The Incremental Commitment Spiral Model (ICSM): Principles and Practices for Successful Systems and Software

Barry Boehm, University of Southern California
boehm@usc.edu; http://csse.usc.edu

ACM Webinar

December 17, 2013

ACM Learning Center

http://learning.acm.org

- 1,350+ trusted technical books and videos by leading publishers including O'Reilly, Morgan Kaufmann, others
- Online courses with assessments and certification-track mentoring, member discounts on tuition at partner institutions
- Learning Webinars on big topics (Cloud/Mobile Development, Cybersecurity, Big Data, Recommender Systems, SaaS, Agile, Machine Learning, Natural Language Processing, Parallel Programming, IPv6, WebGL, Big Data)
- ACM Tech Packs on top current computing topics: Annotated Bibliographies compiled by subject experts
- Popular video tutorials/keynotes from ACM Digital Library, A.M. Turing Centenary talks/panels
- Podcasts with industry leaders/award winners

“Housekeeping”

- Welcome to today’s ACM Webinar. The presentation starts at the top of the hour.
- If you are experiencing any problems/issues, please press the **F5** key on your keyboard if you’re using **Windows**, or **Command + R** if you’re on a **Mac**, to refresh your console, or close and re-launch the presentation. You can also view the Webcast Help Guide, by clicking on the “Help” widget in the bottom dock.
- To control volume, adjust the master volume on your computer.
- If you think of a question during the presentation, please type it into the **Q&A** box and click on the submit button. You do not need to wait until the end of the presentation to begin submitting questions.
- At the end of the presentation, you’ll see a **survey URL** on the final slide. Please take a minute to click on the link and fill it out to help us improve your next webinar experience.
- You can download a PDF of these slides by clicking on the **Resources** widget in the bottom dock.
- This presentation is being recorded and will be available for on-demand viewing in the next 1-2 days. You will receive an **automatic e-mail notification** when the recording is ready.

Talk Back

- Use the Facebook widget in the bottom panel to share this presentation with friends and colleagues

- Use Twitter widget to Tweet your favorite quotes from today's presentation with hashtag #ACMWebinarICSM

- Submit questions and comments via Twitter to @acmeducation – we're reading them!

Goals of Webinar

- Participants to understand
  - Nature of future software and systems engineering and associated development challenges
  - Shortfalls in traditional software/systems engineering and acquisition approaches in addressing challenges
  - Ways to address challenges and enable successful implementation of desired software capabilities using ICSM
  - How to use the ICSM in analyzing software development decision issues in the total system development context
    - Facilitated through case studies

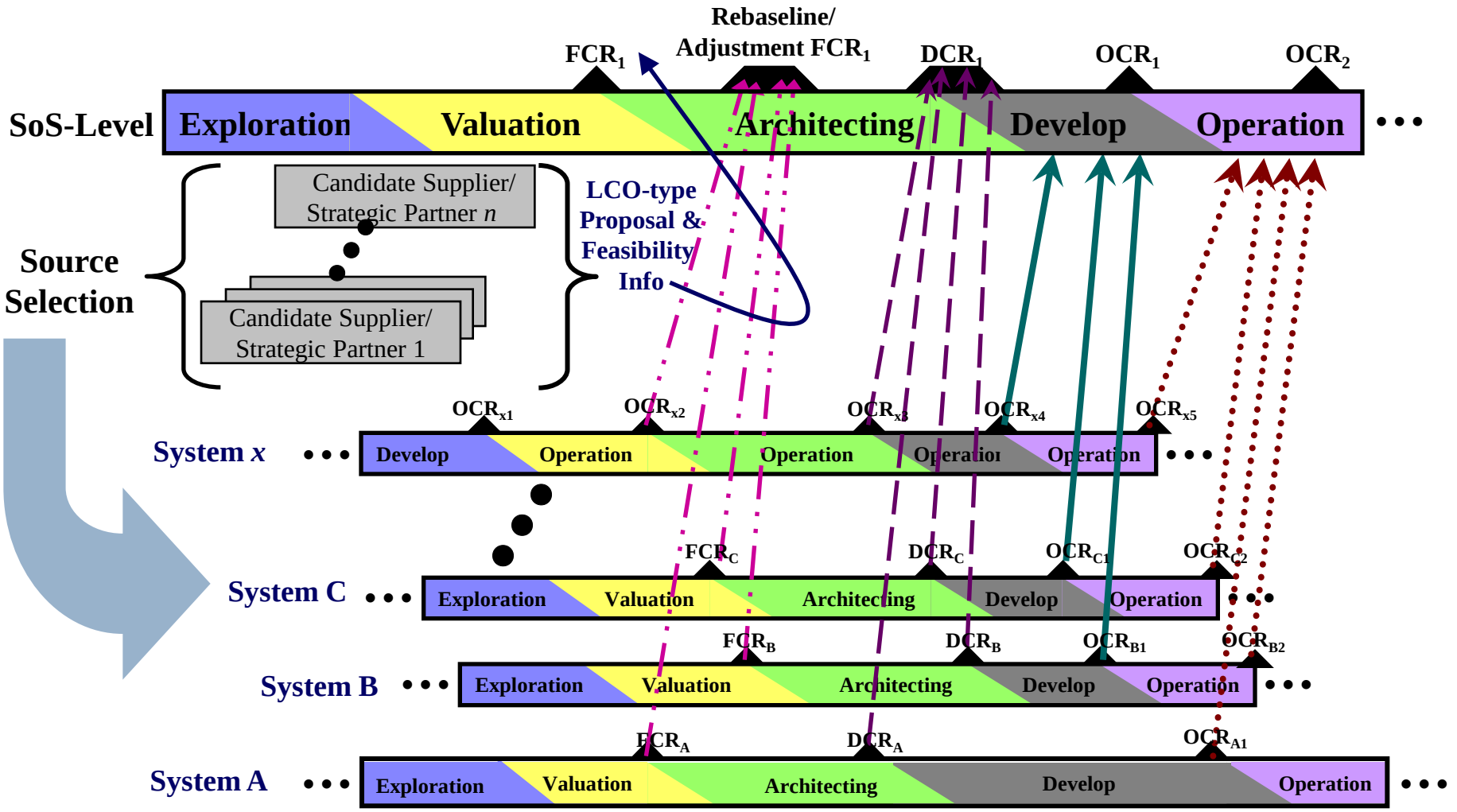
Outline

- *Current and future process challenges*
- Overview of ICSM
- ICSM process decision table
- Guidance and examples for using the ICSM

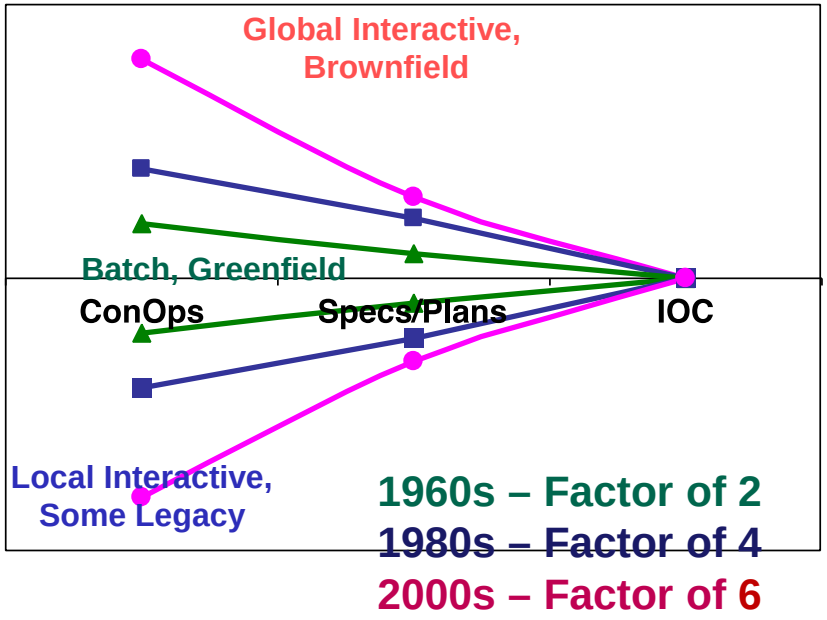
Future Process Challenges-I

- Multi-owner, multi-mission systems of systems (SoS)
  - Integrated supply chain: strategic planning, marketing, merchandising, outsourcing, just-in-time manufacturing, logistics, finance, customer relations management
  - Over 50 separately evolving external systems or services
  - Need to satisfice among multiple stakeholders
  - No one-size-fits-all solutions or processes
- Emergence and human-intensiveness
  - Requirements not pre-specifiable
  - Budgets and schedules not pre-specifiable
  - Need for evolutionary growth
  - Need to manage uncertainty and risk

Example: SoSE Synchronization Points



The Broadening Early Cone of Uncertainty (CU)

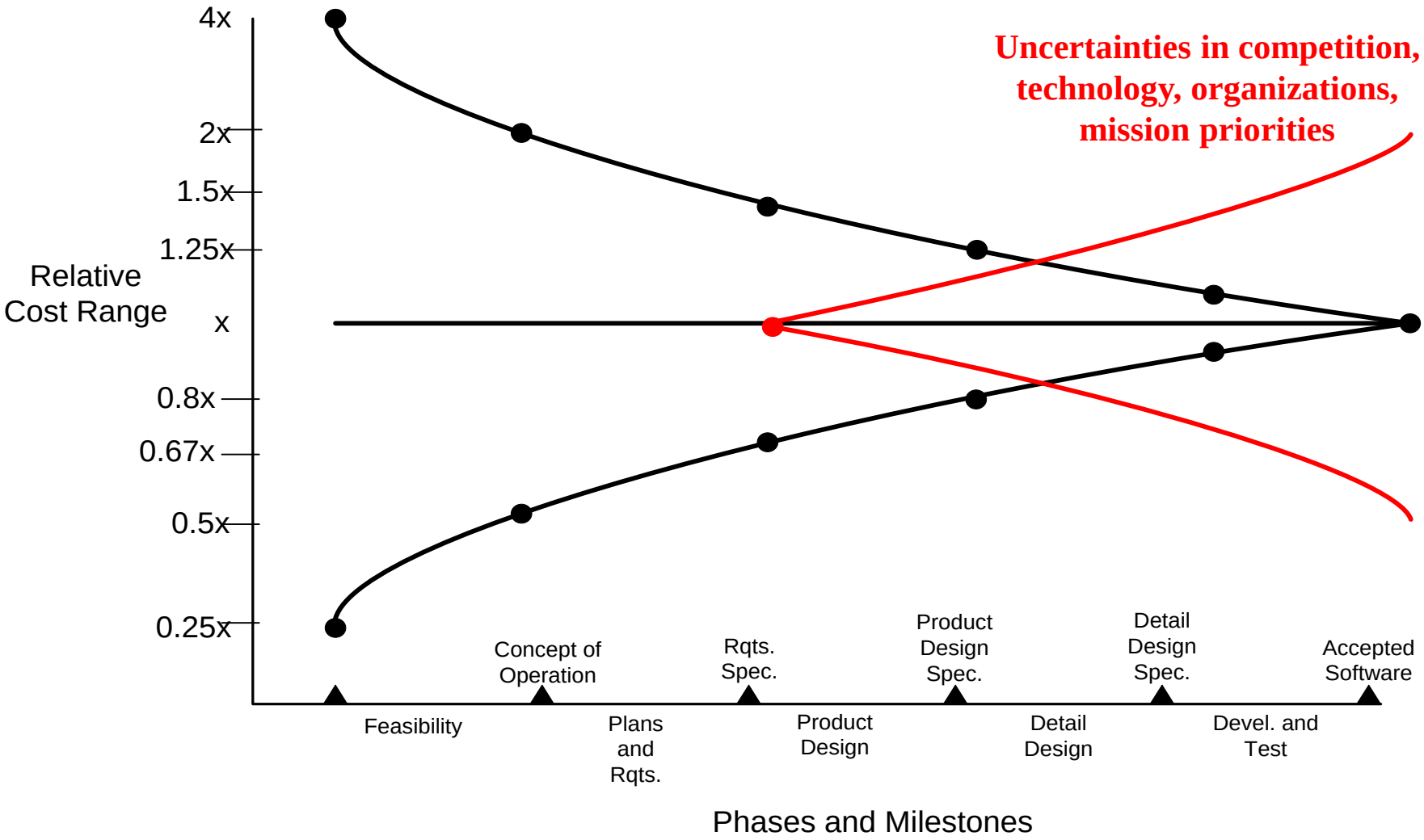


- Need greater investments in narrowing CU
  - Mission, investment, legacy analysis
  - Competitive prototyping
  - Concurrent engineering
  - Associated estimation methods and management metrics
  - More flexible contracts
- Larger systems will often have subsystems with narrower CU's

Future Process Challenges-II

- Rapid pace of change
  - In competition, mission priorities, technology, Commercial Off-the-Shelf (COTS), environment
  - Need incremental development to avoid obsolescence
  - Need concurrent vs. sequential processes
  - Need both prescience and rapid adaptability
    - Software important; humans more important
- Brownfield vs. Greenfield development
  - Need to provide legacy continuity of service
  - Need to accommodate legacy, OTS constraints
- Always-on, never-fail systems
  - Need well-controlled, high-assurance processes
  - Need to synchronize and stabilize concurrency
  - Need to balance assurance and agility

Rapid Change Creates a Late Cone of Uncertainty – Need incremental vs. one-shot development



Need for Evolution-Compatible Acquisition&Development Capabilities

Traditional Metaphor: Purchasing Agent	Needed Metaphor: C2ISR
Complete, consistent, testable requirements before design	Concurrent engineering of requirements and solutions
Single-step development	Evolutionary, incremental system definition and development
One-size-fits-all acquisition instruments	Selectable, tailorable acquisition instruments
Tailorable down from monolithic base	Tailorable up via risk-driven checklist
Premium on low-cost, ambitious first-article performance	Premium on acquisition speed, system flexibility, assurance, total ownership cost

Outline

- Current and future process challenges
- *Overview of ICSM*
- ICSM process decision table
- Guidance and examples for using the ICSM

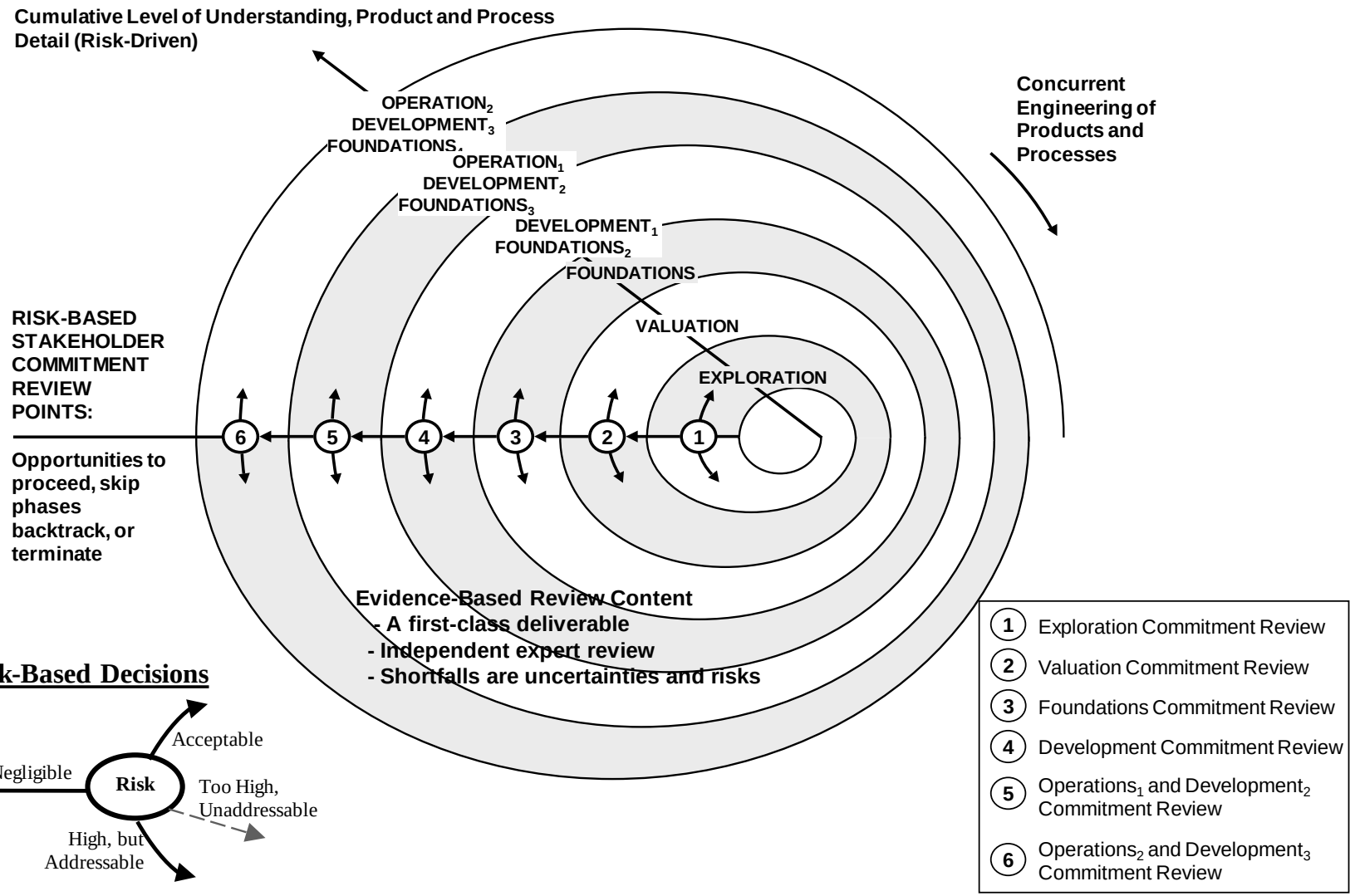
What is the ICSM?

