G E N E R I C   C M

C O N F I G U R A T I O N   M A N A G E M E N T
E X P L A I N E D

A P P E N D I X
Revision 1

By Richard L. Rumage

CONTENTS

FIGURES

TABLES

REVISIONS

Rev. 1, 6/2/06: Inserts “Competitive Data Package” subtitle on page 75. Adds “Revisions” page and renumbers subsequent pages appropriately. Adjusts Tables of Contents accordingly. Adds “Revision” number to headers. Corrects figure captions. Cleans up Table 1.

INTRODUCTION

This Appendix covers a number of side issues. They are not essential to an understanding of Generic CM. However, they are helpful in explaining why things are the way they are. The following sections parallel the sections of the basic text and usually amplify, clarify, or detail that material.

Prerequisite

CM is bounded on one side by the need for a product and on the other by retirement of the product from service. The upper boundary is the highest level of functional requirement. At the bottom, it's the piece part, material, or process. Given this scope I press, once again, the importance of having or acquiring a reasonable understanding of engineering, manufacturing, procurement, deployment, and their relationships. *The lack of such knowledge is the most common cause of major CM failures*.

Experience

The following details the experience upon which this book is based.

I worked at the Pomona Division of the General Dynamics Corporation for 31 years. In the first 4 years I was exposed to every major department in the Division as a Management Systems Analyst. Then I was asked to move to Engineering Management Systems, which eventually merged with Documentation where I was assigned the task of figuring out this thing called Technical Data Management. Having done so, I was assigned the management of that activity.

When CM came along it seemed to be very much like Technical Data so they asked me to figure that out too. The result was appointment to the position of Configuration and Technical Data Manager where I remained until retirement.

There was considerable inter-action between the Pomona Division and other divisions of General Dynamics as well as other companies. There were other temporary assignments but only two have significance in this context. I directed a division-wide study on Design to Production Transition and another one on the probable impact of the future upon the Division's systems.

The Pomona Division came into being as part of Consolidated Vultee (Convair) to develop and produce a defense against the Japanese kamikaze (suicide) bombers of WW II. Among its accomplishments are the Terrier, Tartar, and Standard Anti-Aircraft Missiles for the Navy; the shoulderfired Redeye and Stinger Anti-Aircraft Missiles for the Army; the ARM Anti-Radiation Missile for the Air Force and Navy; the Phalanx Gun (Anti-Aircraft) System for the Navy; and many less spectacular items for one or another of the services.

As part of the great reshaping of aerospace during the 1990's, General Dynamics sold Pomona's product lines and closed the Division. However, most of the products mentioned above remain actively deployed and are supported by other manufacturers.

***

1. VIEW FROM THE TOP

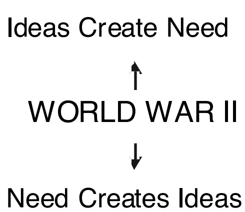
Configuration Management (or CM as we call it) was conceived in scandal, nourished on contention, and matured in ignorance. Nonetheless, done right, it works.

Scandal

Of course scandals are always with us but those of the 1950's and 60's were exceptional. They caught the attention of the media and the people. They rattled the Congress, which took deadly aim at the Military. The Brass, threatened with meltdown, was desperate for a solution. The Air Force led the way and discovered a real oddity. Bad products were seldom the result of conventional skullduggery. Instead they were usually caused by misunderstanding, miscommunication, or similar misadventure.

How could anyone start out to develop one weapon and end up with another? Even so, they did! Obviously, instructions were unclear, misunderstood, or someone was doing his own thing. As it turned out all three were at work.

It's important to know that long established documentation, identification, verification, release, and change control practices were in place. They had been developed by industry years before and they had worked well. They still did but something was clearly wrong. Where was the flaw?



By and large, prior to World War II, inventors created a device and then tried to convince the defense establishment that it was needed. If agreement was reached, the inventor financed further design, test, and production sometimes at his own expense and sometimes backed by others or a procurement guarantee. World War II convinced everyone that this scheme no longer worked. So a need-creates-ideas approach was adopted. The Department of Defense decided what it needed and then sought a contractor willing and able to design and produce it's fulfillment. Figure 1 displays the way it was.

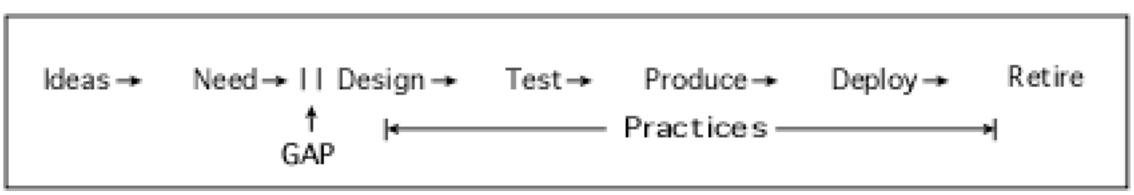


Fig. 1 The Gap

The weapon needed was selected from a group of ideas and passed on for design, test, production, and deployment. The practices began at the mid-point of design and continued throughout the life of the product. Changes, adjustments, corrections, etc. were made <u>using *design* as the point of reference</u>. This self-referencing approach allowed the design to drift or be pushed away from the need. The fix is displayed in Figure 2.

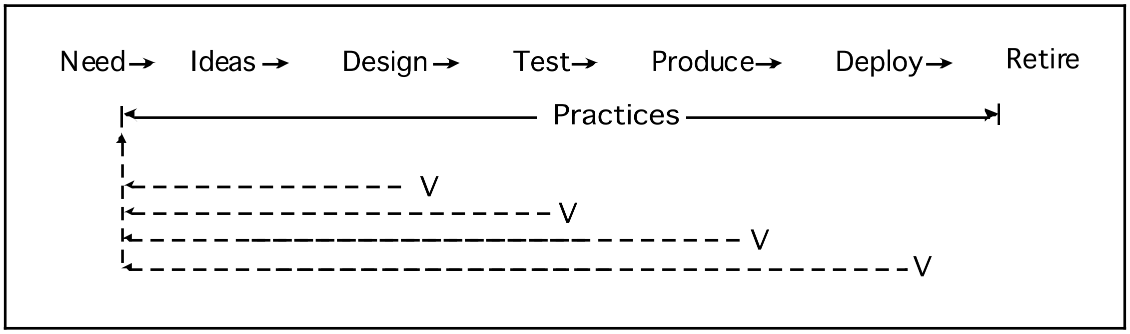


Fig. 2 The Fix

The practices start at the end of need. All subsequent changes, adjustments, corrections, etc. are made using *need* as the point of reference. This approach closes the gap and keeps the design aligned with need. To keep the gap closed, several verification points are imposed. At each point, the current work is examined to assure that it is still aligned with need. If it isn't, corrective action is taken or the program is cancelled. They called this approach Configuration Management or CM for short.

Of course there was more to it. There always is. System Engineering was elevated to a dominant position. The whole effort was wrapped in a scheme called Weapon System Management and there was more.

Contention

However, the proverbial "they" had invaded the provincial halls of Engineering and started internecine war. Design engineers maintained that System Engineering couldn't work and would lead to failure. System engineers swore that it worked marvelously well by keeping the designers in line. Both were ingenious in proving their contentions in practice. It was awkward and it took years to reach an uneasy accommodation.

Simply put the System Engineering task is to translate the need defined by strategists and tacticians into engineering terms suitable as (1) requirements to be met by detail design engineers, and (2) a frame of reference for determining that the design is aligned with need at various points throughout the program. It's a delicate task. Specifying too much hamstrings the detail designers and increases costs. Specifying too little fails as a point of reference and enables design drift. To add spice to the mix, millions of dollars are usually at risk.

To aid this task, the Military spawned dozens of requirements for specifications and wrapped them in complex procedures. It took a specialist to decipher them and all but an Act of Congress to change them. Of course, they were resented and the contention was vociferous.

The second affront was Weapon System Management. It was created to define an orderly, phased process beginning with the strategic and tactical studies that define need and ending when the product was retired from service. Generally, this period lasts 5-15 years or more during which people and conditions change. So another load of procedures was developed to cover most contingencies. To be fair, many of them were adequate and helpful when used properly. But some folks prefer contention to compliance so contend they did.

These conditions were further complicated by the implementation of Program Management and Fixed-Price Contracting.

Generally, defense contractors were organized by department; one for procurement, another for engineering, another for manufacturing, and so on. When more than one product was in work, coordinators, expediters, and project managers were assigned to specific products. Their job was to shepherd their products through the various departments to completion on time and within budget. As the number of products and degree of complexity increased, their job became all but impossible. The Military cure was Program Management. In this scheme, a Program Manager was made fully responsible for a product. He was given rank equal to that of the department heads plus control of the budget. His real power often exceeded that of a General Manager. In too many cases department heads and program managers mounted an offensive against each other. After all, careers were at stake! It was contentious and a number of years passed before an uneasy truce was achieved.

Cost-plus was a primary form of financing during WW II. Keeping it simple, cost-plus meant spend what you must to get the job done but your profit is limited. Fixed-price means that a specified amount of work must be done for a predetermined price including both cost and profit. The objective is to save tax dollars. Most likely it does. However, its impact on organizations was to shrink the money supply. Competition for funding became intense.

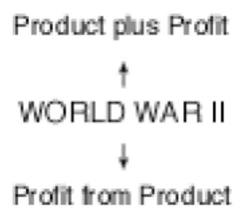
Then, as if all that was not enough, the computer revolution arrived. In the beginning it was used more often to demonstrate its wonders than to enhance the thing computerized. If something could be done, it was done, regardless of good sense. Systems blossomed. Applications flourished. New organizations burgeoned. And, they contended.

Ignorance

Ignorance isn't stupidity. Let's not cross those wires. It's a state of not knowing; an often ignored condition common in humans. The defense establishment is not immune.

CM must operate across an enormous spectrum of activity extending from product need to product obsolescence. The WW II Generation of managers rose through the ranks over many years. They learned the detailed activities of their departments intimately and came to understand the spectrum in which they lived. Unfortunately, as CM came into being, the WW II Generation was retiring. A new generation was taking its place.

Many of the New Generation came to power as college graduates in business and financial management imbued with "maximizing the bottom line". There was a notion at the time that "if he can manage one thing he can manage anything". So they gained management positions early in their careers without knowing the detailed activities of their departments. Of course, many of them had been exposed to the industrial spectrum in school but few of them had hands-on experience with it. They tended to view the details of plant operation as the responsibility of someone else. As a result, they were often unaware of the widespread consequences of their decisions.



The focus of the WW II Generation was the product. Of course, a profit had to be made but it did not dominate every waking moment. Ultimately, the focus of the New Generation was profit. Of course, there had to be a product but profit was the dominant force. Although this distinction may appear to be an insignificant subtlety, it is not! When profit overwhelms product, the ambiance of the plant changes! There are significant consequences not the least of which is corner cutting.

Specialists have a similar problem. Most people in the defense establishment are accomplished specialists of one kind or another. Generally, they are fascinated with their specialty and tend to be parochial. Even so, the WW II Generation Specialists had at least a passing appreciation of the industrial spectrum in which they operated. The New Generation Specialists did not. Things like CM or industrial spectrums were simply too intangible, even too ethereal, to warrant serious attention. However, successful CM is dependent upon their willing cooperation.

Of course all of these conditions didn’t occur at once. However, they appeared, one hard upon the heels of another, and lasted until many of them co-existed. It is important to note *and to remember* that the people involved were not bad people. Each one came at these problems from his perspective and acted according to his perception. However, their perspectives were limited and the problems were vast. Limited perspective continues to be the most common obstacle to understanding CM.

The resulting turbulence engulfed CM. From time to time it became the prisoner of warring engineers, competing empires, shrinking budgets, or careerist power plays. Its development was warped, stunted in some cases and overblown in others. Ultimately, it survived only because of the intransigent demands of the Military Brass. Design drift had to go! Real need had to be met!

***

2. WHAT IS IT?

Give or take a tweak, the following definition of CM has appeared in many places for many years.

> A discipline applying technical and administrative direction and surveillance to (1) identify and document the functional and physical characteristics of a configuration item, (2) control changes to those characteristics, and (3) record and report change processing and implementation.[1]

A more modern version follows.

> A process that establishes and maintains consistency of a product's attributes with its requirements and product configuration information throughout the product's life cycle, [2]

Neither definition is really satisfactory. If the user has the knowledge necessary to understand them he doesn't need them. On the other hand, if the user lacks that knowledge they are of little help.

CM is difficult to define because it borrows from many preexisting fields and applies the borrowings purposefully across an enormous spectrum of activity. The most common definitions focus on Documentation and Change Control because they are tangible and most people are already aware of them. In fact, the association is so close that many people come to believe that CM is just another name for Documentation or Change Control.

Another approach to definition is based on a simple question. What's different? What do we do now that we didn't do before CM was established?

We had documentation, identification, verification, release, and change control before CM was ever heard of. They were loosely collected under the rubric of Documentation. We have them now under the rubric of CM and we use them to define, direct, control, and record. So what is really different?

Before CM they were used to stay in line with the design. After CM, they are used to stay in line with the need. Thus, ---

---

[1] Army AR 70-37, Navy NAVMATINST 4130.1A, Marine Corps MCO 4130.1A, Air Force AFR 65-3, Defense Supply Agency DSAR 8250.4, National Security Agency NSA/CSS 80-14, Defense Communications Agency DCAC 100-50-2, and Defense Nuclear Agency DNA INST 5010.18; issued 1 July 1974 by the Joint DOD Services/Agency.

[2] EIA-649-A, 2004, by the Government Electronics and Information Technology Association.

> CM is a discipline that keeps an evolving product aligned with the need for it.

Most people can readily understand that definition. If not it can be explained quickly and simply. (See: Section 2, Basic Text).

Originally, CM was presented in three parts (a fourth was added later), which are often included in definitions. They are academic in style, jargonistic in language, dependent on prior knowledge, and hard to understand. However, they should be acknowledged and correlated with appropriate parts of this book.

> Configuration Identification: The Documentation of product characteristics or the process of developing such documentation. Correlates with: Documentation, Identification, and Verification.
>
> Configuration Control: Evaluation, coordination, approval, or disapproval of changes, deviations, and waivers of product characteristics. Correlates with: Change Control and Verification.
>
> Configuration Status Accounting: Appropriate records of the Configuration Identification, the status of proposed changes, deviations, or waivers, and their implementation. Correlates with: Release and Change Control.
>
> Configuration Audit: Verification of compliance with the Configuration Identification. Correlates with: Verification.

It would not be surprising to see a new definition before long; something like the following:

> Configuration Management is the collective name for the following practices: Documentation, Identification, Verification, Release, and Change Control.

Note that alignment has disappeared. The argument for omission is that, by now, everyone knows that alignment is necessary so there is no longer any reason to emphasize that fact. In common usage the term “CM” is a collective name for the listed practices. It is better to conform to current usage than it is to press an esoteric approach that we don’t need.

Regrettably, this was the state of things in the 60’s and 70’s when CM was born. We used all of the practices and we knew that alignment was necessary but, too often, it did not happen! Why not?

Until the cause is identified, understood, and eliminated it is imprudent at best and absurd at worst to discard the safeguards established with such difficulty.

***

3. THE DISCIPLINE

Locate applicable engineering requirements.
Perform work in accord with those requirements.
Document the work.
Identify the work.
Verify compliance with the requirements.
Release the documents.
Change Control the documents

Fig. 3 The Elements

The elements of CM have different names in different places but the point is not what they're called but what they do. Each one has its own standards and purposes. They were well established by commercial industry years before the advent of CM because they are necessary for the success of any sizeable industrial project. They were adopted by the military for its industrial projects for the same reason. When the Air Force created CM in the '60s and '70s, it relied on these existing practices for the same reason. Over time, they have become so closely associated with CM that many regard them as "CM Practices". The term implies ownership however CM does not own them! It borrowed them and it shares them with many other activities.

One can go on at great length about each one of them. But it is best to start with simple straightforward definitions based on what they do.

- Requirements specify what has to be accomplished.
- Work accomplishes the things specified.
- Documentation communicates to people and/or machines now or in the future.
- Identification (numbering) distinguishes *precisely* between one thing and another,
- Verification proves that a documented design meets its requirements.
- Release, records verified items and makes them available for use.
- Change Control maintains the integrity of released items.

Although these practices, individually and collectively, are not CM they are essential to its successful operation.

Requirements

The concept of requirements befuddles some people although it's really not that complicated. A requirement is neither more nor less than the statement of something that must be done. There are many kinds of requirements: Engineering, Manufacturing, Procurement, Customer, Contractor, Financial, Schedule, and more. The number and variety may be the source of confusion. Most of them of them have some administrative impact upon CM and that fact may cause the confusion.

The requirements for which CM was created are the Engineering Requirements for a product. The initial set is derived from the work of strategists and tacticians. They are usually documented in a System Specification. All subsequent Engineering Requirements are derived from the initial set and become more detailed as the program proceeds. Ultimately they become the details in Engineering Drawings from which the product can be manufactured.

Work

There are many kinds of work but the work that CM is concerned with is that which is necessary to fulfill the requirements.

Documentation

The purpose of documentation is to communicate information to people or to machines, near or far, now or in the future. The form it takes should be the one that best communicates information to the intended user. Usually it will be text, a list, graphics, a combination, or computer code. The medium used should be accessible, legible, reproducible, and durable. Common mediums include paper, vellum, glass, computer tape, disk, and chip among others.

Documentation has been around since the pyramids. It evolved over time to include formats, conventions, customs, and traditions thought to improve communication. Each currently used method has a vociferous constituency devoted to it along with clamorous critics intent on changing it. And, they fight – until an agreement is forged – whereupon they fight about the meaning of the agreement.

Even after the most meticulous efforts to be clear, accurate, and adequate some user somewhere will misinterpret and worse have a logical reason for doing so. No, computers won't solve it. They do eliminate some problems as they create others. The reason for the hullabaloo is not arrowheads or paragraph headings. It's the human problem of clear communication; a puzzle of perspective and perception not easily solved.

The kinds of documentation are almost limitless. Figure 4 displays the types most often produced during each Phase of a program.

