A searchable repository of designers notebooks from prior projects.

(Early on, the team realized that the term “process” could engender considerable confusion, and so developed a shorthand for different processes and meta-processes. Actual business processes, through which Air Logistics Centers and other organizations fulfill their mission, were referred to as P1; reengineering scenarios, such as those described by Andrews and Stalick, were referred to as P2; the **RAPTR** operational scenarios, which would support and enhance reengineering (P2), were referred to as P3. Thus the Reference Model of Reengineering was frequently referred to as RMP2.)

Model Template RAPTR Conceptual Architecture	
1.	Model Component
2.	Use in RAPTR (include subsystem identification)
3.	Priority
4.	State of Knowledge and Technology in Relevant Domains
4.1	Critical literature sources
4.2	Leading COTS
4.3	Candidate notations
5.	Required advancements
5.1	Digest of technical issues
5.2	Technical detail -- dimensionality
5.3.	Technical detail: content
5.4.	Technical detail: depth management strategy
5.5.	Technical detail: User engagement
6.	Plan for Achieving Required Advances
7.	Interfaces
7.1	Components interfaces with
7.2	Information passed across interface
7.3	Interface translation issues
7.4	External interfaces
8.	User benefit and its measurement
9.	Advice created
10.	Feasibility
11.	Time, skills, and resource requirements
12.	Risks
13.	First draft to be prepared by:
14.	Deadlines

Exhibit 5.2 – Component template

To add further specification to each of these components, ITI created the template shown in Exhibit 5.2; the assemblage of these became the foundation for the Operational Concept Document. After the OCD was completed and we shifted our focus to requirements, the issue of automating each of these twelve components came to the front.

Our approach to analyzing and determining this is described in section 5.2.1 of this chapter.

A second template that was created was the Operational Scenario template, shown in exhibit 5.3. The purpose of this template was to identify the different activities that a user might wish to use **RAPTR** for. The nine user scenarios to be supported by **RAPTR** were:

1. Learning about change management
2. Learning about an organization
3. Initiating a new project
4. Reviewing project status
5. Executing a project
6. Checking out a model
7. Designing a *TO-BE* process
8. Browsing past projects
9. Conducting a directed search

The two critical scenarios in this list are #2, Learning about an organization, which is the assessment functions, and #5, Executing a project, which is the workflow function.

These two templates provided cross-cutting views of **RAPTR** which, when integrated, should fully specify not only its operational concept but also its conceptual architecture. Development guided by these templates extended from May to July, 1996, and resulted in the Operational Concept Document, the first version of which was submitted on August 10, 1996.

Model Template RAPTR Operational Scenario	
1.	Name of scenario or operational mode
2.	User
3.	Process description
4.	Process support and interfaces
5.	Modules/models used
6.	Requirements placed on user
7.	Input data and sources
8.	Output information
9.	Benefit of the RAPTR approach
10.	Measurement of benefit
11.	How does this empower the user
12.	Deadline for this report
13.	Lead effort

Exhibit 5.3 – Operational scenario template

A third, cross-cutting view of **RAPTR** from a user viewpoint was created in the form of an IDEF0 model identifying both high-level and detail-level functions, and the nesting of and interfaces among these. As component-level definition proceeded this model was used less, until component acceptance required validation of the interfaces among the components. At this point the interfaces identified in the IDEF0 model were used to validate component acceptance.

As part of developing the **RAPTR** concept, the team reviewed numerous off-the-shelf tools that could conceivably be modified or adapted to provide **RAPTR** functionality. Among the OTS tools, both commercial and government-owned, that we reviewed, were ICASE, Lotus Notes, Oracle's CDM Advantage, Turbo BPR, KnowledgeWorker System (KWS), and other tools, as described in the previous chapter. This activity extended from early in Phase I to the completion of the Software Design Description on December 17, 1997.

Lotus Notes provides the workflow functionality that was envisioned for **RAPTR**. However, Notes is intended for processes (of type P1) that are repeated multiple times, rather than the unique, one-pass process (P2) that is typical of reengineering projects. Interfaces to other components were also of concern.

I-CASE (Integrated Computer-Aided Software Engineering) was intended as a complete toolkit incorporating every major CASE tool. The integration was never achieved, and at the time of our review I-CASE existed only as an ordering vehicle for software. This was a sobering object lesson in the dangers of placing too much emphasis on integration at the expense of useful and unique functionality.

The **KnowledgeWorker System** appeared to resolve the problem of customization for unique, knowledge-intensive workflow. We were unable, however, to conclude a licensing agreement with the Construction Engineering Research Laboratory that would open up the source code and permit us to integrate other **RAPTR** components with the tool.

Turbo BPR was examined, but its limited analysis potential led the team to not pursue its integration.

CDM Advantage is a tool that Oracle Corporation occasionally advertised. Its purpose was to facilitate systems design and development among collaborative teams, and its workflow capabilities appeared to be suitable for **RAPTR**. The Oracle sales staff, however, did not respond to repeated inquiries for product data, and were clearly not interested in selling the product.

In sum, none of these products was both sufficiently accessible and sufficiently functional to be pursued as a major component of **RAPTR**. When Wizdom succeeded in building a direct interface to the Microsoft Project (MSP) file structure (April 25, 1997),

we concluded that we could achieve the sorts of project planning and workflow support we were looking for by integrating with MSP.

One other set of COTS tools that was reviewed, successfully, was the set of search engines. The team reviewed several search engines, and invited vendors in for demonstrations. Search engines reviewed included **Search 97**, **Livelink**, **Cyber Search**, **Ultraseek**, **Lycos**, and **Alta Vista**. After review, AltaVista was chosen.

Parallel with and supporting the development of the Operational Concept Document, members of the team began conducting fieldwork at Warner Robins AFB. The purpose of this fieldwork was to answer the following questions:

- What were the current methods for undertaking reengineering at Warner Robins ALC?
- What needs could a tool such as **RAPTR** supply?
- What features should be incorporated into the tool to assure maximum usefulness?
- What were the cultural issues in change management, and barriers to organizational effectiveness within an Air Logistics Center?
- What were the current methods for understanding and managing these cultural issues?

WR-ALC/RE was briefed on the results of these last two questions in March, 1997. Findings on the first three were incorporated into the **RAPTR** tool development.

The Phase Review for Phase I was held on January 31, 1997, and approval was given to proceed to Phase II.

5.2 Phase II: From Concept to Prototype

After completing the Operational Concept Document (submitted on August 10, 1996; comments received from AFRL on September 21, 1996; revisions submitted on December 6, 1996), the team began focusing on translating the concepts into designable components.

5.2.1 System Architecture

Five activities led the architectural effort:

1. Defining a band of automation for different components
2. Prioritizing different components
3. Continuing review of OTS tools

4. Ongoing development of the Reference Model
5. Ongoing collection of ethnographic data at Warner Robins ALC

Additional activities supporting the Requirements Specification that followed included

6. Specification of an operating environment
7. Design of a graphical user interface
8. Specification of data interfaces among **RAPTR** components
9. Specification of navigation paths among **RAPTR** components

Two of these require elaboration. Number 1, the team recognized that there were numerous levels of automation possible for each function. The team established a framework for automation, consisting of seven levels:

0. Paper-and-pencil systems
1. Simple transactional systems
2. User configured computing systems (e.g., spreadsheets)
3. Automated reasoning
4. Artificial intelligence (e.g., expert systems)
5. Learning systems
6. Automated learning

For each component we specified a band of automation on this model, ranging from minimal acceptable level to maximum feasible; within that band we identified an optimum level of automation, which became our automation target for the component.

(In retrospect, we estimate that each step up the scale of automation multiplies development costs by a factor of approximately ten; thus a user-configured reasoning system costs ten times as much to develop as a transactional system. See chapter nine, lessons learned.)

In prioritizing the components, the team established three viewpoints: that of the end-user at Warner Robins, that of the project sponsor AFRL, and that of the development team. Each component was rated, from each of these viewpoints, on the following scale:

Must have
Desirable
Don't care

Precedence was given to the Warner Robins viewpoint, and the twelve components were prioritized. One component in particular, the *TO-BE* design, was rated sufficiently low, with sufficient development risk, that it was eventually dropped from the **RAPTR** design.

The team continued its review of off-the-shelf tool alternatives, up through the completion of the software design description.

Development of the Reference Model began conceptually in June of 1996 and lasted well into Phase II. The Reference Model, which is shown in Exhibit 5.4 on the next page, can be considered a methodological superset, embracing change management activities described in numerous standard sources (citations). In building this reference model tradeoffs were made among four design criteria:

- The model had to be tailorable; inasmuch as it was a methodological superset, there should be a procedure for selecting certain activities for actual projects.
- The model had to be integrated, identifying internal dependencies among tasks.
- The model had to be flexible, displaying tasks at different levels of indenture.
- The model had to be recognizable: a redesign team should be able to review the model and identify those activities that comprised their standard methodology.

CHANGE MANAGEMENT REFERENCE MODEL

STAGE I: CONDUCT STRATEGIC ASSESSMENT

- Task 1: Kickoff Stage I
 - Activity 1: Develop design philosophy and project vision
 - Activity 2: Select the executive committee and project team
 - Activity 3: Train the executive committee and project team
- Task 2: Conduct Business Overview
- Task 3: Assess Business Goals and Opportunities
- Task 4: Identify Opportunity Areas
 - Activity 1: Conduct environmental scan
 - Activity 2: Determine project goals and outcomes
 - Activity 3: Develop executive approval
- Task 5: Determine Project Scope
 - Activity 1: Determine project boundaries or sizing
 - Activity 2: Identify project champion
 - Activity 3: Identify stakeholders
 - Activity 4: Determine resource needs
 - Activity 5: Define project management infrastructure
 - Activity 6: Select data collection and analysis methods
 - Activity 7: Obtain executive approval

STAGE II: CONDUCT AS-IS ASSESSMENT

- Task 1: Kickoff Stage II
- Task 2: Assess Internal Company or Agency Characteristics
 - Activity 1: Assess workflow
 - Activity 2: Assess technology
 - Activity 3: Assess organization
 - Step 1: Document organizational structure
 - Step 2: Identify cross-functional coordination processes
 - Step 3: Document job and work design
 - Activity 4: Assess cost
 - Activity 5: Conduct other recommended assessments
 - Option 1: Assess culture
 - Option 2: Assess personnel and HR practices
 - Option 3: Assess communications and information flow
 - Activity 6: Conduct other assessments as indicated
 - Option 1: Assess project management
 - Option 2: Identify product characteristics
 - Option 3: Assess supplier management
- Task 3: Analyze Baseline Performance and Desired Improvement
- Task 4: Assess Environmental Fit
- Task 5: Define or Re-define and Rank Projects
- Task 6: Re-examine and Approve Scope and Resource Needs

STAGE III: CREATE TO-BE DESIGN

- Task 1: Kickoff Stage III
 - Activity 1: Determine or re-iterate TO-BE vision
 - Activity 2: Refine TO-BE scope
 - Activity 3: Determine TO-BE expected outcomes
- Task 2: Develop Design
 - Activity 1: Envision ideal culture
 - Activity 2: Develop workflow design
 - Activity 3: Develop organizational design

- Step 1: Design organization structure
- Step 2: Design cross-functional coordination
- Step 3: Design work and job design
- Activity 3: Develop technology design
- Activity 4: Develop other recommended designs
 - Option 1: Develop personnel/HR support systems
 - Option 2: Develop communications information flow design
- Activity 5: Conduct other design as indicated
 - Option 1: Develop project management design
 - Option 2: Develop supplier management design
- Task 3: Define TO-BE Measurement Strategy
- Task 4: Consider Alternative Designs and Create Final Business Case

STAGE IV: PLAN AND IMPLEMENT CHANGES

- Task 1: Kickoff Stage IV
 - Activity 1: Determine design specifications
 - Activity 2: Define change management program
 - Activity 3: Specify implementation roll-out plan
 - Step 1: Develop implementation communications plan
 - Step 2: Develop & deliver change agents training
 - Option 3: Develop culture change plan
 - Option 4: Develop workflow change plan
 - Option 5: Develop organizational transition plan
 - Option 6: Develop personnel/HR plan
 - Option 7: Develop comm/info flow transition plan
 - Option 8: Develop technology transition plan
 - Option 9: Develop project management plan
 - Option 10: Develop supplier management plan
- Task 2: Pilot Implementation
 - Activity 1: Prototype projects
 - Activity 2: Create change communications plan
 - Activity 3: Choose implementation process
- Task 3: Implement Changes
 - Option 1: Implement culture changes
 - Option 2: Implement workflow changes
 - Option 3: Implement organization changes
 - Option 4: Implement personnel/HR changes
 - Option 5: Implement communication/info flow changes
 - Option 6: Implement technology changes
 - Option 7: Implement project management changes
 - Option 8: Implement supplier management changes

Exhibit 5.4 – Change Management Reference Model

Achieving tradeoffs among these four issues (for example, the more integrated the model, the more difficult it would be to downselect individual activities) was a major focus of design work on the reference model. As the model began taking shape, team members from the Industrial Technology Institute began associating tools and methods with each activity. These became the Team Resources capability of **RAPTR**.

Following the submission and acceptance of the Operational Concept Document, an explicit decision was made to employ Web technology for integrating **RAPTR** platforms. Initially **RAPTR** had been conceptualized as a LAN-based tool; however, with the rapid development and proliferation of Web technology through 1996, the use of HTML, CGI, and Java applets, with easily available Web browsers gave complete access to the functionality that **RAPTR** was intended to supply.

As part of the development of the Software Requirements Specification, the team created diagrams of navigation paths among different components (now to be represented by Web pages). An example of these diagrams is shown in Exhibit 5.5. A student of graphic design at Wayne State, Jennifer Jesse, was brought onto the team to assist with screen design and web page standards. Ms. Jesse traveled to the user location at Warner Robins to collect ideas that were eventually incorporated into the main **RAPTR** home page.

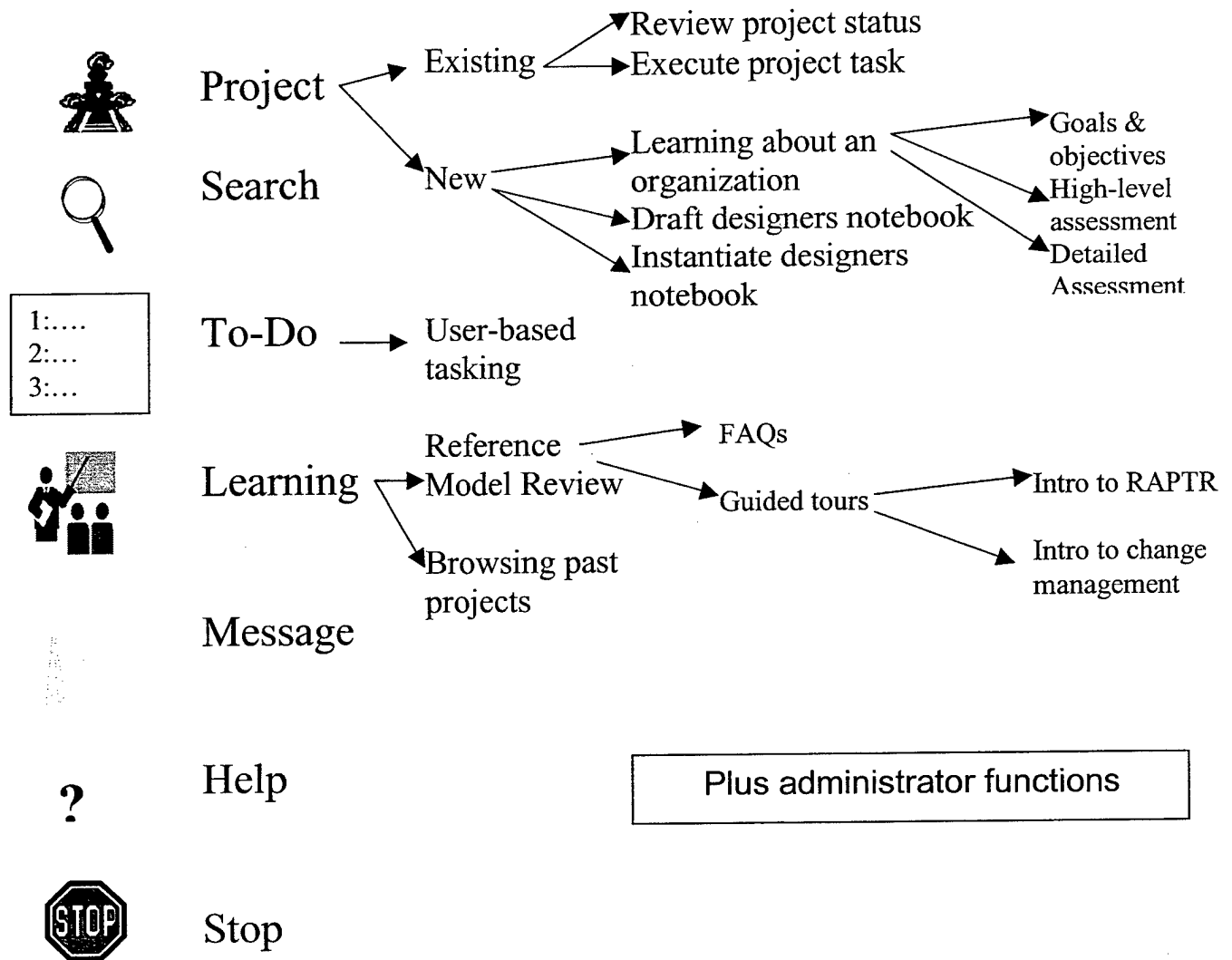


Exhibit 5.5 – **RAPTR** navigation paths

Another activity supporting the Requirements Specification was the modeling of data interfaces among the different components. Although design of data tables was left to Wizdom Systems, Inc., team members from Wayne State and ITI were active in defining data elements that had to be passed from one component/page to another.

An ongoing activity paralleling the development of the Requirements Specification was the fieldwork being conducted at Warner Robins. This activity enabled the developers to understand the user environment, and which **RAPTR** functions would be most beneficial to the users.

Two significant events in the development of the prototype were the creation of an interface with Microsoft Project, and the design specification of **RAPTR** components. In the week of April 25, 1997, Wizdom Systems, Inc., presented the first version of a run-time interface with Microsoft Project data files. Using Visual Basic, they demonstrated the ability to manipulate tasks and schedule information in the MSP data, independent of the MSP application. The significance of this was that it gave **RAPTR** the ability to use MSP functionality, including data management, independent of the MSP application or interface. In this scenario the **RAPTR** user had the ability to create task data in **RAPTR**, export it to a MSP file, calculate schedule information in MSP, and then present the results in a **RAPTR** window:

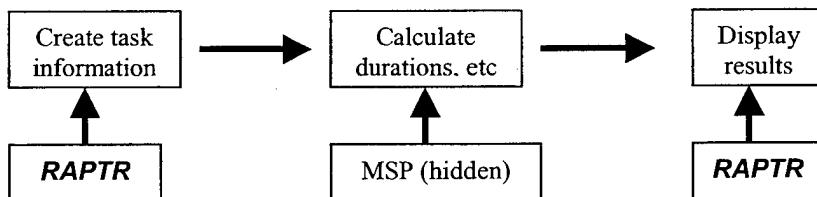


Exhibit 5.6 – Microsoft Project integration

The significance of this for the **RAPTR** architecture was that the integration of a **RAPTR** UI with MSP could supply all the functionality the team had originally desired from the Knowledgeworker System (KWS). At a team meeting on May 12, 1997, this link was extended to Wizdom's Methodology Navigator, and the ability to pass data from one to the other was demonstrated. With this demonstration, and with the difficulty in gaining access to KWS, Microsoft Project replaced KWS as a leading candidate for **RAPTR** integration.

