



Process Decision Frameworks for DoD and e-Services Projects

ASRR 2011

Supannika Koolmanojwong

October 6, 2011



Outline

- **Motivation and Approach**
- **Process Decision Frameworks**
 - **SERC RT-5 Evolutionary Acquisition**
 - **Experimental Results for e-Services projects**
 - **Extension for other Process Patterns**
- **Conclusions**

Motivation and Approach

- **SERC RT-5 Study: SE Implications for Evolutionary Acquisition (EvA)**
 - No one-size-fits-all EvA model
 - Pros and Cons of EvA alternatives
 - Decision Table for EvA alternatives
- **Similar Situation for e-Services applications**
 - Several forms; need for decision table
 - Needed extensions to initial decision table
 - Extensions validated in 84-project experiment
- **Decision framework extended to other classes of systems**
 - Hardware/Software-intensive; system/family of systems, ...

There is No One-Size-Fits-All EvA Model

Type	Examples	Pros	Cons
Prespecified Sequential	Platform base plus PPPIs	Prespecifiable full-capability requirements, scalability when stable	Emergent requirements or rapid change, architecture breakers
Evolutionary Sequential	Small: Agile Larger: Rapid fielding	Adaptability to change, need for usage feedback	Easiest-first; late, costly fixes; SysE time gaps Slow for large systems
Evolutionary Overlapped	Stable development; Maturing technology	Mature technology upgrades	Emergent requirements or rapid change; SysE time gaps
Evolutionary Concurrent	Rapid, emergent development Systems of systems	Emergent requirements or rapid change, SysE continuity	Overkill on small or highly stable systems

Time phasing terms: Scoping; Architecting; Developing; Producing; Operating (SADPO)

Prespecified Sequential: SA; DPO1; DPO2; DPO3; ...

Evolutionary Sequential: SADPO1; SADPO2; SADPO3; ...

Evolutionary Overlapped: SADPO1;
 SADPO2;
 SADPO3; ...

Evolutionary Concurrent: SA; D1 ; PO1...
 SA2; D2 ; PO2...
 SA3; D3; PO3 ...

Evolutionary Acquisition (EvA) Decision Table

Type	Stable, prespecifiable requirements?	OK to wait for full system to be developed?	Need to wait for next-increment priorities?	Need to wait for next-increment enablers*?
Single Step	Yes	Yes		
Prespecified Sequential	Yes	No		
Evolutionary Sequential	No	No	Yes	
Evolutionary Overlapped	No	No	No	Yes
Evolutionary Concurrent	No	No	No	No

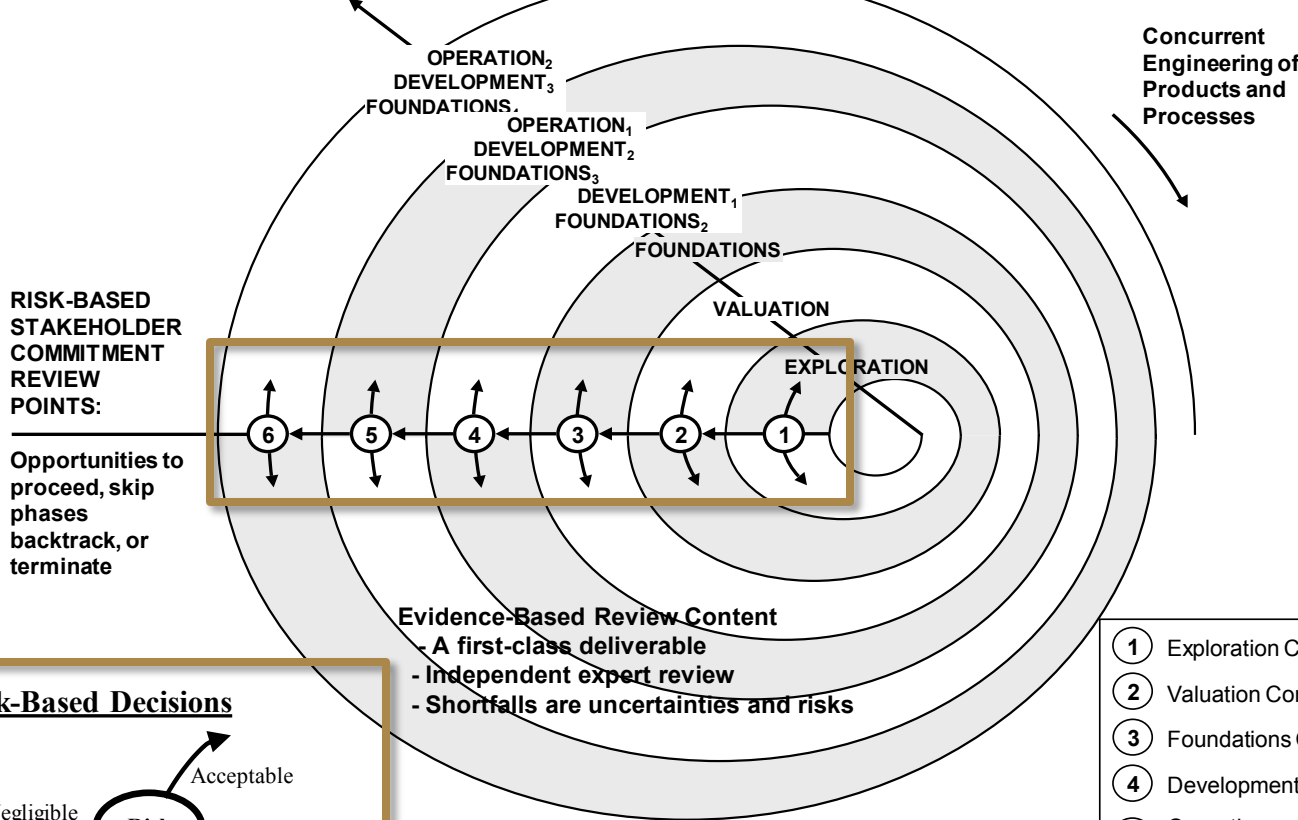
* **Example enablers: Technology maturity; External-system capabilities; Needed resources**

