Balancing Agility and Architecture

Barry Boehm, USC

JAOO 2009

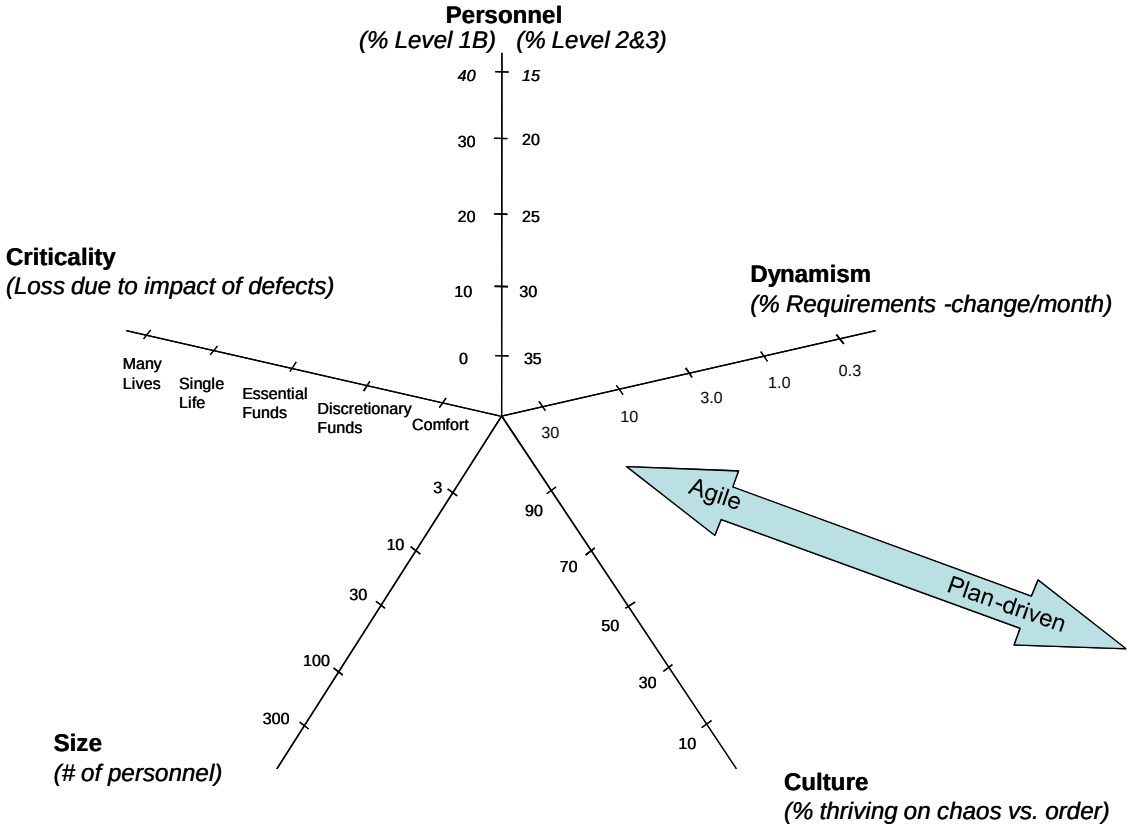
October 5, 2009

Outline

- Agility and Architecture home grounds
- How much architecting is enough
  - A quantitative analysis
- Processes for balancing agility and architecture
  - The Incremental Commitment Model
  - Process decision table
  - Hybrid approaches
- Conclusions

