IRM 5231-02A
W/CH 1

U.S. Marine Corps

System Development Methodology DEVELOPER PERSPECTIVE

PCN 186 523102 00

DEPARTMENT OF THE NAVY
HEADQUARTERS UNITED STATES MARINE CORPS
WASHINGTON, D.C. 20380-0001

IN REPLY REFER TO
5231/02A
CCIS-45
19 SEP 1988

From: Commandant of the Marine Corps

Subj: INFORMATION RESOURCES MANAGEMENT (IRM) SYSTEM DEVELOPMENT METHODOLOGY - DEVELOPER PERSPECTIVE

Ref: (a) MCO 5271.1
(b) MCO P5600.31

Encl: (1) IRM-5231-02A

1. PURPOSE. To provide a model which describes the System Development Methodology (SDM).

2. CANCELLATION. IRM-5231-02.

3. SUMMARY OF REVISION. This revision updates the data dictionary (Appendix F) and adds the next lower level of detail to the data flow diagrams and process descriptions describing the procedures associated with the SDM.

4. AUTHORITY. This publication is published under the auspices of reference (a).

5. APPLICABILITY. The guidance contained in this publication is applicable to all contractors and Marine Corps personnel responsible for the development of an automated system. This standard is applicable to the Marine Corps Reserve.

6. DISTRIBUTION. This technical publication will be distributed as indicated. Appropriate activities will receive updated individual activity Table of Allowances for Publications. Requests for changes in allowance should be submitted in accordance with reference (b).

7. SCOPE

a. Compliance. Compliance with the provisions of this publication is required unless a specific waiver is authorized.

b. Waivers. Waivers to the provisions of this publication will be authorized only by CMC (CC) on a case by case basis.

Subj: INFORMATION RESOURCES MANAGEMENT (IRM) SYSTEM DEVELOPMENT METHODOLOGY - DEVELOPER PERSPECTIVE

8. RECOMMENDATIONS. Recommendations concerning the contents of this technical publication should be forwarded to CMC (CCI) via the appropriate chain of command. All recommended changes will be reviewed upon receipt and implemented if appropriate.

9. SPONSOR. The sponsor of the technical publication is CMC (CCI).

R L Phillips

R. L. PHILLIPS
Brigadier General, U.S. Marine Corps
Director, Command and Control, Communications and
Computer (C4) Division, C4I2 Department

DISTRIBUTION:	L1P	
	L1Q	
	L1R	
	7000006	(1)
	7000007	(2)
	7000012	(2)
	7000014	(2)
	7000015	(2)
	7000017	(1)
	7000019	(1)
	7000020	(1)
	7000029	(40)
	7000031	(10)
	7000045	(1)
	7000050	(10)
	7000051	(4)
	7000062	(2)
	7000066	(2)
	7000067	(1)
	7000070	(1)
	7000071	(20)
	7000072	(10)
	7000093	(2)
	7000142	(1)
	7000148	(4)

Copy to: 8145001

DEPARTMENT OF THE NAVY
HEADQUARTERS UNITED STATES MARINE CORPS
WASHINGTON, D.C. 20380-0001

IN REPLY REFER TO:
5231/02A Ch 1
CTAS-20
29 MAY 1992

From: Commandant of the Marine Corps

Subj: INFORMATION RESOURCES MANAGEMENT (IRM) SYSTEM DEVELOPMENT METHODOLOGY - DEVELOPER PERSPECTIVE

Encl: (1) New page inserts to IRM-5231-02A

1. PURPOSE. To transmit new page inserts and direct pen changes to the basic technical publication of 19 September 1988.

2. ACTION

a. Remove present pages vii, 2-1 to 2-10, and the COMMENTS/REVISIONS page and replace with new pages vii, 2-1 to 2-11, and the new COMMENTS/REVISIONS page.

b. Change the CMC Code in the basic letter of promulgation from "CCI" to "MCCTA" in paragraphs 7.b., 8 and 9.

3. SUMMARY OF CHANGE. This Change clarifies the uniqueness of the DATAMANAGER/PREDICT interfaces, endorses the usage of partitioned datasets, updates the current usage of COM-PLETE, and adds a discussion on generating file descriptions from DATAMANAGER.

4. FILING INSTRUCTIONS. This change transmittal will be filed immediately following the signature page of the basic technical publication.

5. CERTIFICATION. Reviewed and approved this date.

H.W. Jenkins Jr.

H. W. JENKINS, JR.
Assistant Chief of Staff
Command, Control,
Communications,
Computer and Intelligence

DISTRIBUTION: PCN 186 523102A 01

Copy to: 8145001

UNITED STATES MARINE CORPS

Information Resources Management (IRM) Standards and Guidelines Program

System Development Methodology - Developer Perspective

IRM-5231-02A

19 SEP 1988

ENCL (1)

RECORD OF CHANGES

Log completed, change action as indicated.

Change Number	Date of Change	Date Received	Date Entered	Signature of Person Entering Change

SYSTEM DEVELOPMENT METHODOLOGY - DEVELOPER PERSPECTIVE
IRM-5231-02A

PUBLICATION TABLE OF CONTENTS

APPENDIXES

LIST OF FIGURES

Chapter Table of Contents

Chapter 1

GENERAL

(This page intentionally left blank)

Chapter 1

GENERAL

1.1. INTRODUCTION. System development methodologies generally fall into two categories: models which are process oriented, defined in terms of how to perform the phases and activities; and models which are data oriented, defined in terms of documents to be produced. The process type of model is directed toward project managers instead of developers and technicians. In order to better support developers, a "document" based version of the existing U.S. Marine Corps Life Cycle Model for Automated Information Systems is provided in the following manner:

a. IRM-5231-01, SDM-Overview. This material provides a high level perspective for use by the project development office and the project integration team.

b. IRM-5231-02, SDM-Developer Perspective. This material provides a detailed description of the developer activities and their interrelationships for use by project management.

c. The Specification, Plan, and Convention Standards (IRM Standards and Guidelines Program Technical Publications). This material provides general guidelines as well as precise descriptions of the format and content required for specific deliverables for use by project development personnel.

1.2. BACKGROUND. This volume of the System Development Methodology (SDM) presents a graphic model, with supporting narrative, that describes the methodology. The SDM uses data flow diagrams, therefore it is necessary to have a minimum knowledge of these techniques in order to understand and use this volume.

1.2.1. SDM Tools. Chapter 2 provides a description of the tools used to present the SDM, instructions for interpreting the diagrams, the relationship of the process descriptions, and the data dictionary descriptions of the data flows and stores.

1.2.2. SDM Procedures. Chapter 3 presents an overview of the use of the methodology oriented to the planner of applications development activities. Using the guidelines in Chapter 2, a task assignment, task order, statement of work, or similar work order can be developed to support the performance of development activities.

1.2.3. System Development Model. Chapter 4 and appendixes A through E provide the data flow diagrams that actually make up the methodology. The data flows of information that are used and produced during the development activities are depicted, along with the individual processes which describe the activities.

1.2.4. Data Dictionary. Appendix F provides the data dictionary that defines the data elements that are included in the model of the SDM. It includes the sources of information to the processes as well as those that are produced as a result of an activity.

Chapter Table of Contents

Chapter 2

SYSTEM DEVELOPMENT METHODOLOGY TOOLS

(This page intentionally left blank)

Chapter 2

SYSTEM DEVELOPMENT METHODOLOGY TOOLS

2.1. OVERVIEW. The SDM is presented in the form of a structured specification. It contains an events list, a context diagram, and a leveled set of data flow diagrams supported by data dictionary and process descriptions. The data flow diagrams present the logical data and functions necessary to produce a system, the data dictionary describes or defines the data that flows into and out of the processes, and the process descriptions define the activities associated with the processing, or transformation, of the data.

2.2. DATA FLOW DIAGRAMS (DFD)

2.2.1. Definition. A DFD is a network representation of a system; a portrayal of the system in terms of the data that flow through the system and the processes that act on or transform the data. The basic elements that make up a data flow diagram are:

a. Data flows, represented by lines or arcs.

b. Processes, or activities, represented by circles.

c. Files, or stores, represented by sets of parallel lines.

d. Data sources or sinks, represented by boxes.

2.2.2. Data Flow Diagram Symbols. As stated above, there is a basic set of four symbols which make up a data flow diagram. As with any other tool, it is sometimes beneficial to adapt it to a particular application. In this case, some variations were employed to clarify the unique aspects of modeling a system development. A graphic depiction of these symbols is provided in Figure 2-01. A description of both types of symbols follows.

a. The Four Basic Data Flow Diagram Symbols:

(1) Directed Line. Represents a flow of information or objects. The arrow indicates the direction of the data flow. The name of the data flow is written through or next to the line.

(2) Circle. Represents a task or process. It identifies a transformation of input data flows into output data flows. A brief descriptive name and a reference number for the process is written inside the circle.

(3) Two Parallel Lines. Represent a store of information or objects, irrespective of the storage medium. The store identifies a time delay for its contents. The name of the store is written between the lines.

(4) Rectangle. Represents an area where data originates or terminates from the point of view of the system study. It identifies a boundary of the system study; the identification of the originator/terminator is written inside the box.

b. Data Flow Diagram Symbol Variations.

(1) Data Flow Divergence. Indicates a distribution of the data flow with no actual transformation of data content or status. The data flow may be distributed in total or component data flows extracted from the main data flow.

(2) Data Flow Convergence. Indicates a collection of data flows that forms a single data flow with no actual trans-formation of data content or status.

(3) Two-way Data Flow. Indicates a two-way flow of data. The data flows are separate and should be viewed as two independent flows of data. This kind of data flow is usually used to indicate that data flowing in produces the data flowing out, but this rule has been violated to reduce the visual complexity of a diagram by using it for two unrelated data flows.

(4) Dashed Circle. Indicates a management process. Because the agent performing a process is never shown on a data flow diagram, the dashed circle is used to distinguish the processes that are totally managerial from the technical processes.

2.2.3. DFD's Versus Flowcharts. One of the major confusions about DFD's is that they are often mistaken for flowcharts where each process on a diagram is mistakenly viewed as a one-time execution involving a single occurrence of effort and time. A dataflow diagram is a model of a requirement or problem, whereas a flow-chart is a model of how a requirement is accomplished. In a set of data flow diagrams, the way one determines and partitions the processes is the most important consideration of analysis. DFD's show partitioning of a process from the point of view of the data, rather than from the point of view of control. Hence, just as a process can be partitioned, data itself can be partitioned and processed in packets, instead of all at one time. In other words, one can perform a partial process on partial input and produce partial output. Because a DFD does not show control, one can cycle through a DFD until a product is completed or the goal is accomplished.

2.2.4. Example. The use of data flow diagrams is best explained by an analogy. Think of the DFD symbols as a real working environnment. For example, a process is an individual's desk. The data flows are IN and OUT trays on the desk, and stores are file cabinets that an individual can access. A person sits at a desk until a piece of data arrives at the IN tray. If a task can be accomplished on this piece of data by itself, then the person will perform it and place the transformed data in the OUT tray. It may be that two inputs are needed before a process can take place, in which case the person remains inactive until both inputs arrive in the IN tray. The process is data driven.

2.3. PROCESS DESCRIPTIONS. A Process Description is a concise specification of an activity that is presented on the data flow diagram by a circle. The Process Descriptions for the SDM are presented in Chapter 4, "System Development Model", which collects in one place all of the activity specifications. The Process Descriptions are each keyed by a number back to the circle on the data flow diagram chart on which it appears. In order to use the SDM, it is necessary to understand the format of a Process Description, and the relationship of the processes to their respective descriptions. A Process Description consists of four elements.

a. Process Name Element. Identical to the name shown within the associated circle on the data flow diagram chart on which the process appears. It is a descriptive name of the activity associated with the processing of data which flows through the circle.

b. Process Reference Number Element. The number that appears in the process circle along with the name, and is the key that points back and forth between the Process Description and the data flow diagram. Thus the user can locate the appropriate Process Description in Chapter 4, "System Development Model", by the number found in the circle, or locate the process in the Data Flow Diagram by using the number of the description.

c. Process Guided By Element. Identifies all of the data flow elements which provide control, guidance, or management direction to the process being described. This includes government policies or directives, documentation standards, or documents which are products of other processes in the SDM. All of these are described in Appendix F, "Data Dictionary".

d. Process Description Element. This is the heart of the Process Description, and contains detailed specifications of the activities to be performed. Whereas the data flow diagram depicts what is to be done, the Process Description provides the details of how it is to be done, supported by the data elements itemized in the Process Guided By Element.

2.4. DATA DICTIONARY. A general definition of a data dictionary is that it is a repository of data about data. This dictates that all of the information in the data dictionary pertains to the data that flows through the model of a project as depicted in the data flow diagram. Conversely, all of the data elements are described in the data dictionary. It is important to note that this use of the term data dictionary does not refer to any commercial package that is used to physically store data descriptions, such as DATAMANAGER. While the data dictionary will most likely be stored in the project data dictionary, they are not synonymous.

2.5. AUTOMATED TOOLS. The automated tools described in the following paragraphs are intended for use during the system development phases to support the SDM. Each of these tools are provided by the Commandant of the Marine Corps (CMC) with administration of procurement and maintenance contracts handled by C4I Division (Code MCCTA). All automated tools will be considered proprietary software and therefore subject to the provisions of MCO 5234.2. Any deviations from the above will require written approval from CMC (Code MCCTA). Figure 2-02 depicts graphic representation of the required interaction between automated tools during the life cycle. Figure 2-03 cross-references the automated tools to the phases of the life cycle. Automated tools exist today for the SDM areas of data dictionary, logical data modeling and normalization, library management, performance measurement (for ADABAS systems only), and prototyping. Access to these tools will be controlled through the proprietary software package CA-TOP SECRET (TSS). Software packages and automated tools for micro-computers/personal computers are not addressed in this section since no product lines have been designated as a Marine Corps standard at this time. Fourth generation languages (4GLs) and supported automated tools are being evaluated by the Marine Corps using micros and mainframe computers.

2.5.1. DATAMANAGER (Data Dictionary). DATAMANAGER is the approved USMC standard data dictionary and will be used as the master/central data dictionary for SDM. Each site is responsible for the development of an interface between DATAMANAGER and PREDICT, the ADABAS dictionary, to facilitate creation of ADAWAN cards for PREDICT from DATAMANAGER. A Marine Corps standardized interface does not exist. DATAMANAGER use is begun early to support the data flow diagrams during the SDM. It describes or defines the documentation that flows into and out of the process bubbles with the process descriptions defining the activities associated with processing or transforming the data. It also defines data stores. The DATAMANAGER interface to LIBRARIAN may also be used to generate programming file descriptions. The data dictionary contents are used to generate programming language file descriptions directly from DATAMANAGER into LIBRARIAN as file description modules. These modules can then be used as official file descriptions for all program processing. DATAMANAGER will be used throughout the LCM process, to include the maintenance cycle.

2.5.2. DESIGNMANAGER (Logical Data Modeling and Normalization). DESIGNMANAGER is the approved USMC standard logical data modeling and normalization tool. It is used to produce logical data models from the data dictionary.

2.5.3. LIBRARIAN (Library Management). The approved USMC standard for library management is LIBRARIAN. It is a software system that is used to manage source programs and other components of the project. Software for a project will be stored and compressed in a master file. LIBRARIAN control statements are used to add, delete, or update modules on LIBRARIAN. These control statements and their formats are described in the LIBRARIAN User Reference Manual. Partitioned datasets (PDSs) can also be used to store and manage source programs. The increased functionality of TSO within the Marine Corps and the availability of off-the-shelf programming tools which utilize partitioned datasets necessitate the need for an additional software manager.

2.5.4. Teleprocessing (TP) Monitors.

a. COM-PLETE. COM-PLETE is a multi-thread and multi-tasking TP monitor that combines software for all TP functions into a single software facility using a single network of terminals. COM-PLETE is in limited usage at the MCCDPA, Kansas City and on Force Automated Service Centers (FASCs) one, two, and three.

b. Customer Information Control System (CICS). CICS is a general purpose data communications monitor that reduces the effort needed for terminal oriented transaction programming. It interfaces between application programs, teleprocessing access methods and data base management systems.

c. Time-Sharing Option (TSO). TSO is a time-sharing system that provides interactive computing through maintenance of procedure and system libraries and catalogs; through application development and maintenance; and through creation, maintenance, and control of support libraries.

2.5.5. CA-TOP SECRET (TSS) (Access Control). TSS is an access control system that enforces user established access rules, controls access to computer systems and data, prevents accidental or covert disclosure of data, and limits access to authorized users only.

2.5.6. ROSCOE (Interactive Programming Facility). The Remote OS Conversational Operations Environment (ROSCOE) is a comprehensive system for on-line development, offering users a permanent storage library system, full on-line data entry and editing facilities, job submission, output queue viewing and disposal, and security. Other facilities include COBOL and JCL syntax checking, complete OS/VS data management facilities for application and systems programmers, and an interactive procedural language.

2.5.7. PREDICT (ADABAS Data Dictionary). PREDICT is the integrated data dictionary for ADABAS, the data base management system and NATURAL, the ADABAS associated 4GL.

2.5.8. NATURAL SECURITY (Access Control for NATURAL). NATURAL SECURITY is the application which controls access to NATURAL commands, files and libraries. It is separate from TSS with no direct interface.

2.5.9. CA-OPTIMIZER (COBOL Source Code Optimizer). CA-OPTIMIZER is a COBOL optimization package that generates and amplifies optimized COBOL object modules. It assists in the testing, debugging, fine-tuning and maintenance efforts associated with COBOL programming.

2.5.10. ADABAS (Data Base Management System). The Adaptable DataBase System (ADABAS) is an inverted index data base management system.

2.5.11. FOCUS (Data Base Management System). FOCUS is a high-level, user-oriented, English language data base management system. Both mainframe and PC versions are available.