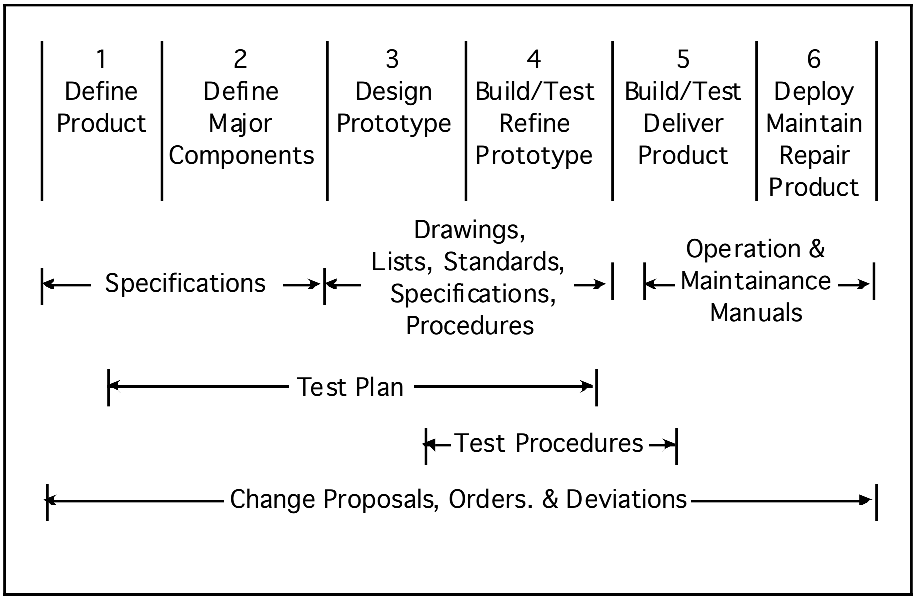


Fig. 4 Typical Types of Documentation

Generally, it is wise to follow the customs and traditions prevalent in the industry involved unless there is a *compelling* reason to do otherwise. Change in this field comes slowly and at significant cost.

The information to be conveyed is the province of various kinds of engineers and technical specialists. Form and format is the job of draftsmen, checkers, technical writers, editors, and now-a-days computer programmers.

Technical Data Management (TDM)

When CM borrowed Documentation as a tool it accepted the rubrics of Technical Data Management (TDM).

Technical data became a discipline because contractors raised unmitigated hell about two things. First, too much very expensive and often useless data was being ordered. Second, the Military was acquiring and compromising contractor proprietary data that it had no right to. Investigations revealed enough truth in the charges to require significant changes. The interesting and complicated subject of proprietary data is best left for another time and place. It is primarily a legal problem. The kind and amount of data ordered is another matter.

The Military adopted a Form (DD 1423, Contract Data Requirements List) upon which all data ordered was to be listed and priced. The Form was made part of the contract. If data wasn't listed on the form, the contractor didn't have to produce it. These changes required people to manage and enforce them and TDM was born. To "simplify" matters, TDM prepared Data Item Description sheets specifying content and format requirements for each kind of data ordered. These sheets soon became the acceptance criteria for data being delivered and a whole new order of complexity blossomed.

There are few jobs harder than writing an adequate Data Item Description. Each item has its own vocal constituency within the Military and in each major contractor's organization. None of them see things the same way. Nonetheless, the sheets get written and issued.

Clearly, the Military needs data that (1) details the progress of a program, (2) permits acceptance, reprocurement, competition, operation, and maintenance of the product, and (3) prevents trouble from developing. All of these are difficult, but preventing trouble is a lollapalooza. The scenario works like this.

Something goes wrong! An investigation assigns the cause to the fact that this, that, or the other was not documented. The corrective action requires such documentation in the future and a new Data Item Description is added to the book. When the next contract comes along, the item is sitting there just waiting to be ordered, so someone does.

Software came along just in time to be entrapped. It was new and evolving. Too many people knew too little about it. They made mistakes. Corrective Data Item Descriptions flowed like water in the tropics. Many of them dealt with programming control. Soon there were so many that something was needed to manage them and there sat CM. To the inexperienced, it looked like the perfect tool and to many it still does.

However, *when CM is used properly it applies a requirement and verification methods but not how-to achieve it.* The how-to belongs to others. Too often software applies a requirement, verification, and how to achieve it in an attempt to control the programmers. Thus, the mangled mess that commingles CM, programmer control, and excessively specific Technical Data Requirements was created and remains to be untangled.

Technical Data Management and CM have much in common. Nonetheless they are separate disciplines. For TDM, the overall objective is effective communication. For CM, it is direct alignment with need. CM uses only some of the documents subject to TDM while TDM relies on CM to help produce only some of the documents it oversees.

Technical Manuals

Technical Manuals deserve special mention because they seldom get adequate attention. Historically, designers designed the product. Then Maintenance Engineers figured out how to maintain and repair it. They documented their conclusions in Operations and Maintenance Manuals intended for use by field personnel.

Weapon System Management seeks to integrate maintainability with the design process.
It's difficult! Maintenance Engineers are generally turned off by the theoretics in the early phases. Typically, designers have little patience with maintenance issues. The trade-offs between Reliability and Maintainability are complicated and unglamorous. Typically, most designers are convinced that their design will not break and do not become

fully engaged until it does. Of course, that's too late. Repair in the field can be impossible. Redesign for maintainability is enormously expensive.

The solution of course is a broader understanding of the whole enterprise by more people. This is unlikely in an age when specialization is king and broad but shallow knowledge is dismissed as useless. CM can do little about the situation but cope with its consequences.

Operation and maintenance requirements, including reliability and maintainability, can be specified in the System Specification (Phase 1) and Component Specifications (Phase 2), accomplished by designers (Phase 3 & 4), documented in manuals (Phase 5), for field use (Phase 6). However, it's very difficult to verify compliance with the requirements in any objective way until some years of field use have past. The most effective method so far has been high-powered technical review by people with enormous field experience.

Technical Manuals are seldom included in the regular release record so it is easy to omit them in the Change Control process. However, if an adequate, accessible release record is otherwise available and if Maintainability is a real requirement in Change Control, it's quite possible to keep them integrated with the rest of the documentation. It is not unusual for the actual release to be made by the organization that prepares the manuals, usually technical writers.

Changes to the product can impact maintenance of the product. Therefore, it is important that they be examined for Maintainability during the Change Control process to assure that Manuals are revised if necessary.

Pure Manual errors can be corrected without any impact on the product. However, error may have occurred because of unclear product documentation. If so, it is appropriate for technical writers to propose clarification changes to the product documentation to correct the condition.

Identification

The purpose of identification is to distinguish one thing from another *precisely*. It can take the form of a name, number, letter, symbol, mark, or some combination of these provided that it is *unique*. Uniqueness is essential to prevent confusion with similar items. Use of the wrong thing is expensive and sometimes dangerous.

There is always some ingenious fellow out there inventing a new "Significant Numbering System". Each digit stands for something such as plant, company, date, product, etc. They can tell you more than you ever wanted to know about the component *if you know the code*. They also require a special assignment and cataloguing system, which can be both cumbersome and costly. The alternative is a "Non-significant Numbering System" in which each digit means nothing but sequence. Assignment and use are easy.

The military assigns a numerical prefix (Code Ident, CAGE Code) to each agency and contractor. It is used to keep their numbers from duplicating each other. Commercial ßcompanies often rely on the company name or trademark as a prefix for the same purpose.

As with documentation, there are endless identification schemes and advocates. Simple but adequate is best. A non-significant sequential document number and revision letter or number has had the best long-term result. However, the wise course is to follow the customs and traditions prevalent in the industry involved unless there is a *compelling* reason to do otherwise. Change in this field is also slow, confusing, complicated, and costly.

The more common identification methods are noted in Table 1.

Companies and Agencies	Code Ident. or CAGE Code	12345
Commercial Organizations	Company Name or Logo	BVD
Documents	Numeric or Alphanumeric	34567825
Drawing Revisions	Revision Letter	A
Other Document Revisions	Revision Number	Rev. 1
Change Proposal	Numeric or Alphanumeric	C34681
Deviation	Numeric or Alphanumeric	D2124
Hardware or Software	• Document Number & Revision Letter of the document used to create it or Part Number referenced to the Document Number. • Serial Number if applicable. (Alternatively a manufacturing lot number may be used.) • Deviation Number if applicable.	34567825A DQ 219890 S/N1111 D2124

Table 1 Common Identification Methods

Figure 5 displays one possible hardware identifier from Table 1.

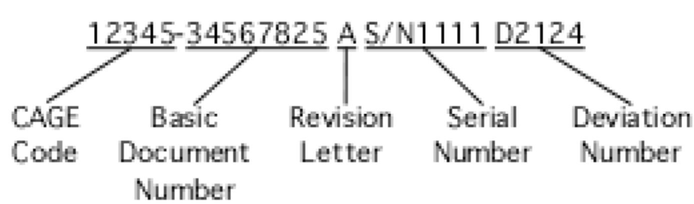


Fig. 5 Sample Identification Number

One of the obvious problems is identifying microminiaturized parts with this amount of data. New methods are constantly being developed. Meanwhile, the most common method is to keep the part in a container marked with the identifier until it is placed in its next assembly. Afterwards deduction from the next assembly drawing is the practical method of identifying an item.

The identification system in use is usually defined in a standard applied by the originator of the document and audited as part of release. The manufacturer marks the identification on the part. Regardless of method, the system must be able to distinguish *precisely* one thing from another. And that brings us to "reidentification" and "interchangeability".

Reidentification & Interchangeability

The decision to reidentify or revise is an integral part of Change Control but the form that it takes is part of Identification

> When a changed item *will not work* *in all* of its next assemblies as well as an unchanged item of the same number, the changed item is not interchangeable and must be reidentified. The Basic Document Number must be changed.
>
> When a changed item *will work* *in all* of its next assemblies as well as the unchanged item of the same number, the changed item is interchangeable and must not be reidentified. Only the Revision Letter is changed. These changes are often called revisions or versions.
>
> Excruciatingly complex definitions of "will work" and "will not work" are available and sometimes helpful. However, the only thing that really works is the informed judgment of knowledgeable people able and willing to understand the use and user of the item.

Wouldn't it be simpler to change the Basic Document Number on everything and avoid the fuss about what does or not constitute a revision or version? Yes, it would on one end of things but not so for the user. In the field, revisions are interchangeable. Reidentifications are not. This information is vital in keeping weapons working. It could be conveyed in other ways however the fuss about what does or not constitute a version would remain. This identification problem backs up throughout the whole system of ordering and stocking spare parts. In that world they don't track versions because any one will work. But they keep careful track of reidentifications because just anyone will not work!

There's a lot more to it. However, for our purposes, the key elements are *uniqueness* and *preciseness*. One thing must be distinguishable from another — *precisely*.

Part Numbering

Some companies follow a tradition of identifying documents and the items they describe separately. Thus, they have Document Numbers and Part Numbers. The Part Numbers are cross-referenced in the document. Other companies use the same number on the document and on the part.

Version

In common English, a version is a slight variation from an original. The term includes alteration, modification, variant, issue, edition, revision, letter change, and anything else that can be considered *a slight variation*. For CM, slight variation means interchangeable; *a version is an interchangeable variation of an origina*l.

Determining interchangeability, particularly functional interchangeability, is an engineering task! The function of upper level assemblies is usually stated in the documentation that defines them. The function of lower level assemblies and parts is inherent in their design and often must be deduced from it.

Unfortunately, there are many ways to identify versions. Generally, versions of drawings are identified with letters of the alphabet. Versions of specifications are identified with Arabic numerals. Versions of software are identified with a variant of the Dewey Decimal system. Versions of other documents may be identified differently but similarly. Once again tradition confronts us. Each identification method was developed by those who controlled the documents affected; separate communities of specialists working in splendid isolation. It is tempting but unwise to call for standardization. The legacy is too broad and too deep to be abandoned easily. The cost exceeds the benefit. So, if you must work with the detail, either rely on a specialist or learn the local conventions.

Version identifiers are recorded in the revised document along with some indication of what has been revised. For drawings, a simple description of the change is recorded in the drawing revision block. For text documents, it's often shown on a "revisions page". Generally, version identifiers are marked on the part.

References in a document to another document do not include version numbers in order to reduce changes and in accord with the general working rule, "Always use the latest version". (After all, they are interchangeable, aren't they?) .

As to drawings and the items they define, if there is an original plus two approved versions of Item 840, the identifiers will be 840, 840A, and 840B. If another interchangeable version comes along, the new identifier will be 840C. However, if that version is not interchangeable, it must be reidentified; that is given a new and unique number.

Reidentification

If the version that would have become 840C is not interchangeable with all other versions of 840, it must be reidentified; given a new unique number AND *it is no longer a version*. It is a new item! The new number will be the next one in order from a log of unused numbers. (A number once used is *never* reused! Even if the document it identifies is cancelled, the number remains to identify the cancelled document.)

Reidentification is required not only for Item 840 but for every item above it until the level of interchangeability is reached.

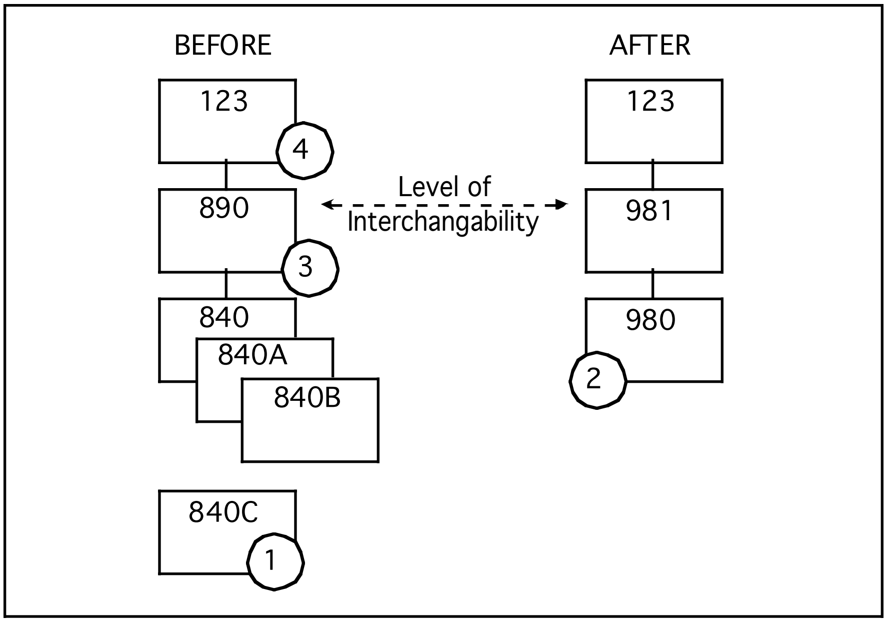


Fig. 6 Reidentification

1. 840C is not interchangeable with the other versions of 840, so -
2. 840C is reidentified to new part 980.
3. Obviously, 840 and 980 are not interchangeable, which forces the creation of 981.
4. Fortunately, 890 and 981 are interchangeable so 123 is not affected.

Changes like this cause delay and expense. A product can have 16 or more levels. Reidentification can ripple upwards from bottom to top and affect every one of them. When it does it can impact at every level in almost every department of the plant. Thus, people turn themselves into pretzels to avoid it.

They have developed a whole folklore of excruciatingly complicated refinements, such as one-way interchangeability, of these fundamentals. This doctrine holds that you need not reidentify a version when it is interchangeable forward (in new units of product) but not backward (for repair of old units); provided that there are no old units to be repaired. The theoretical logic is impeccable. However, the Devil pays no homage to theory. He will find old units, somewhere, that *must* be repaired and you will be out of luck!

These so-called refinements produce almost endless controversy and occasionally real danger. Thus, if you don't know each and every circumstance thoroughly, don't fool with the fundamental. Each component, and its defining document, must have a unique intelligible identifier! If non-interchangeable components have the same number, reidentify one of them! Reidentification must continue upward until the level of interchangeability is reached!

The people who usually make reidentification decisions are the members of the Change Control Board heavily influenced by engineers. Unfortunately, many engineers are very narrowly focused specialists. They know little or nothing of the consequences in manufacturing or the field that can result from lack of reidentification. The solution is to educate engineers about the real and far-flung consequences of their decisions.

Model

Model names or numbers are an old form of identification. It's still widely used. However, there is very little that is uniform about it. Every company and agency that uses Models seems to have a different set of rules for them. So, CM does not use them although there have been exceptions.

As a practical matter, Model identifiers have become the property of marketers both in and out of the Military. Most organizations see little benefit in marketing the same old model year after year so they change it with much "new and improved" hoopla. Conversely, deep within the Congress there are plenty of fixed attitudes. When they do not favor something new, Model numbers don't change. Interchangeability has very little if anything to do with it.

Therefore, model numbers should not be used as CM identifiers. A product may bear both model number and part number. The model number may be used effectively for some inventory control purposes in some systems but the part number is the only positive identification.

Part Marking

Instructions for placing the identifier on the item to be identified are stated in the document that defines that part. They include the actual number to be used, the specific location on the part where it is to be marked, and the method to be used such as stamping, etching, non-conductive ink, etc. Common practice requires version identifiers to be marked on the item.

The miniaturization of parts has created parts too small to mark by conventional methods. The first recourse is "bag & tag". The part number is written on a bag, which contains the part, or the part number is written on a tag, which is tied to the part. Obviously, when the part is next assembled the bag or tag is removed and the identification is lost.

The second recourse is microscopic marking methods. This is an expensive process practical in only the most critical cases. Therefore, it is seldom used.

The third recourse calls for other techniques. For assemblers, a tightly controlled assembly line is employed. The parts are placed in individual bins at each assembly station. The part number is marked on the bin. The assembler is responsible for getting the right part in the right place. Assembly testing is often used to assure that the right part was assembled.

Generally, repair of microminiaturized assemblies is not economical so the whole assembly is replaced. In those rare cases where repair is required, the technician compares the part in question with the assembly drawing to determine its identity.

None of this is picayunish. Getting the right part in the right place is essential! However, it does bring another principle to the fore. *Don't spend to prevent that which is cheaper to correct*. Just don't forget that the cost of correction includes, time lost, damage to reputations, and *above all else – human safety*!

Serialization

A serial number is an additional identifier used to identify each individual unit of an item (part). Example: Part 123, S/N 29 means the 29th unit of item 123. It can be applied to a product or any part of a product subject to the Part Marking constraints discussed above. Serial numbering should not be used simply because it can be done. It is one more cost and delay in production. However, if there is a real need to distinguish one unit of a part from another unit of the same part, serial number is the best method known.

Alternate Parts

Just as more than one kind of light bulb may fit a socket, more than one part may work in an assembly. A part shortage in Manufacturing or Maintenance can spark a hunt for alternate parts that may be available from more than one vendor or from other products. An alternate part, if interchangeable, can be added to the released documentation through an approved change or deviation. The specific method of documentation and release of alternate parts depends upon the local practices of contractors, industries, and buying agencies.

Configuration Item (CI)

The CI originated as part of the allocation process but soon mutated into an "acceptable" technique for moderating Military control. It is defined as "an aggregation of hardware or software, designated by the government for Configuration Management". Translation: a part of the product that the Military really wants to control. (By inference all other parts are free of military control.)

A CI has its own CI Identifier (number) in addition to document and part number. It is designed in accord with a Military controlled specification, which means initial approval of the specification and change control thereafter.

