



TECHNICAL REPORT

CMB4-4

Issued 1982-05
Stabilized 2014-10

Configuration Change Control for Digital Computer Programs

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 CMB4_4

SAE values your input. To provide feedback on this Technical Report, please visit http://www.sae.org/technical/standards/CMB4_4

Documents

Document Sharing Hub



MAY 1982

CONFIGURATION MANAGEMENT

BULLETIN No. 4-4

Configuration Change Control for Digital Computer Programs

NOTE: CMB4, published in 1971, which was originally conceived to cover CM for Digital Computer Programs is currently being revised and superseded with a family of CM Bulletins as follows:

CMB4-0	CM for Digital Computer Programs (Will supersede CMB4)	CMB4 to be revised
CMB4-1	CM for Definitions for Digital Computer Programs (Supersedes Bulletin No. 4A)	Published May, 1982
CMB4-2	Configuration Identification for Digital Computer Programs	Published June, 1981
CMB4-3	Computer Software Libraries	Published February, 1981
CMB4-4	Configuration Change Control for Digital Computer Programs	Published May, 1982

Engineering Department

ELECTRONIC INDUSTRIES ASSOCIATION

Documents
Document Sharing Hub

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 non-member 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 or not 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.

Published by

ELECTRONIC INDUSTRIES ASSOCIATION
Engineering Department
2001 Eye Street, N.W.
Washington, D.C. 20006

PRICE: \$6.00

Printed in U.S.A.

EIA Bulletin No. 4-4
May, 1982

ELECTRONIC INDUSTRIES ASSOCIATION
CONFIGURATION MANAGEMENT
BULLETIN No. 4-4

CONFIGURATION CHANGE CONTROL
FOR
DIGITAL COMPUTER PROGRAMS

- THE SOFTWARE CHANGE CONTROL PROCESS -

Prepared by

EIA G-33 Committee
Computer Resources, Data and Configuration Management

EIA G-33 Task Group (CRTG)
Computer Resources



Documents
Document Sharing Hub

EIA Bulletin No. 4-4
May, 1982

FOREWORD

EIA Bulletin No. 4-4 is presented by the Electronic Industries Association G-33 Computer Resources, Data and Configuration Management Committee to explain the role of Computer Software Control in a configuration management system. The document content was derived from both Government and Industry sources and compiled, coordinated and edited by the Computer Resources Task Group (CRTG) of the G-33 Committee.

Comments or suggestions may be forwarded to:

G-33 Staff Engineer
EIA Headquarters
2001 Eye Street, N.W.
Washington, D.C. 20006

We extend our thanks and appreciation to the members of the Computer Resources Task Group and the full G-33 Committee who have cooperated by providing their time and talent to review the information we have assembled for this Bulletin.

The G-33 Committee and its Subcommittees will continue to devote their efforts to the advancement of knowledge and understanding of the subject of configuration management. Therefore, we invite your comments on this Bulletin and welcome your suggestions concerning publication of additional information to aid contract management in the effective conduct of configuration management.

TABLE OF CONTENTS

	<u>Page</u>
1. INTRODUCTION	1
2. FUNDAMENTAL PROBLEMS IN CONTROL	2
3. DEFINING THE CONTROL PROCESS	4
3.1 • Definition of Configuration Control	4
3.2 • Definition of Configuration Management	4
3.3 • The Role of Configuration Management	5
3.4 • The Software Control Process	6
4. PHASES OF THE CONTROL PROCESS	9
4.1 • Phase I	9
4.2 • Phase II	9
4.3 • Phase III	11
4.4 • Phase IV	12
5. THE TOOLS OF CONTROL	13
5.1 • Program Support Library	13
5.2 • Forms for Communication Changes	13
5.3 • The Unit Development Folder (UDF)	16
5.4 • Procedures for the Control Process	19
6. CONTROL AND COMMUNICATION PROCEDURES	21
6.1 • Requirements	21
6.1.1 • Librarian	21
6.1.2 • Unit Development Folder	22
6.1.3 • Program Trouble Report (PTR)	22
6.1.4 • Engineering Change Proposal and Deviation/Waiver	23
6.2 • Control Procedures	24
6.3 • Firmware	29
7. THE ROLE OF THE SOFTWARE CONFIGURATION CONTROL BOARD (SCCB)	33
- During Control Phase I and II	
- During Control Phase III	
- During Control Phase IV	
7.1 • Combined SCCB and CCB	34
8. CONTROL DURING THE OPERATIONAL AND MAINTENANCE PHASE	35
- Central Support Facility - Developer	
- User Maintenance	
9. CONCLUSIONS	42

