Architecting Systems of Systems with Ilities: an Overview of the SAI Method

Nicola Ricci, Matthew E. Fitzgerald, Adam M. Ross, and Donna H. Rhodes

Massachusetts Institute of Technology

March 21-22, 2014

© 2014 Massachusetts Institute of Technology

Overview

- Motivation
- SAI Method
  - Step-by-Step Description
  - Associated MarSec Examples
- Summary

© 2014 Massachusetts Institute of Technology

Motivation

Systems Engineers practicing environment has undergone a significant **metamorphosis**

- *Advent of the internet* → great increase in amount of resources available
- *Information travels at the speed of light* → instantaneous communication
- *High-speed computation* → Performance of very complex analyses

Systems are subject to highly **dynamic operational environments**

- A multitude of **exogenous uncertainties** can impact a system
  - *Geo-political shifts* (e.g., policy/regulation changes)
  - *Disruptive technologies* (e.g., advent of GPS)
  - *Market variations* (e.g., price &demand variations)
- Unanticipated **shifts in** stakeholder **needs**
  - Change of *preferences*
  - Change of *mission objectives*
- **Systems of Systems**
  - *Managerial and operational independence*
  - *Continually evolving*
  - *Emergent behaviors*

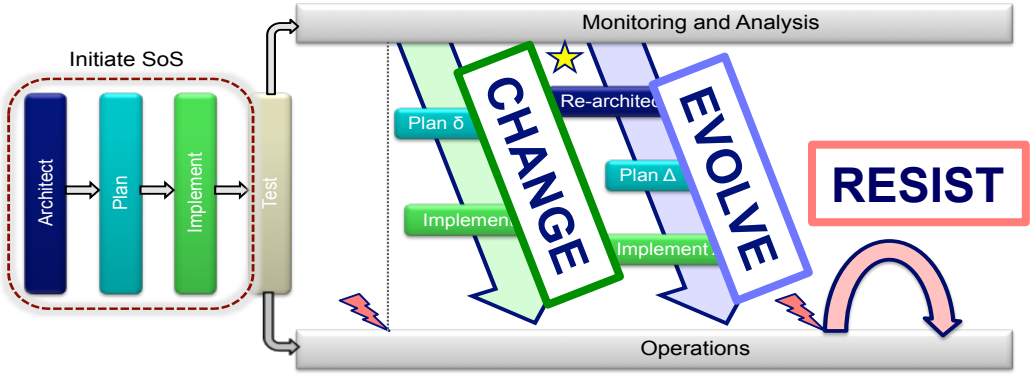
© 2014 Massachusetts Institute of Technology

Ilities

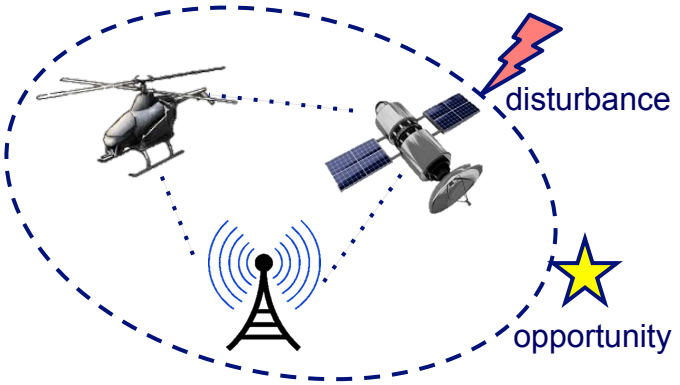
- Modern **ilities** (e.g. flexibility) are one response to mitigate the impact of dynamic complexities on system value over time
- The SAI method builds upon the MIT **Responsive Systems Comparison** method

> *"Properties of engineering systems that often manifest and determine value **after a system is put into initial use**. Rather than being primary functional requirements, these properties concern wider impacts **with respect to time and stakeholders**."*

(de Weck, Ross, and Rhodes – 2012)

