14th Annual NDIA Systems Engineering Conference 2011

A Fresh Look at the Software Technical Review Process

Dr. William Bail

The MITRE Corporation

NAVSEA PEO IWS

The University of Maryland

26 Oct 2011

The authors' affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions or view points expressed by these authors.

Agenda

- The problem
- Current practice and standards
- What really happens
- Recommendations

Reviews

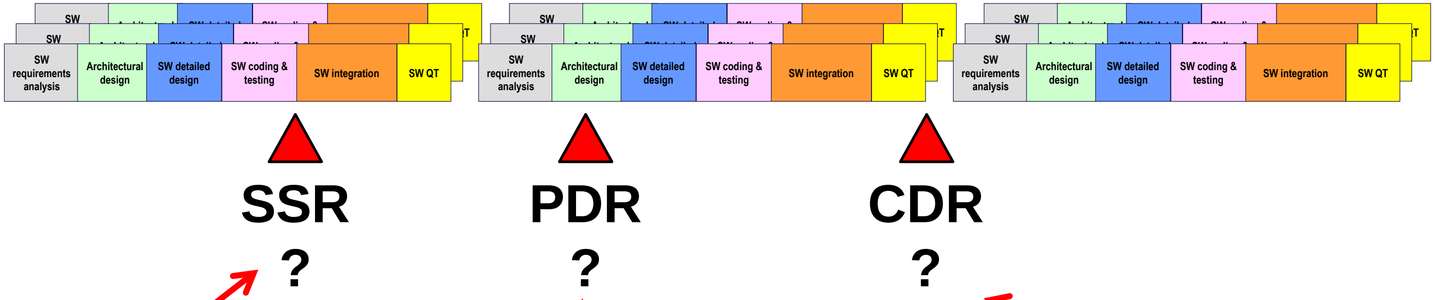
- **Meetings where developers and stakeholders discuss project status, progress, risks, and potential changes**
- **Purpose: to provide opportunity for stakeholders**
  - to determine state of development by
    - assessing progress
    - assessing risks
    - assessing emerging system development approach / design
  - to affirm approach or redirect development effort by
    - re-evaluating priorities
    - mitigating risks
    - revising engineering plan or approach
    - reassessing requirements
  - to authorize continued work by
    - establishing formal baselines
    - identifying any required corrective actions

The problem

- ❑ **Engineering reviews are often based on legacy processes**
  - ➢ which were largely based on the Waterfall process model
- ❑ **Modern development processes are significantly different and improved**
- ❑ **Result is a mismatch between the review approach and the development approach**
  - ➢ Government reviewers expect one thing, developers present another
- ❑ **Examples**
  - ➢ Developer follows an incremental development approach, resulting in some SW builds prior to PDR
    - » PDR criteria fails to include test results from early builds
  - ➢ Developer follows an iterative development approach, resulting in evolving SW requirements (not all defined up-front)
    - » System fails SSR due to incomplete requirements

The challenge

- ❑ **Suppose a program is following an incremental approach**
  - ➢ Multiple SW builds/increments over time
- ❑ **When is best time to hold the reviews?**



SSR ?

PDR ?

CDR ?

But requirements analysis not complete

But architectural design not complete and already has code and test results

But detailed design not complete and already have code and test results

Impacts of the mismatch

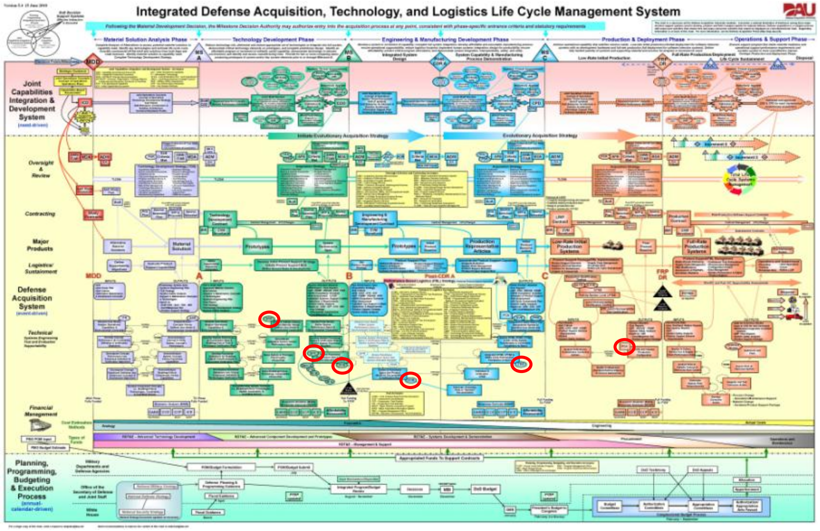
- Reviews tend to be less useful than they could be
- Government customers become frustrated because they don't hear what they are expecting
- Contractors end up spending effort on non-productive tasks just to satisfy legacy expectations
- Excessive effort is spent on formatted deliverables
  - By the Contractor – in developing them
  - By the Government – in reviewing them
- Government focus is misdirected, frittering away valuable resources
- Important information regarding progress and quality is overlooked
  - Potentially raising risk

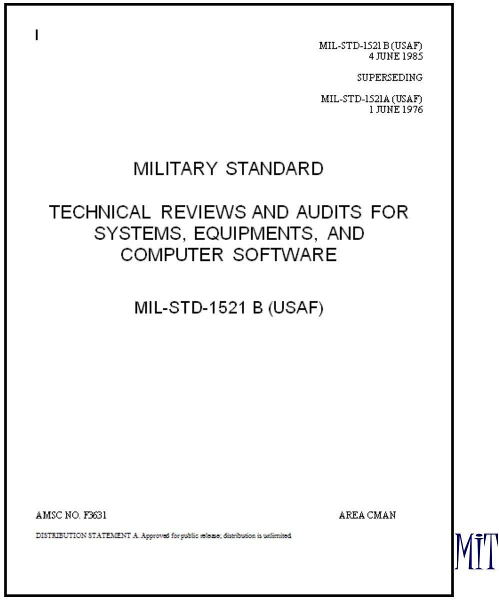
Agenda

- ❑ The problem
- ➡ ❑ Current practice and standards
- ❑ What really happens
- ❑ Recommendations

Drivers

- □ JCIDS
- □ MIL-STDs 498 and 2167
- □ Systems Engineering V-chart
- □ MIL-STD-1521B
- □ 1521B defines the following reviews:
  - ➢ System Requirements Review (SRR)
  - ➢ System Design Review (SDR)
  - ➢ Software Specification Review (SSR)
  - ➢ Preliminary Design Review (PDR)
  - ➢ Critical Design Review (CDR)
  - ➢ Test Readiness Review (TRR)
  - ➢ Functional Configuration Audit (FCA)
  - ➢ Physical Configuration Audit (PCA)
  - ➢ Formal Qualification Review (FQR)
  - ➢ Production Readiness Review (PRR)


