# A RAPID RESPONSE SOFTWARE MODIFICATION PROCESS FOR SLBM TARGETING SOFTWARE

by

**Laurel Ann Brasé**

Project submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

**MASTER OF ENGINEERING**

in

**Systems Engineering**

**APPROVED:**

B. S. Blanchard, Chairman

W. J. Fabrycky

J. R. Pollard

April, 1995
Blacksburg, Virginia

**Key words:** SLBM, Software, Development, Strategic, Targeting

C.2

# A RAPID RESPONSE SOFTWARE MODIFICATION PROCESS FOR SLBM TARGETING SOFTWARE

by

Laurel Ann Brasé

## (ABSTRACT)

The need for a faster, streamlined process for the modification of SLBM targeting software without sacrificing product reliability was identified. Both error correction and changes in software requirements were considered in the design of this process.

Criteria for the design and use of the rapid response software development process were based on the magnitude, time sensitivity, and importance to strategic targeting of the modifications requested. Other concepts addressed included configuration management, customer service, teamwork, empowerment, and process improvement. The process contains four distinct paths based on the time sensitivity and size of the changes requested.

A discussion of the design of the four paths, including use criteria, total process time, work breakdown, and cost is presented. It is accompanied by a hypothetical example of a software change request and its journey through the process.

It was concluded that modifications to the process could be made to further reduce its duration and expand its applicability.

# ACKNOWLEDGEMENTS

I would like to express my gratitude to Dean Ben Blanchard, Dr. Wolter Fabrycky, and Dr. J. Ron Pollard for taking the time to serve on my committee and for the guidance they provided me for this project. I am also indebted to Loretta Tickle for helping me wade through the university regulations. I live alone, so I did not have a family or pets to neglect in the course of completing this project, but I did have to inconvenience some coworkers. I am extremely grateful to those who assisted me by providing documents or by answering my sometimes unorthodox questions. I am also grateful to my supervisor, Janice Cooper, for allowing me some latitude in working on this project. I also owe my thanks to those who lended moral support and to those who simply waited patiently for me to find time for them and to just "finish the darn thing".

# REMARKS

This report was writted for academic use only. It has not been reviewed by SLBM program management. The SLBM program at NSWCDD does not collect data for many of the figures of interest contained in the introduction. Therefore, it was necessary to estimate the values of those parameters. For that reason, citing this report as a reference for any other research effort would be inappropriate.

# TABLE OF CONTENTS

# LIST OF FIGURES

**MAIN TEXT:**

**APPENDIX A:**

**APPENDIX B:**

**APPENDIX C:**

**APPENDIX D:**

# 1.0 SYSTEM DEFINITION

## 1.1 INTRODUCTION

The Submarine-Launched Ballistic Missile (SLBM) Program (K40/K50) at the Naval Surface Warfare Center, Dahlgren Division (NSWCDD) is responsible for producing targeting software. This software is used by the Missile Strike Section (J5231) at United States Strategic Command (USSTRATCOM) in the production of the Single Integrated Operational Plan (SIOP). The SIOP is the US plan for fighting a nuclear war. SLBM forces are a key part of that plan, so it is extremely important that SLBM planning software be accurate and reliable. The rapidly changing world political situation is forcing SIOP planning to follow different constraints and to become increasingly more flexible. As SIOP production evolves, the software used to produce the SIOP must evolve with it. This is especially true for the SLBM planning software, since the SLBM forces are the most capable and most flexible of all of the US nuclear forces.

Creating and maintaining SLBM SIOP production software is an enormous task. It is necessary to do work on the entire suite of software for three SIOP revisions at any given time (current SIOP, next SIOP, and next+1 SIOP). There are 10 NSWCDD computer software configuration items (CSCIs) containing approximately 25

computer software components (CSCs) on three different hardware platforms for the SIOP 95. For SIOP 96, there are 17 CSCIs and approximately 30 CSCs. There are 17 CSCIs are also planned for SIOP 97, but they will contain approximately 35 CSCs.

Production and maintenance of all of this software require a large number of research and development personnel. There are approximately 250 people assigned to the SLBM program at NSWCDD. Of these, 35 are management or administrative personnel. Of the rest, 70 people are available to perform the entire targeting software development process for all of those CSCIs and CSCs. The approximate breakdown is: 35 analysts, 20 programmers, 5 QA specialists, 5 QC specialists, and five media prep specialists. Obviously, most people must work on more than one CSC. Many people must work on several items, often during the same period. The shortage of personnel, particularly in the programming area, has forced K40/K50 to use co-op students and contracted support for some tasks. Also, of the 35 analysts, some physicists, mathematicians, and engineers are group leaders or special staff members who have previously played more of a supervisory or consultant role in the process. The special staff and various others involved in the support of targeting software are also required to spend significant amounts of time supporting other efforts. Others out of that group are non-degreed technicians, who traditionally perform only limited support functions.

The shortage of personnel means that nearly all involved are extremely busy. In addition, the contractors, co-op students, and technicians do not have much training in the tasks required of analysts, programmers, and QA specialists. Both factors increase the risk of errors in the software and its accompanying documentation after delivery to USSTRATCOM. NSWCDD software is expected to have a reliability of 100% for each CSCI. Therefore, the failure rate for every CSCI is expected to be zero, as is the total failure rate. (Note: For this project, the failure rate for a given CSCI is the number of errors per CSCI or related deliverable document per SIOP revision. The failure rates are considered cumulatively over all deliveries of the CSCI throughout its lifecycle.) However, during the current and last SIOP revisions, three or four CSCIs had to be modified after delivery due to significant errors found by the developers or by the customers. Other errors that do not affect the execution results of the software involved are occasionally found, but do not warrant modification of the software before the next scheduled delivery. TRIDENT I software for main frame computer platforms has been delivered for each SIOP revision since the weapon's IOC in 1983. NSWCDD has been delivering main frame-resident TRIDENT II software since 1988. PC software has also been delivered since 1988. Workstation software has only been delivered since 1993. These are the only two revisions for which multiple failures have occurred. No problems with the software or the documentation have been reported during most SIOP revisions.

Therefore, the failure rate for most NSWCDD-produced SLBM software is still less than one failure per SIOP revision. Failures have never been reported for some CSCIs. So the total failure rate is still less that one failure per revision. A conservative estimate of the total reliability is approximately 95%. However, if the number of errors in delivered software continues to increase, this figure will decrease significantly.

Besides causing an increase in errors, the time and training deficiencies have also increased the difficulty in responding to requirement changes from USSTRATCOM. As mentioned previously, there has been an increase in the number of new or different requirements in the past two years. As events in the strategic nuclear community have begun to occur much more rapidly, requirement changes have increased. It is no longer acceptable to wait until the next planned revision to implement all these changes. This places an additional burden on the K40/K50 personnel involved, especially if multiple CSCIs and/or multiple additional deliveries are necessary. For SIOP 94, there were three requirement changes fulfilled before the next scheduled delivery of the CSCIs involved. Only one change was minor in scope. It involved a single line of code, but it affected four CSCIs. The other two were more substantial: one required a new algorithm in one CSCI, while the other required name changes to most of the subroutines in four CSCIs. For SIOP 95, only

one requirement change was time-critical, but it affected three CSCIs.

The current software development process was designed for development efforts and the implementation of changes only once per SIOP revision. This process takes approximately 12 months to complete for most delivery packages, known as computer software versions (CSVs). Therefore, it cannot handle urgent requests for requirement changes or error corrections. As indicated in the preceding paragraphs, both types of requests have been honored recently. Unfortunately, there is a certain reluctance to do that, of which the USSTRATCOM has become aware. Part of the reluctance stems from the time constraints faced at NSWCDD. The other part of the problem is that the current software development process fails to address time-critical changes. It should be recognized in fairness to the authors of this process that it is only a year old, has not been fully completed for any targeting software yet, and is still undergoing improvements. However, the previous system had the same flaws (OD 55688). People are more willing to do difficult tasks if there is explicit guidance provided.

Until now, software changes have been handled on a case-by-case basis. The amount of time needed to make the changes varies, as does the process by which the changes are made. Sometimes, complete redeliveries have been made. Occasionally, a description of the code change is faxed or transmitted electronically. Then the

customer's software integrators must make the change, recompile and relink the software, and run test cases. In one instance, the NSWCDD liaison to USSTRATCOM then, who happens to be a physicist with extremely limited knowledge of software and computers, was instructed to make the software changes herself. It took four pages of extremely detailed instructions and the assistance of a programmer via the telephone to successfully complete this task. This cavalier approach to software modification raises some serious configuration management and quality assurance issues. It also confuses the customer, the liaison, and the developers involved.

The purpose of this project is to design a process to implement software modifications needed to meet new strategic planning requirements or to correct serious errors. It is the intent of this researcher to do this so that it has the least possible impact on normal development efforts.

## 1.2 MISSION DEFINITION

The mission of the entire system (to include software, hardware, personnel, facilities, media, documentation, and regulations) is to produce, process, and maintain strategic targeting data. For the purpose of this project, only the SLBM-related portion of the system will be discussed. The focus of this project is an attempt to improve the software subsystem, in the form of a rapid response software

modification process. Other parts of the system will be discussed primarily in terms of how they relate to the software.

The primary mission of the rapid response software modification process is to enable software developers at NSWCDD to modify critical SLBM targeting software within an acceptable amount of time. This mission will be performed for the following situations: enhancement or adaptation of existing software, rehosting of existing software in a new hardware and/or operating system environment, and the correction of errors in existing software. NSWCDD and USSTRATCOM production tools are the only ones that meet the criteria for this process. The most common mission profile is enhancement or adaptation of existing USSTRATCOM production tools.

## 1.3 OPERATIONAL DEPLOYMENT

The rapid response software modification process will be employed at NSWCDD in the SLBM program. This process will be inherently manpower-intensive in an organization with a scarcity of manpower. It will occur during normal working hours (Monday-Friday, eight hours per person per day, no holidays). Some longer workdays and/or weekend work may be required due to the manpower shortage and the short response time allowed. Development computers currently include the Cray with X-terminals, PCs or VT 220 terminals, the DGSNET with X-terminals, Sun workstations, or

Macintosh computers, and 386 PCs. The work required for the rapid response software modification process will be completed in two office buildings at NSWCDD under SECRET or TOP SECRET security controls. Several products may be undergoing this process at any given time. Changes in SLBM SIOP production requirements will govern usage. The change request form (WORDPERFECT 5.1 template) will be maintained on the DGSNET once it has been retrieved via the SRS datalink.

## 1.4 SOFTWARE

The following software will be employed in the rapid response software modification process:

### 1.4.1 MACPROJECT PRO

MACPROJECT is recommended as the project management tool. People could view the PERT charts showing the process and subprocesses they are to use. Developers could enter actual start dates, finish dates, and work-hours used for each "supertask." They could substitute names for the generic resources. This would allow for easy tracking of progress made, and would help in the process analysis at the end by providing actual time and cost data. This recommendation is based on this scientist's experience, and on the fact that it is currently available at NSWCDD. Other tools (ie. MICROSOFT PROJECT) are available with varying features, platforms,

and costs that would also meet this need. Software PLCs should use MACPROJECT or another project management tool in the normal software development process, because it will make project management tasks easier and more efficient, therefore reducing the amount of time required to perform those functions.

### 1.4.2 WORDPERFECT

WORDPERFECT or WORDPERFECT for WINDOWS, PC version 5.1 or higher should be used for the template for the software change request form. WORDPERFECT versions 2.1 or 3.0 for the Macintosh are also acceptable. It appears that most files readily convert between the versions. WORDPERFECT is the current US Navy word processing standard and is available in some form on most PCs, Macs, the DGSNET, and the SRS datalink. Most people are already familiar with it. Therefore, since wordperfect is almost universally available within the SLBM program and people know how to use it, choosing WORDPERFECT for the software change request form and for all documentation would increase compatability and maintainability. The resulting improvement in efficiency of some tasks should ultimately reduce the amount of work-hours, the task duration, and the cost associated with those types of activities.

### 1.4.3 DRIVERS

Almost all NSWCDD software is delivered with a driver program. Sometimes, that driver is only used during installation of the

software. After the software has been installed, a set of delivery test cases is executed using the NSWCDD driver. The results are compared to the results obtained from the same software installed on a computer at NSWCDD. The comparison is made by an autocompare program that matches the output files line by line and creates a summary of the discrepancies. The purpose of this comparison is to identify any machine dependencies that might significantly affect program execution results.

The second type of driver program used is independently created by K52. The QA specialist uses it during the validation and verification phase of software development. It is also used in the validation and verification phase of the rapid response software modification process. In the normal software development process, the QA specialist might need to take time to create this driver. In this process, it is a vital assumption that the independent driver program already exists.

The third type consists of drivers created by USSTRATCOM or by other integrators working for USSTRATCOM. These drivers are used during the certification of NSWCDD software at USSTRATCOM. The purpose of this certification is to ensure that the software has been integrated properly and the execution results provided to the user by the driver are accurate. At some point after a software modification has been delivered and integrated, the software must be recertified.

## 1.5 HARDWARE

NSWCDD delivers to three computing environments at USSTRATCOM. A fourth computer is used for the transmission of data and programs. There are five computing environments used at NSWCDD for the development of software. A variety of peripheral devices are also used at both sites.

### 1.5.1 MAIN FRAMES

Until recently, TRICOMS was the primary target computing environment. TRICOMS is an IBM main frame computer. Processing in this environment is driven by Job Control Language (JCL). Generally, software and data deliveries to TRICOMS are on 9-track magnetic tape. Occasionally, 3.5 inch disks are used, which requires that the information be uploaded to TRICOMS using a PC. The usage of TRICOMS for missile strike planning is not expected to continue past the end of 1997.

NSWCDD uses a CRAY computer with a UNIX operating system to develop software for TRICOMS. The developer may be using a VT220 or VT340 monitor, a networked PC, an X-terminal, or a networked workstation to access the CRAY. The network in use is the DGSNET, which will be discussed later. Disk drives are only available to users of PCs and workstations. Most magnetic media used on the CRAY are therefore 9-track tapes.

### 1.5.2 PCS

NSWCDD provides three targeting models to USSTRATCOM for use on a PC. A variety of PCs and laptop computers are used, both in standalone and in network configurations. Each PC is expected to be at least a 386 with 640K of RAM. NSWCDD requires MS-DOS 3.3 or higher. Ideally, the PC will have a 3.5 inch disk drive capable of reading double-sided, high density (1.4Mb) disks. If this type of drive is not available, special arrangements can be made to provide 5.25 inch double-sided, high density disks (1.2Mb). For this type of software, NSWCDD also uses PCs for its development.

### 1.5.3 WORKSTATIONS

The missile strike planning section is migrating to a network of SUN SPARCstation 20s operating as "slaves" to a database server and a software server. The servers are SUN SPARCstation 10s. These workstations have a SOLARIS UNIX-based operating system. The database uses a SYBASE SQL server. Compatible compilers for FORTRAN, C, C++, and ADA are available. (Note: FORTRAN and C are the primary languages used in K40/K50.) Appropriate media for the workstations are 3.5 inch 1.4 MB disks and 8mm tapes (USSTRATCOM, 12-3).

At NSWCDD, the DGSNET is used for development work targeted at the SUN workstations. The DGSNET is a collection of UNIX servers with X-terminals, workstations (SUN SPARCstation 2 or 20 and SGI),

PCs, and Macintosh computers networked together. Most development work is done with either an X-terminal or a workstation. DGSNET servers are currently migrating from SUN OS operating systems to SOLARIS. FORTRAN and C compilers are used. A SYBASE SQL database is also used. An attempt is being made to match the significant characteristics of the missile strike network configuration as much as feasible (OD 61711, 3-1)

### 1.5.4 SRS DATALINK

The SLBM Retargeting System (SRS) datalink is a computer network that uses dedicated communication lines between NSWCDD, USSTRATCOM, and other sites. It is controlled by NSWCDD. The datalink uses 486 PCs with MS-DOS. The datalink currently has 3.5 inch 1.4 MB disk drives, 5.25 inch 1.2 MB disk drives, and 9-track tape drives. Replacement of the 9-track tape drives with 8 mm tape drives is being considered because workstations will be the primary computers at USSTRATCOM.

The primary mission of the SRS datalink is to transmit TOP SECRET targeting data rapidly. However, most software is only SECRET. NSWCDD is being added to the Integrated Production Facility (IPF), a USSTRATCOM-managed computer network specifically designed to simplify the development of software for USSTRATCOM by government agencies and defense contractors. This network operates at a SECRET level. However, it is primarily for use with

TRICOMS. The workstation link has not been completed yet, and NSWCDD use is not anticipated to commence until that capability becomes operational. Until that time, the rapid response software modification process will depend upon the SRS datalink. Even after the IPF link is fully operational, the SRS datalink will still be used in this process, though. Generally, "problem cases" sent by USSTRATCOM to NSWCDD for investigation of potential software errors are classified TOP SECRET and therefore may not be sent over the IPF.

### 1.5.5 OTHER HARDWARE

In addition to the software development machines, the rapid response software development process will use other hardware for some documentation changes. DGSNET can be used for documentation and E-mail, but VAX VMS computers are preferred by some developers for documentation and E-mail communications. Others use Macintosh computers for documentation. This process is intended to accommodate all document production and E-mail preferences, if the participants involved in any given implementation of the rapid response software modification process agree on a single platform to use.

## 1.6 SECURITY

The rapid response software development process must accommodate SECRET (and below) and TOP SECRET material. Most of these efforts will be SECRET, with the exception that problem cases sent from USSTRATCOM to be investigated will most likely be TOP SECRET. Security measures are in accordance with OD 61561, NAVSWCINST 5510.1B, and OPNAVINST 5510.1H.

### 1.6.1 COMPUTERS

The USSTRATCOM computer systems involved, except the IPF, are operated at the system-high TOP SECRET level. At NSWCDD, most development is done on system-high SECRET computer networks. There is a TOP SECRET CRAY computer currently in use. A TOP SECRET DGSNET is beginning its implementation phase. A few TOP SECRET PCs and laptop computers also are available. The SRS datalink equipment is also operated at the TOP SECRET level.

### 1.6.2 MEDIA

Magnetic media and paper documents associated with this process are handled according to security regulations based on their classification. Transmittal of SECRET and TOP SECRET material is addressed in the delivery phase description.

### 1.6.3 PERSONNEL

The personnel involved must have at least a SECRET clearance. Nearly all K42 personnel and others routinely involved in work supporting strategic targeting hold TOP SECRET clearances. Participation by foreign nationals is extremely limited, and is often expressly prohibited.

## 1.7 OPERATIONAL LIFE CYCLE

The rapid response software modification process is intended to be open to improvement, replacement, or discard based on lessons learned during its usage and changes in software production requirements. There is also the possibility that this process design may be rejected and therefore never used. Therefore, no finite life span can be projected.

## 1.8 USE CRITERIA

Any process that requires the redirection of the limited human resources will have an impact on normal production. Consequently, every attempt should be made to reduce that impact by minimizing the number of personnel involved and the amount of time used by the rapid response software modification process. It also will be necessary to limit the number of times the process will be used. This can be accomplished by a series of use criteria.

The first test will be the relative importance of the software in question. The software to be changed must be a production tool. A production tool is a program that is used for producing or processing strategic targeting data. Nothing less would justify such a diversion of resources. Secondly, the only acceptable conditions for making a change to the software using this process are: (1) either the customer or a member of the development team discovers an error that affects the program results used for the production and/or processing of strategic targeting data, or (2) the customer levies a new requirement that affects production or processing.

Next, the customer must decide the priority of the change. The priority of the change will be classified as routine, urgent, or critical, based on (1) when the updated or corrected software must be available for use, and (2) the customer's assessment of the importance of the change to performing the mission. Routine changes are those that do not require implementation within six months. Examples of routine changes are enhancements of the "nice to have, but not required" variety, and errors that do not affect program results that are actually used for production. Urgent software changes are those required within three to six months. Critical software changes are those needed within three months. Urgent and critical software changes are errors that affect essential output or requirement changes that affect strategic targeting. Only urgent and critical changes will qualify for the rapid response process.

The magnitude of the software change is also important. Changes will be classified as minor, intermediate, or major. Minor changes involve only a few lines of code. Intermediate changes affect several lines of code in one or two subroutines, or require a new subroutine that can be easily written. Major changes involve several subroutines, a new subroutine for which research is required, or a new program. Major changes will not be considered for the rapid response software development process, although parts of it may be applied if the change is urgent or critical.

## 1.9 PERFORMANCE AND PHYSICAL PARAMETERS

The key performance parameters to be considered are reliability (quality of software produced), process duration, and human resources required.

### 1.9.1 RELIABILITY

For this project, the failure rate for a given CSCI is the number of errors per CSCI or related deliverable document per SIOP revision. The failure rates are considered cumulatively over all deliveries of the CSCI throughout its lifecycle. The reliability of each CSCI is expected to be 100%. In other words, every CSCI is expected to have a failure rate of zero failures per delivery. For this reason, the reliability requirement allocated to every level must also be 100%. These failure rates do not include failures associated with the

related hardware. Hardware (HWCI) reliability is not within the scope of this report, as it is mostly beyond the control of NSWCDD. However, hardware is an important part of the SLBM strategic planning system, and should be addressed at some future time.

### 1.9.2 DURATION

The primary objective of this project is to reduce the amount of time between the identification of an error or a new requirement and delivery of the corrected or updated program. Therefore, certain constraints must be placed on the process duration. The rapid response software modification process is broken down into four paths based on the priorities and sizes discussed in section 1.8: critical-minor (CRITMIN), critical-intermediate (CRITINTER), urgent-minor (URGMIN), and urgent-intermediate (URGINTER). The time between definition of the new requirement (or identifcation of an error) and integration of the modified program at the customer site is defined as the implementation time. The implentation time for the paths are as follows: CRITMIN - 2 weeks (10 days), CRITINTER - 4 weeks (20 days), URGMIN - 6 weeks (30 days), and URGINTER - 8 weeks (40 days). One additional week in each path is alotted for post-delivery activities, making the constraints on total process duration: CRITMIN - 3 weeks (15 days), CRITINTER - 5 weeks (25 days), URGMIN - 7 weeks (35 days), and URGINTER - 9 weeks (45 days).

### 1.9.3 HUMAN RESOURCES

The rapid response software modification process requires a variety of human resources. Depending on the task being performed, a degreed or nondegreed professional, level GS-5 through GS-15 may be required. In order to minimize the impact on other projects and to minimize costs, only one person from each of the functional specialties described in section 1.10 should be involved in a single iteration of this process.

In order to further control costs and collateral damage to other projects, the number of work-hours used in this process should be monitored. A work-hour is defined as one hour of actual work devoted to a particular task by one person. Level of effort is a measure of what portion of a given work period is devoted to a particular task. The task duration can be determined from these two quantities. For example, a task which requires 1 person to spend 2 work-hours at a 50% level of effort has a duration of 4 hours. A work-day is defined as eight work-hours. Since the current software development process is new and no metrics have been collected yet, the figures for work-hours will be planned values rather than statistically-based or actual values. The maximum number of work-hours permitted for the completion of the total process for the CRITMIN and URGMIN paths will be three times the process duration constraint for the CRITMIN path (30 work-days/240 work-hours). The maximum for the CRITINTER and URGINTER paths will be three

times the duration for the CRITINTER path (60 work-days/480 work-hours).

Since there is no empirical basis for the number of work-hours, there is also none for cost. Therefore, an arbitrary figure for the maximum allowable cost must be assigned until appropriate metrics are collected. Each iteration of the process should cost less than $50,000 (in FY-95 dollars). The cost of a particular task can be calculated by (1) multiplying the cost per work-hour for each individual by the number of work-hours they are to spend on the task to get each cost per person per task, and (2) adding up the costs per person per task. The following are the costs per work-hour (in FY-95 dollars), based on grade level, that are assumed for this project: GS-7/$55.28, GS-12/$64.58, and GS-14/$84.32. These figures include both salary expenses and overhead expenses, such as utilities, computer resources, office supplies, support staff (security guards, secretaries, janitors, etc.), and facility maintenance. Expected grade levels for each functional specialty are discussed in section 1.10. In addition, a nominal cost of $1.00 per hour is assessed for shipping, the only non-human resource considered in the project to be a separate entity for cost purposes.

### 1.9.4 PHYSICAL PARAMETERS

Physical parameters that will be investigated include hardware, software (development and information management tools),

electronic media used, and hard copy documentation. Human factors will be considered for the physical aspects of the process. Some more intangible aspects, such as the complexity of the process, workload, schedule flexibility, task variety, and communication between project team members will also be addressed.

## 1.10 PROJECT ORGANIZATION

All phases of the rapid response software development process are conducted within the SLBM Research and Analysis Division (K40) or the SLBM Software Development Division (K50) at the Naval Surface Warfare Center, Dahlgren Division (NSWCDD), or in the Strategic Planning Division (J52) at United States Strategic Command (USSTRATCOM). The following personnel are involved:

### 1.10.1 CUSTOMER

Planners in United States Strategic Command Strategic Planning Division (USSTRATCOM/J52). The primary customer is the Missile Strike Section (J5231). Other organizations at USSTRATCOM are also NSWCDD customers, but are not the focus of this process and will therefore not be discussed here. However, the process may be used to meet their needs.

### 1.10.2 LIAISON

The NSWCDD liaison to USSTRATCOM. For the purposes of this project, the liaison is considered the representative of the customer, though he reports to NSWCDD/K40. The liaison is a GS-12.

### 1.10.3 ANALYST

The scientist, mathematician, or engineer responsible for the methodology contained in the software. The analysts responsible for USSTRATCOM software generally works in the Strategic Targeting Branch (K42), but in certain instances may work for the Fire Control Formulation Branch (K41). Currently, all analysts are GS-12s.

### 1.10.4 PROGRAMMER

The computer scientist or mathematician responsible for writing the software based on the analysts design specifications. Programmers for targeting software belong to the System Simulation Branch (K51). Most programmers are GS-12s.

### 1.10.5 QA SPECIALIST

The scientist, engineer, mathematician, or computer scientist responsible for quality assurance (validation and verification) of the targeting software. QA specialists work in the Quality Asssurance Branch (K52). Most QA specialists are GS-12s.

### 1.10.6 QC SPECIALIST

Quality control specialists are technicians or computer scientists responsible for the correctness of software delivery materials, to include media, documentation, and integration instructions. They also work for K52. For the purposes of this project, the QC specialist will be considered a GS-7 technician.

### 1.10.7 MEDIA PREP SPECIALIST

Media prep specialists are part of the Operational Support Branch (K53). They are responsible for verifying transmittals and arranging shipment for software deliveries. For this project, the media prep specialist is assumed to be a GS-7 technician.

### 1.10.8 K42 BRANCH HEAD

Supervisor of the Strategic Targeting Branch. Occasionally, the K41 branch head may fulfill the role of the K42 branch head in this process if K41 software is involved. It is also possible that both may sometimes become involved. In this case, cost projections for some tasks would need to be altered to reflect the cost of two branch heads. Both are GS-14s. For the purposes of this project, it is assumed that the software involved was created by K42 and that K41 is therefore not involved.

### 1.10.9 K51 BRANCH HEAD

Supervisor of the System Simulation Branch. In this process, branch heads are only involved in the decision to make the software modification and in responding officially to the customer. All other decisions are made by the people doing the work. The K51 branch head is a GS-14.

### 1.10.10 K52 BRANCH HEAD

Supervisor of the Quality Assurance Branch. He is also a GS-14. It should be noted that either the K40 division head or the K50 division head or both are likely to become involved at some point in the decision making. Both are GS-15s and would be assigned the same cost per man-hour as a GS-14. However, since time is the binding constraint in this process, decision making has been attributed to the lowest level qualified to make the decision. Therefore, for the purposes of this project, both division heads have been excluded.

### 1.10.11 PLC/POC

Project leader-coordinator/point of contact for a particular software development effort (OD 61596, 2-4).

## 1.11 OPERATIONAL DEFINITIONS

**CSC:**

Computer Software Component. A collection of Computer Software Units (CSUs) that perform a specific task.

**CSCI:**

Computer Software Configuration Item. A collection of CSCs, documents, test cases, and other items to be controlled and delivered as a single entity.

**CSU:**

Computer Software Unit. A particular algorithm or subroutine that performs one or more functions which contribute to the performance of a specific task.

**CSV:**

Computer Software Version. A set of one or more CSCIs that are delivered and executed on a specific hardware configuration item (HWCI) that represents the system requirements for a given "snapshot in time". Significant changes to requirements frequently force the creation of new CSVs, generally on an annual basis or other previously negotiated schedule.

**HWCI:**

Hardware Configuration Item. The computer equipment, also known as "target hardware" on which a given CSV is designed to be executed (OD 61596, 2-3 2-4).

**IDD:**

Interface Design Document. Design specifications for the interface of an CSCI with otherr CSCIs and HWCIs. Not required if a separate Interface Requirements Specification (IRS) does not exist for a particular CSCI (OD 61596, 2-36).

**IPF:**

Integrated Production Facility. Computer network between USSTRATCOM, NSWCDD, and various contractors that is used for software development up to and including the SECRET level.

**IRS:**

Interface Requirements Specification. Specifies the requirements for the interface(s) between one or more CSCIs (OD 61596, 2-25).

**NSDL:**

National Strategic DGZ List. A list of strategic targets.

**NSWCDD:**

Naval Surface Warfare Center, Dahlgren Division.

**RFTL:**

Reserve Force Target List. Similar in function to the NSDL.

**SDD:**

Software Design Description. Contains the detailed design for a given CSCI (OD 61596, 2-32).

**SDP:**

Software Development Plan. A high level plan containing a description of the software development process as it relates to a particular CSV (OD 61596, 2-16).

**SLBM:**

Submarine-Launched Ballistic Missile. Either TRIDENT I or TRIDENT II weapon system.

**SRS**:

(1) Software Requirements Specification. Contains the detailed requirements for a particular CSCI (OD 61596, 2-27). Serves as the basis for the development of the SDD. (2) SLBM Retargeting System.

**SRS DATALINK:**

Computer link between NSWCDD and USSTRATCOM capable of handling traffic up to and including the TOP SECRET level.

**SSDD:**

System Specification and Design Document. Contains the high-level system requirements and the allocation of requirements to CSVs, CSCIs, HWCIs, and organizations involved (OD 61596, 2-21).

**STP:**

Software Test Plan. The STP is written by the QA specialist. It describes the validation and verification approach to be taken for a particular CSCI (OD 61596, 2-30).

**SUBFOOT:**

A file that contains the targeting data for an SLBM weapon system.

**USER GUIDE:**

A document designed to provide information to the user of a particular CSV about how it can be effectively used to facilitate the performance of the user's duties (OD 61596, 2-46).

**USSTRATCOM:**

United States Strategic Command. Stategic missile strike planners in this organization are the primary customers of SLBM targeting software.

**VALIDATION:**

Evaluation of software to determine whether or not it complies with the requirements contained in the applicable SRS and/or IRS.

**VERIFICATION:**

Evaluation of software to determine if is correct and consistent with respect to the design details contained in the appropriate SDD and/or IDD (OD 61596, 2-11).

# 2.0 FUNCTIONAL FLOW

The rapid response software modification process is described in the sections that follow. Flow diagrams for the critical-minor (CRITMIN) process are included. Appendices A-D contain complete sets of flow diagrams and requirement allocation tables for all four processes.

## 2.1 REQUEST PHASE

The first step in the rapid response software modification process is the customer's request. In general, a customer requesting a software modification that meets the criteria for this process will be a missile strike planner at USSTRATCOM. However, it is possible to get a qualified request from another organization at USSTRATCOM or from the Targeting Processing Group in K42 at NSWCDD.

After a requirement to modify targeting software has been identified, it is necessary to collect sufficient data on the change being requested to make an assessment. The most expedient way to accomplish this is to use a form similar to the one in Figure 1. The customer must provide the following information: the date of the request, the name of the person making the request, the software to be modified, when the modification is needed, whether the request

**SOFTWARE MODIFICATION REQUEST**

DATE OF REQUEST:_____

NAME/ORGANIZATION OF REQUESTOR:_____

SOFTWARE TO MODIFY:_____ SIOP REVISION:____

DATE REQUIRED:_____

SUSPECTED ERROR____ REQUIREMENT CHANGE____

DESCRIBE PROBLEM OR CHANGE IN REQUIREMENT:

SOURCE OF REQUIREMENT CHANGE:

IF ERROR, SUSPECTED IMPACT ON TARGETING IS:
NONE___ MINOR___ MODERATE___ SEVERE___

**FOR NSWCDD USE:**

REVIEWING OFFICIAL:_____

INITIAL ASSESSMENT:
ROUTINE___ URGENT___ CRITICAL___
MAJOR___ INTERMEDIATE___ MINOR___

ANALYST::_____

PROGRAMMER:_____

QA SPECIALIST:_____

Figure 1: Sample Software Modification Request Form

is to correct an error or a requirement change, a description of the change or problem, and the impact of the error on targeting or the source of the requirement change. The form will be completed by the requester and the NSWCDD liaison to USSTRATCOM and faxed immediately to the K42 branch head or other responsible party. The liaison should follow up by providing the form via the SRS datalink before the assessment phase is completed. This will simplify electronic tracking. (Of course, if the request comes from the Targeting Processing Group, the liaison would not participate and the form could be e-mailed or handed in.) It is important if the K42 branch head is unavailable, that the liaison contact someone else in K42 who will accept responsibility for getting the process started.

The first thing the K42 branch head or reviewing official must do is to make an initial assessment of the time sensitivity and size of the requested change. If the change is required is within three months, the request must be classified as "critical." If the date is more than three months but less than six months away, the request is "urgent." If it is more than six months away, the request should be classified as "routine" and dealt with via the normal software development process. If the reviewing official thinks that the change involves only a few lines of code, then the change is "minor." If the change is believed to be more substantial, then it should be classified as "intermediate." However, if the reviewing official believes that several subroutines or a new algorithm that would

require research is needed, then the request should be classified as"major." If the software change is classified as major, then this process is not appropriate, and the change must be negotiated. Based on the determination of time sensitivity and scope, the reviewing official will select the appropriate process: "critical-minor" (CRITMIN, see Figure 2), "critical-intermediate" (CRITINTER), "urgent-minor" (URGMIN), or "urgent-intermediate" (URGINTER).

Next, the K42 branch head must assign an analyst. Generally, this will be the same analyst that was responsible for the latest revision of the affected software. Then, the K51 and K52 branch heads or other representatives from those branches must be informed. They will then assign a programmer and a QA specialist. Like the analyst, it is likely that they are already responsible for the affected software.

The request phase as described above and pictured in Figure 3, is the same for all four of the processes. Also, due to the relative simplicity of this phase, a single day is allocated for its completion in all four processes. This gives the customer confidence that NSWCDD will be responsive to their needs.

**EXAMPLE:**

To illustrate this and the rest of the steps in the rapid response software modification plan, a hypothetical change request to a real

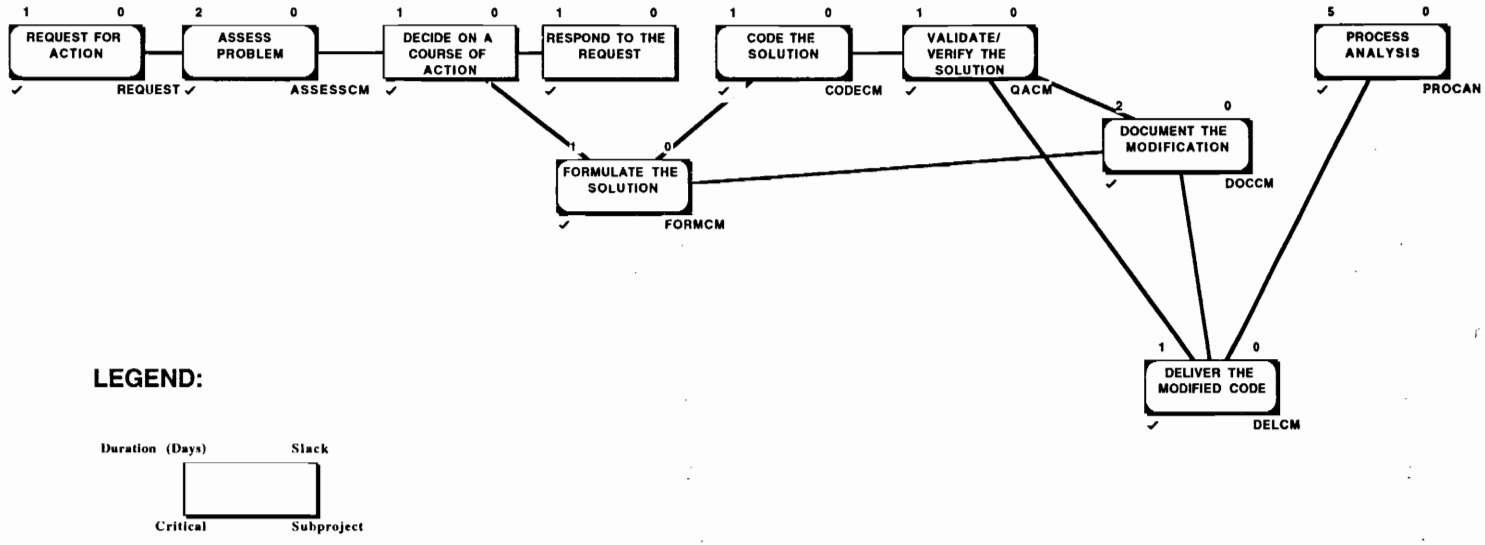



Figure 2: CRITMIN process

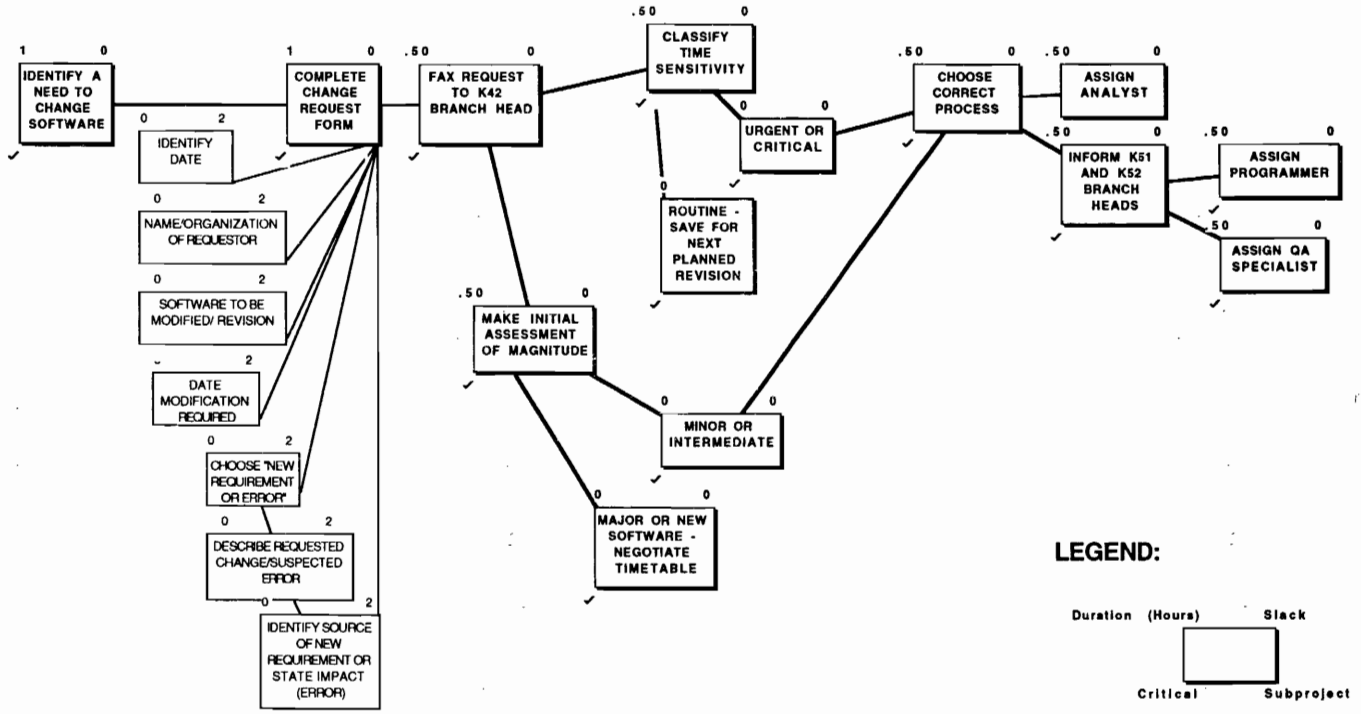



Figure 3: Request Phase

piece of targeting software will be considered. The program is called DATCHK. Its purpose is to check the data on the SUBFOOT file, which contains the targeting information for SLBM targeting assignments, and on the National Strategic DGZ List (NSDL) file, which contains a list of SIOP-eligible targets. DATCHK accomplishes this by comparing the information on one or both files against a set of allowable values and the constraints governing the relationships between those information items. DATCHK can also compare the information items in common between the SUBFOOT and NSDL files (Allen, 1).