Motivation and Approach

- **SERC RT-5 Study: SE Implications for Evolutionary Acquisition (EvA)**
 - No one-size-fits-all EvA model
 - Pros and Cons of EvA alternatives
 - Decision Table for EvA alternatives
- ➔ • **Similar Situation for e-Services applications**
 - Several forms; need for decision table
 - Needed extensions to initial decision table
 - Extensions validated in 84-project experiment
- **Decision framework extended to other classes of systems**
 - Hardware/Software-intensive; system/family of systems, ...

The Incremental Commitment Spiral Model (ICSM)

Cumulative Level of Understanding, Product and Process Detail (Risk-Driven)



Concurrent Engineering of Products and Processes

4 Key Principles:

- Stakeholder value-based system definition and evolution
- Incremental commitment and accountability
- Concurrent system and software definition and development
- Evidence and risk-based decision making

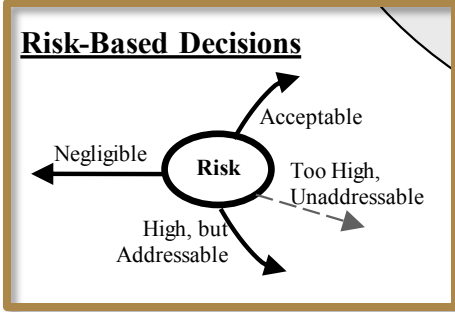
RISK-BASED STAKEHOLDER COMMITMENT REVIEW POINTS:

Opportunities to proceed, skip phases backtrack, or terminate

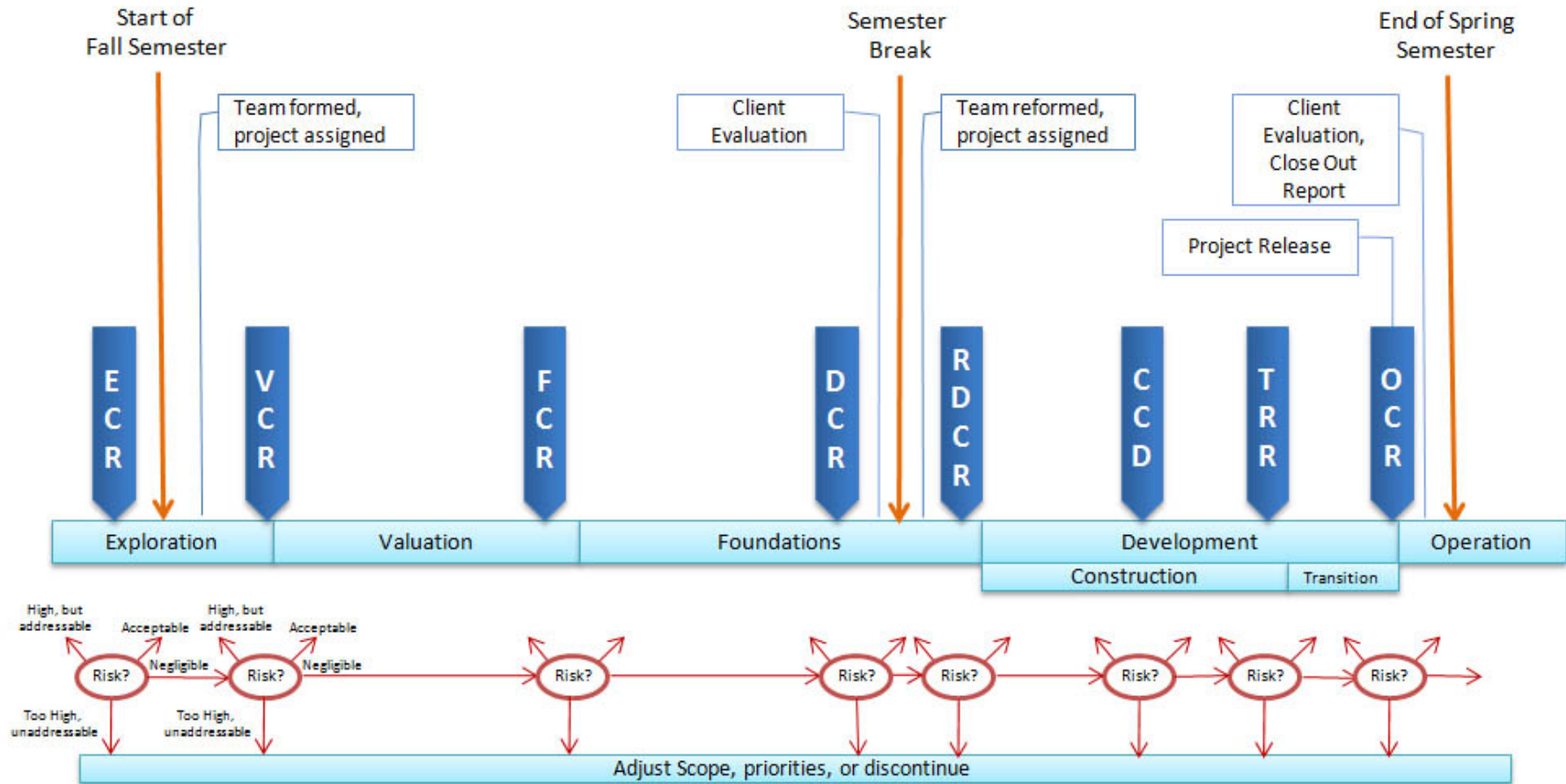
Evidence-Based Review Content

- A first-class deliverable
- Independent expert review
- Shortfalls are uncertainties and risks

- ① Exploration Commitment Review
- ② Valuation Commitment Review
- ③ Foundations Commitment Review
- ④ Development Commitment Review
- ⑤ Operations₁ and Development₂ Commitment Review
- ⑥ Operations₂ and Development₃ Commitment Review

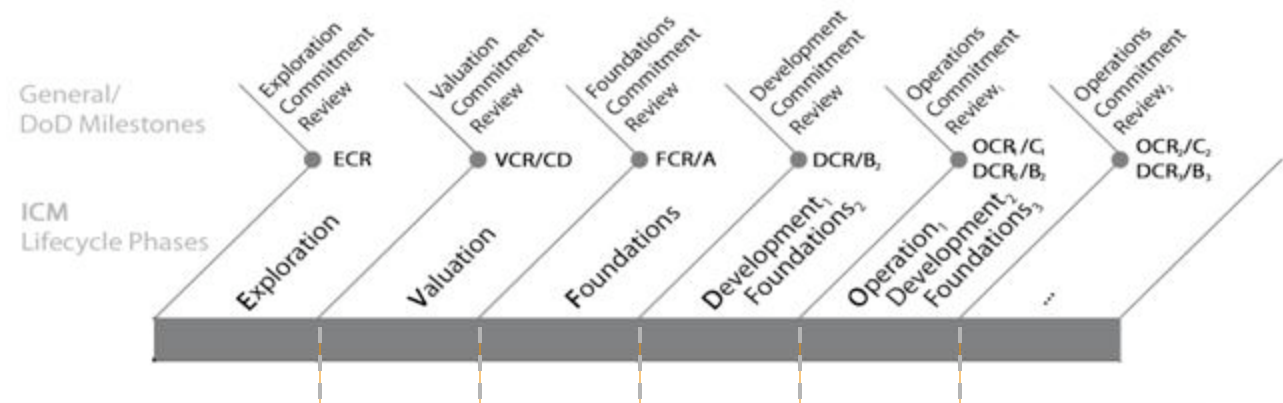


ICSM for 24-week e-services projects



CCD-Core Capability Drivethrough; DCR- Development Commitment Review; ECR-Evaluation Commitment Review; FCR-Foundations Commitment Review; OCR-Operational Commitment Review; RDCR-Rebaselined Development Commitment Review; TRR-Transition Readiness Review; VCR-Valuation Commitment Review

