



TECHNICAL REPORT

CMB6-3

Issued 1991-01
Stabilized 2014-10

Configuration Identification

NOTICE

This document has been taken directly from the original TechAmerica document and contains only minor editorial and format changes required to bring it into conformance with the publishing requirements of SAE Technical Standards. The release of this document is intended to replace the original with the SAE International document. Any numbers established by the original document remain unchanged.

The original document was adopted as an SAE publication under the provisions of the SAE Technical Standards Board (TSB) Rules and Regulations (TSB 001) pertaining to accelerated adoption of specifications and standards. TSB rules provide for (a) the publication of portions of unrevised specifications and standards without consensus voting at the SAE committee level, and (b) the use of the existing specification or standard format.

SAE Technical Standards Board Rules provide that: "This report is published by SAE to advance the state of technical and engineering sciences. The use of this report is entirely voluntary, and its applicability and suitability for any particular use, including any patent infringement arising therefrom, is the sole responsibility of the user."

SAE reviews each technical report at least every five years at which time it may be revised, reaffirmed, stabilized, or cancelled. SAE invites your written comments and suggestions.

Copyright © 2014 SAE International

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of SAE.

TO PLACE A DOCUMENT ORDER: Tel: 877-606-7323 (inside USA and Canada)
Tel: +1 724-776-4970 (outside USA)
Fax: 724-776-0790
Email: CustomerService@sae.org
<http://www.sae.org>

SAE WEB ADDRESS:

Full document details: SAE CMB6_3

SAE values your input. To provide feedback on this Technical Report, please visit http://www.sae.org/technical/standards/CMB6_3

Documents

Document Sharing Hub

CMB6-3

EIA ENGINEERING BULLETIN

Configuration Identification

CMB6-3

JANUARY 1991

ELECTRONIC INDUSTRIES ASSOCIATION
ENGINEERING DEPARTMENT



NOTICE

EIA Engineering Standards and Publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for his particular need. Existence of such Standards and Publications shall not in any respect preclude any member or nonmember of EIA from manufacturing or selling products not conforming to such Standards and Publications, nor shall the existence of such Standards and Publications preclude their voluntary use by those other than EIA members, whether the standard is to be used either domestically or internationally.

Recommended Standards and Publications are adopted by EIA without regard to whether their adoption may involve patents on articles, materials, or processes. By such action, EIA does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the Recommended Standard or Publication.

Technical Bulletins are distinguished from EIA Recommended Standards or Interim Standards, in that they contain a compilation of engineering data or information useful to the technical community, and represent approaches to good engineering practices that are suggested by the formulating committee.

This Bulletin is not intended to preclude or discourage other approaches that similarly represent good engineering practice, or that may be acceptable to, or have been accepted by, appropriate bodies. Parties who wish to bring other approaches to the attention of the formulating committee to be considered for inclusion in future revisions of this Bulletin are encouraged to do so. It is the intention of the formulating committee to revise and update this Bulletin from time to time as may be occasioned by changes in technology, industry practice, or government regulations, or for other appropriate reasons.

Published by

ELECTRONIC INDUSTRIES ASSOCIATION

Engineering Department
2001 Pennsylvania Ave., N.W.
Washington, D.C. 20006

**PRICE: Please refer to the current
Catalog of EIA & JEDEC STANDARDS & ENGINEERING PUBLICATIONS
or call the EIA STANDARDS SALES DEPARTMENT (202/457-4966)**

Copyright 1991

All Rights Reserved
Printed in U.S.A.

Configuration Identification
EIA Bulletin CMB6-3
Acknowledgement

This EIA Bulletin No. CMB6-3, "Textbook for Configuration Identification," is presented by the Electronic Industries Association G-33 Data and Configuration Management Committee. It has been prepared to provide a textbook for use at the undergraduate level to teach the configuration identification CM 201 course developed by the Education and Training Task Group. The co-authors of the text are Elliott A. Whitton and Jo Ann S. Kramer of Martin Marietta Missile Systems, Orlando, Florida.

The G-33 Committee, with the support of its subcommittees, will continue to devote its efforts to the advancement of knowledge and understanding of the subjects of Configuration and Data Management. Therefore, we invite your comments on this Bulletin and welcome your suggestions concerning publication of additional information that would aid in the effective conduct of any activity relative to training and education within the Configuration and Data Management disciplines.

The appearance of a reference in this Bulletin does not necessarily imply endorsement by the EIA.

Configuration Identification EIA Bulletin CMB6-3

TABLE OF CONTENTS

	Page
1.0 INTRODUCTION	1
1.0.1 Overview of Configuration Management	1
1.1 DEFINITIONS	1
1.1.1 Configuration	1
1.1.2 Configuration Item	3
1.2 TECHNIQUES OF CONFIGURATION MANAGEMENT	3
1.2.1 Configuration Identification	3
1.2.2 Configuration Control	4
1.2.3 Configuration Status Accounting	5
1.2.4 Configuration Audits	5
1.3 OBJECTIVE OF TEXT	5
2.0 HISTORY AND BACKGROUND	6
2.1 INTRODUCTION	6
2.2 EVOLUTION OF CONFIGURATION MANAGEMENT	6
2.2.1 Missile Race	6
2.2.2 Inaccurate and Incomplete Documentation	7
2.2.3 Growth of Complexity	7
2.2.4 Army Navy Aeronautics Bulletin	7
2.2.5 Air Force Manuals	7
2.2.6 Department of Defense Task Groups	8
2.2.7 Relationship of Documents	8
2.2.8 Software Procedures	10
2.3 SUMMARY	10
3.0 ACQUISITION PROCESS	12
3.1 LIFE CYCLE PHASES	12
3.1.1 Concept Exploration/Definition Phase	12
3.1.2 Concept Demonstration/Validation Phase	12
3.1.3 Full-Scale Development Phase	13
3.1.4 Production and Initial Deployment Phase	13
3.1.5 Operations Support Phase	13
3.2 REVIEWS AND AUDITS	13
3.2.1 System Requirements Review	13
3.2.2 System Design Review	13
3.2.3 Software Specification Review	14
3.2.4 Preliminary Design Review	14
3.2.5 Critical Design Review	14
3.2.6 Functional Configuration Audit	14
3.2.7 Physical Configuration Audit	14
3.2.8 Test Readiness Review	15

TABLE OF CONTENTS, continued

	Page
3.3 FORMAL BASELINES	15
3.3.1 Functional Baseline	15
3.3.2 Allocated Baseline	15
3.3.3 Product Baseline	17
3.4 SOFTWARE BASELINES	17
3.4.1 Functional and Allocated Baselines	17
3.4.2 Developmental Baseline	17
3.4.3 Formal Product Baseline	19
3.5 INTERNAL BASELINE	19
3.5.1 Application	19
3.6 SUMMARY	19
4.0 ALLOCATION OF CONFIGURATION IDENTIFICATION REQUIREMENTS	20
4.0.1 Introduction	20
4.1 TOP-DOWN BREAKDOWN	20
4.1.1 Hardware Top-Down Breakdown	20
4.1.2 Software Top-Down Breakdown	21
4.2 CONFIGURATION ITEMS	21
4.2.1 Selection of Configuration Items	21
4.2.1.1 Level of Control	21
4.2.1.2 Life Cycle Considerations	21
4.2.1.3 Schedule Considerations	21
4.2.1.4 Common Subsystem Considerations	21
4.2.2 Effects of Configuration Item Selection	22
4.3 SPECIFICATION TREE	23
4.4 DRAWING TREE GENERATION BREAKDOWN	23
4.5 WORK BREAKDOWN STRUCTURE	23
4.5.1 Summary Work Breakdown Structure	26
4.5.2 Project Summary Work Breakdown Structure	26
4.5.3 Contract Work Breakdown Structure	26
4.5.3.1 Contractor Alternatives	26
4.5.4 Project Work Breakdown Structure	28
4.6 SUMMARY	28
5.0 SPECIFICATIONS AND REQUIREMENTS	29
5.0.1 Definition and Usage	29
5.0.2 Relationship of Specifications to Configuration Identification	29
5.0.2.1 Establishing Configuration Identification	29
5.0.2.2 Relationship to Life Cycle Phases and Baselines	29

TABLE OF CONTENTS, continued

	Page
5.1 TYPES AND FORMS OF SPECIFICATIONS	31
5.1.1 Military Specifications and Standards	31
5.1.1.1 Military Specification	31
5.1.1.2 Federal Specification	31
5.1.1.3 Detail Specification	31
5.1.1.4 General Specification	32
5.1.2 Program Peculiar Specifications	32
5.1.2.1 Type A - System/Segment Specification	33
5.1.2.2 Type B - Development Specification	33
5.1.2.2.1 Type B1 - Prime Item Development Specification	33
5.1.2.2.2 Type B2 - Critical Item Development Specification	34
5.1.2.2.3 Type B3 - Non-complex Item Development Specification	34
5.1.2.2.4 Type B4 - Facility or Ship Development Specification	35
5.1.2.2.5 Type B5 - Software Development Specification	35
5.1.2.2.5.1 Software Requirements Specification	35
5.1.2.2.5.2 Interface Requirements Specification	35
5.1.2.3 Type C - Product Specification	36
5.1.2.3.1 Type C1 - Prime Item Product Specification	37
5.1.2.3.1.1 Type C1a - Prime Item Product Function Specification	37
5.1.2.3.1.2 Type C1b - Prime Item Product Fabrication Specification	37
5.1.2.3.2 Type C2 - Critical Item Product Specification	37
5.1.2.3.2.1 Type C2a - Critical Item Product Function Specification	37
5.1.2.3.2.2 Type C2b - Critical Item Product Fabrication Specification	37
5.1.2.3.3 Type C3 - Non-complex Item Product Fabrication Specification	37
5.1.2.3.4 Type C4 - Inventory Item Specification	38
5.1.2.3.5 Type C5 - Software Product Specification	38
5.1.2.4 Type D - Process Specification	38
5.1.2.5 Type E - Material Specification	38
5.1.3 Two-part Specifications	39
5.1.4 Forms of Specifications	39

