

White Paper
CMII 860B

**Comparison of DoD and
CMII Development Processes**



Prepared by
CMII Research Institute

White Paper CMII-860B
Comparison of DoD and CMII Development Processes

Revision Record

<i>Revision</i>	A	B	
<i>Released by</i>	WWG	WWG	
<i>Release date</i>	01/31/08	12/29/09	
<i>Authority</i>	062-WP	063-WP	
Page 1	A	B	
Page 2	A	B	
Page 3	A	B	
Page 4	A	B	
Page 5	A	B	
Page 6	A	B	
Page 7	A	B	
Page 8	A	B	
Page 9	A	B	
Page 10		B	

Purpose of This White Paper

The DoD and CMII development processes are both comprised of four phases or tiers but that is where the similarity ends. This paper serves to describe the major differences.

OUTLINE

- **The DoD Procurement Life Cycle**
 - **Developmental Baselines, Reviews & Audits**
 - **Developmental Phases for Hardware Vs Software**
- **The Four-Tier CMII Development Process**
- **Development Resources By Step Within Each Tier**
- **The CMII V-Model for Development**
- **How the Two Development Approaches Differ**
- **Summary, Conclusions and Recommendations**

The DOD Procurement Life Cycle

DOD Developmental Baselines, Reviews & Audits

The DOD procurement life cycle (or development process) is summarized in a 1-page chart on page 3. It begins with a concept and continues through three milestones represented by three baselines:

- Milestone I is an approved functional baseline.
- Milestone II is an approved allocated baseline.
- Milestone III is an approved product baseline.

The milestone chart is from a book written by Tom Samaras. The latest Defense Acquisition Milestones and Phases provided in DoDI 5000.2, were not used for this comparison because they are at a high level and the Samaras chart provides better insight to the underlying processes.

The Samaras chart provides a good overview of developmental baselines, reviews and audits. The success of each development phase is confirmed with one or more design reviews. Functional and physical configuration audits thereafter serve to confirm that each developed module (or CI) conforms to its design requirements.

Development Phases for Hardware Vs Software

Software represents an increasingly important part of modern weapon systems. Software development also continues to be a major challenge. The MIL-STDs for development did not differentiate between hardware and software prior to DOD-STD-2167. Software developers were expected to follow the processes used to develop hardware.

The chart on page 4 provides a comparison of the hardware and software development phases. Developmental hardware is fabricated in a single phase followed by HWCI testing. Software is constructed in two phases (1) CSU coding and testing and (2) CSC integration and testing, followed by a CSCI testing phase.

The design reviews and configuration audits are shown to be essentially the same for both software and hardware development. The reviews and audits serve to validate achievement of each phase and, in many cases, satisfy the requirements for additional funding.