Agile and Plan-Driven Home Grounds: Five Critical Decision Factors

- Size, Criticality, Dynamism, Personnel, Culture



Relative Size of Agile and Architecture Home Grounds

By % of Projects

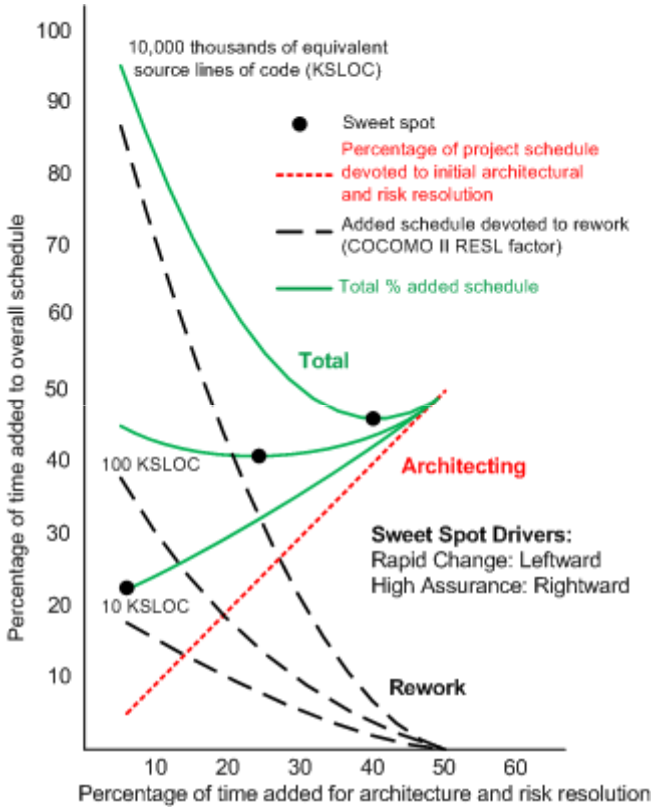
Stability \ Criticality, Size	Low (78%)	High (22%)
High	Either	Arch
Low (80%)	Agile	Both

By % of Costs

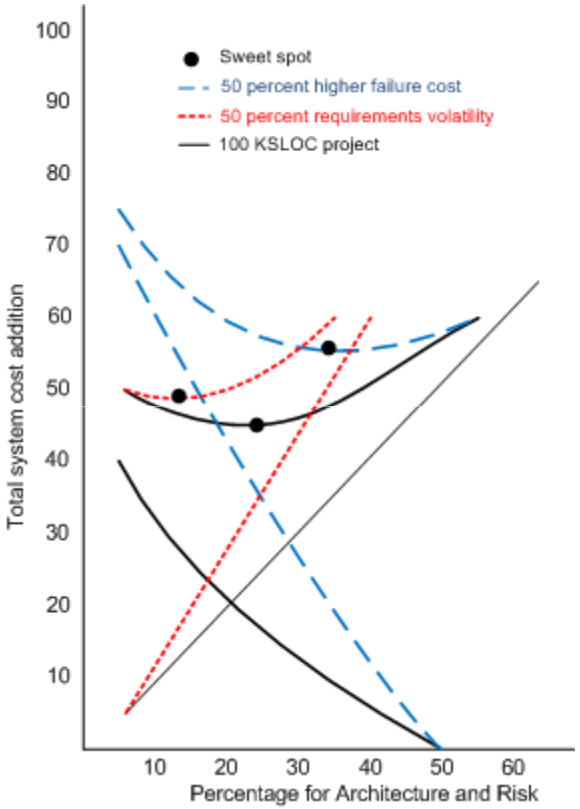
Stability \ Criticality, Size	Low (28%)	High (72%)
High	Either	Architecture
Low (80%)	Agile	Both

- **Based on size distributions in financial sector**
  - **65% small (<10 people)**
  - **25% medium (11-50 people)**
  - **10% large (>50 people)**