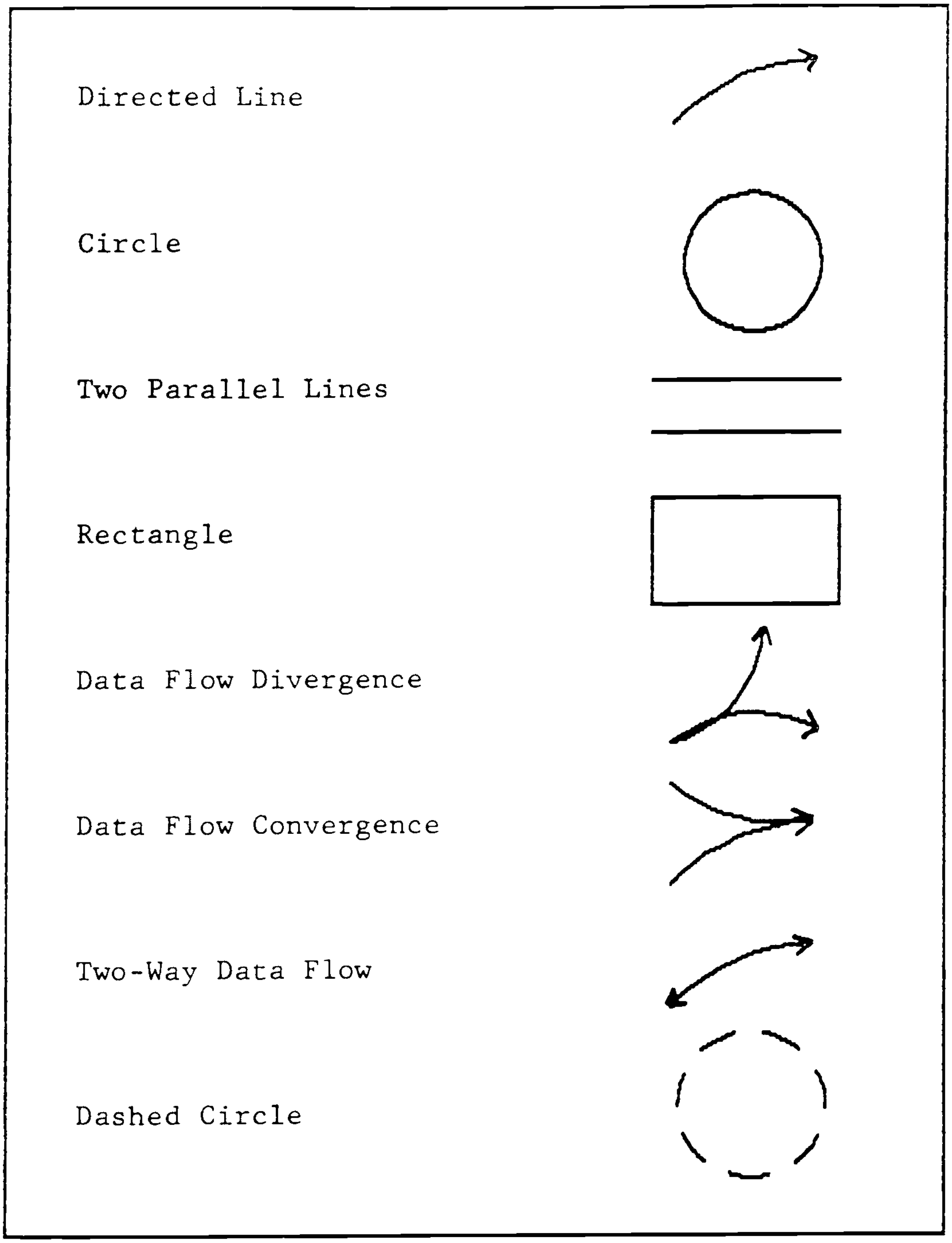

FIGURE 2-01
Data Flow Diagram Symbols

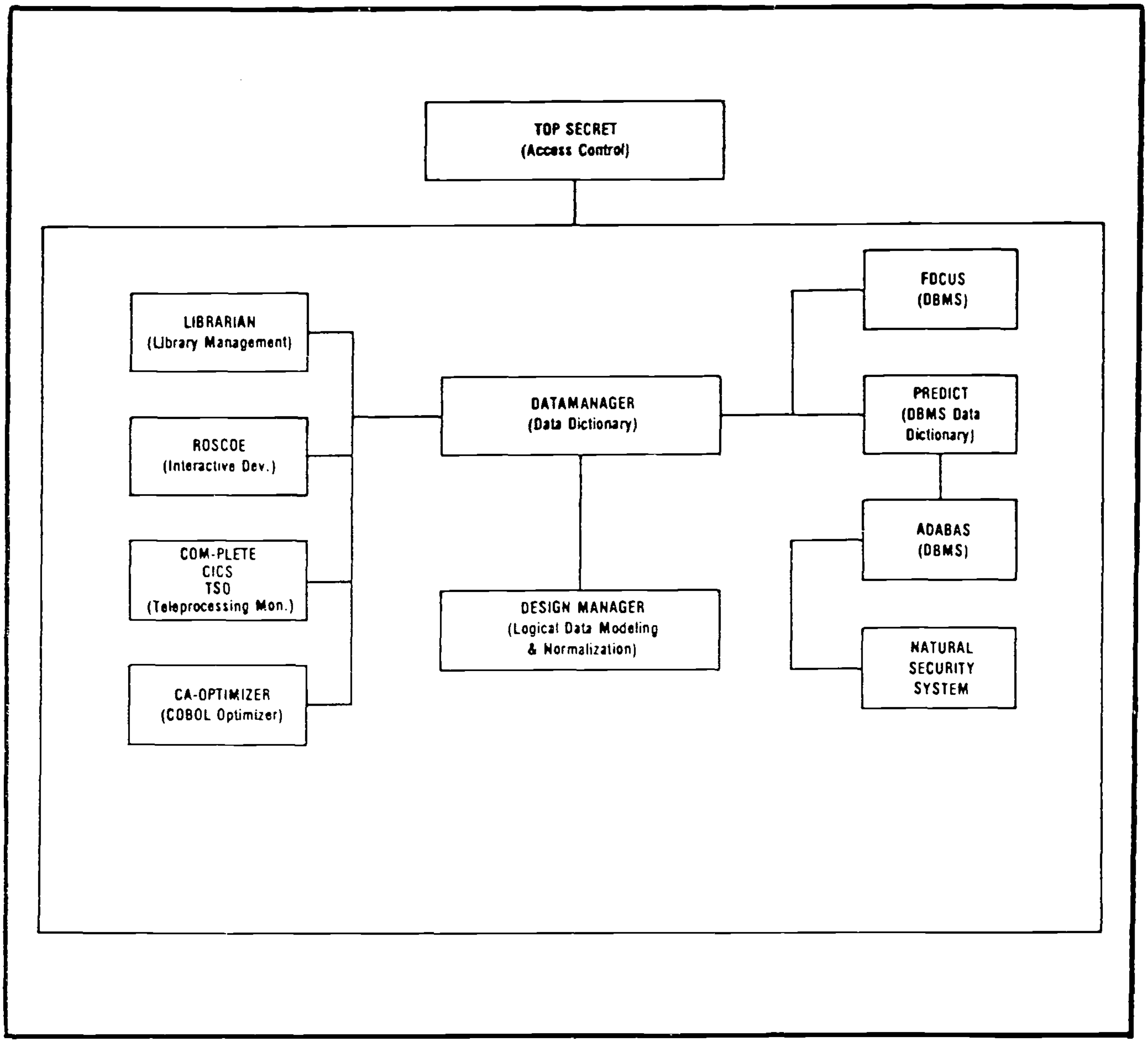

FIGURE 2-02
SDM Tool Interaction

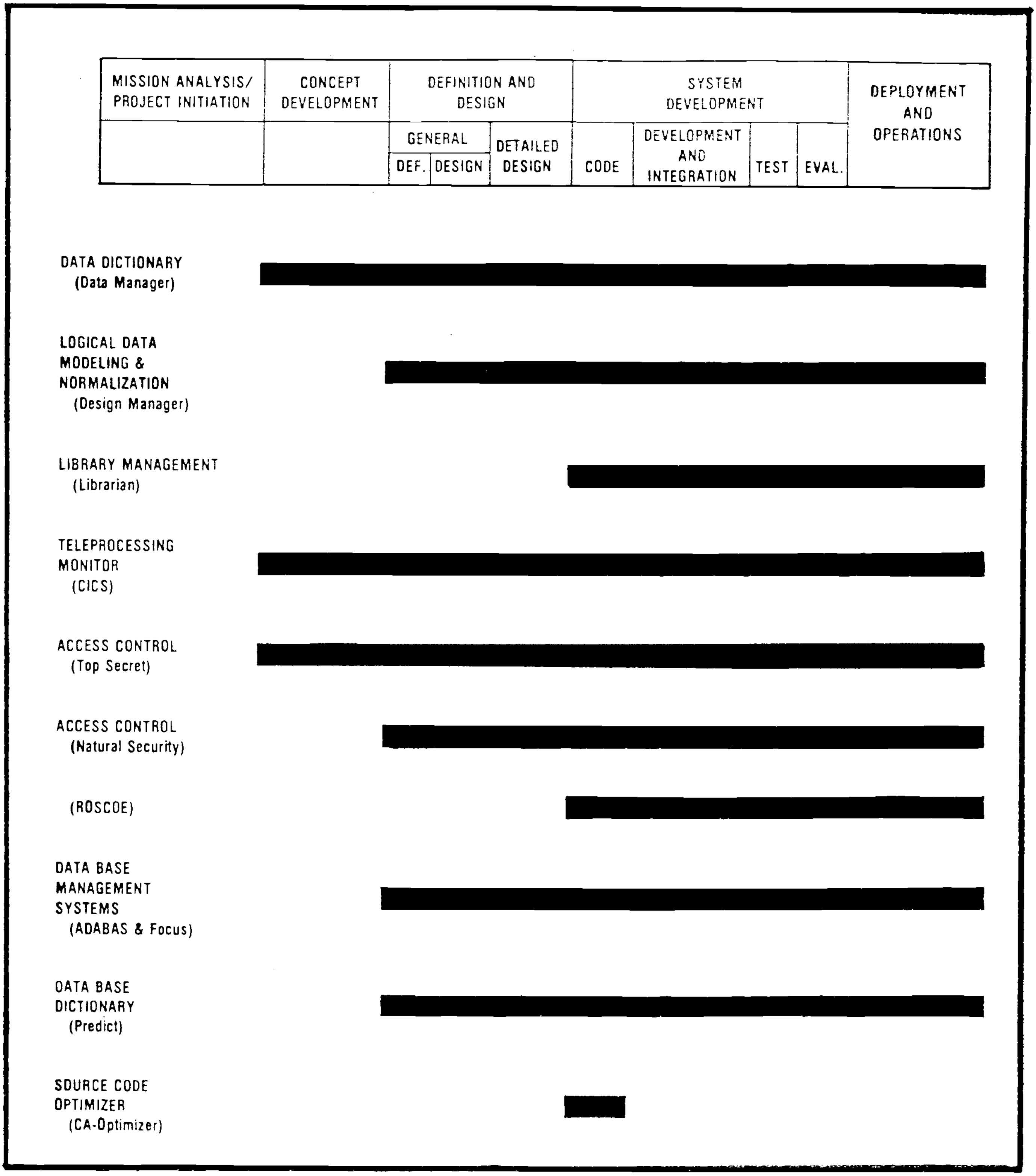

FIGURE 2-03
SDM Tools By Phase

COMMENTS/REVISIONS

Technical publications under the Information Resources Management (IRM) Standards and Guidelines Program (MCO 5271.1) are reviewed annually. Your comments and/or recommendations are strongly encouraged.

IRM Tech Pub Name: ______________________________

IRM-_____-___ (Number) Date of Tech Pub: ____________________

COMMENTS/RECOMMENDATIONS: ______________________________

Name/Rank: ______________________________(optional)
Unit: ______________________________(optional)

Mail To: U.S. Marine Corps
Computer & Telecom Activity (MCCTA Code: CTAS-30)
Building 2006 (3rd Floor)
HQBN, MCCDC
Quantico, Va 22134-5010

Chapter Table of Contents

Chapter 3

SYSTEM DEVELOPMENT METHODOLOGY PROCEDURES

(This page intentionally left blank)

CHAPTER 3

SYSTEM DEVELOPMENT METHODOLOGY PROCEDURES

3.1. OVERVIEW. The SDM is designed to be a working document, to be used by those personnel and organizations involved with the planning of application development efforts in support of the system development. The basic procedures consist of determining from the data flow diagram the portion of the development to be addressed by the particular effort, building a work statement by extracting the process descriptions, and developing a Contract Data Requirements List (CDRL) which consists of those data flows coming out of the processes within the boundaries of the scope of the work effort. The CDRL will be supported by the data dictionary entries which describe the documents, and the standards which apply to the development of the plans and specifications.

3.1.1. Establishing The Boundary. The process of defining a work assignment in support of the system development is greatly simplified when the structured specification is used. The first step in defining the work effort is to locate, on the appropriate data flow diagram, the process or processes to be performed. The next step is to establish the boundary, which is simply to draw a line around the bubbles which depict the activities to be performed. For example, if a requirement exists to develop a Functional Requirements Definition for some application, that activity or process is contained on the Definition and Design Chart (Appendix C). The boundary can encompass one or several related processes, such as shown by the dotted line around the four process bubbles shown in Figure 3-01, "Establishing the Boundary." This boundary accomplishes the function of defining the scope of the work.

3.2. DEFINING THE ACTIVITIES. After the boundary has been established, it is necessary to provide a description of the activities represented by the process bubbles. At this point, the procedure is to extract Process Descriptions associated with the processes. For every process included within the work boundary, there is a Process Description which details the activities to be performed. These are keyed to the data flow diagram by the process number; that is, the number in the bubble is the same as the Process Description Number. At this point, the planner now has a definition of the scope, and a description of the activities. The remaining step is to define the documentation associated with the activities.

3.3. DESCRIBING THE DELIVERABLES. The task of identifying and describing the deliverable documentation consists of two steps: compiling a list of documents, and describing the form and content. At the point where the boundary drawn around the processes crosses any data flow arcs, there is a flow of management information. These represent sources of information or management data that in some way controls the activities. The data going into data stores represents project deliverables, such as reports, data base contents, or project documentation.

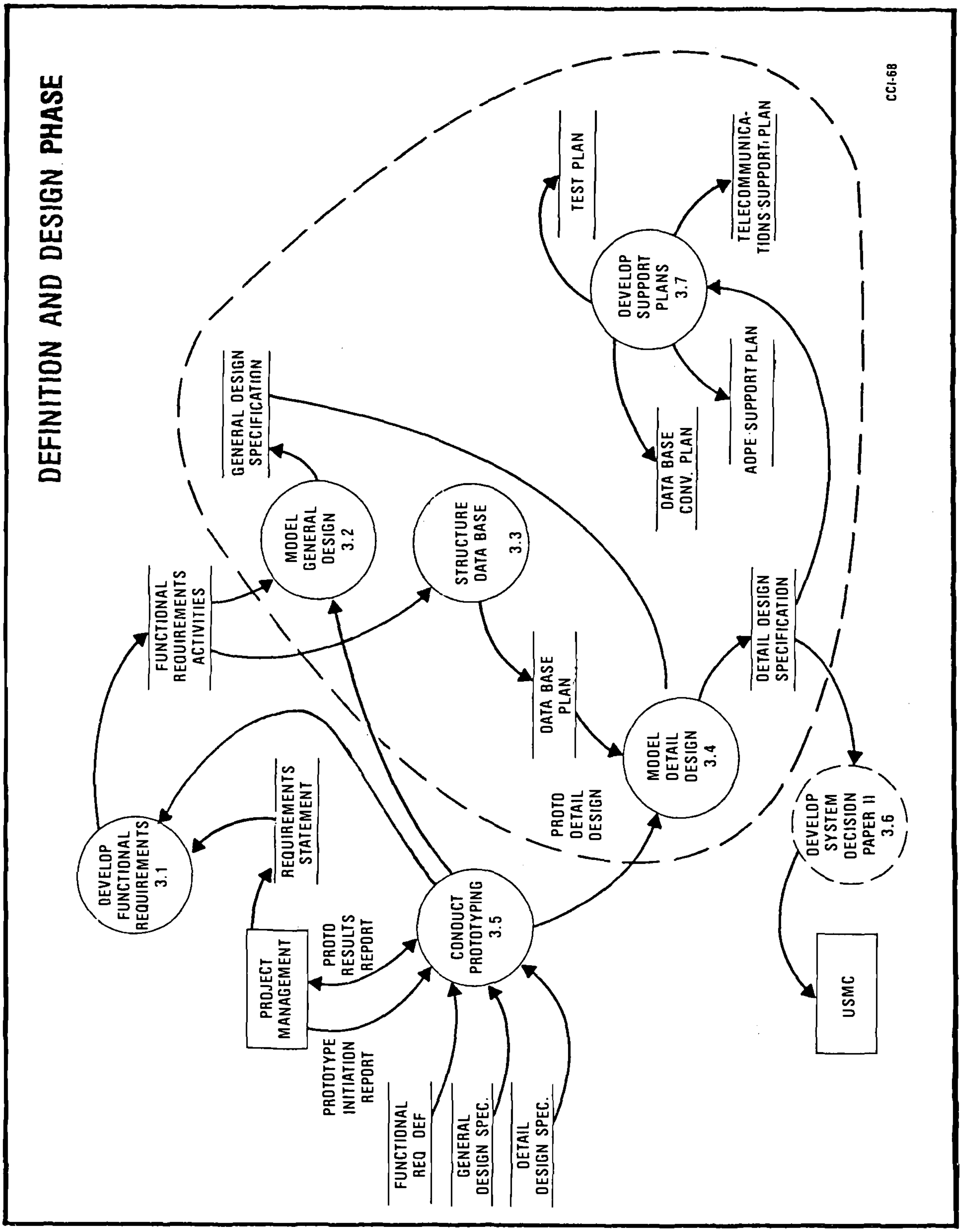

FIGURE 3-01
Establishing the Boundary

Chapter Table of Contents

Chapter 4

MODEL

(This page intentionally left blank)

Chapter 4

SYSTEM DEVELOPMENT MODEL

4.1. OVERVIEW. The following set of data flow diagrams are a graphic model of the System Development Methodology that should be used on a project. In order to keep this model as flexible as possible during the early stages of the project, it only depicts the processes of work on each diagram to the first level of detail. For example, in Data Flow Diagram 1, Process 1.2 - Document Steering Group Charter is obviously not at a functional primitive level; it requires further decomposition. Since the details for each particular application development, using this SDM, can not be totally predicted, decomposition of this process stops here in the model. However, each application using this SDM should decompose this process to whatever level is appropriate to address pertinent details adequately.

4.2. CONTEXT DIAGRAM. Figure 4-01 identifies the boundary of the methodology. It is a top-level graphic representation of the system. It depicts all the net inputs to and outputs from the system, but shows no decomposition or partitioning.

4.3. DIAGRAM 0 - SYSTEM DEVELOPMENT METHODOLOGY. Figure 4-02 represents a one page pictorial overview of the life cycle management process and SDM approach. The relationship between LCM and SDM will be outlined during the following discussion.

4.4. DIAGRAM 1 - MISSION ANALYSIS AND PROJECT INITIATION. Appendix A describes the processes involved with identification of the mission element need and recommendations for the exploration of alternative functional concepts to satisfy that need. This process is the first phase of life cycle management and is included to show the relationship between LCM and SDM. This is a Level 1 Diagram; the lowest level of detail the SDM currently identifies.

4.5. DIAGRAM 2 - CONCEPT DEVELOPMENT. Appendix B describes the processes that establish requirements which will satisfy the need stated in the MENS, evaluate/develop alternative methods that will satisfy those requirements, and recommend one alternative for further development. This is a Level 1 Diagram; the lowest level of detail the SDM currently identifies.

4.6. DIAGRAM 3 - DEFINITION AND DESIGN. Appendix C describes the processes involved with establishing finalized Detail Design Specifications, and developing a system decision paper for approval to continue the project. Upon approval, documentation for the various plans required to continue the effort is completed. This is a Level 1 Diagram; the lowest level of detail the SDM currently identifies.

4.7. DIAGRAM 4 - SYSTEM DEVELOPMENT. Appendix D describes the processes involved with program code development and testing activities that will generate System Decision Paper III for approval to continue the project. Upon approval, documentation of the users and operations manuals is completed and the tested code is implemented. This is a Level 1 Diagram; the lowest level of detail the SDM currently identifies.

4.8. DIAGRAM 5 - DEPLOYMENT. Appendix E describes the processes involved with conducting training for the finished product, the implementation and the validation of the system to requirements. This is a Level 1 Diagram; the lowest level of detail the SDM currently identifies.

