2006-09

Analysis and comparison of various requirements management tools for use in the shipbuilding industry

Clark, Eric D.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

ANALYSIS AND COMPARISON OF VARIOUS REQUIREMENTS MANAGEMENT TOOLS FOR USE IN THE SHIPBUILDING INDUSTRY

by

Eric D. Clark

September 2006

Thesis Advisor: John Osmundson
Second Reader: David M. Hicks

THIS PAGE INTENTIONALLY LEFT BLANK

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 instruction, 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 2006	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE: Analysis and Comparison of Various Requirements Management Tools for Use in the Shipbuilding Industry			**5. FUNDING NUMBERS**
6. AUTHOR(S): Clark, Eric D.			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

Requirements are the cornerstone of all contracts for products and services. If requirements are not well defined and managed, the product or service may fail to meet the customer's needs and costs may go up. This is especially true in the shipbuilding industry where the customer has many requirements. Some are clearly defined while many more are undefined. Some requirements have to be generated from the implication of other requirements while even more have to be pulled from other industry or military standards. This amounts to hundreds or thousands of requirements. Without the proper tools, managing all these requirements would be next to impossible.

This thesis investigates requirements management "Best Practices" and relate them to the needs of Systems Engineering in shipbuilding. This thesis also compares and analyzes several requirements management tools to see what may be the best fit for the shipbuilding industry in vessel design. This thesis provides recommendation of a specific requirements management tool and its suggested use.

14. SUBJECT TERMS Requirements Management, Requirements Development, Systems Engineering, Shipbuilding, Vessel Design, INCOSE			**15. NUMBER OF PAGES** 107
			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

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89) Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

ANALYSIS AND COMPARISON OF VARIOUS REQUIREMENTS MANAGEMENT TOOLS FOR USE IN THE SHIPBUILDING INDUSTRY

Eric D. Clark
Civilian, Northrop Grumman, Avondale, Louisiana
B.S., United States Merchant Marine Academy, 1988

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
September 2006

Author: Eric D. Clark

Approved by: Dr. John Osmundson
Thesis Advisor

David M. Hicks
Second Reader

Dr. David Olwell
Chairman, Department of Systems Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Requirements are the cornerstone of all contracts for products and services. If requirements are not well defined and managed, the product or service may fail to meet the customer's needs and costs may go up. This is especially true in the shipbuilding industry where the customer has many requirements. Some are clearly defined while many more are undefined. Some requirements have to be generated from the implication of other requirements while even more have to be pulled from other industry or military standards. This amounts to hundreds or thousands of requirements. Without the proper tools, managing all these requirements would be next to impossible.

This thesis investigates requirements management "Best Practices" and relate them to the needs of Systems Engineering in shipbuilding. This thesis also compares and analyzes several requirements management tools to see what may be the best fit for the shipbuilding industry in vessel design. This thesis provides recommendation of a specific requirements management tool and its suggested use.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

3SL	Structured Software Systems Limited
ACF(s)	Access Control File(s)
API	Applications Program Interface
ASCII	American Standard Code for Information Interchange
BLOBS	Binary Large Objects
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CAS3	Combined Arms and Services Staff School
CASE	Computer-Aided Software Engineering
CBS	Component Breakdown Structure
CCB	Change Control Board
CM	Configuration Management
CMS	Configuration Management System
COA	Course of Action
CONOPS	Concept of Operations
CPS	Change Proposal System
CPU	Central Processing Unit
CSV	Comma Separated Variable
CWS	Cradle Web Server
DAU	Defense Acquisition University
DBMS(s)	Database Management System(s)
DDM	Distributed Data Management
DECMAT	Decision Matrix

DFD(s)	Data Flow Diagram(s)
DoD	Department of Defense
DoDAF	Department of Defense Architecture Framework
DOORS	Dynamic Object Oriented Requirements System
ECPS	Enterprise Change Proposal System
eFFBD(s)	expanded Functional Flow Block Diagram(s)
EIA	Electronic Industries Alliance
EPS	Encapsulated PostScript
ERD	Element (or Entity) Relationship Diagram
ERS	Enterprise Requirements Suite
FFBD(s)	Functional Flow Block Diagram(s)
GHz	GigaHertz
GUI	Graphical User Interface
HID(s)	Hierarchy Diagram(s)
HTML	HyperText Markup Language
HPGL	Hewlitt-Packard Graphics Language
IEEE	Institute of Electrical and Electronic Engineers
INCOSE	International Council on Systems Engineering
IRS	Interface Requirements Specification
ISO	International Standards Organization
KPP	Key Performance Parameter
MB	MegaByte
MHz	MegaHertz
MNS	Mission Needs Statement
MS	Microsoft

NAVAIR	Naval Air Systems Command
NG	Northrop Grumman
NGSS	Northrop Grumman Ship Systems
ODBMS	Object Database Management System
ORD	Operational Requirements Document
PBD(s)	Physical Block Diagram(s)
PBS	Product Breakdown Structure
QA	Quality Assurance
QC	Quality Control
RA	Requirements Analysis
RAM	Random Access Memory
RD	Requirements Development
RDBMS	Relational Database Management System
RD&A	Requirements Development and Analysis
REQM	Requirements Management
RM	Requirements Management
RTF	Rich Text Format
RTM	Requirements Traceability Matrix
SBS	System Breakdown Structure
SCC	SPAWAR Systems Center
SDD	System Design Description
SE	Systems Engineer
SEMP	Systems Engineering Management Plan
SEPO	Systems Engineering Process Office
SPAWAR	Space and Naval Warfare

SQL	Structured Query Language
SRS	Systems Requirements Specification
SVG	Scalable Vector Graphics
SysML	System Modeling Language
UML	Unified Modeling Language
URL	Uniform Resource Locator
VCRM	Verification Cross Reference Matrix
WBS	Work Breakdown Structure
WYSIWYG	What You See Is What You Get
XML	Extensible Markup Language

EXECUTIVE SUMMARY

Over the last decade or so, the use of a Systems Engineering methodology in vessel design has become more and more apparent. This was a logical step as most other design and manufacturing industries have been employing such a process and just recently it was directed by the Department of Defense that all future acquisition contracts shall be developed using a Systems Engineering methodology (DAU, 2004). This gave most of the large shipyards interested in bidding on government contracts no choice.

Most projects of any size are created from an initial set of customer needs and these needs are analyzed into requirements. The size and complexity of the project dictate the number of requirements necessary to meet all the customer's needs and to build the right thing right. Other than maybe the Navy, Coast Guard or commercial vessels are the largest and most complex products that are designed and built. This size and complexity can equate to over 10,000 requirements. For any project to be successful, all these requirements must be managed. The cornerstone of the Systems Engineering methodology is requirements and therefore, a Requirements Management (RM) process is a necessary part of Systems Engineering.

Managing multiple thousands of requirements takes more than just a spreadsheet, but this is not a new problem. RM tools have been developed and on the market for well over a decade and there are many from which to choose. Although some of these tools are currently being used in the shipbuilding industry, this thesis researches the world of RM tools to determine what tool would best fit in the world of vessel design.

Research into the definition of RM uncovered different concepts of what should actually be included in that process. After relating this information to personal experience in vessel design, the activities of RM were determined which then served as criteria for evaluating RM tools.

This thesis compares the latest version of four RM tools considered to be leaders in the RM and Systems Engineering arenas: Analyst Pro, CORE, Cradle and DOORS. After an in-depth evaluation of each tool and a subsequent objective trade-off analysis,

the Cradle software comes out in front with CORE and DOORS almost tying closely behind. Although Cradle is found to be the best, it is also recommended that DOORS be considered for smaller shipyards or shipyards looking for a simple tool that covered the basics of RM, due to DOORS being able to operate as a stand-alone system without the added enhancements. This thesis also recommends further research into the possibility of ensuring full integration of RM tools so that more effective collaboration can be achieved as multiple organizations commonly team up during vessel design and usually the same RM tool is not being used.

ACKNOWLEDGEMENTS

I would like to thank, first and foremost, God for providing me the strength to persevere through this course of study during the many hardships in life. He has given me the faith to go on when things looked their bleakest, such as in the aftermath of Hurricane Katrina. He has been the Protector of my family during our times of need after Hurricane Katrina.

I would also like to express my sincerest thanks and appreciation to my wife, Tracey, and daughter, Emily, for the support and understanding they have provided. These last two years have taken me away from being able to enjoy spending time with my family and the last several months have been the source of constant stress. My family has had to endure living in a trailer and then in an uncompleted house since Hurricane Katrina, just so I could finish my thesis. My wife and daughter deserve special congratulations for being strong and patient until getting their husband and father back.

I would like to extend a warm-hearted thanks to my fellow COHORT 5 members; all of whom I expect to remain in contact with in the future. Many have been there for support and good times as we worked on several projects together. They will always be remembered and may our paths cross again.

I would like to especially thank David Hicks, my thesis' second reader, Henry Cook and Latan Griffin. Without them, I would never have started and finished my thesis. They all would pester me weekly, but at the same time provide the encouragement I needed.

I would like to thank the executives of Northrop Grumman Ship Systems for selecting me to participate in this program. Their faith in me will allow me to enhance the Systems Engineering department and make advancements in shipbuilding programs. My sincerest thanks also go out to the staff of the Naval Postgraduate School: Dr. John Osmundson, who supported me as my primary thesis advisor; Dr. Wally Owen and Dr. Benjamin Roberts, who have put together an outstanding program; and Lisa Diaz, who took care of everything.

Finally, I would like to thank the developers at Goda Software, Vitech, 3SL and Telelogic for providing such wonderful Requirements Management tools. I especially would like to thank Penny Knelman at Telelogic for answering the many questions I had about DOORS and Fred Manor at Vitech for assisting me with obtaining an evaluation license for CORE and other information he has provided. I would like to thank Mark Walker and Bill Baumler for trying to get me an evaluation license to Cradle. And I would like to thanks the staff at Goda Software for providing an evaluation license so quickly.

I. INTRODUCTION

A. BACKGROUND

When one hears Systems Engineering, Systems Analysis, Requirements Management (RM), or other such terms, the immediate association is with software systems. Over the past decade or more, Systems Engineering has been slowly making its mark in product development and manufacturing processes of large corporations. It has only been recently that the implementation of Systems Engineering into the Shipbuilding Industry has been evident. One driving force for this change is that the Department of Defense (DoD) has directed its implementation on all programs by DoD Directive 5000.1 which requires: “Systems Engineering. Acquisition programs shall be managed through the application of a systems engineering approach that optimizes total system performance and minimizes total ownership costs. A modular open-systems approach shall be employed, where feasible” (as cited in Defense Acquisition University [DAU], 2004). As most large shipyards are competing for the shrinking number of government contracts for Navy and Coast Guard vessels, it is prudent for those shipyards to employ a Systems Engineering methodology in order to capture these new contracts. Not only does Systems Engineering increase the prospects of additional revenue, the results of a study by the International Council on Systems Engineering (INCOSE), began in 2001, concluded that cost and schedule overruns of a project are inversely related to the amount of Systems Engineering effort applied (Haskins, 2006).

1. What is Systems Engineering?

There are many definitions of Systems Engineering, but it is not the intent of this thesis to list them all. INCOSE presents a definition as complete as any in their handbook which states:

> Systems Engineering is an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem. Systems Engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs. (Haskins, 2006)

The common theme here is requirements. Requirements are the cornerstone of any Systems Engineering methodology and the development of any new product involves requirements. Over the years, it has become evident that the management of these requirements is crucial to all ship design contracts. Therefore, keeping in line with the results of the INCOSE study, it can be surmised that proper RM will minimize cost and schedule overruns. RM will also minimize interface problems and customer dissatisfaction. As properly summed up in the collaborative work of INCOSE and the American Institute of Aeronautics and Astronautics (AIAA):

> No matter how effectively an enterprise transitions from a given solution (design) to an end product, if the enterprise does not properly define and manage the evolving requirements set, the ultimate end product will not provide stakeholders with the expected solution. (AIAA and INCOSE, 1997)

2. Characteristics of Requirements

Proper RM, however, can only be effective if the requirements are good. It will be mentioned later that part of RM is ensuring requirements are "good", but it is imperative to present the characteristics of a good requirement here because this area of concern is vitally important throughout the RM process. Good requirements are important because it doesn't matter how well bad requirements are managed, they still result in a bad product.

The characteristics of a good requirement have been described in many texts with some variety, but mostly similarity. Review of several sources and the experience of working with requirements documents have generated the following as minimum criteria that a requirement shall meet:

a. A requirement must be necessary. "Nice to haves" are not necessary. If removing a requirement creates a deficiency that can not be fulfilled by another capability, then it is necessary (Kar and Bailey, 1996).

b. A requirement must be clear and concise. The statement does not have to be grammatically correct, but it must be understandable to everyone who reads it. Every word should have a purpose (NGSS *RD&A*, 2006).

c. Each requirement should be singular. Testing to a requirement that contains several requirements is harder to meet; a failure of just one part means the whole requirement is not met (NGSS *RD&A*, 2006).

d. A requirement shall state what is required, not how it is to be met. The purpose of a requirement is not to provide a solution (Kar and Bailey, 1996).

e. A requirement must be achievable and testable. It is not enough to deliver a product defined by requirements; but to prove that the delivered product meets those requirements (NGSS *RD&A*, 2006).

f. A requirement must be unambiguous. All readers must be able to derive one and only one meaning from the requirement (Kar and Bailey, 1996).

g. Requirements must be consistent. With the thousands of requirements that can be developed, care must be taken to ensure terminology is the same throughout (DAU, 2001).

h. All requirements must be traceable to the top level requirements. If it can not be traced up through the hierarchy, it is probably not needed (NGSS *RD&A*, 2006).

i. Only requirements that contain a "shall" verb are binding and are actually true requirements. The use of "should" implies a recommendation. The use of "will" provides explanation or advises of an association to a requirement. The use of "may" indicates permission (NGSS *RD&A*, 2006). Although in many circles the use of "shall not" is frowned upon, its use is very important. There are some things that just can not be expressed in a positive statement. The use of "shall not" implies a constraint and constraints are just as binding and important as true requirements.

3. Sources of Requirements

These characteristics of good requirements are universal across all types of business. This is not so when it comes to the initial sources of requirements. This can range from requirements being initially developed from customer interviews to having formal Department of Defense documentation that state the customer's needs. In the shipbuilding industry, initial or high-level requirements come from the customer in many forms. Examples of such are the Mission Analysis Report (MAR), Operational

Requirements Document (ORD), Systems Performance Specification (SPS), and Mission Needs Statement (MNS). The information in these documents usually does not reflect the characteristics of good requirements. The RM process assists in deriving proper high-level requirements from these documents. Over the course of preliminary design, these high level requirements will be decomposed into more detailed requirements and customers will inevitably add, delete or and/or modify their needs and wants which must be translated into requirements. Furthermore, in the shipbuilding industry, there are many standards and regulations for building and classing ships from which requirements will be derived. From the start of a contract to delivery of a vessel, these requirements must be managed.

4. Database Management is the Basis of Requirements Management

In order to effectively manage requirements, a good Database Management System (DBMS) is needed. As requirements are typically stored in some sort of database, a DBMS manages and controls the database. DBMSs have been represented by different models over the years, with the following four being the most widely used: Hierarchical, Network, Relational, and Object-oriented (Satzinger, Jackson, & Burd, 2004). Of these four, only the relational and object-oriented models are used for new systems and most existing systems. A Relational Database Management System (RDBMS) organizes stored data into tables similar to those created in Microsoft Access or Excel. An Object Database Management System (ODBMS), based on the Object-oriented methodology made popular by Grady Booch, stores data as objects or interfaces.

There are several different RM tools out on the market that employ one or the other DBMSs with most of the leading tools favoring the Object-oriented methodology. INCOSE continuously surveys most RM tools, with more being reviewed as they hit the market. It is not the intent of this thesis to reiterate those results, but to analyze the results of some of those tools listed herein as they relate to all RM activities. The question to be answered is which tool is best suited for use in the Shipbuilding Industry.

B. PURPOSE

The purpose of this research is to investigate the practice of RM as it pertains to the Shipbuilding Industry and recommend a software tool to assist in meeting the needs of the customer, business unit, Systems Engineer and designer. In so doing, this thesis shall compare the potential effectiveness of tools such as DOORS, Analyst Pro 5.3, CORE 5.1 and Cradle 5.3.

C. RESEARCH QUESTIONS

This research addresses the following questions:

1. What is Systems Engineering relative to Requirements Management?

2. What is Requirements Management?

3. How does Requirements Development & Analysis relate to Requirements Management?

4. What does Requirements Management involve when it comes to vessel design?

5. Why is Requirements Management needed in the shipbuilding industry?

6. What are the benefits of effectively managed requirements?

7. How would the various selected tools implement the practice of Requirements Management in the shipbuilding industry for vessel design?

8. Which tool is best suited for Requirements Management in the shipbuilding industry for vessel design?

D. BENEFITS OF STUDY

This research provides insight to the effective use of a RM tool in the shipbuilding industry for vessel design. The benefits of this research helps promote better management of requirements which results in better ships, better customer satisfaction, a more robust process, less rework, and a better bottom line.

E. SCOPE AND METHODOLGY

1. Scope

This thesis focuses on the activities of RM from a general industry perspective and specifically as experienced by the author. The analysis portion of this thesis is limited to the four RM tools previously mentioned.

2. Methodology

The methodology used in this thesis research consists of the following steps:

a. Conduct research across various industries concerning the scope and definition of RM.

b. Conduct independent review of various RM tools through software company websites and through direct interface with the developers when possible.

c. Compare and analyze the capabilities of RM tools.

d. Discuss advantages and disadvantages of the various tools.

e. Perform a trade-off analysis of the various tools.

f. Make recommendations for further research and improvements.

Analysis of the various tools requires some usage of those tools; therefore downloading tools is necessary.

F. ORGANIZATION OF STUDY

This thesis begins with an introduction that briefly states the background of requirements and RM in shipbuilding, states the purpose and benefits of the research, gives an idea of its nature by means of questions that are answered, and describes how the research will be conducted. The next chapter discusses research performed on RM and Requirements Development & Analysis (RD&A) and the relationships between them. It concludes with what RM is for vessel design. The third chapter reviews the capabilities of the various tools selected for this research. The fourth chapter analyzes and compares the tools to the processes presented in chapter two and presents a trade-off analysis. The final chapter restates key points, makes conclusions and recommendations, and proposes further areas of research.

II. LITERATURE REVIEW

A. INTRODUCTION

A wealth of information regarding RM can be found on the internet, in text books and in published Systems Engineering manuals. As the primary purpose of this thesis is to compare RM tools, the bulk of the research conducted comes from the internet as the latest in what vendors have to offer can only be found there. In order to compare and analyze the capabilities of RM tools, it is necessary to first understand what RM is. Research into this area of the thesis comes from RM tools vendor's websites, textbooks, Systems Engineering handbooks and various requirements definitions from government organizations and standardization societies.

B. DEALING WITH REQUIREMENTS

For the most part, requirements are handled in a similar manner across all industries. Requirements are gathered, analyzed, categorized, allocated, decomposed, traced, changed, verified, validated, formatted into specification documents, and sometimes modeled. These various activities fall under different headings: RM, Requirements Development (RD), Requirements Analysis (RA) and/or RD&A. However, research has found that there is no fine line as to which activities should fall where.

1. What is Requirements Management?

There is a concept that everything having to do with requirements falls under RM. Telelogic adequately defines this scope as:

> Requirements management is the discipline of gathering, expressing, organizing, tracing, analyzing, reviewing, agreeing, changing and validating requirement statements and managing the documents that contain them with the purpose of defining and delivering the right product or service. Requirements management processes span the entire development lifecycle – from inception when requirements are gathered and defined, to the end of development when final testing is carried out with respect to the initial requirements. (Dick, 2004)

The Institute of Electrical and Electronics Engineers (IEEE) describes RM as "the process of controlling the identification, allocation, and flow down of requirements from the system level to the unit level, including interfaces, verification, modifications, and status monitoring" (as cited in Department of Energy Quality Managers Software Quality Assurance Subcommittee, 2000).

In their *Systems Engineering Handbook*, INCOSE describes RM as "the collection, analysis, and validation of requirements with all the communications and negotiations inherent in working with people" (Haskins, 2006). Although this definition is concise and does not seem to include all activities having to do with requirements, the activities listed here are actually covered in more detail in the processes defined in the next section. This would seem to imply that INCOSE views RM as an all-inclusive process and that they have broken out sub-processes only to further define them. To be more specific, INCOSE terms RM as one of the Systems Engineering "enabling" processes (Haskins, 2006). In other words, the RM process facilitates the performance of the other processes. To further the point that INCOSE views RM as an all inclusive process, the research that went into the guidebook comes from the collaborative work by AIAA and INCOSE which defines RM as the:

- Integration of requirements from multiple and separate sources…
- Analysis of these integrated requirements for ambiguities, conflicts, and omissions…
- Identification…of those requirements needing further analysis, simulations, or trade studies to establish quantitative and measurable objectives
- Conversion of analytical results into balanced, derived requirements
- Controlled evolution of requirements throughout a product's life cycle (AIAA and INCOSE, 1997)

In their *Requirements Management Guidebook*, Naval Air Systems Command (NAVAIR) describes the RM process as a framework that supports RD, evaluation of changes to requirements, verification of requirements, traceability and the capturing of decisions and rationale (NAVAIR, 1998). They intend for the process to be iterative based on the single spiral shown in Figure 1, which shows the different RM "sectors" (NAVAIR, 1998).

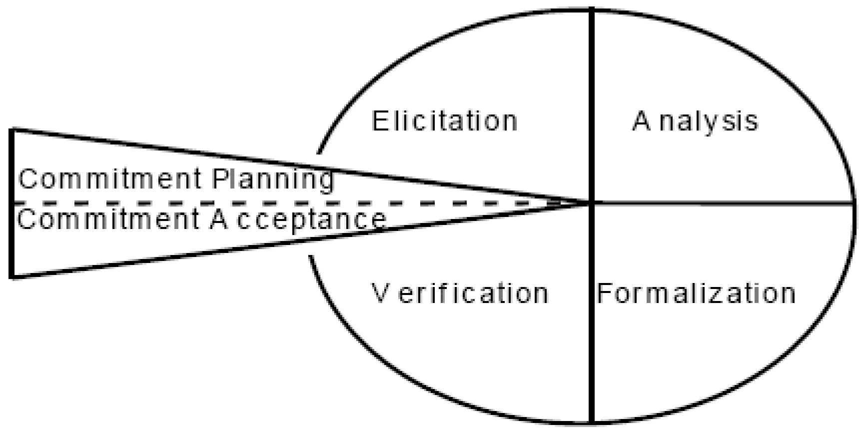


Figure 1. Requirements Management Sectors

The Office of Government Commerce in the United Kingdom defines RM as “the process of eliciting, documenting, organizing, and tracking requirements and communicating this information across the various stakeholders and the project team” (“Requirements Management,” 2006).

Another concept of RM is one that indicates RM is only part of dealing with requirements. The Defense Acquisition Guidebook has a slightly less involved outlook on what RM is by stating that RM is instituted “to (1) maintain the traceability of all requirements from capabilities [and] needs, (2) to document all changes to those requirements, and (3) to record the rationale for those changes” (DAU, 2004).

The Systems Engineering Process Office (SEPO) at SPAWAR Systems Center (SCC) in San Diego describes RM as follows:

> Requirements Management (REQM) involves applying management disciplines to the requirements development process. REQM involves establishing and maintaining an agreement with the customer on the requirements for a project, managing changes to requirements, ensuring consistency between the requirements, the project plans and work products, and maintaining bi-directional traceability for requirements and work products. (SCC SEPO *RM*, 2005)

It would appear that SPAWAR’s definition would seem to fit under the first concept of RM since they seem to indicate that RD is incorporated under RM. This is similar to how INCOSE views RM; however, it will be seen that they do include other tasks under their RD process that don’t directly correlate to the RM definition.

2. What is Requirements Development & Analysis?

According to the Defense Acquisition Guidebook, the RD "process takes all inputs from the relevant stakeholders and translates the inputs into technical requirements… [RD] encompasses the definition and refinement of system-, subsystem- and lower-level functional and performance requirements and interfaces to facilitate the design…" (DAU, 2004).

Instead of providing a definition for RD&A or RD, INCOSE defines a Requirements Definition process and an RA process. The Requirements Definition process purports "to elicit, negotiate, document, and maintain stakeholders' requirements for the system-of-interest within a defined environment" (Haskins, 2006). Some of the activities under this process are identifying stakeholders, identifying constraints, analyzing requirements for "good characteristics", establishing a traceability matrix, and recording requirements. The *Systems Engineering Handbook* goes on to describe that the purpose of the RA process "is to review, assess, prioritize, and balance all stakeholder and derived requirements (including constraints); and to transform those requirements into a functional and technical view of a system description capable of meeting the stakeholders' needs" (Haskins, 2006). Many of the activities listed under these two INCOSE processes seem to align with the definition of RM from the AIAA and INCOSE collaborative work, obviously with good reason.

Then there is SCC SEPO which defines RD as a process which "involves the stakeholders' requirement-driven view of desired services into a technical specification…" (SCC SEPO *RD*, 2005). The tasks included in this process are: commitment/planning, elicitation, analysis, formalization, verification, and commitment/acceptance. These **are** the same tasks or "sectors" mentioned in NAVAIR's documentation. SCC SEPO has appropriately referenced much of the *Systems Engineering Guidebook* and even uses the same figure shown above, although it is titled "Requirements Development Spiral."

3. Putting It All Together

There seems to be some disparity on what RM is supposed to cover, with a heavy emphasis on the all-encompassing concept. Neither concept is incorrect if properly defined. If the concept is that everything having to do with requirements falls under RM, then one should ensure that all is covered. INCOSE, NAVAIR and even SCC SEPO have done just that. If the concept is that RM should be separate from RD, RA and/or RD&A, then the line of demarcation should be clear. Here, also, interfaces between the various requirement processes should be clear. The Defense Acquisition Guidebook does an excellent job with keeping RM separate and distinct from RD. The RM definition is simple and straight forward.

Should it matter if there is a separate RM and RD process in vessel design? As requirements tools are becoming more centered on the whole Systems Engineering process, all requirements activities mentioned above, whether listed in one or more processes, can be performed with one tool. Why not, then, have one over-arching process? INCOSE seems to support this approach.

C. REQUIREMENTS MANAGEMENT IN VESSEL DESIGN

The question to be answered here is: What should the RM process include when it comes to vessel design? As INCOSE is a leading recognized influence in Systems Engineering, the answer would appear to be clear. RM should include the gathering of requirements from source documents and the concurrent management of those source documents, the analysis of the requirements for integrity and completeness, the allocation and decomposition of requirements, the linking of requirements to design, the generation of specification documents at different phases of design, and the management of changes to each and every single requirement. As modeling requirements is fast becoming a useful method of graphically depicting requirements, this may come in handy for certain aspects of vessel design such as in Command and Control. Another aspect of Systems Engineering is the development of a System Architecture which should be directly in tune with the requirements. As such, RM should also ensure proper interfacing of requirements with the architecture during the development of the architecture. This will provide good traceability of requirements to the design.