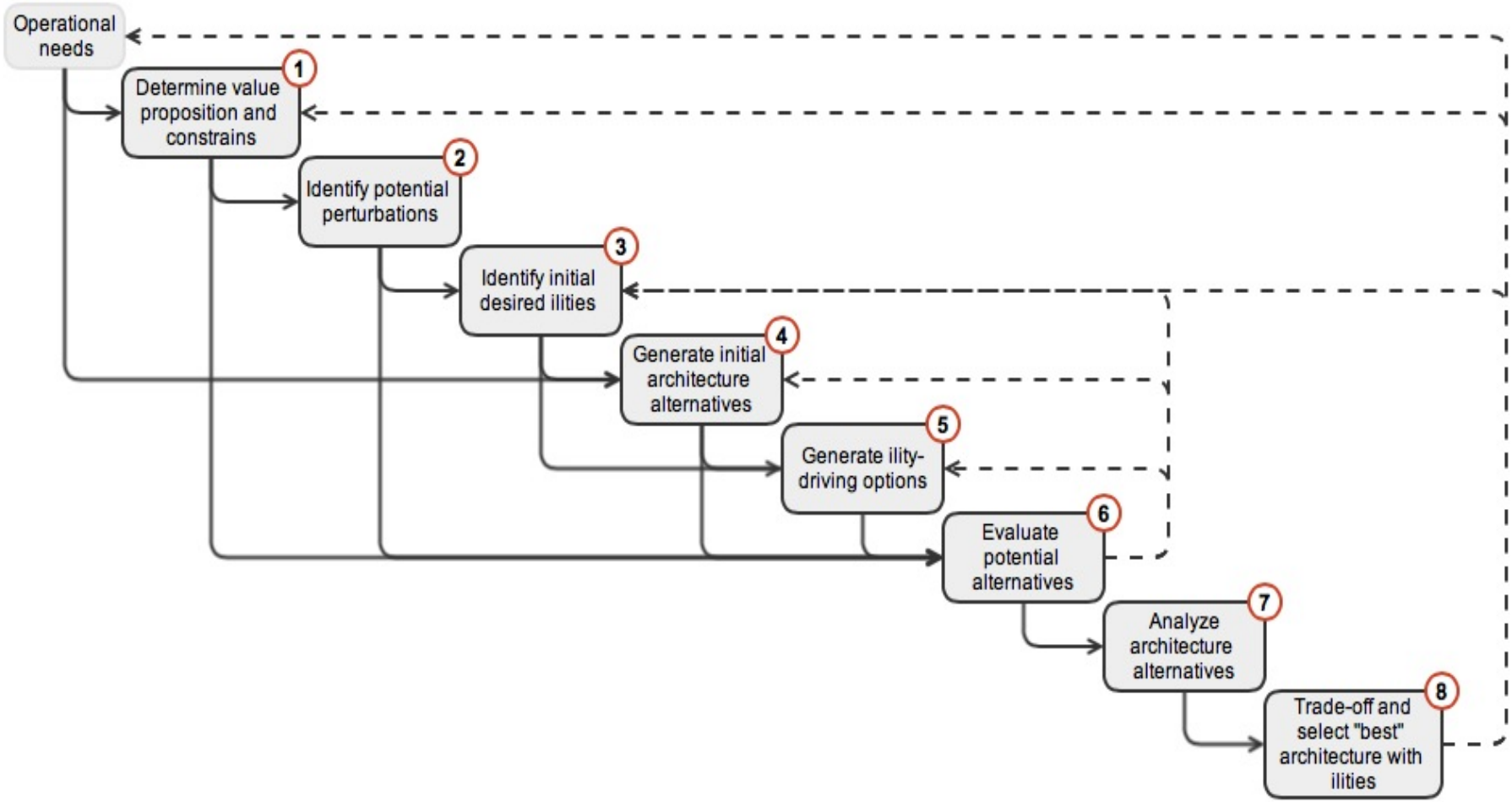


(Ricci, Ross, and Rhodes – 2013)



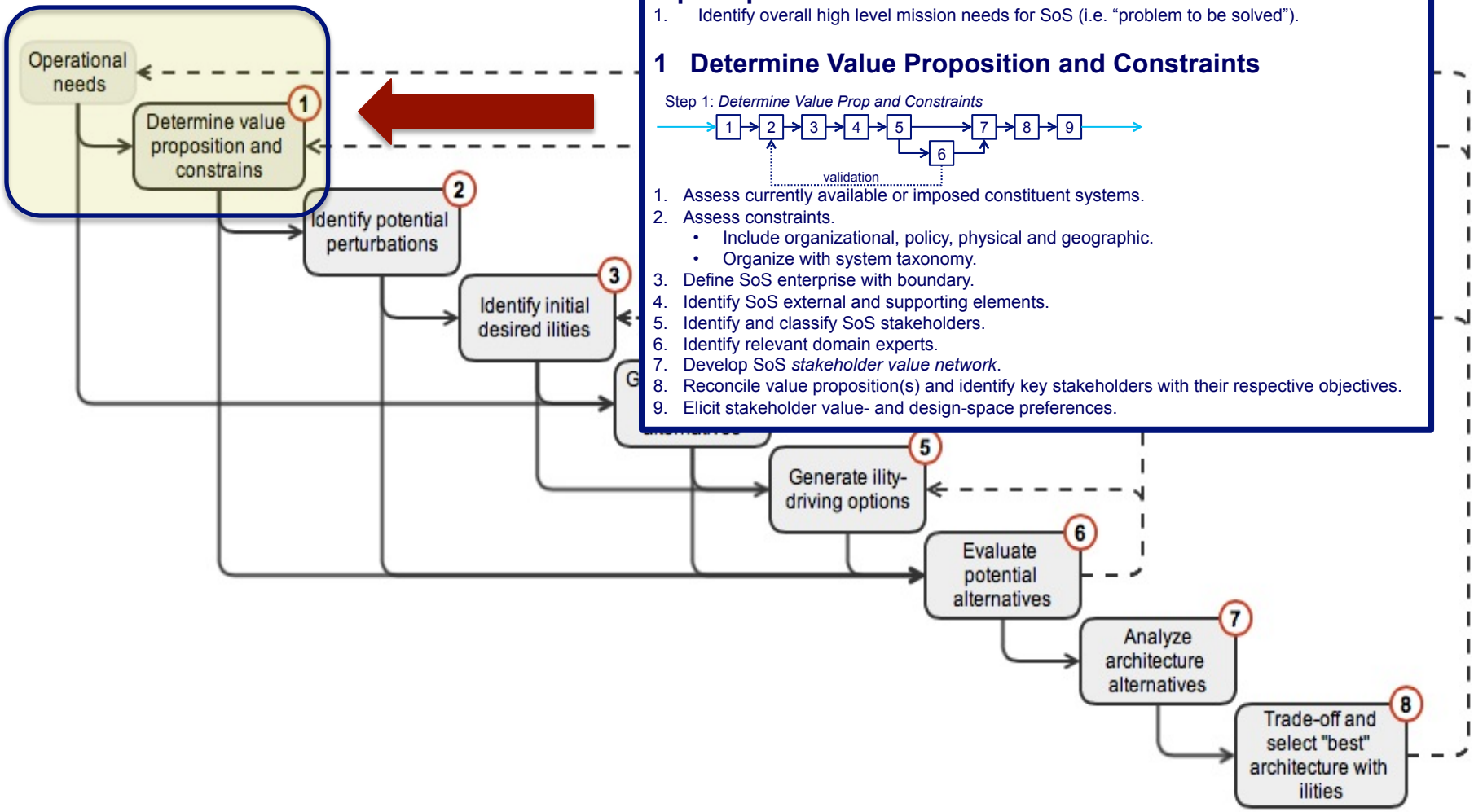
© 2014 Massachusetts Institute of Technology