Integrated Defense Acquisition, Technology, and Logistics Life Cycle Management System
DAU
Joint Capabilities Integration & Development System
Oversight & Review
Contracting
Major Products
Logistics/ Sustainment
Defense Acquisition System
Technical
Financial Management
Planning, Programming, Budgeting & Execution Process



I

MIL-STD-1521 B (USAF)
4 JUNE 1985

SUPERSEDING

MIL-STD-1521A (USAF)
1 JUNE 1976

MILITARY STANDARD

TECHNICAL REVIEWS AND AUDITS FOR
SYSTEMS, EQUIPMENTS, AND
COMPUTER SOFTWARE

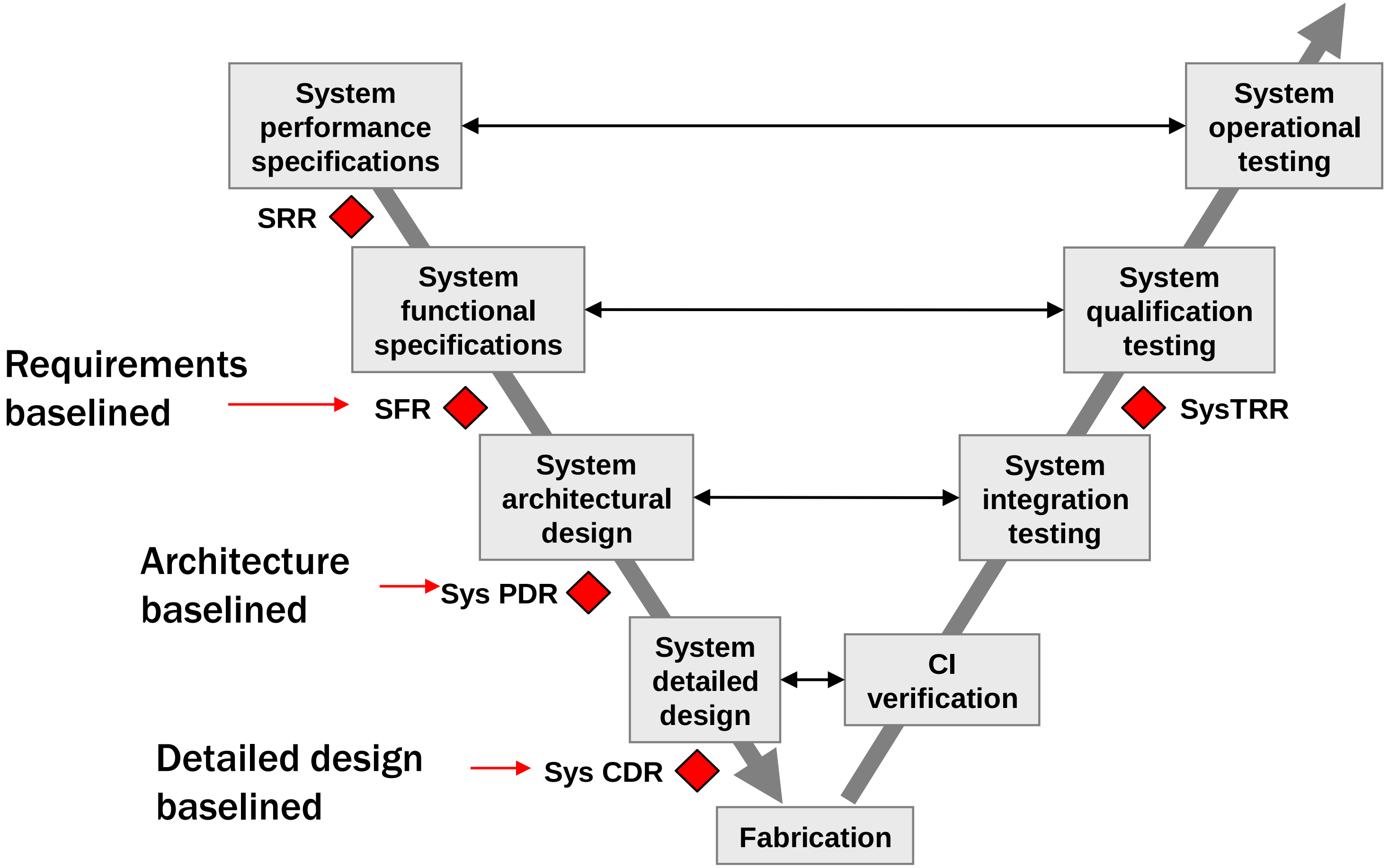
MIL-STD-1521 B (USAF)