The second significant event was an initial presentation of **RAPTR** architecture on March 11, 1997. This architecture, shown with refinements in Exhibit 5.7, provided the basis for allocation of **RAPTR** functions to specific software components. From this point

forward, creation of the Software Design Description was essentially an elaboration on and a formatting of this architecture.

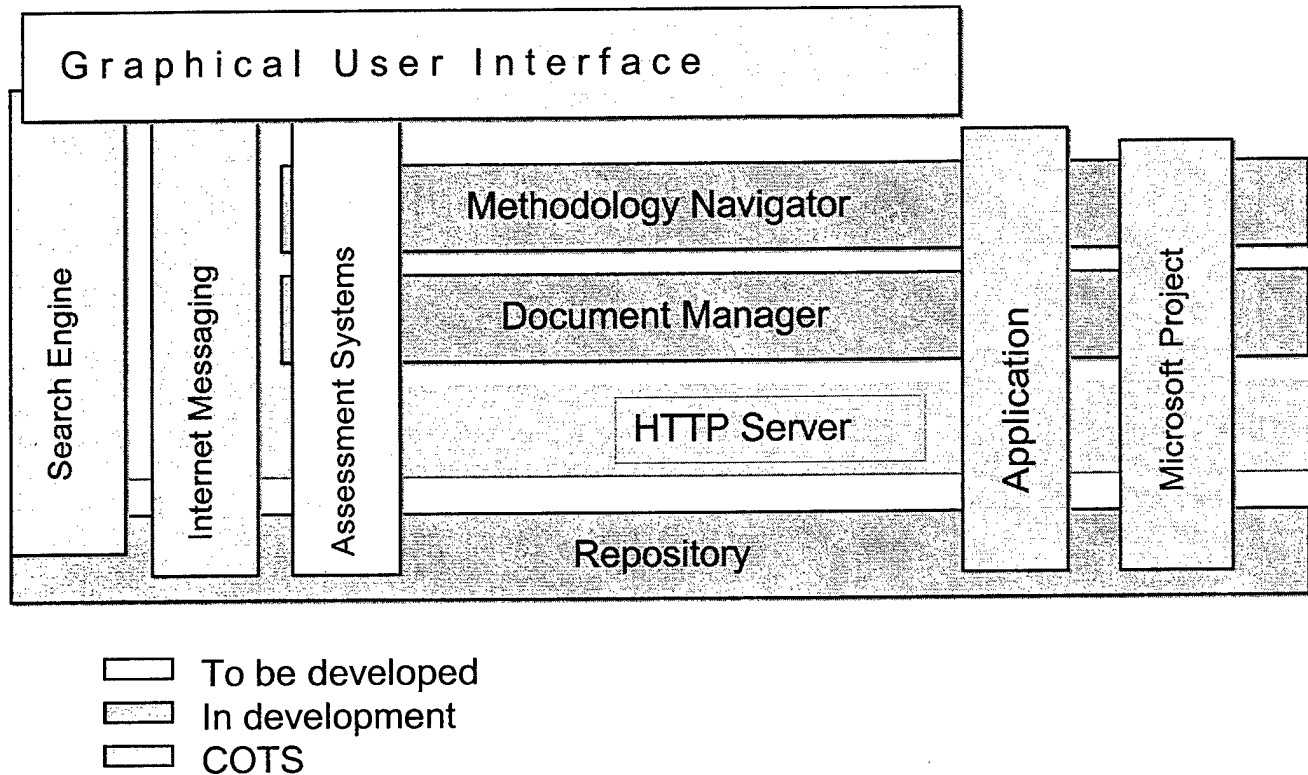


Exhibit 5.7 – RAPTR architecture

A new task that paralleled the development of the SRS, but never became part of it, was the Pacer Lean Lessons Learned (PL3) database. In November of 1996 the **RAPTR** team received a request from HQAFMC to look into using RAPTR technology to capture Lessons Learned from the Pacer Lean implementation. This appeared to be quite feasible, given the fact that **RAPTR** would itself have a Lessons Learned database. Two web pages were mocked up for demonstration purposes. An initial demonstration of these at HQAFMC was received favorably; subsequent to this, however, demonstration at Oklahoma City ALC on March 19, 1997, raised questions regarding who would actually be using and maintaining the database. This effort was suspended pending further guidance from AFMC.

The Software Requirements Specification was submitted to AFRL on April 20, 1997; comments were received in May, and a revised version was submitted in June, 1997.

Inasmuch as the system architecture specified both the creation of new components and the integration of these and OTS components, Wizdom was able at this point to proceed with specific coding tasks; the Software Design Specification, which was due for

submission on December 10, 1997, became an in-progress codification of design decisions that had been made and embodied into an evolving prototype.

As an integration and communication tool, the team created an IDEF0 model describing **RAPTR** functionality. This IDEF0 model, shown in indented list form in Exhibit 5.8, identified 39 functions (at all levels of indenture) that comprised **RAPTR**. The graphical version of the model identified the data interfaces among these functions.

5.2.2 – Knowledge Content and Scenarios

With the operational concept approved and the basic software architecture established (and functions allocated to software components), there were several design activities that could proceed in parallel. Some of these were a matter of passive content, and hence could lag other developments; some pertained to the configuration of existing shells, such as AssessTech (FRAME/WORK); others were critical to the overall integration of **RAPTR**, and hence were given priority.

Guide **RAPTR** Use

Identify Project

- Set up Project identification
- Enter project meta-data
- Initialize designers notebook

Generate team readiness advice

- Describe project
- Describe team

Review Change Management Library

Characterize the Current Situation

- Obtain user answers to questions
- Combine answers for variable values
- ZTYP step pre-select
- Do look-up for variable contribution
- Summarize variable probability table

Plan Project

Tailor task list

- Construct ZTYP task set
- Suggest DetA task steps
- Process task dependencies and prerequisites
- Suggest scale & emphasis advice
- Select time resource tradeoff
- Calculate task difficulty advice
- User edits task list

Do detailed task planning

- Set schedule to tasks
- Detail tasks
- Attach resources to tasks
- Gain team consensus
- Get management approval

Manage and Execute Project
Task status update
Change project definition
Add task
Delete task
Reorder tasks
Review designers notebook
Close out project
Enter completed project metadata

Exhibit 5.8 – *RAPTR* functionality (software view)

5.2.2.1 – Team Self Assessment (standalone function)

The first of these was the team self-assessment (TSA). The *RAPTR* team concluded that one problem with some change management projects was that the change team was unprepared, not properly briefed, or mis-aligned with the scope of the problem. A simple assessment was created, which would return advice of Go-Ahead, Proceed with Caution, or STOP!, using a traffic light icon. This module was completed in June 1997, and subsequently configured into AssessTech by Wizdom.

5.2.2.2 – Assessment Variables and Measurements (configuration item)

Within the actual assessment, definition of variables, measurements, and results was essential to the functioning of *RAPTR*. A team consisting of Bill Hetzner, Mitch Fleischer, Allen Batteau, Ron Kohler, and Ben Mejabi met from November, 1996 to the end of April, 1998 to define the deductive model. The PI imposed design criteria and targets of context sensitivity, non-intrusiveness, and user-interpretable results. Early on the team abandoned the simplistic classification of cultural, technological, and organizational assessment, and instead established seven domains of assessment:

- Process
- Organization
- Personnel and human resources
- Culture
- Technology
- Communication
- Supplier/Customer relations

Within these seven, there were from three to fifteen variables for each domain.

An early formulation of the model is shown in exhibit 5.9. The PI pointed out that essentially this model stated that “everything is related to everything”, and presented no

computational expression of model relationships. The team then refined the model to distinguish among four different types of variables:

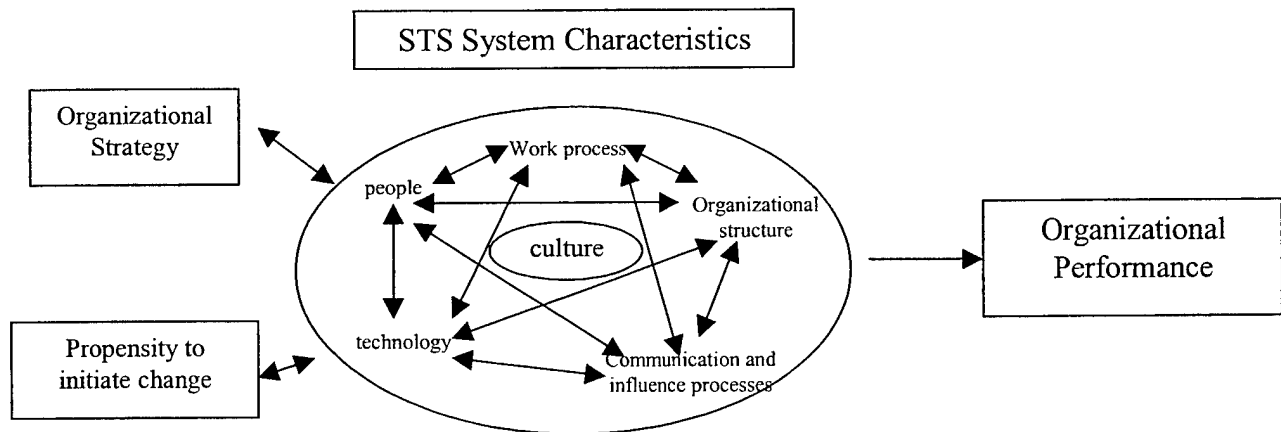


Exhibit 5.9 – Sociotechnical Performance Model (Early version)

Performance measures (PM) – the desired outcomes, in terms of flexibility, responsiveness, efficiency, or other strategic outcomes.

Intermediate outcomes (IO) – system states, such as Leadership Effectiveness, that have a clear “goodness” or “badness” to them given a desired performance outcome. These IO variables are typically not directly changed.

System Features (SF) – aspects of an organization that can be changed by management or other action.

Moderating Variables (M) – aspects of an organization that, within the scope of a typical change project, cannot be changed. An example of an M variable would be “Frequency and Scope of Previous Change” (EPHC).

The variables were then sorted into those that were required to diagnose an organization and scope a project (all the IO variables, plus 13 SF variables, and three Moderating variables) and those that would be used for detailed diagnoses. This established two assessment scenarios, the High-Level Assessment (HLA) and the Detailed Assessment (DetA). The HLA would measure 31 variables, and from this would make recommendations on project scope and plan, and would indicate certain organizational attributes due for further investigation.

Related to this was the requirement to develop and calibrate measurements for the variables. Between three and five measurement items were developed for each variable, usually involving Likert scales of the sort:

How important is quality as a measure of performance for this process?

Unimportant
Somewhat important
Important
Very important
Extremely important.

We practice continuous improvement for the target process.

Strongly agree
Somewhat agree
Neutral
Somewhat disagree
Strongly disagree.

Computational procedures appropriate to ordinal data were developed for establishing values for variables based on these measures.

When a sampling of the change target takes the HLA, one result returned is a report on the condition of the 15 IO variables, compared to USAF norms (established during our fieldwork). Overall scores are represented by the length of a bar next to the variable name; those in an unfavorable range are represented by a red bar, those in a favorable range are represented by a blue bar.

Once the team had preliminary measures for the variables, we tried them out in paper-and-pencil form at Warner Robins and Ogden ALCs. Some consideration was given to using a focus group to establish the value of certain variables; difficulty in managing the focus group led the team to abandon this idea. Henceforth all variable values were to be established by user input in an interactive environment.

5.2.2.3 – Variable descriptions (passive content)

After viewing the results of the HLA, the user can click on one of these bars, and see a description of the variable, and an identification of which variables are related to it. The team developing the assessments also developed short, one-paragraph descriptions of each of the variables. An example is shown in Exhibit 5.10.

Work Group Innovativeness

How willing and able is the workgroup to change? Some workgroups are highly innovative, and able to quickly adopt new ways of doing things, while others are either unwilling, unable, or both. In change efforts, innovative groups may be targeted for the first changes, so that they may set an example for others. On the other hand, change efforts may need to be introduced differently for non-innovative groups.

Some other variables that affect work group innovativeness include:

- Status conferred by Technology
- Rate of Technical Change
- Strategies
- Mechanistic vs. Organic organization structure
- Organization values
- Commitment to existing workforce
- Cross-functional communication
- Variety of skills within process
- Practice of continuous improvement
- Job design for fulfillment
- Motivation systems
- Leadership style

The variables with the strongest relationship to this variable are

- Middle and line management commitment to change
- Value given to learning
- Skill Acquisition systems

Exhibit 5.10 – Example of a variable description

5.2.2.4 – Model integration (configuration item)

Wisdom's AssessTech tool is a dynamically configurable assessment tool in which responses to certain questions can be used either to trigger other questions or to configure other assessment questionnaires. **RAPTR** made use of this capability by having results of the HLA configure a Detailed Assessment. The initial concept was to have detailed assessments focusing on different aspects of culture, technology, etc. As the model was

built, it became clear that having each unfavorable IO result trigger the measurement of selected SF variables was more appropriate. In theory, **RAPTR** can thus configure 1.3×10^{12} (15 !) different Detailed Assessments; as a practical matter, a user will probably drill down on only two or three of the unfavorable IO conditions. However, since there is a many-to-many relationship between IO conditions and SF states, the team had to map out all possible relationships, and decide at what value of an IO variable a particular SF or M became relevant. Doing this involved creating a 70 x 70 matrix and going through each cell to determine if there was a relationship or not. This intensive task was completed on May 5, 1998. Prior to this final release several elements of the detailed assessments had been released and tested.

5.2.2.5 – Change Management Reference Model specification (integration item)

The CMRM continued to evolve, particularly as results came in from the following two tasks. The final result is shown in Exhibit 5.4. This represents the feasible compromise among the design criteria previously presented.

5.2.2.6 – CMRM activity descriptions (passive content)

The Activity Descriptions were a lagging task, inasmuch as they were simply screens that were displayed for each task in the CMRM. Initial efforts in drafting these indicated some changes in the CMRM, particularly the collapsing of some tasks and adding detail to others. This activity was only 40% done by the time of the beta release; however, it had no effect on the functionality of the software.

5.2.2.7 – Scoping scenario and alternatives (integration item)

Probably the most difficult of these elements was integrating the High Level Assessment with the Change Management Reference Model in order to give advice on project planning and scoping. This first involved a decision as to whether the advice given would be indicative (do this v. don't do this), or a matter of relative emphasis (focus on this). The team realized that indicative advice would probably result in the entire CMRM being recommended; hence the decision was made to provide advice on task emphasis and required effort.

Data were collected from experienced project personnel, and from the Principal Investigator's own experience, regarding level of effort for each task in the CMRM for different sizes of organizations. The 31 variables in the HLA were mapped to each of the detail activities and options in the CMRM, and levels of emphasis (low, medium, high) and duration (% of total project effort) were established. This configuration was then turned over to Wizdom, which created the bridge between the AssessTech database and the Microsoft Project data file structure for scoping a project.

In the use scenario, after viewing the results of the HLA, the project manager can conduct an initial scoping, and then add or delete tasks and view the results for the overall project scope. When the project manager is satisfied with the result, a project is created in MSP and the Designers Notebook is configured for ongoing use in a workflow mode.

5.2.3 – Other Opportunities

In addition to the Pacer Lean Lessons Learned database, several other opportunities emerged during Phase II. Several of these remain open as potential areas for further development.

A conference at the USAF Safety Center on Stressed Systems, held on August 5-6, 1997, heard a presentation on **RAPTR**. This presentation, by Major Joyce Adkins, examined **RAPTR** as an organizational assessment tool for dealing with a critical USAF problem: the stress placed on organizations as budgets and resources become increasingly misaligned with mission requirements. Followup inquiry with the authors of the Stressed Systems study suggested possible interest, but none that indicated an alteration of immediate development tasks.

A parallel project with **RAPTR**, the DOME (Depot Operations Modeling Environment) project (prime contractor, University of Arizona), initially seemed to have some areas of convergence (and possibly duplicated functionality) with **RAPTR**. Two meetings were held, one at AFRL on August 13, 1997, and the other at a DOME Phase Review at the University of Arizona on June 26, 1997, to examine these issues. The provisional conclusion emerging from these discussions was that **RAPTR** was primarily a front-end project planning tool, with workflow capability, whereas DOME was a project management environment with integrated analysis capability. An area of convergence that interested the **RAPTR** team was the PIMA (Process Integration, Modeling, and Analysis) function of DOME. This was a function that had been discussed for **RAPTR**, but planning had given it a low priority due to technical risk. The understanding that the DOME project would be building PIMA suggested that **RAPTR** might be able to leverage their work. Shortly after the August 13 meeting plans were made for the team's process expert, Dr. Ben Mejabi, to pursue further technical interchange with DOME. Unexpected events (see next paragraph) cut this short.

The final alteration in the direction of **RAPTR** was a cut in the project budget, affecting the second half of Phase III. Phase III., which had originally been projected to last twelve months, was shortened to six, with a commensurate reduction in the Phase III budget. At the time the team was informed of this, all major content, knowledge, and architectural decisions had been made; the team concluded that descoping could only be on the ATD, not on the basic **RAPTR** functionality. (This same budget cut, incidentally, eliminated the promising PIMA function from the DOME project.)

5.2.4 – Final Build and Integration

The Software Design Description was submitted on December 16, 1997. From this point onward the team began tracking development using the functional list shown in Exhibit 5.8. At the time of the submission of the SDD, six of these functions had already been built and tested. Wizdom Systems, Inc., prepared a build schedule with release dates for each of the remaining 39 functions. This schedule was reviewed on a weekly basis by the entire team. An acceptance procedure was established, with review by Les Sanders, Stan Przybylinski, and Ben Mejabi, to certify that individual **RAPTR** components had been delivered and integrated.

During this time the team began looking ahead to the technology demonstration project at Warner Robins. Initially the team expected that **RAPTR** would be supporting an Activity-Based Costing Pilot project, which was scheduled to begin on February 24. Although it appeared that sufficient functionality would be in hand to make this a useful prototype test, in early February the team was redirected away from the ABC pilot, toward one of the directorate-level ABC projects at Warner Robins. Accordingly, the team began gearing up to support one of these, by collecting project detail from Warner Robins, and sending two team members to ABC classes.

During this period Captain Barlow drafted a demonstration plan/agreement for the participating parties; subsequently Batteau created a Test and Evaluation Plan, which Captain Barlow incorporated into the Demonstration Plan. A videoteleconference was held on March 13, 1998, to review the plan.

On March 27, 1998, Wizdom presented an advance look at the **RAPTR** software to Captain Barlow. On April 10, approximately 1.5 weeks behind schedule, the **RAPTR** team released a working version of **RAPTR**. Although all functions were present, one interface, which tailored the detailed assessment questionnaire, was not completed; this delay was due entirely to the difficulty of completing the deductive model; when this model was completed in early May, 1998, and delivered to Wizdom, this final link was completed.

Also missing in the April 10 release were approximately 50% of the activity descriptions; as non-functional content, these had taken a back seat to completing the deductive model. Over the next six months the remaining activity descriptions were completed. By the start of the demonstration project in July, the only missing activity descriptions were in Stage IV of the deductive model, which had no impact on the demonstration project.

The Phase Review for Phase II was held on April 7, 1998, at AFRL. At this point the **RAPTR** team was good to go for fielding this new tool.

