ANSI/EIA-649-1998
Approved: July 10, 1998

EIA STANDARD

EIA-649

National Consensus Standard for Configuration Management

EIA-649

AUGUST 1998

ELECTRONIC INDUSTRIES ALLIANCE

GOVERNMENT ELECTRONICS AND
INFORMATION TECHNOLOGY ASSOCIATION
ENGINEERING DEPARTMENT

GEIA
Government Electronics and Information Technology Association

A SECTOR OF

EIA
Electronic Industries Alliance

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.

Standards and Publications are adopted by EIA in accordance with the American National Standards Institute (ANSI) patent policy. By such action, EIA does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the Standard or Publication.

This EIA Standard is considered to have International Standardization implication, but the International Electrotechnical Commission activity has not progressed to the point where a valid comparison between the EIA Standard and the IEC document can be made.

This Standard does not purport to address all safety problems associated with its use or all applicable regulatory requirements. It is the responsibility of the user of this Standard to establish appropriate safety and health practices and to determine the applicability of regulatory limitations before its use.

(From Standards Proposal No. 3721, formulated under the cognizance of the G-33 Data and Configuration Management Committee.)

Published by

ELECTRONIC INDUSTRIES ALLIANCE 1998
Engineering Department
2500 Wilson Boulevard
Arlington, VA 22201

PRICE: Please refer to the current
Catalog of EIA, JEDEC, and TIA STANDARDS and ENGINEERING PUBLICATIONS
or call Global Engineering Documents, USA and Canada (1-800-854-7179)
International (303-397-7956)

PLEASE!

DON'T VIOLATE
THE
LAW!

National Consensus Standard for Configuration Management

Contents

Contents *(continued)*

Foreword

In 1994, the Electronic Industries Alliance's G-33 Committee on Data and Configuration Management initiated the task of developing this industry configuration management standard. A core working group was established under ANSI project PN-3414. An interim EIA standard was published in August 1995 after completion of the committee letter ballot process and approval by the EIA Engineering Department Executive Committee (EDEC).

A new ANSI project, PN-3721 was initiated in March 1996 and a new core group was established for the current revision of the standard, which replaces EIA/IS-649 in whole. The EIA letter ballot process was completed in June 1997. An ANSI Ballot version was made available to all interested parties for review in July 1997. In addition to the core working group, a multi-association advisory group participated in review of the standard and disposition of comments to the ANSI Standards Proposal ballot. The ANSI SP Ballot process was successfully completed in May 1998 in accordance with ANSI approved EIA procedures. The publication of the standard was approved by EDEC. The Electronic Industries Alliance would like to acknowledge the unselfish dedication and extraordinary effort that was voluntarily applied by the participants, listed in Annex D, to the accomplishment of these projects.

Significant technical changes between EIA/IS-649 and EIA Standard 649 are as follows:

1. The word "Data" was added to the generic categories to which references to the word "Product" include. (Introduction, Table 2)
2. A methodology for selecting the CM principles and attendant practices that are appropriate to a given product was added. (Clause 1, Scope, Clause 5, Requirements and Annex B).
3. Software Code and Test was relocated from the Build phase to the Definition phase of the product life cycle to be consistent with the statement that a product configuration baseline is normally established at the end of the definition phase.(Introduction, Table 1)
4. Reference to additional operation and disposal phase baselines was added. (5.2.5.2.d.)
5. The minor change classification was modified to reflect the fact that a minor change can correct or modify parts as well as documents and processes, so long as it does not impact the factors that would classify it as major. (5.3.1.2, Table 4)
6. Configuration status accounting information typically available in each life cycle phase was revised to reflect change requests and proposals in the conceptual phase.(5.4.1, Table 5)
7. The inference that design verification plans (for complex designs) typically require customer approval was removed. (5.5.1)
8. Verifications necessary for software products were clarified.(5.5.1)
9. Safety procedures were added to the material necessary to perform a configuration audit. (5.5.2)
10. Application notes were modified to clarify:
    (a) that the standard may be used as a source from which applicable information can be extracted to prepare such items as a request for proposal, or an evaluation or certification checklist, and
    (b) that the application of appropriately selected principles and practices to a product or on a project will enable the user to plan and document an appropriate configuration management program. (Clause 6.)
11. The standard was revised in several respects in accordance with the EIA style guide, which is consistent with ANSI and ISO requirements. Certain material such as acknowledgments, related documents, and a number of statements in the scope and introduction were revised and relocated. Relationship of EIA 649 to other standards and documents was moved to Annex C.

Configuration management, a discipline popularized by its use in the acquisition of defense systems, is widely used for commercial products and services. When configuration management principles are applied using effective practices, return on investment is maximized and both product and service life cycle costs are reduced. Intended users of this document are any organization or individual that can benefit from the application of configuration management. The US Department of Defense and the Internal Revenue Service have adopted EIA Standard 649 for use. This standard

provides an understanding of the technical/program management principles fundamental to configuration management and the best practices used to implement them. It is designed for widespread application on a selective basis. Clauses 1 through 5 are normative. Clause 6 and all annexes in this standard are informative.

Introduction

Configuration management, applied over the life cycle of a product, provides visibility and control of its performance, functional, and physical attributes. Configuration management verifies that a product performs as intended, and is identified and documented in sufficient detail to support its projected life cycle (e.g., fabrication or production, operation, maintenance, repair, replacement, and disposal). References to a product in this standard should be interpreted as applicable to the generic product categories of hardware, software, processes, data, materials, or services (this is consistent with the definition of "product" in ISO Standard 8402).

The configuration management process facilitates orderly management of product information and product changes for such beneficial purposes as to revise capability; improve performance, reliability, or maintainability; extend life; reduce cost; reduce risk and liability; or correct defects. The relatively minimal cost of implementing configuration management is returned many fold in cost avoidance. The lack of configuration management, or its ineffectual implementation, can be very expensive and sometimes can have such catastrophic consequences as failure of equipment or loss of life.

The purpose and *benefits* of configuration management include the following:

- Product attributes are defined. *Provides measurable performance parameters. Both Buyer and Seller have a common basis for acquisition and use of the product.*
- Product configuration is documented and a known basis for making changes is established. *Decisions are based on correct, current information. Production repeatability is enhanced.*
- Products are labeled and correlated with their associated requirements, design and product information. *The applicable data (such as for procurement, design or servicing the product) is accessible, avoiding guesswork and trial and error.*
- Proposed changes are identified and evaluated for impact prior to making change decisions. *Downstream surprises are avoided. Cost and schedule savings are realized.*
- Change activity is managed using a defined process. *Costly errors of ad hoc, erratic change management are avoided.*
- Configuration information captured during the product definition, change management, product build, distribution, operation, and disposal processes, is organized for retrieval of key information and relationships, as needed. *Timely, accurate information avoids costly delays and product down time; ensures proper replacement and repair; and decreases maintenance costs.*
- Actual product configuration is verified against the required attributes. Incorporation of changes to the product is verified and recorded throughout the product life. *A high level of confidence in the product information is established.*

From a historical perspective, configuration management practices were first formalized in the defense and space communities, bringing related industry best practices together under a common framework. These practices have become entrenched as standard approaches in various industry segments for a variety of reasons. Software developers practice configuration management to identify and control versions of their product. The automotive industry practices configuration management to support the spare parts market, to track warranties, and to be able to effect recalls, when necessary. The nuclear power and armament production industries use configuration management to maintain both products and facilities. Each industry segment emphasizes particular aspects of this multi-faceted process and has evolved its own terminology and methodology for the selected configuration management practices they employ.

A major reason for perceived differences in understanding configuration management is that the emphasis placed on various configuration management principles and practices differs in each phase of the product life cycle. In Table 1, the life cycle of a product has been broken down into different phases with a generic title and description given to each phase. These phase names are intended to be as generic as possible so that they can be easily mapped to the myriad of

different life cycle models in use. To encompass a broad range of environments, the table illustrates some of the aliases for each phase. (DoD program life cycle phases are in boldface.)

Table 1— Phases of a Product's Life Cycle

Phases	**Conception**	**Definition**	**Build**	**Distribution**	**Operation**	**Disposal**
Aliases	Marketing **Concept** Study Research Exploration Pre-Development	Development Design Engineering **Program Definition & Risk Reduction** **Engineering & Manufacturing Development** Coding/Software Build[1]	Fabrication **Production** Construction Manufacturing	Sales Delivery Installation **Fielding Deployment**	Operational Maintenance Warranties Service Life Performance **Operation & Support** Repair	Removal From Service Disposition Unsupported
Characteristics	Need Opportunity Mission Analysis Trade-Offs Investigation Survey Functions Pre-Concept & Concept Definitions	System Definition Specification Architecture Preliminary Design Detailed Design Software Code & Test Manufacturing Planning Prototyping Testing Evaluation	Facility Construction Production Assembly Installation[1] Inspection	Order Supply Stock Transport Acceptance Deployment Installation Setup	Use Utilization Operate Maintain Service Depreciate	Mothball Discard Deactivate Destroy Disassemble Scrap Recycle Disposition

Note: 1. Alias or characteristic may apply in more than one product phase.

Regardless of the titles chosen for these phases or whether the product is a facility, computer software, an airplane or a machine screw, at one time in its history, a product will go through all or most of these phases. The phases can have considerable overlap. A product can be in build, distribution and operation phases simultaneously, while changes to its current design are in the definition phase.

The basic principles of configuration management come into play in varying degrees in each phase. To tailor configuration management for a given product application, identify the applicable phase(s), select the appropriate configuration management principles that apply, and determine the degree to which they apply during each phase.

Neutral terms are used in this standard. There is no intent to express preference for any particular set of terminology. A summary table of aliases or equivalents for terms used in this standard is provided in Table 2. When planning and documenting a CM program, these and other aliases may be freely substituted for the neutral terminology.

References to such terms as the enterprise, performing activity, developing activity, or producing activity refer to that organization or agency that has the responsibility for performing configuration management for a given product during some period of its life cycle. This organization could be a commercial enterprise, a contractor, a subcontractor, or a government agency. Configuration management functions related to a given product may be the responsibility of several organizations during its life cycle, e.g., one organization

Table 2— Table of Common Aliases

Terms Used	**Alias** *(Synonymous)* **or Equivalent** *(Conceptually Similar)* **Terms**
Customer - External	Buyer; End User, External end user, Procuring Activity, Acquirer
Customer - Internal	Management; Marketing Department; Specification Activity, Internal user, etc.
Enterprise; Organization, Supplier	Performing Activity; Developing Activity; Design Activity; Agency Contractor; Subcontractor; Vendor, Seller, Provider
Life Cycle Phases	(See Table 1)
Product Servicing; Support	Integrated Logistic Support, Logistic Support, Acquisition Logistics
Configuration Identification Terms:	
Attributes	Characteristics, Requirements
Configuration Documentation	Specification; Requirements Document; Design Basis; Attributes; Interface document: Interface Control Document; Engineering Drawing; Design Information; Design Output; Software Requirements Specification; Software Design Document; Software Product Specification
Configuration Identification	Product Definition
Design Information/Documentation	Engineering Drawings and Associated lists
Design Release Baseline	Developmental Configuration; Release Baseline;
Fixed (as baseline)	Established
Group Product Unit Identifier	Lot Number, Batch Number; Block Number
Individual Product Unit Identifier	Serial Number
Interface Document	Interface Control Document, Interface Control Drawing, Interface Specification
Product	Hardware; Software; Process; Data, Material or Service; Configuration Item; End Item; Item; System; Set; Group; Component; Part; Assembly; Unit, Entity
Product Configuration Baseline	Product Baseline
Product Information	Configuration Documentation, Product Use Documentation, Test Document; Other derived information
Product Structure	Hierarchy; Product Tree; Pyramid; Top-down breakdown; Indentured listing; Bill of Material (BOM)
Product Use Information/Documentation	Operation and Maintenance Instructions and other derived information; Operational Information
Requirements Baseline	Functional Baseline; Performance Requirements; Promise to Customer Allocated Baseline
Requirements Document	Specification
Unique Identifier	Part Number; Name; Dash Number; Product Identifier; Model, Version, Document Identifier
Change Management Terms:	
Change Board	Configuration Control Board; Change Control Board; Change Review Board; Program Review Board; Integrated Product Team
Change Notice; Document Change Notice	Engineering Order; Engineering Change Order; Engineering Change Notice; Specification Change Notice
Change Proposal	Engineering Change Proposal; Engineering Change Package
Change Request	Engineering Change Package; Change Request/Directive
Change Requester	Originator, Preparer of request
Change Sponsor	Cognizant Engineer or Manager
Configuration Change Management	Configuration Control; Change Control
Effectivity	Affected Serial Number(s); Change Applicability; Series; Incorporation Date(s); Block; Lot; Point of embodiment
Variance	Deviation; Waiver; Engineering or Production Departure
Accounting & Audit Terms:	
Configuration Status Accounting	Product Configuration Information
Configuration Verification and Audit	Product Configuration Verification; Functional and Physical Configuration Audit; Product Consistency Verification

that designs and builds a product, performs configuration management during the definition and build phases, and a second organization that is responsible for upgrading the product and servicing malfunctioning units, performs configuration management during the operation phase.

References to the "customer" should be interpreted as the organization(s) that specify requirements (performance attributes) for the product or that acquire and use the product. A customer may be external to the developing and producing organization, or may be an internal customer such as a marketing, management, or using department. When **t**he term "customer" is used alone, either condition applies. When one condition applies but the other does not, customer is modified by either "external" or "internal," as applicable.

Although the term performance relates to functional and physical attributes; and the definition of functional attributes includes performance parameters, the phrase "performance, functional, and physical" is used in this standard to refer to a product's attributes, and to emphasize the importance of performance.

1 Scope

Configuration management (CM) principles underlie sound business practices being used throughout industry and government to provide:

- The orderly establishment, documentation, and maintenance of a product's functional, performance and physical attributes
- Management of changes to the attributes
- Access to accurate information essential to the product's development, fabrication, production, use, maintenance, procurement, and eventual disposal.

This standard presents configuration management from the viewpoint that configuration management practices are employed because they make good business sense rather than because requirements are imposed by an external customer. The standard discusses configuration management principles and practices from an enterprise view; it does not prescribe which CM activities individual organizations or teams within the enterprise should perform. Each enterprise assigns responsibilities in accordance with its own management policy.

The standard explains the major CM functions rather than mandates them. The explanation includes purpose, benefits, and best practices. Within each topic, the basic principles of configuration management are addressed. The principles are selectively applicable to a broad range of customers, products and industries. A tabulated summary of the configuration management principles is provided in Annex A.

All of the configuration management principles are applicable to a product environment in which a robust configuration management approach is necessary. However, in other environments, some of the principles may not apply. Annex B provides a methodology for selecting which principles are appropriate. It prescribes that a benefit-risk assessment be conducted before discarding principles that, although desirable, appear to be too costly. All too often, the "best practices" associated with a principle are equated with "most costly" when, in fact they represent "best investment." In the long run, these investments, like insurance, may prove to be the "least costly" alternative.

Configuration management practices should be applied selectively, and to a degree commensurate with the product application environment. Practices that are unique to specific product types may be covered in individual "Application Practice" annexes to be added to the standard through future revisions as needed.

This standard is not intended for use as a compliance document or an evaluation mechanism for CM programs; rather it is intended for use as a source document for either purpose. The application of the principles in this standard to a product, for a project, or for an enterprise, will enable the user to plan and document an appropriate configuration management program.

2 Normative References

This standard does not contain any provisions that, through reference in the text, constitute provisions of this standard. Therefore there are no normative references. Informative references may be found in Annex C.