SoS Architecting with Ilities (SAI) Method



© 2014 Massachusetts Institute of Technology

Input & Step 1



Input Operational Needs Statement

1. Identify overall high level mission needs for SoS (i.e. "problem to be solved").

1 Determine Value Proposition and Constraints

Step 1: *Determine Value Prop and Constraints*

1. Assess currently available or imposed constituent systems.
2. Assess constraints.
   - Include organizational, policy, physical and geographic.
   - Organize with system taxonomy.
3. Define SoS enterprise with boundary.
4. Identify SoS external and supporting elements.
5. Identify and classify SoS stakeholders.
6. Identify relevant domain experts.
7. Develop SoS *stakeholder value network*.
8. Reconcile value proposition(s) and identify key stakeholders with their respective objectives.
9. Elicit stakeholder value- and design-space preferences.

© 2014 Massachusetts Institute of Technology

Needs Statement

Input Operational Needs Statement

1. Identify overall high level mission needs for SoS (i.e. "problem to be solved").

1. Determine Value Proposition and Constraints

MarSec SoS

High-level Operational Needs Statement:

Provide maritime security for a particular littoral Area of Interest (AOI)

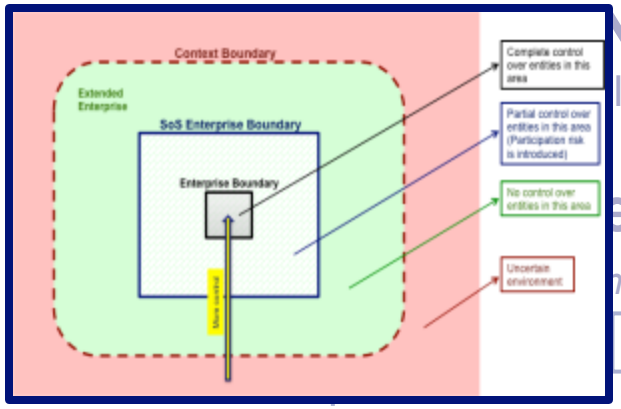
Stakeholders want a *system (SoS)* that:

- **Detects**, **identifies** and **boards** boats entering AOI
- Is capable of carrying out **search and rescue** missions upon request

- Include organizational, policy, physical and geographic
  - Organize with system taxonomy.
3. Define SoS enterprise with boundary.
4. Identify SoS external and supporting elements.
5. Identify and classify SoS stakeholders.
6. Identify relevant domain experts.
7. Develop SoS *stakeholder value network*.
8. Reconcile value proposition(s) and identify key stakeholders with their respective objectives.
9. Elicit stakeholder value- and design-space preferences.

© 2014 Massachusetts Institute of Technology

SoS Enterprise Boundary



Needs Statement

l mission needs for SoS (i.e. "problem to be solved")