- Risk-driven framework for determining and evolving best-fit system life-cycle process
- Integrates the strengths of phased and risk-driven spiral process models
- Synthesizes together principles critical to successful system development
  - Stakeholder value-based system definition and evolution
  - Incremental commitment and accountability
  - Concurrent system definition and development
  - Evidence and risk-driven decisionmaking

Principles trump diagrams…

Principles used by 60-80% of CrossTalk Top-5 projects, 2002-2005

The Incremental Commitment Spiral Model



ICSM and *Lean* Principles

- **Stakeholder value-based system definition and evolution**
  - ***See the whole***
  - ***Empower the team***
- **Incremental commitment and accountability**
  - ***Amplify learning***
  - ***Decide as late as possible***
- **Concurrent multidiscipline system definition and development**
  - ***Deliver as fast as possible***
  - ***Empower the team***
- **Evidence and risk-driven decisionmaking**
  - ***Build integrity in***
  - ***Eliminate waste***

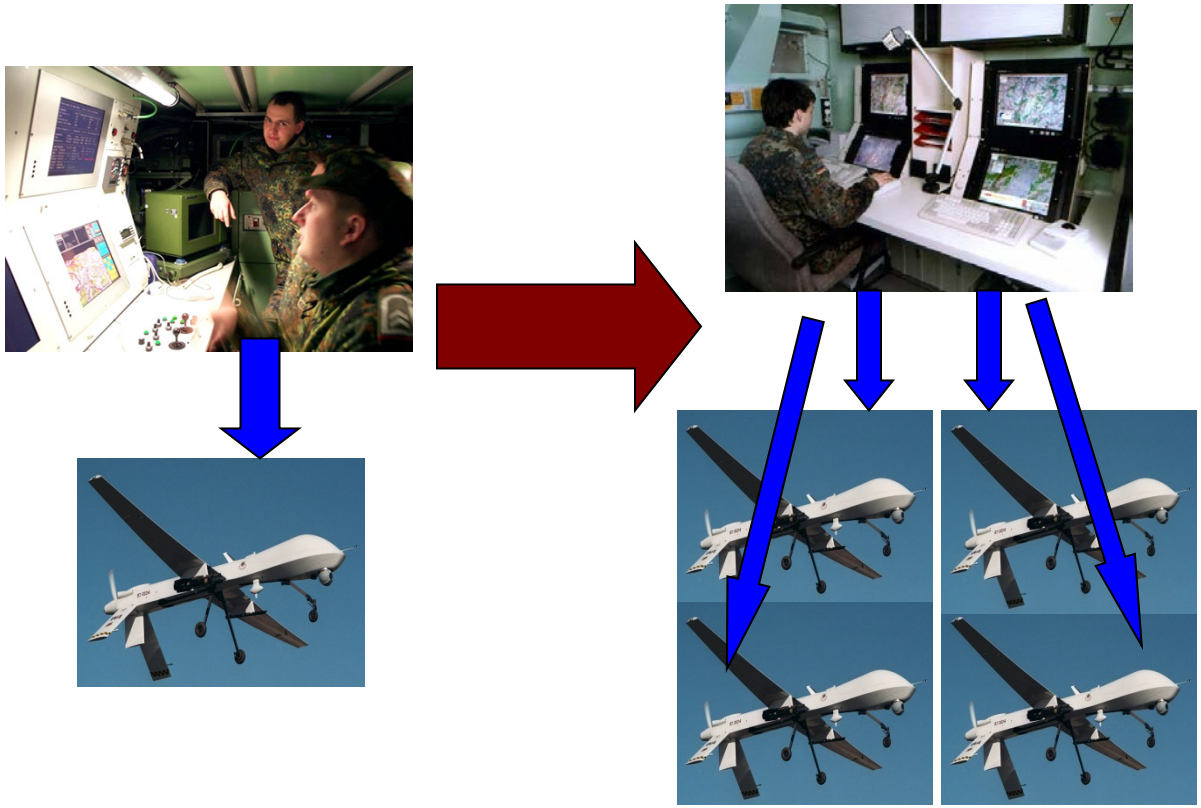
ICSM Nature and Origins

- Integrates hardware, software, and human factors elements of systems engineering
  - Concurrent exploration of needs and opportunities
  - Concurrent engineering of hardware, software, human aspects
  - Concurrency stabilized via anchor point milestones
- Developed in response to a variety of issues
  - Clarify "spiral development" usage
    - Initial phased version (2005)
  - Provide framework for human-systems integration
    - National Research Council report (2007)
- Integrates strengths of current process models
  - But not their weaknesses
- Improves teaching of software project courses
  - Electronic Process Guide (2009)

Incremental Commitment in Gambling

- Total Commitment: Roulette
  - Put your chips on a number
    - E.g., a value of a key performance parameter
  - Wait and see if you win or lose
- Incremental Commitment: Poker, Blackjack
  - Put some chips in
  - See your cards, some of others' cards
  - Decide whether, how much to commit to proceed

Scalable Remotely Controlled Operations



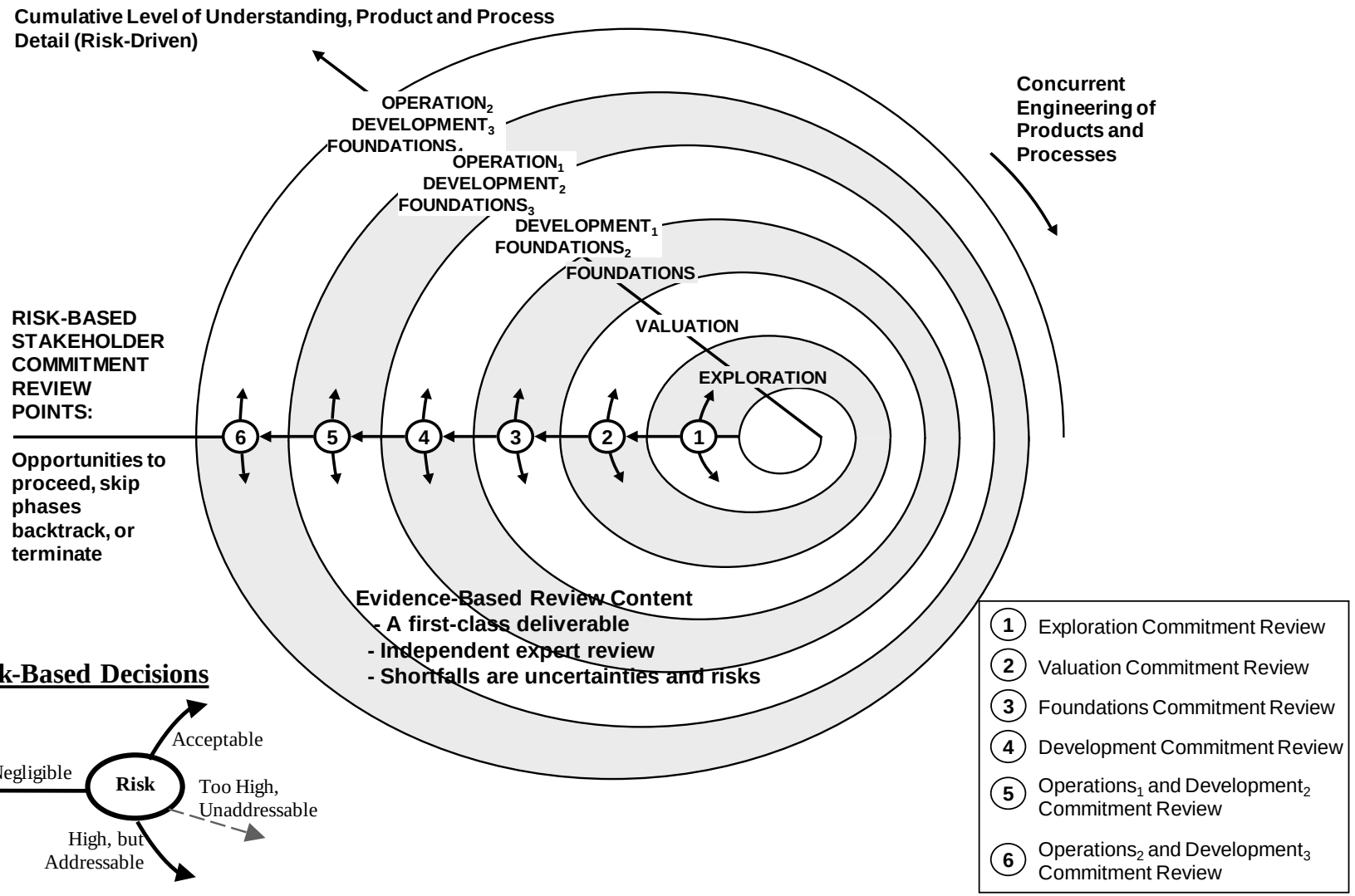
Total vs. Incremental Commitment – 4:1 RPV

- Total Commitment
  - Agent technology demo and PR: Can do 4:1 for $1B
    - Rush to Peak of Inflated Expectations
    - RFP with sunny-day statement of work
  - Winning bidder: $800M; PDR in 120 days; 4:1 capability in 40 months
    - No evidence of achieveability
    - Fixed-price, sunny-day contract requirements
  - PDR: many outstanding risks, undefined interfaces
    - Rainy-day agent, communications performance; multiversion RPVs
  - $800M, 40 months: "halfway" through integration and test
    - Numerous expensive rainy-day Engineering Change Proposals
    - Still no rainy-day test cases
  - 1:1 IOC after $3B, 80 months

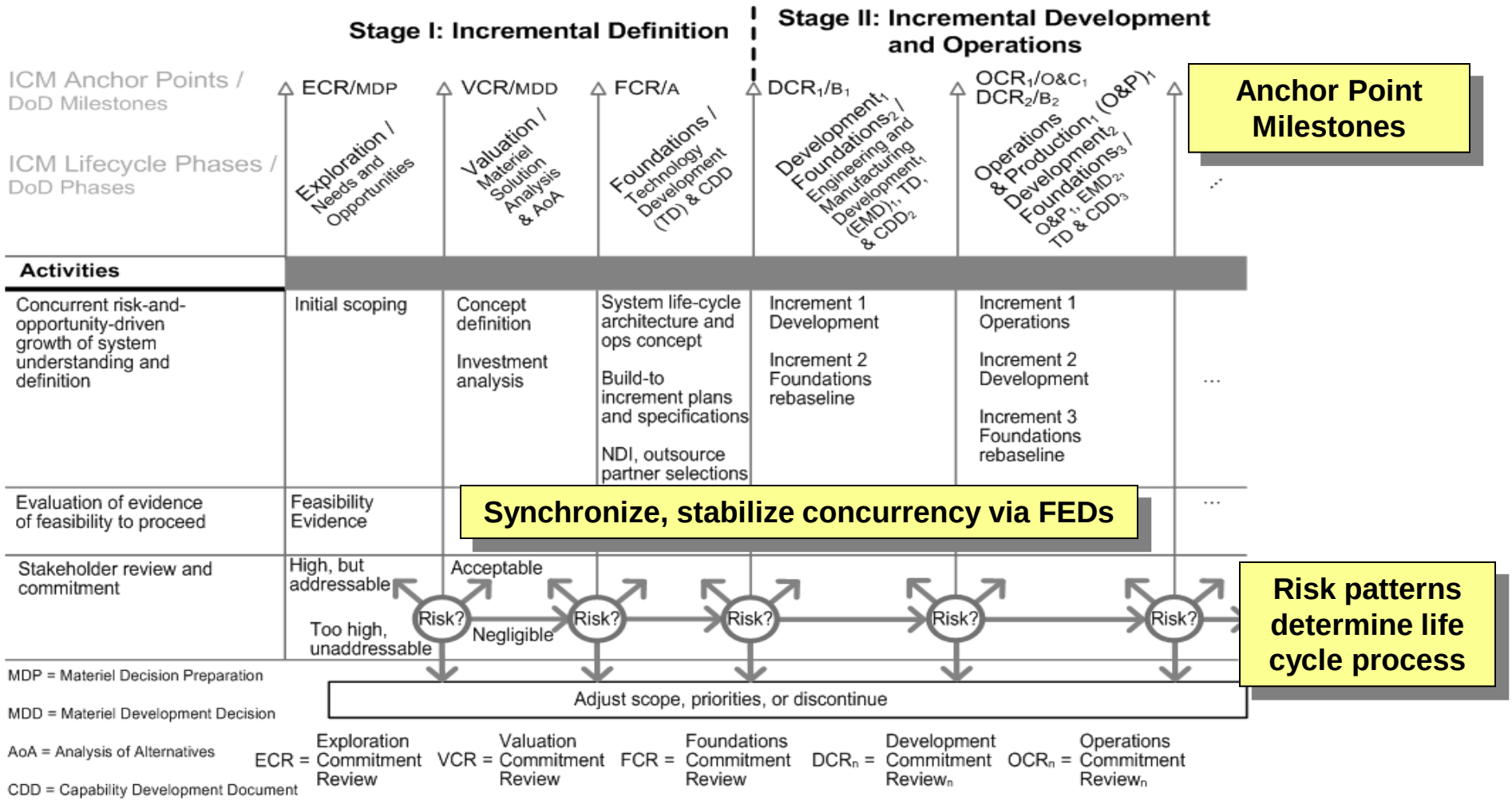
Total vs. Incremental Commitment – 4:1 RPV

- Incremental Commitment [number of competing teams]
  - $25M, 6 mo. to VCR [4]: may beat 1:2 with agent technology, but not 4:1
    - $5M/each for competitors; $5M for evaluation
    - Some rainy-day scenarios
  - $75M, 8 mo. to FCR [3]: agent technology may do 1:1; some risks
    - $20M/each for competitors; scaled-down RPVs; $15M for evaluation
    - More diverse, rainy-day scenarios and operators
  - $225M, 10 mo. to DCR [2]: validated architecture, high-risk elements
    - $80M/each for competitors; full-scale RPVs; $65M for evaluation
    - Participation in full-scale operational exercise
    - Evidence-validated life cycle architecture, IOC plans and budgets
    - Remaining risks covered by risk management plans
  - $675M, 18 mo. to IOC [1]: viable 1:1 capability
  - 1:1 IOC after $1B, 42 months