3 Definitions

For purposes of this standard, the following definitions apply

NOTE — Prominent aliases appear, as follows: "term; synonym." See Table 2 for other aliases. Where more than one meaning is commonly attributed to a term, alternate numbered definitions are provided.

application environment: Where a product is used, for example, defense systems and facilities, energy facilities, aircraft, space systems, automobiles, pharmaceuticals, commercial products.

approval: The agreement that an item is complete and suitable for its intended use.

approved data: Data that has been approved by an appropriate authority and is the official (identified) version of the data until replaced by another approved version.

archived data: Released or approved data that are to be retained for historical purposes.

attributes: Performance, functional and physical characteristics of a product.

baseline: (1) An agreed-to description of the attributes of a product, at a point in time, which serves as a basis for defining change. (2) An approved and released document, or a set of documents, each of a specific revision; the purpose of which is to provide a defined basis for managing change. (3) The currently approved and released configuration documentation. (4) A released set of files comprising a software version and associated configuration documentation.

change: See engineering change.

configuration: (1) The performance, functional, and physical attributes of an existing or planned product, or a combination of products. (2) One of a series of sequentially created variations of a product.

configuration audit: Product configuration verification accomplished by inspecting documents, products and records; and reviewing procedures, processes, and systems of operation to verify that the product has achieved its required attributes (performance requirements and functional constraints) and the product's design is accurately documented. Sometimes divided into separate functional and physical configuration audits.

configuration change management; configuration control: (1) A systematic process which ensures that changes to released configuration documentation are properly identified, documented, evaluated for impact, approved by an appropriate level of authority, incorporated, and verified. (2) The configuration management activity concerning: the systematic proposal, justification, evaluation, coordination, and disposition of proposed changes; and the implementation of all approved and released changes into (a) the applicable configurations of a product, (b) associated product information, and (c) supporting and interfacing products and their associated product information.

configuration documentation: Technical information, the purpose of which is to identify and define a product's performance, functional, and physical attributes (e.g., specifications, drawings).

configuration identification; product definition: (1) The systematic process of selecting the product attributes, organizing associated information about the attributes, and stating the attributes. (2) Unique identifiers for a product and its configuration documents. (3) The configuration management activity that encompasses selecting configuration documents; assigning and applying unique identifiers to a product, its components, and associated documents; and maintaining document revision relationships to product configurations.

configuration management (CM): A management process for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design and operational information throughout its life.

configuration status accounting (CSA); product configuration information: The configuration management activity concerning capture and storage of, and access to, configuration information needed to manage products and product information effectively.

configuration verification: The action verifying that the product has achieved its required attributes (performance requirements and functional constraints) and the product's design is accurately documented.

contract: As used herein, denotes the document (for example, contract, memorandum of agreement or understanding, purchase order) used to implement an agreement between a customer (buyer) and a seller (supplier).

data: Recorded information of any nature (including administrative, managerial, financial, and technical), regardless of medium or characteristics.

design information: Technical information resulting from translating requirements for a product into a complete description of the product.

disapproval: Conclusion by the appropriate authority that an item submitted for approval is either incomplete or not suitable for its intended use.

document representation: A set of digital files that collectively represent a complete digital document

effectivity: A designation defining the product range (e.g., serial, lot numbers, model, dates) or event at which a change to a specific product is to be (or has been) effected or to which a variance applies.

engineering change: Any alteration to a product or its released configuration documentation. Effecting an engineering change may involve modification of the product, product information and associated interfacing products.

firmware: The combination of a hardware device and computer instructions or computer data that reside as read-only software on the hardware device. The software cannot be readily modified under program control..

fit: The ability of a product to interface or interconnect with or become an integral part of another product.

form: The shape, size, dimensions, and other physically measurable parameters that uniquely characterize a product. For software, form denotes the language and media.

function: The action or actions that a product is designed to perform.

functional attributes: Measurable performance parameters including reliability, maintainability, and safety.

hardware: Products made of material and their components (mechanical, electrical, electronic, hydraulic, pneumatic). Computer software and technical documentation are excluded.

interchangeable: A product which possess such functional and physical attributes as to be equivalent in performance to another product of similar or identical purposes; and is capable of being exchanged for the other product without selection for fit or performance, and without alteration of the products themselves or of adjoining products, except for adjustment.

interface: The performance, functional, and physical attributes required to exist at a common boundary.

interface control: The process of identifying, documenting, and controlling all performance, functional, and physical attributes relevant to the interfacing of two or more products provided by one or more organizations.

interface documentation: Interface control drawing or other documentation that depicts physical, functional, and test interfaces of related or co-functioning products.

life cycle: A generic term relating to the entire period of conception, definition, build, distribution, operation and disposal of a product.

nomenclature: (1) Names assigned to kinds and groups of products. (2) Formal designations assigned to products by customer or supplier (such as model number, or model type, design differentiation, specific design series or configuration.)

nonconformance: Nonfulfillment of a specified requirement

operational information: Information that supports the use of a product, for example, operation maintenance and user's manuals/instructions, procedures, and diagrams.

organization code: Designator for a specific organizational entity, e.g., Commercial and Government Entity (CAGE) code.

NOTE — As used in practice, the code denotes the organization responsible for design and/or manufacture of a product. A product may be designed by one organizational entity and manufactured by a different organizational entity with a different organization code.

original: The current design activity's document or digital document representation and associated source data file(s) of record (i.e., for legal purposes.)

performance: A quantitative measure characterizing a physical or functional attribute relating to the execution of an operation or function. Performance attributes include quantity (how many or how much), quality (how well), coverage (how much area, how far), timeliness (how responsive, how frequent), and readiness (availability, mission/operational readiness). Performance is an attribute for all systems, people, products and processes including those for development, production, verification, deployment, operations,

support, training and disposal. Thus, supportability parameters, manufacturing process variability, reliability and so forth, are all performance measures.

physical attributes: Quantitative and qualitative expressions of material features, such as composition, dimensions, finishes, form, fit, and their respective tolerances.

product: Anything that is used or produced to satisfy a need, for example, facilities, systems, hardware, software, firmware, data, processes, materials, or services.

product information: Information related to a product including configuration documentation and other information that is derived from configuration documentation (e.g., instruction manuals, manufacturing instructions, catalogs).

release: The designation by the originating activity that a document or software version is approved by an appropriate authority and is subject to configuration change management procedures.

released data: (1) Data that has been released after review and internal approvals. (2) Data that has been provided to others outside the originating group or team for use (as opposed to for comment).

requirements: Specified essential attributes.

retrofit: The incorporation of new design parts or software code, resulting from an approved engineering change to a product's current approved product configuration documentation, into products already delivered to and accepted by customers.

rework: A procedure applied to a product to eliminate a nonconforming.

software product: The set of computer programs, procedures, and possibly associated documentation and data

software unit: A separately compilable piece of code

specification: A document that explicitly states essential technical attributes/requirements for a product and procedures to determine that the product's performance meets its requirements/attributes.

submitted data: Released data that have been made available to customers.

support equipment: Equipment and computer software required to maintain, test, or operate a product or facility in its intended environment.

unit: One of a quantity of items (products, parts, etc.)

variance; deviation; waiver; departure: A specific written authorization to depart from a particular requirement(s) of a product's current approved configuration documentation for a specific number of units or a specified time period. (A variance differs from an engineering change in that an approved engineering change requires corresponding revision of the product's current approved configuration documentation, whereas a variance does not.).

verification: The act of validating that a requirement has been fulfilled.

version; version identifier: (1) One of several sequentially created configurations of a data product. (2) A supplementary identifier used to distinguish a changed body or set of computer-based data (software) from the previous configuration with the same primary identifier. Version identifiers are usually associated with data (such as files, databases and software) used by, or maintained in, computers.

working data: Data that have not been reviewed or released; any data that are currently controlled solely by the originator including new versions of data that were released, submitted, or approved..

4 Symbols and Abbreviations

The following symbols and abbreviations are used in this standard:

ACDM— Association of Configuration and Data Management
AIA — Aerospace Industries Association
ANSI — American National Standards Institute
ASCII — American Standard Code for Information Interchange
ASME — American Society of Mechanical Engineers
ASQC —American Society of Quality Control
CM — Configuration management
CMAG — Configuration Management Advisory Group
CMM — Capability maturity model
CPIN — Computer program identification number
CSA — Configuration status accounting
DIS — Draft interim standard
DoD — Department of Defense
EDI — Electronic data interchange
EIA — Electronic Industries Alliance
ICD — Interface control document
IEC — International Electrotechnical Commission
IEEE — Institute of Electrical and Electronics Engineers
IGES — International Graphics Exchange Standard
ISO — International Organization for Standardization
JSTD — Joint standard
NDIA — National Defense Industry Association
OTS — Off-the-shelf
PDM — Product data management
SCM — Software configuration management
SEI — Software Engineering Institute
SGML — Standard graphic mark-up language
SOLE — International Society of Logistics
STD — Standard

5 Requirements

The configuration management (CM) process addresses the composition of a product, configuration documentation defining the product, and other data and products that support it. It provides tailored methods and procedures for effectively planning, recording, controlling, and validating product requirements and data that contain the requirements. Where applicable, the CM process also addresses the document representations and associated digital data files that may be accessible on physical media or from on-line information services.

Software configuration management (SCM) is integral to the configuration management process for a program or project that includes software development or maintenance. The same objectives, purpose, and benefits of CM apply to software as to any other product. SCM is also integral to a modern, automated, tool-driven software engineering process.

The CM process may be related to a single product or to an associated collection of products (sometimes referred to as a system or subsystem). In this standard, CM is described in terms of the following interrelated processes that are discussed in the following clauses in terms of their underlying principles:

- Configuration management planning and management
- Configuration identification
- Configuration change management
- Configuration status accounting
- Configuration verification and audit
- Configuration management of digital data.

Typical configuration management activities associated with these CM processes are illustrated in Figure 1.

The principles described in the following clauses are all applicable in a product environment requiring a robust CM approach such as an electronic system, a military weapon, or other item that must be supported over the product life cycle. However all of the principles will not be applied in every phase of the product's life cycle. When configuration management is applied to less complex products and other environments, some of the principles may not apply.

Annex B provides a general methodology for selecting which principles are appropriate.

5.1 Configuration Management Planning and Management

PRINCIPLE 1: Plan CM processes for the context and environment in which they are to be performed and manage in accordance with the planning: assign responsibilities; train personnel; measure performance; and assess measurements/trends to effect process improvements.

CM planning and management over the life cycle of a product are essential to achieve an effective, predictable, and repeatable CM process. CM activities focus on the product and on the customer (see Introduction, page vii) and shape the application of solid, practical procedures that result in cost avoidance and product stability.

CM planning for a specific product builds on the foundation established by an enterprise for similar products. Once the enterprise has implemented effective CM processes, existing plans and procedures normally require relatively minimal fine-tuning to suit specific product, market, or customer needs; or to incorporate improved computer aided tools and methodologies.

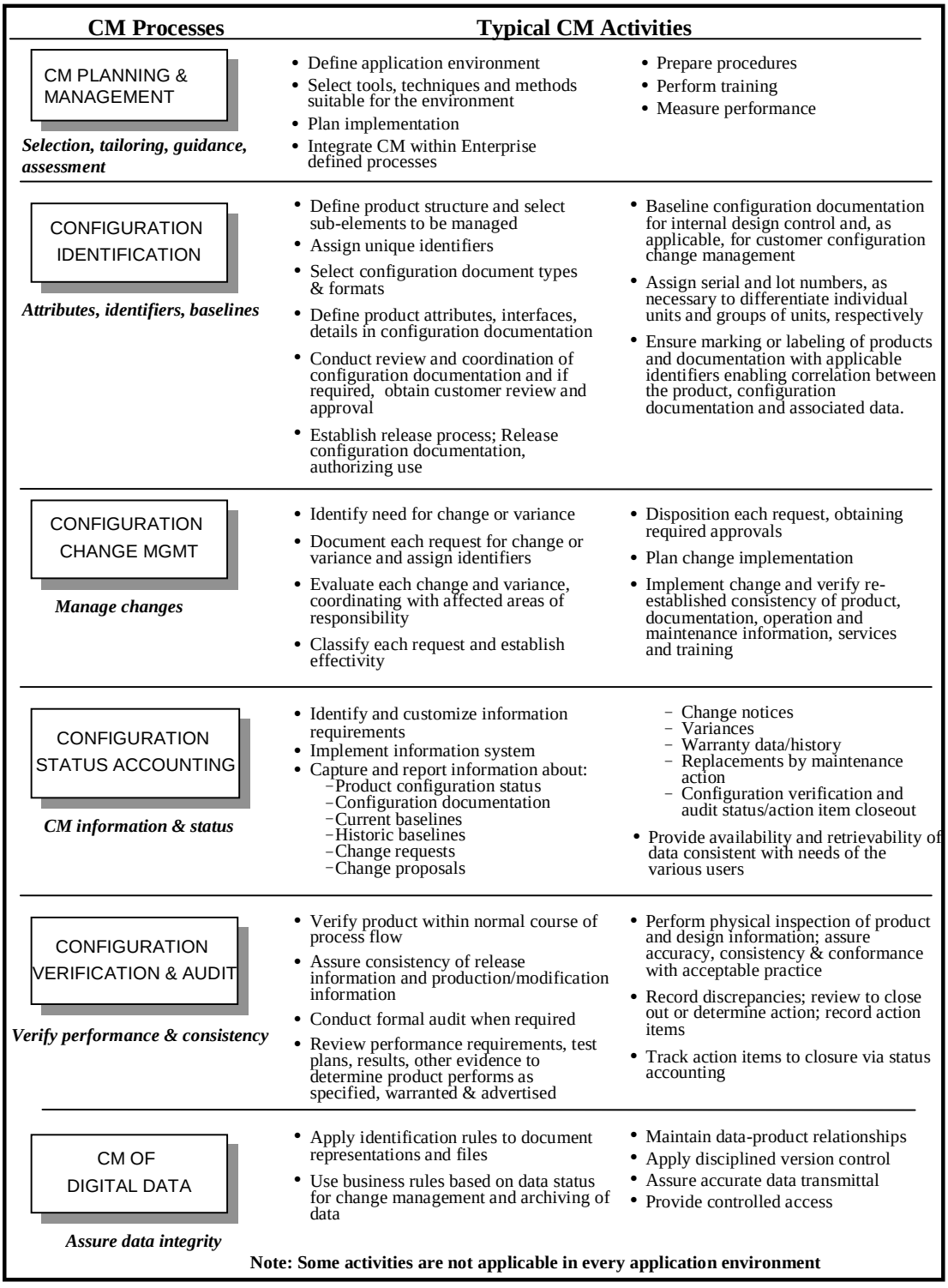

Figure 1 — Typical Configuration Management Activities

Configuration management is often accomplished by distributing various tasks throughout an organization as part of normal line functions or processes. Planning and management ensure that the various distributed tasks are integrated, coordinated, and support the overall objective. Training is provided to ensure that all personnel with CM responsibilities, no matter where they are organizationally situated, understand their roles, responsibilities, and inter-organizational interfaces. The CM process, which may involve many central or distributed functions and activities, is enhanced by the judicious use of measurements that indicate performance trends.

Each application environment warrants a specific selection of CM principles, activities and the tools to implement them. The sophistication of CM implementation can range from elementary paper and pencil methods to the use of highly sophisticated, interactive, shared databases. Table 3 illustrates some of the key attributes of various CM implementations characterized by their salient features. The attributes are additive as you move downward in the table.

Each range of sophistication may represent the best practice for a given application environment. Each element of increased sophistication requires an investment in time and money to establish, but usually results in reduced recurring costs if appropriate to the application environment. In many instances, a highly sophisticated system is totally inappropriate. Evaluation of a CM implementation should focus on its appropriateness to the given environment, the rigor with which the selected processes are applied, and the results in terms of product stability.

Table 3—Attributes of Various Best Practice CM Implementations

Characteristic	Attributes
Planned	• CM planning performed • Planning based on applicable CM principles in this standard • Resources, facilities based on planning • Tools consistent with need/planning
Integrated, consistent	• Coordinated, integrated, and consistent with interacting functions
Rule/work-flow based	• Repeatable, measurable process • Defined by life cycle work flows • CM process integrated with other processes such as program management, engineering, fabrication, manufacturing, operations • Shared information sources (customer, performing activity, subordinate supplier) • Common data directory/dictionary
Flexible, Measured, Transparent	• Tailored by option selection • Process/product metrics used for continuous process adjustment • Integrated with activities such as design to degree that user is largely unaware of CM actions.

The purpose and benefits of CM planning and management include:

- Assurance that the appropriate CM processes, and activities are applied
- Established organizational responsibilities for CM activities
- Necessary resources and facilities
- A basis for continuous improvement
- Enhanced maturity of the enterprise’s process.

5.1.1 Identifying Context and Environment

PRINCIPLE 2: To determine the specific CM value adding functions and levels of emphasis for a particular product, identify the context and environment in which CM is to be implemented.

External constraints (such as product scope, product importance and complexity, production quantity, quality needs, number and size of organizations involved, budgets, and schedules), the application environment (such as defense, energy, automotive, building construction, or consumer products), and the life cycle phases (such as definition, build, distribution, operations, and disposal) are considered in determining the level of CM emphasis to be used for a particular product. Typical questions relating to these topics include:

- Who is/are the customer(s)?
- What are the attributes of the customer's and the end user's environments that need to be addressed by CM? (e.g., Who will support and maintain the product: the customer, the supplier, or a third party?)
- What role will the customer play in decisions about changes?
- What is the current phase of the life cycle and what are the anticipated future phases?
- What is the technical complexity of the product?
- Are there product components that require separate management attention?
- Is the product, or its components, a new design, an existing design, or a modification to an existing design?
- How complex a documentation package is necessary?
- If in the operational phase, what documentation is available, and does it reflect the current product?
- What level of change activity, if any, is anticipated?
- Are customers/users expected to request changes?
- What is the anticipated operational life of the product?
- What, if any, are the required attributes related to disposal; are environmental issues involved?
- What specific information users, maintenance activities, or others will require?

5.1.2 Configuration Management Plan

PRINCIPLE 3: A configuration management plan describes how configuration management is accomplished and how consistency between the product definition, the product's configuration, and the configuration management records is achieved and maintained throughout the applicable phases of the product's life cycle.