e Proposition

nd Constraints

5 → 7 → 8

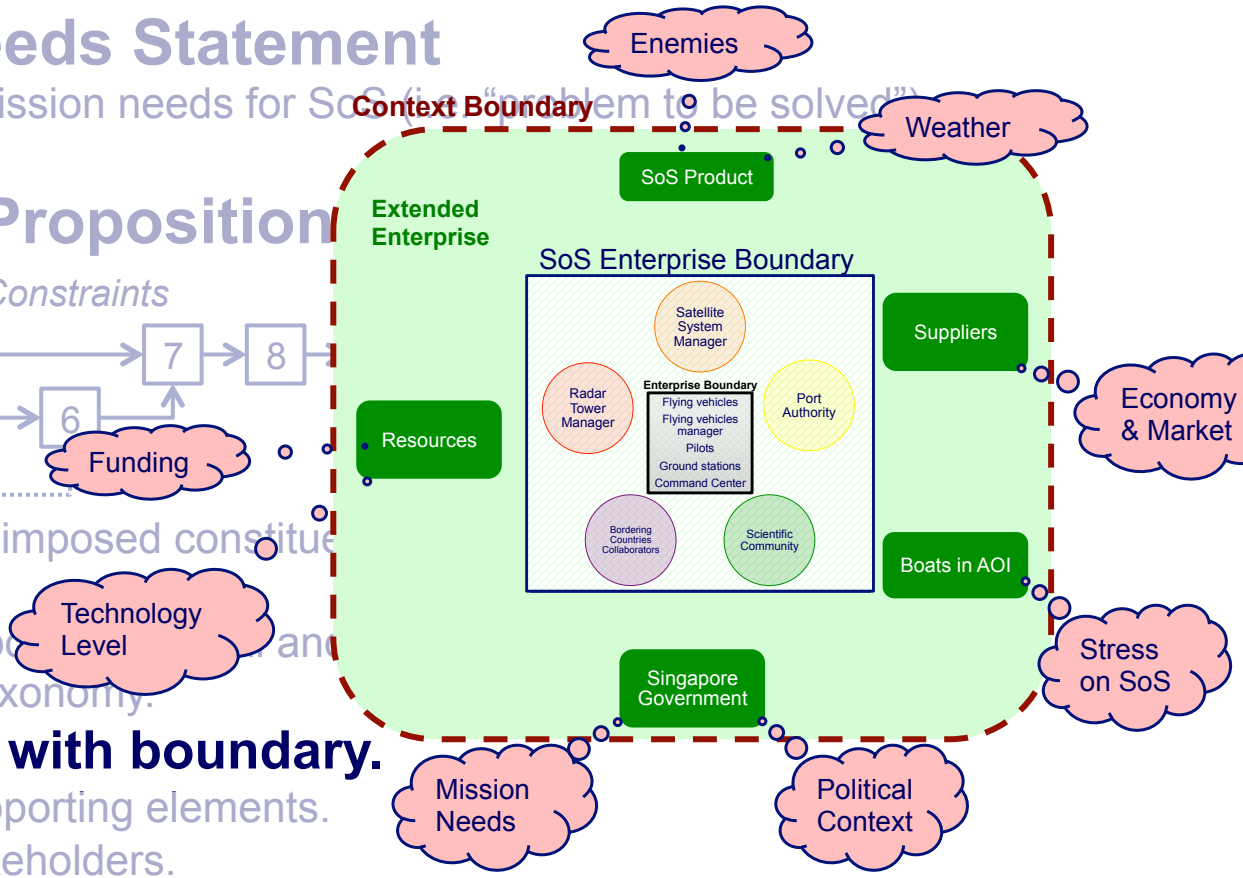
6

validation

1. Assess currently available or imposed constitue
2. Assess constraints.
   - Include organizational, po and
   - Organize with system taxonomy.

3. Define SoS enterprise with boundary.

4. Identify SoS external and supporting elements.
5. Identify and classify SoS stakeholders.
6. Identify relevant domain experts.
7. Develop SoS *stakeholder value network*.
8. Reconcile value proposition(s) and identify key stakeholders with their respective objectives.
9. Elicit stakeholder value- and design-space preferences.



© 2014 Massachusetts Institute of Technology

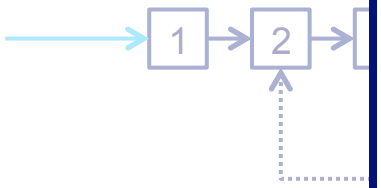
Stakeholder Value Network

Input Operat

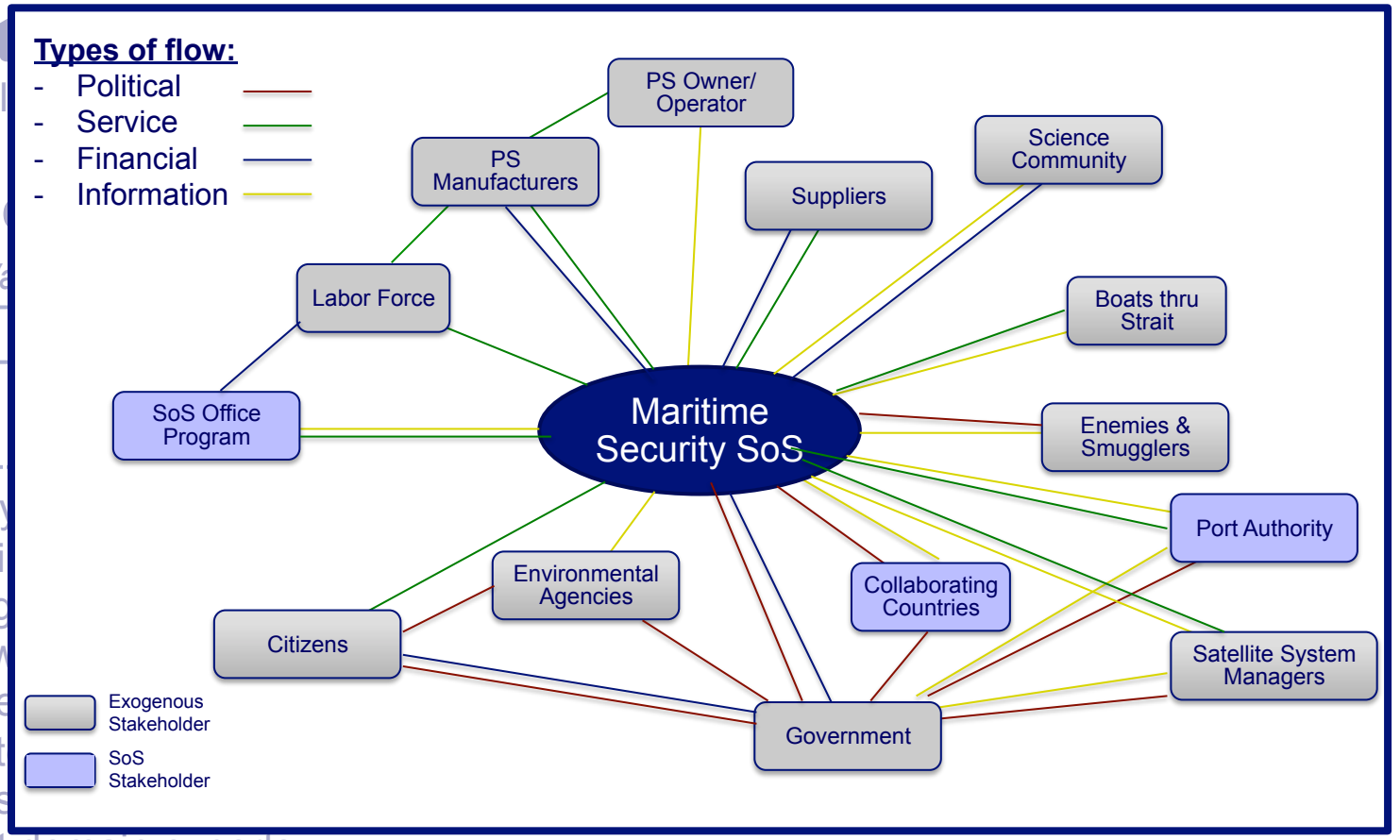
1. Identify overal

1 Determin

Step 1: *Determine Va*



1. Assess currently
2. Assess constrai
   - Include org
   - Organize w
3. Define SoS ente
4. Identify SoS ext
5. Identify and clas
6. Identify relevant domain experts.