D. BENEFITS OF REQUIREMENTS MANAGEMENT IN VESSEL DESIGN

Vessel design also experiences the benefits of RM. RM holds down costs by uncovering errors early. It has been presented in IEEE’s text *Software Requirements Engineering* that, requirements errors found at the requirements stage cost only about one-fifth of what they would cost if found at the testing stage and one-fifteenth of what they would cost after the system is in use (as cited in Department of Energy Quality Managers Software Quality Assurance Subcommittee, 2000). This same conclusion can be made of RM in the shipbuilding industry. It is more costly to discover errors during system testing than during the detailed design phase, which is more costly than finding them during preliminary design. RM improves customer satisfaction by giving the customer the assurance that their objectives are being met. This is done by organizing and tracing the requirements. Tracing the requirements also gives the Systems Engineer (SE) the ability to manage and analyze the impact of changes. Properly managed requirements lead to better specification documents that the customer will be approving. Through the processes of requirements reviews and baselines, RM also gives the SE the ability to track a project's or contract's progress.

E. CHAPTER SUMMARY

Requirements Management means different things to different people or organizations. All over the world, some consider RM to entail the complete lifecycle of a requirement, while others consider RM to include only changes to a requirement. It is hard to determine which is the best approach and maybe it works well either way depending on the organization.

Regardless of the approach, the benefits of managing requirements from conception to deliverance are the same. Proper management of requirements decreases development costs (time and money) in later design phases and the customer ends up being a returning customer or at least a good recommendation for future work. In other words, "Effective requirements management helps to control quality, cost, organization and schedule thus substantially improving your odds of a successful project" (Halbleib, 2006). The goal is to build the right thing the right way, on time, every time.

This chapter covered the many activities of RM and associated most, if not all, of them to what was needed in vessel design. The complexity of vessels and, more importantly, the requirements imposed on government contracts by DoD are leading the large shipbuilder in this direction. Commercial projects may get away with some lesser involved process because of the lack of government intervention, but a commercial vessel exhibits just as much complexity as government vessel these days. Therefore, the RM tools described and analyzed in the following chapters will be applicable for handling the RM tasks of vessel design regardless of the type of vessel.

THIS PAGE INTENTIONALLY LEFT BLANK

III. TOOLS REVIEW

A. INTRODUCTION

As described earlier, the process of managing requirements can be a daunting task, especially when the number of requirements developed is several hundred or several thousand and the changes to those requirements numbers just as many. There are many tools on the market that assist in managing requirements all the way from just storing requirements in a database to actually developing requirements from various documents while performing other functions as well. INCOSE has an on-going survey of RM tools on their website (http://www.paper-review.com/tools/rms/read.php), currently totaling about 30 different tools. The website is set up to allow the RM tool vendor, who desires to be included in the survey, to answer an extensive set of questions, as shown in Appendix A, based on the performance of their offering. It is up to the vendor to be truthful and it is up to the users of the survey to be cautious of the information given as INCOSE does not verify its validity. Occasionally, the Tools Database Working Group, the creators of the survey site, will correct any exaggerations.

It is impossible to consider every RM tool out on the market in the scope of this research. As such, only a select few have warranted further analysis. These include Analyst Pro 5.3, CORE 5.1, CRADLE 5.3 and DOORS. The initial set of tools included RequisitePro and Envision VIP. The reason for this is that the initial set of tools was chosen based solely on the results of the INCOSE survey (a higher number of "Full" capability indications). The final set of tools presented in the following paragraphs was decided upon based on how informative the tool's website was. First impressions of a tool's website and how adaptable to the needs of RM in vessel design it appeared to be were key factors of a tool being chosen for further research. Each of the following paragraphs will describe one of the specific tools and its key capabilities.

B. ANALYST PRO 5.3

The Analyst Pro tool has been developed by Goda Software, a Virginia-based enterprise and systems development company incorporated in 2000. Analyst Pro is concerned with enterprise lifecycle management. With Analyst Pro, Goda Software mainly targets software development companies, as many of these RM tool vendors do, but also provides for systems development. Any product, whether it is a ship or a software program, is considered a system. Analyst Pro provides a concentration on requirements management based on a commercial repository solution[1]. Analyst Pro is scalable and will work with any development process including the Waterfall and Spiral methodologies (Analyst Pro, 2006). The network client server can handle up to 250 concurrent users. The information presented below has been retrieved from Goda Software's website at http://www.analysttool.com/, from use of an evaluation copy of Analyst Pro 5.3, and from the Users Guide.

1. Capabilities

Analyst Pro has many capabilities regarding projects, requirements, repositories, diagrams, traceability, use cases, import/export, database management, change and configuration management, workflow, and report generation. All modules and features can be accessed through the main menu bar shown in Figure 2 or by clicking on buttons or drop down menus from any of the modules.

Users can create projects, project templates, add/delete users, assign users to specific projects, and set various project attributes by selecting the 'Project' module. Project details and options can be entered and changed in the window and tabs shown in Figure 3. Various project attributes can be entered in the window and tabs shown in Figure 4.

In the 'Requirements' module, the user can specify, track, manage and analyze requirements. A unique identifier is automatically generated for each requirement. Links can be created between requirements and to other objects, such as documents, files, models and diagrams in the repository. Change history of requirements can be easily tracked and associated documentation can be generated. Figure 5 represents a typical Requirements window with tabs for the various requirement types, which are editable, and requirements analysis. The requirements editor allows for easy creation of hierarchical specifications and the printing of the same in document form. Analyst Pro

[1] Analyst Pro does not claim adherence to either an Object-oriented or Relational database, but seems to favor a Relational type.

also automatically tracks changes and has diagramming editors for the creation of object-oriented Unified Modeling Language (UML)[2] Use Cases[3].

The ‘Repository’ module allows for the storage of any non-requirements information in the form of documents, diagrams, models and other files created by external tools. This module permits the sharing of this information among users and the information can be linked to actual requirements.

The ‘Traceability’ module allows the user to trace several key issues from the impact of changes to requirements to determining if there is a test memo assigned to every applicable requirement. Sample views of two features of the traceability analysis are shown below. Figure 6 shows a traceability matrix for the design requirements. Here direct and indirect relationships can be shown among all links. From the matrix view, users can generate reports, perform impact analysis and create graphical representations of links. Figure 7 shows traceability views for all requirements in hierarchical format. This gives the user the graphical representation of the relationships among the requirements.

The ‘Output’ module allows the user to create and print several requirements documentation, requirements history and requirements graphs. Figure 8 shows the initial ‘Documents’ screen offering the user the ability to chose tabular documents, requirements documents (with or without attributes), project details documents, and objects list documents. The last two listed document types both require the use of an Export Wizard, shown in Figure 11, which allows output to Microsoft (MS) Word, MS Excel, Adobe Acrobat, HyperText Markup Language (HTML) file, MS Access Database or a simple text file. Output of the previous document types is performed by the use of a Generate Document button. Another option in the ‘Output’ module allows users to print Requirements History and Changed Requirements reports. Figure 9 shows an example of the ‘Changed Requirements’ report setup screen. The user can also create and export pie graphs or bar graphs, as shown in Figure 10, which gives an overview of requirements status.

[2] A non-proprietary specification language for object modeling (“Unified Modeling Language,” 2006).

[3] A technique for capturing functional requirements of systems and system-of-systems (“Use case,” 2006).

2. Screen Shots[4]

Figure 2. Analyst Pro 5.3 Main Menu Bar.
Access to all Modules and Features

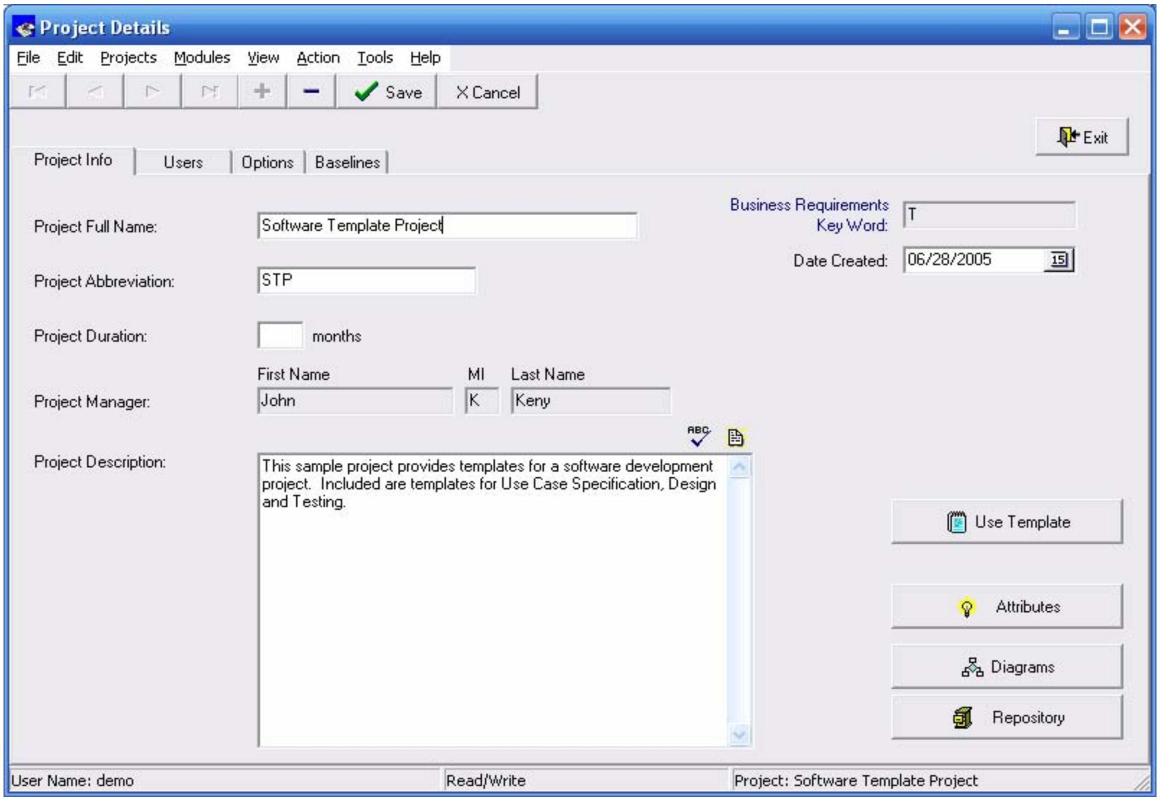


Figure 3. Analyst Pro 5.3 Project Details Info Window.
Initial Project Set-up and Entry Screen

[4] All screen shots were created from the actual Analyst Pro software.

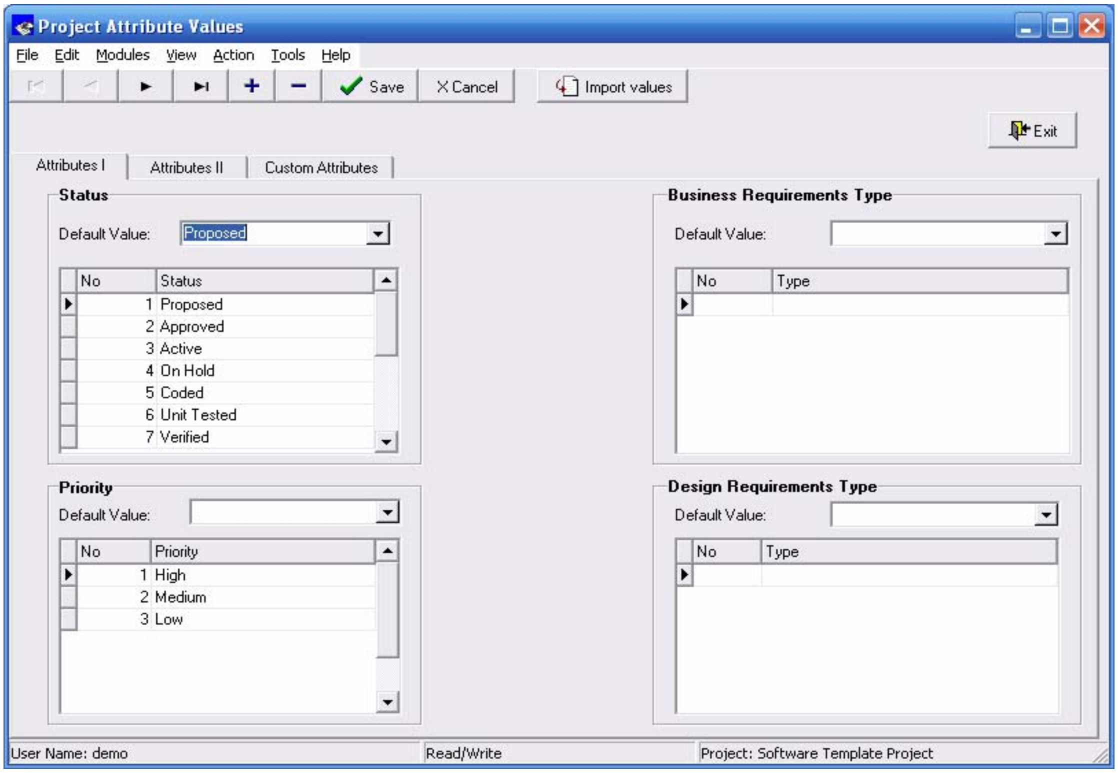


Figure 4. Analyst Pro 5.3 Project Attributes Window.
Entry Screen where all Project Attributes can be entered

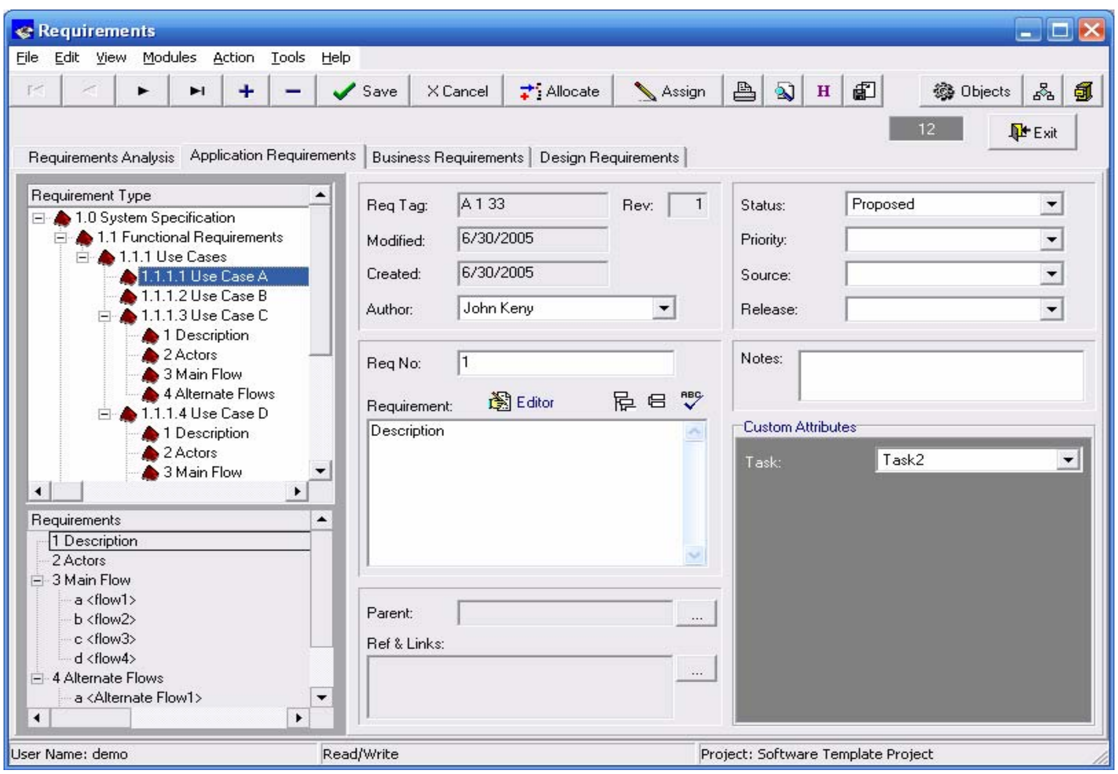


Figure 5. Analyst Pro 5.3 Requirements Window.
All types of Requirements can be viewed and edited

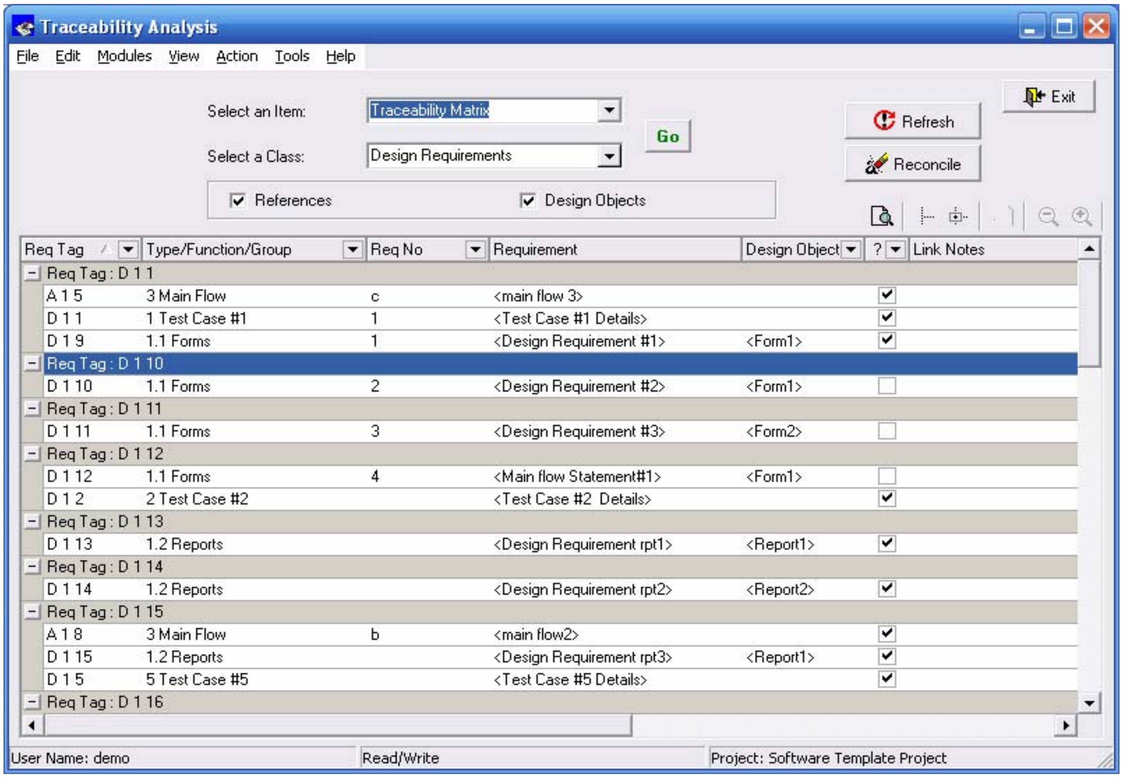


Figure 6. Analyst Pro 5.3 Traceability Analysis Matrix Window. Direct and Indirect Relationships are shown among Design Requirements

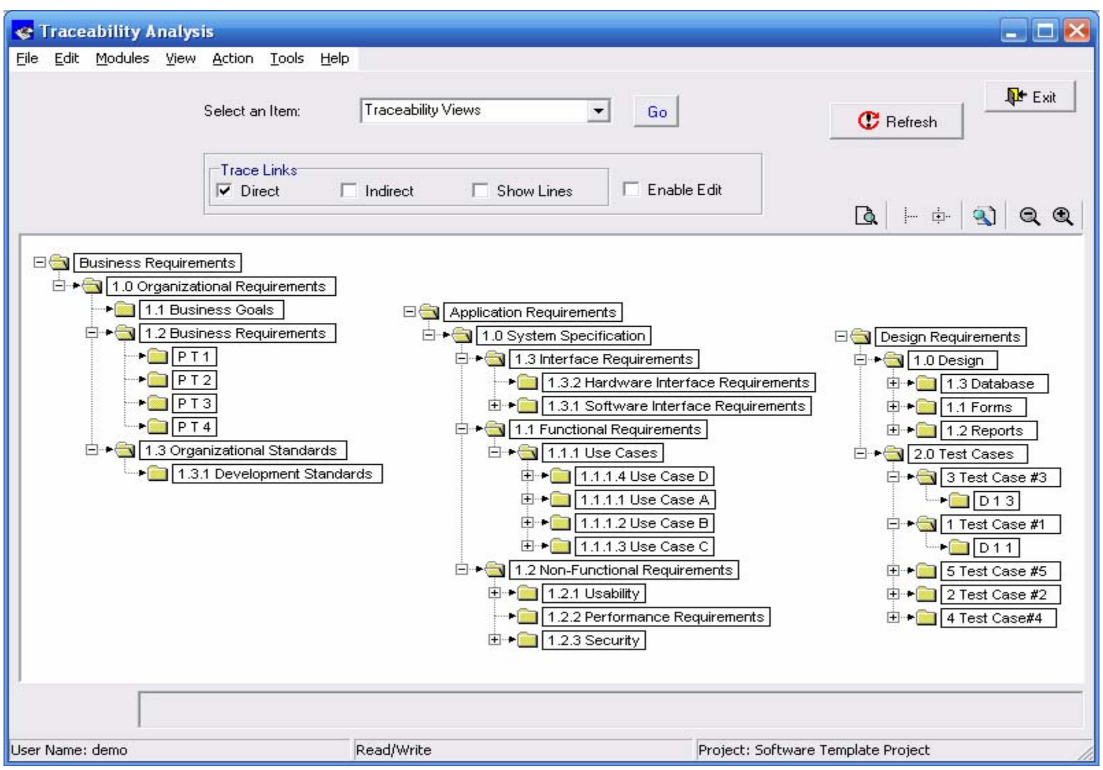


Figure 7. Analyst Pro 5.3 Traceability Analysis Graphical View Window. The Hierarchy of Links is shown for all Requirement Types

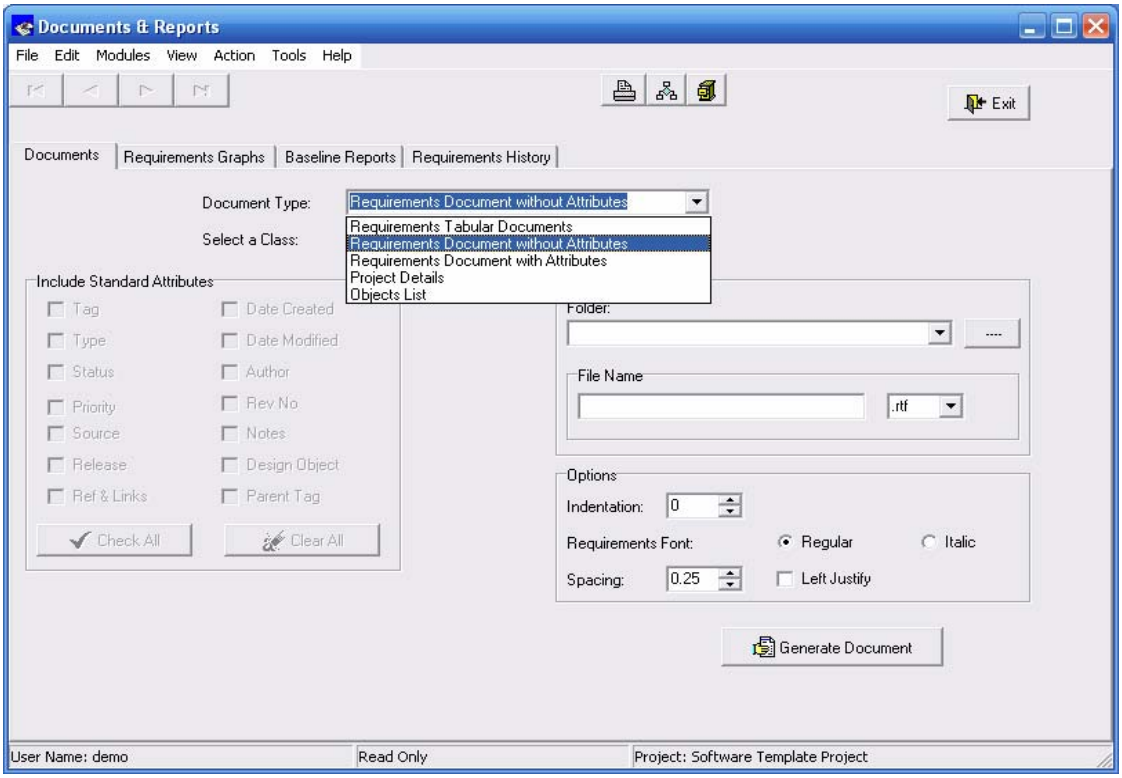
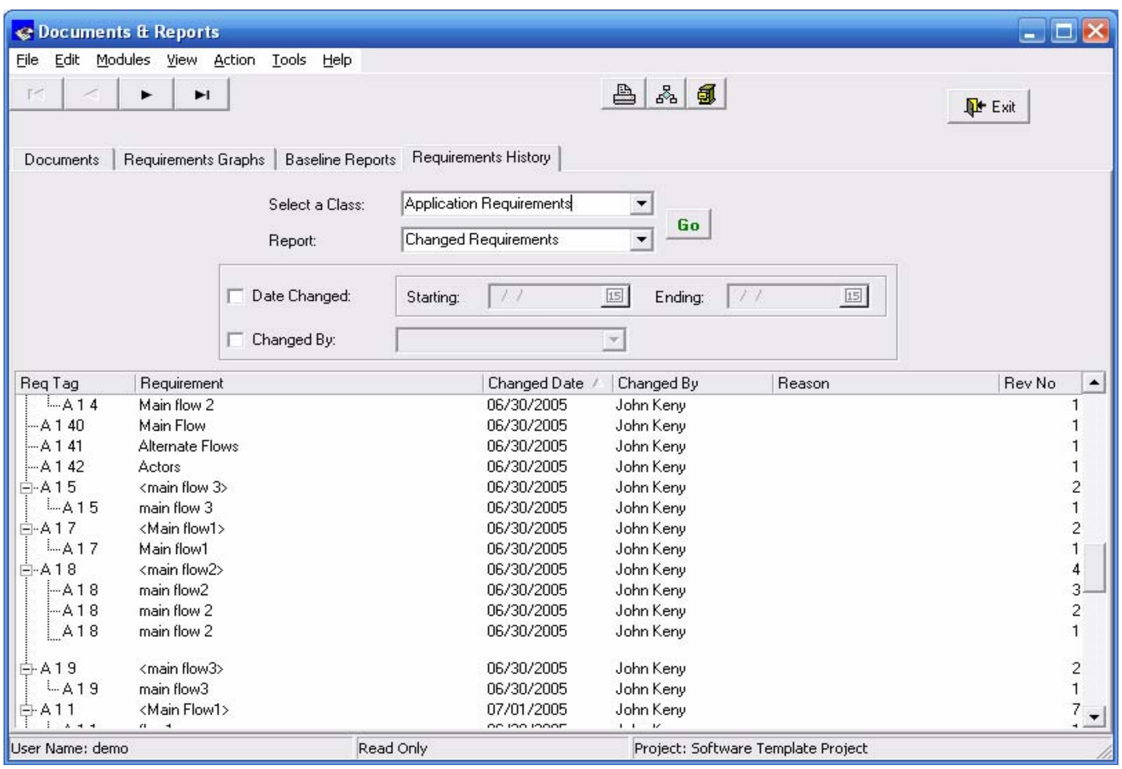