5.3 Phase III: Deployment

Deployment of the **RAPTR** software began well before Phase III, with numerous briefings and demonstrations of functionality. Deployment activities included meetings of the **RAPTR** Users Group, software demonstrations and briefings, training, the selection of a demonstration project, support of a project using **RAPTR**, software maintenance during the ATD phase, and evaluation of the results. Due to delays in completing software functionality, and head start activities like these, there was a time phase overlap between the completion of the final build and the beginning of deployment activities.

In chapter 7 of this report is given a more complete description of the field trial. In this section we describe the activities surrounding and leading up to the field trial.

The **RAPTR** Users Group (Include a table with the list of the RUG organizations) was organized by Captain Cassie Barlow, and met seven times beginning on October 22, 1997. Later meetings included demonstrations by Wizdom personnel of the emerging **RAPTR** system. Most meetings were via video teleconference, with remote access to the **RAPTR** web pages. Responses to the software in these meetings were entirely positive.

Other demonstrations and briefings included presentations at the Logistics Management Agency (Gunter AFB), the USAF Safety Center (Kirtland AFB), Oklahoma City ALC, and HQAFMC.

Efforts to establish a demonstration project for **RAPTR** began in late 1997, with discussion of using the ABC pilot at Warner Robins as a test project. Inasmuch as this project was due to begin in February, 1998, the team made preparations to support the assessment segments on the project in pencil-and-paper mode, while using the then-built components of **RAPTR** to provide project management capabilities. As it turned out, the schedule established for the ABC pilot did not permit time for resolving some of the issues surrounding the use of a web-based project management tool, so the team continued its focus on meeting the release deadline.

With the ABC pilot no longer an option, the next projects available at Warner Robins would be the ABC follow-on projects, due to begin in the summer of 1998. While waiting for designation of a specific project, attention shifted to training.

The first **RAPTR** training session was scheduled for March 30-31, 1998. When it became clear that there would not be a project starting soon after this, this training session was de-scoped to one-half day, and offered at Warner Robins. This session was essentially an orientation to **RAPTR** and a demonstration.

The next training session was scheduled for April 30-May 1, at Warner Robins. This training session was attended by four individuals: two from Ogden ALC, one from AFRL, and a computer support technician from Warner Robins ALC. In an effort to broaden the group of trained users, an additional training session was scheduled for June

16-17, and targeted toward the IPT leads from the WR product directorates. These individuals would be leading Integrated Product Teams charged with conducting an Activity-Based Costing analysis. Subsequent to this training, Mr. Jim Jones deputy director of WR-ALC/RE met with the IPT leads to discuss their experiences with **RAPTR**.

Prior to the second training session, one directorate, Electronic Combat, expressed an interest in using the assessment capabilities of **RAPTR** for evaluating change readiness. Batteau and Stan Przybylinski briefed the division heads of WR-ALC/LN on June 12; a decision on working with LN was postponed pending resolution of **RAPTR** support for an ABC project.

During this training and the next few days, there were two software/network issues that impeded deployment. The first of these, the "refresh problem", was a tendency for the software to lock up when browsed with Netscape. This problem was fixed on June 21, 1998. The second problem was an unacceptable level of net lag, due to the fact that the work was being done on a server located in Naperville, Illinois. On June 24 Wizdom Systems, Inc., transferred the **RAPTR** server to Warner Robins, and installed it at RE; this subsequently fixed the second problem. However these two problems, occurring early in the initial user exposure to **RAPTR**, colored further user acceptance of the tool.

In discussions with the IPT leads, particularly those who had ABC projects about to start, concerns arose over posting ABC data to the **RAPTR** server. Batteau met with the five IPT leads, plus Jim Jones, several times during the week of June 21. On June 28, RE briefed General Goddard regarding the ABC initiative. On June 29 Barlow, Batteau, and Alexia met with the directors of the five product directorates scheduled to begin ABC projects. At this meeting similar reservations were expressed, and a decision was made that **RAPTR** should support a different project.

The project selected for **RAPTR** support was the IMPAC card expansion project, which was scheduled to begin on July 7, 1998.

RAPTR support for the IMPAC project is described in chapter seven. This project began on July 21 (after a two week delay), and as of this writing is still ongoing. The **RAPTR** project concluded on September 30, 1998, and **RAPTR** personnel withdrew from IMPAC support. (Support for the IMPAC project from Wizdom and ERIM continued under a separate contract vehicle.)

Following the termination of support for the IMPAC project, team personnel conducted a series of interviews and focus groups with IMPAC team members to get their evaluations of **RAPTR**. These are reported in chapter 8. The PI attended a final IMPAC team meeting, via VTC, on October 16, 1998.

The final event of the **RAPTR** project was the final briefing on October 30, 1998, at AFRL. At this briefing the history of the project was reviewed, a demonstration of the software was provided, the successes and failures of the project were candidly evaluated, and ongoing maintenance requirements and options were discussed.

The history of the **RAPTR** development provides a model for developing breakthrough software in a collaborative environment. Some of the lessons learned along the way are described in chapter nine. Some of the useful tools have been highlighted in this chapter as exhibits. This chapter has attempted to describe the opportunities, both seized and lost, as the **RAPTR** tool emerged from a vision into working software.

Chapter 6: Description of the Tool

The purpose of this chapter is to describe the functionality of the **RAPTR** tool. With this description, we hope to address all of the sections and capabilities of the **RAPTR** software. **RAPTR** is an enterprise-wide knowledge management tool. **RAPTR** provides:

- Information Retrieval;
- Enterprise Repository;
- Visualization & Navigation;
- Collaboration & Workflow;
- Project Management; and
- Extraction & Authoring.

All of this functionality is accessible through the World Wide Web. Each section in this description corresponds, for the most part, with the gray buttons found along the left side of the **RAPTR** screen. The screens are shown here for illustrative purposes only: further detail on screen content and usage can be obtained through **RAPTR** training available from Wayne State University.

RAPTR is for organizations that must do projects with limited resources, and across different time zones. It automatically develops a complete project plan and designs a work break down schedule for project managers and teams to follow, from assigning tasks to team members to tracking them to their completion. And, it functions as a repository for all information pertaining to the project. That information could be raw data, models, and any documents anywhere. All of this can be done from remote locations utilizing the web or operating on a local server.

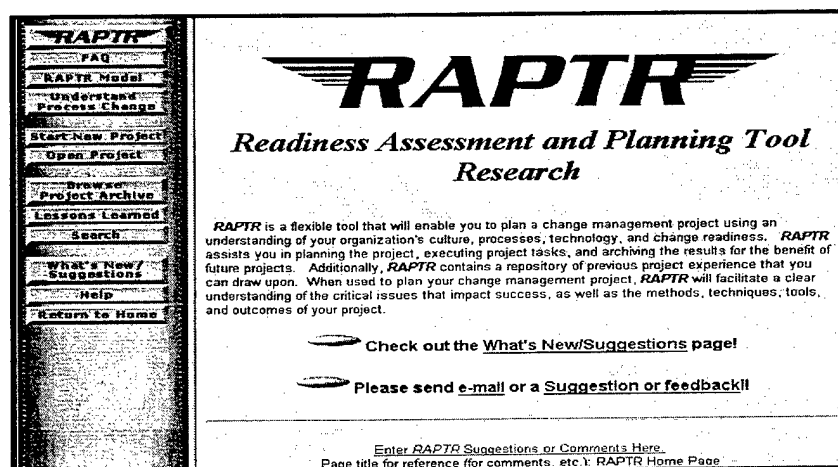
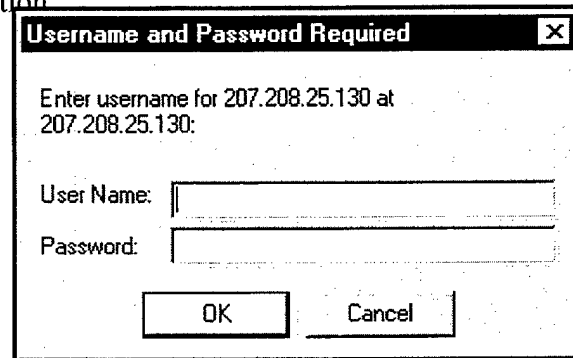


Exhibit 6.1 -- The **RAPTR** Main Screen

6.1 Getting Started

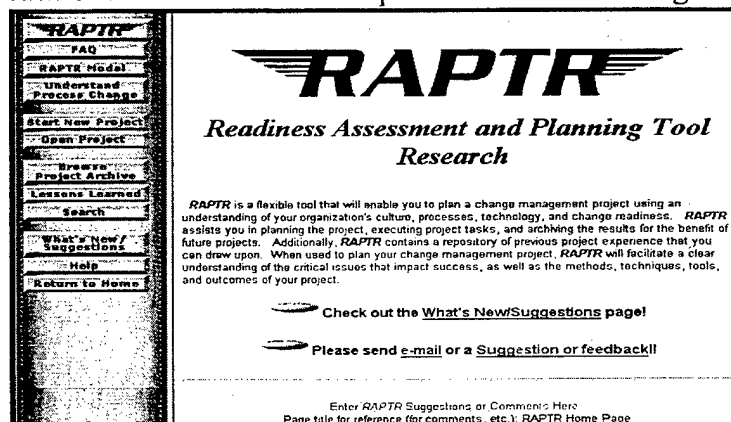
Before doing anything the user must first enter their name and password. The dialog box for entering this information is shown in Exhibit 6.2. The System Administrator will provide this information.



A Windows-style dialog box titled "Username and Password Required" with a close button (X) in the top right corner. The text inside reads: "Enter username for 207.208.25.130 at 207.208.25.130:". Below this, there are two input fields: "User Name:" and "Password:". At the bottom, there are two buttons: "OK" and "Cancel".

Exhibit 6.2 -- The *RAPTR* Logon Dialog Box

The first screen appears upon connecting to the *RAPTR* server is shown in Exhibit 6.3. It does not matter whether the site is Web based or on a local server the screens will appear the same. The user may, through a series screens, choose any number of ways to navigate through *RAPTR* manage one or several projects, and check-out information. The first screen will enable the user to launch into different areas of their projects. The meaning behind each of the buttons will be explained in the following sections.



The RAPTR Home Page. On the left is a vertical navigation menu with the following items: RAPTR, FAQ, RAPTR Model, Understand Process Change, Start New Project, Open Project, Browse Project Archive, Lessons Learned, Search, What's New / Suggestions, Help, and Return to Home. The main content area features the RAPTR logo (stylized with wings) and the text "Readiness Assessment and Planning Tool Research". Below this is a paragraph describing RAPTR as a flexible tool for planning change management projects. At the bottom, there are two links: "Check out the [What's New/Suggestions](#) page!" and "Please send e-mail or a [Suggestion](#) or [feedback](#)!!". At the very bottom, small text reads: "Enter RAPTR Suggestions or Comments Here Page title for reference for comments, etc.): RAPTR Home Page".

Exhibit 6.3 – The *RAPTR* Main Screen

6.2 Frequently Asked Questions (FAQ)

When the *FAQ* button is selected a complete description from A to Z of Business Process Re-engineering (BPR) can be found. Answers are given to the following questions.

Who developed *RAPTR*?
Why was *RAPTR* developed?
How will *RAPTR* help me and my organization?
Why do I need *RAPTR*?
What is the *RAPTR* Supermethodology?
Must I use the complete *RAPTR* Supermethodology?
How do I use *RAPTR*?
What is BPR?
Where Can I learn More About BPR?
Does *RAPTR* do BPR?
What areas do the *RAPTR* assessments measure?
What are the System requirements for *RAPTR*?
How can I be trained on how to use *RAPTR*?

6.3 Understand Process Change

The screen below (Exhibit 6.4) appears upon selection of the button marked *Understand Process Change*. If the highlighted text marked “Outside Resources” is clicked it will bring up the next screen “BPR-Related Sites of Interest” which is linked to many other sites that will aid in the support of their project. These are sites that were identified by ERIM as providing useful examples and documents pertinent to change management and reengineering.

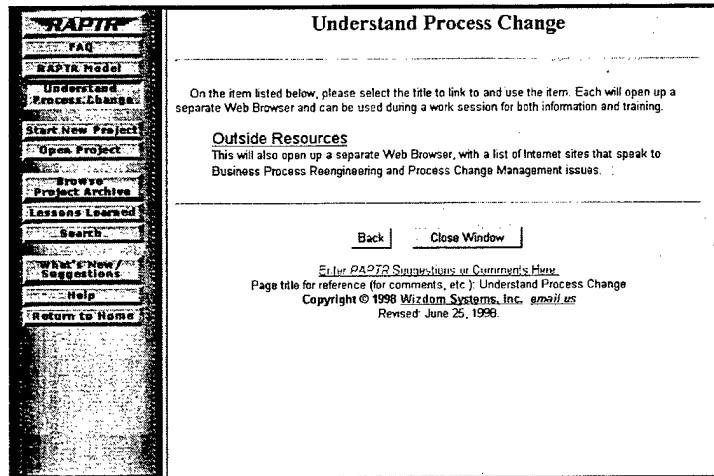


Exhibit 6.4 -- Understand Process Change

Many times, the user wants to understand what actually happens when an enterprise is reinvented: what the critical success factors are, which organizational resources are brought to bear, how those resources are organized for a reengineering effort, what activities are completed—and how, by whom, and why. Any outside information is useful. The added resources linked to this portion will aid in the user's endeavors.

Through the linked sites many of the questions the user may have can be answered.

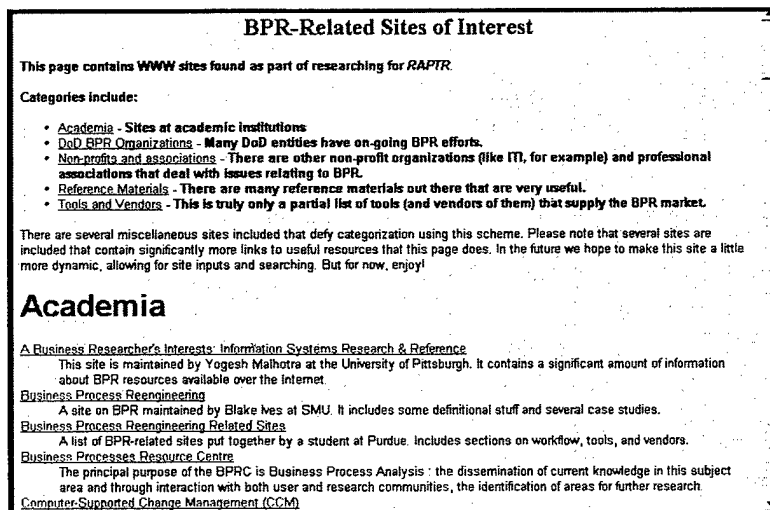


Exhibit 6.5 -- Outside Resources

6.4 Start New Area – Project Manager

RAPTR incorporates a structured process for starting and managing a project. Some segments of this process are dictated by the requirements of initializing project databases. Some segments of the start process are intended to generate project meta-data including the name of the project manager and team members, contact information (phone, fax, e-mail), security information (project password), and other administrative information. One other part of starting a project in **RAPTR** addresses the fundamental question that a project manager should ask, “Do we understand what this project is about, do we have sufficient resources to do it, and is my team good to go?” This final part, the Team Self-Assessment, is described in section 6.6.

When clicking on the “Start New Area”, all of the information needed for a new project must be entered. This information includes:

- Top of Form 1
- Project/Area ID Names (e.g.: AirForceBPR)
- Project/Area Organization:
- Project/Area Start Date:
- Project/Area Manager;
- Project/Area Manager’s Phone;
- Project/Area Manager’s FAX number;
- Project/Area Manager’s E-mail;
- Project/Area Manager’s Office Symbol;
- Project/Area Description;
- Number of Personnel (at target of the project);
- General Password for the Project;
- Type of Project.

RAPTR uses this information to set up the project area, and to begin the process of tailoring a project plan to support the project.

6.5 Open Area

For the general user of the site, most of the **RAPTR** project information may be found by clicking the “Open Area” button to the left of the screen; along the gray panel. The next screen, “Area or Project Work”, opens the drop down window which is labeled (Step 1) “Select a Project”. The user selects the current project that they are working on as the project. The next window labeled (Step 2) “Select Type of Work”, defaults to “Browse Project”. The options as shown in Exhibit 6.6 are:

The following are examples of the screens commonly seen.



When the user chooses the “Project Management” option, the screen in Exhibit 6.7 appears. This screen provides an outline for the tasks to be performed in the management of a **RAPTR** project.

Please select one of the following areas of work:

To begin or redo the initial start of the project & team assessment, select:

☒ There are team members. ☒ Team surveys started. ☒ A Team analysis has been done.

Edit the current Project Team?

Edit the current Project Meta Data?

To configure *WITHOUT* Assessments, select the following button. DO NOT USE if doing assessments OR project is already configured!

To set up the High Level Assessment (HLA), select:

☒ HLA surveys started. ☒ A HLA analysis has been done.

Analysis of the HLA and to generate the change management plan for the project, select:

☒ A HLA analysis has been done. ☒ A project plan has been generated.

Exhibit 6.7 -- The Project Management Screen

After setting up the team information, the project manager normally administers a Team Self Assessment. **RAPTR** will automatically configure an area on the server for storing the assessment, and the manager can e-mail the Universal Resource Locator (URL) to team members. Team members need only to click on this link, and they are taken to the screen shown in Exhibit 6.8.

Team Self Assessment

RAPTR

This easy-to-use questionnaire is designed to conduct an assessment for this project, to identify cultural and organizational issues involved in implementing a project, and to recommend strategies for managing these issues.

The questionnaire will take less than 25 minutes to complete. Answer the questions as accurately as possible, following the directions for each one. Your responses are anonymous and will be compiled with others' by the expert system to recommend appropriate actions. Thank you for participating in this survey!

Wizdom
Copyright © 1998 Wizdom Systems, Inc.

Exhibit 6.8 --The Team Self Assessment Start Screen

Which of the following best describes the goals of this project?

- ☐ We are seeking to make our organization more agile, so that it can more quickly respond to changing customer demand and other environmental circumstances.
- ☐ We are seeking to make our organization more productive, cutting costs and gaining greater operating efficiencies.
- ☐ We are seeking to make our organization flatter, removing layers of middle management and improving communication from top to bottom.
- ☐ We are seeking to improve cycle time, reducing the amount of time it takes us to deliver our products or our services.
- ☐ We are seeking to rightsize our organization, eliminating redundant or non-value-added functions.
- ☐ We are refocusing our organization on its core competencies, eliminating peripheral lines of business.
- ☐ We have been mandated to privatize all or significant elements of our operation.
- ☐ I don't know what the goals of this project are.

[Next](#)
[Exit](#)

Wizdom

Exhibit 6.9 -- The Team Self Assessment Question Screen

The **RAPTR** assessment software then asks questions like that in Exhibit 6.9.

Once enough team members have completed this assessment, the project manager can ask **RAPTR** to analyze the results. When this happens, **RAPTR** returns a screen like that in Exhibit 6.10. This screen will inform the manager of the readiness of the team: green if ready, *yellow* if almost ready, and *red* if not ready.

Briefing3 Team Advice from RAPTR


 Green Light!! You are ready!

Based on what you have told us about your team and your project, thus far **RAPTR** has detected no obstacles to your proceeding with the project. You appear to have adequate sponsorship, an appropriate scope, and the team is prepared and understands the objectives.

Please note that this conclusion is based solely on the team's input. Your next step is to proceed to a broader assessment of issues in the wider organization; this conclusion may be modified based on that wider assessment.

Skill Deficiencies

The team's skills do not seem to be matched to the project requirements. We suggest you review this issue before proceeding further, and perhaps acquire some additional training.