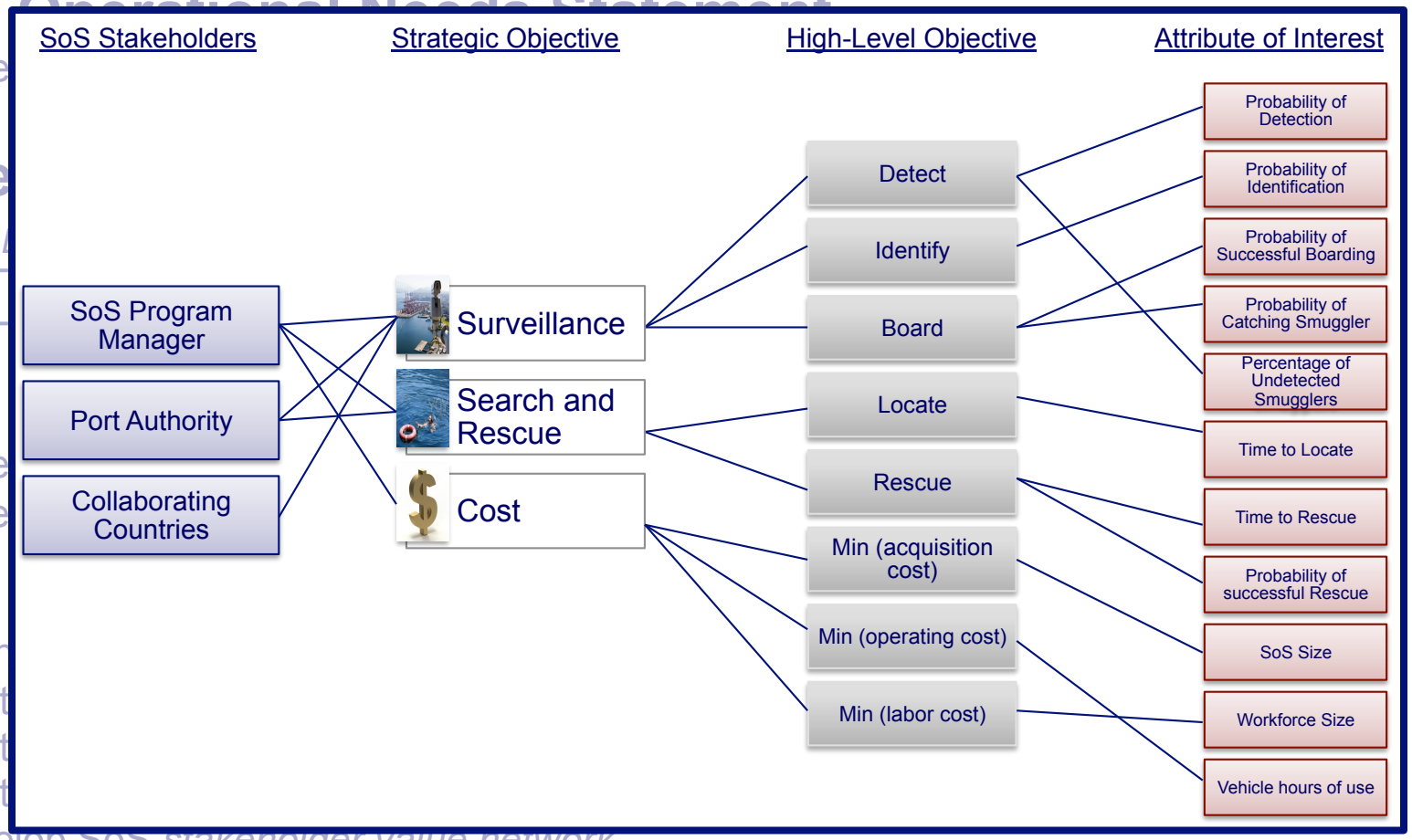


7. Develop SoS *stakeholder value network*.

8. Reconcile value proposition(s) and identify key stakeholders with their respective objectives.
9. Elicit stakeholder value- and design-space preferences.