ILLUSTRATIONS

FIGURE		PAGE
1	SOFTWARE CONTROL PROCESS (FLOW)	7
2	PHASES OF THE CONTROL PROCESS	10
3	SAMPLE OF AN IN-HOUSE (PTR)	14
4	ENGINEERING CHANGE PROPOSAL (ECP)	15
5	COMPUTER SOFTWARE CHANGE ORDER (SCO)	17
6	UNIT DEVELOPMENT FOLDER (UDF) STATUS FORMS	18
7	INITIAL CONTROL	20
8	PROGRAM TROUBLE REPORTS (FLOW)	24
9	CHANGE METHODOLOGY PRIOR TO RELEASE TO FINAL ACCEPTANCE/SELL-OFF	26
10	FORMAL ACCEPTANCE (FLOW)	28
11	SOFTWARE CHANGE METHODOLOGY (FLOW)	30
12	FIRMWARE CHANGE PROCESS (FLOW)	31
13	FUNCTIONAL ORGANIZATION SOFTWARE SUPPORT ACTIVITY	38
14	FUNCTIONAL FLOW - COMMUNICATION AND CHANGES DURING MAINTENANCE PHASE	39
15	CHANGE PROCESS FLOW FOR MAINTENANCE	40

May 1982

1. INTRODUCTION

The very nature of configuration management is the control of the configuration. This was the initial premise under which the control discipline was established and, later, formalized in DoD-D-5010.19 (1969) into a statement which defined control as:

"The systematic evaluation, coordination, approval or disapproval, and implementation of all approved changes in the configuration of a configuration item after formal establishment of its configuration identification".

What this definition does not impart is the scope and magnitude of the configuration change control task, or the rationale for configuration changes. The purpose of this bulletin is to describe the software change control process in the context of a structured programming environment. The bulletin provides a description of:

- o Problems inherent in or to the control process,
- o The control process,
- o The control environment as defined by the software life cycle,
- o The tools, forms, and procedures associated with each life cycle phase and the methodology instituted to attain a free flow of communication,
- o The role of the Configuration Control Board, and
- o Finally, the transition process from completion of the development phases to the Operational and Maintenance phase.

2. FUNDAMENTAL PROBLEMS IN CONTROL

Configuration Change Control is the most apparent and formal aspect of Configuration Management. Change Control is all too often thought to be, by those unfamiliar with the intent of the process, of a restrictive or policing-type nature. It should be the process required to communicate an orderly documentation of the changes necessary to maintain the currently approved configuration of the software systems' configuration items.

Problems resulting from lack of control may often be related to one of the following reasons:

- A. Authority for making changes is not established, resulting in changes being made which have an adverse effect on the software products and supporting documentation.
- B. Failure to properly document and promulgate written change control procedures or provisions for their implementation. This results in everyone implementing differing change methodologies and, thus, creating general confusion and added costs.
- C. Formal control is applied too early or too late in the development process, such as being applied before the design has been well-defined or being applied well into the test process, respectively. The result is either placing unnecessary restrictions on the design and coding process (too early) or a failure to provide traceability back to customer/user performance or design requirements (too late).
- D. Failure to document and record changes to the configuration identification, resulting in confusion and added costs.

- E. Formal controls are applied with too much formality, restrictiveness, or the process itself is too complex or cumbersome. Configuration Control thus becomes a cost driver, dampening overall project progress and resulting in the ineffective use of manpower and project funding.
- F. Failure to provide for the transition to, and application of, effective Configuration Change Control during the Operational and Maintenance phase, resulting in software degradation and inhibition of successful software modification or enhancement.
- G. Failure to tie the Change Control process into other software management techniques, such as unit development folders. The Design Review process, Acceptance and Delivery.

May 1982

3. DEFINING THE CONTROL PROCESS

3.1 DEFINITION OF CONFIGURATION CONTROL

Configuration control is the orderly communication of changes to an item at an established or agreed upon base line. This base line identifies a group of documents at a design milestone, a specified test or procedural event. The purpose of these documents is to define the characteristics needed to specify, build, test, and logistically support an item. Failure to control changes to these characteristics could seriously impact use of the item in its operational environment and escalate costs for both the contractor and user.

As previously stated in Section 1, Introduction, the Department of Defense (DoD) presented the first formal definition of configuration control in DoD Directive 5010.19 in 1969. That definition (repeated below) provided the focal point for any subsequent definitions of the control process.

