NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

A MODEL OF PRE-REQUIREMENTS SPECIFICATION (pre-RS) TRACEABILITY IN THE DEPARTMENT OF DEFENSE

by

Robert C. Laubengayer, LT, USN
and
Jeffrey S. Spearman, LT, USN

June, 1994

Thesis Advisor: B. Ramesh

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
*June 1994

3. REPORT TYPE AND DATES COVERED
Master's Thesis

4. TITLE AND SUBTITLE *A MODEL OF PRE-REQUIREMENTS SPECIFICATION (PRE-RS) TRACEABILITY IN THE DEPARTMENT OF DEFENSE

5. FUNDING NUMBERS

6. AUTHOR(S) *Laubengayer, Robert C. and Spearman, Jeffrey S.

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)

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)*

*The purpose of this thesis is to explore the current DoD initial system development process and develop a model of Pre-Requirements Specification (Pre-RS) traceability. The model is based on a comprehensive study of stakeholder needs during development of large scale, software intensive systems. The motivation for this research is that current DoD standards require traceability and these standards do not specify what information should be captured. A field study of nine independent DoD organizations involved in initial systems development was conducted to determine how traceability is used to ascertain the information needs of various stakeholders. The model developed in this research provides a basis for formulating guidelines on implementing Pre-RS traceability in DoD.

14. SUBJECT TERMS *Pre-Requirements Specification Traceability, Pre-RS Traceability, Requirements Traceability, Traceability.

15. NUMBER OF PAGES 88

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

A Model of pre-Requirements Specification (pre-RS) Traceability in the Department of Defense

by

Robert C. Laubengayer
Lieutenant, United States Navy
B.S., United States Merchant Marine Academy, 1986

and

Jeffrey S. Spearman
Lieutenant, United States Navy
B.S., Southern Illinois University, 1987

Submitted in partial fulfillment
of the requirements for the degree

MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY

from the

NAVAL POSTGRADUATE SCHOOL
June 1994

Author: ______________________
Robert C. Laubengayer

Author: ______________________
Jeffrey S. Spearman

Approved by: ______________________
Balasubramaniam Ramesh, Principal Advisor

____
Carl Jones, Associate Advisor

____
Paul Moose, Chairman
Command, Control, and Communications Academic Group

ABSTRACT

The purpose of this thesis is to explore the current DoD initial system development process and develop a model of Pre-Requirement Specification (Pre-RS) traceability. The model is based on a comprehensive study of stakeholder needs during development of large scale, software intensive systems. The motivation for this research is that current DoD standards require traceability and these standards do not specify what information should be captured. A field study of nine independent DoD organizations involved in initial systems development was conducted to determine how traceability is used to ascertain the information needs of various stakeholders. The model developed in this research provides a basis for formulating guidelines on implementing Pre-RS traceability in DoD.

TABLE OF CONTENTS

I. INTRODUCTION

A. THESIS OBJECTIVES

The objective of this research is to develop a model of pre-Requirements Specification (pre-RS) traceability. The model will be based on an empirical study of the needs of various stakeholders involved in requirements generation during the initial development of large scale, complex systems. Several definitions for Requirements Traceability have been proposed in literature. The following two definitions, that explicitly define the aspects of traceability this research addresses, are used in this thesis:

*"**Requirements Traceability** refers to the ability to describe and follow the life of a requirement, in both a forwards and backwards direction (i.e., from its origins, through its development and specification, to its subsequent deployment and use, and through periods of on-going refinement and iteration in any of these phases)"* (Gotel and Finkelstein, 1993).

*"**Pre-requirements specification (pre-RS) traceability**, is concerned with those aspects of a requirement's life prior to its inclusion in the RS (requirement production)"* (Gotel and Finkelstein, 1993).

This work will explore the practice of pre-RS traceability in the Department of Defense (DoD). Traceability relationships (or linkages) that exist between outputs will be explored. Previous research, at the Naval Postgraduate School, Harrington and Rondeau, 1993, developed a requirements traceability model, depicted in Appendix A, for

Requirements Management. This research identified the system objectives as the primary source of requirements. This thesis will refine their model as it relates to the development of initial requirements.

The development of a pre-Requirement Specification (pre-RS) traceability model which represents various components (agents, outputs, inputs) and the relationships among them is the primary goal of this thesis.

Given the above goal, several questions must be answered:

- What are the components and relationships in the DoD's Requirements Generation Process?
- How should these components and relationships be organized in a traceability model?
- Who are the stakeholders involved in the pre-RS phases of a systems development life cycle?
- How does capturing requirements traceability support the stakeholders tasks in systems development?

B. METHODOLOGIES

Three data collection methods were utilized to determine the current practices of pre-RS traceability: literature review, focus group field studies, and field interviews.

The literature review included previous research on requirements traceability, systems development methodologies and DoD practices This research provided a thorough background of issues associated with initial systems development

A Focus Group is a planned and moderated group discussion designed to obtain information on a specific area of interest in an environment where "disclosures" are encouraged Groups are small and are composed of people who have some homogeneous characteristic that allows meaningful data collection on a particular topic The data gathered is qualitative in nature and can offer rich insights into the subject matter being researched As ideas and perceptions are shared, synergism often develops ıhat provides results not obtainable from other research methods

In this research, a total of 32 subjects in nine focus groups, with 2 to 6 participants each, discussed relationships and possible components of a pre-RS traceability model The discussions focused on the type of information that could support each stakeholder and should be captured by a pre-RS traceability scheme

Focus groups were conducted at:

- Naval Command & Control and Ocean Surveillance Center Research & Development Division (NRaD), TAC-3 Program Management Office, San Diego, CA
- NRaD, Systems Integration Division, San Diego, CA - Space and Electronic Warfare Command (SPAWAR) PMW-152, Program Management Office, Washington, D C

- Defense Information Systems Agency (DISA) Joint Interoperability and Engineering Organization (JIEO), Joint Interoperability and Testing Division, Reston, VA.
- Air Combat Command (ACC), Director of Requirements (DRI), Directorate of Theater Battle Management, Hampton Roads, VA.
- ACC, Langley Air Force Base (AFB), Program Management Office, Langley, VA.
- ACC 1912TH Squadron (CTAPS), Langley AFB, VA.
- Naval Aviation (NAVAIR), F/A-18(E/F) Program Management Office, Washington, D.C.

These organizations procure and manage the development and integration of aircraft, communications, command and control, and weapons systems.

The participants represented many levels of expertise in the areas of Concept Development and Requirements Development. The average years of education (after receiving a high school diploma) was 5.7 years, representing the following degrees: PhD, MBA, MS, MA, BS, BA. These degrees were from various academic areas: Electrical Engineering, Education, Business Administration, Computer Science, Command and Control, Computer Engineering, Mechanical Engineering, Systems Engineering, Business Management, Chemical Engineering, Economics/Mathematics, International Relations, Information Systems, Public Administration, Consumer Studies, Operations Research, and

Psychology. Additionally, the participants had an average experience of 13.4 years in systems development.

The participants had experience in several key areas of initial system development, such as: Project Management, Command and Control interoperability, Software Engineering Training, Functional Process Improvement, Program Reviews, Teaming, Communications and Networking Integration , Requirements Analysis, Requirements Management, Software Testing, System Integration, RDT&E, Configuration Management, Procurement, Acquisition Support, Development Support, and Modeling.

One-on-one interviews were conducted with several individuals that were unavailable to participate during the focus group sessions. Additionally, interviews of focus group participants were conducted where more detail on their responses was needed.

C. SCOPE AND LIMITATIONS OF STUDY

This research is designed to develop a pre-RS traceability model for the early phases of a systems life cycle prior to the specification of requirements. The data collection was limited to DoD organizations that follow a documented Requirements Generation Process.

D. ORGANIZATION OF THESIS

Chapter II provides a background into the DoD requirement generation process, associated documentation, the stakeholders involved, and provides initial guidance for pre-RS traceability during initial systems development.

Chapter III discusses the authors pre-RS traceability research observations of the DoD requirements generation process, and the remarks of initial systems development practitioners.

Chapter IV is the development of a model that depicts the semantics of multiple traceability linkages between system components during initial systems development.

Chapter V furnishes recommendations from the research and summarizes the findings and analysis.

Appendix A shows the Requirements Management Model as developed by Harrington and Rondeau. An "area of interest" is highlighted to refine their model for pre-RS traceability concerns.

Appendix B is a listing of Military Standards that could be referenced during initial systems development.

Appendix C provides a listing of Commercial-off-the-Shelf (COTS) prototyping tools that could be used to supplement pre-RS traceability.

Appendix D presents "examples" for initial systems development approaches to Concept Engineering and Requirements Development that emphasize traceability.

E. ACKNOWLEDGMENT

The authors would like to thank the participants of the focus group interviews. Special thanks to Mr. Larry Core and Mr. Reeve Peterson of NRaD; Captain Kathleen Welch (USAF) of ACC 1912TH Squadron; Lt.Cdr. Mike Quinn (USN) of SPAWAR

(PMW-152); Captain Steve Soderholm (USAF) of ACC Langley AFB; Major Norm Stewart (USAF) of ACC/DRI; Major Steve Maida (USA) and Lt.Col. John Daywalt (USAF) of DISA/JIEO; and Captain Dubois (USN) of NAVAIR.

II. A CURRENT SYSTEMS DEVELOPMENT PROCESS

This chapter discusses the Department of Defense (DoD) Requirements Generation System, associated documentation, the stakeholders involved, and provides initial guidance for pre-RS traceability during initial systems development. The DoD Requirements Generation System is a process. Descriptively, it is the interactions that exist during Federally funded initial systems development. Figure 2-1, shows the DoD acquisition Milestones and Phases that are part of a system(s) life cycle. Our discussion is limited to the "Mission Need Determination" to the successful Milestone 1 approval. Phases and Milestones that occur after Milestone 1, are involved in the systems production aspects of the DoD's acquisition process.

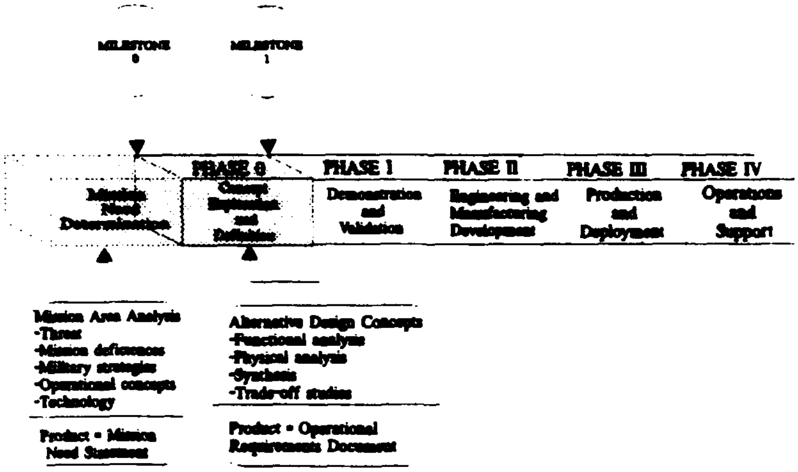
MILESTONE 0
MILESTONE 1
PHASE 0
PHASE 1
PHASE II
PHASE III
PHASE IV
Mission Need Determination
Concept Exploration and Definition
Demonstration and Validation
Engineering and Manufacturing Development
Production and Deployment
Operations and Support
Mission Area Analysis
-Threat
-Mission deficiencies
-Military strategies
-Operational concepts
-Technology
Product = Mission Need Statement
Alternative Design Concepts
-Functional analysis
-Physical analysis
-Synthesis
-Trade-off studies
Product = Operational Requirements Document

Figure 2-1: Systems Lifecycle

Due to the size and complexity of today's DoD systems, the entire systems development process has become quite challenging to manage (Fox, 1988). In the development of large scale, complex systems, it is essential to maintain the traceability of requirements to various outputs produced during the system's initial design process (Roetzheim, 1991). The following is an outline of the initial systems development processes of the DoD Requirements Generation System, the Stakeholders involved, and some insight to the distinct elements that can comprise a pre-RS traceability scheme.