A new plan is necessary only when an established CM Plan and well established CM processes for the enterprise are not in place. With an enterprise plan and processes in place, the plan may be supplemented or used as "boilerplate" for new, but similar, products. The minimum information for the new product is a milestone chart and schedule of resources.

Performing CM planning for the product and its environment is essential. The form and format of the planning documentation, and whether it is included with other project planning, are less important.

Because it clearly and concisely describes the process, a well documented CM plan is useful both for training of personnel and for explaining the process to customers, quality assessors and auditors. It is extremely valuable in correlating the application of configuration management to the ISO 9000 series of quality system criteria. The CM plan also facilitates justifying required resources and facilities, including automated tools.

A comprehensive CM Plan that reflects efficient application of configuration management principles and practices to the identified context and environment would normally include the following topics:

- General product definition and scope;
- Description of CM activities and procedures for each major CM function, e.g.,
  - Configuration planning and management
  - Configuration identification

- Configuration change management
- Configuration status accounting
- Configuration verification and audit
- Configuration management of digital data

— Organization, roles, responsibilities, and resources
— Definitions of terms
— Programmatic and organizational interfaces
— Deliverables, milestones, and schedules
— Subcontract flow down.

5.1.3 Implementation Procedures

PRINCIPLE 4: Prepare procedures to define how each configuration management process will be accomplished.

Procedures provide the detailed "how-to" steps required to implement CM planning, for both manual and automated processes. Procedure styles are usually organization-unique and there is no one "right way" to document a procedure. In general, procedures should be unambiguous and succinct and be written at a level of detail that is commensurate with the intended user skill level. As with planning, procedures may be applied to a range of products or tailored by supplement for individual products. Procedures are evaluated to ensure that they are consistent with CM planning.

5.1.4 Training

PRINCIPLE 5: Conduct training so that all responsible individuals understand their roles and responsibilities and the procedures for implementing configuration management processes.

Training provides the context for understanding CM objectives, techniques, and functional processes for the specific product; in particular it makes organizational interfaces and interactions clear. Best practice training approaches address both performance of CM functions and cross-training with other disciplines.

5.1.5 Performance Measurement

PRINCIPLE 6: Assess the effectiveness of CM plan implementation and performance of the configuration management discipline with defined metrics (performance indicators).

The data derived from metrics are used to understand problems and inefficiencies in products and processes, to assess the extent of those problems and inefficiencies, and to provide insight in making necessary corrections and improvements. The purpose of some CM metrics is to measure project/program performance. CM processes and procedures are reviewed and revised periodically, using metrics data. Metrics are selected and adjusted as appropriate for the program environment and product life cycle phase. Typical CM metrics include:

— Number of configuration documentation releases (Scheduled/Actual)
— Number of engineering changes (by product, by classification, by phase, by time period)
— Average engineering change cycle time (by product, by classification, by major process step including customer review and approval where applicable)
— Average revisions per engineering change (in-house, after submittal to customer)
— Number of changes (by reason for change)
— Number of variances (by product, by type, by phase, by time period, per delivered unit)
— Number of action items per configuration audit (categorized by magnitude)
— Average number of unincorporated changes (attachments) per engineering drawing.

5.1.6 Supplier Configuration Management

PRINCIPLE 7: Performing configuration management includes responsibility for the configuration management performance of subordinate activities (e.g., subcontractors, suppliers).

Configuration management requirements appropriate to the product being acquired are passed down to lower level suppliers, typically via purchase order or other subcontract agreement instrument. Tailoring of requirements for subcontractors is a major CM planning activity. The performing activity takes on the role of customer (buyer) to the supplier. Suppliers are monitored via data reviews, configuration change management, design reviews, product test results, configuration audits, and CM surveillance reviews, as appropriate.

- Data reviews typically include assessment of supplier plans, procedures and configuration documentation
- Configuration change management typically includes review of proposed changes to buyer approved or imposed configuration documentation
- Design reviews generally assess the suppliers progress and provide a level of confidence that the product, when developed, will meet its specified attributes
- Product test results are positive or negative indicators that required attributes will, or will not, be satisfied
- Configuration audits (see 5.5.2) verify that the required attributes have been achieved and the design of the product has been accurately documented
- CM surveillance reviews (see 5.5.3) verify continuing application of supplier CM processes.

The level of configuration change management exercised by the buyer ranges from none to total depending on the nature of the product and the conditions of purchase. For an off-the-shelf product, the buyer generally has no control over the product attributes, but can choose not to buy the product. For a product purchased using a buyer-prepared control drawing, the buyer typically exercises configuration control authority over the specified form, fit, and function attributes. For a product developed to the buyer's specifications, the buyer normally exercises configuration control authority over the product's requirement attributes. The buyer may also exercise control over the product design if more rigorous CM has been flowed down to the supplier.

5.2 Configuration Identification

PRINCIPLE 8: Configuration identification is the basis from which the configuration of products are defined and verified; products and documents are labeled; changes are managed; and accountability is maintained.

The purpose and benefits of configuration identification include:

- Selecting products based on functionality, verifiability, supportability, complexity, risk and management activity
- Determining the structure (hierarchy) of a product and the organization and relationships of its configuration documentation and other product information
- Documenting the performance, interface, and other attributes of a product
- Determining the appropriate level of identification marking of product and documentation
- Providing unique identity to a product or to a component part of a product
- Providing unique identity to the technical documents describing a product
- Modifying identification of product and documents to reflect incorporation of major changes
- Maintaining release control of documents for baseline management
- Enabling a user or a service person to distinguish between product versions
- Enabling a user or a service person to correlate a product to related user or maintenance instructions
- Facilitating management of information including that in digital format (See 5.6.)
- Correlating individual product units to warranties and service life obligations
- Enabling correlation of document revision level to product version/configuration
- Providing a reference point for defining changes and corrective actions.

These benefits are realized only if there is consistency between a product itself and the information about the product. To achieve consistency of the product attributes and the information about them, the performance, functional and physical attributes are first defined in configuration documentation. These attributes must then be achieved in a product. Both the product and its documentation must also be verified for consistency. Consistency is maintained throughout the product life cycle by identifying and evaluating the impact of all proposed changes through the configuration change management process (see 5.3) including the verification that the product and all associated product information have been updated and continue to be consistent. To bring this underlying conceptual view into reality requires the disciplined application of configuration identification principles expressed in the following subclauses.

5.2.1 Product Information

PRINCIPLE 9: Configuration documentation defines the functional, performance, and physical attributes of a product. Other product information is derived from configuration documentation.

Figure 2 illustrates the generic categories of product information and how they relate. The two major sets of information are configuration documentation and operational information. Only product information necessary for control and correlation of product attributes is included in the product's configuration documentation.

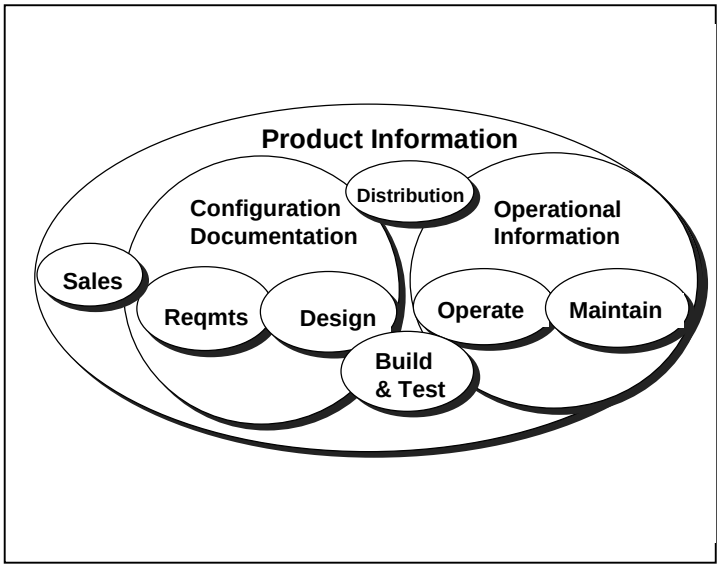

Figure 2— Composition of Product Information

Configuration documentation, the product's defining documentation, defines a product's performance, functional, and physical attributes. It consists of:

- Requirements documentation defining the performance (capabilities) and functional boundaries of the product and its physical and functional interfaces. Definition of boundaries and interfaces reflect the end-use of the product to satisfy performance requirements and include product test, product servicing and support considerations. The major components of a complex product are defined by distributing (allocating) product attributes to the components. Requirements documents are typically in the form of specifications and interface documents (see 5.2.7). Statements that define product requirement attributes include a parameter, a value for the parameter, a tolerance on the value, and a level of confidence (e.g., verification method).
- Design documentation defining the physical and functional attributes necessary to completely define the design details of the product implementing the specified requirements. Examples of design documentation are engineering drawings and parts lists (see ASME Y14.24M and Y14.34M.), and software design documents.

Operational information, as well as build and test information, is derived from the configuration documentation. It includes information necessary to use and operate the product (e.g., operating procedures) and other documentation necessary to service and maintain the product. Statements that describe product use are supported by, and consistent with, the statements defining the product's attributes.

Each enterprise's internal operating standards are based upon the intended end use(s) for their products throughout the product life cycle and delineate the appropriate configuration documentation and other product information to be provided. The entire set of selected product information is organized (i.e., grouped, categorized) to facilitate effective and efficient retrieval and maintenance. The product information is stated such that it is complete and usable whenever and wherever it is needed throughout the product's life.

5.2.2 Product Structure

PRINCIPLE 10: The product composition (i.e., relationship and quantity of parts that comprise the product) is determinable from its configuration documentation.

A product structure is a common technique for organizing the composition of a product. It is a representation of the breakdown hierarchy (i.e., product tree/pyramid) of a complex product, from the top down to the lowest level. Each level references associated configuration documentation (e.g., engineering drawings, bills of material, specifications, software requirements and design documents, and processes/procedures). The product structure shows the top-down relationships among the various parts that make up the product and the quantity of each. The product structure is complete when all parts and configuration documentation are included. The product structure is very useful in visualizing the relationships, in determining the level(s) at which to apply CM, and in evaluating the impact(s) of proposed changes to the product.

5.2.3 Product Identifiers

PRINCIPLE 11: All products are assigned unique identifiers so that one product can be distinguished from other products; one configuration of a product can be distinguished from another; the source of a product can be determined; and the correct product information can be retrieved.

There are basically two levels of product and document identifiers: (1) the identifier level that is visible to an external customer or user of the product during the operational period, and (2) the level of identifiers of the product and its component parts that is internally necessary for the developing and manufacturing activity to manage the configuration during the definition, build, and operation phases. The customer needs identifiers of the product and all parts that can be ordered as spare, replacement, or accessory parts, and of any parts that have specific service life attributes (such as shelf life or limited period of operation) or specific warranties. The customer also needs unique identifiers for documents that reflect the configuration of the product to the same level. The developer needs all of that information plus the complete set of identifiers for all parts comprising the product, and their associated documents.

Conventions for product identifiers vary by industry. However, the most commonly used identifier for a part, consists of a part number and a code representing the design activity/manufacturer/supplier. These elements are necessary for automated systems. A widely accepted practice is for the design document (engineering drawing) number to be the same as, or included within, the part number. The code and/or the supplier's name and logo may appear on the part marking (e.g., nameplate, stamp, etch, silk-screen, etc.). Both normally appear in design documents.

A product and each of its component parts are assigned unique identifiers, as follows:

a. Unique identifiers (design organization, and part numbers or in some cases names, such as “nylon”) are assigned for all new products down to the lowest level in each branch of the product structure for which that activity has development responsibility.

b. Already developed products used as components of the product retain their existing identifiers (design organization and part numbers), unless modified to the extent that interchangeability is affected.

c. Parts of the product developed by, or acquired from, suppliers retain the unique identifiers assigned by the supplier. If the acquiring activity has special requirements, it may choose to provide a unique identifier in addition to the supplier’s identifier, to correlate the part to its specification document. The additional identifier distinguishes the part meeting special requirements, such as additional test or parts screening, from like parts that do not.

d. When a change is applied to a product or part of a product, its descriptive configuration documentation (engineering drawing, product model) is updated to reflect the change. The unique identifier assigned to a product, or part, and the marking on the part itself, are changed to distinguish one configuration of the product from another, when:

- The new or updated part is no longer interchangeable functionally or physically with the previously delivered part,
- The new or updated part is no longer interchangeable functionally or physically with previous undelivered parts that will remain in a different configuration,
- The new part requires new or revised testing, maintenance, repair, training, operating procedures, equipment, or software,
- The part is altered, selected or is a source controlled item (ASME Y14.24M), or
- The updated part has different application, use, safety, or other restrictions.

e. When a repair part within a product is changed so that it is no longer interchangeable with its previous version, it is assigned a new identifier. The general rule is to re-identify the next higher assembly and all subsequent higher assemblies up to and including the level at which interchangeability is re-established, or an identifiable end product (against which configuration changes are tracked) is reached.

Unique software identifiers are assigned for each software product and for the software products used in the software engineering and test environments. The software identifier includes the version of the entity (typically, a dash or decimal point is used to separate the version number from the root software number). Software units are assigned a name or number that is unique within the software product. The marking and labeling of software is commonly accomplished as follows:

- The software identifier is embedded in the source and executable code header
- Each software medium is labeled with the supplier’s code/name identification and the software identifiers of the software products it contains. If it is impracticable to include all software identifiers, the medium is labeled with a reference to an embedded list (such as a read.me file) containing the identifiers.
- Wherever practicable, electronically reprogrammable hardware with resident software, is labeled with the software identification number of the resident software in addition to the hardware part number.[1]

Many products have catalog numbers assigned to them by distributors; and parts stocked by the military have National Stock Numbers. Software developed for the military (Air Force) may also be assigned a catalog number known as a Computer Program Identification Number (CPIN).

[1] Firmware devices with resident software that is altered as a normal course of operation is treated much like a general purpose computer in that the configuration of the resident software is not labeled on the device but is maintained in a separate record.

5.2.3.1 Identifying Individual Units of a Product

PRINCIPLE 12: Individual units of a product are assigned a unique product unit identifier when there is a need to distinguish one unit of the product from another unit of the product.

PRINCIPLE 13: When a product is modified, it retains its original product unit identifier even though its part identifying number is altered to reflect a new configuration.

The most widely accepted method of identifying an individual unit within a series of like units is by assigning a unique serial number to each unit and applying the manufacturer's identification (organization code) and that serial number to the unit. Some illustrative examples of when product units should be serialized are:

- When products with the same basic identifying number can be provided with customer options (e.g., paint color), serialization (such as the vehicle identification number on a car) provides the means to direct the customization and to maintain appropriate records.
- When products have warranties, the serial number is used to correlate information concerning dates of manufacture and sale and the warranty period for each individual unit.
- Whenever each unit must be subject to individual functional and performance testing or screening, such as acceptance testing, serialization provides the means to correlate each unit to its test records.

When products are modified, they retain their original serial number, even though their part identifying number is altered to reflect a new configuration. This practice enables the history of each specific part to be maintained. In order to achieve this level of integrity, it is essential that the serialization take place using a nonchanging identifier as a base. The source of this non-changing base is dependent upon the internal procedures of the enterprise assigning the serial numbers. The complete part identifying number is usually not suitable because it must change if the configuration of the part is changed. Alternative choices are the root portion of the engineering drawing number (exclusive of dash numbers), a model number, or some other specially assigned designator indicative of the product line.

5.2.3.2 Identifying Groups of Units of a Product

PRINCIPLE 14: A series of like units of a product is assigned a unique product group identifier when it is unnecessary or impracticable to identify individual units but nonetheless necessary to correlate units to a process, date, event, or test.

Typically, the manufacturer's identification (organization code) and batch or lot number identify groups of units. The batch or lot number distinguishes units to a lesser degree than a serial number. It enables an individual unit to be correlated with the test or process records for a quantity of units rather than an individual unit. In the event of a latent defect in the product, the lot or batch number enables the problem to be isolated to the number of units in a suspect lot or group of lots. The lot or batch identifier can simply be a sequential number or a code with some significance. For example, in high volume production situations, the lot identifier might be a date code identifying the units produced on a given date.

Other batch or group designators are assigned for convenience in referring to a collection of products. These discretionary indicators may or may not be marked on the products themselves. An example of a discretionary designator is a “block identifier” used to refer to a group of units of similar or identical configuration.

Like serial numbers, it is essential that the lot or batch numbering takes place using a nonchanging identifier as a base. When a product is modified, it retains its original product group identifier even though its part identifying number is altered to reflect a new configuration, unless the modification process involves a new grouping. Unlike serial numbers, a lot number assigned to an item may be changed if two or more lots, or parts of lots, are reworked as a new amalgamated lot.

5.2.4 Document Identification

PRINCIPLE 15: All documents reflecting product performance, functional, or physical requirements and other product information are uniquely identified so that they can be correctly associated with the applicable configuration of the product.