The CI is best understood as a technique, useful to some and overdone for others, depending upon the parochial practices of the contractor and the buying agency. Where it works, use it. Where it doesn't, avoid it. Either way, it does not alter the basics of CM!

Verification

The purpose of verification is to prove that an item complies with the requirements for it. That item may be documentation or hardware. Documentation includes software documentation. Hardware includes the software used in it or with it.

Proof

The only certain Verification is use of the actual product to satisfy the specified need. For a weapon, that means use of the weapon in war. For a commercial product, it means use of the product by a large number of consumers. Thankfully, wars are not always available to verify weapons. Obviously, it is imprudent to wait for a commercial product to be marketed. So, the next best thing is devised to prove the work at various stages of the program.

Proof is that which convinces the mind. So it is not surprising that there are almost as many methods of proofing, as there are items to be verified. Most commonly, Technical Compliance Review, Checking, or Editing are used to verify design. Inspection (Examination or Test) is used to verify hardware. However, the acid test is to actually build hardware from the documentation and then demonstrate that the hardware meets the need.

A Technical Compliance Review consists of competent people, familiar with the technology involved, but not part of the design team. They examine the item against its requirements and reach a conclusion that the item does or does not meet them.

Checking and editing are age-old methods dealing as much with communication as content. However, they do examine technical coherence and compatibility.

Inspection (Examination or Test) is a well-established Quality Assurance method of long standing.

A successful demonstration, conducted near the end of Phase 4, is the crowning achievement of most development programs. The intent is to demonstrate satisfactory performance of one or more prototypes in the field. Every reasonable effort is made to duplicate the operating conditions that the product was designed to meet. Verification is usually performed by a wide variety of technical types selected for their knowledge of the technology involved or the verification method used.

Figure 7 displays the most common methods in relation to program phase.

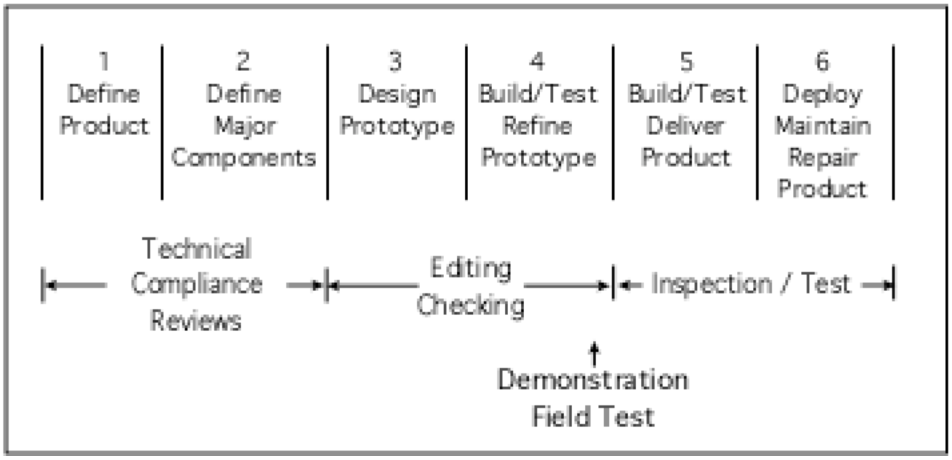


Fig. 7 Common Verification Methods

Technical Compliance Review

Of all the verification methods available the Technical Compliance Review is the most difficult to conduct successfully. It would seem to be simple. Gather a group of technically competent people. Give them a copy of the applicable requirement. Make the current work available to them. Ask the question, "Does this work comply with these requirement?" However, unless they have a chairman with iron control they will never get to the compliance question. Instead they will come up with every possible change that in their opinion will *improve* the product. Many of these ideas will be practical and attractive *but they are diversions*!

Product Improvement Reviews, also called Design Reviews or Technical Reviews, are good and useful techniques for getting the best possible product from the engineering effort. Compliance Reviews are not. They are conducted not to improve the product but to see if the product in its current state complies with the requirements for it. These are very different activities. However the same people perform them. Their usual work is creation or improvement of product design and they will revert to that mode of behavior whenever they have the chance without even being aware of it. Only a strong Chairman can keep them focused on compliance long enough to get an answer.

Do not take this matter lightly. It is one of the primary causes for the failure of Technical Compliance Reviews to detect errors that must be corrected.

Documentation Standards

Compliance with documentation standards imposed by Technical Data Management is not part of CM Verification. Thus, "verified document" is a shop term (slang), which refers to the content of the document rather than to its form and format. Draftsmen, checkers, technical writers or editors determine compliance with form and format standards as the documents are created.

However, these same people examine the documents for errors and inconsistencies. This part of their work is considered to be an adequate CM verification for Prototype documentation primarily because there is no other practical method available at that point in a program.

Even so, grizzled gray-heads usually say that the only way to prove a documentation package is to build it and do so more than once. In my opinion they are right particularly if the documentation package is intended for use in competitive reprocurement.

Configuration Audit

Of the verification methods available, Configuration Audit became the most expensive for the least value added. In simple terms, it is a customer-imposed practice that duplicates verifications already completed. This consequence had two causes. First, formal CM definitions were ambiguous enough to permit it. Second, the performance of verification, conducted by both military and contractor personnel, was poor enough to require it.

Unfortunately, large organizations are prone to add another unit of people or duty rather than correct those that are failing. Thus, instead of audit being treated as a synonym for verification, it became a cause célèbre with a life of its own.

In very simple terms, a Functional Configuration Audit verifies that finished hardware complies with the applicable Functional or Allocated Baseline; that is, it functions, as it should. A Physical Configuration Audit verifies that finished hardware complies with the applicable Product Baseline; that is, it was built correctly.

Generally, Configuration Audits are conducted on the first unit of production hardware but some other unit may be selected. Functional Audits became unpopular quickly because they require examination of lots of test data and that gets dull fast. They usually get short shrift. Conversely, Physical Audits became quite popular because they capture lots of attention and can be dramatic.

A physical audit consists of disassembling a finished product piece by piece until the limits of non-destructive disassembly are reached. Each part is examined against its documentation to determine conformance. Discrepancies require correction of the documents or rework of the hardware. All well and good except that *it should have been done through verification much earlier in the program*. Further, much of the hardware can't be checked because disassembly destroys it.

Thus, Configuration Audit is not a technique to be used routinely on every program. The better practice is to see that Verification is properly conducted in the first place. If that effort fails, use audit to recover.

Release

The purpose of Release is to make *verified* documentation available to down-stream functions and maintain an accurate record of it. Before Release, a document is deemed to be a working draft, error prone, incomplete, and changeable without notice. After Release the document is deemed to be authentic, error free, and changeable only with notice. Release stands between Engineering and Manufacturing, Procurement, and other major functions. It protects downstream users from investing time and money based on invalid information. However, schedule pressure causes most organizations to permit some form of “Advance Release” (sometimes called “Pre-release”) which means, subject to change - use at your own risk! An Advance Release takes place before the documentation has been verified.

The release act consists of (1) placing a release symbol on the master of the document (so that all copies will show it) or making a “released” entry in a widely accessible record and (2) making the document and record available for distribution. Masters of Advance Release documents are marked or the record shows “Advance Release”. They are replaced by the verified documentation release as soon as it is available.

A record of the document is made before it is released. Generally, the document identification, date, author, verifier, and where and when the document is to be used are sufficient. However, the advent of computers has greatly complicated the process. Release is now expected to record many kinds of information for dissemination through integrated systems to widely dispersed users. So a release is not made until all of that data is available. The process needs careful management to avoid creating an enormous choke point.

The master (or an equivalent) of a released document is placed in bond and jealously guarded thereafter to prevent any *unauthorized* change.

Release is really quite deceptive. The act itself is exceedingly simple however the record generated is an enormously complicated database widely used throughout the organization. The need for information is usually acute and the pressure for timeliness and accuracy is remorseless.

Release clerks usually perform the release task, which consists of the following parts:

<u>Approval</u>

Originally the only approval required for release came from the engineer who created the document. As complexity increased mistakes became more costly so management added specialized approvals such as reliability, materials, etc. to prevent error. These approvals can easily grow to 10 or more and degenerate into last minute signature collection and wrangling that has almost no value. It is also an oft-cited bottleneck. Such approvals should be integrated either with the work of the phase or with Verification. *The only approval required by the CM Discipline is Verification*. However, it is common to also require Technical Data Management clearance as to form and format to assure usability.

<u>Recording</u>

Originally, the name and number of the document, the name of the engineer approving it, and the date was the only record made. With the advent of computers it became possible to construct databases containing much more. Release was the obvious spot to capture the data. Thus, the release record has become a critical central source of information for most functions in the plant and for many in the Military. However, such added data is ancillary to the CM Discipline.

In addition to verification, the Release Record data *required by the CM Discipline* is the identification of documents released (including the version), where they are used (next assembly), and the effectivity of the release. Identification was discussed above. Where-used and Effectivity are summarized below.

<u>Where-Used (Next Assembly)</u>

"Where-used" is slang for "The assembly(s) in which the item is used".

Normally, parts are assembled into assemblies and assemblies are assembled into other assemblies and so on until the top assembly is reached and the product is completed. Each assembly is considered to be the "next assembly" of the parts, materials, and assemblies it contains.

> For instance, a doorknob is a part, which next assembles into a doorknob assembly, which next assembles into a door, which next assembles into a house. Thus, the house is the next assembly of the door. The door is the next assembly of the doorknob assembly. The doorknob assembly is the next assembly of the doorknob.

Of course, it's often not that simple. Parts and assemblies can be used in many different next assemblies. Thus, the "where-used" data for doorknob #4040 could be doorknob assemblies #9122, #8043, #3060, and #9203. But it doesn't stop there. Next assemblies #9122 and #8042 are used in House #50126, while #3060 and #9203 are used in House #60131.

These data and their relationships are recorded in the Release Record for later use in Change Control. If a change is proposed to the doorknob the change must be acceptable for all four next assemblies or it must be reidentified.

<u>Effectivity</u>

"Effectivity" is "when the item is to be used". Effectivity can be expressed as a calendar date, an event, a quantity of hardware, a lot number, or as a unit serial number. The most common manner is unit serial number. But most next assemblies are not serialized. What then? For this purpose, they are treated "as if" they are serialized.

> A new version of #4040 is to be introduced as 4040A. Effectivity has to be established for each usage (next assembly). However, in this case it will be used in only one next assembly until its desirability is established.
>
> Figure 8 shows that ten units of 8043 are on order in the factory. Two of them are complete and two of them are in work so it is most economic to set the effectivity

of 4040A at 5 and-on for 8043. Factory systems will pick up this information and translate it into whatever system they are currently using. Most likely they will call for a quantity of 4 containing 4040 and a quantity of 6 containing 4040A,

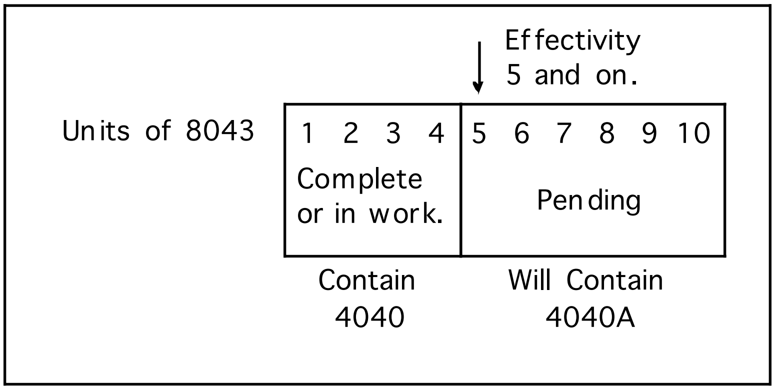


Fig. 8 Effectivity

The phrase "and-on" means "every unit that follows until otherwise directed".

If all of this seems somewhat confusing rest assured that it is but, happily, we don't have to cope with it. Manufacturing Control calculates effectivity and they are quite good at it. Effectivity, however expressed, is recorded in the Release Record.

<u>Bonding</u>

Formerly, the original of the recorded document was placed "in bond"; that is locked up. Copies were made available to users and authors. The objective was to prevent unauthorized change to the original. However, modern reproduction can produce a duplicate original indistinguishable from the real thing. Therefore, "bonding the original" is an obsolete practice.

Still, unauthorized change to the released document must be prevented. Any legible, reproducible, bonded copy can serve the purpose including digital media provided that humans can easily read it. This bonded copy becomes the "master".

Note: Technical Data Management may impose archival requirements for durability and reproducibility of masters. However, though important, they are ancillary to CM.

<u>Distribution</u>

Any economic, efficient method that makes true copies of the "master" available to users is acceptable. However, it is important that timely notice of a release reaches actual users. Such notice is doubly important when releasing a change. Notice may be accomplished by distributing the released documents or by any other timely means that alerts users.

The release record is also made available, usually via an automated system. Sometimes such systems stand alone. More often they are integrated with, or at least connected to, various management systems such as inventory or material control.

Pre-release

Originally, the use of a document prior to release was cause for dismissal. Compressed schedules have eliminated the rule. Many companies allow the use of clearly marked "pre-release" copies *at the risk of the user*. The risk is *change* of the pre-released documents *without notice* to the user. Pre-release documents may also contain *substantial error*. Thus, limited planning is a reasonable risk. Large quantity purchase orders are not.

Some companies have installed separate "pre-release control systems", an utterly foolish, deceptive, and expensive practice. *A document is either released or it is not!* If it is not, all of the safeguards imposed between the creator and the user are absent. If you really don't need those safeguards, abandoning them is safer than going around them. Yes, change orders can be pre-released but with even greater caution. It is a dangerous practice because people cannot seem to resist the urge to implement them at once.

If the receivers of pre-released documents are mature, experienced, and well supervised the practice works rather well for advance planning. However, if their attitudes are adolescent and their supervisors are gung-ho they are better off without the temptation to damn the risk and proceed at full speed ahead.

Configuration Accounting

The Release Record is sometimes referred to as Configuration Accounting, a formal term from original CM definitions. Technically speaking Configuration Accounting includes the information called for in the release record plus the status of proposed changes, and sometimes more. However, the specific elements vary from time to time and place to place.

Efforts have been made to specify and standardize the elements to be required. Standardization eases automated transfer of such data between contractors and agencies. However, the ease of transfer must be weighed against the colossal disruption caused by imposing such a standard on contractors and agencies of every hue and stripe. Contractors, industries, and products cannot be standardized routinely! The elements required for an effective record will continue to vary.

Therefore, standardize where and when practical but do not let the compromises essential for standardization distort the elements needed by the particular contractor, industry, or product. The price of doing otherwise exceeds the benefit to be achieved. *The objective is effective CM not standardization.* Again, one size does not fit all.

The following cautions apply to any Release Record or Configuration Accounting System.

- Don't assume a need to know everything about everything. Record the elements that will most likely be needed *and used* during the life of the product.

- Enforce existing basic systems; don't reinvent them.
- Use proven rather than leading edge automation technology and equipment.
- Discard "nice-to-haves" when requirements, costs, or schedules are impacted.
- Watch out for "just-might-as-well" features. They always cost more than predicted and often are never used.
- Keep it simple. Complex systems require costly maintenance and sophisticated operators. They are hard to modify. *Automation is an aid not an objective or a religion.*
- Keep the system user friendly or it won't be used!

Traceability

Despite best efforts, bad parts occasionally get into good hardware. Once known, the problem is to find them. For most commercial items, that task is left to the customer. Replacement is covered by a wide variety of warranties. However, given product liability developments, that course is not always wise. For military items, the task is a joint effort of the contractor and the military custodian of the delivered products. Usually, the task is urgent because bad parts cause malfunctions which can generate a disaster overnight. So, there is always pressure for "a better system".

Beware! Better systems generally require serialization of everything and retained assembly records stating which serially numbered part went into which serially numbered next assembly all the way up to the serially numbered top assembly of the product. It sounds easy, fast, and cheap? It's not! Furthermore, miniaturization prevents serialization of many parts.

A decision to adopt such an approach rests upon the probability of having bad parts assembled, plus the current ability to find them, plus cost – versus – the efficiency and cost of the more elaborate method. The manned space program of NASA is a good example of the need for a very elaborate traceability system. The cost is justified by the need to protect human life.

The first defense against bad parts is a well-run, intelligently self-disciplined organization. Every executive, manager, supervisor, and employee needs to appreciate and guard against the consequences of bad parts. The reason for this seemingly platitudinous statement is that bad parts seldom result from deliberate wrongdoing or even simple neglect. They most often develop subtly from ignorance.

The second defense is good, not excessive, common records well kept in every major department.

Keeping it simple, a search starts with the Release Record. When were the governing documents released? Where are they used? What is their effectivity (how many and when)? The next step is Procurement. When did the parts arrive in-plant? When and how many were released to Manufacturing? Then in Manufacturing, when were the parts next assembled? How many were scrapped? Which end-product units were in line to receive the assembled parts and when did they ship? Then in the field, where are the end products now? Find, test, fix, and keep track of the bad parts until all of them are accounted for.

It sounds forbidding, and it is. However, most experience indicates that this loosely organized but existing capability is the preferable method. It will be available because its records exist for other real needs. The more elaborate approaches fall into disrepair because their maintenance and costs are resented!

Traceability is an excellent illustration of the fact that CM is a discipline applied by many, not an organization. Please note that no one but the General Manager controls all the departments involved in a bad parts problem. Only those departments can and will maintain records for their own reasons which can also support a search. Any “CM Manager” confronted with a traceability problem is completely dependant upon the active support of people he cannot control.

Lot Control

A manufacturing lot is a number of units of the same item to be created at the same time through a series of manufacturing operations. The number of units is based on economic lot size. The contract quantity of simple parts, such as stampings, usually constitutes one lot because it costs more and takes longer to setup the machine than it does to make the parts. The lot size for intermediate assemblies will have more to do with efficient movement from one manufacturing operation to another. The top assembly (product) is often made in lots of one. However, economic lot size is almost totally dependent upon the kind of product and type of manufacturing operation.

The existence of manufacturing lots is an almost irresistible temptation for the advocates of traceability. The associated data, captured in an automated system *as part of an as-built baseline*, could be of tremendous help in tracing bad parts. It could help in diagnosing failures and a lot of other things. It could also bankrupt the company.

The key word is *could*. The cost of having the data must be weighed against the consequences of not having the data. So far, not having has won. As the kinds of interrelated data increase, the system becomes more complex and costly to design, maintain, and operate. Eventually, more will be possible but eventually isn’t now! *Because something can be done does not mean that it should be done!*

As noted earlier under Traceability, lot data is usually available as a normal part of manufacturing records. It’s just awkward to use in that form.

Change Control

Change Control maintains the integrity of released (verified) items by approving or disapproving changes to them before the changes are made.

Hard Facts

We start with some hard facts of life.