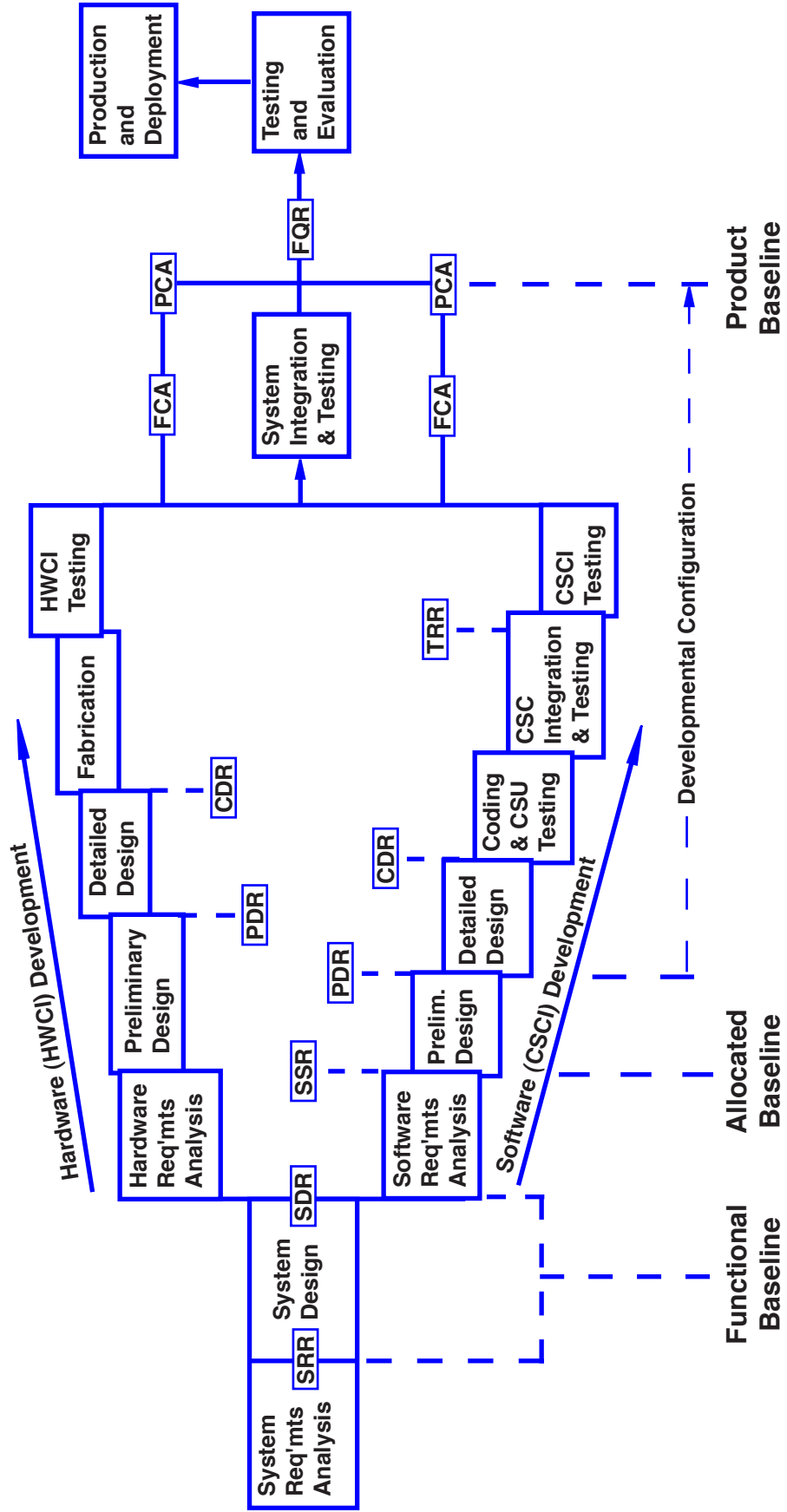
DOD Developmental Baselines, Reviews & Audits

CM Deskbook by T. Samaras, 1988

NEW SYSTEM	MILESTONE I	MILESTONE II	MILESTONE III
PHASES	CONCEPT EXPLANATION	DEMONSTRATION AND VALIDATION	FULL SCALE DEVELOPMENT
PURPOSE OF PHASE	<ul style="list-style-type: none"> Conceptual Studies Investigation of Alternative Concepts and Approaches 	<ul style="list-style-type: none"> Competitive Demonstration Definition & Analysis of System Alternatives Select Preferred System 	<ul style="list-style-type: none"> Design Development of Selected System Test and Evaluation of System Elements
REVIEWS & AUDITS	<div style="border: 1px solid black; padding: 2px; display: inline-block;">SRR</div> <div style="border: 1px solid black; padding: 2px; display: inline-block;">SDR</div>	<div style="border: 1px solid black; padding: 2px; display: inline-block;">PDR_s</div> <div style="border: 1px solid black; padding: 2px; display: inline-block;">SSR</div>	<div style="border: 1px solid black; padding: 2px; display: inline-block;">CDR_s</div> <div style="border: 1px solid black; padding: 2px; display: inline-block;">TRR_s</div> <div style="border: 1px solid black; padding: 2px; display: inline-block;">PRR_s</div> <div style="border: 1px solid black; padding: 2px; display: inline-block;">FCA_s</div> <div style="border: 1px solid black; padding: 2px; display: inline-block;">PCA_s</div> <div style="border: 1px solid black; padding: 2px; display: inline-block;">FQR</div>
BASELINES	Functional	Allocated	Product
ENGINEERING EMPHASIS	<ul style="list-style-type: none"> Operational Analysis and Allocation Alternative System Concept Synthesis Technology Assessment Risk Analysis Logistic Support Analysis System and Cost Effectiveness Analysis System Level Trades 	<ul style="list-style-type: none"> Requirement Analysis and Allocation Definition and Evaluation of Alternative Systems Recommendation of Preferred System Risk Reduction Cost and Schedule Validation Development of FSD Program Plans Element Level Trades 	<ul style="list-style-type: none"> Detailed Design of Selected System Engineering Development Testing (DT&E) (IQT&E) Software Design, Coding and Testing Design Optimization Development of Production and Deployment Plans Complete Logistical Support Designs