Figure 8. Analyst Pro 5.3 Documents Window.
Initial Documents & Reports Screen where type of Documents to be processed is selected



Req Tag	Requirement	Changed Date	Changed By	Reason	Rev No
A 1 4	Main flow 2	06/30/2005	John Keny		1
A 1 40	Main Flow	06/30/2005	John Keny		1
A 1 41	Alternate Flows	06/30/2005	John Keny		1
A 1 42	Actors	06/30/2005	John Keny		1
A 1 5	<main flow 3>	06/30/2005	John Keny		2
A 1 5	main flow 3	06/30/2005	John Keny		1
A 1 7	<Main flow1>	06/30/2005	John Keny		2
A 1 7	Main flow1	06/30/2005	John Keny		1
A 1 8	<main flow2>	06/30/2005	John Keny		4
A 1 8	main flow2	06/30/2005	John Keny		3
A 1 8	main flow 2	06/30/2005	John Keny		2
A 1 8	main flow 2	06/30/2005	John Keny		1
A 1 9	<main flow3>	06/30/2005	John Keny		2
A 1 9	main flow3	06/30/2005	John Keny		1
A 1 1	<Main Flow1>	07/01/2005	John Keny		7

Figure 9. Analyst Pro 5.3 Requirements History Window.
The Report Setup Screen showing a Preview of Changed Requirements

Figure 10. Analyst Pro 5.3 Requirements Graphs Example.
A Bar Graph showing the Status of all Requirements

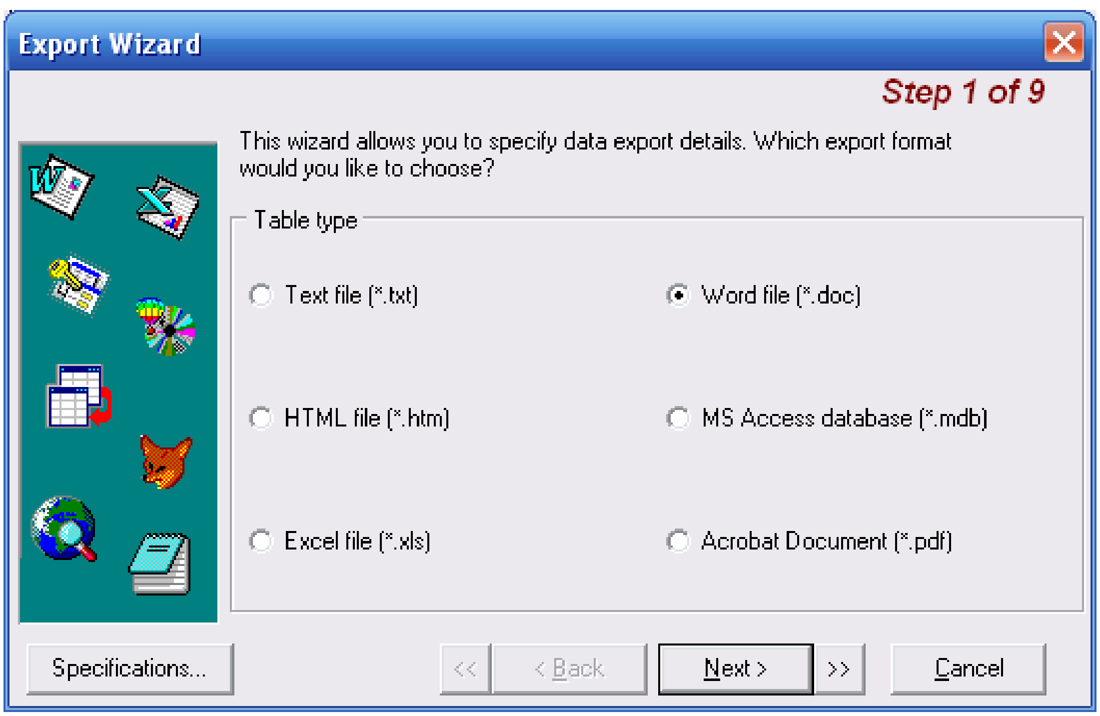


Figure 11. Analyst Pro 5.3 Export Wizard Window.
Initial Export Window Screen allows selection of Export format

C. CORE 5.1

The CORE tool has been developed by Vitech Corporation, another Virginia-based company founded in 1992. Vitech provides both a workstation version for the single user and an Enterprise client-server version of CORE. Either version utilizes an ODBMS to provide a collaborative solution for the synchronization of requirements, analysis and architecture. CORE covers the whole Systems Engineering methodology: the analysis, decomposition, allocation and validation of system requirements; the definition of system functional and operational behaviors; the definition of system architecture including internal and external interfaces; and the definition of system verification and validation requirements (Fluhr, 2006).

The information presented in this section about CORE has been retrieved from Vitech's website at http://www.vitechcorp.com/, actual usage of CORE Workstation 5.1 and as noted.

1. Capabilities

CORE is supported by several key technologies. At the center of the system is the repository which handles multiple users adding, deleting, changing, and reviewing information which culminate in a system specification. This centralization ensures all users are working from the same baseline and provides consistency throughout the product development. In order to eliminate ambiguity in defining a system, CORE uses a System Definition Language consisting of "elements" grouped into one of several classes; "relationships" which define links between two elements; "attributes" which provide further description of "elements"; "attributes on relationships" which provide further description of "attributes"; and "structures" for behavior notation (CORE: Guided Tour, 2005). CORE employs a dynamic diagram generator which ensures changes made in the repository are reflected in the diagrams and vice versa. CORE comes preloaded with several diagrams including the Element Relationship Diagram (ERD)[5], Physical Block Diagram (PBD)[6], Functional Flow Block Diagram (FFBD)[7], N^2 Diagram[8], and

[5] Displays the element and its relationships to other elements (CORE 5, 2005).

[6] Shows composition and connectivity of physical architecture (CORE 5, 2005).

[7] Shows functional flow including control logic (CORE 5, 2005).

[8] Displays functions and their internal and external inputs/outputs in a matrix format (CORE 5, 2005)

Hierarchy Diagram (HID)[9]. CORE also has automatic document generation. Any information can be extracted from the repository into any format desired from the simple query to the formal specification document. The use of scripts allows the user to automate the information retrieval process.

The main starting screen is the Element Browser view of the CORE Project Explorer, as shown in Figure 12. From here the user can navigate throughout the database. The various panes are standard, but they can be resized and shown or hidden as desired. The information within the panes is completely customizable to meet the user's needs. The Project pane consists of a standard set of database classes, to which more can be added. The project pane also includes information on the project's schema, which is also customizable. The Element pane shows the elements of the selected class. The Element Property Sheet gives all the details of a selected element including any assigned relationships. There are well over 100 possible relationships depending on the class being viewed. Relationships enhance CORE's traceability feature by making it easy to locate unfulfilled requirements and unresolved issues. The user also has the option to open an Element Table view, as shown in Figure 13. This allows users to view, update and add elements in spreadsheet format similar to MS Excel.

From the Project Explorer, the menu bar in the bottom right gives the user several viewing choices. Figure 14 shows how the hierarchy view of requirements traceability might look. Much information can be seen in this view: the type of element at the bottom of each block, the element name, the relationship between the elements, etc. The black dot in the upper right corner of a block indicates that the element is elsewhere on the diagram. A black dot in the upper left corner of the block indicates that this element has further traceability. An N^2 diagram[10] can also be viewed in Project Explorer like the one shown in Figure 15. Other system or physical interfaces can be represented by PBDs. All diagrams can be viewed full screen for better visibility and the scale factor, among other things, can be modified. All diagrams are automatically generated from the information in the repository. Double-clicking on any element block will open up a

[9] Graphically displays several layers of relationships between elements on a single diagram such as functional, physical, and traceability hierarchy views (CORE 5, 2005)

[10] CORE uses one type of N^2 diagram showing only the functional interfaces, whereas another diagram (or chart) commonly used in vessel design shows physical, technical and logical (functional) interfaces.

separate Element Property Sheet, as shown in Figure 16, allowing the user to make changes without leaving the diagram view. Any changes will be automatically reflected in the diagram once saved.

CORE also makes an easy job of manually extracting elements from source documents. By invoking the Element Extractor, shown in Figure 17, and then loading an external document file, the user can just highlight the text in the source document and click on the transfer buttons (name, number, etc.) in the right pane. Relationships, targets and attributes can be assigned at this point also in order to establish traceability.

Other features noticeable in Figure 12 in the Project pane are the classes for "Issues", "Risks" and "Verification Requirements". All these can be manually entered, updated and related to other elements just like any other element. CORE can generate specific documents highlighting issues and/or risks, and can generate a Test and Evaluation Plan. All interfaces between elements are captured in the "Links" class.

2. Screen Shots[11]

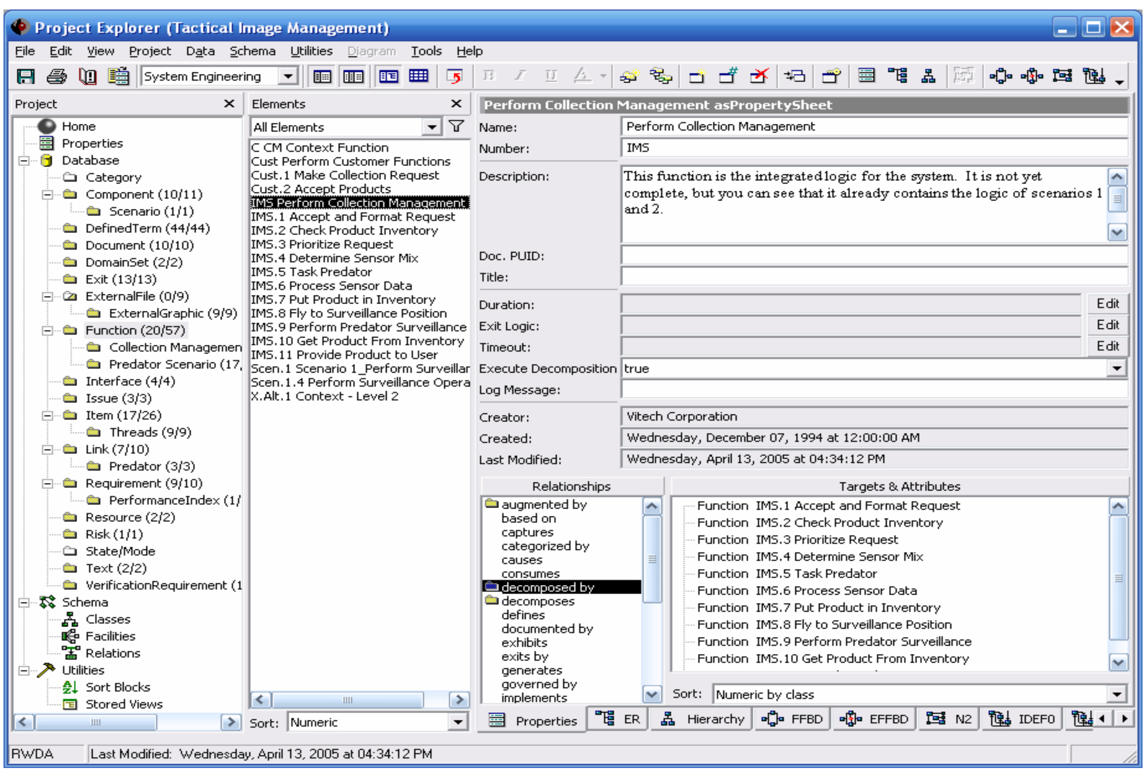


Figure 12. CORE Element Browser View.
Navigation from this Screen to any part of the Database is possible

---

[11] All screen shots were created from the actual CORE software.

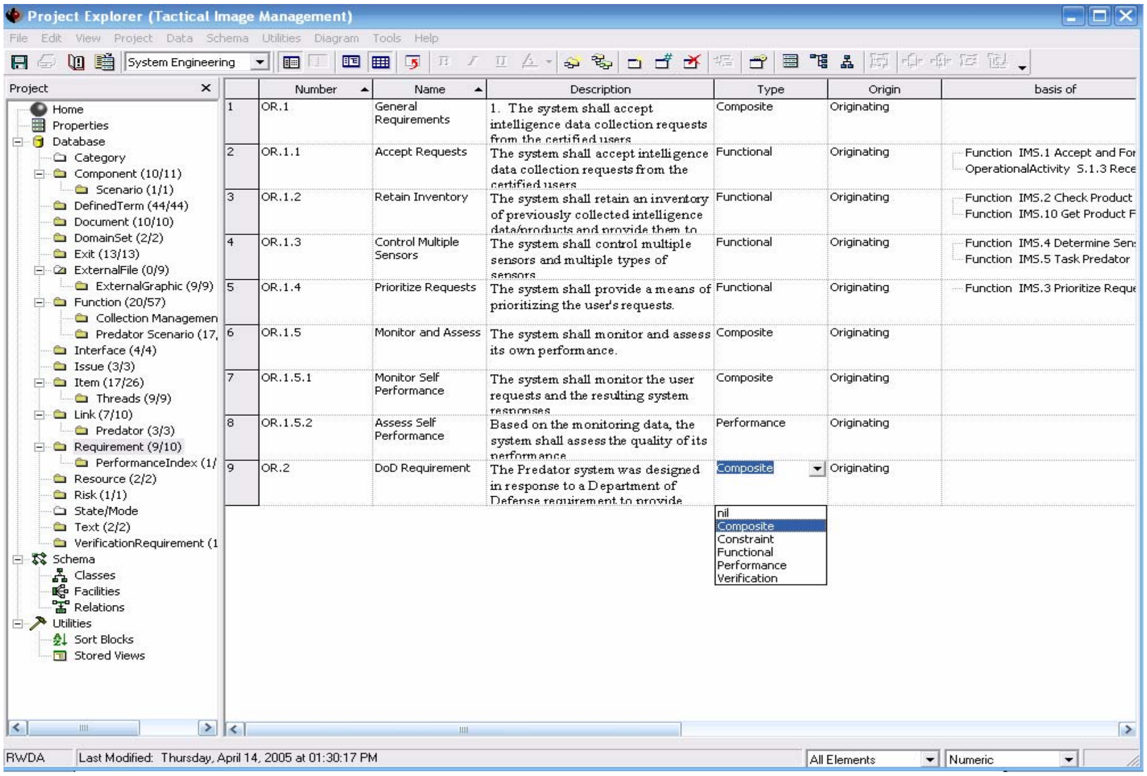


Figure 13. CORE Element Table View.
A Spreadsheet View of the Elements of a Specific Class

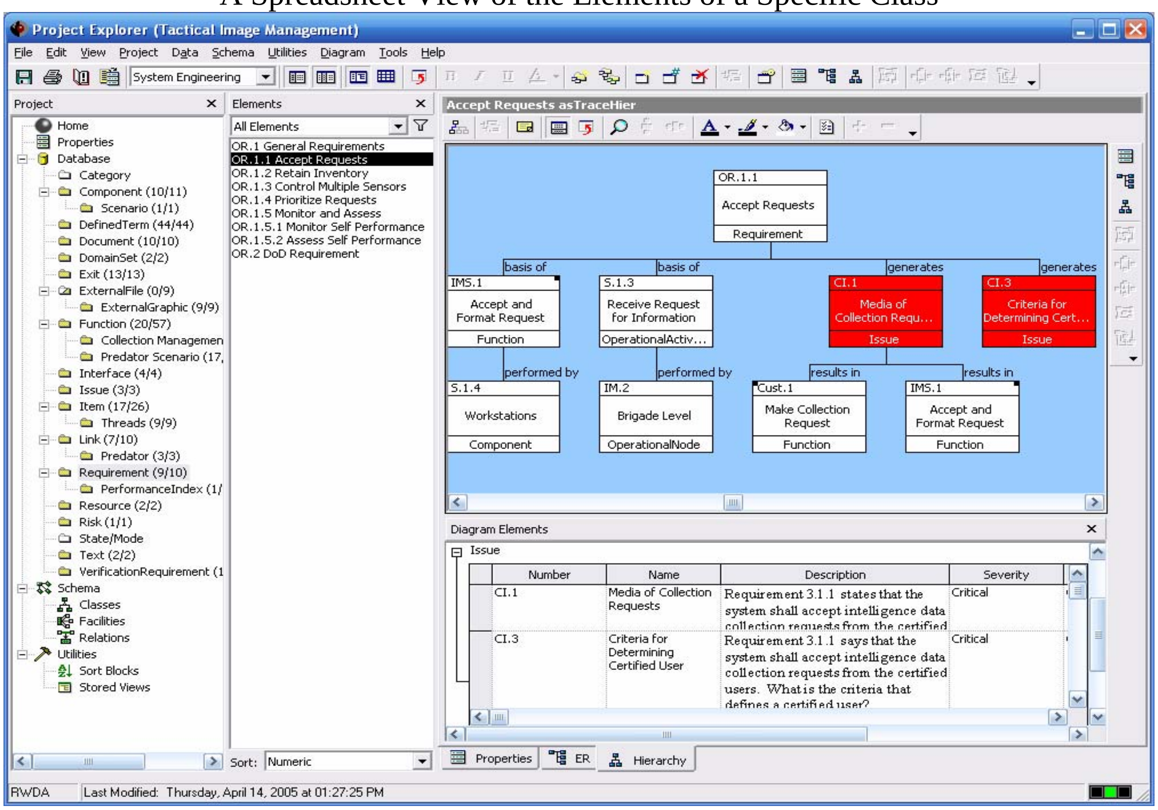


Figure 14. CORE Project Explorer showing Hierarchy View.
Much Information can be gathered from this Hierarchical Tree

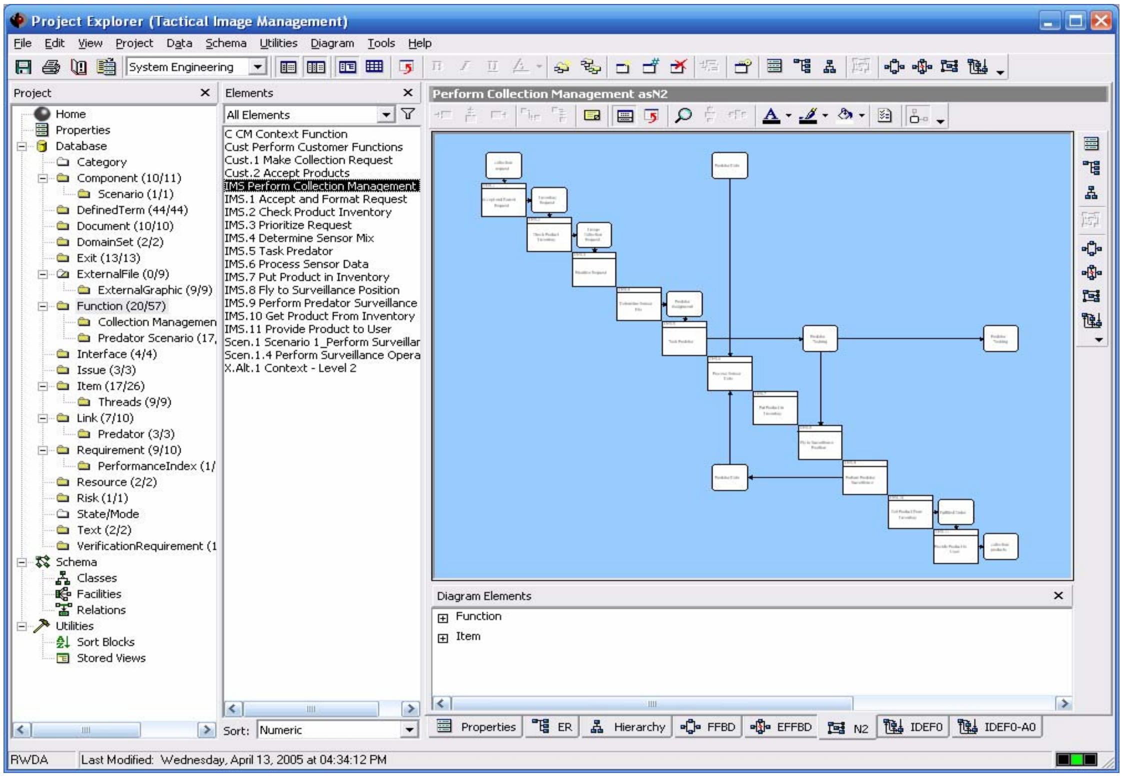


Figure 15. CORE Project Explorer showing an N^2 Diagram.
Selecting the N2 tab shows the Functional Interfaces between Requirements

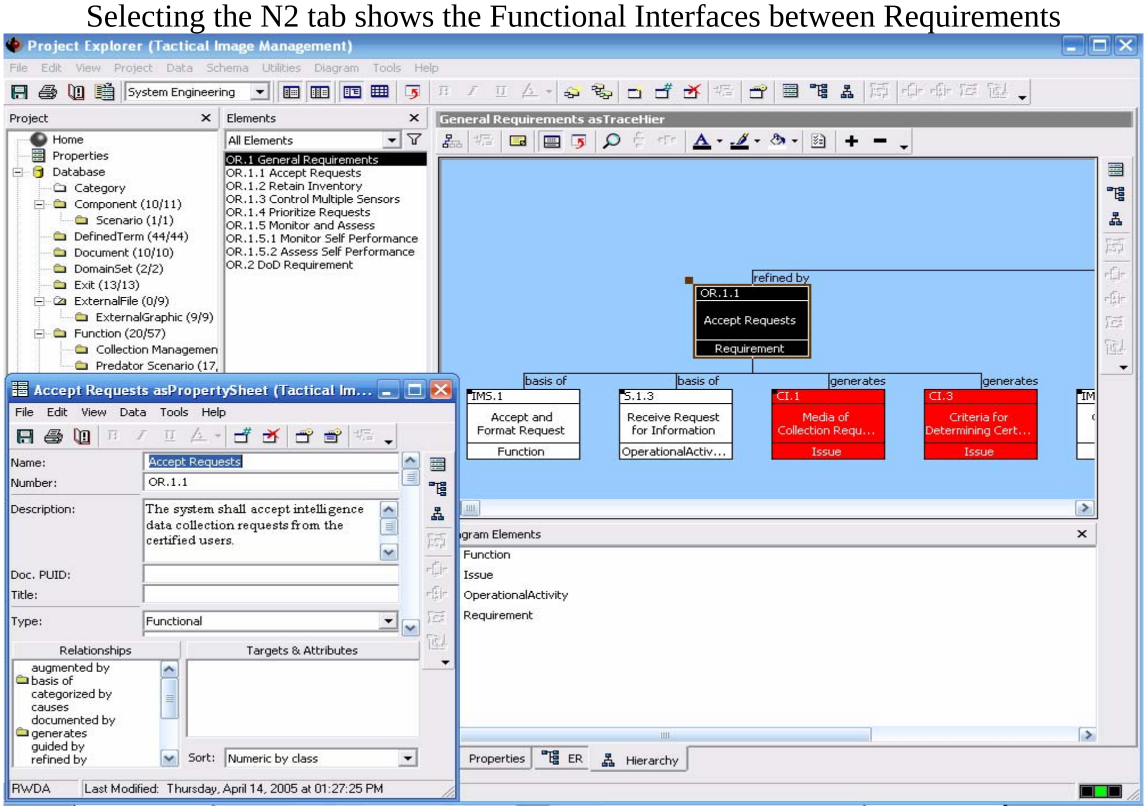


Figure 16. CORE Project Explorer with Element Property Sheet.
Double-clicking a block opens a Requirement Property Sheet for editing

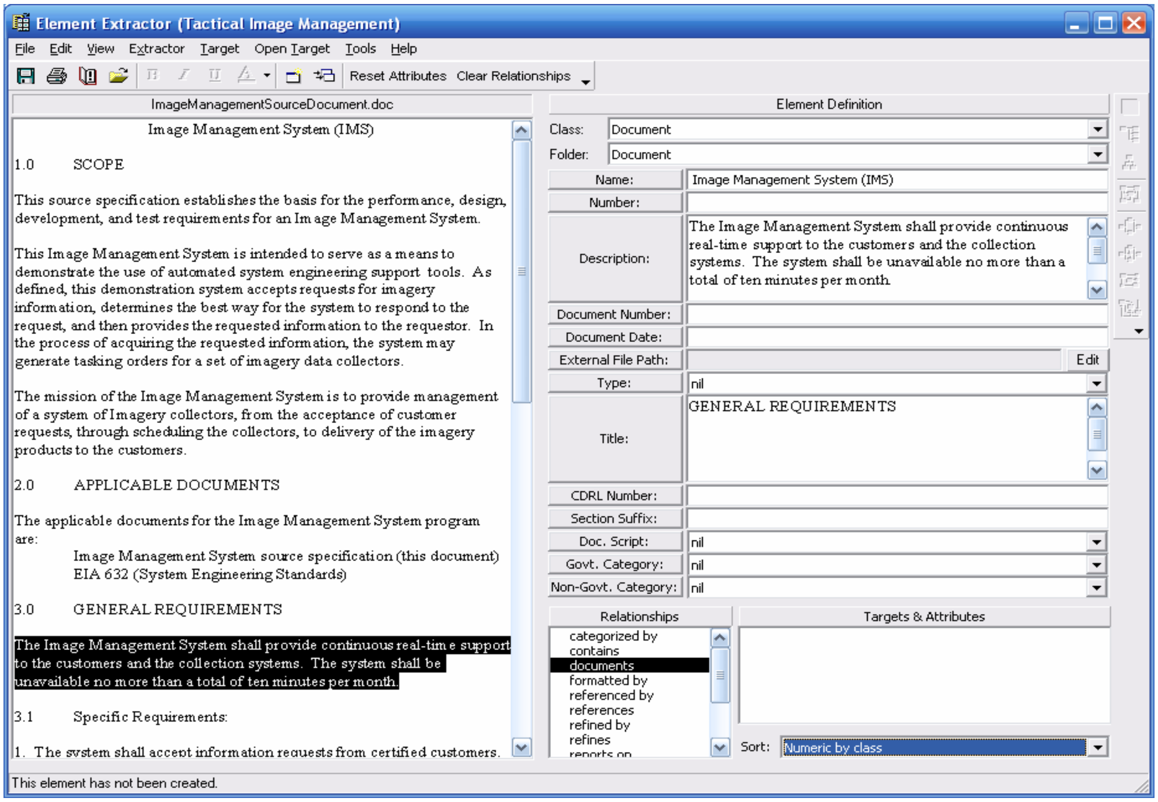


Figure 17. CORE Element Extractor Window.
The Left Pane shows the Requirements Document; the Right Pane manages the Extracted Requirements

D. CRADLE 5.3

The Cradle tool has been developed by Structured Software Systems Limited (3SL), an England-based company specializing in systems engineering processes and products. Cradle is an extensive requirements management and systems engineering tool able to support multi-users (up to 8,192) and multi-projects through the use of an RDBMS. Cradle consists of 10 modules that can be used in conjunction with each other or as separate entities. These modules are: Cradle-PDM, Cradle-WRK, Cradle-REQ, Cradle-MET, Cradle-SYS, Cradle-PERF, Cradle-SWE, Cradle-DOC, Cradle-WEBP, and Cradle-WEBA. Cradle's Toolset will invoke seven of the modules, not including WRK, WEBP and WEBA, and is geared toward the core systems engineers. Cradle-WRK, Cradle's Workbench, invokes the same seven modules and is geared toward the rest of a project's team members and any external reviewers. WEBP and WEBA are Cradle's Web Publisher and Web Access modules giving users the capability to extend a project into the World Wide Web in order for occasional and remote users to have review access.