- Engineering is an iterative process; that is something is done from which something is learned which causes something more to be done from which something more is learned and so on until the boss says "good enough" – or they run out of money.

- Manufacturing is an iterative process until it stabilizes. Then it becomes repetitive.

- Suppliers discontinue some items, initiate new ones, go out of business, or fail to deliver as promised. Replacements are necessary.

- *Given these conditions, change is inevitable!* But, if it flows without restraint chaos and bankruptcy loom. Release was developed to bring order and Change Control was instituted to maintain it.

- *Change Control was not designed to reduce the number of changes!* Although it has some impact on volume the result is relatively slight. The volume of change is driven by the difficulty of the technical work, the experience of the personnel doing it, and the schedule!

We can accommodate these facts or we can ignore them and suffer the consequences. They will not evaporate.

The Change Control Process -

- Determines the desirability or undesirability of each proposed change.

- Facilitates the implementation of desirable changes.

- Prevents the implementation of undesirable changes.

Figure 9 shows the primary elements of the process

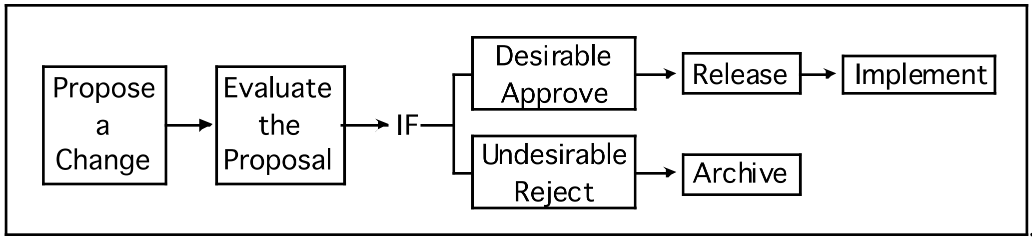


Fig. 9 Change Control Process

The hang-up of course is the definition and interpretation of "desirable". There is no general or lasting solution to that quandary. Numerous categories, definitions, and guidelines exist but none of them, none of them, are adequate over time. The hard fact is that the desirability of change varies from industry to industry, product to product, time to time, place to place, and situation to situation.

Obviously, any change necessary to align the product to need is desirable. Any change that lowers cost is desirable – unless it wrecks the schedule. Any change that increases cost is undesirable – or is it? There simply are no good generalizations. The harder we try to define, refine, and apply them the more muscle-bound we become. At some point, we must rely on informed judgment!

Change Control applies to all released documents and all completed hardware. Changes are proposed at any time during any phase for any number of reasons. Usually, a Change Control Board is responsible for evaluating them. The board members represent all elements of the organization that have direct responsibility for the product. They determine desirability or undesirability in accordance with criteria supplied by management, the buyer, the situation, and their own good sense difficult though that may be. Desirable changes are approved, released, and implemented. Later they are incorporated into the documents that they change. The revised documents are subject to re-release.

The Change Board's task is further complicated by the fact that the desirability of a particular change is not always clear. It is often a matter of technical judgment subject to vigorous debate.

Proposing Changes

The process starts with a written Change Request. So who can write one? At first blush you might think that any employee qualifies but that's not necessarily so. Some organizations limit the kinds of personnel authorized to write a change proposal in the hope that this filter will keep foolish or unnecessary changes out of the system. It may have been decreed that only a supervisor or manager can submit a change. Perhaps only Manufacturing Engineering can propose factory changes or maybe there's a group of change engineers who do all the proposing. Whatever, these schemes always add cost and time to the system without significant benefit.

First, it is deceptive to think of all this activity as happening outside the change system. In reality it is part of the system whether it's recognized as such or not. Second, it is a major error to think of most changes as unnecessary or frivolous. To prove this fact you need only setup an anti-frivolous filter and see what's caught in the net. Significant frivolous change is an industrial myth, good for emoting but little else. Third, these supervisors, managers, or special engineers are removed from the situation by one-degree or more. That means that they must take the time to gather facts and make a decision.

Generally organizations with great concern for quality will permit any employee to propose a change because they are concerned with getting all of the kinks, quirks, and errors out of their products. In my opinion any employee of the organization should be allowed to propose a change. At most I would insist that he get the approval or acknowledgement of his own supervisor to gain a little more maturity in the decision. However, I stress the fact that the system must fit the organization it serves or it will be destroyed. If that requires special filters or writers, so be it.

Regardless of who prepares it, the proposal must state what is to be changed, how it is to be changed, and why it is to be changed. It should be as clear, complete, and accurate as possible to communicate adequately.

Where-used (next assembly) data for the item to be changed must be gathered from the Release System and included in the proposal. It is unwise to think of most items as having *a single usage*. Use of an item in more than one next assembly or product is quite common and all uses must be considered in evaluating the change.

Evaluating Changes

Generally changes are evaluated by a Change Control Board, which consists of a member from each element of the organization that has responsibility for the product. A so-called "neutral party" chairs it. The members must be quite adept at recognizing the limits of their own competence. They apply some version of the following questions *to each usage* (next assembly) of each change and they answer for their organizations if they can.

Technical
- Will the product function if the change (a) is made, *and* (b) is not made?
- Is reidentification required?
- Should the change (a) be made, *and* (b) why?

Financial
- What is the cost to your organization of (a) making the change *and* (b) of not making the change?

Schedule
- Upon approval, how long will it take your organization to implement the change?
- What effectivity should be assigned?

If board members cannot answer these questions, they get the answers from their organizations. When conflict or ambiguity arises, they pull the right people together and seek resolution. Regardless of the method used, the questions must be answered *for each usage* (next assembly) of the item being changed. Based on the circumstances, the decision is documented in one or more of the following forms. These decisions also verify the content of the document and authorize its release.

Document	What	When
Stop Order	A written order to stop work until further notice.	Continued work or procurement will result in expensive rework or replacement.
Deviation	Written authorization, before an item is manufactured, to vary from released documents for a specific number of units or period of time.	Permanent change to the design is inappropriate but a temporary deviation is necessary.
Waiver	Written authorization, after an item(s) is manufactured to vary from released documents.	Such units are suitable for use "as is",
Change Order	A written order to change a released document, as specified in the order, at a specified effectivity.	A permanent change to the design is required.
Rejection	Written notice of the reason for rejection.	The Change Request is not acceptable.

Table 2 Change Board Decisions

Approving Changes

Change decisions can be made by a Change Control Board or by a number of other methods. Frankly, no one method is clearly better than another as long as it (1) reaches every affected element of the organization not for political reasons but *because that's where the knowledge is*, (2) is able to *force* timely trade-offs and final decisions when necessary, and (3) fits the organization it serves. The personality of the organization, most of its customs and traditions, and the powers that be, must be accommodated! And that means that very few of these methods transplant from one organization to another without significant modification. *One size does not fit all!*

If the Board is unable to reach a decision the proposed change is presented to the Program Manager(s) for decision.

After the Decision

Each Stop Order, Deviation, Waiver, and Change Order has its own Identification. When approved, they are promptly released which places them under Change Control. The rest of the organization responds by implementing these released documents at the effectivity specified. Rejection Notices are sent to the originator and Change Requests are filed.

Incorporation

In due course, the content of Change Orders is incorporated into the documents that they change, the identification of changed documents is adjusted, and the revised documents are released as replacements for the previously released Change Orders. Most organizations limit the number of Change Orders that can be outstanding (unincorporated) at any one time in order to reduce confusion.

Stop Orders, Deviations, and Waivers are never incorporated but remain a permanent part of the Release Record. Hardware subject to Deviations and Waivers is marked with the Deviation or Waiver Identifier.

Change Classification

Some organizations use elaborate systems of classification to route changes through variations of processing. Class I vs. Class II, mandatory vs. non-mandatory, major vs. minor, supplier vs. contractor, customer vs. contractor are examples. For instance, a Class I change alters contract cost and generally goes to a headquarters command for review and approval (slow) while Class II changes do not alter contract cost and are reviewed and approved by a local customer representative (faster).

Generally, classifications emerge as compromises in a constant struggle to make Change Control simpler and faster. However, they soon become parochial in nature, barnacled with local practices bordering on superstition, and have far more to do with power than efficiency. The fundamental questions will always be who has the power to say, “Yes” and who has a veto.

As for the CM Discipline, there are only two logical questions to be answered. Do we know enough about the change and its ramifications to make a decision? Should we make the change in spite of (or because of) the ramifications? Yes or No! Whatever method produces these answers, completely and accurately, is adequate!

Change Status Reporting

An automated Change Status System is a necessity. The data recorded must include identification of Change Requests, the documents or items they would change, the classification of the change, location of the Change Request in the processing cycle, the final disposition of the Request, and the date of everything. The primary capabilities of the system should include random access, sorting by any of the elements recorded, counting everything sorted, and time span calculation.

The kind and volume of data available in the Change Control process is almost infinite. The demand for it is sporadic and usually capricious. The effective use of it is very uneven. The cost of recording, processing, and preserving it is relatively high. Even so, some amount of it must be available! The problem is simple. The solution is not.

Normally, no one except Change Control Administrators and Processors has much interest in Change Control data until something goes wrong. Thereupon, the wronged-ones demand information for defense and counter-attack. The problem can be about anything from any level in any department. Unless satisfied, it can swell until it involves many levels and many departments.

If data is available, it will be beaten to death by the wronged-ones until they exhaust themselves. If data is not available, Change Control will likely be “reorganized”. Thereupon, in either case, all will return to business-as-usual until the next time.

The key factor in all this is easy to spot but hard to deal with. *By the time that a change has a status to report, the problem that generated it exists*. The need is to solve that problem not to status the change. However, the change is the presumed solution. Everyone concerned wants to know where it is in the process and how soon it will be approved. Usually, the location is known and processing time can be estimated. However, that information is seldom enough.

So the Change System itself becomes the culprit. Why is it so slow? What is being done to expedite? The answers to these questions and a dozen others are seldom acceptable so the emoting goes on and another log is tossed on the fire of controversy.

In the broader sense, the study of change statistics can produce a number of insights into plant operation. The problem is that almost no one is prepared to do anything constructive about them. It’s not hard to find that most changes are coming from a particular department. It’s very hard for that department head to do something timely about it. It’s not hard to see that span time is expanding and to locate the delays. It can be almost impossible for the delayer to do anything timely about it.

It is difficult for most people to recognize that the causes of Change Control problems are rarely found in the system or the processors. The causes lie deep within the operating philosophy of the enterprise. By the time they are recognized, the fat is in the fire and it’s too late for prevention. The obvious time to correct such problems is in the calm between programs when managers have more time. However, the incentive is usually obscure during such lulls. When the incentive becomes obvious, the lull has passed and there is no time to make general improvements.

One more time – there is simply no substitute for a thorough understanding of the industrial process and the way that it functions in your workplace. Change Control is difficult with it and nearly impossible without it!

Customer Change Control

Customer Change Control is always controversial.

First, it takes time, a lot of time. The customer is cautious because he doesn't know what he's getting into. The contractor is impatient because he's anxious to implement his decision. Second, many contracts have compressed, even incentivized, schedules which are often impossible of performance on their face. They also include Customer Change Control with no time limit on that process. Thus, the contractor strives to meet schedules, which the Customer thwarts with untimely Change Control. Third, in adversarial settings professional conceit swells, "professional differences of opinion" proliferate. They take time and cost money. Fourth, anyone with the right or obligation to intervene is clearly culpable for the consequences. "You approved (or disapproved) so it's your fault!"

Therefore, customers are well advised to be very selective about imposing controls on contractors. Even so, common sense dictates customer control of any change that would alter the following basic parameters of the contract.

- Contract cost, schedule, terms, and conditions.
- Requirements that the finished product must meet. (Functional Baseline)
- Verification methods used to assure that the finished product actually complies with its requirements. (Demonstration & Product Baseline)

Military customers have paid dearly for their unwillingness to accept these limits. Without deep and detailed control, they fear that poor contractor judgment or chicanery will leave them holding the bag. So they tend to require review and approval of every change to make sure it's OK. The irony is that the Military is no more adept than the contractor at making sure. Having spent much time and money on prevention they are still left holding the bag – but – they tried. This is a real problem. Every attempt to solve it produces new manifestations because it is inherent in the government-contractor relationship!

This problem won't go away. It's too subtle to solve. But it could be eased if Customer Change Control applied only to the basic parameters of the contract as stated above.

As a safeguard, the customer can require information copies of all Change Control decisions. If the contractor's independent judgment is unwise, it can be suspended or reversed through various contract provisions. This approach allows the contractor to be a contractor instead of an employee and reduces the customer's culpability. It also provides the Military with a brake to apply when necessary and it will reduce overall cost.

Perhaps the most important factor is that a contract is supposed to be a voluntary agreement to cooperate. When it becomes an adversarial invitation to combat, the product suffers! Customer Change Control should be defined and exercised in that light.

Field Modification

Generally, modifications developed and implemented by field personnel are unwise. It is almost impossible for them to know all the ramifications of their actions which may come back to bite.

However, it is not uncommon for field personnel do develop a modification as an improvement. Such modifications are treated as changes and are Identified, Documented, Verified, and Released into the Product Baseline. They are fabricated as kits and shipped to the field for installation. They are installed, verified, and recorded in the Product Log.

Retrofit & Refurbishment

A retrofit usually involves major modifications. Normally, these will be Identified, Documented, Verified, Released, and Change Controlled either as a major change or as a different product. A refurbishment usually replaces old or worn parts. It makes no change in the design or performance of the product. In both cases, implementation is usually done in a factory.

Fixing Change Control

Every so often a novice, with the power to impose, declares that he "is going to fix Change Control". Life becomes quite difficult until he learns that neither he nor anyone else can "fix" that which is intrinsic. The best he can do is to improve administration that seldom betters overall performance by more than 10 percent, if at all.

The intrinsic factors are the volume of changes which should be lower, the number of people involved which should be fewer, the behavior displayed which should be better, the span time which should be shorter, the cost which should be lower, and the complexity which should be simpler. You, like hundreds before you, will wonder why these goals cannot be achieved once and for all. The answer follows.

Far too many people think that the responses to a Change Request can be routine, timely, and clear; with trade-offs made promptly; and decisions made quickly. Most of the time they're wrong. Information is often late, ambiguous, and conflicting. A change, needed in one usage, can be acceptable in another, and be a disaster for a third yet reidentification is opposed. The manufacturing costs could be lowered but the hardware will have been completed before the change can be implemented. Engineers are irked because their work has been challenged. Change requestors are irritated because their work is delayed. Managers are harried because their work is increased. And, each one wants his problem solved his way!

Before Release, relatively few people are affected. After Release, hoards of people base their work on it. As the program progresses, other designers, test engineers, manufacturing and inspection planners, tool engineers, buyers, suppliers, vendors, budget managers, etc. become dependant on it. If that release is changed, at least some of their work is lost and must be redone. Frustration and stress are abundant. Only the most sophisticated handle it well.

Another problem is the "no change" syndrome! Unfortunately, neophytes sometimes gain the power to issue no-change policies that inevitably embarrass them and everyone else. Change is inherent to engineering, manufacturing, and procurement. If nothing else happens (and it will), vendors go out of business resulting in a necessary change of parts.

In the engineering world, there is almost no such thing as "doing it right the first time" no matter how good they are or how hard they try. Engineering is an iterative process. Each iteration produces new knowledge that quite often exposes both new and old errors and possibilities. Similarly, Manufacturing produces new knowledge that poses problems even though it was "done right the first time". Yes, these circumstances can be used to mask incompetence and that's a significant problem for supervisors. Nonetheless, these circumstances are real and they do produce changes that have to be handled!

There are as many schemes to manage Change Control, as there are people involved in it. None of them really alleviate the frustration because the intrinsic elements remain intrinsic. There is no sweeping panacea awaiting discovery. Some things in life are simply hard work. Coping with the intrinsic elements of Change Control is one of them. However the following can help.

Competent personnel, deeper understanding of industrial practice, minimum Customer Change Control, realistic expectations, sound program planning, and adequate time (beware of compressed schedules) are the elements most apt to improve performance. Even so, don't expect to "fix" Change Control. Learn to cope with it.

Warning

Hopefully the foregoing discussion of Change Control has explained at least a few of the most significant reasons why it is so often controversial. Not so clear is a common consequence. Problems pile up and emotions run riot. The attention they demand swells Change Control until it displaces CM. In fact there are many managers today who think that CM is no more than a fancy name for Change Control. They are wrong! Essential though it is, Change Control cannot keep "an evolving product in alignment with the specified need for it" unless the CM Process was properly applied.

If all you want is Change Control, call it that and do it. If you want to keep your product aligned with need, apply the CM Process and all of its tools.

The Never-Again Syndrome

Like many other things all of these elements of CM are subject to the "never-again" syndrome. A mishap occurs. A task force is formed to see that it never happens again! A check, balance, procedure, audit, or the like is added to the practice and the mishap does not recur. However, years later the safeguard appears to be useless. No one remembers its purpose. Indeed the original purpose may no longer exist.

Such accretions are hard to remove. What appears to be useless may really be needed somewhere deep in the organization. No one knows all the nooks, crannies, and consequences. So it's easier to leave them than to run the risk of removal.

Contrary to popular belief, this condition is not peculiar to government. It is a malady of large organizations, public and private. It is more prevalent in government because almost anyone from a crank to the press can criticize governmental mishaps making overcorrection probable. Generally, there is no cure but knowledgeable people can manage the condition.

There is no doubt that the CM practices should be reconditioned periodically. Their essential elements may be rearranged or updated to accommodate new *proven* technology. However, any greater effort is likely to reinvent the wheel under a different name. It's wise to remember – a wheel by any other name is still a wheel.

***

4. PHASES & BASELINES

Tailoring

Originally, the Military thought that one size would fit all. They persisted primarily because they had few who really understood the fundamentals of an industrial plant. As a result, they got an inadequate system, muddled allocations, and a product developed independently of both. So they came up with the idea of "tailoring" which allows substantial adjustment and selective application. It also allows the less scrupulous to duck and avoid. However, the objective is still to keep the product directly aligned with the specified need throughout the program, no more and no less!

Allocated Baseline

The Allocated Baseline seldom worked well in its original form. The primary cause is the disparate methods used by various industries, contractors, and subcontractors to get requirements from the Functional Baseline into their design organizations. Their methods are as varied as the people using them and it is wise to use those methods unless a *compelling* reason dictates otherwise. Consequently, the Allocated Baseline finally became optional.

This decision put the power to require an Allocated Baseline into the hands of various buying commands. They have used it for a wide variety of purposes or not at all. The problem is complicated because it's rooted in the subtleties of engineering practice, politics, and technology. Even so, variations of the Allocated Baseline are easier to understand if you know a bit about the forces that shape them.