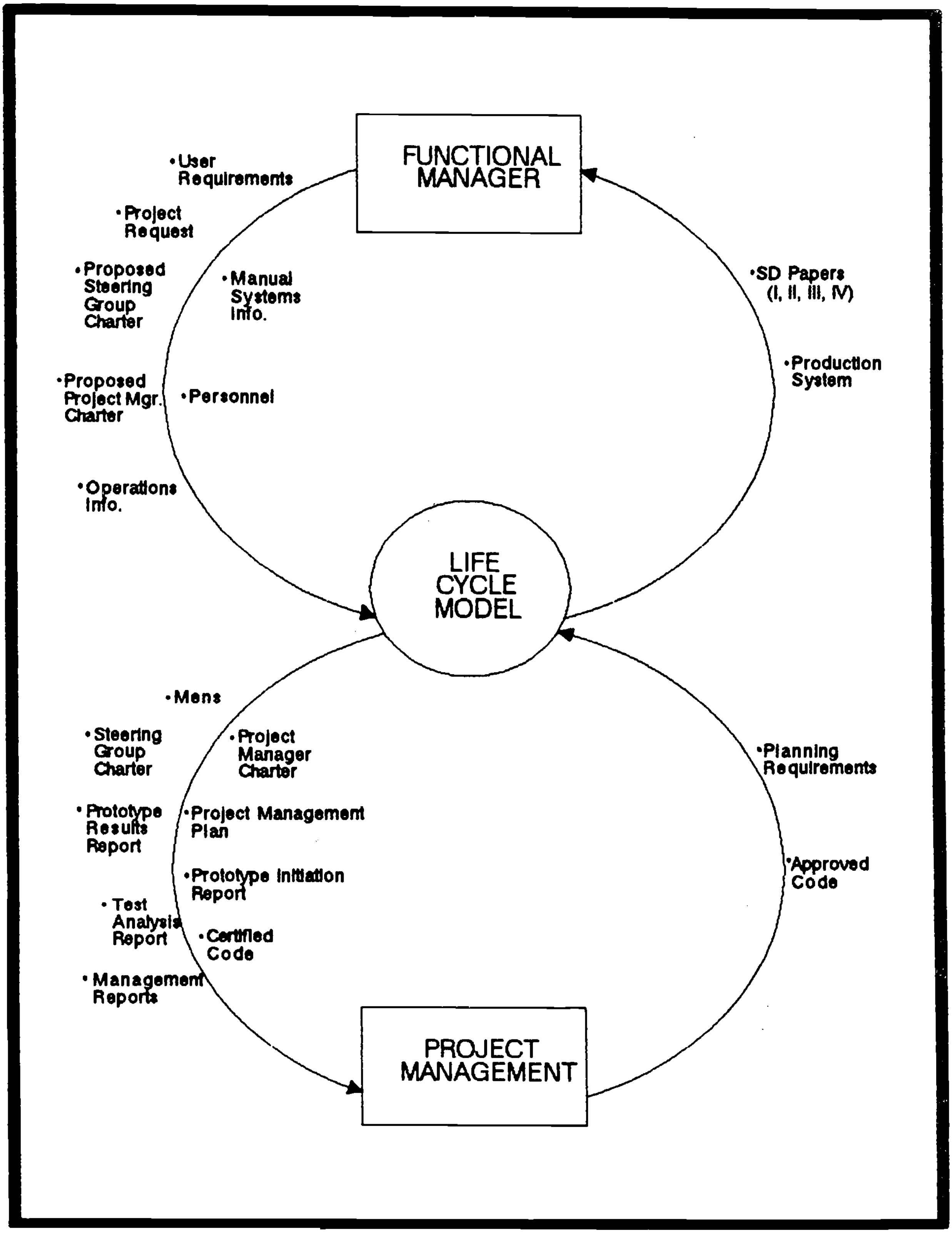

FIGURE 4-01
Context Diagram

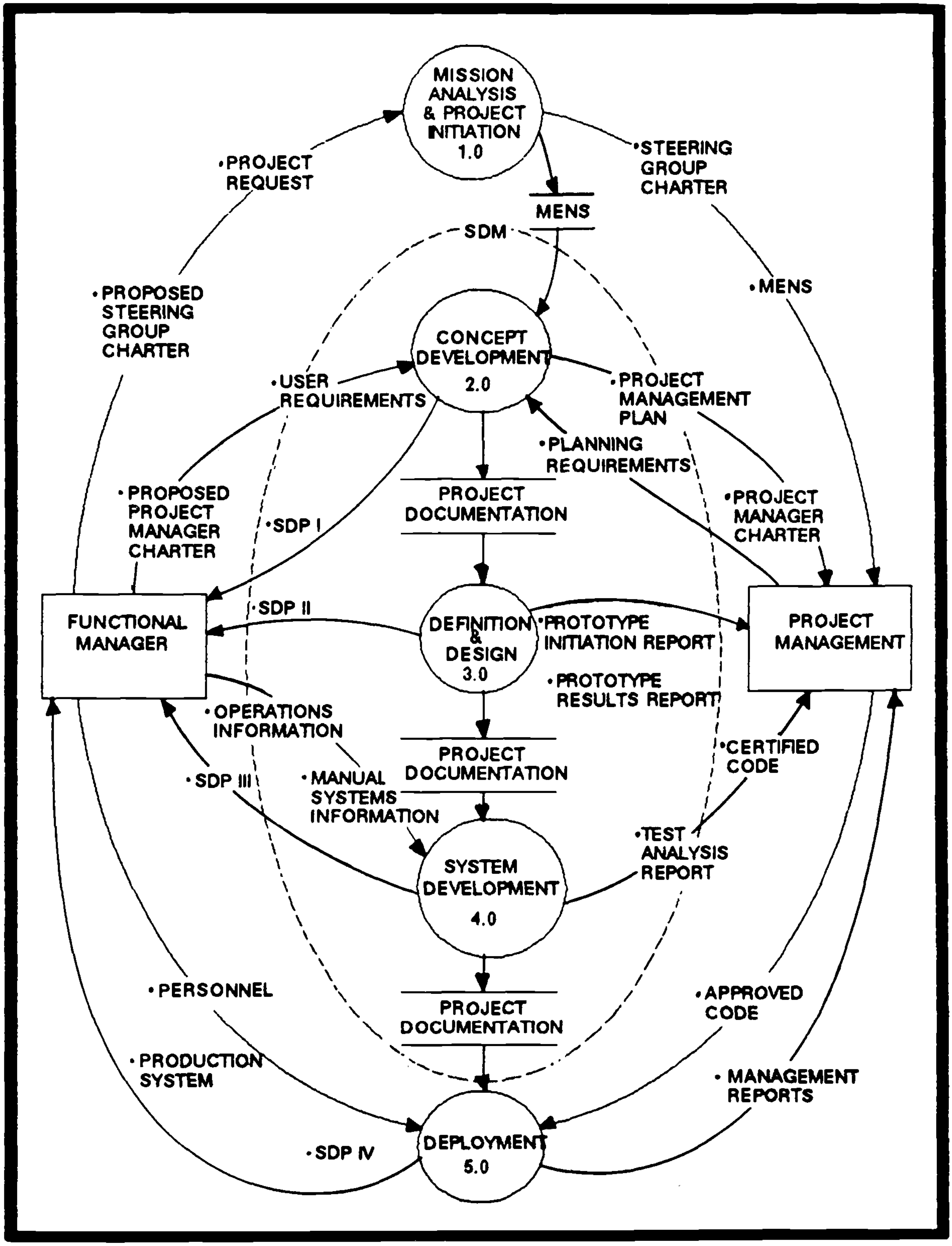

FIGURE 4-02
DIAGRAM 0 - Life Cycle Management and
System Development Methodology

Appendix A

DIAGRAM 1 - MISSION ANALYSIS AND PROJECT INITIATION

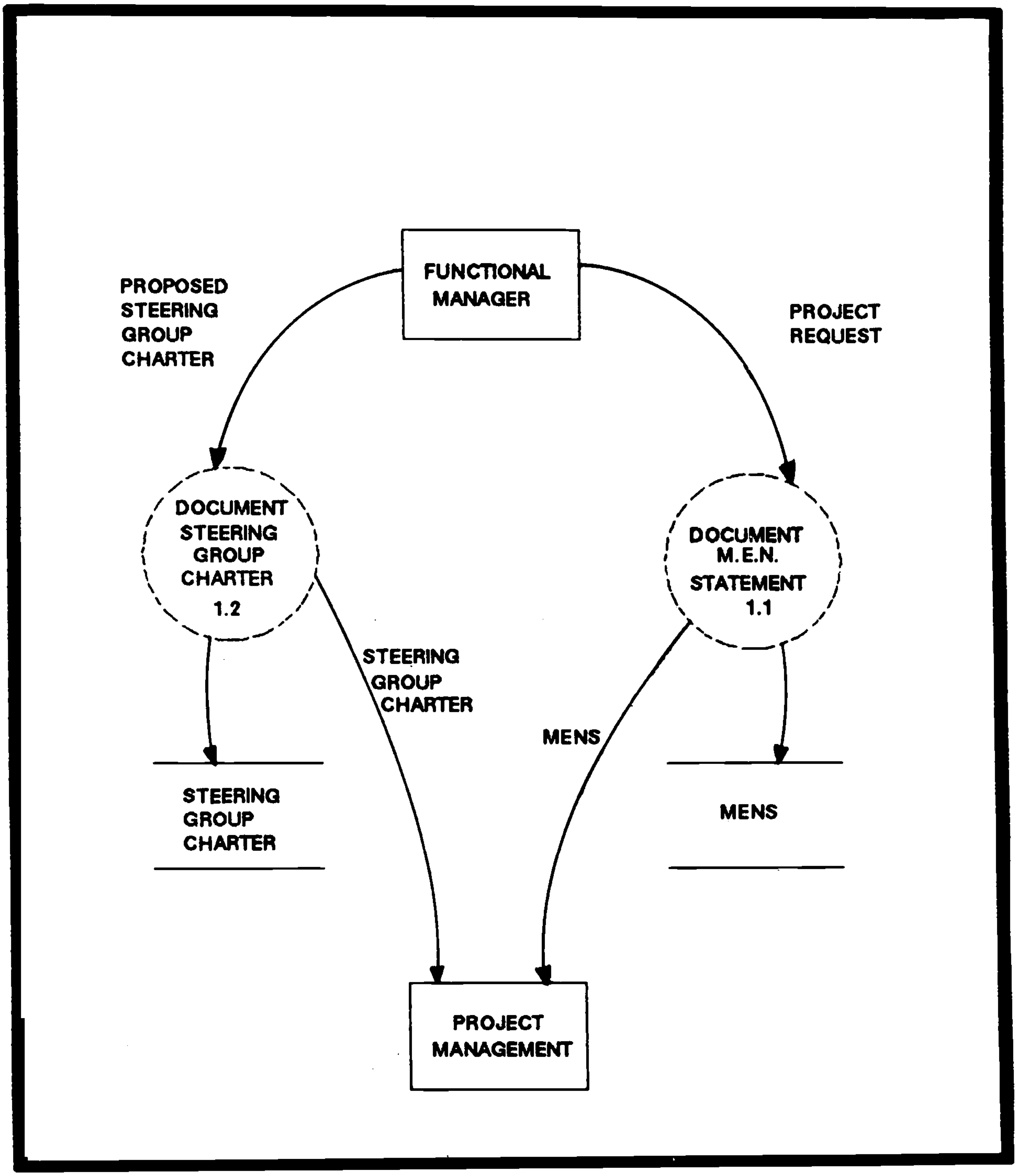

FIGURE A-01
DIAGRAM 1 - Mission Analysis and Project Initiation

PROCESS NAME: Document Mission Element Need Statement (MENS)

PROCESS REFERENCE NUMBER: 1.1

PROCESS GUIDED BY: MCO P5231.1A

PROCESS DESCRIPTION:

All development efforts start with documentation of a Mission Element Need Statement (MENS) based on a user's project request. The preparer of this document will obtain the required data and information from interviews, analysis, and the production, procedures, development standards, and general information project libraries. Once the statement is documented and staffing requirements have been determined and met, the MENS is forwarded to Headquarters Marine Corps for approvals/disapprovals.

PROCESS NAME: Document Steering Group Charter

PROCESS REFERENCE NUMBER: 1.2

PROCESS GUIDED BY: MCO P5231.1A

PROCESS DESCRIPTION:

When the scope of the system impacts upon more than one functional area, a steering group composed of representatives from each functional area will be established. The MENS approval authority is responsible for establishing the steering group and staffing of the charter to each member of the proposed group.

Appendix B

DIAGRAM 2 - Concept Development

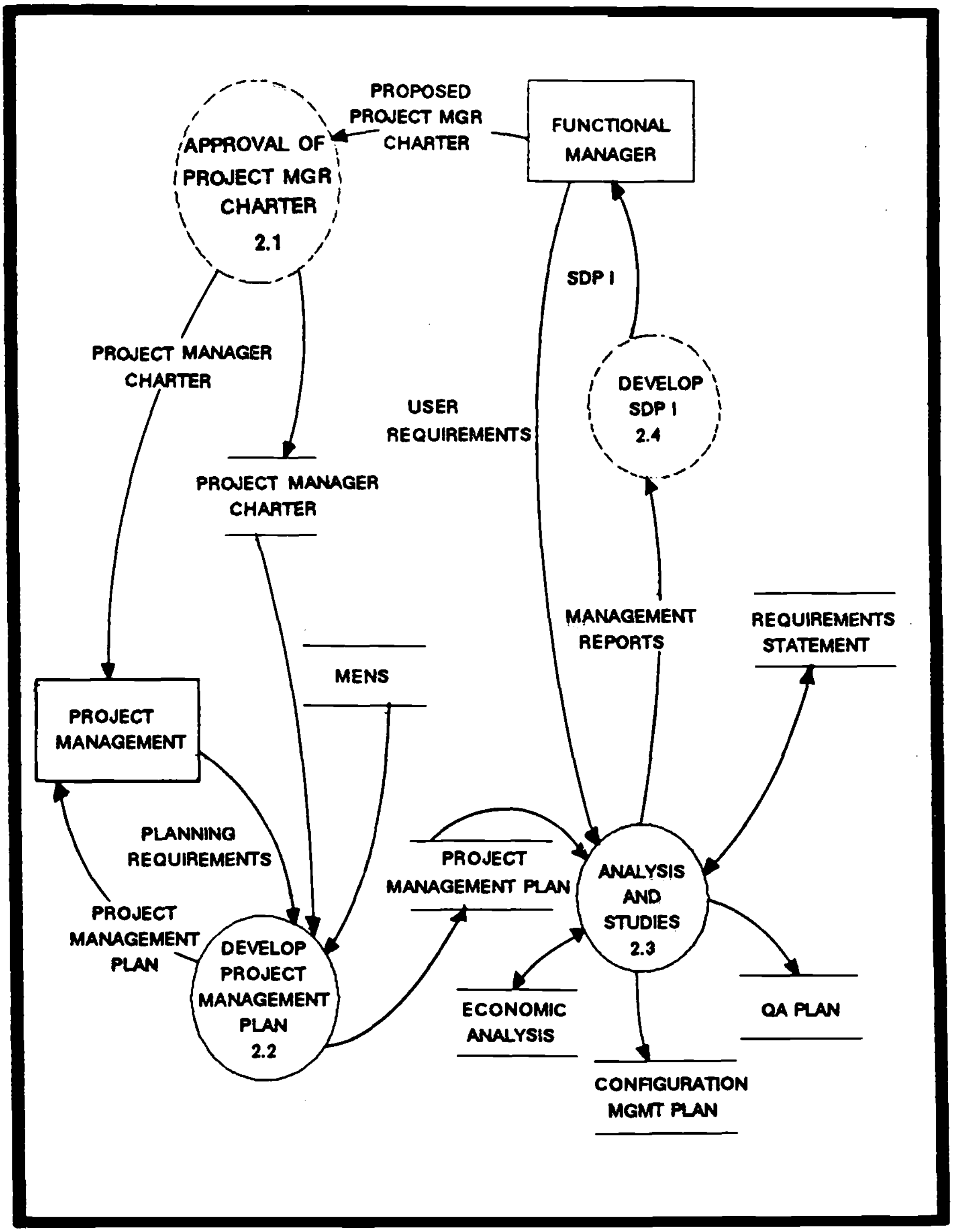

FIGURE B-01
DIAGRAM 2 - Concept Development

PROCESS NAME: Approval of Proposed Project Manager Charter

PROCESS REFERENCE NUMBER: 2.1

PROCESS GUIDED BY: MCO P5231.1A

PROCESS DESCRIPTION:

The approval authority, as defined by MCO P5231.1A, will review and approve/disapprove the proposed project management charter submitted by the functional manager. This document will identify the Project Manager by name. The document will be filed in the project library and will serve as the Project Manager's delegated authority to provide the daily direction, coordination, and control of the project.

PROCESS NAME:	Develop Project Management Plan
PROCESS REFERENCE NUMBER:	2.2
PROCESS GUIDED BY:	MCO P5231.1A
PROCESS DESCRIPTION:	Project Development Standard - Project Management Plan (IRM-5231-19)

Based upon information contained in the MENS and PMC, the Project Manager will develop a plan that outlines the resources, tasks and milestones to develop the system.

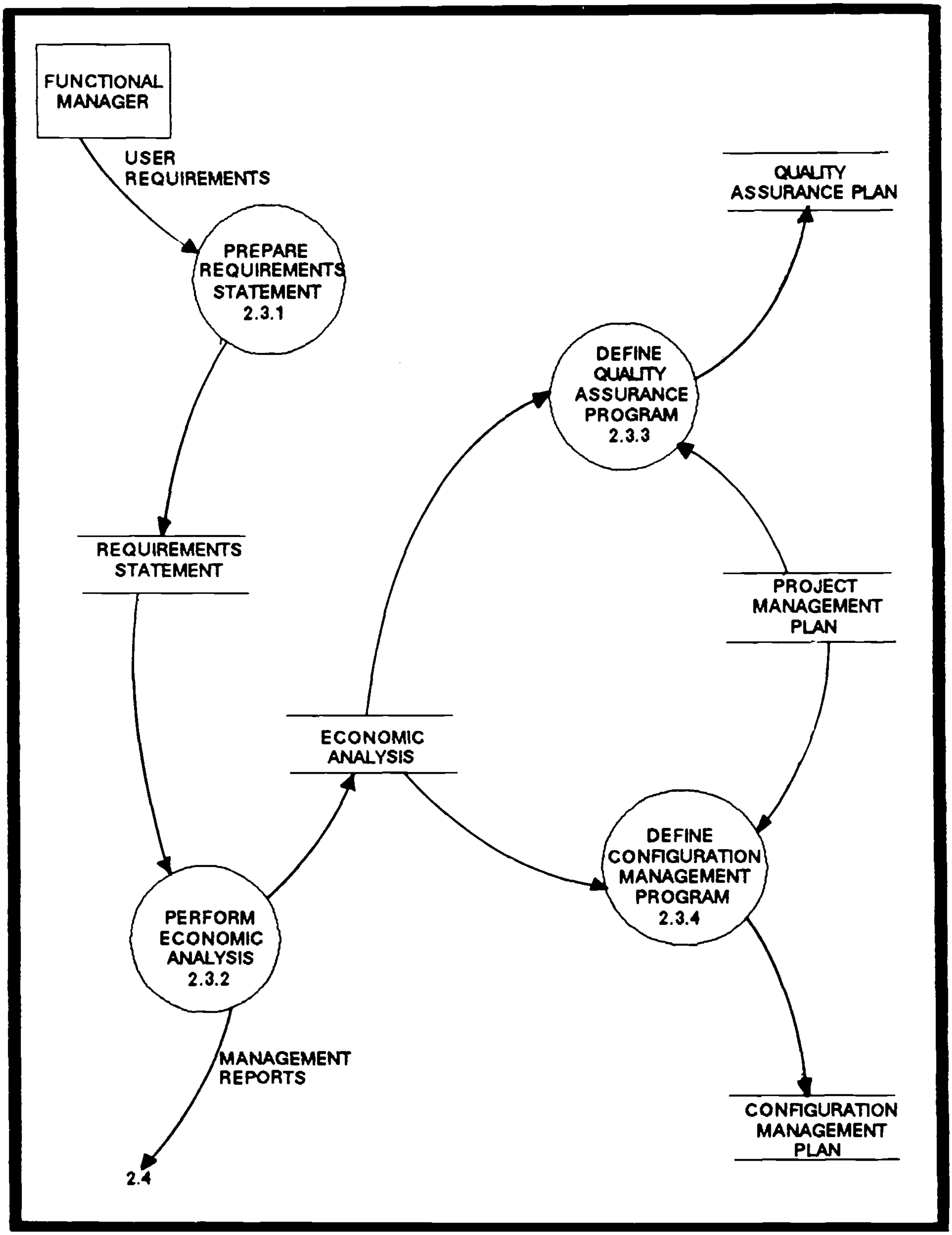
FUNCTIONAL MANAGER
USER REQUIREMENTS
PREPARE REQUIREMENTS STATEMENT 2.3.1
QUALITY ASSURANCE PLAN
DEFINE QUALITY ASSURANCE PROGRAM 2.3.3
REQUIREMENTS STATEMENT
PROJECT MANAGEMENT PLAN
ECONOMIC ANALYSIS
DEFINE CONFIGURATION MANAGEMENT PROGRAM 2.3.4
PERFORM ECONOMIC ANALYSIS 2.3.2
MANAGEMENT REPORTS
2.4
CONFIGURATION MANAGEMENT PLAN

FIGURE B-02
DIAGRAM 2.3 - Analysis and Studies

PROCESS NAME: Develop Requirements Statement

PROCESS REFERENCE NUMBER: 2.3.1

PROCESS GUIDED BY: Requirements Statement (IRM-5231-20)

PROCESS DESCRIPTION:

This process will result in a clearly defined set of functions required of the system. These requirements must be specific in terms of what is to be done, but should not address how it is to be accomplished.