Regardless of the number of modules used on any given project, Cradle will recognize each module invoked automatically so there is no need transfer data between modules.

As this thesis relates to RM, only those modules pertaining to requirements and the enhancement of the RM process will be presented. The information presented about Cradle in this section has been retrieved from 3SL's website at http://www.threesl.com/. Actual use of the various modules of Cradle was, unfortunately, not possible. However, this should in no way adversely bias the subsequent comparison of tools in the next chapter.

1. Capabilities

Cradle is a complete "cradle to grave", full development lifecycle supporting tool which is perfect for projects dealing with all phases of RM, system design and analysis, architecture, and business process modeling. The modules mentioned below are suitable for RM, systems analysis, architecture design, functional allocation, metrics, risk management, document generation, configuration management, and workflow management. Activities such as creating a Work Breakdown Structure (WBS), capturing requirements with parsers and MS Word/Excel plug-ins, formatting the requirements to generate useful reports, creating UML and functional models to which to allocate requirements, and allocating requirements and models to architecture can be accomplished with Cradle (Cradle Overview, 2005).

Cradle is made up of projects, where all work is done. Cradle-PDM, the project data management module, is the infrastructure for all the other modules. Everything about a project is defined in this module. All other modules will inherit the properties of Cradle-PDM, respective to each project (Walker, 2006). Each project has its own database. Each database contains items, of which there can be many item types. Each item contains many customizable attributes. The items represent requirements, architecture components, issues, risks, processes, functions, classes, etc. All of this, including link types, user profiles, and rules, make up a project's "schema" (Cradle Overview, 2005).

The Cradle-REQ module covers the RM domain. This module can maintain any type of requirement, including functional and non-functional, user, operational and system. As Cradle can link to any application and has built-in interfaces to such products as MS Word, MS Excel, MS PowerPoint, Adobe Acrobat, Adobe FrameMaker, Claris FileMaker, Adobe PhotoShop and Adobe Illustrator, it is easy to capture requirements through automatic parsers or simple data exchange. Figure 18 shows a typical example of capturing requirements by directly launching the "CRADLE Capture" screen right from MS Word (Cradle REQ, 2006). Once requirements are captured, Cradle provides many tools to search for issues such as ambiguity, contradiction and duplication. Cradle provides tools for impact analysis and traceability and can show the requirements in various formats such as trees, lists, tables or hierarchies. Cradle-WRK, discussed later, provides the ability to show some of these formats in one screen, as presented in Figure 19, and they can be viewed separately within Cradle-REQ (Cradle REQ, 2006). Requirements can contain graphics, videos, tables, figures, web links, equations or any other data retrieved from the applications with which Cradle can integrate. Changes to any of the requirements can be automatically placed in history and alerts can be automatically sent out to affected users. Cradle offers full traceability: requirements can be linked to tests, risks, other critical issues, and any other project data. Requirements can also be linked to models, architectural entities, and analyses when used with Cradle-SYS.

The Cradle-WRK module is simply a workbench area that allows users or groups to customize the viewing, browsing and manipulating of the Cradle databases. Figure 19, mentioned above, is an example of a workbench. Each session contains a master tree showing everything associated with a particular project. The windows to the right can be situated in any manner and be of any size. The content of these screens can show any information the user requires and in any format (e.g. tables, trees, matrices, lists or custom forms). The custom form view on the bottom right of Figure 19 is easily created by the user from a form details screen. Not only can items be viewed, but also edited. Cross-references can also be created and manipulated. As Cradle-WRK is completely integrated with all the other Cradle modules, any work done in the workbench directly affects the other modules and vice-versa.

The Cradle-SYS module provides the user an interface in which create UML, functional, process, organizational, data and architectural models for all aspects of the project. These models are easily linked to the requirements bringing a 3-D integrity to the systems being developed. Like requirements, models can also contain pictures, videos, tables, figures, equations and link to other MS desktop tools. There are various tools within this module to provide consistency, animation, comparison and automated checkers to ensure models are created correctly from the beginning (Overview, 2006).

There is also Cradle-DOC, Cradle's Document Generation module, which allows users to create and print customizable documents. There are two tools available here: Document Printer and Document Publisher. Document Printer creates the printable documents from Workbench, Toolset, or the other modules individually. Document Publisher allows the automatic generation of a specification document which is a compilation of all information (as needed) from the database created by the other modules. Specification documents are critical to any project because it is these documents that give the customer the complete story of the product being developed and continued assurance of met requirements. Document Publisher tool enables the user to create a MS Word template and add markup (user defined tags) to this template. As requirements, models, tables, lists, etc. are being developed, each is identifiable by certain criteria. When the document publisher is started and a particular template is opened, Cradle will systematically search and retrieve information from the database by matching tags with criteria and populate the MS Word document. The finished product will come complete with table of contents, list of figures and list of tables (Cradle Tutorial, 2005).

Cradle-MET is the module that provides the user the ability to define, measure and report project metrics. The metrics can be output to the web, MS Word or MS Excel into the form of user controlled table. They are no more than queries into the database to retrieve items after which they are analyzed by simple counts and/or full attribute analysis.

2. Screen Shots

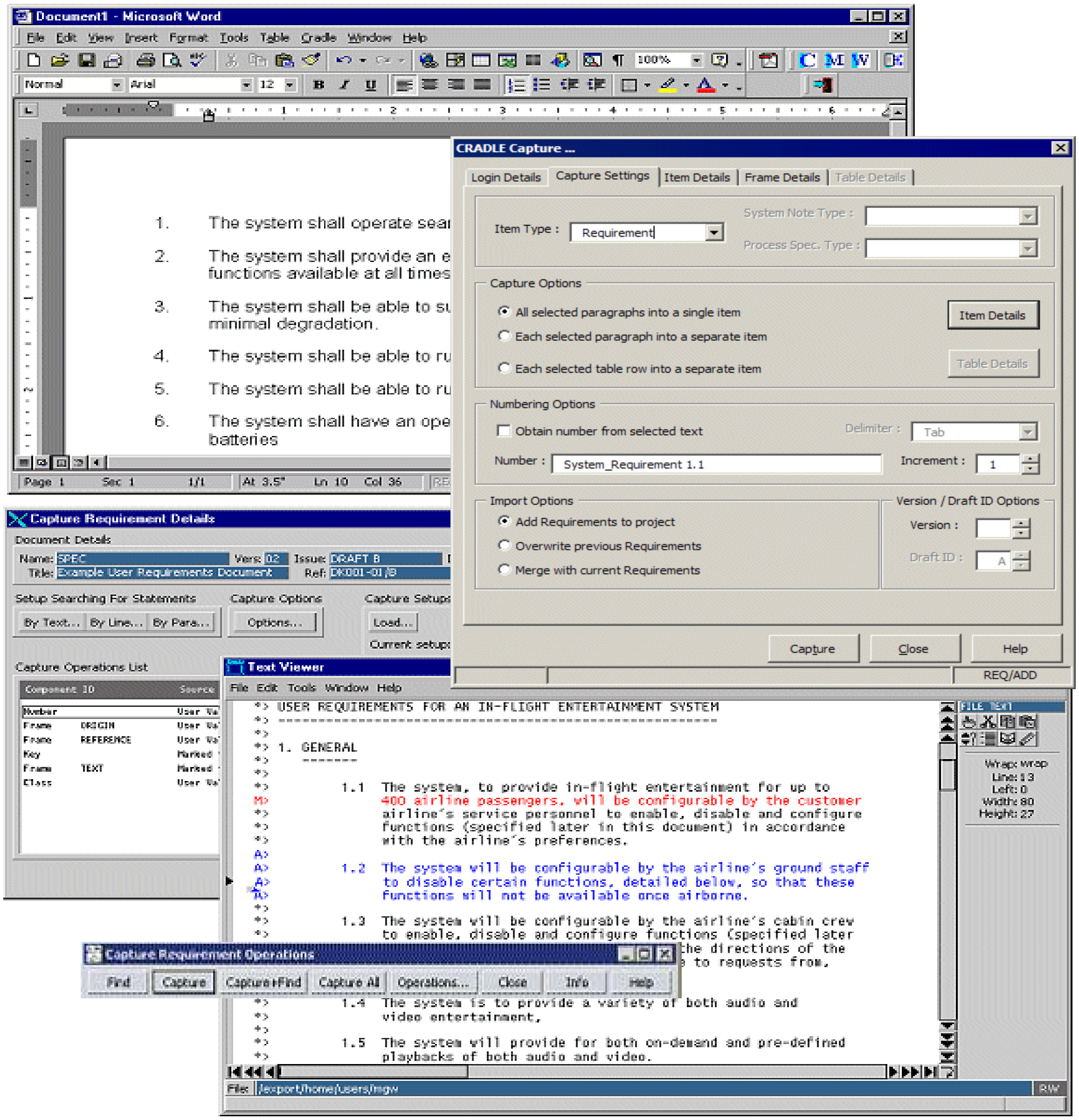


Figure 18. Example of Capturing Requirements with Cradle[12].
Cradle opens Requirements Document in host program (i.e. MS Word) allowing user to select text and Capture Requirement by using a Toolbar

[12] Created by combining several pictures from Cradle REQ, Cradle Functionality and Overview.

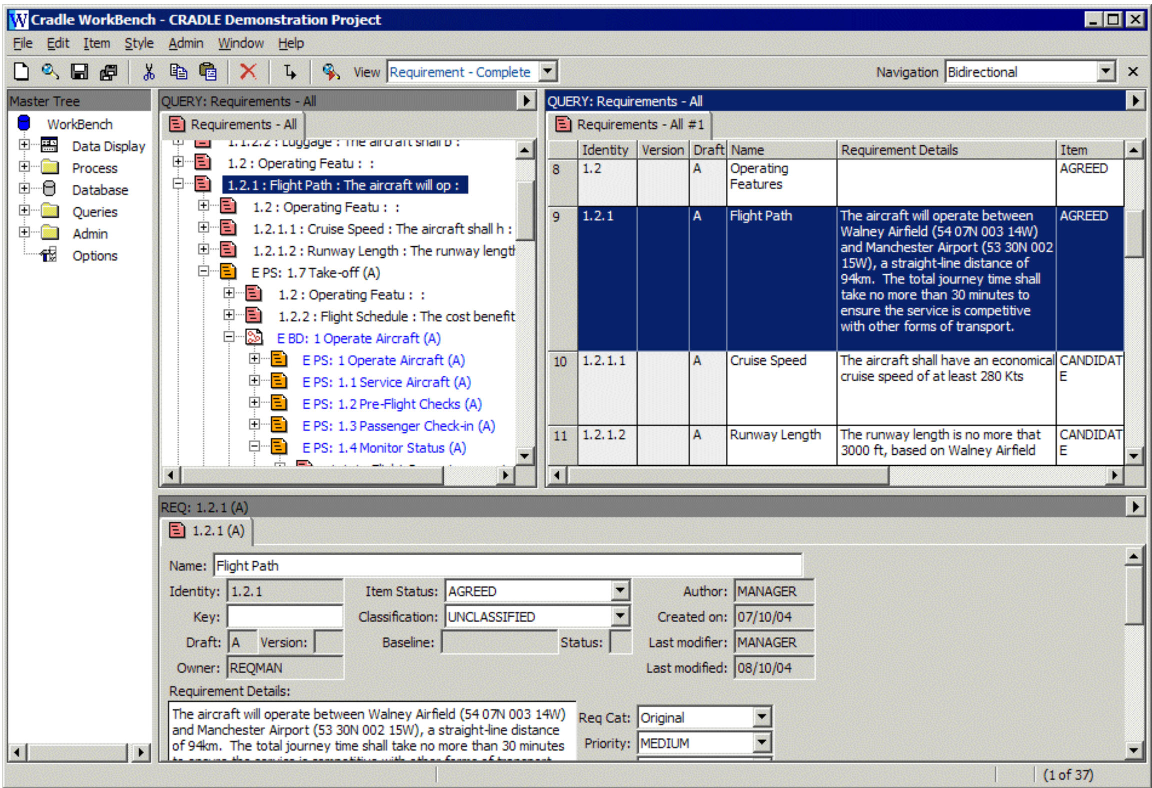


Figure 19. Example of Various Requirements Views in Cradle-WRK[13]. The Workbench can be Customized to show any Combination of Informational Panes

E. DOORS

The DOORS tool has been developed by Telelogic, a Sweden-based company with a U.S. headquarter in California founded in 1983. Considered one of the world's leading RM tools and received the 2005 Yphise Certification Award as Best Requirements Management Software (Yphise, 2005), DOORS provides a collaborative RM environment, a requirements change management system, a powerful traceability feature, and is fully scalable. DOORS, as part of an Enterprise Requirements Suite (ERS), also has many other tools with which it can interact in order to increase its RM capability. DOORS/Analyst allows graphical modeling and more detailed analysis of requirements. DOORS XT allows better integration of virtual workgroups. DOORSnet allows infrequent or remote users controlled access to the requirements database in order to keep those users updated. Telelogic DASHBOARD provides management with an overview of the status of requirements' changes, trends being encountered in a project, and insight to project risk. Integrating with Telelogic SYNERGY/Change allows for the

[13] From Cradle REQ.

creation of more complex approval workflows and a more accountable change control process than DOORS alone can accomplish. This may be of crucial importance considering DOORS can support thousands of users. Telelogic's SYSTEM ARCHITECT, although more for developing Enterprise Architecture, may greatly increase RM capabilities when it is desired to integrate architecture with requirements gathering and design.

The information presented in this section and below has been retrieved from Telelogic's website at http://www.telelogic.com/corp/index.cfm and from use of a full version of DOORS 7.1.

1. Capabilities

DOORS is capable of identifying, gathering, linking, analyzing, tracing and managing many source documents, standards and requirements. Requirements can be imported into DOORS in many formats: MS Word (plain or rich text format), as objects or ASCII files, spreadsheets, MS Project, MS Excel, Adobe FrameMaker, etc. Information can be exported out in the same formats as above plus HTML, MS PowerPoint, and MS Outlook. DOORS is very user friendly in that it behaves like Windows Explorer, as shown in Figure 20, in displaying project and folder hierarchies and in navigating among all documents. This allows projects to be organized easily. Projects are indicated by 'blue' folder icons, folders are indicated by 'yellow' folder icons, and database modules are indicated by a 'document' icon with a 'black bowtie'.

Upon opening any of the modules, the user can access all the requirements in that module, any links that have be set up, and all requirement attributes. Different views of the information in a module can be easily set up, showing as many or as few attributes as desired. Any number and type of attributes can be added to suit project needs. Figure 21 represents a requirements module of a particular database showing many attributes. All objects (requirements and non-requirements) are assigned a unique object identifier (left column). The attributes created for the database are shown across the top in the 'pink' bar. A small vertical column (currently containing 'yellow' bars) indicate the change status of each object: 'green' means unchanged since last baseline, 'yellow' means changed since last baseline but saved to the database, and 'red' means changed since last baseline and not yet saved to the database. Links are indicated by arrows in one of the attribute columns. Changes to any object are automatically captured within DOORS. Right-clicking on any object allows the user to access its properties within which the user can view all attributes, history and links (see Figure 22).

Not only are changes within a database module captured, but DOORS offers a Change Proposal System (CPS). The CPS is normally used to control modifications to requirements after they have been baselined. From any database module, the user can open the CPS which will allow the application of a unique change to a single object or the change of one or more attributes of several objects. Figure 23 shows a typical entry screen for the CPS. All changes created in the CPS are stored in another database module where each proposed change is automatically linked to the source object.

DOORS is also unique in its offering of traceability. Traceability can easily be performed by simply dragging and dropping between objects. It can also be performed by selecting objects from a list or even creating an attribute, inserting object identifiers, and letting DOORS make the link. Informative reports are available which show end-to-end traceability in a single view.

The addition of DOORS/Analyst allows users to augment requirements with diagrams, pictures and models through the application of UML modeling techniques. Modeling brings the requirements to life and makes it easier for users and customers to visualize functionality. It enhances requirements capture, traceability, communication, verification and collaboration. UML diagrams can be created automatically in the requirements documents and links between requirements and the models make getting around easy. There is no need for users to learn to use additional software as the DOORS modules can support storage of the models. Furthermore, traceability ensures that any changes made to either model or requirement results in the other being automatically updated.

2. Screen Shots[14]

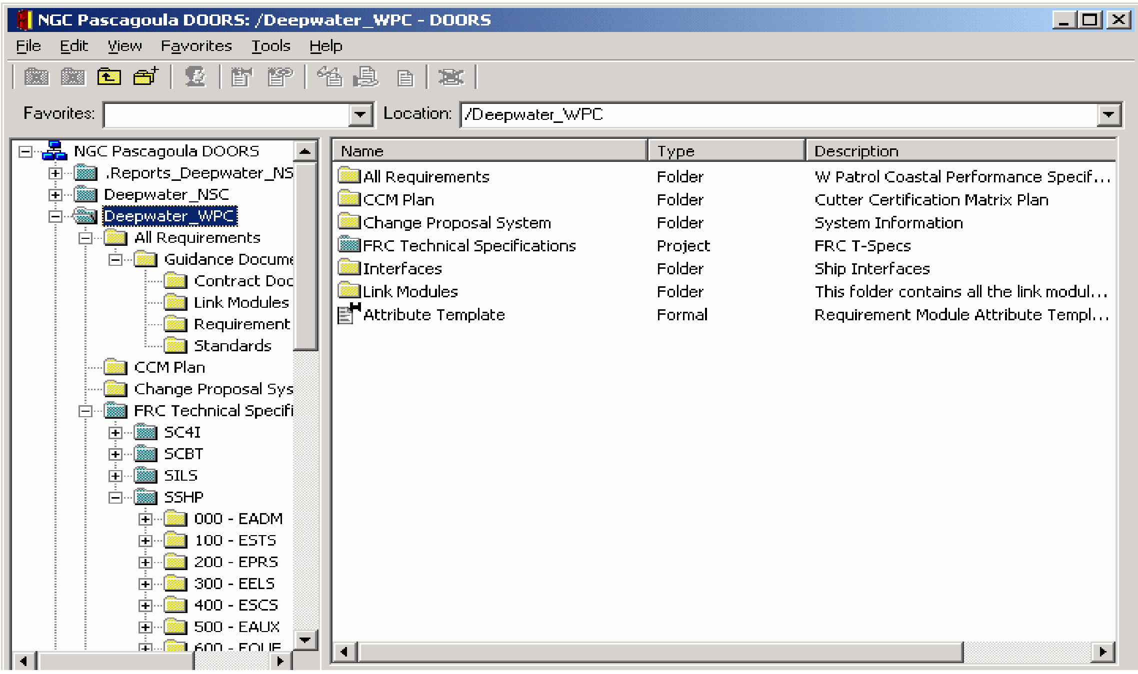


Figure 20. DOORS Database View.
Projects and Folders are easily Viewed and Navigated in this Hierarchical View

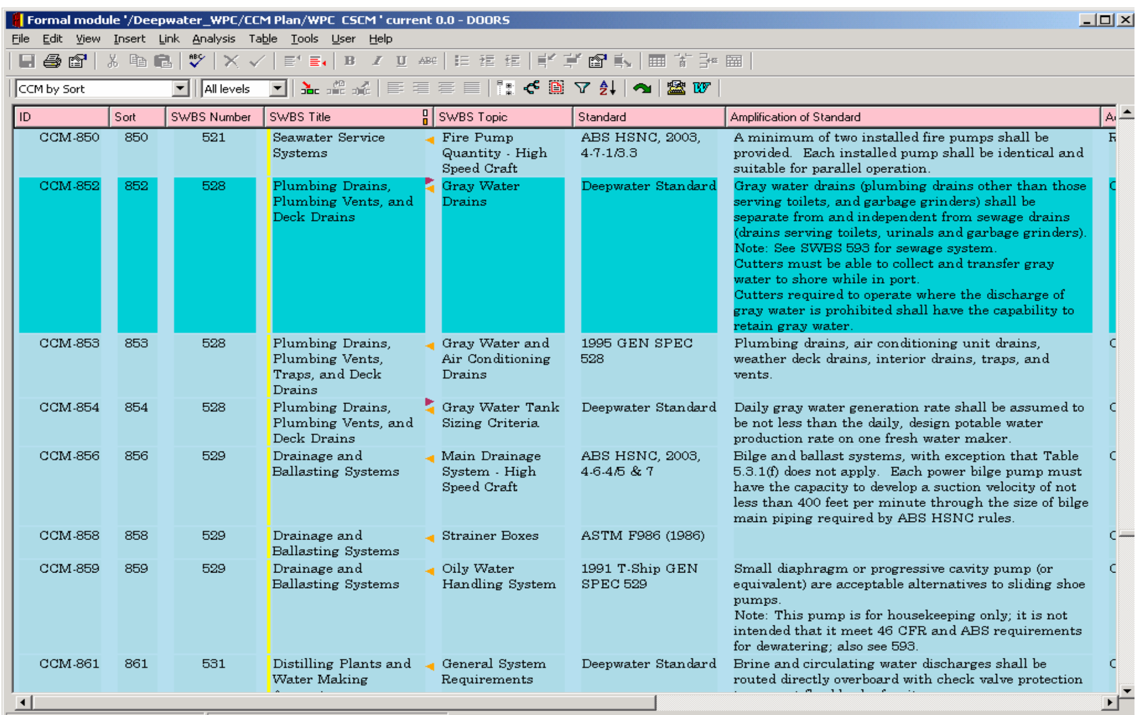


Figure 21. DOORS Document View of Database.
All Requirements' Attributes can be shown in this Table-Like View

---

14 All figures except 23 are screen shots created from the actual DOORS software. Figure 23 is from Telelogic's website.

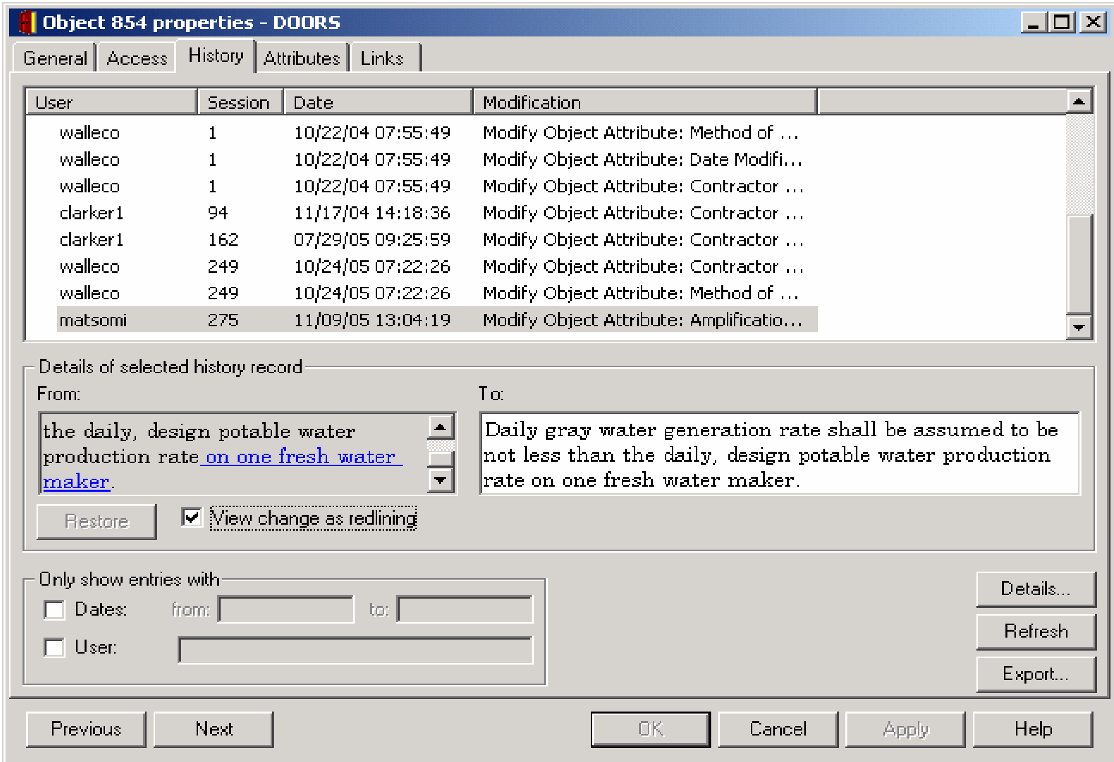


Figure 22. DOORS Object Properties View.
Right-Clicking any Requirement and Selecting "Properties" provides useful Information

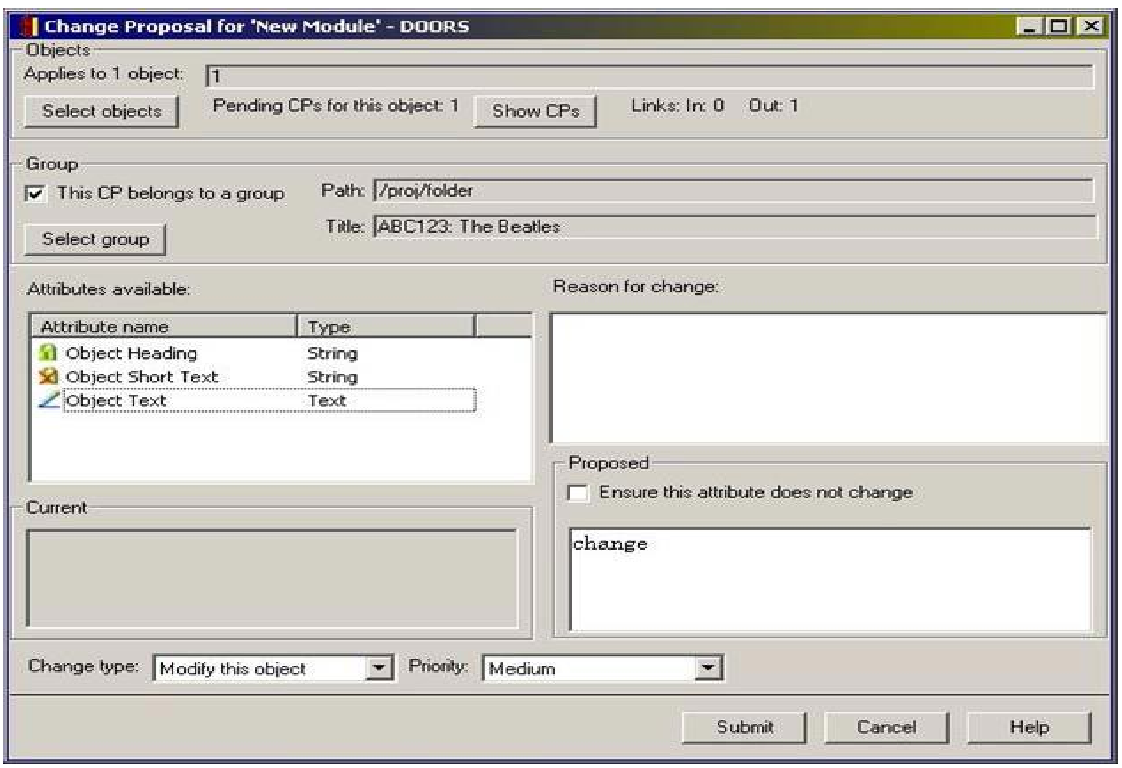


Figure 23. DOORS Change Proposal System View.
All Changes to a Requirement can be done in this Window

F. CHAPTER SUMMARY

RM can be a huge undertaking for any corporation regardless of their product output. Consequently, the business of providing RM tools is also large, and there are many solution providers out on the World Wide Web. For this thesis, only a small handful of tools were selected based on first impression of their websites, availability of information on their websites, and apparent functionality exhibited through their websites. It was apparent that most solution providers catered to the software developer, so capabilities beyond just the software side also influenced the choice of tools for this thesis.