When the Military decided to define need first and then design a product to meet it, the first large scale use of System Engineering (a more or less new engineering discipline) came into being. This discipline brought at least two new concepts into play, system and allocation.

System

The definition of system will remain forever controversial. There have been conferences, symposiums, and powwows convened to define the term. They have come up with such things as a self-sufficient entity including hardware, facilities, techniques, and personnel. Few are satisfied and the debate continues.

Further examination will reveal that these efforts have little to do with the realities of a system and much to do with protecting the interests of powerful factions. As one wag put it, "He who defines system, controls the (engineering) world", an exaggeration but not without merit. So, what can be done with the problem?

Simply put, stay out of it. Use a working definition. *For CM purposes, a system is all of functions, including outside interfaces, necessary to meet the specified need.* If it finally turns out to be a subsystem, component, or something else, rename it. A system in one use becomes a subsystem in another and a mere assembly in a third. All of these terms are relative to the context in which they are used.

Regardless of what it's called, the idea of treating the end-item to be developed as a functioning whole rather than a patched together collection has great merit. The emphasis on interfaces has even more merit. You may well ask, "Wasn't this always necessary?" And, I must answer, "Yes, it was." but it was often haphazard. Elements such as maintainability, training, and even interfaces were often overlooked or left for later in the program. Such practice always produces problems, sometimes very serious ones.

The critical concern is to define an entity that includes all of the functions and outside interfaces necessary to meet the specified need. In most cases, that entity will be a system.

Interfaces

An interface is the point, line, or plain where two items meet or interact. It may be electrical, mechanical, or functional. It may even be specified empty space between items.

- Outside Interface: An interface between a product and an item outside of it. Example: an aircraft radio receiver and a control tower transmitter or an automobile gas tank and a gas pump nozzle.

- Inside Interface: An interface between components of the product. Example: An aircraft wing and the gas tank inside that wing.

Outside interface characteristics can be documented in the product specification, in a referenced interface drawing, or in a referenced interface specification. Such characteristics are subject to Verification, Release, and Change Control as part of the Functional Baseline.

Inside interfaces can be documented with interface drawings. However, they are more commonly recorded in the documentation of each interfacing item. Thus, a comparison of documents is required to see both sides of the interface. The management of interfaces is an engineering task that should not be left to novices!

Allocation

Once the functions of the system have been defined and verified they are allocated (assigned) to sub-systems. Each sub-system is evaluated as a separate functioning entity and then verified. The sum of the sub-systems equals the system.

There are two advantages to this approach. (1) Developing and verifying sub-systems is another way to examine the system as a whole, another chance to find and correct errors. (2) Sub-systems divide the System into more visible and manageable work packages at least in theory.

The work of system definition and allocation is done by System Engineers. It is then handed to Design Engineers who are expected to design physical items that perform the functions of the sub-systems. Theoretically this scheme produces the neat result depicted in Figure 10. Please note the Baselines are provided for reference.

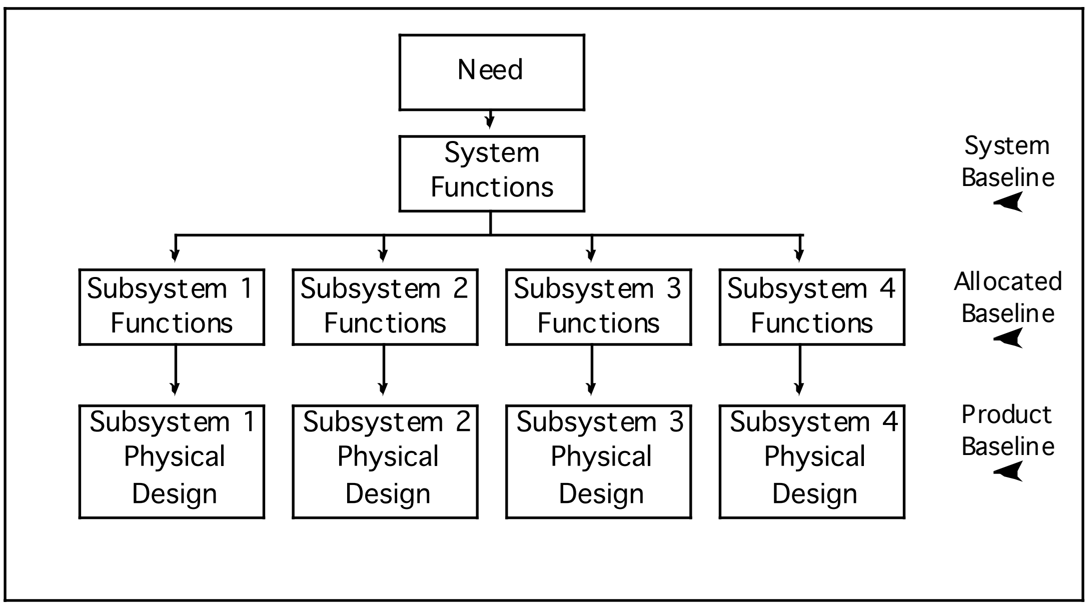


Fig. 10 The Original Allocation Concept

However, this scheme seldom produces the neat result shown. Some of the more significant reasons are noted below.

Cookie Cutter

The approach in Figure 10 was designed for new products and applied as a cookie cutter. However, one size does not fit all. It doesn't even fit most.

In reality, there are very few totally new products. Most of them are modifications of existing items or technology to produce a new result. It's logical to ask why functional sub-systems should be developed for existing physical items whose functions are already established. There are some theoretical reasons but they fade rapidly when cost versus value is examined.

Figure 10 implies that there is a one-for-one relationship between sub-systems and physical items. Sometimes that's true as when a propulsion sub-system becomes a rocket motor. More often it's not true as when an automobile electrical sub-system becomes headlights, taillights, and many things in-between. Figure 11 presents a more likely outcome.

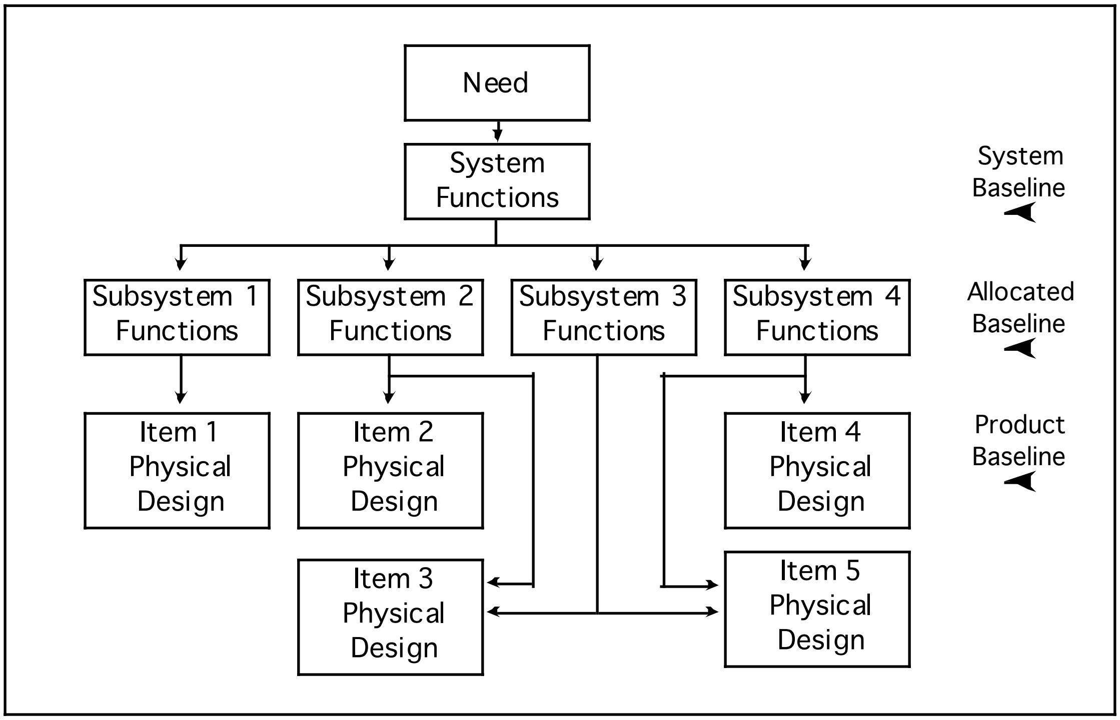


Fig. 11 A Typical Allocation Outcome

<u>One-For-One</u>

The desire for a one-to-one relationship between the functional and physical is more than compulsive symmetry. It does make tracking the functional to the physical and back much easier. This notion played a major role in the genesis of "Two Part Specifications" to document a single item. Part 1 was Functional and Part 2 was Physical. However, the practice requires that physical items be known before the functional allocation is made. This is quite a compromise of the original idea and an invasion of Design Engineering Territory.

<u>Turf Wars</u>

The allocation concept implies that function is the province of System Engineering while physical design belongs to Design Engineering. Even though Figure 11 shows only two levels of functional design, there is nothing to prevent allocation to several additional levels. Such action tends to dictate physical design. It wasn't long before over enthusiastic System Engineers found themselves in small enclaves surrounded by a much larger group of hostile Design Engineers (with the power to disable) who felt that they had been invaded and trashed by a foreign force. Turf wars, neither hot nor cold, but uncomfortably warm, developed and continue in some quarters to this day.

In reality, while the focus of Systems Engineering is functional and the focus of Design Engineering is physical, both must deal effectively with both functional and physical aspects of the product. Turf wars simply complicate an already complex problem and contribute little to its solution.

The Upshot

Having lost sight of the objective, they redoubled their efforts. Prime Items, Critical Items, two-part specifications, and Configuration Items, are among the compromises that still litter the landscape. The uproar finally became so debilitating, that the Military made the Allocated Baseline optional. Of course, this action questions the need for allocation to sub-systems in the first place. In fact, it has been used to avoid the sub-system level altogether by cramming such detail into the definition of the System. Although that approach can work, it can also produce 3 or more inches of inordinately complicated system specification.

Please remember that the cause of controversy is rooted in the subtleties of engineering practice. Resolution depends upon (1) the practices of the buying agencies and their contractors, (2) the attitudes and competence of the engineering organizations involved, and (3) the maturity of the technology needed for the product. Given the number of variables, resolution must be found System-by-System, one at a time. The resolution for one seldom transplants to another without major tinkering.

The engineers of the buying agency and the contractor must reach such resolutions. Otherwise, it fails! CM has no contribution to make to such resolutions. However, there is another way to approach the problem.

A Tailored Allocated Baseline

Sooner or later systems must be divided according to the organizations that will have detail design responsibility for the physical items in the system. Usually that division produces the following.

- Customer Directed Items
- Subcontracted Design Items
- Contractor Designed Items

Customer Directed Items

Customer Directed Items usually fall into one of two categories: (1) Customer Supplied Items such as warheads or (2) Existing Items from specific contractors such as rocket motors. There are all kinds of reasons, usually economic, why a customer decides to follow this route but regardless of reason he has the right to do so. However, he remains responsible for the detail design of the item!

As a minimum, the interfaces between these items and the product of this program must be documented, verified and released as a part of this program's product! And, the Cus-

tomer must agree not to permit change to these interfaces without the prior agreement of the contractor for this program!

Usually Customer Directed Items are identified in the System Specification (System Baseline). However, if such identification is missing or inadequate it can be added to the Allocated Baseline. If the Items exist the existing documentation is usually adequate. If they do not yet exist original documentation must be created.

People often overlook the fact that the interfaces of such Customer Directed Items become design constraints on the product of this program. That is everything else must be designed to be compatible with them. If those interfaces are not imposed upon this program they will be ignored and significant trouble will follow.

Subcontracted New Design

A contractor subcontracts design because he doesn't have (1) the capability in terms of personnel or facilities who can do it as well or (2) the capacity in terms of enough personnel, space, or equipment. Obviously, he has to tell the subcontractor what he wants done. This task is usually accomplished through a functional procurement specification that states the functions the item must perform, *and the functional and physical interfaces that the item must meet*. (It also includes verification methods and handling requirements.) The proper place for this sort of item is the Allocated Baseline.

Contractor Designed Items

This category includes everything required by the System other than Customer Directed Items and Subcontracted New Design.

Every contractor has evolved a way in which to divide and assign Design Engineering tasks. It is based on powerful personalities, organization, tradition, experience, and myriad other elements. No two are the same. *Tamper with it at your peril.*

The Military has never learned this lesson and they pay dearly because of it. The result is that the contractor pursues his own method and converts the results into the form the Military prefers. The Military reviews, comments, amends, and approves these results. Thereupon the contractor reverts to his own method. This means that parallel methods are operating at the same time. They are the cause of much expense and no little confusion.

A far more practical approach is to let the contractor proceed in his established manner. He may divide the work into recognizable subsystems and verify and release them into an Allocated Baseline. He may divide the work into dozens of work assignments and control them through some variation of a release system. He may choose some other device. *In any case, he is allocating functions from the System Specification to Design Engineering.*

For CM purposes, the System Specification is lurking in the background as the source of functional requirements to be met by Contractor Designed Items. Technically speaking this means that the source of requirements for Contractor Design is really the Functional

Baseline. For coherency, you may think of the System Specification as if it had been released into the Allocated Baseline (for reference) even though such a release is rare.

Figure 12 displays the foregoing approach to an Allocated Baseline.

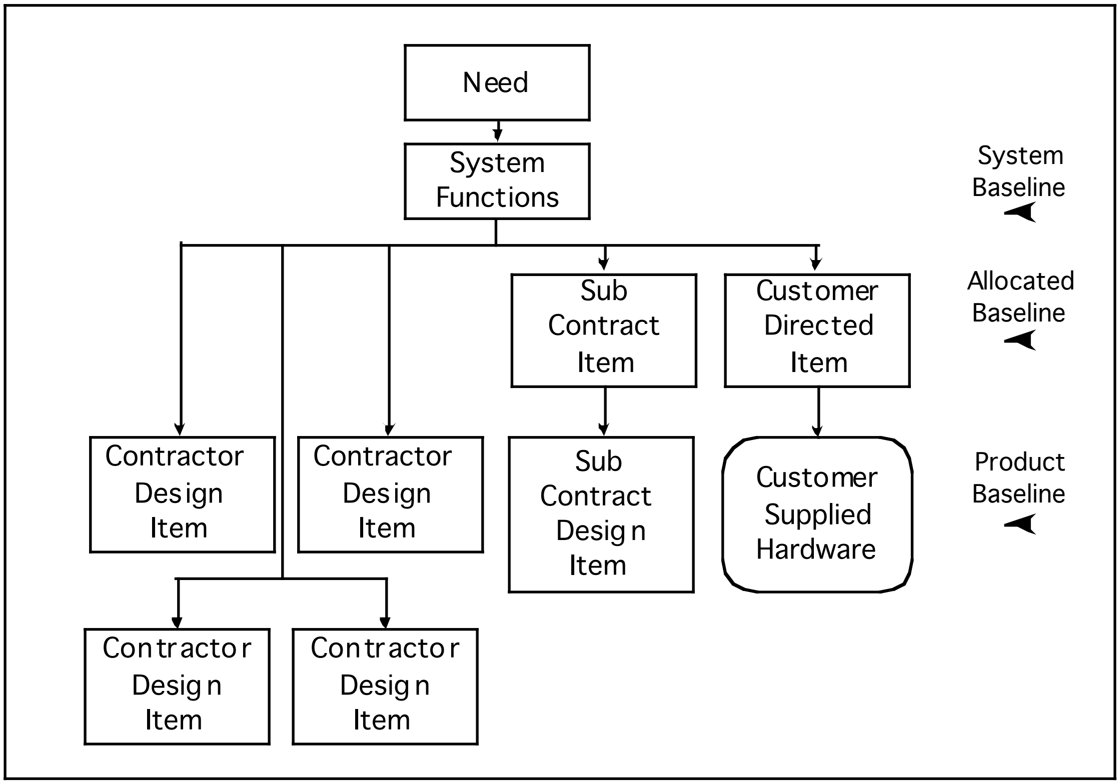


Fig. 12 A Tailored Approach

The Lesson

There is no virtue in an Allocated Baseline as such. It must contribute to keeping the evolving product in direct alignment with the specified need or it is superfluous. When the real effort is to impose or avoid an Allocated Baseline the result will be useless and often harmful. Real-life solutions depend upon the product to be developed, the maturity of the technology, the experience and maturity of the personnel, and prevailing practice in the industry and agency. Once again, one size does not fit all.

The As-Built Baseline

The notion of an As-Built Baseline arises from the fact that there can be permissible uncertainties between the Product Baseline and the hardware built from it.

Alternate or substitute parts are one obvious example. Perhaps the Baseline calls for Part 902 *or* Part 888 in an assembly. The factory installs first one and then the other depending on availability. No special record is made. Later in production or in the field it is difficult to tell which part was installed. It can be impossible if the part is miniaturized. Doesn't it just make good sense to keep a record of the part actually used?

Not necessarily. If we insured against all possible perils we would be insurance paupers. So we insure against the *most probable worst* perils. *Simple prudence demands that the cost of a probable consequence be weighed against the cost of preventing it.*

When the perils of manned space flight are considered, the consequences of not doing something can be catastrophic. Consequently, NASA is much more concerned with the as-built problem. The concern drops when most defense systems are examined and drops again for most commercial products.

There is no decision formula for this problem. Solution depends entirely upon specific situations, probabilities, and experience. So far, in most situations there is little justification for an as-built baseline.

***

5. HOW IT WORKS

Phase 1. Define the Product

Correcting Need

What do you do when you discover that the specified need is really and truly wrong? Fix it! Change it! Now! And, accept the consequences. *They don't get better with age*.

That last statement is not always true. If the Congress has set the need in stone or if the Brass is career dependent upon a particular need, changing it can be career limiting for them and for you! Programs can be cancelled and people can be fired! Sometimes waiting removes the obstacle through retirement or reassignment and the consequences look better. However, *those are political decisions*! The CM Disciple can only contribute facts, which are often unwelcome. It has no part in making the decision itself.

This Devil's Brew is real! It's often the root cause of CM controversy. After all, without the surveillance made possible by CM there would be no unavoidable facts to disturb minds already made up. I have no easy solution. It is one of life's conundrums to be solved by each of us in his own way. It is also the reason why integrity is often prized in theory and demeaned in practice.

If the need is no longer legitimate, fix it. Doing otherwise can cost lives. But understand that CM cannot make the decision for you.

Generally, need is altered by submitting a Change Request to modify the System Specification. It is assumed that the customer will examine that change against need and alter need or reject the change. This is a weak spot in the system for there is nothing that forces the customer to do that job. If it isn't done the gap between need and design reappears and design drift sets in.

Phase 2. Define Major Components

One could go on for some time about major components to no particular benefit. However, it is quite important to understand the material presented under "Allocated Baseline" in Section 4 above. It applies.