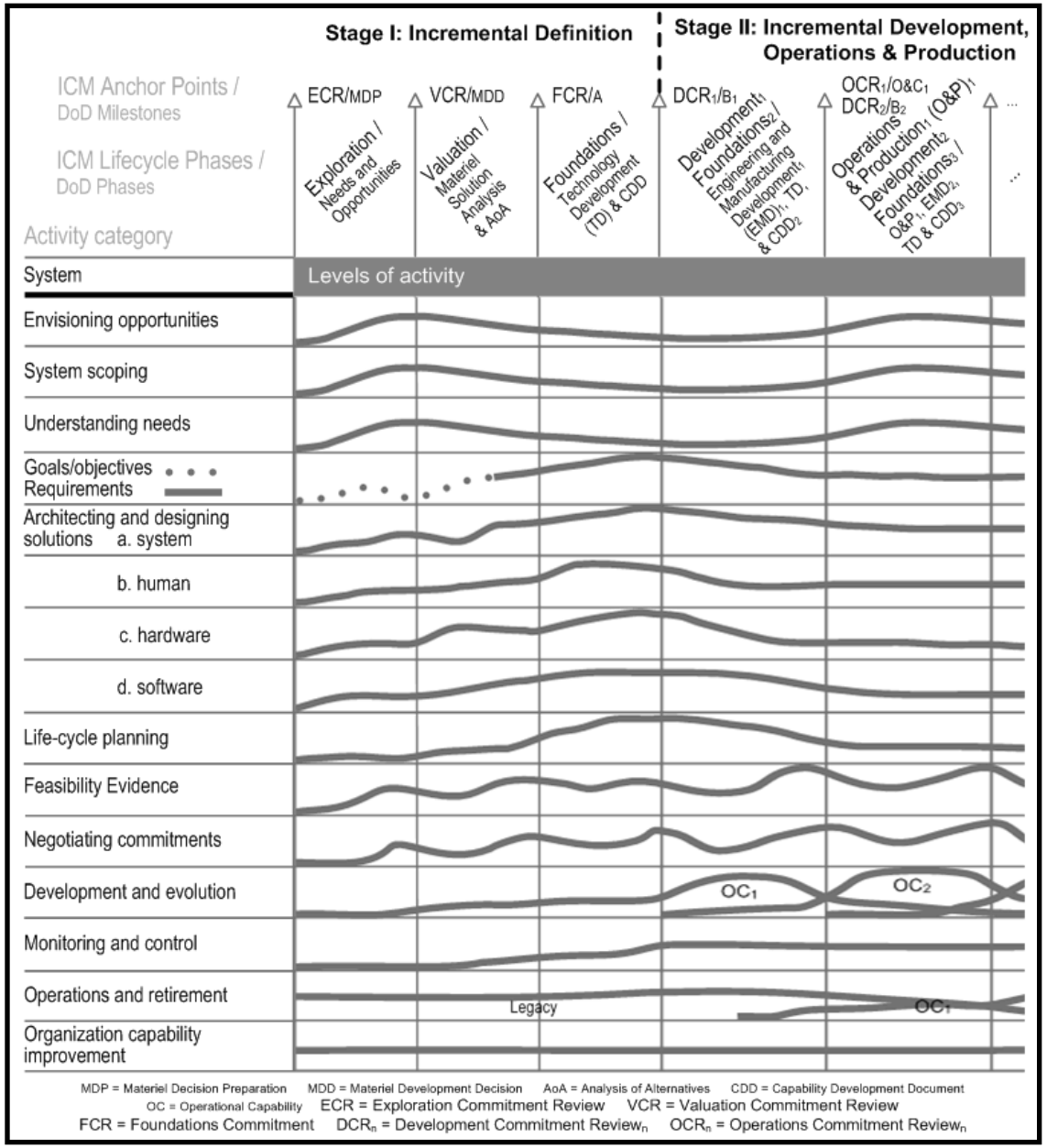
The Incremental Commitment Spiral Model



The Incremental Commitment Spiral Process: Phased View



ICSM Activity Levels for Complex Systems



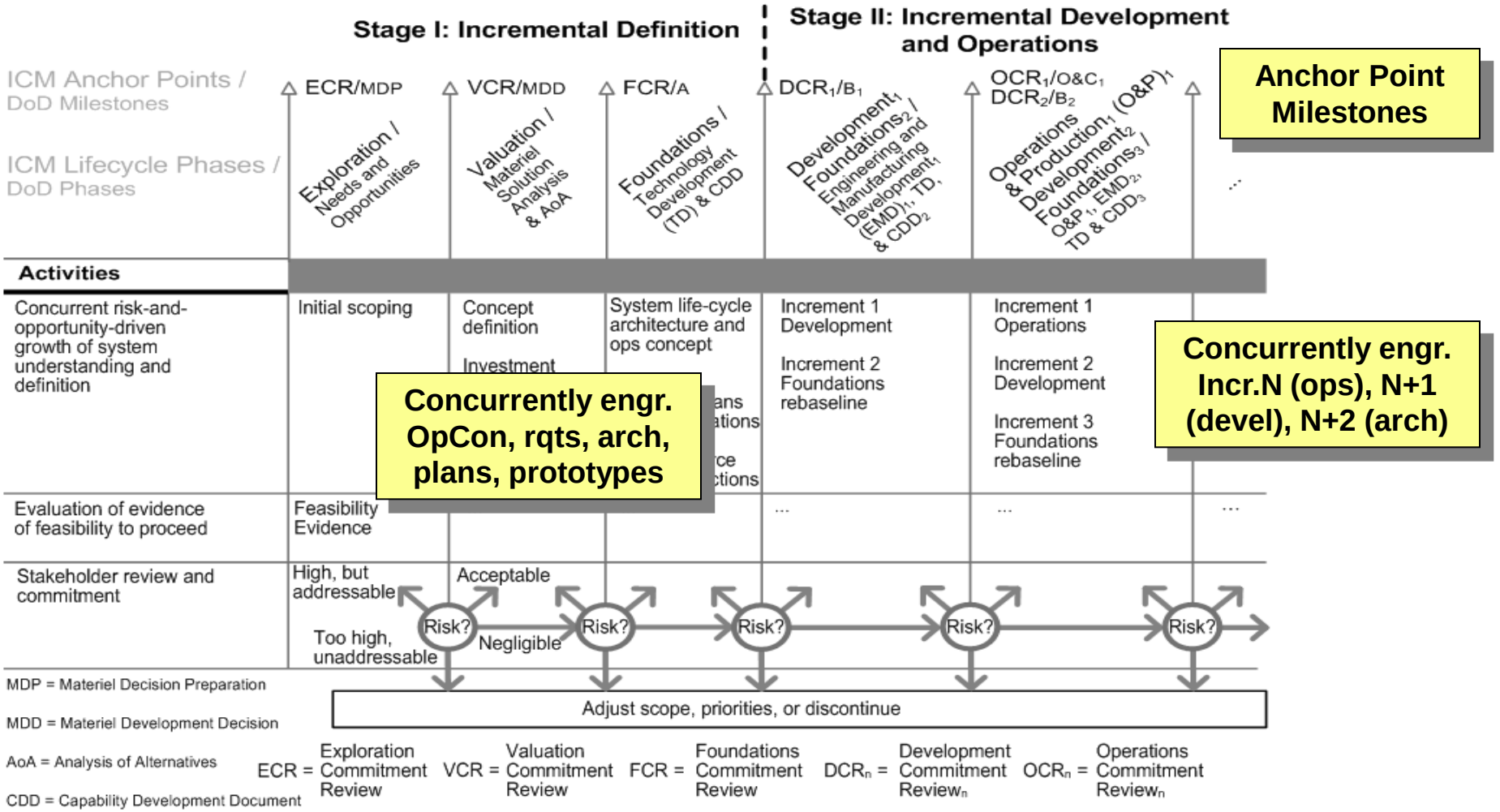
Anchor Point Feasibility Evidence Descriptions

- **Evidence provided by developer and validated by independent experts that:**

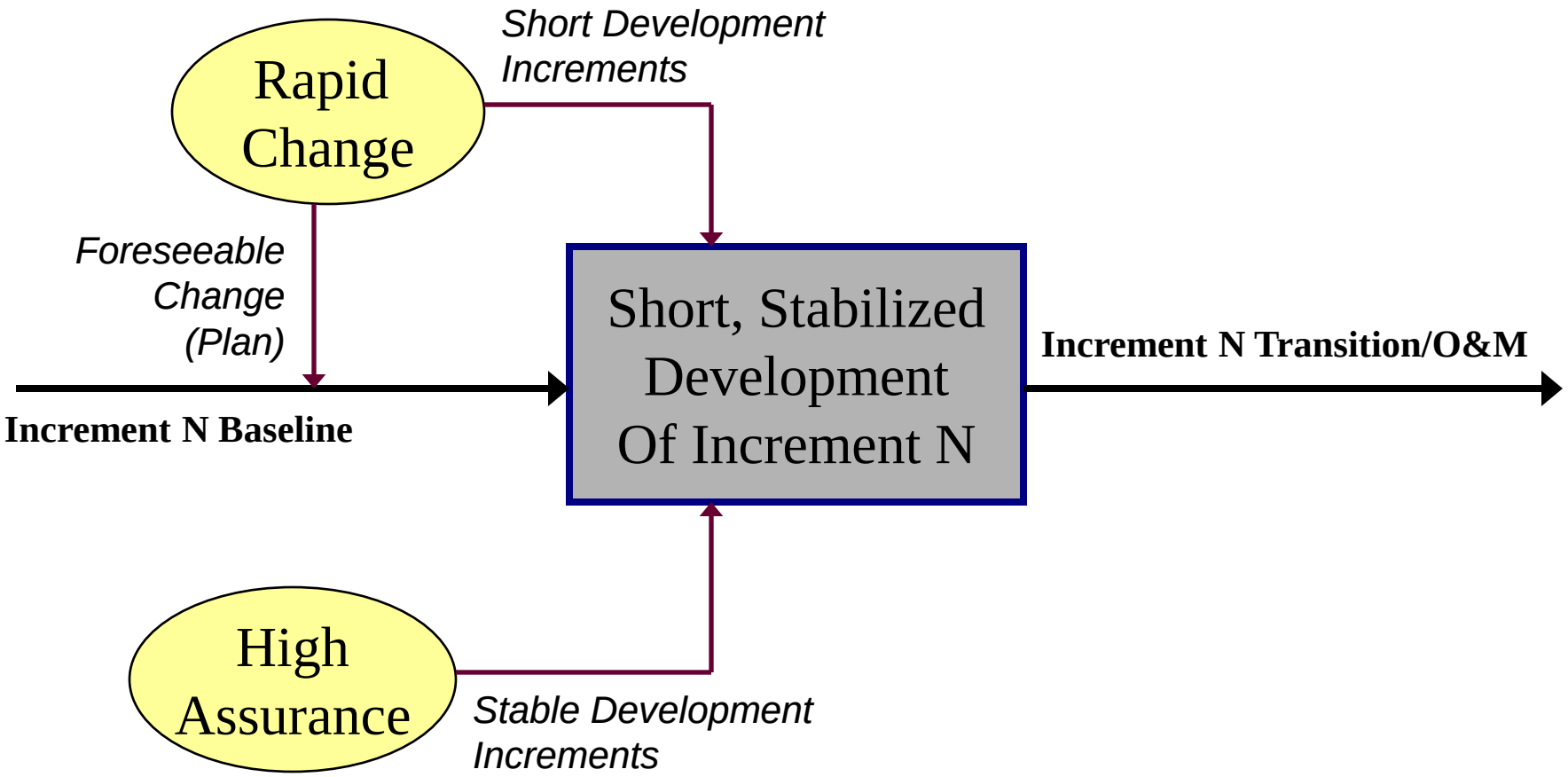
  If the system is built to the specified architecture, it will
  - **Satisfy the requirements: capability, interfaces, level of service, and evolution**
  - **Support the operational concept**
  - **Be buildable within the budgets and schedules in the plan**
  - **Generate a viable return on investment**
  - **Generate satisfactory outcomes for all of the success-critical stakeholders**
- **All major risks resolved or covered by risk management plans**
- **Serves as basis for stakeholders' commitment to proceed**

Can be used to strengthen current schedule- or event-based reviews

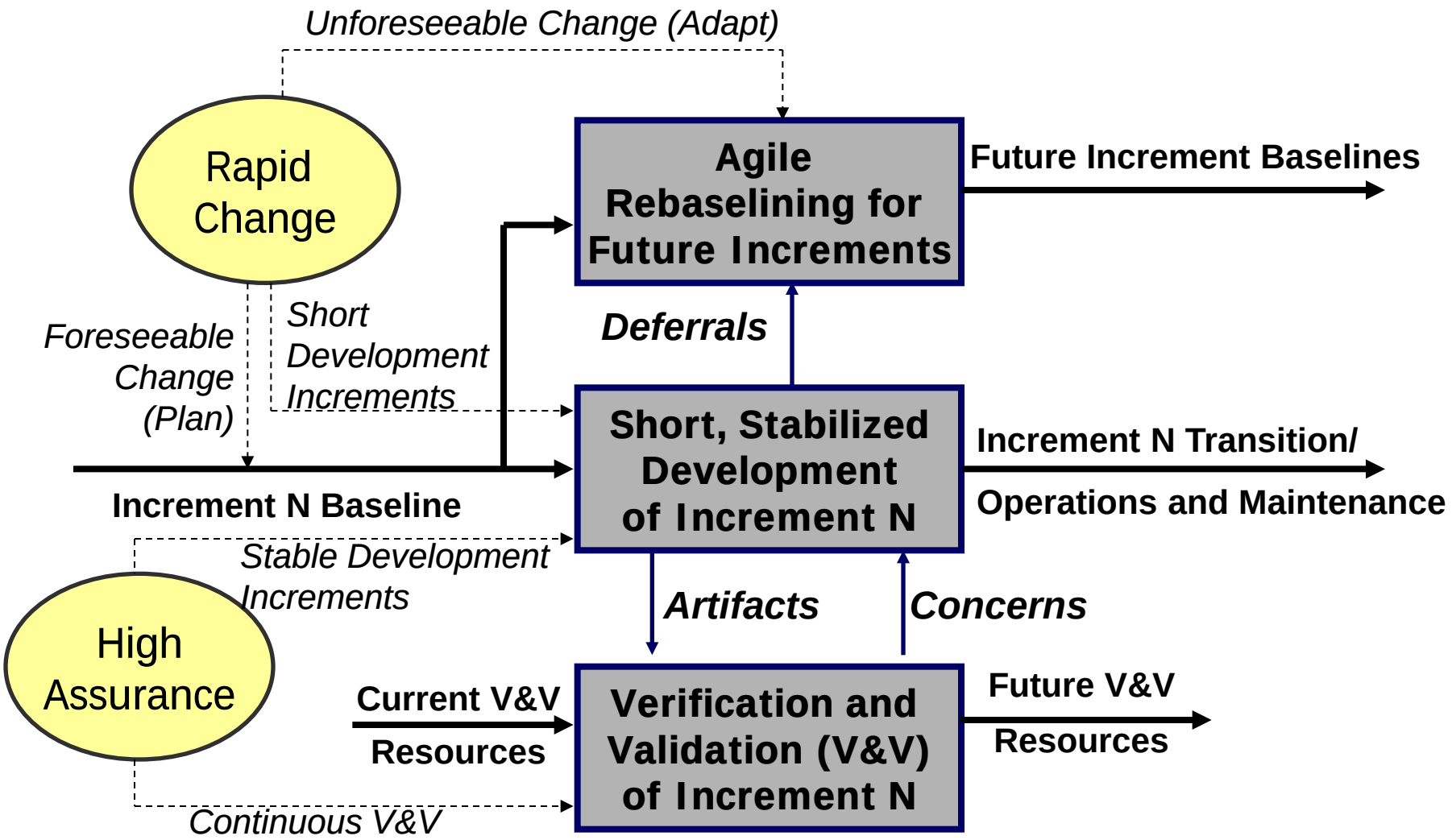
The Incremental Commitment Spiral Process: Phased View