Configuration Control - The systematic evaluation, coordination, approval or disapproval, and implementation of all approved changes in the configuration of a configuration item after formal establishment of its configuration identification.

This definition states the necessary activities employed to communicate a change. If anything is required to be added to it, then the incorporation of the term "communication" could be considered to enhance the description of the methodology used to document a change to a configuration item, thereby changing established base line points.

3.2 DEFINITION OF CONFIGURATION MANAGEMENT

The accepted general definition of Configuration Management⁽¹⁾ emphasizes the three basic elements of Identification, Control, and

(1) See EIA CM Bulletin 4-1 - DEFINITIONS for further amplification

Accounting. Control is the focal point of this hierarchy -- we must have a viable identity to control to, and we must have control activity to initiate accountability.

The Federal Information Processing Standards' (FIPS) definition of Configuration Management closely follows the DoD definition stated in Paragraph 3.1, and a survey of industry procedures further demonstrates very little deviation from the DoD's definition. One should refer to EIA CM Bulletin 4-1 for additional software/CM definitions.

3.3 THE ROLE OF CONFIGURATION MANAGEMENT

Configuration Management is the focal point for communication and is responsible for establishing a well-conceived control process that embodies all of the elements of good communication in a well-phased software development cycle and the subsequent operational phase.

A Software Configuration Management Plan (SCMP), as developed by Software Configuration Management (SCM), spells out the control process to be implemented by SCM during each phase of the software development cycle. (NOTE 1)

The SCMP's Change Control Chapter describes the base lines to be established by SCM, what will comprise each of the base lines, and the milestone points at which they will become firm. In a structured, top-down, programming environment, a series of design reviews will be held, resulting in the final build base line. Structured, top-down programming precludes the establishment of traditional single-event design reviews, normally conducted in the classical development environment.

NOTE 1: This does not preclude use of standard operating procedures when one is involved with an established, continuing and homogenous product line.

May 1982

In addition, the SCMP includes the procedures, forms, and control methodology planned for implementation by SCM. Procedures and forms, in particular, are best conceived and developed by SCM at the time the proposal is first submitted; thus they are ready to be implemented by SCM at the time of contract go-ahead.

3.4 THE SOFTWARE CONTROL PROCESS

The flow illustrated in Figure 1 describes a typical software control process. The control process begins with receipt, or establishment of, a Systems Specification (Type A per Air Force MIL-STD-490) or equivalent. The Navy's terminology would be a Systems Operational Specification/Document (SOS/SOD). The process evolves from the System Specification through the establishment and initiation of the Performance Specification and, finally, the Design Specification.

As the specifications are approved, released and base lined, internal control is established. If acceptance is made by the customer, formal controls requiring customer approval of changes are also invoked.

At the point when the coding process is considered complete, i.e., end of module test, and the software is under Library control,⁽²⁾ the internal control process becomes more involved.

More detailed analysis, and more formal reviews are required prior to a change being implemented.

These internal developmental phase controls then give way, on acceptance by the customer of documents such as the Part II Specification or Program Description Document, to the

(2) See EIA CM Bulletin 4-3, Computer Software Libraries.

The flowchart illustrates the Software Change Control (SCC) process, organized into three main phases: I (Contractual Baselines), II (Internal Baselines), and III (Control Phase).

Phase I: Contractual Baselines

- FUNCTIONAL BASELINE** (Oval) leads to **FUNCTIONAL DFN** (Rectangular box).
- ALLOCATED BASELINE** (Oval) leads to **REQMT'S DEFINITION** (Rectangular box).
- DESIGN RELEASE** (Rectangular box) leads to **DESIGN SPEC** (Parallelogram).

Phase II: Internal Baselines

- SYSTEM SPECIFICATION** (Parallelogram) leads to **PERFORMANCE SPEC** (Parallelogram).
- PERFORMANCE SPEC** leads to **CODE SOURCE OBJECT** (Rectangular box).
- CODE SOURCE OBJECT** leads to **MODULE TEST** (Rectangular box).
- MODULE TEST** leads to **INTEGRATION TEST** (Rectangular box).
- INTEGRATION TEST** leads to **DESCRIPTION DOC** (Parallelogram).
- DESCRIPTION DOC** leads to **DELIVERY & ACCEPT** (Rectangular box).
- DELIVERY & ACCEPT** leads to **PRODUCT & OPERATIONAL BASELINE** (Oval).