Of course care should be taken to see that all specifications are adequate and accurate. But those dealing with Subcontracted or Customer Directed Items need special attention precisely because people are apt to give them short shrift. After all subcontractors are very knowledgeable and should need little direction. Customer Directed Items already exist and should be quite easy to define.

However, a mistake under the control of the prime contractor is relatively easy to correct. But one that crosses a contract line is subject to correction by at least two parties. That makes it difficult. It also opens subcontracts to some degree of renegotiation that can lead to price increases. It's wise to put a little extra effort into Verification for this phase.

Phase 3. Design Prototype

The work of design is the selection of characteristics in the form of raw material, detail parts, subassemblies, and assemblies, up to the top of the product. It includes the integration of Customer Directed Items, Subcontractor Design Items, and Vendor supplied materials, parts, and supplies. Hundreds of people are involved. Thousands of documents result.

Verification of this design really takes place in Phase 4 when the Demonstration Tests are conducted. Even so, checkers and editors perform an analysis almost as complete as a Compliance Review even though it is done piecemeal, document by document, over a longer time period. This verification is essential to reduce the manufacturing costs in Phase 4.

Demonstration Plan

The Demonstration Plan is critical because it will be used in Phase 4 to prove that the Prototype Design meets the Need.

The Plan is derived from the verification requirements of the System Specification (Functional Baseline), Component Specifications (including the Allocated Baseline) and the detailed design of the Prototype. It will demonstrate one or more manufactured units of the Prototype under field conditions as close as possible to those for which it was designed. Contractor and customer test engineers working in close cooperation usually develop the plan. A Compliance Review verifies it.

Most often the detailed plan for Prototype demonstration is released and controlled by the organization that develops it. The obvious question is why. They had it under control long before the advent of CM. At that time Release and Change Control were seen as support to production rather than control systems for the enterprise and Test Engineers were isolated in their own little world. They had to do everything for themselves. Thus, tradition, organizational power, and rivalry played a part in both customer and contractor organizations.

The Demonstration Plan is used by relatively few people and there is no CM reason to change the method of control. In fact, it is a good example of CM being performed by the people doing the technical work. As long as good performance is maintained the only reason to change this approach is a centralized system argument.

The best arguments for centralized systems are economics and discipline. Complexity and inflexibility are the best arguments against it. The deciding factors are found in the personality and operation of the enterprise. If the CM Discipline can be maintained without centralization, there is no good CM argument to centralize. In most cases a hybrid approach develops and works.

Phase 4. Build, Test, and Refine Prototype

Refinement for Producibility

Many of the standard techniques of manufacturing are uneconomic when only one or a few units are to be manufactured. So, they are truncated or modified during the manufacture of Prototypes. Manufacturing planning and routing, inspection planning, tooling, castings, and test equipment are examples. Some parts and assemblies are fabricated or tweaked under laboratory conditions. The completed Prototype Baseline contains these abbreviated methods.

However, these cost saving abbreviations are not economic when production quantities are required. So, the Prototype Documentation is examined for producibility and various refinements are made through Change Control to assure that it will be producible. Producibility has two elements of concern. (1) Can the item be produced in production quantities and (2) can it be produced economically?

Some companies release these changes into the Prototype Baseline and then simply call it the Product Baseline. While this method is slightly less expensive, it also destroys the record of the Prototype Configuration that was demonstrated. During production, trouble may require a comparison of the current production configuration with the configuration that was demonstrated (the Prototype). Of course, that can't be done without reconstructing it; an added cost in time and money. Maintaining the Prototype Baseline is not that expensive and it can avoid a good deal of effort later on.

Copying the Prototype Documentation and calling it the "intended" Product Baseline accomplishes this result; usually a simple automated process. Producibility changes are released into this *intended* baseline. When the upgrade is complete, the "*intended*" designation is removed. The fully released Product Baseline becomes available and the Prototype Baseline record remains intact.

Remember, Murphy's Law is still alive and well. To deal with its mischief, it's wise to keep the Prototype Baseline record complete and accessible. *There is nothing like knowing what worked when you need to do it again.*

Phase 5. Build/Test Deliver Product

Pilot Lines

Once upon a time, production began with a Pilot Line, followed by Initial Production, and then annual Follow-on Production Contracts that could continue for years.

- The purpose of a Pilot Line is to debug manufacturing methods. When these methods are used for the first time, some elements don't work. Tools and test equipment fail. Processes are awkward. People are inexperienced. Product design is misinterpreted. So a limited number of production units are used to flush out problems for correction. A high change rate is expected. (Pilot Line Units are ultimately delivered and function as well as subsequent ones.)

- Initial Production proves that the Technical Data Package (Product Baseline Documentation) is adequate to support follow-on production including competitive or multi-source procurement if required. Initial Production yields the first hardware produced with debugged manufacturing methods. A moderate change rate is expected.

- Follow-on production provides more production units. The Technical Data Package has been stabilized so a low change rate is expected. A full-fledged competition for continuing production can be conducted. If a second source is introduced, operating differences between contractors will cause the change rate to rise and then stabilize.

However, competition suddenly became the source of all good things (according to Congressional mythology). The Military was directed to reduce unit prices by getting to competition faster. The objections of prime contractors were summarily dismissed as obviously self-serving. Something of a contest developed between program managers to see who could get there fastest. Things changed!

The need to debug manufacturing methods (Pilot Line) or prove the Technical Data Package (Initial Production) was deemed to be little more than prattle designed to obscure the benefits of competition. So, a slogan, “Do it right the first time!”, replaced experience and the Pilot Line and Initial Production were combined. And — since it is now being done right the first time, we might as well conduct competitive biding for follow-on production right away. Various combinations and permutations followed along with a few problems.

The unproved Technical Data Package provoked all kinds of dirty pool complaints from competing contractors because it was unstable. It really wasn’t pool but it certainly was dirty. The Package kept changing while they were bidding and it continued to change afterwards. However, any experienced contractor should have expected it.

The second difficulty appeared when the change rate soared. Of course, the glow of early competition faded. However, it was summarily decided that the change control system or the people running it must be the culprits. The critics conducted the usual witch-hunt until they wore themselves out.

We should note that second source and competition is not necessarily the same thing. Second sources are added (by competition or otherwise) when the original contractor cannot supply the volume of product units wanted. Competition is conducted to drive the production unit price down. However, the impact of bug-filled manufacturing methods and unstable Technical Data Packages will raise change rates in either case.

CM has no part in the decision that competition or a second source is necessary sooner, later, or at all. At the higher levels of government, where charts and graphs are often the basis for decisions, early competition looks good but there is a price to be paid.

Compressed schedules, buggy manufacturing methods, unproven Technical Data Packages, competition, and second sources inevitably raise change rates, which cost money and delay schedules!

Although "do it right the first time" is a laudable goal to be pursued vigorously, no experienced manager believes that it is a practical solution to real engineering or manufacturing problems! Either take time to debug production methods and prove the Technical Data Package or plan for a high change rate. Neither practice will please speeders.

Maintenance Manuals

Most often Maintenance Manuals are released and controlled by the organization that developed them. The reasons are the same as those given for the Demonstration plan – essentially tradition, organizational power, and rivalry in both customer and contractor organizations. There is no CM reason to change the method of control. In fact, it is another good example of CM being performed by the people doing the technical work. As long as good performance is maintained the only reason to change this approach is to achieve a centralized system, not always a desirable goal.

Phase 6. Deploy, Maintain, Repair

Significant experience accumulates in the field. Unfortunately it is too often ignored. Most designers have little interest in design performance once the customer has accepted it. After all, they are submerged in new assignments. Field personnel are soon discouraged by this attitude and communication simply dries up.

Many companies try to correct this situation by creating a group of liaison engineers to deal with the field. Another approach is the assignment of designers to the field for a period of time as part of their training. However, almost any approach is nearly impossible to fund unless the customer issues a support contract. In the absence of funding it's impossible to staff the operation.

CM can help this situation, at least a little bit, by keeping the Change Control System easily accessible and responsive to field personnel.

***

6. CHAIN OF VERIFICATION

See Section 6 of the basic text.

***

7. SOFTWARE

Modern Trial And Error

The Electronic Battlefield is a good example of an area in which Software is essential although the requirements for it vacillate endlessly. This battlefield concept is new and evolving rapidly. Need can change almost faster than it can be defined. It is an unstable base from which to derive requirements. Situations like this have spawned a so-called "new approach". Why not send programmers into the field? Let them work with the computer operator and develop the programming needed on the spot. Well, why not?

There is no good reason not to do so, if ----

- Requirements *really* cannot be stabilized,
- Need is *genuine and compelling*,
- Success is both possible and *probable*, and
- You can afford to finance the effort.

However, this is not a "new approach"! It's a return to the venerable "educated trial, error, discovery, and invention" approach. Some of its advantages and limitations are noted below.

- Enthusiasm or desperation for what might be tends to override the probability of what will be.
- Realistic cost and schedule limits are unlikely. Arbitrary limits must be set and will become culprits if the venture fails.
- The computer operator is unlikely to know his real interface limitations. The result is apt to be scads of function that can't be used and loads of data that simply confuse.
- Necessary changes to interfacing elements of the system must be made in the field. On-the-spot changes are notoriously fickle.
- Documentation of the result will be slovenly to the point of uselessness unless mayhem has been threatened repeatedly.
- Ultimately, a successful result must be integrated with the formal program for future use. This is a significant added expense usually ignored until it has to be paid.

- Necessary changes can be made, tested, and revised on the spot. This is a huge advantage when time is really of the essence. But remember on-the-spot changes are notoriously fickle.

There are conditions in this world that compel radical, risky action. When a sober analysis of risk and probability shows that success is possible, perhaps probable, and appropriate disaster safeguards are in place, variations of this approach can be used successfully. However, it indicates an effort to push the state of the art, an effort more suitable for feasibility than design development destined for production.

Software is by no means the only area with problems amenable to this approach. However, the use of the CM Discipline is not practical in these situations even though appropriate use of some CM elements will be helpful.

***

8. GOVERNMENT CONTROL

Methods

Most contract changes are accomplished by mutual agreement of the parties. However, the government, as Customer and a party, usually retains the right to direct unilateral changes "for the convenience of the government" subject to "an equitable adjustment of cost and schedule". The Contractor has the right to appeal. The process is generally accepted as fair. However, such unilateral changes are relatively rare.

Customer Approval is a more common form of Customer Control. Normally, Customer Approval of certain items is required before further work can proceed. Once approved, the item is subject to Customer Change Control.

Those new to the scene often wonder why the Contractor can't be given a contract and left alone to design, produce, and deliver the product. From the Customer's point of view the history of this approach is too grim, the risk is too great, and penalty is too severe.

From the Contractor's point of view Customer Approval may be appropriate but it can present some serious difficulty. Generally, if a change meets requirements it should be approved without comment. However, many reviewers try to improve the change; normal behavior for most engineers. Thus, Change Control becomes a contest rather than a process.

Both views are correct! But take note. *The more the Customer fiddles the less stringent the Contractor becomes*. Disgruntled employees are inclined to "let the government do it". And, the more the Customer does it the more culpable he becomes. The best resolution for both parties is to limit Customer Approval and Control to minimum essentials and make them as efficient as possible.

Duplicitous Contractors

The image of the nefarious, duplicitous, thieving contractor is greatly over done but it persists for good reasons. First, there always are some. Second, the buying command, as the agent of others, doesn't want to be snookered.

However, no amount of "Customer Control" will eliminate the deliberate malefactor. Increasing the stringency of CM is also a near useless response. CM was designed to help knowledgeable cooperative people perform a complex job. When it is used for gotcha purposes it seldom gets anything but more expensive and less effective.

The best course is common sense before the fact and severe penalties afterwards. When they claim that they will do a better job for half the money in half the time you are dealing with genius or fraud. Find out which! When everything said is what you want to hear, beware. Life seldom works that way. Be prudent! When you are conned in spite of prudence, sock them with a penalty, a severe penalty. That action will do more for the industry, decent contractors, and good products than anything else.

With that said, take note: the deliberate malefactor is seldom the problem. Most of the time it is customer and contractor untutored hypesters and imprudent risk takers. They unwittingly lure the naïve and themselves into non-performable commitments. Such is not their intent. It is their destiny. Do not make it yours!

Stringent Customer Control is not an antidote to this witches brew. The best defense is prudence; a thorough understanding of the industrial process, practical skepticism, realistic risk evaluation, and a healthy respect for Murphy's Law.

Testing

Extensive testing or examination (inspection) is often seen as protection from duplicitous contractors. Most of the time it is. But there are limits.

Testing is enormously expensive. It is impractical to test or examine every characteristic of any product. So, the dependent characteristic approach is used. To check an electric circuit in an ordinary residence, we turn on the light. We do not examine the power source, wiring, switch, socket, and bulb unless the bulb doesn't light. A lit bulb is the dependent characteristic because it depends upon the other characteristics of the circuit. If they are absent the bulb won't light. Another cost saving approach is to test for the most probable and serious failures. After all if a loose screw has no impact, there is little point in testing for it. However, neither of these methods takes durability, longevity, corrosion, etc. into account.

Almost everyone can understand these humble examples and think of dozens of others to illustrate the dilemma. Actual performance over time is the only real proof of performance but it is not a practical testing method.

Product Baseline Customer Control

Given the foregoing problems, most military customers will take the belt and suspenders approach. They will require an extensive test program plus control of the Product Baseline.

The most common objection to the Baseline approach is cost. But the cost is trivial when compared to the cost of a major failure. The root problem is time. Customer personnel must approve hundreds of initial documents and thousands of subsequent changes. Usually the customer will locate an office in or near the contractor's plant to facilitate the process. However, local offices are limited in the type of change they can approve. Major changes will be forwarded to the buying command for approval. If the project is a joint procurement it's possible to have two buying commands, two change boards, and all of the confusion and contention one would expect.

In this situation, both the customer and the contractor will be urged to buy the latest transmission and reproduction technologies to reduce time and cost. Obviously, Xerox is better than blue line and computers can be better than Xerox. However, the saving in time and cost, in relation to the total, is miniscule. The time-eater is personnel. The solution is *enough really competent people* to do the job. The Military seldom has the resources. The result is delayed schedules, increased cost, and an indictment of Change Control.

Acquisition Reform

One of the proposed improvements is:

> Stop imposing CM requirements by citing military standards. Instead, ask the contractor how he applies CM. Evaluate his method against industry standards and accept it if appropriate.

It sounds good but it contains the seeds of its own destruction. It assumes that "industry standards" exist. They don't! Therefore, the approach fails. Of course, there is an effort to develop such standards. The Government Electronics and Information Technology Association (GEIA) has issued the "National Consensus Standard for Configuration Management", EIA-649-A, The concept needs further development and they intend to do it. There is a good chance that this approach will be successful in the short term. Long-term survival is dubious.

Industry writes and maintains only those standards that it finds necessary for its purposes. Right now it finds that CM standards are necessary to get the government out of its hair. Once that objective has been accomplished, it is likely that the standard will become obsolete because industry will no longer see it as necessary.

It is true that industry was the original developer of documentation, identification, verification, release, and change control *but not as elements of CM*. It is true that each company developed its own standards for verification, release, and change control because *the company* needed them. The companies did join to develop common standards for documentation and identification because *they* needed them.

However, industry has not developed standards to "keep an evolving design aligned with the need for it" (the essence of CM) because industry doesn't need them. Industry's obligation is to fulfill the contracts it signs. It is the buyer's responsibility to decide if the contract product meets his need.

Over the years many have tried and all have failed to convince industry that it has more than a marketing stake in the customer's need. There would have been no CM at all if government had not written the standard and imposed it with implacable intensity. I see nothing in the future to mitigate these conditions.

***

9. CONTRACTS

CONTRACTING

Department of Defense instructions require the Military to apply CM under various circumstances. However, *nothing but a contract requires any contractor to comply* with those instructions. Once upon a time, Procurement Commands handled the problem by contractually imposing a top CM military instruction that, in turn, invoked almost everything ever written on the subject and had nothing to do with the program as such.

Things have improved though they are still difficult. There are many ways to call for the use of CM in a contract including Addenda, Supplements, Attachments, and Annexes. The following four are the most common.

- Technical Data Requirements
- A CM Plan
- A Special Provision; a section of customized text.
- Insertion into text on some other subject.

Technical Data Requirements

Many of the CM Requirements, both explicit and implicit, are contained in Technical Data Requirements for specifications, engineering drawings, and other documents. CM and TDM are tied very closely together. However, they are not the same thing. Therefore, it's wise to pay close attention to the Technical Data Requirements when determining the CM imposed by contract.

CM Plans

A CM Plan is usually called for as a Technical Data Requirement. The Plan is supposed to detail the CM to be applied during the contract. The prime contractor and each subcontractor submit CM Plans as part of their proposals. Unfortunately, many CM Plans tend to be sales documents; compendiums of commonly accepted CM hype. They are often written and reviewed by the equivalent of interns. If the boilerplate and buzz are right, the plan is approved *and made a part of the contract*. Thereafter, it usually gathers dust unless a dispute develops.

Some people believe that vague and indefinite language will help them prevail in a dispute. However, "the government" will prevail unless the Contractor's negotiators are exceptional! That lightly considered boilerplate and buzz could turn out to have sharp teeth.

To avoid this unhappy result, the CM Plan should be a realistic statement of requirements that fit the product, program, contract, and customer. Do not overstate them in the sales mode and expect to avoid them later.

Special Provision

A Special Provision is simply one or more contract paragraphs setting forth requirements that are usually out of the ordinary. For instance, a complicated test requirement might be spelled out in a Special Provision. In a similar fashion, CM Requirement additions, extensions, modifications, or exceptions can appear in Special Provisions.

Insertion

One might be reading about prototypes and see "… which shall be made part of the Product Baseline." If it makes sense, so be it. If it doesn't, get it fixed as early as possible in the contracting cycle.

The Contracting Cycle

The contracting cycle consists of the Customer's Request for Proposal; the Contractor's response (a work statement, cost estimate, and schedule); negotiation; a Contract document; Contract interpretation or modification; and Contract closeout.

If competition is intended the Customer's Request will go to several Contractors. The Customer will *award* a contract to one or more Contractors based on their proposals rather than negotiation. It is also possible for a Contractor to begin the cycle by offering an unsolicited proposal for which there was no Customer Request. In most cases a contracting cycle between the Contractor and his Subcontractors parallels the cycle between Customer and Prime Contractor.

CM Requirements first appear in the Customer's Request for Proposal. The Contractor's response may accept them "as is" or include a counterproposal. Differences should be resolved and a better understanding achieved during negotiations. The results of negotiations are documented in the Contract Document.