TABLE OF CONTENTS, continued

	Page
5.2 CHANGES TO SPECIFICATIONS	39
5.2.1 Changes to Military Specifications	39
5.2.1.1 Supersession	39
5.2.1.2 Specification Revision	41
5.2.1.2.1 Format	41
5.2.1.2.2 Notation of Revision	41
5.2.1.2.3 Summary Sheet for Proposed Coordinated Specification	41
5.2.1.2.4 Revision Indicator	42
5.2.1.3 Specification Supplement	42
5.2.1.3.1 Contents	42
5.2.1.3.2 Format	42
5.2.1.3.3 Preamble	42
5.2.1.3.4 Captions for Supplements	43
5.2.1.3.5 Page Numbering and Document Identification of Supplements	43
5.2.1.4 Specification Amendment	43
5.2.1.4.1 Document Identifier	43
5.2.1.4.2 Headings and Titles	43
5.2.1.5 Notices	43
5.2.1.5.1 Document Identifier	43
5.2.2 Changes to Program Peculiar Specifications	44
5.2.2.1 General	44
5.2.2.2 Changes	44
5.2.2.2.1 Specification Change Notice	44
5.2.2.2.2 Notice of Revision	46
5.2.2.2.3 Changed Pages	46
5.2.2.2.4 Change Numbering	46
5.2.2.2.5 Identification of Changed Pages	46
5.2.2.2.6 Numbering of Changed Pages	48
5.2.2.3 Revisions	48
5.3 IDENTIFICATION OF SPECIFICATIONS	48
5.3.1 Identification of Military Specifications	48
5.3.1.1 Military Specifications	48
5.3.1.2 Document Identifiers for Detail Specification Referring to a General Specification	49
5.3.1.3 Date of Specification	49
5.3.2 Identification of Program Peculiar Specifications.	49
5.3.2.1 Sectional Arrangement of Specifications	49
5.3.2.1.1 Specification Sections Description	49
5.3.2.2 Identification of Two-part Specifications	50
5.4 SUMMARY	51

TABLE OF CONTENTS, continued

	Page
6.0 ENGINEERING DRAWINGS	52
6.0.1 General Drawings	52
6.0.2 Relationship of Drawings to Configuration Identification	52
6.0.3 Classification of Drawing Levels	52
6.0.3.1 Classification	52
6.0.3.1.1 Levels	53
6.0.3.1.2 Requirements	53
6.0.4 Description of Drawing Levels	53
6.0.4.1 Level 1, Conceptual and Development Design	53
6.0.4.1.1 Level 1 Requirements	53
6.0.4.2 Level 2, Production Prototype and Limited Production	53
6.0.4.2.1 Level 2 Requirements	53
6.0.4.3 Level 3, Production	55
6.0.4.3.1 Level 3 Requirements	55
6.0.5 Differences Between Level 2 and Level 3 Drawings	55
6.0.6 Variables Affecting Drawing Content	58
6.0.6.1 Determination of Drawing Type	58
6.0.6.1.1 Type and Number of Drawings	58
6.0.6.1.2 Responsibility for Drawing Selection	58
6.0.7 MIL-T-31000 Drawing Categories	59
6.0.7.1 Conceptual Design Drawings and Associated Lists	59
6.0.7.2 Developmental Design Drawings and Associated Lists	59
6.0.7.3 Product Drawings and Associated Lists	59
6.0.7.4 Commercial Drawings and Associated Lists	59
6.0.7.5 Special Inspection Equipment Drawings and Associated Lists	59
6.0.7.6 Special Tooling Drawings and Associated Lists	59
6.1 TYPES OF DRAWINGS	60
6.1.1 Detail Drawing	61
6.1.2 Monodetail Drawing	61
6.1.3 Multi-Detail Drawing	61
6.1.4 Assembly Drawing	61
6.1.5 Detail Assembly Drawing	61
6.1.6 Control Drawing	64
6.1.7 Envelope Drawing	64
6.1.8 Specification Control Drawing	64
6.1.9 Source Control Drawing	64

TABLE OF CONTENTS, continued

	Page
6.1.10 Altered Item Drawing	69
6.1.11 Selected Item Drawing	69
6.1.12 Interface Control Drawing	69
6.1.13 Installation Control Drawing	73
6.1.14 Installation Drawing	73
6.1.15 Diagrammatic Drawing	73
6.1.16 Schematic Diagram	73
6.1.17 Connection or Wiring Diagram	73
6.1.18 Construction Drawing	73
6.1.19 Book-Form Drawing	74
6.1.20 Wiring List	74
6.1.21 Wiring Harness Drawing	74
6.1.22 Cable Assembly Drawing	74
6.1.23 Printed Wiring Master (Stable Base Artwork) Pattern Drawing	74
6.1.24 Kit Drawing	74
6.1.25 Combination of Adopted Items Drawing	74
6.1.26 Modification Drawing	74
6.1.27 Computer-Aided Design Drawing	78
6.2 REVISION OF ENGINEERING DRAWINGS	78
6.2.1 Change in Dimensions	78
6.2.2 Crossing Out	78
6.2.3 Drawing Practices	78
6.2.4 Identifying Revision Locations	79
6.2.5 Revision Letters	79
6.2.6 Multiple Changes	79
6.2.7 Revision of Multi-Sheet and Book-Form Drawings	79
6.2.8 Identification of Configuration Management Baselines	82
6.3 NUMBERING, CODING AND IDENTIFICATION OF DRAWINGS	82
6.3.1 Numbering Systems	82
6.3.1.1 Nonsignificant Numbering Systems	82
6.3.1.2 Significant Numbering Systems	84
6.3.2 Contractor and Government Entity Code	84
6.3.3 Document Identification Number	84
6.3.3.1 Referenced Documents	86
6.3.4 Drawing Number	86
6.3.4.1 Drawing Number Requirements	86
6.3.5 Part Number	87
6.3.5.1 Part Number Requirements	87
6.4 SUMMARY	88

TABLE OF CONTENTS, continued

	Page
7.0 TECHNICAL DATA PACKAGE ELEMENTS AND ASSOCIATED DOCUMENTS	89
7.1 IDENTIFICATION OF TECHNICAL DATA PACKAGE ELEMENTS . .	89
7.2 SPECIAL INSPECTION EQUIPMENT	89
7.2.1 Special Inspection Equipment Documentation	90
7.2.1.1 Special Inspection Equipment Operating Instructions	90
7.2.1.2 Special Inspection Equipment Descriptive Documentation	90
7.2.1.3 Special Inspection Equipment Calibration Procedures	90
7.2.2 Relationship of Special Inspection Equipment Drawings to Configuration Identification	90
7.3 PRESERVATION, PACKAGING, PACKING AND MARKING DATA . .	90
7.3.1 Relationship of Packaging Requirements to Configuration Identification	90
7.3.2 Submittal of Packaging Data	91
7.4 QUALITY ENGINEERING PLANNING LIST	91
7.5 SOFTWARE AND SOFTWARE DOCUMENTATION	91
7.6 TEST REQUIREMENTS DOCUMENTS	91
7.7 QUALITY ASSURANCE PROVISIONS	92
7.7.1 Quality Assurance Procedure Description	92
7.7.2 Relationship of Quality Assurance Provisions to Configuration Identification	92
7.7.3 Documents Requiring Quality Assurance Provisions	92
7.7.4 Documents Not Requiring Quality Assurance Provisions	92
7.7.5 Classification of Characteristics	93
7.8 OPERATIONAL PHASE TECHNICAL DATA PACKAGE ELEMENTS . .	93
7.8.1 Field Configuration Management	93
7.8.2 Identification of Operational Phase Technical Data Package Elements	93
7.9 SUMMARY	93
8.0 ENGINEERING RELEASE SYSTEM	95
8.0.1 Introduction	95
8.0.2 Types of Released Documentation	95
8.1 RELEASE CLASSIFICATIONS	96
8.1.1 Formal Release	96
8.1.2 Informal Release	96
8.2 PRODUCTION RELEASE REVIEW	96
8.3 SUPPORTING THE RELEASE	97