[Return to Briefing3 Work Page](#)

[Enter RAPTR Suggestions or Comments Here](#)
 Page title for reference (for comments, etc.) Team Advice
 Copyright © 1998 Wizdom Systems, Inc. [powered by](#)
 Revised: May 1, 1998

Exhibit 6.10 --The TSA Analysis Screen

After the Team Self Assessment is completed, the manager configures the High Level Analysis. This is similar to the Team Self Assessment, except that it is distributed to the knowledge workers as well as the team members. Once completed, the screen shown in Exhibit 6.11 shows **RAPTR's** assessment of the readiness of the organization with respect to a number of variables. Notice the red-flagged issue, indicating an issue that jeopardizes the success of a change effort.

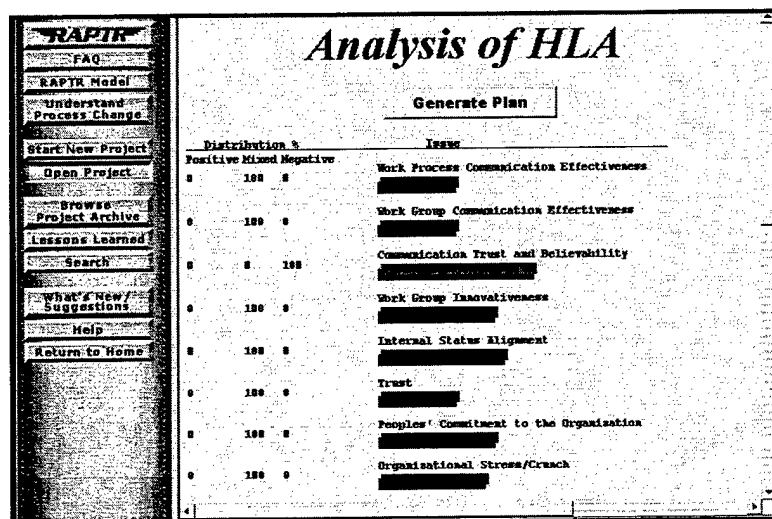


Exhibit 6.11 -- Analysis of the High-Level Assessment

With this information, the project manager can enter the *Generate Plan* activity, where **RAPTR** recommends tasks that are deemed necessary to successfully complete the change, including tasks that address the red-flagged issues. Results are presented in a screen like that in Exhibit 6.12.

OK	Task Description	CF
☒	PLAN CHANGE MANAGEMENT	1314
☒	Conduct Strategic Assessment	15
☒	Kickoff Stage 1	3
☒	Develop Design Philosophy and Project Vision	1
☒	Select the Executive Committee and Project Team	1
☒	Train Executive Committee and Project Team	1
☒	Conduct Business Overview	1
☒	Assess Business Goals and Objectives	1
☒	Identify Opportunity Areas	3
☐	Conduct Environmental Scan	1
☒	Determine Project Goals and Outcomes	1
☐	Develop Executive Approval	1
☒	Determine Project Scope	7
☒	Determine Project Boundaries or Sizing	1
☐	Identify Project Champion	1
☐	Identify Stakeholders	1
☐	Determine Resource Needs	1

Exhibit 6.12 -- Generating a Plan

The next step for the project manager is to assign scoping information to the selected tasks. **RAPTR** presents the screen shown in Exhibit 6.13 to recommend the number of days that should be allotted for each task.

Exhibit 6.13 -- Scope Plan Screen

After this step, **RAPTR** configures a project plan file, in MS Project format, that reflects the information agreed upon by the project manager. The manager can now utilize this file by using the screen in Exhibit 6.14 to download it

Task Description	CF	Duration
PLAN CHANGE MANAGEMENT	96.42	14
Conduct Strategic Assessment	7	7
Kickoff Stage I	3	3
Develop Design Philosophy and Project Vision	1	1
Select the Executive Committee and Project Team	1	1
Train Executive Committee and Project Team	1	1
Conduct Business Overview	1	1
Assess Business Goals and Objectives	1	1
Identify Opportunity Areas	1	1
Determine Project Goals and Outcomes	1	1
Determine Project Scope	1	1
Determine Project Boundaries or Sizing	1	1
Conduct AS-IS Assessment	12	1
Assess Internal Company or Agency Characteristics	12	1
Create TO-BE Design	10	1

Exhibit 6.14 -- The Task Management Screen

Other management functions supported by *RAPTR* are:

- Task Assignment (via MS Project)
- Task Status (via MS Project)
- Detailed Assessment (drill-down in red-flagged issues)

6.7 Open for Work

When the user chooses the “Open for Work” option, the screen in Exhibit 6.15 appears. This screen provides access to two functions. First, the user can select the “Check Tasks” button. This will take them to Microsoft Project for Workgroups, where they can check messages for task assignments or status updates. This screen is pictured in Exhibit 6.16.

**Designer's Notebook
for DLA Activities**

Before working on one of the Activities/Tasks listed below, be sure and check the *Project Tasks Mail Box* by pressing the button:

Download the *Wizdom Works Process Model Viewer* for work with process models.
To Review the DLA Plan in Microsoft Project 98, Select: *Review Plan in MSPProject*

Click on a Node to work on a Task.

Task Description	CF	D
PLAN CHANGE MANAGEMENT	209	92
Conduct Strategic Assessment	105	30
Kickoff Stage I	33	6
Develop Design Philosophy and Project Vision	10	2

Annotations:
A: Criticality Factor
B: Duration

Exhibit 6.15 -- Project Repository Screen

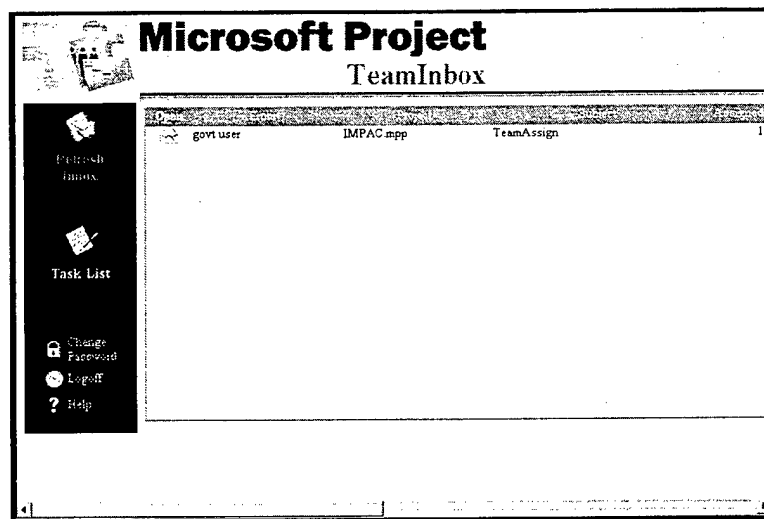


Exhibit 6.16 -- Microsoft Project for Workgroups

The screen in Exhibit 6.15 also provides access to the repository areas for each task. These areas provide access-controlled storage and retrieval of task-specific documents and artifacts. The area seen in the lower portion will have list the tasks and provide links to pages that present the task repository area. Additional fields in the indented list are the columns to the right of the list labeled A and B. The *Criticality Factor* (CF) is the value determined by the software and assigned and relates to the level of importance of that particular level within the project. The higher that number the more important that level is within the project. *Duration* (D) is also determined by the software and represents the anticipated project duration in days.

When a level of the indented list is selected, such as "Conduct Strategic Assessment", the following screen will be seen.

Activity Task

Conduct Strategic Assessment

- For detailed descriptions, advice, and instructions concerning this particular task/activity please select: [Activity Instructions](#)
- For general references select: [Change Management references](#)
- To search for previous lessons on this activity select: [Lessons Learned](#)
- The above links all open in separate windows for reference during Activity work.

In the table listed below are all the documents, files, and templates for this Task/Activity. For *Checking Out* items, go to the table and enter your name first and then select in the check boxes on the right side, the items you want. You may enter or change a description of the document. You can also just *Browse* which does not check out the file. When done with a document or if you need to add a file, follow the next instructions for *Uploading and/or Checking In* documents.

For adding and checking in files, select

There are NO documents available for this task.

[Return To Task Listing](#) [Go to SuccessfulProject Home Page](#) [Go to User's Home Page](#)
[Enter Questions or Comments Here](#)
 Page title for reference (for comments, etc.): Task Chores
 Copyright © 1998 Wisdom Systems, Inc. [email us](#)
 Revised: August 2, 1998

Exhibit 6.17 -- Task Repository Area

There are three areas of the screen. The first has links to other areas.

- (a) *Activity Instructions* is linked back to the Task Activity Description & Instructions. Which are descriptions of who, what, when, and how to do each level of the tasking. The next link
- (b) *Change Management* Instructions will take the user to BPR – related site and interest. The last link is to
- (c) *Lessons Learned* is an area where information gathered through experience is stored.

6.8 Uploading Information

The next area is for adding and checking files in.

Upload Files for SuccessfulProject

A1: Conduct Strategic Assessment

To Upload Project Documents (Word, Excel, etc.) and files of any type, please fill in the file upload form below.

*Please enter a description for each file. If a document is complete and being checked in for the last time for this task, please check the complete box. After selecting files, press the **Upload** button near the bottom of the page. No wildcards (*) are accepted. You may enter up to four files on this form. After selecting the **Upload** button, the next page will require you to **Check in** the files.*

The document/file to be uploaded:

Please enter a description of the file:

Please check here if the document is complete: ☐

The document/file to be uploaded:

Please enter a description of the file:

Exhibit 6.18 -- Repository Upload Screen

When the user selects *Check In Files* they will see the screen shown in Exhibit 6.18. This screen allows team members to upload anything saved as a file. The user selects the *Browse* button to locate the file. The user may enter up to four files on this form. The file will be transfer to the upload section. The user should add a short description of the file. This description will appear of the bottom of the *Activity Task* screen and will be easier for others to understand exactly what the file is. After the user has named the files, they can move to the bottom portion of the screen and select *Upload* in order to begin the upload process.

The **RAPTR** Repository function provides a document management capability with version maintenance and logging. When documents are checked in, if they have been altered they are checked in as a subsequent version of the original document. Documents can be downloaded for read-only (browsing), or for purposes of revision.

6.9 Browsing

Browsing through the project will allow a user to view the documents without the ability to modify any part of the documents. This “read-only” access will allow users to copy files, and to inspect file contents.

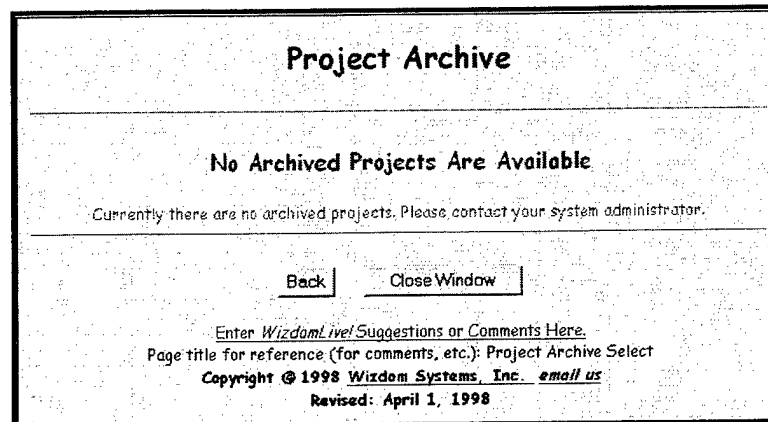


Exhibit 6.19 -- The Browse Area Archive Screen

6.10 Browse Area Archive

This area contains any archived information, e.g., Activity Models, Data Models, or Process Flow models. The stored information would be listed in the space provided and check-out and checked-in when needed. The advantage to this archive area is that if information has been stored relevant to the project that the user are working on currently could be within drawn examined and if applicable could be directly applied saving valuable time and resources.

6.11 Lessons Learned

Lessons Learned, like any history, is best written down so that it is not repeated. The *Lessons Learned* section can be a valuable resource for saving time. Exhibit 6.20 illustrates the Lessons Learned Screen.

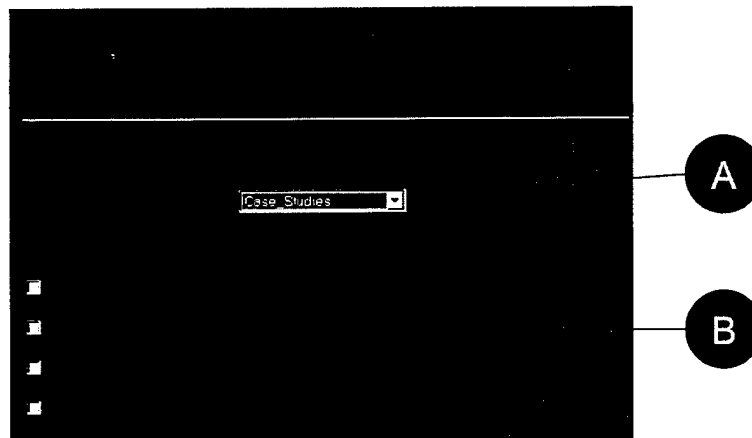


Exhibit 6.20 -- Lessons Learned Screen

First, from the (a) drop down window, the user selects the project on which they are currently working. Move to the (b) topic areas below; select that category that best coincides with the information. Move to the lower section of the screen (c) and enter the information in the window provided. It may help to type the text in a word processing program, such as Word and then cut and past into the area provided.

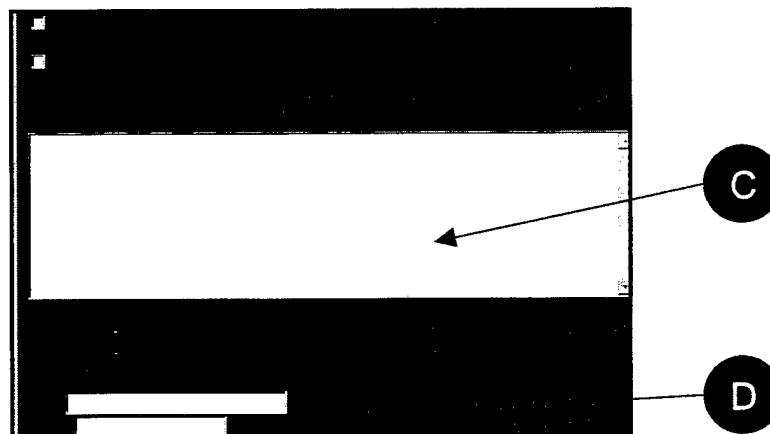


Exhibit 6.21 -- Lessons Learned Entry Screen

Proceed to the next step (d) and enter you name, office, and fax number in the appropriate boxes.

6.12 Search

The *Search* engine is based on the Internet product developed by Digital Equipment Corporation called AltaVista. This search function can be used like any other search engine to perform search functions and work just as all search engine operate.

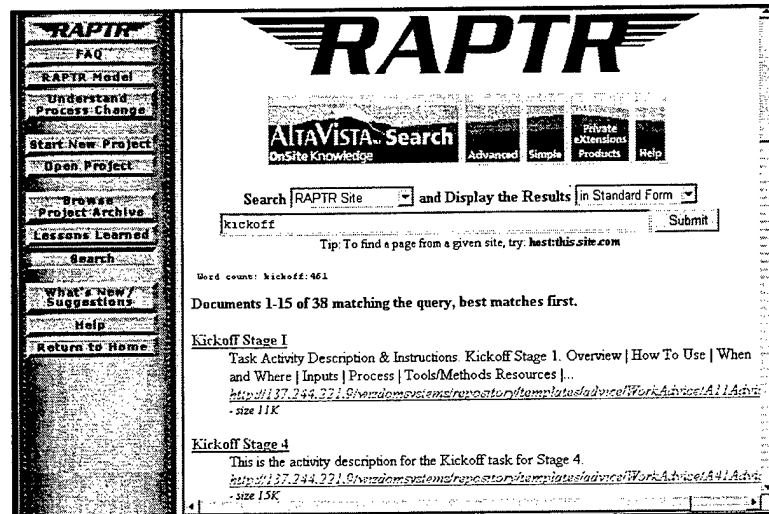


Exhibit 6.22 -- RAPTR Search Screen

6.13 What's New/Suggestions

The *What's New/Suggestions*, contains an area has a list of the major changes in *RAPTR* with the most recent changes shown first. If possible, a *hyperlink* will be available to go to the new change or new instructions. For suggestions, it allows you to submit suggestions for changes that you feel may helpful straight to the technical developers.

6.14 Help

As with any help section, the user may ask for assistance directly to the group that can best help the user when they find that they are having a problem with a particular area *RAPTR*.

6.15 Return Home

This button does just what it says, it will return the user to the home page of *RAPTR*.

Chapter 7: *RAPTR* Field Trials and Field Evaluation

During Phase III of the *RAPTR* project, there were two projects on which *RAPTR* was used to provide project planning and execution support. On one of these projects, undertaken by Warner Robins ALC, *RAPTR* team personnel provided on-site support; the other, undertaken by the Air Mobility Command, was a spontaneous effort that found *RAPTR* useful in executing a new methodology, Action Work Out (AWO). Analysis of both of these suggests both some opportunities for further development of *RAPTR*, and some requirements for its successful use.

7.1 Selecting Pilot Sites

7.1.1 Criteria for pilot sites

In choosing a pilot project to demonstrate *RAPTR*, we had several specific criteria in mind. We wanted a project that would:

- be broad enough to exercise as many of *RAPTR*'s capabilities as possible,
- have strong support across all levels,
- have a timeline that would allow users to be familiarized with the tool before actual use,
- be staffed by people who understand and embrace the basic tenets of teamwork
- be scheduled so as to coincide with the test period called for in the *RAPTR* development plan
- allow us access to the meetings and documents necessary to begin populating the database and,
- would involve significant changes to a process, and or technology.

7.1.2 ABC

The Reengineering (RE) office at Warner Robins described to us an upcoming Activity Based Costing (ABC) project that sounded like a good match for our needs. After further discussion with RE, we decided to approach the ABC Integrated Product Teams to determine their interest in participating. On the surface, this looked like a good test. It was a Center-wide program that worked to a common agreed-upon plan. The stated intent was to gather and share information across product directorates from the ABC deployment.

We spent some time with RE staff to get up to speed on their work and the ABC tools they employed. We attended several IPT meetings and ABC modeling sessions,

facilitated by an ABC contractor. We met with the contractor to discuss our work and how we might work together. Using a Microsoft Project file that they supplied, we structured a Designer's Notebook around their existing project plan and began to populate it with artifacts collected from RE, e.g., meeting notices, meeting minutes, reports, etc. While the stated intent was to share information, the ABC tools employed (EasyABC) did not allow model information to be readily shared (later in the effort a network version of the tool was to be acquired that would facilitate information sharing). As a test case, a **RAPTR** team member installed the ABC tool and used graphics tools and an HTML editor to create a Web version of several of the team's models. We were not, however, allowed to incorporate actual numbers in these models. Even though the cost and resource figures were not included, we believed that sharing the structure and contents of the models would be useful to the different ABC teams.

We conducted information meetings with several IPT members, showing our work to date in making their ABC information available in **RAPTR**. While there was some interest, RE left the final decision on whether to participate in the **RAPTR** pilot to a vote of the IPT. In the end, the IPT decided not to work with the **RAPTR** team. We believe there are several reasons for this decision:

A common reason to not participate in a pilot is that it will result in "extra" work, above and beyond the tasks to be supported by the new technology. As often happens at WR-ALC, people were expected to conduct this ABC effort while maintaining their previous level of throughput. While we tried to illustrate that the benefits of using **RAPTR** would exceed these perceived costs, we could not persuade them. There was reluctance to take the time to learn a new tool.