A. REQUIREMENTS GENERATION SYSTEM

DoD Directive 5000.1 (Defense Acquisition) establishes policies for an effective interface among the three major decisionmaking support systems affecting defense acquisition. The three support systems are the Requirements Generation System, the Acquisition Management System, and the Planning, Programming, and Budgeting System as shown in Figure 2-2 . A discussion of the requirements generation system is appropriate, as it provides a process description of how requirements are generated from a concept. A "concept" is defined as a "possible" solution to an articulated operational need or deficiency.

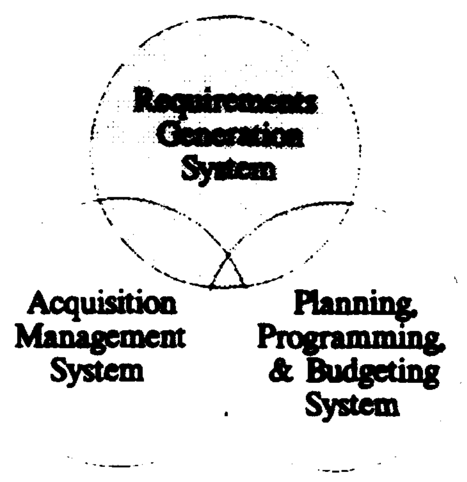

Figure 2-2: Three Major Decision Making Support Systems

The requirements generation system is intended to be uniform throughout the DoD. Specifically, the generation of requirements consist of the following four phases. definition, documentation, validation, and approval (CJCS MOP 77, p.3, 1992).

1. Definition

Definition is the activity that initially defines and justifies a mission need to fulfill a capability deficiency or exploit a technological opportunity. Through a Mission Area Analysis (MAA), DoD components evaluate current and projected capabilities with respect to changing threats, policy, guidance, military strategy, and assigned missions to identify deficiencies. This analysis should delineate the need for a material or non-material solution. If a material solution must be pursued, the Definition activity translates the deficiency or technological opportunity into a Mission Needs Statement (MNS) in broad operational capability terms (non-system specific) and operational constraints.

2. Documentation

Documentation is the formal preparation of the required and standardized documents in accordance with DoD 5000.2-M.

a. *The Mission Need Statement*

The MNS will be a nonsystem specific statement of operational capability need. The MNS will comply with the format as discussed in DoD 5000.2-M, page 2-1-1. Five elements are outlined as required documentation:

1. Defense Planning Guidance: Identifies the major program planning objective or section of the Defense Planning Guidance to which this need responds.

2. Mission and Threat Analysis: Identifies and describes the mission need or deficiency.

3. Nonmaterial Alternatives: Discusses the results of the mission area analysis.

4. Potential Material Alternatives: Identifies known systems or programs addressing similar needs that are deployed or are in development or production by any of the Services or Allied nations.

5. Constraints: Describes key boundary conditions related to infrastructure support that may impact on satisfying the need.

b. The Operational Requirements Document

The Operational Requirements Document (ORD) contains performance and related operational elements for the proposed concept or system. The ORD is an evolutionary document that describes a concept and reflects "system level" performance capabilities. The elements that are required in the document are:

1. General Description of Operational Capability.

2. Threat.

3. Shortcomings of Existing Systems.

4. Capabilities Required.

   a. System Performance
   b. Logistics and Readiness
   c. Critical System Characteristics

5. Integrated Logistics Support.

   a. Maintenance Planning
   b. Support Equipment
   c. Human Systems Integration
   d. Computer Resources
   e. Other Logistics Considerations

6. Infrastructure Support and Interoperability.

   a. Command, Control, Communications, and Intelligence
   b. Transportation and Baseing
   c. Standardization, Interoperability, and Commonality
   d. Mapping, Charting, and Geodesy Support
   e. Environmental Support

7. Force Structure.

8. Schedule Considerations.

3. Validation

Validation is the formal review process of the documentation by an operational authority other than the user to confirm the identified need and operational requirement.

4. Approval

Approval is the formal or official sanction of the identified need and/or operational capabilities described in the documentation.

Each concept proposed at Milestone I, Concept Demonstration Approval, will be described in the initial ORD in terms of minimum acceptable requirements (thresholds) that define the system capabilities needed to satisfy the MNS. Once a program is approved, the operational requirements for the concept(s) selected will progressively evolve from broad operational capability needs described in the MNS to system specific functional/non-functional requirements found in the ORD.

B. STAKEHOLDERS IN THE REQUIREMENTS GENERATION SYSTEM

A number of stakeholders, each having a different set of goals and priorities, are involved in the requirements generation system. The Definition and Documentation (Def/Doc) activity is the "central" stakeholder who is accountable for the program throughout the requirements generation system. The remaining stakeholders provide the need, assistance, validation, and approval.

1. The End-User

The "need" originates from the end-user or warfighter. The Commanders-in-Chief's (CINCs), Component Commands, and Services are the end user's. In DoD the end-user is the only entity who can submit a MNS (DoDD 5000.1, p.2-2, 1991). CINCs and Component Commands identify their mission needs to the responsible Service Component Commander. The Component Commanders will then coordinate the Def/Doc activities through their own Service's requirements generation system and keep the CINCs apprised on the MNS's status (CJCS MOP 77, p.10, 1992). The Services will define mission needs and operational requirements and will develop and coordinate the documentation with affected Services, CINCs, and Agencies (CJCS MOP 77, p.11, 1992).

2. Assistance

The Services keep assistance facilities (i.e., NRaD) seperate from the Def/Doc activities. This facilitates the Def/Doc activity to only query these facilities when technological, engineering, or other assistance is needed. This allows the facilities to continue with research in their areas of expertise, and not be burdened by the acquisition process.

3. Validation

Stakeholders in this activity are the Defense Intelligence Agency (DIA), Director J-6, and the Joint Requirements Oversight Council (JROC).

DIA validates the potential threat to be countered and the projected threat environment, and certifies intelligence requirements.

For any Command, Control, Communications, Computers (C4) capability, the Director, J-6, Joint Staff, must certify the need and operational requirements for conformance to joint (multi-service) C4 policy and doctrine, interoperability, architectural integrity, and joint potential.

The JROC validates all potential joint programs. The JROC utilizes the Defense Information Systems Agency, Joint Interoperability and Engineering Organization (DISA/JIEO) as their validation stakeholders.

4. Approval

Approval activity stakeholders ensure the MNS and ORD conform to the DoD 5000 series, indicate a joint potential designator (JPD), and may recommend the lead Service or Agency for programs involving more than one DoD component.

C. FOUNDATIONS FOR PRE-REQUIREMENTS SPECIFICATION TRACEABILITY

The Def/Doc activity responds to any changes, recommendations, alternatives, and has the most knowledge of the end-user's Need, the MAA, the MNS, and the ORD. Therefore, the authors believe that the primary responsibility for a pre-RS traceability scheme should reside with the Def/Doc activity. The distinct elements that can comprise a pre-RS traceability scheme are the interactions that exist between the Def/Doc activity, the end-user, and other stakeholders as they relate to the MAA, the MNS, and the ORD. Figure 2-3 shows interactions that exist in the DoD Requirements Generation System.

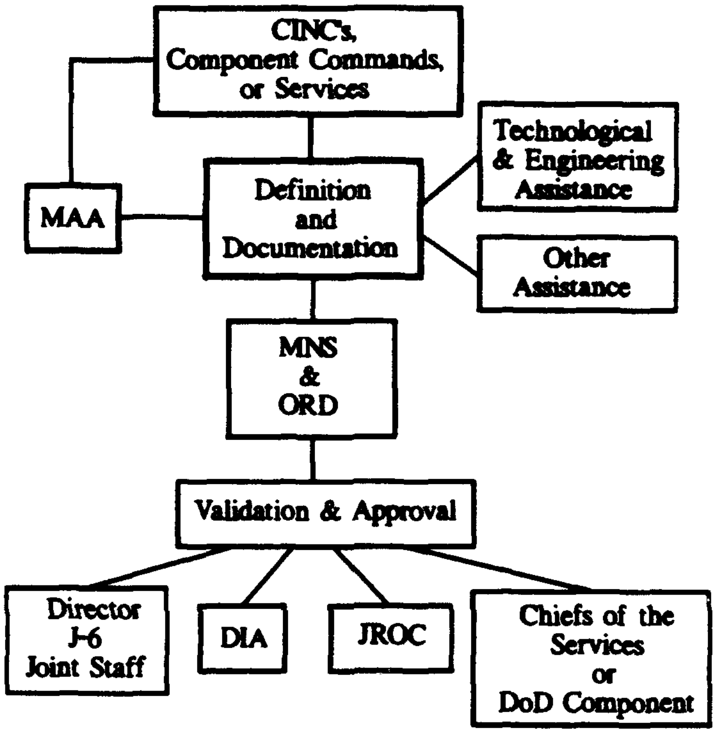
CINC's, Component Commands, or Services
MAA
Definition and Documentation
Technological & Engineering Assistance
Other Assistance
MNS & ORD
Validation & Approval
Director J-6 Joint Staff
DIA
JROC
Chiefs of the Services or DoD Component

Figure 2-3: Stakeholder Interactions

1. The Mission Area Analysis (MAA)

The end-user articulates a need or deficiency to the Def/Doc activity. The Def/Doc activity attempts to rationalize the end-users "problem space" through the MAA. Problem space is defined as the environmental and strategic needs that help define the operational need. The MAA is provided by the end-user and evaluates the problem space in respect to threat, National policy, technology, budget, capabilities required, military strategy, and current doctrine. This problem space analysis sets the boundaries where the operational need must reside. Capturing the environment in which the end-user defines their needs is an important element of traceability. Therefore, the MAA should be kept as an element of the pre-RS traceability scheme.

2. The Mission Need Statement (MNS)

The MNS is generated in response to the MAA and refines the initial problem space where the need or deficiency resides. The Def/Doc activity collects the information that generates the elements of the MNS. The elements that comprise the MNS are important artifacts for a pre-RS traceability scheme, as these elements can be traced back to the boundary conditions of the MAA and provide the foundation for the ORD.

3. The Operational Requirements Document (ORD)

The Def/Doc activity queries technological, engineering, and various other "assistance activities" to formulate the initial elements of the ORD. Alternative concepts are generated to provide the end-user with a satisfactory choice of a concept or characteristics of concepts that satisfy the need. A pre-RS traceability scheme could include all of the elements of the ORD. The elements are used to develop requirements for contract specifications during each acquisition phase (DoD 5000.2-M, p. 3-2, 1991).