All documents, including information related to the product (such as reports, letters, and memoranda), are uniquely identified so that they may be referred to precisely and retrieved when necessary. Configuration documents and other product related documentation cite, or are linked (in a database or other cross-reference) to the product identifier and revision with which they associate. Methods of identifying documents can vary considerably; however, unique identification includes at least originator/design organization, a unique identifier assigned by the originator/design organization (for example a number, alphanumeric identifier, or title/subject) and a revision indicator (number, letter, or date). The document identifier may include the type of document and/or a customer's contract or purchase order number.

5.2.5 Baselines

PRINCIPLE 16: A baseline identifies an agreed-to description of the attributes of a product at a point in time and provides a known configuration to which changes are addressed.

Baselines are the fundamental basis of configuration management. Baselines provide affected parties an assurance of the stability and consistency of information needed for their subsequent activities. They provide a common communication of product definition and also a vehicle permitting transfer of authority over all or a portion of a product's definition. The release of a requirements definition document before beginning a product's design represents a baseline; the released detailed design, as production begins, represents another. The operational characteristics of an industrial plant or facility, against which performance deterioration is measured, could also represent a baseline. Conceptually, the release of each document establishes a baseline for that portion of the overall product definition and the release of each revision represents an update to the baseline.

5.2.5.1 Establishing Baselines

PRINCIPLE 17: Baselines are established by agreeing to the stated definition of a product's attributes.

The degree of formality and the way in which baselines are used can vary widely. A baseline may be quite simple and under internal control of a small group, or be formal, detailed and under external contractual control. Any agreed point of departure can represent a baseline.

In creating a new product, the definition of desired performance attributes will normally be established as the initial baseline. To aid the management of complex developments, intermediate baselines may be used to decompose an overall design approach and allocate requirements to various subdivisions. These initial baselines are later supplemented with a detailed description (e.g., a set of drawings) of the resulting product. The adoption of incremental baselines usually occurs at project or contract milestone such as a design review or the decision to enter production.

Additional detailed baselines are established when the attributes of the sub-divisions stabilize, or when control of a greater level of design detail becomes necessary. These points usually signify changes in activities or commitments. Examples of such events may include the following:

- A facility becomes operational
- Advertising of a product's features begins
- Production begins
- Customer approves design concept
- A major change or addition is approved.

These events typically represent a need for stability by a larger or different group and therefore may also signify a transfer of change authority. The documentation defining a product's baselines must be mutually consistent and compatible. Each more detailed baseline level must be traceable to, and be a detailed extension of, its predecessor(s). The authority to approve changes to a baseline is transferred when an expectation of stability is needed by others. For example,

- When a drawing is completed and released, a designer relinquishes control (i.e., can no longer make changes unilaterally) so that others can be assured of stability while creating related dependent drawings.
- The authority for approval of changes to a product baseline may be transferred to program management when significant investment in production tooling and facilities is committed
- A customer may need to review changes to a product's detailed design to ensure compatibility with other products, or with the same product acquired from other suppliers.

Before any document or data set is considered part of a baseline, it must be reviewed to ensure that the document is complete, valid and suitable for use. A release system/process is employed to validate the document and file integrity.

Configuration change management, described in 5.3, is essentially a process for managing baselines. Baselines are tools to match the need for consistency with the authority to approve changes. A configuration management system or plan must recognize:

- What baselines are to be established
- When and how they will be defined
- The process for assuring document and file integrity
- The authority to approve baseline changes
- If and when change authority will transfer
- The process by which proposed changes will be dispositioned.

5.2.5.2 Types of Baselines

PRINCIPLE 18: The configuration of any product, or any document, plus the approved changes to be incorporated is the current baseline.

The current approved configuration defines the baseline and becomes the basis for the next change. Baselines are normally established within the enterprise developing and manufacturing a product. The baselines frequently become formalized at the interface between customer and supplier, depending on the practices of the industry and the contractual involvement of the customer in the product change process. (See 5.3)

Identified baselines are often categorized by the degree of detail defined (e.g., requirements, design release, and product configuration baselines) or by the placement of authority for change approval (e.g., external customer baseline, program baseline, engineering baseline). To provide a consistent reference when describing the principles of configuration management, the baseline terminology and protocol, described below and illustrated in Figure 3 are used in this standard:

a. Requirements Baseline The customer baseline established for a product represents firm performance requirements. The requirements are normally documented in a specification. The customer with whom the requirements baseline is established may be one of the following:

- An external customer, i.e., a buyer for whom the product is to be developed and produced
- A management level or function, such as a marketing department, that represents many customers.

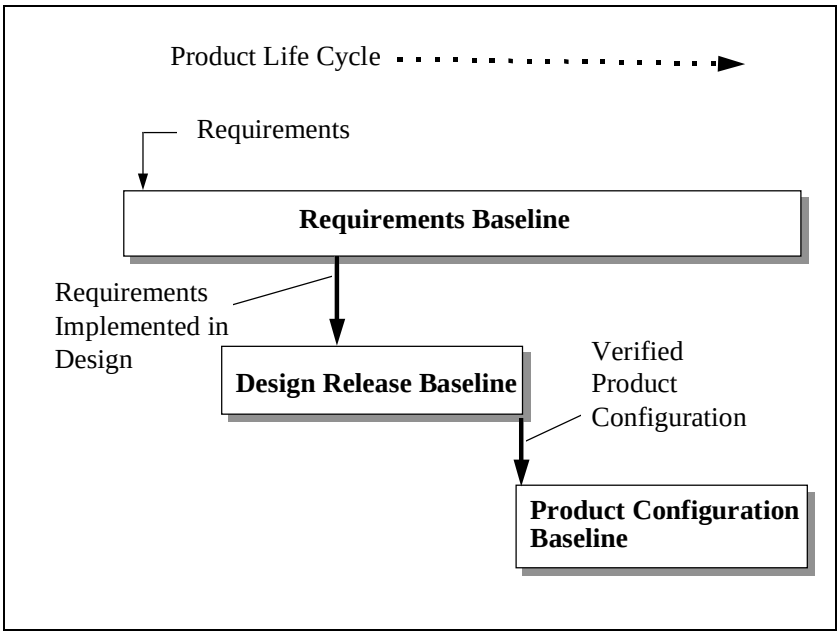

Figure 3 — Configuration Baselines

For new development, the requirements baseline is often a product of the conception phase. The requirements baseline documents the common understanding of what the product is expected to do (its functional and performance requirements). It defines the capabilities the customer expects to receive from the product. The requirements baseline provides a basis for commencing the definition phase. For already developed products, a requirements baseline could be as simple as a purchase order stipulating the product by reference to a catalog describing the product's performance and physical attributes.

Subordinate requirements baselines are established for major elements of complex products where it is determined that separate specification of performance requirements for selected elements is appropriate. These subordinate "allocated requirements," or "allocated" baselines, as they are frequently referred to, often establish requirements for suppliers of components of the complex product. In that case, the performing activity for the complex product assumes the role of customer to the supplier of the component. Requirements allocated to the supplier must be consistent with the requirements baseline for the complex product. Allocated baselines are of concern to the customer for the complex product only if contractually stipulated, or when allocated baselines are employed to incrementally verify performance of the complex product.

b. Design Release Baseline Design information is normally created, reviewed, and incrementally released over a period of time during the definition phase of a product's life cycle. Once design information is released, it becomes part of the design release baseline controlled by the developing activity. To establish and maintain a design release baseline requires a process for initial release of design information and for release of approved engineering changes to the design information.

Changes to released documentation must be evaluated, as described in 5.3, for the effect they have on requirements and on users of the released information who may have begun implementation actions such as ordering material, designing tooling, planning manufacturing, coding software or producing parts. The database of information resulting from these processes, including the design information itself, should be capable of providing current and historical information about each product. (See 5.4.)

c. Product Configuration Baseline A baseline is normally established for a product at the end of the definition phase, when the design of the product is “frozen” for the Build phase. The establishment of the product configuration baseline normally represents a change in the management level involved in configuration control. It is considered a customer baseline where the customer, as previously stated in table 2, could be a change authority internal to the enterprise or an external customer with a contractual relationship to the enterprise. This “product” baseline is highly significant because it normally becomes the basis for commitments for such things as production resources, materials, sales literature, training devices and material, and operating and service instructions. It is defined by the complete set of current product configuration documentation constituting the current design release baseline. Once the product configuration is fixed, it is changed only through appropriate configuration change management action (See 5.3). Typically this baseline is established after verification that the product is in compliance with its requirements baseline. (See 5.5).

When a customer purchases an already developed product, the current product configuration baseline defines the configuration of the product to be supplied, unless a configuration change is approved. The unit(s) of product delivered in the current product baseline configuration may differ from the configuration of earlier units, delivered to a then-current baseline configuration.

d. Additional Operation and Disposal Phase Baselines Supplemental baselines that are either location-oriented views (extracts) of the product configuration baseline, or that add supplemental information of concern to the product operation, support and maintenance are typically identified and controlled at operational sites. (See 5.4.1.e.) In some instances, a disposal configuration baseline may be necessary.

5.2.6 Product Identification Recovery

PRINCIPLE 19: Maintaining product information is important because time consuming and expensive recovery may be necessary if records of operational units of a product do not match the actual units (as reported by maintenance activities) or such records do not exist.

Specific manpower-intensive actions will be necessary to recover the configuration identification, or establish workable configuration identification for an old product with no adequate documentation. Each unit must be inspected to determine its configuration. If performance and design information are not available, it may be necessary to determine product attributes essential for operation and repair decisions through the expensive process of reverse engineering. The product identification recovery process consists of the following steps:

a. Understand and scope the problem The product's attributes and information about the product that are essential to be recovered are identified. To avoid unnecessary or excessive recovery activities, recovery priorities are defined based on the need and use of these components, attributes, and information.

b. Recover attributes Product disassembly and physical examination are performed if necessary to determine the physical attributes of a product's components. If necessary, product testing is performed to determine functional attributes. This data is retained as information about the product.

c. Recover information If necessary, searches are performed to identify sources of information about the product. The available information is evaluated and validated as to applicability and currency. Required information that is not recovered through searches is re-created by performing analysis, re-enactment, testing, or interviewing (to capture best recollections).

d. Evaluate impact on consistency For all recovered attributes and information, the impact on related attributes and information is evaluated to ensure consistency. Where information and attributes are not internally consistent or consistent with each other, discrepancies are identified and resolved.

e. Merge recovered information The recovered, validated, consistent information is integrated into the established information records and systems.

5.2.7 Interface Control

PRINCIPLE 20: For product interfaces external to the enterprise, establish an interface agreement and a mutually agreed to documentation of common attributes.

Product attributes include defined interfaces with products that are developed, produced, and supplied by organizations outside the enterprise. External interfaces are documented in a product's configuration documentation. To document and control the interface, there must be a relationship between the interfacing organizations.

If the relationship is a buyer-seller relationship, the interface definition is included as part of the purchase agreement (e.g., by reference to a defined catalog item, or by use of a control drawing). If there is no direct relationship, an interface agreement is established between the developing enterprises. It delineates procedures for defining and maintaining the common interface. The procedures (for defining complex interfaces and coordinating proposed changes to them) may employ a joint interface control working group. A mutually agreed upon interface definition (including performance, functional, and physical attributes) is typically detailed in an interface document or drawing.

5.3 Configuration Change Management

PRINCIPLE 21: Changes to a product are accomplished using a systematic, measurable change process.

Configuration change management is a process for managing product configuration changes and variances. The purpose and benefits of the change management process include the following:

- Enable change decisions to be based on knowledge of complete change impact
- Limit changes to those which are necessary or offer significant benefit
- Facilitate evaluation of cost, savings, and trade-offs
- Ensure customer interests are considered
- Provide orderly communication of change information
- Preserve configuration control at product interfaces
- Maintain and control a current configuration baseline
- Maintain consistency between product and documentation
- Document and limit variances
- Facilitate continued supportability of the product after change.

Regardless of the type of product or phase of its life cycle, a change to a product is accomplished using a systematic, measurable change process. The process includes identifying the need for a change; documenting change impact; evaluating and coordinating the proposed change (including approval/disapproval); incorporating the approved change in the product and its related configuration documents; and verifying consistency of the product definition. In addition, the process also encompasses the identification, documentation, approval, and implementation of variances from configuration baseline requirements.

The initial baseline for change management consists of the configuration documentation defining the requirements that the performing activity (i.e., the product developer or product supplier) has agreed to meet. This agreement may be with an external customer in the case of a product to be designed specifically for that customer, or with an internal customer (such as a marketing department) in the case of a product produced for the commercial market. Any change which would cause the product performance or interfaces to differ from the requirements baseline; or which would impact cost (price), delivery schedule, warranties or guarantees; should be reviewed with, and approved by the customer. The "approved" change then becomes part of the current baseline. As the product progresses through the later phases of its

life cycle (e.g., build, operation), the customer's requirements continue to be a primary baseline for customer configuration change management.

The design release baseline for change management consists of the detail design documentation used to manufacture, construct, build, or code the product. The design release baseline incrementally defines the design that is developed to meet the customer's performance requirements. Internal configuration control measures are applied to the configuration documentation constituting the design release baseline for a product, during the definition phase.

The product configuration baseline for change management consists of the detailed design documentation from the design release baseline which defines the product configuration that has been proven to meet the requirements for the product. The product configuration is considered a mature configuration. Internal configuration control measures are also applied to the product configuration baseline during the build, distribution and operation phases. However, elements of the product configuration may be under developer/supplier configuration control, under customer configuration control, or distributed as defined by customer/supplier agreement (contract, purchase order, etc.).

Changes to the current requirements, design release, or product configuration baselines may result from discovery of a problem, a suggestion for product improvement or enhancement, a customer request, or a condition dictated by the marketplace or by public law. The generic change management process model shown in Figure 4, which is valid for either internal change management or management of changes to products under customer configuration control, consists of the following processes:

a. Change identification process The requested change is conceptually visualized to determine one or more approaches to accomplishing it. The process steps are applied to: describe the change; make preliminary assessments of the effect of making (or not making) the change; choose a proposed point of incorporation (effectivity); and classify the change as to required levels of processing and approval authority (including customer approval or concurrence, when required). (See 5.3.1.)

b. Evaluation and coordination process Evaluation and coordination includes, as applicable:
- The cost or savings to both the supplier and customer(s);
- The current work scope and schedules affected;
- The design, development and test effort involved;
- The product documentation revision or replacement required;
- The effect on warranty and other contractual considerations;
- The effect on delivered product (e.g., does it require recall, retrofit, replacement);
- The effect on spare/replacement parts;
- The effect on environmental considerations
- The modification required to be made to manufacturing, assembly, installation, test, operating or maintenance instructions or to training devices and training materials; and
- any effects on the performance or on the functional and physical interfaces with other products.

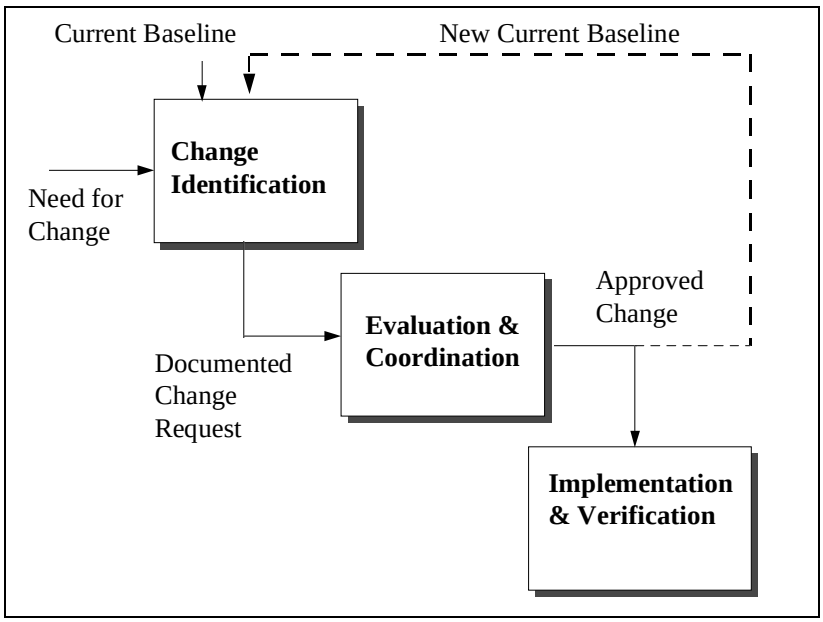

Figure 4 — Change Management Process Model

If changes to customer provided or approved requirements are involved, the decision to proceed with a proposed change may require proposal of the change to the customer and customer approval. (See 5.3.2.)

c. Incorporation and verification process An approved change is planned and scheduled for implementation in all documents, facilities, hardware, and software impacted by the change. The implementation is carried out in accordance with the plan. Change accomplishment in all impacted areas is monitored and verified to maintain consistency between each unit of the product and its applicable configuration and supporting documentation. (See 5.3.3.)