SRR - System Requirements Review SDR - System Design Review SSR - Software Specification Review PDR - Preliminary Design Review	CRD - Critical Design Review TRR - Test Readiness Review FCA - Functional Configuration Audit PRR - Production Readiness Review PCA - Physical Configuration Audit FQR - Formal Qualification Review DT&E - Development Test & Evaluation IQT&E - Initial Qualification Test and Evaluation
--	--

Development Phases for Hardware Vs Software



- SRR - System Requirements Review
- SDR - System Design Review
- SSR - Software Specification Review
- PDR - Preliminary Design Review
- CRD - Critical Design Review
- TRR - Test Readiness Review
- FCA - Functional Configuration Audit
- PCA - Physical Configuration Audit
- FQR - Formal Qualification Review
- CSU - Computer Software Unit
- CSC - Computer Software Component
- CSCI - Computer Software CI
- HWCi - Hardware CI

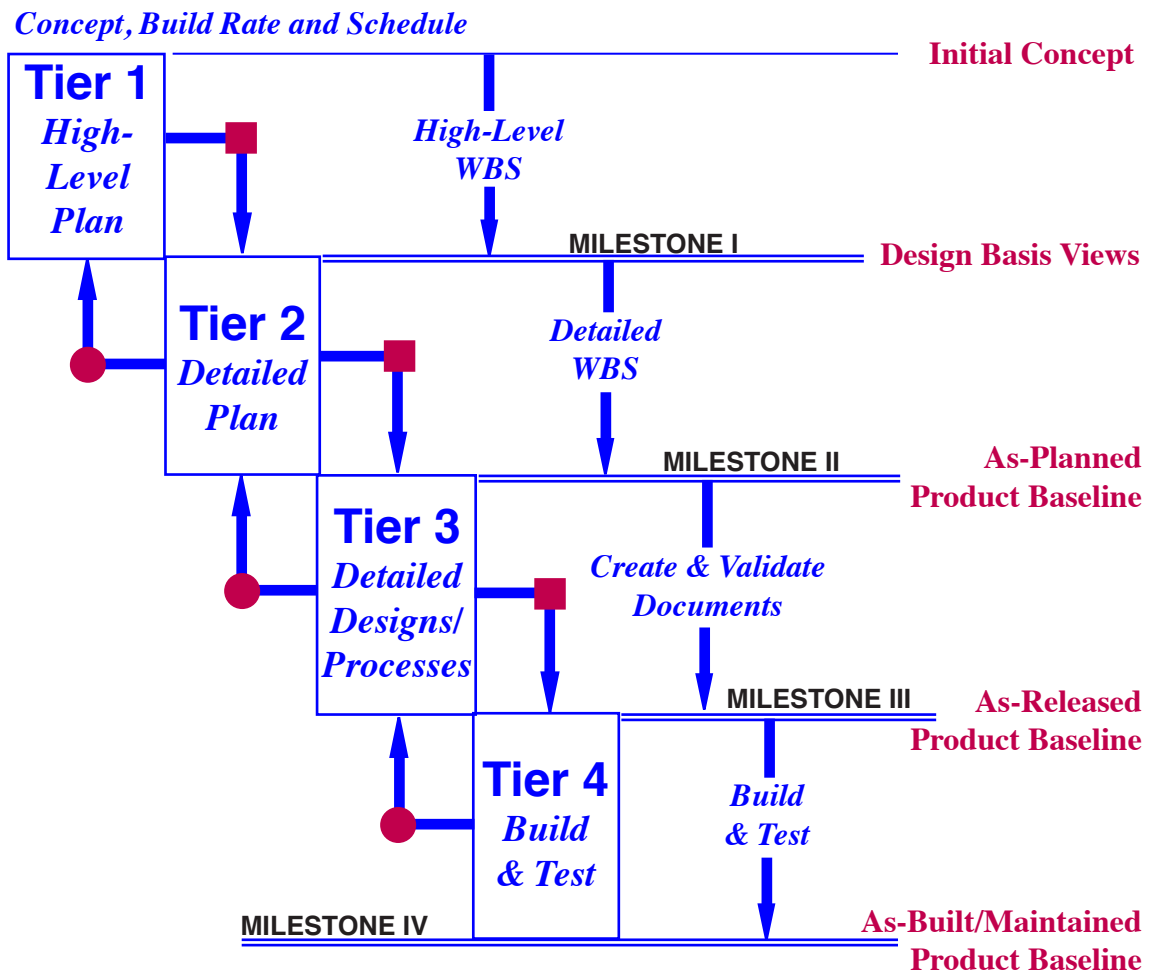
DOD-STD-2167A Defense System Software Development