The hypothetical scenario to be considered here is as follows: LCDR Schmidt, a missile strike planner at USSTRATCOM, has found out that an additional type of unit "TA" will be required for TRIDENT I SSBNs in the Atlantic and "DP" for TRIDENT II SSBNs in the Pacific. In addition, the larger number of target packages generated by this sudden change in the force commit will cause the number of aimpoints in the SUBFOOT to exceed the 20,000 aimpoint limit in DATCHK (Allen, 3). LCDR Schmidt estimates that the maximum number of aimpoints that could appear in the SUBFOOT is probably about 30,000. The SUBFOOT for SIOP 96 must be sent for processing in mid-June. Therefore, his software integrators will need the software by 1 May. It was decided that all other software affected by this change would not be needed for several months, and could therefore be handled through the normal software development

process. After discussing the situation with the NSWCDD liaison to USSTRATCOM, LCDR Schmidt submitted a software change request form for DATCHK, as shown in Figure 4. The liaison faxed the change request to K42. The K42 branch head is on travel for the next two days, so Sue Roberts, a group leader, has taken responsibility for the software change request. She has made an initial determination that the request is critical and that the level of effort required is intermediate. She has assigned Laura Hoffmann as the analyst. Sue contacted the K51 and K52 branch heads, who assigned Ron Jackson as the programmer and Karla Krause as the QA specialist.

## 2.2 ASSESSMENT PHASE

After the software change request has been evaluated, and a process has been chosen, the analyst and programmer assigned must make an assessment of the change. The first step in the assessment is to determine exactly which computer software version (CSV) and computer software configuration item (CSCI) will be affected. Next, the computer software component(s) (CSCs) and computer software unit(s) (CSUs) that will be affected should be identified from the software design document (SDD) belonging to the CSCI of interest.

If the change request involves a suspected error, the next step is to verify the error condition by running an appropriate test case. The test case may be designed by the programmer and analyst. Or, it may be a test case from some phase of the software development

## SOFTWARE MODIFICATION REQUEST

DATE OF REQUEST: 28 FEBRUARY 1995

NAME/ORGANIZATION OF REQUESTOR: LCDR SCHMIDT, J5231

SOFTWARE TO MODIFY: DATCHK SIOP REVISION: 96

DATE REQUIRED: 1 MAY 1995

SUSPECTED ERROR____ REQUIREMENT CHANGE X

DESCRIBE PROBLEM OR CHANGE IN REQUIREMENT:

-NEED TO PROCESS AT LEAST 30,000 AIMPOINTS IN SUBFOOT
-ADD "TA" (TRIDENT I ATLANTIC) AND "DP" (TRIDENT II PACIFIC) TO THE LIST OF VALID UNITS

SOURCE OF REQUIREMENT CHANGE:

LARGER NUMBER OF TARGET PACKAGES/FOOTPRINTS DUE TO RESTRUCTURING OF FORCE COMMIT

IF ERROR, SUSPECTED IMPACT ON TARGETING IS:

NONE___ MINOR___ MODERATE___ SEVERE___

**FOR NSWCDD USE:**

REVIEWING OFFICIAL: SUE ROBERTS

INITIAL ASSESSMENT:

ROUTINE___ URGENT___ CRITICAL_X_

MAJOR___ INTERMEDIATE_X_ MINOR___

ANALYST: LAURA HOFFMANN

PROGRAMMER: RON JACKSON

QA SPECIALIST: KARLA KRAUSE

Figure 4: Hypothetical DATCHK Modification Request

process, or the actual case that the customer was using when the suspected problem was encountered. If an actual case is used, it should be sent via datalink by the liaison, and it is likely to be classified TOP SECRET. If they are unable to duplicate the problem, the liaison must be told to investigate the driver software into which this CSC has been integrated. If the error condition can be duplicated, the analyst and programmer may proceed with the assessment by checking the SDD logic. If the SDD is incorrect, they must estimate the corrections that will be required to both the SDD and the code. If the SDD is correct, only the code changes need to be assessed.

If the request involves a change to the requirements, the next step after identification of the CSCs and CSUs involved is to verify that the requirements actually have changed. This can easily be accomplished by examining the SDD. Assuming that there is a difference, the impact on the SDD must be estimated, and also the impact on the code and on other software documentation.

No matter whether the request involves an error or a requirement change, the next step is to form a general opinion of the size and complexity of the modifications needed. At this point, if the opinion of the analyst and programmer differs from the K42 branch head's initial evaluation, then they may choose to switch to one of the other three processes. If they decide that the request involves a major change, then the amount of time required to make the

necessary modifications should be estimated. At this point they should notify the K42, K51, and K52 branch heads of the results of the assessment.

Whatever process is used, the tasks that must be performed are the same (see Figure 5). In the CRITMIN path, only one day is allocated for the assessment phase. Two days are allotted in the CRITINTER because there is more to review. In both CRIT paths, the level of effort is expected to be 100%. Two days are allowed in the URGMIN path and four days in the URGINTER path, at an effort level of 50%.

**EXAMPLE:**

The assessment of the DATCHK was completed by Laura, the analyst, and Ron, the programmer. They noted that both the CSCI and the CSV of interest are DATCHK 96, as DATCHK does not have a higher level NSWCDD driver program. DATCHK is composed of five CSCs: an executive routine, a routine that creates a sorted file of unique aimpoints in the SUBFOOT file (unique aimpoints routine), a SUBFOOT data checking routine, an NSDL data checking routine, and a SUBFOOT to NSDL comparison routine (Allen, 3). Laura and Ron determined from the SDD that the addition of two new unit types affects only the SUBFOOT data checking CSC. The increase in aimpoints affects the executive routine (for the NSDL limit) and the unique aimpoints routine (for the SUBFOOT limit) CSCs. On further

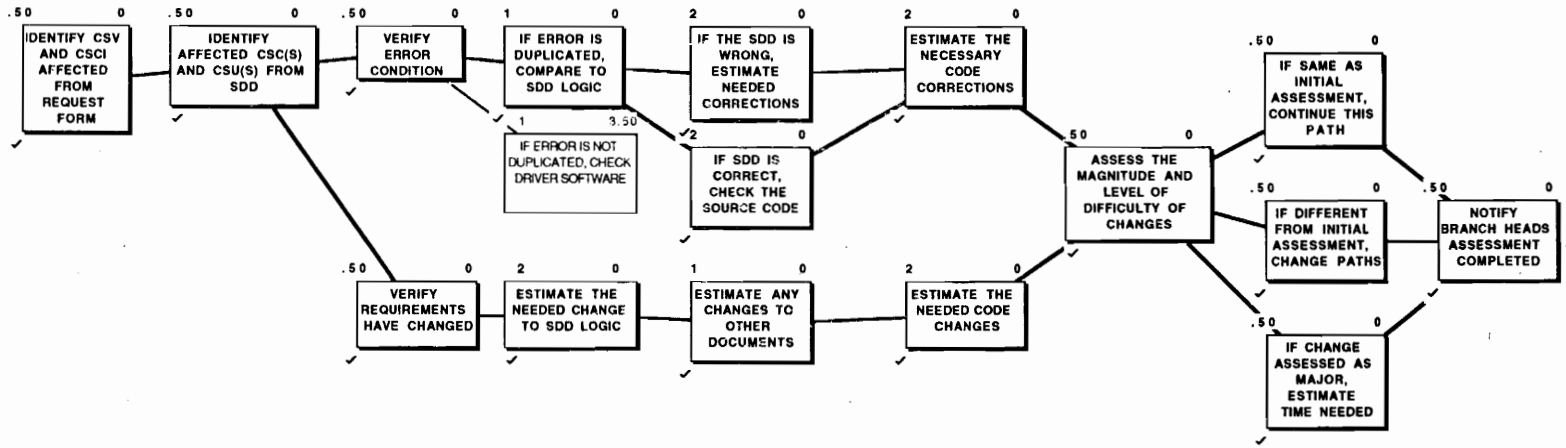



Figure 5: Assessment Phase

review of the SDD, they noticed that although three CSCs are involved, each will require a coding change of only one line. The SDD will also only require minor changes. However, references to the changing parameters are scattered throughout the SDD (OD 61596, 2-32 2-36). It is also suspected that the user guide will also require some modification. No other documents are related to DATCHK since it is self-contained. Laura and Ron decide that although the code changes are minimal, they could use the two weeks of extra time to find all of the changes in the SDD. So they decide against reclassifying the request as minor, since approximately eight weeks are available before the modified version of DATCHK is required at USSTRATCOM.

## 2.3 DECISION PHASE

After the assessment of the software change request is completed, the K42, K51, and K52 branch heads must decide whether to go forward with the change. If there is more than one possible implementation, they must also choose that. In all four paths, this step is only allowed one day. These managers will only spend 25% of their time that day on this task. As stated in the project organization section, the division heads (K40 and K50) may choose to participate, depending on the magnitude of the request. The K41, K43, or K44 branch heads might participate also if the analyst involved is not part of K42. The actual cost computations should be adjusted accordingly.

**EXAMPLE:**

Ron and Laura notify the K42, K51, and K52 branch heads that they have completed their assessment of the software change request for DATCHK 96. They report that the requirements changes will be easy to do and recommend that the branch heads approve the changes. After about fifteen minutes of discussion, the changes are approved.

## 2.4 RESPONSE PHASE

After the managers make the decision to fulfill the request (or to reject it), the K42 branch head or other responsible party must

write a command letter in response to the request. The letter should describe the action to be taken if the request will be honored, or suggest alternatives if the request was rejected. The delivery timetable and name and telephone number of the analyst responsible should be provided. The letter should be signed by the K40 division head and addressed to the O-6 in charge of the requesting branch, with copies to the requester and the liaison. If the K40 division head is unavailable, the acting K40 division head, the K50 division head, or a participating branch heads may sign the letter. Since all that is required of the K40 division head is a signature, it was not deemed necessary to include a cost for his time. For the Target Processing Group's requests, a memorandum signed by a participating branch head is all that is needed. A response should be faxed or sent via datalink within one day. This will help assure the customer that their requests are considered important by the SLBM program and allow them to plan for the delivery.

**EXAMPLE:**

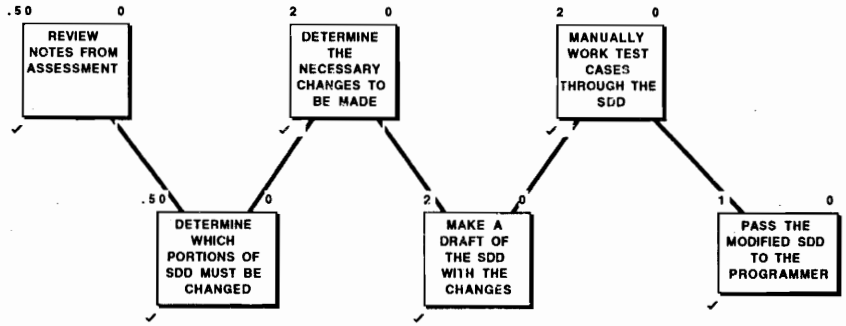
The K42 branch head returns to her office, phones the liaison with the decision, and writes the official response letter to be signed by the K40 division head within 24 hours.

## 2.5 FORMULATION PHASE

The analyst first reviews any notes from the assessment. If the request is to correct an error and it was learned in the assessment phase that only the code contains the error, the formulation phase may be skipped. Next, the analyst must figure out which portions of the SDD must be changed and what those changes should be. Then the analyst can make a draft of the portions of the SDD with the modifications in it. The modified portions of the SDD should be italicized, boldfaced, or otherwise clearly marked.

After the SDD has been modified, the analyst should manually work test cases through the SDD logic, including CSU and CSC interfaces, to ascertain that the logic has been modified correctly. This should also help the analyst determine which other documents might be affected by the change. When the analyst is satisfied that the SDD has been modified correctly, then copies of the modified SDD should be given to the programmer and the QA specialist. It would be helpful, but is not required, for the analyst to also provide the test cases used during this phase of the process.

The formulation phase contains the same basic tasks for all four paths (see Figure 6). In the CRITMIN path, only one day is allotted for modification of the SDD. In the CRITINTER path, four days are allowed. It is expected that the analyst would devote 100% of his or her time to this part of the project. In the URGMIN path, however, the




**LEGEND:**

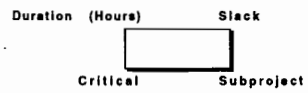



Figure 6: Formulation Phase

analyst is only expected to expend a 20% effort, and is allowed five days to complete these tasks. For the URGINTER path, eight days are allowed, with the analyst spending 50% of his or her time on this effort.

**EXAMPLE:**

After reviewing her notes from the assessment phase, the analyst examines the SDD. She determined that preliminary design subsections 2.2.1 (executive CSC), 2.2.2 (unique aimpoints CSC), and 2.2.3 (SUBFOOT checking CSC) are affected. Detailed design subsections 3.1 (executive CSC - restrictions), 3.1.5 (executive CSC - create sorted NSDL files CSU), 3.2 (unique aimpoints CSC - restrictions), and 3.1.2 (SUBFOOT checking CSC - check footprint header data CSU) (Henderson, 8 13 21). Laura double-checks the logic with three test cases she has designed. After deciding that her logic changes are correct, she makes the changes in italics and passes the modified SDD, a summary off her changes, and her test cases to the programmer and the QA specialist.

## 2.6 CODING PHASE

Unlike the formulation phase, which can be skipped if the only impact of a change request is to the code, the coding phase must always be done. Software change requests that affect documentation

only do not meet the criteria for the rapid response software modification process.

The first thing that the programmer must do is review the modified SDD to make sure that he or she understands the changes. If there is any question about comprehension, or about the correctness of the modifications, the analyst should be consulted immediately. The next thing the programmer must do is to find out which subroutine(s) that must be modified. Then he or she must decide exactly what changes need to be made. The programmer should make a copy of each affected subroutine before altering the code.

After the affected subroutine(s) have been modified, the programmer should compile them and link them to the rest of the program. It is then possible to run test cases through both the original and modified versions. The difference in the results should match the analyst's expectations. If both the programmer and the analyst are satisfied with the results, the programmer can release the modified subroutines and the modified executable to the QA specialist.

For the URGMIN path, the programmer is expected to spend approximately 20% of his or her time each day on coding these changes. Up to five days are allowed for this part of the process. For URGINTER, the programmer has eight days to complete these tasks, and is expected to work on it 50% of the day. For both CRIT paths,

the programmer is to devote his or her time exclusively to these tasks until they are completed. In the CRITINTER path, he or she is allowed four days, but only one day is budgeted for coding in the CRITMIN path (see Figure 7). The tasks are the same for all four paths.

**EXAMPLE:**

Ron reviews the SDD changes and decides that no further clarification is needed. He makes a copy of the SRTNDL, UNIAIM, and CHKHDR subroutines. Then he makes the changes, which are a single line each (two integer variable assignment statements and one character array), compiles the modified subroutines, and links them with the rest of the subroutines. Next, he runs the design test cases through the modified executable and shows the results to Laura. She agrees that the results are correct, so Ron gives the QA specialist permission to the modified executable, and the source code of the modified subroutines. He also types up a brief summary of the changes he made.

## 2.7 VALIDATION & VERIFICATION PHASE

It is the job of the quality assurance specialist to make sure that everything the programmer and the analyst have done up to this point is consistent and makes sense. The first thing that the QA specialist must do is to collect everything (see Figure 8). A copy of the

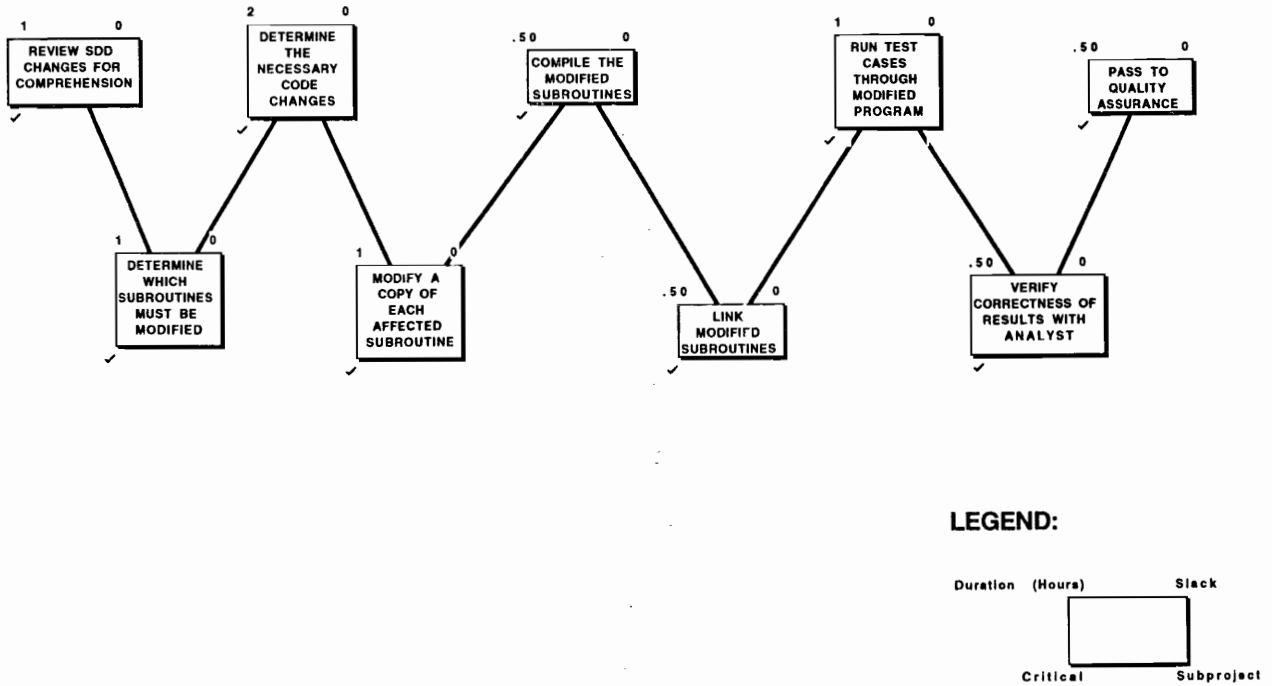



Figure 7: Coding Phase

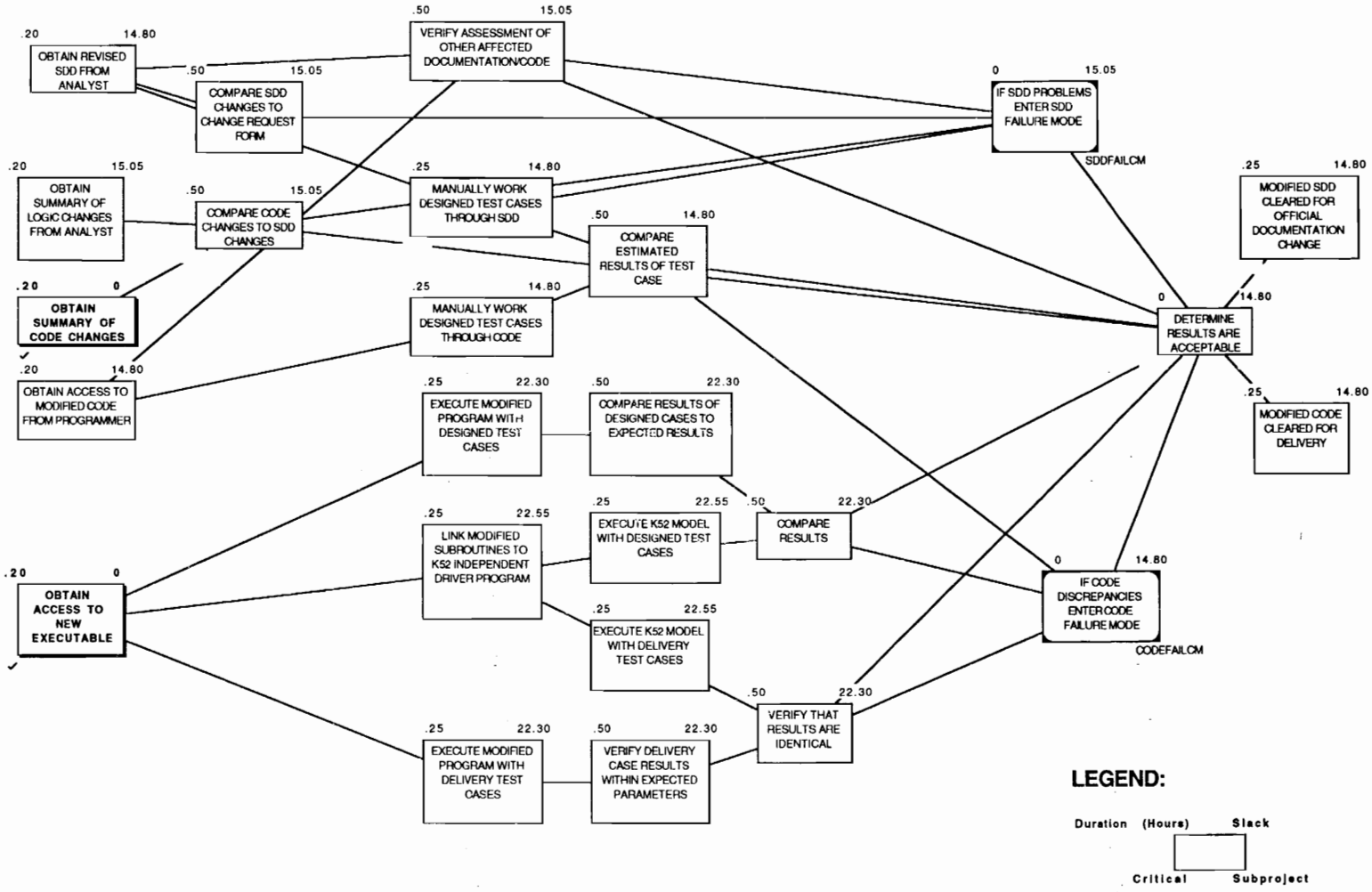



Figure 8: Validation and Verification Phase

modified SDD and a summary of the changes are obtained from the analyst. Access to the modified code, the modified executable, and a summary of the code changes is obtained from the programmer. The programmer and the analyst may provide their summaries verbally if the changes are simple.

The next step is to conduct static testing. The QA specialist should compare the SDD changes to the request form information, and the code changes to the SDD changes. The assessment notes should be examined and the determination of the impact to other documentation should be verified. The QA specialist should then manually work designed test cases through both the code and the SDD logic and compare the results. It may not be feasible to do all of the calculations manually, so the results will not be exact. The designed test cases are those specifically intended to produce results relevant to the change. The same ones used by the programmer during the coding phase and/or by the analyst during the formulation phase may be used, or the QA specialist may create his or her own test cases (OD 61868, 6-2).

If the results of the designed test cases do not match sufficiently, an attempt must be made to detect if the SDD, the code, or both are incorrect. The validation and verification phase then enters one of the failure modes based on this determination. The basic flow of either failure mode (Figures 9 and 10) is notification of either the analyst (if an SDD failure is diagnosed) or the

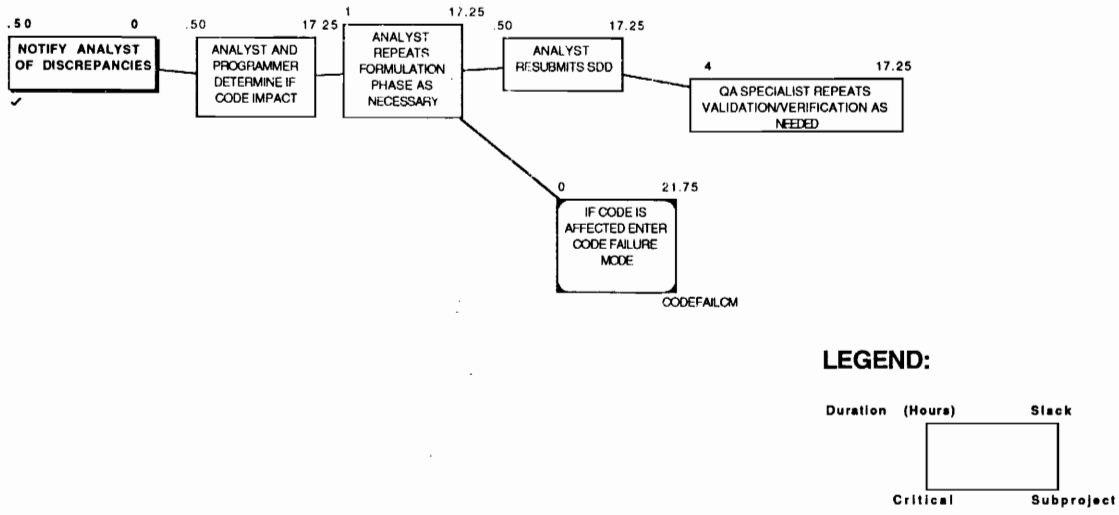
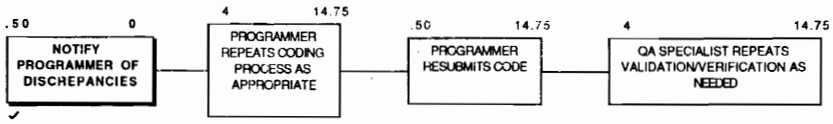



Figure 9: SDD Failure Mode




**LEGEND:**

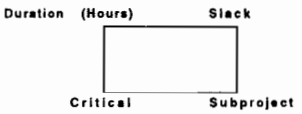



Figure 10: Code Failure Mode

programmer (if a coding problem is discovered). Then the responsible party repeats the portions of either the formulation phase or the coding phase necessary to correct the problem and resubmits the correct copy to the QA specialist. It should be noted that correction of the SDD could possibly drive a correction to the code.re the code failure mode can be entered from the SDD failure mode.

In the URGMIN and URGINTER processes, static testing involves an additional step. The QA specialist will examine the code changes to ensure that they conform to SLBM program standards. This is not part of the CRITMIN or CRITINTER processes due to the shorter amount of time available (OD 61868, 6-1).

The QA specialist must also conduct dynamic testing. This involves executing the modified code with test cases, as opposed to manually working through the test cases using a printout of the code. The affected CSC(s) is then linked to the K52 independent driver program created for use in the validation and verification of normal deliveries of this software. The K52 model is executed with the same test cases as the modified executable that was obtained from the programmer. The results are then compared, either visually and/or with the assistance of comparison software. These tasks are done for the designed test cases and for the set of test cases from the most recent normal delivery. If the results do not match, the process enters the code failure mode as described in the static testing. If the results from both the static and dynamic testing are

acceptable, then the QA specialist should clear the SDD changes and the modified code for delivery (OD 61868, 6-3 6-12).

As stated before, the URGMIN and URGINTER processes contain an extra task. The amount of time allotted for these processes is four days for both, with levels of effort of 25% and 50%, respectively. The level of effort expected in the CRITMIN and CRITINTER processes is 100%. The amount of time allowed for the validation and verification phase in these processes is one day for CRITMIN and two days for CRITINTER.

**EXAMPLE:**

Karla collects the SDD, the modified executable, the modified subroutines, and the summaries. She first compares the SDD changes to the request form. They match, so she compares the SDD changes to the source code of the modified subroutines. Next, she looks at the assessment notes and agrees that the user guide will also have to be changed. Since she does not have that yet, Karla works Laura's designed test cases through the SDD logic and the source code. A comparison reveals that the results would be the same. She decides that the SDD is correct. Next, Karla turns her attention to the modified executable. She runs the designed test cases through the modified executable and saves the results. She does the same with the test cases from the most recent delivery. Then Karla takes the modified subroutines, compiles them, and links them to the K52

driver for DATCHK. She runs the same test cases through this model. Then she compares the results and verifies that they match. Karla clears both the modified code and the modified SDD for delivery.

## 2.8 DOCUMENTATION PHASE

After the SDD changes have been made, it is necessary for the analyst to make any changes to related documentation that may be required. This phase may begin immediately after the formulation phase has ended, but should not be completed until after the validation and verification phase has been completed, since that investigation may uncover errors in the SDD. The other reason that the documentation phase may not be completed until after the validation and verification phase has finished is that the QA specialist must also participate in the documentation phase.

The first thing that the analyst should do is to make official change pages for the SDD (see Figure 11). Next, he or she should review the notes on additional documentation changes from the assessment phase to find out what needs to be done next. If the software modifications affect the CSCI interface with other CSCIs, it may be necessary to change the interface requirements specification (IRS) and the accompanying interface design document (IDD). This is more likely to happen in the CRITINTER and URGINTER scenarios. Any changes that have only an internal impact on the CSCI of interest will not require performance of those two tasks. Another

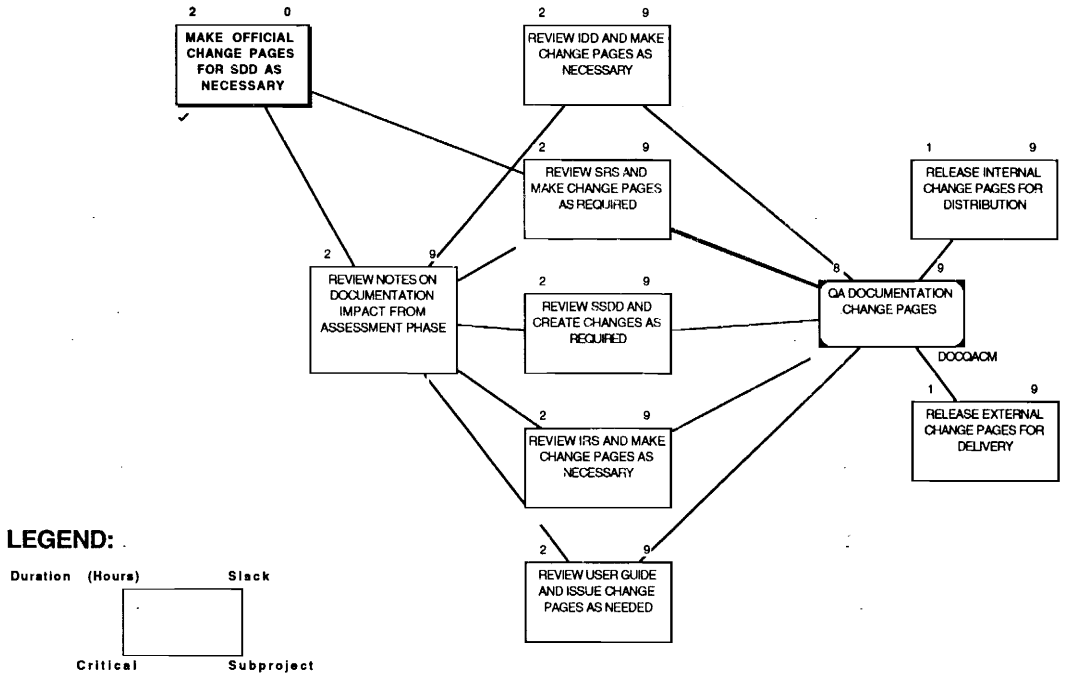



Figure 11: Documentation Phase

document that might require modification if more than one CSCI is involved is the system specification and design document (SSDD). This is a high-level document that contains the requirements for various CSVs and CSCIs and how they interact with hardware and the supporting functions. A change that would affect an SSDD would most likely be classified as "major." Therefore it will not be encountered in this process except in unusual circumstances, and was included mostly for completeness. The fifth type of document that may need to be updated is the system requirements specification (SRS). The SRS contains detailed design requirements for the CSCI. Therefore it is highly possible that this document would require changes in any scenario, except errors found in the code but not in the SDD. If the requirements have changed, it is also likely that a user guide change must be made.

After the analyst has completed the changes to a particular document, they may be passed to the QA specialist for validation and verification (see Figure 12). For the SDD changes, the QA specialist will compare the official change pages to the changes approved during the validation and verification phase. He or she will trace the SDD changes through the other related documents. This enables the QA specialist to verify each set of documentation changes when they are completed and handed in. If the changes to any document are unacceptable, they are returned to the analyst for correction and resubmission.

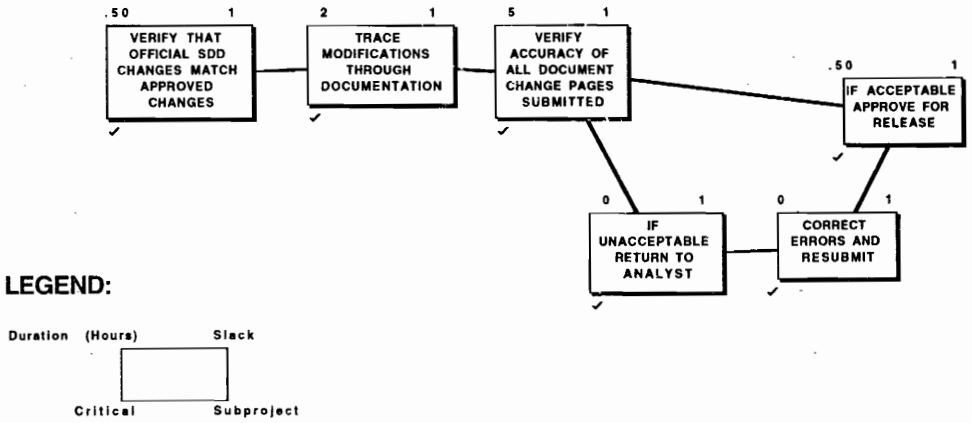



Figure 12: Documentation Quality Assurance Mode

When all of the documentation changes have been completed and have passed quality assurance, then they are approved for internal (to NSWCDD) release or external delivery. All documents are released internally, but only the IRS and user guide are delivered externally. This entire phase should only take two days to complete at an effort level of 100% for CRITMIN and five days at the same level of effort for CRITINTER. Five days are also allowed for completion of this phase of URGMIN. However, the analyst and QA specialist are only expected to spend 40% of their work day on it. For URGINTER, five days' worth of work on this phase is done at a 50% level of effort, making the total amount of time allotted ten days.

**EXAMPLE:**

When she has passed the SDD to the programmer and the QA specialist, Laura makes the official SDD change pages, including the list of effective pages. She gives the official change pages to Karla. Karla will verify that the changes are the same ones that she approved, when she has finished the validation and verification phase. Next, Laura changes the "restrictions" section in the user guide to show the increased limit, and the description of the SUBFOOT checks made to reflect the new allowable unit types. Laura also passes the new user guide pages to the QA specialist, who by this time has completed the validation and verification phase, and

has also approved the official SDD change pages. Karla verifies the new user guide pages and approves them for delivery.

## 2.9 DELIVERY PHASE

This is the last phase of the rapid response software modification process that is time sensitive, because this is the last phase that is of interest to the customer. After all of the documentation changes and code changes have passed quality assurance, the delivery process can begin. First, the programmer and the analyst must make a delivery copy of all code, the document change pages, and the code integration instructions to be released to USSTRATCOM. For the CRIT paths (see Figure 13), the delivery copies take the form of a floppy disk. For the URG paths, a tape is required for code delivery, and hard copies of the document changes and integration instructions are used.

In all paths, the next step is for the quality control specialist to compare the delivery copies to the master copies that were previously approved by the QA specialist. Assuming they match, the QC specialist will then perform an integration using the instructions. He or she will run at least one test case through this model and compare the results to those obtained from the executable approved by the QA specialist. Assuming those results are acceptable, the documentation changes, the code, and the integration instructions are ready for delivery.

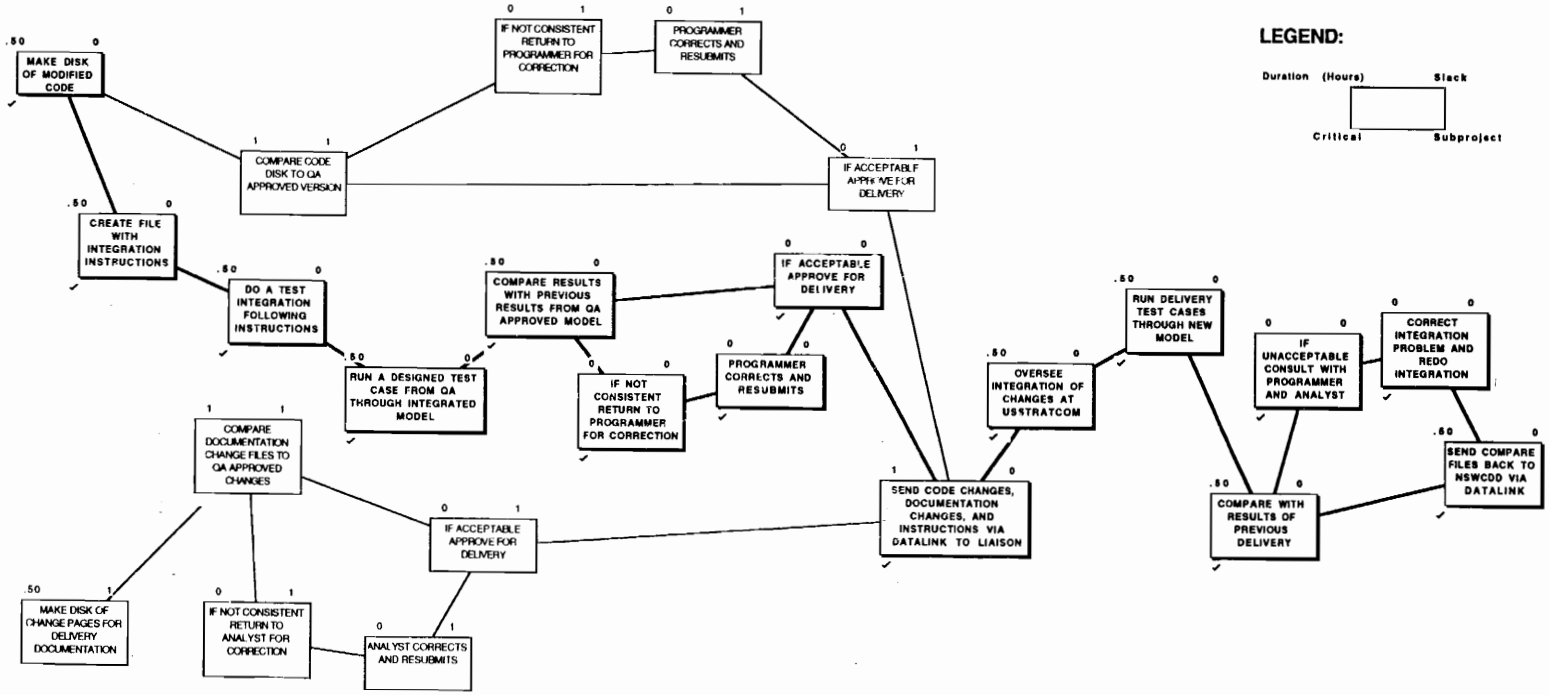



Figure 13: Delivery Phase

In the CRIT paths, delivery is made by the QA specialist and the analyst. If K52 prefers, the QC specialist may replace or augment the QA specialist in this task. If that occurs, however, the actual cost computations should be adjusted accordingly. The disks are placed in the SRS datalink computer and the files are sent to USSTRATCOM. The files are available at USSTRATCOM immediately. The SRS datalink may be used for all classification levels through TOP SECRET.