5.3.1 Change Identification

PRINCIPLE 22: Each change is uniquely identified.

The objectives of change identification are to provide a description of the requested change and its impact sufficient for evaluation, determine the appropriate level of approval, choose the appropriate format (information content) for describing the change, and provide a unique identifier for the change request.

Figure 5 illustrates the key elements of this process and the questions to be answered to prepare the change request for evaluation. For simplicity, the originator of a change request is referred to as the "change requester." The change requester may be a member of the product development or production team, an operator or user of the product, the product test activity, or a customer in any other capacity. The activity or individual with technical responsibility for the product (who could also be the change requester) is referred to as the "change sponsor."

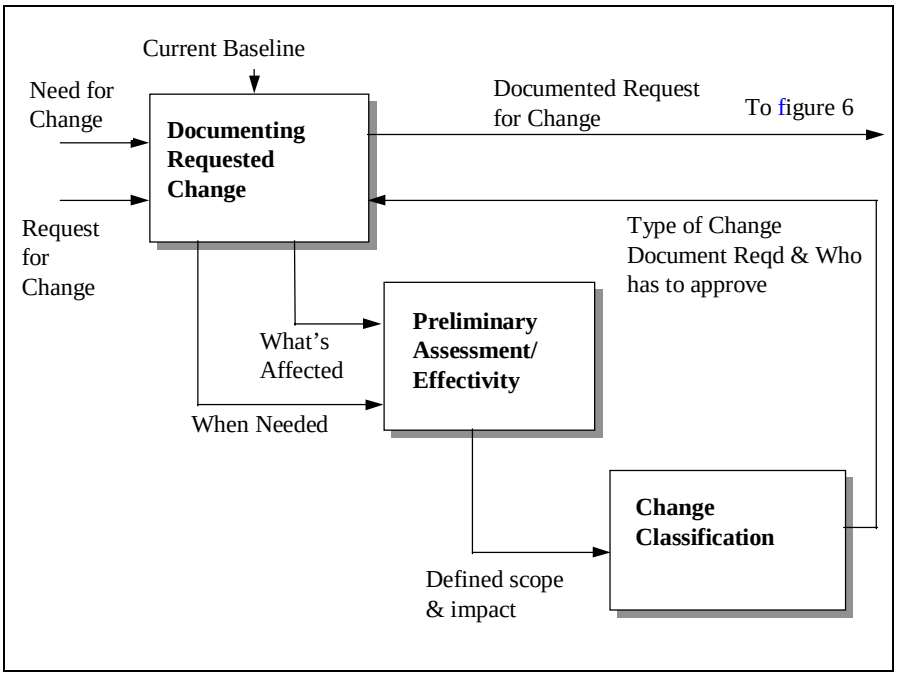

Figure 5 — Change Identification Process Model

5.3.1.1 Requesting Changes

PRINCIPLE 23: Changes represent opportunities for improvement.

Changes are initiated for a variety of reasons, such as:

- To provide new capabilities desired by a customer(s)
- To enhance product support
- To insert new technology
- To effect product improvements
- To correct product defects or deficiencies
- To correct problems and prevent recurrence
- To eliminate safety hazard conditions
- To implement preplanned product improvement
- To reduce production costs/ improve production efficiency
- To prevent schedule slippage.

Changes may be required as a result of problems reported during test, operations and service activities. All reported problems are documented, dispositioned, and tracked to closure. Upon review, the problem/change request is entered into the configuration change management process if it requires a change to baselined configuration documentation.

Requests for change are documented either by the requester directly or by the sponsor after communication with the requester. The originator and sponsor make preliminary judgments to identify the proper change authority and to determine the processing method and document format that are most appropriate. These judgments concern:

- The need for the requested change
- The basic scope and description of the requested change
- The definition of its impacts
- The desired effectivity
- Its urgency and importance.

While this process can be very simple and intuitive, it often requires careful weighing of alternatives and cost benefit trade-offs.

5.3.1.2 Classifying Changes

PRINCIPLE 24: Classify requested changes to aid in determining the appropriate levels of review and approval.

A compilation of "best practice" factors (engineering change characteristics) to be considered in determining change classification are provided in Table 4. Some of these factors may not be applicable for some product types or may require tailoring for a particular application. The customer could be an internal customer, such as a marketing or program management department or other corporate function with the responsibility of specifying the product requirements. The classifications, defined as follows, differentiate between major and minor changes. Criteria are provided for each classification to determine if external customer (buyer) review in addition to internal review is required.

a. Major A change classified as major is a change to the requirements of baselined configuration documentation (requirements, design release or product configuration baselines) that has significant impact. It requires coordination and review by all affected functional groups or product development teams and approval by a designated approval authority (usually an individual who can authorize the resources needed for change implementation.)

A major change requires external customer approval[2] only if:

- It impacts external customer (buyer) requirements
- It impacts any factors in Table 4 that relate to external customer controlled activities (e.g., the external customer has to replace an inventory of spare or repair parts or must take other implementing action).

b. Minor A minor change corrects or modifies configuration documentation (released design information), processes or parts but does not impact any characteristics in Table 4 that would cause it to be classified as major (e.g., product interchangeability). Minor changes do not impact customer requirements (i.e., should be invisible to the customer) and therefore require only internal approval at an appropriate level commensurate with their impact.

A minor change will require external customer involvement[3] under the following circumstances:

- The product configuration baseline is established, and
- The external customer is concerned with (or controls) the product's detail design in addition to its performance and interface attributes, and has imposed his own management procedures on the detailed design; and
- The contractual agreement stipulates either that the external customer or his representative must review the change classification, or must approve minor changes. (This provision normally applies when the external customer is the design activity and the product is being built to external customer documentation.)

[2] In DoD application practice, a major change that requires government approval is designated as a Class I change.
[3] In DoD application practice. a minor change that requires government review is designated as a Class II change.

5.3.1.3 Documenting Requests for Changes

PRINCIPLE 25: Change requests must be clearly documented.

To adequately evaluate a request for change, the change request must be clearly documented. It is important to accurately describe even minor changes so that an audit trail can be constructed in the event that there are unanticipated consequences or unexpected product failures. Saving the cost of the research involved in one such incident by having accurate accessible records may be sufficient to fully offset diligent, disciplined change processing.

Table 4—Classification of Engineering Changes

Characteristics of Engineering Change	Classification	
Factors	**Major**	**Minor**
a. Affects approved, baseline specification requirement[1] (i.e., requirement would be out of the specification limits)	●	
b. Affects one or more of the following, after product baseline: — Products furnished by a customer — Safety — Compatibility with interfacing products (including such products as test equipment, support equipment and associated software) — Delivered operation or servicing instructions[2] — Preset adjustments[3] — Interchangeability or substitutability of replaceable products, assemblies, or components — Change to a previously nonselected supplier, where supplier selection is specified — User skills or physical attributes — Operator or maintenance training	●	
c. Requires retrofit of delivered products, e.g., by product recall, modification kit installation, attrition (replacement during maintenance by modified spares), etc.	●	
d. Affects cost/price to customer(s) (including incentives and fees), guarantees, warranties, contracted deliveries or milestones; and is an engineering change that does not impact factors a. through c.	●	
e. Affects configuration documentation (released design information), product or processes but does not affect factors in a. through d., above		●

1. Such as performance, reliability, maintainability, weight, balance, moment of inertia, interface characteristics, electromagnetic characteristics, etc.
2. For which there is no planned and funded update requirement, such as for periodic or continual maintenance of the instructions
3. To the extent that product identification should be changed

Documentation of major changes includes the following information that is required to make an informed evaluation of the change and to clearly define the change:

— Unique change identifier
— Originator organization and responsible individual
— Class of change
— Product(s), major components, interfacing products affected
— Contract and configuration documents affected
— Scope and description of change

- Effects on specified performance, operation, maintenance, servicing, operation and maintenance training, spare and repair parts, support and test equipment, catalogs, marketing literature, etc.
- Reason and justification for the change; consequences of not doing the change
- Priority/urgency of the change
- Proposed change effectivity
- Requested approval date
- Change implementation and delivery schedules
- Estimated cost increase or savings
- Alternatives.

Minor changes are documented in the format used to release and communicate design changes. As a minimum, the following information is necessary:

- Unique change identifier
- Originator organization and responsible individual
- Class of change
- Product, assemblies, and components affected
- Configuration documents affected
- Description of change
- Reason for the change
- Proposed change effectivity.

5.3.2 Change Evaluation and Coordination

PRINCIPLE 26: Consider the technical, support, schedule, and cost impacts of a requested change before making a judgment as to whether the change should be approved for implementation and incorporation in the product and its documentation.

The change evaluation and coordination process, modeled in Figure 6, encompasses reviewing the preliminary impact assessments; determining the required change effectivity; establishing the cost/price; and dispositioning (approving, deferring for more research, or disapproving) the change.

5.3.2.1 Change Impact Assessment

PRINCIPLE 27: Determine all potential effects of a change and coordinate potential impacts with the impacted areas of responsibility.

The impact assessment details what would be affected by the change and ensures that all potential effects are known. This information is essential to determine effectivity. The effectivity enables the impact to be quantified, and the total implementation of the change to be priced and scheduled. The functional areas normally possess the specific detailed knowledge required for an accurate assessment. For example, they may know that a needed test facility will be closed during a critical time period, necessitating an alternative or more costly approach.

Change Boards are a common means of achieving the coordination necessary to evaluate a change and assess its impact. Change boards go by many aliases, but whether it is a committee, team, program review board, configuration control board, change review board, or some other name, it should have the following characteristics:

- A change board should be chaired by someone with the authority to commit the resources of the enterprise to implement the change (See 5.3.2.4)

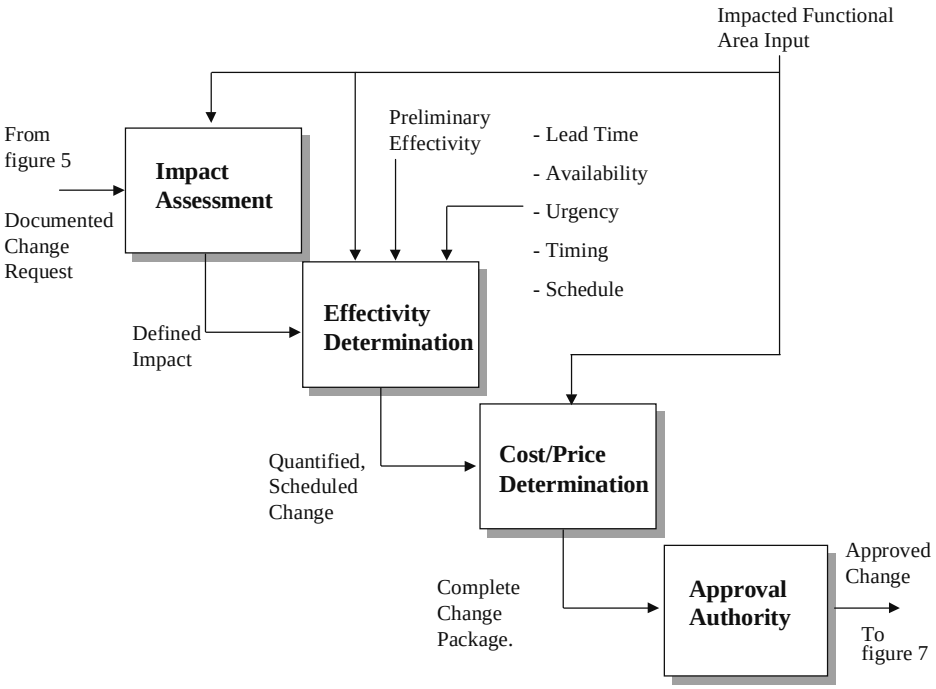

Figure 6 — Change Evaluation and Coordination Process Model

— Members of the board should represent the functional activities (e.g., Engineering, Maintenance, Operations, Training) or product development teams impacted by the proposed changes and have the authority to commit the group they represent to the implementation actions required for the changes
— Review agendas and documents should be provided to board members prior to the meeting
— Board direction and decisions should be documented and disseminated to all affected activities and retained as a record.

5.3.2.2 Change Effectivity Determination

PRINCIPLE 28: Change documentation delineates which unit(s) of the product are to be changed. Change effectivity includes both production break-in and retrofit/recall, as applicable.

PRINCIPLE 29: A changed product should not be distributed until support and service areas are able to support it.

Lack of timely and concurrent support was one of the primary factors leading to the formulation of the configuration management discipline. Therefore, choosing an effectivity requires knowledge not only of the lead times associated with changing the product (whether in production or by retrofit, recall or other means) but of the actions and lead times necessary to effect the associated change in all impacted support areas (such as the update of support software, availability of spare and repair parts, or revision to operating and maintenance instructions). In addition, effectivity determination typically requires the balancing of a number of other considerations, such as:

— Urgency of the change, e.g., is safety involved
— Parts and materials on hand (can implementation be delayed until they are depleted, can they be modified, or do they need to be scrapped?)

— The need to support multiple configurations because all existing units of the product will not be updated, or will not be updated at the same time
— Timing of the introduction of the changed product with respect to customer preferences and needs, competition, marketing strategies, etc.

Effectivity is expressed in different ways depending on product type and quantity or rate of production. Two common means of expressing effectivity are by product unit identifying number (e.g., serial number) and by date code (e.g., date of manufacture). Other examples of effectivity expression are by model year, model designation, version number, and by product group identifying number (e.g., lot number, batch number). All of the methods of expression are intended to delineate, as clearly and precisely as practicable, which unit(s) of the product are and are not to be changed.

5.3.2.3 Change Cost/Price Determination

PRINCIPLE 30: The decision-maker is aware of all cost factors in making the decision.

Determining change costs (or savings) is usually one of the most critical factors that must be addressed in making a decision about a change. The decision should be based on a cost/benefit analysis covering the remaining product life cycle. Cost estimating and pricing of a change cannot be effectively accomplished without the knowledge resulting from the impact assessment and effectivity determination. That knowledge facilitates determination of not only the immediate cost of making the change but also the expected costs that will be incurred in the future because the change is made. All cost factors are considered, such as whether the cost relates to the increased/decreased price for new parts, or in the scrapping of parts on hand, or to new tooling/software for manufacturing the product, or to the cost of recall and retrofit of already purchased units, or to the increased/decreased cost to fulfill service contracts for customers. Such future costs can influence the change in selling price (and competitiveness) of the product, the change in cost of future service contracts with customers
and other factors that may affect future business. The CM process ensures that the decision-maker is aware of all of these cost factors in making the decision.

5.3.2.4 Change Approval Authority

PRINCIPLE 31: Change disposition decisions are made by an appropriate authority who can commit resources to implement an approved change.

The approval authority for a change can be delegated depending upon the classification of the change (refer to Table 4). As the life cycle progresses, the change authority often transitions to individuals with greater management and fiscal responsibility because the effect of a change can be more widespread and, as a result, the cost impact of change decisions are normally greater. Each organization should establish its own change authority levels taking some or all of the following factors into consideration:

— The phase of the product life cycle
— The extent of technical, support, schedule and cost impacts
— The customer involvement
— The product attributes subject to formal control
— The organizational structure and relationships within the enterprise
— The various levels at which change authority is vested
— The period of performance during which that level of control applies.

An individual author of a document practices the simplest, most informal level of configuration control. The individual controls the document's content and may unilaterally change it. If generation of the complete document is a group effort, each author working on his assigned portion also has unilateral control until such time as the individual portions are consolidated into a common draft. The initial consolidated draft then becomes the governing revision. Responsibility for its management passes to a lead individual. The individual contributors can no longer change their portion of the governing revision (current configuration) unilaterally without coordinating with the leader. When the "nth" draft

revision of the document is provided to “users” outside the group to perform required work, the authoring group relinquishes its authority to unilaterally modify their “product” without determining the effect such a change might have on the users. If work in process, or even scheduled future work, is impacted by the document change, control rises to a level of management with the authority to approve the work scope changes. The process for obtaining approval may be relatively informal, or more structured, depending on the organizational interfaces and customer impact.

The most complex multi-level configuration control is practiced on projects in which authority for making change decisions is structured around tiered “Change Boards." The change approval authority spans many contractor and customer organizations. Each organization has defined limited authority to disposition changes to the portion of the “system” under its cognizance. Those changes that exceed the change approval authority of lower levels are elevated to a higher level.

5.3.3 Change Implementation and Verification

PRINCIPLE 32: Implement an approved change in accordance with documented direction approved by the appropriate level of authority.

PRINCIPLE 33: Verify implementation of a change to ensure consistency between the product, its documentation and its support elements.

A model for the change implementation and verification process is shown in Figure 7. Planning for, and implementing, a change can be very simple or can involve many complex and interrelated considerations. The basic planning for implementation of the change is accomplished during the engineering change evaluation before the change is approved. Once the change is approved, detailed implementation planning, which expands but remains consistent with the basic planning, is normally required.