AMSC NO. F3631 AREA CMAN

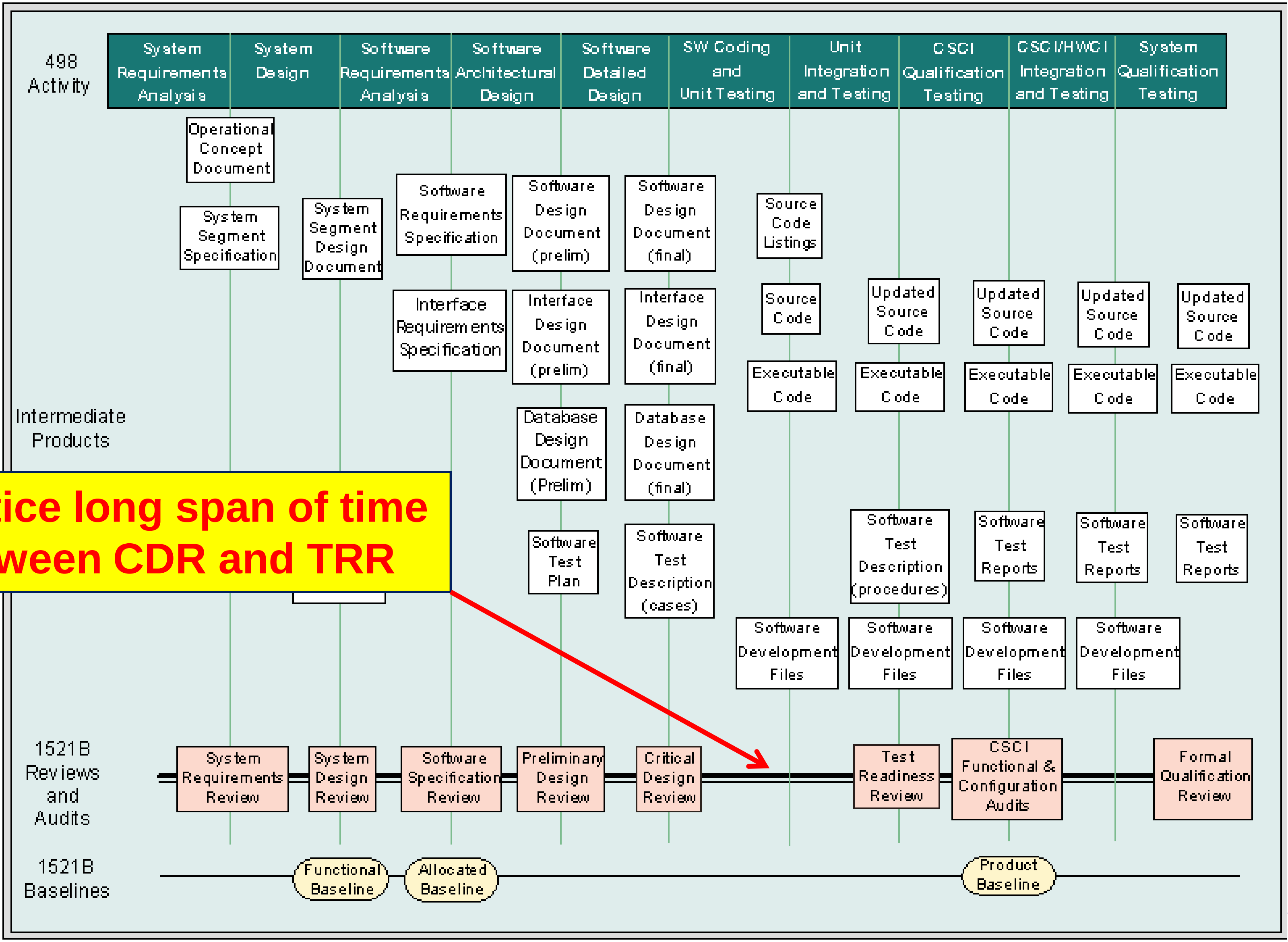
DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited

Underlying model

- ❑ Based on engineering V chart
- ❑ Relies on a sequential set of events, progressive in nature


System performance specifications
System operational testing
SRR
System functional specifications
System qualification testing
Requirements baselined
SFR
SysTRR
System architectural design
System integration testing
Architecture baselined
Sys PDR
System detailed design
CI verification
Detailed design baselined
Sys CDR
Fabrication

Sample MIL-STD life cycle



498 Activity

System Requirements Analysis | System Design | Software Requirements Analysis | Software Architectural Design | Software Detailed Design | SW Coding and Unit Testing | Unit Integration and Testing | CSCI Qualification Testing | CSCI/HWCI Integration and Testing | System Qualification Testing

Intermediate Products

Operational Concept Document

System Segment Specification

System Segment Design Document

Software Requirements Specification

Interface Requirements Specification

Software Design Document (prelim)

Interface Design Document (prelim)

Database Design Document (Prelim)

Software Test Plan

Software Design Document (final)

Interface Design Document (final)

Database Design Document (final)

Software Test Description (cases)

Source Code Listings

Source Code

Executable Code

Software Development Files

Updated Source Code

Executable Code

Software Test Description (procedures)

Software Development Files

Updated Source Code

Executable Code

Software Test Reports

Software Development Files

Updated Source Code

Executable Code

Software Test Reports

Software Development Files

Updated Source Code

Executable Code

Software Test Reports

1521B Reviews and Audits

System Requirements Review

System Design Review

Software Specification Review

Preliminary Design Review

Critical Design Review

Test Readiness Review

CSCI Functional & Configuration Audits

Formal Qualification Review

1521B Baselines

Functional Baseline

Allocated Baseline

Product Baseline

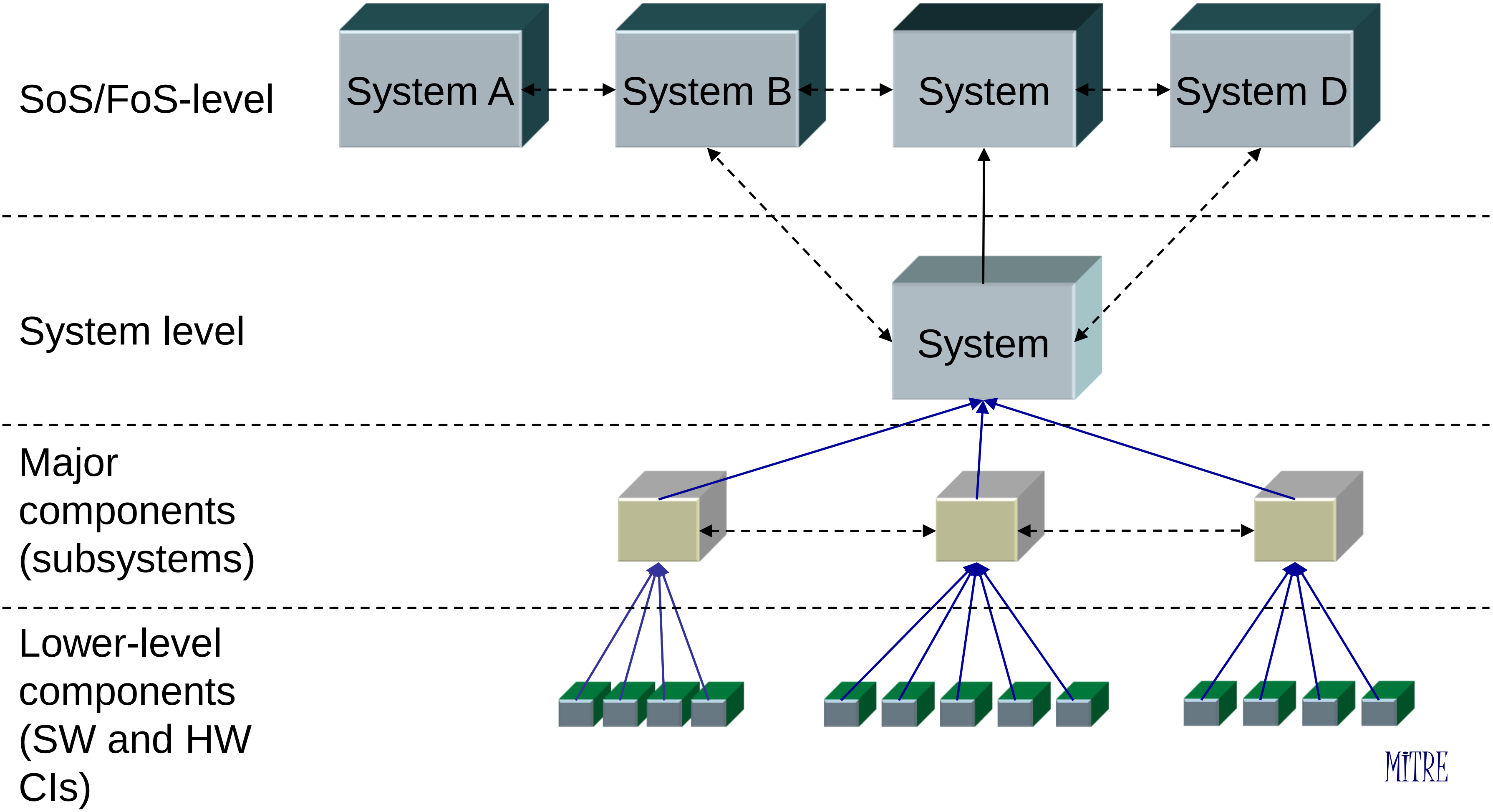
Notice long span of time between CDR and TRR