PROCESS NAME: Conduct Economic Analysis

PROCESS REFERENCE NUMBER: 2.3.2

PROCESS GUIDED BY: Economic Analysis (IRM-5236-03)

PROCESS DESCRIPTION:

The economic analysis process is a systematic approach to the problem of choosing how to employ scarce resources and an investigation of the full implications of achieving a given objective in the most efficient and effective manner. The purpose of an economic analysis is to estimate the cost and benefits of system alternatives in a standardized manner. It also will provide a cost and schedule baseline.

To help guide the system developers through the economic analysis, we recommend that the work team follow the approach presented below:

2.3.2.1 Examine Requirements Statement and related documents

2.3.2.2 Review characteristics of current system

2.3.2.3 Review new system requirements

2.3.2.4 Interview users to confirm information

2.3.2.5 Identify alternatives

2.3.2.5.1 Define feasible alternatives for analysis
2.3.2.5.2 Assemble Alternatives Definition information
2.3.2.5.3 Identify assumptions and constraints

2.3.2.6 Analyze costs and benefits

2.3.2.6.1 Identify cost categories
2.3.2.6.2 Separate capital costs from reccurring costs
2.3.2.6.3 Estimate values for each cost category
2.3.2.6.4 Prepare list of anticipated benefits (tangible and intangible)
2.3.2.6.5 Identify the benefits that are quantifiable
2.3.2.6.6 Estimate values for each quantifiable benefit
2.3.2.6.7 Perform Cost/Benefit Analysis
2.3.2.6.8 Perform present value and sensitivity analysis
2.3.2.6.9 Assemble Cost/Benefit Analysis information

2.3.2.7 Select best alternative

2.3.2.8 Assemble Final Economic Analysis (EA)

2.3.2.8.1 Prepare Alternatives Definition document
2.3.2.8.2 Prepare Cost/Benefit Analysis document
2.3.2.8.3 Prepare Funding Profile document (if any)
2.3.2.8.4 Prepare Acquisition Strategy document (if any)

The economic analysis will document the estimated costs to develop, operate, and maintain the system over the stated life cycle, compared to the benefits that will result. Optional parts of the analysis are the acquisition strategy and the funding profile, which are prepared as separate documents. The economic analysis must be updated during each phase of the IS life cycle.

PROCESS NAME: Develop Quality Assurance Program

PROCESS REFERENCE NUMBER: 2.3.3

PROCESS GUIDED BY: Quality Assurance Plan (IRM-5231-10)

PROCESS DESCRIPTION:

The purpose of quality assurance planning is to detail the auditing and review process and to provide a definitive list of the subsequent design documents that are subject to audit and review. The resulting quality assurance program will be performed separately from this initial activity.

To help guide the system developers through the quality assurance planning process, the work team should follow the approach presented below:

2.3.3.1 Research quality assurance references and PMP

2.3.3.2 Identify tools and techniques

2.3.2.3 Define responsibilities/schedules

2.3.3.4 Develop a specific review approach for the Selected Alternative from Economic Analysis

2.3.3.4.1 Identify control and reporting requirements
2.3.3.4.2 Examine reference material on reporting forms
2.3.3.4.3 Prepare checklists for reporting by milestone
2.3.3.4.4 Develop procedures for filling forms

2.3.3.5 Prepare Quality Assurance Plan (QA)

The quality assurance planning process will result in a document that presents the requirements of a quality assurance program tailored to meet the specific needs of the system being developed. A single quality assurance plan (and resulting quality assurance program) should be developed to address all parts of a system. In cases where parts of the system are themselves large, an abbreviated quality assurance plan should also be developed for each part, making extensive use of the existing quality assurance plan with substitutions tailored to that particular part of the system.

PROCESS NAME: Develop Configuration Management Program

PROCESS REFERENCE NUMBER: 2.3.4

PROCESS GUIDED BY: Configuration Management Plan (IRM-5231-09)

PROCESS DESCRIPTION:

The purpose of configuration management planning is to detail the identification, control, accounting, and auditing process and to provide a definitive list of the design documents that are subject to this process. The resulting configuration management program will be performed separately from this initial activity.

To help guide the system developers through the configuration management planning effort, we recommend that the work team follow the approach presented below:

2.3.4.1 Research configuration management references and PMP

2.3.4.2 Develop a configuration management policy for the selected alternative from economic analysis

2.3.4.3 Identify specific document control requirements

2.3.4.4 Develop specific control procedures

2.3.4.4.1 Define configuration identification procedures
2.3.4.4.2 Define configuration control procedures
2.3.4.4.3 Define status accounting procedures
2.3.4.4.4 Define configuration audit procedures

2.3.4.5 Assign organizational responsibilities by directive

2.3.4.6 Prepare Configuration Management Plan

The configuration management planning process will result in a document that presents the requirements of a configuration management program tailored to meet the specific needs of a system being developed. A single configuration management plan (and resulting configuration management program) should be developed to address all parts of a system. In cases of large systems composed of more than one project, a configuration management plan is developed for the control program with an addendum to that plan written for each project.

PROCESS NAME: Develop System Decision Paper Number One

PROCESS REFERENCE NUMBER: 2.4

PROCESS GUIDED BY: MCO P5231.1A

PROCESS DESCRIPTION:

The first System Decision Paper is the second formal management control point that allows responsible management to cancel the project. If a decision to continue the project is made, the necessary resources will be committed at this point in time to initiate the actual design of the system.

Appendix C

DIAGRAM 3 - DEFINITION AND DESIGN

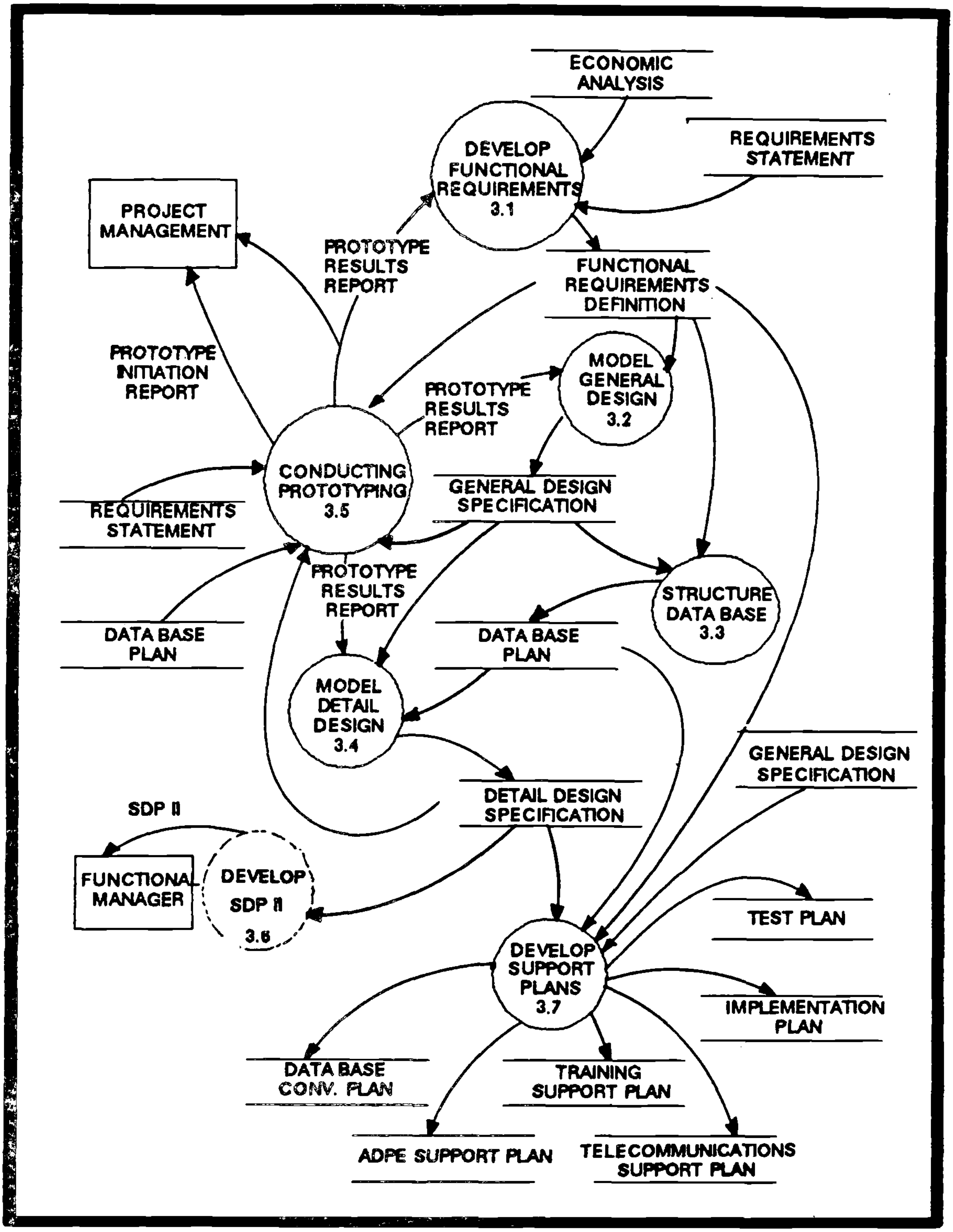

FIGURE C-01
DIAGRAM 3 - Definition and Design

PROCESS NAME: Develop Functional Requirements

PROCESS REFERENCE NUMBER: 3.1

PROCESS GUIDED BY: Project Development Standards
- Functional Requirements Definition (IRM-5231-04)
- Data Dictionary (IRM-5235-01)
- Economic Analysis (IRM-5236-03)

PROCESS DESCRIPTION:

This process performed by the project development team guided by the Project Manager, will develop the Functional Requirements Definition (FRD) following the instructions contained in IRM-5231-04. IRM-5235-01 will control development of all the data dictionary items associated with this effort.

The functional requirements development process is the first place in the development cycle that the functions of a system are explicitly defined in terms of specific processing requirements. Until this time, the system's objectives are only documented in general terms such as lists of functions, examples of desired reports, current system descriptions, and the Mission Element Need Statement (MENS). The purpose of a Functional Requirements Definition is to formalize the description of the user's needs and expand upon those requirements in a hierarchical presentation. This is an analysis and documentation of the essential processes before any design work is performed.

To help guide the system developers through the functional requirements process, we recommend that the work team follow the approach presented below:

3.1.1 Examine project documents for new requirements

3.1.2 Investigate the current system(s)

3.1.2.1 Assess current operations for existing capabilities
3.1.2.2 Prepare written description of current data processing organization, environment, and performance

3.1.3 Model the current physical system(s)

3.1.3.1 Develop context diagram of the current system
3.1.3.2 Prepare initial partitioning
3.1.3.2.1 Define first level breakdown (level 0) of business processes

3.1.3.2.2 Define the flow of information between those processes

3.1.3.3 Interview users to verify the definition of each process

3.1.3.4 Interview users to obtain more process information

3.1.3.5 Partition processes (iteratively) to functional primitives

3.1.3.6 Prepare mini-specifications for each functional primitive

3.1.3.7 Define interfaces

3.1.3.7.1 Document the man-machine boundary (automated processes)

3.1.3.7.2 Describe each flow of data across the man-machine boundary

3.1.3.8 Assemble material into a formal Current Physical Model

3.1.4 Review material with users and revise as necessary

3.1.5 Enter process, file, group, and item descriptions into DATAMANAGER

3.1.6 Model the current logical processes

3.1.6.1 Identify the current processes which are purely physical

3.1.6.2 Re-define the purely physical processes in terms of policy

3.1.6.3 Re-partition processes to incorporate redefined processes

3.1.6.4 Assemble material into a formal Current Logical Model

3.1.7 Review material with users and revise as necessary

3.1.8 Enter revised process, file, group, and item descriptions into DATAMANAGER

3.1.9 Model the expanded logical processes

3.1.9.1 Assess new requirements (identified by step 3.1.1)

3.1.9.2 Determine the processes required to provide the new functions

3.1.9.3 Perform additional analysis

3.1.9.3.1 Incorporate the new processes (re-partition) into the Current Logical Model

3.1.9.3.2 Perform analysis on data stores and assemble data schema

3.1.9.4 Assemble material into a formal New Logical Model

3.1.10 Review material with users and revise as necessary

3.1.11 Enter revised process, file, group, and item descriptions into DATAMANAGER

3.1.12 Assemble Functional Requirements Definition (FRD)

The functional requirements definition process will result in a document that presents data flow diagrams from three perspectives in sequence of analysis: (1) current physical processes; (2) current logical processes; and (3) new logical processes. The bulk of the material in the Functional Requirements Definition will be in Section 2, "Structured Specification." It is there that the data flow diagrams, the data dictionary, and the initial mini-specifications are presented.

PROCESS NAME:	Model General Design
PROCESS REFERENCE NUMBER:	3.2
PROCESS GUIDED BY	Project Development Standards - General Design Specification (IRM-5231-05) - Data Dictionary (IRM-5235-01)

PROCESS DESCRIPTION:

This process, performed by the project development team guided by the Project Manager, will develop the General Design Specification (GDS) following the instructions contained in IRM-5231-05. IRM-5235-01 will control development of all data dictionary items associated with this effort.

The general design modeling process is the first place in the development cycle that the new functions of a system are grouped according to physical similarities. By this time, the logical requirements are thoroughly documented in a set of leveled data flow diagrams developed during functional requirements definition, depicting each process only in terms of their essential functions. The purpose of general design is to now regroup these functions and then provide a few more levels of detail to define the physical manner in which the system will perform these functions. The general design modeling process is a transition step before actually determining the program's structural hierarchy.

To help guide the system developers through the general design process, we recommend that the work team follow the approach presented below:

3.2.1 Examine Functional Requirements Definition

3.2.2 List capability and capacity of data processing environment

3.2.3 Relate user organizations to data processing environment

3.2.4 Assess any requirements for distributed data processing

3.2.5 List any new performance criteria/requirements

3.2.6 Model the revised physical processes

3.2.6.1 Define man-machine boundary on New Logical Model
3.2.6.2 Refine definitions of all automated processes
3.2.6.3 Expand physical aspects of custodial processes
3.2.6.4 Define additional custodial functions as new processes

3.2.6.5 Refine definitions of data stores into data structure charts

3.2.6.6 Add security/control constraints onto each automated process

3.2.6.7 Assemble material into a New Physical Model format

3.2.7 Examine off-the-shelf products against physical model

3.2.8 Summarize all information requirements

3.2.9 Review material with users and revise as necessary

3.2.10 Enter revised process, file, group, and item descriptions into DATAMANAGER

3.2.11 Assemble General Design Specification (GDS)

The general design process will result in a document that presents the physical groups of data flow diagrams and will include dictionary entries. The bulk of the material packaged as the GDS will be in Section 2, "Structured Specification." It is there that the data flow diagrams, the data dictionary, and the mini-specifications are presented. Usually a single GDS is developed to address a single set of functional requirements, and would probably consist of about four or five levels of data flow diagrams and associated material.

PROCESS NAME: Structure Data Base

PROCESS REFERENCE NUMBER: 3.3

PROCESS GUIDED BY: Project Development Standard
- Data Base Plan (IRM-5231-11)

PROCESS DESCRIPTION:

This process performed by the project development team guided by the Project Manager, will structure the data base used on the project. This effort is accomplished following the instructions contained in IRM-5231-11.

The data base structuring process is the only place in the development cycle where data requirements are analyzed independent of program requirements. By this time, the data storage requirements are documented in a logical sense on the data flow diagrams developed during functional requirements definition, in terms of data stores and the resulting data schema. The purpose of data base design is to develop full data architecture that defines the data structure, access points, and subsequent navigation to meet the system's objectives. This is done in parallel with program design.

To help guide the system developers through the data base design process, we recommend that the work team follow the approach presented below:

3.3.1 Examine Functional Requirements Definition

3.3.2 Develop logical model

3.3.2.1 Break down user defined entities and attributes into data elements
3.3.2.2 Verify data elements with DATAMANAGER standard definitions
3.3.2.3 Develop Entity-Relationship (E-R) diagram with entities and relationships
3.3.2.4 Review E-R diagram into DESIGN MANAGER and develop third normal form logical model

3.3.3. Develop physical data model

3.3.3.1 Examine user access needs to DBMS
3.3.3.2 Compare physical constraints of DBMS environment with third normal form logical model
3.3.3.3 Review material with Data Base Manager/Data Base Administrator (DBM/DBA) and revise as necessary
3.3.3.4 Install and implement the data base with the DBMS

3.3.4 Develop DBMS maintenance process for users

3.3.4.1 Prepare file recovery and backup procedures
3.3.4.2 Define data element error criteria
3.3.4.3 Define data element error handling process

3.3.5 Enter file and data item names into DATAMANAGER

3.3.6 Assemble Data Base Plan (DBP) document

The data base design process will result in a document that presents both logical and physical perspectives of the data base structure, and will document its naming conventions, maintenance, and utilization characteristics. A single DBP (and resulting data base structure) should be developed to address the functional requirements of all parts of a system. In cases when additional parts of a system are derived after the data base design is complete, an abbreviated data base design effort should be undertaken as an assessment/modification effort that follows the same steps as the initial design but makes extensive use of the existing data base design with reference to its documentation.

PROCESS NAME:	Model Detailed Design
PROCESS REFERENCE NUMBER:	3.4
PROCESS GUIDED BY:	Project Development Standards - Detailed Design Specification (IRM-5231-06) - Data Dictionary (IRM-5235-01)

PROCESS DESCRIPTION:

This process, performed by the project development team guided by the Project Manager, will develop the Detailed Design Specification (DDS) following the instructions contained in IRM-5231-06. IRM-5235-01 will control development of all data dictionary items associated with this effort.

The detailed design modeling process is the first place in the development cycle that specific software, programming, and computer environment issues are addressed by the designers. By this time, the software analysis and design are well under way and documented, primarily through the use of data flow diagrams from the general design. The purpose of detailed design is to now transform these physical yet conceptual requirements into a set of very specific structure charts. This is the final step before program coding.

To help guide the system developers through the detailed design process, we recommend that the work team follow the approach presented below:

3.4.1 Examine General Design Specification (GDS)

3.4.2 Prepare list of programs and map to data flow diagrams

3.4.3 Define the input/output requirements

3.4.3.1 Prepare list of output reports
3.4.3.2 Prepare rough layouts for each report
3.4.3.3 List known/anticipated output data elements
3.4.3.4 List known/anticipated input data elements
3.4.3.5 Identify existing/new data file requirements
3.4.3.6 Review material with data base design personnel

3.4.4 Present full sideview structure chart of data flow diagrams

3.4.5 Re-work the system structure

3.4.5.1 Identify/add the master process for each program
3.4.5.2 Prepare individual structure charts for each program

3.4.5.3 Re-structure processes that are related by common transforms
3.4.5.4 Re-structure processes that have common transactions
3.4.5.5 Perform additional top-down factoring on process modules
3.4.5.6 Verify revised structure against functional requirements

3.4.6 Finalize the design details

3.4.6.1 Define each program in terms of jobs, steps, and load units
3.4.6.2 Write module specifications for all process modules
3.4.6.3 Identify and define data couples between connected modules
3.4.6.4 Prepare state transition diagrams for all input/output modules
3.4.6.5 Finalize format for all reports and data entry screens
3.4.6.6 Assemble material into structured format

3.4.7 Review material with users and revise as necessary

3.4.8 Enter revised process, file, group, and item description into DATAMANAGER

3.4.9 Assemble Detailed Design Specification (DDS)

The detailed design process will result in a document that presents the final levels of detail developed for a system's design, primarily using the mini-specs from the general design as starting points to develop structure charts. The bulk of the material packaged as the DDS will be in Section 2, "Structured Design." It is there that the structure charts, the final data dictionary, and the final module specifications are presented. One or more DDS should be derived from a single general design, and each should consist of a dozen or more structure charts and associated material.