The Four-Tier CMII Development Process

The CMII development process is comprised of four tiers. Satisfactory completion of Tiers 1, 2, 3 and 4 achieves Milestones I, II, III and IV, as shown below.

With CMII, end-item application requirements are defined and its design basis is established at Milestone I. An as-planned product baseline is achieved at Milestone II. An as-released version and the as-built/as-maintained version are achieved at Milestones III and IV.

Any uncertainty about the concept is resolved while creating the high-level plan. A high-level work breakdown structure (WBS) is used to create the high-level plan. A detailed WBS is used to extend the high-level plan into a detailed plan. Minor spiraling between levels is expected.



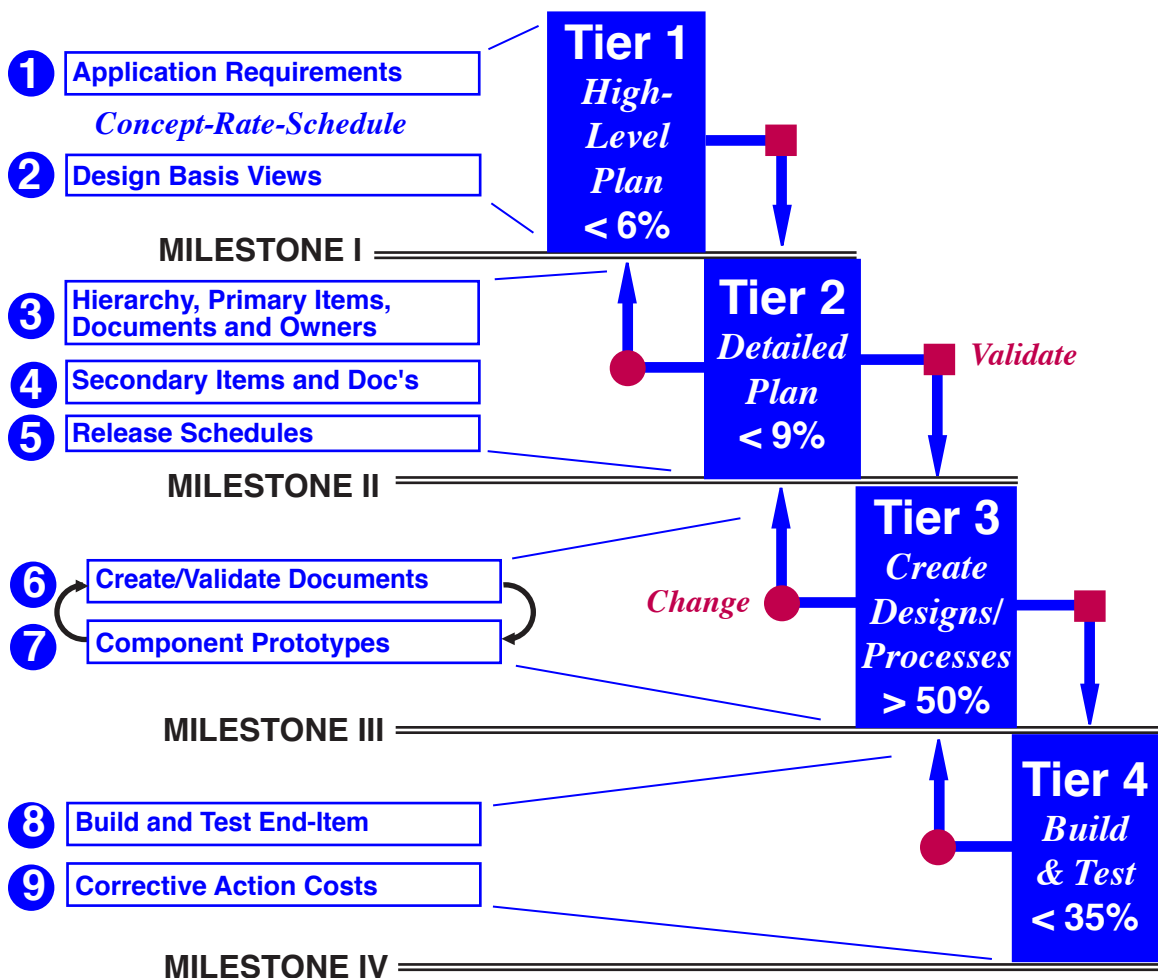
Development Resources By Step Within Each Tier