Agenda

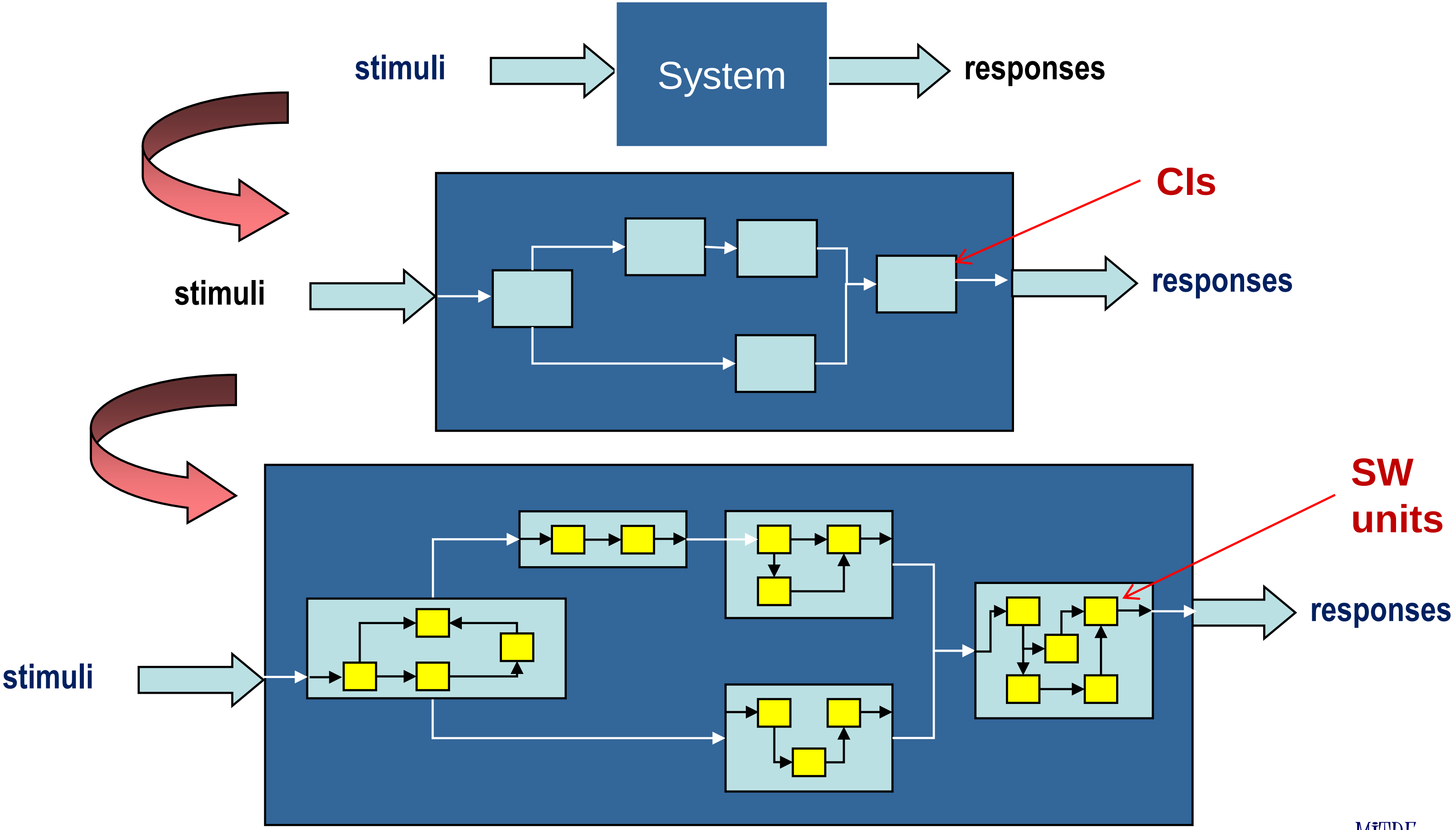
- ❑ The problem
- ❑ Current practice and standards
- ➡ ❑ What really happens
- ❑ Recommendations

SW in relation to the system

- ❑ SW is developed as part of a larger system, each level having its own requirements, architecture, and design

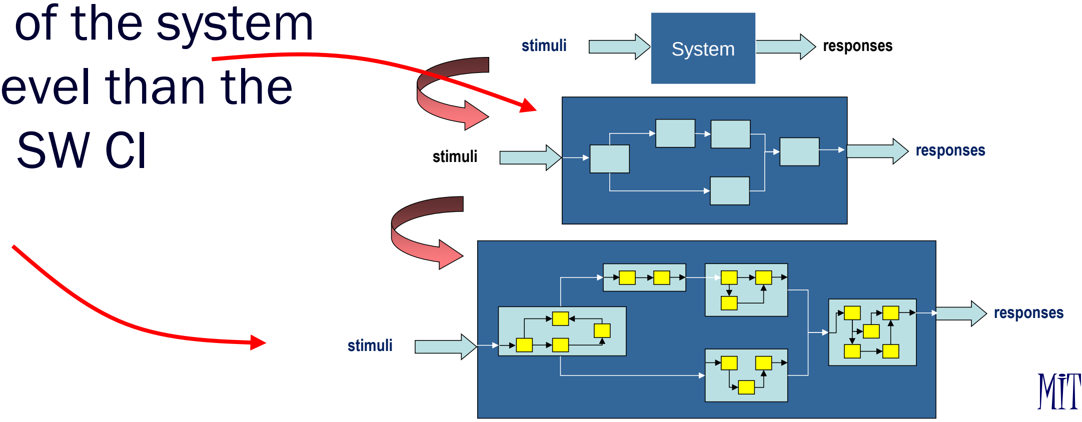

SoS/FoS-level
System A
System B
System
System D
System level
System
Major components (subsystems)
Lower-level components (SW and HW CIs)