PROCESS NAME:	Conduct Prototyping
PROCESS REFERENCE NUMBER:	3.5
PROCESS GUIDED BY:	Project Development Standards - Prototyping Standard (IRM-5231-18) - Data Dictionary (IRM-5235-01)

PROCESS DESCRIPTION:

During the definition and design activity, the project development team may decide to conduct prototyping as an aid to verifying and validating the functional requirements, general design, and/or detail design specifications.

This process will be guided and controlled by IRM-5231-18. When prototyping is tasked by project management and has been completed, the outputs of the process will be used as input to the functional requirements definition, general, and/or detailed design process. Prototyping will be initiated with data inputs from the following sources:

- Project Management
  - o Constraints
  - o Task Orders
- Project Libraries
  - o Constraints
  - o IRM-5231-18
  - o IRM-5235-01
- Project Documents
  - o Requirements Statement
  - o Functional Requirements Definition
  - o General Design Specifications
  - o Detail Design Specifications
  - o Data Base Plan

When the prototyping process has been completed, copies of the outputs resulting from the effort will be placed in the project documentation for use by other personnel and processes. These outputs are:

- Prototyping Initiation Report

- Prototyping Results Report

Copies of the Prototyping Initial Report and Prototyping Results Reports are forwarded to project management for review and approval.

PROCESS NAME: Develop System Decision Paper Number Two

PROCESS REFERENCE NUMBER: 3.6

PROCESS GUIDED BY: MCO P5231.1

PROCESS DESCRIPTION:

This System Decision Paper is the third formal management control point that allows responsible management to cancel the project. If a decision to continue the project is made, the necessary resources will be committed at this point in time to initiate the actual development of the system.

This process is initiated when the detailed design specification is completed and responsible management makes the decision to continue development. The completed detailed design specification is the base from which the decision paper is documented.

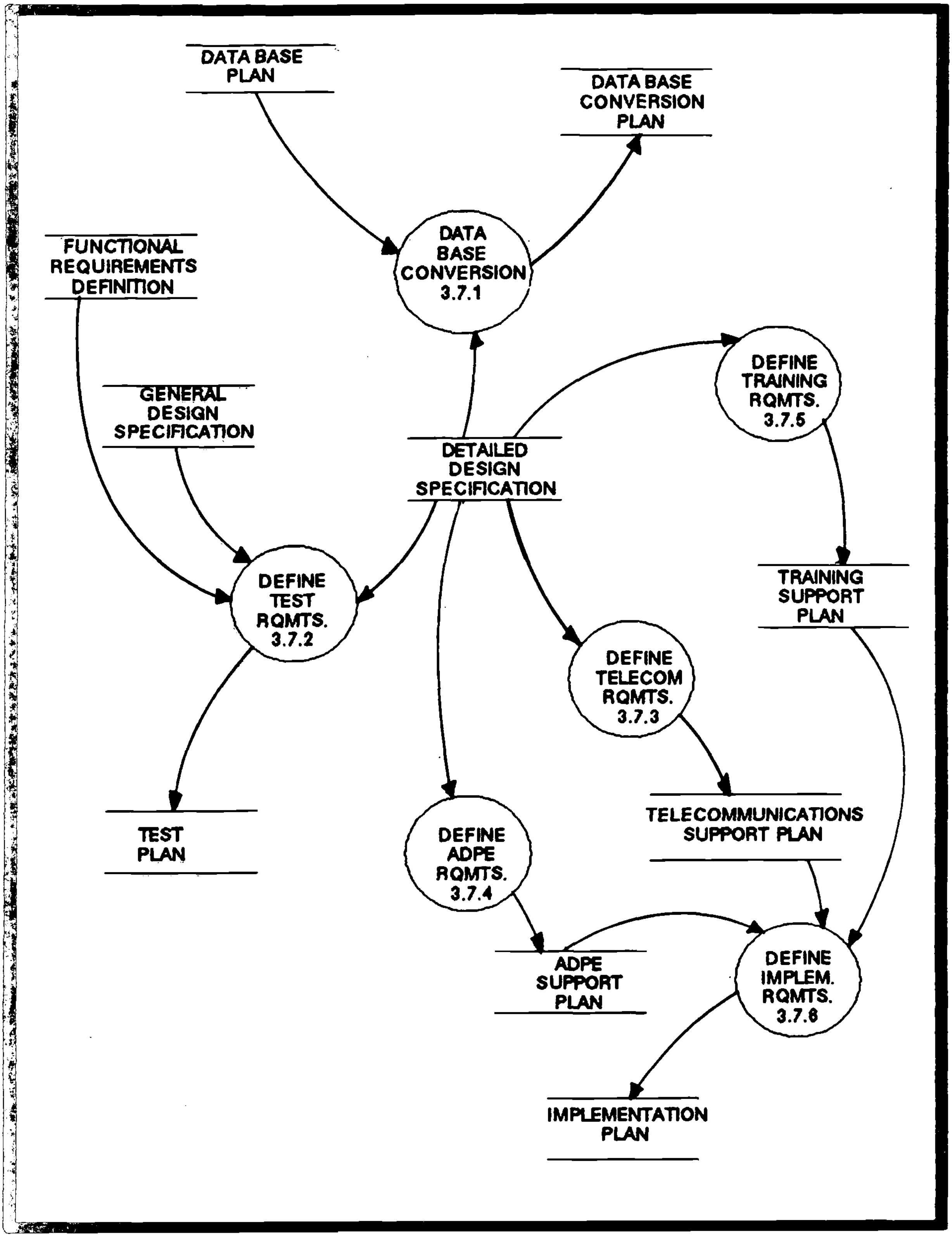
DATA BASE PLAN
DATA BASE CONVERSION PLAN
DATA BASE CONVERSION 3.7.1
FUNCTIONAL REQUIREMENTS DEFINITION
DEFINE TRAINING RQMTS. 3.7.5
GENERAL DESIGN SPECIFICATION
DETAILED DESIGN SPECIFICATION
TRAINING SUPPORT PLAN
DEFINE TEST RQMTS. 3.7.2
DEFINE TELECOM RQMTS. 3.7.3
TEST PLAN
TELECOMMUNICATIONS SUPPORT PLAN
DEFINE ADPE RQMTS. 3.7.4
ADPE SUPPORT PLAN
DEFINE IMPLEM. RQMTS. 3.7.6
IMPLEMENTATION PLAN

FIGURE C-02
DIAGRAM 3.7 - Develop Support Plans

PROCESS NAME: Data Base Conversion

PROCESS REFERENCE NUMBER: 3.7.1

PROCESS GUIDED BY: Data Base Conversion Plan (IRM-5231-13)

PROCESS DESCRIPTION:

The data base conversion process is an optional part of the development cycle provided specifically to load the data base from existing manual or computerized data files, if any have been in use. By this time, the new data base has been designed to meet the needs of the application programs. The purpose of data base conversion is to plan and perform the migration from an existing data structure in terms of a sequence of events, the control mechanisms, and any audit requirements. This is done in parallel with new system programming.

To help guide the system developers through the data base conversion process, we recommend that the work team follow the approach presented below:

3.7.1.1 Examine data base documentation

3.7.1.2 Examine Detailed Design Specification

3.7.1.3 Examine existing data base structure and content

3.7.1.4 Develop the approach for orderly data conversion

3.7.1.4.1 Plan conversion and define controls
3.7.1.4.2 Document data base and data processing environment
3.7.1.4.3 Define storage, loading, and data capture requirements
3.7.1.4.4 Define data conversion verification process
3.7.1.4.5 Document transition program/JCL requirements
3.7.1.4.6 Examine existing documentation to identify change areas
3.7.1.4.7 Define parallel operating requirement during transition
3.7.1.4.8 Assemble Data Base Conversion Plan (DBC)

3.7.1.5 Review material with DBM/DBA

3.7.1.6 Convert data base according to plan

The data base conversion process will result in a converted data base and a document that guides that process. That document presents both the detailed plan to convert the data from one file structure to another as well as documenting any time-phasing requirements to maintain the processing integrity of the current systems during the transition period. A single DBC should be developed to include all parts of a system being developed, however, the data conversion may be performed separately for various parts of the system upon completion of the detailed design for those parts.

PROCESS NAME: Define Test Requirements

PROCESS REFERENCE NUMBER: 3.7.2

PROCESS GUIDED BY: Test Plan (TP) (IRM-5231-14)

PROCESS DESCRIPTION:

The test requirements definition process is the place in the development cycle at which the desired functions and capabilities of the system are transformed into specific tests and test parameters overlayed on the detailed design of the system. The purpose of test planning is to perform initial testing of program modules before delivery, and to define the full tests to be performed for final acceptance. The final acceptance testing will be performed separately from this initial activity.

To help the system developers define the test requirements, we recommend that the work team follow the approach presented below:

3.7.2.1 Examine design specifications (functional, general, detailed)

3.7.2.2 Define specific tests for verification of design

3.7.2.2.1 List unit/module tests to be performed
3.7.2.2.2 List component tests to be performed
3.7.2.2.3 List integration tests to be performed
3.7.2.2.4 List functional tests to be performed
3.7.2.2.5 List system tests to be performed

3.7.2.3 Identify testing locations for each test

3.7.2.4 Prepare milestone chart for pre-delivery testing

3.7.2.5 Define testing support requirements

3.7.2.5.1 List equipment requirements for each test
3.7.2.5.2 List software requirements for each test
3.7.2.5.3 List personnel requirements for each test

3.7.2.6 Document test requirements

3.7.2.6.1 Define pre-delivery testing responsibilities
3.7.2.6.2 Identify and document testing materials
3.7.2.6.3 Document security issues related to test data
3.7.2.6.4 Assemble sections 1, 2, and 3 of Pre-Delivery TP

3.7.2.7. Review Pre-Delivery TP with development programmers

3.7.2.8 Define specific tests for validation of design

3.7.2.8.1 List acceptance tests to be performed
3.7.2.8.2 List structural tests to be performed

3.7.2.9 Package testing information

3.7.2.9.1 Examine Pre-Delivery TP for common information
3.7.2.9.2 Review milestone chart to include acceptance tests
3.7.2.9.3 List additional requirements of acceptance testing
3.7.2.9.4 List materials/responsibilities of acceptance testing
3.7.2.9.5 Assemble material into Acceptance TP

3.7.2.10 Review Acceptance TP with users

The test planning process will result in a document that details a schedule of events and activities to be done to adequately test that the program operates properly and addresses the original requirements for the system. A separate TP should be developed for each part of a system that is being developed and delivered separately.

PROCESS NAME: Define Telecommunications Requirements

PROCESS REFERENCE NUMBER: 3.7.3

PROCESS GUIDED BY: Telecommunications Support Plan (IRM-5239-05)

PROCESS DESCRIPTION:

The telecommunications requirements definition process is one place in the development cycle at which hardware support requirements of a new system design are documented. The purpose of specifically defining the telecommunications requirements is to ensure that adequate capacity and capability are available and/or planned to accommodate the new system before the design is taken to production.

To help the system developers define the telecommunications requirements, we recommend that the work team follow the approach presented below:

3.7.3.1 Examine capabilities of existing telecommunications facilities

3.7.3.2 Examine detailed design to determine telecommunications needs

3.7.3.3 Develop plan to meet telecommunications needs

3.7.3.3.1 Document deficiencies (if any) in current telecommunications
3.7.3.3.2 Define course of action to meet telecommunications requirements
3.7.3.3.3 Assess feasibility and document costs
3.7.3.3.4 Assemble Telecommunications Support Plan (TSP)

3.7.3.4 Review TSP with management

3.7.3.5 Revise and submit final TSP

The telecommunications planning process will result in a TSP. A separate TSP should be developed for each part of the system that is being developed and delivered separately to ensure that the total support requirements are addressed.

PROCESS NAME: Define ADPE Requirements

PROCESS REFERENCE NUMBER: 3.7.4

PROCESS DESCRIPTION: ADPE Support Plan (IRM-5231-12)

PROCESS DESCRIPTION:

The ADPE requirements definition process is one place in the development cycle at which hardware support requirements of a new system design are documented. The purpose of specifically defining the ADPE support requirements is to ensure that adequate processing, storage, and interface equipment is available and/or planned to accommodate the system before the design is taken to production.

To help the system developers define the ADPE requirements, we recommend that the work team follow the approach presented below:

3.7.4.1 Examine capabilities of existing data processing equipment

3.7.4.2 Examine detailed design to determine needs

3.7.4.3 Develop plan to meet equipment needs

3.7.4.3.1 Document deficiencies (if any) of current equipment
3.7.4.3.2 Define course of action to meet equipment requirements
3.7.4.3.3 List site preparation and logistics requirements
3.7.4.3.4 List general operational requirements
3.7.4.3.5 Define procurement recommendations/responsibilities
3.7.4.3.6 Assemble ADPE Support Plan

3.7.4.4 Review ADPE Support Plan with management

3.7.4.5 Review and submit final ADPE Support Plan

The ADPE planning process will result in an ADPE Support Plan. A separate ADPE Support Plan should be developed for each part of the system that is being developed and delivered separately to ensure that the total support requirement are addressed.

PROCESS NAME: Define Implementation Requirements

PROCESS REFERENCE NUMBER: 3.7.6

PROCESS GUIDED BY: Implementation Plan (IP) (IRM-5231-16)

PROCESS DESCRIPTION:

The implementation requirements definition process is the first place in the development cycle at which preparations are made for operations. The purpose of documenting the planned approach is to coordinate the installation of a new system, possibly composed of many separate parts, and to minimize any disruption of current systems. The resulting implementation will be done separately from this initial activity.

To help the system developers define the implementation requirements, we recommend that the work team follow the approach presented below:

3.7.6.1 Examine documentation for ADPE, telecommunications, and training needs

3.7.6.2 Develop approach for orderly migration to new system(s)

3.7.6.2.1 Define implementation process and schedule
3.7.6.2.2 Document hardware/software system components
3.7.6.2.3 Define site preparation requirements for implementation
3.7.6.2.4 Assemble IP

3.7.6.3 Review IP with affected personnel

3.7.6.4 Review and submit final IP

The implementation planning process will result in an IP. A separate IP should be developed for each part of the system that is being developed and delivered separately, each one at least 30 days prior to the subsequent implementation activity.

PROCESS NAME: Define Training Requirements

PROCESS REFERENCE NUMBER: 3.7.5

PROCESS GUIDED BY: Training Support Plan (IRM-5231-15)

PROCESS DESCRIPTION:

The training requirements definition process is the final step in the development cycle before actual deployment and operations. The purpose of planning training is to detail the requirements and objectives for educating the user in the operation of the developed system. The resulting training will be done separately from this initial activity.

To help the system developers define the training requirements, we recommend that the work team follow the approach presented below:

3.7.5.1 Identify/appoint the training coordinator for the system

3.7.5.2 Develop list of participants for training activities

3.7.5.3 Examine existing USMC training regulations/programs

3.7.5.4 Examine detailed design for training insights

3.7.5.5 Develop plan to meeting training needs

3.7.5.5.1 Document scope and overview of training requirements
3.7.5.5.2 Identify training materials and facilities anticipated
3.7.5.5.3 Describe training course content for the system
3.7.5.5.4 Assemble Training Support Plan (TRP)

3.7.5.6 Review TRP with affected personnel

3.7.5.7 Revise and submit final TRP

The training planning process will result in a TRP. A separate training plan should be developed for each part of the system that is being developed and delivered separately, each one at least 30 days prior to the subsequent training activity.