Effect on Size on Sweet Spots



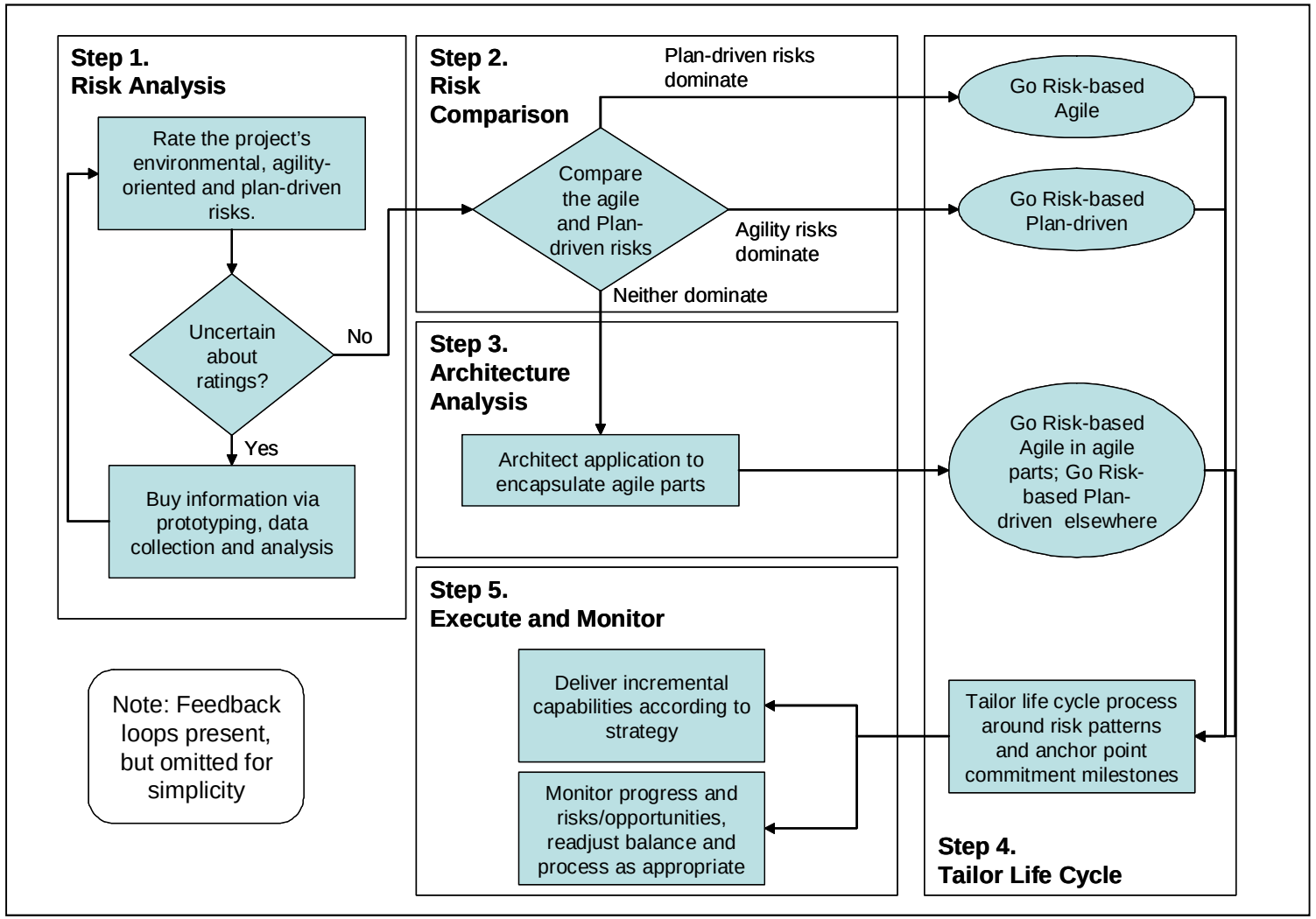
Effect of Volatility and Criticality on Sweet Spots