Different Risks/Opportunities Yield Different Processes














Although the patterns look similar,
 NDI and services have different risks

With addressable risk(s), the project moves on the next phase

With provided architecture and functionalities from NDI,
 the team could spend close to no effort in

The team spends more effort in assessing NDI(s) and their
 interoperability, enter Operation phase sooner

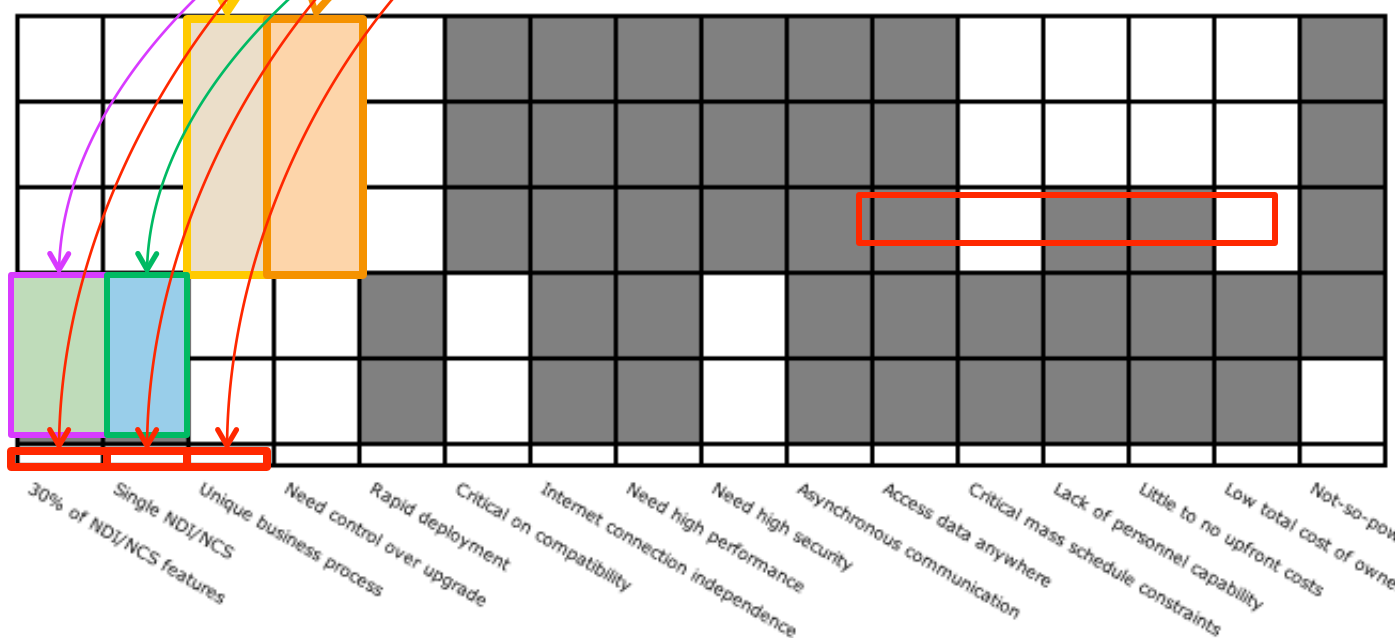
ICSM Process Patterns

Process Pattern	Example
<p>Architected Agile</p>  	<p><i>Business data processing</i></p> 
<p>Use Single NDI</p>  <p>OR</p> 	<p><i>Small website</i></p>
<p>NDI- intensive</p> <p>at least 30% *</p>  <p>AND/OR</p> 	<p><i>Supply chain management</i></p> <p>+ at most 70% CUSTOM CODE</p>
<p>Services- Intensive</p> <p>at least 30% *</p>   <p>AND/OR</p>  	<p><i>Community Services</i></p> <p>+ at most 70% CUSTOM CODE</p>