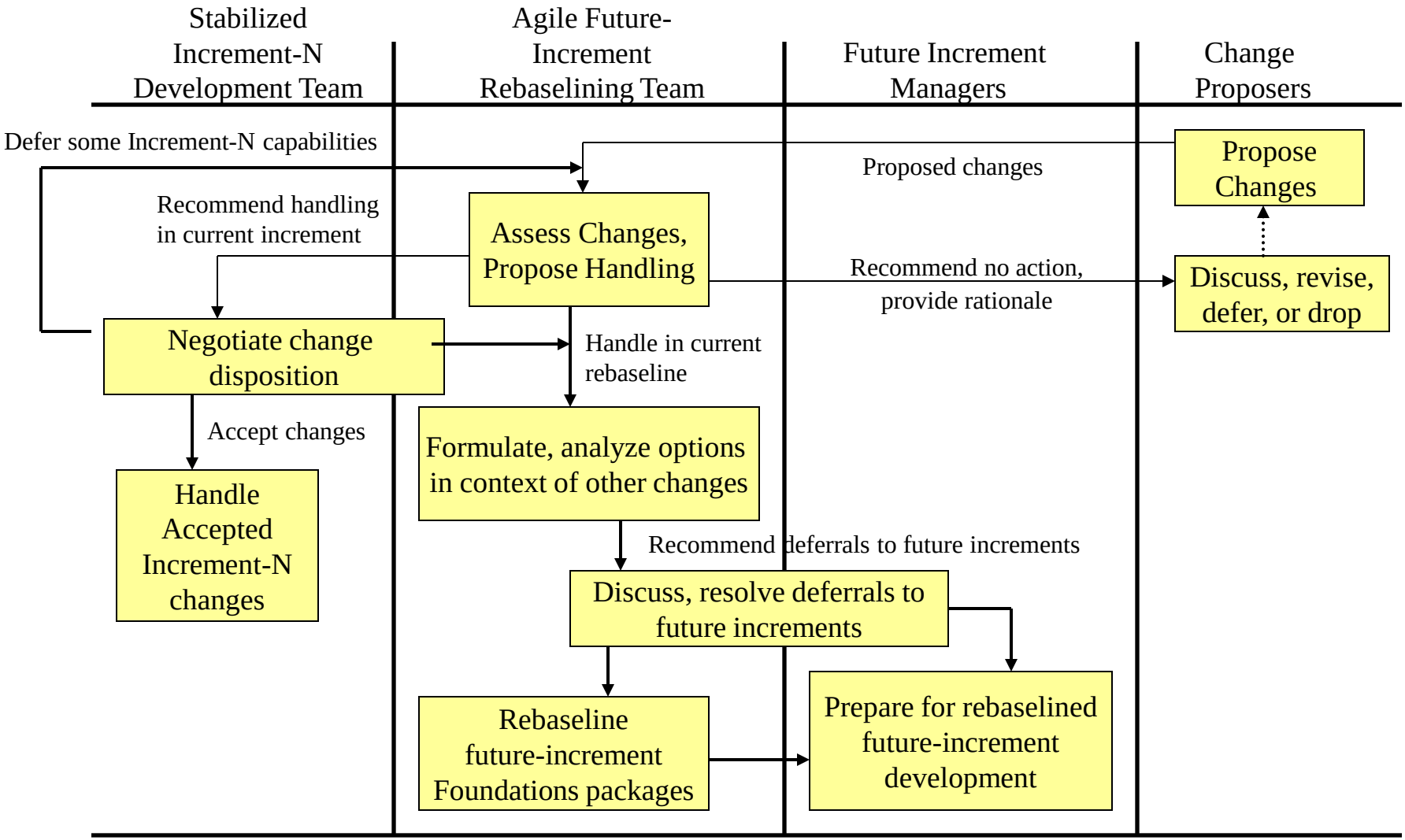
Risk-Driven Scalable Spiral Model: Increment View

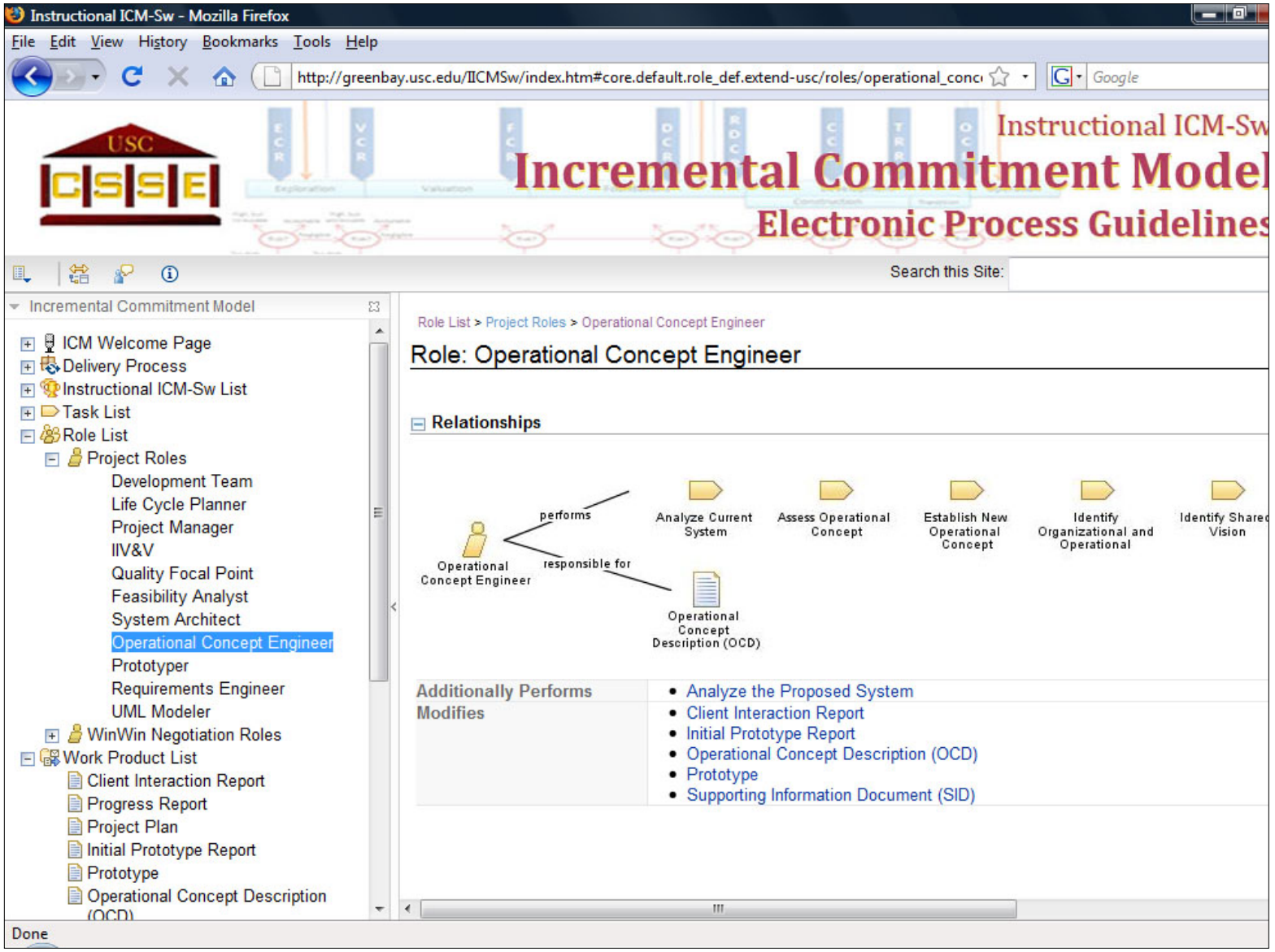


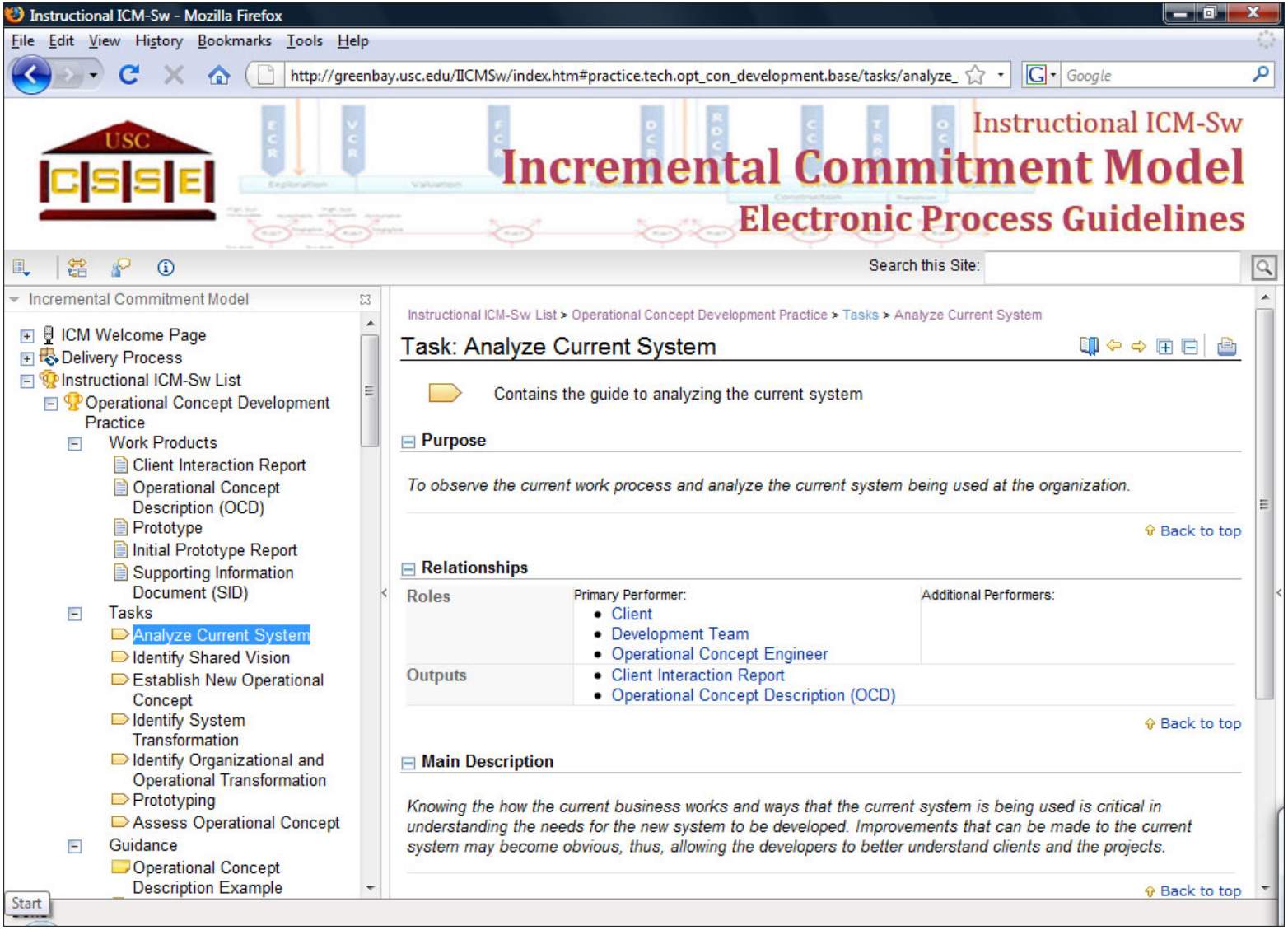
Risk-Driven Scalable Spiral Model: Increment View



Agile Change Processing and Rebaselining







Principles Trump Diagrams

1. Stakeholder value-based system definition, evolution

1. Incremental commitment and accountability

1. Concurrent system definition and development

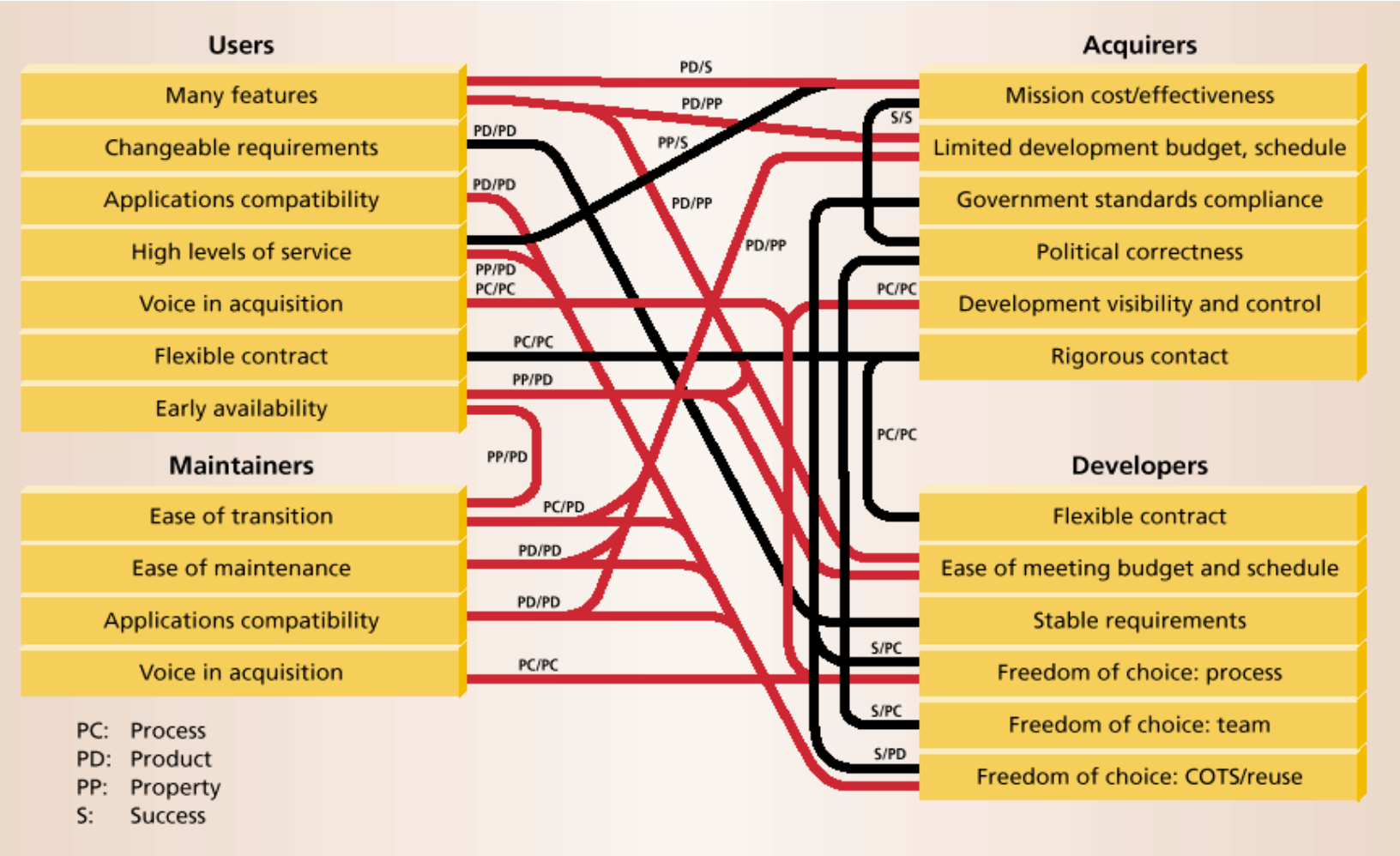
2. Evidence and risk-driven decisionmaking

Counterexample: Bank of America Master Net

Good example: Symbiq Medical Infusion Pump

ICSM Principles Counterexample:

Bank of America Master Net



Principles Trump Diagrams: Master Net

1. **Stakeholder value-based system definition, evolution**
   - Overconcern with Voice of Customer: 3.5 MSLOC of rqts.
   - No concern with maintainers, interoperators: Prime vs. IBM
2. **Incremental commitment and accountability**
   - Total commitment to infeasible budget and schedule
   - No contract award fees or penalties for under/overruns
3. **Concurrent system definition and development**
   - No prioritization of features for incremental development
   - No prototyping of operational scenarios and usage
4. **Evidence and risk-driven decisionmaking**
   - No evaluation of Premier Systems scalability, performance
   - No evidence of ability to satisfy budgets and schedules

Example ICSM Commercial Application: Symbiq Medical Infusion Pump

Winner of 2006 HFES Best New Design Award

Described in NRC HSI Report, Chapter 5

Symbiq IV Pump ICSM Process - I

- Exploration Phase
  - Stakeholder needs interviews, field observations
  - Initial user interface prototypes
  - Competitive analysis, system scoping
  - Commitment to proceed
- Valuation Phase
  - Feature analysis and prioritization
  - Display vendor option prototyping and analysis
  - Top-level life cycle plan, business case analysis
  - Safety and business risk assessment
  - Commitment to proceed while addressing risks

Symbiq IV Pump ICSM Process - II

- **Architecting Phase**
  - Modularity of pumping channels
  - Safety feature and alarms prototyping and iteration
  - Programmable therapy types, touchscreen analysis
  - Failure modes and effects analyses (FMEAs)
  - Prototype usage in teaching hospital
  - Commitment to proceed into development
- **Development Phase**
  - Extensive usability criteria and testing
  - Iterated FMEAs and safety analyses
  - Patient-simulator testing; adaptation to concerns
  - Commitment to production and business plans

Principles Satisfaction: Symbiq IV Pump

1. **Stakeholder value-based system definition, evolution**
   - Extensive involvement of users, buyers, funders, regulators
   - Extensive use of prototyping, safety analysis methods
2. **Incremental commitment and accountability**
   - Expanding system definition and evidence elaboration
   - Decision to start with composable 1- and 2-channel pumps
3. **Concurrent system definition and development**
   - Concurrent evaluation of display, alarm, pump suppiiers
   - Concurrent definition, evaluation of safety and business cases
4. **Evidence and risk-driven decisionmaking**
   - Evidence-based reviews of technical and business feasibility
   - Outstanding risks covered by next-phase risk mitigation plans

ICSM Summary

- **Current processes not well matched to future challenges**
  - **Emergent, rapidly changing requirements**
  - **High assurance of scalable performance and qualities**
- **ICSM addresses challenges**
  - **Assurance via evidence-based milestone commitment reviews, stabilized incremental builds with concurrent V&V**
    - **Evidence shortfalls treated as risks**
  - **Adaptability via concurrent agile team handling change traffic and providing evidence-based rebaselining of next-increment specifications and plans**
  - **Use of critical success factor principles: stakeholder value-based, incremental commitment and accountability, concurrent system definition and development, evidence and risk-driven decisionmaking**
- **Major implications for funding, contracting, career paths**

- ***Backup Charts***

Is the ICSM a One-Size-Fits-All Process?