4. Stakeholders Input

A pre-RS traceability scheme relies on the rationale, assumptions, decisions, and motivation for a systems existence. The stakeholders that normally have the most impact during the pre-RS phases of a systems development are the Def/Doc activity, the end-user, and the assistance activities. Capturing these stakeholders input to the systems development process is the most difficult task for the Def/Doc activity in formulating a pre-RS traceability scheme.

D. CONCLUSIONS

The goal of pre-RS traceability is to make systems requirements as well defined as possible. This should allow for a smooth transition into the contracted design production of a system. Historically, bridging the gap between systems requirements and design specifications of large scale, complex systems has been the most difficult aspect of systems development. Pre-RS traceability will aid in this transition by providing the contractor with stakeholders rationale, assumptions, and motivation for the systems existence.

III. DEVELOPING PRE-REQUIREMENTS SPECIFICATION TRACEABILITY FOR THE DOD

This chapter discusses observations of pre-RS traceability efforts in the DoD requirements generation process, based on data from the author's empirical research. The intent of this discussion is to provide a framework for a model of pre-RS traceability.

A. OBSERVATIONS OF INITIAL SYSTEMS DEVELOPMENT PROCESS

This section analyzes the DoD's initial systems development process and issues identified by the author's research while exploring the characteristics of a comprehensive traceability scheme.

1. A Lack of pre-RS Traceability Guidance

MIL-STD-2167A, *Defense System Software De ɜlopment*, is the primary DoD document that mandates requirements traceability. The author's observed that initial systems development should not initially address software issues. The basis of initial systems development should be on defining the "problem space" for systems where the needs are specified (i.e., in the MAA, MNS, ORD). A typical MIL-STD document is intended for "contractors", while pre-RS traceability is intended to be performed by Def/Doc activities where requirements get defined. Therefore, a major concern of all research participants, is that pre-RS traceability is not practiced by DoD. These participants also felt that pre-RS traceability would greatly assist them in their respective systems development activities.

2. Problem Space vs. Solutions

The preceeding chapter outlined the major "Problem Space" analysis documents that seek to bound the system under development. In large organizations like DoD, one stakeholders mission need is their supervisors operational need and so forth. These various levels of systems abstraction can be best described through problem space analysis. Too often DoD's initial systems development process pursues ad hoc solutions to satisfying the articulated need. With such an early commitment to system specific solution characteristics (i.e., baud rates, frequencies, etc.), it is easy to lose the intent of the customer. Pre-RS traceability hopes to aid the development team in distinguishing between the "problem space" [need] and the "solution" [system].

3. No Structured pre-RS Traceability Approach

Historically, the DoD has given contractors reams of documentation with no structure. The current method used by the DoD to specify requirements uses mostly a narrative, English format with supporting diagrams and charts. Ambiguities are frequent, as English specifications are inexact. If requirements are formally stated and can be transformed into designs in a formal manner, traceability between requirements and designs is a by-product of the design process itself. The MIL-STD-499B, *Systems Engineering*, seeks to provide guidance on the design process, but is also intended for contractors and does not address pre-RS stages. A listing of Military Standards and other references that encompass initial systems development in the DoD are provided in Appendix B.

4. The Need For Innovative pre-RS Approaches

Innovative approaches to systems development should seek to remove the ambiguities that reside in English narratives, supporting diagrams, and charts that are delivered to contractors. Low cost Commercial Off The Shelf (COTS) prototyping (or Storyboarding) and simulation software programs can be used to animate, design user interfaces, show various levels of the system (i.e., Strategic vs. Tactical), and can be brought to any stakeholder's organization to ensure that their needs are properly understood. The use of these COTS programs allows stakeholders to "try before they buy", and can be given to a contractor to build from. For a partial listing of COTS programs that could aid pre-RS stages of systems development, refer to Appendix C.

5. Adopting a Structured pre-RS Traceability Approach

Establishing a systems development hierarchy can greatly assist the development team in conceiving a traceability scheme. A *hierarchy of levels* can lead to a systematic approach to systems which have broad applications (Blanchard & Fabrycky, 1990). Appendix D presents outlines of possible approach's to the initial systems development process, that emphasize pre-RS traceability.

B. OBSERVATIONS FROM FOCUS GROUPS

This section analyzes the concerns and issues addressed by focus groups exploring the characteristics of a comprehensive pre-RS traceability scheme.

1. Customer is Driver for systems development

The customer is defined as the end-user's representative (i.e., an entity that holds accountability for the end-user). The customer is the driver for systems development as well as an invaluable source of traceability information. Department of Defense (DoD) policy is that the customer develop the operational requirements for all future systems [Chap. 2.B.1]. The ultimate customers for U.S. military systems are the geographical area Commander and Chiefs (CINC's). The CINC represents the needs of every entity that will use the system down to the operator. The customer's need must be satisfied with regard to quality, completeness, and accuracy of any system that is fielded. ("The system will most likely be reworked several times before finally being accepted by the customer, when the customer is not involved at virtually every level.")[1] Therefore, the customer should be involved throughout the systems design process, beginning at concept inception.

2. Development Teams mission

A common problem is that the customer is not necessarily able to articulate their need or operational requirement. Def/Doc activities [Chap. 2.A.1] are able to suggest

[1] This is a direct quote from a focus group subject. Henceforth subject quotes will be enclosed in parentheses and quotation marks, but no specific reference will be made.

possible alternatives and aid the customer in articulating their needs. ("We often help work the ORD because many times what the fleet (customer) says it wants and what it really wants are two different things.")

3. Team Building

Large scale projects consist of many design elements. The teams developing these elements must understand the customer need the system is being developed to meet, and how various design elements relate to each other. ("One of the things that every team has to do is to figure out what is going on. Even if they have domain knowledge, they have to figure out what is different about this one from any other one they've worked on."). The ability of team members to capture the intent and rationale of the customer is a fundamental necessity for any model of traceability. ("At each phase when you build a team, typically there are multiple teams because nobody has the skills to follow a program all the way through.") Team members capable of grasping the many complex relationships between the projects design elements are also essential to successful system design. Over time, these individuals possess the "Corporate Knowledge" of any project. ("Once you crystallize the need, you start injecting the domain knowledge of what it is you are building. That comes from the peoples heads who are working on the project.") The "corporate knowledge" should permeate throughout the project and a comprehensive traceability scheme would aid this process. Traceability serves as an excellent means of augmenting the skills and knowledge of the development team associated with a project, if it can capture the various process histories and the alternatives. The capture of the justifications, assumptions, and reasoning behind the teams iteraction by a traceability tool could reduce costs of the overall system. Traceability then becomes the means to access the "corporate knowledge".

4. Solutions instead of needs

In the course of developing systems, often the focus is on technology to be used rather than on addressing the fundamental requirements of the user that the system is intended to satisfy. With focus shifted to systems requirements, the customer's need that the system was designed to address can be lost. ("The technology is rapidly changing, but the technology does not have the requirement. The system doesn't have a requirement. The operator using the system has a deficiency.") One method of ensuring that the customer's need is well understood is through rapid prototyping, but several issues arise from this method of systems development. Rapid prototype systems ensure customer involvement throughout, but are extremely difficult to upgrade or maintain because of the lack of documentation on the development history of the system. ("We fielded a rapid prototype system that's a conglomeration of several things, it is not well designed especially from the maintenance point of view where it is missing the documentation and rationale. But, it does meet a lot of the operators needs. It has succeeded where a lot of the traditional acquisition attempts have failed.")

5. Traceability for pre-RS

Traceability should capture the requirements rationale (the why) of the customer requirements. This rationale can then be used to develop the working documents of a system such as development options papers (DOP). ("The Systems Commands [Navy] develop what is called the DOP. This goes back and says in order to meet this mission need, these are the technical risks, schedule risks, cost risks, and here are several alternatives to go about doing it.") The justifications used and decisions made should be captured to ensure that the intent of the system is documented. ("At all levels you want to be able to get an idea, the spirit. I think this really important. I think one of the boogie men of requirements over the years has been how do we convey the spirit. What we've settled for is the letter of the law [acquisition regulations].")

6. Requirements Hierarchy and Linkage to Other Systems

At each successive level of a systems development, requirements are generated. A traceability model should capture relationships between these levels. The hierarchy of requirements is horizontally linked to design specifications which have a tendency to be lumped into or confused with requirements. A hierarchy of design specifications should also be horizontally linked back to relevant requirements. The stakeholders should be able to distinguish between requirements and design specifications. ("Then you start having design artifacts, and at some point you are not only tracing requirements to other requirements you are also tracing requirements over to the design. You have both vertical and horizontal tracing.") The ability to identify the requirements verses design specifications allows changes to be evaluated by the concerned stakeholders. ("We trace requirements, maybe, but what we really want to do is trace design decisions. By the time you get to the ORD statement you might say: *"We are going to have modulation in that."*

We are inputting design decisions in it that break it into further requirements and that is what we don't trace.") Traceability should, at a minimum, capture what requirements exist and which design specification it is horizontally linked to. Traceability should identify which stakeholder made a decision and what requirements were derived. Traceability should capture the decision as it relates to other requirements, stakeholders, and system components that are affected by the decision. ("If you had the requirements traceability, then its not that much to ask for to say: *"this requirement came from this requirement up here"*.")

7. Information Needs of Stakeholders

System requirements originate from the customer and many other stakeholders. Each of these stakeholders have varying needs for the system. Stakeholders range from the technical sergeant involved with the system at the user level, to the Chairman of the Joint Chiefs of Staff (CJCS) answering national concerns, to a software programmer at the Computer Software Unit (CSU) level. The sergeant is interested in getting the console configured correctly, the CJCS would like to know that the system can operate as intended, and the software programmer is interested in what is to be programmed and how it interfaces with other CSU's. These dissimilar interests of the various stakeholders establish needs and in turn system requirements. A traceability scheme would allow each stakeholder to "observe" how their needs are being satisfied by accessing information that meets their concerns. A traceability scheme should allow stakeholders to identify the system requirements associated with the needs they have articulated. System designers can look at the stakeholders needs and ascertain if these needs are changing and what design specifications must also be change.

8. Interpretation and definition

A difficult part of system design is determining what the requirements are in a text document. ("Currently, we hand over a system specification and we can't tell them how many requirements are in there, because we typically go by paragraphs, and SHALL's make requirements WILL's don't. We typically can't come out and say we gave you 135 requirements.") By having no standard method for translating text into unambiguous requirements, it becomes difficult for the designers to clearly interpret the customer and stakeholders needs and expectations. Consistency of definitions, such as an object in one system translating to the same object in another system, would be extremely beneficial in translating requirements. ("So when you model over here, over here, and over here, and use three separate tools, "takeoff time" is the same for each system.") A traceability scheme should relate requirements templates to other requirements templates and contain a data dictionary that ensures text definitions are validated.

9. System Redundancy

Another key feature that traceability should be able to accomplish is to identify relationships between different systems and organizations that have comparable needs. ("I think traceability is important because it would allow other organizations to avoid duplication of ideas and concepts. Also because there is potential for one systems change to effect two or three other systems down the road that depend on that system for something.") Traceability of initial systems development would allow systems to be evaluated with respect to redundancy. ("We currently have two funded systems where redundancy exists. We know each of the systems do roughly what the other does, but with some alterations we could completely do away with one of them. Now, how did two systems get off the ground doing the same thing? Nobody can tell us what each of these was meant to do from the inception.")