A more important reason here was the issue of information sharing. According to the vision and mission created for this effort, ABC information was to be readily shared across product directorates (PDs). In some respect the sharing was left to the contractor who facilitated all of the sessions and applied their knowledge of all of the models across the PDs. However, while the PDs would share the hierarchical structure of their models across PDs, they would not share any quantitative data. Even though the goal was to model and learn across PDs, IPT members saw resources as power and their acquisition as a zero sum game. If they are performing some particular function more efficiently than someone else is, they may lose resources. If they are doing worse, attention will be called to their shortfall. Managing information sharing issues of this sort is a command issue.

Another issue was that **RAPTR** was still in development. As the developers, we expected that there would be some bugs in the system, and were not discouraged to discover them. However, as potential users, the ABC group was not willing to expend effort on something that was still in its Beta stage. Learning a new system would be bad enough, spending time on something that did not always perform as they expected was too much. Responsibility for this issue rested, of course, with the development team.

7.1.3 IMPAC

When it became clear that ABC was not going to be feasible as a test case, RE invited the team to pilot **RAPTR** on another new project. This project was to develop the new processes necessary to move the AF to using a credit card for purchases under \$2500, thus avoiding the cumbersome paper intensive purchasing process. The project, named IMPAC, would be run out of the RE office at WR with a WR/RE person as lead.

The IMPAC project was led by the project manager in RE and was comprised of team-members from other ALCs as well as HQ/AFMC. In this sense, **RAPTR** was ideal, since one of its primary functions is to facilitate distributed workgroups. The field trial began on July 23rd and went through September 30th, **RAPTR**'s original end date.

From an overall test perspective, IMPAC was not ideal since it clearly would not allow for exercising many of **RAPTR**'s functions. That is, it was not a true reengineering effort. However, there were some strong reasons to proceed.

- It did fit into the timeframe we needed.
- The team seemed amenable to the testing.
- The team was distributed.
- The team lead was already familiar with the **RAPTR** effort.
- It was a new project, so we could start from day one.

7.2 Field Trial at Warner Robins ALC

The Reengineering Directorate at Warner Robins Air Logistics Center (WR-ALC/RE) was the primary customer for the **RAPTR** effort. . As the development team worked with RE to understand their change management methods and tools, there was the objective that one of the change management projects at RE would provide a field trial for the beta version of **RAPTR**.

Due to several external circumstances, the field trial that began in late July of 1998, 70 days before the end of the **RAPTR** project, can only be considered a limited success. **RAPTR** displayed the full range of its functionality, and provided positive benefit for the team using it. However, the project did not make full use of the assessment capabilities, or integrated project planning and management capabilities of **RAPTR**. IMPAC use of **RAPTR** was limited to document management. Some of the reasons for this will be discussed at the end of this chapter.

7.2.1 Project context, selection and preparation

The project selected for the field trial was the IMPAC card expansion project. The IMPAC card (International Merchant Purchase Authorization Card) is a debit card that prior to the project could be used for purchases of and payments on commercial items valued under \$2500. The project was chartered to expand this usage into other types of items and payment processes. The immediate context for this project was the Reengineering Directorate at Warner Robins Air Logistics Center (WR-ALC/RE).

The IMPAC card is a debit card that is used for making multiple types of purchases, typically from local vendors. An individual who has IMPAC purchasing authority can purchase small items without going through base supply or requisitioning processes; for small items the IMPAC card eliminates the purchasing and item management roles.

An Air Logistics Center is a large maintenance, repair, and materials management facility. Within the Air Force Materiel Command there are three ALCs: at Ogden UT, Oklahoma City OK, and Warner Robins, GA. The major systems supported by Warner Robins include the F15, the F16, and the C130.

Warner Robins ALC has undertaken an aggressive improvement strategy to maintain and enhance their position as a premier logistics facility: the leadership of the center has recognized that in an era of downsizing and base closures, depots and repair centers are in a competitive environment where they must be able to offer their services faster, cheaper, and with better quality than other centers or private alternatives. They have identified their objectives as 50-30-20, meaning a 50% reduction in cycle time for repairs, a 30% reduction in costs, and a 20% reduction in personnel through retirement and attrition.

This aggressive posture led the center to create a directorate-level reengineering function (WR-ALC/RE). This directorate brought together expertise from across the base to provide change leadership. In 1995, when approached by the **RAPTR** team, RE eagerly agreed to be the **RAPTR** customer. Through the development period of **RAPTR**, in 1996 and 1997, the leadership of RE, had good visibility on **RAPTR** and its evolving capabilities. In April of 1997 when it came time to launch the field trial, they were fully supportive and took every measure possible to assure a successful field trial, within the limits of the timetables of the directorate and the **RAPTR** project. **RAPTR** had been briefed at a command level in the center, so center support for the field trial was also assured.

Two training sessions were held at Warner Robins in the course of Phase III. The first, on April 30-May 1, 1997, was attended by only four individuals. A second session was held for the ABC project team leaders on June 16-17, 1997.

This second training session provided the first user-level exposure to **RAPTR** at Warner Robins outside the RE directorate. There was a high level of concern among the users over storing financial performance information on a server: The type of data that the ABC project collected were critical to the competitive posture of the different product directorates, and there was concern that these data might fall into the wrong hands. The **RAPTR** team briefed the directors of the product directorates on June 29; subsequent to this briefing, the directors decided that the center should find a project other than ABC for **RAPTR** to support.

The next project available was the IMPAC card expansion, which was scheduled to start the following week. (At the last minute, the start date was postponed two weeks.) With the clock running out on Phase III, the team quickly agreed .

There were three results of this decision that, in retrospect, made the IMPAC project less than ideal as a field trial. In the first regard, the project manager and the **RAPTR** team both learned that **RAPTR** would be supporting IMPAC less than three weeks before the project started. There was no opportunity for orientation and review.

Second, no members of the IMPAC project team were trained on **RAPTR**. Subsequently Wayne State graduate students, supported by Wizdom engineers, made the rounds of team members in Warner Robins, Dayton, and Washington DC, to explain how to use the browser, and explain basic **RAPTR** functionality. These sessions typically lasted less than two hours, and covered browsers, basic concepts, and how to use the repository. However, they provided no substitute for the authority that would have been conferred by **RAPTR's** standard, two-day classroom training session. As such, many members of the team saw use of **RAPTR** as optional, a conclusion that severely undermines the effectiveness of a collaboration tool such as **RAPTR**.

(An additional result of these support sessions at the user's workstation was the discovery that most members of the IMPAC team did not have web browsers installed on their computers. Since the Air Force regulates what software can be installed on desktop computers, including which versions of web browsers, additional delay was encountered in getting the users ready to use **RAPTR**. As a result, on the IMPAC team, use of **RAPTR** tended to concentrate among those who were proficient in terms of their PC configurations and personal skills.)

Third, as IMPAC unfolded, it revealed a basic limitation of the tool: **RAPTR** was designed for organizational change, whereas IMPAC was a process that spanned multiple organizational boundaries. This challenge is an increasingly common one in change management. The IMPAC project was an effort to expand use of the IMPAC card, initially to use it to replace small contracts. This would require new procedures for ordering and receiving materials, and for approving invoices and payments. Although these activities span multiple components – potentially including contracting (PK), financial management (FM), materials management and logistics (LG), as well as the line components that use the material, none of these functions or components per se was being changed: only the ordering and payment process.

This said, the IMPAC project had several positives that did make it a useful trial for **RAPTR**. The IMPAC Expansion Team was well-motivated, and very much focussed on fulfilling its mission. The project manager was an effective team leader, and supportive of **RAPTR**. RE leadership assured that **RAPTR** and the team were adequately supported during the field trial.

The IMPAC team consisted of approximately 25 individuals representing four functions, five locations, and multiple levels of USAF command. Although the majority were from Warner Robins, six team members were from HQAFMC, two from Secretary of the Air Force, two from Defense Finance and Accounting System (in Denver, CO), and two from the private sector (representing the contracting community and the bank that processes IMPAC payments). This provided an excellent, robust test for **RAPTR**'s ability to integrate multiple locations.

The IMPAC project began on July 21, 1998. An initial, three-day meeting was held to provide team orientation, define the project scope and vision, and identify any other team requirements. During this initial meeting a project plan was developed, the tasks of which are listed here:

Kickoff

- Define project scope/vision
- Define roles and responsibilities
- Validate team and identify additional team members
- Review budget needs
- Review/brief DFAS and SAF/IL IMPAC issues
- Identify potential impact
- Identify approving authority and cycle
- Review and validate project scope, plan, and schedule
 - Schedule additional meetings, VTC, as required
 - Determine times and attendees for approval briefings
 - Assign action items and completion dates

AS-IS Assessment

- Identify constraints: DFAS, FM, PK, etc.
- Develop process flow
 - Systems
 - Document flow
 - Critical approvals/accountability
- Identify other barriers to change
 - RAPTR** High-Level Assessment
 - RAPTR** Detailed Assessment

TO-BE Design

- Review/Validate TO-BE Vision and brainstorm implementation
- Map TO-BE data flow, document flow, approvals
- Design/Specify Final systems
 - Status of legacy systems re upgrades and changes
 - Identify/obtain funding commitment for legacy system change
 - Timeline for CSRDs/implementation
- Design/specify interim systems
 - Assess potential for workaround test/implementation

- Determine timeline and cost of interim systems
- Assess manpower needs for future process
- Create Test/Proof-of-concept methodology
 - Timeline for CSRDs/implementation
 - Identify COTS alternatives for TP-BE test
 - Success criteria
 - Metrics for evaluation
- Brief TO-BE design

Test, and Plan Implementation

- Secure approval to test: briefings, coordination reqs, approval levels, timeline
- Conduct test
 - Training
 - Interim technology insertion
 - Sign up defense contractors: bank and government
 - Monitor test
- Secure full-scale approval
- Brief technology and system changes
 - Brief/propose regulatory changes
 - Create training plan
 - Sign up defense contractors
- Present final briefing

Three comments are worth making regarding this plan. First, the plan was developed in collaboration between the IMPAC project manager and the **RAPTR** PI; the IMPAC project manager had an excellent knowledge of the substantive issues involved with the expansion of the IMPAC card, and the **RAPTR** PI had extensive project management experience. We submit that, except in hindsight, it would be difficult to improve upon this plan.

Second, in developing the plan, the **RAPTR** PI annotated it in terms of the critical attributes of each of the 47 tasks (at all levels of indenture). For tasks characterized by consensus-building we judged that face-to-face (FTF) meetings would be the preferred mechanism of coordination, whereas for tasks characterized by fact-gathering, analysis, and dissemination, **RAPTR** would be more effective than any other medium. Tasks that just required dissemination of information could be handled by e-mail, whereas periodic re-alignment of the team could be accomplished through videoteleconference (VTC). Based on this analysis, we identified 15 of the 47 tasks that would be best accomplished through **RAPTR** (these were almost exclusively in stages II and III), and 22 where FTF or VTC would be preferable.

Third, in retrospect, this ratio (22:15) should have alerted us to the fact that this project had more to do with relationship change and consensus-building, rather than process analysis and change. In the analytic stages (II and III) **RAPTR** was appropriate, but the overall tenor of the project required more intensive collaboration among far-flung team members.

In the Air Force use of the IMPAC card has been limited to small commercial purchases (under \$2500). The IMPAC expansion project was chartered to expand this use in three

areas: Base procurement (retail), Central procurement spares (wholesale), and Industrial Fund purchases.

The first major task in the IMPAC project was documenting the process flows and systems for different types of contract purchases: MSD, GSD, FMS, etc. Altogether there are eight different type of contracts that IMPAC could be used for processing payments. Each of these has a different sequence of payment activities. For example, the AS-IS of the MSD (DO-35A) process was mapped as follows:

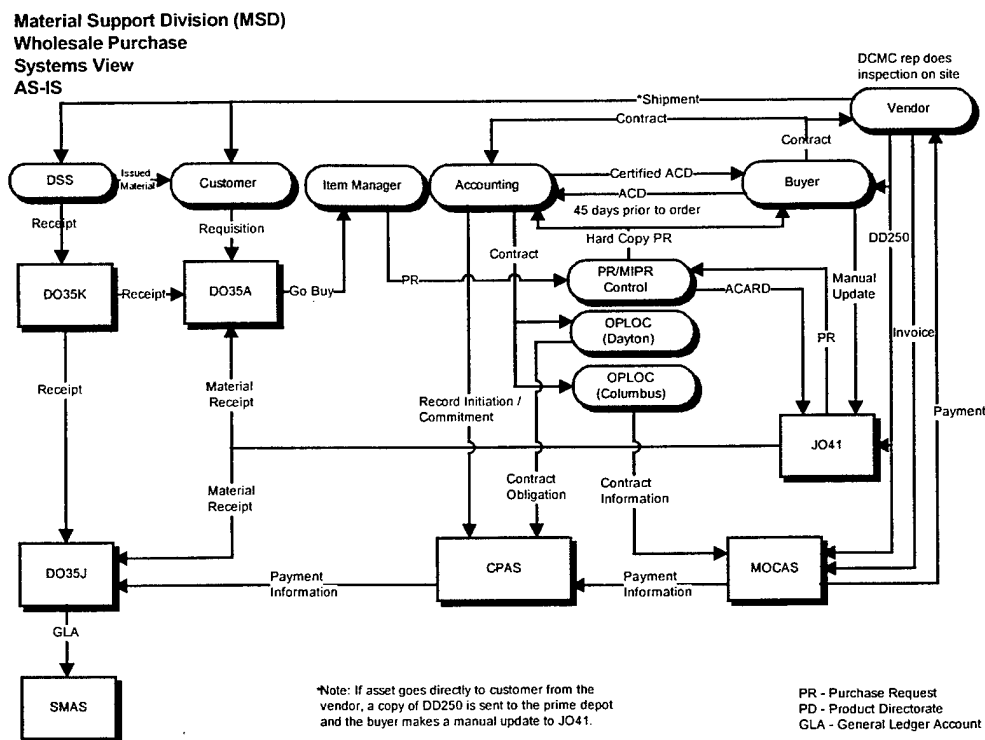


Exhibit 7.1 – MSD Process Map (AS-IS)

During the period of the **RAPTR** evaluation (which began with the project, but concluded before the project finished), there were four different types of interchanges among team members:

1. Dissemination of project documents: charter, schedule, team lists, etc.
2. Team alignment: identifying and involving team members

3. Process and system discovery and mapping
4. Solution development and validation

The second of these was the critical success factor for the IMPAC Expansion project, and the team leader devoted significant attention to this. The first is largely an administrative task, and the **RAPTR** repository to some extent was able to replace the fax machine as the mechanism of distribution. The fourth type was best accomplished in on-site meetings and videoteleconferences.

The third type of interchange provided a critical test of **RAPTR**. Initially the team developed process flow diagrams in WizdomWorks, yet the file structure of WizdomWorks, similar to that of other modeling tools such as Easy ABC, made **RAPTR** difficult for the seamless exchange of models. Difficulties in exchanging documents with complex file structures eventually led the team to use PowerPoint for all exchange of graphic documents.

This is a technical problem that the development team wrestled with for more than a year. In brief, the **RAPTR** repository is single-file oriented; as files are created, they are uploaded to the repository, from whence they can be either checked out, or browsed using a helper application on the user's desktop computer. Easy ABC, on the other hand, has a file structure using approximately 20 files supporting a relational database for each model. Utilities such as PKZIP will archive and compress these, but this adds an additional processing step for both storage and retrieval. Other applications such as WizdomWorks have their own built-in utilities for exporting models. Additionally, for viewing these models, the user must have specialized viewing software. Often such software is provided gratis; Wizdom distributes the WizdomWorks viewer for free.

Despite this the overall process for storing and retrieving documents with complex file structures proved to be most cumbersome and frustrating for the users. To create, store, and retrieve a process model requires the following 15 steps, using three different applications:

To create and upload

- Create and save model
- Export model in IDL format
- Load browser and open **RAPTR**
- Enter project with password
- Go to appropriate node in designers notebook
- FTP model to **RAPTR** server
- Check in model to Designers notebook

To download and view

- Load browser and open **RAPTR**
- Enter project with password
- Go to appropriate node in designers notebook
- Check out model from Designers notebook
- Download model to personal workstation

- Load model viewer
- Import model
- Open model and view

Most users preferred the three-step, single application approach of create, print, and fax, despite the paper clutter and version control problems this frequently created. Also, the create/print/fax scenario provides a limited or nonexistent project record.

This 8-step download process is simpler with applications such as PowerPoint that use a flat file structure. In these, two or three steps are eliminated, depending on whether one wishes to view or download the file. The efficient use of **RAPTR** with relational database applications is an unresolved issue.

During the course of the field trial (July 21 to September 30, 1998) the IMPAC expansion team held two team meetings at Warner Robins, and conducted three video-teleconferences. For each of the nine weeks of the trial, senior members of the **RAPTR** team were on-site, attending teleconferences and team meetings, and working with team members to facilitate their use of the software. By intention the development team was so closely involved, supporting the users, to assure that the field trial was a test of the software, not a test of the users. On two occasions the **RAPTR** PI facilitated team meetings, and other **RAPTR**-related personnel kept minutes or created and maintained models. By the end of the field trial the IMPAC Expansion team had several notable accomplishments, including a fully formed and aligned team, the AS-IS models, identification of the systems involved in processing payments, and a listing of the TO-BE issues. In the course of the field trial work was not begun on the TO-BE model.

7.2.2 Issues and challenges

Several unanticipated challenges arose during the IMPAC trial. All of these are instructive of the benefits of **RAPTR**, and the requirements for its effective use.

Perhaps the most basic of these was the browsers that team personnel had installed on their PCs. **RAPTR** makes use of a push technology to notify team members of task assignments and due dates. However, policy within many USAF components is to disable push features on browsers such as Netscape and IE, rendering this **RAPTR** function inoperable. In addition, several team members did not have web browsers installed on their computers; several weeks were spent discovering and correcting this situation.

In the course of this field trial the team wanted to test out the assessment features, and collect some assessment data. Since the project was already underway, the Team Self Assessment was beside the point. For conducting a High-Level Assessment to determine possible sources of resistance to change, it was unclear who should be assessed: whether a typical product directorate, or the item managers within the PD, or the item managers across all PDs. The last of these was chosen, with the result that because there was no

central direction for the assessment, the response rates did not support any statistically valid conclusions.

In one of the IMPAC meetings the **RAPTR** PI briefed the IMPAC team on the benefits of **RAPTR**, and observed that the purpose of **RAPTR** was to get away from the "big binder" syndrome. This was a friendly reference to the habit of the IMPAC project manager, Ms. Connie Black of carrying relevant project documents in a large, 3-ring loose-leaf binder. Ms. Black found such a binder a useful administrative tool; the **RAPTR** team saw it as a challenge to make **RAPTR** functionality sufficiently accessible and robust that project managers including Ms. Black would find **RAPTR** more useful than the big binder.

Achieving this would have required two actions within **RAPTR**. First, we suggested that in the Designers Notebook, in addition to the public storage areas, there should be a private, searchable folder for each team member. The search feature would speed the retrieval of e-mail and other team communications.

Second, the storage procedure should be more transparent. Currently to store a document, it must first be uploaded to the **RAPTR** server, and then stored in a specific bin in the Designers Notebook. This two-step procedure is useful for maintaining version control (allowing for the option of incrementing version counters), but it makes saving something less than automatic. A drag-and-drop option for storing and retrieving documents would have created far more storage-and-retrieval activity on **RAPTR**.