Levels of design



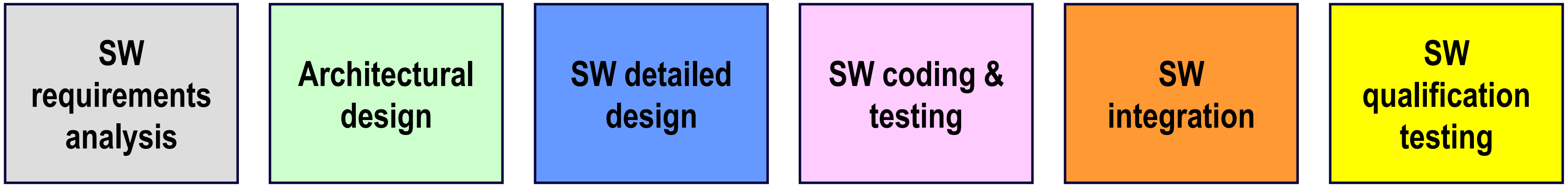
Levels of system design

- **Each level has its own design process**
  - System requirements → system architecture → system design → system integration → system verification
  - Subsystem requirements → subsystem architecture → subsystem design → subsystem integration → subsystem verification
  - CI requirements → CI architecture → CI design → CI integration → CI verification
- **A review of the system architecture (e.g., at a System PDR) is fundamentally different than a review of the architecture for a SW CI PDR**
  - The architecture of the system is at a different level than the architecture of a SW CI



SW SCI development activities

- All SW is developed using the activities defined in IEEE/EIA 12207.0 regardless of the process followed
- When these are performed is defined by the overall process followed
  - e.g., projects can perform requirements analysis towards end of development, concurrently with integration and test
  - e.g., user interfaces can be defined late in development, after sufficient usability analysis has been performed – still a part of requirements definition
  - e.g., when coding during an early build, activity is still part of code and unit test