TABLE OF CONTENTS, continued

	Page
8.4 TYPICAL RELEASE TECHNIQUES	97
8.4.1 Bulk Release	97
8.4.1.1 Action Required	98
8.4.1.2 Resulting Action	98
8.4.1.3 De-allocation	98
8.4.2 Advance Release	98
8.4.3 Partial Release	98
8.5 ENGINEERING RELEASE RECORDS	98
8.5.1 Identification Criteria for Configuration Items	98
8.5.2 Elements of Data Required	100
8.6 SOFTWARE CONSIDERATIONS	101
8.6.1 General Requirements	101
8.6.1.1 Kind of Data	101
8.6.1.2 When to Release	101
8.6.2 Types of Software Release	102
8.6.2.1 Incremental Release	102
8.6.2.2 Cumulative Release	102
8.6.2.3 Total Release	102
8.6.3 Release Procedure	102
8.6.3.1 The Release Package	102
8.6.3.2 Storage of Master Copies	102
8.6.4 Artwork and Word Processor Magnetic Files	102
8.7 SUMMARY	103
9.0 CONFIGURATION MANAGEMENT PLAN	105
9.1 INTRODUCTION	105
9.2 MIL-STD-1456 CONFIGURATION IDENTIFICATION REQUIREMENTS	105
9.2.1 Baseline Identification	105
9.2.2 Functional Baseline	105
9.2.3 Allocated Baseline	105
9.2.4 Product Baseline	106
9.3 MIL-STD-483 CONFIGURATION IDENTIFICATION REQUIREMENTS	106
9.3.1 Specifications	106
9.3.2 Drawings	106
9.4 NORTH ATLANTIC TREATY ORGANIZATION CONFIGURATION IDENTIFICATION REQUIREMENTS	107
9.5 SOFTWARE CONFIGURATION IDENTIFICATION REQUIREMENTS	107
9.6 SUMMARY	107
10.0 SUMMARY AND CONCLUSIONS	108
LIST OF ACRONYMS	109

LIST OF FIGURES

Figure	Title	Page
1.1-1	Configuration Characteristics	2
2.2.7-1	Time Line for Configuration Management	9
2.2.8-1	Software Documentation Standards Chronology	11
3.3-1	Configuration Management Life Cycle	16
3.4-1	Typical Software Development Baseline Establishment and Control	18
4.3-1	Specification Tree	24
4.4-1	Drawing Tree	25
4.5.3-1	Contract Work Breakdown Structure (Partial)	27
5.0.2.2-1	Specification-Baseline Relationship	30
5.1.4-1	Forms of Specifications	40
5.2.2.2.1-1	Specification Change Notice	45
5.2.2.2.2-1	Notice of Revision	47
6.0.3.1.1-1	Documentation Life Cycle Phases	54
6.0.4.3.1-1	Drawing Level Requirements	56
6.1.2-1	Monodetail Drawing	62
6.1.4-1	Assembly Drawing	63
6.1.6-1	Commercial Item Drawing Guide	65
6.1.8-1	Specification Control Drawing	66
6.1.9-1	Source Control Drawing (with replaceable parts)	67
6.1.9-2	Source Control Drawing (Replacement Part)	68
6.1.10-1	Altered Item Drawing (part controlled by Military Standard)	70
6.1.11-1	Selected Item Drawing (part controlled by Military Standard)	71
6.1.12-1	Interface Control Drawing	72
6.1.21-1	Wiring Harness Drawing	75
6.1.24-1	Kit Drawing	76
6.1.25-1	Combination of Adopted Items Drawing	77
6.2.4-1	Example of Revision Block Notation for Multi-Sheet Drawing	80
6.2.4-2	Example of Revision Block Notation for Changes in Drawing Rights	80
6.2.7-1	Revision Status Block and Revision Status of Drawing Sheets	81
6.2.8-1	Example of Notation for Allocation Baseline	83
6.2.8-2	Example of Notation for Product Baseline	83
6.3.2-1	Example of CAGE Code, Drawing Number, Design Activity Relationship as Originally Specified	85
7.7.5-1	Example of QAP Notes on Drawings	94
8.5-1	Engineering Release Record	99
8.6.4-1	Magnetic File Label	104

Configuration Identification
EIA Bulletin CMB6-3
Chapter 1
Introduction

1.0 INTRODUCTION

1.0.1 Overview of Configuration Management. To manage the development and procurement of modern-day hardware and software systems, the project management concept and many engineering, logistic, and management disciplines have developed over the years.

Configuration Management (CM) is the special name that has been coined to describe those specialized engineering management practices used by industry and the Government to define and control items (sometimes referred to as configuration items (CIs)) in their inventories. CM supports the establishment of a product service network for all hardware, software, and supporting data in the form of drawings, design documents, operating manuals, repair instructions, etc., being developed, and ensures the continued serviceability of these items as changes are made throughout their service lives.

The overall objective of CM is to guarantee the procuring activity that a given product is what it was intended to be - functionally and physically, as defined by contractual drawings and specifications, and to identify the configuration to the lowest level required to assure repeatable performance, quality, and reliability in future products of the same type.

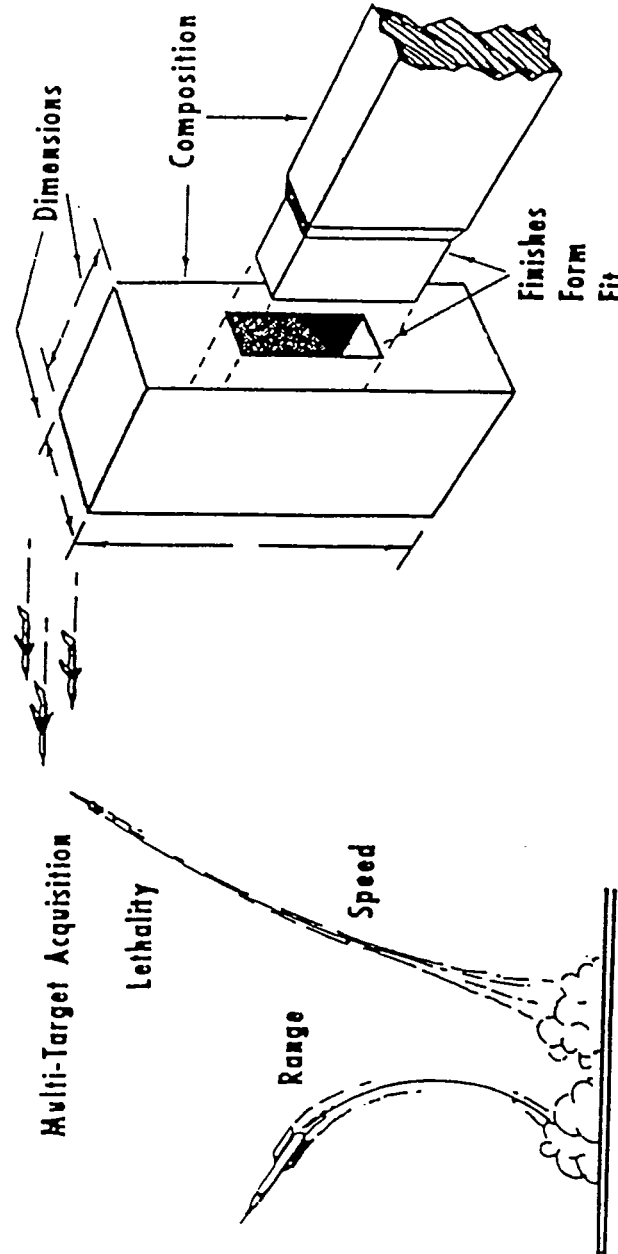
1.1 DEFINITIONS

The following definitions are basic to the understanding of CM:

1.1.1 Configuration. The configuration of an item is defined in both functional and physical terms. Functional requirements, such as accuracy, stability, linearity, etc., establish the performance the item is expected to achieve; physical requirements, such as weight, size, material, etc., define the specific design and appearance of the item to be built. The term configuration also applies to technical descriptions required to build, test, operate, and repair a CI (see Figure 1.1-1). In practice, an item being developed is described, and its development governed, primarily by its intended functional characteristics. Following development, however, an item being produced for customer use is primarily described and governed by its physical characteristics, although the functional characteristics continue to be controlled.

WHAT IS CONFIGURATION ?

The **FUNCTIONAL** and/or **PHYSICAL** CHARACTERISTICS of **HARDWARE/SOFTWARE** as SET FORTH in **TECHNICAL DOCUMENTATION** and ACHIEVED in a **PRODUCT**



FUNCTIONAL

RANGE - SPEED - RELIABILITY - MAINTAINABILITY
LETHALITY - MULTI-TARGET ACQUISITION - ETC.

PHYSICAL

COMPOSITION - DIMENSIONS - FINISHES - FORM
FIT - INTERCHANGABILITY - ETC.

Figure 1.1-1 Configuration Characteristics

1.1.2 Configuration Item. A CI is an aggregation of hardware/software, or any of its discrete portions, which satisfies an end-use function and is designated by the Government or other procuring activity for CM. CIs may vary widely in complexity, size and type as reflected below:

- ♦ Systems, such as aircraft, missiles, ordnance, ships, submarines, space and surface vehicles, electronic installations (including their computer programs) and facilities.
- ♦ Equipment, such as vehicles, artillery ammunition and radar sets.
- ♦ End items selected for reasons of system integration or interface control.

During development and manufacture of the initial (prototype) production configuration, CIs are those specification items whose functions and performance parameters must be defined (specified) and controlled to achieve the overall end-use function and performance.

1.2 TECHNIQUES OF CONFIGURATION MANAGEMENT

CM practices, basically, are common sense management concepts which have been formalized into set practices to address the peculiarities of managing large quantities of items in multiple product lines. For convenience sake in describing these practices, CM is divided into four functional areas: Configuration Identification, Control, Status Accounting, and Audits. Each of the four functions has a specific purpose and a specific area of emphasis.

1.2.1 Configuration Identification. Configuration Identification establishes the requirements for a CI and the hardware/software needed to meet these requirements. It consists of the current approved or conditionally approved technical documentation for a CI as set forth in specifications, drawings and associated lists, and documents referenced therein. Approved technical documentation for a CI describes: (1) all necessary functional characteristics; (2) the tests required to demonstrate achievement of specified functional characteristics; (3) the necessary interface characteristics with associated CIs; (4) key functional characteristics of CIs and key lower level CIs, if any; and (5) design constraints, such as envelope dimensions or component standardization, use of inventory items, and integrated logistic support policies.