Outline

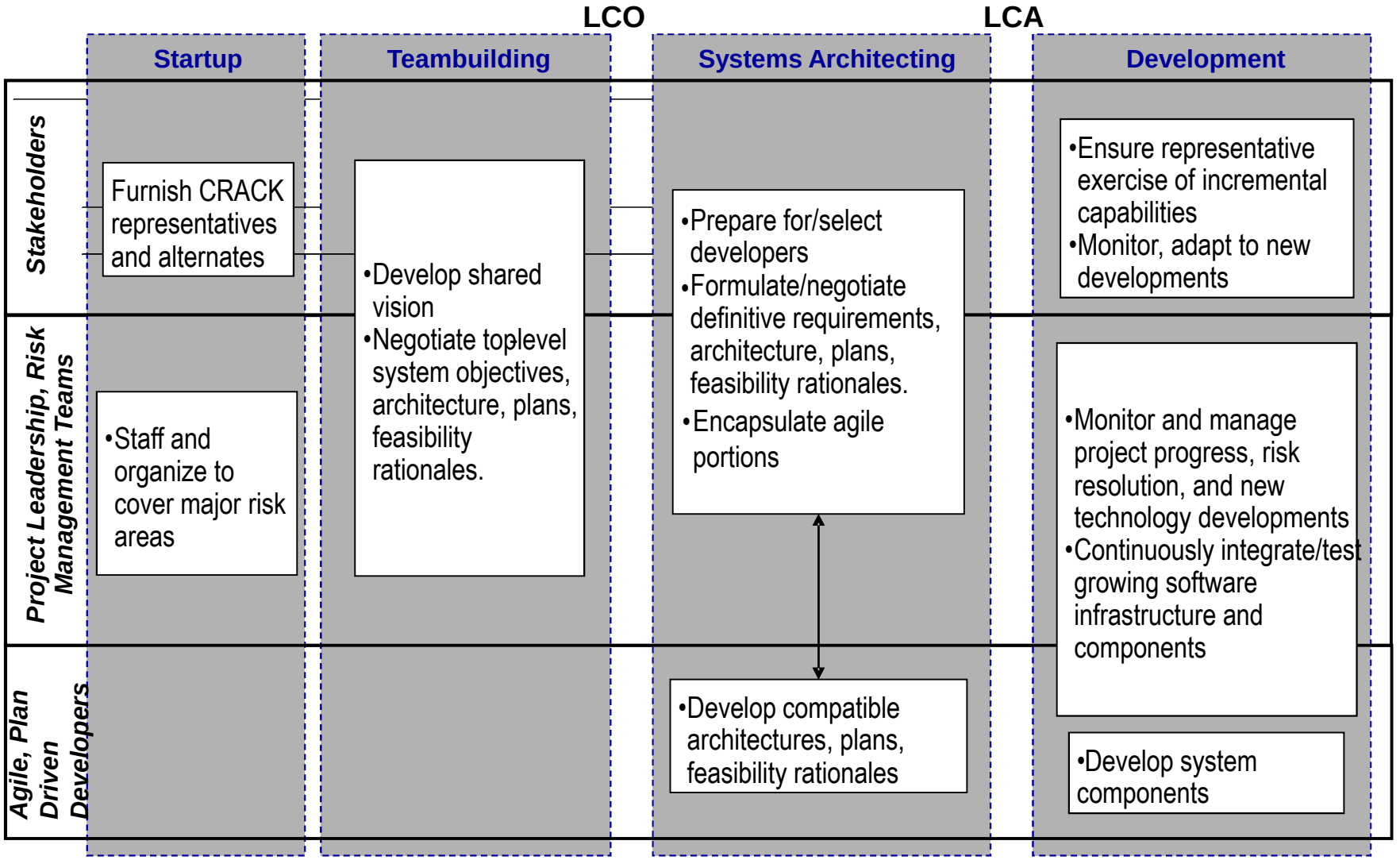
- Increasing importance of both agility and quality
- Challenges of achieving both agility and quality
- Approaches for achieving both agility and quality
- Case studies and critical success factors
- Conclusions