- **Frequently-Asked Question**
  - **I can see how the ICSM can help on large, highly critical projects, but we have simpler projects too. Wouldn't process models like Agile be better for these?**
- **Answer (to be elaborated in the next session)**
  - **The ICSM is actually a risk-driven process model generator**
  - **For some risk patterns, pure Agile is the best choice**
  - **For other risk patterns, where pure Agile would encounter scalability or system assurance problems, an alternative process called Architected Agile would be better**
  - **Several such common risk patterns will be discussed next. The best choice can generally be determined in the ICSM Exploration phase.**

Exercise: Apply Principles to 4:1 RPV Case Study

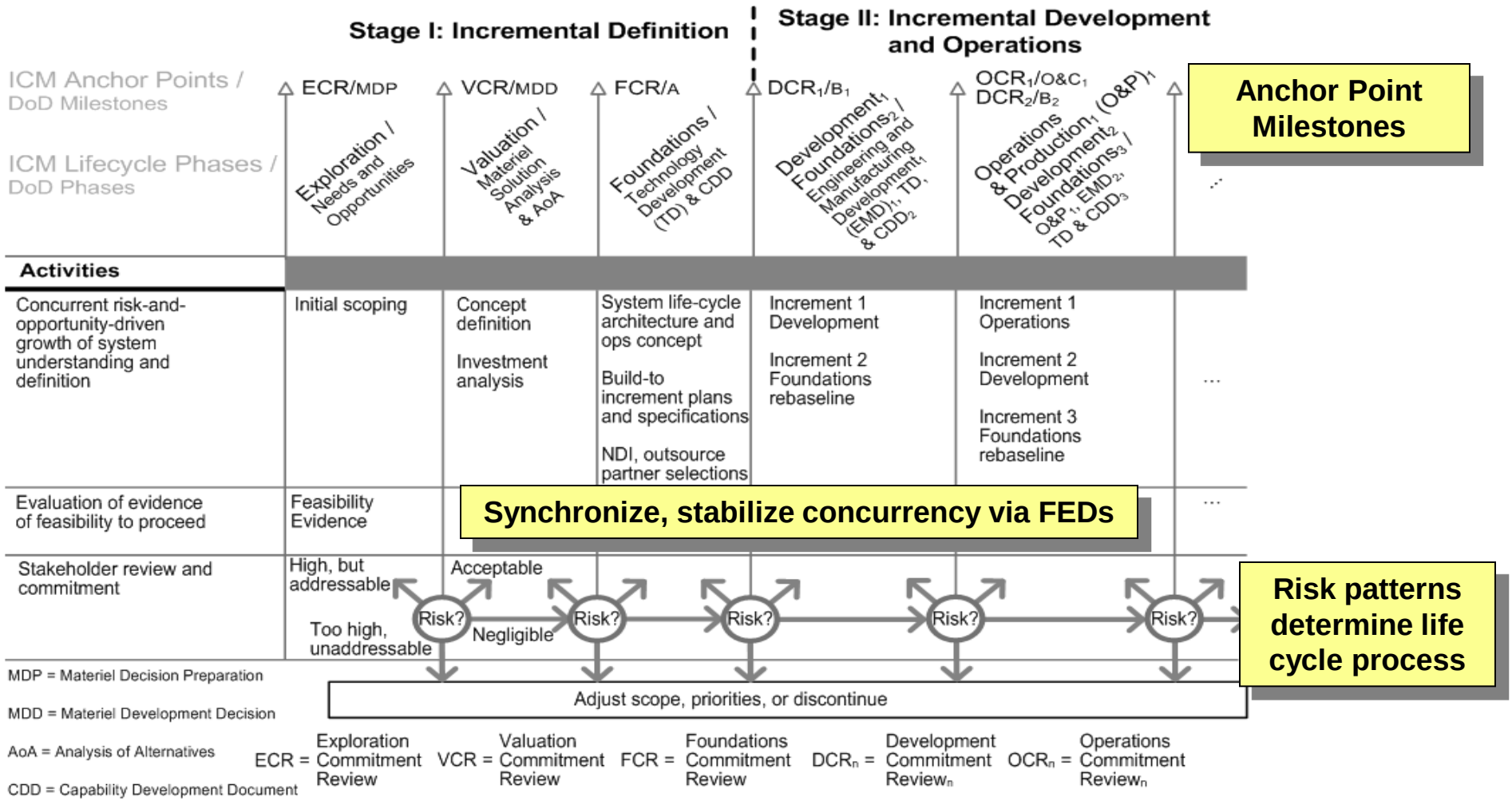
- Agent technology demo and PR: Can do 4:1 for $1B
  - Rush to Peak of Inflated Expectations
  - RFP with sunny-day statement of work
- Winning bidder: $800M; PDR in 120 days; 4:1 capability in 40 months
  - No evidence of achieveability
  - Fixed-price, sunny-day contract requirements
- PDR: many outstanding risks, undefined interfaces
  - Rainy-day agent, communications performance; multiversion RPVs
- $800M, 40 months: "halfway" through integration and test
  - Numerous expensive rainy-day Engineering Change Proposals
  - Still no rainy-day test cases
- 1:1 IOC after $3B, 80 months

1. Stakeholder value-based system definition and evolution
2. Incremental commitment and accountability
3. Concurrent system definition and development
4. Evidence and risk-driven decisionmaking

Outline

- Current and future process challenges
- Overview of ICSM
- *ICSM process decision table*
- Guidance and examples for using the ICSM

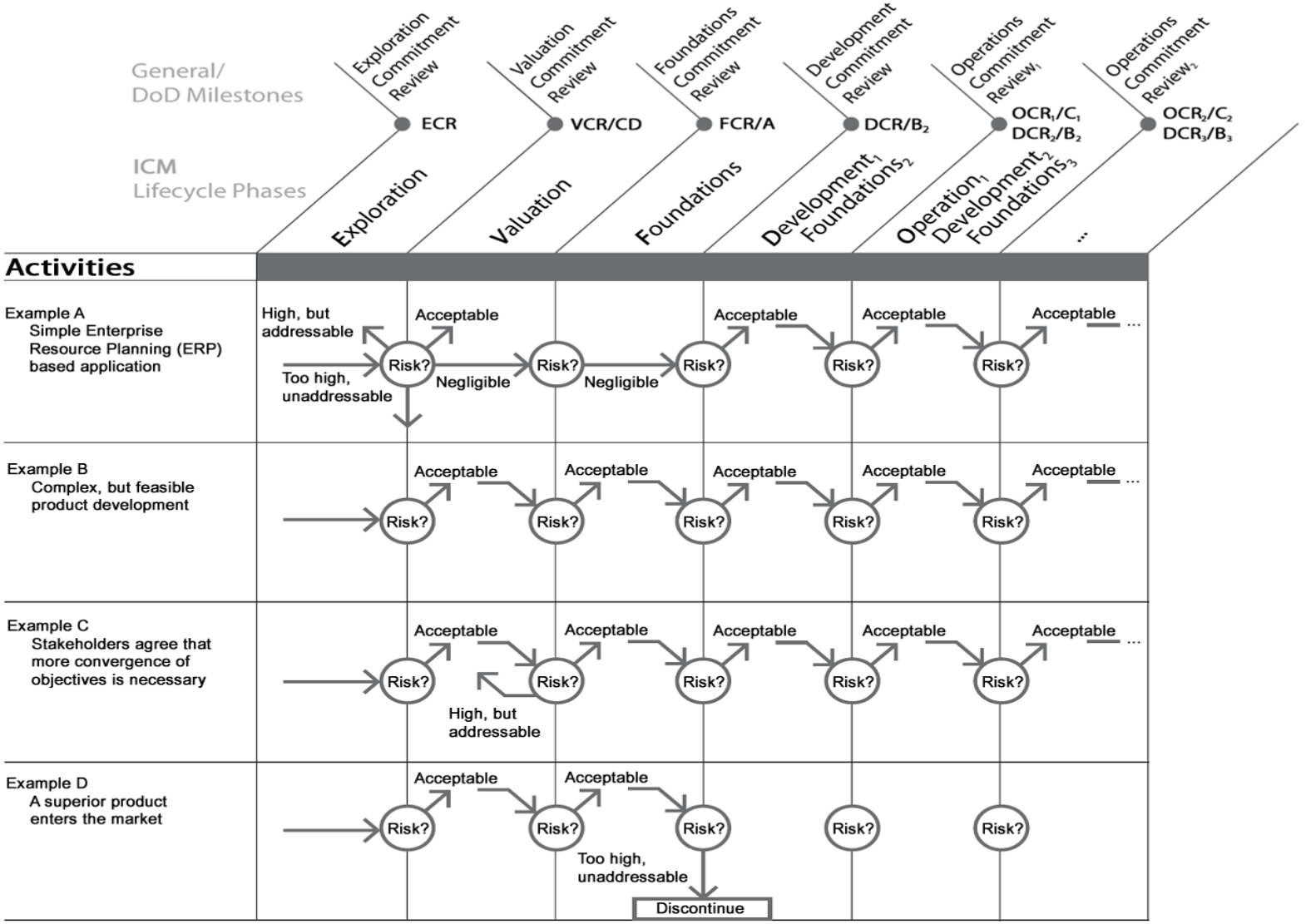
The Incremental Commitment Life Cycle Process: Overview



The ICSM as Risk-Driven Process Generator

- Stage I of the ICSM has 3 decision nodes with 4 options/node
  - Culminating with incremental development in Stage II
  - Some options involve go-backs
  - Results in many possible process paths
- Can use ICSM risk patterns to generate frequently-used processes
  - With confidence that they fit the situation
- Can generally determine this in the Exploration phase
  - Develop as proposed plan with risk-based evidence at VCR milestone
  - Adjustable in later phases

Different Risk Patterns Yield Different Processes



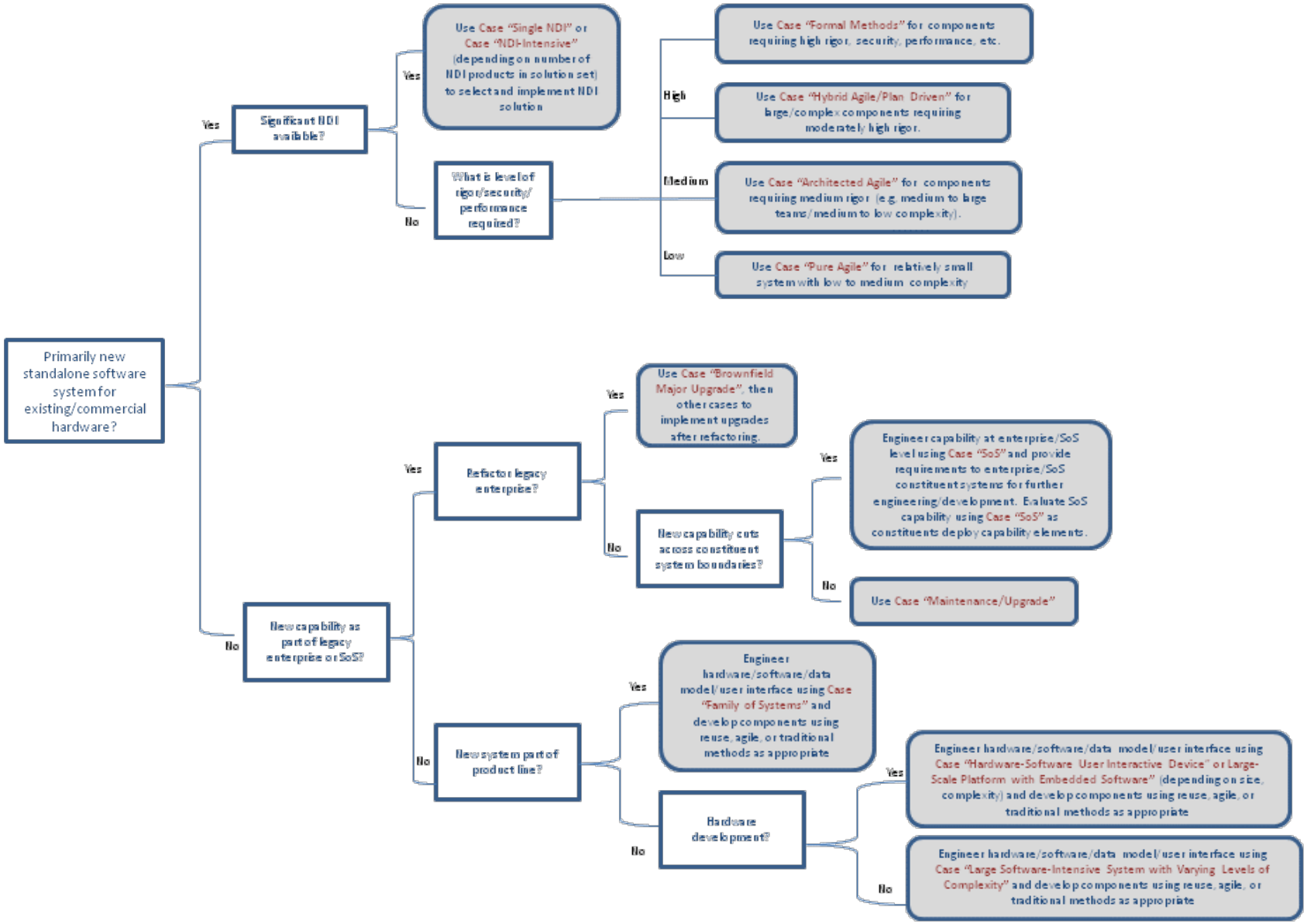
The ICSM Process Decision Table: Key Decision Inputs

- Product and project size and complexity
- Requirements volatility
- Mission criticality
- Nature of Non-Developmental/COTS/Services support
  - Commercial, open-source, reused components
  - Cloud services
- Organizational and Personnel Capability

The ICSM Process Decision Table: Key Decision Outputs

- Key Stage I activities: incremental definition
- Key Stage II activities: incremental development and operations
- Suggested calendar time per build, per deliverable increment

ICSM Common Case Decision Table



Common Risk-Driven Special Cases of the ICSM (Cases 1-4)

Case 1: Use NDI

Example: Small accounting system
Size, Complexity: Size variable, complexity low
Typical Change Rate/Month: Negligible
Criticality: n/a
NDI Support: Complete
Organizational Personnel Capability: NDI-experienced (medium)
Key Stage I Activities (Incremental Definition): Acquire NDI
Key Stage II Activities (Incremental Development/Operations): Use NDI
Time/Build: n/a
Time/Increment: Vendor-driven

Case 2: Agile

Example: E-services
Size, Complexity: Low
Typical Change Rate/Month: 1-30%
Criticality: Low to medium
NDI Support: Good, in place
Organizational Personnel Capability: Agile-ready, medium-high experience
Key Stage I Activities (Incremental Definition): Skip Valuation and Architecting phases
Key Stage II Activities (Incremental Development/Operations): Scrum plus agile methods of choice
Time/Build: <= 1 day
Time/Increment: 2-6 weeks

Case 3: Architected Agile

Example: Business data processing
Size, Complexity: Medium
Typical Change Rate/Month: 1-10 %
Criticality: Medium to high
NDI Support: Good, most in place
Organizational Personnel Capability: Agile-ready, medium to high experience
Key Stage I Activities (Incremental Definition): Combine Valuation, Architecting phases. Complete NDI preparation.
Key Stage II Activities (Incremental Development/Operations): Architecture-based Scrum of Scrums
Time/Build: 2-4 weeks
Time/Increment: 2-6 months

Case 4: Formal Methods

Example: Security kernel; Safety-critical LSI chip
Size, Complexity: Low
Typical Change Rate/Month: 0.3%
Criticality: Extra high
NDI Support: None
Organizational Personnel Capability: Strong formal methods experience
Key Stage I Activities (Incremental Definition): Precise formal specification
Key Stage II Activities (Incremental Development/Operations): Formally-based programming language; formal verification
Time/Build: 1-5 days
Time/Increment: 1-4 weeks

Common Risk-Driven Special Cases of the ICSM (Cases 5-8)