The four-tier CMII development process is comprised of 9 steps. Step 1 serves to define the application requirements. Step 2 serves to create the design basis views. These first two steps are most important since they consume less than 6% of the total development costs and lock-in over 90% of the total life cycle costs.

The second tier contains steps 3, 4 and 5 which serve to create the detail plan and consume no more than 9% of the development costs.

Steps 6 and 7 are the most resource intensive. Step 6 serves to create and validate the detailed designs and processes. Step 7 is used to build prototypes as needed to further validate the detailed documents.

The end-item is built and tested in Step 8. Corrective action costs are collected in Step 9 and should not exceed 10% of the development costs.



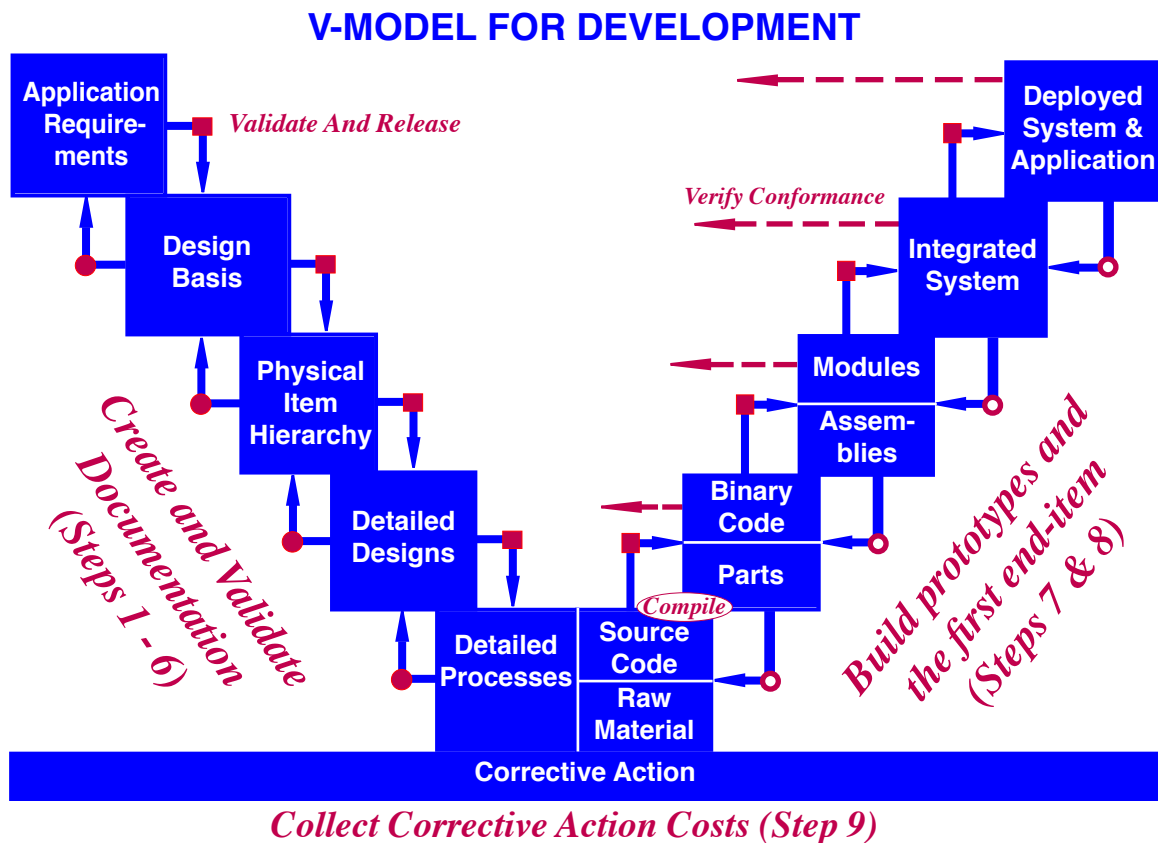
The CMII "V" Model For Development

The CMII development process represents a V-model as shown below. Designs and processes are created in the left leg. Hardware and software items are produced in the right leg.

The left leg of the V-model will experience minor spiraling at each level as designs and processes evolve. With proper validation processes in place, changes at one level should not ripple above the parent level. The extra costs to resolve those that do represent corrective action.

As-built items on the right must conform to their respective requirements on the left. Any nonconformances detected in the right leg are symptoms of deficiencies in the documents created in the left leg. The extra effort required to fix a nonconforming item is corrective action.

The primary product of engineering is documentation. The purpose of a prototype is to validate the documentation. The most reliable and efficient developers are those who can accommodate change and keep their documented requirements clear, concise and valid.



How the Two Development Approaches Differ

DOD Versus CMII Baselines and Milestones

The DoD's functional and allocated baselines, once approved, are fixed. The current configuration is the approved baseline plus approved changes. Baselined documents, however, may have unincorporated changes. Overall efficiency is compromised accordingly.

The CMII baseline is a moving baseline which ratchets forward with each change. Each document, once released, is up-to-date. There are no unincorporated changes. CMII achieves the initial Product Baseline at Milestone II rather than Milestone III.

Design Reviews and Audits

Formal design reviews and audits in a DOD development effort serve to validate or verify something that should already be known. Follow-on funding, however, is often tied to formal reviews and/or audits.

With CMII, design reviews and configuration audits are continuous and built into each step. Proof of accomplishment is retained in the records. Additional reviews and audits would serve no purpose.