Using Risk to Balance Discipline and Agility - Overview



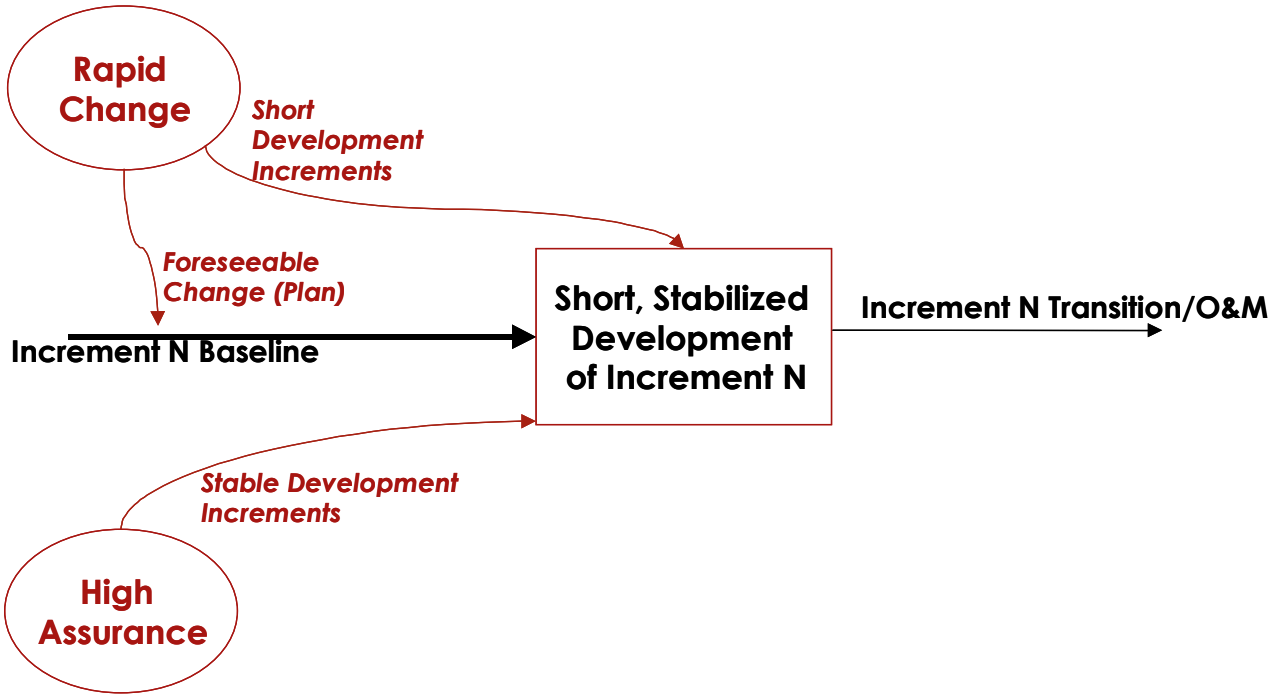
Hybrid Agile/Plan-Driven Strategy

– CRACK: collaborative, representative, authorized, committed, knowledgeable



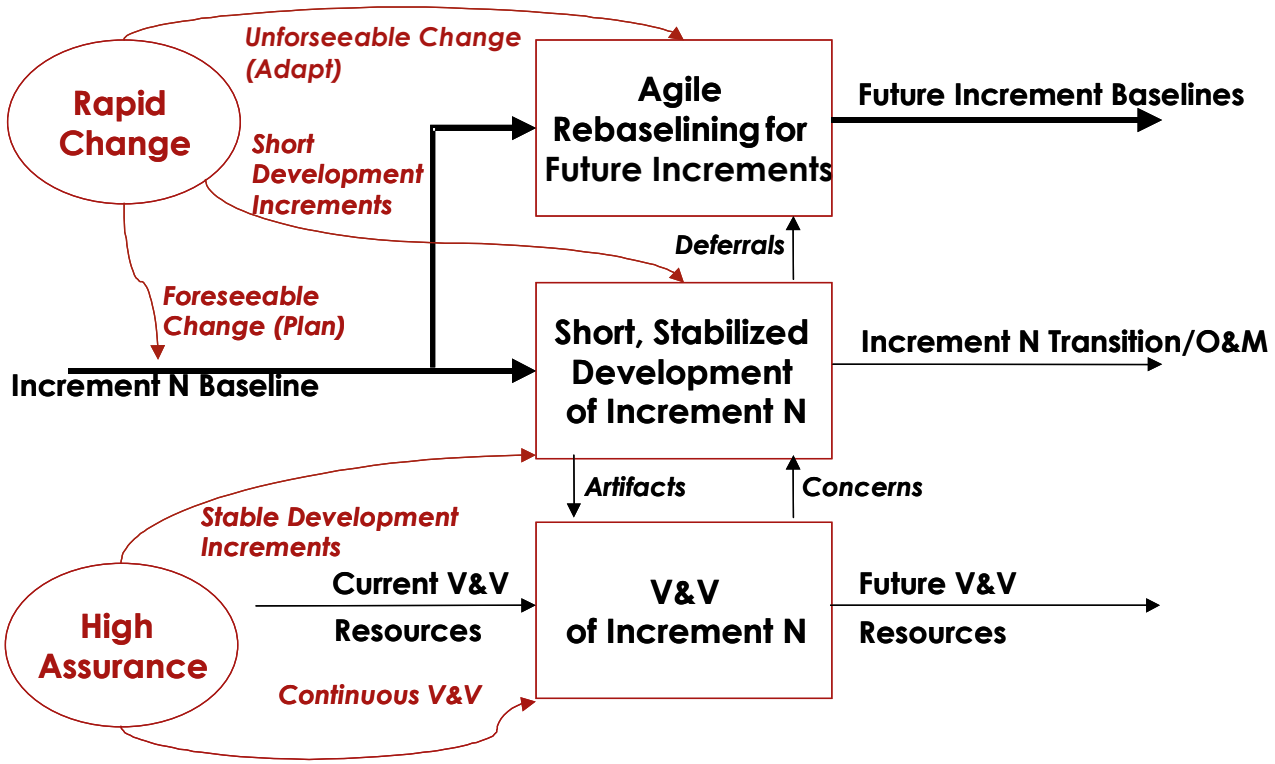
Incremental Commitment Model:

Single Increment View

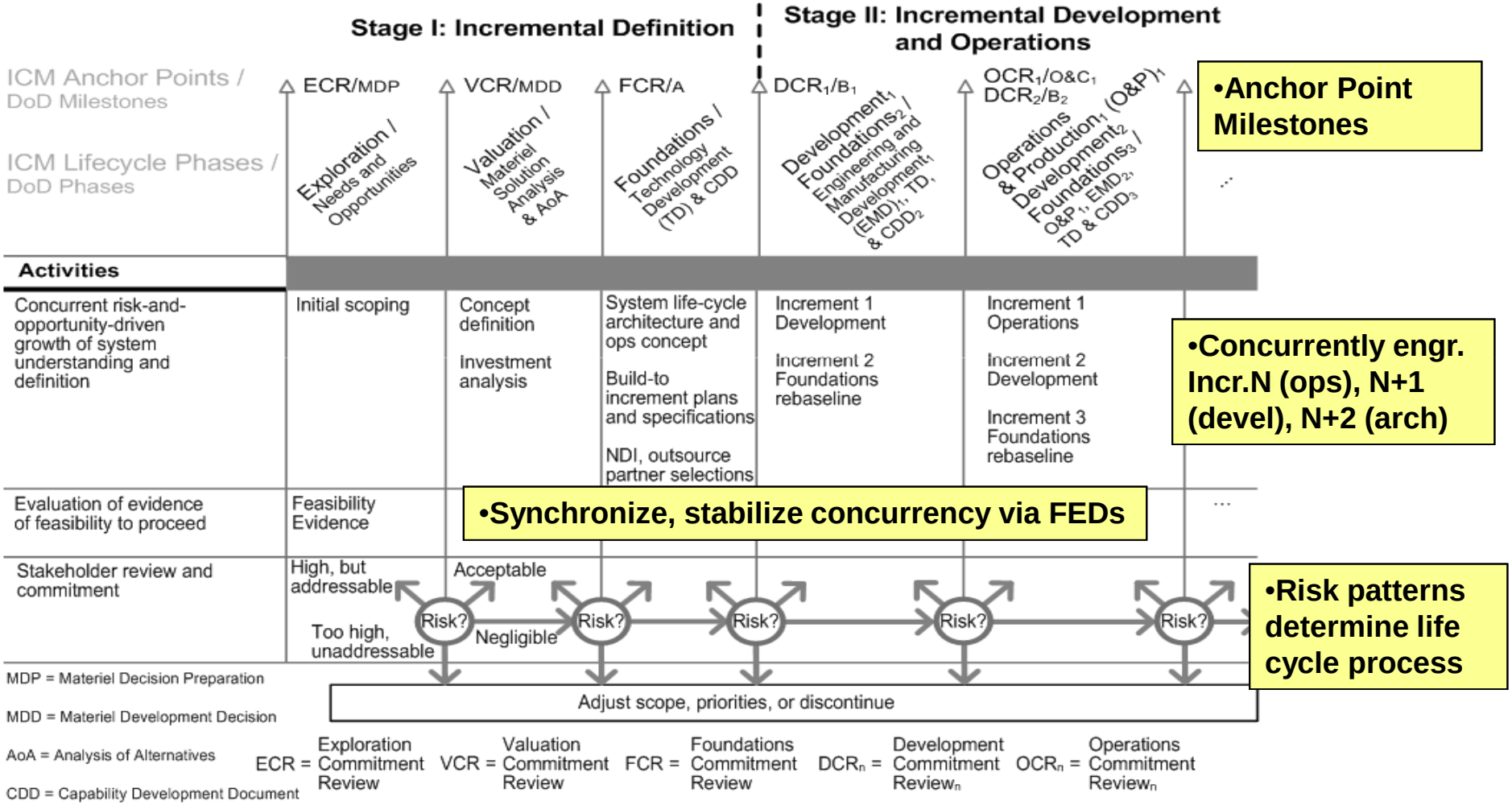


Incremental Commitment Model:

Single Increment View



The Incremental Commitment Life Cycle Process: Overview



Milestone Feasibility Rationales

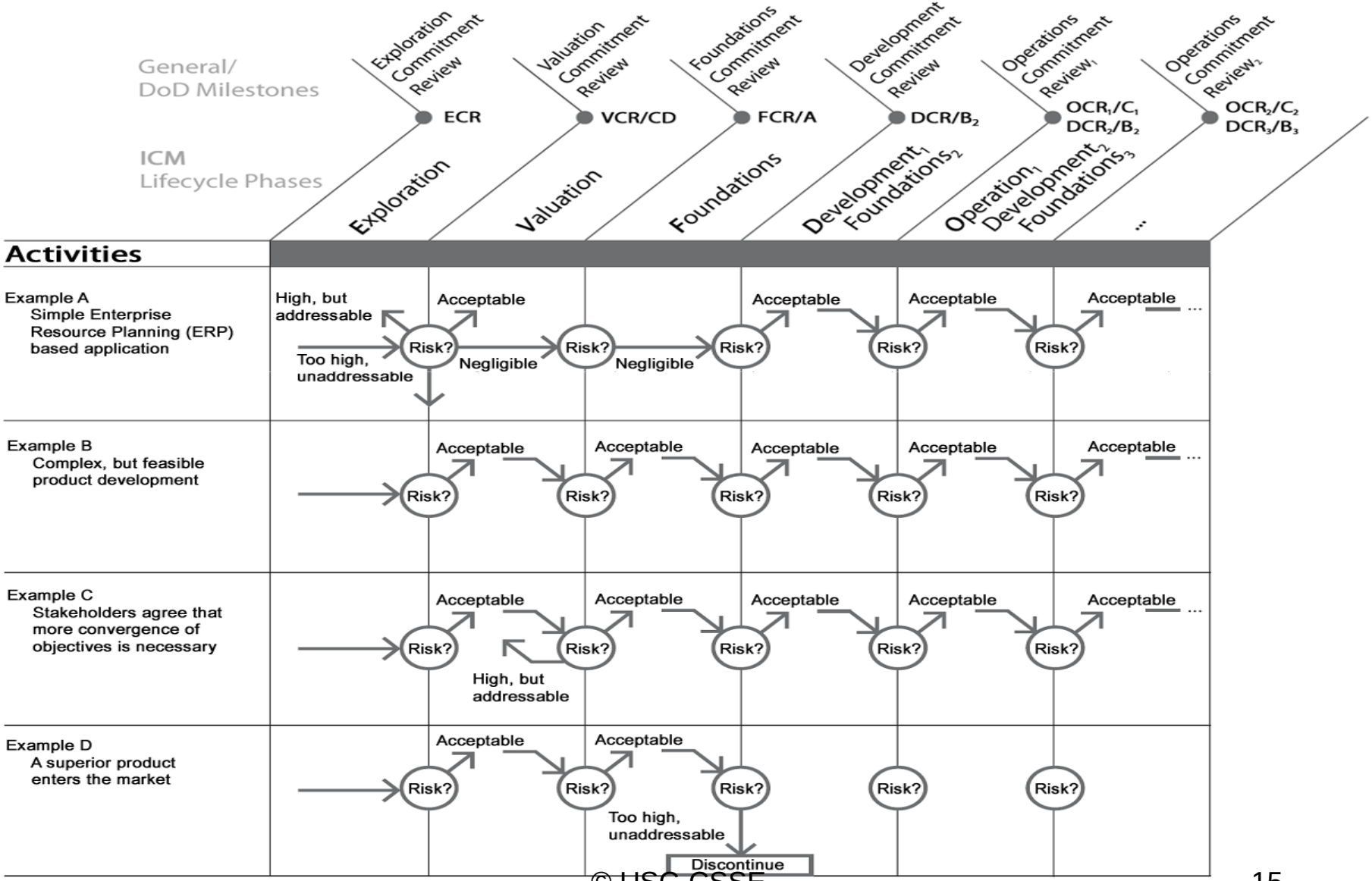
- **Evidence provided by developer and validated by independent experts that:**

  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**

The ICM as Risk-Driven Process Generator

- Stage I of the ICM 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 ICM 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 ICM Process Decision Table

- Key Decision Inputs
  - Product and project size and complexity
  - Requirements volatility
  - Mission criticality
  - Nature of Non-Developmental Item (NDI)* support
    - Commercial, open-source, reused components
  - Organizational and Personnel Capability
- 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