Implementation of a change involves the release of new or revised configuration documentation including requirements and design information. It may involve changes to operation and maintenance information, build and test information, and sales information (such as catalogs, marketing literature). The new or revised information is identified and released (see 5.2). The release process should correlate the document revisions to the change, or changes, incorporated. A common method of disseminating document changes is by means of document change notices that establish a permanent record of the specific changes as well as facilitate distribution. The format used to reflect the changes to the document depends on the method of document creation and reproduction. For a word-processed document, one perfectly acceptable change record is a redlined/strike-through version of the affected portions of the document.

Implementation of the change in the first (or only) affected unit of the product should be verified to ensure consistency between the product, its documentation and its support elements. In accordance with the change implementation plan, this process may involve a detailed audit of the product against its documentation, a validation of operation, maintenance, installation, or modification instructions, or a simple inspection. The choice depends upon the nature of the product and the complexity of the change. If the change is being introduced into a production line, and all future units will have the change incorporated via the normal production process, it is normally sufficient to ensure that the manufacturing instructions contain the change, that they are released for use (such as with a work order) and that the first articles produced are inspected for compliance. However, when support elements are impacted by a change, or the change is being retrofitted over a period of time to a large number of units, complete implementation and verification of the change can be a lengthy, incremental process.

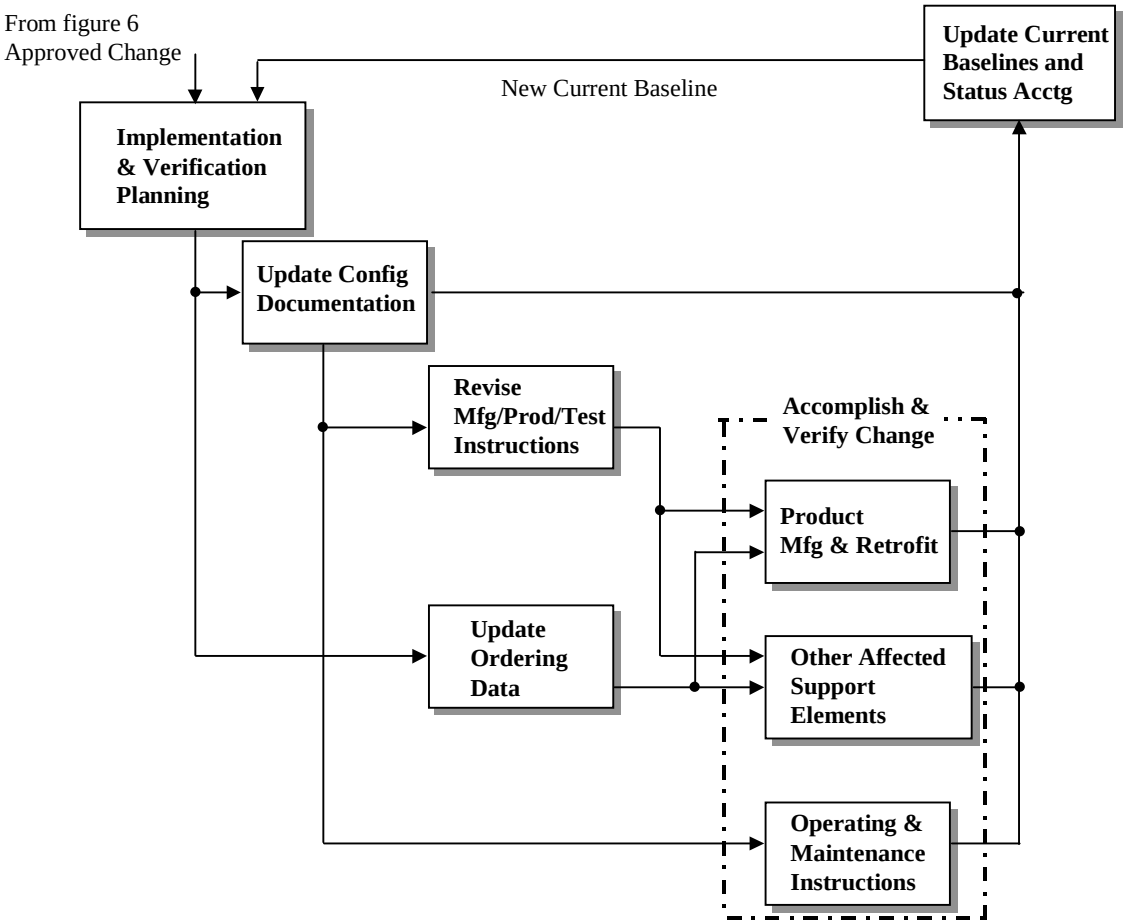

Figure 7 — Change Implementation and Verification Process Model

Under these circumstances, the implementation plan should define the extent to which the change to each unit, or support commodity, is to be verified; and the records that are to be maintained (see 5.4). When the total quantity of materials, or parts, or kits, is ordered in incremental stages (e.g., per year, per month, etc.), it is also necessary to verify that the incremental ordering and supply operations are being completed.

5.3.4 Change Management Process applied to Variances

PRINCIPLE 34: If it is considered necessary to temporarily depart from specified baseline requirements, a variance is documented and authorized by the appropriate level of authority.

Products that incorporate a known departure from requirements (even if the requirements are internally specified) should not be delivered to a customer unless a variance has been documented and authorized.

Authorized variances are temporary departures from requirements and do not constitute a change to the configuration documentation. If it is decided that a departure will be permanent, an engineering change is processed. Similarly, unless unusual circumstances exist, a variance should not be processed if it would affect operation, support, or maintenance (for example, repair parts, operation or maintenance procedures, or compatibility with training devices or test sets); or if it would include the entire remaining number of deliverable units of the product. Rather an engineering change should be proposed.

Requests for variance should always be documented in writing, either by the requester or by the sponsor, after verbal communication with the requester. Requests for variances should include the following information:

- Unique identifier for the variance
- Originator organization and responsible individual
- Category of variance, if applicable
- Identifiers of the product(s) and components affected
- Description of variance including any impacts to performance, operation, maintenance, servicing, operation and maintenance training, spare and repair parts, support and test equipment, catalogs, marketing literature, etc.
- Reason/justification for the variance
- Priority/Urgency
- Proposed effectivity of the variance (limited quantity or time)
- Corrective action to prevent recurrence and/or to eliminate the variance
- Consideration, if any, for accepting variant products
- Alternatives.

Variances may be categorized, or classified[4] to facilitate determination of the appropriate level of approval required for the variance and action to be taken to prevent recurrence. After approval of a variance, there is normally a corrective action to prevent recurrence of the variance or to eliminate it completely as, for example, with an engineering change.

5.4 Configuration Status Accounting

PRINCIPLE 35: An accurate, timely information base concerning a product and its associated product information is important throughout the product life cycle.

Effectiveness of configuration management is closely linked to the flow and availability of configuration information about the product. Information is collected while performing activities associated with the CM processes (planning and management, identification, change management, and verification and audit).

Configuration status accounting (CSA) correlates, stores, maintains, and provides readily available views of this organized collection of information. CSA provides access to accurate, timely information about a product and its documentation throughout the product life cycle.

CSA involves the storage and maintenance of:

- Information about the configuration documentation (such as document identifiers and effective dates)
- Information about the product's configuration (such as part numbers or changes installed in a given unit)
- Information about the product's operational and maintenance documentation (such as the documents affected by each change and their update status), and
- Information about the CM process (such as the status of change requests.)

CSA satisfies both supplier and customer needs for access to information, as needed. CSA improves capabilities to identify, produce, inspect, deliver, operate, maintain, repair, and refurbish products. CSA:

- Enables retrieval of information concerning change decisions
- Supports inquiries concerning future planning of design changes, investigations of design problems, warranties, shelf and operating life calculations, etc.
- Provides access to complete configuration information on a product, any individual product unit, or group of product units

[4] In DoD application practice, variances are called deviations (formerly deviations and waivers) and are classified as Critical, Major and Minor. (See MIL-STD-2549.)

— Provides access to accurate identification of the configuration of each delivered product unit (or batch/lot of product units)
— Enhances availability of accurate information on spare parts and maintenance support
— Provides a source for configuration history of a product and all of its configuration documentation.

5.4.1 CSA Information

PRINCIPLE 36: Configuration information, appropriate to the product, is systematically recorded, safeguarded, validated, and disseminated.

PRINCIPLE 37: Configuration information content evolves and is captured over the product life cycle as tasks occur.

Since the CSA information is a by-product of all the other CM processes, the effectiveness of CSA is dependent on the quality of CM implementation, supported by CM processes that ensure the information is systematically recorded, safeguarded, validated, and disseminated. Decisions on the information to be captured in the CSA system should be based on such factors as the nature of the product, the environment in which the product will be operated, the anticipated volume and complexity of change activity, and the information needs of the customer(s). For example, the critical nature of military aircraft escape systems, which contain explosive devices with limited life, makes accurate complete and traceable change history records for each serial numbered unit essential so that future inquiries and investigations can be supported. As another example, the nature of liability and warranty obligations on a commercial vehicle make it necessary to capture the dates of manufacture and sale, as well as the installation dates of changes.

Application of CSA requires selection of the information to be captured and made available during each phase. Typically, required CSA information is extracted during performance of various tasks by including information recording actions. Table 5 tabulates CSA information typically available during each of the life cycle phases; it represents data availability, without regard to specific organizational responsibility:

a. Conception phase During this phase, some requirements documents may be generated. Information about these requirements documents and their change history are recorded in the CSA system.

b. Definition phase During the definition phase, a product structure is created and dynamically updated as the product's remaining requirements documents and detailed configuration documents are generated, released and baselined. Information about configuration documentation (specifications, engineering drawings, software design documents, software code), test plans and procedures, and any other documents required to design, develop, build, test, and verify the product configuration are recorded in the CSA system.. All change activity information is captured including the status, history and effectivity of all changes that are proposed, dispositioned, and incorporated in the product and affected documentation. The status and history of variances are captured as well. As the design of the product evolves, a record of the release of each configuration document and each subsequent revision to update the design release baseline enter the CSA system. Data accompanying this release record provides the applicability and effectivity of parts, or software units.

Information on the current baseline and changes being processed is always readily accessible during the definition phase. A variety of views into the CSA system are accessed using such keys as product identifier, document identifier, change identifier, variance identifier. The following additional information can be determined:

— The composition of any product, and any component part, in terms of subordinate parts (including supplier parts) and the configuration documents associated with each
— The product identifier (part number) and effectivity on which any subordinate part is used
— The next higher part (or next assembly) in which any part is used
— The as-designed (released) configuration of each product including each applicable engineering change.

Table 5—Typical Status Accounting Information Across the Product Life Cycle

Life cycle Phases Typical CSA Information (Select, where applicable and appropriate)	CONCEPTION	DEFINITION	BUILD	DISTRIBUTION	OPERATION	DISPOSAL
Requirements documentation	●	●	●	●	●	●
Product structure information		●	●	●	●	●
Configuration documentation		●	●	●	●	●
Configuration documentation change notice		●	●	●	●	
Change request and proposal	●	●	●	●	●	
Engineering change effectivity		●	●	●	●	
Variance documentation		●	●	●	●	●
Verification and audit action item status		●	●	●	●	●
Event date entries		●	●	●	●	●
Product as-built record			●	●	●	
Product as-delivered record				●	●	
Product warranty information				●	●	●
Product as maintained, as modified					●	●
Limited use, shelf life restrictions, etc.			●	●	●	●
Product operation and maintenance information revision status					●	●
Product information change requests and change notices					●	●
On-line information access directory or index					●	●
Restrictions due to facility/product performance degradation					●	●
Product replacement information						●
Environmental impact information (where applicable)	●	●	●	●	●	●
Product or Parts salvage information						●

At or near the end of the definition phase or the beginning of the build phase, the current design release baseline is verified and becomes the product configuration baseline. During the verification process, information concerning the results of configuration verification and audits (including action item status) is captured and is entered into the CSA system.

c. Build phase All of the information accessible during the prior phases continues to be available in the build phase. Now, however, additional information about the product, and changes to the product are

recorded and tracked. This additional information concerns the verified as-built configuration of each unit of the product. It includes, where applicable, differences from the as-designed, any associated variances, major changes incorporated, and serial/lot numbers of installed components. In addition, such data as the dates associated with shelf life, or other restrictions are recorded. Superseded database records, which provide configuration history and might reflect the configuration of products still to be modified, are retained.

Additional information that can be extracted from the data base in the build phase includes:

- The as-built configuration of each unit (by serial number), including installation and removal of serialized and lot numbered components
- The identifiers and product unit serial numbers which constitute effectivity of each approved major engineering change; the identifiers of the changes which have been released for any specific serial-numbered unit of a product
- Superseded configuration records that reflect prior configurations of the product.

d. Distribution phase Much of the CSA information previously listed is continued since the distribution phase typically overlaps the build phase. If a product is one that requires installation, changes during the installation process that affects the product are added. During this phase, additional information about customers, deliveries, installations, warranty starts, etc. are also added to the CSA system.

The distribution phase information that can be extracted from the database includes:

- Customers and dates of delivery
- Installation Configuration
- Warranty expiration dates for each unit delivered or installed
- Service agreement type and expiration.

e. Operation phase Status accounting during the operation phase varies considerably based on the type of product and the contractual allocation of CM responsibilities. Changes to the product, which occur due to parts replaced during maintenance, modification of the product by retrofit of engineering changes, or installation of software upgrades, are recorded. During operation, information about changes to the product behavior may be monitored and captured.

Primary concerns during operations are the revision status and accessibility of operation and maintenance documentation. Ideally, stored product information is effectively indexed and provided by direct on-line access. Changes processed in this phase often concern improvements to operator and maintenance information, based on experience, that do not impact the product itself. The current “Operational baseline” consisting of the current status accounting information concerning these documents is maintained through appropriate change management procedures to ensure that the improvements are authorized and reflected in the product information.

Additional information extracted from the CSA system during operations, therefore includes:

- Product as-maintained and as-modified configuration
- Product operation and maintenance information revision status
- Product information change requests and change notices
- On-line information access directory or index
- Restrictions due to facility/product performance degradation.

f. Disposal phase In this phase, during which the product is removed from service and disposed of, the information captured in the CSA system depends upon the product type. It also depends upon such factors as whether disposing of a product may have adverse environmental implications; if there are product replacements; or if parts salvage is involved. Some legal and contractual statutes require retention of specified data even after disposal.

5.4.2 CSA System

PRINCIPLE 38: Data collection and information processing system requirements are determined by the need for configuration information.

Configuration records are generated and maintained by an effective and timely CSA information management system. Most information resulting from such CM activities as release, change evaluation, change approval, and change verification, are collected, correlated and maintained in a database. The extent, complexity, and inter-relationships of product information to be managed typically requires the use of information systems products such as a product data manager and a work flow manager. The supporting information system should also provide storage and security of product information and traceability of product history.

CSA data is collected from dedicated data sources within the engineering, project management, manufacturing, quality assurance, logistic support organizations, and the customers/users. The accumulated information is processed into the CSA system to provide structured records on the product and its related documentation. The requirements for the depth of product configuration information needed for management, purchasing, test and evaluation, operation, support, and spares, determines the content of the data collection, storage, processing, and distribution.

Modern applications generally collect CSA data as a normal course of business using integrated information systems serving many users and functions, efficiently entering data one time for multiple uses. Such systems are designed using data and process models that include common data dictionaries to facilitate data interchange. The automated integrated information system typically provides real time access and transfer of CSA (as well as other information) between customers, product development teams, subcontractors and others using an external electronic data interchange (EDI) network. Using an external EDI to provide information eliminates the need for extensive paper reports. In situations where such systems are not applicable, a standalone CSA system (or even manual records) may be appropriate.

In a common example of an integrated system approach, software configuration information, status, and meaningful SCM metrics, as needed, are typically compiled using an automated SCM tool within a software engineering environment. Status accounting for software products is the by-product of:

- Baselining software versions
- Providing library control
- Change management related to software versions and their documentation

5.5 Configuration Verification and Audit

PRINCIPLE 39: Verification that a product's requirement attributes have been met and the product design meeting those attributes has been accurately documented is required to baseline the product configuration.

Configuration verification and audit establish that the performance and functional requirements defined in the configuration documentation have been achieved by the design and that the design has been accurately documented in the configuration documentation. The purpose and benefits of the process include the following:

- Ensure that the product design provides the agreed-to performance capabilities
- Validate the integrity of the configuration documentation
- Verify the consistency between a product and its configuration documentation
- Determine that an adequate process is in place to provide continuing control of the configuration
- Provide confidence in establishing a product baseline
- Ensure a known configuration as the basis for operation and maintenance instructions, training, spare and repair parts, etc.

5.5.1 Design and Document Verification

PRINCIPLE 40: Verification that a design achieves its goals is accomplished by a systematic comparison of requirements with the results of tests, analyses or inspections.

PRINCIPLE 41: Documentation of a product's definition must be complete and accurate enough to permit reproduction of the product without further design effort.