Quality of Design Documents

A document with unincorporated changes is not a quality document. Each document must be updated with each change and validated by its creator and one or more designated users. With proper validation, the need for deviations and/or waivers is essentially eliminated.

Configuration Items (CIs)

Assigning CI numbers to developmental items served two purposes. Only items with a CI number are placed under configuration management controls. Physical items are serialized against their CI number.

With CMII, there are no CIs. Everything is subjected to configuration management controls. Items are serialized against model numbers. CIs serve no useful purpose.

The Change Process

The DOD process focuses on controlling and statusing changes rather than accommodating change. As a result, documents are rarely up-to-date. That, in turn, is why product quality suffers and why so many deviation and waivers are needed.

Summary, Conclusions and Recommendations

Summary

The DoD uses a 4-phase development process with 3 milestones. The concept phase concludes with a functional baseline, which is Milestone I. The demonstration and validation phase concludes with an allocated baseline, which is Milestone II. The full scale development phase concludes with a Product Baseline, which is Milestone III. Production and Deployment, which follow, represent the fourth phase.

Development, per CMII, is a 4-tier, 9-step effort. The first tier serves to create the high-level plan (Milestone I). The second tier serves to create the as-planned product baseline (Milestone II). The third tier serves to create the as-released product baseline (Milestone III). The fourth tier serves to create the as-built/as-maintained product baseline.

Major differences in the DoD versus CMII approaches:

- fixed baselines versus moving baselines;
- design reviews and audits — extra effort versus built-in;
- configuration items (CIs) — required versus not used;
- change process — slow and costly versus fast and efficient;
- documentation rules and ownership — weak versus strong;
- Product baseline achieved at Milestone II versus Milestone III.

Conclusions and Recommendations

The CMII development process is much more robust than the DoD development process. Moving baselines are more efficient than fixed baselines. Any organization that cannot accommodate change and keep requirements clear, concise and valid is destined to operate in the corrective action mode — which leads to poor quality, missed schedules and budget overruns.

There are no rules that prevent a DoD contractor from using practices proven to be more effective. There are no rules that prevent a DoD contractor from accommodating change and keeping requirements clear, concise and valid. Good internal controls and records are better than reviews and audits for demonstrating conformance to requirements.