Process Pattern Decision Driver

Questions	Importance	Architect ed Agile	Use Single NDI	NDI- Intensive	Services- Intensive
Alternatives					
How likely that more than 30% of features are available in NDI/NCS ?		0 – 1	2 – 3	3 – 4	3 – 4
How likely that there is a single NDI/NCS that satisfies a complete solution ?		0 – 1	4	2 – 3	2 – 3
How unique/ inflexible business process your project is?		2 – 4	0 – 1	0 – 1	0 – 1
Life Cycle					
How likely that the system needs control over upgrade / maintenance ?		2 – 4	0 – 1	0 – 1	0 – 1
How fast do you need your pro					
Architecture					
How critical on compatibility y					
How likely that the system will					
How likely that the system will					
How likely that the system v					
How likely that the system w					
How likely that the system w					
Resources					
How critical on mass schedule					
How likely that your organizati					
How likely that your project r					
How likely that your project r					
How likely that your system					

Architected Agile Process Pattern



Note: Development team d
Decision Criteria Rating S
Importance Rating Scale: 1=Low, 2=medium, 3=high

Online tool is available at <http://greenbay.usc.edu/KoolmanDG/index.php>

An Example of a team that follows the Architected Agile Process Pattern :

Shields For Family Project

– Develop various reports for LA city-based Family Housing Project

Questions	Importance	Project Status
Alternatives	Alternatives	Alternatives
How likely that more than 30% of features are available in NDI/NCS ?	1	1
How likely that there is a single NDI/NCS that satisfies a complete solution ?	1	0
How unique/ inflexible business process your project is?	1	2
Life Cycle	Life Cycle	Life Cycle
How likely that the system needs control over upgrade / maintenance ?	3	4
How fast do you need your project to be up and running or go to market?	1	1
Architecture	Architecture	Architecture
How critical on compatibility your project is?	1	2
How likely that the system will be processed offline ?	1	2
How likely that the system will need high level of services / performance ?	3	1
How likely that the system will need high security ?	1	2
How likely that the system will require asynchronous communication?	1	3
How likely that the system will be accessed from anywhere ?	1	3
Resources	Resources	Resources
How critical on mass schedule constraints for your project?	1	1
How likely that your organization will be lack of personnel capability ?	1	4
How likely that your project requires little upfront costs (hardware /software)?	1	1
How likely that your project requires low total cost of ownership?	1	1
How likely that your system will be operable on not-so-powerful local machines ?	3	2

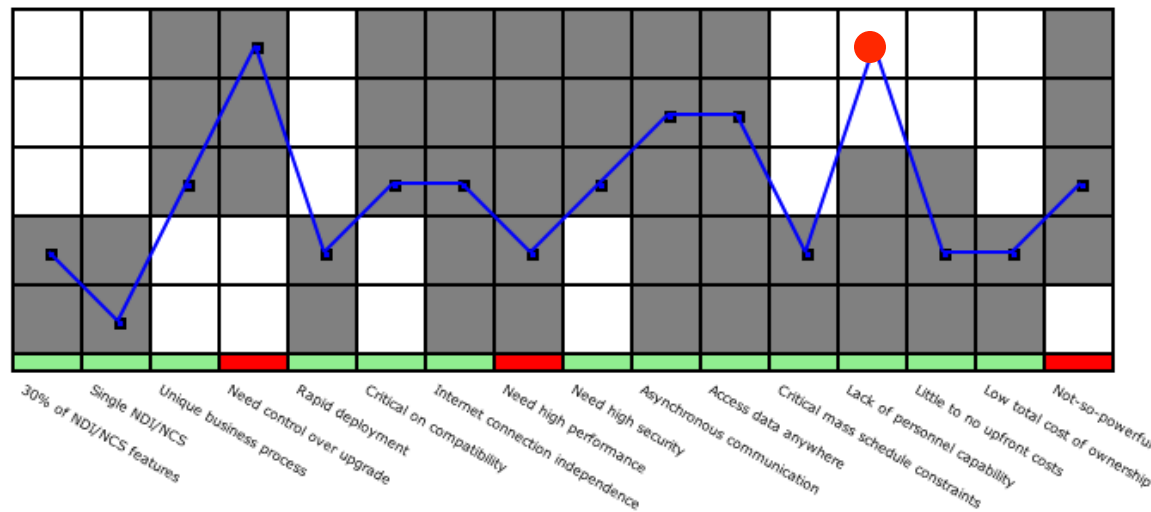
Note: Development team discusses with stakeholders on important drivers and project status