SW requirements analysis

Architectural design

SW detailed design

SW coding & testing

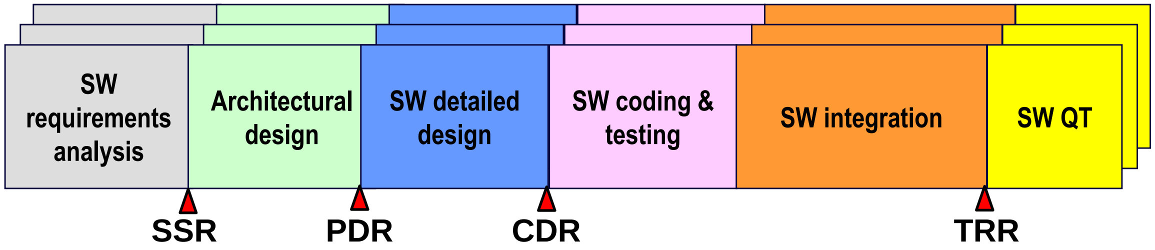
SW integration

SW qualification testing

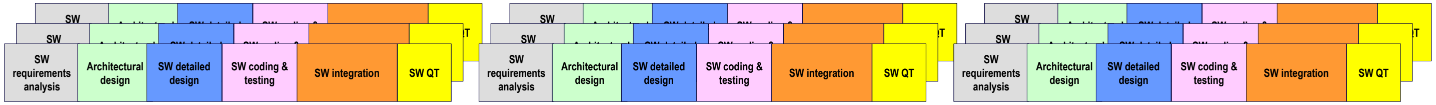
For each SCI

Some alternative development strategies

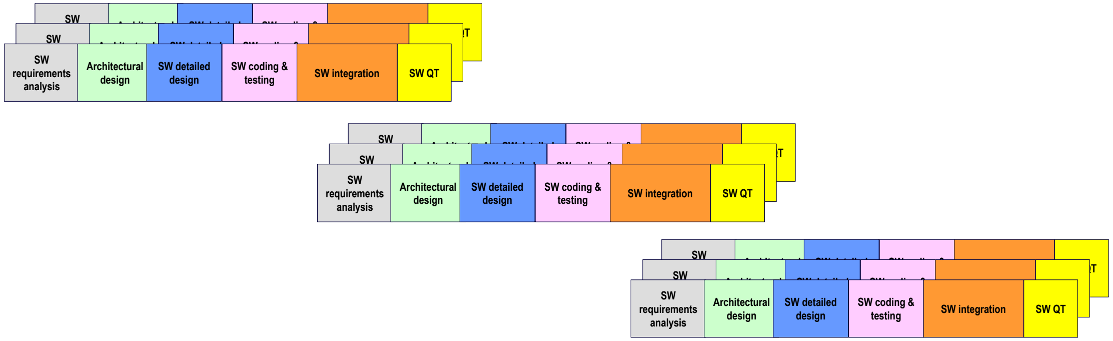
Waterfall



Incremental
– no overlap

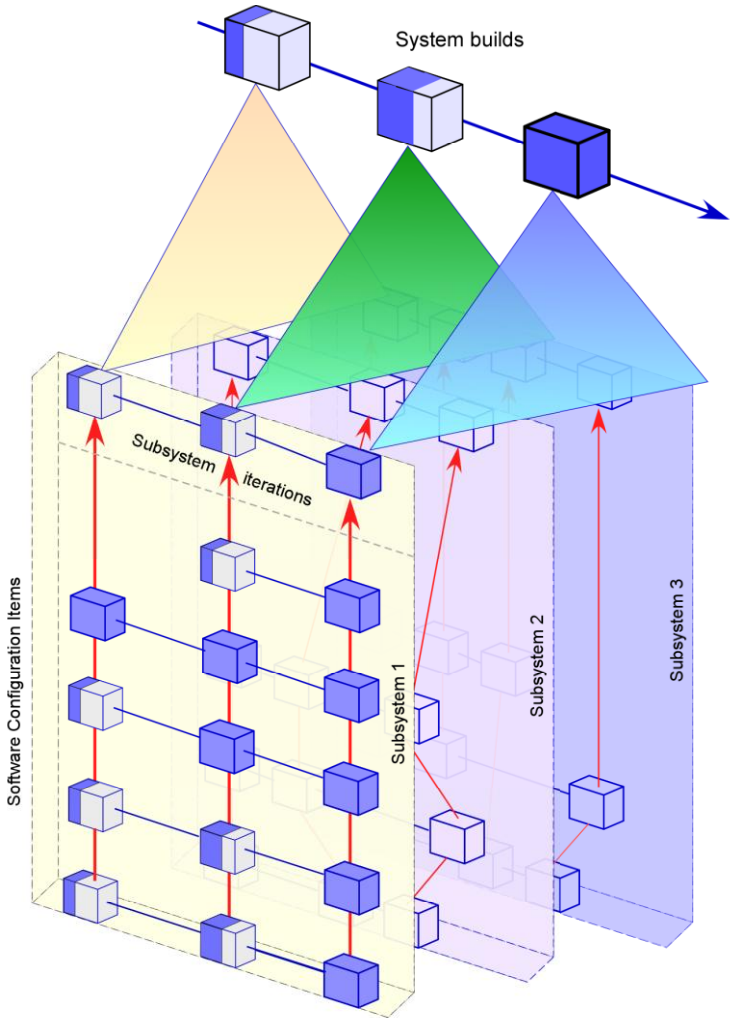


Incremental
– overlap



SW developed incrementally

- ❑ Development programs have multiple levels
  - ➢ System level
  - ➢ Subsystem level
  - ➢ CI level
- ❑ CIs developed incrementally feed into incremental integrations for subsystems
- ❑ Subsystems developed incrementally feed into incremental integration for the system



MITRE

Agenda

- ❑ The problem
- ❑ Current practice and standards
- ❑ What really happens
- ➡ ❑ Recommendations

Perspective

- **Experience and lessons-learned have resulted in current best practice to place increasing reliance on early builds of system**
  - Risk reduction by exploration of requirements and design via early implementation (serving as prototypes)
  - Provides for early deployable systems as off-ramps
  - Provides early and on-going insight into progress and emerging risk
- **Important to exploit early increment experience as a key source of information for assessing progress and maturity**
  - Using an incremental approach to programmatic and technical reviews

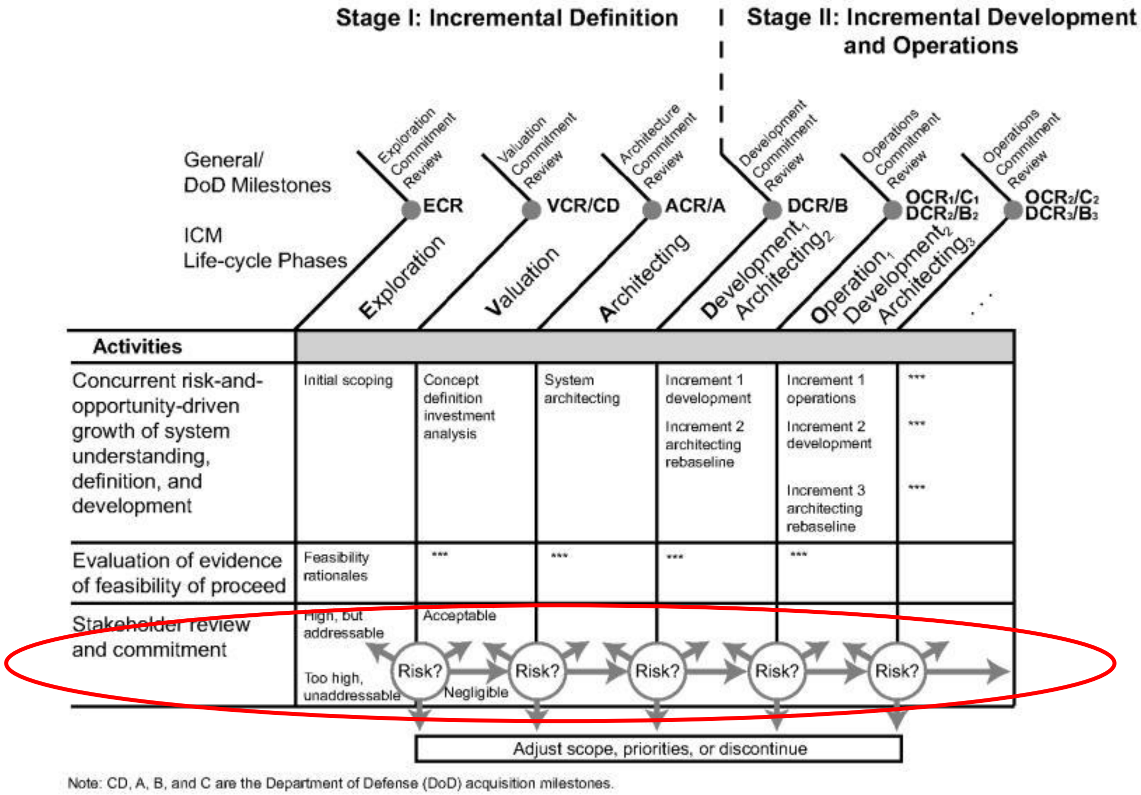
Overall recommendations

- ❑ **Encourage (via RFP/SOW) risk-aware development processes, including incremental approaches – select those that best suit needs of system to be developed**
  - ➢ Especially the Incremental Commitment Model
- ❑ **Plan acquisition to incorporate reviews**
  - ➢ At the right time
  - ➢ Focused on maximizing insight and likelihood of identifying risks before they become problems
- ❑ **Ensure that nature and general scheduling of reviews is clearly defined in the SOW**
  - ➢ With specific need dates for system capabilities
  - ➢ Require periodic technical reviews to correspond to contractor development events
- ❑ **Define criteria for content and scheduling of reviews up-front to ensure adequate review coverage is achievable**

Incremental commitment model

- ❑ Overall concept of risk reduction via incremental development clearly described by the Incremental Commitment Model (ICM)

Boehm, Barry, and Jo Ann Lane. "Using the Incremental Commitment Model to Integrate System Acquisition, Systems Engineering, and Software Engineering". *STSC Crosstalk* Oct 2007


Stage I: Incremental Definition
Stage II: Incremental Development and Operations
General/ DoD Milestones
Exploration Commitment Review
ECR
Valuation Commitment Review
VCR/CD
Architecture Commitment Review
ACR/A
Development Commitment Review
DCR/B
Operations Commitment Review
OCR$_1$/C$_1$ DCR$_2$/B$_2$
Operations Commitment Review
OCR$_2$/C$_2$ DCR$_3$/B$_3$
ICM Life-cycle Phases
Risk?
Risk?
Risk?
Risk?
Risk?