10. Resources

One of the critical processes for any given project is the allocation of resources and the impact of resource constraints on the overall system. Man-hours, funding, and schedule impact every system design. Traceability is only mandated by DoD for software embedded systems. It is not considered crucial by some systems managers to the efficient management of their systems during initial systems development. Due to budget constraints, traceability is not done until absolutely necessary. Several focus group participants stated that if traceability is accomplished during initial systems development, it would reduce the amount of resources required to field a system. ("All of sudden, all of this traceability becomes very valuable. So, in terms of development costs you many not see traceability benefits, but in life cycle costs it will surely be beneficial.") Resources might have to be re-allocated to meet a deficiency that a system was designed to meet, if the system does not perform the tasks the customer had intended. ("So now when the product hits the street you get into the really expensive part of requirements, enhancing the product to do what the customer wanted in the beginning.") Traceability provides a structure to monitor resource allocations and provides the ability to identify those activities that could be cut during times of limited resources while preserving the customer need.

11. Critical Issues

a. Change Management

Requirements and needs of the customer are constantly changing with the dynamic nature of the technology available, the threat, and the mission. To keep abreast of this changing environment, a system should be evaluated to ensure that the needs of the customer are being met. The needs might change, and a traceability scheme could be used to follow the customer need, and may also allow the need to be altered to conform with the new environment.

b. Available Technology

Technology is changing at a rapid rate and what may have been the state-of-the-art even a year before can be out dated prior to any system production. A whole generation of computer hardware is currently being developed every eighteen months, possibly requiring that the system be changed to comply. Traceability could be used to capture this wholesale change. ("There are many cases that the current technology may have already been considered at the DOP level, but the technology may have matured by now, and what may have been eliminated earlier because of the risks associated with it in the past is now the technology of choice.")

c. *Reuse*

DoD policy at present has the systems designers evaluating Commercial or Government Off The Shelf (COTS/GOTS) systems which have already been fielded to meet the needs of other DoD components. Traceability of a fielded system may or may not exist. ("COTS often falls short of what is actually needed. If COTS can satisfy four out of five requirements...is that all right.") Traceability of the customer need that drives the requirements could aid in determining what the critical requirements are for a system.

d. *Stakeholder Interactions*

The ORD, sometimes, does not express the customer's intent properly. ("The ORD is at a very high level and intentionally ambiguous. The options and alternatives reside with the DOP [Navy].") The ORD defines the operational parameters for the formal DoD acquisition process, but the actual requirements are developed from the discussions between the development team and the customer. ("We take the technological options to the customer and they decide on which option serves their purpose. We then take that option, along with the original high level documents [MAA,MNS,ORD] and start generating specific requirements.") Traceability should capture this interaction to assist the designers, maintainers, and operators of a system throughout its life cycle.

e. *Modeling Needs*

The customer is important when modeling is used to help define the needs that the systems requirements are intended to meet. Several efforts are currently trying to model the functional processes involved with several DoD systems. The customer describes the activities and functions needed in order to complete their mission. ("The functional model which we build is a collection of activities performed and we associate this with current systems. And now it is essentially a baseline model of operator needs.") Traceability should capture the information from such sources.

C. CONCLUSIONS

The author's have presented their views of the DoD requirements generation process and those of the practitioners involved. Pre-RS traceability is a "needed" capability that would assist the development team in nearly all areas of initial systems development. These discussions suggest that pre-RS traceability must be performed for all large scale, complex systems that are developed by and for the DoD.

IV. A MODEL FOR PRE-RS TRACEABILITY

A. INTRODUCTION

A major challenge in this thesis is the development of a model that depicts the semantics of multiple traceability linkages between system components during initial systems development. Components can be described as tasks, agents, inputs, and outputs of the development process. Linkages describe the relationships between components. Focus groups consisting of initial systems development practitioners identified the components and their relationships to each other.

In this chapter, traceability linkages will be distinguished by uppercase, bold faced letters (**LINKAGES**), while components that they link are shown with uppercase, italic letters (*COMPONENTS*). For every link in the model an inverse may be defined.

B. A MODEL FOR PRE-REQUIREMENTS SPECIFICATION

A recurring premise of the focus group subjects was that traceability, when implemented correctly, would be highly beneficial and would ease understanding, capture, tracking, and verification of requirements generated later in the lifecycle. The following is a discussion of the pre-RS traceability model presented in Figure 4-1.

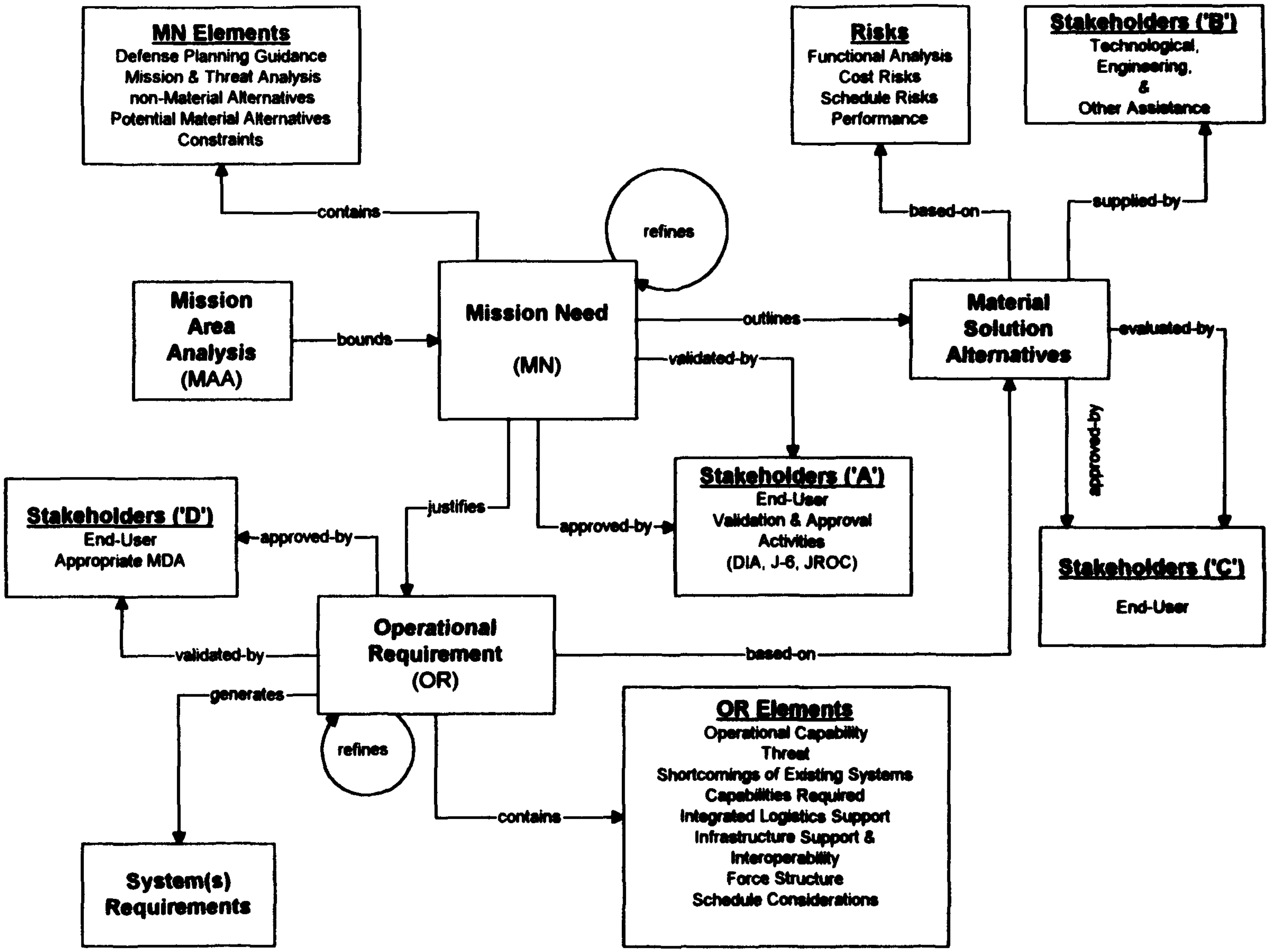

Figure 4-1: pre-RS Traceability Model

1. Mission Need

a. *MISSION AREA ANALYSIS* **BOUNDS** *MISSION NEED*

MISSION AREA ANALYSIS is the determination and exploration of the environmental and strategic needs based on future technologies, threats, and organizational goals. **BOUNDS** on what the *MISSION NEED* should and should not include are established by *MISSION AREA ANALYSIS* . The environment of the cold war and the possibility of nuclear war bounded most systems to the Soviet threat from the 1940s until recently. *MISSION NEED* is the operational capability to meet a deficiency found with regard to the strategic and environmental needs ("...the mission needs statement is based on the operational requirements documents so its a process.") For example a *MISSION NEED* could suggest development of an automated system to generate a comprehensive report including target, take-off, landing, and fuel information to fulfill the strategic and organizational needs of a military service.

b. *MISSION NEED* is **VALIDATED-BY** *STAKEHOLDERS ('A')*

The link **VALIDATED-BY** refers to the determination of whether the *MISSION NEED* meets the strategic and environmental needs as defined and understood by the *STAKEHOLDERS*. For instance, the draft Mission Needs Statements (MNS) is sent to the Joint Interoperability and Engineering Organization (JIEO) of the Defense Information Systems Agency (DISA). JIEO validates the draft MNS with regards to MNS ability to meet its joint needs as defined by the Joint Requirements Oversight Council (JROC) ("....we receive draft and approved MNS and ORDs (Operational

Requirements Documents) from the services and subsequently staff them out throughout JIEO centers, CINCs, Services, and Agencies and provide an assessment on interoperability, capability, and integration.") *STAKEHOLDERS* ('A') are those organizations that have validation responsibility for, or have a vested interest in, the *MISSION NEED*.

c. *MISSION NEED* is **APPROVED-BY** *STAKEHOLDERS ('A')*

APPROVED-BY is the link which specifies approval of the *MISSION NEED* by the *STAKEHOLDERS ('A')* that the *MISSION NEED* expresses the operational need of the *STAKEHOLDERS ('A')*. ("For Joint Services systems, the recommendation for approval is given to the Joint Requirements Oversight Council and this recommendation is an important wicket which the services must pass.") This illustrates that approval of a MNS by JIEO is required for further development of a system when interoperablity is a strategic need. *STAKEHOLDERS ('A')* are those organizations that have approval responsibility. DISA JIEO is a *STAKEHOLDER ('A')* because it has to approve the MNS and acts to insure the interests of the Joint Chiefs of Staff (JCS) via the JROC.

d. *MISSION NEED* **CONTAINS** *MISSION NEED ELEMENTS*

Analysis and planning to meet the strategic and environmental needs are contained within the *MISSION NEED*. Examples of *MISSION NEED ELEMENTS* are Defense Planning Guidance, Mission Analysis and Threat Analysis. *MISSION NEED ELEMENTS* also include material alternatives, which are systems, and nonmaterial alternatives which are changes in procedures or policy. *MISSION NEEDS* expressing a material alternative discuss the nonmaterial alternatives explored. A specific example is that the Marine Corps Combat Development Command (MCCDC) has developed Doctrine which is incorporated into the MNS for systems that involve the Marine Corps.

e. *MISSION NEED* **REFINES** *MISSION NEED*

The link **REFINES** is alterations to perfect and elaborate the *MISSION NEED* in order to meet the strategic and organizational needs. The MNS for DoD systems is reworked to meet the organizational needs and concerns for the Joint Chiefs of Staff.