Case 5: Hardware with Embedded Software Component

Example: Multi-sensor control device
Size, Complexity: Low
Typical Change Rate/Month: 0.3 - 1 %
Criticality: Medium to very high
NDI Support: Good, in place
Organizational Personnel Capability: Experienced, medium-high
Key Stage I Activities (Incremental Definition): Concurrent hardware/software engineering. CDR-level ICSM DCR
Key Stage II Activities (Incremental Development/Operations): IOC development, LRIP, FRP. Concurrent version N+1 engineering
Time/Build: Software 1-5 days
Time/Increment: Market-driven

Case 6: Indivisible IOC

Example: Complete vehicle platform
Size, Complexity: Medium to high
Typical Change Rate/Month: 0.3 – 1%
Criticality: High to very high
NDI Support: Some in place
Organizational Personnel Capability: Experienced, medium to high
Key Stage I Activities (Incremental Definition): Determine minimum-IOC likely, conservative cost. Add deferrable software features as risk reserve
Key Stage II Activities (Incremental Development/Operations): Drop deferrable features to meet conservative cost. Strong award free for features not dropped.
Time/Build: Software: 2-6 weeks
Time/Increment: Platform: 6-18 months

Case 7: NDI-Intensive

Example: Supply chain management
Size, Complexity: Medium to high
Typical Change Rate/Month: 0.3 – 3%
Criticality: Medium to very high
NDI Support: NDI-driven architecture
Organizational Personnel Capability: NDI-experienced, medium to high
Key Stage I Activities (Incremental Definition): Thorough NDI-suite life cycle cost-benefit analysis, selection, concurrent requirements/architecture definition
Key Stage II Activities (Incremental Development/Operations): Pro-active NDI evolution influencing, NDI upgrade synchronization
Time/Build: Software: 1-4 weeks
Time/Increment: Systems: 6-18 months

Case 8: Hybrid Agile/Plan-Driven System

Example: C4ISR system
Size, Complexity: Medium to very high
Typical Change Rate/Month: Mixed parts; 1-10%
Criticality: Mixed parts; Medium to very high
NDI Support: Mixed parts
Organizational Personnel Capability: Mixed parts
Key Stage I Activities (Incremental Definition): Full ICSM, encapsulated agile in high change, low-medium criticality parts (Often HMI, external interfaces)
Key Stage II Activities (Incremental Development/Operations): Full ICSM, three-team incremental development, concurrent V&V, next-increment rebaselining
Time/Build: 1-2 months
Time/Increment: 9-18 months

Common Risk-Driven Special Cases of the ICSM (Cases 9-11)

Case 9: Multi-Owner Directed System of Systems

Example: Net-centric military operations
Size, Complexity: Very high
Typical Change Rate/Month: Mixed parts; 1-10 %
Criticality: Very high
NDI Support: Many NDIs, some in place
Organizational Personnel Capability: Related experience, medium to high
Key Stage I Activities (Incremental Definition): Full ICSM; extensive multi-owner team building, negotiation
Key Stage II Activities (Incremental Development/Operations): Full ICSM; large ongoing system/software engineering effort
Time/Build: 2-4 months
Time/Increment: 18-24 months

Case 10: Family of Systems

Example: Medical device product line
Size, Complexity: Medium to very high
Typical Change Rate/Month: 1-3%
Criticality: Medium to very high
NDI Support: Some in place
Organizational Personnel Capability: Related experience, medium to high
Key Stage I Activities (Incremental Definition): Skip Valuation and Architecting phases
Key Stage II Activities (Incremental Development/Operations): Scrum plus agile methods of choice
Time/Build: 1-2 months
Time/Increment: 9-18 months

Case 11: Brownfield

Example: Incremental legacy phaseout
Size, Complexity: High to very high
Typical Change Rate/Month: 0.3-3%
Criticality: Medium-high
NDI Support: NDI as legacy replacement
Organizational Personnel Capability: Legacy re-engineering
Key Stage I Activities (Incremental Definition): Re-engineer/refactor legacy into services
Key Stage II Activities (Incremental Development/Operations): Incremental legacy phaseout
Time/Build: 2-6 weeks/refactor
Time/Increment: 2-6 months

Common Risk-Driven Special Cases of the ICSM (Cases 12a/b)

Case 12a: Net-Centric Services – Community Support	Case 12b: Net-Centric Services or Rapid Fielding – Quick Response Mission Support
Example: Community services or special interest group	**Example:** Response to competitor initiative
Size, Complexity: Low to medium	**Size, Complexity:** Medium to high
Typical Change Rate/Month: 0.3-3%	**Typical Change Rate/Month:** 3-30%
Criticality: Low to medium	**Criticality:** Medium to high
NDI Support: Tailorable service elements	**NDI Support:** Tailorable service or product elements
Organizational Personnel Capability: NDI-experienced	**Organizational Personnel Capability:** NDI-experienced
Key Stage I Activities (Incremental Definition): Filter, select, compose, tailor NDI	**Key Stage I Activities (Incremental Definition):** Filter, select, compose, tailor NDI
Key Stage II Activities (Incremental Development/Operations): Evolve tailoring to meet community needs	**Key Stage II Activities (Incremental Development/Operations):** Satisfy quick response; evolve or phase out
Time/Build: <= 1 day	**Time/Build:** <= 1 day
Time/Increment: 2-12 months	**Time/Increment:** Quick response-driven

LEGEND

C4ISR: Command, Control, Computing, Communications, Intelligence, Surveillance, Reconnaissance.
CDR: Critical Design Review.
DCR: Development Commitment Review.
FRP: Full-Rate Production.
HMI: Human-Machine Interface.
HW: Hardware.
IOC: Initial Operational Capability.
LSI: Large Scale Integration.
LRIP: Low-Rate Initial Production.
NDI: Non-Development Item.
SW: Software

Relations to Recent Draft DoDI 5000.02

- 1. Hardware-Intensive Program
  - ICSM 5: Simple Hardware-Intensive System: Sensor Control
  - ICSM 6: Indivisible IOC: Vehicle Platform
- 2. Defense-Unique Software-Intensive Program
  - ICSM 8: Hybrid Agile/Plan-Driven System: C4ISR
- 3. Incrementally-Fielded Software-Intensive Program
  - ICSM 3: Architected Agile: Business Data Processing
  - ICSM 7: NDI-Intensive: Supply Chain Management
  - ICSM 8: Hybrid Agile/Plan-Driven System: C4ISR
- 4. Accelerated Acquisition Program
  - ICSM 12b: Quick-Response Mission Support
- 5a,b. Hybrid Hardware- or Software-Dominant Platform
  - Combinations of ICSM 6 and 8

Outline

- Current and future process challenges
- Overview of ICSM
- ICSM process decision table
- *Guidance and examples for using the ICSM*
  - Common cases: Architected Agile, Brownfield
  - A Feasibility Evidence Data Item Description

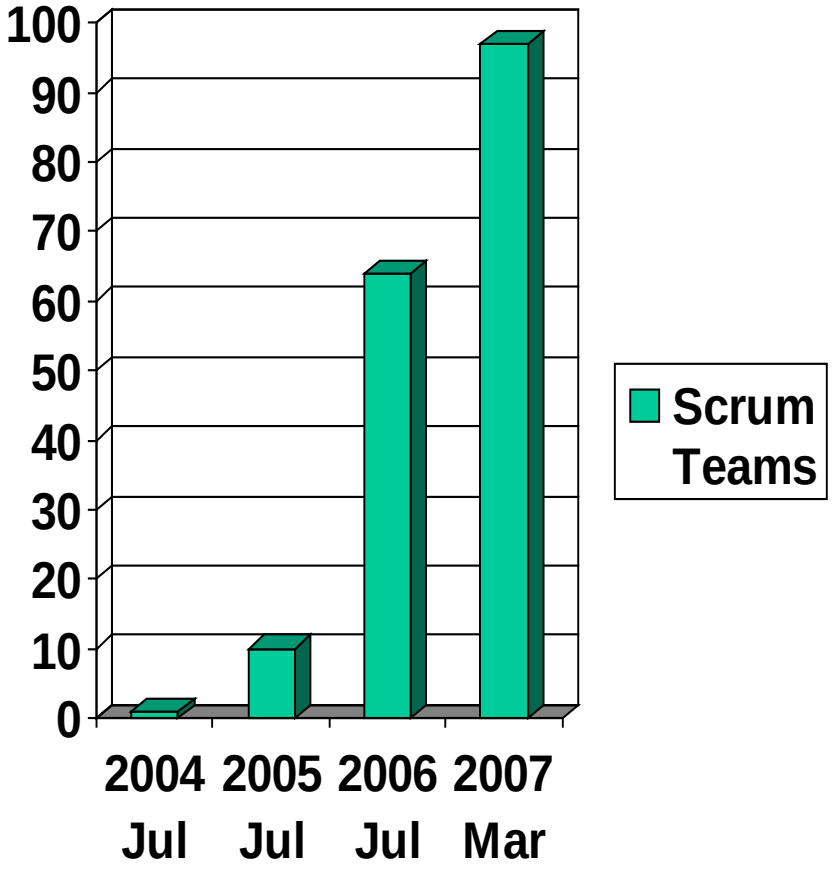
Case 3: Architected Agile

- Exploration phase determines
  - Need to accommodate fairly rapid change, emergent requirements, early user capability
  - Low risk of scalability up to 100 people
  - NDI support of growth envelope
  - Nucleus of highly agile-capable personnel
  - Moderate to high loss due to increment defects
- Example: Business data processing
- Size/complexity: Medium
- Anticipated change rate (% per month): 1-10%
- Criticality: Medium to high
- NDI support: Good, most in place
- Organizational and personnel capability: Agile-ready, med-high capability
- Key Stage I activities: Combined Valuation and Architecting phase, complete NDI preparation
- Key Stage II activities: Architecture-based scrum of scrums
- Time/build: 2-4 weeks Time/increment: 2-6 months

USA Medical Case Study

- 1400 software people; 7M SLOC; 7 sites
  - 4 in Europe, 2 in India
- 500 medical applications; 500 financial; others
- Survivability-critical software problems
  - Reliability, productivity, performance, interoperability
  - Sarbanes-Oxley requirements
  - Management receptive to radical change
- Some limited experimental use of agile methods
  - Led by top software technologist/manager
- Committed to total change around Scrum and XP

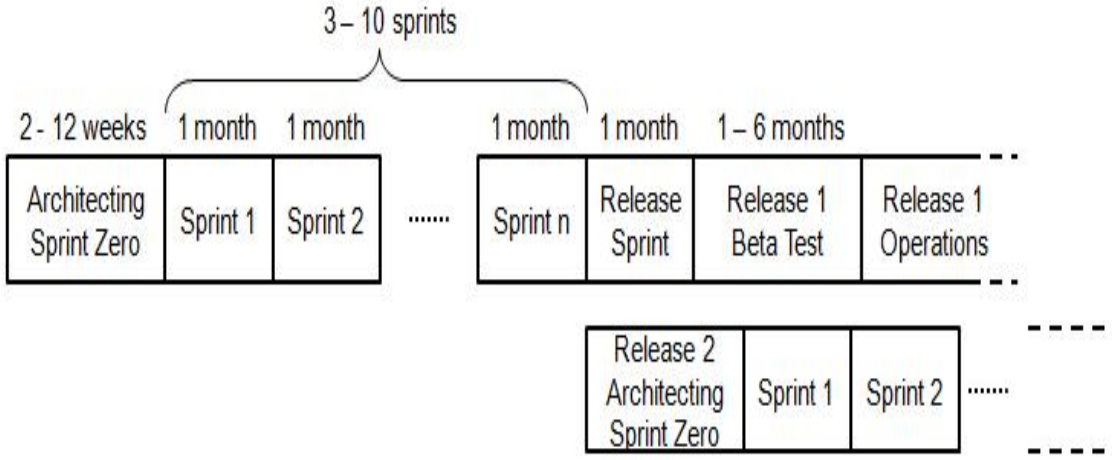
USA Medical Adoption Profile



- July 2004 - July 2005
  - Recruit top people from all sites into core team(s)
  - Get external expert help
  - Develop architecture
  - Early Scrum successes with infrastructure
  - Revise policies and practices
  - Train, reculture everyone
  - Manage expectations
- July 2005 – July 2006
  - Begin full-scale development
  - Core teams as mentors

Architected Agile Approach

- Uses Scrum of Scrums approach
  - Up to 10 Scrum teams of 10 people each
  - Has worked for distributed international teams
  - Going to three levels generally infeasible
- General approach shown below
  - Often tailored to special circumstances



Architected Agile – USA Medical

- Include customers and marketers
  - New roles; do's/don'ts/opportunities; CRACK personnel; full collaboration and teamwork; expectations management
- Scrum; most XP practices; added company practices
  - 6-12 person teams with team rooms, dedicated servers
  - Hourly smoke test; nightly build and regression test
  - Just-in-time analysis; story-point estimates; fail fast; detailed short-term plans; company architecture compliance
  - Embrace change in applications and practices
  - Global teams: wikis, daily virtual meetings, act as if next-door
- Release management
  - 2-12 week architecting Sprint Zero; 3-10 1-month Sprints; Release Sprint; 1-6 month beta test
  - Next Sprint Zero concurrent with Release Sprint
- Initiative manager and team
  - Define practices; evolve infrastructure; provide training; guide implementation; evaluate compliance/usage; continuous improvement

Case 11: Brownfield

- Example: Incremental legacy phaseout
- Size/complexity: High to very high
- Anticipated change rate (% per month): 0.3-3
- Criticality: Medium-high
- NDI support: NDI as legacy replacement
- Organizational and personnel capability: Legacy re-engineering
- Key Stage I activities: Re-engineer/refactor legacy into services
- Key Stage II activities: Incremental legacy phaseout
- Time/build: 2-6 week/refactor
- Time/increment: 2-6 months

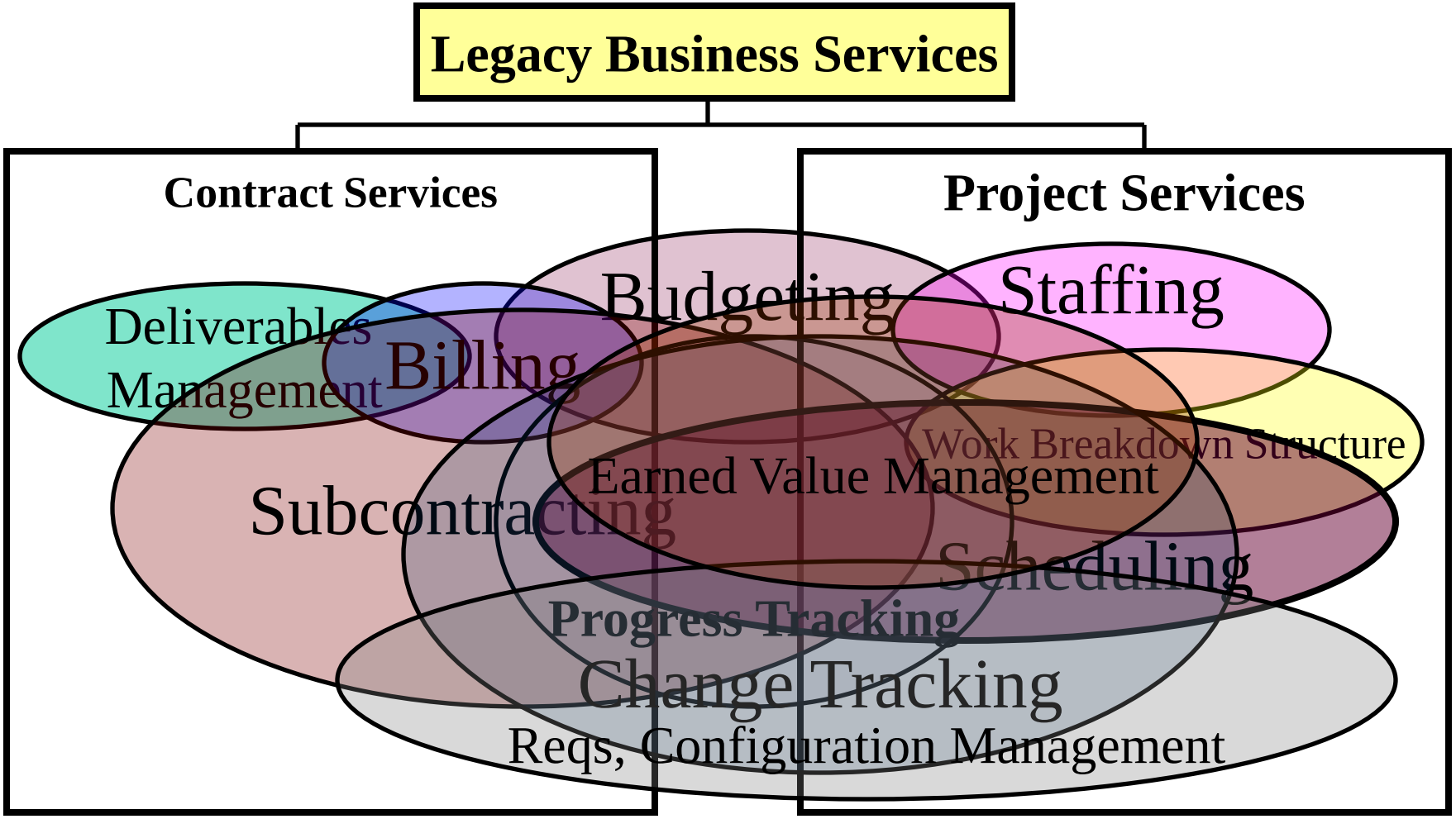
ICSM and Brownfield Development

- Many process models are Greenfield-oriented
  - Requirements→Design→Develop→Test→Operate
- Failed Greenfield project example
  - Corporate central financial system
  - To replace spaghetti-code collection of COBOL programs
- Improved ICSM Brownfield approach
  - Concurrent new-system definition and legacy system re-engineering