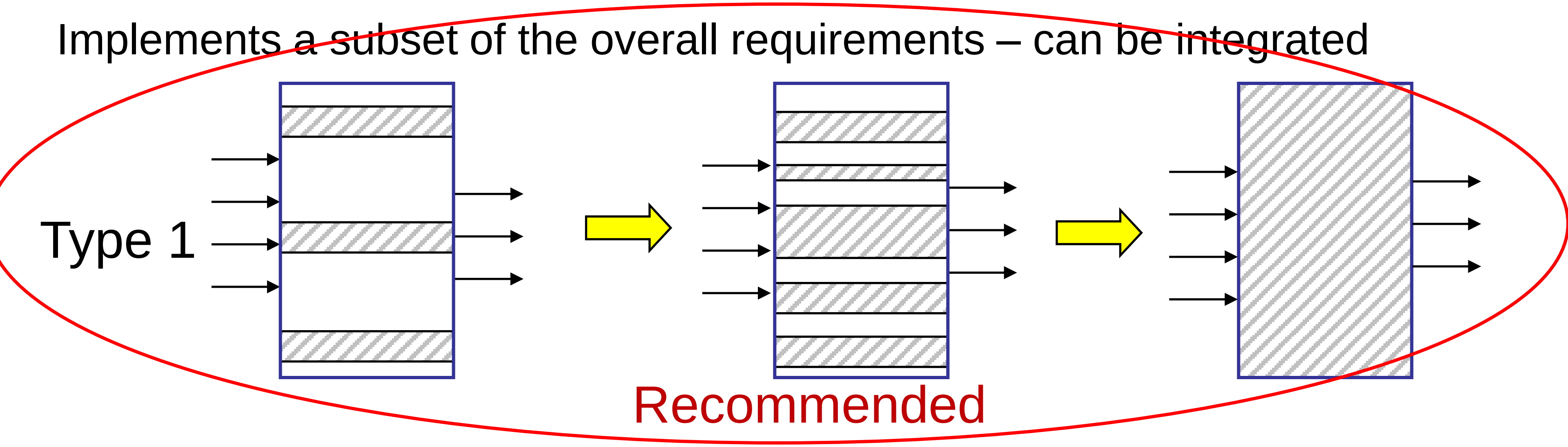
Activities	Exploration	Valuation	Architecting	Development$_1$ Architecting$_2$	Operation$_1$ Development$_2$ Architecting$_3$	. . .
Concurrent risk-and-opportunity-driven growth of system understanding, definition, and development	Initial scoping	Concept definition investment analysis	System architecting	Increment 1 development Increment 2 architecting rebaseline	Increment 1 operations Increment 2 development Increment 3 architecting rebaseline	*** *** ***
Evaluation of evidence of feasibility of proceed	Feasibility rationales	***	***	***	***	
Stakeholder review and commitment	High, but addressable Too high, unaddressable	Acceptable Negligible				

Adjust scope, priorities, or discontinue

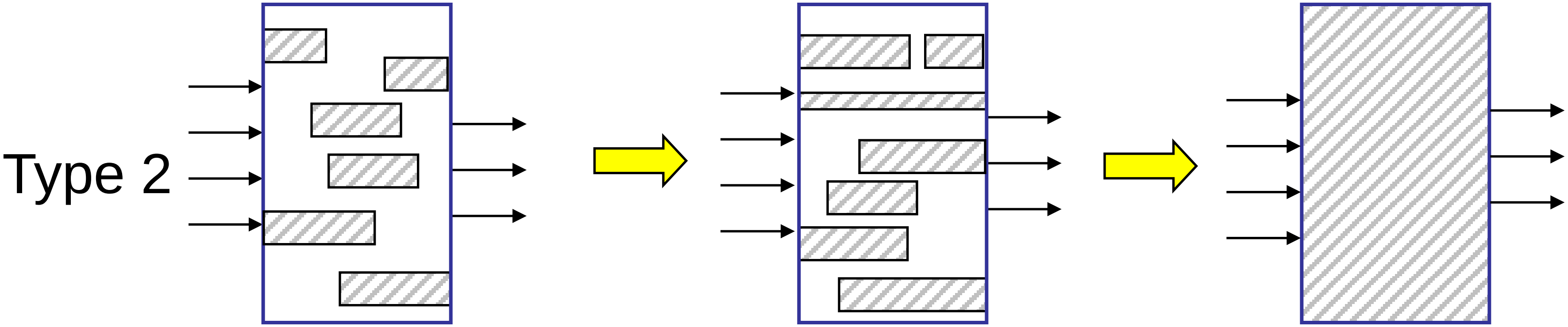
Note: CD, A, B, and C are the Department of Defense (DoD) acquisition milestones.

Two general approaches to iterations

Implements a subset of the overall requirements – can be integrated



Implements portions of the overall system, but not enough to integrate



Risk reduction opportunities

- ❑ **Situation 1 of 1,000,000**
  - ➢ Uncertain how to design a complex series of functions
  - ➢ Solution
    - » Start with core set of functions, build with multiple iterations, use lessons-learned to improve each iteration
- ❑ **Situation 2 of 1,000,000**
  - ➢ Uncertain about real requirements when interacting with “physics”
  - ➢ Solution
    - » Build initial solution with approximate parameters for first iteration
    - » Include data extraction as a part of design
    - » Use each iteration to refine knowledge
- ❑ **Result – gain benefits of prototyping without throwaway artifacts**
  - ➢ and maintain full quality of design and code

Overall recommendations *(cont'd)*

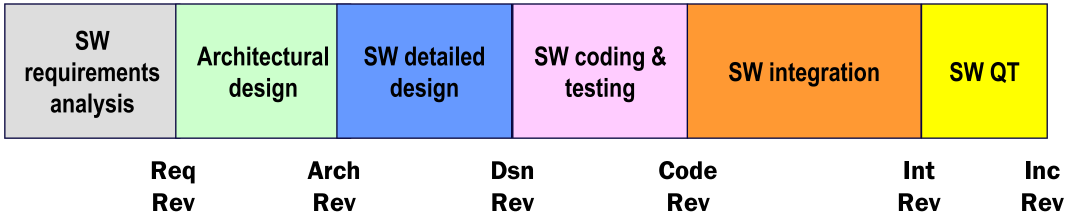
- **With incremental approach, require contractors to describe their planned development approach and the role of reviews, to include:**
  - Requirements refinement
  - Architectural and detailed design maturation
  - Code and test results
  - Changes to planned increment content
- **Require review content to provide information necessary to assess**
  - Actual progress vs planned progress
  - Actual product quality and content
  - Quality based on expectations
  - Risk of proceeding
- **Ensure that each level of review is defined to correspond to the information appropriate to that review**
  - SW PDR ≠ Sys PDR