The technique requires the development of formal technical descriptions called baselines, which are defined and documented at designated points in the life cycle of a CI (baselines and life cycle phases are defined in Chapter 3).

Configuration Identification begins with the development of the first of a series of documents which broadly describe a CI's overall functional characteristics. This identification becomes more specific as the design progresses. As development continues, documentation is prepared for each CI, complete with a definition of the qualification requirements the item must meet to be acceptable.

Before volume production is authorized, another set of documents completes the identification of the hardware/software to the precise detail required for manufacture or procurement. This is the Technical Data Package (TDP). The TDP is used and modified throughout the service life of the hardware or software.

1.2.2 Configuration Control. Configuration Control is the discipline employed to levy control on those documents which provide the identification. It embodies systematic evaluation, coordination, approval or disapproval, and implementation of all approved changes in the configuration of a CI after the formal establishment of its configuration identification. Configuration Control has three basic elements:

- (1) Design change reviews.
- (2) The actual processing, evaluation, and approval or disapproval of Engineering Change Proposals (ECPs) and other change documents.
- (3) Change implementation by affected functional activities (e.g., engineering, contracts, provisioning, supply management, maintenance, training, support, and manufacturing).

Once the basic identification documents have been released for internal use, all changes to these documents must be controlled. The technique provides for flexibility in determining the degree of control required, according to the stage of development and nature of the item being identified. Only that documentation approved by the procuring activity is placed under formal (customer) control. The details released vary from broad performance parameters as part of the preliminary design process, to dimensions and tolerances for critical components as part of the detail design process.

1.2.3 Configuration Status Accounting. Configuration Status Accounting allows for the continual monitoring of identification status versus item configuration. It provides traceability of the configuration baselines and is maintained by recording and reporting of the information that is used to define and change the configuration, including a listing of the approved configuration identification, the status of proposed changes to the configuration, and the implementation status of approved changes.

Configuration Status Accounting also includes the tracking of changes to delivered hardware/software, particularly the incorporation of alteration/modification/retrofit parts into delivered units.

1.2.4 Configuration Audits. Configuration Audits (See Chapter 3) serve to verify that compliance with baseline requirements has been achieved. The audit function verifies achievement of development requirements and the accuracy of a CI's detail design documentation. Configuration audits include the Functional Configuration Audit (FCA), the Formal Qualification Review (FQR), and the Physical Configuration Audit (PCA). When practicable and appropriate, the configuration audits for similar CIs may be accomplished together (i.e., the FCAs for three CIs may be conducted as part of a single meeting).

1.3 OBJECTIVE OF TEXT

The objective of this text is to provide an in-depth understanding of the concepts and principles involved in the Configuration Identification process. It is geared to the individual who has completed a basic introductory course in CM and who requires the more in-depth knowledge of a CM practitioner as opposed to a superficial overview.

The general concepts within the text apply during the design and delivery of products to various types of customers, and include elements of information which must be monitored for the continued support and service of these products. Because of the fact that the Department of Defense (DOD) represents a major customer of such goods and services, a highly structured program utilizing Configuration Identification, along with the related CM disciplines, exists in the environment of military item acquisition. The authors, therefore, have placed emphasis on DOD-related systems and requirements.

Configuration Identification
EIA Bulletin CMB6-3
Chapter 2
History and Background

2.0 HISTORY AND BACKGROUND

2.1 INTRODUCTION

CM is a discipline utilized to simplify the management of complex hardware and software systems. It reduces the elements involved to their simplest terms, equips each problem with a convenient management approach and displays the whole situation in a manner that enhances comprehension, analysis, and control. CM defines the product, its components, and their interfaces; it restricts unnecessary change, and it requires precise records of the changes that are authorized.

2.2 EVOLUTION OF CONFIGURATION MANAGEMENT

2.2.1 Missile Race. DOD procedures and direction on CM are the result of gradual evolution through the years. A significant number of the techniques and some of the formalization of CM have been derived from good business practices. Initially, CM disciplines lacked substance and uniformity, most notably relative to configuration control and accounting. This shallow depth of CM, especially the lack of proper records, became apparent to the DOD in the race for a successful missile launch in the 1950s. With time being critical, the promulgation of changes was accelerated to resolve incompatibilities among elements supplied by many prime contractors. When the Government directed that contractors build a certain number of missiles identical to original prototypes, both the Government and industry found themselves in the following circumstances:

- (1) Prototypes were expended.
- (2) Adequate records of part numbers identification, and sequence of changes were not available.
- (3) Technical documentation did not reflect all the changes that were made.

It became apparent that each successive item would not be a replica of the previous item and, therefore, the success of this replica in performing its mission would not be guaranteed.

2.2.2 Inaccurate and Incomplete Documentation. Major General William B. Bunker of the US Army Materiel Command (AMC) described the situation in these terms:

"Expensive and large systems were being delivered with no information as to how they were built and how to keep them running. Deficiencies and failures in tests started long sleuthing operations to determine what had failed and what design alternatives were available to solve the problem. Illustrated parts lists appeared months, sometimes even years after the equipment was delivered, and were usually inaccurate and incomplete. We qualified certain components and tests and then found, usually after they failed in service, that the production design had been altered, and we found that only the original manufacturer could supply the system or its component because we couldn't tell anyone else how it was made."

2.2.3 Growth of Complexity. Aircraft weapon systems were becoming more complex, with military aircraft carrying sophisticated electronic equipment which required on-line maintenance. Air bases and aircraft carriers now devoted great proportions of their space to electronic and avionic support facilities. Multiple configurations were often undiscovered until maintenance, diagnostic analysis, interchangeable parts, and supporting documentation presented compatibility problems.

2.2.4 Army Navy Aeronautics Bulletin. The formal set of procedures establishing CM have generally grown out of military specifications and contract requirements. The acquisition of military hardware during World War II taught some bitter lessons. Engineering changes levied on increasingly sophisticated systems too often led to products that turned out to be unpleasant surprises. During the post war period, the Federal Government issued Army Navy Aeronautical (ANA) Bulletin 390 to establish uniform industry guidelines for proposing aircraft changes. Shortly thereafter ANA 391A was issued. This later bulletin categorized engineering changes by priority of classification and extended the guidelines to include electronic and ground support equipment manufacturers. In 1963 these two bulletins were superseded by ANA 445. The new bulletin reiterated the necessity for including technical, fiscal and logistic supporting data to define the impact of engineering changes; it also added maintainability and reliability data requirements.

2.2.5 Air Force Manuals. During the period of ANA Bulletin development, the Air Force was establishing CM techniques to be applied to industrial contractors. These techniques were published in June, 1962, in a series of manuals containing requirements for resource management of a military system during its acquisition phase. Volume I of this set of manuals, Air Force System Command Manual (AFSCM) 375, was on CM.

2.2.6 Department of Defense Task Groups. At a DOD Technical Documentation Conference in May, 1964, the Army and Navy expressed real interest in the Air Force CM program, since they too had developed similar, less formal procedures. The need for more uniform procedures was recognized by all three branches of the Armed Services. Subsequently, in June, 1964, the Director of Defense Research and Engineering and the Assistant Secretary of Defense for Installations and Logistics established an Ad Hoc Steering Group on CM. Four Task Groups with representatives from the Air Force, Army, and Navy were set up under the Steering Group to prepare a family of directives, standards, and specifications to cover policy, configuration control, configuration identification, and configuration status accounting. The Task Groups' objectives were to attain the optimum degree of uniformity in CM policy, procedures, data, forms, and reports at all interfaces within DOD and between DOD and industry.

2.2.7 Relationship of Documents. The Ad Hoc Group was responsible for publication of DOD Directive (DODD) 5010.19 and various military standards (MIL-STDs) as shown on the time-line drawing in Figure 2.2.7-1.

The directive applied to all echelons and all Government-industry interfaces. The document clearly stated that CM application should be carefully tailored to be consistent with the quantity, size, scope, stage of life cycle, nature and complexity of the product involved. The directive gave general policy guidance on the initiation, duration, and responsibility for CM. It also called for joint CM when more than one DOD component (military service or agency) was involved in the acquisition, modification, or support of a product.

Concurrent with the release of DODD 5010.19 in August, 1968, DOD Instruction (DODI) 5010.21, "Configuration Management Implementation Guidance," was approved. This document provided guidance for the implementation of DODD 5010.19. It expanded the policy and provided some firm guidelines for implementation.

The implementation of this DOD policy and guidance took the form of a limited number of mandatory, coordinated military standards and specifications for contractual application in lieu of the variety of existing exhibits and other documentation used for this purpose. Although DOD was the prime mover toward standardization, the National Aeronautics and Space Administration (NASA) contributed to this effort and was an advisory member to the DOD team responsible for establishing uniform procedures and documents.

DODI 5010.21 was cancelled in July, 1974, and superseded by a joint service regulation which established the mandatory requirements of DOD military standards and specifications and implemented CM within the respective services.