In the URG paths, the tape(s), the change pages, and the instructions are listed on a classified document transmittal. The quality control process includes checking to make sure the delivery items are labeled with the proper classification markings, downgrading instructions, and registry numbers. The delivery items are then given to the media preparation specialist, who verifies that the delivery items match the information on the transmittal. If everything matches, the media prep specialist then wraps the delivery items and takes the package to the mail room for shipment via registered mail (for SECRET deliveries or below). For TOP SECRET deliveries, the package is instead taken to the TS control vault, where it is barcoded and shipped via Defense Courier Service (DCS). Shipment by registered mail takes approximately one week to ten days. For SECRET deliveries, registered express mail may also be used; this reduces shipping delays to approximately three days. Shipment via DCS takes 1-3 weeks ("Procedures", 1 2).

When the delivery reaches USSTRATCOM, the liaison should oversee the integration of the changes to the software, and ensure that change pages are placed in the user guides and in the IRS (where applicable). He or she should run the test cases from the most recent normal delivery through the revised model and compare the results to those obtained with the previous version of the model. The comparison results should be sent back to NSWCDD via the SRS datalink.

For the CRITMIN and CRITINTER paths, all personnel are expected to put 100% of their effort into those tasks until they are complete. Delivery should take only one day. For URGMIN and URGINTER, ten days are allowed for delivery (five of them are for shipping). Personnel are to devote 50% of their time to the delivery phase. Shipping accounts for five days, so that leaves only five days for the actual work to be accomplished. It is recognized that when deliveries are shipped, NSWCDD has little control over the shipping time, and therefore also has little control over the total time required for any given delivery.

**EXAMPLE:**

Laura makes a disk copy of her documentation changes. Ron makes a copy of his modified subroutines. Then he writes an integration procedure and puts that on a disk also. The QC specialist compares the code and SDD files against the ones approved by the QA

specialist earlier. Since they match, the QC specialist clears them for delivery. He tries Ron's integration instructions, but decides that they are ambiguous. Ron fixes the steps in question and resubmits his procedure. This time it passes, so the Karla and Laura take the disks down to the datalink room. They place the files on the datalink computer and telephone the liaison, who removes the files and gives them to the USSTRATCOM programmer in charge of integrating the DATCHK software. They upload the files from the datalink disks to TRICOMS. Then they perform the integration on TRICOMS together, and run the test cases from the previous delivery. The results match with those from the previous version of the code, so the liaison dumps the output files onto a disk and places them on the datalink. He telephones the analyst to inform her that the delivery is complete.

## 2.10 PROCESS ANALYSIS PHASE

Process analysis is the final step in the rapid response software modification process. This phase primarily involves the analyst, the programmer, and the QA specialist, but any of the people involved may participate. The process analysis phase is the same for all four paths (see Figure 14).

For software change requests involving errors, the software test plan (STP) and the project folder should be obtained from the project leader/coordinator (PLC). All of the original documentation and

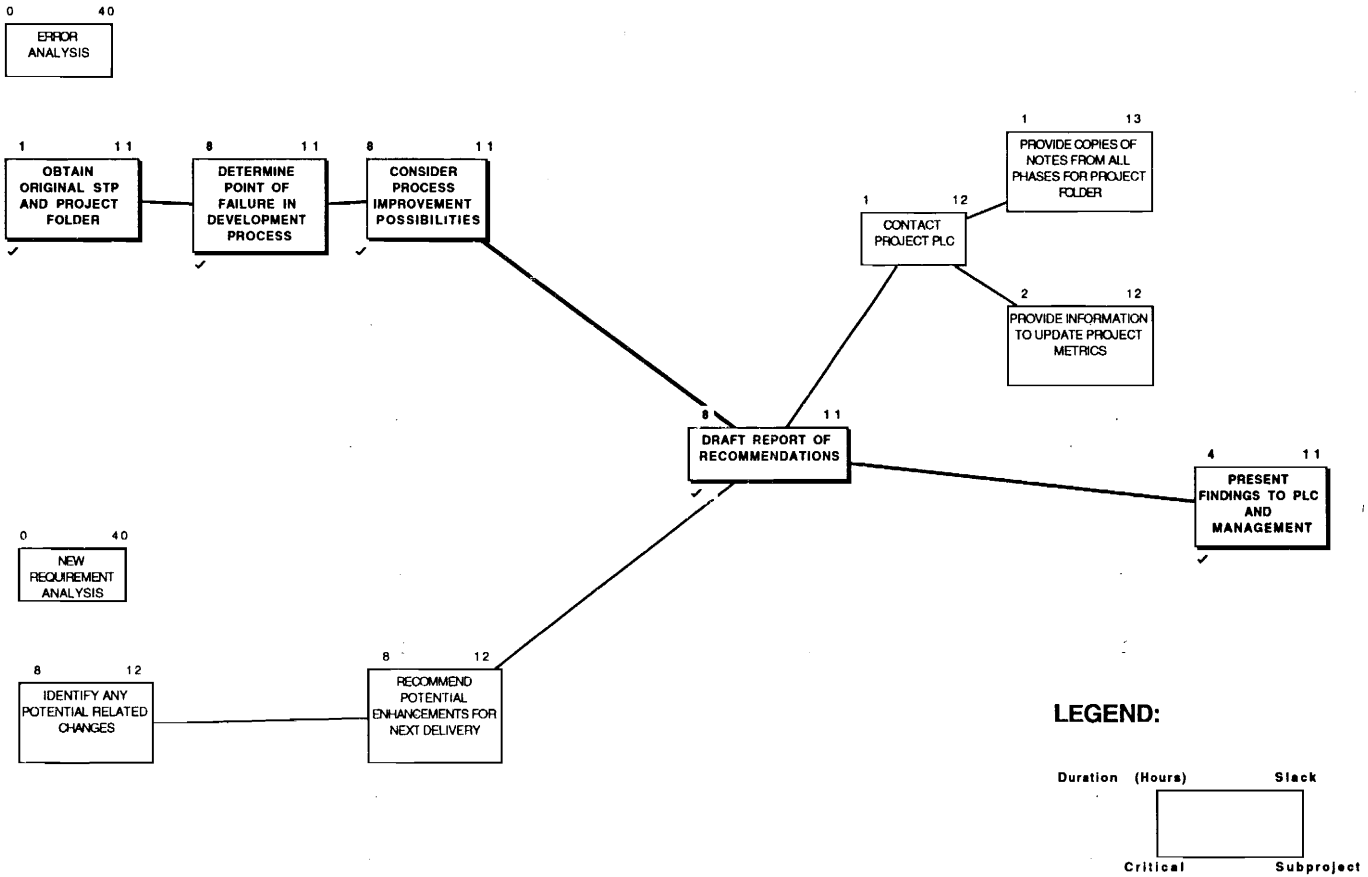



Figure 14: Process Analysis Phase

notes should be examined to detect where in the software development process the point of failure was. Then it is up to these three individuals to discuss possible ways to improve the software development process to prevent further errors of the same type.

For software change requests involving new or modified requirements, any potentially related changes should be identified. Recommendations should be made for possible enhancements for the next regular delivery.

The rapid response software modification process should also be evaluated for possible improvements, whatever the reason for the software change request. A written report describing all actions taken, all applicable metrics, any deviations from the process, and recommendations will be presented to management. The PLC should be given copies of notes for the project folder. The PLC should also be given time and cost information, and any other information necessary to update the project metrics. A single review for the PLC, management, and any other interested parties that covers everything done as part of this process should be given. This review replaces several separate reviews in the normal software development process. Five days are budgeted for process analysis. However, since the delivery has already been made, additional time may be used if necessary.

# 3.0 SYSTEM ANALYSIS

## 3.1 PROCESS ANALYSIS

At this point, it is important to review the requirements defined in section 1.9 to verify that the rapid response software modification process described in section 2.0 complies with them. As this is a new process, and since no statistical data was available upon which to base the design, it is impossible at this point to determine the accuracy of the time, effort, and cost estimates are accurate. This analysis should be revisited after the process has been used a few times.

### 3.1.1 RELIABILITY

As stated in section 1.9.1, the reliability requirement for all parts of this process is 100%. It is impossible at this stage to determne if that requirement has been met, since the process is new. However, there are several tasks of a quality assurance or quality control nature in the process, which will hopefully keep reliability high once the process is in use.

### 3.1.2 DURATION

The requirements for the total process duration defined in section 1.9.2 are: 3 weeks (15 days) for the CRITMIN path, 5 weeks (25 days) for the CRITINTER path, 7 weeks (35 days) for the URGMIN

path, and 9 weeks (45 days) for the URGINTER path. The expected total process durations are shown in Figure 15. In the CRITMIN path, nearly one-third of that time is allocated to the process analysis phase. But for the URGMIN and URGINTER paths, delivery becomes the most lengthy phase (see Figure 16). Subtracting out the duration of the process analysis phase, which happens after delivery is completed, gives the implementation time. The implementation time requirements have also been met, as shown in Figure 17.

### 3.1.3 HUMAN RESOURCES

As many as six people were involved in some phases of the rapid response software development process. Obviously, that level of participation can become expensive. Therefore, it is necessary to place some constraint on the amount of work-hours involved in the process. For this analysis, work-days will be used instead, since it is easier to compare work-days to the process duration. (Recall that a work-day equals eight work hours.) Some portions of the process are human resource-intensive, using two or three times more work-days (see Figure 18) than the durations of those tasks. Other parts of the process involve fewer work-days than the task duration. Due to this wide range of values, as well as the high degree of uncertainty in them, it makes more sense to impose a limitation on the total number of work-days expended on the process, rather than setting individual requirements for each phase. These limits, as discussed in section 1.9.3, are 30 work-days for the CRITMIN and URGMIN paths,

| PATH | DURATION |
|---|---|
| CRITMIN | 14 DAYS |
| CRITINTER | 25 DAYS |
| URGMIN | 35 DAYS |
| URGINTER | 40 DAYS |

Figure 15: Total Process Duration

| PHASE | CRITMIN | CRITINTER | URGMIN | URGINTER |
|---|---|---|---|---|
| REQUEST | 1 | 1 | 1 | 1 |
| ASSESSMENT | 1 | 2 | 2 | 4 |
| DECISION | 1 | 1 | 1 | 1 |
| RESPONSE | 1 | 1 | 1 | 1 |
| FORMULATION | 1 | 4 | 5 | 8 |
| CODING | 1 | 4 | 5 | 8 |
| V&V | 1 | 2 | 4 | 4 |
| DOCUMENTATION | 2 | 5 | 5 | 10 |
| DELIVERY | 1 | 1 | 10 | 10 |
| PROCESS ANALYSIS | 5 | 5 | 5 | 5 |

Figure 16: Durations of Individual Phases (Days)

| PATH | IMPLEMENTATION TIME |
|---|---|
| CRITMIN | 9 DAYS |
| CRITINTER | 20 DAYS |
| URGMIN | 30 DAYS |
| URGINTER | 35 DAYS |

Figure 17: Process Implementation Time

| PHASE | CRITMIN | CRITINTER | URGMIN | URGINTER |
|---|---|---|---|---|
| REQUEST | 2.000 | 2.000 | 2.000 | 2.000 |
| ASSESSMENT | 2.125 | 4.125 | 2.125 | 4.250 |
| DECISION | 0.750 | 0.750 | 0.750 | 0.750 |
| RESPONSE | 0.250 | 0.250 | 0.250 | 0.250 |
| FORMULATION | 1.000 | 4.000 | 1.000 | 4.000 |
| CODING | 1.000 | 4.000 | 1.000 | 4.000 |
| V&V | 1.000 | 2.000 | 1.000 | 2.000 |
| DOCUMENTATION | 2.000 | 5.000 | 2.000 | 5.125 |
| DELIVERY | 1.125 | 2.125 | 2.500 | 2.500 |
| PROCESS ANALYSIS | 15.000 | 15.000 | 15.000 | 15.000 |

Figure 18: Number of Work-Days in Each Phase

and 60 work-days for the CRITINTER and URGINTER paths. These constraints are met with plenty of time left over. The actual total number of work-days alotted for each path is listed in Figure 19.

### 3.1.4 COST

As stated before, the nature of this constraint is somewhat arbitrary due to insufficient data. The maximum cost constraint for the entire rapid response software modification process is $50,000 (in FY-95 dollars). The basis for the planned costs are derived from the number of work-hours and the cost per work-hour assigned to each of the participants in the process. Some assumed costs per work-hour for various GS levels in the SLBM program at NSWCDD for fiscal year 1995 are presented in section 1.9.3. Typical GS levels for each functional specialty are assigned in section 1.10. Figure 20 ties that information together. The next step in cost computation is to determine the cost of each phase of the process for each path. The results of those calculation can be viewed in Figure 21. Finally, those costs are tabulated to obtain a total process cost for each path. The total costs (see Figure 22) computed for this process are well under the maximum limit. However, it is suspect that the amount of work-hours required in the process analysis phase has been overestimated, and since that phase accounts for approximately one-third to one-half of the total process cost, the actual total process costs will probably be lower than planned, assuming a reasonable degree of accuracy in all other estimates.

| PATH | WORK-DAYS |
|---|---|
| CRITMIN | 26.250 |
| CRITINTER | 39.250 |
| URGMIN | 27.625 |
| URGINTER | 39.875 |

FIGURE 19: Projected Total Expenditure of Work-Days

| POSITION | COST PER MAN-HOUR | COST PER MAN-DAY |
|---|---|---|
| ANALYST | $64.58 | $516.64 |
| K42 BRANCH HEAD | $84.32 | $674.56 |
| K51 BRANCH HEAD | $84.32 | $674.56 |
| K52 BRANCH HEAD | $84.32 | $674.56 |
| LIAISON | $64.58 | $516.64 |
| MEDIA PREP SPEC | $55.28 | $442.24 |
| PROGRAMMER | $64.58 | $516.64 |
| QA SPECIALIST | $64.58 | $516.64 |
| QC SPECIALIST | $55.28 | $442.24 |
| SHIPPING | $1.00 | $8.00 |

Figure 20: Cost of Human Resources

| PHASE | CRITMIN | CRITINTER | URGMIN | URGINTER |
|---|---|---|---|---|
| REQUEST | $1725.75 | $1725.75 | $1725.75 | $1725.75 |
| ASSESSMENT | $2451.78 | $4712.08 | $2452.28 | $4776.66 |
| -DECISION | $505.92 | $505.92 | $505.92 | $505.92 |
| RESPONSE | $168.64 | $168.64 | $168.64 | $168.64 |
| FORMULATION | $1033.28 | $4133.12 | $1033.28 | $4133.12 |
| CODING | $1033.28 | $4133.12 | $1033.28 | $4133.12 |
| V&V | $2599.35 | $4617.47 | $2131.14 | $4197.70 |
| DOCUMENTATION | $3616.48 | $6199.68 | $3099.84 | $6361.13 |
| DELIVERY | $1097.34 | $2065.52 | $1671.80 | $1671.80 |
| PROCESS ANALYSIS | $17242.86 | $17242.86 | $17242.86 | $17242.86 |

Figure 21: Projected Cost of Each Path

| PATH | PROJECTED COST |
|---|---|
| CRITMIN | $31474.67 |
| CRITINTER | $45504.16 |
| URGMIN | $31064.79 |
| URGINTER | $44916.70 |

FIGURE 19: Projected Total Process Cost

## 3.2 HUMAN FACTORS

No process design would be complete without some analysis of human factors. The rapid response software modification process is only intended to encompass short periods. Therefore it is not appropriate to expect any workplace redesign to support it unless one or more tasks could not be performed using the current workplace design. However, certain human factors issues can be considered in the process design.

### 3.2.1 OFFICE DESIGN

Most offices are shared by two people. They are relatively spacious, except offices that contain a third person or an additional common work area. Each person has a desk with hutch and two under-desk file cabinets, a five-drawer file cabinet, a bookcase, a terminal table, printer table, two adjustable chairs (one with arm rests and one without armrests), a waste basket, and a telephone. Each worker has at least one computer, but several have more than one.

The terminal tables are specially designed for computers. They are sufficiently deep to allow the developer to move the monitor forward or backward to adjust for visual distance from the screen. The keyboard may be placed on the table or in one's lap. There is also enough room to shift the equipment right or left to place the mouse

on either side. This is important because there are many left-handed people in K40/K50. The height of the terminal table is adjustable.

The heights of both office chairs are adjustable, as are the backs of the chairs. They are padded and covered in woven cloth. Both chairs have casters.

The desks are not adjustable. However, since the chair height can be adjusted, this is not a problem. The under-desk file cabinets can be positioned one on each side under the desk, or both on the same side under the desk, or not under the desk at all. Therefore the worker may sit in the center or on either side, depending on individual preference. The hutch does interfere with illumination of the workspace, but adjustable desk lamps that fit under the hutch are available.

There is little that can be done about the computers or the office furniture. However, people can be educated in correct seated posture and in the proper chair and table heights. That way they can adjust their furniture to meet their own needs. Employees should be allowed to ask for foot rests. Since this process will sometimes require certain individuals to spend lengthy periods on their computer, keyboard trays and wrist rests should also be available to reduce the possibility of repetitive motion injuries like Carpal Tunnel Syndrome (Kodak 2, 246-257).

### 3.2.2 ENVIRONMENT

The office areas in K40/K50 do have some minor environmental problems. Overhead fluorescent lighting is used in all offices. This type of lighting increases screen glare. Each office has two switches to turn the lights off and on, thus placing a certain amount of control of illumination levels in the hands of the employees. However, most people need both sets of lights turned on anyway, due to the intensely visual nature of the task. The resulting screen glare can be mitigated by changing the background color of the screen and by placing a filter over the screen. In cases where this is not adequate, employees should be encouraged to have their vision checked. Often, glasses can be tinted to lessen the effects of excess illumination and/or glare (Kodak 1, 225-234).

There is a certain amount of background noise from the hallways and neighboring offices. The levels of noise are not great enough to cause any more than a nuisance. In some offices, employees may close the doors to reduce background noise. However, certain office areas require that doors remain open for security reasons. In these cases, most people can adapt to the background noise. The only exception to that assessment is the noise made while the janitor is vacuuming, which is done once per week. People can be encouraged to take breaks while this task is done (Kodak 1, 210-219). It may be prudent to address this issue when the janitorial contract is

renegotiated; vacuuming should not be done during peak hours (0900-1500).

The temperature of the work areas also presents a distraction. Some areas, while not outside the safety limits, are too cold for the average person to be comfortable. Other areas are too warm. Although it would be conceivable to adapt to the office temperature by wearing heavier or lighter clothing, this becomes impractical when it is considered that people move between office areas frequently during the day. Regardless, there is little else that can be done (Kodak 1, 244-262).

During winter months, the lack of humidity also produces discomfort. Increase fluid consumption is necessary. The levels are low enough to increase the amount of electrostatic charge dramatically. It is not unusual for people in certain work areas to receive small electric shocks from computer equipment, badge readers, and metal surfaces. (One such shock was sufficient to create visible sparks and short out a badge reader. Another such shock from a computer screen lifted an employee out of her chair.) This condition also presents a hazard to equipment. Antistatic mats should be provided, and the carpeting should be sprayed with chemicals to inhibit static buildup. Employees should be educated about the hazards associated with this condition.

### 3.2.3 HUMAN/COMPUTER INTERFACE

Everyone has at least computer in their office. Many have two or more computers with different equipment configurations. This presents a difficult challenge .

There are several keyboard models used in the SLBM program. All of them have the "QWERTY" layout of alphanumeric keys. Some have an additional numeric keypad. The major differences, however, are in the physical design of the keyboard and in the function keys. Some keyboards are flat, while others are inclined slightly, which is preferable. The size and shape of the keys, the amount of pressure required to activate them, and the spacing of the keys also vary. Those people fortunate enough to have newer equipment have the keyboards with better ergonomic designs. People who use laptop or notebook computers regularly must adapt to smaller keyboards with less space between the keys. Others still have aging "dumb terminals" with flat keyboards that require much more force in striking the keys, which can cause discomfort in the hands and wrists after extended periods of use. Because so many diferent keyborad designs are in use, it becomes difficult to work on one computer and then another. Differences in key size, spacing, and location tend to increase the frequency of data entry errors (Kodak 1, 134-140). The differences in the function keys available (both on keyboards connected to different systems and on different keyboard designs for the same system) often cause confusion and lower

productivity because "F11 isn't there", or because "the delete key and backspace key definitions are reversed."

Differences in keyboard definitions are not the only source of confusion in switching from one computer to another. Each computer network has a different operating system. The Macintosh computers (object-oriented and menu-driven) and PCs (MS-DOS) have the easiest operating systems to learn. The VAX VMS operating system is somewhat more difficult. The Cray and the DGSNET, which are the primary software development systems, use UNIX (but not the same version) and are considerably more challenging. Each additional system that a worker must use increases the amount of training required and therefore the cost. Another problem lies within the use of the same commands for different purposes on different systems. For example, the command "quit" exits and doesn't save changes made when it is used in VAX "news", and the command "exit" leaves "news" but saves changes. On the DGSNET e-mail, the definitions of those commands are reversed. Mixing up these particular commands could lead to a person being forced to redo work. Additionally, switching from network to another, or from a Macintosh to a network, for example, may be further complicated by the fact that in some cases, file transfers between computers is difficult, if not impossible.

Monitors also merit some consideration. Older terminals and most PCs have monitors with 13"-15" screen widths. X-terminals

and some of the Macs have 17" screens or larger. The bigger screens are easier to see. They also permit a full-page view of word processing documents and allow the user to view multiple task windows at the same time. The curvature of the screen on the older monitors is also greater, causing increased distortion, which is exacerbated by screen flicker (Kodak 1, 101-106). Another issue is the size and font of the characters displayed on the screen. These features are adjustable on Macs, and in word processors on some of the other computers. The X-terminal default font is very small and can present a challenge to the eyes of a "fresh" worker. For a person nearing the end of his or her shift, who is likely to be experiencing visual and mental fatigue, the characters displayed on their X-terminal are easily misread and can cause headaches. A large, (12 point) simple font (such as "Geneva" or "Times") would be preferable. A related problem involves the background color and intensity. The objective is to combine a background and screen print that create the most contrast. Black characters on a white background present the most contrast in a workplace with a high amount of illumination. When the ambient lighting is lower, white letters on a black background presents the most contrast. Color backgrounds and print are generally discouraged, but some workers may choose to use color, particularly for the background, because they find a slightly lower level of contrast more pleasing to the eye. As long as there are no indications that job performance is

suffering, the use of personally preferred colors and fonts is acceptable (Kodak 1, 170-174).

Currently there is concern about the potentially hazardous effects of extra low frequency (ELF) electromagnetic radiation that emanates from computer equipment and other office machines. Some workers have expressed anxiety over this topic since they spend the majority of their work-day in front of a computer, or in the same office with one or more computers that are turned on all day. There is no consensus in the scientific community on the validity of these claims. Regardless of whether or not this radiation increases the risk of cancer or other health problems, most people are exposed to more ELF radiation from household appliances, such as electric blankets, irons, and toasters. Therefore, ELF radiation hazards were not considered in this project. However, should more definitive evidence be produced linking ELF radiation of that order of magnitude to health problems, this issue should be revisited.

### 3.2.4 WORK HOURS

The tasks involved in this process are highly complex and therefore place considerable mental and visual demands on the individuals performing them. If the task duration is longer, the person performing the task is more likely to experience visual and/or mental fatigue. Extending the work day provides only a limited benefit because this fatigue decreases productivity (Kodak

2, 222-223). Consequently, the rapid response software modification process has been designed to be completed during normal duty hours: eight hours per day, five days per week. K40/K50 uses flexible scheduling, so employees have a certain amount of latitude in adjusting their schedules to meet personal physiological needs and family commitments. Schedules are allowed to fall within the period of 0600-1800 hours. Each employee is responsible for choosing a schedule that not only meets their personal needs, but also the requirements of their job. It should be noted, that things occasionally go wrong, or take longer than expected. In those instances, some overtime may be necessary to complete the process on schedule. Because of security requirements, it is generally easier to work extended hours than to work extra days.

Even if overtime is necessary, people should be encouraged to take a short break at least every couple of hours. It is preferable that the person walks around for part of this break so that normal circulation will be restored. Other breaks may be included throughout the day to lessen fatigue. Those breaks may involve the performance of other types of tasks, such as reading departmental memos, returning phone calls, filing, and doing paperwork. This should help in the prevention of repetitive motion injuries besides reducing mental fatigue (Kodak 2, 223-226). This is more important in the critical-minimum or critical-intermediate processes, since the urgent-minimum and

urgent-intermediate processes require less than 100% of the work day be spent on these tasks.

### 3.2.5 INFORMATION TRANSFER

The last human factors issue to be considered in the rapid response software modification process is that of information transfer. To facilitate accurate information transfers, each phase of the process begins or ends with a step during which the information is passed. During this step, the information is discussed between the people responsible, to ensure comprehension by both parties. The validation and verification phase, the quality assurance tasks in the documentation phase, and the quality control tasks in the delivery phases are designed to catch any problems created by erroneous or ambiguous information transfers (Kodak 1, 179-190). Also, the software change request form has been designed to be simple and easy to be read (Kodak 1, 164-166). Any problems noted should be pointed out during the process analysis phase.

## 3.3 IMPLEMENTATION

The rapid response software modification process has not yet been presented to K40/K50 management. That must be the first step in the implementation of this process. It is anticipated that 30 days would be required to present the plan and make any changes necessary to gain approval. Following approval, copies of the plan

must be made and distributed throughout the SLBM program at NSWCDD. Training should take 30-60 days. During that period, a presentation should also be made to USSTRATCOM. The liaison may be briefed then as well. It would also be advisable to brief the sponsor organization at some point. Therefore, the approximate amount of time required for implementation of this process is 90 days.

## 3.4 TRAINING

Everyone involved is familiar enough with his or her job that job-specific training is not needed. However, the rapid response software modification process is different from standard procedures to require a small amount of training in the performance of tasks and their specific ramifications with respect to the process. This can be accomplished by a series of brief presentations to groups likely to be involved. These sessions may be more effective if they are given to project teams, so that there are analysts, programmers, QA specialists, etc. in the room simultaneously. This will help to provide a more balanced view of the process. The other approach would be to hold one session for each branch. This method is not preferred because it separates analysts from programmers, and programmers from QA specialists, etc., and creates an "us against them" view of the process. Training sessions should take less than one hour. They would include a 30

minute presentation and 30 minutes to address questions and step through the process with some hypothetical examples. Particular attention should be paid to the request and assessment phases.

In addition to training in the process, it is the opinion of this scientist that managers and supervisors need to be given more training in empowering their people. A general lack of empowerment is prevalent in K40/K50. The impact of this problem to the customer is that if the manager is not available, decisions simply do not get made, and actions are not taken. This forces the liaison to determine who is most likely to take the initiative who will do nothing when a request is made. In order for the rapid response software modification process to succeed, this problem must be corrected.

# 4.0 CONCLUSIONS

Many conclusions can be drawn from this project. The rapid response software modification process should be implemented by the SLBM program. In addition, changes should be made to equipment, computer systems, the work environment, the way that the SLBM program does business, and ultimately, in the rapid response software modification process itself.

## 4.1 HUMAN FACTORS

Transmission of data between participants in the process should be done electronically. This saves time that would be spent waiting for printouts and delivering documents to other offices, and it saves the cost of the paper and its disposal later. It also simplifies the security classification issues, because it is easier to handle electronic data than paper copies, and the risk of compromise is much smaller.

Electronic transmission would be easier if the computer systems were more compatible. It would probably take two or three years, but I think that everyone's computer (regardless of what type it is) should be connected to DGSNET. They should be able to multitask, both between networks, and between DGSNET and their PC or Macintosh. This capability is available in a limited way now. In a perfect world, everyone would do work on the same network with the same equipment. But given the mission requirements of the SLBM program and the capabilities currently available on each system, this would be neither feasible nor desirable.

Computer equipment should be updated. The ancient VT220 terminals must be replaced with X-terminals or PCs. Old PCs and Macs with limited capabilities (particularly the monitors) should also be replaced at a rate permitted by the budget with more human-friendly equipment. A standard keyboard should be adopted where

possible (An example of when this would not be possible currently is the Macintosh keyboard, which must have a different design due to its operating system.). Conflicts between command definitions on different networks should be resolved. One possibility would be to choose one operating system's command set, and "alias" it on all of the others. That way the user could learn one set of commands that would work on all systems. Standardizing both the equipment and the network would decrease training time and increase maintainability. However, the improvement in hardware that is needed most is correcting the reliability problems of the DGSNET servers, which seem to crash at least twice per week.

As far as the work environment goes, minor changes to office equipment and changes to environmental controls should be made. The office furniture provided is adequate for the needs of the mission. However, wrist rests and foot rests should be provided to every employee, plus every common work area, to reduce fatigue due to poor circulation and muscle soreness, and to reduce the risk of injury caused by static loading and repetitive motion. Tables containing proper chair, terminal table, keyboard, and monitor adjustments based on anthropometric data should be provided to all employees. The task of measuring themselves and making the adjustments can be left to them. Polarizing filters or anti-glare coating should be put on all monitors. This would reduce eye strain and data entry errors. This could further be accomplished by

reducing the amount of ambient light in work spaces, either by requiring people to turn one of the light switches off, or by removing half of the fluorescent tubes. The second option would be more effective. Both options would also reduce electricity costs.

## 4.2 ORGANIZATIONAL CULTURE

It is noted that the rapid response software modification process represents a significant departure from current practices in that most of the decisions are being made at the lowest level of the organization. Managers should empower their people to make these types of decisions and take certain calculated risks. Eventually, the decision and response phases should also be turned over to front-line employees, perhaps after a transitional period in which group leaders (first-line supervisors) or project leader-coordinators (PLCs) perform those functions. Management involvement would then only be required to assign personnel to the process and to review the actions after the process has been completed. Even the assignments should be automatic most of the time. I suggest an initial trial period of six months, during which the process should be implemented as designed. The transitional period should be an additional six months. Therefore, after one year, the process can be turned over to work teams.

The standards that exist for documentation and code should be enforced more rigorously than is currently done. Additional

standards are in the process of being draft, but even those initiatives don't cover everything. The more that documents, code structure, terminology, equipment, and procedures can be standardized, the more maintainable the system will be. This is especially important for people who have responsibilities on several projects. What should be done, for example, if more than one software modification request is received and their schedules overlap, if both of the affected programs are maintained by the same people? What if the analyst responsible for that software is on leave and cannot be reached? It is obvious that someone else must be able to pick up that project and immediately be able to figure out what needs to be done in order to make the requested change. Standardization also makes it easier to assign the non-traditional analysts (non-degreed professionals and scientists and engineers from other areas of expertise) to these projects. It also reduces the frequency of errors caused by the non-traditional analysts. Since more of the workload could be picked up by these people, the workers who normally perform those tasks would experience a reduction in their workloads, which should make them more effective and decrease the rate at which they make errors caused by distractions and general fatigue.

Flexible work schedules present a minor problem in this process. In order to accomplish certain tasks, the people assigned to the process must work together, meaning not only cooperate, but also

working at the same time. This problem can easily be solved by having the participants agree to a common work schedule for the duration of the process. Any deviations from that schedule would have to be coordinated in advance.

Since time is a scarce commodity, the temptation exists to cut corners. One part of the rapid response softeware modification process that is particularly vulnerable to this temptation is the process analysis phase. This part of the process should never be skipped or delayed. Participants must take the time to perform the process analysis thoroughly, rather than simply writing the required report. Also, although participants are expected to provide the PLC with whatever information is necessary to update the project metrics and the project software development folder, no metrics are specified. The software development process in OD 61596 recommends several metrics and references another publication that contains additional metrics. However, it allows the PLC to choose which metrics are used on the project. It is the opinion of this researcher that a set of core metrics should be defined and subsequently required for all projects. The use of any additional metrics may remain optional. This core group should contain at a minimum failure rate and reliability metrics (both for hardware and software), task duration, level of effort, work-hours used, number and type of human resources, and cost. The metrics should be descriptive enough to permit identification of points of failure and

areas for improvement. These metrics can easily be collected using project management software.

## 4.3 PROCESS IMPROVEMENT

If the changes described in sections 4.1 and 4.2 are implemented, several process improvements could be made. Once software reliability is at 98%, some of the quality control functions (particularly those tasks that involve verifying that the version used in a given step of the process matches the version from the previous step) can be eliminated. This would further reduce costs and human resource requirements. When decision making is completely in the hands of the work teams, the decision phase and the response phase can be collapsed into the assessment phase. The datalink or integrated production facility (IPF) could be used in the URGMIN and URGINTER paths in place of shipping, reducing the delivery phase duration for those paths by at least five days, and eliminating the need for a media prep specialist. The process analysis phase will take less time as people become familiar with the process.

I think that the combination of all of these changes would reduce the software failure rate by at least 50%. Assigning tasks to less costly human resources, standardization, automation, and stremlining the process should reduce work-hours by at least 25% and costs by at least 30%, assuming that cost components associated with base-wide expenses do not rise dramatically.

Implementation time and total process duration could be reduced by 25% to 50%.

## 4.4 FUTURE PLANS

After the rapid response software modification process has been used a few times and appropriate statistics have been collected, parts of this project should be repeated. The purpose of doing the work again would be to identify task durations, human resource requirements, and costs that are more realistic. Hardware should be included to a greater degree. Reliability issues should be addressed more definitively.

This process should be expanded. A rapid response software development process shoul evolve out of the modification process. The expanded process could hopefully have a duration of less than three months. It would provide a swift method of dealing with the larger requests and requests for new items that are beyond the scope of the modification process. This could be accomplished by determining what steps are missing from the modification process that would apply to a larger effort. It could also be done by examining the current software development process and removing any steps that add little or no value.

## REFERENCES


Allen, Shirley, DATCHK Formulation. SIOP Revision 96, NSWCDD, K42, 22 August 1994.

Blanchard, Benjamin S., and Wolter J. Fabrycky. Systems Engineering and Analysis. 2nd ed. Englewood Cliffs, NJ: Prentice Hall, 1990.

Eastman Kodak Company. Ergonomic Desing for People at Work. 2 Vols. New York: Van Nostrand Reinhold, 1983.

Henderson, John, DATCHK Design. SIOP Revision 96, NSWCDD, K51, 26 August 1994.

NAVSWCINST 5510.1B, Information Security Manual. FOUO, NSWCDD, 8 March 1991.

NAVSEA OD 55688, NSWC K40/K50Configuration Management Master Plan. Revision A, NSWC, K52, 1 July 1987.

OPNAVINST 5510.1H, Information and Personnel Security Program Regulation. FOUO, Department of the Navy, 24 August 1990.

"Procedures for Supporting Software Deliveries for K53 Fleet Support." Revision 1, NSWCDD, K53 memorandum, 9 August 1994.

SSP OD 61561, SLBM Security Handbook. FOUO, NSWCDD, 1 April 1993.

SSP OD 61596, NSWCDD SLBM Software Development Process. Revision A, NSWCDD, SEPIG, 1 August 1994.

SSP OD 61711, SLBM Integrated Planning System (SIPS) Software Development Plan. Revision B (Draft), NSWCDD, K41, 1 December 1994.

SSP OD 61805, SUBFOOT/NSDL Data Check (DATCHK) User Guide. Revision B, NSWCDD, K42, 1 February 1995.

SSP OD 61868, SLBM Integrated Planning System (SIPS) Patrol Area Analysis. Revision A, Vol. 3, NSWCDD, K52, 1 August 1994.

USSTRATCOM / SSP Force Planning Siftware Development Plan. (Draft), NSWCDD, K42, April 1994.