Understandably, CM Requirements are not the primary objective of the Customer or the Contractor. Unfortunately, CM is often left to the last minute and then jammed into place by exhausted participants. Obviously, this is bad practice and often produces mangled requirements best suited for the trashcan. When this practice cannot be avoided, undertake clarification as soon as practical! Many will object but it is better to deal with it early and avoid the almost impossible snarls that are certain to develop later in the program. Clarification can be accomplished through Contract modification or interpretation.

A modification is a rewrite of parts of the existing contract, an addition to it, or deletion from it. Normally, customer personnel resist modification because it opens the contract to cost (usually higher) and schedule (usually longer) adjustments. Assessing the risk is a matter of judgment. There is no formula. Unfortunately, most CM problems are usually regarded as low risk until the Contractor or the Customer is in deep trouble. It's wise to avoid this consequence.

An interpretation is easier to accomplish. A simple letter agreement can state that paragraphs x through z really mean the following. However, any such agreement must withstand scrutiny if a dispute develops during contract performance or closeout. To survive, it must be a *reasonable* interpretation that does not strain credulity. To avoid disallowance by the Contracting Officer, make him a party to the letter agreement.

Contract Kinds

The CM Discipline has been presented as a smooth-flowing, uninterrupted continuum covering the life cycle of the product and that is the way it should operate. However, if care is not taken to preserve that continuum, disconnects will develop as the program moves from one kind of contract to another.

A program can be covered by a single all-in-one contract but this method appears to have been unsatisfactory for both Customers and Contractors. Many kinds of contracts are available but the following ones are most often used. Purchase Orders usually apply to Vendors. The others are used between Customers and Contractors and between Contractors and Subcontractors.

> Feasibility Contract. Covers all kinds of efforts too immature for use in full-scale development and production programs. The CM Discipline should not be imposed but tailored elements of CM may be helpful.
>
> Development Contract. Usually covers design, prototype manufacture, and test. The CM Discipline applies. In a competitive effort, a separate Development Contract for each winning Contractor will be awarded.
>
> Production Contract. Usually covers the manufacture of hardware designed under a Development Contract. Operating and Maintenance Methods and the Maintenance Document Package are often made a requirement of the Production Contract. The CM Discipline applies. Over time, there may be many Production Contracts, one per year for instance. In a competitive effort, a separate Production Contract for each winning Contractor will be awarded.
>
> Support Contract. Covers all sorts of ancillary tasks such as maintenance, retrofit, field service, spare parts, etc. The CM Discipline may apply. The number of Support Contracts depends upon the practices of the Procuring Command and the kind of product supported.
>
> Purchase Order. Covers the procurement of ready-made (off-the-shelf) parts or materials from Vendors. The CM Discipline applies but in a reversed sort of way. In simple terms, Vendors have already defined and identified the characteristics of their own products in their own documentation. New design is not involved. Contractors use the Vendor's documentation or a Specification Control Drawing to order items from the Vendor. Contractors may impose additional examinations or tests to Confirm Compliance.

Chain of Reference

Documents call for (invoke, reference) the use of other documents in whole or part. *To the extent that they are referenced,* such 'other documents' are just as binding as if they had been incorporated, word for word, into the referencing document. Drawings reference parts lists, which reference specifications and other drawings, which reference standards, which may reference other standards, which in turn may reference … etc. This condition is called a “chain of reference” and may extend through 16 or more levels of referencing.

This method of referencing is used because it eliminates the need to repeat the same text (describing for instance a common method such as welding) in document after document. Instead, the common method is simply referenced. There are many cost and management advantages.

However, sloppy referencing can produce serious conflicts. It is not uncommon for a lower tier of reference to invoke a document that is a useless, even damaging, application for the product being developed. Knowledgeable, intelligent interpretation is the best (but seldom used) solution. More often harried functionaries pursue their own agendas and each other.

Referencing a document without knowing the applicability of its chain of reference (rote referencing) should be stopped but that is unlikely. However, a special contract provision can be helpful. It can limit applicability of a document’s chain of reference to the first two or three tiers unless that document was specifically prepared for the product being developed. Regrettably, this is an expedient that won’t help future users because they won’t have the same contract. The elimination of rote referencing coupled with knowledgeable interpretation is still the best solution.

It is always possible to give a document a new identity by attaching a cover sheet that specifies which parts of that document are to be applied. This Technical Data Management technique can change the Chain of Reference to eliminate inappropriate lower tier references while retaining the applicable information without an expensive rewrite.

***

10. CM ORGANIZATION

The Primary Actors

The primary actors are:

- The Customer as Buyer from the Contractor.
- The Contractor as Seller to the customer, and

  As buyer from Subcontractors and Vendors.
- Subcontractors as Sellers to the Contractor, and

  As buyers from Vendors.
- Vendors as Sellers to Contractors and Subcontractors.

The term "supplier" can apply to any seller.

Customer

The Customer's role is to specify when and what he needs, to negotiate an agreement with a company capable of supplying it, to assure that the product delivered is the product that was ordered, and to pay for it.

Technically, the military Customer is "the government". Its agent is a Contracting Officer, the only person who can legally commit the government to an agreement.

Practically, the military Customer is an Armed Service Procurement Command consisting of a Commanding Officer and staff including Program Managers and technical specialists. The procuring command is almost never the using command. The relationship between these military commands is significant although it is seldom readily visible to the seller.

Although a military officer is in charge, other personnel may be military or civilian. Typically, the Program Manager is the primary spokesman for the Customer. However, technical specialists usually handle the details of the program. All personnel are "guided" by extensive and cumbersome regulations imposed by the Congress, the Administration, the Department of Defense, or one of the armed services.

Non-military customers may be commercial companies, non-profit organizations, or other government agencies. Their duties are the same but their organizations are more like those of a contractor.

<u>Contractor</u>

The Contractor's role is to supply the product ordered and collect a reasonable price for it.

Typically, the Contractor is a company. Its agent is one or more officers of the company supported by a contracting staff. The agent is usually the only person who can legally commit the company to an agreement but a corporate staff of some size almost always oversees his actions.

Practically, the Contractor takes the form of a Program Manager, technical staff, and liaison with many other functions scattered throughout the company. The Program Manager is usually the primary spokesman for the Contractor. However, his technical specialists usually handle the details of the program. All personnel are "guided" by practices imposed by the company's hierarchy.

"Prime Contractor" is the term most often used to identify the Contractor who deals directly with the Customer.

<u>Subcontractor</u>

The Subcontractor's role is to supply the Prime Contractor with custom designed components such as rocket motors, aircraft engines, generators, radar units, etc. He may be a designer, manufacturer, or both.

Typically, a Subcontractor is organized and operates like any other contractor. However, his products are for use in a Prime Contractor's product.

<u>Vendor</u>

The Vendor's role is to supply Sub or Prime Contractors with ready-made (off-the-shelf) parts or materials such as nuts, bolts, screws, solder, electronic parts, raw materials, adhesives, lubricants, etc. He is usually a manufacturer or broker.

Typically, a Vendor is organized and operates like any other contractor. However, he makes or distributes standardized ready-made parts or materials for wide markets, both military and commercial. Generally, he does not customize anything for a special use.

Customer and Contractor specialists, interacting with one another, make most of the detail program decisions. The same is true of Contractors and Subcontractors. These people develop necessary, reliable, personal, business relationships that lead to agreements between organizations.

In daily practice, the primary actors can appear to be individual people but they are not! The primary actors are organizations represented by people who may or not remain in place. Transfers, promotions, resignations, death, etc. can destroy the most carefully developed unwritten agreement at the most inopportune time. Consequently, the only reliable way to maintain agreements between organizations is by written contract.

The representatives of each organization are responsible for maintaining a clear contract record for their successors. When enthusiasm obscures that responsibility, the contract becomes uncertain and the organizations are subject to arbitrary or capricious interpretations by uninformed successors. CM is unusually vulnerable to this unfortunate consequence.

***

11. IS IT REALLY NECESSARY?

If anyone has done a real cost-effectiveness (or similar) study of CM, I have not seen it. I suspect that the complexity and cost of doing such a study would discourage any but the most determined and naïve MBA candidate.

Even so, I remain convinced after all these years that CM, *properly applied*, and in spite of its many annoyances and confusions, does save money. However, my proof is anecdotal and the savings are almost impossible to isolate because they occur in such diffuse areas as design time or procurement and manufacturing error reduction. So the cost-effectiveness debate will likely continue inconclusively.

***

12. A TALE OF TWO PROGRAMS

This story is true - as far as I know it. The actual happenings were so many and so complicated, strewn across a wide swath through time, that it is doubtful that anyone knows “the whole truth”. Still, it’s useful to us because it contains real life examples of errors and their consequences, as part of actual happenings.

Our tale begins with the Applied Physics Lab (APL) at Johns Hopkins University. They had come up with a promising concept for a sea-to-air missile that would foil future Kamikaze-like attacks, the Japanese suicide missions of WW II. The Navy built a Naval Reserve Industrial Plant at Pomona, California to develop the idea and engaged the Consolidated Vultee Aircraft Corporation (Convair) to operate it. The result was the Pomona Division, first of Convair and then of General Dynamics when it acquired Convair.

12A. The Standard Missile (SM) Program

Terrier and Tartar were among the early manifestations of the APL concept. Several versions of those missiles followed during the late 1950’s. In the early 60’s a standardization effort gave birth to the notion of a Standard Missile. Of course, the Pomona Division wanted that business.

Meanwhile other developments were converging on this project. The consequences will be easier to understand if Figure 13 is consulted as we go along. The upper panel displays a relatively common program plan in use at the time. The lower panel displays the controversial plan for Standard Missile (SM). I can no longer vouch for the exactness of the numbers but they are accurate enough to illustrate events.

The Problems

Compression

Every so often the Congress rediscovers competition and behaves like it’s a cure for all the ills that beset the Nation. This was one of those times. Defense Industry scandals were abnormally prevalent and severe. The Congress was plaguing the Military, which was doing everything possible to placate it. Thus, the heat was on to reach competition as soon as possible.

The SM Plan got to competition 1.5 years faster than the old plan. This feat was accomplished by shortening Development to 18 months and omitting the 12-month Pilot Line altogether. Never mind that there were half again as many prototypes to be manufactured in 75% of the time normally allowed. “Just eliminate the waste and you’ll be fine”.

Nominally, it took 36 months to reach bug-free, economic, volume production. Regardless of what the bits and pieces were called the amount of work was the same. But a different idea had taken hold. The contractors’ recommendation for a Pilot Line was seen as a ruse to delay competition for as long as possible. So the SM Plan eliminated it.

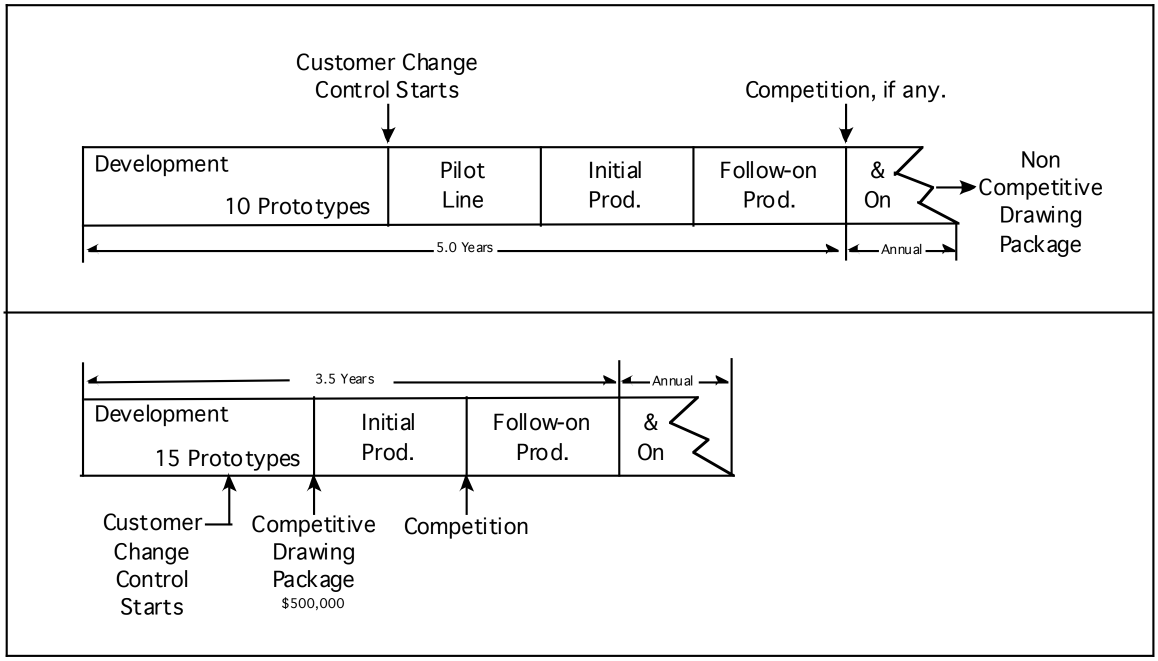


Fig. 13 Traditional vs. Standard Missile

Change Control

Normally, Customer Change Control was imposed at the end of development. The SM Plan imposed it during the last third of Development, specifically on the last half of the Prototypes. This scheme was curious for it was a much-debated compromise without clear intent. The first 8 prototypes, as I recall, were left under contractor Change Control. The last 7 were under customer Change Control. Most likely the intent was to bolster the effort to reach early competition by limiting changes. This mistaken notion prevails in spite of experience.

The signification of this action to the SM Program was a built-in conflict between Customer Change Control, which takes time, and a schedule already compressed beyond reason.

Competition

For a time there was a debate about competing the Development Phase of the program. Fortunately the Navy found that it didn't have the money to fully fund two major contractors during Development. The next effort was to introduce competition at the end of Development.

This approach was resisted on the grounds that a fully proven data package was necessary for competition and a competitive package was impossible without production experience. So it was finally agreed that it wouldn't be imposed until the end of Initial Production.

Competitive Data Package

However, a full-fledged competition takes about 12 months. That means that the data package must be available at the end of Development. A $500,000 bonus was the incentive to produce the package on time.

Basically, a competitive data package is a set of drawings and other documents that can be used by any competent manufacturer to produce the item depicted. However, it is practically impossible to know that a package is suitable for competition until it has been used to produce the item depicted in significant quantities.

Technical Data Management

The concept of a Form (DD-1423) defining all of the data that had to be produced under a contract was introduced for the first time. While there had been much debate no one had elected to put the policy into practice prior to this time.

Fixed-Price

The fixed price concept was applied to this program. It means that for a stated number of dollars (covering all costs and profit) the contractor will perform all of the work listed. This was a revolutionary idea for Development Contracts that had been cost-plus in the past.

Change

The Pomona Division of General Dynamics was a large organization. So was the U.S. Navy. It takes large organizations a lot of time to change. This is not a question of resistance although that counts. It is far more about people perceiving – correctly – what's wanted, figuring out how to do it, and cooperating with each other to get it done. That takes time! The sheer number of major changes listed above was more than enough to guarantee failure when a compressed schedule was one of them!

Why would an experienced contractor agree to a program that requires early Change Control, early competition, and a new approach to Technical Data inside of a compressed schedule for a fixed-price? And, for that matter, why would any seasoned Customer want to deal with a Contractor who would agree to such a thing?

The Navy's motive was fairly clear and they made no secret of it. They had to placate the Congress. The planners simply over reached by gathering the new ideas under consideration and, ready or not, slapping them on this program. Wisdom played no role.

Hubris played a major role in the Pomona Division's decision; "We can do anything!", so to speak. However, there was another far more significant reason. Without the Terrier/Tartar program there would be no Pomona Division. Terrier/Tartar was certain to end when the Standard Missile went on line. More bluntly, the Division had to go where the money was and the Navy had it. The Navy had to go where the skill was and the Pomona Division had it.

But there was one more thing that had a heavy influence on both parties. The idea that "they don't really mean it" had been around for several years and experience indicated that it was true. Many excessive contract provisions were simply window dressing, ignored once the contract was signed. There was nothing but prudence and great unrest in the defense industry to indicate that it would change.

Negotiation

The die was cast when the Division elected to bid on the SM Development Contract. The negotiation was bumpy but made progress until the very end. It broke down over two points. The Contractor wanted $5 Million more and the Navy claimed not to have it. The Navy wanted data in accord with a DD-1423 that the Contractor couldn't fathom.

Time was short so the Division put together a special team to seek resolution. The Contract Manager, the Program Planner, the Chief Estimator, and I were directed to go to Washington and get it settled. My responsibility was the DD-1423 since I seemed to be the only person in the Division who understood it.

It had been priced at $1.2 Million provided that the Navy dropped its demand for Tooling and Test Equipment drawings and Manufacturing Planning. The Navy negotiators were convinced that these items contained the secrets of successful manufacturing. The price for them was huge because of the sheer volume of such data.

This issue was in an unseemly snarl because a number of Navy people were convinced that we were suppressing this data to gain an advantage. This belief was embedded even though their own procurement regulations counseled against the procurement of such data. The Navy had found it to be useless in the past. Ultimately, I must have been convincing enough because the requirement finally disappeared from the DD-1423.

I was not so persuasive with the $500,000 bonus for timely delivery of a competitive data package. I was not surprised because I had been the only voice, Navy or Contractor, in opposition to this item. I took that position because there was no mutually agreeable acceptance inspection method for something so nebulous. I foresaw a major conflict when the Navy refused the package as inadequate while the Contractor insisted that it was.

We had struggled in Washington for 10 days when our Corporate Office told us to go home. They had settled. No one ever did fess up but it was rumored that the Corporate CEO and the Admiral in charge of procurement met and agreed to split the $5 Million difference 50-50. The final figures supported that view.

Development

We had entered into a fixed-price contract for $2.5 Million less than our bid and it wasn't padded! Internally, that difference had to be absorbed and much of it came out of the funding for the DD-1423.

Development proceeded remarkably well until we reached the point of applying Customer Change Control to the last of the Prototypes. A frustrated company accused the Navy of dragging its feet so that we would miss the due date for the procurement package and the $500,000 bonus that went with it. Of course, the Navy disagreed. The truth was not in foot dragging. It was in the time it takes to process changes but the atmosphere had been poisoned.

Those involved with Documentation had pretty good proof that we could not meet the date for a competitive data package. It was suggested that the company announce that fact and give up the bonus. Upper management went ballistic! We were told to make that date or get out.

The Manager of Documentation was replaced with a company hatchet man. He started by rearranging the supervisory structure and moving the physical location of the group. Then he let it be known that we would make the date and earn the bonus or else. Generally he was merciless on the working level personnel.

It wasn't long before everyone got the message and began presenting drawings to the Navy for review and acceptance. The Navy personnel had never conducted such a review and had little idea of how to do it. But do it they did and ownership of the drawings passed to them, something they had coveted from the beginning but would live to regret. Ultimately we met the delivery date, earned the bonus and the SM passed its demonstration tests with flying colors. Management was thrilled as we had proved that "We could do anything!" But had we? Several million dollars of overrun was hardly offset by the $500,000 bonus.