© 2014 Massachusetts Institute of Technology

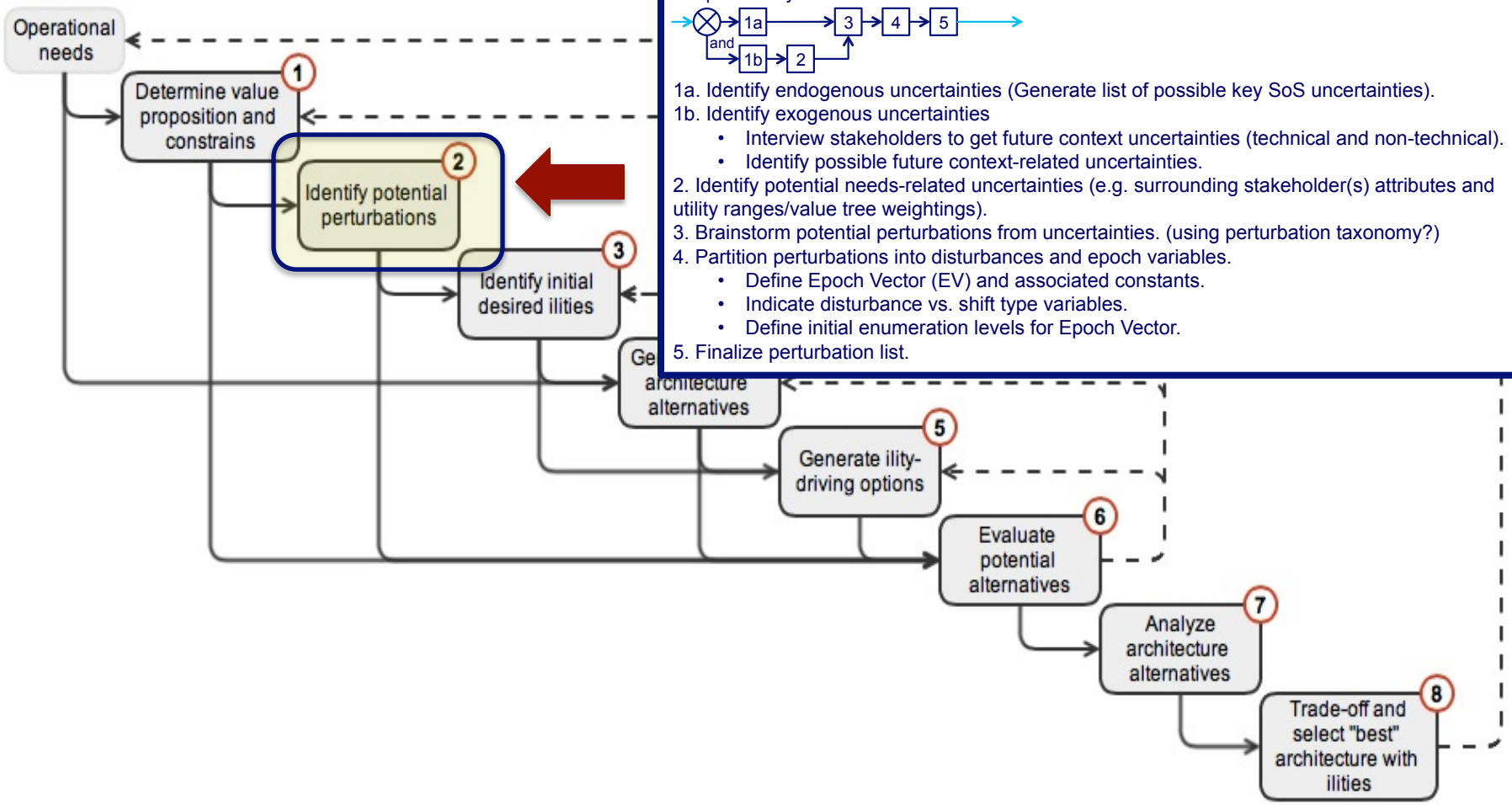
Value Proposition



8. Reconcile value proposition(s) and identify key stakeholders with their respective objectives

© 2014 Massachusetts Institute of Technology

Step 2



2 Identify Potential Perturbations

Step 2: *Identify Potential Perturbations*

1a. Identify endogenous uncertainties (Generate list of possible key SoS uncertainties).
1b. Identify exogenous uncertainties
- Interview stakeholders to get future context uncertainties (technical and non-technical).
- Identify possible future context-related uncertainties.

2. Identify potential needs-related uncertainties (e.g. surrounding stakeholder(s) attributes and utility ranges/value tree weightings).
3. Brainstorm potential perturbations from uncertainties. (using perturbation taxonomy?)
4. Partition perturbations into disturbances and epoch variables.
- Define Epoch Vector (EV) and associated constants.
- Indicate disturbance vs. shift type variables.
- Define initial enumeration levels for Epoch Vector.

5. Finalize perturbation list.

© 2014 Massachusetts Institute of Technology

Brainstorm Perturbations: Uncertainty Characterization

2 Identify Potential

PERTURBATION

"Unintended state change in a system's design, context, or stakeholder needs that could jeopardize value delivery"

1b. Identify exogenous uncertain
- Interview stakeholders t
- Identify possible future

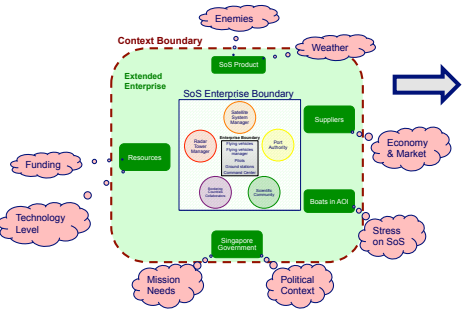
2. Identify potential needs-relate
utility ranges/value tree weightings).

3. Brainstorm potential perturbations from uncertainties.

4. Partition perturbations into disturbances and epoch variables.
- Define Epoch Vector (EV) and associated constants.
- Indicate disturbance vs. shift type variables.
- Define initial enumeration levels for Epoch Vector.

5. Finalize perturbation list.

Enterprise scoping exercise informed the types of "epoch variables" encountered by program