Appendix D

DIAGRAM 4 - SYSTEM DEVELOPMENT

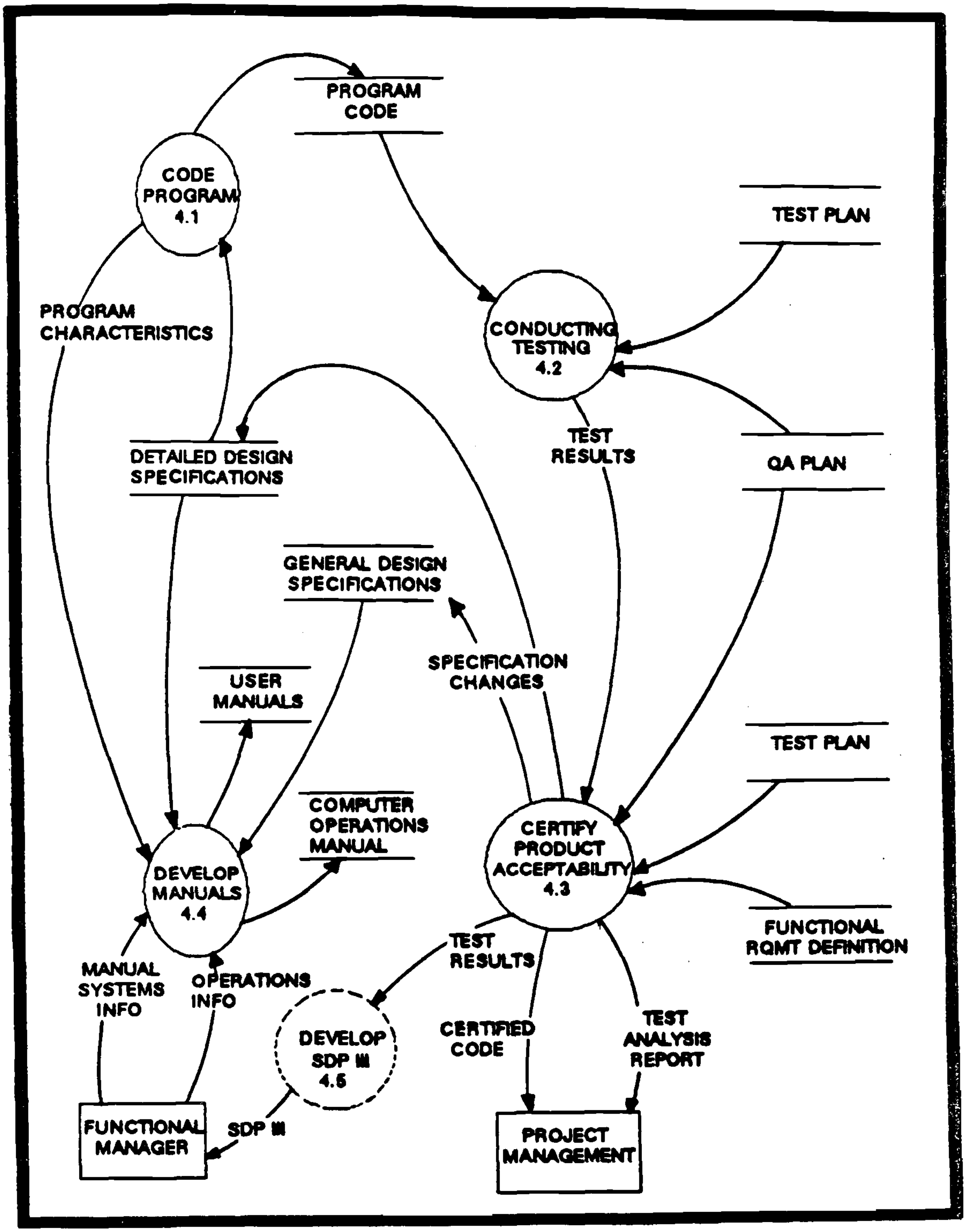

FIGURE D-01
DIAGRAM 4 - System Development

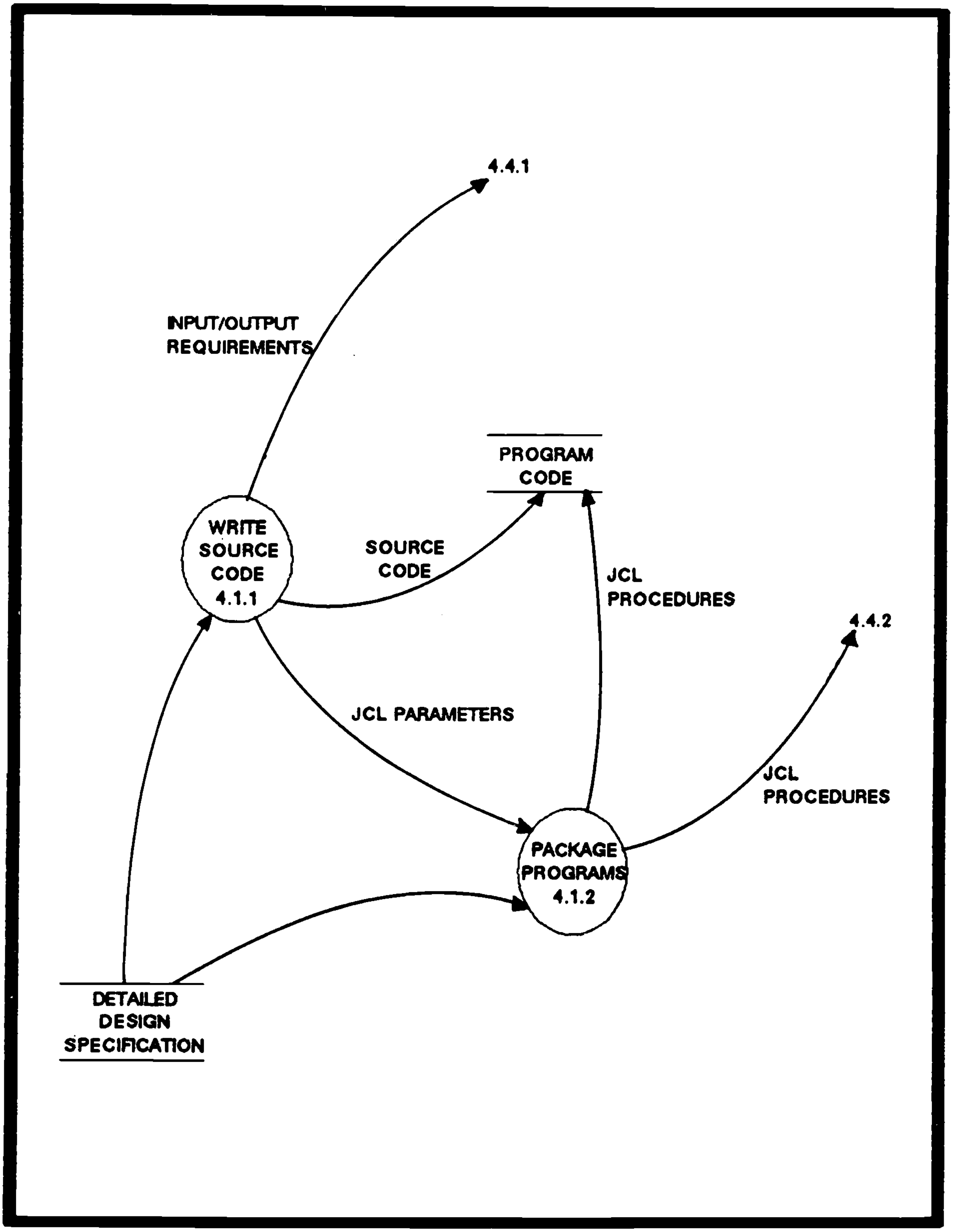

FIGURE D-02
DIAGRAM 4.1 - Code Program

PROCESS NAME: Write Source Code

PROCESS REFERENCE NUMBER: 4.1.1

PROCESS GUIDED BY: Programming Standard (IRM-5234-01)

PROCESS DESCRIPTION:

The source code generation is the place in the development cycle at which all the design work is transformed into its final and most tangible results. By this time, the design is documented in a hierarchical manner such that source code can be written directly from the module specifications developed during detailed design. The purpose of programming is to write, compile, and unit test this code.

To help system developers write the source code, we recommend that the work team follow the approach presented below:

4.1.1.1 Examine Detailed Design Specifications

4.1.1.2 Code programs and key into system

4.1.1.2.1 Identify separate load units on structure charts
4.1.1.2.2 Identify separate programs within load units
4.1.1.2.3 Code program modules from structured module specs

4.1.1.3 Compile programs and correct initial errors

4.1.1.4 List input/output requirements

4.1.1.4.1 Prepare JCL parameters
4.1.1.4.2 Prepare DBMS interface calls
4.1.1.4.3 List and describe input parameters and forms
4.1.1.4.4 List and describe output parameters and reports
4.1.1.4.5 Document data entry, edit, and error handling process
4.1.1.4.6 Document processing error messages

PROCESS NAME: Package Programs

PROCESS REFERENCE NUMBER: 4.1.2

PROCESS GUIDED BY: Library Management System (IRM-5233-06)

PROCESS DESCRIPTION:

The program packaging is the place in the development cycle at which all the components of the design are arranged in terms of job sequencing and control. By this time, the design is documented in a hierarchical manner such that job compositions and characteristics are defined on the program structure charts. The purpose of program packaging is to develop the job control information.

To help the system developers package the programs, we recommend that the work team follow the approach presented below:

4.1.2.1 Examine Detailed Design Specification

4.1.2.2 Define jobs and job steps

4.1.2.2.1 Prepare list of programs and utilities
4.1.2.2.2 Organize programs into discrete jobs

4.1.2.3 Examine JCL parameters of programs/files

4.1.2.4 Code JCL parameters

4.1.2.4.1 Document local security controls
4.1.2.4.2 Determine backout/recovery process
4.1.2.4.3 List all checkpoints and restart points
4.1.2.4.4 Flowchart job submission/scheduling process
4.1.2.4.5 Breakdown a list of steps for each job
4.1.2.4.6 Write JCL code for each job/job-step
4.1.2.4.7 Write symbolic parameters for catalogued procedures

4.1.2.5 Estimate data storage/format requirements

PROCESS NAME: Conduct Testing

PROCESS REFERENCE NUMBER: 4.2

PROCESS IS GUIDED BY: Project Development Standards

- Test Plan (IRM-5231-14)
- Quality Assurance Plan (IRM-5231-10)

PROCESS DESCRIPTION:

The project team will conduct testing as specified in the Test Plan in accordance with procedures outlined in the Quality Assurance Plan. Upon completion of testing, the test results in the form of a technical report will be forwarded to project management for review. Test results will also provide the basis for the fourth formal management decision point that initiates implementation of the developed system.

PROCESS NAME:	Certify Product Acceptability
PROCESS REFERENCE NUMBER:	4.3
PROCESS GUIDED BY:	Project Development Standards - Test Plan (IRM-5231-14) - Functional Requirements Definition (IRM-5231-04) - Quality Assurance Plan (IRM-5231-10)

PROCESS DESCRIPTION:

This process will validate that the system fully satisfies the user requirements as stated in IRM-5231-04. The testing activity ensures that the required functional capabilities are operational in the hardware, software, and DBMS environment in which it is expected to be deployed.

Upon completion of this process, a report of activity results will be sent to project management for information and review purposes.

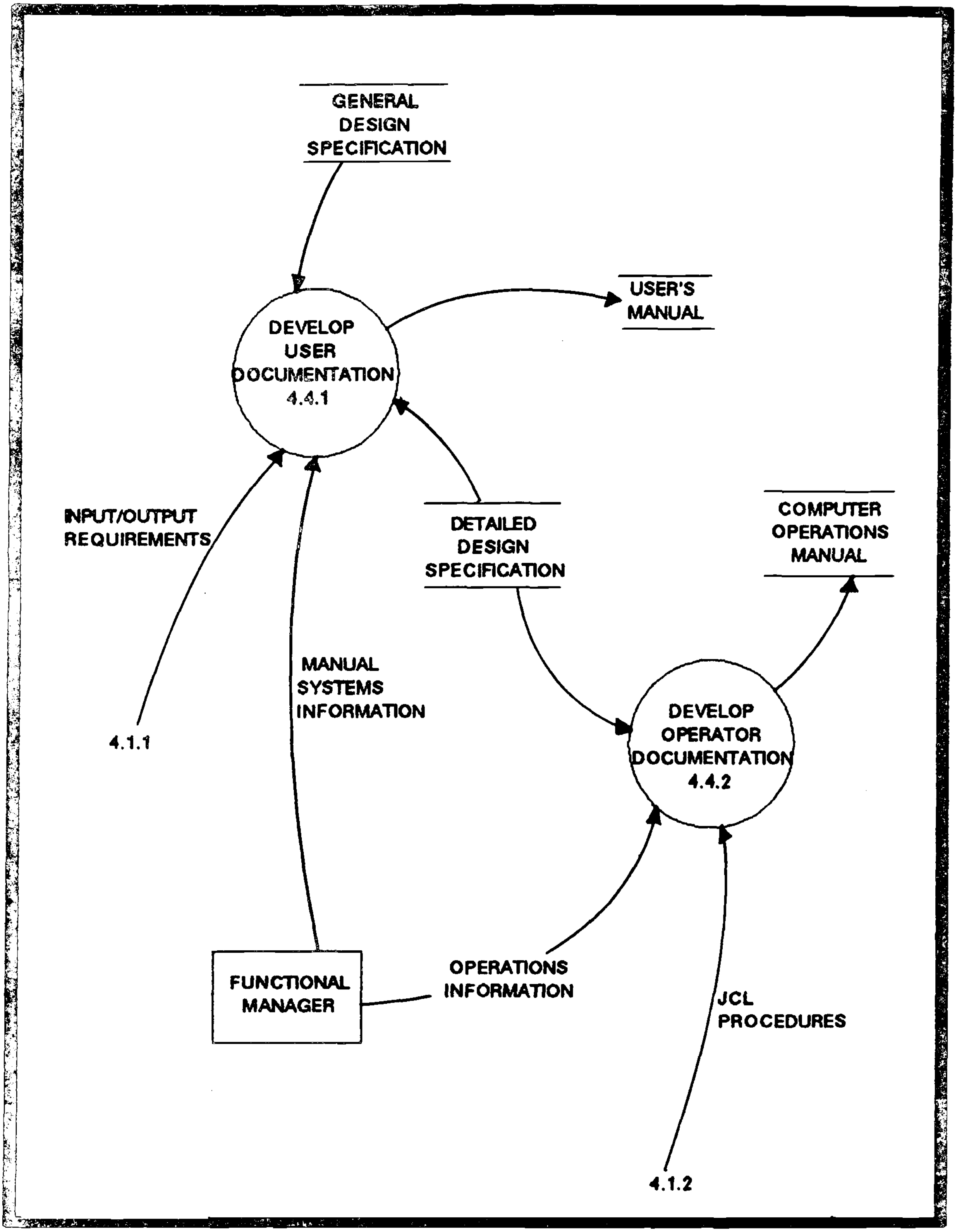
GENERAL DESIGN SPECIFICATION
USER'S MANUAL
DEVELOP USER DOCUMENTATION 4.4.1
INPUT/OUTPUT REQUIREMENTS
DETAILED DESIGN SPECIFICATION
COMPUTER OPERATIONS MANUAL
MANUAL SYSTEMS INFORMATION
4.1.1
DEVELOP OPERATOR DOCUMENTATION 4.4.2
FUNCTIONAL MANAGER
OPERATIONS INFORMATION
JCL PROCEDURES
4.1.2

FIGURE D-03
DIAGRAM 4.4 - Develop Manuals

PROCESS NAME: Develop User Documentation

PROCESS REFERENCE NUMBER: 4.4.1

PROCESS GUIDED BY: User Manual (UM) (IRM-5231-07)

PROCESS DESCRIPTION:

The user documentation process will result in a document that presents operational information at a level so that user personnel can successfully operate the system. The bulk of the material in a Users Manual should be in Section 2, "Application," and most of that in paragraph 2.1, "Process Description." It is there that structure and access characteristics of the program are broken down by the author and presented from the user's perspective. Several Users Manuals may result from each group of detailed designs, one for each major topic identified in the design, although the topics may not correspond one-to-one with the separate design specifications. These manuals are intended to cover those user topics not already addressed in user type publications that are part of the formal Marine Corps directives systems. We recommend that the work team follow the approach presented below:

4.4.1.1 Describe source programs

4.4.1.1.1 Make list of functional topics
4.4.1.1.2 Define classes of users
4.4.1.1.3 Prepare user/topic matrix

4.4.1.2 Assign groups of topics to separate manuals if appropriate

4.4.1.3 Describe operational procedures

4.4.1.3.1 Develop list(s) of specific sub-topics
4.4.1.3.2 Identify user procedures for each sub-topic
4.4.1.3.3 Write brief specification of each user procedure
4.4.1.3.4 Organize specifications within "Process Description" section

4.4.1.4 Assemble full draft of UM(s)

4.4.1.5 Review material with users and revise as necessary

4.4.1.6 Assemble final UM(s)

PROCESS NAME: Develop Operator Documentation

PROCESS REFERENCE NUMBER: 4.4.2

PROCESS GUIDED BY: Computer Operations Manual (IRM-5231-08)

PROCESS DESCRIPTION:

The operator documentation process will result in a document that provides an overview of the job in terms of the job steps and programs, and will define the specific operational procedures with the forms in Section 3, "Job Run Book." It is there that day-to-day operations information is presented. A single Computer Operations Manual will usually encompass all of the detailed designs generated for one or more previous general designs. The development team should follow the approach presented below:

4.4.2.1 Prepare Job Run Book(s)

4.4.2.1.1 Writer overview description of each job
4.4.2.1.2 Writer brief description of each job
4.4.2.1.3 Prepare a run book for each job

4.4.2.2 Assemble material into draft Computer Operations Manual (COM)

4.4.2.3 Review material with operations and revise as necessary

4.4.2.4 Assembly final COM

PROCESS NAME: Develop System Decision Paper Number Three

PROCESS REFERENCE NUMBER: 4.5

PROCESS GUIDED BY: MCO P5231.1A

PROCESS DESCRIPTION:

This System Decision Paper is the fourth formal management control point that allows responsible management to cancel the project. If a decision to continue the project is made, the necessary resources will be committed at this point in time to initiate the implementation of the tested system, and development/documentation of user and computer operation manuals. This decision paper is based upon the results of code testing.

Appendix E

DIAGRAM 5 - DEPLOYMENT

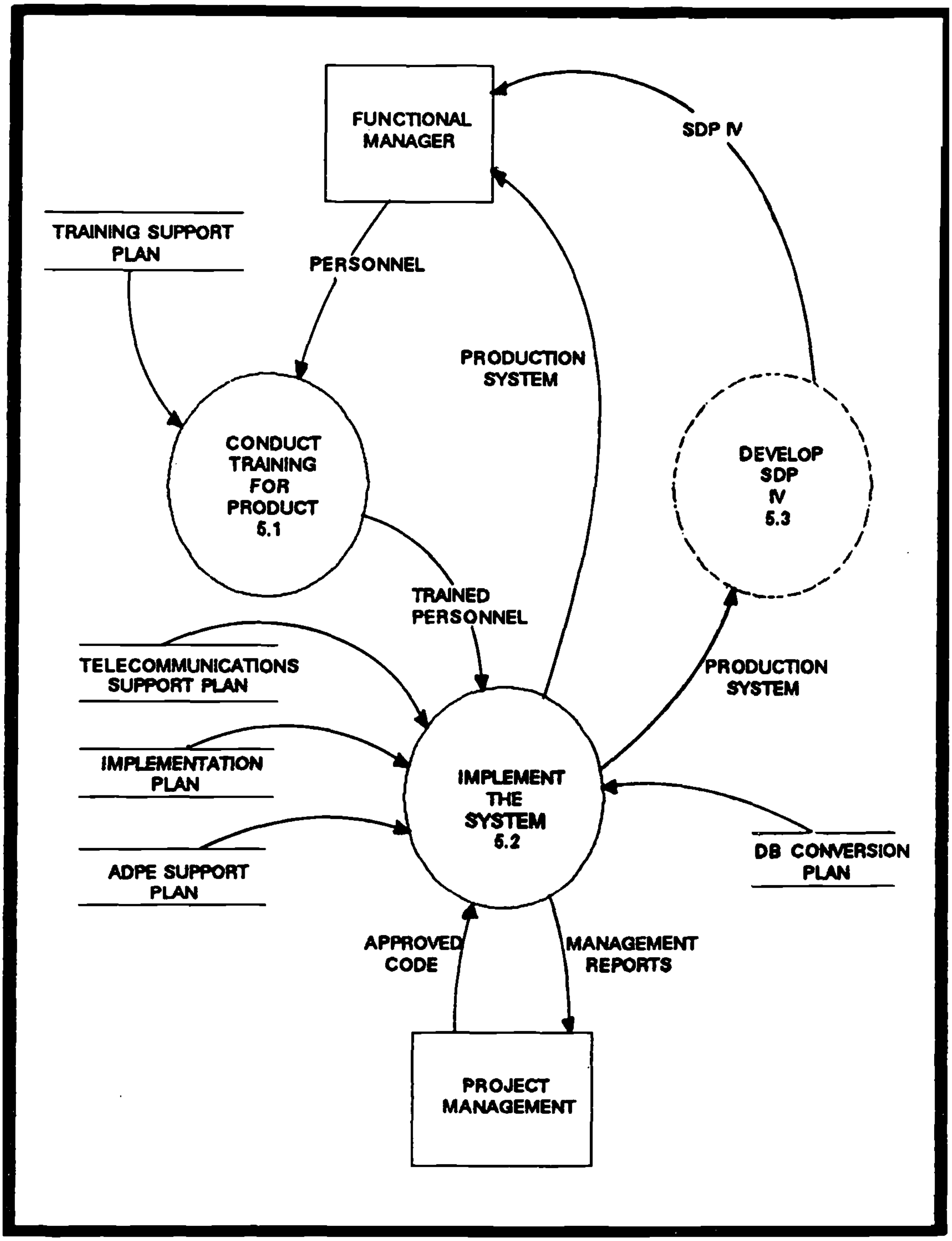

FIGURE E-01
DIAGRAM 5 - Deployment

PROCESS NAME: Conduct Training For Product

PROCESS REFERENCE NUMBER: 5.1

PROCESS GUIDED BY: Project Development Standards
- Training Support Plan (IRM-5231-15)

PROCESS DESCRIPTION:

The Training Support Plan is implemented. At a minimum, personnel who are to be trained under this plan will include functional users, operations personnel, and program maintenance personnel.

PROCESS NAME: Implement The System

PROCESS REFERENCE NUMBER: 5.2

PROCESS GUIDED BY: Project Development Standards

- Telecommunications Support Plan (IRM-5239-05)
- Implementation Plan (IRM-5231-16)
- ADPE Support Plan (IRM-5231-12)
- Data Base Conversion Plan (IRM-5231-13)

PROCESS DESCRIPTION:

When a task order from project management is obtained, a team of personnel following the above referenced plans will implement the system. The implementation team should include the existing development team and those personnel identified in the training plan that should be trained to effectively operate and maintain the ongoing system. This team should also implement the software and hardware purchased and/or developed by any particular vendor. Management reports documenting this activity will be transmitted to project management and the vendors for contract and management information purposes.

PROCESS NAME:	Develop System Decision Paper Four
PROCESS REFERENCE NUMBER:	5.3
PROCESS GUIDED BY:	MCO P5231.1A

PROCESS DESCRIPTION:

This System Decision Paper is the fifth formal Management Control point. Each system will be re-evaluated on a periodic basis to assure the functional requirements have been satisfied and that the technical specifications are effective and efficient.

Appendix F

DATA DICTIONARY

OVERVIEW. The data dictionary is a set of definitions for the data contained in a leveled data flow diagram set. The data items are defined with narratives and occasionally with examples, to aid in data item definitions. These are needed as the methodology covers a number of classes of systems. The actual value and even components of a data item may differ depending on the system being developed. Using the data dictionary format, one can define the necessary component for the specific needs and purposes.

The following three symbols are used to document entries in the data dictionary:

= An equal sign means "is defined as" or "is composed of."

+ A plus sign means "and."

() Parentheses mean "optional." If more than one item is within the parentheses, the whole set is optional.

Acceptance Testing = The final process of verification and validation in order to determine if the product will operate within the environment and in accordance with the user requirements.

Activity Precedence Diagram = The graphical portrayal of distinct work segments. This indicates the logical precedent relationships between activities.