# APPENDIX A: CRITMIN PROCESS

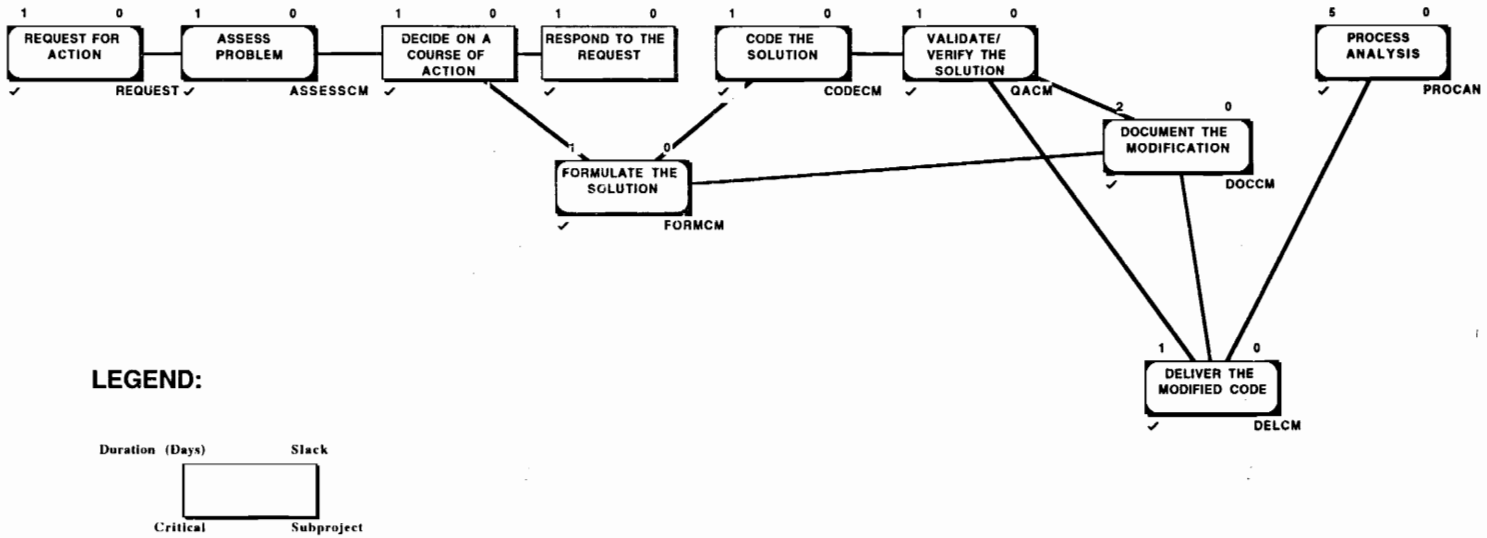



Figure A1: CRITMIN Process

CRITMIN
4/7/95
Project Overview

| | |
|---|---|
| • Tasks and Activities | |
| Tasks | 2 |
| Milestones | 0 |
| Supertasks | 8 |
| Completed Tasks | 0 |
| Completed Milestones | 0 |
| Summary Activities | 0 |
| • Project Dates | |
| Project Start Date | 7/1/95 |
| Project Finish Date | 7/20/95 |
| Actual Start | 7/3/95 |
| Actual Finish | 7/20/95 |
| • Project Costs | |
| Total Planned Cost | 31,474.67 |
| Total Planned Income | 0.00 |
| Total Actual Cost | 0.00 |
| Total Actual Income | 0.00 |
| • Other | |
| Projects in Family Tree | 13 |
| Projects in Resource Scope | 0 |
| Project % Done | 0 |
| Number of Resources | 8 |
| Number of Calendars | 1 |

Figure A2: CRITMIN Project Overview

| Name | Subproject | # Resources | Duration (Days) | Slack | Total Cost |
|---|---|---|---|---|---|
| REQUEST FOR ACTION | REQUEST | 4 | 1 | 0 | 1,725.75 |
| ASSESS PROBLEM | ASSESSCM | 3 | 1 | 0 | 2,451.78 |
| DECIDE ON A COURSE OF ACTION | | 3 | 1 | 0 | 505.92 |
| RESPOND TO THE REQUEST | | 1 | 1 | 0 | 168.64 |
| FORMULATE THE SOLUTION | FORMCM | 1 | 1 | 0 | 1,033.28 |
| CODE THE SOLUTION | CODECM | 1 | 1 | 0 | 1,033.28 |
| VALIDATE/ VERIFY THE SOLUTION | QACM | 1 | 1 | 0 | 2,599.35 |
| DOCUMENT THE MODIFICATION | DOCCM | 2 | 2 | 0 | 3,616.48 |
| DELIVER THE MODIFIED CODE | DELCM | 5 | 1 | 0 | 1,097.34 |
| PROCESS ANALYSIS | PROCAN | 3 | 5 | 0 | 17,242.86 |

Figure A3: CRITMIN Process Total Resource Allocation

| Name | Resource | Work-Days | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REQUEST FOR ACTION | LIAISON | .50 | 50 | 1 | 258.32 |
| REQUEST FOR ACTION | K42 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| REQUEST FOR ACTION | K51 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| REQUEST FOR ACTION | K52 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| ASSESS PROBLEM | ANALYST | 1 | 100 | 1 | 516.64 |
| ASSESS PROBLEM | PROGRAMMER | 1 | 100 | 1 | 516.64 |
| ASSESS PROBLEM | LIAISON | .12 | 100 | .12 | 64.58 |
| DECIDE ON A COURSE OF ACTION | K42 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| DECIDE ON A COURSE OF ACTION | K51 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| DECIDE ON A COURSE OF ACTION | K52 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| RESPOND TO THE REQUEST | K42 BRANCH HEAD | .25 | 50 | .50 | 168.64 |
| FORMULATE THE SOLUTION | ANALYST | 1 | 100 | 1 | 516.64 |
| CODE THE SOLUTION | PROGRAMMER | 1 | 100 | 1 | 516.64 |
| VALIDATE/ VERIFY THE SOLUTION | QA SPECIALIST | 1 | 100 | 1 | 516.64 |
| DOCUMENT THE MODIFICATION | ANALYST | 2 | 100 | 2 | 1,033.28 |
| DOCUMENT THE MODIFICATION | QA SPECIALIST | 1 | 100 | 1 | 516.64 |
| DELIVER THE MODIFIED CODE | ANALYST | .19 | 100 | .19 | 96.87 |
| DELIVER THE MODIFIED CODE | QA SPECIALIST | .12 | 100 | .12 | 64.58 |
| DELIVER THE MODIFIED CODE | QC SPECIALIST | .44 | 100 | .44 | 193.48 |
| DELIVER THE MODIFIED CODE | PROGRAMMER | .12 | 100 | .12 | 64.58 |
| DELIVER THE MODIFIED CODE | LIAISON | .25 | 100 | .25 | 129.16 |
| PROCESS ANALYSIS | ANALYST | 5 | 100 | 5 | 2,583.20 |
| PROCESS ANALYSIS | PROGRAMMER | 5 | 100 | 5 | 2,583.20 |
| PROCESS ANALYSIS | QA SPECIALIST | 5 | 100 | 5 | 2,583.20 |

Figure A4: CRITMIN Individual Resource Allocation

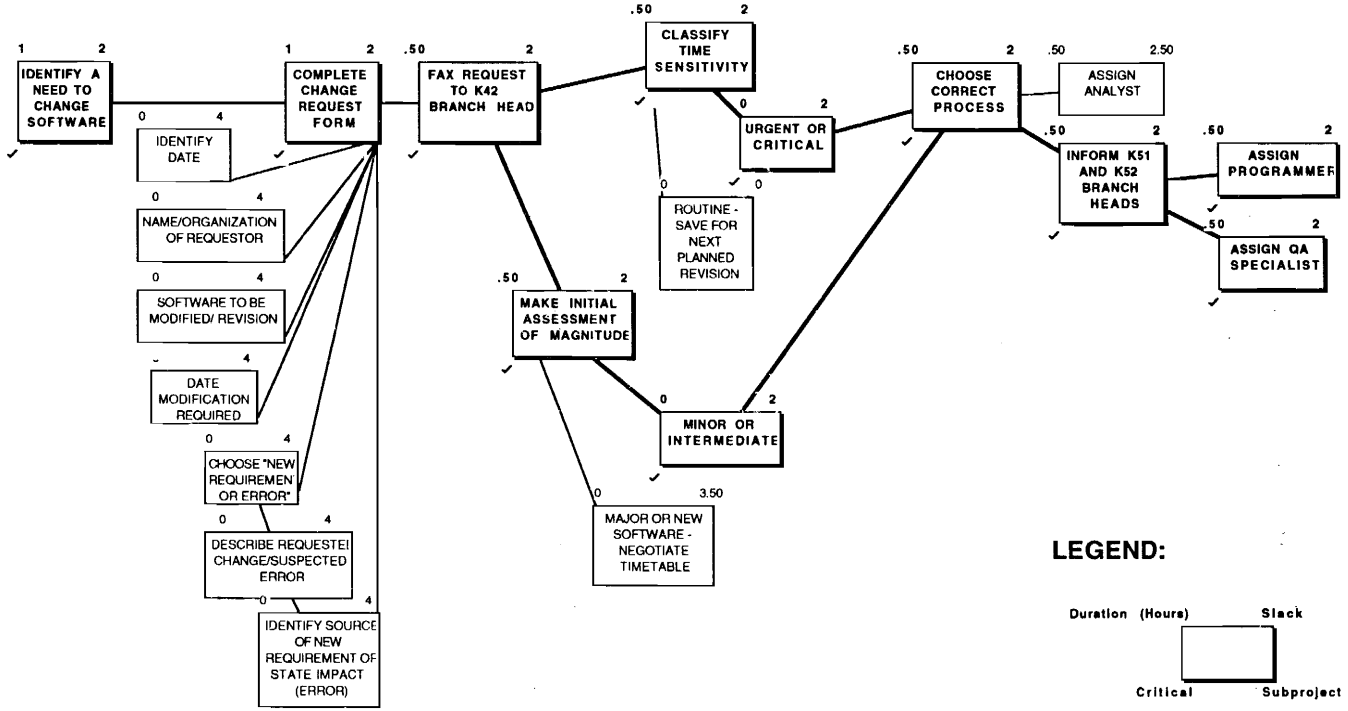



Figure A5: CRITMIN Request Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| IDENTIFY A NEED TO CHANGE SOFTWARE | LIAISON | 1 | 100 | 1 | 64.58 |
| COMPLETE CHANGE REQUEST FORM | LIAISON | 1 | 100 | 1 | 64.58 |
| FAX REQUEST TO K42 BRANCH HEAD | LIAISON | .50 | 100 | .50 | 32.29 |
| CLASSIFY TIME SENSITIVITY | K42 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| MAKE INITIAL ASSESSMENT OF MAGNITUDE | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| CHOOSE CORRECT PROCESS | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| ASSIGN ANALYST | K42 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| INFORM K51 AND K52 BRANCH HEADS | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| ASSIGN PROGRAMMER | K51 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| ASSIGN QA SPECIALIST | K52 BRANCH HEAD | .50 | 100 | .50 | 42.16 |

Figure A6: CRITMIN Request Phase Resource Allocation

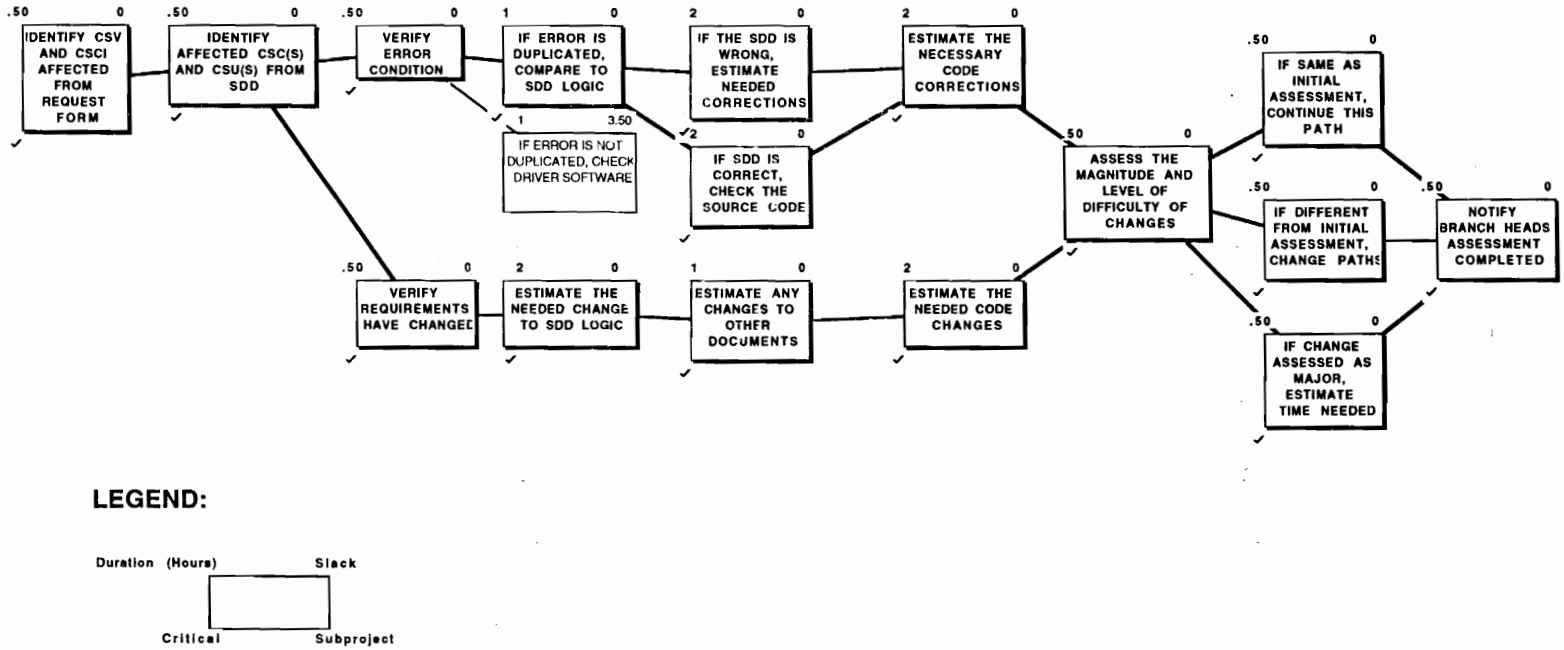



Figure A7: CRITMIN Assessment Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| IDENTIFY CSV AND CSCI AFFECTED FROM REQUEST FORM | ANALYST | .50 | 100 | .50 | 32.29 |
| IDENTIFY CSV AND CSCI AFFECTED FROM REQUEST FORM | PROGRAMMER | .50 | 100 | .50 | 32.29 |
| IDENTIFY AFFECTED CSC(S) AND CSU(S) FROM SDD | ANALYST | .50 | 100 | .50 | 32.04 |
| IDENTIFY AFFECTED CSC(S) AND CSU(S) FROM SDD | PROGRAMMER | .50 | 100 | .50 | 32.04 |
| VERIFY ERROR CONDITION BY RUNNING A TEST CASE | ANALYST | .50 | 100 | .50 | 32.29 |
| IF ERROR IS NOT DUPLICATED, CHECK DRIVER SOFTWARE | LIAISON | 1 | 100 | 1 | 64.58 |
| IF ERROR IS DUPLICATED, COMPARE TO SDD LOGIC | ANALYST | 1 | 100 | 1 | 64.58 |
| IF THE SDD IS WRONG, ESTIMATE NEEDED CORRECTIONS | ANALYST | 2 | 100 | 2 | 129.16 |
| IF SDD IS CORRECT, CHECK THE SOURCE CODE | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| ESTIMATE THE NECESSARY CODE CORRECTIONS | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| VERIFY REQUIREMENTS HAVE CHANGED | ANALYST | .50 | 100 | .50 | 32.29 |
| ESTIMATE THE NEEDED CHANGE TO SDD LOGIC | ANALYST | 2 | 100 | 2 | 129.16 |
| ESTIMATE ANY CHANGES TO OTHER DOCUMENTS | ANALYST | 1 | 100 | 1 | 64.58 |
| ESTIMATE THE NEEDED CODE CHANGES | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| ASSESS THE MAGNITUDE AND LEVEL OF DIFFICULTY OF CHANGES | ANALYST | .50 | 100 | .50 | 32.04 |
| ASSESS THE MAGNITUDE AND LEVEL OF DIFFICULTY OF CHANGES | PROGRAMMER | .50 | 100 | .50 | 32.04 |
| IF SAME AS INITIAL ASSESSMENT, CONTINUE THIS PATH | ANALYST | .50 | 100 | .50 | 32.29 |
| IF SAME AS INITIAL ASSESSMENT, CONTINUE THIS PATH | PROGRAMMER | .50 | 100 | .50 | 32.04 |
| IF DIFFERENT FROM INITIAL ASSESSMENT, CHANGE PATHS | ANALYST | .50 | 100 | .50 | 32.04 |
| IF DIFFERENT FROM INITIAL ASSESSMENT, CHANGE PATHS | PROGRAMMER | .50 | 100 | .50 | 32.04 |
| IF CHANGE ASSESSED AS MAJOR, ESTIMATE TIME NEEDED | ANALYST | .50 | 100 | .50 | 32.04 |
| IF CHANGE ASSESSED AS MAJOR, ESTIMATE TIME NEEDED | PROGRAMMER | .50 | 100 | .50 | 32.04 |
| NOTIFY BRANCH HEADS ASSESSMENT COMPLETED | ANALYST | .50 | 100 | .50 | 32.29 |
| NOTIFY BRANCH HEADS ASSESSMENT COMPLETED | PROGRAMMER | .50 | 100 | .50 | 32.29 |



Figure A8: CRITMIN Assessment Phase Resource Allocation

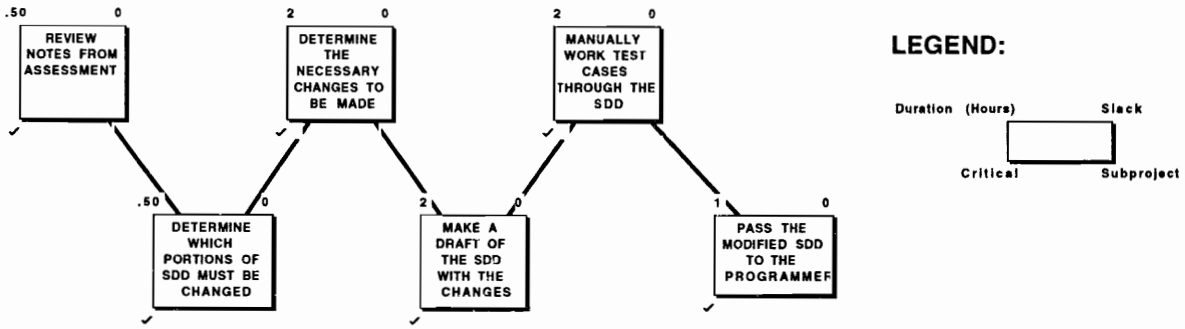



Figure A9: CRITMIN Formulation Phase



| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REVIEW NOTES FROM ASSESSMENT | ANALYST | .50 | 100 | .50 | 32.29 |
| DETERMINE WHICH PORTIONS OF SDD MUST BE CHANGED | ANALYST | .50 | 100 | .50 | 32.29 |
| DETERMINE THE NECESSARY CHANGES TO BE MADE | ANALYST | 2 | 100 | 2 | 129.16 |
| MAKE A DRAFT OF THE SDD WITH THE CHANGES | ANALYST | 2 | 100 | 2 | 129.16 |
| MANUALLY WORK TEST CASES THROUGH THE SDD | ANALYST | 2 | 100 | 2 | 129.16 |
| PASS THE MODIFIED SDD TO THE PROGRAMMER | ANALYST | 1 | 100 | 1 | 64.58 |

Figure A10: CRITMIN Formulation Phase Resource Allocation

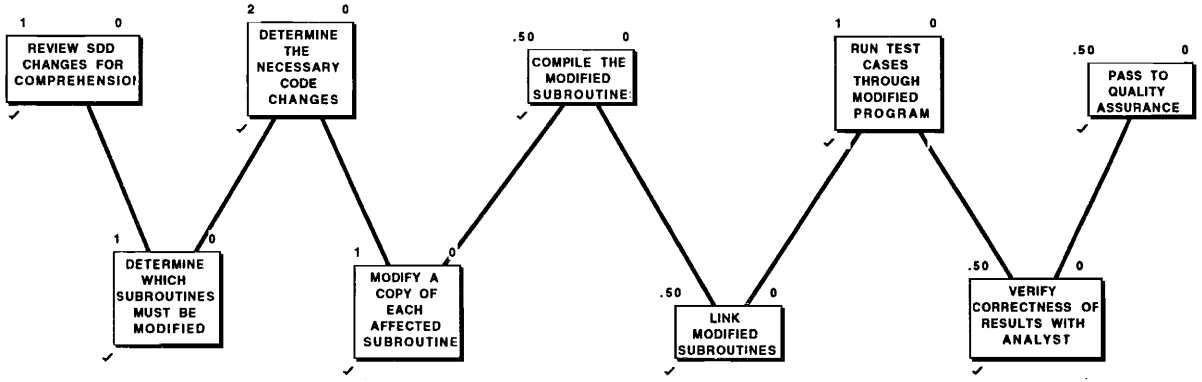



Figure A11: CRITMIN Coding Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REVIEW SDD CHANGES FOR COMPREHENSION | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| DETERMINE WHICH SUBROUTINES MUST BE MODIFIED | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| DETERMINE THE NECESSARY CODE CHANGES | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| MODIFY A COPY OF EACH AFFECTED SUBROUTINE | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| COMPILE THE MODIFIED SUBROUTINES | PROGRAMMER | .50 | 100 | .50 | 32.29 |
| LINK MODIFIED SUBROUTINES WITH COPY OF REST OF CODE | PROGRAMMER | .50 | 100 | .50 | 32.29 |
| RUN TEST CASES THROUGH MODIFIED PROGRAM | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| VERIFY CORRECTNESS OF RESULTS WITH ANALYST | PROGRAMMER | .50 | 100 | .50 | 32.29 |
| PASS TO QUALITY ASSURANCE SPECIALIST | PROGRAMMER | .50 | 100 | .50 | 32.29 |

Figure A12: CRITMIN Coding Phase Resource Allocation

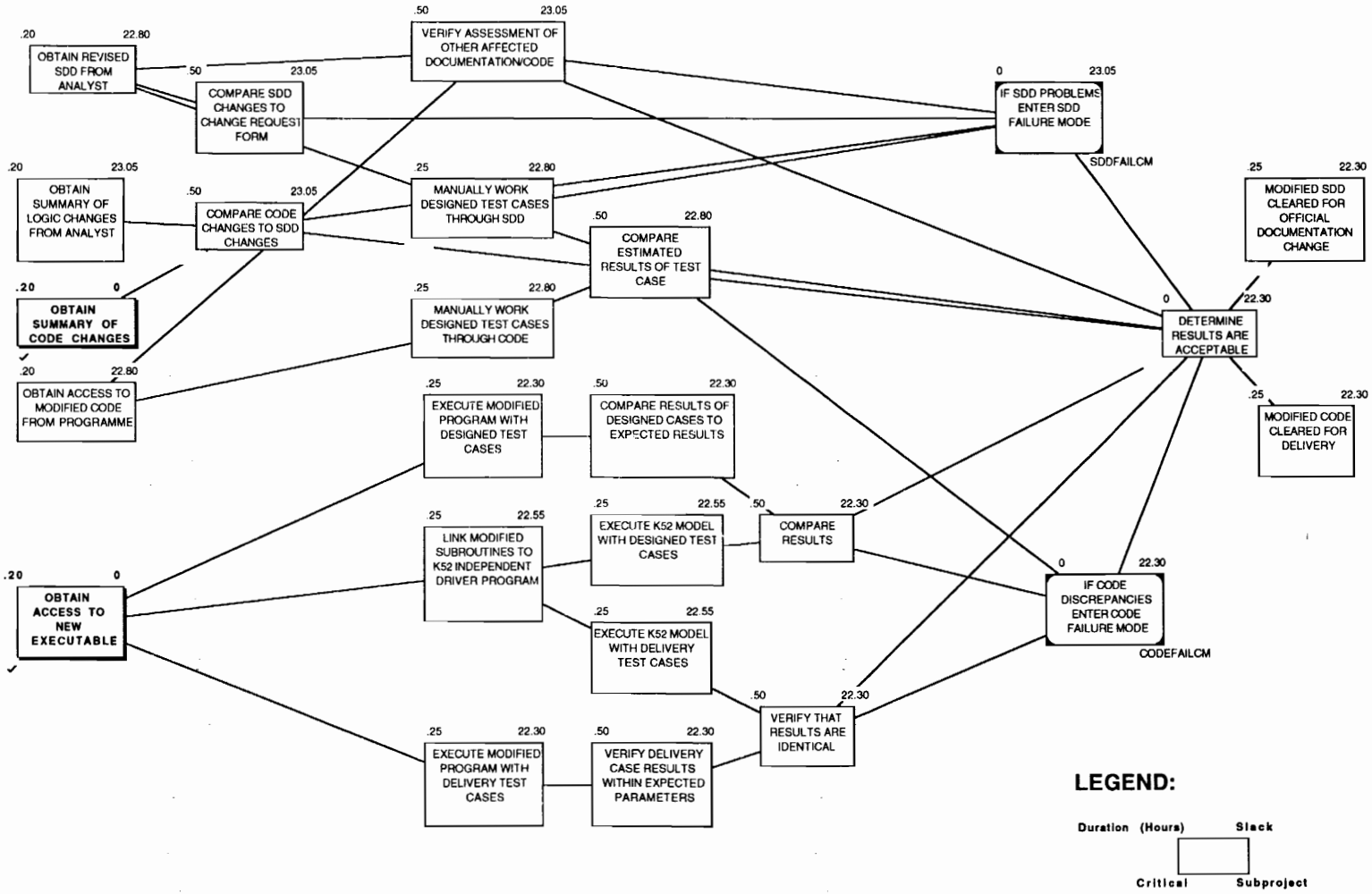



Figure A13: CRITMIN Validation and Verification Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| OBTAIN REVISED SDD FROM ANALYST | QA SPECIALIST | .20 | 100 | .20 | 12.92 |
| OBTAIN SUMMARY OF LOGIC CHANGES FROM ANALYST | QA SPECIALIST | .20 | 100 | .20 | 12.92 |
| OBTAIN ACCESS TO MODIFIED CODE FROM PROGRAMMER | QA SPECIALIST | .20 | 100 | .20 | 12.92 |
| **OBTAIN ACCESS TO NEW EXECUTABLE** | **QA SPECIALIST** | **.20** | **100** | **.20** | **12.92** |
| **OBTAIN SUMMARY OF CODE CHANGES** | **QA SPECIALIST** | **.20** | **100** | **.20** | **12.92** |
| COMPARE SDD CHANGES TO CHANGE REQUEST FORM | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| COMPARE CODE CHANGES TO SDD CHANGES | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| VERIFY ASSESSMENT OF OTHER AFFECTED DOCUMENTATION/CODE | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| MANUALLY WORK DESIGNED TEST CASES THROUGH SDD | QA SPECIALIST | .25 | 100 | .25 | 16.15 |
| MANUALLY WORK DESIGNED TEST CASES THROUGH CODE | QA SPECIALIST | .25 | 100 | .25 | 16.15 |
| COMPARE ESTIMATED RESULTS OF TEST CASE | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| **EXECUTE MODIFIED PROGRAM WITH DESIGNED TEST CASES** | **QA SPECIALIST** | **.25** | **100** | **.25** | **16.15** |
| **COMPARE RESULTS OF DESIGNED CASES TO EXPECTED RESULTS** | **QA SPECIALIST** | **.50** | **100** | **.50** | **32.29** |
| **EXECUTE MODIFIED PROGRAM WITH DELIVERY TEST CASES** | **QA SPECIALIST** | **.25** | **100** | **.25** | **16.15** |
| **VERIFY DELIVERY CASE RESULTS WITHIN EXPECTED PARAMETERS** | **QA SPECIALIST** | **.50** | **100** | **.50** | **32.29** |
| LINK MODIFIED SUBROUTINES TO K52 INDEPENDENT DRIVER PROGRAM | QA SPECIALIST | .25 | 100 | .25 | 16.15 |
| EXECUTE K52 MODEL WITH DESIGNED TEST CASES | QA SPECIALIST | .25 | 100 | .25 | 16.15 |
| **COMPARE RESULTS** | **QA SPECIALIST** | **.50** | **100** | **.50** | **32.29** |
| EXECUTE K52 MODEL WITH DELIVERY TEST CASES | QA SPECIALIST | .25 | 100 | .25 | 16.15 |
| **VERIFY THAT RESULTS ARE IDENTICAL** | **QA SPECIALIST** | **.50** | **100** | **.50** | **32.29** |
| **MODIFIED CODE CLEARED FOR DELIVERY** | **QA SPECIALIST** | **.25** | **100** | **.25** | **16.15** |
| **MODIFIED SDD CLEARED FOR OFFICIAL DOCUMENTATION CHANGE** | **QA SPECIALIST** | **.25** | **100** | **.25** | **16.15** |



Figure A14: CRITMIN Validation and Verification Phase Resource Allocation

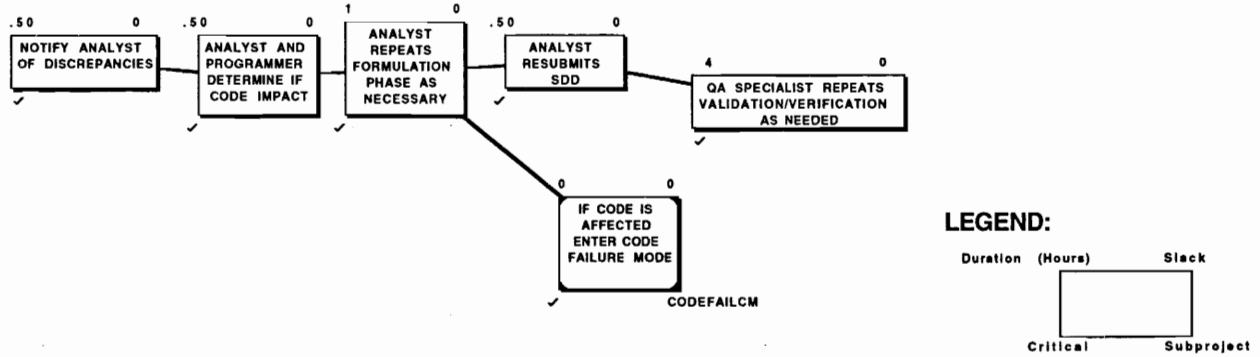



Figure A15: CRITMIN SDD Code Failure Mode

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| NOTIFY ANALYST OF DISCREPANCIES | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| ANALYST AND PROGRAMMER DETERMINE IF CODE IMPACT | ANALYST | .50 | 100 | .50 | 32.29 |
| ANALYST AND PROGRAMMER DETERMINE IF CODE IMPACT | PROGRAMMER | .50 | 100 | .50 | 32.29 |
| ANALYST REPEATS FORMULATION PHASE AS NECESSARY | ANALYST | 1 | 100 | 1 | 64.58 |
| ANALYST RESUBMITS SDD | ANALYST | .50 | 100 | .50 | 32.29 |
| QA SPECIALIST REPEATS VALIDATION/VERIFICATION AS NEEDED | QA SPECIALIST | 4 | 100 | 4 | 258.32 |

Figure A16: CRITMIN SDD Failure Mode Resource Allocation

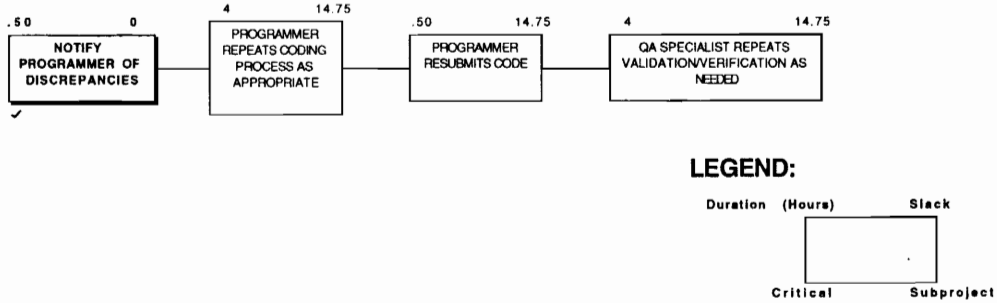



Figure A17: CRITMIN Code Failure Mode

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| NOTIFY PROGRAMMER OF DISCREPANCIES | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| PROGRAMMER REPEATS CODING PROCESS AS APPROPRIATE | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| PROGRAMMER RESUBMITS CODE | PROGRAMMER | .50 | 100 | .50 | 32.29 |
| QA SPECIALIST REPEATS VALIDATION/VERIFICATION AS NEEDED | QA SPECIALIST | 4 | 100 | 4 | 258.32 |

Figure A18: CRITMIN Code Failure Mode Resource Allocation

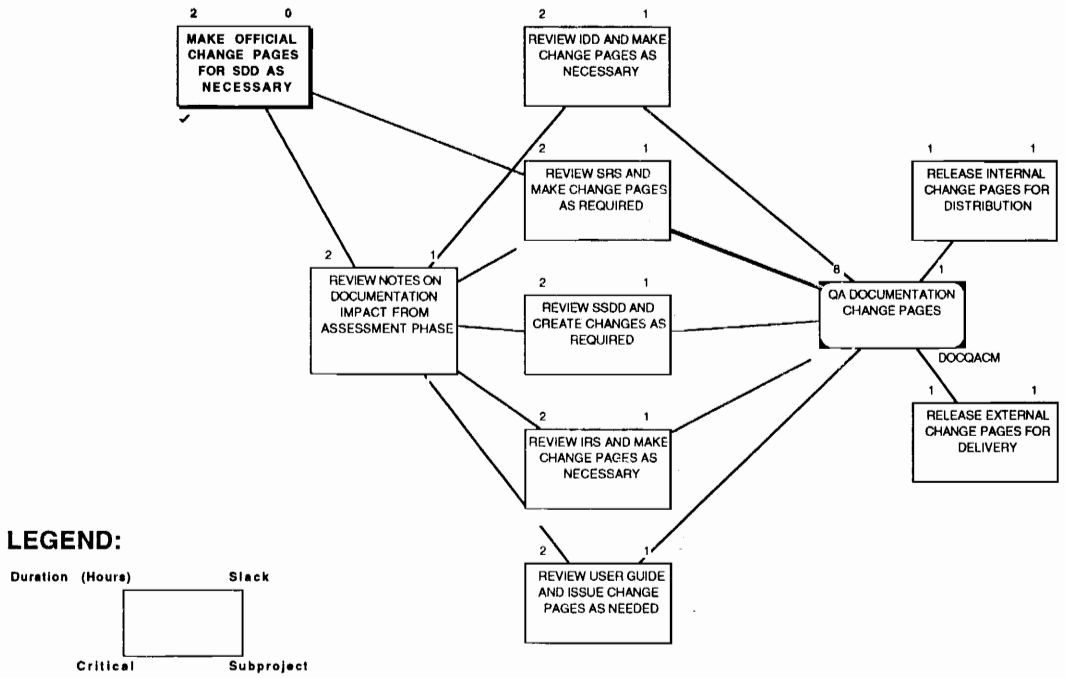



Figure A19: CRITMIN Documentation Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| MAKE OFFICIAL CHANGE PAGES FOR SDD AS NECESSARY | ANALYST | 2 | 100 | 2 | 129.16 |
| REVIEW NOTES ON DOCUMENTATION IMPACT FROM ASSESSMENT | ANALYST | 2 | 100 | 2 | 129.16 |
| REVIEW IDD AND MAKE CHANGE PAGES AS NECESSARY | ANALYST | 2 | 100 | 2 | 129.16 |
| REVIEW SRS AND MAKE CHANGE PAGES AS REQUIRED | ANALYST | 2 | 100 | 2 | 129.16 |
| REVIEW SSDD AND CREATE CHANGES AS REQUIRED | ANALYST | 2 | 100 | 2 | 129.16 |
| REVIEW IRS AND MAKE CHANGE PAGES AS NECESSARY | ANALYST | 2 | 100 | 2 | 129.16 |
| REVIEW USER GUIDE AND ISSUE CHANGE PAGES AS NEEDED | ANALYST | 2 | 100 | 2 | 129.16 |
| QA DOCUMENTATION CHANGE PAGES | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| RELEASE INTERNAL CHANGE PAGES FOR DISTRIBUTION | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| RELEASE EXTERNAL CHANGE PAGES FOR DELIVERY | QA SPECIALIST | 1 | 100 | 1 | 64.58 |

Figure A20: CRITMIN Documentation Phase Resource Allocation

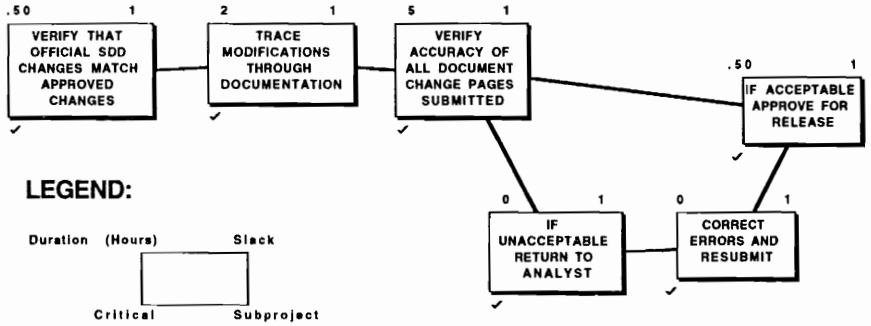



Figure A21: CRITMIN Documentation Quality Assurance Mode

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| VERIFY THAT OFFICIAL SDD CHANGES MATCH APPROVED CHANGES | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| TRACE MODIFICATIONS THROUGH DOCUMENTATION | QA SPECIALIST | 2 | 100 | 2 | 129.16 |
| VERIFY ACCURACY OF ALL DOCUMENT CHANGE PAGES SUBMITTED | QA SPECIALIST | 5 | 100 | 5 | 322.90 |
| IF UNACCEPTABLE RETURN TO ANALYST | ANALYST | 0 | 100 | 0 | 0.00 |
| CORRECT ERRORS AND RESUBMIT | ANALYST | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE APPROVE FOR RELEASE | QA SPECIALIST | .50 | 100 | .50 | 32.29 |

Figure A22: CRITMIN Documentation Quality Assurance Mode Resource Allocation

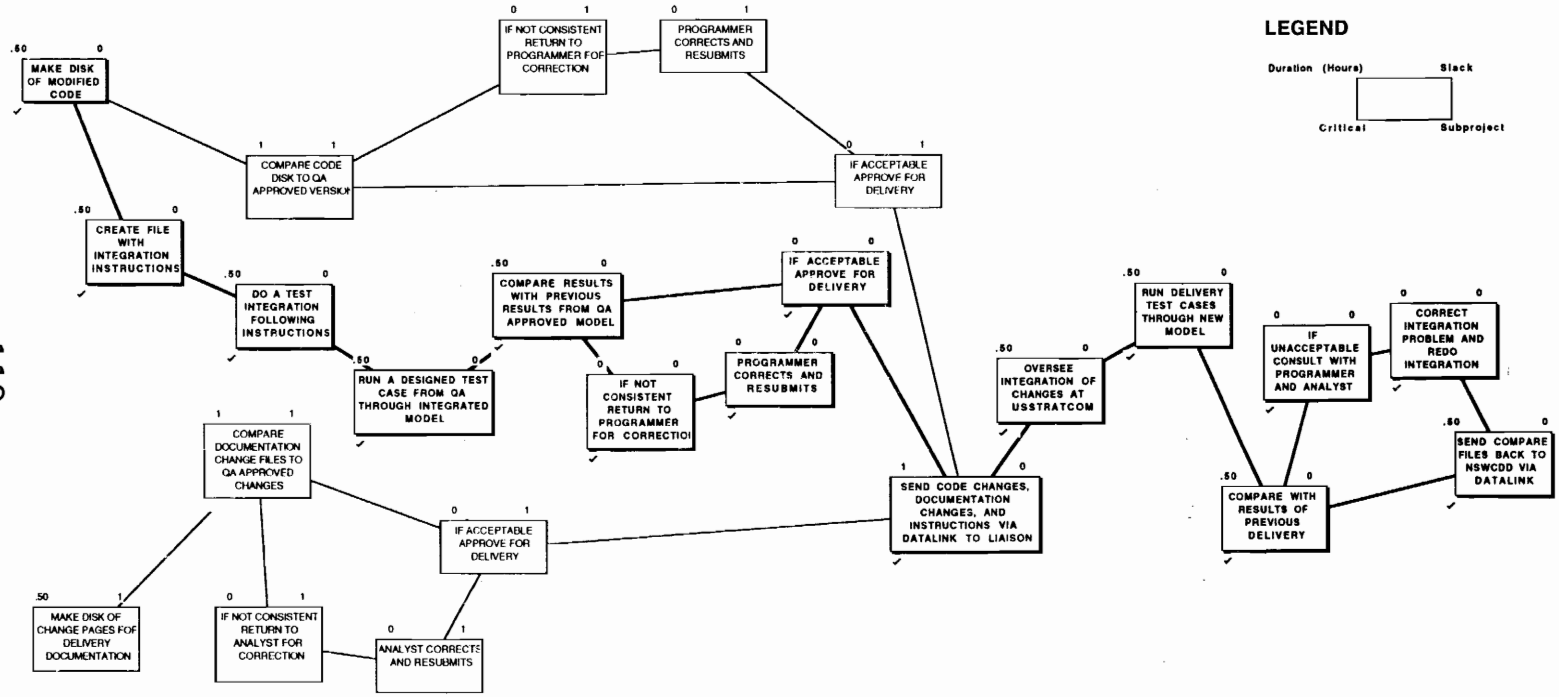



Figure A23: CRITMIN Delivery Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| MAKE DISK OF MODIFIED CODE | PROGRAMMER | .50 | 100 | .50 | 32.29 |
| MAKE DISK OF CHANGE PAGES FOR DELIVERY DOCUMENTATION | ANALYST | .50 | 100 | .50 | 32.29 |
| CREATE FILE WITH INTEGRATION INSTRUCTIONS | PROGRAMMER | .50 | 100 | .50 | 32.29 |
| COMPARE CODE DISK TO QA APPROVED VERSION | QC SPECIALIST | 1 | 100 | 1 | 55.28 |
| IF NOT CONSISTENT RETURN TO PROGRAMMER FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| PROGRAMMER CORRECTS AND RESUBMITS | PROGRAMMER | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE APPROVE FOR DELIVERY | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| COMPARE DOCUMENTATION CHANGE FILES TO QA APPROVED CHANGES | QC SPECIALIST | 1 | 100 | 1 | 55.28 |
| IF NOT CONSISTENT RETURN TO ANALYST FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| ANALYST CORRECTS AND RESUBMITS | ANALYST | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE APPROVE FOR DELIVERY | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| DO A TEST INTEGRATION FOLLOWING INSTRUCTIONS | QC SPECIALIST | .50 | 100 | .50 | 27.64 |
| RUN A DESIGNED TEST CASE FROM QA THROUGH INTEGRATED MODEL | QC SPECIALIST | .50 | 100 | .50 | 27.64 |
| COMPARE RESULTS WITH PREVIOUS RESULTS FROM QA APPROVED MODEL | QC SPECIALIST | .50 | 100 | .50 | 27.64 |
| IF NOT CONSISTENT RETURN TO PROGRAMMER FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| PROGRAMMER CORRECTS AND RESUBMITS | PROGRAMMER | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE APPROVE FOR DELIVERY | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| SEND CODE CHANGES, DOCUMENTATION CHANGES, AND INSTRUCTIONS VIA DATA | ANALYST | 1 | 100 | 1 | 64.58 |
| SEND CODE CHANGES, DOCUMENTATION CHANGES, AND INSTRUCTIONS VIA DATA | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| OVERSEE INTEGRATION OF CHANGES AT USSTRATCOM | LIAISON | .50 | 100 | .50 | 32.29 |
| RUN DELIVERY TEST CASES THROUGH NEW MODEL | LIAISON | .50 | 100 | .50 | 32.29 |
| COMPARE WITH RESULTS OF PREVIOUS DELIVERY | LIAISON | .50 | 100 | .50 | 32.29 |
| IF UNACCEPTABLE CONSULT WITH PROGRAMMER AND ANALYST | LIAISON | 0 | 100 | 0 | 0.00 |
| CORRECT INTEGRATION PROBLEM AND REDO INTEGRATION | LIAISON | 0 | 100 | 0 | 0.00 |
| SEND COMPARE FILES BACK TO NSWCDD VIA DATALINK | LIAISON | .50 | 100 | .50 | 32.29 |