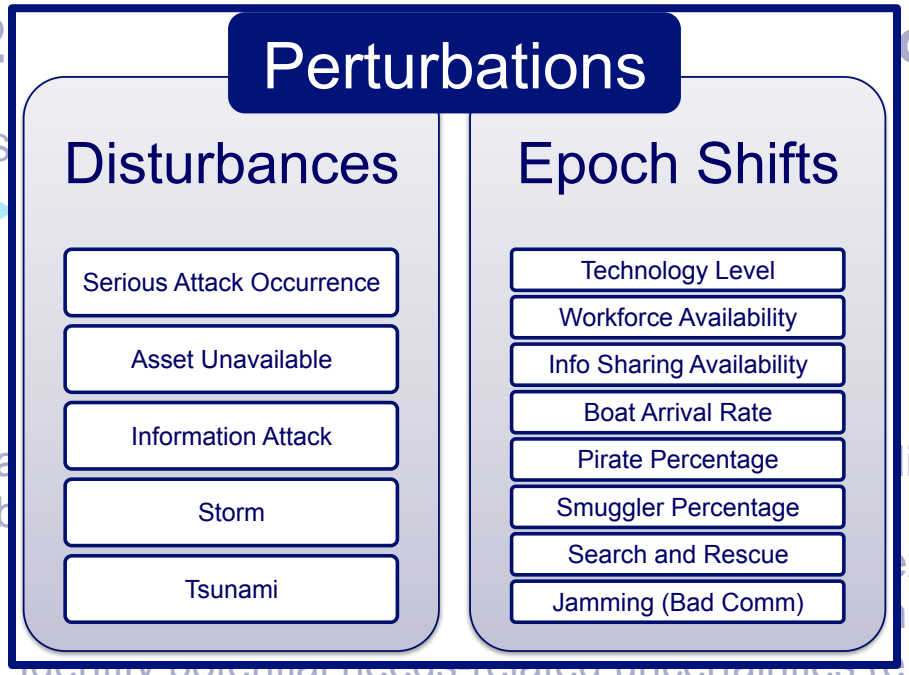


Epoch variables allow for parameterization of some "context" drivers for system value

	Exogenous Uncertainty Category	Possible Factor	Rank w/in Category	Description
Epoch Vector	*Technology Level*	New UAV	1	A new UAV with enhanced capabilities is available
		Detection Methods	3	More reliable detection methods are developed
		Communication	2	Higher gain receivers are on market
	Enemies	Smugglers Volume	1	Illegal activity near the area increases
		Pirate Attacks	2	For some reasons, pirates are more (or less) active
		Terrorists Attacks	3	Possibility of terroristic attacks to boats and/or SoS
	Economy and Market	Alliance with other countries	3	Allows for beneficial deals with other countries
		Goods' price	1	Fuel price increase
		Workforce salary and availability	2	Reduction in workforce salary and size
	Stress on SoS	Communication interruption	2	Lightning strike interrupts communication b/w UAV and ground station
		UAV out of order	3	Pirate attack brings UAV down
		Increased traffic volume	1	Boat arrival rate increases for a specific period
	Mission Needs	Intercept boats	2	Need to take down a dangerous enemy in the AOI
		Search & Rescue	1	Must be able to perform S&R in case of emergency
		Random Search	3	New identification policy
	Funding	Budget cuts on research	2	Will not be able to investigate new technologies
		Budget cuts on Operations	1	Can not pay the current workforce and have to downsize
		No working overtime	3	Can not ask for extra hours in the case of intense activity periods
	Weather	Lightning strike	2	Lightning strike put UAV out of service
		Tsunami	3	Tsunami causes damage to the whole SoS
		Storm	1	Storm reduces visibility and situational awareness
	Political Context	War time	3	The AOI might become a military intense zone
		Conflict with bordering country	2	Might undermine the state of the operating SoS
		Environmental Policy	1	Must fly less UAVs

© 2014 Massachusetts Institute of Technology

Perturbation List: Uncertainty Space



Disturbances

Definition:
"Finite-(short) duration changes of a system design (i.e., forms and operations), needs, or context that could affect value delivery"*

Epoch Shifts

Definition:
Unlikely to revert (e.g. "long" duration) changes in context and/or needs*

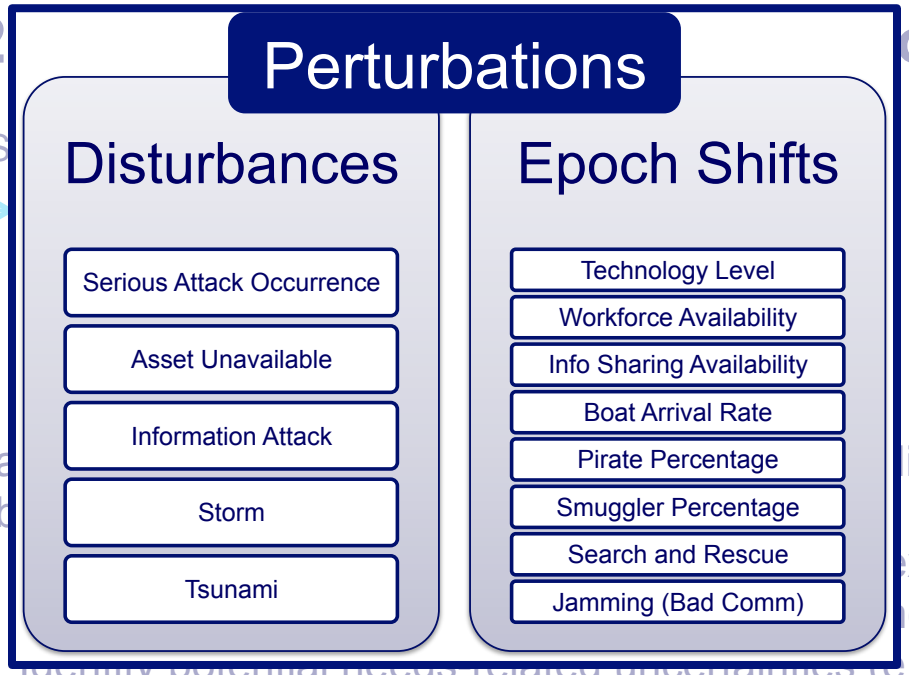
*Beesemyer, J.C., *Empirically Characterizing Evolvability and Changeability in Engineering Systems*, Master of Science Thesis, Aeronautics and Astronautics, MIT, June 2012.