The design of a product must be verified to ascertain that it has achieved specified requirements and desired goals; that the documentation of the design is accurate; and that the product can be produced from the documentation. Conceptually, the verification occurs in sequence by first determining the acceptability of the design and then confirming that the documentation portrays that design. In practice, it may be accomplished in separate events or audits. It is often more practicable to verify these aspects incrementally during the course of the definition phase and to incorporate the verification into the design/manufacturing process flow, so that it occurs on a continuous basis.

The verification methods should be carefully planned to ensure that all requirements are addressed and the individual verification methods chosen are appropriate. Verification of complex designs usually involves a formal plan, which may require customer approval. Requirements analysis and test tools which flow down, account for, and verify all attributes facilitate the design verification process. Results are typically recorded in a matrix indicating each discrete requirement, the method of verification, verification procedure, and the verification results.

The design output, consisting of the complete set of design information, must be accurately documented to permit reproduction of the product without further design effort. Beyond this fundamental requirement, other factors (such as the need to procure from other sources or future maintenance needs) may influence the content and formality of documentation. A product should be able to be produced from its documentation with confidence that it will meet all requirements.

A software product should also be in compliance with published design and coding standards so that it can be maintained, modified and upgraded. In addition, the following should be verified:

- The documentation library control system
- Uniqueness of the product identifier
- Validity of interfaces
- Internal audit records of CM processes and procedures

Verifying the documentation determines that it is adequate for its intended purposes and accurately reflects compliant design.

The verification of design and documentation must be planned to permit its accomplishment at minimum cost. In complex physical products, the comparison of the documentation with the prototype or test article can often be accomplished incrementally, during assembly of the article, to avoid the need for later disassembly. These verifications are considered complete upon resolution of discrepancies or departures found and correction of associated documentation.

5.5.2 Configuration Audit

PRINCIPLE 42: Where necessary, verification is accomplished by configuration audit.

An effective configuration audit typically performed at the conclusion of the definition phase or at the start of the build phase includes performance verification (functional configuration audit) and design verification (physical configuration audit). Other types of configuration audits are conducted in various phases of a product life cycle. Some examples are: (1) an audit to determine the currently installed complement and configuration of equipment in an operational facility or a vehicle; and (2) an audit to assess the quantities and configurations of spare parts in inventory.

Audits may be conducted by the organization responsible for the product development, by the customer, or by a third party designated by the customer. A chairperson representing each party to the audit participates in audit planning and preparation. Audit plans and agendas are reviewed and agreed to prior to the audit. Audit of a complex product may be accomplished in a series of incremental audits.

During the conduct of an audit, audit participants record significant questions, discrepancies or anomalies, and recommended courses of action. Chairpersons review audit findings and determine appropriate actions. Affected parties agree to action items and the plan for effecting their successful closure. Audit minutes provide a record of the audit findings, conclusions, recommendations, and action items. Follow-up occurs until all required action items are complete.

The necessary resources and material to perform an audit include the following items to the extent appropriate for the type and scope of audit:

- An audit plan and agenda
- Adequate facilities and unencumbered access
- Assignment and availability of personnel
- Applicable specifications, drawings, manuals, schedules, and design data, test results, inspection reports, process sheets, data sheets, safety procedures, and other documentation as deemed necessary
- Tools and inspection equipment necessary for evaluation and verification
- Access to the product(s) and detailed parts to be reviewed.

5.5.3 Continuing Performance Audits and Surveillance

PRINCIPLE 43: Periodic reviews verify continued achievement of requirements, identify and document changes in performance, and ensure consistency with documentation.

As appropriate, ongoing production and processes are audited, tested or inspected to determine continued suitability and consistency of the product or process with its documentation. This review process may be accomplished by methods ranging from manual inspection to statistical trend analysis. The choice of method is influenced by the opportunity for variability and the impacts of such variability on the performance of the product.

Operation of products or facilities is periodically reviewed to identify and monitor changes or degradation of performance, or to compare existing elements with new criteria or requirements. Associated documentation is changed or updated to maintain consistency between the product and its definition.

Reviews of all or portions of the CM process and procedures are conducted periodically or when indicated by process metrics. Periodic assessments enhance the effectiveness of the CM process.

5.6 Configuration Management of Digital Data

Note: *While some principles in this section are encompassed by principles in other sections, they are specifically tailored to add a specific focus for digital data. The CM practitioner will also find that the principles identified in other sections of this standard are also applicable to digital data.*

PRINCIPLE 44: Apply configuration management principles to ensure the integrity of digital representations of product information and other data.

As the predominant media for communicating information evolves from transmitting paper to interactively accessing information resident in databases, the scope of configuration management expands to encompass all digital data. Digital data is information prepared and maintained by electronic means, and provided by electronic data access, interchange, transfer, or on electronic media.

The purpose and benefits of data configuration management are to ensure the integrity of digital data and enhance good data management practice by providing:

- Effective file and database management
- Unique identification of documents, files and document representations
- Retention of essential file and version relationships
- Known data status
- Controlled access to digital data.

The application of configuration management to digital data is expressed in terms of the following elements:

- Digital data identification
- Data status level management
- Maintenance of data dependency and product configuration relationships
- Data version control and management of review, comment, annotation, and disposition
- Digital data transmittal
- Data access control.

5.6.1 Digital Data Identification

PRINCIPLE 45: Apply digital data identification rules to maintain document, document representation, and file version relationships.

Digital data files should be carefully identified to differentiate between similar files and to maintain traceability to specific product configurations and document representations. Because file-naming conventions vary widely among operating systems and application programs, it is often necessary to store appropriate information about the files (meta-data) providing the correct relationships and associations. Document management system software is often a practical necessity as the amount and complexity of digital information expands.

A document is digitally represented by one or more electronic data files. Document representations consisting of digital data files (whether word processor, CAD/CAM representations, or software deliverables) are often the official or de facto original of a document and, as in the case of software deliverables, may well represent the product itself. Each document representation represents one document and includes all the files necessary for the complete document, such as:

- A set of files which were created by a particular application software (such as CAD or off-the-shelf (OTS) word-processing software)
- A set of files in a standard neutral format (such as Integrated Graphics Exchange Standard (IGES), ASCII, SGML-tagged ASCII, Computer Graphics Metafile)
- A set of related files in different formats created by different application software.

One document, however, can have several different, equally valid, representations (such as the same document represented in two different OTS word processing formats). Any individual file (such as a raster graphics file, an ASCII file, or a spread sheet file) may be part of any number of document representations of the same document at the same revision; of iterations (different revisions) of the same document; or of different documents.

To facilitate the proper relationships, apply the following digital data identification rules:

- Assign a unique identifier to each file
- Assign a unique identifier to each document representation
- Assign a version identifier to each file
- Maintain, in a database, the relationship between:
  - Document identifier and its revision level (see 5.2.4)
  - Associated document representation(s)
  - File identifiers and versions
- Retain multiple versions of files as necessary to recreate prior document revisions and provide a traceable history of each document.

5.6.2 Data Status Level Management

PRINCIPLE 46: Apply business rules using data status levels for access, change management, and archiving of digital documents.

Data status level management is a process that defines and applies business rules based on the status of a digital data document. It facilitates data workflow management and enhances data integrity. Figure 8 illustrates a standard data life cycle model. It shows common data status levels, also referred to as states, of a specific document or document revision.

Business rules are established for each document type by defining requirements such as the following:

- Document/document revision identifier and file version identifier scheme (see 5.6.1)
- Application software and data format
- If submittal to (or access by) customer(s) is required
- Who will be granted access privileges to the data in each of the applicable states
- What are the approval requirements (reviewers/approvers) and method of approval (e.g., electronic signature) to enter the released state; the approved state
- What are the archiving rules for this document type (e.g., all released versions upon release of a superseding version, all released versions, 90 days after release of a superseding version, etc.)

Working data represents work in progress, which is highly subject to change without notice. Sub-states are often defined for working data to distinguish data that are in draft state, review state, etc. Any revision of released data reverts to the working data state.

Released data are data that have been reviewed and authorized for use or, if required, that have been authorized for submittal to, or access by, a customer. Released data are under change management rules. A new version cannot replace a released document until it has also been reviewed and authorized by the appropriate authority. The configuration of a document is fixed, once it is in the released state. It is only changed by release of a superseding revision.

Subsequent status levels apply only to documents requiring customer action. They indicate that the released document was promoted without change to another state. Submitted data are data in the same configuration as the released data but have been made available for customer review. If a submitted document is commented to, or disapproved, a new working version may be started. An approved data status means that the data has been

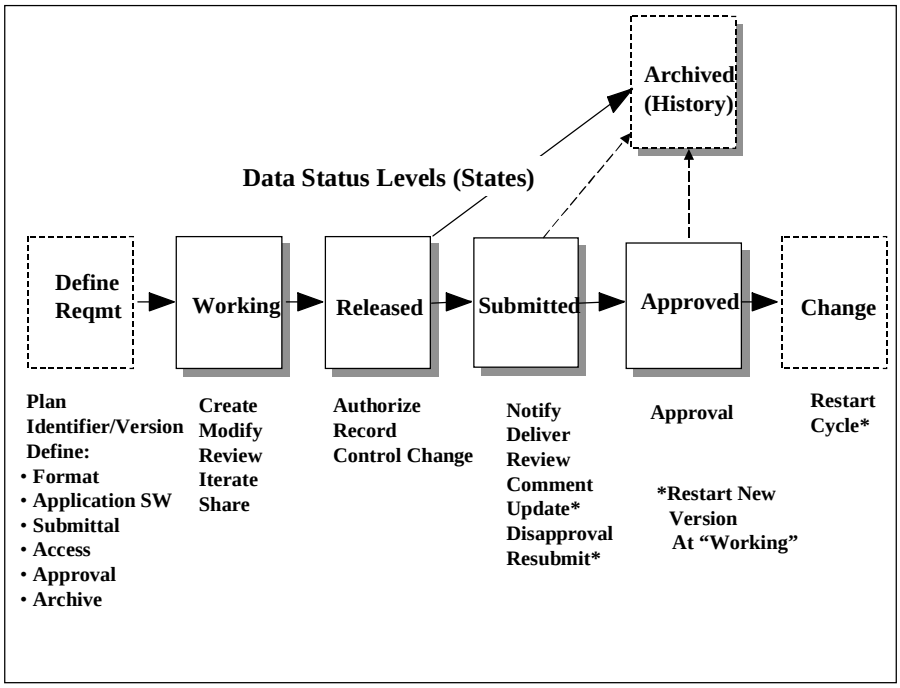

Figure 8 — Standard Data Life Cycle Model

approved by a customer. Depending on its type and purpose, a document may not progress past the working or released state. Data that are not required to be submitted remain in the released state; some data exist only at the working level. Archived data are data (in the same configuration as released, submitted, and approved) that have been removed from an active access storage mode.

5.6.3 Maintenance of Data and Product Configuration Relationships

PRINCIPLE 47: Maintain relationships between digital data, data requirements, and the related product configuration to ensure accurate data access.

An effective system of managing the relationships of data files, document representations and key data elements must be employed if data are to be accessed or retrieved in a controlled manner. Data files are related to documents via document representations. The documents are related to:

- Program/project and/or contractual agreement
- Contract data item identifiers
- Document change authorization
- Document name/document title/identifier
- Document revision number and date
- Associated product (hardware or software) name
- Associated product part or software identifying number
- Status (working, released, submitted, approved. archived)
- Associated data
- Associated correspondence.

If customers or other users have access to data base applications, key relationships within the file or database must be preserved to maintain integrity of data they may extract. These relationships (dependencies) ensure, for example, that extracted information concerning a given product or part is associated correctly with the configuration and effectivity (serial number) of the product that uses the part, and not with an earlier or a later configuration that does not.

5.6.4 Data Version Control and Management of Review, Comment, Annotation, and Disposition

PRINCIPLE 48: Apply disciplined version control to manage document review electronically.

Managing the document review process electronically is essentially a process of version control to establish an audit trail of comments or annotations by reviewers, and the disposition of each comment. Each version of each document representation provided to, or received from, each reviewer must be uniquely identified. Since a single document representation can consist of many files, a very disciplined process is necessary. The versions in one typical review cycle, as follows, illustrate the document representation versions that might occur; each representation requires unique identification and correct association:

- The revision distributed for review
- The redline/strike-out version of the same revision provided to aid in viewing the changes from the prior issue
- The annotated version which has been marked up to indicate reviewer comments. If there are many reviewers, there could be many of these files.
- The dispositioned version of the reviewer annotated file(s)
- The version (next revision) that is updated as a result of the dispositioned comments
- The redline/strike-out annotated version of the new revision to document the incorporation of comments.

5.6.5 Digital Data Transmittal

PRINCIPLE 49: Ensure that a transmitted digital data product is usable.

When data are provided on physical media such as floppy disks or tapes, appropriate identification, similar to software media identification, is affixed to the media to clearly identify its contents. When it is impracticable to include all of the file identifications, a reference to an accompanying listing or to a read me file is necessary.

Whether transmittal is by floppy disk, electronic data interchange (EDI), modem, or other means, the supplier should ensure that the deliverable data product can be recreated in human readable form and processed by the user. The supplier's instructions ("read.me" files, reference to standard protocols, on-line help) include (as applicable):

- Identification of the files included in the transfer by file name, description, version, data status level, and application/file type
- Applicable references to associate the data with the basis (requirement) for its transmittal, approval, and payment, where applicable
- If there are multiple files, such as separate text and graphics, how to assemble each included data item for reading, review or annotation, as applicable
- The naming convention for file versions and data status level that distinguishes altered (for example, annotated or red-line/strike-out) file versions from unaltered files.
- If and how changes from previous versions are indicated
- How to acknowledge receipt of the data, provide comments, and/or indicate disposition of the data digitally
- Time constraints, if any, relating to review and disposition.

5.6.6 Data Access Control

PRINCIPLE 50: Effective digital data access fulfills requirements, preserves rights, and provides users with data they are entitled to in the correct version.

For electronic data access to be effective, a process of access privileges is employed to limit access to applicable users. Access privileges vary according to the data status level, the nature of the data and the needs of the user.
Typically, working data are made available only to the originating individual, group, or team (such as an integrated product development team); or to other designated reviewers of the data. If customers are part of the team, they have the same access as the other members. With this exception, customer access to digital data (including data retrieved from databases) is limited to released, submitted, and approved data that is contractually agreed to.

Users of accessed data must respect all contractual and legal requirements for data rights, security, licenses, copyrights, and other secondary distribution restrictions that apply to the data.

Irrespective of the means of storage of digital data, the supplier must be capable of retrieving the appropriate files necessary to produce, view, and provide the correct version of each digital data document. Where access is interactive (i.e., pre-defined query and extraction of data), the supplier's process should provide the following information, as applicable:

- How data is to be accessed
- Request for access and logging of access for read-only or annotation
- Naming of temporary working version of the file(s) for purpose of annotation/mark up
- Means of indicating whether a comment/annotation is essential/suggested
- Re-identification of marked up versions, as required
- Method of indicating acceptance, approval, or rejection, as applicable
- Time constraints, if any, on data acceptance
- Tracking of disposition of required actions
- Re-identification of changed files.

Page intentionally blank

6 Application Notes

NOTE 1 — EIA 649 is a guidance document. It is not intended to be made a compliance document in a contract for purchase/acquisition of a product. It may be used as a source from which applicable information can be extracted to prepare such items as a request for proposal, or an evaluation or certification checklist. The application of appropriately selected principles and practices to a product or on a project will enable the user to plan and document an appropriate configuration management program for that product or project environment.

NOTE 2 — Configuration management practices should be applied selectively, and to a degree commensurate with the product application environment. A configuration management practice should not be implemented solely because an evaluation standard (such as the ISO-9000/10000 series of Quality Programs) addresses the subject. In circumstances where given practices are not necessary, evidence of configuration management planning and a rational basis for selection of appropriate practices should be adequate to satisfy the evaluation criteria.