Phase III: Control Phase

- OPERATIONAL DEFINITION** (Rectangular box) leads to **DESCRIPTION DOC**.
- LIBRARY CONTROL** (Rectangular box) leads to **INTEGRATION TEST**.
- PROG PRG** (Parallelogram) leads to **DESCRIPTION DOC**.
- PROBLEMS, ERRORS** (Text) leads to **MODULE TEST**.
- ITERATE CODE & TEST (SOFTWARE MANAGER APPROVAL)** (Text) leads to **CODE SOURCE OBJECT**.
- ITERATE DESIGN (SOFTWARE MANAGER APPROVAL)** (Text) leads to **PERFORMANCE SPEC**.
- SOFTWARE CHANGE ORDERS** (Parallelogram) leads to **REVIS AS DIRECTED & UPDATE BASELINES** (Text).
- REVIS AS DIRECTED & UPDATE BASELINES** leads to **PERFORMANCE SPEC**.
- PROPOSED CHANGE** (Text) leads to **PROID B.I. ESTABLISHED?** (Decision diamond).
- PROID B.I. ESTABLISHED?** (Decision diamond):
 - YES** leads to **FUNCTION OR ALLOC B.I. IMPACTED?** (Decision diamond).
 - NO** leads to **SOFTWARE CHANGE ORDERS**.
- FUNCTION OR ALLOC B.I. IMPACTED?** (Decision diamond):
 - YES** leads to **SOFTWARE CHANGE ORDERS**.
 - NO** leads to **SOFTWARE CHANGE ORDERS**.
- SOFTWARE CHANGE ORDERS** leads to **CLASSIFICATION AND DIRECTION** (Decision diamond).
- CLASSIFICATION AND DIRECTION** (Decision diamond):
 - CLASS I** leads to **CUSTOMER APPROVAL** (Rectangular box).
 - CLASS II** leads to **CUSTOMER APPROVAL**.
 - CLASS III** leads to **CUSTOMER APPROVAL**.
- CUSTOMER APPROVAL** leads to **SCCB APPROVAL** (Text).
- SCCB APPROVAL** leads to **SOFTWARE CONFIG CONTROL BOARD (SCCB)** (Rectangular box).
- SOFTWARE CONFIG CONTROL BOARD (SCCB)** leads to **SOFTWARE CHANGE ORDER** (Parallelogram).
- SOFTWARE CHANGE ORDER** leads to **IMPLEMENT (REVIEW APPROVAL)** (Text).
- IMPLEMENT (REVIEW APPROVAL)** leads to **SOFTWARE CHANGE ORDER**.
- SOFTWARE CHANGE ORDER** leads to **SOFTWARE CHANGE REVIEW (1)** (Rectangular box).
- SOFTWARE CHANGE REVIEW (1)** leads to **BASELINES AFFECTED?** (Decision diamond).
- BASELINES AFFECTED?** (Decision diamond):
 - YES** leads to **SOFTWARE IN LIBRARY?** (Decision diamond).
 - NO** leads to **SOFTWARE IN LIBRARY?**.
- SOFTWARE IN LIBRARY?** (Decision diamond):
 - YES** leads to **MINOR CODE DOCUMENT OR PROCEDURE CHANGE ONLY?** (Decision diamond).
 - NO** leads to **MINOR CODE DOCUMENT OR PROCEDURE CHANGE ONLY?**.
- MINOR CODE DOCUMENT OR PROCEDURE CHANGE ONLY?** (Decision diamond):
 - YES** leads to **IMPLEMENT (REVIEW APPROVAL)**.
 - NO** leads to **IMPLEMENT (REVIEW APPROVAL)**.

Legend:

- IN SCOPE**
- ORG DESIGN**
- CLASS II**
- CLASS I**
- COMPATIBILITY**

Notes:

- *ECP INCLUDES SOFTWARE CHANGE PROPOSAL SOFTWARE ENHANCEMENT PROPOSAL AS APPROPRIATE**
- (1)** SOFTWARE CHANGE REVIEW
- (2)** SOFTWARE CHANGE ORDER



Document Sharing Hub

PROBLEM HARD COPY

May 1982

Operational/Deployment Phase Controls. These controls require the same processing but are now under a field operations and maintenance environment. As such, the changes received are oriented to adaptive, modification and enhancement-type changes rather than corrective.

The SCMP must, therefore, provide the guidance establishing and implementing the software control procedures in an evolutionary and effective manner by describing the formal base line points and the internal base lines, such as the requirements, design, release, and control points.