Analyst Pro, CORE, Cradle and DOORS are among the top in the industry. With various add-ons or modules, they all exhibit similar core features with differences showing up in ease-of-use or apparent user-friendliness, if actual use was not possible. Comparison and further analysis of these tools is presented in the following chapter.

IV. COMPARISON OF TOOLS

A. INTRODUCTION

The primary focus of this thesis is to compare various RM tools and their ability to handle all the activities inherent to the management of requirements. Chapter II presented what those activities were and what special needs were inherent to vessel design. Chapter III presented an in-depth look at four RM tools that are considered leaders in the RM arena. This chapter presents the pros and cons of the various tools relative to their ability to perform as a RM tool in the shipbuilding industry. This chapter also concludes with a trade-off analysis of these various tools indicating which tool may be the best tool for RM in the shipbuilding industry.

Before proceeding, it is necessary to recap what RM entails. RM involves: the elicitation, gathering or capturing of requirements from top-level source documents; the analysis of requirements to ensure good form, uncover risks and issues, and ensure applicability; the decomposition of requirements in concert with the development of the architecture to ensure proper traceability; the linking of requirements to design; the generation of specification documents at different phases of design; and the management of changes to each and every single requirement. No RM tool will automatically perform all this without some human intervention even with the best devised scripts; however, a good tool will provide a substantial repository to store and manage all documents, requirements, architectures, and models and provide the applications to perform the above activities in a user-friendly environment.

B. PROS AND CONS

This section deals mainly with the capability of the various tools with respect to the activities inherent to RM, not a comparison between the tools. The comparison is presented in the next section. Most of the pros and cons are derived, as was for the previous chapter, from use of the tool and in-depth research through websites. For this thesis, unfortunately, only Analyst Pro, CORE and DOORS were actually used. However, I believe enough information was retrieved regarding Cradle to ensure equal representation.

1. Analyst Pro 5.3

From the outset this tool appears to be strictly for software development companies with most of the standard screen and report templates being software centric. However, some of this can be customized, such as requirement attributes, to fit the needs of the shipbuilding industry. One of the key benefits of this tool is that all applications are accessible from one main menu and from one or more different modules. Separate licenses are not required for each module. It is a very easy-to-navigate tool and switching from one task to another is quick. Analyst Pro is also extremely easy to learn even without the User Manual.

Analyst Pro can effectively handle many of the activities mentioned in the introduction, except perform or link to architecture development. Although, RM does not include the development of the system architecture, linking requirements to the architecture as they are both developed in unison is very important to effective RM. Analyst Pro can import and store documents, objects, diagrams, etc. in its repository. One can even link to items in the repository, but not to unique elements within an item. Analyst will support multiple concurrent users, but only up to a maximum of 250 on one server. Where this may not be a problem for a smaller shipyard, larger shipyards may find this a hindrance.

2. CORE 5.1

Although not as simple in appearance as Analyst Pro, CORE is full of functionality. The tool handles the whole systems engineering approach and as a result automatically integrates the activities of RM. It may take a little time to learn all that the tool can do, but that is only because there is so much that it can do. As the tool can interface with many different formats of documents, capturing the requirements from and then storing those documents is easy. And if the source document has a certain structure, requirements can be automatically parsed into the CORE repository. One of the key benefits is the amount of information available to the user from the “Project Explorer” screen. One can easily get to and switch between all classes of elements using the left window and see all information about each element in the other windows. Another key benefit of this tool is that the various models, charts and architectures are created automatically based on the characteristics and relationships assigned to each element.

Although not necessary for RM, these forms of pictures do allow for a better representation of requirements. Risks and issues can also be captured and managed within CORE.

The most damaging drawback to this tool is its inability to track the history of changes to requirements. The tool will record the last change, but that's it.

3. Cradle 5.3

Like CORE, Cradle is a tool supporting the whole systems engineering process. As already mentioned above, this is beneficial because the RM activities are fully integrated with the rest of the process. The key benefit of Cradle's WorkBench and Toolset is that is allows the user access to all the modules necessary for implementing a systems engineering process. The Cradle-REQ module can adequately assist the SE in efficiently performing all the RM activities. Capturing of requirements from source documents is extremely efficient given the tool's automated parsers, allowing the SE to devote more time to other activities. Requirements capture is backed up by automated completeness checks to ensure nothing is missed from the source documents. The tool can also load requirements from other tools, such as DOORS and CORE, and return data in the same format. This allows for an efficient exchange of information with customers who don't use Cradle. What makes this tool exceptional is its ability to correct structural deficiencies (i.e. ambiguity, contradiction, duplication) of requirements usually found in source documents. The tool also benefits the SE with a system to manage the complete evolution of a requirement.

The main drawback to Cradle is likely the cost. Although Toolset and WorkBench invoke all the modules, each module is separately priced per user.

4. DOORS

The key benefit of DOORS is that it does one thing and one thing well. It manages requirements. With regard to the other tools, one would think that DOORS is less functional. This may be true for supporting the entire systems engineering methodology, but when it comes to RM, DOORS has just as much functionality. The key benefit of this tool's automated parsers is that source documents are loaded directly into a unique module where each heading and sentence becomes objects. This allows the user to easily analyze each object and pull out the real requirements. The tool is completely based on an "explorer-type" view with a tree to the left and folder details to the right. This allows for easy navigation. The benefit of the module views is that it reflects the appearance of MS Excel with a few added features. This allows any and all information about a module to be shown in any format that a user finds acceptable. Also, linking requirements is extremely easy; just a couple clicks and it's done.

For the shipbuilding industry, this tool would suffice if all that was needed was a robust repository for requirements. However, more and more the industry needs to follow a systems engineering methodology to round out all that is involved in RM. Another benefit of DOORS is the family of products it comes from. One can move to DOORS/Analyst in order to include modeling in an already strong RM tool. Telelogic also has other tools that integrate with DOORS and DOORS/Analyst in order to increase its functionality as a RM tool.

Unfortunately, DOORS does have a few negatives. It can be slow in practice, but this may be due to the size of a particular module or the amount of resources being used at the time. This, however, may be true for all tools. As it is not an all-inclusive systems engineering tool, DOORS does not create models automatically and DOORS/Analyst is relatively new with a not-too-instructive guide for using it. Although DOORS can generate customizable reports from any module, the hierarchy of the output is dependant on the information presented in the module at the time. DOORS does not have the capability to search the database and populate a document template based on pre-loaded criteria; one would have to have the information in the module ordered the way it was desired to be view in the document.

C. TRADE-OFF ANALYSIS

The difficulty in comparing tools from leading RM tool vendors is that they are so close in capability. One can get a good feel for a particular tool from the vendor's website, as mentioned in the previous chapter as to the selection of the tools for this thesis. Analyst Pro's website was simplistic and allowed ready access to necessary documentation and "no-hassle" free trials. Cradle's website was more informative and easy to get around, but obtaining the privilege to an evaluation copy was not possible given the time constraints of this thesis. CORE's website was very informative and still easy to get around with the possibility of downloading academic versions of the tool requiring only a few conversations with a sale representative. Telelogic's website had the most information available covering requirements, modeling, architecture, etc. with white papers, webinars (web-based seminars), and other documentation. The site was easy to navigate and a sales person always called within 24 hours upon registering to navigate to certain areas of the website. But all this says nothing about the tool's functionality in the RM arena, except it indicates the vendor's skill in selling its product. It was, however, an introduction as to what might be expected of the tool.

1. Decision Matrix (DECMAT)

The pros and cons mentioned above try to highlight the important benefits or shortcomings of the four tools in question. However, to properly and objectively compare the tools, a trade-off analysis must be performed. An analysis can be performed manually by creating a table, developing criteria, assigning weights, and then ranking the alternatives. There are also probably many tools out there that will help in doing some of the tasks. For this thesis, DECMAT was chosen for nothing more than its simplicity and its ability to assign weights to criteria based on relative importance factors assigned by the user. DECMAT version 2.2 was created by Captain Richard B. Stikkers in October 1998 after graduating from the Combined Arms and Services Staff School (CAS3) at the U.S. Army Command and General Staff College, Ft. Leavenworth, Kansas (Stikkers, 1998). In a later section, the actual matrix created for this thesis is presented along with a Sensitivity Analysis and a Pairwise Comparison. First, however, criteria must be developed. Note that each alternative, or each RM tool being analyzed, is considered a Course of Action (COA) by DECMAT.

2. Developing Criteria

This thesis has succeeded in presenting the results of independent research up to this point rather than reiterating the answers from the RM tools survey on the INCOSE website. However, most of the questions asked in that survey and presented in Appendix A adequately define relevant criteria that can be used in a trade-off analysis. As only four tools are the focus of this thesis, not all questions will be necessary since either the responses are relatively close in comparison (and confirmed by research) or the questions are not considered vitally important to RM. Although the answers to the survey are somewhat biased coming from the vendors themselves, the research conducted as part of this thesis confirms most of answers given. As such, it is prudent to present those answers as well. All the answers are presented in Appendix B.

As mentioned above, not all questions become criteria for the trade-off analysis to follow. Question 9 did not become a criterion because all tools performed on similar platforms (all operated in a Windows environment, while some were also capable of operating in a UNIX environment) and required similar processing speed, memory and storage capability. The difference in operating environment capabilities was considered irrelevant since most companies operate in a Windows environment. Question 11 did not become a criterion because it was decided that, even though Analyst Pro doesn't ascribe to complying with any standards, meeting standards was not critical in the selection of a RM tool for use in the shipbuilding industry. Question 12 did not become a criterion because all tools provided some level of support and maintenance and it was thought that a criterion based on this would not influence the outcome. The criteria derived from the remaining questions follow.

a. Capturing

The capturing of requirements and structure pertains to survey questions 1 and 2. This criterion is treated a little differently than the rest since the tools vary with respect to the sub-criteria mentioned below. A separate trade-off analysis is done for capturing in order to find the correct ranking of different tools in the full trade-off analysis.

(1) Document Analysis. Each of the tools has a different level of automated analysis of the imported documents.

(2) Automatic Parsing. Requirements can be automatically identified and extracted out of a source document in some tools. Other tools only have semi-automatic parsing.

(3) Interactive Identification. This is similar to semi-automatic parsing of requirements.

(4) Manual Identification. Manual identification of requirements will be inherent in all the tools; it's just a matter of how user-friendly the transfer.

(5) Classification. This refers to how much categorizing, grouping, linking can be done during the actual capturing of requirements from a source document.

(6) Modeling. Not all tools have fully integrated modeling functions, but even those that don't may have a separate application that performs just as well.

b. Flowdown

The flowdown of requirements pertains to survey question 3. Flowdown entails allocating requirements to architecture and level-appropriate decomposition of requirements. This allows users to locate childless requirements (e.g. requirements not decomposed at the next level down). This criterion considers how well each tool allows the user to perform this task and to what level of integration.

c. Traceability

Traceability pertains to survey question 4. Traceability of requirements is extremely important in RM as the user must be able to see upstream and downstream from a requirement and why it is a requirement. This also allows users to locate "orphan" requirements (e.g. requirements not linked to a higher-level requirement), inconsistencies and bad links. This criterion considers how links can be made and what information can be gleaned from those links.

d. History

History is one key aspect of the configuration management of all requirements and pertains to survey question 5. It was already mentioned before that CORE does not maintain a requirements change history. Other tools do have some level of tracking changes.

e. Output

Output of information pertains to survey question 6. Managing requirements is no good if one can not present the information to others, especially the customer when they may not have the software to view it in the form it is being worked. This criterion considers what extent of standard templates each tool has and how easy customized templates can be created.

f. Groupware

Groupware pertains to survey question 7. This criterion considers how each tool handles multiple users and to what level in a database concurrency can occur. Surprisingly, not all are the same.

g. Interfacing

Interfacing pertains to survey question 8. The functionality of an RM tool depends, in part, on how well it works with other tools and applications, both externally and among the same tool. This criterion considers how robust and widespread the interfacing capability is of each tool.

h. Usability

This is where user-friendliness comes into the picture and pertains to survey question 10. Just as important as a website's ease of navigation is to selling a product, the ease of use of a RM tool is to the quality of work generated from it. The functionality (i.e. everything a tool allows a user to do) of a tool directly affects the quality of work, but if it is not easy for the user to produce that quality, functionality is worthless. This criterion considers ease of navigation and understandability of each tool.

i. Learnable

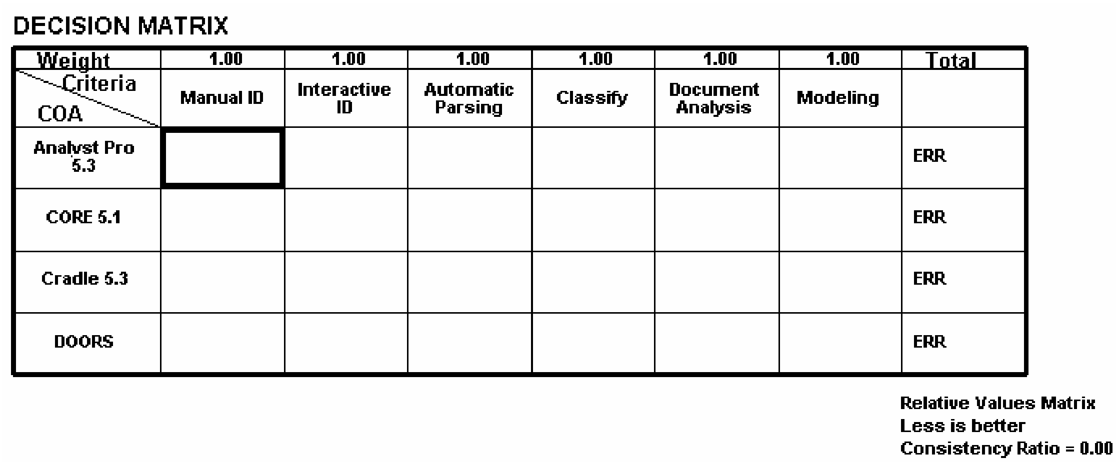
This pertains to training which correlates to survey question 13. Initially training was not going to be considered because most tools require some learning curve. It was determined that the more independent the modules of the various tools could be, the more specialized training was needed. This criterion considers the vendor-recommended training required to be fully versed in all modules that make up or enhance the RM suite of each tool.

j. Cost

Cost is not relative to one of the survey questions, but was deemed important to include as a criterion. In an effort to compare the four tools equally (i.e. same basic capabilities), any number of modules required to get to a common ground were considered. This criterion considers the cost per license/seat of each tool or tool suite.

3. Weighting Criteria

As with any trade-off analysis, criteria must next be weighted. DECMAT utilizes what is known as "Pairwise Comparison" to assign weights to the criteria. This gives some objectivity to assigning weights to criteria rather than one trying to determine weights across all criteria all at the same time. DECMAT requires that the criteria initially be placed in the column headings in order of importance. Since a trade-off analysis of the "Capturing" criteria must be performed first, this is what is represented in Figure 24.[15]



DECISION MATRIX

Weight / Criteria / COA	1.00 Manual ID	1.00 Interactive ID	1.00 Automatic Parsing	1.00 Classify	1.00 Document Analysis	1.00 Modeling	Total
Analyst Pro 5.3							ERR
CORE 5.1							ERR
Cradle 5.3							ERR
DOORS							ERR

Relative Values Matrix
Less is better
Consistency Ratio = 0.00

Figure 24. Initial DECMAT Matrix for Capture Trade-Off Analysis

DECMAT then has a matrix pull-down where the actual pairwise comparisons can be made. Notice the comparison scale in Figure 25 and then the resultant DECMAT matrix showing the criteria weights in Figure 26.

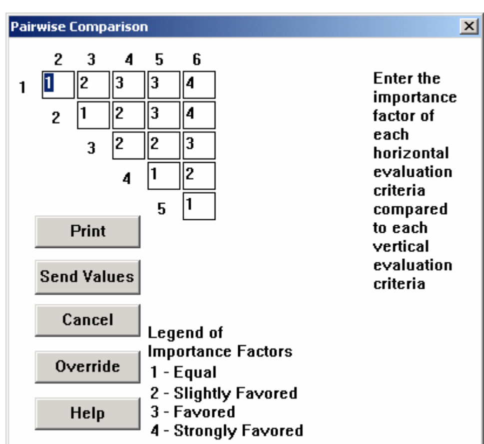


Figure 25. Pairwise Comparison window for Capture Trade-Off Analysis

[15] All screen shots in this chapter were created from using the DECMAT analysis tool.

DECISION MATRIX

Weight / Criteria / COA	4.36 Manual ID	3.62 Interactive ID	2.88 Automatic Parsing	1.58 Classify	1.32 Document Analysis	1.00 Modeling	Total
Analyst Pro 5.3							ERR
CORE 5.1							ERR
Cradle 5.3							ERR
DOORS							ERR

Relative Values Matrix
Less is better
Consistency Ratio = 98.90

Figure 26. Weighted DECMAT Matrix for Capture Trade-Off Analysis

The matrix is now ready for ranking the various tools with respect to the criteria. The same process above is conducted on the full RM trade-off analysis matrix and is shown below in Figure 27, Figure 28 and Figure 29, respectively.

DECISION MATRIX

Weight / Criteria / COA	1.00 Capturing	1.00 Traceability	1.00 History	1.00 Output	1.00 Usability	1.00 Flowdown	1.00 Interfacing	1.00 Groupware	1.00 Cost	1.00 Learnable	Total
Analyst Pro 5.3											ERR
CORE 5.1											ERR
Cradle 5.3											ERR
DOORS											ERR

Relative Values Matrix
Less is better
Consistency Ratio = 0.00

Figure 27. Initial DECMAT Matrix for full RM Trade-Off Analysis

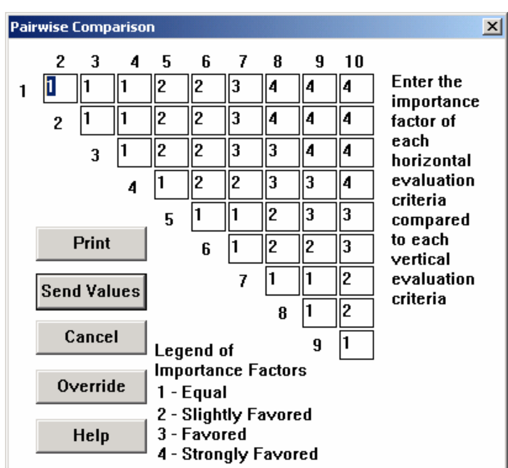


Figure 28. Pairwise Comparison window for full RM Trade-Off Analysis

DECISION MATRIX

Weight / Criteria / COA	4.80 Capturing	4.80 Traceability	4.66 History	4.09 Output	2.73 Usability	2.41 Flowdown	1.81 Interfacing	1.41 Groupware	1.23 Cost	1.00 Learnable	Total
Analyst Pro 5.3											ERR
CORE 5.1											ERR
Cradle 5.3											ERR
DOORS											ERR

Relative Values Matrix
Less is better
Consistency Ratio = 98.99

Figure 29. Weighted DECMAT Matrix for full RM Trade-Off Analysis

4. Ranking RM Tools

With the DECMAT tool it is possible to either perform a multiplication or relative value matrix. The multiplication matrix uses raw data and is therefore more accurate, but the relative value matrix is easier especially when raw numbers are not available. For this trade-off analysis, it was necessary to choose the relative value matrix. The only raw data that could be obtained was training time and cost. It would have been possible to perform a preliminary trade-off for each of the criteria from the full RM analysis matrix as was done with “Capturing”, but that seemed too extreme.

DECMAT requires that successive numbers (1, 2, 3 …) be used in ranking the alternatives. Any instance where two alternatives rank the same, they shall receive the average of the ordered ranking (e.g. two alternatives that would have received 2 and 3, should receive 2.5 if they need to be ranked equal). The following figures represent the final ranking of both trade-off analyses and respective sensitivity analysis. Figure 30 and Figure 31 give the matrix and sensitivity analysis for the “Capturing” trade-off analysis. Figure 32 and Figure 33 give the matrix and sensitivity analysis for the full RM trade-off analysis. The purpose of the sensitivity analysis is to show if any minor change in the criteria weightings would change the outcome. The rankings of the “Capturing” criterion in the latter matrix are based on the relative ranking of the results from the former matrix.

DECISION MATRIX

Weight / Criteria / COA	4.36 Manual ID	3.62 Interactive ID	2.88 Automatic Parsing	1.58 Classify	1.32 Document Analysis	1.00 Modeling	Total
Analyst Pro 5.3	2	1.5	4	3	4	4	39.716
CORE 5.1	1	1.5	2	2	3	1	23.687
Cradle 5.3	4	3	1	1	1	2	36.099
DOORS	3	4	3	4	2	3	48.199

Relative Values Matrix
Less is better
Consistency Ratio = 98.90

Figure 30. Final DECMAT Matrix for Capture Trade-Off Analysis

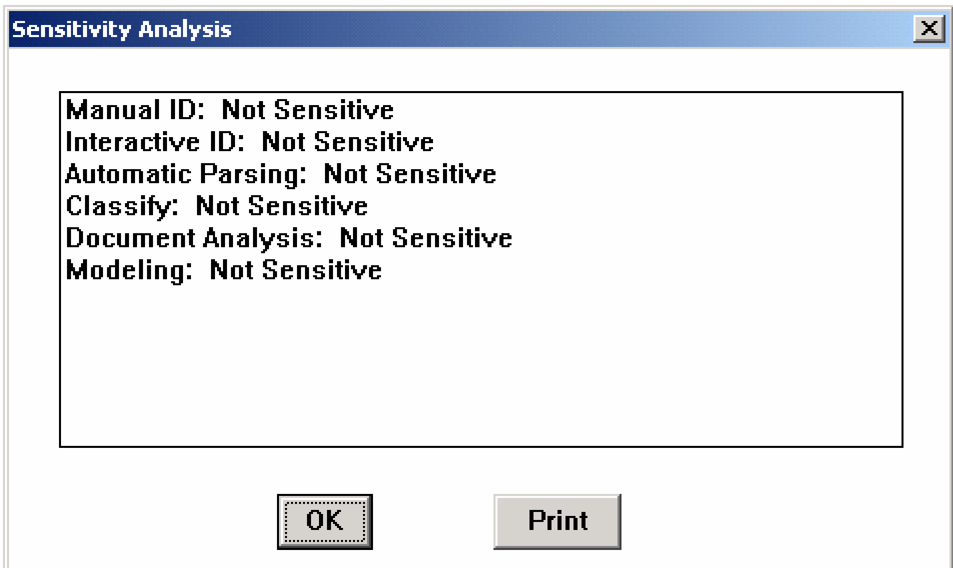


Figure 31. DECMAT Sensitivity Analysis for Capture Trade-Off Analysis

DECISION MATRIX

Weight / Criteria / COA	4.80 Capturing	4.80 Traceability	4.66 History	4.09 Output	2.73 Usability	2.41 Flowdown	1.81 Interfacing	1.41 Groupware	1.23 Cost	1.00 Learnable	Total
Analyst Pro 5.3	2.96	4	2.5	4	3	4	4	4	1	1	94.319
CORE 5.1	1	2.5	4	3	1	2	3	3	3	3	71.588
Cradle 5.3	2.52	1	1	1	4	1	2	1	4	4	52.917
DOORS	4	2.5	2.5	2	2	3	1	2	2	2	72.768

Relative Values Matrix
Less is better
Consistency Ratio = 98.9

Figure 32. Final DECMAT Matrix for full RM Trade-Off Analysis[16]

[16] Relative values for “Capturing” interpolated from results of Capture Trade-Off Analysis

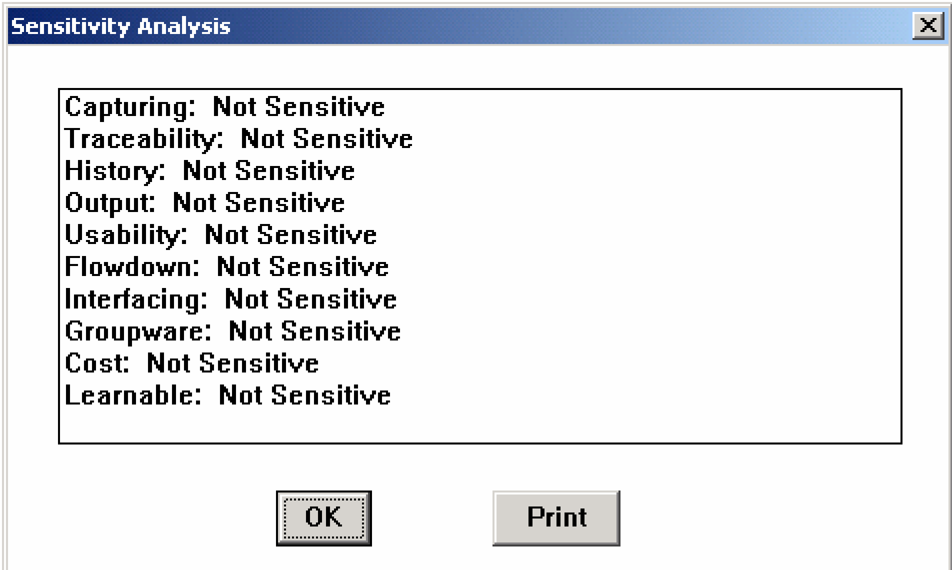


Figure 33. DECMAT Sensitivity Analysis for full RM Trade-Off Analysis

5. Trade-Off Analysis Results

Based on the outcome of the trade-off analysis above, Cradle 5.3 is clearly the best fit for RM in the shipbuilding industry. This outcome is peculiar since actual use of the tool was not possible. However, enough documentation was available to achieve a comparable evaluation. During the first trade-off analysis, CORE 5.1 ranked above the other tools in capturing requirements. In the second phase, CORE and DOORS were about equal in the final ranking.