Allocated Baseline = A configuration identification document or a set of such documents formally designated and fixed at a specific time during the life cycle of a configuration item (CI) or a computer software configuration item (CSCI). Baselines, plus approved changes from those baselines, constitute the current configuration identification. For configuration management the allocated base-line is the initial approved allocated configuration identification.

Application Configuration Management Plan Standard (IRM-5231-09) = The guideline establishing uniform procedures for development of a CM Plan and implementation of configuration management for all automated information systems under authority of the Project Directorate.

Section 1 Introduction Purpose and Scope + Definition + Configuration Management Summary +

Section 2 Applicable Documents +

Section 3 Configuration Management Organization +

Section 4 Configuration Identification +

Section 5 Configuration Control +

Section 6 Software Configuration Authentication +

Section 7 Configuration Status Accounting +

Section 8 Interface Management +

Section 9 Configuration Audits +

Section 10 Subcontractor/Vendor Control

Approvals/Disapprovals = A flow of information out of a process or data source for an approval/disapproval determination on arrival at its destination.

Approved Code = Certified code which has also been determined acceptable to the USMC in consideration of SDP III.

Automatic Data Processing Equipment (ADPE) Support Plan = The plan that assures the required delegations of procurement authority (DPA) have been received, that funds are available, and that the necessary systems software, hardware, telecommunications, and logistical support is acquired to support the implementation of applications.

Automated Data Processing Equipment (ADPE) Support Plan Standard (IRM-5231-12) = The guideline that defines the format and content for all ADPE created for the Project.

Section 1 Introduction Purpose and Scope + Objectives + Responsibilities + References + Terms and Abbreviations +

Section 2 Background and Approval Present ADPE Configuration + ADPE Deficiencies + Additional ADPE Requirements +

Section 3 Conclusions and Recommendations +

Section 4 ADPE Support Requirements Environmental Requirements + Logistics Requirements +

Section 5 Operational Requirements Personnel Requirements + Training Requirements + Maintenance Requirements +

Section 6 Procurement Authority Requirements Central Funding Requirements + Procurement Authority Procedures + Funding Responsibilities + ADPE Acquisitions + Support Equipment and Services + Contracted Service Responsibilities

Automated System Information = That group of information describing the automated portion of a system in total.

Batched Transmission = A transmission that is not on-line, such as store and forward.

Benefit = The total value to be derived, less any disadvantages encountered, by the USMC operational forces that will rely on the use of the system.

Capital Cost = The non-recurring or first time cost to acquire a part of a system at the beginning of its replacement cycle.

Category = A grouping of cost or benefit values.

Certified Code = Program code which has been determined to meet all the requirements of the Test Plan based upon examination of the Test Results.

Child Diagram = A lower level graphic representing the further partitioning of a parent mini-system from the previous level.

Code Checking = The act of verifying the conformance of computer code to standards and naming conventions.

Cohesion = The degree to which a process or module is independent measured by the amount of shared data.

Component Testing = The testing of a sub-structure of software less than a full function, as determined from the structure chart.

Computer Operations Manual (COM) = Those manuals that document the detail operating instructions for a system.

Computer Operations Manual Standard (IRM-5231-08) = The guideline establishing the format and content for all developer prepared manuals that define the operational procedures to run an application.

Section 1 System Overview System Summary + System Operation + Program Inventory + File Inventory + Processing Overview + Security +

Section 2 Job Run Description General Information + Run Inventory + Job Schedule +

Section 3 Job Run Book Run Book Forms + Item Definition

Computer Software Configuration Item (CSCI) = Software components such as compilers, data base maintenance programs, report generation programs, executive programs, operational programs, and self-testing packages. See the definition for configuration item.

Configuration Control Board (CCB) = The decision-making and review body which decides upon the acceptability of configuration changes. The CCB reviews changes to determine that changes reflect applicable requirements and that all impacts of changes are addressed.

Configuration Item (CI) = An aggregation of hardware/software, or any of its discrete portions, which satisfies an end use function and is designated by the government for configuration management. During development and initial production, CIs are only those specification items that are referenced directly in a contract (or an equivalent in-house agreement). During the operation and maintenance period, any reparable item designated for separate procurement is a configuration item. (Refer to DOD Directive 5010.19).

Configuration Management Plan (CM) = The plan by which all changes to baselines will be controlled, and evaluations of the change impact on project cost and schedules will be made.

Configuration Status Accounting = The recording and reporting of the information that is needed to manage configuration effectively, including a listing of the approved configuration identification, the status of proposed changes to configuration, and implementation status of approved changes.

Constraints = Those things that are defined as limits, boundaries, or parameters that a development effort must stay within.

Context Diagram = The top-level graphic representation of the system or subsystem under study. It depicts all the net inputs to and outputs from the system, but shows no decomposition or partitioning of the system.

Contract Data Requirements List (CDRL) = Lists specific contract requirements by number and cross references those numbers to detailed Data Item Descriptions (DIDs).

Contracting Officer (CO) = The CO monitors contractor activities to assure compliance with the terms of the contract. The CO issues approvals required under the contract so that work proceeds in an orderly manner. The CO also modifies the contract where appropriate, accepts deliverables, and closes out the contract upon completion.

Contracting Officer's Technical Representative (COTR) = The COTR represents the CO in technical matters involving deliverables. The COTR notifies the appropriate personnel on the availability of deliverables, ensures that dates are met, and acts as the liaison between the contractor and user activities in resolving minor deficiencies in the deliverables. The COTR also maintains the status of the deliverables through a logging process.

Control Job = A job that produces Privacy Act Information and the customer designates, in writing, the personnel that are authorized to submit the work request to process the job and receive its output.

Coupling = The degree of module interdependence.

Cost = The total of all required expenditures, less any savings, incurred by or on behalf of the USMC in the development of the system.

Critical Job = A job designated by the customer as being critical and that must be completed. Any personnel required to correct, restart, and complete the job will be recalled.

Crossover Resource = Information from one environment to another.

Current Logical Model = The model defining the fundamental or essential functions which the existing system performs as well as the fundamental or essential stored data to support those functions.

Current Physical Model = The model defining the operational environment of the existing system.

Custodial Functions = Those activities that establish and maintain the current system's essential memory by acquiring and storing data needed by the fundamental to essential functions.

Data Base Conversion Plan (DBC) = The plan that details the methodology and documentation necessary to control the transfer of existing data into a new data base environment.

Data Base Conversion Plan Standard (IRM-5231-13) = The guideline that defines the format and content of the data base conversion plan. This plan addresses issues concerning the hardware/software environment, conversion and capture of data, conversion scheduling, and supporting administrative activity.

Section 1 Introduction Purpose and Scope + Relationship to Other Activities + Responsibilities + Reference + Terms and Abbreviations +

Section 2 Data Base Conversion Planning and Control Task Analyst + Task Precedence + Project Schedule + Project Organization + Project Control + Identification of Deficiencies + Corrective Action +

Section 3 Technical Activities Environment Definition + Data Conversion + Space Allocation + Data Loading + Data Synthesis + Data Verification

Data Base Management (DBM) = Approval levels for the establishment of user requirement for data base manipulation.

Data Base Management System (DBMS) = The results of a structuring process in the definition and design phase that addresses all issues relative to the establishment, maintenance, and usage of the DBMS.

Data Base Plan (DBP) = The plan that provides the guidelines to design a data base structure.

Data Base Plan Standard (IRM-5231-11) = The guidelines by which the Data Base Management System (DBMS) Plan will be created to address all issues relative to the establishment, maintenance, and usage of the DBMS.

Section 1 Introduction Purpose and Scope + Relationship to Other Activities + Terms and Abbreviations + Reference +

Section 2 DBMS Implementation Responsibilities + Logical Data Model + Logical Data Base + Inversion Characteristics + Physical Data Model + Physical Structure + Data Compression + Naming Conventions + Installation and Implementation + Verification +

Section 3 DBMS Maintenance Responsibilities + DBMS Software + Change Procedures + Data Integrity + Data Security + Performance Monitoring +

Section 4 DBMS Utilization Data Base Access + Application Program Interfaces + Queries and Report Generation + Teleprocessing Interfaces

Data Couple = An arrow with a ring or small circle on one end representing data. The arrow indicates the direction of the data flow. A couple with an open ring represents data, a couple with a closed or darkened ring indicates a control flag. The name of the couple is written next to it.

Data Dictionary (DD) Standard (IRM-5235-01) = The guideline that defines the format and content of the DD Specification to ensure that all issues relative to usage of the Data Dictionary have been adequately addressed.

Section 1 Introduction Purpose and Scope + Relationship to Other Activities + Responsibilities + Terms and Abbreviations + Reference +

Section 2 Operation and Use Data Definition + Member Types + Member Control + Member Relationships + Data Responsibilities + Multiple Dictionaries + Dictionary Features + Data Manager Reporting

Data Flow = A curved line, or arc, that represents the flow of information or data. Each arc will have an arrow indicating the direction of the data flow. The name of the data flow is written next to the arc and must summarize the content of the data flow. Rules governing the format of the Data Flow Numbers are in the Data Dictionary Standard.

Data Navigation = Entry point into a data base in terms of the first data file used and all subordinate data files.

Data Store = A set of parallel lines on a data flow diagram that represent a repository of data.

Design Model = A model of a part of the system as defined in a General Design or Detailed Design Specification.

Detailed Design Specifications (DDS) = The document that defines the users software requirements for new, changed, or enhanced applications and define the software environment necessary to meet the proposed systems objectives.

Detailed Design Specification Standard (IRM-5231-06) = The guideline that provides the documentation requirements and evaluation criteria necessary to ensure production of a well-defined Detailed Design Specification of the user's software requirements.

Section 1 General Objective + Scope + Responsibilities + Detailed Design Action Plan +

Section 2 Structured Design Specification +

Section 3 Supporting Documentation New Performance Characteristics + ADPE Environment

Deviation = A specific written authorization, granted prior to the development of an item, to depart from a particular performance or design requirement of a specification, drawing or other document for a specific number of units or a specified period of time. A deviation differs from an engineering change in that, an approved engineering change requires corresponding revision of the documentation defining the affected item, whereas a deviation does not contemplate revision of the applicable specification or drawing. (DOD-Standard-480).

Direct Access Tables = Data tables accessed during program execution.

Documentation Testing = The determination of whether the user documentation is accurate, clear, and useful. It is also the determination of whether the procedures outlined in the objectives and user documentation are met, such as procedures to be followed by the system operator, data-base administrator, terminal user, and functional user. Any examples illustrated in the documentation should be included in test cases and executed by the program.

Economic Analysis (EA) = The document that provides an evaluation of the relative worth of each alternative proposed in a feasibility study.

Economic Analysis Standard (IRM-5236-03) = The guidelines that provide the documentation requirements and evaluation criteria necessary to ensure development of a complete Economic Analysis.

Section 1 Introduction +

Section 2 Objective +

Section 3 Assumptions +

Section 4 Cost/Benefit Analysis Life Cycle Period + Basis for Comparison + Alternatives + Cost Categories + Cost Measurements + Benefit Categories + Benefit Measurements +

Section 5 Evaluation +

Section 6 Conclusions +

Section 7 Recommendations

Engineering Change = An alteration in the configuration of a configuration item or items, delivered, to be delivered, or under development, after formal establishment of its configuration identification. (DOD-Standard-480)

(1) Engineering Change, Class I = An engineering change is defined as Class I if the criteria of DOD-Standard-480, Configuration Control-Engineering Changes, Deviations and Waivers, are met. These are generally changes in form, fit, or function of an item.

(2) Engineering Change, Class II = An engineering change is defined as Class II if the criteria of DOD-Standard-480, Configuration Control-Engineering Changes, Deviations and Waivers, are met. These are generally changes to correct errors in documentation.

Engineering Change Proposal (ECP) = A term which includes both a proposed engineering change and the documentation by which the change is described and suggested. (DOD-Standard-480).

Entity = An object, or store of data. The entity is represented on the ERD as a box.

Entity Relationship Diagram (ERD) = A graphical network which describes the stored data layout of a system.

Equipment/Hardware Testing = The determination of whether the system operates on the hardware minimum and maximum configuration as specified in the objectives.

Equivalent Annual Amount = The amount of money needed in each year for a specified period in order to pay for one or more purchases of different amounts in particular years, assuming that the value will increase at a prescribed interest rate.

Essential Functions = Those activities or processes that are fundamental to a system's performance.

Essential Memory = The total of all data elements remembered by the system and required by the essential functions.

Event List = Describes the events in the environment to which the system must respond.

Existing System Information = The body of pertinent information that describes the current in-place hardware and software configurations.

Facility Testing = The determination of whether each facility in the objectives was actually implemented.

Figure 0 = The next level graphic representation after the context diagram, showing a high level partitioning of the system under study.

Functional Baseline = A configuration identification document or a set of such documents formally designated and fixed at a specific time during a CI's/CPCI's life cycle. Baselines, plus approved changes from those baselines, constitute the current configuration identification. For configuration management the functional baseline is the initial approved Functional Configuration Identification.

Functional Requirements Definition (FRD) = The document that defines the users functional requirements for new, changed, or enhanced applications.

Functional Requirements Definition Standard (IRM-5231-04) = The guideline that provides the documentation requirements and evaluation criteria necessary to ensure a well-defined Functional Requirements Definition of the user's functional requirements.

CURRENT PHYSICAL MODEL +

Section 1 General Objective + Scope + Responsibilities + Current Physical Model Action Plan +

Section 2 Structured Specification +

Section 3 Supporting Documentation Current Organizational Content + Current ADPE Environment + Current Performance Characteristics +

CURRENT LOGICAL MODEL +

Section 1 General Objective + Scope + Responsibilities + Current Logical Model Action Plan +

Section 2 Structured Specification +

NEW LOGICAL MODEL +

Section 1 General Objective + Scope + Responsibilities + Functional Requirements Definition Action Plan +

Section 2 Structured Specification +

Section 3 Supporting Documentation Data Schema Documentation + Summary of New Requirements

Functional Testing = An attempt to locate errors in the performance of the functions which the software is required to provide.

Funding = The supply of dollars at a particular point in time to pay for specific expenditures.

Future Worth = The amount of money needed later to make payment for a purchase in one or more previous years, assuming that the value will increase at a prescribed interest rate.

Gantt Chart = A diagram displaying an estimated schedule for project completion. It is a graphical representation of scheduling relationships. In its basic form, the Gantt chart is a graph of resource allocation over time.

General Design Specifications (GDS) = The document that defines the system requirements for new, changed, or enhanced applications and define the environment necessary to meet the systems proposed stated purpose.

General Design Specification Standard (IRM-5231-05) = The guideline that provides the documentation requirements and evaluation criteria necessary to ensure a well-defined General Design Specification of the user's functional requirements.

New Physical Model

Section 1 General Objective + Scope + Responsibilities + General Design Action Plan +

Section 2 Structured Specification +

Section 3 Supporting Documentation Organizational Context + ADPE Environment + New Performance Characteristics

Hardware Products = Computing machinery, its associated input/output peripherals, and all of the miscellaneous support equipment and devices that accompany each of them.

Hierarchical = A mutually exclusive breakdown of categories into a more precise set of sub-categories; nested groups.

Impact Analysis = Determination of the affects a portion of the system design has on the total project effort.

Implementation Plan (IP) = The plan that provides necessary information to the users and data processing personnel to implement functional segments of the system, and to achieve operational status at each installation site.

Implementation Plan Standard (IRM-5231-16) = The guideline establishing uniform procedures for development of an implementation plan to accomplish the installation of functional segments of the system, and achieve the segments' operational status at each installation site.

Section 1 Introduction Purpose + References + Terms and Abbreviations +

Section 2 Implementation Process Description + Organizational Responsibilities/Tasks + Tracking Methods + Implementation Schedule + Support Materials + Training + Personnel Orientation + Site Personnel Requirements + Security +

Section 3 System Components System Software + Application Software + Jobstream Conversion + System Verification +

Section 4 Site Information Site Identifications + Site Schedule + Hardware and Software Requirements + Facility Requirements + Orientation Facilities + Implementation Team Workspace + Detailed Implementation Procedures

Implemented System = System in production after acceptance testing is complete.

Inspection and Acceptance (IA) Standard (IRM-5231-17) = The guidelines by which the receipt, inspection, evaluation, and acceptance of deliverables are governed so as to be handled in a consistent manner among all developers.

Instructional Training Products = Training supplied as a specific product itself. This includes both initial and ongoing training.

Integration Team = The entity responsible for assisting the project in the overall management and development of the system.

Integration Testing = The testing of the total system including the interfaces between and among the system parts.

Interactive Transmission = Any more than one transmission, such as voice and data.

Input/Output (I/O) Requirements = List of data items supplied by and/or returned to the user by the computer program as designed.

Job Control Language (JCL) Parameters = List of computer operating system variables (such as file numbers) and characteristics (such as file types) required for the computer program as designed.

JCL Procedures = The completed set of computer operating system commands to control the flow of program jobs and job steps, and to set the JCL Parameters to their proper values at program run time.

Job Run Dependencies = Definition of the sequence in which jobs in a job stream must be run, in order to obtain correct results.

Job Stream Name = Identifier for a job stream.

Library Management System (LMS) Standard (IRM-5233-06) = The guideline established for development and implementation of a Library Management System (LMS) Standard Operating Procedure (SOP).

Section 1 General Purpose and Scope + Responsibilities +

Section 2 Library Management Library Data Sets + Partitioned Data Sets + Direct Access Data Sets + Naming Conventions +

Section 3 Local Procedures Statement of Work + Library Management Systems Procedures +

Section 4 Exported Systems Exported Systems + Release Schedule + LOADLIB Destination + LOADLIB Request + Methods of Transmitting LOADLIB(s) + Specially Handled LOADLIB(s) + Follow-up Process +

Section 5 Imported Systems External MCCDPA Sponsored Class I Systems + Contractor Developed Systems + DoD Sponsored and Other Class I Systems

Life Cycle = The period of time which a system will be fully utilized, after which it is then obsolete and un-usable.

Logical Data Model = See New Logical Model.

Management Reports = The body of reports generated during a development process to provide management information in a timely manner.

Man-Machine Dialogue Standard (MDS) (IRM-5234-02) = Conventions for the design of the user interfaces for a computer program.

Manual System Information = That group of information describing the manual portion of the system in total.

Measurement = The value(s) of costs and/or benefits.

MENS Information = That body of information required to be gathered, analyzed, and documented in the Mission Element Need Statement (MENS).

Mini-specification (mini-spec) = Statement of the logical requirements governing the transformation of input data flows into output data flows at the functional primitive mini-system or process level.

Mini-system (or process) = Drawn as a circle on a data flow diagram to represent a transformation of incoming data into outgoing data.

Miscellaneous Project Information = A random array of information required by various processes at different times during the development cycle.

Mission Element Needs Statement (MENS) = The document that describes the mission deficiency and justifies exploration of alternative solutions for satisfying the deficiency.

Section 1 Name of Project +

Section 2 Mission and Objectives +

Section 3 Responsibilities and Accountability Project Manager + Users +

Section 4 Authorities +

Section 5 Relationships and Changes of Communication +

Section 6 Organization +

Section 7 Location and Administrative Support +

Section 8 Project Transition/Disestablishment

Modeling = The simulation or portrayal of a system or entity within a defined environment.

Module = A box that represents a set of instructions that will cause the transformation of data in a planned way. The name of the module is written within the box and will describe the function the module is to perform each time it is called.

Module Specification = Statement of the rules governing the processing of associated data couples.

Network Procedures Manual (NPM) Standard (IRM-5239-01) = The guideline that ensures the reliability of the network and the coordination and control of changes being made to the network.

Section 1 Objective +

Section 2 Scope +

Section 3 Approach +

Section 4 Documentation + Front-End Processor GEN Submission Standards + Network Software Changes + Class-of-Service Tables + Network Software Responsibility + Reporting + Network Back-Up + Network Back-Off

New Logical Model = The model defining the fundamental or essential functions which the new system must perform as well as the fundamental or essential stored data to support those functions.

New Physical Model = The model defining the specification of the proposed operational environment.

On-line System Reliability Testing = The determination of whether the on-line functions can accept and correctly process any input without causing total on-line system failure.

Operations Information = That group of information describing the systems environment that an application operates in currently or will in the future.