Failed Greenfield Corporate Financial System

- Used waterfall approach
  - Gathered requirements
  - Chose best-fit ERP system
  - Provided remaining enhancements
- Needed to ensure continuity of service
  - Planned incremental phase-in of new services
- Failed due to inability to selectively phase out legacy services
  - Dropped after 2 failed tries at cost of $40M

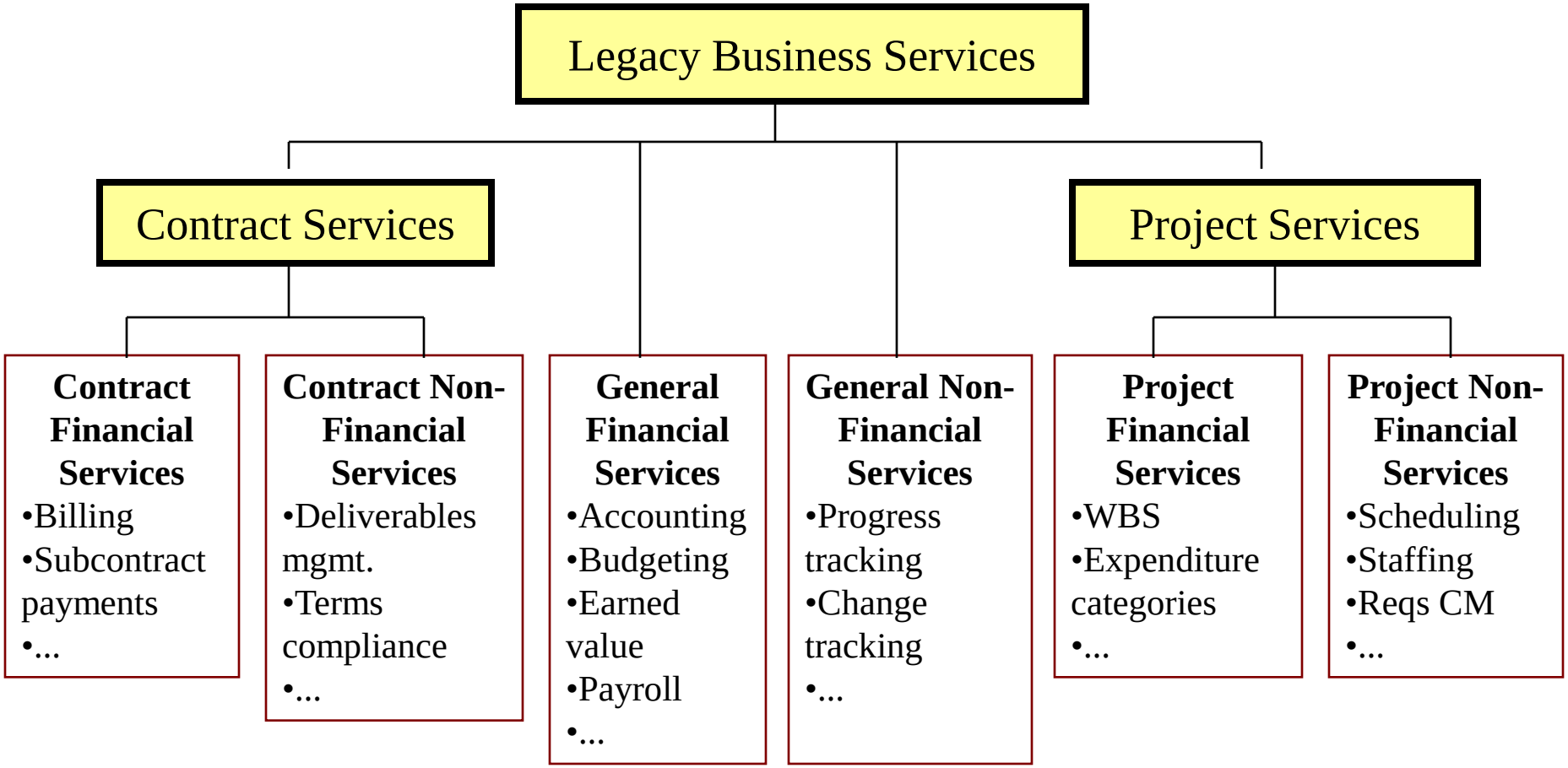
Legacy Systems Patched, Highly Coupled Financial and Non-Financial Services



ICSM Approach to Brownfield Engineering

- Understanding needs
  - Analysis of legacy system difficulties
- Envisioning opportunities
  - Concurrently decouple legacy financial and non-financial services, explore new system phase-in and architecture options
- System scoping and architecting
  - Extract legacy financial, non-financial services
  - Prioritize, plan for incremental financial services phase-in/out
- Feasibility evidence development
  - Successful examples of representative service extractions
  - Evidence of cost, schedule, performance feasibility

Result of Legacy Re-engineering



Another Frequently Asked Question

- **Q: Having all that ICSM generality and then using the decision table to come back to a simple model seems like an overkill.**
  - **If my risk patterns are stable, can't I just use the special case indicated by the decision table?**

- **A: Yes, you can and should – as long as your risk patterns stay stable. But as you encounter change, the ICSM helps you adapt to it.**
  - **And it helps you collaborate with other organizations that may use different special cases.**

A Feasibility Evidence Data Item Description

- Schedule-based and event-based reviews are risk-prone
  - Their DIDs focus on specifications and traceability
  - Optional evidence preparation is frequently absent
- Evidence-based reviews enable early risk resolution
  - They require more up-front systems engineering effort
  - They have a high ROI for high-risk projects
  - They synchronize and stabilize concurrent engineering
  - The evidence becomes a first-class deliverable
    - It requires planning and earned value management
- There are no DIDs for feasibility evidence
  - Path of least resistance is to use existing DIDs
- Proposed DID provides an evidence-based alternative
  - Based on successful use on related very large and small projects
  - Enables tailoring-up vs. always tailoring down

Types of Milestone Reviews

- Schedule-based reviews (contract-driven)
  - We'll hold the PDR on April 1 whether we have a design or not
  - High probability of proceeding into a Death March
- Event-based reviews (artifact-driven)
  - The design will be done by June 1, so we'll have the review then
  - Large "Death by PowerPoint and UML" event
    - Hard to avoid proceeding with many unresolved risks and interfaces
- Evidence-based commitment reviews (risk-driven)
  - Evidence provided in Feasibility Evidence Description (FED)
    - A first-class deliverable
  - Shortfalls in evidence are uncertainties and risks
  - Should be covered by risk mitigation plans
  - Stakeholders decide to commit based on risks of going forward

Nature of FEDs and Evidence-Based Milestones

- ***Evidence*** **provided by developer and validated by independent experts that: If the system is built to the specified architecture, it will**
  - **Satisfy the specified operational concept and requirements**
    - **Capability, interfaces, level of service, and evolution**
  - **Be buildable within the budgets and schedules in the plan**
  - **Generate a viable return on investment**
  - **Generate satisfactory outcomes for all of the success-critical stakeholders**
- **Shortfalls in evidence are uncertainties and risks**
  - **Should be resolved or covered by risk management plans**
- **Assessed in increasing detail at major anchor point milestones**
  - **Serves as basis for stakeholders' commitment to proceed**
  - **Serves to synchronize and stabilize concurrently engineered elements**

Can be used to strengthen current schedule- or event-based reviews

Nature of Feasibility Evidence

- Not just traceability matrices and PowerPoint charts
- Evidence can include results of
  - Prototypes: of networks, robots, user interfaces, COTS interoperability
  - Benchmarks: for performance, scalability, accuracy
  - Exercises: for mission performance, interoperability, security
  - Models: for cost, schedule, performance, reliability; tradeoffs
  - Simulations: for mission scalability, performance, reliability
  - Early working versions: of infrastructure, data fusion, legacy compatibility
  - Previous experience
  - Combinations of the above
- Validated by independent experts
  - Realism of assumptions
  - Representativeness of scenarios
  - Thoroughness of analysis
  - Coverage of key off-nominal conditions

Steps for Developing FED

Step	Description	Examples/Detail
A	Develop phase work-products/artifacts	For a Development Commitment Review, this would include the system's operational concept, prototypes, requirements, architecture, life cycle plans, and associated assumptions
B	Determine most critical feasibility assurance issues	Issues for which lack of feasibility evidence is program-critical
C	Evaluate feasibility assessment options	Cost-effectiveness; necessary tool, data, scenario availability
D	Select options, develop feasibility assessment plans	What, who, when, where, how, how much
E	Prepare FED assessment plans and earned value milestones	Example to follow…
F	Begin monitoring progress with respect to plans	Also monitor changes to the project, technology, and objectives, and adapt plans
G	Prepare evidence-generation enablers	Assessment criteria Parametric models, parameter values, bases of estimate COTS assessment criteria and plans Benchmarking candidates, test cases Prototypes/simulations, evaluation plans, subjects, and scenarios Instrumentation, data analysis capabilities
H	Perform pilot assessments; evaluate and iterate plans and enablers	Short bottom-line summaries and pointers to evidence files are generally sufficient
I	Assess readiness for Commitment Review	Shortfalls identified as risks and covered by risk mitigation plans Proceed to Commitment Review if ready
J	Hold Commitment Review when ready; adjust plans based on review outcomes	Review of evidence and independent experts' assessments

NOTE: "Steps" are denoted by letters rather than numbers to indicate that many are done concurrently.

Feasibility Evidence DID Overview

- Tailorable up from simple-project version
  - Criteria provided for simple, intermediate, and complex projects
- Complex-project version based on key SE studies
  - NRC Early Systems Engineering study
  - Services Probability of Program Success frameworks
  - NDIA-SEI SE Effectiveness Survey
  - INCOSE SE Leading Indicators
  - SISAIG SE Early Warning Indicators
- Organized into Goal-Critical Success Factor-Question Hierarchy
  - Tailorable up at each hierarchy level

Criteria for Simple, Intermediate, and Complex Projects

Criterion	Size	Complexity	Criticality	Capability
Criterion Content	Number of personnel	Novelty; Technical Risk; Stakeholder Conflicts; External Constraints	Loss due to defects	Personnel; Organization: relative to complexity & criticality
Simple Level	1 – 10	Low	Comfort; Discretionary funds	High - Very High
Intermediate Level	10 – 100	Mixed	Serious funds; Quality of life factors	Mixed
Complex Level	Over 100	All high to very high	Essential funds; Loss of human life	Low

FED DID General Information for Simple Projects

Project Name: _____

Project Primary Objective: _____

Success-Critical Stakeholders: _____

_____

_____

(Includes Role, Organization, Authorized Representatives and Contact Info for each stakeholder)

Life Cycle Process: ☐ Agile ☐ Architected Agile ☐ IC Spiral ☐ RUP ☐ Vee ☐ Other _____

Decision Milestone: _____

Key FED Dates:

Review Version Complete _____

Review Complete ____

Decision Meeting and Outcome Decided _____

The DID Tailoring-Up Framework: Goals, Critical Success Factors, and Questions

Goal 1. Concurrent definition of system requirements and solutions

CSF 1.1 Understanding of stakeholder needs: capabilities, operational concept, key performance parameters, enterprise fit (legacy)

1. At Milestone A, have the Key Performance Parameters (KPPs) been identified in clear, comprehensive, concise terms that are understandable to the users of the system?

2. Has a Concept of Operations (CONOPS) been developed showing that the system can be operated to handle both nominal and off-nominal workloads and meet response time requirements?

3. Has the ability of the system to meet mission effectiveness goals been verified through the use of modeling and simulation?

4. Have the success-critical stakeholders been identified and their roles and responsibilities negotiated?

(a) Have questions about the fit of the system into the stakeholders' context—acquirers, end users, administrators, interoperators, maintainers, etc.—been adequately explored?

Can Tailor DID Up at Goal or CSF Level

High-level Goals	Critical Success Factors
Concurrent definition of system requirements & solutions	Understanding of stakeholder needs
	Concurrent exploration of solutions
	System scoping & requirements definition
	Prioritization/allocation of requirements
System life-cycle organization, planning & staffing	Establishment of stakeholder RAAs
	Establishment of IPT RAAs
	Establishment of resources to meet objectives
	Establishment of selection/contracting/incentives
	Assurance of necessary personnel competencies
Technology maturing & architecting	COTS/NDI evaluation, selection, validation
	Life-cycle architecture definition & validation
	Use of prototypes, models, etc. to validate maturity
	Validated budgets & schedules
Evidence-based progress monitoring & commitment reviews	Monitoring of system definition
	Monitoring of feasibility evidence development
	Monitoring/assessment/re-planning for changes
	Identification and mitigation for feasibility risks

Example of Tailoring-Up Use

- Quantitative Methods, Inc. (QMI) is a leader in developing complex object-recognition systems (ORS)
- Coast Guard contracting with QMI for an ORS
  - Simpler than ORSs developed for Navy, Air Force
  - But includes new university-research algorithms
  - Uncertainty in performance leads to KPP ranges in contract
- Only a few of Goals and CSFs need to be tailored in
  - CSF 1.1 Understanding of stakeholder needs: key performance parameters
  - Question 1 on KPP identification covered by KPP ranges
  - Question 3 on effectiveness verification tailored in
  - CSF 1.2 Concurrent exploration of solution opportunities tailored in to address alternative high-performance-computing platforms
  - CSF 1.3 on system scoping and CSF 1.4 on requirements prioritization tailored out due to being already covered

Spreadsheet Tool Enables Risk Monitoring

Exposure	Question #	Impact: Critical / 40-100%	Impact: Significant / 20-40%	Impact: Moderate / 2-20%	Impact: Little-No impact / 0-2%	Evidence/Risk: Little-None / p(0.4-1.0)	Evidence/Risk: Weak / p(0.2-0.4)	Evidence/Risk: Partial / p(0.02-0.2)	Evidence/Risk: Strong / p(0.0-0.02)	NOTE: Impact and evidence/risk ratings should be done independently. The impact rating should estimate the effect a failure to address the specified item might have on the program. The evidence rating should specify the qualtity of evidence that has been provided, which demonstrates that the specified risk item has been satisfactorily addressed.	Reset / Risk Exposure	
	Goal 1:									**Concurrent definition of system requirements and solutions**		
	Critical Success Factor 1.1									**Understanding of stakeholder needs: capabilities, operational concept, key performance parameters, enterprise fit (legacy)**	4	
1	1.1(a)	●	○	○	◉	◉	○	○	○	At Milestone A, have the KPPs been identified in clear, comprehensive, concise terms that are understandable to all stakeholders?		No forma
3	1.1(b)	●	○	◉	○	○	◉	○	○	Has a CONOPS been developed showing that the system can be operated to handle both nominal and off-nominal workloads, to meet response time requirements, and generally to meet the defined KPPs?		IT system
3	1.1(c)	●	○	◉	○	○	◉	○	○	Has the ability of the system to meet mission effectiveness goals been verified through the use of modeling and simulation?		IT system effectiven
4	1.1(d)	◉	○	○	○	○	○	◉	○	Have the success-critical stakeholders been identified, their roles and responsibilities negotiated, and their needs clearly represented by the KPPs and CONOPS?		Developm Stakehold
4	1.1(e)	◉	○	○	○	○	○	◉	○	Have issues about the fit of the system into the stakeholders' context -- acquirers, end users, administrators, interoperators, maintainers, etc. -- been adequately explored?		Explored after syst related to different