Decision Criteria Rating Scale; 0:Very Low; 1:Low; 2: Medium; 3:High; 4:Very High

Importance Rating Scale: 1:Low; 2: Medium; 3:High

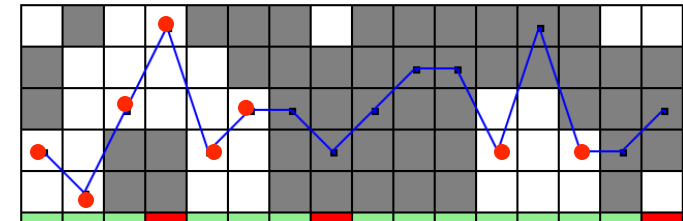
An Example of a team that follows the Architected Agile Process Pattern

Architected Agile

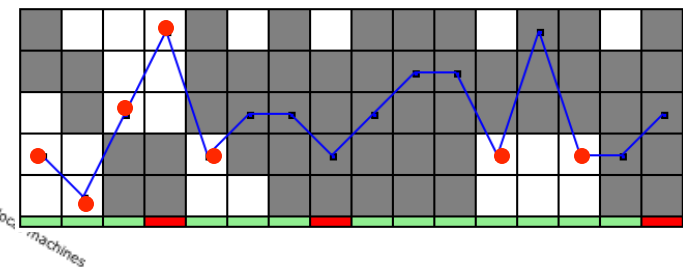


- Non-conforming point
- High importance level
- Low importance level
- Project Status

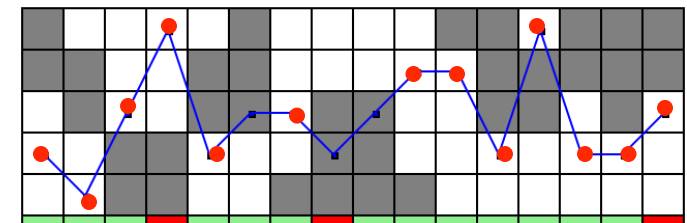
Use single NDI



NDI-Intensive



Services -Intensive



Point Lost from selecting wrong process

- **Teams with no process decision drivers selected wrong processes and loss unnecessary points**

	Without Process Decision Framework			With Process Decision Framework		
	Client Satisfaction (20)	Point Lost	Effort (hour)	Client Satisfaction (20)	Point Lost	Effort (hour)
Average of all teams	17.73	64.95	1522	18.37	76.77	1412
Average of incorrect process selection teams	18	78.24	1652	18.5	78.14	1501

- **Teams spent unnecessary effort on**
 - **System and Software Architecture, Cost Estimation, Requirements**
- **Against the Lean concept, “the right job and doing the job right” [Oppenheim 2010]**

Motivation and Approach

- **SERC RT-5 Study: SE Implications for Evolutionary Acquisition (EvA)**
 - No one-size-fits-all EvA model
 - Pros and Cons of EvA alternatives
 - Decision Table for EvA alternatives
- **Similar Situation for e-Services applications**
 - Several forms; need for decision table
 - Needed extensions to initial decision table
 - Extensions validated in 84-project experiment
- ➔ • **Decision framework extended to other classes of systems**
 - Hardware/Software-intensive; system/family of systems, ...



Applying Process Decision Framework to other patterns

Process Patterns

Software-Intensive Application

Software-Intensive Devices

Platform, Embedded Software

Large, Diversified Software-Intensive Systems

Family of Systems

System of Systems

Enterprise System

Brownfield Modernization

Process Decision Criteria

	Hardware Engineering Needed	Type of Hardware	# of Products in Family	Level of Control over constituent of System of Interest	Continuity of Service; Legacy Migration	# of Systems or Families
SW-Intensive Apps	No	Computer	1	M – H	No	1
SW-Intensive Devices	Yes	Single Device	1	M – H	No	1
Platform, Embedded SW	Yes	Single Platform	1	M – H	No	1
Large, Diversified SIS	Yes	Multi Device & Platform	1	M – H		1
Family of Systems			> 1	H		1
System of Systems				L – M		1
Enterprise System				L - M		> 1
Brownfield Modernization				L - H	Yes	

Conclusions

- **ICSM risk driven framework enables generation of process decision tables in several domains**
- **Wrong e-services process patterns caused poor performance on both unnecessary effort and project results.**
- **These risks and problems could have been mitigated by using process decision criteria to select the appropriate process common case**
- **Approach extended to cover additional common DoD process patterns**