Packaging = The process of grouping model components, typically modules in structure charts (DDS), into programs, steps and jobs.

Partitioning = The process of breaking a model (GDS) into subordinate parts, successive levels of detail.

Periodic Cost = The recurring or annual cost to operate or maintain a part of a system throughout its replacement cycle.

Performance Testing = The determination of whether the system performs within the response times and throughput rates specified in the objectives.

Personnel = Persons who will operate/use the system.

Physical Configuration Audit (PCA) = The formal examination and comparison of the "as built" configuration of a CI unit against its technical documentation in order to establish the CI's initial product configuration identification. (DOD-Standard-480)

Physical Data Model = See New Physical Model.

Planning Requirements = Objectives/constraints used to manage the system development project.

Pre-Delivery Test Results Report = The document outlining results of the pre-delivery testing of the unit/module, integration, function and system testing required to be performed by the developing organization.

Preferred Alternative = The alternative that is selected from those listed in the feasibility study that is presented to the steering group for approval concurrently with the economic analysis.

Product Baseline = A configuration identification document or a set of such documents formally designated and fixed at a specific time during a CI's/CSCI's life cycle. Baselines, plus approved changes from those baselines, constitute the current configuration identification. For configuration management the product baseline is the initial approved or conditionally approved product configuration identification. The approval of the product baseline generally indicates the hardware and software is entering the operational phase.

Production System = Approved Code having been successfully implemented and used for regular operation.

Program Characteristics = I/O Requirements & JCL Procedures.

Program Code = The product resulting from a coding process conducted using the Detail Design Specifications.

Program Evaluation and Review Techniques (PERT) = A tool for the planning and scheduling of large, nonrepetitive projects.

Programming Standard (PS) (IRM-5234-01) = The guideline establishing a methodology for assuring the readability and maintainability of source code.

Section 1 Environment Standards Job Control Language + Libraries + Standardized Procedures + Other Facilities +

Section 2 Support Software LIBRARIAN + ROSCOE + TSO + IBM Utilities

Section 3 Application Development Standards COBOL + NATURAL + FOCUS + ADABAS + VSAM + ADA Conventions +

Section 4 Evaluation Criteria

Project Deliverable Style Manual (SM) = The document that provides guidance on standardization and consistency of documentation. Front Cover + Cover Letter + Title Page + Record of Changes + Table of Contents + (List of Figures) + Text + (Bibliography) + Appendices + (Attachments) + (Index) + Distribution List + Back Cover.

Project Deliverable Style Manual Standard (IRM-5230-02) = The guideline that standardizes the format for written documentation to ensure consistency with respect to content requirements and format.

Project Libraries = The source(s) of miscellaneous information utilized in development efforts.

Mission Element Needs Statement Information + Miscellaneous Project Information + Project Development Standards + Existing System Information + Constraints + Operations Information + Computer Operations Manuals + User Manuals + System Decision Paper III,

(See Individual Definitions)

Project Documentation = The set of formal specifications and plans generated in the design of a system, governed by the SDM standards.

Project Management Plan (PMP) = The working document that assists the Project Manager in the day-to-day management of the project.

Project Management Plan Updates = The results from conducting certain processes in the methodology that are utilized to update the project management plan.

Project Manager (PM) = Individual responsible for the development of the system as assigned by the functional manager for the system.

Project Manager Charter = A documented charter that grants the Project Manager authority to provide management functions for a development effort.

Section 1 Name of Project +

Section 2 Mission and Objectives +

Section 3 Responsibilities and Accountability Project Manager + Users +

Section 4 Authorities +

Section 5 Relationships and Channels of Communications+

Section 6 Organization +

Section 7 Location and Administrative Support +

Section 8 Project Transition/Disestablishment

Project Plans = Project Management Plan + Quality Assurance Plan + Configuration Management Plan + Data Base Plan + Data Base Conversion Plan + Telecommunications Support Plan + Training Support Plan + Test Plan + Implementation Plan + Automatic Data Processing Equipment Plan

(See Individual Definitions)

Project Request = A formal request from a user that will initiate the development process.

Project Standards = The standards used as guidelines for the processes defined in the system development methodology.

Standard	Standard Title
IRM-5230-02	Project Deliverable Style Manual
IRM-5231-01	SDM - Overview
IRM-5231-02	SDM - Developer Perspective
IRM-5231-04	Functional Requirements Definition
IRM-5231-05	General Design Specification
IRM-5231-06	Detailed Design Specification
IRM-5231-07	Users Manual
IRM-5231-08	Computer Operations Manual
IRM-5231-09	Application Configuration Management Plan
IRM-5231-10	Quality Assurance Plan
IRM-5231-11	Data Base Plan
IRM-5231-12	ADPE Support Plan
IRM-5231-13	Data Base Conversion Plan
IRM-5231-14	Test Plan
IRM-5231-15	Training Support Plan
IRM-5231-16	Implementation Plan
IRM-5231-17	Inspection and Acceptance
IRM-5231-18	Prototyping Standard
IRM-5231-19	Project Management Plan
IRM-5231-20	Requirements Statement
IRM-5233-06	Library Management Standard
IRM-5234-01	Programming Standard
IRM-5234-02	Man-Machine Dialogue
IRM-5234-04	Naming Conventions
IRM-5235-01	Data Dictionary
IRM-5236-03	Economic Analysis
IRM-5239-01	Network Procedures Manual
IRM-5239-05	Telecommunication Support Plan

Proposed Steering Group Charter = The proposed authority to be granted a Steering Group being formed for a particular project.

Proposed Project Manager Charter = A proposal outlining the Project Manager's functions for a development effort and definitions of the authority required to carry them out (see Project Manager Charter).

Prototype = A working model built specifically to identify design problems such as the integration of components and the interaction of the user with these components.

Prototyping Initiation Report = A report provided to management that documents the object of the prototyping effort, the prototype specification, and how the effort will be controlled and funded.

Prototyping Results Report = A report to management that documents the results obtained from the prototyping effort.

Prototyping Specification = The documentation of the implementation environment, prototype design, and the prototype's operation.

Prototyping Standard (PRS) (IRM-5231-18) = The guideline that governs the documentation produced when prototypes are used during structured systems analysis and design of new, changed, or enhanced applications.

Prototype Initiation Report

Section 1 General Problem Statement + Scope + Responsibilities + Prototype Action Plan +

Section 2 Prototype Specification +

Section 3 Prototype Control Resource Estimates + Approval Authority

Prototype Results Report

Section 1 General Problem Statement + Scope + Responsibilities + Prototype Action Plan +

Section 2 Prototype Results Summary of Results + Detail Results +

Section 3 Prototype Control Resource Expenditures + Approval Authority

Quality Assurance (QA) Plan = The plan whereby the quality of the product to be delivered is both controlled and assured.

Quality Assurance (QA) Plan Standard (IRM-5231-10) = The guideline that defines the format and content of all QA Plans to be developed for a project. Adherence to this standard will assure uniform, applicable plans that provide for the required QA functions.

Section 1 General Purpose and Scope + Relationship to Other Activities + Responsibilities + References + Terms and Abbreviations +

Section 2 Tools and Techniques Reviews + Walkthroughs + Audits + Code Checking + Testing + Subcontractor Control +

Section 3 Plan Implementation Tasks + Responsibilities + Resource Requirements +

Section 4 Control and Reporting Monitoring Overall Development Status + Identification of Deficiencies + Support of Decision Making + Corrective Action.

Reading Grade Level (RGL) = The level at which a document should be written commensurate with the capability of the target audience.

Recovery Testing = The determination of whether the system recovers from programming errors, hardware failures, and data errors according to the objectives.

Relationship = A set of connections that relate two or more entities. The relationship is represented on the entity relationship diagram as a diamond.

Relative Context Diagram = See Context Diagram.

Replacement Cycles = The period of time after which a part of a system must be replaced (when less than the system life cycle).

Replacement Life Span = Number of years in the replacement life cycle of a part of a system.

Requirements Statement (RS) = The document that lists the users' functional requirements to be satisfied by the development effort.

Section 1 General Purpose + Point of Contact +

Section 2 Current System Project Reference + Problem Description + Existing System +

Section 3 Required Capabilities Capability Identification + Organizational Structure + Interface with other systems + Operating Environment + Communication Requirements + Classification + Performance Requirements + Requirements for back up Capability +

Section 4 Validation of User Requirements

Responsibility Matrix = A graphic description describing of organizational responsibilities and tasks associated with actual implementation.

Risk = In assessing possible cost reduction strategies, the amount of anticipated savings expected to be lost due to a failure to achieve the proper result when instituting the cost reduction strategy; measured as probability of failure times the penalty cost.

Savings = A reduction in cost due to a change in the method or system used; a negative cost.

Security Testing = The process of attempting to devise test cases that subvert the system's security checks as specified in the objectives.

Software Changes Proposals (SCPs) = A document prepared to propose changes to established computer program baselines. The SCP is used in instances where the change will be an addition, deletion, or modification of a capability which would be evident to the user/operator of the system and/or affect the compatibility of the software with previous unchanged versions.

Software, Computer = A collection of associated computer programs and computer data required to enable the computer hardware to perform computational or control functions.

Software Products = The programming deliverables that drive the hardware, and the applications that operate on the hardware.

Software Trouble Report (STR) = A software trouble report is a report used when fielded software is not in conformance with the approved baseline documentation which is under configuration control.

Source Code = The completed set of computer program commands, written in the selected programming language, to accomplish the requirements of the computer program as designed.

Specification Changes = Modifications to a system under development, as determined by failures of that system to meet the requirements of the Test Plan during the testing process.

State Transition Diagram = A diagram showing the "cause/reaction" relationship between any modules which can be identified as steady state (i.e. at rest requiring a "cause" to change state). This is usually a sub-set of a module structure chart (DDS) showing only those modules that have a user-interface, where the user supplies the "cause" and the system performs a "reaction."

Steering Group Charter (SGC) = A documented charter granting authority to a group of personnel to conduct progress reviews at intermediate review points, and make decisions at each life cycle decision point.

Storage Testing = The determination of whether the system meets the storage requirements as specified in the objectives.

Stress Testing = The determination of whether the system can perform using a peak volume of data over a short period of time.

Structural Testing = The attempt to detect errors in the code structure, the statement, condition, and decision coverage in regard to the integration of a specific module.

Structure Chart = A graphic tool for depicting modules, the hierarchy, and organization of those modules, and the data that interfaces those modules.

Structured English = A subset of the English language, using limited syntax and grammar to convey prime functions in an organized manner, commonly used when writing mini-specifications and similar to pseudo-code.

Structured Specification = A model of the system under investigation and the documentation of that system.

Subject Architecture = The result of the initial user/topic analysis. This provides the topics to be used as user manual subjects.

Subordination = An unnamed arrow connecting two modules indicating the control of one module over another the direction of the arrow indicates subordination.

System Decision Paper I (SDP I) = The first formal decision point that allows management to cancel the project or commit the necessary resources to initiate definition and design activities for the system.

Section 1 Executive Summary Purpose + Discussion +

Section 2 Essential Issues General + Schedule + Resources + Problem Area +

Section 3 Approval

System Decision Paper II (SDP II) = The second format decision point that allows management to cancel the project or commit the necessary resources to initiate actual development of the system.

Section 1 Executive Summary Purpose + Discussion +

Section 2 Essential Issues General + Schedule + Resources + Problem Area

Section 3 Approval

System Decision Paper III (SDP III) = The third formal decision point that allows management to cancel the project or commit the necessary resources to initiate implementation of the application system and document manuals.

Section 1 Executive Summary Purpose + Discussion +

Section 2 Essential Issues General + Schedule + Resources + Problem Area +

Section 3 Approval

System Decision Paper IV (SDP IV) = The fourth formal decision point based upon a periodic review of the system to ensure that the functional requirements are still being satisfied.

System Life Span = Number of years in the system life cycle.

System Testing = The comparison of the developed system to its original objectives.

Target Environment = The computer installation in which a tested, developmental software system is installed to produce an operable production system.

Task Audit Procedures = Methods of monitoring activities associated with implementation.

Task Matrix = See Responsibility Matrix.

Technical Representative (TR) = The TR is responsible for the inspection and evaluation of deliverables.

Telecommunications Network Naming Standard (TNS) (Contained in Naming Conventions - IRM-5234-04) = The guideline that ensures all authorized users have access to network resources regardless of the users' physical location and the location of the network resource.

Section 1 Objective +

Section 2 Scope +

Section 3 Approach +

Section 4 Documentation Standards Resource Identification + MCDN Subarea Assignments + Network Control Program Name + Group Macro Label for Line Groups + Line Macro Label for Lines

Telecommunication Support Plan (TSP) = The plan that provides a detailed description of the teleprocessing and telecommunication traffic volume, communication interface points and methods, communication system utilization plans, and the unique communication, hardware, facility, or service requirements for the development product.

Telecommunication Support Plan Standard (IRM-5239-05) = The guideline that provides a structure for the development of a Telecommunications Support Plan (TSP) that defines the teleprocessing requirements for the system.

Section 1 Introduction Scope + Constraints + Limitations + Responsibilities + Terms and Abbreviations +

Section 2 Background and Approach Requirements Solution + Current Telecommunications Deficiencies +

Section 3 Telecommunications Requirements Methodology + Planning Factors + Component Requirements + Support Concepts + Summary +

Section 4 Conclusion Technical Feasibility + Operational Feasibility + Recommendation + Costs

Teleprocessing Interface = An input/output device used to facilitate the transfer of data.

Test Analysis Report = The document that outlines the results of testing, allocates responsibility for deficiency correction and follow-up, provides a basis for preparation of the statement of test completion, and establishes user approval in the operation of the system. (See IRM-5231-14, Test Plan Standard, for format)

Test Plan (TP) = The plan that directs and guides the testing of the product that has been developed.

Test Plan Standard (IRM-5231-14) = The guidelines that define the format and content of all formal Pre-Delivery and Acceptance Test Plans to be developed in support of the Project.

Pre-Delivery Test Plan

Section 1 Introduction Purpose of the Pre-Delivery Test Plan + Project References + Terms and Abbreviations + Privacy and Protection of Test Plan +

Section 2 Responsibilities Developer's Responsibilities + Approval Authority Responsibilities + Project Development Office Responsibilities + CDPA/RASC/DFASC Responsibilities + Functional User Responsibilities +

Section 3 Pre-Delivery Testing Activities Pre-Test Activities + Testing Methodologies +

Section 4 Pre-Delivery Test Results Pre-Delivery Test Results + Volume Test + Stress Test + Performance Test + Known System Deficiencies

Acceptance Test Plan

Section 1 Introduction Purpose of the Acceptance Test Plan + Project References + Terms and Abbreviations + Privacy and Protection of Test Plan +

Section 2 Responsibilities Developer's Responsibilities + Approval Authority Responsibilities + Project Development Office Responsibilities + CDPA/RASC/DFASC Responsibilities + Functional User Responsibilities +

Section 3 Test Plan System Description + Testing Schedule + Testing Location + Milestone Chart + Equipment Requirements + Software + Personnel + Personnel Responsibilities + Training Orientation + User Training + Operator Training + Maintenance Training + Test Material + Deliverable Material + Site-Supplied Material + Security and Privacy + Waivers of Standards +

Section 4 Test Specification and Evaluation Test Specification + Requirements + System Functions + Test/Function Relationships + Equivalence Partitioning + Boundary-Value Analysis + Cause-Effect Graphing + Error Guessing + Test Methods and Constraints + System Test Conditions + Extent of System Test + Data Recording + System Test Constraints + Test Progression + Test Evaluation + Test Data Criteria + Test Data Reduction + System Test Interfaces + Test Methods and Documentation +

Section 5 Test Description Test Description + Test Control + System Test Means of Control + Test Data + Input Data + Input Commands + Output Data + Output Notification + Test Procedures + Test Setup + Test Data Development + Acceptance Testing Test Data + Test Initialization + Test Steps + Test Termination + Restart Procedures +

Test Results = Quantifiable information describing the testing of program code as directed by the Test Plan and presented as a technical report including input data, subsequent output data, number and classes of failures and other such statistics explicitly and specifically related to the functions of the program code.

Tested Code = The program code that has successfully passed all of the applicable processes specified in the Test Plan.

Testing Line = A numeric or graphical representation of actual costs in each of several years throughout a specified period.

Top Down Factoring = Functional decomposition from most general terms to most detailed terms.

Tracking Methods = A means of monitoring activities associated with a specific task.

Trained Personnel = Persons who know how to operate/use the system.

Training Support Plan (TRP) = The plan that provides a description of the training requirements for the implementation and operation of the product being developed.

Training Support Plan (TRP) Standard (IRM-5231-15) = The guideline that establishes the format for a TRP, its documentation requirements, and the completion criteria necessary to ensure that all TRPs developed from this standard are uniform and complete.

Section 1 Training Support Plan Overview Plan Objective + Plan Scope + Resource Requirements + Planned Duration + MCO 1510.37 Inconsistencies + Terms and Abbreviations +

Section 2 Training Support Plan Information Training Coordination + Training Responsibility/Task Matrix + Responsibility/Task Matrix Definition + Responsibility Codes + Training Schedule + Training Materials + Training Facilities + Training Staffing + Training Cost + Personnel to be Trained +

Section 3 Course Description Information Course Name + Course Objective + Course Scope + Course Teaching Techniques + Course Content and Description + Course Prerequisites + Course Completion Criteria

Transaction Analysis = A method of re-structuring same-level processors on a DFD into a structure chart based upon physical grouping of data.

Transform Analysis = A method of re-structuring same-level processors on a DFD into a structure chart based upon sequentually revising the same initial data element.

Usability Testing = The determination of whether the system provides for the human factors and ease of use specified in the objectives.

User = Any person or organization who requests initiation of the system development process to provide a data processing solution to a particular set of requirements.

Users Manuals (UM) = Those manuals that provide a user with the information required to perform manual tasks, operate, and use data processing hardware and software.

Users Manual Standard (IRM-5231-07) = The guideline that defines the format and content of user manuals that present operational features and their usage to the functional and technical users.

Section 1 Introduction Purpose and Scope + Relationship to Other Activities + Terms and Abbreviations + References +

Section 2 Application Process Descriptions + Pictorials + Inputs + Outputs + Data Relationships +

Section 3 Procedures Tools + Hints and Warnings + System Organization Chart + Process Flowchart

User/Procedure Matrix = The decomposition of topics from the User/Topic Matrix, into sub-topics, breakdown of sub-topics into procedures, and the relationship of procedures to users.

User Requirements = The specific requirements that a project request is initiated to address.

User/Topic Matrix = Indicates the topics that need to be developed as part of the analysis conducted in developing a subject architecture.

Validate = To check the adequacy of a system design in terms of its original functional objectives. (Does it do what it is supposed to do?)

Validated System = Approved system not formally in production. A system that is awaiting real-time testing.

Vendors = The commercial organizations that supply hardware and software to the project.

Verify = To measure level of conformance of a system design to technical standards and specifications. (Does it do what it does right?)

Volume Testing = The determination of whether the system can handle the volume of data specified in the objectives.

Waiver = A written authorization to accept a configuration item or other designated item(s) which, either during production or after having been submitted for inspection, are found to depart from specified requirements, but nevertheless are considered suitable for use "as-is" or after rework by an approved method.

Walkthrough - Organized procedure for a group of peers to review a technical product for accuracy.

Work Breakdown Structure (WBS) = Level of detail in the top-down division of work planned to perform the task. It is this breakdown that defines the individual work packages.

COMMENTS/REVISIONS

Technical publications under the Information Resources Management (IRM) Standards and Guidelines Program (MCO 5271.1) are reviewed annually. Your comments and/or recommendations are strongly encouraged.

IRM Tech Pub Name: ______________________________

IRM-_____-___ (Number) Date of Tech Pub: ______________

COMMENTS/RECOMMENDATIONS: ______________________________

____

____

____

____

____

____

____

____

____

____

____

____

____

____

Name/Rank: ______________________________(optional)

Unit: __________________________________(optional)

Mail To: Commandant of the Marine Corps (Code CCI)
Headquarters, United States Marine Corps
Washington, D.C. 20380-0001