Figure A24: CRITMIN Delivery Phase Resource Allocation

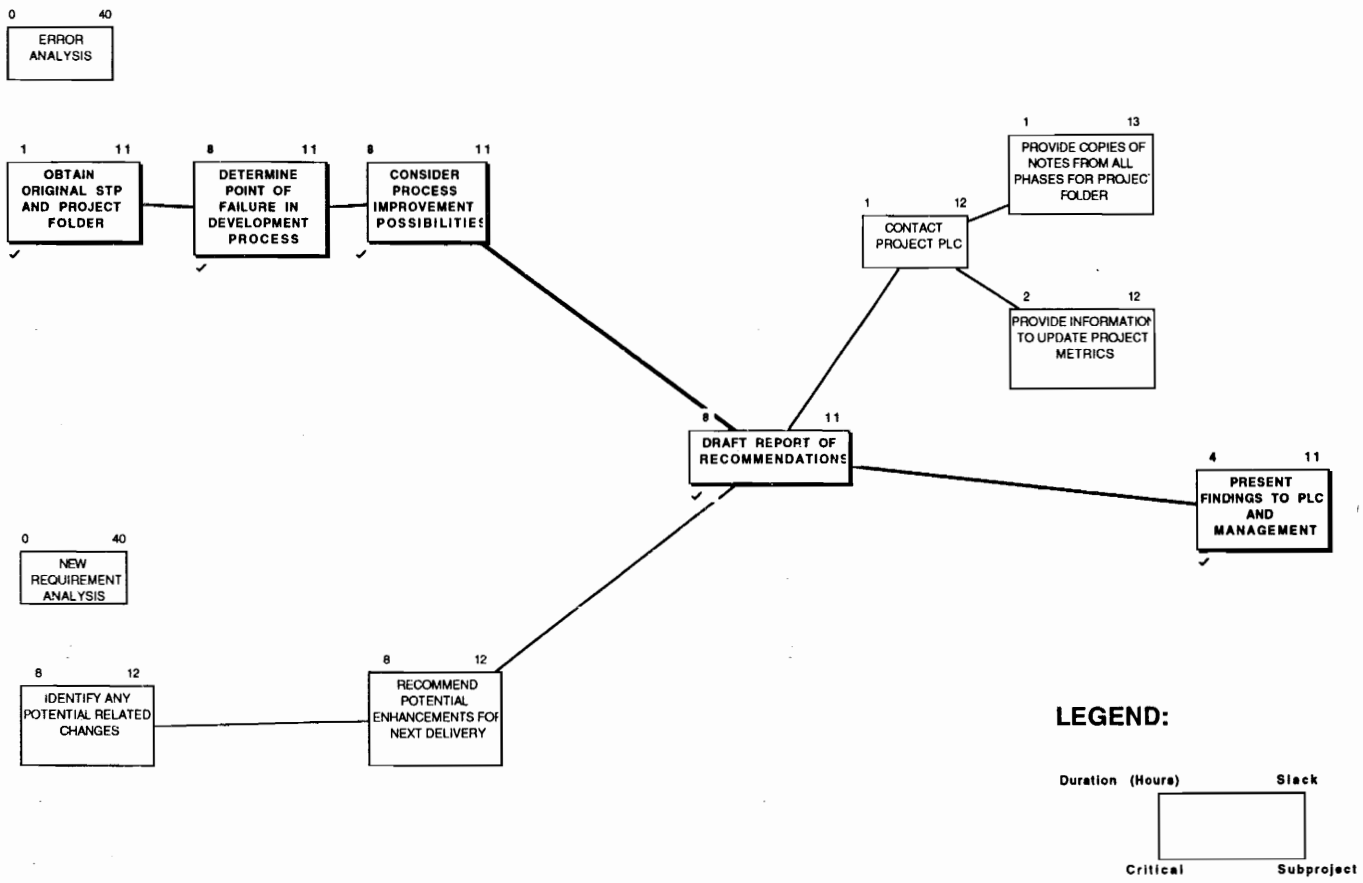



Figure A25: CRITMIN Process Analysis Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | ANALYST | 1 | 100 | 1 | 64.58 |
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | ANALYST | 8 | 100 | 8 | 516.64 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | ANALYST | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | ANALYST | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | ANALYST | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | ANALYST | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| CONTACT PROJECT PLC | ANALYST | 1 | 100 | 1 | 64.58 |
| CONTACT PROJECT PLC | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| CONTACT PROJECT PLC | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | ANALYST | 2 | 100 | 2 | 129.16 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | QA SPECIALIST | 2 | 100 | 2 | 129.16 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | ANALYST | 1 | 100 | 1 | 64.58 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | ANALYST | 4 | 100 | 4 | 258.32 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | QA SPECIALIST | 4 | 100 | 4 | 258.32 |

Figure A26: CRITMIN Process Analysis Phase Resource Allocation

# APPENDIX B: CRITINTER PROCESS

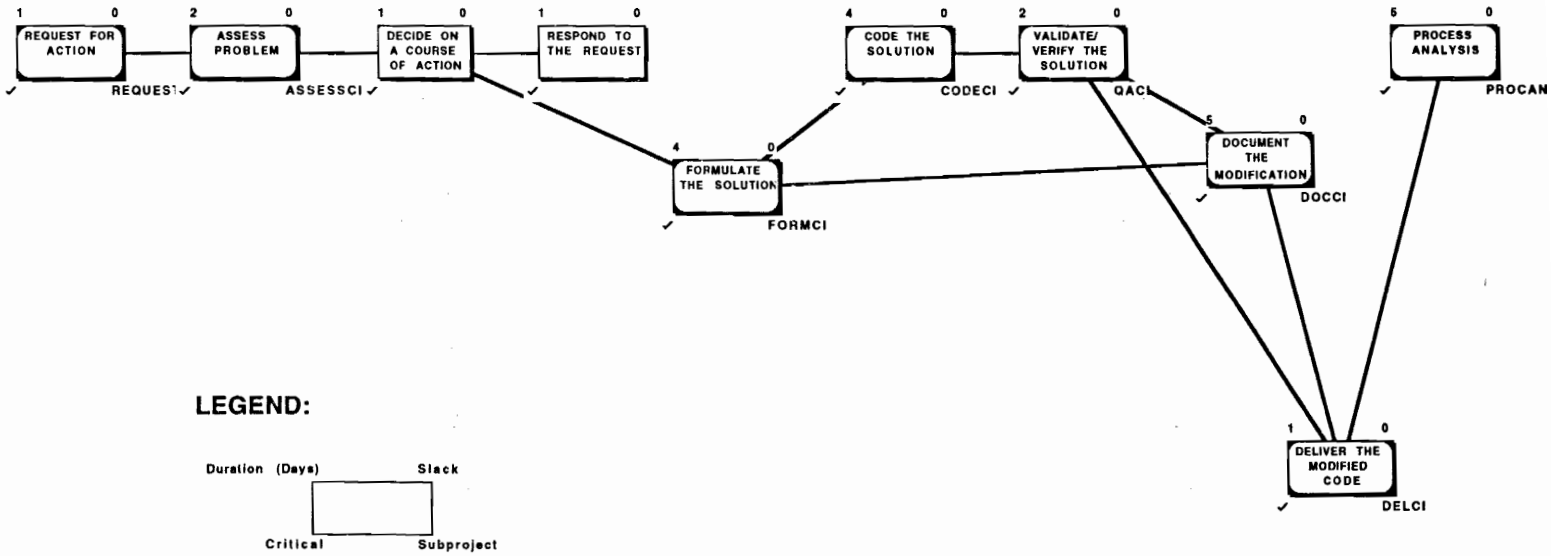



Figure B1: CRITINTER Process

CRITINTER
4/7/95
Project Overview

| | |
|---|---|
| • Tasks and Activities | |
| Tasks | 2 |
| Milestones | 0 |
| Supertasks | 8 |
| Completed Tasks | 0 |
| Completed Milestones | 0 |
| Summary Activities | 0 |
| • Project Dates | |
| Project Start Date | 7/1/95 |
| Project Finish Date | 8/1/95 |
| Actual Start | 7/3/95 |
| Actual Finish | 8/1/95 |
| • Project Costs | |
| Total Planned Cost | 45,504.16 |
| Total Planned Income | 0.00 |
| Total Actual Cost | 0.00 |
| Total Actual Income | 0.00 |
| • Other | |
| Projects in Family Tree | 13 |
| Projects in Resource Scope | 0 |
| Project % Done | 0 |
| Number of Resources | 10 |
| Number of Calendars | 1 |

Figure B2: CRITINTER Project Overview

| Name | Subproject | # Resources | Duration (Days) | Slack | Total Cost |
|---|---|---|---|---|---|
| REQUEST FOR ACTION | REQUEST | 4 | 1 | 0 | 1,725.75 |
| ASSESS PROBLEM | ASSESSCI | 3 | 2 | 0 | 4,712.08 |
| DECIDE ON A COURSE OF ACTION | | 3 | 1 | 0 | 505.92 |
| RESPOND TO THE REQUEST | | 1 | 1 | 0 | 168.64 |
| FORMULATE THE SOLUTION | FORMCI | 1 | 4 | 0 | 4,133.12 |
| CODE THE SOLUTION | CODECI | 1 | 4 | 0 | 4,133.12 |
| VALIDATE/ VERIFY THE SOLUTION | QACI | 1 | 2 | 0 | 4,617.47 |
| DOCUMENT THE MODIFICATION | DOCCI | 2 | 5 | 0 | 6,199.68 |
| DELIVER THE MODIFIED CODE | DELCI | 5 | 1 | 0 | 2,065.52 |
| PROCESS ANALYSIS | PROCAN | 3 | 5 | 0 | 17,242.86 |

Figure B3: CRITINTER Total Resource Allocation

| Name | Resource | Work-Days | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REQUEST FOR ACTION | LIAISON | .50 | 50 | 1 | 258.32 |
| REQUEST FOR ACTION | K42 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| REQUEST FOR ACTION | K51 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| REQUEST FOR ACTION | K52 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| ASSESS PROBLEM | ANALYST | 2 | 100 | 2 | 1,033.28 |
| ASSESS PROBLEM | PROGRAMMER | 2 | 100 | 2 | 1,033.28 |
| ASSESS PROBLEM | LIAISON | .12 | 100 | .12 | 64.58 |
| DECIDE ON A COURSE OF ACTION | K42 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| DECIDE ON A COURSE OF ACTION | K51 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| DECIDE ON A COURSE OF ACTION | K52 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| RESPOND TO THE REQUEST | K42 BRANCH HEAD | .25 | 50 | .50 | 168.64 |
| FORMULATE THE SOLUTION | ANALYST | 4 | 100 | 4 | 2,066.56 |
| CODE THE SOLUTION | PROGRAMMER | 4 | 100 | 4 | 2,066.56 |
| VALIDATE/ VERIFY THE SOLUTION | QA SPECIALIST | 2 | 100 | 2 | 1,033.28 |
| DOCUMENT THE MODIFICATION | ANALYST | 3 | 100 | 3 | 1,549.92 |
| DOCUMENT THE MODIFICATION | QA SPECIALIST | 2 | 100 | 2 | 1,033.28 |
| DELIVER THE MODIFIED CODE | ANALYST | .25 | 100 | .25 | 129.16 |
| DELIVER THE MODIFIED CODE | QA SPECIALIST | .12 | 100 | .12 | 64.58 |
| DELIVER THE MODIFIED CODE | QC SPECIALIST | .88 | 100 | .88 | 386.96 |
| DELIVER THE MODIFIED CODE | LIAISON | .62 | 100 | .62 | 322.90 |
| DELIVER THE MODIFIED CODE | PROGRAMMER | .25 | 100 | .25 | 129.16 |
| PROCESS ANALYSIS | PROGRAMMER | 5 | 100 | 5 | 2,583.20 |
| PROCESS ANALYSIS | QA SPECIALIST | 5 | 100 | 5 | 2,583.20 |
| PROCESS ANALYSIS | ANALYST | 5 | 100 | 5 | 2,583.20 |

Figure B4: CRITINTER Process Individual Resource Allocation

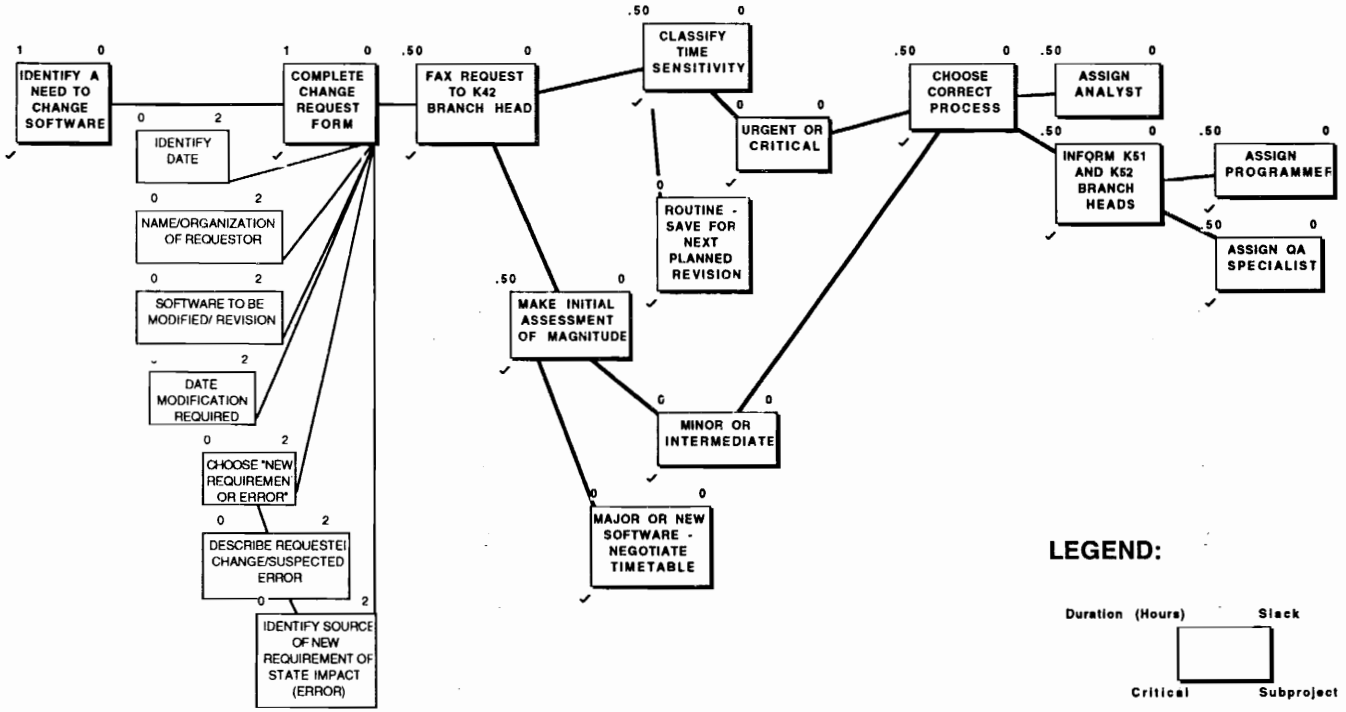



Figure B5: CRITINTER Request Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| IDENTIFY A NEED TO CHANGE SOFTWARE | LIAISON | 1 | 100 | 1 | 64.58 |
| COMPLETE CHANGE REQUEST FORM | LIAISON | 1 | 100 | 1 | 64.58 |
| FAX REQUEST TO K42 BRANCH HEAD | LIAISON | .50 | 100 | .50 | 32.29 |
| CLASSIFY TIME SENSITIVITY | K42 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| MAKE INITIAL ASSESSMENT OF MAGNITUDE | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| CHOOSE CORRECT PROCESS | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| ASSIGN ANALYST | K42 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| INFORM K51 AND K52 BRANCH HEADS | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| ASSIGN PROGRAMMER | K51 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| ASSIGN QA SPECIALIST | K52 BRANCH HEAD | .50 | 100 | .50 | 42.16 |

Figure B6: CRITINTER Request Phase Resource Allocation

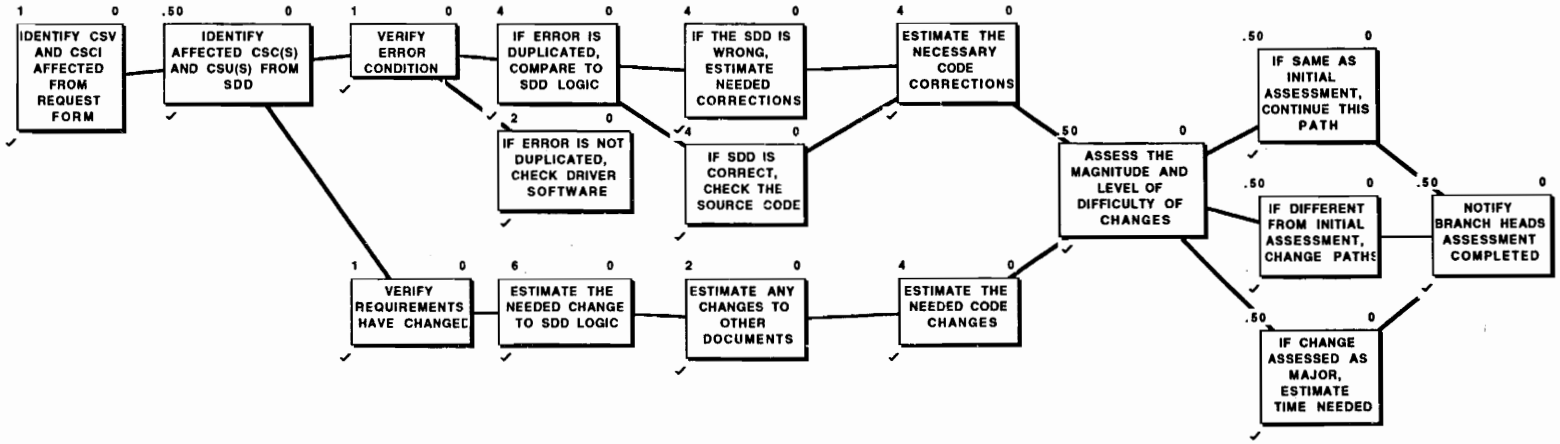



Figure B7: CRITINTER Assessment Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| IDENTIFY CSV AND CSCI AFFECTED FROM REQUEST FORM | ANALYST | 1 | 100 | 1 | 64.58 |
| IDENTIFY CSV AND CSCI AFFECTED FROM REQUEST FORM | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| IDENTIFY AFFECTED CSC(S) AND CSU(S) FROM SDD | ANALYST | .50 | 100 | .50 | 32.04 |
| IDENTIFY AFFECTED CSC(S) AND CSU(S) FROM SDD | PROGRAMMER | .50 | 100 | .50 | 32.04 |
| VERIFY ERROR CONDITION BY RUNNING A TEST CASE | ANALYST | 1 | 100 | 1 | 64.58 |
| IF ERROR IS NOT DUPLICATED, CHECK DRIVER SOFTWARE | LIAISON | 2 | 100 | 2 | 129.16 |
| IF ERROR IS DUPLICATED, COMPARE TO SDD LOGIC | ANALYST | 4 | 100 | 4 | 258.32 |
| IF THE SDD IS WRONG, ESTIMATE NEEDED CORRECTIONS | ANALYST | 4 | 100 | 4 | 258.32 |
| IF SDD IS CORRECT, CHECK THE SOURCE CODE | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| ESTIMATE THE NECESSARY CODE CORRECTIONS | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| VERIFY REQUIREMENTS HAVE CHANGED | ANALYST | 1 | 100 | 1 | 64.58 |
| ESTIMATE THE NEEDED CHANGE TO SDD LOGIC | ANALYST | 6 | 100 | 6 | 387.48 |
| ESTIMATE ANY CHANGES TO OTHER DOCUMENTS | ANALYST | 2 | 100 | 2 | 129.16 |
| ESTIMATE THE NEEDED CODE CHANGES | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| ASSESS THE MAGNITUDE AND LEVEL OF DIFFICULTY OF CHANGES | ANALYST | .50 | 100 | .50 | 32.04 |
| ASSESS THE MAGNITUDE AND LEVEL OF DIFFICULTY OF CHANGES | PROGRAMMER | .50 | 100 | .50 | 32.04 |
| IF SAME AS INITIAL ASSESSMENT, CONTINUE THIS PATH | ANALYST | .50 | 100 | .50 | 32.29 |
| IF SAME AS INITIAL ASSESSMENT, CONTINUE THIS PATH | PROGRAMMER | .50 | 100 | .50 | 32.04 |
| IF DIFFERENT FROM INITIAL ASSESSMENT, CHANGE PATHS | ANALYST | .50 | 100 | .50 | 32.04 |
| IF DIFFERENT FROM INITIAL ASSESSMENT, CHANGE PATHS | PROGRAMMER | .50 | 100 | .50 | 32.04 |
| IF CHANGE ASSESSED AS MAJOR, ESTIMATE TIME NEEDED | ANALYST | .50 | 100 | .50 | 32.04 |
| IF CHANGE ASSESSED AS MAJOR, ESTIMATE TIME NEEDED | PROGRAMMER | .50 | 100 | .50 | 32.04 |
| NOTIFY BRANCH HEADS ASSESSMENT COMPLETED | ANALYST | .50 | 100 | .50 | 32.29 |
| NOTIFY BRANCH HEADS ASSESSMENT COMPLETED | PROGRAMMER | .50 | 100 | .50 | 32.29 |

Figure B8: CRITINTER Assessment Phase Resource Allocation

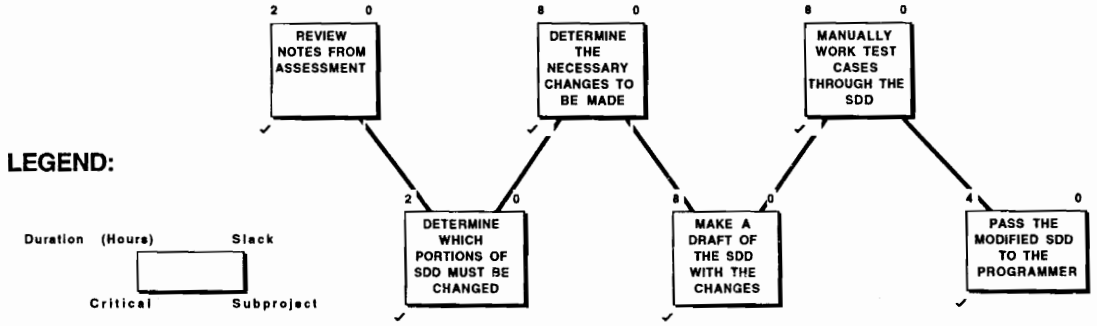



Figure B9: CRITINTER FORMULATION PHASE

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REVIEW NOTES FROM ASSESSMENT | ANALYST | 2 | 100 | 2 | 129.16 |
| DETERMINE WHICH PORTIONS OF SDD MUST BE CHANGED | ANALYST | 2 | 100 | 2 | 129.16 |
| DETERMINE THE NECESSARY CHANGES TO BE MADE | ANALYST | 8 | 100 | 8 | 516.64 |
| MAKE A DRAFT OF THE SDD WITH THE CHANGES | ANALYST | 8 | 100 | 8 | 516.64 |
| MANUALLY WORK TEST CASES THROUGH THE SDD | ANALYST | 8 | 100 | 8 | 516.64 |
| PASS THE MODIFIED SDD TO THE PROGRAMMER | ANALYST | 4 | 100 | 4 | 258.32 |

Figure B10: CRITINTER Formulation Phase Resource Allocation

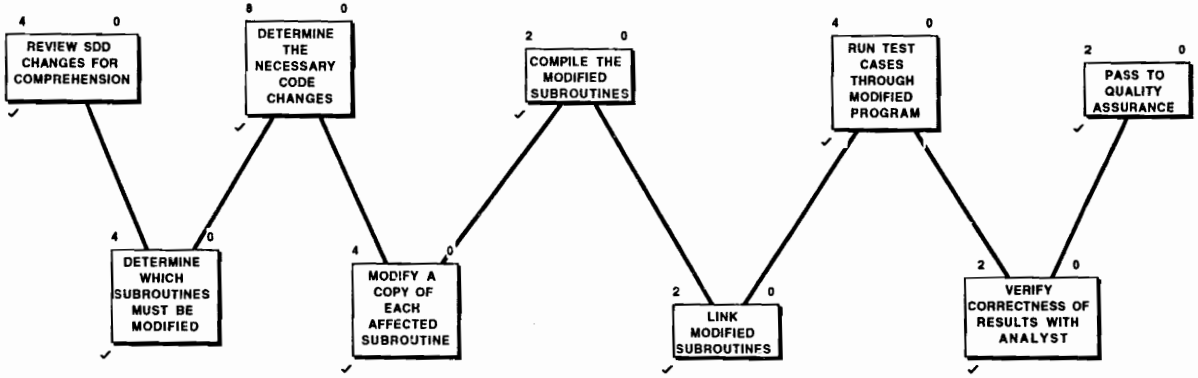



Figure B11: CRITINTER Coding Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REVIEW SDD CHANGES FOR COMPREHENSION | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| DETERMINE WHICH SUBROUTINES MUST BE MODIFIED | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| DETERMINE THE NECESSARY CODE CHANGES | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| MODIFY A COPY OF EACH AFFECTED SUBROUTINE | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| COMPILE THE MODIFIED SUBROUTINES | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| LINK MODIFIED SUBROUTINES WITH COPY OF REST OF CODE | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| RUN TEST CASES THROUGH MODIFIED PROGRAM | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| VERIFY CORRECTNESS OF RESULTS WITH ANALYST | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| PASS TO QUALITY ASSURANCE SPECIALIST | PROGRAMMER | 2 | 100 | 2 | 129.16 |

Figure B12: CRITINTER Coding Phase Resource Allocation

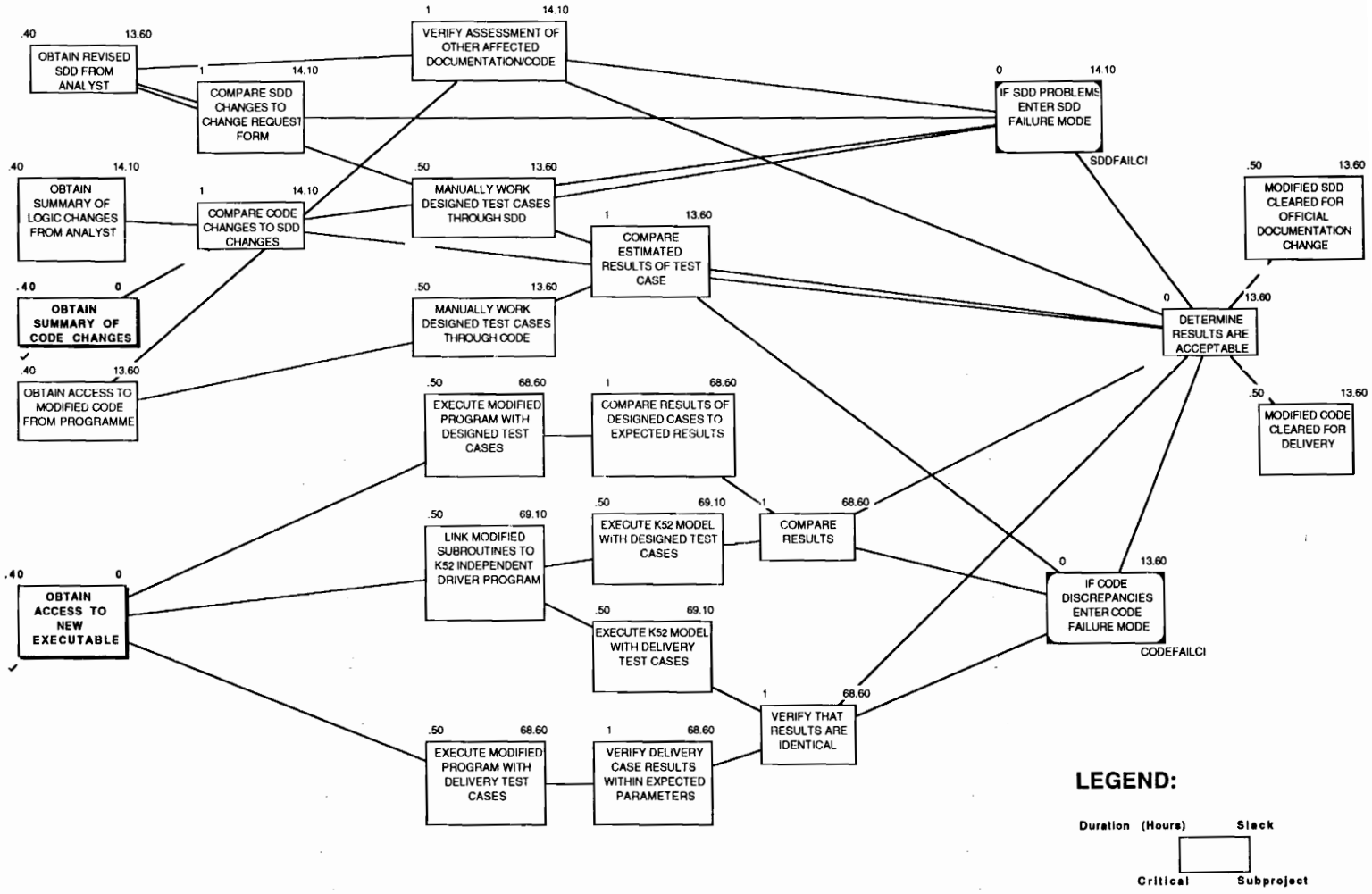



Figure B13: CRITINTER Validation and Verification Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| OBTAIN REVISED SDD FROM ANALYST | QA SPECIALIST | .40 | 100 | .40 | 25.83 |
| OBTAIN SUMMARY OF LOGIC CHANGES FROM ANALYST | QA SPECIALIST | .40 | 100 | .40 | 25.83 |
| OBTAIN ACCESS TO MODIFIED CODE FROM PROGRAMMER | QA SPECIALIST | .40 | 100 | .40 | 25.83 |
| OBTAIN ACCESS TO NEW EXECUTABLE | QA SPECIALIST | .40 | 100 | .40 | 25.83 |
| OBTAIN SUMMARY OF CODE CHANGES | QA SPECIALIST | .40 | 100 | .40 | 25.83 |
| COMPARE SDD CHANGES TO CHANGE REQUEST FORM | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| COMPARE CODE CHANGES TO SDD CHANGES | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| VERIFY ASSESSMENT OF OTHER AFFECTED DOCUMENTATION/CODE | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| MANUALLY WORK DESIGNED TEST CASES THROUGH SDD | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| MANUALLY WORK DESIGNED TEST CASES THROUGH CODE | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| COMPARE ESTIMATED RESULTS OF TEST CASE | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| EXECUTE MODIFIED PROGRAM WITH DESIGNED TEST CASES | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| COMPARE RESULTS OF DESIGNED CASES TO EXPECTED RESULTS | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| EXECUTE MODIFIED PROGRAM WITH DELIVERY TEST CASES | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| VERIFY DELIVERY CASE RESULTS WITHIN EXPECTED PARAMETERS | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| LINK MODIFIED SUBROUTINES TO K52 INDEPENDENT DRIVER PROGRAM | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| EXECUTE K52 MODEL WITH DESIGNED TEST CASES | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| COMPARE RESULTS | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| EXECUTE K52 MODEL WITH DELIVERY TEST CASES | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| VERIFY THAT RESULTS ARE IDENTICAL | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| MODIFIED CODE CLEARED FOR DELIVERY | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| MODIFIED SDD CLEARED FOR OFFICIAL DOCUMENTATION CHANGE | QA SPECIALIST | .50 | 100 | .50 | 32.29 |

Figure B14: CRITINTER Validation and Verification Phase Resource Allocation

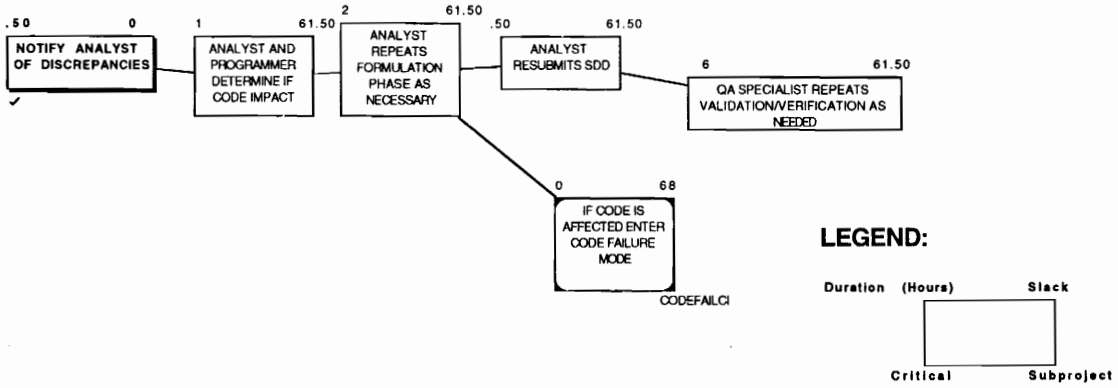



Figure B15: CRITINTER SDD Failure Mode

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| NOTIFY ANALYST OF DISCREPANCIES | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| ANALYST AND PROGRAMMER DETERMINE IF CODE IMPACT | ANALYST | 1 | 100 | 1 | 64.58 |
| ANALYST AND PROGRAMMER DETERMINE IF CODE IMPACT | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| ANALYST REPEATS FORMULATION PHASE AS NECESSARY | ANALYST | 2 | 100 | 2 | 129.16 |
| ANALYST RESUBMITS SDD | ANALYST | .50 | 100 | .50 | 32.29 |
| QA SPECIALIST REPEATS VALIDATION/VERIFICATION AS NEEDED | QA SPECIALIST | 6 | 100 | 6 | 387.48 |

Figure B16: CRITINTER SDD Failure Mode Resource Allocation

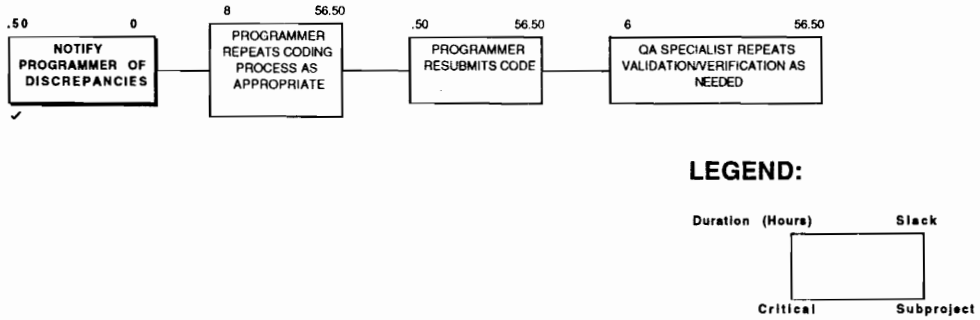



Figure B17: CRITINTER Code Failure Mode

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| NOTIFY PROGRAMMER OF DISCREPANCIES | QA SPECIALIST | .50 | 100 | .50 | 32.29 |
| PROGRAMMER REPEATS CODING PROCESS AS APPROPRIATE | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| PROGRAMMER RESUBMITS CODE | PROGRAMMER | .50 | 100 | .50 | 32.29 |
| QA SPECIALIST REPEATS VALIDATION/VERIFICATION AS NEEDED | QA SPECIALIST | 6 | 100 | 6 | 387.48 |

Figure B18: CRITINTER Code Failure Mode Resource Allocation

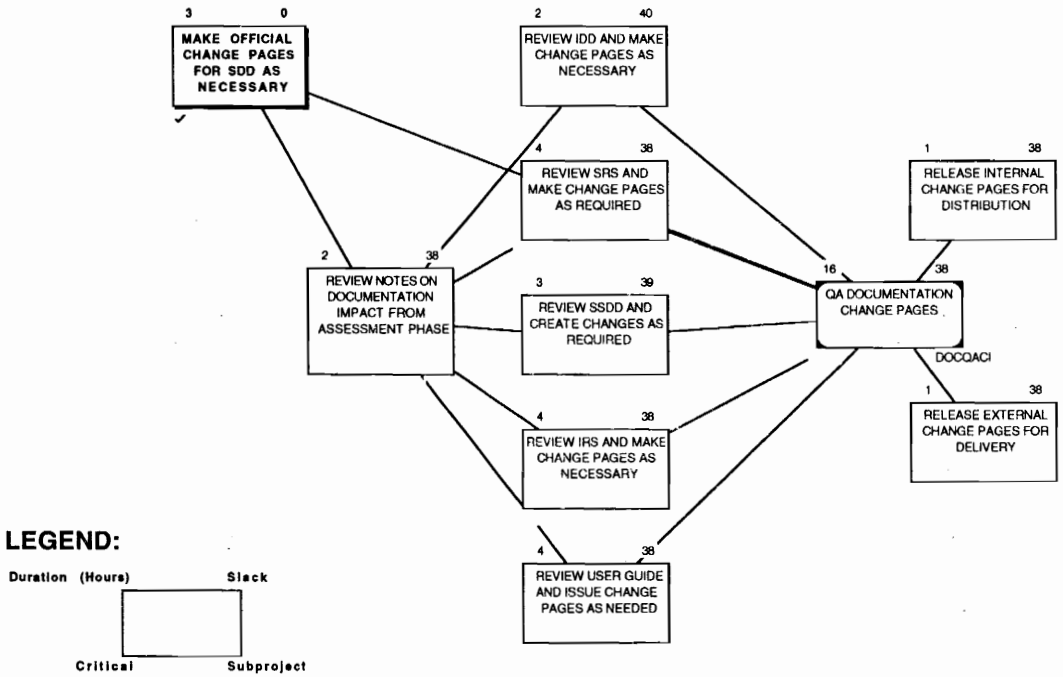



Figure B19: CRITINTER Documentation Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| MAKE OFFICIAL CHANGE PAGES FOR SDD AS NECESSARY | ANALYST | 3 | 100 | 3 | 193.74 |
| REVIEW NOTES ON DOCUMENTATION IMPACT FROM ASSESSMENT | ANALYST | 2 | 100 | 2 | 129.16 |
| REVIEW IDD AND MAKE CHANGE PAGES AS NECESSARY | ANALYST | 2 | 100 | 2 | 129.16 |
| REVIEW SRS AND MAKE CHANGE PAGES AS REQUIRED | ANALYST | 4 | 100 | 4 | 258.32 |
| REVIEW SSDD AND CREATE CHANGES AS REQUIRED | ANALYST | 3 | 100 | 3 | 193.74 |
| REVIEW IRS AND MAKE CHANGE PAGES AS NECESSARY | ANALYST | 4 | 100 | 4 | 258.32 |
| REVIEW USER GUIDE AND ISSUE CHANGE PAGES AS NEEDED | ANALYST | 4 | 100 | 4 | 258.32 |
| QA DOCUMENTATION CHANGE PAGES | QA SPECIALIST | 16 | 100 | 16 | 1,033.28 |
| RELEASE INTERNAL CHANGE PAGES FOR DISTRIBUTION | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| RELEASE EXTERNAL CHANGE PAGES FOR DELIVERY | QA SPECIALIST | 1 | 100 | 1 | 64.58 |