2. Material Solution Alternatives

a. *MISSION NEED* **OUTLINES** *MATERIAL SOLUTION ALTERNATIVES*

MATERIAL SOLUTION ALTERNATIVES are material options that are capable of meeting the operational need as defined by the *MISSION NEED*. An example of this are the alternatives explored by DoD for the FA-18 E/F program. The program matured from a study called Hornet 2000 which explored *MATERIAL SOLUTION ALTERNATIVES* for a export model of FA-18 aircraft in 1987. ("There were three major configurations developed for this project and there were subconfigurations. Anyway configuration 3C looks a lot like what the aircraft looks today.") The *MISSION NEED* **OUTLINES** or provides the guidelines for a search of possible solutions to meet operational need. The DoD MNS is the basis for developing trade studies on the operational need. ("There are a lot of trade studies done in a cost and evaluation kinda phase before the actual ORD.")

b. *MATERIAL SOLUTION ALTERNATIVES* are **BASED-ON** *RISKS*

The *RISKS* associated with solutions provide the basis for *MATERIAL SOLUTION ALTERNATIVES*. *RISKS* are the dangers and hazards associated with technology, performance, and costs. For example the Navy program for an advanced attack aircraft A12 was canceled because the program costs became excessive due to the *RISKS* associated with it. The evaluations of the *RISKS* associated with cost, threat, and performance are considered essential. ("All those trade studies are cost, threat, performance.")

c. *MATERIAL SOLUTION ALTERNATIVES* are **SUPPLIED-BY** *STAKEHOLDERS ('B')*

Possible *MATERIAL SOLUTION ALTERNATIVES* to meet the operational need are furnished or **SUPPLIED-BY** *STAKEHOLDERS('B')*. e.g. The DoD research and development laboratories provide *MATERIAL SOLUTION ALTERNATIVES* to the Definition and Documentation Activity [2.A.1]. ("I dangled the Alternatives in front of them along with the rough costs and what kind of capability they could get for how much money and when.") *STAKEHOLDER('B')* are those organizations providing possible *MATERIAL SOLUTION ALTERNATIVES* to meet the operational need. e.g. Private and public research laboratories.

d. *MATERIAL SOLUTION ALTERNATIVES* are **EVALUATED-BY** *STAKEHOLDERS ('C')*

The link **EVALUATED-BY** is the determination and appraisal by *STAKEHOLDERS ('C')* as to how the *MATERIAL SOLUTION ALTERNATIVES* meet the operational need. *STAKEHOLDERS('C')* are those organizations and entities that assess the *MATERIAL SOLUTION ALTERNATIVES* . An example is that CNO staff members evaluates the different options and alternatives provided.

e. *MATERIAL SOLUTION ALTERNATIVES* are **APPROVED-BY** *STAKEHOLDERS ('C')*

APPROVED-BY refers to the approval of the *MATERIAL SOLUTION ALTERNATIVES* to meet the operational need as defined and understood by the *STAKEHOLDERS('C')* ("...there are several alternatives to go about doing it. We recommend, in order of priority, this way; this way; this way; based on risk. CNO then comes back and says "Okay, from your DOP [Chap 3.B.5] we going to take option X.") The *STAKEHOLDERS('C')* are those organizations and entities that approve the *MATERIAL SOLUTION ALTERNATIVES.*

3. Operational Requirement

a. *OPERATION REQUIREMENT* is **BASED-ON** *MATERIAL SOLUTION ALTERNATIVES*

OPERATIONAL REQUIREMENT is the requirement developed to meet the operational need as developed by the *MISSION NEED*. For DoD the Operational

Requirement Document expresses the *OPERATIONAL REQUIREMENT* for a system **BASED-ON** (i.e., developed from or supported by) the *MATERIAL SOLUTION ALTERNATIVES*. "So, the ORD for this program became a derived document based on these studies").

b. *OPERATIONAL REQUIREMENT* **CONTAINS** *OPERATIONAL REQUIREMENT ELEMENTS*

The link **CONTAINS** is defined as including standards previously employed so that the *OPERATIONAL REQUIREMENT* can meet the operational need. For DoD systems *OPERATIONAL REQUIREMENTs* must embody the force structure, logistic considerations, threat, and operational capability. *OPERATIONAL REQUIREMENT ELEMENTS* are constraints that mold the *OPERATIONAL REQUIREMENT* and define standards which the *PERATI NAL REQUIREMENT* must adhere to. DoD standards provide the structure and the ORD must include them ("MOP(Memorandum of Policy) 6212 says that...a standards profile is supposed to be part of the ORD.")

c. *OPERATIONAL REQUIREMENT* is **VALIDATED-BY** *STAKEHOLDERS ('D')*

The **VALIDATED-BY** link refers to the determination of whether the *OPERATIONAL REQUIREMENT* adheres to the operational need as defined and understood by the *STAKEHOLDERS('D')*. *STAKEHOLDERS('D')* are those organizations providing oversight that the *OPERATIONAL REQUIREMENT* meets the operational need that was developed in the *MISSION NEED*.

d. *OPERATIONAL REQUIREMENT* is **APPROVED-BY** *STAKEHOLDERS ('D')*

The link **APPROVED-BY** refers to the approval of the *OPERATIONAL REQUIREMENT* by *STAKEHOLDERS('D')* that it meets the operational need. For DoD the ORD must be approved according to law by the appropriate Milestone Decision Authorities [chap 2.A.4]. *STAKEHOLDERS('D')* are those organizations and entities that approve the *OPERATIONAL REQUIREMENT*.

e. *OPERATIONAL REQUIREMENT* **REFINES** *OPERATIONAL REQUIREMENT*

REFINES refers to alterations to perfect and polish the *OPERATIONAL REQUIREMENT* in order to meet the operational need of the *STAKEHOLDERS*.

f. *MISSION NEED* **JUSTIFIES** *OPERATIONAL REQUIREMENT*

The **JUSTIFIES** link prescribes that the *OPERATIONAL REQUIREMENT* must maintain and assert the operational need the *MISSION NEED* expresses.

g. *OPERATIONAL REQUIREMENT* **GENERATES** *SYSTEM REQUIREMENT*

The *OPERATIONAL REQUIREMENT* **GENERATES** or creates and brings into existence the *SYSTEM REQUIREMENTS*. *SYSTEMS REQUIREMENTS* are system specific requirements that are developed to meet the *OPERATIONAL REQUIREMENT*.

V. CONCLUSIONS AND RECOMMENDATIONS

A. DOD INITIAL SYSTEMS DEVELOPMENT PROCESS

The Department of Defense has in place a formal acquisition process that provides a foundation for systems development. DoD Standard 2167A requires traceability documentation for software intensive systems. However, this standard does not explicitly detail what traceability information should be captured. This research identifies the information needs of the stakeholders during initial requirements development using the DoD acquisition process.

B. PRE-RS TRACEABILITY MODEL

This research has developed a model of pre-Requirements Specification traceability, as described in Chapter IV, based on the information needs of various stakeholders in initial system development. This model provides a basic structure from which pre-RS traceability can be conducted. This model is based on information gathered from the focus groups and a review of the DoD acquisition process. Such a model would provide a conceptual basis for formulating guidelines on implementing pre-RS traceability in DoD.

C. PARTICIPANTS

Recommendations from the parties to improve the pre-RS traceability of DoD systems were varied and numerous. Some stakeholders needed pre-RS traceability to act as a repository of system histories. While others needed pre-RS traceability to explore the functions within system development. All recommended traceability to reduce the costs associated with redevelopment or system alterations.

D. THE NEED FOR FORMAL GUIDANCE

There is an articulated need to have pre-RS traceability information, but there are no formal requirements or guidance on the subject within theDoD acquisition process.This research indicates a strong need for guidance on how traceability information should be captured and how the information should be used. The ability to follow the systems development from inception to completion and back will provide stakeholders involved the ability to adopt to change in a more efficient manner. In the current dynamic environment, comprehensive pre-RS traceability would prove extremely beneficial.

E. SUGGESTIONS FOR FUTURE RESEARCH

Research to follow-on to this work should include:

- A validation of the pre-RS Traceability Model using a DoD system currently under development.
- Further investigation into adopting a standardized approach in concept development to requirements development stages, such that traceability is a natural outcome of the process.
- Development of a model that details the interactions that exist during the transfer from systems requirements -to- design specifications.

APPENDIX A.

REQUIREMENTS MANAGEMENT MODEL

A. INTRODUCTION

Figure A-1 shows the Requirements Management Model as developed by previous research conducted at the Naval Postgraduate School (Harrington, Rondeau, 1993).

An "area-of-interest" is highlighted to depict the portion of the model that is concerned with initial systems development and pre-RS traceability.