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

Case 1: Use NDI	Case 2: Agile
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	**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	**Case 4: Formal Methods**
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	**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 ICM (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 ICM 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 ICM, encapsulated agile in high change, low-medium criticality parts (Often HMI, external interfaces)
Key Stage II Activities (Incremental Development/Operations): Full ICM, 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 ICM (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 ICM; extensive multi-owner team building, negotiation
Key Stage II Activities (Incremental Development/Operations): Full ICM; 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 ICM (Cases 12a/b)

Case 12a: Net-Centric Services – Community Support	Case 12b: Net-Centric Services – Quick Response Decision Support
Example: Community services or special interest group **Size, Complexity:** Low to medium **Typical Change Rate/Month:** 0.3-3% **Criticality:** Low to medium **NDI Support:** Tailorable service elements **Organizational Personnel Capability:** NDI-experienced **Key Stage I Activities (Incremental Definition):** Filter, select, compose, tailor NDI **Key Stage II Activities (Incremental Development/Operations):** Evolve tailoring to meet community needs **Time/Build:** <= 1 day **Time/Increment:** 2-12 months	**Example:** Response to competitor initiative **Size, Complexity:** Medium to high **Typical Change Rate/Month:** 3-30% **Criticality:** Medium to high **NDI Support:** Tailorable service elements **Organizational Personnel Capability:** NDI-experienced **Key Stage I Activities (Incremental Definition):** Filter, select, compose, tailor NDI **Key Stage II Activities (Incremental Development/Operations):** Satisfy quick response; evolve or phase out **Time/Build:** <= 1 day **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:** Hard ware.
- **IOC:** Initial Operational Capability.
- **LSI:** Large Scale Integration.
- **LRIP:** Low-Rate Initial Production.
- **NDI:** Non-Development Item.
- **SW:** Software.

Large Systems Combine Special Cases: Need Overarching Architecture

- **Use Parnas architecting-for-change approach**
  - **Identify primary sources of change**
  - **Encapsulate these in modules or services**
  - **Confines ripple effects of change within modules**
- **Frequent large-change special cases**
  - **User interfaces**
  - **Device drivers**
  - **Interfaces to external systems**
  - **Rapid-competition features**
  - **Deferred user features**

Frequently Asked Question

- Q: Having all that ICM 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 new situations, the ICM helps you adapt to them.
  - And it helps you collaborate with other organizations that may use different special cases.

Conclusions

- Many future systems will need both agility and architecture
- Risk analysis helps determine how much of each is enough
  - Balancing risks of doing too little, too much of each
  - Often varies across subsystems
- Parnas approach helps develop overarching architecture
  - Identify primary sources of change
    - Examples: User interfaces, Device drivers, Interfaces to external systems, Rapid-competition features, Deferred user features
  - Encapsulate these in modules or services
  - Confines ripple effects of change within modules
- Incremental Commitment Model provides tailorable risk-driven framework
  - And decision table for common special-case processes

References

Beck, K., *Extreme Programming Explained*, Addison Wesley, 1999.

Boehm, B., and Lane, J., "Guide for Using the Incremental Commitment Model (ICM) for Systems Engineering of DoD Projects, v0.5," USC-CSSE-TR-2009-500.

Boehm, B., "Some Future Trends and Implications for Systems and Software Engineering Processes", *Systems Engineering* 9(1), pp. 1-19, 2006.

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., "Using the ICM to Integrate System Acquisition, Systems Engineering, and Software Engineering," *CrossTalk*, October 2007, pp. 4-9.

Boehm, B. andTurner, R., *Balancing Agility and Discipline: A Guide for the Perplexed,* Addison Wesley, 2004.

Checkland, P., *Systems Thinking, Systems Practice*, Wiley, 1980 (2nd ed., 1999).

Department of Defense (DoD), *Instruction 5000.02, Operation of the Defense Acquisition System,* December 2008.

Department of Defense (DoD), *Systems Engineering Guide for System of Systems*, June 2008.

Lane, J. and Dahmann, J., "Process Evolution to Support System of Systems Engineering," Proceedings, ICSE 2008 ULSSIS Workshop, May 2008.

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.

Backup Charts

ICM HSI Levels of Activity for Complex Systems