Overall recommendations *(cont'd)*

- **At reviews, evaluate**
  - Results of previous increments
  - Progress for current increment
  - Plans for next increment
- **Select attendees at each review to correspond to the appropriate stakeholders**
- **Focus on key information at each level**
  - CI-level reviews – focus on the CI and how it is being developed in the context of the requirements allocated to it
  - Increment reviews – can be performed at all levels of increments – CIs, subsystems, and system
  - System-level reviews – focus on overall system and how the pieces are coming together
    - Correlate with system-level increments

Suggested scheduling

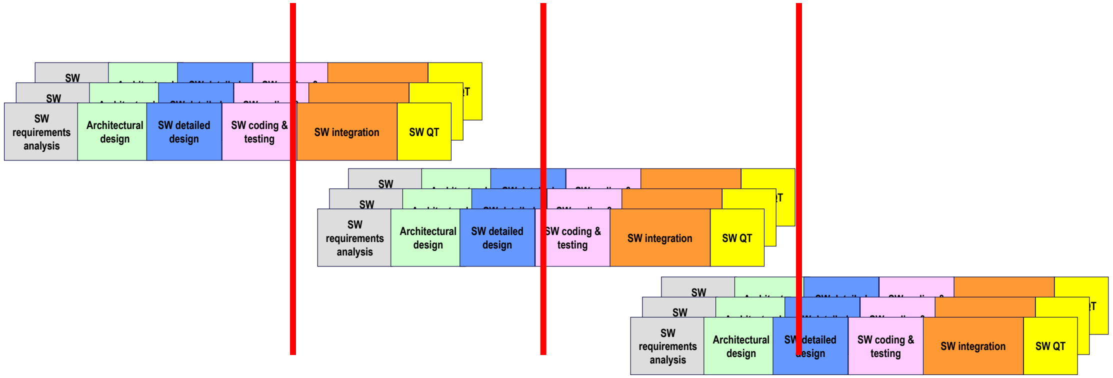
- ❑ **Planning review schedule**
  - ➢ Define reviews at all levels for each increment
    - » CI / subsystem / system
  - ➢ For SW CIs, hold review after each activity



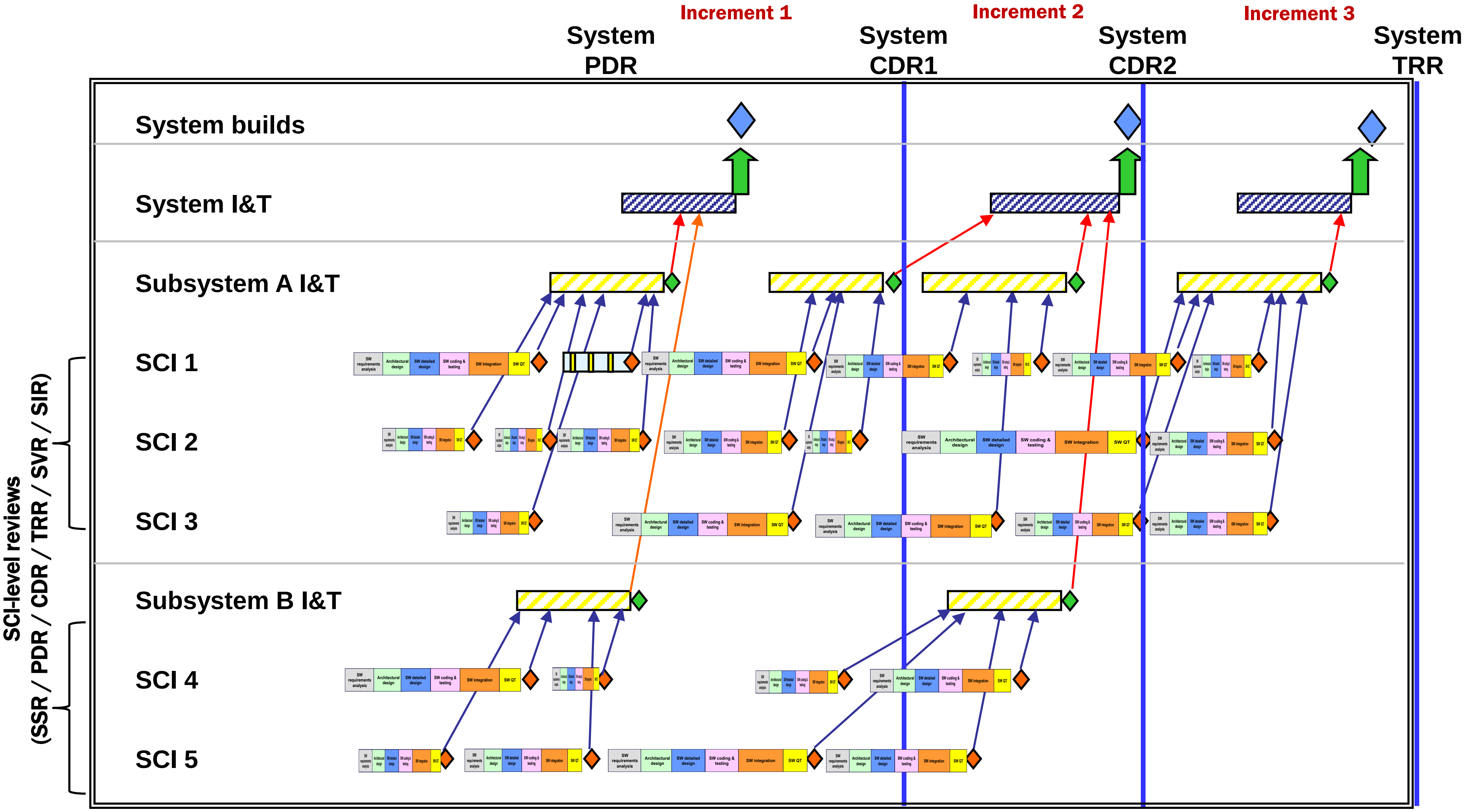
SW requirements analysis | Architectural design | SW detailed design | SW coding & testing | SW integration | SW QT

Req Rev | Arch Rev | Dsn Rev | Code Rev | Int Rev | Inc Rev

  - ➢ For increments, consider holding reviews at beginning, middle, and end of each increment to assess overall design, across all CIs



Sample strategy



SCI Iteration review ◆

SW requirements analysis | Architectural design | SW detailed design | SW coding & testing | SW integration | SW QT

Summary

- ❑ Modern development processes are significantly different than (and much improved over) legacy processes
- ❑ They have evolved from traditional Waterfall approach in which each phase of development was sequentially performed
- ❑ Important to plan for reviews to exploit the strengths of incremental approaches

End

- ❑ **Any questions?....**

Contact info:
Dr. William Bail
The MITRE Corporation
wbail@mitre.org
571.205.7002