It typically happens in many organizations that once a project is concluded, the project assets are quickly filed and forgotten. Months or years later another team will want to see some of the project results from the first project, which are unavailable. Thus project teams are continually reinventing the wheel.

The solution to this, of course, is to mandate that project teams use **RAPTR** or a similar document management system having good version control and easy uploads. Project managers who did not use such a system would be responsible for maintaining their project files and providing access to them, even after their project ended. This procedure would solve a problem that was evident early in the **RAPTR** project, the fact that the early users of archives incur the costs yet reap few of the benefits. Mandated use of document management systems would assure that change agents could build on the experience of their predecessors.

Overall this field trial resulted in several valuable ideas for improving the **RAPTR** software, insight into some unanticipated problems, and a better understanding of the requirements for integrating effective project management with project planning and management tools such as **RAPTR**. Some of these insights and understandings are discussed in chapter 10.

7.3 Field Trial at Air Mobility Command

A more successful test of **RAPTR** was conducted by the Air Mobility Command. The Quality office at AMC was chartered to implement the 5-S (Sort, Straighten, Shine, Standardize, Sustain) methodology at multiple AMC wings/locations. They had learned about **RAPTR** through one of the briefings conducted by **RAPTR** Program Manager, and had inquired into the possible use of the tool.

AMC used a team-building methodology called AWO (Action Workout) to visit multiple locations for the 5-S implementation. AWO is an intensive, two-week methodology for rapid process change. The AWO team would arrive at the implementing location and conduct the two-week workshop.

One of the members of the AMC Quality team characterized some of their early efforts at this as "walking into a dark room." Prior to arriving on-site the team had no indication of the readiness of the site for the AWO process.

After learning about **RAPTR**, the AMC Quality office used the Team Self Assessment to gain insight into new locations. The advice returned by the Team Self-Assessment (green for go, yellow for caution, red for stop) was useful in understanding the readiness and the issues they might encounter at the site. This prepared them for issues they might encounter, and arrival on-site in fact confirmed the issues that **RAPTR** had raised.

Overall the AMC Quality office reported that they were quite pleased with the insight and support that **RAPTR** had provided to the 5-S project, particularly in the way it enabled them to anticipate certain issues. It should be noted that this team was able to use **RAPTR** effectively without any training and without any support from the **RAPTR** development team.

7.4 Field Evaluation

7.4.1 Data Collection and Analysis

In connection with the field trials, the **RAPTR** team collected data on the users' experiences with the tool. Data were collected by four means: a survey, a focus group, observations of the support engineer, and telephone and e-mail inquiries. These research activities covered a wide spectrum of those who had used or been exposed to the **RAPTR** tool.

Survey

Most surveys were conducted by phone because it was much easier for the team members to fit it in their schedules. Another set of surveys was administered to the group in time set aside for it at an IMPAC meeting.

Focus Group

A focus group was conducted involving the LN directorate. There were mixed reactions. Although some interest in the tool was expressed, the consensus of the group was that further guidance was required before they should use it.

Software Support

A **RAPTR** software engineer was stationed full time at WR-ALC for a period of four weeks. During this time, he was working on completing **RAPTR** functionality as well as troubleshooting any problems that arose in using the system. On-line support was also available from Wizdom. In all cases, problems, questions, or issues that arose were documented as part of the data gathering for the project.

Phone and Email Queries

The **RAPTR** team at Wayne State University performed many phone and email queries over the course of the trial and analysis. No email replies were received. It was difficult to keep up with all team members due to their constantly changing email addresses. Some came back as undeliverable. The phone conversations were generally useful in obtaining current information about the level of competency regarding **RAPTR**, which tended to be at the level of basic functionality (filling out assessments and browsing project documents).

Participation and Sponsorship

Executive sponsorship remained constant within RE at WR-ALC. There was, however, only moderate participation from team-members. We believe that the participation would have been higher if the makeup of the group had remained more constant throughout the duration of the project, and if the benefits of **RAPTR** had been more transparent. Our efforts to provide one-on-one training to the IMPAC team members was hindered by the changing composition of the team.

7.4.2 Data Analysis Results

RAPTR Functions

Most of the team members used **RAPTR** to retrieve administrative related information about IMPAC such as Action Item Lists, Team Member Information and Meeting Minutes. The IMPAC project did not use **RAPTR** to create a project plan. Since the makeup of the team was predetermined, they chose not to conduct the Team Self-Assessment. They did conduct the "High Level Assessment", but chose to forego the detailed assessment. The document check-out/check-in was used by about half of the team members.

User Feedback

41 members of the **RAPTR** Users Group and the IMPAC team were contacted. Of these, 20 responded either to a telephone inquiry or an e-mail survey. Of the respondents, 60% (N=12) felt that the **RAPTR** system was useful in achieving their project goals.

50% (N=10) of the respondents felt that **RAPTR** was necessary to do their assigned work.

80% (N=16) felt that they understood how to navigate **RAPTR** somewhat to fairly well.

Of the 8 respondents who felt **RAPTR** was not useful, 2 said the way IMPAC was set up did not lend itself to using **RAPTR**.

Summary

The use of **RAPTR** on the IMPAC project was somewhat problematic due to the fact that the IMPAC team were only using certain portions of the system. This prevented the team members from receiving the full benefit of the system. (You are asserting a fact rather than demonstrating a fact. However, there was positive feedback with regard to the possibility of using the system on other projects. The functions used were reported to be useful and convenient to the team members.

It is important to note that as with most knowledge management software the early users are not able to reap the full benefit. Early users bear the burden of populating the knowledge base, whereas later users are able to harvest that which was earlier planted. **RAPTR** is designed to be a better information resource after each use. For example, the Notebook Library should contain detailed histories about previous projects. Obviously, this cannot happen until there have been previous projects. We populated the Library with data from other reengineering projects, but since they did not use **RAPTR** for their projects, they simply aren't as relevant. The same is true regarding lessons learned. Lessons learned without **RAPTR** may not be applicable to a project that is using **RAPTR**. Furthermore, for **RAPTR** users, many of the important lessons may directly concern the tool itself.

7.4.3 Discussion

Our initial expectations were very high for **RAPTR**. We intended to have a pilot project that would ideally use **RAPTR** from beginning to end to help plan, run and implement the necessary changes associated with their particular effort. Scheduling issues forced the IMPAC project to use their own project plan, eliminating the assessment portion of **RAPTR** functionality. IMPAC team members were able to use **RAPTR**'s data storage and retrieval as well as the electronic interface designed for helping to make email more

easily accessible between team members. Other functions were not used in the field trial.

7.4.4 Cost-Benefit Analysis

To provide further guidance on the use of **RAPTR** on different projects, the team compiled observations on the costs and benefits of using **RAPTR**. Although the shortness of the field trial did not permit collection of quantitative data, we describe them in detail and also supply trial metrics for each. An experienced project manager should thus be able to construct a rough estimate of the net benefits for using **RAPTR** on a specific project. The list below summarizes three classes of costs and six classes of benefits, along with trial metrics for each:

RAPTR Costs

Training:

An experienced computer user should be able to understand and gain familiarity with **RAPTR** without extensive training. Because change team members often have a wide range of computer experience, hands-on training designed for **RAPTR** users may best communicate the capabilities of the tool.

Trial metric: Standard **RAPTR** training requires two days, including hands-on exercises.

Server Administration:

Personnel for server administration are necessary in order to assign user identification and passwords. For optimal **RAPTR** usage, an administrator should be assigned to the **RAPTR** server.

Trial metric: After setting initial users and passwords, administration consist primarily of maintaining passwords for changing team composition. Time involved is a function of the server OS and the experience of the administrator.

Initial Process Redundancies:

Until change teams establish and accept standard routines for communicating, collaborating, and storing information through **RAPTR**, workloads may appear to increase as previously "paper" information is loaded into the **RAPTR** notebook. This redundancy can be countered by careful front-end planning. The "paperless" capabilities of **RAPTR** can be very cost effective, just as e-mail is cost-effective when compared to paper-based systems.

Trial metric: This is a function of the experience and discipline of the change management team. In the long run, "doing it right the first time" improves productivity; in the short run, learning disciplined project management may require a week or more of training. An appropriate metric here would be perceptions of false starts and wasted effort in communicating within a change management team.

RAPTR Benefits

Project Data Management

Project data files are centrally located on one server for team member access and sharing. The most up-to-date task documentation can be easily accessed by authorized team members. Password control is available to maintain desired security. A search engine assists team members with locating appropriate files. **Trial metric:** Time associated with hunting for missing documents should be dramatically reduced.

Virtual Teaming Tool:

RAPTR can be utilized to minimize the need for phone, fax, face-to-face team communications.

Trial metric: Time associated with phone tag, TDY, and meetings.

Project Management Tasking:

Automated e-mail tasking capabilities are available to assist team leaders in making task assignments.

Trial metric: The benefit here will be more in the clarity of the assignments and their scheduling, rather than saving the team leader any time. Assignments will still need to be discussed with team members.

Insight into Potential Change Targets:

Front-end cultural assessment indicates organizational areas that may benefit from change initiatives.

Trial metric: Fewer false starts on change management projects.

Insight into readiness of the IPT:

For newly formed teams, a clarification of steps needed to prepare for the project.

Trial metric: Fewer false starts on change management projects.

Knowledge Management:

Lesson Learned from previous change projects accumulates as projects are added to **RAPTR** library of Designers Notebooks.

Trial metric: Reduced confusion and ambiguity within the IPT regarding appropriate steps to take in a change management project.

As is evident from the foregoing, the costs are closer to being validated than the benefits. Numerous potential benefits offer the possibility that they will outweigh the costs.

7.5 Conclusion

At the final briefing for **RAPTR** on October 30, we observed that sufficient interest existed in the tool's capabilities to make it worth deploying. However, we concluded from the IMPAC experience that in its present state, the tool is not self-starting: some level of orientation and training are required for a project team to make effective use of the tool.

Note should also be made here of maintenance issues that remain outstanding with **RAPTR**. Since the assessments were not used in the IMPAC project, there was no opportunity to tune or calibrate the instruments. As presented in our final briefing, model maintenance is one of the ongoing requirements for a tool such as **RAPTR**: as assessment experience increases, and as uses and circumstances change, some modification to the models is indicated. At this point even modest maintenance would significantly improve the usefulness of **RAPTR**.

In chapter 10 we examine the how, where, and why of using **RAPTR** on a specific project; these issues should be the focus of orientation prior to a decision to use **RAPTR**. Additionally, project managers using **RAPTR** should be fully conversant with

RAPTR was designed to support process redesign and change management within the context of an organization that was undertaking significant organizational restructuring or improvement. This is consistent with the change management literature of the mid-1990s, which saw processes within their organizational context as the appropriate targets of change.

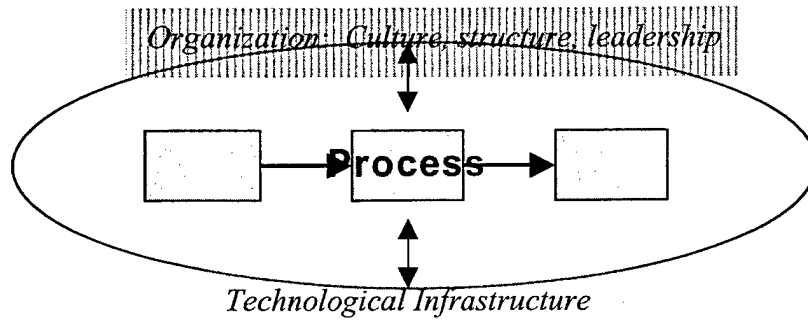


Exhibit 7.2 – Traditional Reengineering Context

Experience on the IMPAC project, and more recent organizational literature, suggests a revision of this strategy of organizational change. The IMPAC process was a highly structured process that cut across multiple functional and operational units in multiple locations.

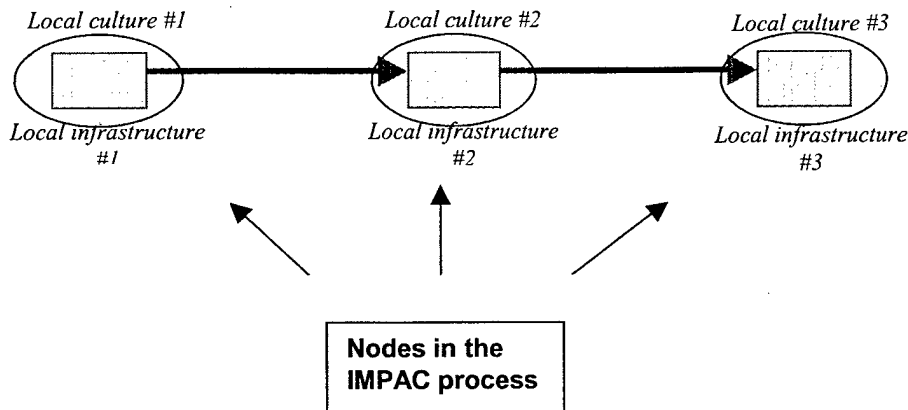


Exhibit 7.3 – IMPAC as exemplifying the new reengineering context

When we attempted to conduct an assessment of resistance factors, the organizational factors described here defeated the assessment. In brief, we faced three alternatives:

1. Assess the culture and organizational structure of AFMC, using a sampling strategy;
2. Construct an articulated model of item managers, purchasing agents, financial managers, contracting officers, and shop managers to identify differential sources of acceptance and resistance; or

3. Redesign the IMPAC project as a pilot project focusing on a single location.

Alternative #1 was far beyond the scope of the IMPAC project, and would have required buy-in from much higher levels; alternative #2 exceeded the technical capabilities of **RAPTR**; alternative #3 was precluded by the pre-set nature of the IMPAC project.

The development of **RAPTR** was based on the mainstream change management and reengineering literature, such as Hammer and Champy, Andrews and Stalick, Davenport, and Kotter.²⁰ This mainstream change management literature is based on the traditional view of the corporation associated with Chester Barnard and Herbert Simon.²¹

Recent organizational literature has come to view organizations not as determinative entities as in the traditional view, but rather as accidental congeries of strategies, processes, personnel, and infrastructure, contingently subject to some form of leadership.²² In other words, the traditional, top-down model of leadership determining the organizational form is being replaced by a more negotiated view that sees leadership operating within an interpretive context that it can influence but not control

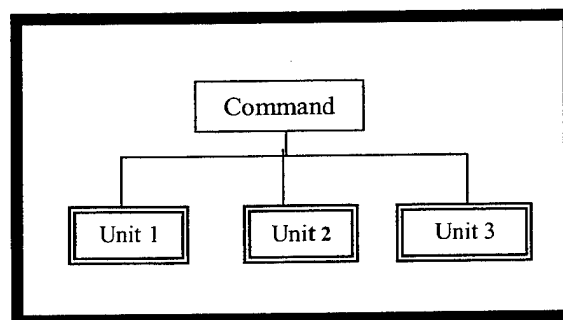


Exhibit 7.4 – traditional chain of command management

²⁰ Hammer, Michael, and James Champy. Reengineering the Corporation. New York. HarperBusiness. 1993. Davenport, Thomas. Process Innovation: Reengineering Work through Information Technology. Boston. Harvard Business School Press. 1993. Kotter, John P. Leading Change. Boston. Harvard Business School Press. 1996.

²¹ Barnard, Chester I. The Functions of the Executive. Cambridge. Harvard University Press. 1938. Simon, Herbert. Administrative Behavior. New York. The Free Press. 1945.

²² Barley, Stephen. The Alignment of Technology and Structure Through Roles and Networks. Administrative Science Quarterly 35, 61-103. 1990. Cohen, Michael D., James G. March, and Johan P. Olsen. A Garbage Can Model of Organizational Choice. In March, James G., ed., Decisions and Organizations. Oxford, Basil Blackwell. 1988. March, James G., and Roger Weissinger-Baylon, eds. Ambiguity and Command: Organizational Perspectives in Military Decision Making. Marshfield, Mass. Pitman, 1986. Sproull, Lee, Stephen Weiner, and David Wolf. Organizing an Anarchy: Belief, Bureaucracy, and Politics in the National Institute of Education. Chicago. University of Chicago Press. 1988. Daft, Richard, and Karl E. Weick. Toward a Model of Organizations as Interpretation Systems. Academy of Management Review 9, 284-295. 1984. Alvesson, Mats, and Hugh Willmott. Making Sense of Management. Thousand Oaks. Sage Publications. 1996. Martin, Joanne. Cultures in Organizations: Three Perspectives. New York. Oxford University Press. 1992.

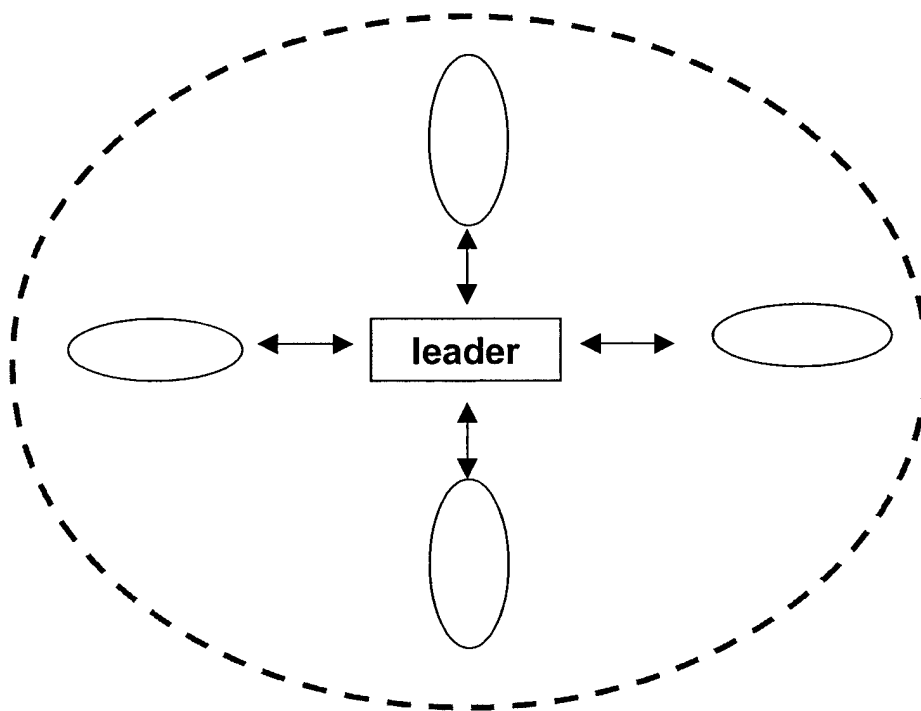


Exhibit 7.5 New Context of Change Management

Understood from this viewpoint, the IMPAC project was not the traditional BPR project executed with the mandate and sponsorship of the principal officer of an organization; instead, IMPAC might portend a new project model, bringing together:

- A pre-defined process (in this case, a payment process) spanning multiple units
- An executive mandate (in this case, from an assistant secretary of defense)
- A strategic objective (expanded use of the IMPAC card)
- A project team consisting of multiple personnel from multiple locations
- A heterogeneous technological infrastructure that was a given for the project
- An experienced, effective project leader

None of these were effectively “owned” or the responsibility of any single individual, activity, or command. This challenge, of managing change in an environment of shared, negotiated leadership, will be discussed in our conclusion.

In our view, the **RAPTR** demonstration was a qualified success. Unfortunately, due to factors beyond our control, we were not able to fully exercise all components of the system. We did, however, clearly demonstrate that it has the potential to be a very useful tool to those involved in change projects. We also were able to identify and resolve several glitches in the system, which is a major goal of such a pilot.