TIME LINE FOR CONFIGURATION MANAGEMENT

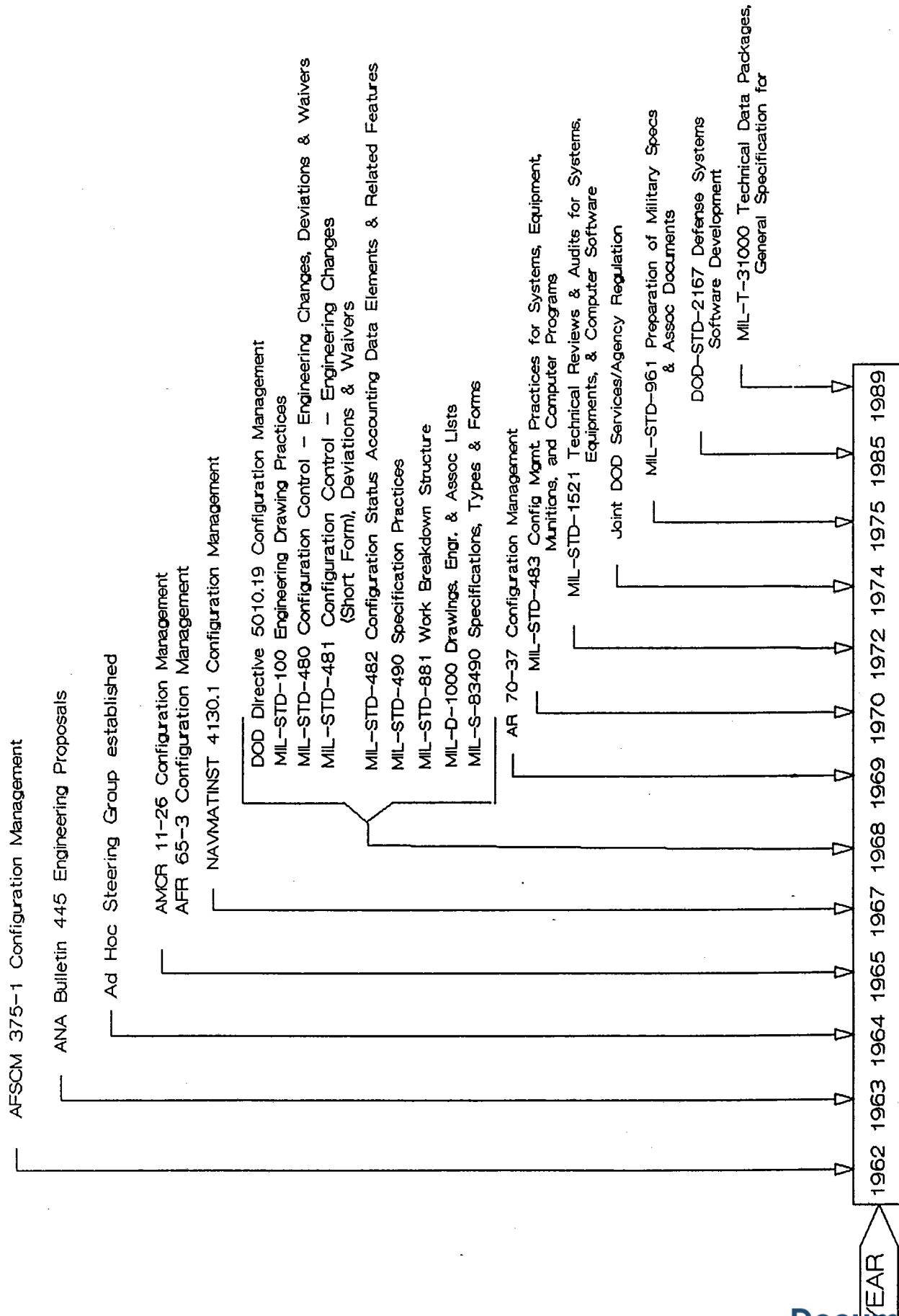


Figure 2.2.7--1

2.2.8 Software Procedures. In the mid-1970s, Government and industry began to become aware of the increasing impact of software on the systems acquisition environment. In 1976, DOD, recognizing the significant impact of software on weapons system procurements, mandated in DODD 5000.29 that:

"Defense system computer resources, including
both hardware and computer software, will be
specified and treated as configuration items."

Previously, software had been treated contractually as an item of data. Managing software as a CI elevated it to the level of a prime hardware item, and put it under the closer scrutiny of the procuring activity, theoretically assuring greater management attention.

In the ten years from 1976 to 1986, the software development process began to mature from what was once a "black art" to something approaching an engineering discipline. The stormy history of standards regulating the documentation of software and, therefore, providing a basis for its configuration control, parallels the early chaotic history of CM requirements documents. Figure 2.2.8-1 shows the early chronology of these standards.

A single tri-service Software Development Standard was achieved in June, 1985, with the issuance of DOD-STD-2167. This standard addresses software acquisition requirements, including CM.

2.3 SUMMARY

The history of CM as a formalized discipline is relatively brief, covering only the last 25 to 30 years. During this time industry, the Government and other procuring activities have made significant progress in attaining procedural commonality. It is hoped that continued verbal and written interfacing among CM personnel in these organizations will further enhance such efforts in the future.

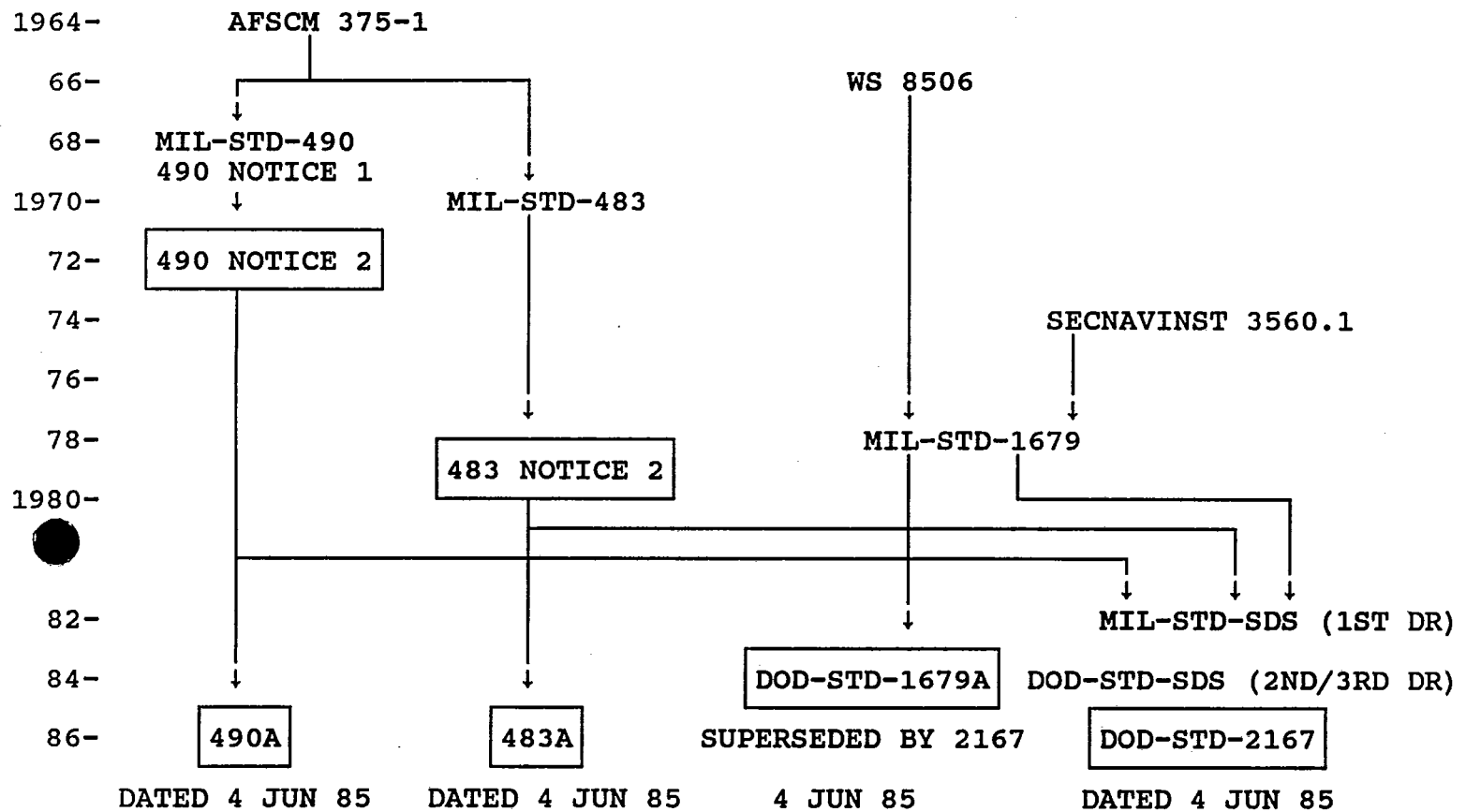


Figure 2.2.8-1 Software Documentation Standards Chronology

Configuration Identification
EIA Bulletin CMB6-3
Chapter 3
Acquisition Process

3.0 ACQUISITION PROCESS

The Government acquisition process involves program segments known as life cycle phases. The boundaries or separations between life cycle phases are called baselines. The baselines consist of documentation requirements which establish contractual relations between the procuring organization and the design activity. Baselines establish the basis for change control. The purpose of this chapter is to address the life cycle phases and the associated baselines.

3.1 LIFE CYCLE PHASES

DODD 5000.1 identifies the phases in the life cycle of a CI. These phases are:

- ◆ Concept Exploration/Definition Phase
- ◆ Concept Demonstration/Validation Phase
- ◆ Full-Scale Development (FSD) Phase
- ◆ Production and Initial Deployment Phase
- ◆ Operations Support Phase

3.1.1 Concept Exploration/Definition Phase. DODDs identify the initial phase of the CM Life Cycle as the Concept Exploration/Definition Phase. Ideas for new equipment normally emanate during this phase. Activities for the DOD components commonly include design studies, mission analysis and development planning, while the design activity may be involved in feasibility studies and the development of system alternatives. Primary interfaces are jointly identified by the procuring and design activities. This phase normally culminates with the preparation of an initial system specification and the establishment of the Functional Baseline.