D. CHAPTER SUMMARY

The focus of this chapter was to compare the four RM tools presented in this thesis. Without repeating all the capabilities discussed in Chapter III, this chapter focused on the key benefits that each tool possessed. In order to be considered worthy of analysis, all tools had to meet certain minimum requirements management abilities. The majority of this chapter dealt with the actual trade-off analysis of the RM tools, with surprising results. Although the author is more familiar with DOORS, after evaluation and comparison, the data suggest that CORE is a better fit for the shipbuilding industry based on the usability and general handling of requirements. However, after the final trade-off analysis was applied, Cradle was shown to lead by a large margin.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSIONS AND RECOMMENDATIONS

A. INTRODUCTION

This thesis introduced the rising need to implement a Systems Engineering approach, and in particular RM, into the Shipbuilding Industry for vessel design. It was found that the definition of RM was not always clear cut among organizations involved in its application. It was also found that, regardless of the terminology, most of the activities associated to RM throughout the various organizations also applied to RM for vessel design. The primary focus of this thesis was to research the various RM tools in order to propose a tool for use in the shipbuilding industry based on several criteria. Only four tools were chosen out of more than thirty, but they did represent most of the leaders in the industry.

B. KEY POINTS, CONCLUSIONS AND RECOMMENDATIONS

As stated before, requirements are the cornerstone of any project. In vessel design, depending on the complexity, the number of requirements can number into the thousands or even tens of thousands. Proper management of these requirements is the only way to help ensure project success. Managing requirements isn't the only thing that must be done, but requirements are critical.

A good RM process is key and it includes many activities, some happening in iterative steps and others happening continuously. The actual process followed (the order of the activities) is not important to this thesis and may be slightly different for various users. Even at Northrop Grumman Ship Systems (NGSS), the process is continuously evolving as more improvements are made. The primary RM activities are:

- Elicitation, gathering or capturing of requirements from top-level source documents
- Analysis of requirements to ensure good form, uncover risks and issues, and ensure applicability
- Decomposition of requirements in concert with the development of the architecture to ensure proper traceability

- Linking of requirements to design, other requirements and source documents
- Generation of specification documents at different phases of design
- Management of changes to each and every single requirement

There are many RM tools out on the market that range from nothing more than a repository to hold requirements and possibly track changes to those requirements to the tool that addresses the whole Systems Engineering approach. Analyst Pro, CORE, Cradle and DOORS were the four tools chosen for analysis. More could have been incorporated into the analysis, but time was a limiting factor.

CORE and DOORS were the only ones to employ an ODBMS, while Cradle employed an RDBMS and Analyst Pro had just a standard commercial database system. DOORS was the only tool, if not integrated with any other Telelogic applications, that effectively and efficiently met the Defense Acquisition Guidebook's definition of RM. The other tools were too integrated to achieve that level of simplicity.

All four tools appeared to be about evenly matched based on the results of the INCOSE tools survey, but final trade-off analysis was able to cut-out the salesperson subjectivity and come up with a more objective result. The Cradle tool came out on top after the final trade-off analysis, and after reviewing the steps in the process of performing the trade-off analysis, that conclusion is still valid. Initially, it was thought that CORE would be the best tool for the shipbuilding industry because of its user friendliness, overall handling of requirements, and all the other applications that enhance RM seemed to be integrated better than the other tools. The other tools had separate add-on applications (except Analyst Pro). Also, CORE has an explorer-type view of requirements similar to DOORS with which this author is familiar. However, after logically placing criteria in order of importance based on experience working with requirements and then performing a pairwise comparison to assign weights to the criteria, the criteria under which CORE performed best were weighted low. DOORS experienced the same outcome despite the author's familiarity.

Although the results indicate that Cradle is the best fit for the shipbuilding industry, in doing vessel design, it should not be concluded that all shipbuilders need to change what they are currently using if they are indeed using a RM tool. To do so would introduce a learning curve for a new tool which may weigh heavily if figured into a trade-off analysis. The trade-off analysis in this thesis attempted to include the learnability of a tool, but it pertained more to the actual learning curve of a tool and did not take into account the resistance a shipyard may experience from changing tools.

A recommendation is not needed here, since the results speak for themselves. However, for most shipbuilders not already using a RM tool and needing something simple to start off with, DOORS would be the tool of choice. Functionality could be increased in the future by the addition of other Telelogic applications.

C. POTENTIAL AREAS FOR FURTHER RESEARCH

SLATE was one lifecycle tool that was not included in this thesis. It was considered but, unfortunately, too late in the thesis process. Further research to include SLATE would be beneficial since SLATE is probably one of the top RM tools and would likely rank with Cradle. In addition to including SLATE in further research, another potential area, that would be more internal to Northrop Grumman (NG), could be conducted. This research could not have been done for this thesis due to the proprietary nature of the information that would be presented, however the basis of the situation can be mentioned. Three different tools are used at three different sectors: DOORS at NGSS, Cradle at NG Newport News, and SLATE at NG Electronic Systems. The problem is that the tools can't be fully integrated with each other. This may be a stumbling block to collaboration, especially within a corporation that is geographically dispersed. Enhancements to these tools that allow complete integration or the development of a new tool that provides a robust and secure link between the tools could be suggested.

This area of research could also lead to another broader, external area of potential research. More and more, shipyards are partnering with each other and other defense contractors in the pursuit of building new vessels. In order to have smooth collaboration during RM, it would be prudent that the partners have similar tools or at least tools that can effectively and efficiently integrate. Research into which combination of tools provide the best external integration and recommendation to tool vendors of improvements that could be made, may produce beneficial results for all of the United States as we struggle to keep pace with other worldwide shipbuilders.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

American Institute of Aeronautics and Astronautics (AIAA) and International Council on Systems Engineering (INCOSE). (1997, August). *Systems Engineering*. Retrieved August 27, 2006, from http://www.incose.org/ProductsPubs/products/seprimer.aspx

Analyst Pro website. Last visited August 28, 2006, at http://www.analysttool.com/

Analyst Pro: Building the right system the right way [Brochure]. (2006). Retrieved July 21, 2006, from http://www.analysttool.com/doc/AP_Details.pdf

Cradle Functionality. Retrieved July 21, 2006, from http://www.threesl.com/pages/products/functional.php

Cradle Overview (RM018). (2005, November). United Kingdom: Structured Software Systems Limited. Retrieved July 21, 2006, from http://www.threesl.com/pages/downloads/download.php?section=lit53&filename=rm01804-Overview.pdf

Cradle Tutorial (RM006). (2005, November). United Kingdom: Structured Software Systems Limited. Retrieved July 21, 2006, from http://www.threesl.com/pages/downloads/download.php?section=docs53&filename=rm00609-Tutorial.pdf

Cradle REQ. Retrieved July 21, 2006, from http://www.threesl.com/pages/products/req.php

Cradle website. Last visited August 28, 2006, at http://www.threesl.com/

CORE 5: Systems Engineering Guided Tour. (2005, September). Vienna, VA: Vitech Corporation. Retrieved July 24, 2006, from http://www.vitechcorp.com/library/docs/CORE5GuidedTour.zip

CORE website. Last visited August 28, 2006, at http://www.vitechcorp.com/

Defense Acquisition University. (2004, November). *Defense Acquisition Guidebook, Chapter 4.0 Systems Engineering* (Version 1.5). Fort Belvoir, VA: DAU. Retrieved August 31, 2006, from Defense Acquisition University: http://akss.dau.mil/dag/DoD5000.asp?view=document

Defense Acquisition University. (2001, January). *Systems Engineering Fundamentals*. Fort Belvoir, VA: DAU Press. Retrieved August 25, 2006, from http://www.dau.mil/pubs/gdbks/sys_eng_fund.asp

Department of Energy Quality Managers Software Quality Assurance Subcommittee. (2000). *Guidelines for Requirements Management* (SQAS19.01.00). New Mexico: Albuquerque Operations Office

Dick, Jeremy (2004, November 5). What is Requirements Management. Telelogic. Retrieved June 14, 2006, from http://download.telelogic.com/download/paper/WPWhatIsRMNov04.pdf

Fluhr, Jody. (2006, July 2). *CORE: A Model-Based Approach to System Engineering and Architecting* [Presentation]. Received via email July 19, 2006, from Fred Manar of Vitech Corporation.

Halbleib, Harold. (2006). Requirements Management. Excel Software. Retrieved August 29, 2006, from http://www.excelsoftware.com/requirements_management.pdf

Haskins, Cecilia. (2006, June). INCOSE Systems Engineering Handbook (Version 3). International Council on Systems Engineering. Retrieved August 24, 2006, from http://www.incose.org/

INCOSE Requirements Management Tools Survey. Retrieved July 21, 2006, from http://www.paper-review.com/tools/rms/read.php

INCOSE Requirements Management Tools survey questions. Retrieved July 21, 2006, from http://www.paper-review.com/tools/rms/INCOSERMToolSurvey.doc

Kar, Pradip and Bailey, Michelle (1996, July 7-11). *Characteristics of Good Requirements.* Seminar presented at the 1996 INCOSE Symposium, Boston, MA.

Naval Air Systems Command (NAVAIR). (1998, September). *Requirements Management Guidebook*. Retrieved August 27, 2006, from http://sepo.spawar.navy.mil/

Northrop Grumman Ship Systems (NGSS). (2006, April 12). *Deepwater Requirements Development Process*. Retrieved August 28, 2006, from http://cm.ss.northgrum.com/Livelink_p/Livelink.exe/Deepwater_Requirements_Development_Process.doc?func=doc.Fetch&nodeId=583501&docTitle=Deepwater+Requirements+Development+Process

Office of Government Commerce. (2006). *Requirements Management*. Retrieved August 29, 2006, from http://www.ogc.gov.uk/delivery_lifecycle_requirements_management.asp

Overview. (2006). Retrieved July 21, 2006, from http://www.threesl.com/pages/downloads/download.php?section=lit53&filename=us_flyers.pdf

Satzinger, John W., Jackson, Robert B., Burd, Stephen D. (2004). *Systems Analysis and Design in a Changing World* (3rd ed.). Boston, MA: Course Technology.

Space and Naval Warfare (SPAWAR) Systems Center (SCC) San Diego Systems Engineering Process Office (SEPO). (2005, January 4). *Requirements Development Process (Expert Mode)*. Retrieved August 27, 2006, from http://sepo.spawar.navy.mil/

Space and Naval Warfare (SPAWAR) Systems Center (SCC) San Diego Systems Engineering Process Office (SEPO). (2005, January 4). *Requirements Management Process (Expert Mode)*. Retrieved August 27, 2006, from http://sepo.spawar.navy.mil/

Stikkers, Richard B., Capt. (1998, October). DECMAT Help File (Ver. 2.2). Combined Arms and Services Staff School. U.S. Army Command and General Staff College. Ft. Leavenworth, KS.

Telelogic website. Last visited August 28, 2006, at http://www.telelogic.com/corp/index.cfm

Unified Modeling Language. (2006). *Wikipedia: The Free Encyclopedia*. Retrieved August 28, 2006, from http://en.wikipedia.org/wiki/Unified_Modeling_Language

Use case. (2006). *Wikipedia: The Free Encyclopedia*. Retrieved August 28, 2006, from http://en.wikipedia.org/wiki/Use_Cases

Walker, Mark (2006). *Cradle-5: From concept to creation*. [Presentation]. United Kingdom: Structured Software Systems Limited. Retrieved July 21, 2006, from http://www.threesl.com/pages/downloads/download.php?section=lit53&filename=rr00108-UK_Briefing.ppt

Yphise. (2005). *Requirements-driven Application Lifecycle Management* [Report]. Retrieved July 21, 2006 from http://www.telelogic.com/campaigns/2005/global/doors_yphise_report/index.cfm?campaigncode=000547-002318

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. INCOSE RM TOOL SURVEY QUESTIONS

The following is the list of questions from the INCOSE Requirements Management Tool Survey site at http://www.paper-review.com/tools/rms/INCOSERMToolSurvey.doc.

1. Capturing Requirements/identification

1.1. Input document enrichment/analysis

Use of existing document information (such as glossary, index, etc.) to aid the user in requirements analysis, identification of requirements, etc.

1.1.1. Input document change/comparison analysis

The ability to compare/contrast two different versions of a source document

1.2. Automatic parsing of requirements

A mechanism for automatic identification of requirements by key words, structure, unique identifiers, etc. to create requirements from the text

1.3. Interactive/semi-automatic requirement identification

The ability to identify requirements from a text file via interactive means such as mouse highlighting of the requirement text or prompting by the system "is this a requirement?"

1.4. Manual requirement identification

A manual means of identifying or creating requirements.

1.5. Batch mode operation

A mechanism for inputting/identifying requirements from outside of the tool

1.5.1. Batch-mode document/source-link update

Does the tool have the ability to update existing linked documents from new/changed versions of the source documents without having to re-establish traceability links?

1.6. Requirement classification

Does the tool have the ability to classify/categorize requirements during identification?

2. Capturing system element structure

Once the requirements have been captured, the allocation of requirements to sub-system elements takes place. The tool must capture these elements so links/allocations can be made to those sub-systems elements.

2.1. Graphically capture systems structure

Can the tool graphically capture system implementation (such as architecture, functional decomposition, WBS, etc.) and display them graphically such that requirements can be linked to them.

2.2. Textural capture of systems structure

Can the tool textually capture system implementation (such as architecture, functional decomposition, WBS, etc.) and display them textually such that requirements can be linked to them.

3. Requirements flowdown

Once the requirements have been captured and system architecture captured, requirements are allocated to the various system elements.

3.1. Requirements derivation (req. to req, req. to analysis/text)

The ability to derive/create additional requirements and link between them such as requirement to requirement, or requirement to text (representing trade studies) to derived requirements

3.2. Allocation of performance requirements to system elements (weight, risk, cost, etc.)

This is the ability to link performance requirements to system elements such as weight, cost, throughput, etc. This also includes the ability to allocate portions of that performance requirement to system elements.

3.3. Bi-directional requirement linking to system elements

The linking of requirements to system elements can be accomplished from either end of the link--from the implementation back to the requirement or from the requirement down to the system element.

3.4. Capture of allocation rationale, accountability, test/validation, criticality, issues, etc., if so how and what mechanism does it use?

Also critical, is the ability to attach rationale, assignments, criticality, test/validation and many other issues to the requirement, allocation, and the system element to which a requirement is linked.

4. Traceability analysis

Once the allocations are complete, the user will want the ability to see the links where they come from, where they go, and why they apply.

4.1. Identify inconsistencies (orphans, etc., if so what kind of...)

The tool should allow the user to identify inconsistencies such as unlinked requirements or system elements (orphans).

4.2. Visibility into existing links from source to implementation (i.e. follow the links)

With the requirement links in place, the user needs the ability to follow the links to see where they come from and where they go to.

4.3. Verification of requirement (was it done, how was it done)

Throughout the life of the project, the requirement management tool will be used to verify that the requirements have been met. The tool should provide the ability to document that the requirement was fulfilled, how it was done, and who was responsible.

4.4. Requirement performance verification from system elements (roll up of actuals)

Once performance requirements have been allocated to system elements, the requirements management tool should support the verification of those requirements by rolling up actuals and reporting on variances (this is the allocated weight versus the actual weight).

5. Configuration Management

5.1. History of requirement changes; who, what, when, where, why, how.

Once requirements have been captured, the requirement management tool should maintain a history of requirement changes, who changed it, when it was done, why it was done, etc. Some of this tracking could be automatic, others could be procedural such as a rationale for the change and how the change is to be accomplished.

5.2. Baseline/Version control

At various times the requirements will need to be baselined (saved and locked away). The requirements management tool should support this along with the ability to compare and contrast between various baselines.

5.3. Access control (modification, viewing, etc.)

The requirements should be able to be protected from modification, viewing, etc. by individuals or groups.

6. Documents and other output media

6.1. Standard specification output (if so what kind)

The requirements management tool should output documentation in various military/commercial standard formats (490, 2167, etc.).

6.2. Quality and consistency checking (spell, data dictionary)

The tool should also support document quality and consistency checking through spell checking, data dictionaries, acronym tables, etc.

6.3. Presentation output

Once the information is loaded, the requirements management tool should support the generation of presentation quality charts and graphs.

6.4. Custom output features and markings (user definable tables, figures, security markings..)

The tool should support the output of documents in finished form including page security markings, graphics/figures, user definable tables, indexes, etc.

6.5. WYSIWYG previewing of finished output

The tool should allow the user to view the document on-screen in finished format.

6.6. Status reporting

Tool users need to status information in the requirements management tool.

6.6.1. Technical Performance Measurement status accounting

Status current technical performance of various allocated performance requirements and monitor progress towards goals.

6.6.2. Requirement progress/status reporting

Status reporting on current compliance/non-compliance to various requirements

6.6.3. Other ad hoc query's and searches

The requirements management tool should support ad hoc queries and searches per the user's discretion.

7. Groupware

Since Systems Engineers rarely work as individuals, the ability for a team of engineers to look/work on the same information at the same time is critical.

7.1. Support of concurrent review, markup, and comment

The tool should support a team of engineers reviewing, marking up, and commenting on requirements or implementation alternatives.

7.2. Multi-level assignment/access control

Access by the team to the database must be tempered by multi-level access control (i.e. the ability to protect things from being modified). This also includes the ability to submit changes into an approval cycle (for acceptance/voting) before committing the changes to the tool for everyone to see.

8. Interfaces to other tools

8.1. Inter-tool communications

Requirements management must have the ability to communicate requirements to other domain-specific design tools (CASE, EE, etc.).

8.1.1. Interfaces to other tools ________?

What tools will your requirements management tool interface with or talk to?

8.1.2. External Applications Program Interface available

To support the wide variety of tools in use by engineers, the requirements management tool should have programmable access to the information contained in the tool's database (to get access to and deposit information).

8.1.3. Support Open database system (standard query access)

Does the tool support Open Database standards such as standard query languages or exchange formats?

8.1.4. Import of existing data from various standard file formats?

Does the tool have the ability to import existing data (such as an ASCII text file containing link information) to create structures within the tool without having to re-enter the information?

8.1.5 Support Data Exchange Standards (AP-233, XML, etc.)

8.2. Intra-tool communications

8.2.1. Exchange of information among same tool, but different installations

Since the tool will be used at different sites and different projects, how does the tool exchange information between different tool installations or databases?

8.2.2. Consistency/comparison checking between same-tool datasets

Does the tool support comparing/contrasting of different same-tool datasets to allow consistency and verification checking?

9. System Environment

9.1. Single user/multiple concurrent users

Is the tool support a single user or multiple concurrent users?

9.2. Multiple Platforms/Operating Systems ______?

Which platforms and operating systems does the tool run on?

9.3. Commercial vs. unique database

Does the tool use a proprietary or commercially available database?

9.4. Resource requirements

Please identify hardware/software configuration requirements:

9.4.1. Memory requirements

9.4.2. CPU requirements

9.4.3. Disk space requirements

10. User Interfaces

10.1. Doing one thing while you are looking at another

Does the user have the ability run a report and look at a requirement at the same time?

10.2. Simultaneous update of open views

If the tool allows for multiple windows/views into the tool, does a change in one view automatically reflect in all other views?

10.3. Interactive graphical input/control of data

Does the tool support graphical input and manipulation of data?

10.4. Which window's standard do you follow?

If your tool supports a window's standard, which one(s)?

10.5. Executable via scripts (recordable) or macros

Does the tool allow the user to create and playback commands or macros that allow the user to automate various tedious tasks?

10.6 Web browser interface?

10.7 Edit Undo Function Support

11. Standards--which ones do you comply with?

Which military/commercial standards does your tool comply with including database standards, output document standards, exchange standards, display/graphics standards, etc.?

12. Support and maintenance

12.1. Warranty

Does your tool have a warranty, if so what is it?

12.2. Network license policy

Does the tool support network licensing (floating, node locked, etc.), if so which license manager?

12.3. Maintenance and upgrade policy

How often are software updates released; are updates separately priced items, etc.?

12.4. On-line help

Are the user's manuals on-line; is there on-line help with the tool?

12.5. Internet Web page location

Does the tool supplier have an Internet address or home page location?

12.6. Phone support

What type of phone support is available from the tool supplier?

12.7 Support Users Group

13. Training

13.1 Tool-specific training classes

13.2 Training available at customer's location

13.3 Recommended training time

What is the recommended training time for a user to become proficient in using the tool?

13.4 Software installation with only basic training

14. What other requirements management features do you as a tool supplier think are important (modeling, etc.)?

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. VENDORS' ANSWERS TO SURVEY QUESTIONS

The following two tables, Table 1 and Table 2, present the answers to the survey questions listed in Appendix A of the four RM tools covered in this thesis. These answers have been retrieved from the INCOSE Requirements Management Tool Survey site at http://www.paper-review.com/tools/rms/read.php.

Questions	Analyst Pro 5.3	CORE 5.1
1. Capturing Requirements/ Identification	Requirements can be captured in spread sheets or word processor and can easily be imported. Flexibility to use use cases for requirements capture. Analyst Pro identifies requirements automatically.	CORE can import data using standard file exchange formats such as ASCII, Rich Text Format (RTF), Microsoft Word, Microsoft Excel, etc. There are several mechanisms for input, including parsers (formatted text and delimited text), semiautomatic (select and transfer), and manual entry.
1.1. Input document enrichment/analysis	Yes	After importing requirements via an originating document, CORE allows the user to create additional requirements or edit existing ones, elements, by just a few clicks of the mouse or automatically with scripting.
1.1.1 Input document change/comparison analysis	Analyst Pro automatically updates if a requirement is changed in the input document.	CORE's scripting capabilities can easily facilitate the comparison/contrast of two different versions of a source document.
1.2 Automatic parsing of requirements	Analyst Pro allows importing requirements from tabular documents by mapping. Analyst Pro also allows importing requirements from documents semi-automatically.	CORE provides an automatic parsing utility. The time required to "capture" requirements is dependent on their format. For example, 100 requirements contained in an EXCEL file or other "parsable" format can be automatically parsed in just seconds.
1.3 Interactive/semi-automatic requirement identification	Analyst Pro also provides semi-automatic tool for importing requirements from documents. It allows importing requirements by highlighting them.	CORE's easy-to-use Element Extractor feature allows the user to easily import information into the database using simple text formats (ASCII, Rich Text Format-RTF) or Microsoft Word or Excel file formats with a click of a button. Users may then decompose elements, such as Requirements or Documents, into lower level elements by linking them with an appropriate relationship.
1.4 Manual requirement identification	Yes. Analyst Pro allows user defined requirements numbering.	CORE's Database Editor allows quick and easy entry of new requirements. When entering data from source documents, the Element Extractor is helpful for analyzing requirements and eliminating compound requirements and ambiguous language.
1.5 Batch-mode operation	Allows importing requirements in batch mode loading from various data formats.	CORE's Applications Program Interface (API) provides a C/C++ interface for batch-mode operations.
1.6 Requirement classification	Analyst Pro provides upfront classification of requirements. Ability to create as many as classes as needed for a project. Additional classification is possible with attributes.	While utilizing CORE's Element Extractor in identifying the requirements, one can simply change the classification of requirements (Type attribute) and categorize them (Origin attribute) at the same time.
2. Capture System Element structure (if so, how? As document paragraphs? Product structures?...)	Yes.	Using its System Definition Language, CORE captures requirements, models behavior, supports analysis and simulation, and documents processes and products. All elements are linked via a central integrated design repository. If an element is changed, the change is reflected through the design due to the unique identification of the element within the design repository.
2.1 Graphically capture systems structure	Analyst Pro provides built-in diagramming tool for capturing system structure graphically. It also allows using external diagramming tools.	CORE is founded upon the concept of using models to support systems engineering. Graphical representations are key to providing the necessary understanding and communications to engineer a successful system or process. Representations available within CORE include: expanded functional flow block diagrams (eFFBDs), physical block diagrams (PBDs), functional flow block diagrams (FFBDs), IDEF0 or data flow diagrams (DFDs), interface charts (N2), and hierarchy diagrams.

