Practical Software and Systems Measurement

Objective Information for Decision Makers

Expediting Systems Engineering within an SoS - Outbrief

Friday, 3 August 2012

Jo Ann Lane, USC
Supannika Koolmanojwong, USC

Expediting Systems Engineering within an SoS

- **Many systems belong to one or more SoS**

- **Approaches often driven by schedule as an independent variable**

- **Workshop attempted to explore**

  - **Approaches for expediting new capabilities within an SoS**

  - **Associated technical debt for approaches**

  - **Ways to quantify technical debt within an SoS**

Workshop Participants

- **Facilitators**
  - Sue Koolmanojwong
  - Jo Ann Lane
- **Participants**
  - Mauricio Aguiar
  - Diana Baklizky
  - Francine Bis
  - Paul Kohl
  - Greg Niemann

Summary

- **Workshop accomplishments**
  - *Identified*
    - *"Expediting" enablers and inhibitors for*
      - *Single system (new)*
      - *Single system (existing)*
      - *System of systems (SoS)*
    - *Estimated impact of each enabler/inhibitor on schedule*
      - *High, medium, low, none*

Top Inhibitors
for expediting systems engineering

#	Single System: New	Single System: Existing	SoS
	29	35	35
1	Requirements Volatility	Requirements Volatility	Lack of Interoperability
2	Unprecedentedness	High # of external interfaces	Lack of / incompatible standard & protocol
3	Delayed authority to proceed with fixed milestone	Unprecedentedness	Requirements Volatility
4	Infeasible schedule/staffing profile	Vague Requirements	Unprecedentedness
5	Lack of Domain Experience	Embedded poor quality software	High # of external interfaces
6	Technology Volatility	Conflicting Stakeholders	Infeasible schedule/staffing profile
7	High # of external interfaces	Delayed authority to proceed with fixed milestone	Inability to test across systems
8	Vague Requirements	Infeasible schedule/staffing profile	Delayed authority to proceed with fixed milestone
9		Technical debt	

Top Enablers
for expediting systems engineering

	Single System: New	Single System: Existing	SoS
#	26	28	30
1	Rapid Prototyping	Target / test / simulation lab	Customer /Tech requirements flexibility
2	Target / test / simulation lab	Incremental test and feedback	Rapid Prototyping
3	Customer /tech requirements flexibility	Incremental Delivery & feedback	Target / test / simulation lab
4	Incremental test and feedback	flexible / tailorable rules	Incremental test and feedback
5	Incremental Delivery & feedback	Agile/lean approach	Common standard and protocol
6	Decision making authority	Rapid Prototyping	Reusing assets
7	Best people		Tools and automation
8	Agile/lean approach		Common standard, interface
9	Less context switching when doing multiple projects		

Next Steps/Action Items

- **Consolidate list of enablers and inhibitors**
  - *Considerable overlap*
- **Continue**
  - *Identification of enablers and inhibitors*
  - *Solicit additional evaluations of enablers and inhibitors*
- **Evaluate related technical debt issues**
- **Map enablers and inhibitors to cost model parameters**
- **Refine cost models using**
  - *Enablers/inhibitors*
  - *Technical debt information*

Backup Charts

Observations

- ***Considerable overlap of the enablers and inhibitors across new system/existing system/SoS***

  - *But the impact ratings tended to differ*

- ***Many inhibitors are the "flip side" of a corresponding enabler***

Conclusions, Recommendations, and Results

Single System New	Number Identified	Estimated Impact Levels (5 assessments)		
		High	Medium	Low
Enablers	26	37	55	35
Inhibitors	29	29	70	42

Single System Existing	Number Identified	Estimated Impact Levels (5 assessments)		
		High	Medium	Low
Enablers	28	30	54	48
Inhibitors	35	32	89	46

SoS	Number Identified	Estimated Impact Levels (5 assessments)		
		High	Medium	Low
Enablers	30	38	61	43
Inhibitors	35	49	84	37

Scope of enablers and inhibitors: People, process, tools, product