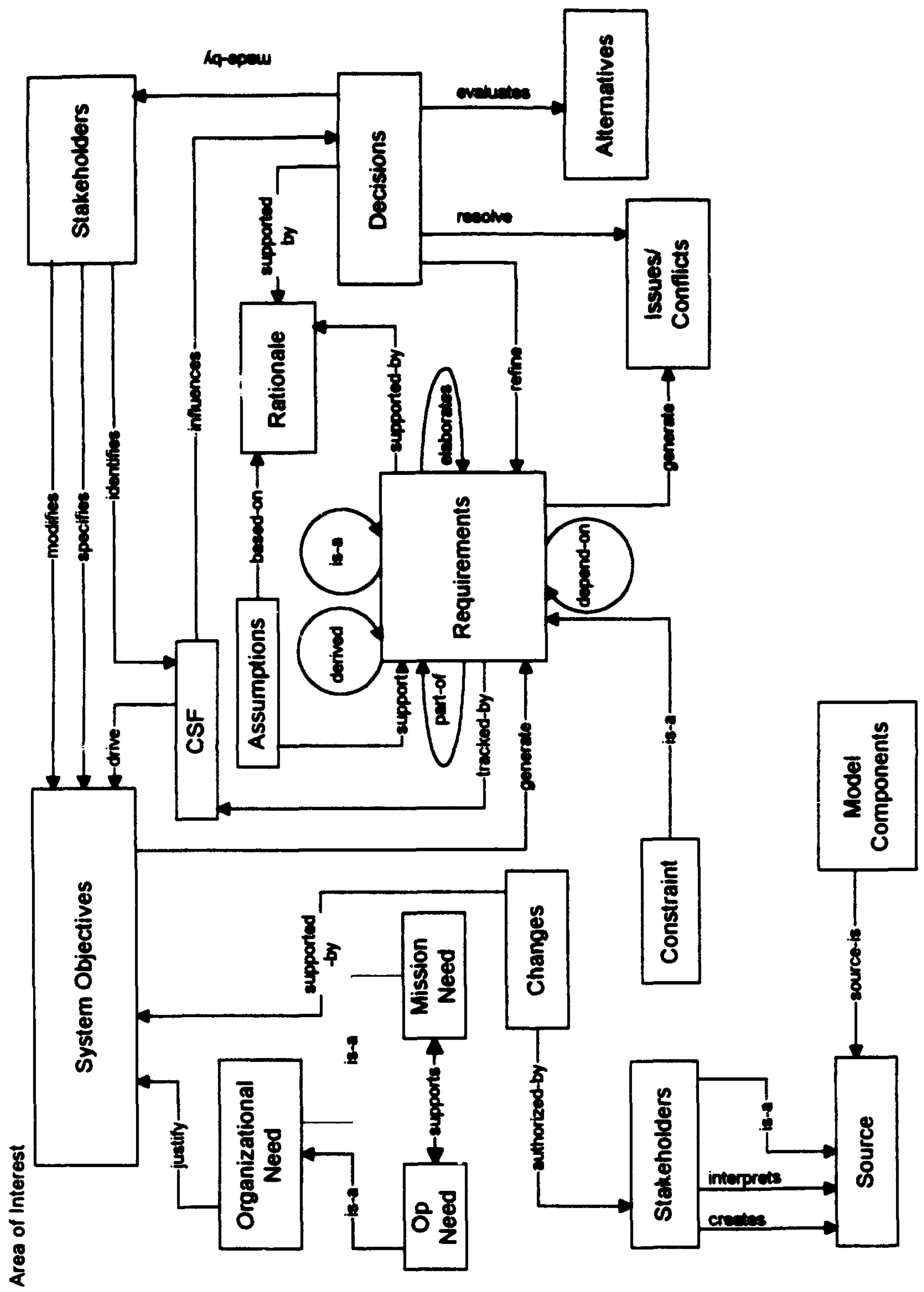

Figure A-1: Requirements Management Model

APPENDIX B.

DOD REFERENCES FOR INITIAL SYSTEMS DEVELOPMENT

TECHNICAL DISCIPLINE	REFERENCE
Acquisition Streamlining	DoDD 5000.43
Automated Information System Strategic Planning	DoDD 7740.2
Automated Information System Life-Cycle Management Process, Review and Milestone Approval Procedures	DoDI 8120.2
Automated Information System Life-Cycle Management Manual	DoD Manual 8120.2-M
A Guide for DoD-STD-2168	MIL-HDBK-268
Automated Interchange of Technical Information	MIL-STD-1840
Baselining of Automated Information Systems	DoDI 7920.4
Configuration Management and Audits	MIL-STD-973 MIL-HDBK-61
Climate Information	MIL-STD-210
Computer Aided Acquisition and Logistics Support	MIL-HDBK-59

Configuration Management of Automated Systems	DoDD 5010.19
Configuration Management Plans	MIL-STD-1456
Defense Acquisition	DoDD 5000.1
Defense Acquisition Management Policies and Procedures	DoDI 5000.2
Defense Acquisition Management Documentation and Reports	DoDM 5000.2-M
DoD Information Resources Management Program	DoDD 7740.1
Drawing Practices	MIL-STD-100
Defense Information Management Program	DoDD 8000.1
Defense System Software Development Handbook: A Tailoring Guide for DoD-STD-2167A	MIL-HDBK-287
Design to Cost	MIL-STD-337 MIL-HDBK-766
Engineering Drawing Practices	DoD-STD-100C
Systems Engineering	MIL-STD-499B
Operating Environments	MIL-STD-810
Human Factors	MIL-STD-1472C MIL-STD-1794 MIL-STD-1800 DoD-HDBK-763 MIL-H-46855
Integrated Diagnostics	MIL-STD-1814
Life-Cycle Management of Automated Information Systems	DoDD 8120.1

Manufacturing	MIL-STD-1528
Microcomputer Software and Hardware Guidelines	MIL-HDBK-805
Order of Preference for Selection of Standards and Specifications	MIL-STD-970
Parts, Materials, and Process Control	MIL-STD-965 MIL-HDBK-402 MIL-STD-1836
Preparation of Military Specifications and Associated Documents	MIL-STD-961C
Producibility	MIL-STD-1528 MIL-HDBK-727
Quality	MIL-Q-9858 MIL-I-45208
Quality Assurance Terms and Definitions	MIL-STD-109
Reliability / Durability	MIL-STD-785 MIL-STD-1530 MIL-STD-1543 MIL-STD-1783 MIL-STD-2164 MIL-STD-1798
Software	DoD-STD-2167A MIL-STD-1803 MIL-STD-1815 MIL-HDBK-287
Software Test and Evaluation	DoDM 5000.3-M3
Software Support Environment Acquisition	MIL-HDBK-782
Software Quality Assurance	DoD-STD-2168 MIL-HDBK-286
Specification Practices	MIL-STD-490 MIL-STD-961
Statement of Work Preparation	MIL-HDBK-245
Survivability	MIL-STD-1799 MIL-HDBK-336 MIL-STD-2069

System Safety	MIL-STD-882
System Security	MIL-STD-1785
Technical Reviews	MIL-STD-1521
Telecommunications	MIL-STD-188
Test and Evaluation	DoDD 5000.3
Testability	MIL-STD-2165
Training	MIL-STD-1379
Transportability	MIL-STD-1367 MIL-HDBK-157
Value Engineering	MIL-STD-1771
Work Breakdown Structure	MIL-STD-881B

APPENDIX C.

TOOLS

Dr. Stephen Andriole of Drexel University, derived the following listing of some (not all) of the COTS tools that can be utilized during the initial stages of systems development. Requirements traceability may not be a by-product of some of the tools. Yet, if the development team is able to validate the users needs with these prototyping tools, then the tool and its associated process can be stored in the overall traceability scheme.

Tools are listed by their associated platform (i.e., Macintosh, DOS/Windows, UNIX).

A. APPLE MACINTOSH PROTOTYPING TOOLS (NON-CASE)

* *Cricket Presents* by Computer Associates International (San Diego, Ca.):

One of the oldest screen creation and playback programs that can be used with the entire "cricket" family of software products. Strong on the linear playback of screens; under $200.

* *Microsoft Powerpoint* by Microsoft Corporation (Redmond, Wa.):

A powerful screen creation and playback program that supports 32-bit color for imported pictures and drawings. Easy to use program that permits nonprogrammers to experiment with alternative screen displays and user-computer interface designs; $300.

* *MacDraw* by Apple Computer Inc. and *Mac Draw II* by Claris Corporation (Santa Clara, Ca.):

MacDraw II supports color. Is compatible across Macintosh architectures. Screens created in MacDraw and MacDraw II can be imported in many playback programs. MacDraw $200, MacDraw II $300.

* *MacPaint* by Apple Computer Inc. and MacPaint 2.0 by Claris Corporation (Santa Clara, Ca.):

The first freehand drawing/painting program bundled with early Macintoshes.

* *Prototyper* by SmethersBarnes (Portland, Or.):

Strong on the design and development of Macintosh-like user-computer interfaces. Prototyper also generates high level (C, Pascal) code and Macintosh data resource structures. It is an easy to use program that will permit nonprogrammers to design and develop user-computer interface prototypes. $300.

* *The Slide Show Magician* by Magnum Software (Chatsworth, Ca.):

The Slide Show Magician permits playback with a surprising amount of flexibility. Screens can be played back at different intervals, wiped via several techniques, and has simulated functions via invisible "go-to" buttons placed under simulated menu options. $75.

* *MacBriefer* by FGM Inc. (Herndon, Va.):

This is a tool for creating screen displays and presenting them sequencially to designers and/or users. $500.

* *FilmMaker* by Live Software (Charenton-Le-Pont, France):

Powerful tool for creating and presenting animated displays. $500.

* *Videoworks II* (& Accessories) by Macromind Inc. (San Francisco, Ca.):

One of the first systems to support useful animation in color. $300.

* *Macromind Director* y Macromind Inc. (San Francisco, Ca.):

Capa le of delivering high fidelity animation and simulation especially as such capa ilities involve multimedia. $500.

* *Hypercard* (& Accessories) by Apple Computer Inc. (Cupertino, Ca.):

A multipurpose applications program and programming environment that supports the design and development of user-computer interfaces and simulations of fully functional prototypes. $50.

* *Supercard* by Silicon Beach Software Inc. (San Diego, Ca.):

Supports easy to use color applications, utilizes the full screen of the larger Macintosh monitors, and it has some novel animation capabilities. Supercard can be used to build prototypes or actual systems. $200.

* *PowerVision* by knowledge Vision (Myrtle Beach, South Carolina):

Supports screen creation, playback, animation, and the integration of multimedia applications. $1000.

* *Aldus Persuation* by Aldus Corp. (Seattle, Wa.):

A prototyping presentation tool that supports screen creation and sequential playback. $500.

* *Storyboarder* by American Intelliware Corp. (Torrence, Ca.):

Has an imbedded drawing/screen display capability as well as a playback capability. $500.

B. APPLE MACINTOSH PROTOTYPING TOOLS (CASE FOR SOFTWARE SPECIFICATIONS)

* *AppMaker, The Application Generator* by Bowers Development Corporation (Lincoln Center, Mass.):

Supports the design and development of user-computer interface design and development. $300.

* *Silverrun-SRL* by XA Systems (Los Gatos, Ca.):

Supports screen layouts. $2000.

C. UNIX PROTOTYPING TOOLS

* *Vermont Views* by Vermont Creative Software (Richford, Vermont):

Comprehensive user interface development environment that permits the design of screens, data entry forms, windows and menus. Runs under and poris to UNIX, DOS, OS/2, VMS, and Xenix. $5000.

* *Human Interface Manager* by Allsoft (Albuquerque, New Mexico):

A user interface development system. DOS: $350, Xenix: $550, UNIX and VMS: $900.

* *C-scape* by Oakland Group (Cambridge, Massachusetts):

An object-oriented interface system that supports all sorts of menuing, windowing, data entry, text entry, and help functions. UNIX: $1500, DOS & OS/2: less.

* *XBUILD* by Siemens/Nixdorf (Cambridge, Massachusetts):

An X-based user interface design and development tool designed to permit the development and testing of OSF/Motif user interfaces. $2000.

* *ExoCode & AutoCode* by Expert Object Corp. (Lincolnwood, Illinois):

Programs permit prototyping of user computer interfaces in several environments. Generates C language source code and supports the design and testing of GUI's. each: $1500.

* *The Builder Xcessary* by Integrated Computer Solutions Inc. (Cambridge, Massachusetts):

Permits the design and playback of complex user-computer interfaces in the OSF/Motif environment. This is a user interface management system with code generation capabilities. $2500.

* *The Open Windows Developer's Guide* by Sun Microsystems Inc. (Mountain View, Ca.):

A graphical user interface design editor which permits designers to design, develop, and test user-computer interfaces and then generate C code that can subsequently be compiled and linked with the larger applications program elements.

* *Transportable Applications Environment (TAE) Plus* by NASA/Goddard Space Flight Center (Greenbelt, Maryland):

Supports the design, development, testing of user-computer interfaces, and larger prototyping efforts on nearly all UNIX machines. Estimate: $300- $500.

* *Serpent* by the Software Engineering Institute (SEI) (Pittsburgh, Pennsylvania):

A User Interface Management System (UIMS) supports Sun 3/60 or higher, X11, and other environment. It uses the X Window system to interact with users and can be used for producing and prototyping systems. Available for testing & evaluation.