2.2 Textual capture of system structure	Yes. It allows capturing system architecture, functional decomposition, etc textually and allows to link requirements to them.	In addition to the graphical views, each element within the integrated design repository can be displayed in an easy-to-use text view so that all information associated with the element can be viewed and edited: name, number, description, attributes, relationships, targets, etc.
3. Requirements Flowdown	1. Allocate requirements to system components 2. Ability to create sub-requirements 3. Link requirements to documents and other complex objects	CORE allows the users to build and maintain a single, multi-access system design database that retains hierarchical linkage of requirements, function flow threads, associated requirement allocations, technical basis for allocation and traceability and linkages to lower tier requirements where the flow can be up, down, or lateral as controlled by the user.
3.1 Requirements derivation (req. to req., req. to analysis/text)	Easy to create sub-requirements and link requirements by parent-child and peer-to-peer relationship.	Once entered, CORE allows the user to allocate, link, build relationships between requirements, originating documents, physical structure, etc.
3.2 Allocation of performance requirements to system elements (weight, risk, cost, etc.)	Easy to attach requirements to system elements (weight, risk, cost, etc.) by defining custom attributes	CORE allows the user to create/manipulate many attributes of the requirements, functions, physical components, etc, through that element's property sheet; weight, cost (units), value, KPP, etc. The assignment of risk to a requirement is performed by creating a risk element and linking the two through the System Definition Language's "caused by" relationship.
3.3 Bi-directional requirement linking to system elements	Yes.	CORE's inherent System Definition Language automatically links elements within the database in the reverse direction of that created (i.e. When a requirement is given a relationship to a lower level requirement of "refined by" the child requirement will automatically be assigned the relationship to the parent of "refines."
3.4 Capture of allocation rationale, accountability, test/validation, criticality, issues, etc. If so, how and what mechanism does it use.	Yes.	CORE allows the user to create/manipulate many attributes of the requirements, functions, physical components, etc, through that element's property sheet; including rationale, assignments, criticality, test/validation, allocations, flowdowns, etc..
4. Traceability Analysis	Provides a full set of traceability tools.	See below.
4.1 Identify inconsistencies (orphans ...) If so, what kind of...	Inconsistencies can be found by: 1. Traceability Reports 2. Traceability Matrix	Several document templates, as well as queries, contain information on inconsistent items. In addition, the variety of graphical and text representations available with CORE are useful for analyzing the design information.
4.2 Visibility into existing links from source to implementation--i.e. follow the links	Analyst Pro provides powerful graphical view that shows all the linkage.	Verification of requirements can be done visually through CORE's graphic and text representations. More importantly, COREsim allows the user to perform simulation of the system behavior that has been modeled to ensure that the proposed architecture is executable.
4.3 Verification of requirements (was it done, how it was done...)	Yes.	Leading to a testing event, Verification requirements are created for each test, linking back to originating or derived requirements. It is the Verification Requirements which denote whether or not a requirement was satisfied within the testing event.
4.4 Requirement performance verification from system elements (roll up of actuals)	Yes.	CORE is delivered with a robust script generator that allows the generation and tailoring of output reports in user defined format. Dynamic roll-ups of weighted requirement values could easily be satisfied by a simple script.
5. Configuration Management	Analyst Pro provide the following for thorough configuration management. 1. Base lining 2. Automatic Change History 3. Tracking 4. Rollback to back any previous version 5. Locking	CORE offers robust change management and access protection for the data resident within the system design repository.
5.1 History of requirement changes, who, what, when, where, why, how	Analyst Pro automatically tracks requirements change history.	CORE records information on: who made the last change, when the change was made, what was changed, and when the element was last exported to file. Each user may have a unique login name and password that governs what information they have access to. CORE records the username and time of the last attribute change for each element in the database.
5.2 Baseline/Version control	Analyst Pro has built-in support for base lining.	Versions of the data are maintained simply by exporting and archiving the dataset. The easiest and most straightforward method of base lining and preventing changes to the included elements is to change the permissions to Read Only and export/archive a copy for safety.

5.3 Access control (modification, viewing, etc.)	User groups can create and users can be linked to them.	Access Control for the CORE system is available through several complementary mechanisms. The User/Group Tool; allows the system administrator to define new users and specify system privileges. Administrators also assign users and groups permission to access projects, elements, and even individual attributes if desired. With these permissions defined, all access to data in CORE – whether via the CORE GUI, the report generator, the API, or the CORE2net web interface – is governed by these permissions. In CORE Workstation, all user definitions and properties are maintained in encrypted Access Control Files (ACFs) in the CORE directory. These files can be shared among users at a single CORE site or multiple distributed sites. In Enterprise, all user information is automatically maintained in the Enterprise repository.
6. Documents and Other Output Media	Analyst Pro allows generating requirements documents, reports by grouping and custom filtering and exporting in various formats.	See below.
6.1 Standard specification output (if so, what kind)	Yes.	490, 498, 632, 2167
6.2 Quality and consistency checking (spelling, data dictionary, ...)	Yes. Provides spell checker and traceability matrix.	CORE includes standard spell checking, data dictionary, etc. CORE allows you to check within a single element or within the entire database and any portion thereof.
6.3 Presentation output	It allows generating graphs and charts.	CORE is delivered with a robust script generator that allows the generation and tailoring of output reports in user defined format, including tabular and multiple graphical output styles.
6.4 Custom output features & markings (definable tables, security marking)	No Response Added.	CORE is delivered with a robust script generator that allows the generation and tailoring of output reports in user defined format. A Rich Text Format (RTF) output feature allows Microsoft Word to read, edit, and print the produced documents.
6.5 WYSIWYG previewing of finished output	Yes.	CORE is delivered with a robust script generator that allows the generation and tailoring of output reports in user defined format. A Rich Text Format (RTF) output feature allows Microsoft Word to read, edit, and print the produced documents, real time, on the screen in its finished format.
6.6 Status Reporting	Analyst Pro provides ability to filter and create reports with statuses.	All Elements within all classes have the ability to have their attributes updated / status provided at any point in time through CORE's element Property Sheet.
6.6.1 Technical Performance Measurement status accounting	No Response Added.	Functional flows created to satisfy performance requirements can be simulated in CORESim to monitor design progress at any point in the design process.
6.6.2 Requirement progress/status reporting	Analyst Pro provides: 1. Ability to create Reports 2. Ability to generate graphs.	CORE is delivered with a robust script generator that allows the generation and tailoring of output reports in user defined format, including reporting on the status of requirements satisfied through linkages to functional and physical design elements.
6.6.3 Other ad hoc queries & searches	Analyst Pro allows generate repots by custom filtering and grouping.	CORE comes with a drag and drop based query builder (Script generator) to allow the generation or editing of queries. CORE also provides the ability to do simple queries by specifying relationships desired to produce graphical hierarchies.
7. Groupware	Yes. Analyst Pro is a multi-user, multi project tool. It avoids update conflicts with powerful concurrency control.	The CORE product family provides a flexible combination of modeling and simulation tools supporting product and process engineering. CORE's object-oriented environment delivers the same functionality from a single user workstation to large, distributed, client-server teams via its Enterprise configuration which allows multiple users to access the same model. CORE2net provides web-based access to the team to the project model.
7.1 Support of concurrent review, markup, & comment	Yes. Analyst Pro is a multi-user, multi project tool. It allows concurrent review, markup and comment.	Through using a CORE Enterprise Server License & multiple Enterprise Client Licenses, multiple users may interact with the design repository at any point in time. Updating/changing element attributes are only limited to the first user to access the element having changing rights until they "release" the element for another to work with. Through using CORE Workstation Licenses, multiple instantiations of a database can be created, then integrated by the ultimate administrator of the programs database, allowing complete freedom between database

		instantiations.
7.2 Multi-level assignment/access control	Yes.	Each program must assign an administrator to the CORE Database, who can assign multi-level access control through all elements and the all lower level attributes, as well as accept or reject changes (should a workstation license be employed).
8. Interfaces to Other Tools	See below.	See below.
8.1 Inter-tool communications	Powerful import/export features and built-in repository.	See below.
8.1.1 Interfaces to other tools?	Yes.	CORE interfaces to DOORS, RDD-100, Vital Link, Software thru Pictures, Rational Rose, Microsoft Office (Word, Excel, PowerPoint), and Microsoft Project.
8.1.2 External Applications Program Interface available	Provides import/export facilities that are customizable and reusable for exchanging information.	CORE's C/C++ API provides open access to the database from other engineering or project management tools or applications. Thus, CORE's database can be utilized as a central repository for all project-related data.
8.1.3 Support Open database system (standard query access)	No Response Added.	CORE provides support via simply query language.
8.1.4 Import of existing data from various standard file formats	Allows import requirements from various documents such as rtf, doc, html, txt, etc.	CORE can import data using standard file exchange formats such as ASCII, Rich Text Format (RTF), Comma Separated Variable (CSV), XML, etc.
8.1.5 Support Data Exchange Standards (AP-233, XML,..)	Powerful Import/Export tools provided.	CORE is actively working with XML and AP-233 data exchange standards.
8.2 Intra-tool communications	Export/import Features	See below.
8.2.1 Exchange of information between same-tool different installations	Yes. Analyst Pro provides powerful import and export features.	Should an Enterprise Server/Client license be employed, all the "clients" directly interact with the same database, housed on the Enterprise server. Location has no effect on this. Should several Workstation Licenses be employed an ultimate Administrator of the database will need to retrieve database updates from the outlying other Workstations and integrate them.
8.2.2. Consistency/ comparison checking between same-tool datasets	Traceability Matrix and Base lining Comparison allows consistency checking.	This instantiation could only occur using several Workstation Licenses. In this case it would be the responsibility of the ultimate Administrator of the database to generate a script to perform the comparing/contrasting between the various received database updates.
9. System Environment	Analyst Pro is multi-user, multi-project tool. Analyst Pro is a Windows based application. Analyst Pro uses commercial database.	CORE's object-oriented environment delivers the same functionality from a single user workstation to large, distributed, client-server teams.
9.1 Single user/multiple concurrent users	Yes.	CORE supports both. Single and multiple concurrent users are supported via our Workstation or Enterprise products. In addition, the CORE2net Enterprise Web Server provides access to all information and models contained in a CORE database. A separately licensed component of the CORE Enterprise Server, CORE2net turns the CORE Enterprise system into a Web server. CORE2net users require only appropriate access permissions and an Internet Browser. CORE2net is a "CORE viewer" that does not require any special software to be installed on a user's workstation.CORE2net makes the information contained in CORE available to an entire organization. System engineering, like every other art or science, has creators and consumers. The creators are the engineers that create systems engineering designs. The consumers are the rest of the world that needs access to the work of the creators. CORE is a product designed for the creators; CORE2net is for the consumers. The CORE2net Enterprise Web Server is compatible with Internet Explorer and Netscape.
9.2 Multiple Platforms/Operating Systems?	Client: Windows 98, NT, 2000, XPServers: Windows NT, 2000, 2003	CORE can run on Windows 98/NT/ME/2000/XP/2003.
9.3 Commercial vs. Proprietary database	Commercial	CORE utilizes a true object oriented database (OODB) in both versions: CORE Workstation and CORE Enterprise. To produce an integrated and executable model of an architecture, an OODB is critical for this type of application as it allows for each element of an architecture definition to be treated, stored, and accessed as an object. Utilizing an OODB also

		facilitates the export of entire architecture definitions to one data file such as an XML file. It also allows organizations to easily modify the meta-model to suit their needs without impacting the functionality of the tool. For CORE Workstation, a proprietary OODB is utilized. For CORE Enterprise, a third party OODB from Gemstone Systems Incorporated, is utilized. Both versions interoperate and can be utilized in one environment. To the user, the difference between the versions of CORE and the associated database engines is not perceptible.
9.4 Resource Requirements	Windows server (windows NT, 2000, 2003) for more than 10 concurrent users.	CORE is not a resource-intensive application. A simple rule of thumb is to determine if your machine is capable of running Microsoft Word® 2000. If so, your system can run CORE.
9.4.1 Memory requirements	32 MB RAM (256 MB Recommended)	For Workstation installation: 256 MB RAM (Note: More may be required when accessing large amounts of data such as executing CORE scripts on large databases.)
9.4.2 CPU Requirements	Minimum Pentium 200 MHZ	For Workstation installation: 1 GHz processor or higher. The faster the CPU, the faster script and simulation (using COREsim) execution will be.
9.4.3 Disk space requirements	100 MB available (250 MB Recommended)	For Workstation installation: 100 MB free disk space for the software, documentation and online help files. Additional disk space required for CORE Workstation project databases.
10. User Interfaces	GUI	CORE has a user-friendly interface based on the familiar Windows motif.
10.1 Doing one thing while you are looking at another	Yes.	CORE provides the ability to easily work on multiple tasks with various methods to enter data or execute tasks.
10.2 Simultaneous update of open views	No Response Added.	Yes, the tool allows multiple activities to be performed at the same time. Should a change be made to a requirement while another user is viewing that requirement, the viewing user would see the changes real-time.
10.3 Interactive input/control of data	No Response Added.	CORE dynamically generates views and diagrams directly from the database ensuring that views are consistent with current design details.
10.4 Which window standard do you follow?	Uses Microsoft Windows for look and feel	Microsoft Windows
10.5 Executable via scripts (recordable) or macros	No Response Added.	The addition of discrete event simulation capability to CORE with the COREsim option allows execution of the integrated architecture by dynamically interpreting the behavior model that resides in the system design repository. Discrete-event simulation logic identifies timing, resource utilization, and model inconsistency. CORE is delivered with a robust script generator that allows the generation and tailoring of custom scripts as well.
10.6 Web browser interface?	Internet and VPN connections can be used.	The CORE2net Enterprise Web Server provides access to all information and models contained in a CORE database. A separately licensed component of the CORE Enterprise Server, CORE2net turns the CORE Enterprise system into a Web server. CORE2net users require only appropriate access permissions and an Internet Browser. CORE2net is a "CORE viewer" that does not require any special software to be installed on a user's workstation. CORE2net makes the information contained in CORE available to an entire organization. System engineering, like every other art or science, has creators and consumers. The creators are the engineers that create systems engineering designs. The consumers are the rest of the world that needs access to the work of the creators. CORE is a product designed for the creators; CORE2net is for the consumers. The CORE2net Enterprise Web Server is compatible with Internet Explorer and Netscape.
10.7 Edit Undo Function Support	Yes. Support Windows standard functions.	CORE does not have an Edit Undo Feature.
11. Standards--which one(s) do you comply with?	No Response Added.	CORE has been considered compliant with several engineering standards, including IEEE-1220, EIA-632, and Mil-Std-490A. The tool was built as a System Engineering tool and is intended to support variations in engineering process and reporting standards. Information contained within the tool repository may be expanded to allow compliance with most of today's evolving standards. Vitech is an active participant in emerging efforts for ISO10303-233 (AP233) – the standard for systems engineering data exchange and the Object Management Group's Systems Modeling Language (SysML) extension to UML.
12. Support and	First one year free support and	A full-range of support and maintenance is available for CORE including

Maintenance	upgrades.	online, telephone and email support.
12.1 Warranty	Analyst Pro comes with first one year free upgrades and support.	Vitech offers a 30 day warranty on all software products. Subscription to the annual Maintenance Plan includes all upgrades and telephone/email technical support.
12.2 License policy (Network, Node Locked,..)	Dedicated, floating licenses. Very simple built in license management. External license managers are not required.	CORE supports three licensing options: floppy key, node-locked, and network licensing mechanisms.
12.3 Maintenance & upgrade policy	First year upgrades and updates are Free.	CORE publishes two major releases per year; with point releases more often. All releases and upgrades are included in the Maintenance plan.
12.4 On-line help	No Response Added.	All support information is available on-line. Additional information is available on-line as well, including methodology, case studies, technical notes, Department of Defense Architecture Framework resources, and Capability Maturity Model information.
12.5 Internet access/Website	http://www.analysttool.com	http://www.vitechcorp.com; info@vitechcorp.com
12.6 Phone support	703-351-5032	The CORE support hotline is staffed 10 hours per day. Messages are recorded at all other times. E-mail requests are preferred for quickest response and can be submitted via the website or sent to support@vitechcorp.com.
12.7 Support users group		CORE has a formal Users Group that meets twice a year. There is also a web-based CORE Users Group with a wealth of resources available to users.
13. Training	Web based training and onsite training are available.	A full range of training resources are available. Hands-on training courses in the use of CORE and COREsim are available monthly for individual users. Dedicated team training is available upon request. The Vitech technical library also offers a self-guided introduction to CORE as well as in-depth technical notes and white papers.
13.1 Tool-specific training classes	No Response Added.	Vitech offers CORE-specific training classes that are scheduled throughout the year at our Vienna, VA, USA location: Model-based Systems Engineering with CORE - An Introductory Course; Advanced CORE - Unleashing the Power of CORE.
13.2 Training available at customer's location	Yes. Onsite training is available.	All of the CORE training courses can be delivered at a customer location: Model-based Systems Engineering with CORE - An Introductory Course; Advanced CORE - Unleashing the Power of CORE. Our worldwide representatives also offer convenient CORE training upon request. Courses may also be tailored to the customer's needs.
13.3 Recommended training time	No Response Added.	We suggest that new users attend a 4-day introductory class that exposes them to the majority of the tools capabilities, however many users are successful without any formal training. Many users can be self-taught using our comprehensive Guided Tour, which typically takes no more than three hours.
13.4 Software installation with only basic training	No special training is required for installation.	CORE is easily installed on a client workstation. Workstation requirements are available at http://www.vitechcorp.com

Table 1. INCOSE RM Survey Answers for Analyst Pro 5.3 and CORE 5.1

Questions	Cradle 5.3	DOORS/ERS
1. Capturing Requirements/ Identification	Cradle provides a range of tools for importing requirements information into items in the database from Microsoft Office tools. These import tools all provide three fundamental options for importing information:(a) Overwrite Off, in which information is imported only if the item does not already exist in the database(b) Overwrite On, in which information is imported irrespective of whether it already exists in the database or not, but when importing, any pre-existing content of the information is overwritten(c) Overwrite Merge, in which information is imported irrespective of whether it already exists in the database or not, but when importing, all data in attributes that are not being imported are preserved and the only attributes to be updated are those contained in the import file. The significance of the import Merge option is that is allows items to be augmented with additional attributes taken from multiple data sources. For	DOORS/ERS is an integrated, RM suite designed to capture, link, trace, analyze and manage a wide range of information to ensure a project’s compliance to specified requirements and standards. DOORS/ERS combines a powerful, RM database engine with an intuitive document-style interface that provides thousands of concurrent users with a single, customizable view of the most up-to-date requirements information

	example, a hierarchy of requirements could be imported from a Word document, and then columns from an Excel spreadsheet could be used to augment these requirements with additional attributes.	
1.1. Input document enrichment/analysis	Using existing document information (such as glossary, index, etc.) aids the user in requirement analysis, identification of requirements, etc.	Automatic input parsers analyze text for keywords and create attributes for recognized data such as references and security classification. Following parsing requirements can be automatically labeled based on any search criteria.
1.1.1 Input document change/comparison analysis	Cradle can identify differences between successive versions of external source documents, and identify which requirements or other items may be affected by these changes. Associated impact analyses can then be performed. Cradle does not store multiple copies of items for different versions, effectively managing the version histories such that database size is controlled and performance impacts are minimized.	The spreadsheet import does automatic updates of new versions. Other parsers may be user modified to compare existing with new data. Also, a document compare function can be used to compare two documents that have been separately imported.
1.2 Automatic parsing of requirements	The Cradle REQ module allows the user to parse requirements based on identified key words, paragraph number or structure, or unique identifiers. Requirements (and the desired numbering system) can be created directly from the text. Cradle also has Word and Excel Plug-ins, facilitating data capture from these tools and automated entry into the Cradle database. When capturing from Word or Excel tables, Cradle can selectively import from just those columns that you want, skipping columns not needed. Additionally, Cradle has a read/write API to support special purpose database access requirements.	Multiple parsers are available to read all kinds of data. All parsers may be configured to fit the users' particular needs.
1.3 Interactive/semi-automatic requirement identification	Cradle REQ allows for manual (interactive) and semi-automatic (e.g., groups at a time) parsing and identification of requirements. In both this and the fully automatic mode above, automated completeness checks are provided to validate the extent of requirements capture. Plug-ins for Microsoft Word and Excel allow the user to capture desired paragraphs and tables from Word and spreadsheet rows from Excel as Cradle requirements, system notes or process specifications.	Automatic parsers will recognize requirements without intervention unless input data is ambiguous in which case the user will be prompted.
1.4 Manual requirement identification	Cradle REQ provides the mechanism to scan and individually identify and create requirements and their definitions. Cradle also provides an MS Word/Excel interface that supports creating requirements directly in the database from either Word or Excel. Cradle provides a Word plug-in that allows any collection of paragraphs or table rows to be imported into new items in the database. When importing paragraphs, users can specify values for all attributes to be created in the new database items. When importing from a table, users can define a mapping between the table columns and the attributes of the items to be created in the database, optionally omitting some of the table columns. Cradle provides an Excel plug-in that allows any range of cells to be imported into new items in the database. When importing from a spreadsheet, users can define a mapping between the data columns and the attributes of the items to be created in the database, optionally omitting some of the columns. To simplify the creation of this mapping, Cradle will attempt to automatically match data columns to database item attributes by reading the column headings.	Yes.
1.5 Batch-mode operation	Cradle REQ provides the mechanism to scan and individually identify and create requirements and their definitions. Cradle also provides an MS Word/Excel interface that supports creating requirements directly in the database from either Word or Excel.	Full batch loading of requirements from multiple source formats is provided
1.6 Requirement classification	Yes, requirements grouping, classification, and categorization are all fully supported. Cradle can store arbitrarily many, arbitrarily large attributes of any data type to support the classification process. Cradle databases are extended by a point-and-click interface. Cradle can store requirements in any number of hierarchies or other structures concurrently. Any requirement(s) can appear in any document.	Requirements that are updated, either directly or in batch operations, retain their links. New versions of documents may be used to update the requirements; however, the use of constant requirements identifiers in the source documents significantly aids the process.

		It should be noted however, that DOORS provides a fully featured Microsoft Word-like editing environment to negate the need for external modification and in many instances remove the need for repeated update from external sources. Where needed, links can also be loaded in batch mode from external files.
2. Capture System Element structure (if so, how? As document paragraphs? Product structures? ...)	See below.	See below.
2.1 Graphically capture systems structure	Yes, Cradle provides an extensive library of modeling notations that allows the user to capture system implementation and associated linking. Representative Cradle notation sets include: functional (data flow, state transition, entity relationship, data structure, structure charts); behavior; process; architectural (function block, hierarchy, physical, software) and object oriented (class, use case, sequence, collaboration, statechart, activity, component, deployment).	DOORS/Analyst provides a mechanism of describing functional decomposition and analysis in the form of UML 2, stored, viewed and edited directly within DOORS.
2.2 Textual capture of system structure	Yes, the user defines the appropriate (database) item types and the corresponding data dictionary or specification textual descriptions. Requirements can then be cross-referenced into these data types as well as associated system model items.	DOORS can be used on its own to textually describe systems structure or in conjunction with DOORS/Analyst which enables the textual descriptions and the graphical representations to be kept in synch with each other.
3. Requirements Flowdown	Cradle allows arbitrarily complex relationships between requirements, in which any requirement can simultaneously be linked into any number of hierarchies or directed acyclic graphs. The requirement structures can be depicted in Explorer-style trees, nested tables, matrices or Hierarchy Diagrams (HIDs).	See below.
3.1 Requirements derivation (req. to req., req. to analysis/text)	Cradle fully supports requirements derivation hierarchies, as well as decomposition, merging and history tracking. In Cradle, you create links as needed (provided you have the right to do so). Cradle provides the facility for user-defined cross reference link attributes (attributes within cross references in which the user can store information) that can be used within search criteria. These link attributes can be displayed in Tables using Cradle WorkBench. Cradle allows the user to group links, to search based on any link type defined in a link group, and to create any number of link groups (each containing any number of types of link).	Derived or additional requirements may be created directly using DOORS full requirements editor or using decomposition tools to automatically allocate, sub-divide or combine requirements or other data. Links are created automatically by the tool when this is done. DOORS/Analyst can be used to provide rationale behind how requirements have been derived – i.e. capture requirements, perform some sort of analysis on them with DOORS/Analyst and then write down any derived requirements from the Analysis. Full traceability is captured from high level requirement through to analysis and onto to derived requirements.
3.2 Allocation of performance requirements to system elements (weight, risk, cost, etc.)	The Cradle user defines these system elements as (database) item types. Performance requirements can then be cross-referenced into these data types as well as items in the system models (e.g., other requirements, functions, interfaces, etc.).	May be achieved using attributes associated with the requirements or independent data linked to the requirements. Performance metrics and other calculations may then be performed on the data. Metrics and other numerical data can be calculated/allocated/dispersed between requirements or other data at the same level or across different levels through links.
3.3 Bi-directional	Cradle provides bi-directional, many-to-many, typed, attributed, transitive	This is the natural way of working in

requirement linking to system elements	cross-references to system elements. Cradle has the capability of mapping and linking Work Breakdown Structure (WBS), System breakdown Structure (SBS) or Product Breakdown Structure (PBS) elements with requirements.	DOORS and very little difference is made between linking up, down or at the same level. Note that linking is done through a simple drag-and-drop approach within the same document or between documents.
3.4 Capture of allocation rationale, accountability, test/validation, criticality, issues, etc. If so, how and what mechanism does it use.	Cradle provides complete flexibility in attaching frame information to items, including text, spreadsheets, tables, equations, video, sound clips, CAD/CAM drawings, etc. These frames can be typed, for defining new data types, the privileges associated with them, and customized interfaces to interact with data of such types. Cradle can color code its requirement attribute values, based on the value of another related attribute. For example, criticality can be addressed by coding all Mandatory requirements (and their frame information) red, Highly Desirable requirements blue, etc. This color coding makes it easy to visually scan and understand a set of information. The colors are reproduced when attributes are printed, can be used in tables, and appears in output RTF and HTML. Cradle can also reference associated requirements data (e.g., related test data) in external files, at URLs, and within external environments. Cradle provides for regular expressions (regex), the most powerful and flexible search expression for retrieving the data. Cradle can also display matrices or nested tables (such as tests for each acceptance criteria for each requirement). Tables can be generated with any amount of nesting and exported directly to RTF, HTML and Excel. Users can define any number of these tables through a simple point-and-click interface.	All rationale, test/validation, critical issues may be associated with the requirements using assigned attributes or other associated objects in the database. Importantly, this information may also be associated with the link directly as well as the object/requirements itself so that relationships may include a rationale. If required, DOORS can be set to prompt the user for rationale, etc, upon creation or change of any requirement or data object.
4. Traceability Analysis	All searches into the Cradle database use queries. A query searches for database items (such as requirements) using any variety of criteria, among which is whether the item is, or is not, linked to any other items, or items of a particular type (such as other requirements), optionally by links of any type or a particular type or any link type of a link group.	See below.
4.1 Identify inconsistencies (orphans ...) If so, what kind of...	Hierarchical diagrams provide a graphical depiction of requirements links to display anomalies. Cradle's hierarchy diagrams always show the existence of all children of a node, and provide full control over the expansion and collapse of nodes. In its hierarchy diagrams, Cradle also provides full control over which item attributes are shown; the user can also specify the diagram colors, appearance (sizing, alignment) and style. When the user has resolved all inconsistencies in the hierarchy diagram, he/she can save the diagram as a static snapshot of the database structure (to be included in a report, for example). These diagrams are not the only means to identify database inconsistencies. Cradle provides many, independent tree views of the database structure. Cradle's WorkBench module also provides coverage analysis by allowing user-defined queries into the database to look for all unlinked items, undefined items, unspecified inputs/outputs and other anomalies of interest. For example, when searching for orphan items (such as requirements), Cradle provides 'Orphan', 'Unconnected', 'Disconnected' and other such alternatives as single-click options, in addition to the above functionality. Finally, analysis tools are provided for identification and correction of ambiguity, contradiction, duplication and omission instances within the database.	DOORS can identify objects/requirements with no links at all, with no outward links or with no incoming links or with no links of certain type. This can also be conditional on other data types. For example, show all unlinked tests that have not yet been performed or all links to failed tests.
4.2 Visibility into existing links from source to implementation--i.e. follow the links	Cradle allows any items, such as requirements, functions, architecture components, verifications, acceptance criteria, trade studies etc) to be linked (optionally by links of specific types) in any manner. Link rules can be defined to impose rigor into such relationships and ensure consistency and integrity. Linking can be done through drag-and-drop between Explorer-style requirements/function/component hierarchies, and through graphical HIDs. Cradle allows for project-defined link types and groups; several traceability templates, traceability diagrams and reports are provided with the software. WorkBench provides predefined queries that allow the user to immediately examine and assess requirements bi-directional traceability. Extended sorting, table viewing and querying capabilities all exist in WorkBench to help assess requirements definition and linkage. Hierarchy diagrams are available in Cradle that provide a view showing all items that	This may be performed on line or printed in the form of a matrix style report. DOORS also offers the users visible "link-tips" to show links directly in the document and traverse them with the simple click of the mouse. Importantly DOORS can show an unlimited number of link traversals on the same screen at the same time for powerful analysis. Links are also visible in exported HTML versions of the documents and in data viewed directly via the Internet/Intranet using