3.1.2 Concept Demonstration/Validation Phase. The Concept Demonstration/Validation Phase is the period when the military utilizes a well-defined set of requirements to invite industry participation in this co-involvement, and from which a best solution approach should evolve. During this phase, the system specification is updated and CI development specifications are prepared. The end product of this phase is the establishment of the Allocated Baseline.

3.1.3 Full-Scale Development Phase. The FSD Phase begins when the system and end items are defined. During this phase requirements are updated formally as required, and the Preliminary Design Review (PDR) and the Critical Design Review (CDR) are conducted. The product, material and process specifications and other elements of the TDP are prepared and revised as necessary. Toward the end of the phase, after qualification of the equipment, the FCA is conducted. Even later, the PCA may be conducted using the TDP and prototype preproduction CIs. The successful PCA represents the establishment of the Product Baseline at the conclusion of the phase.

3.1.4 Production and Initial Deployment Phase. During this phase, the TDP is utilized for the manufacture of CIs. Any revision to the TDP is by formal update and change control. Status accounting is performed during this phase to correlate the TDP with the configuration status of units made available for inventory.

3.1.5 Operations Support Phase. Initial deployment marks the beginning of the Operations Support Phase. Operational readiness must be achieved and maintained by logistics personnel during this phase, and retrofit action must be taken when necessary.

3.2 REVIEWS AND AUDITS

The activities performed during each life cycle phase are numerous and varied, and may be tailored as necessary to meet the unique requirements of a program. Noteworthy among these activities are the following reviews and audits:

3.2.1 System Requirements Review. The objective of the System Requirements Review (SRR) is to ascertain the adequacy of the design activity's efforts in defining system requirements. It is normally conducted when most of the system functional requirements have been established near the end of the Concept Exploration/Definition Phase.

3.2.2 System Design Review. The System Design Review (SDR) is conducted to evaluate the optimization, correlation, completeness, and risks associated with the allocated technical requirements. It includes a summary review of the system engineering process which produced these requirements and of planning for forthcoming engineering efforts. Basic manufacturing considerations and planning for future production engineering are also reviewed. The SDR is conducted when the system definition effort has proceeded to the point where system characteristics are defined and the CIs are identified, typically during the Concept Demonstration/Validation Phase.

3.2.3 Software Specification Review. The Software Specification Review (SSR) is a review of the finalized Computer Software Configuration Item (CSCI) requirements and operational concept. The SSR is conducted when CSCI requirements have been sufficiently defined to evaluate the design activity's responsiveness to and interpretation of the system, segment, or prime item level requirements. A successful SSR is predicated upon the procuring activity's determination that the Software Requirements Specification (SRS) and Interface Requirements Specification(s) (IRS(s)) and other documentation form a satisfactory basis for proceeding into preliminary software design.

3.2.4 Preliminary Design Review. A PDR is used for each CI to address its functional elements and their related performance constraints, as selected and defined by the design activity's systems engineering analyses, prior to proceeding with the detail design. It is used to assess the chances that the selected functional elements will meet all of the CI's specified (Allocated Baseline) performance requirements. The series of reviews is normally completed early in the FSD Phase.

3.2.5 Critical Design Review. A CDR is conducted for each CI to address the design activity's detail design drawings (or software detail design documentation) prior to releasing them for manufacture/coding of qualification test (pre-production) articles. It is used to assess the chances that the selected detail design elements will meet all of the CI's specified (Allocated Baseline) performance requirements. The series of reviews is normally completed during the first half of the FSD Phase.

3.2.6 Functional Configuration Audit. An FCA is utilized for each CI to check the results of its qualification testing in order to verify that CI performance meets or exceeds development specification (Allocated Baseline) requirements. The series of audits is normally accomplished at the end of the FSD Phase.

3.2.7 Physical Configuration Audit. A PCA is conducted to verify that a CI's detail design documentation referenced in the CI's product specification matches the design of a unit being delivered to the procuring activity, and to verify that the design activity's engineering release system is capable of controlling the release of changes to the detail design documentation. The series of audits is held near the beginning of the Production and Initial Deployment Phase when a unit of the operational configuration is available. The audits may be accomplished at the end of the FSD Phase.

3.2.8 Test Readiness Review. A Test Readiness Review (TRR) is held for each CSCI to determine whether the software test procedures are complete and to assure that the design activity is prepared for formal CSCI testing. Software test procedures are evaluated for compliance with software test plans and descriptions, and for adequacy in accomplishing test requirements. At TRR, the procuring activity also reviews the results of informal software testing and any updates to the operation and support documents. A successful TRR is predicated on the procuring activity's determination that the software test procedures and informal test results form a satisfactory basis for proceeding into formal CSCI testing.

3.3 FORMAL BASELINES

The use of baselines is an integral CM concept. A baseline is commonly considered as a boundary between two program phases; in that context it serves as both the completion milestone of one program phase and the initiation point of the following phase. The three boundaries or baselines between the first four phases discussed above are the:

- ◆ Functional Baseline
- ◆ Allocated Baseline
- ◆ Product Baseline

A basic CM life cycle showing these phases and baselines is depicted in Figure 3.3-1.

3.3.1 Functional Baseline. The Functional Baseline is established by the Functional Configuration Identification (FCI) documentation. This baseline consists of approved elements of the TDP, namely performance oriented specifications prepared in accordance with military specification MIL-S-83490 and/or MIL-STD-490. This baseline also consists of any documents referenced within the specifications, such as drawings, associated lists, and other specifications. The approved FCI, i.e. the Functional Baseline, is a product of the Concept Exploration/Validation Phase and represents the culmination of that phase. The Functional Baseline is commonly the initiation point for the Concept Demonstration/Validation Phase.

3.3.2 Allocated Baseline. The Allocated Baseline is established by the Allocated Configuration Identification (ACI) documentation. This identification is used when CIs are part of a total system or a higher level CI; it documents functional requirements allocated to a CI from a system or higher level CI. This identification may be needed because of the complexity of the CI, for ease of project management, or for contractual integration or division of the total task. The Allocated Baseline is normally the starting point for the FSD Phase.

CONFIGURATION MANAGEMENT LIFE CYCLE

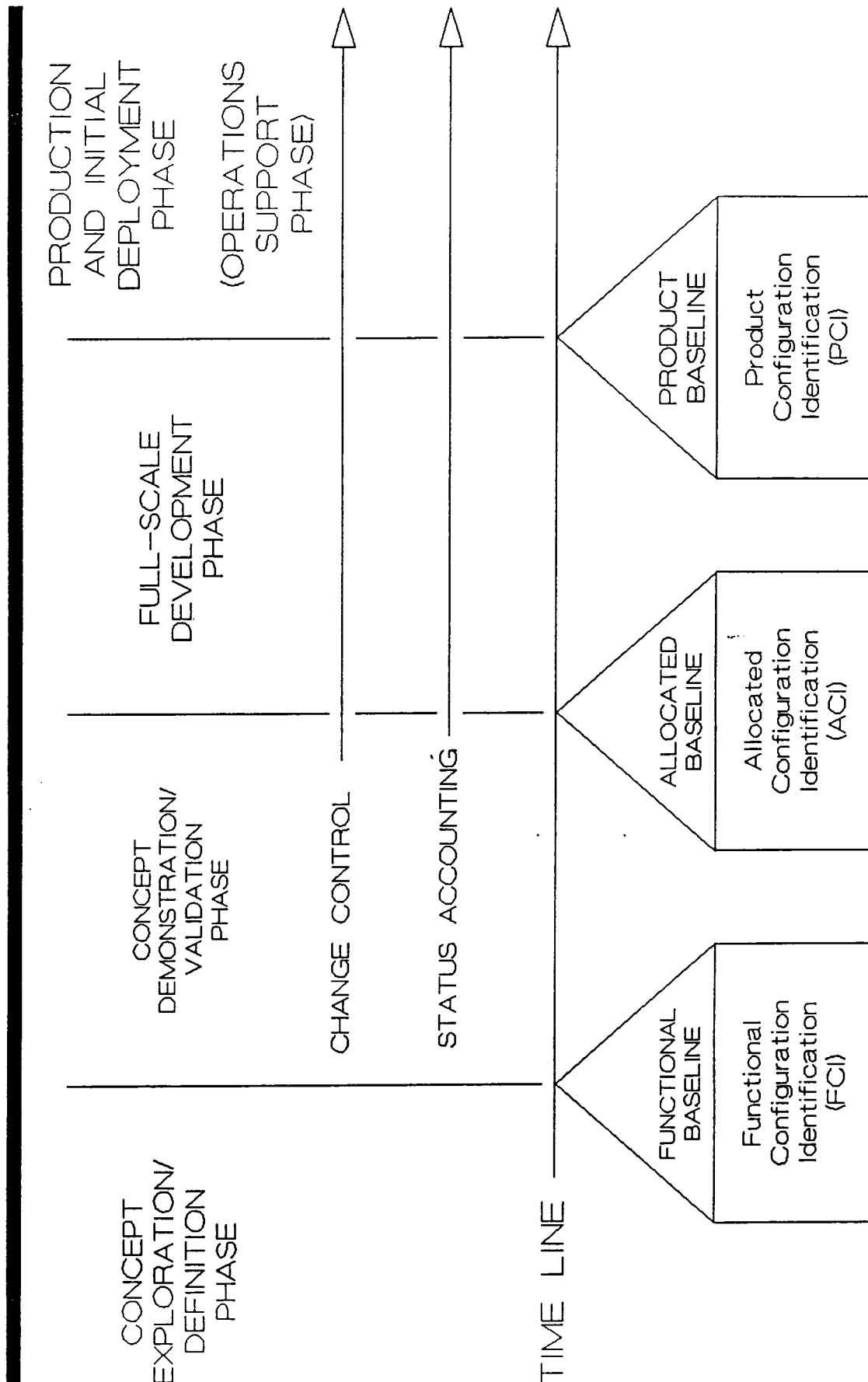


Figure 3.3-1

The ACI consists primarily of performance-oriented CI specifications prepared in accordance with MIL-S-83490 and/or MIL-STD-490. Each specification should include the necessary interface requirements with other associated CIs which comprise the higher level CI.