* *The Dialogue System* by Microfocus Inc. (Wayne, Pennsylvania):

Supports the design and interactive demonstration of screen displays and generates COBOL code. DOS, UNIX, OS/2: $600.

* *Vitamin C 4.0* by Creative Programming (Carrollton, Texas):

A tool for creating user interfaces based upon an extensive UCI library. Object oriented design permits custom designs of complex UCI system concepts. $1000.

D. SPECIAL PURPOSE PROTOTYPING TOOLS

* *InterMAPhics* by prior Data Sciences (Kantana, Ontario, Canada):

Designed for developing interactive command and control display systems which present dynamically changing information on a geographic background. The product thus supports realtime systems design and development. UNIX: $40000.

* *LabVIEW 2* by National Instruments (Austin, Texas):

Supports the design and development of UCI's as they pertain to interaction with data and knowledge in an instrument setting. Program permits the design and development of UCI's for cockpits, control panels, and software systems. Macintosh: $2000.

* *XPort* by perfect Products (Papillon, Nebraska):

Permits Macintosh applications to run on Sun workstations running X Window. Permits users on X11 workstations to log onto Macintoshes over TCP/IP networks as though the Macintoshes were UNIX servers. $495.

* *HOOPS* by Ithaca Software (Alameda, Ca.):

Permits porting of graphics applications across development platforms. Similar to XPort. $2100.

* *ESYview* by E-Systems (Dallas, Texas):

An X Window toolkit that supports map and graphics interaction. Suitable where maps and geographic data/knowledge interaction is required in an application. Particularilty suited to command and control system design problems. $3000.

E. IBM PC & COMPATIBLE PROTOTYPING TOOLS (NON-CASE)

* *Dan Bricklin's Demo II Program* by the Software Garden Inc. (Syracuse, N.Y.):

Can be used to create and playback screen displays. Has shown success in use on developing demonstration versions of major application programs. $200.

* *Dialogue System* by MicroFocus Inc. (Wayne, Pennsylvania):

A tool for creating screen displays and complex menu structures. $600.

* *ShowPartner & ShowPartner FX* by Brightbill-Roberts & Co. (Syracuse, N.Y.):

Showpartner supports limited screen design and somewhat more powerful slide editor. ShowPartner FX is more powerful presentation tool that can be adapted for prototyping.

* *Dr. Halo III* by Media Cybernetics (Silver Spring, Maryland):

Primarily a screen creation program that is flexible. Once screens are developed, they can be played back in a presentation format.

* *Skylights & Skylights GX* by Skylight Software Inc. (Wilmington, Massachusetts):

Strong in the design and development of PC-based prototypes, especially where high system interactivity is required. Can be used with Dr. Halo III via its ability to call screens into the application. Full version: $750.

* *Layout* by Matrix Software Technology Corporation (Cambridge, Massachusetts):

Comprised of four modules: Desktop for file manipulation; Paint for graphics and screen creation; Helpmaker for on-line help; and Layout for prototyping via cardfiles and flowcharts. $200.

* *Instant Replay III & Instant Replay Professional* by Nostradamus Inc. (Salt Lake City, Utah):

Can be used to design and develop prototypes, demonstrations, and tutorials. Instant Replay III: $200; Instant Replay Professional: $600.

F. IBM PC & COMPATIBLE PROTOTYPING TOOLS (CASE)

* *Excelerator* by Index Technology Corporation (Cambridge, Massachusetts):

Permits the design and development of data flow diagrams, entity-relationship diagrams, and other software specification models. Also permits the design and development of screen displays.

* *Picture Oriented Software Engineering (POSE)* by Computer Systems Advisors (Woodcliff Lake, New Jersey):

Supports conventional software engineering via a graphics environment.

* *CARDtools* by Ready Systems Corporation (Palo Alto, Ca.):

Supports simulation of system capabilities and prototyping.

* *Microstep* by Syscorp International (Austin, Texas):

Supports simulation of systems capabilities and prototyping.

All of the above listed tools can be utilized to communicate the needs, processes, and interactions required of the various stakeholders involved in the initial development phases of a system. These tools are characteristically easy to use and are relatively cost effective.

APPENDIX D.

STRUCTURED APPROACHES TO

CONCEPT ENGINEERING

AND

REQUIREMENTS DEVELOPMENT

A. CONCEPT ENGINEERING

Gary W. Burchill, in his Doctorial Thesis: *"Concept Engineering"*, 1993, defines concept engineering as a two level process. At the first level a process for developing product or service concepts that strive to meet or exceed customer articulated needs. The second level dictates that concept engineering is a decision support process. This is slightly differing from a decision support system, in that, a decision support process relies on problem solving systems using computers and on the human interaction that exists outside of the computers.

Concept Engineering, according to Burchill, is a five stage process. Each stage has embedded steps that guide the concept engineer toward a concept selection.

Stage 1: Understanding Customer's Environment

The objective is for the development team to develop empathy for the customer, in the actual use environment of the product or service. The process begins with an articulation of the project scope.

Step 1: Plan for Exploration

Exploration of what is needed in a product or service is accomplished by researching the activity and customer types. Prior to visiting the selected customers, the team members develop an open ended interview guide.

Step 2: Collect the Voice of the Customer

Pairs of team members (usually cross-functional) visit customers and conduct the interviews at the customer's site and take verbatim notes of customer comments and their own observations.

Step 3: Develop Common Image of Environment

Upon completion of the interview process, images of the customer's use environment are selected and analyzed. This image is a link to the customer's real world and acts as a contextual anchor for all future product concept decisions.

Stage 2: Converting Understanding into Requirements

The objective of this stage is to gather what was learned from the customer exploration into a small set of well understood, critical customer requirements.

Step 4: Transform Voices into Requirements

The transformation process converts the customer's language, often laden with emotion, into a customer requirement statement better suited for use in downstream development activities. Each customer's voice is explicitly linked to an image of the customers environment.

Step 5: Select Significant Requirements

The entire team then selects the vital few customer requirements, from the useful many, through a democratic and iterative process. Identifying the most important requirements based on their respective understandings of the opportunity (need).

Step 6: Develop Insight into Requirements

The image of the customers environment is again employed to develop new insight and team consensus regarding the relationships among requirements.

Stage 3: Operationalizing What You Have Learned

The objective is to ensure that the key customer requirements are clearly, concisely, and unambiguously communicated in measurable terms. The key customer requirements are validated with customers, operationally defined in measurable terms and the resulting information is displayed in such a way that the relationships between requirements, metrics, and customer feedback is easily seen.

Step 7: Develop and Administer Questionnaires

Questionnaires developed for this step should address the relative importance of requirements to the customer.

Step 8: Generate Metrics for Requirements

The team develops and structures metrics in order to measure, quantitatively, requirement realization. The use of tree diagrams, showing hierarchical relationships, is an approved method.

Step 9: Integrate Understanding

This stage concludes with the development of a Quality Chart and Operational Definitions (Deming 1986, Hauser & Clausing 1988, Juran 1988, Akao 1990) to integrate customer requirement understanding.

Stage 4: Concept Generation

This stage marks the transition in the development team's thinking from the "requirement or problem space" to the "solution space." The objective is to develop the largest number of potential solution ideas possible. Multiple perspectives of the development project are used to generate ideas from distinct vantage points.

Step 10: Decomposition

The complex design problem is decomposed into smaller, independent sub-problems based on the customer's and the engineering development perspectives.

Step 11: Idea Generation

The team creates, through individual and group collaboration efforts, an exhaustive list of ideas (both feasible and un-feasible) for each sub-problem; working first from the customer's vantage point before exploring the internal engineering perspective.

Step 12: Solution Generation

This stage concludes when each team member creates their ideal solution concept from the generated list of ideas.

Stage 5: Concept Selection

The objective of this stage is to select the "best" product or service concept for downstream development. Concepts are systematically reviewed, compared, combined and enhanced in an iterative process of concept development. Concepts are evaluated against customer requirements and organizational/environmental constraints.

Step 13: Solution Screening

The development team thinks individually and together, seeks expert assistance, and experiments in the laboratory in an iterative process of combining and improving initial solution concepts to develop a small number of superior concepts.

Step 14: Concept Selection

The "surviving" complete concepts are evaluated in detail against customer requirements and organizational constraints in order to select the dominate concept(s).

Step 15: Reflection (Traceability)

When completed, an audit trail exists for tracing the entire decision process from project scope determination through detailed concept analysis as this Concept Engineering process is self-documenting.

B. REQUIREMENTS DEVELOPMENT

Bell Communications Research (Bellcore), Special Report (SR-NWT-002159), Issue 1, 1992, provides a structured approach to requirements development. Key aspects of this report include: a. A robust process that encourages dialogue and participation among all stakeholders; b. Requirements documents that exhibit clarity of expression and ease of use; c. Consistency and uniformity among related requirements documents; d. Use of practical technology to enhance requirements development, review, use, and traceability.

The scope of this report extends into the more general realm of systems engineering including both hardware and software. [R-( )] indicates a requirement of the organizations requirements development process.

a. Focus

[R-1]: Organizations shall have in place a requirements process and contents guideline as part of their corporate policies and practices.

There is no overall "best" contents for requirements. A requirements template should be developed and address width (coverage) and uniformity of overall presentation of the requirements. Coupled with format and style issues the template should result in a better reading, more useful (set of) requirements document(s).

b. The Requirements Template

[R-2]: Adopt or customize a requirements template for each product.

A requirements template provides a frame of reference, identifies needed information, and indicates an order of presentation. However no single template can meet the needs of every project. The template should be tailored, and considered as a collection of building blocks serving as a checklist to reduce the chance of inadvertent omission.

[R-3]: Each organization shall establish a standard template or set of templates for given application domains.

When talking about similar systems or system components, one should use the same or similar templates. The degree of tailoring or customization is both a management and technical issue. Customizing for a class of systems such as: security, performance, user interface, are often quite useful.

c. The Requirements Data Base

[R-4]: All requirements information shall be stored within the requirements data base.

After determining what requirement elements are to be included within the requirements data base, this data becomes the repository for ail requirements information (this determination of elements should be gleaned from the concept engineering phase that has previously transpired).

[R-5]: The requirements data base shall be kept current at all times.

Different views of the data base containing subsets of the elements and/or different levels of detail, shall be provided for different audiences (stakeholders).

[R-6]: The requirements changes shall be expressed as a change to the requirements data base.

Change management is another key issue impacting the requirements data base. All changes must be reflected in the data base.

[R-7]: Establish a list of data base elements needed to describe each single requirement.

The individual requirement statement itself is only the beginning. There is other information that is needed to support and communicate the requirement. An example follows:

[EXAMPLE]

Elements for a single requirement

Tech. Content	Admin. Content
Req. Statement	Unique ID
Graphics	Change/Configuration Data
Attributes	Version/Phase Data
Verification	Audience Views/Extracts
Structure	Working Notes
Metrics/Sizing	Decision History
Comments/Remarks	Status

[R-8]: Explicitly identify which data elements and what level of abstraction are necessary and appropriate for each participant in the audience and/or stakeholders (this information is also gleaned from concept engineering phase.)

[R-9]: Maintain a decision history, based on working notes, highlighting major decisions and actions.