Initial Production

Initial Production did not proceed as easily as development had. The Navy had officially delivered copies of *its Competitive Data Package* to us and to the company that would bid against us. Our factory was getting tooling and testing stabilized along with all the other things usually done during a Pilot Line. But an unusual volume of changes to the data package kept the effort in flux. These were changes that were skipped during Development in order to make the delivery date and earn the bonus.

They were also disrupting the competitor's effort to put his bid together. It wasn't long before he complained that a constantly changing data package was not suitable for competition. Of course, the Navy insisted that it was and the wrangling went on for some time before the Navy came up with a solution.

We would simply freeze the data package for bidding purposes. That way both contractors would be bidding on a stable package. Meanwhile, the Pomona Division would continue with the Initial Production program. That meant that there were two data packages in play, *never a good idea*, one for Initial Production and the other for bidding purposes.

As the high change rate continued the two packages diverged. The Navy accused us of having hoodwinked them into accepting an inadequate Competitive Data Package. Our answer was that, in spite of our protests, "You reviewed it and you approved it as adequate. You wanted to own it and now you do. It's your package that's causing trouble." Thus it was that we wrangled our way to the end of Initial Production.

The SM was produced and it worked. The competition was conducted. The Navy split the first year's follow-on production awarding one-half to the Pomona Division and the other half to the competitor and the troubles began again.

Follow-On Production

The Navy had to increase the money award to cover integration of the changes made during Initial Production but not incorporated into the frozen Competitive Data Package. The competitor had to absorb all of those changes before he could start Follow-On Production, a very awkward situation.

Following this contract, Follow-on Production contracts were awarded on a competitive basis year after year. The total number of missiles was split in various quantities between the Pomona Division and its competitor. As far as I know that practice continued until General Dynamics sold its interests and closed the plant. The missile did work and became the primary anti-aircraft defense for the fleets of most countries in what was then known as the free world. However, there are two more bits to this tale before we close.

The Claim

At the end of Initial Production the company filed a claim against the Navy to recover its costs for correcting "the Navy's inadequate Data Package". The General Manager of the Pomona Division retired unexpectedly and without explanation.

The claim provoked a gasp heard throughout the plant and all the way to Washington and back. After all, we had produced the package. It was undeniably inadequate. We had some responsibility. However, we had protested and argued for the time to do it right. The Navy had denied our plea. It was convinced that we were resisting competition by deliberately delaying. (Yes, we resisted competition quite openly but not by delay.)

We were told to present the drawings so that Navy personnel could decide whether they were competitive or not. We warned that neither our people nor theirs could make that determination accurately. When we were told to do it anyway we elected to earn the bonus and let the Devil take the hindmost and he did.

The claim went forward. It was based on the fact that the Navy's Data Package, furnished to us for Initial Production, was inadequate. The Navy had admitted that fact (1) by taking "ownership" of the package at the end of Development and (2) by approving all of the changes that had been made in order to correct it. We were entitled to reimbursement for the costs of those changes. The claim was quite large for its time – rumored to be somewhere between $20 and $40 Million. The exact figures were withheld from most of us. The company recovered most of what it claimed.

Thus, the feeling of mutual betrayal that had infected both parties was set in concrete. It was never resolved and the Pomona Division's reputation never fully recovered.

Fixing Change Control

Some months after the claim had been settled I was summoned by the Vice President and General Counsel, sworn to secrecy, and told that the Corporate Office would not let the Division accept the next SM Production Contract unless and until Change Control was fixed. The question was what should we do? I assured him that Change Control was not out of control. It was not the cause of the SM change rate problem. He made clear to me that, true or not, such an answer was unacceptable to the corporate office and would result in no contract! Something more had to be done! I was given the task of figuring out what to do and getting it done promptly.

The corrective action that should have been taken was the adjustment of contract provisions to accommodate a compressed schedule, or rejection of the contract, or knowledgeable acceptance of the consequences. It would also have been wise to tone down the notion that "we can do anything" because we couldn't. But it was quite clear that we would get nowhere with that kind of an answer. There were several things about our approach to Change Control that could be improved so I set about getting that done.

We were a multi-program Division. Each program had developed its own approach to Change Control. The differences often caused confusion and delay in change processing so we made the system uniform division-wide. The only differences retained were those required by the various customers, Army, Navy, etc., and they were few.

The other major modification was the elimination of a formal Change Board. The logic was that most of the time the members of the Board had to consult with the departments they represented before they could act. So we decided to forward changes directly to those departments for action without the middlemen.

Engineering and program personnel went berserk more over rumor than fact. Several Vice Presidents met with them and read the riot act so to speak. Accept the modifications or do without a contract. Most accepted it but many continued a subversive resistance that caused constant local criticism. Management had an article published in the corporate-wide newspaper advising the world of the wonderful improvements that we had made in Change Control. I resisted publishing this stuff because what we had done was not that great. Hubris won the day. The Corporate Office lifted its restrictions and I got to host delegations from our other Divisions. They wanted to know what was so great about Pomona's Change Control.

And, the beat went on. Several Program Managers authorized individuals to examine the Change System independently and thoroughly and prove that it wasn't working. Generally, these individuals were quite competent and I instructed all personnel to cooperate with them in every way. They all reported that there was no correctable problem with Change Control. However the carping continued. The primary complaint became span time. It took too long to process a change. Allegedly, the lack of a Change Board was at fault.

So, I took advantage of a situation to establish a Change Board for a particular Army Program. I chaired that Board personally. An influential Design Manager was assigned to the Board to oversee the engineers. Generally, the other departments assigned top-notch people to represent them. A room was dedicated and several phones were installed for rapid communication. This Board met everyday from 1:00 P. M. until every open change for that program had been reviewed. Any delay by any department was pounced upon and pursued. At the end of ninety days we had improved the span time for the average change by about 1.5 days, a little less than 4%.

No company or agency can afford to devote that amount and quality of effort to its Change Board routinely for a 4% improvement in span time. The volume and quality of change reflects the operation of the entire organization. If big money is to be spent on it, a greater gain will be realized by improving performance in each part of the organization. Too much pressure on reducing change span time acts exactly like a compressed schedule. Any number of ills develop not the least of which is personnel burnout.

This became obvious when most members of management were incentivized to fix the Change Control Problem. Each one had a high old time fixing the problem at the expense of everyone else. And, then there was the period when the Chief Engineer decided to review each change before it entered the Change System. He intended to fix the problem by keeping unnecessary changes out of the system. It took him six weeks to discover that the vast majority of changes are necessary. He gave up.

So it was that we arrived in the 1990's still wrangling over a change problem that had started in the 1960's. There were several other convulsive efforts to fix things including the transfer of Change Control to another Branch of the Engineering Department where it remained for a couple of years (without significant modification) before being returned to its former home. Shortly after that it became evident that the Division would close so the critics had more significant things to worry about.

12B. The Standard ARM Program (Anti-Radiation Missile)

In the mid 1960's the Vietnamese were destroying our aircraft and pilots at a catastrophic rate. They had gained the ability to have a missile ride up our radar beam to destroy our aircraft with uncanny accuracy.

The Navy teamed with the Air Force to develop a counter-measure. They came to the Pomona Division with a proposal. They wanted us to modify the Standard Missile so that it could ride down the Vietnamese radar beam and destroy their ground station and its missiles before they could be launched. Essentially this action would convert a sea-to-air missile into an air-to-ground missile. We agreed.

The Navy and Air Force set up a joint Program Office. We agreed to bend every constraint as far as we could to get this weapon into the field at the earliest possible date and they agreed to help. Cost was not to be a constraint.

By this time CM had come into being and was being pressed by all customers on all programs. Our *tailored* approach for this program is displayed in Figure 14.

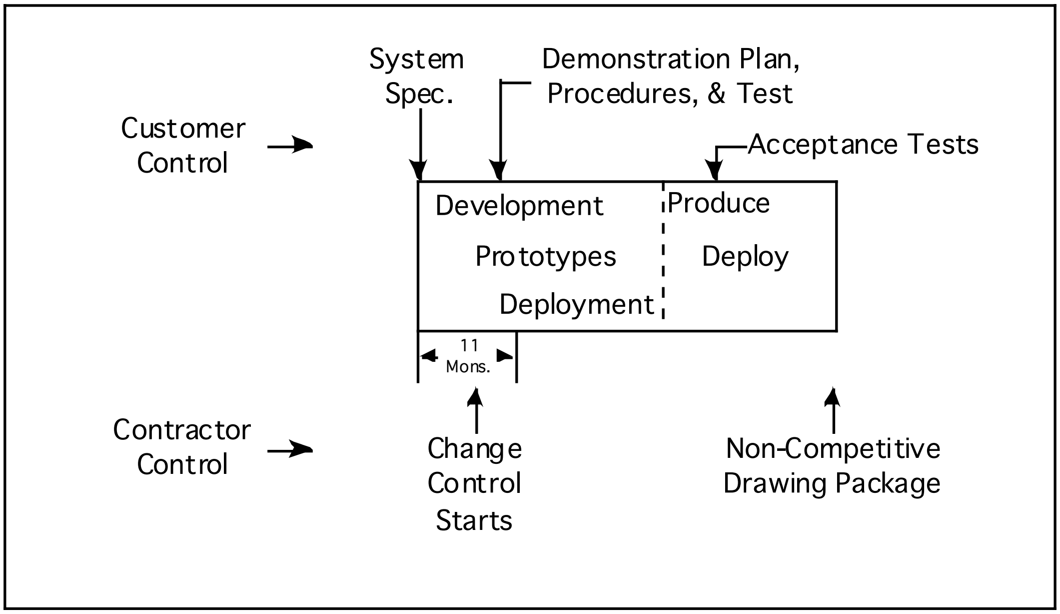


Fig. 14 CM on Standard ARM

Of course it was not as neat as the diagram implies. But it's accurate enough to illustrate the approach.

The System Specification defined the performance required. The Demonstration Plan, Procedures, and Tests proved that the required performance had been achieved in the hardware. (These tests were completed in the actual theater of war.) Some of these tests evolved into simpler Acceptance Tests for follow-on missiles. All of these items were placed under Customer Change Control, which gave it the control of performance that it needed. The Contractor was obligated to comply.

Contractor Change Control started with Prototypes destined for the field and continued with the following missiles. The drawing package was to be no more than what was necessary to allow us to produce the hardware in the labs and the Experimental Factory. This approach gave us the flexibility that we needed to meet the "as soon as possible" schedule. The Customer was obligated to comply.

While Technical Reviews were conducted almost monthly their purpose was perfection of the design rather than compliance assurance. We believed that the risk of drift was low because Development was short, the objective was clear, and scrutiny was constant and intense. However, the door to cost increase was open because cost was not a constraint. Higher cost often makes up for lack of time.

The missile was wildly successful. From program initiation to first working hardware took eleven months, an almost unheard of accomplishment. It was very costly but it saved lives and won battles. Basically, the Vietnamese gave up this fight by shutting off their ground radar to prevent its total destruction. The slaughter of our pilots stopped.

No Good Deed …

The Navy decided that it wanted more missiles but at a substantially lower unit cost. They established a new Program Office that became intent on lowering the cost by introducing competition. The outrage was palpable when they realized that the Standard ARM Drawing Package was not suitable for competition. Worse, it was not even suitable for production in our own Production Factory. We were still producing units for the field in our Experimental Factory supported by our Labs.

We were flabbergasted because the quality of the drawing package had been carefully defined to avoid this specific complaint. We explained that we would need at least a Mini-Development Program to make the drawings suitable for quantity production. Pilot Line and Initial Production would be necessary to assure that the package was suitable for competition. The more that we explained the madder they got.

It became evident that the two Navy Program Offices were not speaking and nothing we could do would get the Production Program people to talk to the Development people. So, we remained the unscrupulous scoundrels that had hoodwinked the Navy into accepting a defective data package. There is little doubt that the quarrels about the Standard Missile were heavily influencing Navy thinking on Standard ARM. They took exceptional steps to see that the Navy was not “fooled again” which did nothing but delay the effort.

A Mini-Development contract was awarded and the drawing package was improved for production and eventual competition. Once again time for Pilot Line and Initial Production was not available but the program continued until later designs by other companies replaced the Standard ARM completely.

12C. Lessons

This Tale is well worth reviewing more than once. It contains many subtle lessons not so much about CM as about the impact of program planning and management. It’s tempting to call them “lessons learned” but it’s more likely that they were merely lessons endured. Perhaps the best description is “lessons to be learned”. Some of them are highlighted below.

Compression

It’s true that things can become so sloppy over time that they need to be wrung out every now and then. In these circumstances compressed schedules become popular. They can be useful but they are always dangerous and often produce unintended consequences. The compressed schedule for SM Development did not allow enough time to complete the producibility effort, abandoned a Pilot Line, and distorted Initial Production. Design development had to be completed during Initial Production under Change Control. All of this made preparation of a Competitive Data Package in the time allowed impossible.

The compressed schedule created a pent up demand for changes that was not satisfied during Development or Initial Production. It fed upon itself until it became a long-term plague, cost an enormous sum unnecessarily, and damaged the reputations of everyone associated with it.

Competitive Data Package

We need a series of seminars on Competitive Data Packages. It was originally thought that "suitable for manufacturing by any competent manufacturer" was an adequate definition. In theory, it was accomplished by:

- Strict adherence to Military Specifications,
- The elimination of proprietary restrictions, and
- The removal of manufacturing methods ("how-to" data) from the package.

Military Specifications put everything in a common language understood by most established manufacturers. The elimination of proprietary restrictions removed legal constraints. The removal of manufacturing methods gave the competitor freedom to apply his methods, which is necessary and does reduce cost. These efforts would have allowed any competent manufacturer to produce the item but not in a smooth operation within a month or two after start-up. *Yet smooth operation is what the military customer demanded.*

This viewpoint is wholly unrealistic even between two identical plants operated by the same company; even between an Engineering Department and a Manufacturing Department operated by the same company at the same location. No matter how much effort has gone into making documentation clear, precise, and accurate some mechanic (or engineer) will point to something and ask, "What do you think they meant by that?"

By direction, methods are not included. There are always elements, sometimes called "know-how", that cannot be documented adequately. One contractor's methods will not necessarily work in another contractor's plant! They have to be developed through practice. As a minimum the new Contractor needs a real Pilot Line during which he learns to accommodate the design *in his plant*.

The ability to increase production by quick start-up of another plant has been a pie-in-the-sky dream of military procurement for years. It was tried through the creation of identical plants during WWII. It fails because the people in the two plants cannot be made identical!

Bonus

Bonuses are useful incentives particularly when they are awarded after the fact for outstanding work accomplished. They are more complicated when they are promised in advance for future performance unless that performance is defined objectively. They are plain dangerous when the work to be rewarded cannot be measured objectively.

A Competitive Data Package is all but impossible to measure objectively. After one or two years of smooth production the package is *probably* competitive. After a year or two of smooth production *by the second contractor* the package *is* competitive. To expect more is to believe in fairy dust.

The bonus for the SM Data Package distorted the program. The Contractor focused on getting the money. The Navy focused on timely delivery. Unfortunately, no one focused on the content of the package.

Fixing Change Control

It is impossible to fix Change Control when the elements to be fixed are intrinsic to it. Volume will always be too high and span time will always be too long. Sometimes they can be improved through administrative adjustments but it will seldom gain more than 10% if that. More often efforts to fix intrinsic problems complicate the process, increase the cost, extend the time, and fail! Using the Change Control System to control volume is akin to tying-down the safety valve on a steam boiler. Sooner or later it blows up!

Volume, span time, and most other elements are driven by the operation of the plant and the conduct of the program. By the time they are evident as problems it's too late to correct them for the impacted program. If volume and span time are nominal, leave them alone! They are a normal part of any program. If they are excessive, look for the cause in (1) the conduct of the program, (2) the operation of the plant, and lastly (3) the Change Control System.

If a program contains requirements that will increase changes, such as compressed scheduling, either modify the requirements or accept the fact and plan on it. Don't deny it or hide from it. It will not go away. Nine times out of ten it is not fixable! Most people will not accept this fact so it cannot be taught. They simply have to live with it long enough to realize that it is real and unavoidable.

CM

As for CM, the Standard Missile Program was under way before CM was created. Even so, most of the elements of CM were used and the program illustrates many of the ways that management decisions impact Change Control.

"Alignment with need" was simply assumed as it always had been. There will be those who say that this proves that CM is unnecessary but that's too great a leap. The program's "need" was to stop the Kamikaze by making the APL concept work. It was clearly defined and widely known and there was no time for drift.

Does this mean that a clear need and a compressed schedule makes CM unnecessary? It's possible but unlikely. A clear need really means that CM is possible. A compressed schedule means trouble aplenty. Don't forget the overruns and the Claim!

CM program planning could have made the inherent complications more obvious. However, it's unlikely that they would have been accommodated. The Navy was fixated on achieving early competition no matter what. They got it but at an enormous cost in money, stress, and reputation.

CM was very much in being when the Standard ARM came along. Once again time was a major constraint. A compressed schedule was unavoidable. But this time the Navy's primary goal was an effective solution to a raging problem. They cooperated to develop appropriate accommodations, the greatest of which was high cost.

It may sound odd but it is often possible to substitute higher cost for time. Higher cost makes possible the assignment of more people and people of greater capability (higher paid) to the problem. The search for lowest cost parts and materials is dropped in favor of availability of those that work. Greater risks are taken. If they turn to junk they are simply scrapped and something else is tried. Economy of operation is subordinate to effective operation. These things, skillfully managed, do save time. The change rate will be high but expected. Little time is lost in emoting about it.

The Standard ARM program illustrates that CM can be effectively tailored to fit almost any set of conditions if the tailors know what they are doing and work at it in good faith.

Competition wasn't a consideration until the design problem was solved. However, when it appeared it became an almost insane fixation all over again. Nothing had been learned from the Standard Missile's problems. After the storm had abated somewhat, we set about solving the problem but the amiable relationship that existed during the Development Program was gone forever.

***

Afterword

Be mindful of the fact that perception depends upon perspective. The Tale of Two Programs has been told from my perspective. There are others. Many of the decisions that appear to have been irresponsible may be justified if seen from the perspective of upper management. Even so, decisions have consequences. They must be accommodated or suffered.

The Standard Missile program suffered them in ignorance for almost 30 years. The Standard ARM Development Program accommodated them and produced amazing results. The Standard ARM Production Program tried to suppress them and suffered accordingly.

Perhaps the most important lesson to be learned is that CM is not a task for novices. It requires a thorough understanding of the industrial plant and the way it operates. Without that understanding it provides little more than expensive window dressing, comforting to some but of little other benefit.

***