3.3.3 Product Baseline. The Product Configuration Identification (PCI) is used to prescribe necessary "build-to" or form, fit and function requirements for a CI and the acceptance tests for these requirements. The type and level of detail to be contained in the PCI should be determined, taking into consideration requirements for anticipated method of reprourement, for configuration audits, and for logistic support of potentially reparable items which are part of a CI. PCI documentation (the TDP) should be in accordance with MIL-S-83490, MIL-STD-490, Defense Standardization Manual 4120.3M, DOD-D-1000, DOD-STD-100, or other criteria, as applicable.

3.4 SOFTWARE BASELINES

Relative to the software baselines, DOD-STD-2167 refers to "developmental configuration" when addressing design activity controlled developmental baselines. The standard describes the activities, products and reviews associated with each step in the development of an item of software. These steps are: software requirements analysis, preliminary design, detailed design, code and unit test, component integration and test, and CSCI testing. Figure 3.4-1 shows a simplistic view of the internal software baseline process.

3.4.1 Functional and Allocated Baselines. The Functional and Allocated Baselines are established by the System Segment Specification and the SRS, respectively. The IRS is optional as another Allocated Baseline document, and is designed for use when a complex interface exists; otherwise, its content is covered by the SRS. Since these documents define the requirements for the CSCI, it is appropriate that they be baselined and released.

3.4.2 Developmental Baseline. As the development effort continues, the Software Design Document (SDD) is created to define the top level software design. It provides definition of the functional allocation and design features of the software at all levels of its hierarchy. The Interface Design Document (IDD) is an optional document to be used when interfaces are complex and it is desirable to define a shared database externally from a specific SDD. The SDD (along with the IDD, when used) is controlled internally as part of the developmental configuration. As software is coded and unit tested, it too is added to the developmental configuration. This is done by releasing the source and object code into a Software Document Control Library.

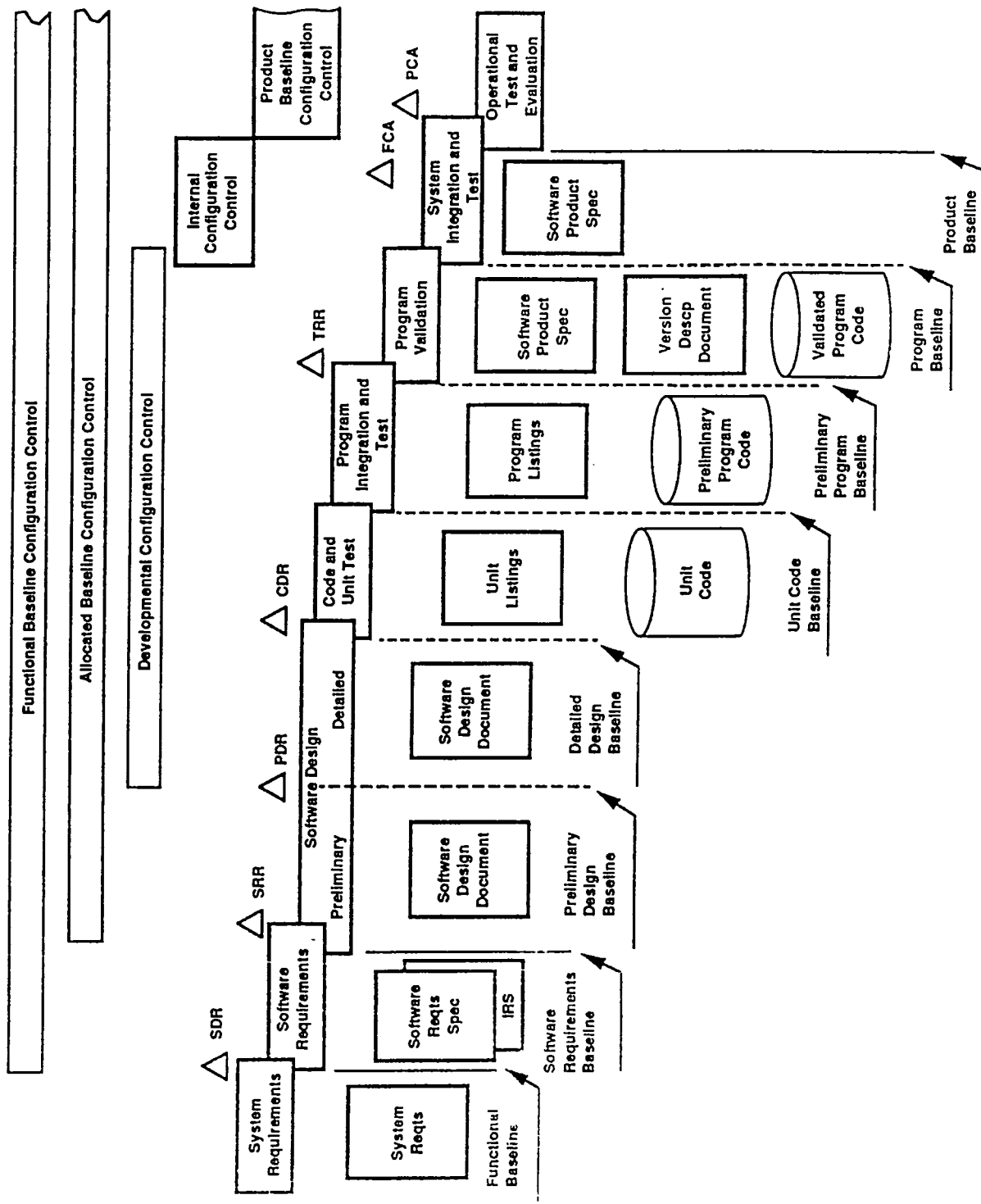


Figure 3.4-1 Typical Software Development Baseline Establishment and Control

3.4.3 Formal Product Baseline. When software development is complete, as proven at FCA and PCA, the developmental configuration documents are repackaged along with the applicable software source and object code listings into a Software Product Specification. The developmental configuration is superseded by the Product Baseline. The difference now is that the Product Baseline is established and is under formal control of the Government or other procuring activity; i.e., any impact to the design documents or code must be treated as a Class I or II engineering change.

3.5 INTERNAL BASELINE

An internal baseline is a control established by the design activity to ensure that proper internal coordination of software and/or hardware documentation is achieved. Internal baseline documentation normally does not include procuring activity review and/or approval.

3.5.1 Application. When drawings, specifications or other TDP elements are needed for design activity internal use prior to the time they are issued to and controlled by the procuring activity, it is usually advantageous to the design activity to coordinate changes to that documentation internally. This coordination may be needed for documentation, whether the documentation is being used for procurement purposes, for establishing requirements for development hardware/software, or for establishing system requirements for the program. Whenever it is desirable to establish an internal baseline for any of the TDP documents, the design activity needs to identify which program documents are to be internally controlled, and which internal organization(s) should participate in the coordination.

3.6 SUMMARY

This chapter addresses the life-cycle phases which are set forth by DODDs. It also discusses the formal and developmental baselines which form the groundwork for the phasing of many critical CM functions. Though these phases and baselines are addressed as fixed entities, quite frequently both are tailored to meet the particular needs of a program. Therefore, in many instances, programs may implement some derivation of or deviation to the milestones set forth herein.

Configuration Identification
EIA Bulletin CMB6-3
Chapter 4
Allocation of Configuration Identification Requirements

4.0 ALLOCATION OF CONFIGURATION IDENTIFICATION REQUIREMENTS

4.0.1 Introduction. One of the major functions in the identification of configurations is the top down division of program elements into appropriate and logical parts. This task is necessary regardless of whether the program is developing an entire hardware and/or software system or a component of an end item. This chapter provides guidance on those parameters that should be considerations in top-down breakdown tasks. The chapter also addresses some of the by-products which result from the allocations. One of the initial considerations should be the Work Breakdown Structure (WBS) requirements. Another consideration should be the criteria used in the selection of CIs. The products of the allocation decisions include hardware and software breakdowns as well as specification and drawing trees.

4.1 TOP-DOWN BREAKDOWN

An important facet of program element allocation is the top-down breakdown below the CI level. Consideration should be given to the subdivisions needed within both hardware and software. These are discussed in the following paragraphs.