Several other AF units have expressed an active interest in using **RAPTR** to facilitate planned or ongoing change projects at their installations. We advise in the strongest terms that they allow the time and resources for proper training and familiarization with this tool before using it. To do otherwise risks disappointment and a premature demise of what promises to be an excellent tool in the change management kit.

Chapter 8: Lessons Learned

Over the course of the *RAPTR* development and abbreviated field trial, the team had the opportunity to reflect on both its successes and its failures. In this team effort members of the team were encouraged to comment on and evaluate the directions and progress of the team. Presented here, in summary form, are some of the dominant conclusions and lessons learned that the team drew from the entire project.

8.1 Component-level Lessons Learned

1. The power of simple assessments

One of the least expected lessons was the power of simple assessments. The Team Self-Assessment, perhaps because it required so little effort, and returned information in easy-to-understand categories, was the best received of the three. The Team Self-Assessment, in contrast to the High-Level Assessment and the Detailed Assessment, had a small number of questions and was based on a two-dimensional linear model of team readiness. This simplicity made it easy to build, easy to take, easy to understand, and easy to maintain.

The development team at times referred to such simple assessments as “Readers Digest Assessments” (After the linear, 20-question self-scoring questionnaires found in that publication); this was a condescending reference to the low level of intellectual content found in such assessments. This characterization, however, could be read another way: the simplicity and transparency make the assessment accessible in ways that the *RAPTR* assessments are not. Far more copies of Readers Digest than of *RAPTR* have been fielded, although the uses of Readers Digest assessments are typically not for critical organizational evaluation.

2. The usefulness of the reference model

The Reference Model of Change Management turned out to be one of the most difficult and useful components of *RAPTR*. On the one hand, in building it required extensive data-gathering on change management methodologies, numerous adjustments among these, and frequent tradeoffs between levels of detail and abstraction. In building the model we discovered that standard change management sources tend to be in agreement in stages II and III (analysis and design), and vary most widely in stages I (Preparation and Strategic Assessment) and IV (Implementation).

Once built, however, the Reference Model turned out to be a supple mechanism for providing guidance on change management. The Reference Model is supported by activity descriptions for each of its 91 nodes; this provides a comprehensive guidance to all the technical aspects of change management. As a strategy for providing guidance on specific projects, this comprehensive, tailorable Reference Model is a useful approach; members of the **RAPTR** team have since applied this strategy to such diverse areas as EDI implementation and supply chain management.

3. Building assessments from the ground up

RAPTR contains three assessments: the Team Self-Assessment (TSA), the High-Level Assessment (HLA), and the Detailed Assessment (DetA). These assessments run on top of three different models to provide four different sorts of information and advice:

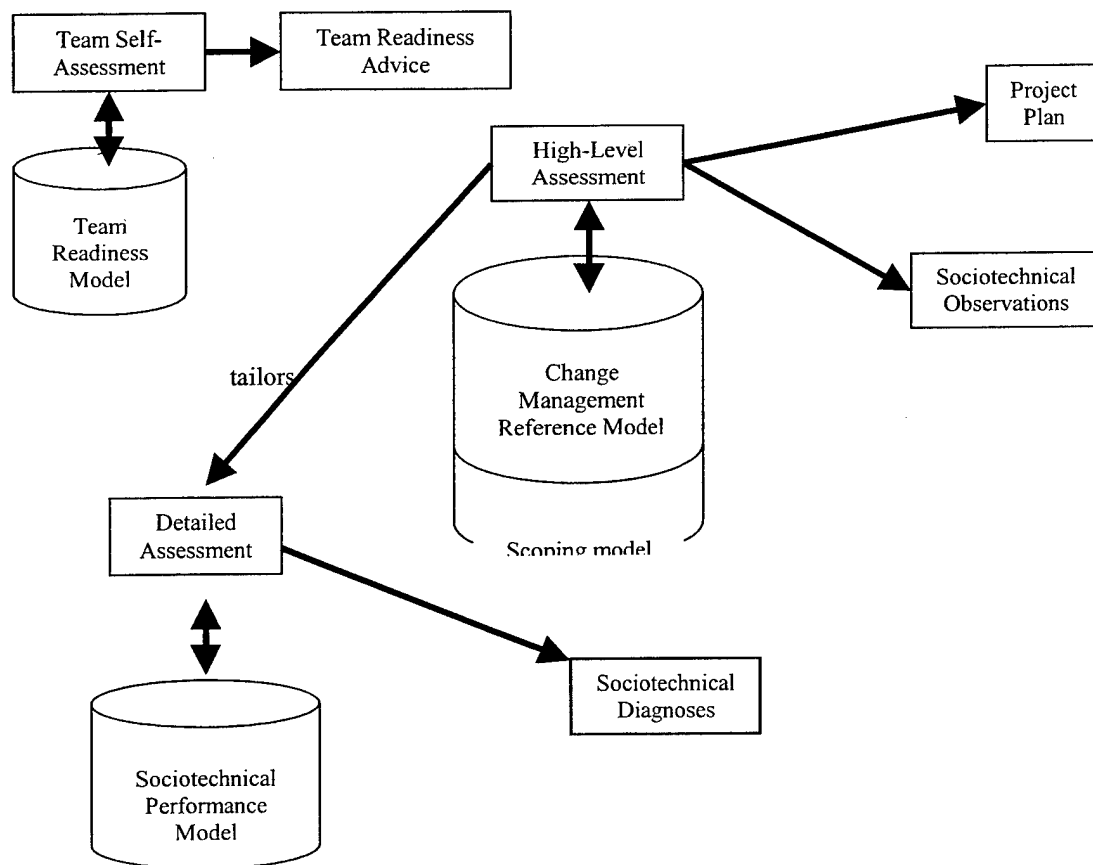


Exhibit 8.1 – RAPTR Assessments

In this diagram the Team Readiness Model is an experience-based model (based largely on the Principal Investigator's experience with numerous reengineering projects),

whereas the Change Management Reference Model and the Sociotechnical Performance Model (described in our proposal as the Deductive Model) are based on multiple literature and theoretical sources.²³ The sociotechnical performance model identifies those factors in the domains of process, technology, organization, communication, culture, and human resources practices that are associated with effective organizational performance. Further descriptions of the sociotechnical model and the Change Management Reference Model are contained in chapter five of this report. The choice of a deductive model was made based on the experience of the FRAME/WORK project, which used an inductive model of technology implementation: The PI had concluded that such inductive models were too difficult to build, and too context-specific to be of general utility.

In building the High Level Assessment, empirical research conducted at Warner Robins ALC was extensively used. Those components of the model and the assessment that are closest to this research have turned out to be more accessible to the users, suggesting that user accessibility needs to be strategically factored into further such efforts at model-building.

8.2 Architecture and Integration Lessons Learned

4. More Robust Modularity

RAPTR pursued a modular design for its software, which proved to be a useful strategy for designing and building complex software. However, the tight coupling between the CMRM and the Scoping Model (consisting of a set of rules linking assessment results to task durations) essentially meant that update of one of these models would require update of the other. A solution to this dilemma is proposed in the concluding chapter.

Large scale software requires modular design. However, when the design falls short of modularity, there should be an explicit understanding of the maintenance costs that will be incurred.

5. Modules should be field-tested before integration

The original *RAPTR* development approach called for paper-and-pencil tests of components prior to building them. For the assessment components, this was

²³ Schein, Edgar. Organizational Culture and Leadership. San Francisco. Jossey-Bass. 1992. Daft, R. Organizational Theory and Design. New York. West Publishing. 1995. Majchrzak, Anne, Mitch Fleischer, and D. Roitman. Reference Manual for Performing a HITOP Analysis. Ann Arbor. Industrial Technology Institute. Mintzberg, H. Structure in Fives: Designing Effective Organizations. Englewood Cliffs. Prentice-Hall. 1983. Adler, Paul. New Technologies, New Skills. California Management Review 28, 9-28. 1986.

accomplished. However, other components (such as the Designers Notebook and the Notebook Library) did not lend themselves to paper-and-pencil tests.

At the risk of elongating the development schedule in the future, articulated software such as **RAPTR** should be fielded at the module level, prior to integration. Although often the greatest value of such software comes from the ability to pass information from component A to component B, if component A goes unused this value is lost.

Most components of **RAPTR** did not receive a field trial; of those that did (Netscape Browser, Team Self-Assessment, Designers Notebook, MSP Interface and messaging, and the Altavista Search Engine), some performed beautifully and others clearly needed more work. Had this been known before starting (what was supposed to be) a full-scale field trial, the actual field trial would have been more successful. It would have been beneficial to have less of a rush to complete the integrated package, and a more deliberate approach to testing: the development team takes responsibility for not pursuing this strategy.

6. Reliance on networks adds risk

As a concept **RAPTR** evolved from a standalone PC tool, to a LAN tool, to an extranet (a wide-area network using internet protocols) tool. Each of these steps added unforeseen technical risk, particularly in the environment of rapidly evolving internet standards.

Due to the compressed nature of the field trial, **RAPTR** needed to “work the first time” for all users. Instead, at all IMPAC locations user visits were required to make sure **RAPTR** worked properly. These validation and verification steps should have taken place before the field trial started.

Part of the risk was the use of internet protocols and the Internet Explorer browser; users who had Netscape on their desktop computers experienced minor but annoying problems with **RAPTR**. Browser independence should be a design goal.

7. Training should begin with proficiency on existing tools, including basic computer skills.

RAPTR was designed to integrate and enhance the tools used by change management teams. These tools-in-current-use included the Microsoft Office suite, Windows 95, and assorted modeling tools. In the course of the field trial, we discovered that skills with the existing tools were unevenly distributed.

A **RAPTR**-supported project should be able to assume not only a team trained to use **RAPTR**, but that the same team has a common baseline in the other tools they use. Pegging this baseline of skills at some pre-determined level is less important than having

all team members together at the same baseline; otherwise there is a likelihood that **RAPTR** use will devolve to the level of the least skilled member of the team.

8. The development team should have participated on a project team

RAPTR development team members interviewed RE personnel, but did not actually participate in an RE project. Had this been negotiated in advance, some of the field issues might have been better anticipated.

9. An explicit model of integration would have been useful

Although the goal of **RAPTR** was to produce a tool that would integrate assessment, project planning, and project management capabilities making maximum use of COTS tools, the conceptual presentation did not distinguish among different forms of integration. These different forms include (in increasing order of difficulty):

- 1 Offline exchanges of single files on a common platform
- 2 Transfer of single files across a network
- 3 Uploads and downloads of record data across a network.
- 4 Use of common data resources on a common platform
- 5 Use of common data resources across a network
- 6 Transfer of complex data resources (e.g., relational databases) across a network
- 7 Exchange of field- or record-level data between applications on a single platform
- 8 Exchange of field- or record-level data across a network
- 9 Extraction and exchange of semantic information between applications

All but the first two of these tend to be application-specific: Knowing what files, fields, or records to extract and exchange requires knowing what application they will be used in. (Some applications are more robust than others in terms of what data resources they can use.) **RAPTR** integration tends to be at levels 2 and 3, less as a design goal and more as a default because **RAPTR** was not engineered to support specific applications. Although this was probably inevitable, a clearer understanding of it in conceptual development would have been useful.

10. Need to better understand automation tradeoffs

As described in chapter 5, automation targets were set for each component of **RAPTR**. In this chapter we commented that higher levels of automation entailed higher costs and higher development risks. There were at least two other tradeoffs with higher levels of automation that were not anticipated at the time: First, the more highly automated components, such as the assessments, required greater effort to maintain and upgrade.

Second, when two components were integrated, the more highly automated of the two drove the maintenance requirements of the less automated. An example of this would be the link between the reference model and the assessments. Changes in the assessments required corresponding changes in the reference model.

8.3 Deployment and Lifecycle Lessons Learned

11. Importance of the field trial

When the *RAPTR* budget was cut halfway through Phase II, several development decisions were frozen: the team decided to complete the development as planned, and shorten the field trial.

An alternative strategy would have been to radically truncate the development, and proceed as quickly as possible to the field trial. Given the difficulties we experienced in coordinating the field trial (see next item), this would have allowed for a more complete field hardening of the tool.

Of the lessons learned here, most resulted from our limited field experience. A longer field trial, with more than one cycle, would have created the opportunity to correct several limitations of the tool that were revealed only in the course of the field trial.

12. Difficulty of synchronizing schedules in field trials

Although this would seem obvious, it was not sufficiently anticipated. Stated briefly, when one's site for a field trial has an operational mission, there needs to be an explicit strategy for accommodating the schedule of the development team to the mission requirements of the field site. At a minimum this must include:

- Scheduling flexibility for the developers
- Contract flexibility for the program, including the ability to carry over obligated funds
- An understanding by the development team that development efforts may be suspended while waiting for a field trial to come along
- Anticipation in the project schedule and budget for such times of "suspended animation" in the development effort

13. First field trial should be a made-up project, not mission-critical

In retrospect this seems fairly obvious: The PI has a principle that one should not wait until the night before a big project is due to try out new software. Likewise, in a program

of multiple field trials, the first trial should be a “toy” or artificial project, where there is no penalty for delays or mis-steps.

14. The partnership with the field site should have been reinforced

Techniques for this might have included loaning a team member to other IPTs at the field site, or a more sustained presence there. Although there was contractual sensitivity to having the development team provide project support to the field site, in retrospect such an activity might have given the team further insight into the nuances of project management there.

15. The owner of the field trial should have management authority over the field project; avoid ragged starts.

There was a split ownership of the field trial, with the program manager owning the IMPAC project, and the *RAPTR* PI owning the *RAPTR* trial. There would seem to be some desirability of coordinating these two under a single manager, which would have avoided some of the scheduling difficulties in integrating the two (particularly given that both people had additional responsibilities in addition to IMPAC and the field trial).

16. It is insufficient that the software be built to spec; greater sensitivity to field issues would have been useful during the development period.

During the entire two months of the field trial, *RAPTR*'s primary developer was on site at Warner Robins. At the risk of inefficiency, it would probably have been desirable to have additional individuals responsible for interface design to spend additional time on site. (The interfaces – with the user, and between components, are the most difficult to spec; over-specifying locks in potentially inappropriate designs; under-specifying can cause a mis-interpretation of the requirement. More intensive on-site exposure is the proposed solution here.)

17. Greater investments should have been made in training, including training in basic computer skills, as part of an explicit field strategy.

Given our findings of significant unevenness in the users' computer skills, using training as a field strategy might have enabled us to surmount some of the scheduling problems that surrounded the field trial. By integrated and extended training on browser and search skills, project management tools and skills, *RAPTR* components, and finally the integrated package, we would have had much richer data on *RAPTR* use, and possibly much more rapid user acceptance of *RAPTR*.

18. Training should have begun during Phase I of the project; training in computer skills, and orientation to concepts such as project management, knowledge management, and cultural assessment would have been useful.

Although **RAPTR** was designed to meet a need identified at higher levels of management within RE, this need was perceived and understood quite differently by the project teams that used RE. A much more intensive conceptual orientation could have closed this gap.

19. A command briefing early in the project might have resolved some of the deployment issues.

Given the Air Force's investments in **RAPTR**, it is not inappropriate to suggest that as early in the project as possible the commanding general of the field site at the time of the field trial should have been briefed on the tool.

8.4 Strategic Lessons Learned

20. Understanding user accessibility

User accessibility is more than just look-and-feel; it includes user understanding of software concepts, and it also includes technological infrastructure.

The **RAPTR** team devoted significant attention to assuring that the software would have an acceptable look-and-feel: a graphic designer was brought onto the team, and images from the Air Logistics Center were emulated. The development team could have made the software more accessible by creating vernacular views of both the assessments and the document management functions. (The second of these is treated in Lesson Learned #21). A vernacular view of the assessments would have begun with how the users understood issues of culture, communication, etc., and would have stepped the users through an improved understanding of their own culture etc. The multiple-choice question format, while not unfamiliar to the users, failed to give them an understanding of the importance of the cultural assessment.

The other barrier to accessibility was the technological resources available to the users. **RAPTR** was engineered to be used with current version browsers (Internet Explorer 4 or Netscape 3.0). At the beginning of the field trial most users did not have these installed on their desktop computers, a condition that required several weeks to correct.

21. Emulating management styles

An understanding of different management styles should be integrated into the concept of any management support tool. **RAPTR** used a textbook understanding of project management, which gives importance to workflow, version control of documents, and the intensive use of project management tools (such as Microsoft Project).

In retrospect, the project management styles that we observed were far more varied and far less structured than this. Reference was made in chapter 7 to the “big binder”, a record-keeping tool that brought together all key project records in a portable, easily-browsed format. The “big binder” is the preferred management tool of many project managers. Other tools, such as spreadsheets and chartmaking tools, are delegated to contract support personnel, freeing up the project manager to focus on the interpersonal aspects of the project.

A key principle in developing **RAPTR** was not to second-guess project managers: we assumed that if a project was managed in a certain way (such as with a “big binder”), it was because the project manager had a good reason for doing so. More time should have been invested in understanding these styles so that **RAPTR** would emulate and enhance them.

22. Make management philosophy assumptions explicit.

In addition to management style, management philosophy is coded into management support tools; if the tool supports a pre-determined philosophy, the tool will be accepted. Some of the basic distinctions in management philosophy (contrasted here to management style) would include:

- What does one manage? Schedules? Budgets? Deliverables? Relationships? Knowledge? Values?
- What are the feedbacks built into a management system? How is good management recognized and rewarded? How is bad management corrected?
- How are management responsibilities and capabilities aligned?
- Are management and leadership the same thing, or are they different?

These issues differ from management style, in that style tends to be an individual-level issue, whereas philosophy is more dependent on the entire organization. One could say that answers to questions such as these comprise the management culture of an organization such as the US Air Force.

Chapter 9: Using *RAPTR*: Where, How, Why

Based on observations during and evaluations of the field trial, and conversations with other users, we have developed some guides for the appropriate and effective use of *RAPTR*. These involve locating specific projects within arrays of problem types, context types, and team characteristics. We also present here recommended management direction, and the maintenance requirements for *RAPTR* to continue to be a useful tool to the Air Force.

9.1 Problem types

As described in chapter 5, part of the *RAPTR* planning model was a characterization of different project types. To review, this variable, *ZTYP*, a nominal variable, had 8 different possible values:

- A. **Project definition** – projects that are intended to define other improvement projects
- B. **Process improvement** – projects, typically follow-on to type A, that improve specific processes
- C. **Classic reengineering** – full-scope efforts beginning with a strategic assessment and concluding with implementation of new systems
- D. **Change management** – Making pre-determined process changes.
- E. **Project management** – a user-defined alternative with user-selected steps in the reference model
- F. **Analysis paralysis** – a project that concludes with the analysis of the AS-IS
- G. **Greenfield projects** – a project that starts by defining the TO-BE
- H. **System implementation** – involving no process change, just the introduction of new systems

Although this exhausts the logical possibilities within the Change Management Reference Model, it does not exhaust the types of change projects confronting the Air Force. Among the types of projects that *RAPTR* is not designed to support are:

- **Systems design** – the technical aspects of designing databases, applications, and communications systems.
- **Strategic change** – developing new policies and priorities for committing organizational resources
- **Relationship change** – projects that alter the allocation of authority, responsibility, or resources among different units.