Page intentionally blank

Annex A Summary of Configuration Management Principles (informative)

No.	Principle	Clause
1.	**Plan CM processes for the context and environment in which they are to be performed and manage in accordance with the planning: assign responsibilities; train personnel; measure performance; and assess measurements/trends to effect process improvements.**	**5.1**
2.	**To determine the specific CM value adding functions and levels of emphasis for a particular product, identify the context and environment in which CM is to be implemented.**	**5.1.1**
3.	**A configuration management plan describes how configuration management is accomplished and how consistency between the product definition, the product's configuration, and the configuration management records is achieved and maintained throughout the applicable phases of the product's life cycle.**	**5.1.2**
4.	**Prepare procedures to define how each configuration management process will be accomplished.**	**5.1.3**
5.	**Conduct training so that all responsible individuals understand their roles and responsibilities and the procedures for implementing configuration management processes.**	**5.1.4**
6.	**Assess the effectiveness of CM plan implementation and performance of the configuration management discipline with defined metrics (performance indicators).**	**5.1.5**
7.	**Performing configuration management includes responsibility for the configuration management performance of subordinate activities (e.g., subcontractors, suppliers).**	**5.1.6**
8.	**Configuration identification is the basis from which the configuration of products are defined and verified; products and documents are labeled; changes are managed; and accountability is maintained.**	**5.2**
9.	**Configuration documentation defines the functional, performance, and physical attributes of a product. Other product information is derived from configuration documentation.**	**5.2.1**
10.	**The product composition (i.e. relationship and quantity of parts that comprise the product) is determinable from its configuration documentation.**	**5.2.2**
11.	**All products are assigned unique identifiers so that one product can be distinguished from other products; one configuration of a product can be distinguished from another; the source of a product can be determined; and the correct product information can be retrieved.**	**5.2.3**
12.	**Individual units of a product are assigned a unique product unit identifier when there is a need to distinguish one unit of the product from another unit of the product.**	**5.2.3.1**
13.	**When a product is modified, it retains its original product unit identifier even though its part identifying number is altered to reflect a new configuration.**	**5.2.3.1**

No.	Principle	Clause
14.	**A series of like units of a product is assigned a unique product group identifier when it is unnecessary or impracticable to identify individual units but nonetheless necessary to correlate units to a process, date, event, or test.**	**5.2.3.2**
15.	**All documents reflecting product performance, functional, or physical requirements and other product information are uniquely identified so that they can be correctly associated with the applicable configuration of the product.**	**5.2.4**
16.	**A baseline identifies an agreed-to description of the attributes of a product at a point in time and provides a known configuration to which changes are addressed.**	**5.2.5**
17.	**Baselines are established by agreeing to the stated definition of a product's attributes.**	**5.2.5.1**
18.	**The configuration of any product, or any document, plus the approved changes to be incorporated is the current baseline.**	**5.2.5.2**
19.	**Maintaining product information is important because time consuming and expensive recovery may be necessary if records of operational units of a product do not match the actual units (as reported by maintenance activities) or such records do not exist.**	**5.2.6**
20.	**For product interfaces external to the enterprise, establish an interface agreement and a mutually agreed to documentation of common attributes.**	**5.2.7**
21.	**Changes to a product are accomplished using a systematic, measurable change process.**	**5.3**
22.	**Each change is uniquely identified.**	**5.3.1**
23.	**Changes represent opportunities for improvement.**	**5.3.1.1**
24.	**Classify requested changes to aid in determining the appropriate levels of review and approval.**	**5.3.1.2**
25.	**Change requests must be clearly documented.**	**5.3.1.3**
26.	**Consider the technical, support, schedule, and cost impacts of a requested change before making a judgment as to whether the change should be approved for implementation and incorporation in the product and its documentation.**	**5.3.2**
27.	**Determine all potential effects of a change and coordinate potential impacts with the impacted areas of responsibility.**	**5.3.2.1**
28.	**Change documentation delineates which unit(s) of the product are to be changed. Change effectivity includes both production break-in and retrofit/recall, as applicable.**	**5.3.2.2**
29.	**A changed product should not be distributed until support and service areas are able to support it.**	**5.3.2.2**
30.	**The decision-maker is aware of all cost factors in making the decision.**	**5.3.2.3**

No.	Principle	Clause
31.	**Change approval decisions are made by an appropriate authority who can commit resources to implement the change.**	**5.3.2.4**
32.	**Implement an approved change in accordance with documented direction approved by the appropriate level of authority.**	**5.3.3**
33.	**Verify implementation of a change to ensure consistency between the product, its documentation and its support elements.**	**5.3.3**
34.	**If it is considered necessary to temporarily depart from specified baseline requirements, a variance is documented and authorized by the appropriate level of authority.**	**5.3.4**
35.	**An accurate, timely information base concerning a product and its associated product information is important throughout the product life cycle.**	**5.4**
36.	**Configuration information, appropriate to the product, is systematically recorded, safeguarded, validated, and disseminated.**	**5.4.1**
37.	**Configuration information content evolves and is captured over the product life cycle as tasks occur.**	**5.4.1**
38.	**Data collection and information processing system requirements are determined by the need for configuration information.**	**5.4.2**
39.	**Verification that a product's requirement attributes have been met and the product design meeting those attributes has been accurately documented is required to baseline the product configuration.**	**5.5**
40.	**Verification that a design achieves its goals is accomplished by a systematic comparison of requirements with the results of tests, analyses, or inspections.**	**5.5.1**
41.	**Documentation of a product's definition must be complete and accurate enough to permit reproduction of the product without further design effort.**	**5.5.1**
42.	**Where necessary, verification is accomplished by configuration audit.**	**5.5.2**
43.	**Periodic reviews verify continued achievement of requirements, identify and document changes in performance, and ensure consistency with documentation.**	**5.5.3**
44.	**Apply configuration management principles to ensure the integrity of digital representations of product information and other data.**	**5.6**
45.	**Apply digital data identification rules to maintain document, document representation, and file version relationships.**	**5.6.1**
46.	**Apply business rules using data status levels for access, change management, and archiving of digital data documents.**	**5.6.2**
47.	**Maintain relationships between digital data, data requirements, and the related product configuration to ensure accurate data access.**	**5.6.3**

No.	Principle	Clause
48.	**Apply disciplined version control to manage document review electronically.**	**5.6.4**
49.	**Ensure that a transmitted digital data product is usable.**	**5.6.5**
50.	**Effective digital data access fulfills requirements, preserves rights, and provides users with data they are entitled to in the correct version.**	**5.6.6**

Annex B Selection of Appropriate Principles/Practices for a Given Product (Including Hardware, Software, Materials, Processes and Services) (Informative)

B.1 Scope.

All of the configuration management principles described in this standard are applicable to a product environment in which a robust configuration management approach is necessary. Such an environment is typically a complex product such as an electronic system, a military weapon, an air or land vehicle, or any other product containing hardware, software or a combination of both that must be supported over the product life cycle. However all of the principles will not be applied in every phase of the product's life cycle.

There is a wide variety of end product types to which configuration management can provide benefit, for example, a document, a customer database for a service, construction of a building or an information network facility. When configuration management is applied to these and other environments or to less complex products, some of the principles may not apply. It is imprudent to prescribe precisely the specific principles and practices that are applicable to each product type and environment; each case must be examined individually taking in to consideration the answers to questions, such as:

- Who is the customer(s)?
- What is the nature of the product?
- Is the product a unique design for one application?
- Are there many products of the same general nature in the marketplace?
- What is the current phase of the product life cycle?
- What is the expected future life and employment of the product?
- What associated services such as training and product maintenance are to be provided?
- Under what circumstances can the product be changed?
- Is the product a consumable?
- What is the condition of the data defining the product configuration?
- Is the product presently being produced?
- To what extent is the product repairable?
- Is a precise same-configuration replacement item necessary or can a generic substitution be made?

This annex provides a general methodology for selecting which principles are appropriate. It prescribes a benefit-risk assessment before discarding principles which, although desirable, appear to be too costly. All too often, the "best practices" associated with a principle are equated with "most costly" when, in fact they represent "best investment." In the long run, these investments, like insurance, may prove to be the "least costly" alternative. The selection of applicable CM principles and practices to apply in a given instance is a CM planning function that is based on judgment and common sense. The practices implementing the CM principles can be shaped in a wide variety of way to achieve the essence of CM as appropriate for the circumstances.

B.2 Selection Methodology

To select the appropriate principles and their implementing practices for a given product, the following is a recommended method of assessment; any method that considers the product characteristics, environment, and life cycle may be performed:

1. Determine the most significant characteristics about the product in the following categories, using the typical summary of characteristics in Table B.1 as a general guide:
   - Product nature.....This category helps in determining the nature to which a given principle should be applicable
   - Product complexity……. This category focuses on the degree to which a given principle should be applicable

- Customer .....This category focuses attention on customer expectations
- Product environment/life cycle...This category highlights the product use/business environment.

Table B.1—Typical Product Categorization for Use in Selecting Applicable Principles and Practices

SAMPLE PRODUCT CHARACTERISTICS			
Product Nature	**Hardware**	Hardware/Software	Electric
	System	Subsystem	Component
	Electronic	Mechanical	Pneumatic
	Mobile	Stationary	Disposable
	Software	Operating System	Utility
	Application	Recreational/Game	Development Tool
	Mission	Process Control	Computational
	Network	Communications	Options
	Material	Consumable	Perishable
	Limited Shelf Life	Liquid/Gas	Chemical
	Document	Paper	File
	Composite	Metal	Fabric
	Patterns	Sizes	Variety
	Process	Manufacturing	Assembly
	Analytic	Legal	Distribution
	Facility	Production	Operation
	Maintenance	Disposal	Storage
	Office	Power Generating	Sales
	Service	Documentation	Delivery
	Consulting	Maintenance	Design
Product Complexity	Standard	Unique	Hi-Tech
	Critical	Safety	State-of-the-art
	Simple	Standalone	
Customer	Single	Many	Contract
	Multi-Contract	Prime/Integrator	Consumer
	Government	Commercial	Military
Environment And Life Cycle	Concept	Development	Production
	Support	Service	Spares
	Factory Test/Simple	Factory Test/Complex	Distribution
	Installation/User	Installation Complex/User	Installation /Provider
	Operation-None	Operation-Simple	Operation –Complex
	Operator Training Required	Maintenance-User	Maintenance-Provider
	Service Contract	Warranty	Drive-in Service
	On-Site Support		

2. Examine the selected product characteristics with each of the 50 principles of CM (See Annex A); determine whether each principle is:

- Obviously required
- Desirable but not essential
- Obviously not applicable

3. For the desirable but not essential principles, consider the benefits if implemented and the risks if not. Consider both customer and provider concerns. Briefly describe how the principle should be implemented in practice, estimate the cost (or cost savings) as high, medium, or low, and rank the principles

4. In light of the factors in step one and the benefit/risk/cost ranking in step 3, determine which of the desirable principles are affordable. A cost/risk chart as shown in Figure B.1 can be helpful in this analysis.

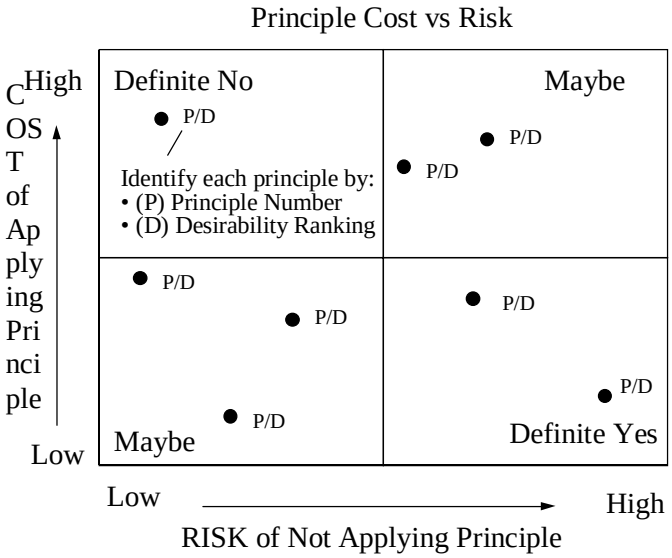

Figure B.1 — Affordability of Desirable CM Principles

As a check, sketch out a rough implementation plan using the “obviously required" and the selected “desirable but not essential” principles.

Page intentionally blank

Annex C Related Documents (informative)

This industry standard relates to other existing documents as follows:

C.1 Relation to Other Industry Standards

EIA 649 is compatible with EIA 632, "Processes for Engineering a System," ANSI[5]/EIA JSTD-016 "Software Development Process," and ISO[6] /IEC 12207, "Information Technology —Software Life Cycle Processes."

The following ASME[7] standards that relate to engineering drawing practices are of primary importance since engineering drawings and associated lists are an essential element in defining product configuration (design information):

- ASME Y14.24M, "Engineering Drawing Types"
- ASME Y14.34M, "Associated Lists"
- ASME Y14.35M, "Revisions to Engineering drawings."
- ASME Y14.100M, "Engineering Drawing Practices."

C.2 Relation to ISO 9000 and 10000 Series Quality Program Standards

The (ISO) 9000 Series Standards and their ASQC[8] equivalents are used to audit and evaluate a contractor's quality program. Within these standards there are broadly stated requirements that are met if the contractor has an adequate configuration management program implemented using the principles in EIA 649. Of particular importance are the following clauses within each of the ISO and ASQC documents:

- Design Control
- Document and Data Control
- Product Identification and Traceability
- Control of Nonconforming Product.

A similar relationship exists with ISO 10007 "Quality management -- Guidelines for configuration management." ISO 10007 provides broad general guidelines that can easily be mapped to the principles and best practices in EIA 649.

C.3 Relation to ISO, ANSI and IEEE[9] Software Standards

The following are useful references for configuration management practices unique to software, the requirements of which are compatible with the principles and practices in EIA 649:

- ANSI/IEEE Standard 610.12-1990, "Standard Glossary of Software Engineering Terminology"
- IEEE Standard 828-1990, "Software Configuration Management Plans"
- ANSI/IEEE Standard 830-1994, "Software Requirements Specification, Guide to"
- ANSI/IEEE Standard 1042-1987, "Guide to Software Configuration Management."

In addition, ISO 9000-3, "Quality Program Standard for Software Development," bears the same relationship to EIA -649 as do the other quality program standards.

The Software Engineering Institute's Software Development Capability Maturity Model (SEI/CMM) delineated in SEI-93-TR-024, provides a best practice view of software development. In this view, CM is a defined process integrated as

[5] American National Standards Institute
[6] International Organization for Standardization
[7] American Society of Mechanical Engineers
[8] American Society of Quality Control
[9] Institute of Electrical and Electronics Engineers

an essential element in the software engineering process. The CMM articulates the need for inter-organizational relationships but it does not prescribe any specific organization or methodology. The CMM and EIA 649 are fully compatible and complementary,

C.4 Relation to US Military Standards and Handbooks on Configuration Management

EIA 649 was prepared in partnership with the DoD Configuration Management Advisory Group (CMAG), with the intent that MIL-STD-973 be canceled when a non-government standard (EIA 649) and MIL-STD-2549, Configuration Management Data Interface Standard, are available. However, by direction of the DoD Specifications and Standards Improvement Council, MIL-STD-973 will be retained for a period of time to allow for transition to a shared data environment.

MIL-STD-2549, prepared by the CMAG, with significant EIA and other industry association participation, addresses the DoD CM information requirements, defines the terminology used by DoD, and contains the conceptual schema (business rules view) of a DoD CM Automated Information System (CM AIS). The conceptual schema is presented as a relational data model and data dictionary. It standardizes the interface with the CM AIS enabling contractors to interface directly and supply information electronically.

Military Handbook 61 (MIL-HDBK-61), also prepared by the CMAG with industry association participation provides guidance to military acquisition program and CM managers in relating Industry EIA 649 to their programs in the environment of the standard DoD CM AIS and the practices associated with performance-based acquisition.

MIL-STD-100 provides essential guidance in numbering, coding and identifying of products and parts on engineering drawings and, by extension, in engineering model digital data sets. It supplements the information in the industry drawing standards listed above.

Another useful reference is MIL-STD-280, “Definition of Item Levels, Item Exchangeability, Models, and Related Terms,” which defines the standard terminology used in naming equipment used by the military.

Annex D Acknowledgment of Participants in EIA Standard 649 (Informative)

EIA/IS-649 Core Working Group

Alan Lager - MLR Associates
Bill Brummer – Westinghouse Savannah River
Don Noble - Lockheed Martin
Wayne Starkweather – Raytheon TI Systems
Mitch Kaarlela - Lockheed Martin
Leon Waldron - Boeing
Fred Bahrs - CMstat Corporation
Nan Barnes - Litton Data Systems
Lee Foley - Honeywell
Bill Dean - US Air Force Institute of Technology
Bob Beermann - US Army (ARDEC)
Chris Denham – EIA

EIA 649 Core Working Group

Alan Lager - MLR Associates
Don Noble - Lockheed Martin
Mitch Kaarlela - Lockheed Martin
Leon Waldron - Boeing Defense & Space Group
Fred Bahrs - CMstat Corporation
Lee Foley - Honeywell
Joseph Mystkowski, Northrop Grumman
James Leonard - Lockheed Martin
Bill Bichel - McDonnell Douglas
Bill Dean - US Air Force Institute of Technology
Adrienne Scott - TRW
James Martin, Raytheon Systems
Chris Denham - EIA

Multi-Association Advisory Group

Ron Berlack IEEE
Mike Daniels NDIA & SOLE
Chris Denham EIA
Linda Fowlbe DoD
Harvey Schock ASQC
David Purvis ACDM

Many others, too numerous to list, provided invaluable guidance, critique, and feedback during EIA Quarterly meetings and on Panels at the annual EIA Engineering and Management Workshops from 1994 - 1997. In addition, the core groups would like to acknowledge those who took the time to provide verbal, written and e-mail comments, observations, and words of encouragement.

Page intentionally blank


Electronic Industries Alliance