4.1.1 Hardware Top-Down Breakdown. To establish a hardware top-down breakdown, the common practice is to start with the end article or CI. Next, the first item that logically can be taken off the CI is listed. This is the last item applied to the CI during assembly. All fasteners, label plates, etc., must also be listed in their proper sequence (indenture) to have a complete breakdown. Provisioning Parts Breakdown (PPB) and Illustrated Parts Breakdown (IPB) are two of the many examples requiring this form of indentured listing. The PPB reflects how the equipment is assembled and accounts for all the parts contained within the CI, assembly by assembly. The total part usage can then be used as a factor in calculating the quantity of spare parts to buy for a specified time period. The IPB is used to facilitate repair, maintenance and other corrective action where the technician is not familiar with the specific CI. The exploded views and the indenturing of the parts give the technician the immediate understanding of the order of assembly and how to gain access to parts, components and assemblies requiring attention. MIL-B-5000 is one of the specifications that describes the process of indenturing in support of both IPBs and PPBs.

4.1.2 Software Top-Down Breakdown. The rationale applied here is no different than that discussed in the foregoing paragraph. The only difference is the progression through a series of software functions which start with the major subsystem level and proceed through the lowest tier of computer program functions.

4.2 CONFIGURATION ITEMS

4.2.1 Selection of Configuration Items. The selection of CIs is a key decision making process as allocation of functional design elements takes place. MIL-STD-480 defines a CI as, "an aggregation of hardware, firmware, software, or any of its discrete portions, which satisfies an end use function and is designated for CM." CIs may vary widely in complexity, size and type, from an aircraft, ship or electronic system to a test meter or round of ammunition. During development and manufacture of the initial (prototype) production configuration, CIs are those items whose performance parameters and physical characteristics must be separately defined (specified) and controlled to provide management insight needed to achieve the overall end use function and performance.

According to MIL-STD-483, the major considerations that are to be given in the selection of CIs are as follows:

4.2.1.1 Level of Control. Because CIs must exist at manageable levels of assembly, their selection is usually limited to major subsystems/subassemblies. CI designation reflects an optimum management level at which the contracting agency specifies, contracts for, and accepts individual elements of a system.

4.2.1.2 Life Cycle Considerations. The selection is a management decision normally accomplished through the system engineering process in conjunction with CM and with the participation of logistics. Choosing too many CIs increases the cost of control; choosing too few or the wrong elements as CIs runs the risk of too little control through lack of management visibility. It must be determined what control needs to be exercised in light of cost/benefit trade-offs.

4.2.1.3 Schedule Considerations. Early identification as CIs of the major elements comprising a system will facilitate effective management as development progresses. As development is finalized and new logistics or technical considerations surface, additional hardware/software can be designated as CIs.

4.2.1.4 Common Subsystem Considerations. On development programs for subsystems or support equipment that will be common to more than one system, the basic CI should be that assembly which is common to all applications.

4.2.2 Effects of Configuration Item Selection. CI selection affects cost, schedule and performance for the contracting agency, prime contractors, subcontractors and suppliers. Selection of an item as a CI for manageability may be based on its administrative complexity, technical (engineering) criticality, or maintenance (logistics) criticality. The following is a MIL-STD-483 listing of the usual effects of CI designation:

- (1) Formal preparation of discrete configuration identification, most often in the form of a discrete development or requirements specification and a companion product specification.
- (2) Individual design review activity (PDRs and CDRs, for example) during development.
- (3) Individual qualification testing and reporting.
- (4) Individual PCAs and FCAs at the conclusion of development.
- (5) Discrete and separate "related" development preparation, review, approval and negotiation of ECPs (for changes to CIs).
- (6) Government approval of changes over the configuration identification governing the CI.
- (7) Continuing an accurate recording of the exact configuration status of the CI, including providing field activities precise data dealing with impending or completed modification actions.
- (8) Providing traceability of detailed design for follow-on activity, including historical data and individual status information for accident investigations, failure analysis, etc.
- (9) Separate identification indices and qualification records.
- (10) Separate nameplates and discrete CI identifiers (i.e., CI number, type, model, series, etc.).
- (11) Preparation of separate operator's and user's manuals.

4.3 SPECIFICATION TREE

In order to show the indentured relationships among specification-type documents or requirements documents, a schematic or block diagram called a specification tree is used. A specification tree (see Figure 4.3-1), shows the hierarchical relationship of the program specification documents and discloses the structure for the CM of such documents. The structure of the specification tree should reflect the indentured relationships of system segments, systems, subsystems, equipments, and related design requirements and constraints. A detailed discussion of specifications is included in chapter 5.

4.4 DRAWING TREE GENERATION BREAKDOWN

The purpose of a drawing tree (or indentured drawing list) is to show the relationship and structure of the various drawings comprising the TDP. The types of drawings are discussed in detail in Chapter 6. Drawing trees not only depict the drawings, but the related specifications, test documents, and lists as well. A drawing tree is normally an extension of, and should be consistent with, the specification tree. An example of a drawing tree is shown in Figure 4.4-1. In the figure, the top drawing is broken down into lower level assembly drawings. The assembly drawings are then broken down further into detail drawings, schematics, parts lists, and material, process and test documents. It is common practice to depict the drawing tree in list form known as a Generation Breakdown (GB). A GB usually shows the indentured level for all assemblies, parts, materials and processes which constitute the configuration of a CI or other piece of equipment.

4.5 WORK BREAKDOWN STRUCTURE

The WBS is a significant tracking framework used in Government contracts to analyze and allocate program elements. By definition, the WBS is a product oriented family tree composed of hardware, software, services, and other work tasks which completely define the project/program. A WBS displays and defines the product to be developed or produced and relates the elements of work to be accomplished to one another and to the end product. The WBS provides a consistent and visible framework that facilitates:

- ♦ A more effective management and technical base for planning and assigning management and technical responsibilities by operations within those Governmental offices responsible for the acquisition of defense and other materiel items, and those contractors furnishing the items.

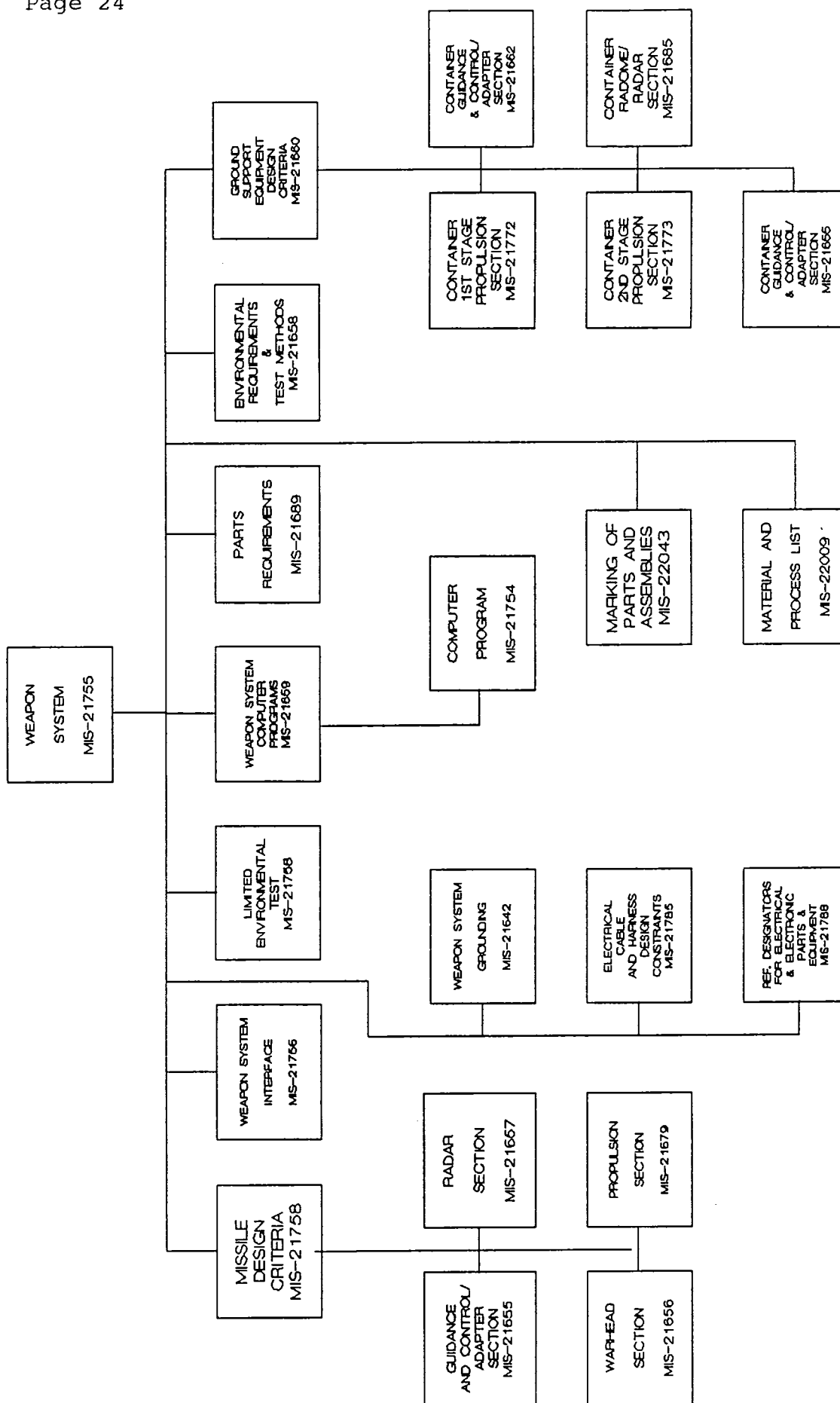


FIGURE 4.3-1
SPECIFICATION TREE