Figure B20: CRITINTER Documentation Phase Resource Allocation

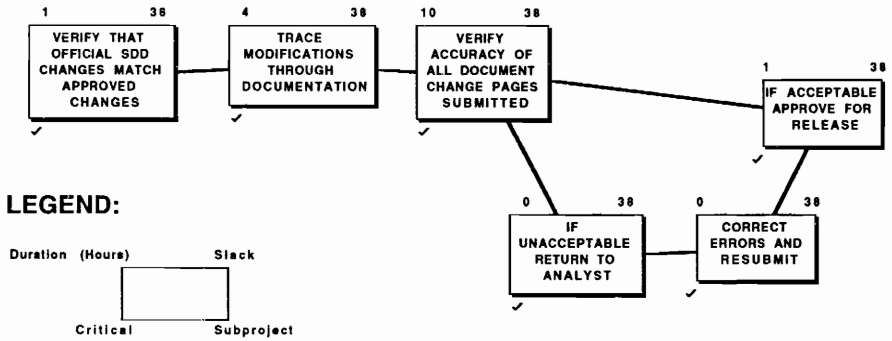



Figure B21: Documentation Quality Assurance Mode

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| VERIFY THAT OFFICIAL SDD CHANGES MATCH APPROVED CHANGE | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| TRACE MODIFICATIONS THROUGH DOCUMENTATION | QA SPECIALIST | 4 | 100 | 4 | 258.32 |
| VERIFY ACCURACY OF ALL DOCUMENT CHANGE PAGES SUBMITTE | QA SPECIALIST | 10 | 100 | 10 | 645.80 |
| IF UNACCEPTABLE RETURN TO ANALYST | ANALYST | 0 | 100 | 0 | 0.00 |
| CORRECT ERRORS AND RESUBMIT | ANALYST | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE APPROVE FOR RELEASE | QA SPECIALIST | 1 | 100 | 1 | 64.58 |

Figure B22: CRITINTER Documentation Quality Assurance Mode Resource Allocation

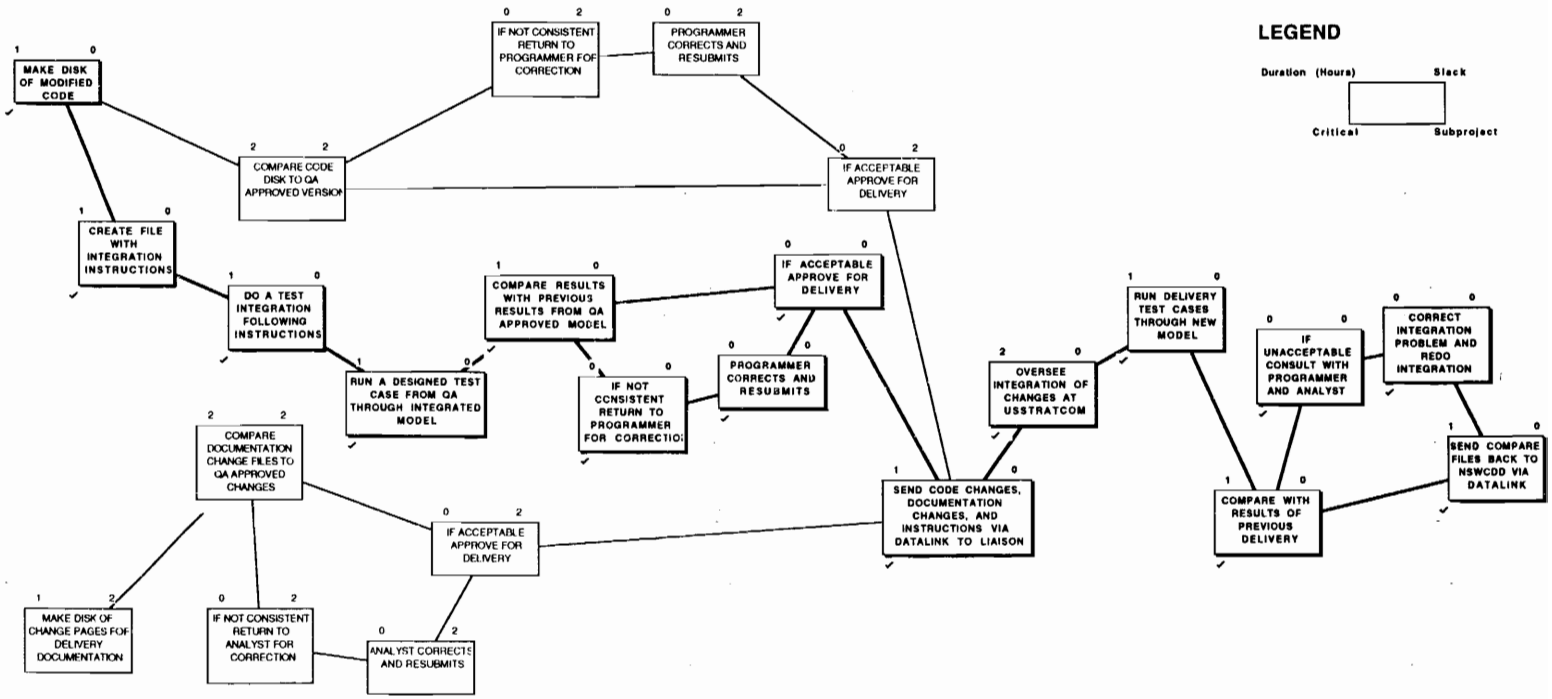



Figure B23: CRITINTER Delivery Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| MAKE DISK OF MODIFIED CODE | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| MAKE DISK OF CHANGE PAGES FOR DELIVERY DOCUMENTATION | ANALYST | 1 | 100 | 1 | 64.58 |
| CREATE FILE WITH INTEGRATION INSTRUCTIONS | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| COMPARE CODE DISK TO QA APPROVED VERSION | QC SPECIALIST | 2 | 100 | 2 | 110.56 |
| IF NOT CONSISTENT RETURN TO PROGRAMMER FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| PROGRAMMER CORRECTS AND RESUBMITS | PROGRAMMER | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE APPROVE FOR DELIVERY | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| COMPARE DOCUMENTATION CHANGE FILES TO QA APPROVED CHANGES | QC SPECIALIST | 2 | 100 | 2 | 110.56 |
| IF NOT CONSISTENT RETURN TO ANALYST FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| ANALYST CORRECTS AND RESUBMITS | ANALYST | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE APPROVE FOR DELIVERY | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| DO A TEST INTEGRATION FOLLOWING INSTRUCTIONS | QC SPECIALIST | 1 | 100 | 1 | 55.28 |
| RUN A DESIGNED TEST CASE FROM QA THROUGH INTEGRATED MODEL | QC SPECIALIST | 1 | 100 | 1 | 55.28 |
| COMPARE RESULTS WITH PREVIOUS RESULTS FROM QA APPROVED MODEL | QC SPECIALIST | 1 | 100 | 1 | 55.28 |
| IF NOT CONSISTENT RETURN TO PROGRAMMER FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| PROGRAMMER CORRECTS AND RESUBMITS | PROGRAMMER | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE APPROVE FOR DELIVERY | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| SEND CODE CHANGES, DOCUMENTATION CHANGES, AND INSTRUCTIONS VIA DATA | ANALYST | 1 | 100 | 1 | 64.58 |
| SEND CODE CHANGES, DOCUMENTATION CHANGES, AND INSTRUCTIONS VIA DATA | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| OVERSEE INTEGRATION OF CHANGES AT USSTRATCOM | LIAISON | 2 | 100 | 2 | 129.16 |
| RUN DELIVERY TEST CASES THROUGH NEW MODEL | LIAISON | 1 | 100 | 1 | 64.58 |
| COMPARE WITH RESULTS OF PREVIOUS DELIVERY | LIAISON | 1 | 100 | 1 | 64.58 |
| IF UNACCEPTABLE CONSULT WITH PROGRAMMER AND ANALYST | LIAISON | 0 | 100 | 0 | 0.00 |
| CORRECT INTEGRATION PROBLEM AND REDO INTEGRATION | LIAISON | 0 | 100 | 0 | 0.00 |
| SEND COMPARE FILES BACK TO NSWCDD VIA DATALINK | LIAISON | 1 | 100 | 1 | 64.58 |

Figure B24: CRITINTER Delivery Phase Resource Allocation

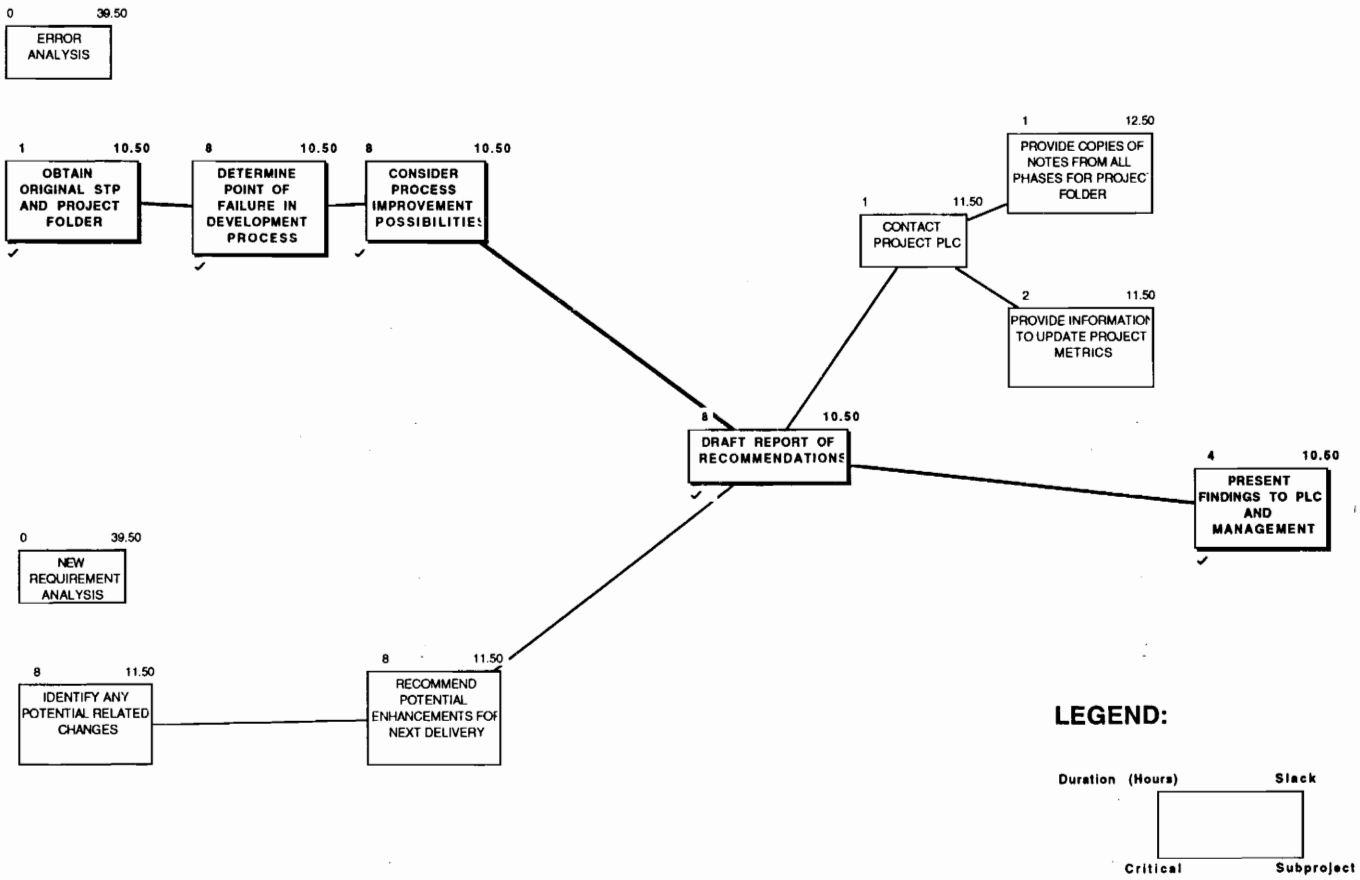



Figure B25: CRITINTER Process Analysis Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | ANALYST | 1 | 100 | 1 | 64.58 |
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | ANALYST | 8 | 100 | 8 | 516.64 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | ANALYST | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | ANALYST | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | ANALYST | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | ANALYST | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| CONTACT PROJECT PLC | ANALYST | 1 | 100 | 1 | 64.58 |
| CONTACT PROJECT PLC | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| CONTACT PROJECT PLC | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | ANALYST | 2 | 100 | 2 | 129.16 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | QA SPECIALIST | 2 | 100 | 2 | 129.16 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | ANALYST | 1 | 100 | 1 | 64.58 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | ANALYST | 4 | 100 | 4 | 258.32 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | QA SPECIALIST | 4 | 100 | 4 | 258.32 |



Figure B26: CRITINTER Process Analysis Phase Resource Allocation

# APPENDIX C: URGMIN PROCESS

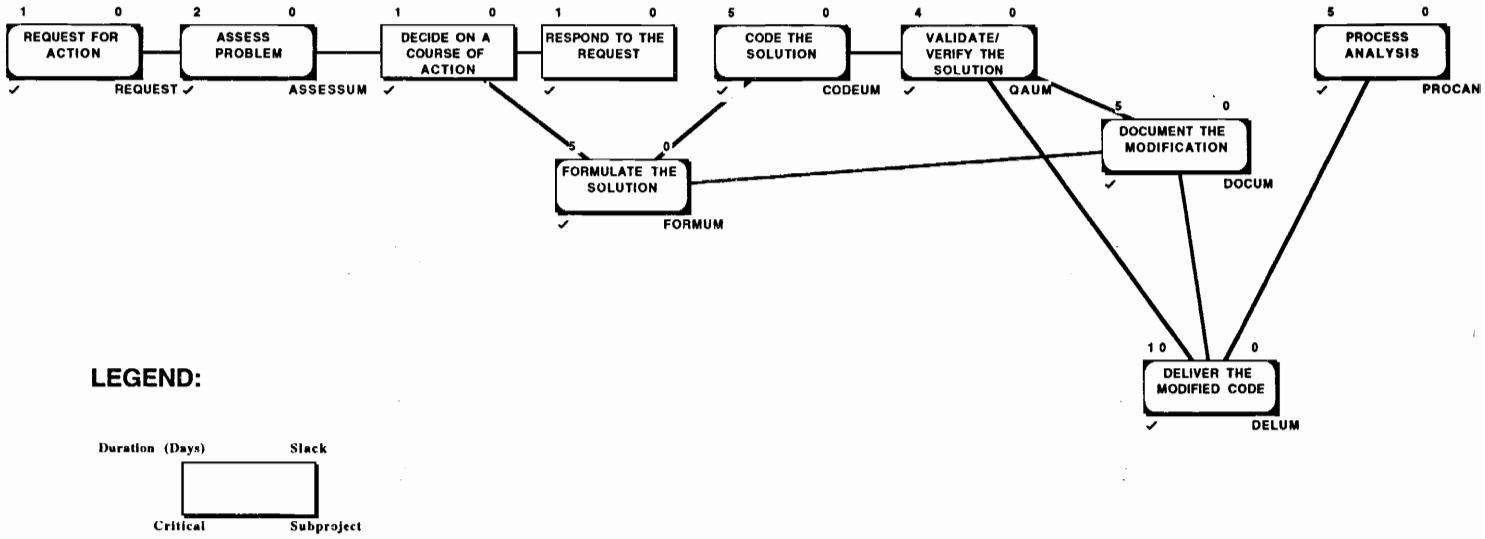



Figure C1: URGMIN Process

URGMIN
4/7/95
Project Overview

| | |
|---|---|
| • Tasks and Activities | |
| Tasks | 2 |
| Milestones | 0 |
| Supertasks | 8 |
| Completed Tasks | 0 |
| Completed Milestones | 0 |
| Summary Activities | 0 |
| • Project Dates | |
| Project Start Date | 7/1/95 |
| Project Finish Date | 8/18/95 |
| Actual Start | 7/3/95 |
| Actual Finish | 8/18/95 |
| • Project Costs | |
| Total Planned Cost | 31,064.79 |
| Total Planned Income | 0.00 |
| Total Actual Cost | 0.00 |
| Total Actual Income | 0.00 |
| • Other | |
| Projects in Family Tree | 13 |
| Projects in Resource Scope | 0 |
| Project % Done | 0 |
| Number of Resources | 10 |
| Number of Calendars | 1 |

Figure C2: URGMIN Project Overview

| Name | Subproject | # Resources | Duration (Days) | Slack | Total Cost |
|---|---|---|---|---|---|
| REQUEST FOR ACTION | REQUEST | 4 | 1 | 0 | 1,725.75 |
| ASSESS PROBLEM | ASSESSUM | 3 | 2 | 0 | 2,452.28 |
| DECIDE ON A COURSE OF ACTION | | 3 | 1 | 0 | 505.92 |
| RESPOND TO THE REQUEST | | 1 | 1 | 0 | 168.64 |
| FORMULATE THE SOLUTION | FORMUM | 1 | 5 | 0 | 1,033.28 |
| CODE THE SOLUTION | CODEUM | 1 | 5 | 0 | 1,033.28 |
| VALIDATE/ VERIFY THE SOLUTION | QAUM | 1 | 4 | 0 | 2,131.14 |
| DOCUMENT THE MODIFICATION | DOCUM | 2 | 5 | 0 | 3,099.84 |
| DELIVER THE MODIFIED CODE | DELUM | 6 | 10 | 0 | 1,671.80 |
| PROCESS ANALYSIS | PROCAN | 3 | 5 | 0 | 17,242.86 |

Figure C3: URGMIN Total Resource Allocation

| Name | Resource | Work-Days | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REQUEST FOR ACTION | LIAISON | .50 | 50 | 1 | 258.32 |
| REQUEST FOR ACTION | K42 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| REQUEST FOR ACTION | K51 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| REQUEST FOR ACTION | K52 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| ASSESS PROBLEM | ANALYST | 1 | 50 | 2 | 516.64 |
| ASSESS PROBLEM | PROGRAMMER | 1 | 50 | 2 | 516.64 |
| ASSESS PROBLEM | LIAISON | .12 | 50 | .25 | 64.58 |
| DECIDE ON A COURSE OF ACTION | K42 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| DECIDE ON A COURSE OF ACTION | K51 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| DECIDE ON A COURSE OF ACTION | K52 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| RESPOND TO THE REQUEST | K42 BRANCH HEAD | .25 | 50 | .50 | 168.64 |
| FORMULATE THE SOLUTION | ANALYST | 1 | 20 | 5 | 516.64 |
| CODE THE SOLUTION | PROGRAMMER | 1 | 20 | 5 | 516.64 |
| VALIDATE/ VERIFY THE SOLUTION | QA SPECIALIST | 1 | 25 | 4 | 516.64 |
| DOCUMENT THE MODIFICATION | ANALYST | 1 | 20 | 5 | 516.64 |
| DOCUMENT THE MODIFICATION | QA SPECIALIST | 1 | 20 | 5 | 516.64 |
| DELIVER THE MODIFIED CODE | ANALYST | .25 | 50 | .50 | 129.16 |
| DELIVER THE MODIFIED CODE | QC SPECIALIST | 1 | 50 | 2 | 442.24 |
| DELIVER THE MODIFIED CODE | PROGRAMMER | .25 | 50 | .50 | 129.16 |
| DELIVER THE MODIFIED CODE | LIAISON | .50 | 50 | 1 | 258.32 |
| DELIVER THE MODIFIED CODE | MEDIA PREP SPECIALIST | .50 | 50 | 1 | 221.12 |
| DELIVER THE MODIFIED CODE | SHIPPING | 5 | 100 | 5 | 40.00 |
| PROCESS ANALYSIS | ANALYST | 5 | 100 | 5 | 2,583.20 |
| PROCESS ANALYSIS | PROGRAMMER | 5 | 100 | 5 | 2,583.20 |
| PROCESS ANALYSIS | QA SPECIALIST | 5 | 100 | 5 | 2,583.20 |

Figure C4: URGMIN Individual Resource Allocation

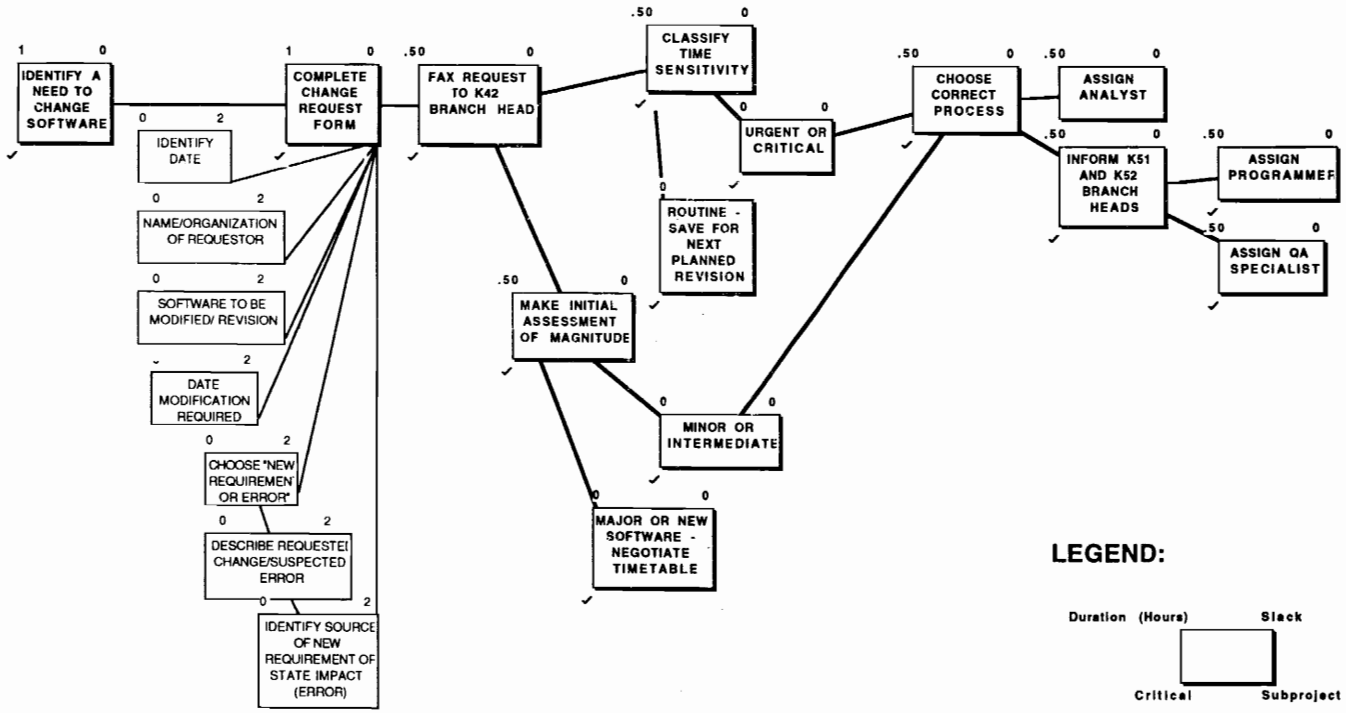



Figure C5: URGMIN Request Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| IDENTIFY A NEED TO CHANGE SOFTWARE | LIAISON | 1 | 100 | 1 | 64.58 |
| COMPLETE CHANGE REQUEST FORM | LIAISON | 1 | 100 | 1 | 64.58 |
| FAX REQUEST TO K42 BRANCH HEAD | LIAISON | .50 | 100 | .50 | 32.29 |
| CLASSIFY TIME SENSITIVITY | K42 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| MAKE INITIAL ASSESSMENT OF MAGNITUDE | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| CHOOSE CORRECT PROCESS | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| ASSIGN ANALYST | K42 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| INFORM K51 AND K52 BRANCH HEADS | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| ASSIGN PROGRAMMER | K51 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| ASSIGN QA SPECIALIST | K52 BRANCH HEAD | .50 | 100 | .50 | 42.16 |



Figure C6: URGMIN Request Phase Resource Allocation

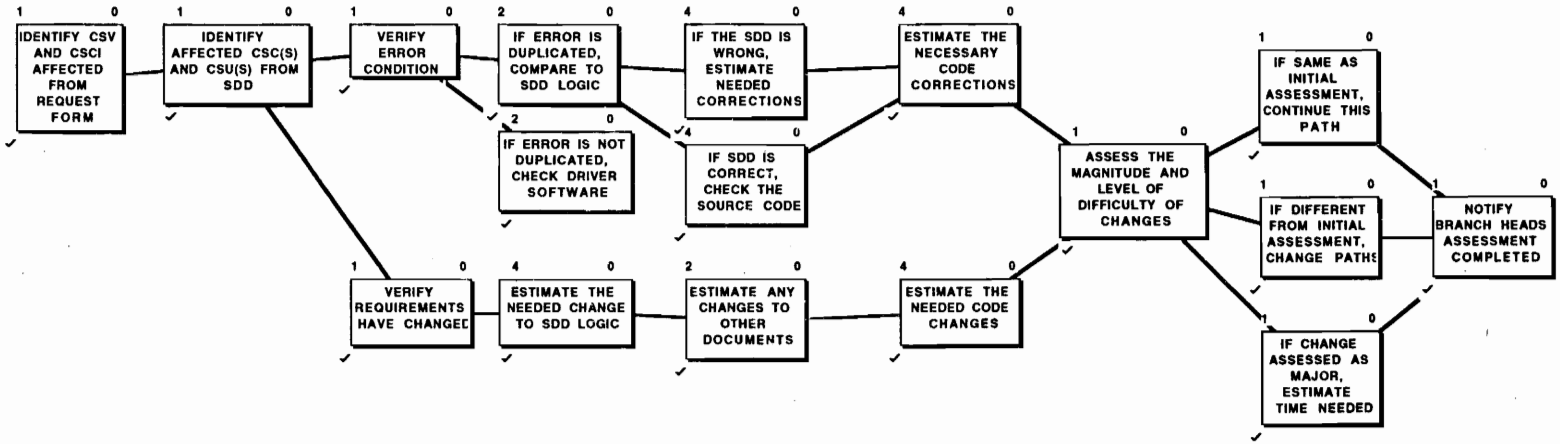





Figure C7: URGMIN Assessment Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| IDENTIFY CSV AND CSCI AFFECTED FROM REQUEST FORM | ANALYST | .50 | 50 | 1 | 32.29 |
| IDENTIFY CSV AND CSCI AFFECTED FROM REQUEST FORM | PROGRAMMER | .50 | 50 | 1 | 32.29 |
| IDENTIFY AFFECTED CSC(S) AND CSU(S) FROM SDD | ANALYST | .50 | 50 | 1 | 32.29 |
| IDENTIFY AFFECTED CSC(S) AND CSU(S) FROM SDD | PROGRAMMER | .50 | 50 | 1 | 32.29 |
| VERIFY ERROR CONDITION BY RUNNING A TEST CASE | ANALYST | .50 | 50 | 1 | 32.29 |
| IF ERROR IS NOT DUPLICATED, CHECK DRIVER SOFTWARE | LIAISON | 1 | 50 | 2 | 64.58 |
| IF ERROR IS DUPLICATED, COMPARE TO SDD LOGIC | ANALYST | 1 | 50 | 2 | 64.58 |
| IF THE SDD IS WRONG, ESTIMATE NEEDED CORRECTIONS | ANALYST | 2 | 50 | 4 | 129.16 |
| IF SDD IS CORRECT, CHECK THE SOURCE CODE | PROGRAMMER | 2 | 50 | 4 | 129.16 |
| ESTIMATE THE NECESSARY CODE CORRECTIONS | PROGRAMMER | 2 | 50 | 4 | 129.16 |
| VERIFY REQUIREMENTS HAVE CHANGED | ANALYST | .50 | 50 | 1 | 32.29 |
| ESTIMATE THE NEEDED CHANGE TO SDD LOGIC | ANALYST | 2 | 50 | 4 | 129.16 |
| ESTIMATE ANY CHANGES TO OTHER DOCUMENTS | ANALYST | 1 | 50 | 2 | 64.58 |
| ESTIMATE THE NEEDED CODE CHANGES | PROGRAMMER | 2 | 50 | 4 | 129.16 |
| ASSESS THE MAGNITUDE AND LEVEL OF DIFFICULTY OF CHANGES | ANALYST | .50 | 50 | .99 | 32.04 |
| ASSESS THE MAGNITUDE AND LEVEL OF DIFFICULTY OF CHANGES | PROGRAMMER | .50 | 50 | .99 | 32.04 |
| IF SAME AS INITIAL ASSESSMENT, CONTINUE THIS PATH | ANALYST | .50 | 50 | 1 | 32.29 |
| IF SAME AS INITIAL ASSESSMENT, CONTINUE THIS PATH | PROGRAMMER | .50 | 50 | .99 | 32.04 |
| IF DIFFERENT FROM INITIAL ASSESSMENT, CHANGE PATHS | ANALYST | .50 | 50 | .99 | 32.04 |
| IF DIFFERENT FROM INITIAL ASSESSMENT, CHANGE PATHS | PROGRAMMER | .50 | 50 | .99 | 32.04 |
| IF CHANGE ASSESSED AS MAJOR, ESTIMATE TIME NEEDED | ANALYST | .50 | 50 | .99 | 32.04 |
| IF CHANGE ASSESSED AS MAJOR, ESTIMATE TIME NEEDED | PROGRAMMER | .50 | 50 | .99 | 32.04 |
| NOTIFY BRANCH HEADS ASSESSMENT COMPLETED | ANALYST | .50 | 50 | 1 | 32.29 |
| NOTIFY BRANCH HEADS ASSESSMENT COMPLETED | PROGRAMMER | .50 | 50 | 1 | 32.29 |

Figure C8: URGMIN Assessment Phase Resource Allocation

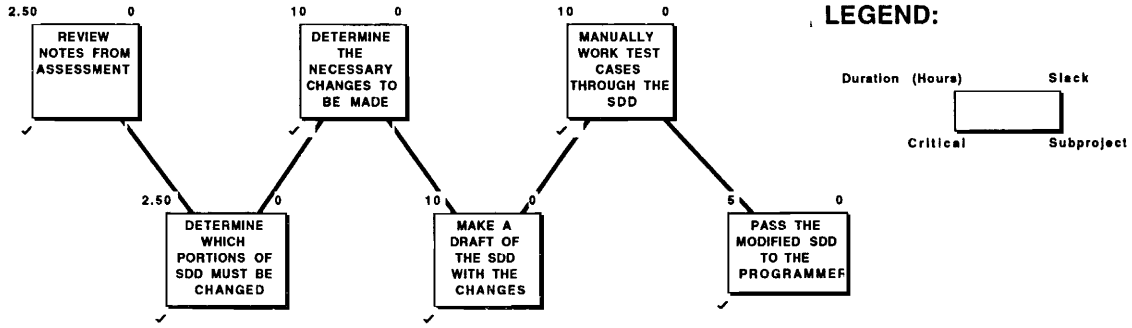



Figure C9: URGMIN Formulation Phase



| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REVIEW NOTES FROM ASSESSMENT | ANALYST | .50 | 20 | 2.50 | 32.29 |
| DETERMINE WHICH PORTIONS OF SDD MUST BE CHANGED | ANALYST | .50 | 20 | 2.50 | 32.29 |
| DETERMINE THE NECESSARY CHANGES TO BE MADE | ANALYST | 2 | 20 | 10 | 129.16 |
| MAKE A DRAFT OF THE SDD WITH THE CHANGES | ANALYST | 2 | 20 | 10 | 129.16 |
| MANUALLY WORK TEST CASES THROUGH THE SDD | ANALYST | 2 | 20 | 10 | 129.16 |
| PASS THE MODIFIED SDD TO THE PROGRAMMER | ANALYST | 1 | 20 | 5 | 64.58 |

Figure C10: URGMIN Formulation Phase Resource Allocation

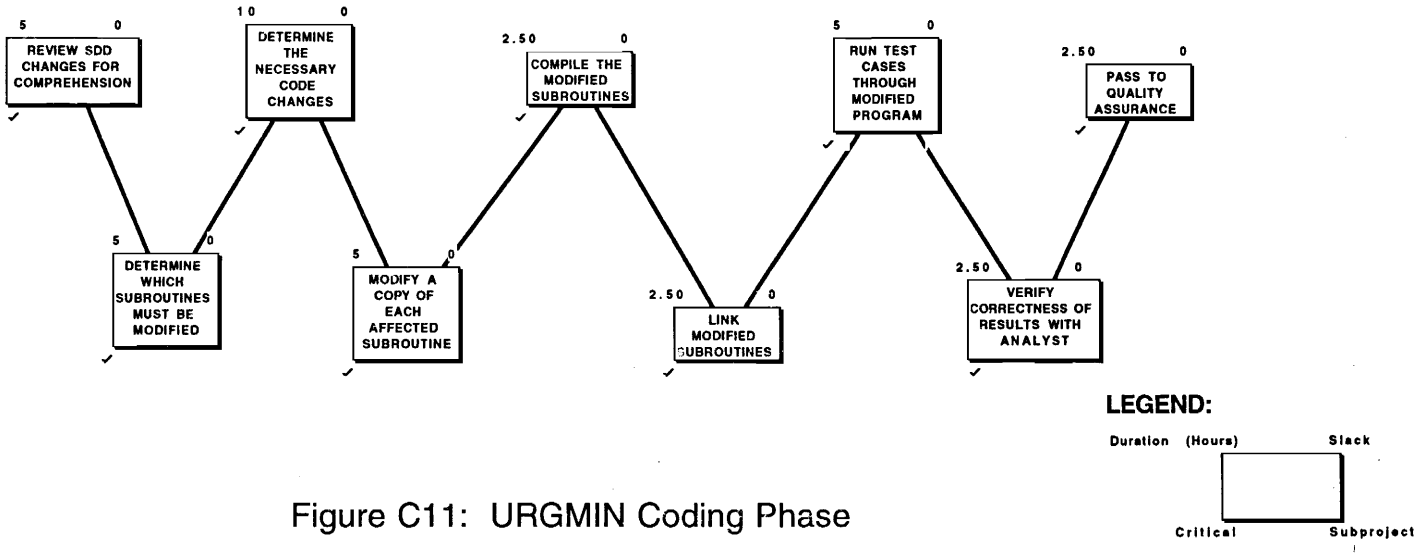



Figure C11: URGMIN Coding Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REVIEW SDD CHANGES FOR COMPREHENSION | PROGRAMMER | 1 | 20 | 5 | 64.58 |
| DETERMINE WHICH SUBROUTINES MUST BE MODIFIED | PROGRAMMER | 1 | 20 | 5 | 64.58 |
| DETERMINE THE NECESSARY CODE CHANGES | PROGRAMMER | 2 | 20 | 10 | 129.16 |
| MODIFY A COPY OF EACH AFFECTED SUBROUTINE | PROGRAMMER | 1 | 20 | 5 | 64.58 |
| COMPILE THE MODIFIED SUBROUTINES | PROGRAMMER | .50 | 20 | 2.50 | 32.29 |
| LINK MODIFIED SUBROUTINES WITH COPY OF REST OF CODE | PROGRAMMER | .50 | 20 | 2.50 | 32.29 |
| RUN TEST CASES THROUGH MODIFIED PROGRAM | PROGRAMMER | 1 | 20 | 5 | 64.58 |
| VERIFY CORRECTNESS OF RESULTS WITH ANALYST | PROGRAMMER | .50 | 20 | 2.50 | 32.29 |
| PASS TO QUALITY ASSURANCE SPECIALIST | PROGRAMMER | .50 | 20 | 2.50 | 32.29 |