	are directly linked to the current item and, for each of them, the items that they are cross referenced to, for a user-specified number of levels. A formal Requirements Traceability Matrix (RTM) template is provided in the Document Publisher library. Generating this matrix will also provide complete visibility into all project database links from source to implementation. Completeness assessment and impact analyses of requirements to system cross referencing can be accomplished using these mechanisms.	DOORSnet.
4.3 Verification of requirements (was it done, how it was done...)	Cradle provides complete traceability throughout the lifecycle, with the ability to attach item frame data to capture the complete requirements history. Impact analyses and change assessments of this history can be performed. A formal Verification Cross Reference Matrix (VCRM) template is available in the Document Publisher library. Generating this matrix will provide the user complete visibility into the plans for verifying the requirement (how and when), and who is responsible for the verification test. Holes or inconsistencies in the verification plans will become evident after review of this matrix.	This is achieved by the use of links and/or attributes and is the whole basis for using DOORS.
4.4 Requirement performance verification from system elements (roll up of actuals)	Performance requirements are linked to constraints on performance of the system design model. Cradle then assesses the system performance by executing the design model using mathematical models based on user configurable performance parameters. Results of this analysis provide allocated versus actual system parameter comparisons.	Pre-written scripts provided in the DOORS library support roll up values either within a set of requirements or across data linked from many modules. Analysis may also be automated to highlight requirements or other objects where actuals exceed allocated values. If necessary, such discrepancies may be automatically e-mailed to key users. DOORS also offers a statistics tool to automatically generate graphical displays of metrics or calculated values.
5. Configuration Management	Cradle contains a built-in CM system that provides mechanisms for the formal review of information, baselines and version control, formal change control (change requests and change tasks) and an audit trail. This audit trail records all CM events related to each and every item (requirement, function, CBS, SBS, solution concept and so on).	Telelogic provide two variants of version management. DOORS CPS is a built in change proposal system. DOORS ECPS is an enterprise change proposal system built on the top of Telelogic SYNERGY.
5.1 History of requirement changes, who, what, when, where, why, how	Cradle provides complete edit histories of requirement (or any item) changes. In addition, Cradle can record full details of each and every edit to each and every item, recording the date and time of the edit, who did the edit, the reason for the edit, and the old and new values of all attributes modified by the edit. A Pending Delete facility is available, where deleted requirements can be recovered if necessary. Total requirements evolution is provided through history records, external user annotations and CM baselines. Cradle also provides full configuration audit logs within its internal Configuration Management System. Cradle provides command line utilities to create automated incremental and full database backups.	DOORS automatically records who, what (down to the actual attribute changed), when and how automatically. The why can also be captured either through the voluntary entry of data by the user, or forced via DOORS' automated trigger mechanism such that the user would not be able to save the change without entering a rationale. DOORS also supports a full Change Proposal System (CPS or ECPS) for collecting change requests and formally reviewing them via a CCB before changes get into the documents or data sets. The CPS is also available in DOORSNet for Internet/Intranet access.
5.2 Baseline/Version control	Cradle provides a free, integrated, extensible Configuration Management System (CMS), offering complete support of formal review and change, and subsequent base lining and version control and management. Cradle has extensible database schema to allow encapsulation and use of existing QA documents within existing QA/QC procedures. Using the Web Publisher Module, Cradle can also publish different baselines, or works-in-progress, into separate website components, for comparison between approved and	Full base lining is supported as well as the comparison of any two baselines. Baselines in DOORS are saved, locked versions of data sets such as requirements, system elements, tests, etc. that may be viewed, flexibly reported, but never changed.

	current activities.	
5.3 Access control (modification, viewing, etc.)	Cradle provides complete access controls based on project-configurable combinations of security classifications, user privileges and project organization. Formal change control is provided via Change Requests and Change Tasks within the CMS. For cleared viewers, Cradle offers an electronic post-it annotation mechanism for read-only external reviewers who can record their comments in annotations to items in the database.	Access control in DOORS may be achieved at three levels in DOORS. First, sets of data such as requirements in a document or all test cases may be controlled. Second, individual objects such as a single requirement may be controlled. Third, access controls may be imposed on attributes within objects. For example, users may be able to edit a comments attribute but not modify the allocated cost attribute. Access rights can also inherited by children from parent objects. Access levels include the ability to read, create, modify, delete and control access itself. Further, a mechanism called propagation allows access right to be imposed on documents or requirements not yet in existence.
6. Documents and Other Output Media	Cradle provides a range of facilities for exporting information to Office tools, and supports all current versions of Office including Office 97, 2000 (SP1 or later), XP and 2003. Cradle can generate exports of any tabular view or matrix, including views of cross referenced information with arbitrarily many levels of nested cross reference information, to Word and to Excel. This includes Cradle's ability to generate matrices showing inter-relationships between the items in either one, or two or three collections of items. Cradle allows any diagrams to be exported to Word and to a variety of other export formats including SVG for web pages. The Document Publisher tool uses any format specified in any user-created Word document or Word template file, and combines this format with queries to the database that are embedded in tags in hidden bookmarks embedded into the Word document using the tool's point-and-click UIs.	See below.
6.1 Standard specification output (if so, what kind)	The new Cradle Document Publisher enables the user to use Microsoft Word to construct arbitrarily large and complex documents from datasets in a Cradle project database. Word documents or reports are user-defined through Word templates (outlines) constructed with tags or bookmarks that are keyed to access database information and the layout/format of the information in the output document. User-defined templates provided, with reports producible in the following formats: Mil-Std-490, 498, 961 and 2167; IEEE-1220; DoD-5000; EIA-632; and DoDAF. Cradle provides typical systems engineering report templates as starting points, such as the Systems Engineering Management Plan (SEMP), Performance Requirements Based Specification (PBS), Systems Requirements Specification (SRS), Interface Requirements Specification (IRS) and System Design Description (SDD), to mention a few. In any case, the user can define his/her own template or modify existing ones to suit individual needs. The Requirements Traceability Matrix (RTM) and Verification Cross Reference Matrix (VCRM) templates are also included in the Cradle template library. Finally, the Web Publisher Module allows you to publish sections of your Cradle database (or document) as fully hyperlinked website components, with diagrams published as hyperlinked SVG files for ease of use and display.	No Response Added.
6.2 Quality and consistency checking (spelling, data dictionary, ...)	Data dictionaries and acronym tables are an inherent part of the Cradle infrastructure. Since the Cradle Document Publisher creates all reports in Word, the user has access to all the associated word processing quality and consistency checking features (e.g., spell checking and grammar) available in Word.	DOORS includes a spell checker on both the UNIX and PC versions. Other mechanisms such as acronym tables may be implemented by the user with DOORS modules making use of the Object Oriented database.
6.3 Presentation output	The new Cradle Document Publisher produces quality Word reports and presentation charts with contents including tables, diagrams, graphics and images. Output directly to Excel by the Document Publisher will be	DOORS can generate color graphs and charts for displaying metrics data, results of calculations and statistics.

	available soon (currently have an option to output tables in CSV for reading in Excel). Additionally, Cradle supports a wide spectrum of output formats, including Postscript, RTF (for PowerPoint), HTML, Interleaf, SVG, FrameMaker, HP LaserJet, HPGL, and EPS.	Examples of such charts produced by DOORS include a volatility chart showing numbers of changes over time for a document or data set. These charts and graphs can be generated and printed directly from DOORS without the use of an external graphics package. DOORS document hierarchies may be viewed graphically and traceability may be viewed as "tree" structures in the Traceability Explorer.
6.4 Custom output features & markings (definable tables, security marking)	The Cradle Document Publisher creates and publishes all of these features. User defined tables and graphics are specified as tags or bookmarks (by a point-and-click interface) on a Word template, and the corresponding data is extracted from the project database by the Publisher and output in a finished report. Cover page information, security markings, disclaimers, etc. are all part of the report template, again defined by the user. The Table of Contents, List of Figures and List of Tables are all dynamically generated as the report is being produced. For users without access to Word, Cradle's Document Printer is used to define (again by a point-and-click interface) templates depicting the desired document structure, layout and contents.	Title pages, contents pages, headers, footers, graphics, etc. are all part of the DOORS output to Postscript printers on UNIX or the Windows Print Manager on PC.
6.5 WYSIWYG previewing of finished output	The Document Publisher report output file can be previewed as a Word document on the screen immediately after it is produced. Alternately, the Web Publisher Module will present the output as a fully hyperlinked website component.	Yes.
6.6 Status Reporting	All Cradle tools provide status information or dialog boxes prompting the user for the function of its UI controls, for the completeness and integrity of requirements and other items as they are manipulated, and for both system-level and project-level reporting.	See below.
6.6.1 Technical Performance Measurement status accounting	Cradle users accomplish performance status accounting by defining a combination of user defined item types (system notes) that are assigned for tracking and annotating performance related requirements into any number of models created in the Cradle design modeling domain. Cradle WorkBench is then used to query the progress/status of these item types.	DOORS configurable/programmable attributes allow status monitoring of any held information, either within a document or across links in multiple documents.
6.6.2 Requirement progress/status reporting	Cradle users accomplish status reporting by defining a combination of user defined item types (system notes) that are assigned for tracking and annotating cross references linking requirements into the analysis and design domain models and to supplemental project risk and issue items. Cradle WorkBench is then used to query the progress/status of these item types.	Links and link attributes combined with configurable/programmable attributes support reporting and statistical analysis of compliance or non-compliance
6.6.3 Other ad hoc queries & searches	Cradle WorkBench provides a complete (point-and-click) library of the more common requirements-related user inquiries and searches of interest, with the capability to easily create additional user-defined queries as needed.	DOORS supports searches and queries on any data according to users needs either by sets that fulfill criteria or each next object that meets defined criteria. Search and filtering can be on attribute value, substring searches or the presence or absence of links.
7. Groupware	Many Cradle projects are intercontinental or international, with simultaneous access to shared or replicated databases.	See below.
7.1 Support of concurrent review, markup, & comment	The Cradle infrastructure fully supports a multi-user project database with distributed data sources and users. Cradle provides a mechanism called "Alerts" to allow users to send instantaneous status messages to other users, their team, or the entire project. Additionally, the electronic post-its annotation process is extremely useful for read-only external reviewers to record their comments relative to items in the database. The Cradle Configuration Management System (CMS) generates urgent alerts to the reviewers of information and to everyone in the project when a baseline is opened, closed or restored.	This is supported in DOORS through a variety of mechanisms. Each project can choose the most appropriate method for them. - First, DOORS supports multi-user concurrent write access to the same document. - Second, DOORS' CPS (Change Proposal System) allows proposed changes from multiple users to be reviewed together and either the best taken or a combination generated. This feature is also available through the

		DOORS web interface, DOORSnet. - Third, DOORS Distributed Data Management (DDM) allows portions of the DOORS database to be taken out, worked on and returned for resynchronization with the database. - Fourth, DOORSrequireIT can be used to extract a document from DOORS into Word. Here, the requirements and their attributes can be managed, modified, deleted and new ones created before returning it to the DOORS database for an update. - Fifth, DOORSnet allows users from remote locations to also participate in the team work by making changes and creating new requirements directly to the DOORS database using the internet or an intranet.
7.2 Multi-level assignment/access control	Cradle provides fully user-defined access controls by the project to include: project organizational structures (users, teams, groups); controllable item ownerships; customizable user privileges, item and user security clearances, personnel skills and roles; and customizable user access rights to tools in the environment. These controls are used to implement and monitor the review and approval of informal or evolving changes. Cradle's internal Configuration Management System provides the necessary change request/action/review/control processes associated with formal change approval cycles.	DOORS provides a formal Change Process for submission of proposed changes. Specified users then review proposed changes either on-line (with sign off fields) or by committee (CCB). Accepted changes are promoted into the document or dataset automatically. DOORSnet supports the submission of changes for review from remote locations
8. Interfaces to Other Tools	Cradle provides a library of tool interface mechanisms, including data converters, plug-ins, an Applications Program Interface (API) and full CSV (comma separated value) capability for data exchange.	Requirements management must have the ability to communicate requirements to other domain-specific design tools (CASE, EE, etc.).
8.1 Inter-tool communications	See below.	DOORS has the most flexible method of inter-tool communication. Not only is there a full API, but the DOORS extension language (DXL) can be used to write imports and exports to other tools in almost any format. DXL being C-like is very easy to learn and use. DOORS on the PC can also use OLE automation for integration such as those used by Microsoft tools
8.1.1 Interfaces to other tools?	Cradle has data converters for RDD-100, CORE, Teamwork and BPWin, and uses the CSV standard for data exchange with DOORS, StP and RTM. Cradle also has Word and Excel Plug-ins, facilitating data capture from these tools and automated entry into the Cradle database. In the case of DOORS, the plug-in connects to a DOORS server and asks the user to authenticate. After this, the user can choose to export any module(s) from DOORS into the plug-in, where they are converted to AP233 format files which are then imported into the Cradle database. Cradle provides a separate utility to browse the AP233 files through a tree-viewer. The same plug-in can be used to import requirements back into DOORS (typically a new module since DOORS cannot overwrite existing items in the database). Additionally, Cradle has a read/write API to support special purpose database access requirements. Cradle has a data converter menu that prompts the user on how to convert a data file from many of these other products into a specified Cradle import/export file.	The Telelogic program for interfaces with other tools is the most comprehensive in the industry. We now have over 25 interfaces to the most popular tools for design, analysis, text, CM, etc.
8.1.2 External Applications Program Interface available	Cradle has a C/C++ Applications Program Interface (API) that provides open access to the database from other engineering tools or project management applications. This API supports getting data from the database as well as writing into the database. The API also supports VB and VBA	No Response Added.

	applications. Thus, Cradle's database can be utilized as a central repository for all project-related data. 3SL uses the Cradle API to create Cradle tools. Specifically, the Cradle Document Publisher and Spell Checker are examples of tools that we have created with the API.	
8.1.3 Support Open database system (standard query access)	Yes, Cradle supports UML Modeling; RTF, MIF, OPS for document exchange; SQL via external RDB engine; and CSV for data exchange.	The DOORS extension language allows data to be accessed more easily than SQL using standard high level programming techniques. DXL is easily programmable by those conversant with C such that SQL knowledge is unnecessary
8.1.4 Import of existing data from various standard file formats	Yes. Cradle provides several mechanisms for direct data import without the need to re-enter information: full CSV (comma separated value) capability for data exchange; the Cradle REQ parsing engine; the Cradle Word/Excel Plug-ins (to directly capture Microsoft Word and Excel data); the Cradle Applications Program Interface (API); and external tool data converters.	DOORS can import information in many forms such as MS-Word, ASCII, Spreadsheet, FrameMaker, Interleaf and RTF, so that structures, attributes and links may be set up automatically without manual input
8.1.5 Support Data Exchange Standards (AP-233, XML,..)	Yes, Cradle supports these Data Exchange Standards. The Cradle DOORS plug-in converts to AP-233 format files for import into the Cradle database. Cradle supports XML based file formats for information exchange, including: (a) XMI files for the exchange of model data, (b) SVG files for the presentation and exchange of diagrams.	No Response Added.
8.2 Intra-tool communications	See below.	See below.
8.2.1 Exchange of information between same-tool different installations	Cradle supports true client/server environments for information exchange. A user can also create export file(s) for information transmission and exchange. Cradle supports fully distributed projects (e.g., international), with simultaneous access to central or distributed data repositories by widely distributed users. This capability means that exchanges of data between Cradle installations can be minimized by simply distributing the project across the organizations involved.	DOORS offers a feature called DDM (Distributed Data Management) where users can export controlled sections of the database with read or read/write access to other databases. Data can be returned to the master database at any time with full merge abilities. Distributed parts can be defined down to the single requirement/single attribute level
8.2.2. Consistency/ comparison checking between same-tool datasets	Cradle provides a complete set of data consistency and integrity functions. However, with Cradle the need for such functions is minimal, since Cradle is fully multi-user and multi-project compatible, allowing users shared access to a common, shared database – irrespective of how widely separated the groups involved in the project are. It is far better to combine all users into a single shared repository as Cradle does, than to have to synchronize the efforts of separate user groups into an integrated whole.	DDM (see above) and the import methods provided by DOORS include the capability to compare data to see if it is new or existing. These comparisons allow updating of existing data and creation of new requirements. Updates can show inconsistencies and variations between data sets
9. System Environment	The Cradle infrastructure is fully robust and completely scalable.	See below.
9.1 Single user/multiple concurrent users	Cradle supports both user environments. Cradle is a fully multi-user environment, has provided these facilities since 1990, and has demonstrated continued success in large distributed projects. Cradle has been deployed in very large installations with over 1,000 active desktops and more than 500 concurrent users.	Multiple concurrent
9.2 Multiple Platforms/Operating Systems?	Cradle supports all versions of Windows including 95, 98, ME, NT4 (SP3 or later), 2000 SP1 or later, XP and 2003; SunOS 4.1.4, Solaris 2.5.1 or later; IBM AIX 4.2 and 5.1 or later; HP-UX 10.20 or later; and all types of Linux built on a version 2.4 or later kernel (such as Mandrake, RedHat Enterprise 3). Any Cradle component running on any platform can link to the same or any other component of Cradle running on the same, or any other, platform. The format of all project databases is identical across all platforms. This means that databases can be moved between Cradle systems platforms without any need to convert them.	Microsoft Windows NT 4, 2000, XP, 2003 server, Sun Solaris 7 and 8HP/UX 11
9.3 Commercial vs. Proprietary database	Cradle has a proprietary and open database. Cradle databases are processed as relational databases whose attributes can contain, or reference, BLOBS (binary large objects). Each Cradle database is a collection of 46 CISAM file groups. Each file group contains an arbitrarily large collection of data records that are simultaneously indexed by multiple indexes (some file	Proprietary, open object oriented repository

	groups have over 30 concurrent indexes) to provide pre-sorted information retrievals in multiple different sort orders. This eliminates the need for Cradle tools to retrieve large amounts of data and then sort it locally.	
9.4 Resource Requirements	See below.	System Requirements – Windows Server Windows 2000, Professional and Server Edition (SP2), Windows XP, Windows NT Version 4 Service Pack 6a TCP/IP System Requirements – Windows Client Windows 2000, Professional and Server Edition (SP2), Windows XP, Windows NT Version 4 Service Pack 6a TCP/IP for client install System Requirements - UNIX Hewlett-Packard HP-UX B.11.0 (64-bit version) with the following patch: QPK1100 B.11.00.58.5 Quality Pack for HP-UX 11.00, September 2002 Contact HP directly if you require this patch. Sun Solaris 8 or 9 Redhat Linux 8.0
9.4.1 Memory requirements	Memory requirements: For Windows platforms: 64MB minimum, 128MB if host is also running CDS. Additional 64MB for XP, 2003. For UNIX and Linux platforms: 32MB minimum.	Windows Server - 128Mb of RAM MORE than recommended for your particular windows system. Windows Client - On Windows 2000 , Windows XP and Windows NT, at least 128Mb RAM UNIX - Follow the manufacturer's recommendation for RAM
9.4.2 CPU Requirements	CPU requirements: For Windows platforms: 233MHz or faster is preferred. For UNIX platforms: minimal, any configuration is acceptable. In practice, the more users there are, the higher performance server that is needed.	Windows Server - 350 MHz Pentium processor. Windows Client - On Windows 2000, Windows XP and Windows NT, at least a 350 MHz Pentium processor.
9.4.3 Disk space requirements	For Windows platforms: 70-220 MB, depending on components loaded. For UNIX platforms: 220 MB.	Windows Server - At least 100MB of free disc space. Windows Client - On Windows 2000, Windows XP and Windows NT, at least 128Mb RAM. For standalone/typical client install, at least 50Mb of free disc space. UNIX - At least 100Mb disk space recommended for installation and use.
10. User Interfaces	See below.	See below.
10.1 Doing one thing while you are looking at another	Yes. The modular architecture of Cradle allows a user (or users) to perform simultaneous operations on the database. For example, while viewing and analyzing a set of requirements (or building a design model), the user can also be running a report with the Document Publisher.	Yes.
10.2 Simultaneous update of open views	Yes. In the WorkBench module, Cradle allows any number of views to be opened on the same or different sets of information at the same time. Any	Yes.

	changes made through one view will be automatically propagated through all other views.	
10.3 Interactive input/control of data	Yes. Cradle provides a graphical editor for creating multiple types of diagrams, including: functional, behavioral, hierarchical, process, data flow, architecture and object oriented. Numerous notation styles within each graphical type are supported. The graphical models are input via a point-and-click interface to the graphics icon library, and manipulated or modified by a drag-and-drop process. The graphics icon symbols can be replaced by (GIF or JPEG) images to more clearly convey the intent of the graphics. Graphics (diagrams and images) in Cradle can easily be viewed and output for publication or presentation via the Document Publisher. Similarly, Cradle database entities (items, attributes, types, etc.) can be input, controlled and viewed using the WorkBench query and data form user panes.	Yes.
10.4 Which window standard do you follow?	Microsoft Windows.	The DOORS user interface is based on Windows 2000 wherever appropriate
10.5 Executable via scripts (recordable) or macros	All repetitive operations in Cradle have a macro capability, such as requirements parsing, import/export, and performance analysis. These macros can be defined, stored and reused. Cradle tasking can also be controlled through its API, or through a fully user-definable Message Programming Interface (MPI).	Automation of tasks is supported through a user configurable language. These automated scripts may then be added as menu items and appear just like other functions of the tool. Alternatively scripts may be run from the operating system command line for batch operation
10.6 Web browser interface?	Yes. Cradle can support secure web access and provides secure access control logic within its web and non-web environments. The Cradle Web Access (WEBA) module allows remote users access to Cradle databases over the Internet or corporate intranet using only a web browser and an appropriate project authorization. Users access via browsers pointed at the Cradle Web Server (CWS), where they can login to a database via a login screen. Users can create, view, edit and delete items, and manipulate and follow cross-references using customizable screens. Also, by using report writing procedures provided in the Web Publishing Module (WEBP), a user can create HTML documents that can be viewed by a web browser.	Yes, DOORSnet can be used as an interface to the central requirements repository through a web browser.
10.7 Edit Undo Function Support	Yes, in both text and diagrams.	Yes. Single level undo and the ability to revert back to any previous version of an attribute in the history without having to step back through all of the intermediate changes
11. Standards--which one(s) do you comply with?	All web-based and non-web-based access to Cradle databases is controlled by a password protected login process that is compliant with BS 7799 and ISO 7799. Links between web browsers and the Cradle Web Server (CWS) can use HTTP or HTTPS protocols. See Section 6.1 for documentation standards.	Documents adhering to various military and commercial standards in multiple output formats can be produced from DOORS. DOORS also supports the needs of the disabled as defined in Section 508 of the US Disabilities Act. Our development processes are ISO 9001 compliant
12. Support and Maintenance	3SL is extremely proud of its excellent training, technical support and tool maintenance programs.	See below.
12.1 Warranty	Cradle has a 90-day warranty. We also offer an Annual Maintenance Program that provides free software upgrades and tool operational support.	Yes. 30 days.
12.2 License policy (Network, Node Locked,..)	Cradle has connection and tool licenses; both are floating (available from any machine on the network) and dynamic (active when a user starts a tool, released when user closes tool). Connection licenses are Read-Write, or Read-Only. The Cradle License Manager is proprietary.	Yes, FLEXlm
12.3 Maintenance & upgrade policy	Major software releases are provided at least annually, available at no cost under the Annual Maintenance Program. Official upgrade downloads with release notes are also available quarterly from the 3SL corporate ftp site to users under the maintenance program.	DOORS is normally released every 6 - 12 months with additional patches to facilitate support.
12.4 On-line help	Yes. The Cradle CD, the 3SL web site and the corporate ftp site all maintain a current full set of Manuals and Data Sheets in PDF format (with browser	Yes.

	to access and read). The Cradle software comes with online help. 3SL now offers Internet-based support to all customers with active maintenance contracts who have access to a PC or UNIX workstation with a web browser and Internet access. Our web-based service allows us to create a web meeting in which our engineers can run Cradle systems that appear on the user's screen. The user can control the Cradle systems (that are running on a 3SL PC) to ask a question, illustrate a problem, or propose a desired enhancement.	
12.5 Internet access/Website	E-mail: info@threesl.com Web Site URL: http://www.threesl.com	http://www.telelogic.com/doors
12.6 Phone support	Telephone Hotline Support available 12 hours per day, 7:00am to 7:00pm (EST in US, GMT in UK) US: 1-937-832-8529 UK: +44 (0) 1229 838867 E-mail requests encouraged: support@threesl.com	Tool support is available during normal business hours around the world. Weekend and out of office support can also be arranged.
12.7 Support users group	Formulation of Cradle Users Group under evaluation.	Yes. Telelogic hold annual User Conferences around the world. Please visit our website for details.
13. Training	See below.	See below.
13.1 Tool-specific training classes	Wide variety of Cradle training available - from overview sessions to comprehensive tool usage workshops.	Yes. Please see our website for details.
13.2 Training available at customer's location	Yes, training available onsite as needed. Typical class size is 8 - 10 students.	Yes.
13.3 Recommended training time	1-2 days for Requirements Management and Analysis. Additional training available for other Cradle modules.	The tool can be learnt in a single day although Telelogic provide addition training courses around Requirements Management. Please see our web site for details.
13.4 Software installation with only basic training	Available to fit customer needs. Our consultants are trained to provide onsite or online installation support and Cradle Systems administration training.	Tool training is not required for installation

Table 2. INCOSE RM Survey Answers for Cradle 5.3 and DOORS/ERS

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Dr. John Osmundson
   Naval Postgraduate School
   Monterey, California

4. David M. Hicks
   Northrop Grumman Ship Systems
   Pascagoula, Mississippi

5. Dave W. Whiddon
   Northrop Grumman Ship Systems
   Pascagoula, Mississippi

6. Dave T. Williams
   Northrop Grumman Ship Systems
   Pascagoula, Mississippi

7. Robert Rifley
   Northrop Grumman Ship Systems
   Pascagoula, Mississippi

8. Schuyler Otis Bland Library
   United States Merchant Marine Academy
   Kings Point, New York