None of these three, we note here, are the sort of reengineering or change management projects that **RAPTR** was intended to support. However, some projects, such as the IMPAC project, while appearing on the surface to be process change, do, in fact, involve significant relationship changes. A useful upgrade to **RAPTR** would be, when setting the value of ZTYP (Team Self-Assessment, project leader questionnaire) to identify those types of projects for which **RAPTR** was not recommended. If the project fit one of these three types, then the advice returned would be to not use **RAPTR** on the project.

9.2 Project contexts

RAPTR was designed to support process change projects chartered by the unit commander of a mid-size unit. A project involving a single office would be too small to justify use of **RAPTR**, whereas a project involving multiple processes in an entire command would exceed **RAPTR's** technical capabilities. Between these two extremes are located projects of an appropriate size for **RAPTR**. The table below is a rough guide for determining if the project scale and scope are of an appropriate size to make **RAPTR** use beneficial.

Levels of authority >>	1	2	3	4	5
Varieties of processes					
Few, focused on one or two specific functions	OK	OK	OK	OK	use advisedly
Intermediate, spanning multiple functions	OK	OK	OK	use advisedly	use advisedly
Broad, encompassing all functions found within a center or a wing	null	null	use advisedly	use advisedly	not recommended

**Exhibit 9.1 –
Contexts in which to use **RAPTR** – Scope and Complexity**

In this table we illustrate the interaction of project complexity and unit span of control “OK” states that use of **RAPTR** is indicated, “Use advisedly” states that while use of **RAPTR** is recommended, the political sensitivities of dealing with multiple relationships, levels of authority, and functions indicates that an experienced project manager should add his or her own judgment to recommendations made by **RAPTR**. In the lower right-hand cell “Not recommended” indicates that the project is going to be too large and too

freighted with strategic and relationship issues for effective use of **RAPTR**; “Null” cells are those where it is improbable that such a project will exist.

The second element of **RAPTR** use is the involvement of leadership. The issues that **RAPTR** assesses – culture, organizational structure, technology – are not issues that a manager, three levels down from the unit leader, can manage. Each has strategic and other implications that only the commander has responsibility for. For a branch chief to address cultural or technological change issues within an Air Logistics Center, without the close involvement and support of the center command, is an exercise in futility.

Since projects are often delegated multiple levels of authority below the project sponsor, some guidance will be useful to determine the appropriate level of authority for directing a **RAPTR** project. Like the previous template this grid provides rough guidance, rather than definitive solutions. This chart suggests those situations (indicated as “graduate”) where a change manager should enlist a high level of sponsorship, rather than attempting the change himself. This chart does have the advantage of closely mapping to **RAPTR**: If a High-Level Assessment reveals, for example, that the leading problems are problems of organizational structure (rather than, for example, process), then intervention by higher authority is indicated.

Sponsor/Manager (see note)	0	1	2	3
Problem				
Process	OK	OK	OK	OK
Communications	OK	OK	OK	OK
Human Resources	OK	OK	OK	graduate
Technology	OK	OK	OK	graduate
Culture	OK	OK	graduate	graduate
Organization Structure	OK	graduate	graduate	graduate

OK = acceptable level of project management

graduate = this sort of problem requires higher authority

Note: The horizontal axis is the difference between the project sponsor's level and the project manager's level. Thus projects involving organization structure should have the active involvement or management of the project sponsor; projects involving communications change, however, can be delegated down two levels.

Exhibit 9.2 – Levels of Sponsorship

10.3 The Change Team

The knowledge elements of **RAPTR** were conceptualized as supplying the experience and insight of a seasoned and conscientious consultant. This resulted from the realization that the success or failure of many change projects results from the quality of the consultants that are engaged to support the project. It was thought that **RAPTR**, through its knowledge bases, lessons learned, and project repository, could provide much of this experience and insight.

Appreciation and use of such experience and insight depends in part on the character of the change team and its leadership. There is a threshold of value that **RAPTR** adds, which is a function of the experience of the team and the complexity of the project. An experienced team that is familiar with the project context would probably have little use for **RAPTR**'s project planning advice, although they could be expected to make robust use of its document management capabilities. An inexperienced team on the other hand might see little value in the project archiving capabilities.

RAPTR provides a full spectrum of project planning and support functions. Different types of teams in different project situations will, however, be expected to focus on certain parts of **RAPTR**. The table below identifies which aspects of **RAPTR** we would recommend a team begin with as it approaches a new project, based on the team and the context:

Project Scope >>>	Small: a branch or single office	Medium: A division or three-letter unit	Upper medium: An entire directorate or two-letter organization	Large: An entire center	Very large: an entire USAF command
Description of change management team					
Completely inexperienced	Use of RAPTR is critical: Team self assessment, High Level Assessment, Methodology Advice	Use of RAPTR is critical: Team self assessment, High Level Assessment, Methodology Advice	The team may be out of its element	A more experienced team is required	A more experienced team is required
A few members have had experience on one or two projects	Use of RAPTR is critical: Team self assessment, High Level Assessment, Methodology Advice	Use of RAPTR is critical: High-Level Assessment, Methodology Advice, Document Management	Use of RAPTR is critical: High-Level Assessment, Methodology Advice, Document Management	The team may be out of its element	A more experienced team is required
Most members have experience on two or more projects	RAPTR functionality would be overkill	RAPTR functionality will be useful: Methodology Advice, Document Management	Use of RAPTR is critical: High-Level and Detailed Assessments, Document Management	Use of RAPTR is critical: High-Level and Detailed Assessments, Document Management	Use of RAPTR is critical: High-Level and Detailed Assessments, Document Management
Nearly all members of the team are highly experienced at change management	RAPTR functionality would be overkill	RAPTR functionality will be useful: Methodology Advice, Document Management	RAPTR functionality will be useful: High-Level Assessment, Methodology Advice, Document Management	Use of RAPTR is critical: High-Level and Detailed Assessments, Document Management	Use of RAPTR is critical: High-Level and Detailed Assessments, Document Management

Exhibit 9.3 – Use of *RAPTR* functions

In the view presented here, **RAPTR** functionality gets its greatest use on projects where the project scope presents significant but not insurmountable challenges to a team's skill and experience: **RAPTR** extends the team's capability.

9.4 Maintenance requirements

At our final briefing on October 30, 1998, we identified the maintenance requirements of **RAPTR**. These fall under the headings of maintaining the models, maintaining the knowledge bases, and maintaining the software.

Model maintenance for the Change Management Reference model is a matter of keeping this model up to date as methods and philosophies of change management evolve, and as we accumulate experience and lessons learned for change management activities. Additionally, attention should periodically be given to re-calibration of activity durations as project experience is accumulated.

Model maintenance for the Assessment Models is a more detailed task. Each of the assessments is built upon a model (of team readiness, of organizational symptoms, of organizational diagnostics) that was originally based on the experience of **RAPTR** team members including those from Warner Robins. Early in the project, as reported in chapter 5, it was decided not to make **RAPTR** a learning tool, largely because of the additional costs that would have been incurred. Hence there is a need for **RAPTR** to learn and grow off-line. As the assessments are used, managers and maintainers should be asking the following questions:

- Did the results make sense?
- Did the results help me figure out what to do?
- What parts of the assessment seemed more/less correct.

Periodic review of the models in light of these responses will evolve the assessments into quite powerful tools.

Maintaining the knowledge bases requires attention to another set of issues:

- What do the users want to know?
- To what extent is **RAPTR** addressing their concerns?
- What would they like to learn that **RAPTR** is not telling them?

As **RAPTR** evolves, additional content could be added to the knowledge bases, including knowledge of:

- Operating principles
- Case Studies
- Lessons Learned
- Operating Standards

The **RAPTR** repository, when coupled with the Altavista Search Engine, is a powerful tool for knowledge management and maintenance. It could be configured around model projects that would be created purely for capturing knowledge of this sort.

9.5 Management direction

The final element of **RAPTR** use is understanding the forms of management direction and guidance required for its effective use. Like any other powerful tool, if misapplied, **RAPTR** can end up doing more harm than good.

Management direction in the use of **RAPTR** is required in two areas: understanding the broader context of a change management project, and setting project conventions and rules-of-use for the tool.

It is a requirement for leadership to demonstrate that change management is an ongoing enterprise, and not a one-time event or a series of disconnected projects. If the members of a project team see their involvement as a diversion from their “real” jobs, to be completed and left behind as quickly as possible, then they will see using **RAPTR** as a burden with little benefit. If, on the other hand, they understand that the issues, analyses, and results of today’s project will be revisited in one, two, or three years, then team members will understand that they or their co-workers will benefit from using **RAPTR** to maintain institutional memory.

A scenario:

Jim (hanging up the phone): Well, I've just been tasked to redesign our inventory records.
Mary: That sounds like the project I worked on two years ago.
Jim: Do you have any of the process maps you created for that project.
Mary: No. We threw those out as soon as the final report was written.
(A dark look crosses Jim's face)

The dilemma here, as the **RAPTR** team noted during the development, is that the first users of an archiving capability such as that of **RAPTR** get all of the pain and none of the benefit. It is only as the archive grows that subsequent projects can begin to benefit from it.

The growing of the **RAPTR** archive, for any given command, should have strategic direction. It is much like building a library: An assemblage of odd volumes, purchased haphazardly, has little value other than to collect dust, whereas a collection of books, strategically assembled and maintained, eventually becomes a library of priceless value.

In a dynamic environment effective action depends on making sense out of a confusing environment, a task for which knowledge resources and learning from experience are essential²⁴ This requirement for strategic knowledge indicates a second requirement for management direction. Not all projects are worth archiving, and not all project materials deserve to be saved. At the conclusion of a project the questions should be asked:

²⁴ Weick, Karl. Sensemaking in Organizations. Thousand Oaks. Sage Publications. 1995.

1. Do I want to remember what we did on this project? Do I want my successor or co-workers to know what we did, warts and all?
2. Which documents are most worth saving? Which should be thrown out?

The easiest answer to question #1 is to trash the entire project. The easiest answer to question #2 is to save everything. Between these two extremes lies a set of intelligent decisions that create a project record that is useful and accessible to later projects.

Accomplishing this requires ongoing management of project documents in the course of a project. **RAPTR**'s document management functions support this, although guidance is required (a) to assure that **RAPTR** is not bypassed when circulating critical project documents, and (b) to decide which documents to save after tasks and projects are closed. Two general guidelines we can offer here are:

Save:

- Management documents (team charters, schedules, budgets, team rosters)
- Technical documents (process maps, analyses)
- Project results (proposals, reports)

Do not save:

- Routine administrative messages: meeting scheduling, acknowledgements.
- Anything that could be handled with a telephone call or short e-mail message.
- Earlier drafts, unless version history is important.

By communicating these guidelines to the change management team, adhering to them consistently, and adding a narrative commentary on each archived document after the project is concluded, a valuable resource will be created for later project teams.

Chapter 10: Conclusion: Areas for future research

I identify here the four core problems of change management in the Air Force that **RAPTR** uncovered. These are issues that **RAPTR** was never intended to address, yet as the tool was implemented, their importance became apparent. They are presented as critical issues confronting the Air Force today as it attempts to sustain an evolving global mission.

The concepts presented here reflect only the conclusions of the Principal Investigator, and do not represent a position of Wayne State University, its contractors, or the Air Force Research Laboratory.

10.1 Enterprise models of management and change

In peacetime, away from a warfighting or other high performance environment, command and control is dead. Large, top-down, monolithic organizations subject to effective, imperative control, if they ever existed, are certainly no more. Large-scale societal changes (rising affluence, increasingly diverse workforces) require leaders in both the military and the civilian world to govern by persuasion and coalition-building. Few officers are rising through the ranks with the idea that leadership begins and ends with issuing orders. Often, in fact, the most effective exercise of leadership is not through the chain of command.

Despite this demonstrable fact, most of the tools and methods for change management presume a top-down model of organization. When these methods fail, it is sometimes attributed to the stubbornness or "cultural resistance" of those affected by the change.

Within industry, there is an explicit effort to recognize that alliances and alignments of companies are the key elements of successful competition. Manufacturers no longer talk about their "suppliers"; instead they enforce the terminology of "trading partners". The more progressive retailers no longer have employees; instead, they have "associates." The horizontal enterprise – an alignment of multiple companies along a value-added chain – has replaced the vertical firm as the dominant economic fact.

Within the military more models are needed for this sort of cooperation and alliance. Although leadership by persuasion, coalition-building, and alliance are everyday facts of life within the military, the available models for making change presume that if just the right policy or doctrine can be crafted, if just the right order will be issued, if just the right channels of communication will be employed, when change happens, it will be the desired change. This is not guaranteed. Within the Air Force there are many examples of leadership by persuasion, leadership by coalition-building, leadership by example.

Experience with alliance partners such as the NATO forces demonstrates this. These need to be made into models, describing not an exemplary situation, but the norm for an adaptive, flexible Air Force.

10.2 Non-linear processes involving people, organizations, and technology

One characteristic of the command-and-control view of organizational process is a presumption of linearity: unidirectional influences and short, closed-loop feedback. In the real world, nothing could be further from the truth. Whether in an IPT struggling to streamline payment processes and coordinate multiple jurisdictions and levels of accountability, or in the operation of a highly automated aircraft, most real-world processes involve multiple, semi-autonomous agents, including people, organizations, and the technologies through which these act. Even the systems can be viewed as semi-autonomous agents: every user of a computer, whether on the desktop or in a cockpit, has had the experience at least once of the computer behaving in an (apparently) uncontrolled and inexplicable manner. In most cases the results are not disastrous.

An emerging science of complexity has begun to model these processes.²⁵ In contrast to this the **RAPTR** Change Management Reference Model presumed a linear, non-iterative model of change: Strategic Assessment, AS-IS Assessment, TO-BE Design, and Implementation. The IMPAC project plan presumed a similar sequence, as does nearly all of the change management literature on which the **RAPTR** model was based.

Contrast these presumptions of linearity to the view expressed by an experienced technologist, Donald N. Frey, a former VP for Engineering at Ford and former CEO of Bell and Howell, now Professor of Industrial Engineering at Northwestern University. According to Frey, the process of technological innovation

...is less of a linear flow and more of a complex ecosystem, an Everglade which, though requiring fresh infusions of new knowledge, goes through many diversions, shunts, diffusions, percolations, and budget cycles before emerging as new products. The reality is much more indirect and complex than the linear model suggests.²⁶

A military example of this can be seen in the CALS program, which is not so much a technology as an acronym that has sustained numerous initiatives for systems integration

²⁵ Cf. Epstein, Joshua M., *Nonlinear Dynamics, Mathematical Biology, and Social Science*. Reading, Massachusetts. Addison-Wesley, 1997, for example.

²⁶ Frey, Donald N., *A Technology Development Strategy for Illinois*. Supplied by the author. 1989.

through more than a dozen budget cycles.²⁷ The only failure of CALS is that its implementation has been less than linear. As we suggest here, this would be an unrealistic expectation anyhow.

Rather than creating an otherworldly standard of linearity against which a program like CALS can be judged a failure, it would be better to accept that change and innovation are non-linear processes involving people, organizations, and technology, and establish models and standards for managing what by its very nature is a chaotic, unpredictable, and not controllable process.

Such processes cannot be controlled, but they can be constrained, and their results guided in desired directions. Efforts to control that which can only be constrained are at times counterproductive.

10.3 Resources for sensemaking in complex environments

Military culture, especially away from the battlefield, contains strong reinforcements for linear thinking. Personnel systems, the entire scheme of ranking, and the nature of the military mission (with its enviable singularity of purpose) all select for and reinforce a culture that presumes a lack of complexity in the behavior of people, organizations, and technology. This culture is the product of thousands of years of fighting experience in which the warrior was required to suspend his normally complex motivations and focus solely on overcoming the enemy. Simplicity is a principle of war.

Every major trend affecting the Air Force is subverting this culture: from the changing mission (from warfighting to peacekeeping, disaster relief, and humanitarian missions, to the workforce changes noted previously, to the increasing questioning of authority in the society at large, to the substitution of increasingly complex and opaque technology for human effort and judgment. Things ain't as simple as they used to be.

It should be little surprise that in the course of the *RATPR* fieldwork we observed numerous examples of chronic degradation of sensemaking. Personnel "just going through the motions", burn-out, early retirements, the "slow roll" in response to unwelcome initiatives, youngish short-timers. These diverse symptoms, we would suggest, have as their common etiology a subtle but pervasive breakdown in sensemaking: a loss of understanding of what is going on.

²⁷ CALS: Computer-Aided Acquisition and Logistics Support (1983-1988); Computer-aided Acquisition and Logistics Systems (1989-1995); Continuously Automated Lifecycle Support (1996 - ??). Dates are approximate.

Over the last four years there has emerged an ample literature on sensemaking in organizations.²⁹ Applying the findings of this research in the USAF context would yield great dividends in morale, productivity, and organizational effectiveness.

10.4 Confronting USAF Management culture

The *RAPTR* project was occasioned by the observation that cultural issues often constrained possibilities of change within the Air Force. The findings of our fieldwork strongly supported this initial observation.

The findings of the fieldwork also support an elaboration and refinement of this initial hypothesis. Within organizations one can observe both horizontal and vertical cultures, or perhaps intra-group and inter-group cultures. The former of these is what is frequently described in the literature as "subcultures":³⁰ the cultures of occupational groups, regional groups, generational groups, and sub rosa networks ("the good old boys"). The latter consists of shared (even if differentially evaluated) understandings and expectations of intergroup relationships: how authority is to be exercised, the appropriate forms for inter-group relationships, how open can communication be between subordinates and commanders.

In our fieldwork at Warner Robins ALC we observed and documented both sorts of cultures: We observed regional culture, strongly rooted in the American South, a bureaucratic culture that is typical of government organizations, and a military culture that traces back to the military orders of the middle ages. The first of these can be considered a horizontal culture and the second a vertical culture. The third, the military culture, embraces both, although it contains two caste-like groups, officers and enlisted, each of which has its own culture. In a presentation at WR-ALC/RE we described these cultures and proposed that their mis-alignments were a source of organizational underperformance. These findings may be typical of Air Force installations that heavily draw upon the surrounding civilian workforce.

As a plausible suggestion for further investigation, we would hypothesize that in their enactment at specific locations such as Warner Robins, these three cultures are finely adjusted to each other, and that one cannot be changed without altering the others. The behavior of the civilian employees, for example, is predicated in part on what is perceived as the values and expectations of the military personnel and the civilian managers. In other words it is insufficient for a military commander to wish for or expect cultural change among the civilian workforce, without recognizing that there will have to

²⁹ Karl E. Weick, *Sensemaking in Organizations*, Thousand Oaks, CA: Sage Publications, 1995 is a recent, authoritative source.

³⁰ Cf. Martin, Joanne, *Cultures in Organizations: Three Perspectives*. New York. Oxford University Press. 1992.

be a corresponding cultural change with respect to the scheme of bureaucratic relationships, and among the military personnel.

Some aspects of military culture may be up for review. Although it would be presumptuous for a civilian to pass judgment here on any given practice, we can offer that in keeping with our proffered enterprise model of change management suggested earlier in this chapter, cultural change within an organization like the Air Force should be seen as a collaborative process, not a command initiative.

Within its mission of supporting the nation's security, the Air Force is fully committed to understanding and molding its organizational culture. This can be seen from the numerous cultural surveys, TQM initiatives, and other innovations in the personnel arena. Like industry, the Air Force sees culture as an issue in world-class performance. The **RAPTR** project added to the Air Force's toolkit for collaborative management of organizational culture. Unlike the hard tooling of aircraft, machinery, computer systems, and C3I networks, these "soft tools" – management practices, shared values, cultural norms, assessment instruments – are the next frontier in sustaining and supporting the Air Force's evolving global defense mission.