Figure C12: URGMIN Coding Phase Resource Allocation

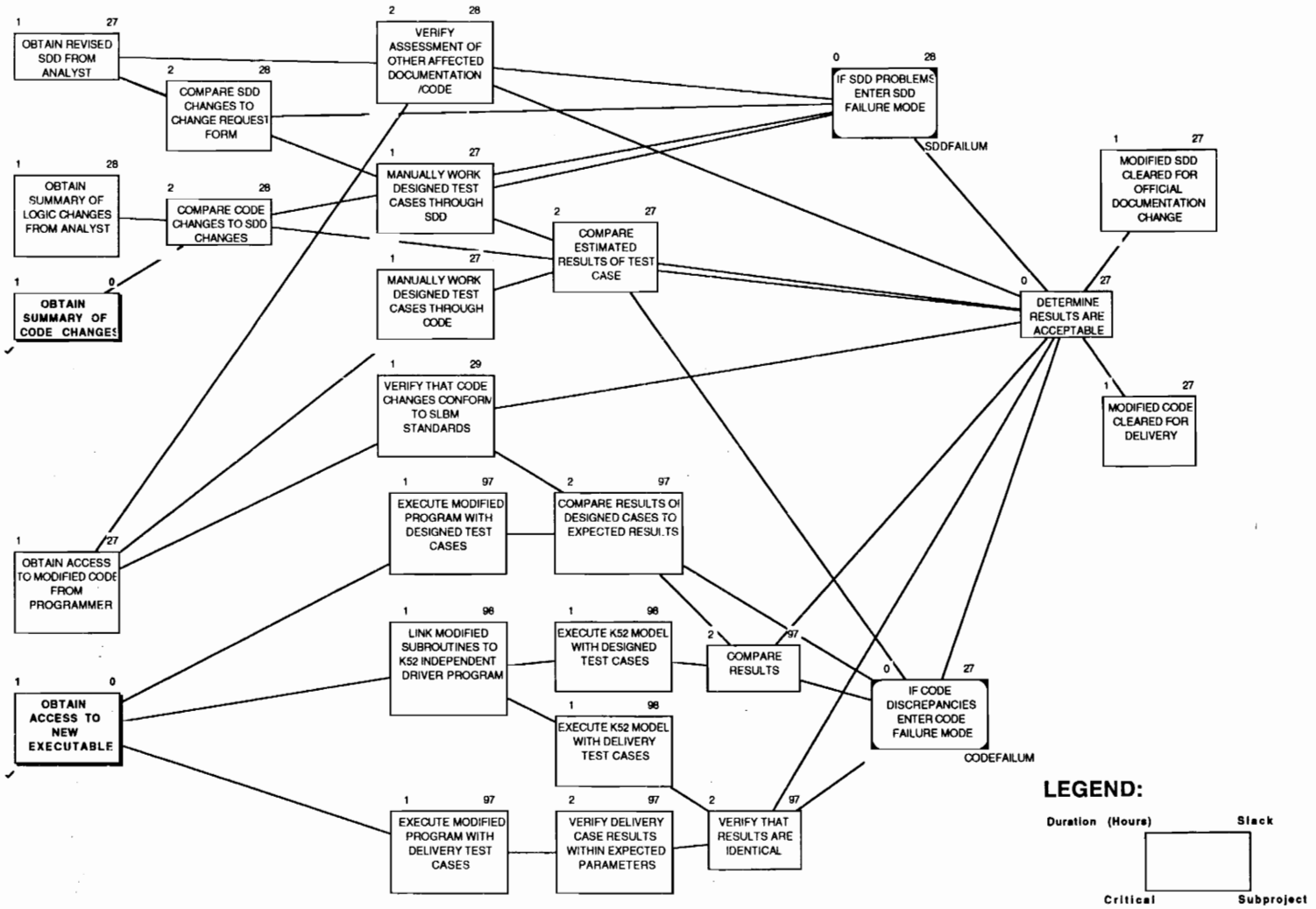



Figure C13: URGMIN Validation and Verification Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| OBTAIN REVISED SDD FROM ANALYST | QA SPECIALIST | .20 | 20 | 1 | 0.00 |
| OBTAIN SUMMARY OF LOGIC CHANGES FROM ANALYST | QA SPECIALIST | .20 | 20 | 1 | 0.00 |
| OBTAIN SUMMARY OF CODE CHANGES | QA SPECIALIST | .20 | 20 | 1 | 0.00 |
| OBTAIN ACCESS TO MODIFIED CODE FROM PROGRAMMER | QA SPECIALIST | .20 | 20 | 1 | 0.00 |
| OBTAIN ACCESS TO NEW EXECUTABLE | QA SPECIALIST | .20 | 20 | 1 | 0.00 |
| COMPARE SDD CHANGES TO CHANGE REQUEST FORM | QA SPECIALIST | .50 | 25 | 2 | 0.00 |
| COMPARE CODE CHANGES TO SDD CHANGES | QA SPECIALIST | .50 | 25 | 2 | 0.00 |
| VERIFY ASSESSMENT OF OTHER AFFECTED DOCUMENTATION /CODE | QA SPECIALIST | .50 | 25 | 2 | 0.00 |
| VERIFY THAT CODE CHANGES CONFORM TO SLBM STANDARDS | QA SPECIALIST | .25 | 25 | 1 | 0.00 |
| MODIFIED SDD CLEARED FOR OFFICIAL DOCUMENTATION | QA SPECIALIST | .25 | 25 | 1 | 0.00 |
| MODIFIED CODE CLEARED FOR DELIVERY | QA SPECIALIST | .25 | 25 | 1 | 0.00 |
| MANUALLY WORK DESIGNED TEST CASES THROUGH SDD | QA SPECIALIST | 25 | 25 | 1 | 0.00 |
| MANUALLY WORK DESIGNED TEST CASES THROUGH CODE | QA SPECIALIST | .25 | 25 | 1 | 0.00 |
| LINK MODIFIED SUBROUTINES TO K52 INDEPENDENT DRIVER PROGRAM | QA SPECIALIST | .25 | 25 | 1 | 0.00 |
| EXECUTE MODIFIED PROGRAM WITH DESIGNED TEST CASES | QA SPECIALIST | .25 | 25 | 1 | 0.00 |
| EXECUTE MODIFIED PROGRAM WITH DELIVERY TEST CASES | QA SPECIALIST | .25 | 25 | 1 | 0.00 |
| COMPARE ESTIMATED RESULTS OF TEST CASE | QA SPECIALIST | .50 | 25 | 2 | 0.00 |
| COMPARE RESULTS | QA SPECIALIST | .50 | 25 | 2 | 0.00 |
| VERIFY THAT RESULTS ARE IDENTICAL | QA SPECIALIST | .50 | 25 | 2 | 0.00 |
| EXECUTE K52 MODEL WITH DESIGNED TEST CASES | QA SPECIALIST | .25 | 25 | 1 | 0.00 |
| EXECUTE K52 MODEL WITH DELIVERY TEST CASES | QA SPECIALIST | .25 | 25 | 1 | 0.00 |
| COMPARE RESULTS OF DESIGNED CASES TO EXPECTED | QA SPECIALIST | .50 | 25 | 2 | 0.00 |
| VERIFY DELIVERY CASE RESULTS WITHIN EXPECTED PARAMETERS | QA SPECIALIST | .50 | 25 | 2 | 0.00 |

Figure C14: URGMIN Validation and Verification Phase Resource Allocation

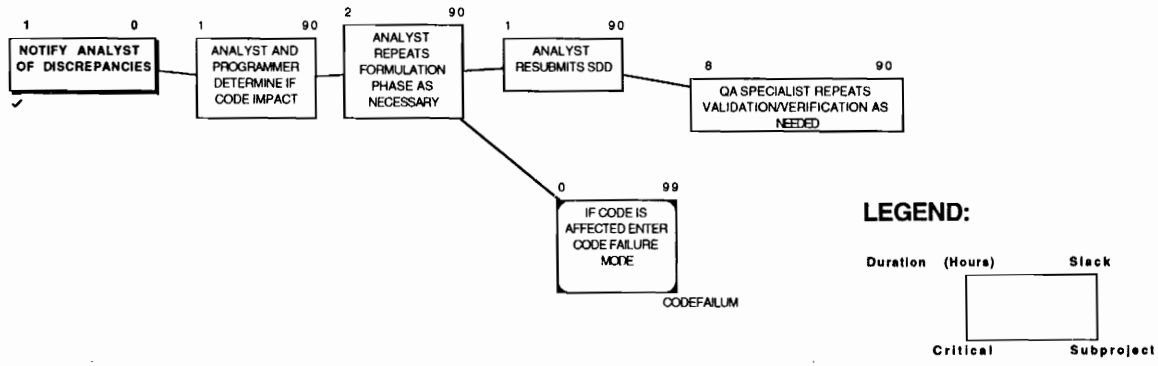



Figure C15: URGMIN SDD Failure Mode



| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| NOTIFY ANALYST OF DISCREPANCIES | QA SPECIALIST | .50 | 50 | 1 | 32.29 |
| ANALYST AND PROGRAMMER DETERMINE IF CODE IMPACT | ANALYST | .50 | 50 | 1 | 32.29 |
| ANALYST AND PROGRAMMER DETERMINE IF CODE IMPACT | PROGRAMMER | .50 | 50 | 1 | 32.29 |
| ANALYST REPEATS FORMULATION PHASE AS NECESSARY | ANALYST | 1 | 50 | 2 | 64.58 |
| ANALYST RESUBMITS SDD | ANALYST | .50 | 50 | 1 | 32.29 |
| QA SPECIALIST REPEATS VALIDATION/VERIFICATION AS NEEDED | QA SPECIALIST | 4 | 50 | 8 | 258.32 |

Figure C16: URGMIN SDD Failure Mode Resource Allocation

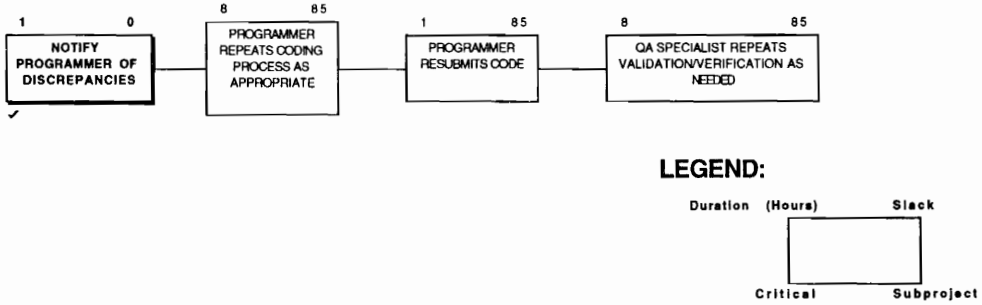



Figure C17: URGMIN Code Failure Mode



| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| NOTIFY PROGRAMMER OF DISCREPANCIES | QA SPECIALIST | .50 | 50 | 1 | 32.29 |
| PROGRAMMER REPEATS CODING PROCESS AS APPROPRIATE | PROGRAMMER | 4 | 50 | 8 | 258.32 |
| PROGRAMMER RESUBMITS CODE | PROGRAMMER | .50 | 50 | 1 | 32.29 |
| QA SPECIALIST REPEATS VALIDATION/VERIFICATION AS NEEDED | QA SPECIALIST | 4 | 50 | 8 | 258.32 |

Figure C18: URGMIN Code Failure Mode Resource Allocation

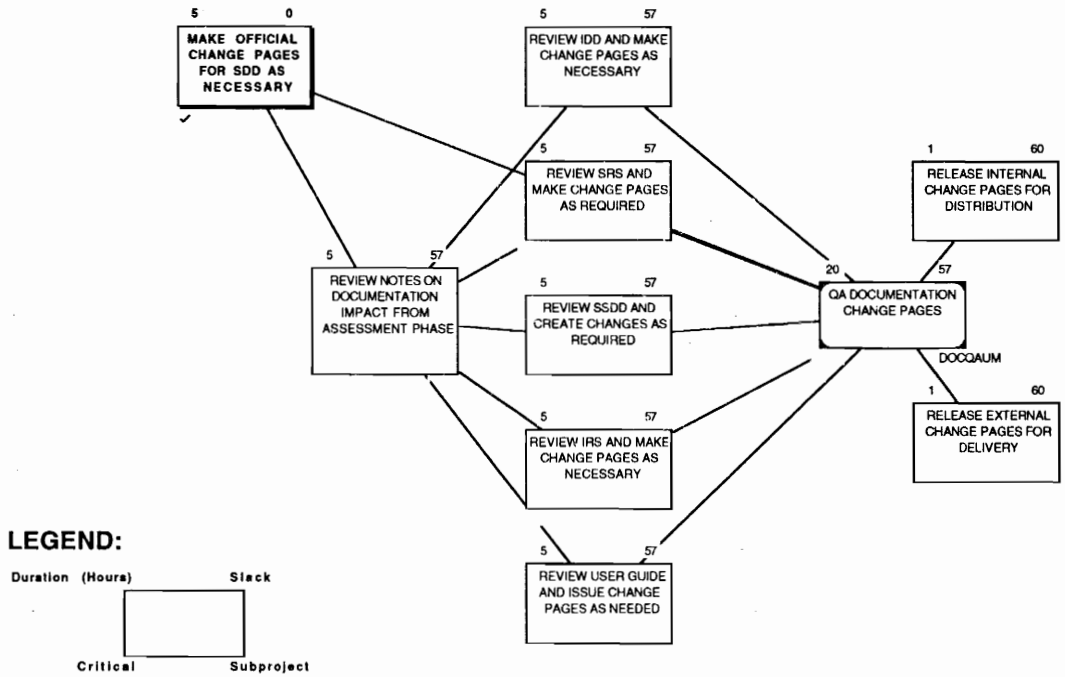



Figure C19: URGMIN Documentation Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| MAKE OFFICIAL CHANGE PAGES FOR SDD AS NECESSARY | ANALYST | 2 | 40 | 5 | 129.16 |
| REVIEW NOTES ON DOCUMENTATION IMPACT FROM ASSESSMENT | ANALYST | 2 | 40 | 5 | 129.16 |
| REVIEW IDD AND MAKE CHANGE PAGES AS NECESSARY | ANALYST | 2 | 40 | 5 | 129.16 |
| REVIEW SRS AND MAKE CHANGE PAGES AS REQUIRED | ANALYST | 2 | 40 | 5 | 129.16 |
| REVIEW SSDD AND CREATE CHANGES AS REQUIRED | ANALYST | 2 | 40 | 5 | 129.16 |
| REVIEW IRS AND MAKE CHANGE PAGES AS NECESSARY | ANALYST | 2 | 40 | 5 | 129.16 |
| REVIEW USER GUIDE AND ISSUE CHANGE PAGES AS NEEDED | ANALYST | 2 | 40 | 5 | 129.16 |
| QA DOCUMENTATION CHANGE PAGES | QA SPECIALIST | 8 | 40 | 20 | 516.64 |
| RELEASE INTERNAL CHANGE PAGES FOR DISTRIBUTION | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| RELEASE EXTERNAL CHANGE PAGES FOR DELIVERY | QA SPECIALIST | 1 | 100 | 1 | 64.58 |

Figure C20: URGMIN Documentation Phase Resource Allocation

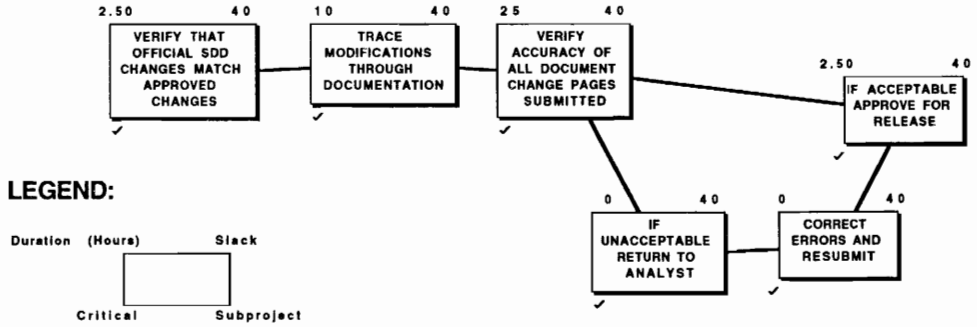



Figure C21: URGMIN Documentation Quality Assurance Mode

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| VERIFY THAT OFFICIAL SDD CHANGES MATCH APPROVED CHANGES | QA SPECIALIST | .50 | 20 | 2.50 | 32.29 |
| TRACE MODIFICATIONS THROUGH DOCUMENTATION | QA SPECIALIST | 2 | 20 | 10 | 129.16 |
| VERIFY ACCURACY OF ALL DOCUMENT CHANGE PAGES SUBMITTED | QA SPECIALIST | 5 | 20 | 25 | 322.90 |
| IF UNACCEPTABLE RETURN TO ANALYST | ANALYST | 0 | 100 | 0 | 0.00 |
| CORRECT ERRORS AND RESUBMIT | ANALYST | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE APPROVE FOR RELEASE | QA SPECIALIST | .50 | 20 | 2.50 | 32.29 |

Figure C22: URGMIN Documentation Quality Assurance Mode Resource Allocation

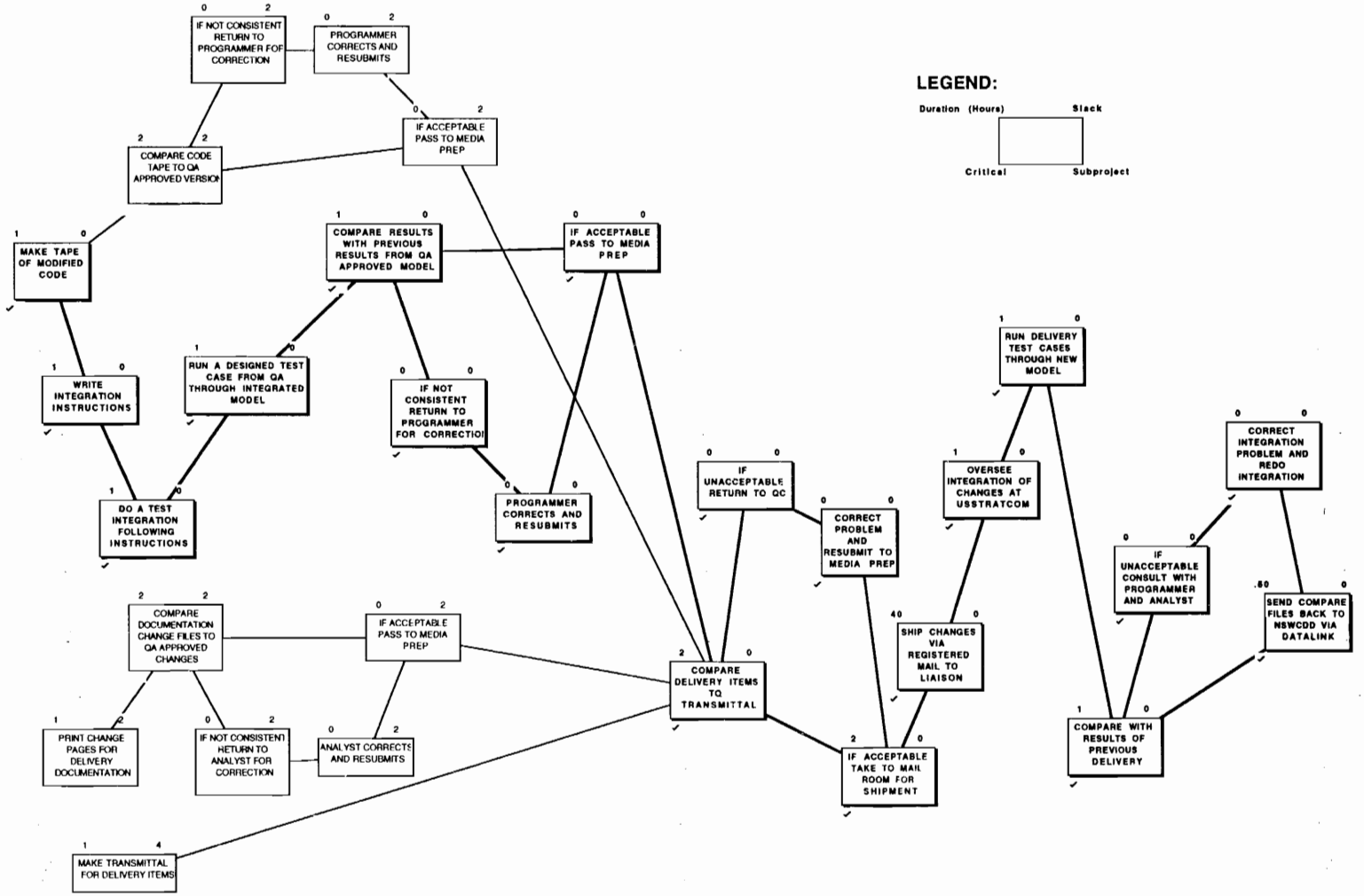



Figure C23: URGMIN Delivery Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| MAKE TAPE OF MODIFIED CODE | PROGRAMMER | .50 | 50 | 1 | 32.29 |
| PRINT CHANGE PAGES FOR DELIVERY DOCUMENTATION | ANALYST | .50 | 50 | 1 | 32.29 |
| WRITE INTEGRATION INSTRUCTIONS | PROGRAMMER | .50 | 50 | 1 | 32.29 |
| MAKE TRANSMITTAL FOR DELIVERY ITEMS | ANALYST | .50 | 50 | 1 | 32.29 |
| COMPARE CODE TAPE TO QA APPROVED VERSION | QC SPECIALIST | 1 | 50 | 2 | 55.28 |
| IF NOT CONSISTENT RETURN TO PROGRAMMER FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| PROGRAMMER CORRECTS AND RESUBMITS | PROGRAMMER | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE PASS TO MEDIA PREP | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| COMPARE DOCUMENTATION CHANGE FILES TO QA APPROVED CHANGES | QC SPECIALIST | 1 | 50 | 2 | 55.28 |
| IF NOT CONSISTENT RETURN TO ANALYST FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| ANALYST CORRECTS AND RESUBMITS | ANALYST | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE PASS TO MEDIA PREP | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| DO A TEST INTEGRATION FOLLOWING INSTRUCTIONS | QC SPECIALIST | .50 | 50 | 1 | 27.64 |
| RUN A DESIGNED TEST CASE FROM QA THROUGH INTEGRATED MODEL | QC SPECIALIST | .50 | 50 | 1 | 27.64 |
| COMPARE RESULTS WITH PREVIOUS RESULTS FROM QA APPROVED MODEL | QC SPECIALIST | .50 | 50 | 1 | 27.64 |
| IF NOT CONSISTENT RETURN TO PROGRAMMER FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| PROGRAMMER CORRECTS AND RESUBMITS | PROGRAMMER | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE PASS TO MEDIA PREP | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| COMPARE DELIVERY ITEMS TO TRANSMITTAL | MEDIA PREP SPECIALIST | 1 | 50 | 2 | 0.00 |
| IF UNACCEPTABLE RETURN TO QC | MEDIA PREP SPECIALIST | 0 | 100 | 0 | 0.00 |
| CORRECT PROBLEM AND RESUBMIT TO MEDIA PREP | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE TAKE TO MAIL ROOM FOR SHIPMENT | MEDIA PREP SPECIALIST | 1 | 50 | 2 | 0.00 |
| SHIP CHANGES VIA REGISTERED MAIL TO LIAISON | SHIPPING | 40 | 100 | 40 | 0.00 |
| OVERSEE INTEGRATION OF CHANGES AT USSTRATCOM | LIAISON | .50 | 50 | 1 | 32.29 |
| RUN DELIVERY TEST CASES THROUGH NEW MODEL | LIAISON | .50 | 50 | 1 | 32.29 |
| COMPARE WITH RESULTS OF PREVIOUS DELIVERY | LIAISON | .50 | 50 | 1 | 32.29 |
| IF UNACCEPTABLE CONSULT WITH PROGRAMMER AND ANALYST | LIAISON | 0 | 100 | 0 | 0.00 |
| CORRECT INTEGRATION PROBLEM AND REDO INTEGRATION | LIAISON | 0 | 100 | 0 | 0.00 |
| SEND COMPARE FILES BACK TO NSWCDD VIA DATALINK | LIAISON | .50 | 100 | .50 | 32.29 |



Figure C24: URGMIN Delivery Phase Resource Allocation

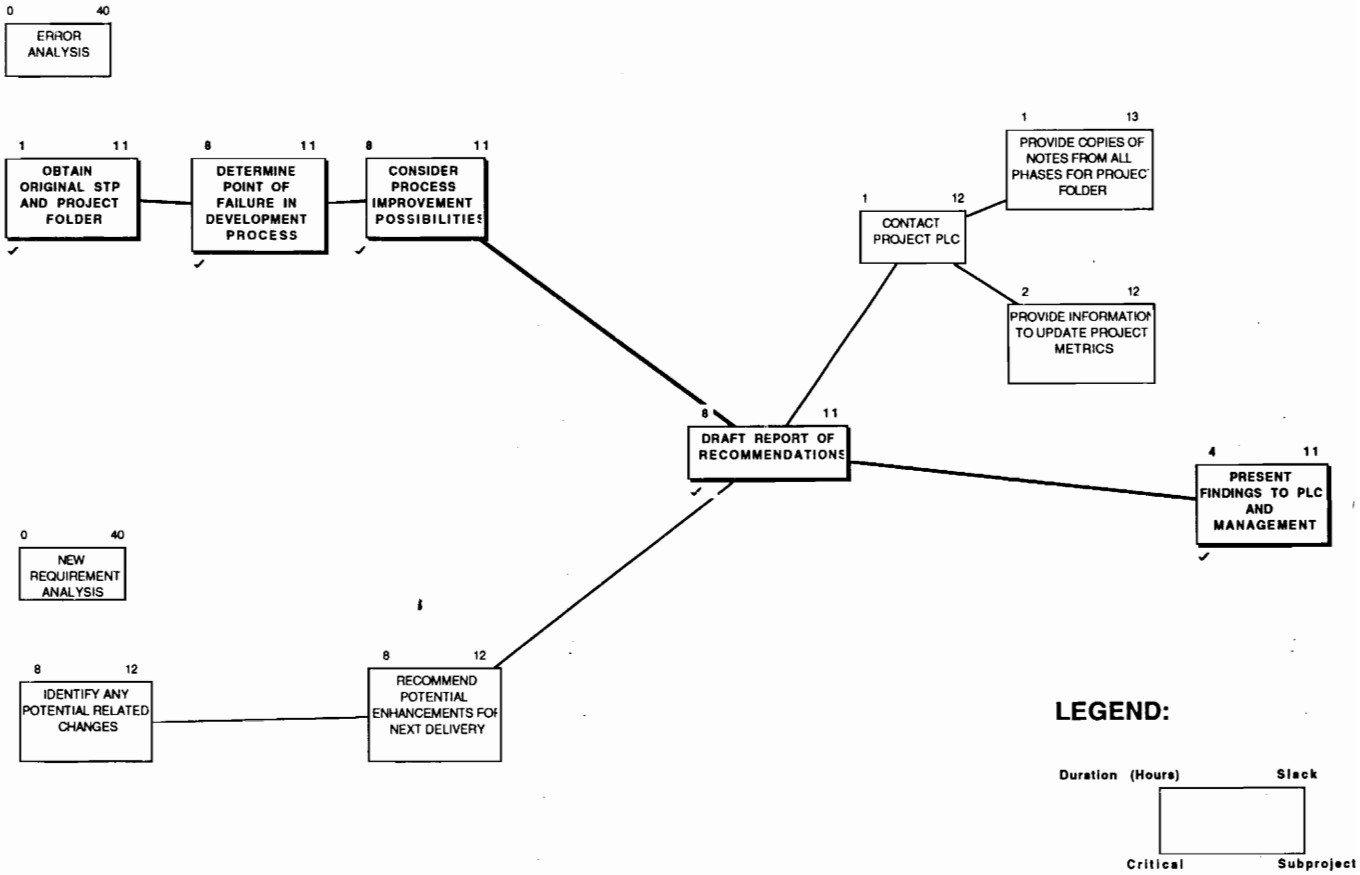



Figure C25: URGMIN Process Analysis Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | ANALYST | 1 | 100 | 1 | 64.58 |
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | ANALYST | 8 | 100 | 8 | 516.64 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | ANALYST | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | ANALYST | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | ANALYST | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | ANALYST | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| CONTACT PROJECT PLC | ANALYST | 1 | 100 | 1 | 64.58 |
| CONTACT PROJECT PLC | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| CONTACT PROJECT PLC | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | ANALYST | 2 | 100 | 2 | 129.16 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | QA SPECIALIST | 2 | 100 | 2 | 129.16 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | ANALYST | 1 | 100 | 1 | 64.58 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | ANALYST | 4 | 100 | 4 | 258.32 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | QA SPECIALIST | 4 | 100 | 4 | 258.32 |



Figure C26: URGMIN Process Analysis Phase Resource Allocation

# APPENDIX D: URGINTER PROCESS

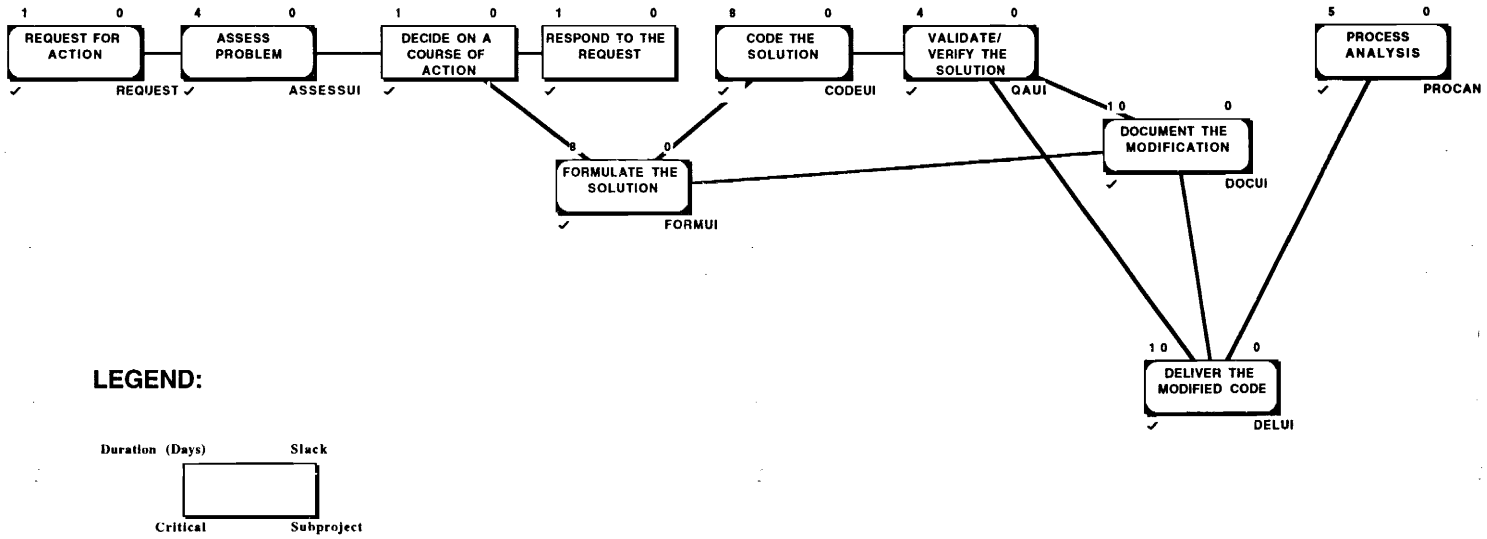



Figure D1: URGINTER Process

URGINTER
4/7/95
Project Overview

| | |
|---|---|
| • Tasks and Activities | |
| Tasks | 2 |
| Milestones | 0 |
| Supertasks | 8 |
| Completed Tasks | 0 |
| Completed Milestones | 0 |
| Summary Activities | 0 |
| • Project Dates | |
| Project Start Date | 7/1/95 |
| Project Finish Date | 8/30/95 |
| Actual Start | 7/3/95 |
| Actual Finish | 8/30/95 |
| • Project Costs | |
| Total Planned Cost | 44,916.70 |
| Total Planned Income | 0.00 |
| Total Actual Cost | 0.00 |
| Total Actual Income | 0.00 |
| • Other | |
| Projects in Family Tree | 13 |
| Projects in Resource Scope | 0 |
| Project % Done | 0 |
| Number of Resources | 10 |
| Number of Calendars | 1 |

Figure D2: URGINTER Project Overview

| Name | Subproject | # Resources | Duration (Days) | Slack | Total Cost |
|---|---|---|---|---|---|
| REQUEST FOR ACTION | REQUEST | 4 | 1 | 0 | 1,725.75 |
| ASSESS PROBLEM | ASSESSUI | 3 | 4 | 0 | 4,776.66 |
| DECIDE ON A COURSE OF ACTION | | 3 | 1 | 0 | 505.92 |
| RESPOND TO THE REQUEST | | 1 | 1 | 0 | 168.64 |
| FORMULATE THE SOLUTION | FORMUI | 1 | 8 | 0 | 4,133.12 |
| CODE THE SOLUTION | CODEUI | 1 | 8 | 0 | 4,133.12 |
| VALIDATE/ VERIFY THE SOLUTION | QAUI | 1 | 4 | 0 | 4,197.70 |
| DOCUMENT THE MODIFICATION | DOCUI | 2 | 10 | 0 | 6,361.13 |
| DELIVER THE MODIFIED CODE | DELUI | 6 | 10 | 0 | 1,671.80 |
| PROCESS ANALYSIS | PROCAN | 3 | 5 | 0 | 17,242.86 |

Figure D3: URGINTER Total Resource Allocation

| Name | Resource | Work-Days | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REQUEST FOR ACTION | LIAISON | .50 | 50 | 1 | 258.32 |
| REQUEST FOR ACTION | K42 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| REQUEST FOR ACTION | K51 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| REQUEST FOR ACTION | K52 BRANCH HEAD | .50 | 50 | 1 | 337.28 |
| ASSESS PROBLEM | ANALYST | 2 | 50 | 4 | 1,033.28 |
| ASSESS PROBLEM | PROGRAMMER | 2 | 50 | 4 | 1,033.28 |
| ASSESS PROBLEM | LIAISON | .25 | 50 | .50 | 129.16 |
| DECIDE ON A COURSE OF ACTION | K42 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| DECIDE ON A COURSE OF ACTION | K51 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| DECIDE ON A COURSE OF ACTION | K52 BRANCH HEAD | .25 | 25 | 1 | 168.64 |
| RESPOND TO THE REQUEST | K42 BRANCH HEAD | .25 | 50 | .50 | 168.64 |
| FORMULATE THE SOLUTION | ANALYST | 4 | 50 | 8 | 2,066.56 |
| CODE THE SOLUTION | PROGRAMMER | 4 | 50 | 8 | 2,066.56 |
| VALIDATE/ VERIFY THE SOLUTION | QA SPECIALIST | 2 | 50 | 4 | 1,033.28 |
| DOCUMENT THE MODIFICATION | ANALYST | 3 | 50 | 6 | 1,549.92 |
| DOCUMENT THE MODIFICATION | QA SPECIALIST | 2.12 | 50 | 4.25 | 1,097.86 |
| DELIVER THE MODIFIED CODE | ANALYST | .25 | 50 | .50 | 129.16 |
| DELIVER THE MODIFIED CODE | QC SPECIALIST | 1 | 50 | 2 | 442.24 |
| DELIVER THE MODIFIED CODE | PROGRAMMER | .25 | 50 | .50 | 129.16 |
| DELIVER THE MODIFIED CODE | LIAISON | .50 | 50 | 1 | 258.32 |
| DELIVER THE MODIFIED CODE | MEDIA PREP SPECIALIST | .50 | 50 | 1 | 221.12 |
| DELIVER THE MODIFIED CODE | SHIPPING | 5 | 100 | 5 | 40.00 |
| PROCESS ANALYSIS | ANALYST | 5 | 100 | 5 | 2,583.20 |
| PROCESS ANALYSIS | PROGRAMMER | 5 | 100 | 5 | 2,583.20 |
| PROCESS ANALYSIS | QA SPECIALIST | 5 | 100 | 5 | 2,583.20 |

Figure D4: URGINTER Individual Resource Allocation

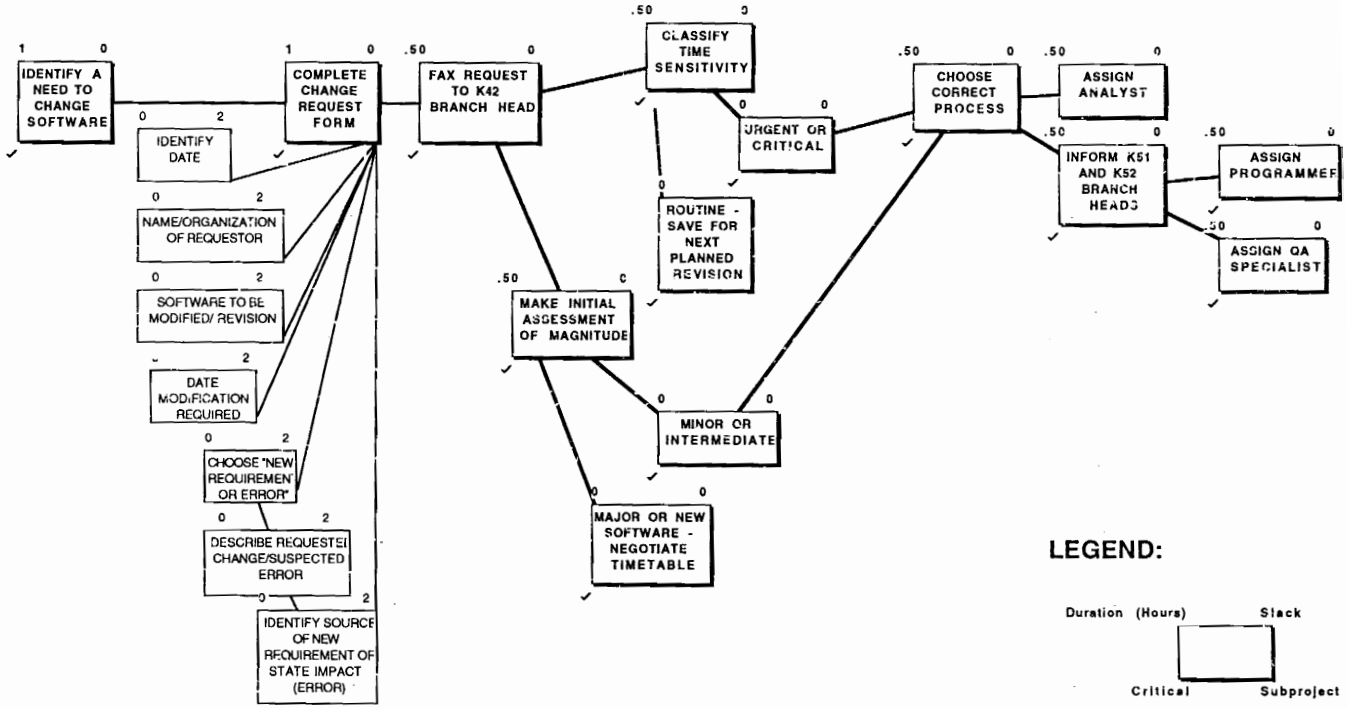
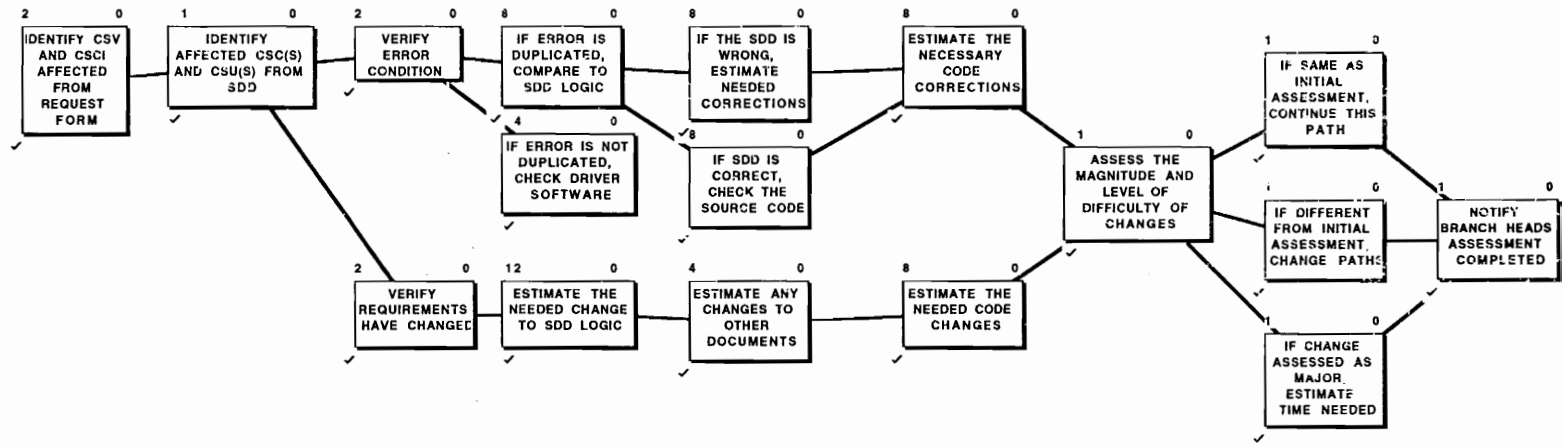