Summary

- Schedule-based and event-based reviews are risk-prone
  - Their DIDs focus on specifications and traceability
  - Optional evidence preparation is frequently absent
- Evidence-based reviews enable early risk resolution
  - They require more up-front systems engineering effort
  - They have a high ROI for high-risk projects
  - They synchronize and stabilize concurrent engineering
  - The evidence becomes a first-class deliverable
    - It requires planning and earned value management
- There are no DIDs for feasibility evidence
  - Path of least resistance is to use existing DIDs
- Proposed DID provides an evidence-based alternative
  - Based on successful use on related very large and small projects
  - Enables taioring-up vs. always tailoring down

ICSM Transition Paths

- Existing programs may benefit from some ICSM principles and practices, but not others
- Problem programs may find some ICSM practices helpful in recovering viability
- Primary opportunities for incremental adoption of ICSM principles and practices
  - Supplementing traditional requirements and design reviews with development and review of feasibility evidence
  - Stabilized incremental development and concurrent architecture rebaselining
  - Using schedule as independent variable and prioritizing features to be delivered
  - Continuous verification and validation
  - Using the process decision table
- For additional ICSM information, see http://csse.usc.edu (Tech Report 2009-500)

ICSM Summary

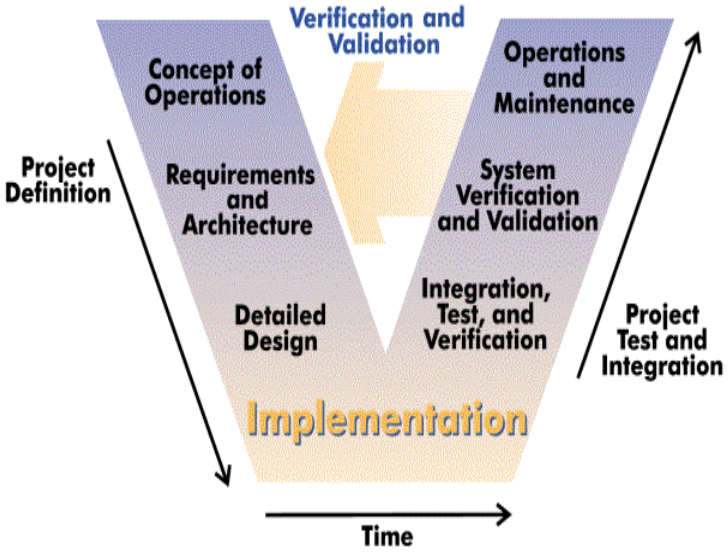
- Current acquisition processes not well suited to future challenges
  - Emergent, rapidly changing requirements
  - High assurance of scalable performance and qualities
- Incremental Commitment Model addresses challenges
  - Assurance via evidence-based milestone commitment reviews, stabilized incremental builds with concurrent V&V
    - Evidence shortfalls treated as risks
  - Adaptability via concurrent agile team handling change traffic and providing evidence-based rebaselining of next-increment specifications and plans
  - Use of critical success factor principles: stakeholder satisficing, incremental growth, concurrent engineering, iterative development, risk-based activities and milestones
  - Can be adopted incrementally
- Major implications for funding, contracting, career paths

Implications for funding, contracting, career paths

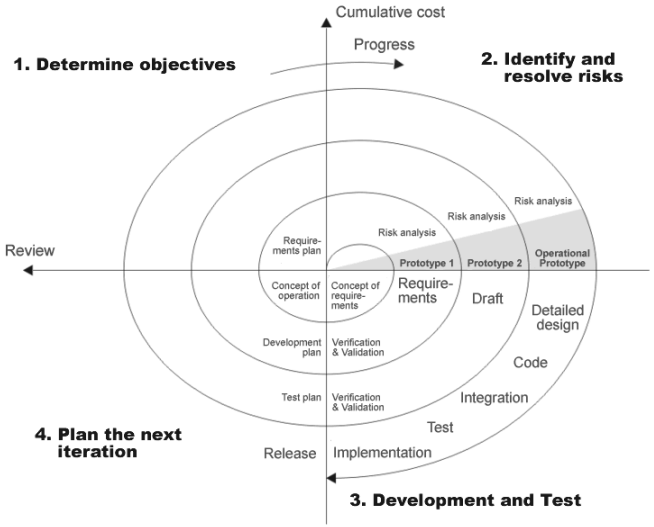
- Incremental vs. total funding
  - Often with evidence-based competitive downselect
- No one-size-fits all contracting
  - Separate instruments for build-to-spec, agile rebaselining, V&V teams
    - With funding and award fees for collaboration, risk management
    - Compatible regulations, specifications, and standards
    - Compatible acquisition corps education and training
  - Generally, schedule/cost/quality as independent variable
    - Prioritized feature set as dependent variable
- Multiple career paths
  - For people good at build-to-spec, agile rebaselining, V&V
  - For people good at all three
    - Future program managers and chief engineers

Current System Acquisition Methods
Too easy to misinterpret as one-size-fits-all

- **V-Model**[1]



- **Spiral Model**[2]



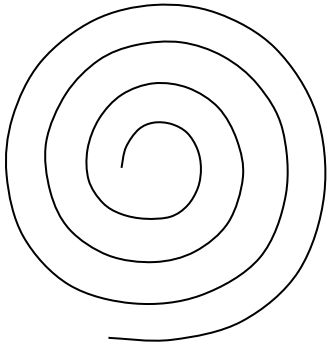
High level guidance assumes that acquirers have extensive acquisition experience...
Without experience, too easy to misinterpret and auger in with disastrous results...

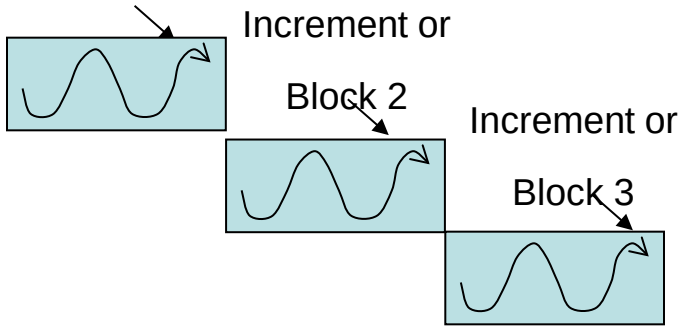
[1] http://en.wikipedia.org/wiki/V-Model

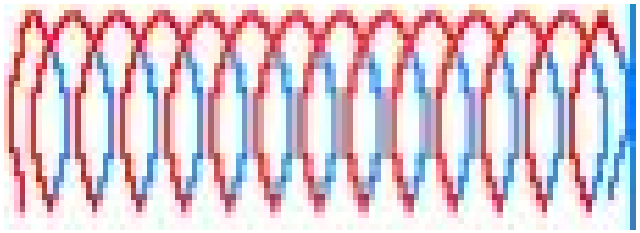
[2] http://en.wikipedia.org/wiki/Spiral_model

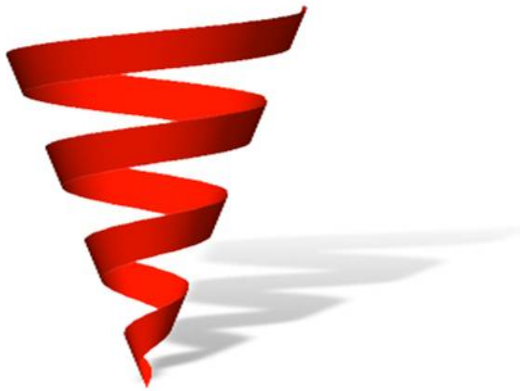
Principles Trump Diagrams: Spiral

Several US Government Programs









Where's risk-driven process branching?

Incremental Commitment In Systems and Life: Anchor Point Milestones

- Common System/Software stakeholder commitment points
  - Defined in concert with Government, industry organizations
  - Initially coordinated with Rational's Unified Software Development Process
- Exploration Commitment Review (ECR)
  - Stakeholders' commitment to support initial system scoping
  - Like dating
- Validation Commitment Review (VCR)
  - Stakeholders' commitment to support system concept definition and investment analysis
  - Like going steady

Incremental Commitment In Systems and Life: Anchor Point Milestones *(continued)*

- **Foundations Commitment Review (FCR)**
  - **Stakeholders' commitment to support system architecting**
  - **Like getting engaged**
- **Development Commitment Review (DCR)**
  - **Stakeholders' commitment to support system development**
  - **Like getting married**
- **Incremental Operational Capabilities (OCs)**
  - **Stakeholders' commitment to support operations**
  - **Like having children**

ICSM Anchor Point Milestone Content (1)

(Risk-driven level of detail for each element)

Milestone Element	Foundations Commitment Review	Development Commitment Review
Definition of Operational Concept	• Top-level system objectives and scope – System boundary – Environment parameters and assumptions – Evolution parameters • Operational concept – Operations and maintenance scenarios and parameters – Organizational life-cycle responsibilities (stakeholders)	• Elaboration of system objectives and scope of increment • Elaboration of operational concept by increment
System Prototype(s)	• Exercise key usage scenarios • Resolve critical risks	• Exercise range of usage scenarios • Resolve major outstanding risks
Definition of System Requirements	• Top-level functions, interfaces, quality attribute levels, including – Growth vectors and priorities – Prototypes • Stakeholders' concurrence on essentials	• Elaboration of functions, interfaces, quality attributes, and prototypes by increment • Identification of TBD's (to-be-determined items) • Stakeholders' concurrence on their priority concerns

ICSM Anchor Point Milestone Content (2)

(Risk-driven level of detail for each element)

Milestone Element	Foundations Commitment Review	Development Commitment Review
Definition of System and Software Architecture	• Top-level definition of at least one feasible architecture – Physical and logical elements and relationships – Choices of COTS and reusable software elements • Identification of infeasible architecture options	• Choice of architecture and elaboration by increment – Physical and logical components, connectors, configurations, constraints – COTS, reuse choices – Domain-architecture and architectural style choices • Architecture evolution parameters
Definition of Life-Cycle Plan	• Identification of life-cycle stakeholders – Users, customer, developers, maintainers, interoperators, general public, others • Identification of life-cycle process model – Top-level stages, increments • Top-level WWWWWHH* by stage	• Elaboration of WWWWWHH* for Initial Operational Capability (IOC) – Partial elaboration, identification of key TBD's for later increments
Feasibility Evidence	• Assurance of consistency among elements above – Via analysis, measurement, prototyping, simulation, etc. – Business case analysis for requirements, feasible architectures	• Assurance of consistency among elements above • All major risks resolved or covered by risk management plan

*WWWWWHH: Why, What, When, Who, Where, How, How Much

References - I

- Beck, K., Extreme Programming Explained, Addison Wesley, 1999.
- Boehm, B., "Some Future Software Engineering Opportunities and Challenges," In Sebastian Nanz (Ed.): The Future of Software Engineering, Springer Berlin Heidelberg, 2011, pp. 1-32.
- Boehm, B., Brown, W., Basili, V., and Turner, R., "Spiral Acquisition of Software-Intensive Systems of Systems, CrossTalk, Vol. 17, No. 5, pp. 4-9, 2004.
- Boehm, B. and Lane J., "21st Century Processes for Acquiring 21st Century Software-Intensive Systems of Systems." CrossTalk: Vol. 19, No. 5, pp.4-9, 2006.
- Boehm, B., and Lane, J., "Using the ICSM to Integrate System Acquisition, Systems Engineering, and Software Engineering," CrossTalk, October 2007, pp. 4-9.
- Boehm, B., and Lane, J., "A Process Decision Table for Integrated Systems and Software Engineering," Proceedings, CSER 2008, April 2008.
- Boehm, B. et al., Software Cost Estimation with COCOMO II, Prentice Hall, 2000.
- Boehm, B. and Lane, J., "Evidence-Based Software Processes," New Modeling Concepts for Today's Software Processes, Springer Lecture Notes in Computer Science, 2010, Volume 6195/2010, pp. 62-73.
- Boehm, B., Lane, J., Koolmanojwong, S., and Turner, R., "An Evidence-Based SE Data Item Description," Proceedings, CSER 2013, Elsevier, www.sciencedirect.com
- Checkland, P., Systems Thinking, Systems Practice, Wiley, 1980 (2nd ed., 1999).
- Electronic Industries Alliance (1999); EIA Standard 632: Processes for Engineering a System
- Hall, E.T., Beyond Culture, Anchor Books/Doubleday, 1976.

References -II

- Highsmith, J., Adaptive Software Development, Dorset House, 2000.
- International Standards Organization, Information Technology Software Life Cycle Processes, ISO/IEC 12207, 1995
- ISO, Systems Engineering – System Life Cycle Processes, ISO/IEC 15288, 2002.
- Krygiel, A., Behind the Wizard's Curtain; CCRP Publication Series, July, 1999, p. 33
- Lane, J. and Boehm, B., "System of Systems Cost Estimation: Analysis of Lead System Integrator Engineering Activities", Information Resources Management Journal, Vol. 20, No. 2, pp. 23-32, 2007.
- Lane, J. and Valerdi, R., "Synthesizing SoS Concepts for Use in Cost Estimation", Proceedings of IEEE Systems, Man, and Cybernetics Conference, 2005.
- Madachy, R., Boehm, B., Lane, J., "Assessing Hybrid Incremental Processes for SISOS Development", USC CSSE Technical Report USC-CSSE-2006-623, 2006.
- Maier, M., "Architecting Principles for Systems-of-Systems"; Systems Engineering, Vol. 1, No. 4 (pp 267-284).
- Maier, M., "System and Software Architecture Reconciliation," Systems Engineering 9 (2), 2006, pp. 146-159.
- Northrop, L., et al., Ultra-Large-Scale Systems: The Software Challenge of the Future, Software Engineering Institute, 2006.
- Pew, R. W., and Mavor, A. S., Human-System Integration in the System Development Process: A New Look, National Academy Press, 2007.
- Rechtin, E. Systems Architecting, Prentice Hall, 1991.
- Schroeder, T., "Integrating Systems and Software Engineering: Observations in Practice," OSD/USC Integrating Systems and Software Engineering Workshop, http://csse.usc.edu/events/2007/CIIForum/pages/program.html, October 2007.
- USC CSSE, ICSM Electronic Process Guide, http://greenbay.usc.edu/IICMSw/index.htm#publish.icm.base-usc/customcategories/icm_welcome_page_D99DA7B2.html

List of Acronyms

B/L	Baselined
C4ISR	Command, Control, Computing, Communications, Intelligence, Surveillance, Reconnaissance
CD	Concept Development
CDR	Critical Design Review
COTS	Commercial Off-the-Shelf
DCR	Development Commitment Review
DI	Development Increment
DoD	Department of Defense
ECR	Exploration Commitment Review
EVMS	Earned Value Management System
FCR	Foundations Commitment Review
FED	Feasibility Evidence Description
FMEA	Failure Modes and Effects Analysis
FRP	Full-Rate Production
GAO	Government Accountability Office
GUI	Graphical User Interface

List of Acronyms *(continued)*

HMI	Human-Machine Interface
HSI	Human-System Interface
HW	Hardware
ICSM	Incremental Commitment Model
IOC	Initial Operational Capability
IRR	Inception Readiness Review
IS&SE	Integrating Systems and Software Engineering
LCO	Life Cycle Objectives
LRIP	Low-Rate Initial Production
MBASE	Model-Based Architecting and Software Engineering
NDI	Non-Developmental Item
NRC	National Research Council
OC	Operational Capability
OCR	Operations Commitment Review
OO&D	Observe, Orient and Decide
OODA	Observe, Orient, Decide, Act
O&M	Operations and Maintenance

List of Acronyms *(continued)*

PDR	Preliminary Design Review
PM	Program Manager
PR	Public Relations
PRR	Product Release Review
RUP	Rational Unified Process
SoS	System of Systems
SoSE	System of Systems Engineering
SSE	Systems and Software Engineering
SW	Software
SwE	Software Engineering
SysE	Systems Engineering
Sys Engr	Systems Engineer
S&SE	Systems and Software Engineering
USD (AT&L)	Under Secretary of Defense for Acquisition, Technology, and Logistics
VCR	Validation Commitment Review
V&V	Verification and Validation
WBS	Work Breakdown Structure
WMI	Warfighter-Machine Interface

ACM: The Learning Continues...

- Questions about this webcast? learning@acm.org
- ACM Learning Webinars (including archives): http://learning.acm.org/webinar
- ACM Learning Center: http://learning.acm.org
- ACM SIGSOFT: http://www.sigsoft.org/
- ACM Queue: http://queue.acm.org