The decision history may simply be a sequential accumulation of working notes or it may be a more formal structured data base including a listing of alternatives considered, evaluation and selection criteria. A decision history is quite useful for maintaining requirements, especially when the requirements have long "shelf life", when there is staff turnover, or when it becomes necessary to reexamine a requirement or reconsider alternatives.

d. Requirements Tools

[R-10]: Maintain current documentation of all tools being used to support the requirements efforts.

This documentation is to include a tools inventory, applicability guidelines, usage guidelines, and all information needed to use the tools.

e. Format and Style

[R-11]: Use a set of explicite labels to distinguish among requirements, commentary, examples and other categories of information.

"[R]" indictates a requirement. "[RP]" indicates a recommended practice. "[EXAMPLE]" and other text categories can be explicitly labeled. Attributes such as importance, "[CRITICAL]", and dependencies may be denoted. When available, highlighting may be appropriate.

[R-12]: Decompose compound requirements into seperate, singular requirements.

[R-13]: State only one single requirement at a time.

It is vital that each requirement be visible and stand alone. The decomposition of requirements into singular individual requirements may result in a choppier, less flowing narrative. This tradeoff of accuracy vs. flowing style should favor accuracy.

[R-14]: Employ an organized requirements numbering scheme with unique identifiers.

There are a number of alternatives for establishing and managing unique identifiers. Try to employ a scheme that closely resembles the outputs from the concept engineering phase.

[R-15]: Maintain a requirements trace (traceability) throughout the system life cycle.

[R-16]: Explicitly identify links among requirements.

R-16 can facilitate R-15 if the linkages are traced. Linkages are especially important during the requirements change process. Contemplated changes to requirements need to be considered in the light of their impact on other requirements. Linkage facilitates the change impact analysis process.

[R-17]: Determine which attributes shall be included within the requirements document.

[R-18]: Clearly label all attributes.

Attributes of the requirement include: uncertainty, volatility, importance, source, etc. Capturing and communicating attributes is an aid towards quality requirements.

[R-19]: Requirements must be verifiable.

[R-20]: Subjective requirements shall have associated criteria for making the associated binary subjective judgement of (requirement) met or not met.

[R-21]: A test method and decision criteria including statistical distributions, sample size and pass/fail criteria shall be provided for statistical requirements.

As with all requirements, statistical requirements need to be presented in such a fashion that all parties (stakeholders) can determine and agree to whether or not the requirement is being met. Statistical requirements may have significant legal, contractual, and technical implications.

f. Requirements Categories

There may be a fine line between requirements and design. A requirement is something believed to be necessary to meet the needs of the users. An explicit requirement, [R], focuses on "what" not "how".

A conditional requirement, [R_{cond}], is one that is invoked only if a specific condition (state) or event occurs. The term "conditional" is preferred to "optional" in that "optional" opens the question, "whose option?"

A phased requirement, [R_{phased}], may be invoked if the product(s), and their underlying requirements may be available in multiple versions or with planned changes over time. This leads to the following conditional requirements if a phased requirement is stated.

[R-22_{cond}]: Clearly identify multiple effectivities (phases) when they are present.

[R-23$_{cond}$]: Clearly identify multiple deliveries and the delivery schedule when there are multiple deliveries.

It may be desirable to include a category of requirements called constraining requirements, [R$_{constraint}$], or constraints. These add emphasis or weight to mandated requirements. An example of this would be: *[R$_{constraint}$-963] Use an ACME Model 256-B Tape Drive.*

[R-26]: There shall be no unrecorded requirements.

This statement speaks for itself.

[R-27]: Explicitly state assumptions.

By stating all assumptions explicitly, "catch-all" requirements can be made such as: *[R-200] The system shall not inpact any other system.*

g. Requirements Changes

Requirements change as a result of additional information and analysis during the requirement's development process, customer's needs, external factors, and as a result of errors discovered during the system's life cycle.

Establishing a requirements baseline is critical to project success. Additionally, managing requirements changes is also vital to project success.

[R-28]: All changes to the system shall be initiated as a change to the requirements document.

[R-29]: All changes to the requirements document shall be explicitly identified.

[R-30]: All changes to the system shall be integrated into the requirements data base.

[R-31]: Any proposed change to the system shall be expressed as a proposed change to the requirements data base and requirements document.

[R-32]: Administrative and technical data supporting the requirements change shall be stored in a data base.

h. Teamwork

One cannot overemphasize the human side of systems development, and of requirements. Cooperation and teamwork, fostered by leadership, common goals and good communications, are vital to successful development.

In a step-by-step development process, there is a tendancy to compartmentalize tasks for each participant. Many "hand-offs" occur as the development process continues towards completion. Early participation by persons responsible for subsequent tasks and continued availability of appropriate experts is a positive step towards teamwork and higher quality systems.

i. Managing Expectations

Pundits usually describe success as having three characteristics; good, fast, and cheap. A systems development effort can produce these characteristics when they are clearly defined, expectations are properly managed, objectives are explicitly stated, a reasonable work plan is developed, and participants work the plan.

[R-34]: Criteria for success shall be explicitly stated and agreed upon by all project participants including the customer, management, and development staff.

Criteria for success must incorporate user expectations regarding general performance, schedule, and cost.

[R-35]: Monitor the project.

Encourage feedback and take any necessary corrective action.

j. Process Customization and Tailoring

A quality requirements process should be tailored to meet the needs of the system and organizations that it impacts. Tailoring begins by developing a taxonomy of the system and the systems development, and is completed by the customizing of the requirements template and the selection of requirements database elements.

[R-36]: Develop a taxonomy describing the system, the system development process, environment and life cycle.

Developing a taxonomy will help focus on process issues to enable an advantageous selection of methods and tools. A taxonomy also helps clarify differences and similarities among projects.

k. The Audience

[R-37]: Identify the audience(s) for the requirement and their special needs.

Requirements are a communications vehicle that must look beyond the product to the audience(s) that will use them for many purposes.

l. Sequence of Events

[R-38]: Establish a detailed sequence of events for accomplishing the project.

The following is a generic sequencing process, like the requirements template, it must be tailored to the specific system/project:

[EXAMPLE]

1. Strategic/Business Issues Assessment

   Concept-Operational capability objectives
   Business plan
   Budget
   Assess technology futures
   Identify need for requirements document
   Identify the users

2. Assignment

   What Organization/Team to do analysis
   What Organization/Team to write document
   What Organization/Team to manage activities
   Assign individuals

3. Planning

   Determine type of document
   Relationship to other documents
   Resource allocation (time/money/staff)

4. Project Planning

   Establish scope and identify major interfaces
   Tailor requirements and database templates to:
     fit with coordinating documents
     meet perceived needs of project
   Review and finalize templates
   Refine resource allocation
   Determine probable information sources

5. Requirements Gathering and Development

- Data gathering (from/with users)
- Analysis
- Defining system components
  - defining functionality
  - define feature interactions
  - defining constraints
  - defining the user interfaces
  - defining system interfaces
- Writing/Authoring
- Quality reviews
- Administrative reviews
- Technical reviews

Outlining a sequence of events is a good management practice and greatly improves the projects ability to communicate with all participants.

m. The change Process

[R-39]: There shall be a change approval process that considers both technical and business issues.

[R-40]: An up-to-date change history data base shall be maintained.

[R-41]: The requirements data base shall be under configuration management.

An established process must assure that all proposed changes are analyzed and reviewed. Each proposed change is analyzed by a responsible person using: 1)decomposition, 2)links, 3)key word text searches, and 4)communication with knowledgeable participants.

LIST OF REFERENCES

Andriole, S.J., "Interactive Requirements Modeling & Prototyping for Enhanced Software Requirements Specification", IEEE International Conference on Requirements Engineering, Colorado Springs, Colorado, April 1994.

Bell Communications Research (Bellcore), "A Requirements Tutorial: Quality Systems and Software Requirements", Special Report SR-NWT-002159, December 1992.

Blanchard, B.S. & Fabrycky, W.J., "Systems Engineering and Analysis", Prentice Hall Publishers, Englewood Cliffs, New Jersey, 1990.

Burchill, G.W., "Concept Engineering: An Investigation of Time vs. Market Orientation in Product Concept Development", Massachusetts Institute of Technology, June 1993.

Chairman of the Joint Chiefs of Staff Memorandum of Policy, *Requirements Generation System Policies and Procedures*, CJCS MOP 77, 17 September 1992.

Department of Defense Directive, *Defense Acquisition,* DoDD 5000.1, 23 February 1991.

Department of Defense Manual, *Defense Acquisition Management Documentation and Reports*, DoD 5000.2-M, February 1991.

Fox, J. R., "The Defense Management Challege, Weapons Acquisition," Harvard Business School Press, Boston, Massachusetts, 1988.

Gotel O.C.Z., Finkelstein, A.C.W., "An Analysis of the Requirements Traceability Problem," Imperial College of Science, Technology, and Medicine, Dept. of Computing, London, UK, 1993.1

Harrington, G.A., Rondeau, K.,M., *An Investigation of Requirements Traceability to support Systems Development,* Naval Postgraduate School, Monterey, California, September 1993.

Military Standard, *Defense System Software Development*, MIL-STD-2167A, 29 February 1988.

Military Standard, *Systems Engineering*, MIL-STD-499B (DRAFT), 6 May 1992.

Roetzheim, W. H., "Developing Software to Government Standards, " Prentice Hall, Englewood Cliffs, New Jersey, 1991.

BIBLIOGRAPHY

Abbott, A.G., Busch, M.R., Edwards, M., Ramesh,B., *An Initial Model of Requirements Traceability, an Empirical Study*, Naval Postgraduate School, Monterey, California, September 1992.

Air Force Pamphlet, *Guildelines for Sucessful Acquisition and Management of Software Intensive Systems: Weapons Systems and Mangement Information Systems,*Department of the Air Force, Washington, D. C., 1993.

Davis, A. M., "Software Requirements, Analysis and Specification," Prentice Hall, Englewood Cliffs, New Jersey, 1990.

Department of Defense Instruction 5000.2, "Defense Aquisition Management Policies and Procedures", Department of Defense (Aquisition), Washington D.C., 1991.

Gause, D.C., Weinberg, G.M., "Exploring Requirements: Quality before Design," Dorset House Publishing, 1989.

Krueger R. A., "Focus Groups, A Practical Guide for Applied Research," Sage Publications, Newbury Park, New York, 1988.

Powers, P. P., Stubbs, C.D., *A Study on Current Practices of Requirements Traceability in Systems Development,* Naval Postgraduate School, Monterey, California, September 1993.

Shamdansani, P.N., Stewart, D.W., "Focus Groups, Theory and Practice," Sage Publications, Newbury Park, New York, 1990.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Professor B. Ramesh Naval Postgraduate School Monterey, California 93943-5000	2
4.	Professor Carl Jones Naval Postgraduate School Monterey, California 93943-5000	2
5.	Professor Michael Sovereign Naval Postgraduate School Monterey, California 93943-5000	1
6.	Capt. Stu White 1912th CSGP/SCP 37 Elm St. Suite 205 Langley AFB, Virgina 23665	1
7.	LT Robert C. Laubengayer 8376 Alaterna Court Orangevale, California 95662	2
8.	LT Jeffrey S. Spearman 7361 Milner Drive Colorado Springs, Colorado 80918	2