Figure D5: URGINTER Request Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| IDENTIFY A NEED TO CHANGE SOFTWARE | LIAISON | 1 | 100 | 1 | 64.58 |
| COMPLETE CHANGE REQUEST FORM | LIAISON | 1 | 100 | 1 | 64.58 |
| FAX REQUEST TO K42 BRANCH HEAD | LIAISON | .50 | 100 | .50 | 32.29 |
| CLASSIFY TIME SENSITIVITY | K42 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| MAKE INITIAL ASSESSMENT OF MAGNITUDE | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| CHOOSE CORRECT PROCESS | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| ASSIGN ANALYST | K42 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| INFORM K51 AND K52 BRANCH HEADS | K42 BRANCH HEAD | .50 | 100 | .50 | 41.83 |
| ASSIGN PROGRAMMER | K51 BRANCH HEAD | .50 | 100 | .50 | 42.16 |
| ASSIGN QA SPECIALIST | K52 BRANCH HEAD | .50 | 100 | .50 | 42.16 |

Figure D6: URGINTER Request Phase Resource Allocation




**LEGEND:**

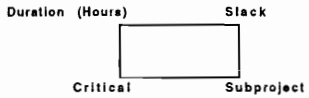



Figure D7: URGINTER Assessment Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| IDENTIFY CSV AND CSCI AFFECTED FROM REQUEST FORM | ANALYST | 1 | 50 | 2 | 64.58 |
| IDENTIFY CSV AND CSCI AFFECTED FROM REQUEST FORM | PROGRAMMER | 1 | 50 | 2 | 64.58 |
| IDENTIFY AFFECTED CSC(S) AND CSU(S) FROM SDD | ANALYST | .50 | 50 | .99 | 32.04 |
| IDENTIFY AFFECTED CSC(S) AND CSU(S) FROM SDD | PROGRAMMER | .50 | 50 | .99 | 32.04 |
| VERIFY ERROR CONDITION BY RUNNING A TEST CASE | ANALYST | 1 | 50 | 2 | 64.58 |
| IF ERROR IS NOT DUPLICATED, CHECK DRIVER SOFTWARE | LIAISON | 2 | 50 | 4 | 129.16 |
| IF ERROR IS DUPLICATED, COMPARE TO SDD LOGIC | ANALYST | 4 | 50 | 8 | 258.32 |
| IF THE SDD IS WRONG, ESTIMATE NEEDED CORRECTIONS | ANALYST | 4 | 50 | 8 | 258.32 |
| IF SDD IS CORRECT, CHECK THE SOURCE CODE | PROGRAMMER | 4 | 50 | 8 | 258.32 |
| ESTIMATE THE NECESSARY CODE CORRECTIONS | PROGRAMMER | 4 | 50 | 8 | 258.32 |
| VERIFY REQUIREMENTS HAVE CHANGED | ANALYST | 1 | 50 | 2 | 64.58 |
| ESTIMATE THE NEEDED CHANGE TO SDD LOGIC | ANALYST | 6 | 50 | 12 | 387.48 |
| ESTIMATE ANY CHANGES TO OTHER DOCUMENTS | ANALYST | 2 | 50 | 4 | 129.16 |
| ESTIMATE THE NEEDED CODE CHANGES | PROGRAMMER | 4 | 50 | 8 | 258.32 |
| ASSESS THE MAGNITUDE AND LEVEL OF DIFFICULTY OF CHANGES | ANALYST | .50 | 50 | .99 | 32.04 |
| ASSESS THE MAGNITUDE AND LEVEL OF DIFFICULTY OF CHANGES | PROGRAMMER | .50 | 50 | .99 | 32.04 |
| IF SAME AS INITIAL ASSESSMENT, CONTINUE THIS PATH | ANALYST | .50 | 50 | 1 | 32.29 |
| IF SAME AS INITIAL ASSESSMENT, CONTINUE THIS PATH | PROGRAMMER | .50 | 50 | .99 | 32.04 |
| IF DIFFERENT FROM INITIAL ASSESSMENT, CHANGE PATHS | ANALYST | .50 | 50 | .99 | 32.04 |
| IF DIFFERENT FROM INITIAL ASSESSMENT, CHANGE PATHS | PROGRAMMER | .50 | 50 | .99 | 32.04 |
| IF CHANGE ASSESSED AS MAJOR, ESTIMATE TIME NEEDED | ANALYST | .50 | 50 | .99 | 32.04 |
| IF CHANGE ASSESSED AS MAJOR, ESTIMATE TIME NEEDED | PROGRAMMER | .50 | 50 | .99 | 32.04 |
| NOTIFY BRANCH HEADS ASSESSMENT COMPLETED | ANALYST | .50 | 50 | 1 | 32.29 |
| NOTIFY BRANCH HEADS ASSESSMENT COMPLETED | PROGRAMMER | .50 | 50 | 1 | 32.29 |

Figure D8: URGINTER Assessment Phase Resource Allocation

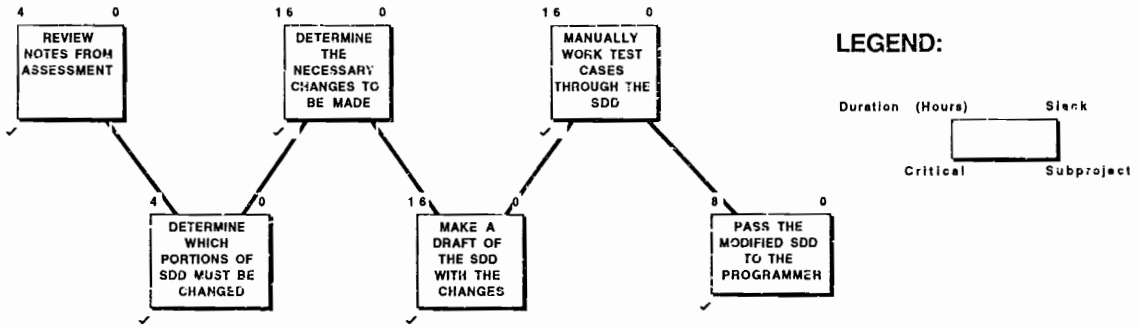



Figure D9: URGINTER Formulation Phase



| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REVIEW NOTES FROM ASSESSMENT | ANALYST | 2 | 50 | 4 | 129.16 |
| DETERMINE WHICH PORTIONS OF SDD MUST BE CHANGED | ANALYST | 2 | 50 | 4 | 129.16 |
| DETERMINE THE NECESSARY CHANGES TO BE MADE | ANALYST | 8 | 50 | 16 | 516.64 |
| MAKE A DRAFT OF THE SDD WITH THE CHANGES | ANALYST | 8 | 50 | 16 | 516.64 |
| MANUALLY WORK TEST CASES THROUGH THE SDD | ANALYST | 8 | 50 | 16 | 516.64 |
| PASS THE MODIFIED SDD TO THE PROGRAMMER | ANALYST | 4 | 50 | 8 | 258.32 |

Figure D10: URGINTER Formulation Phase Resource Allocation

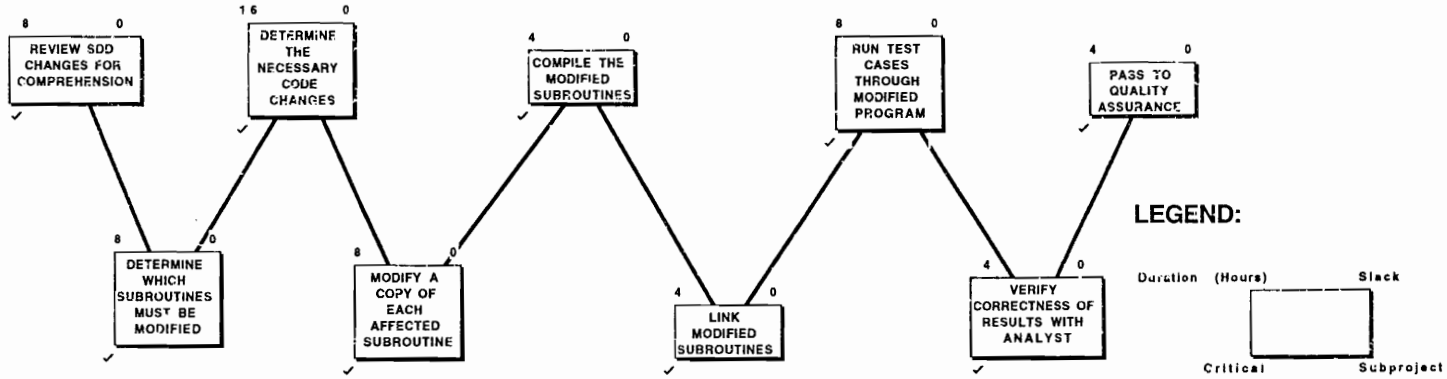



Figure D11: URGINTER Coding Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| REVIEW SDD CHANGES FOR COMPREHENSION | PROGRAMMER | 4 | 50 | 8 | 258.32 |
| DETERMINE WHICH SUBROUTINES MUST BE MODIFIED | PROGRAMMER | 4 | 50 | 8 | 258.32 |
| DETERMINE THE NECESSARY CODE CHANGES | PROGRAMMER | 8 | 50 | 16 | 516.64 |
| MODIFY A COPY OF EACH AFFECTED SUBROUTINE | PROGRAMMER | 4 | 50 | 8 | 258.32 |
| COMPILE THE MODIFIED SUBROUTINES | PROGRAMMER | 2 | 50 | 4 | 129.16 |
| LINK MODIFIED SUBROUTINES WITH COPY OF REST OF CODE | PROGRAMMER | 2 | 50 | 4 | 129.16 |
| RUN TEST CASES THROUGH MODIFIED PROGRAM | PROGRAMMER | 4 | 50 | 8 | 258.32 |
| VERIFY CORRECTNESS OF RESULTS WITH ANALYST | PROGRAMMER | 2 | 50 | 4 | 129.16 |
| PASS TO QUALITY ASSURANCE SPECIALIST | PROGRAMMER | 2 | 50 | 4 | 129.16 |

Figure D12: URGINTER Coding Phase Resource Allocation

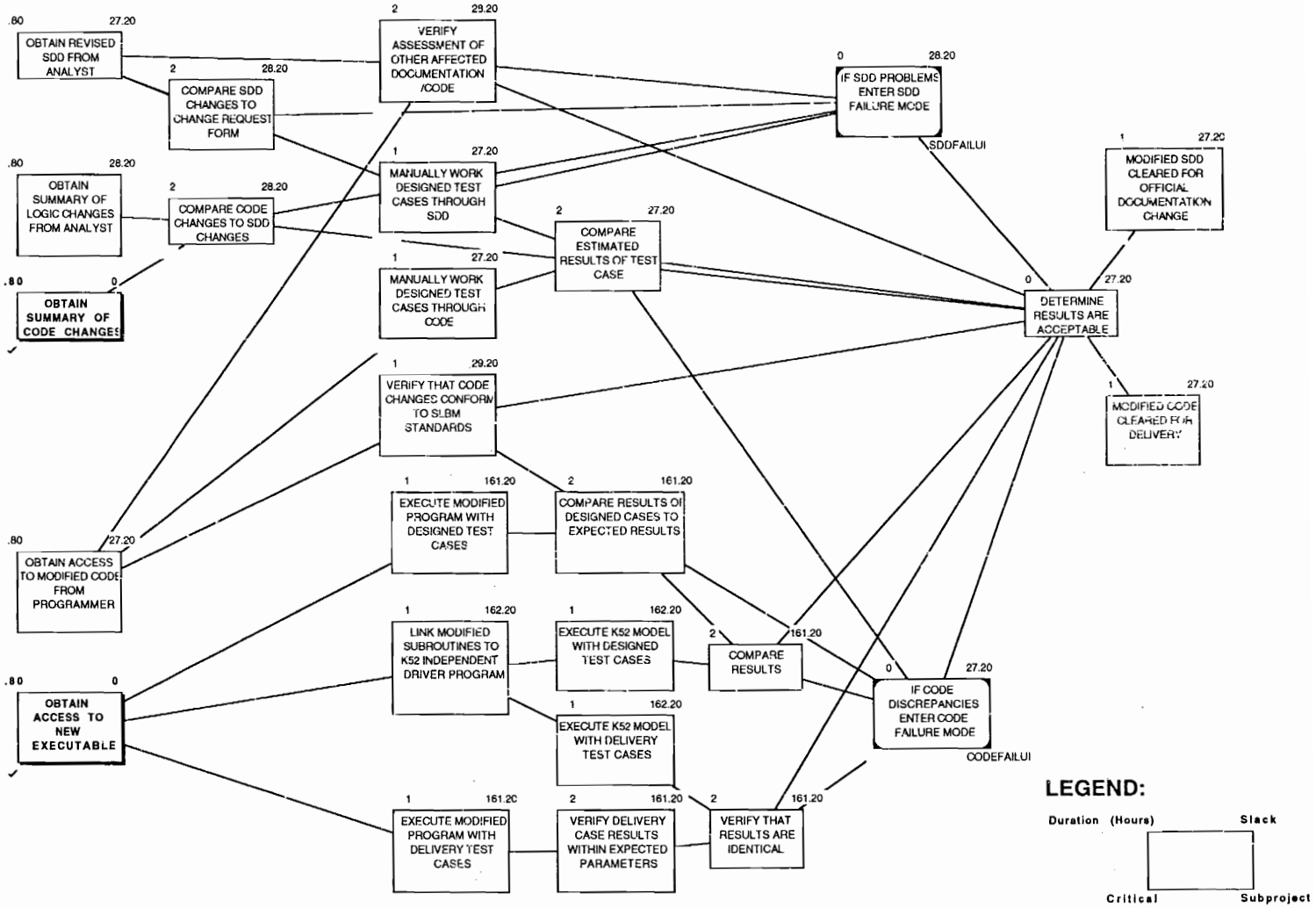



Figure D13: URGINTER Validation and Verification Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| OBTAIN REVISED SDD FROM ANALYST | QA SPECIALIST | .40 | 50 | .80 | 0.00 |
| OBTAIN SUMMARY OF LOGIC CHANGES FROM ANALYST | QA SPECIALIST | .40 | 50 | .80 | 0.00 |
| OBTAIN ACCESS TO MODIFIED CODE FROM PROGRAMMER | QA SPECIALIST | .40 | 50 | .80 | 0.00 |
| **OBTAIN ACCESS TO NEW EXECUTABLE** | **QA SPECIALIST** | **.40** | **50** | **.80** | **0.00** |
| **OBTAIN SUMMARY OF CODE CHANGES** | **QA SPECIALIST** | **.40** | **50** | **.80** | **0.00** |
| COMPARE SDD CHANGES TO CHANGE REQUEST FORM | QA SPECIALIST | 1 | 50 | 2 | 0.00 |
| COMPARE CODE CHANGES TO SDD CHANGES | QA SPECIALIST | 1 | 50 | 2 | 0.00 |
| VERIFY ASSESSMENT OF OTHER AFFECTED DOCUMENTATION/CODE | QA SPECIALIST | 1 | 50 | 2 | 0.00 |
| MANUALLY WORK DESIGNED TEST CASES THROUGH SDD | QA SPECIALIST | .50 | 50 | 1 | 0.00 |
| MANUALLY WORK DESIGNED TEST CASES THROUGH CODE | QA SPECIALIST | .50 | 50 | 1 | 0.00 |
| COMPARE ESTIMATED RESULTS OF TEST CASE | QA SPECIALIST | 1 | 50 | 2 | 0.00 |
| VERIFY THAT CODE CHANGES CONFORM TO SLBM STANDARDS | QA SPECIALIST | .50 | 50 | 1 | 0.00 |
| **EXECUTE MODIFIED PROGRAM WITH DESIGNED TEST CASES** | **QA SPECIALIST** | **.50** | **50** | **1** | **0.00** |
| **COMPARE RESULTS OF DESIGNED CASES TO EXPECTED** | **QA SPECIALIST** | **1** | **50** | **2** | **0.00** |
| **EXECUTE MODIFIED PROGRAM WITH DELIVERY TEST CASES** | **QA SPECIALIST** | **.50** | **50** | **1** | **0.00** |
| **VERIFY DELIVERY CASE RESULTS WITHIN EXPECTED** | **QA SPECIALIST** | **1** | **50** | **2** | **0.00** |
| LINK MODIFIED SUBROUTINES TO K52 INDEPENDENT DRIVER PROGRAM | QA SPECIALIST | .50 | 50 | 1 | 0.00 |
| EXECUTE K52 MODEL WITH DESIGNED TEST CASES | QA SPECIALIST | .50 | 50 | 1 | 0.00 |
| **COMPARE RESULTS** | **QA SPECIALIST** | **1** | **50** | **2** | **0.00** |
| EXECUTE K52 MODEL WITH DELIVERY TEST CASES | QA SPECIALIST | .50 | 50 | 1 | 0.00 |
| **VERIFY THAT RESULTS ARE IDENTICAL** | **QA SPECIALIST** | **1** | **50** | **2** | **0.00** |
| **MODIFIED CODE CLEARED FOR DELIVERY** | **QA SPECIALIST** | **.50** | **50** | **1** | **0.00** |
| **MODIFIED SDD CLEARED FOR OFFICIAL DOCUMENTATION CHANGE** | **QA SPECIALIST** | **.50** | **50** | **1** | **0.00** |

Figure D14: URGINTER Validation and Verification Resource Allocation Phase

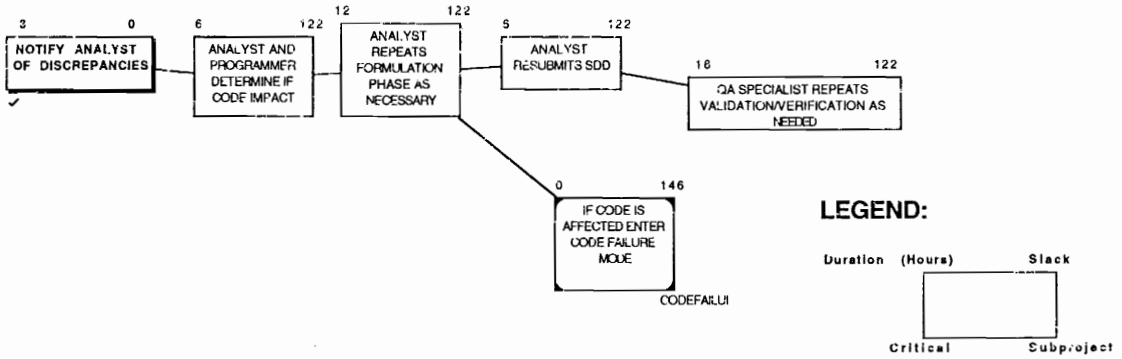



Figure D15: URGINTER SDD Failure Mode



| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| NOTIFY ANALYST OF DISCREPANCIES | QA SPECIALIST | 1 | 34 | 2.94 | 64.58 |
| ANALYST AND PROGRAMMER DETERMINE IF CODE IMPACT | ANALYST | 2 | 34 | 5.88 | 129.16 |
| ANALYST AND PROGRAMMER DETERMINE IF CODE IMPACT | PROGRAMMER | 2 | 34 | 5.88 | 129.16 |
| ANALYST REPEATS FORMULATION PHASE AS NECESSARY | ANALYST | 4 | 34 | 11.76 | 258.32 |
| ANALYST RESUBMITS SDD | ANALYST | 2 | 34 | 5.88 | 129.16 |
| QA SPECIALIST REPEATS VALIDATION/VERIFICATION AS NEEDED | QA SPECIALIST | 6 | 34 | 17.65 | 387.48 |

Figure D16: URGINTER SDD Failure Mode Resource Allocation

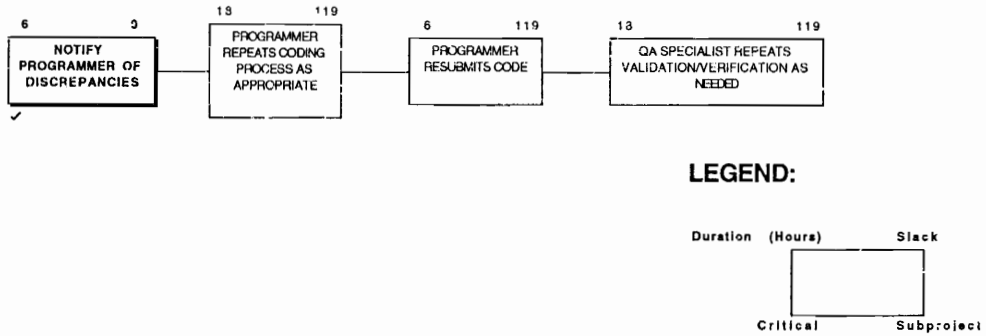



Figure D17: URGINTER Code Failure Mode



| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| NOTIFY PROGRAMMER OF DISCREPANCIES | QA SPECIALIST | 2 | 34 | 5.88 | 129.16 |
| PROGRAMMER REPEATS CODING PROCESS AS APPROPRIATE | PROGRAMMER | 6 | 34 | 17.65 | 387.48 |
| PROGRAMMER RESUBMITS CODE | PROGRAMMER | 2 | 34 | 5.88 | 129.16 |
| QA SPECIALIST REPEATS VALIDATION/VERIFICATION AS NEEDED | QA SPECIALIST | 6 | 34 | 17.65 | 387.48 |

Figure D18: URGINTER Code Failure Mode Resource Allocation

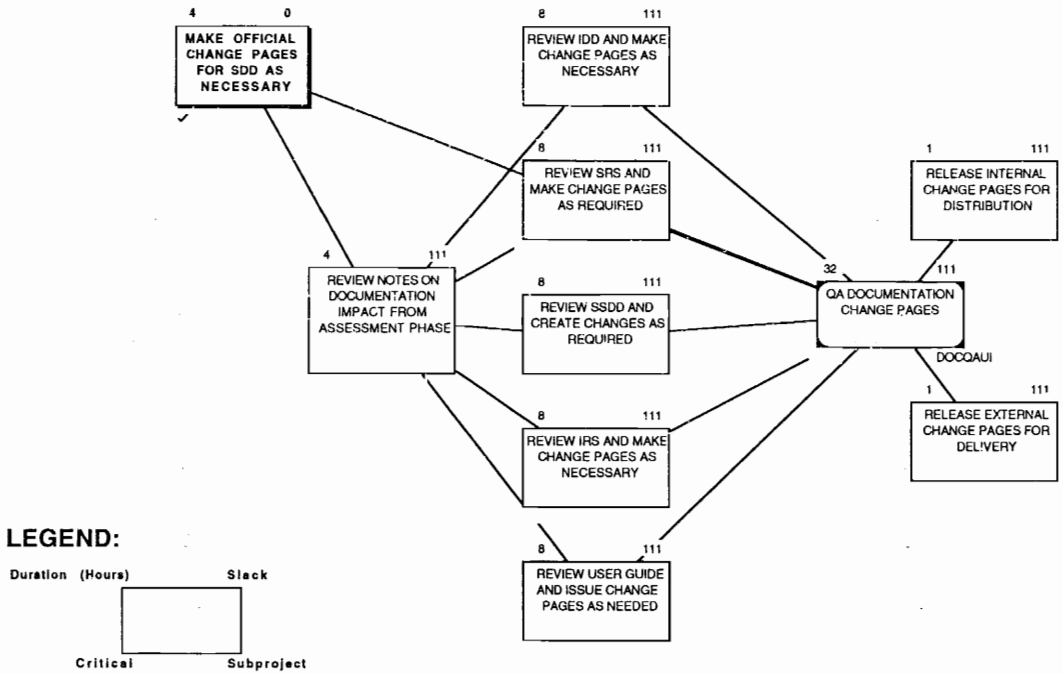



Figure D19: URGINTER Documentation Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| MAKE OFFICIAL CHANGE PAGES FOR SDD AS NECESSARY | ANALYST | 2 | 50 | 4 | 129.16 |
| REVIEW NOTES ON DOCUMENTATION IMPACT FROM ASSESSMENT | ANALYST | 2 | 50 | 4 | 129.16 |
| REVIEW IDD AND MAKE CHANGE PAGES AS NECESSARY | ANALYST | 4 | 50 | 8 | 258.32 |
| REVIEW SRS AND MAKE CHANGE PAGES AS REQUIRED | ANALYST | 4 | 50 | 8 | 258.32 |
| REVIEW SSDD AND CREATE CHANGES AS REQUIRED | ANALYST | 4 | 50 | 8 | 258.32 |
| REVIEW IRS AND MAKE CHANGE PAGES AS NECESSARY | ANALYST | 4 | 50 | 8 | 258.32 |
| REVIEW USER GUIDE AND ISSUE CHANGE PAGES AS NEEDED | ANALYST | 4 | 50 | 8 | 258.32 |
| QA DOCUMENTATION CHANGE PAGES | QA SPECIALIST | 16 | 50 | 32 | 1,033.28 |
| RELEASE INTERNAL CHANGE PAGES FOR DISTRIBUTION | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| RELEASE EXTERNAL CHANGE PAGES FOR DELIVERY | QA SPECIALIST | 1 | 100 | 1 | 64.58 |

Figure D20: URGINTER Documentation Phase Resource Allocation

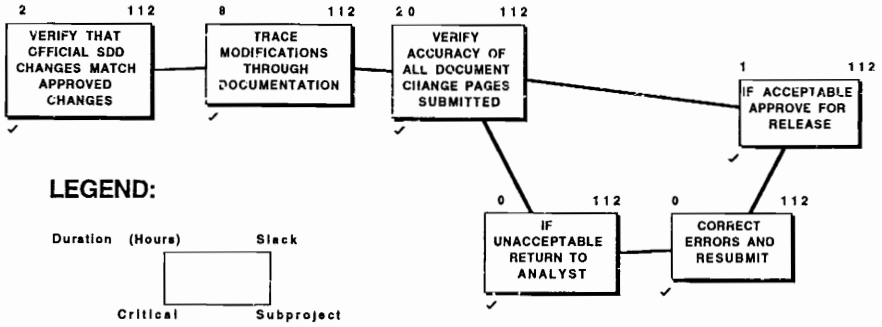



Figure D21: URGINTER Documentation Quality Assurance Mode



| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| VERIFY THAT OFFICIAL SDD CHANGES MATCH APPROVED CHANGES | QA SPECIALIST | 1 | 50 | 2 | 64.58 |
| TRACE MODIFICATIONS THROUGH DOCUMENTATION | QA SPECIALIST | 4 | 50 | 8 | 258.32 |
| VERIFY ACCURACY OF ALL DOCUMENT CHANGE PAGES SUBMITTED | QA SPECIALIST | 10 | 50 | 20 | 645.80 |
| IF UNACCEPTABLE RETURN TO ANALYST | ANALYST | 0 | 100 | 0 | 0.00 |
| CORRECT ERRORS AND RESUBMIT | ANALYST | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE APPROVE FOR RELEASE | QA SPECIALIST | .50 | 50 | 1 | 32.29 |

Figure D22: URGINTER Documentation Quality Assurance Mode Resource Allocation

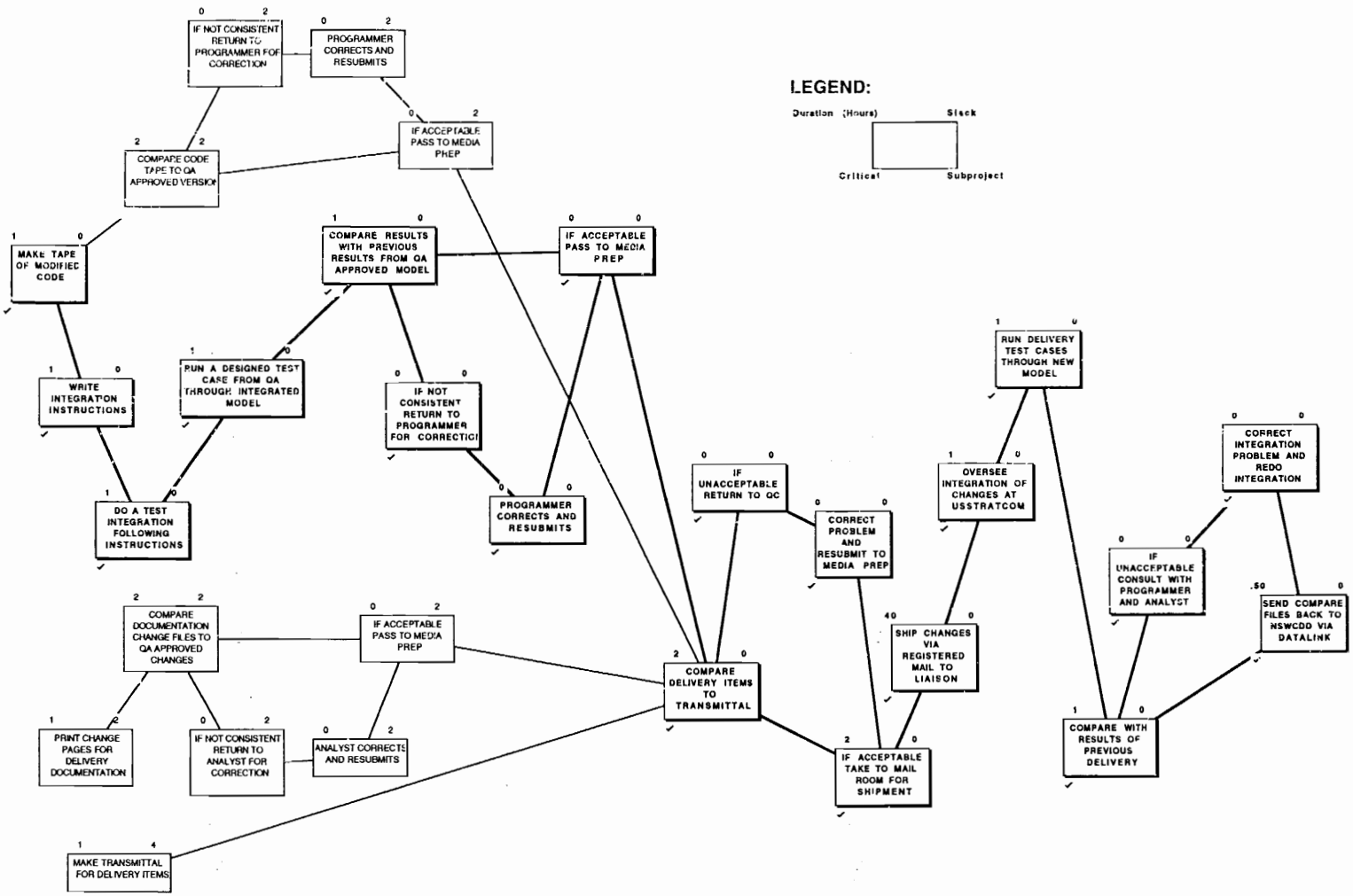



Figure D23: URGINTER Delivery Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| MAKE TAPE OF MODIFIED CODE | PROGRAMMER | .50 | 50 | 1 | 32.29 |
| PRINT CHANGE PAGES FOR DELIVERY DOCUMENTATION | ANALYST | .50 | 50 | 1 | 32.29 |
| WRITE INTEGRATION INSTRUCTIONS | PROGRAMMER | .50 | 50 | 1 | 32.29 |
| MAKE TRANSMITTAL FOR DELIVERY ITEMS | ANALYST | .50 | 50 | 1 | 32.29 |
| COMPARE CODE TAPE TO QA APPROVED VERSION | QC SPECIALIST | 1 | 50 | 2 | 55.28 |
| IF NOT CONSISTENT RETURN TO PROGRAMMER FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| PROGRAMMER CORRECTS AND RESUBMITS | PROGRAMMER | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE PASS TO MEDIA PREP | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| COMPARE DOCUMENTATION CHANGE FILES TO QA APPROVED CHANGES | QC SPECIALIST | 1 | 50 | 2 | 55.28 |
| IF NOT CONSISTENT RETURN TO ANALYST FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| ANALYST CORRECTS AND RESUBMITS | ANALYST | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE PASS TO MEDIA PREP | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| DO A TEST INTEGRATION FOLLOWING INSTRUCTIONS | QC SPECIALIST | .50 | 50 | 1 | 27.64 |
| RUN A DESIGNED TEST CASE FROM QA THROUGH INTEGRATED MODEL | QC SPECIALIST | .50 | 50 | 1 | 27.64 |
| COMPARE RESULTS WITH PREVIOUS RESULTS FROM QA APPROVED MODEL | QC SPECIALIST | .50 | 50 | 1 | 27.64 |
| IF NOT CONSISTENT RETURN TO PROGRAMMER FOR CORRECTION | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| PROGRAMMER CORRECTS AND RESUBMITS | PROGRAMMER | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE PASS TO MEDIA PREP | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| COMPARE DELIVERY ITEMS TO TRANSMITTAL | MEDIA PREP SPECIALIST | 1 | 50 | 2 | 0.00 |
| IF UNACCEPTABLE RETURN TO QC | MEDIA PREP SPECIALIST | 0 | 100 | 0 | 0.00 |
| CORRECT PROBLEM AND RESUBMIT TO MEDIA PREP | QC SPECIALIST | 0 | 100 | 0 | 0.00 |
| IF ACCEPTABLE TAKE TO MAIL ROOM FOR SHIPMENT | MEDIA PREP SPECIALIST | 1 | 50 | 2 | 0.00 |
| SHIP CHANGES VIA REGISTERED MAIL TO LIAISON | SHIPPING | 40 | 100 | 40 | 0.00 |
| OVERSEE INTEGRATION OF CHANGES AT USSTRATCOM | LIAISON | .50 | 50 | 1 | 32.29 |
| RUN DELIVERY TEST CASES THROUGH NEW MODEL | LIAISON | .50 | 50 | 1 | 32.29 |
| COMPARE WITH RESULTS OF PREVIOUS DELIVERY | LIAISON | .50 | 50 | 1 | 32.29 |
| IF UNACCEPTABLE CONSULT WITH PROGRAMMER AND ANALYST | LIAISON | 0 | 100 | 0 | 0.00 |
| CORRECT INTEGRATION PROBLEM AND REDO INTEGRATION | LIAISON | 0 | 100 | 0 | 0.00 |
| SEND COMPARE FILES BACK TO NSWCDD VIA DATALINK | LIAISON | .50 | 100 | .50 | 32.29 |

Figure D24: URGINTER Delivery Phase Resource Allocation

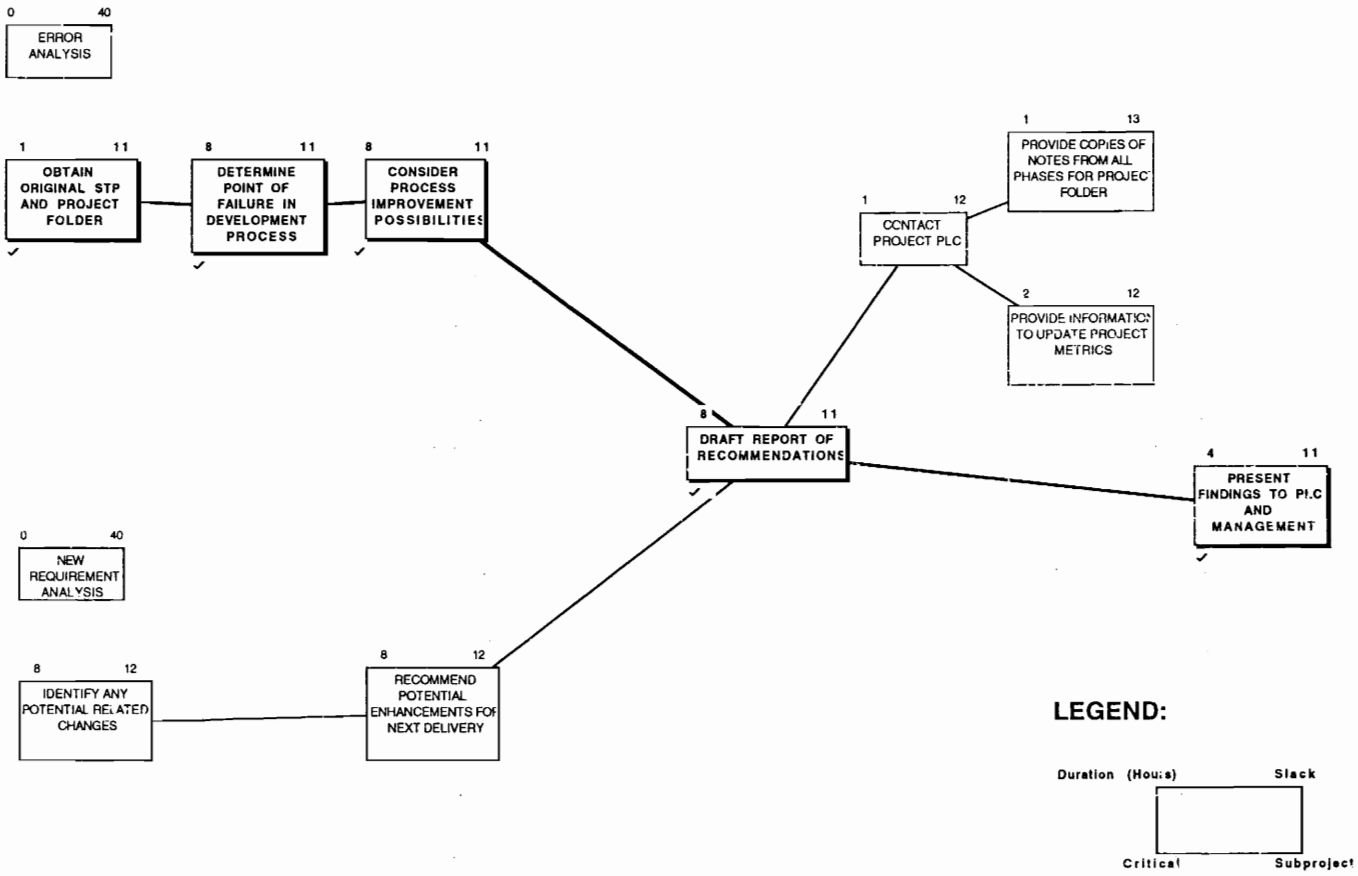



Figure D25: URGINTER Process Analysis Phase

| Name | Resource | Work-Hours | % Effort | Duration | Allocation Cost |
|---|---|---|---|---|---|
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | ANALYST | 1 | 100 | 1 | 64.58 |
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| OBTAIN ORIGINAL STP AND PROJECT FOLDER | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | ANALYST | 8 | 100 | 8 | 516.64 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| DETERMINE POINT OF FAILURE IN DEVELOPMENT PROCESS | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | ANALYST | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| CONSIDER PROCESS IMPROVEMENT POSSIBILITIES | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | ANALYST | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| DRAFT REPORT OF RECOMMENDATIONS | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | ANALYST | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| IDENTIFY ANY POTENTIAL RELATED CHANGES | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | ANALYST | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | PROGRAMMER | 8 | 100 | 8 | 516.64 |
| RECOMMEND POTENTIAL ENHANCEMENTS FOR NEXT DELIVERY | QA SPECIALIST | 8 | 100 | 8 | 516.64 |
| CONTACT PROJECT PLC | ANALYST | 1 | 100 | 1 | 64.58 |
| CONTACT PROJECT PLC | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| CONTACT PROJECT PLC | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | ANALYST | 2 | 100 | 2 | 129.16 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | PROGRAMMER | 2 | 100 | 2 | 129.16 |
| PROVIDE INFORMATION TO UPDATE PROJECT METRICS | QA SPECIALIST | 2 | 100 | 2 | 129.16 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | ANALYST | 1 | 100 | 1 | 64.58 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | PROGRAMMER | 1 | 100 | 1 | 64.58 |
| PROVIDE COPIES OF NOTES FROM ALL PHASES FOR PROJECT FOLDER | QA SPECIALIST | 1 | 100 | 1 | 64.58 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | ANALYST | 4 | 100 | 4 | 258.32 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | PROGRAMMER | 4 | 100 | 4 | 258.32 |
| PRESENT FINDINGS TO PLC AND MANAGEMENT | QA SPECIALIST | 4 | 100 | 4 | 258.32 |



Figure D26: URGINTER Process Analysis Phase Resource Allocation