2. Identify potential needs related uncertainties (e.g. surrounding stakeholder(s) attributes and utility ranges/value tree weightings).
3. Brainstorm potential perturbations from uncertainties.
4. Partition perturbations into disturbances and epoch variables.
   - Define Epoch Vector (EV) and associated constants.
   - Indicate disturbance vs. shift type variables.
   - Define initial enumeration levels for Epoch Vector.

5. Finalize perturbation list.

© 2014 Massachusetts Institute of Technology

Perturbation Taxonomy



Disturbances

Definition:
"Finite-(short) duration changes of a system design (i.e., forms and operations), needs, or context that could affect value delivery"*

Epoch Shifts

Definition:
Unlikely to revert (e.g. "long" duration) changes in context and/or needs*

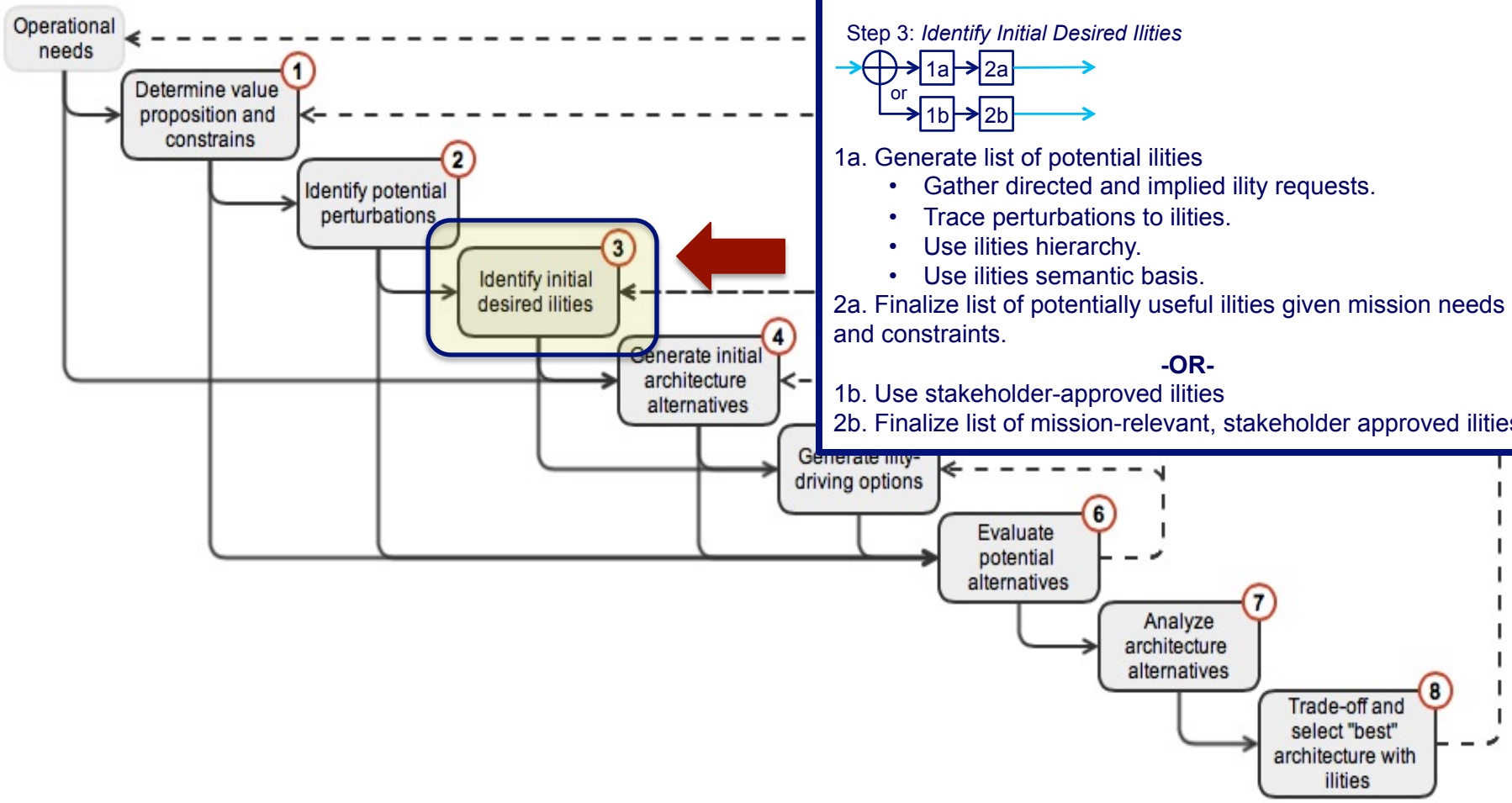
*Beesemyer, J.C., *Empirically Characterizing Evolvability and Changeability in Engineering Systems*, Master of Science Thesis, Aeronautics and Astronautics, MIT, June 2012.

Perturbation	Type	Space	Origin	Intent	Nature	Consequence	Effect
Name	Disruption	Design	Internal	Yes	Natural	Positive	Various
	Disturbance	Context	External	No		Negative	
	Shift	Needs	Either	Either	Artificial	Either	

5. Finalize perturbation list.

© 2014 Massachusetts Institute of Technology

Step 3



3 Identify Initial Desired Ilities

Step 3: *Identify Initial Desired Ilities*

1a. Generate list of potential ilities
- Gather directed and implied ility requests.
- Trace perturbations to ilities.
- Use ilities hierarchy.
- Use ilities semantic basis.

2a. Finalize list of potentially useful ilities given mission needs and constraints.

-OR-

1b. Use stakeholder-approved ilities

2b. Finalize list of mission-relevant, stakeholder approved ilities.

© 2014 Massachusetts Institute of Technology

Desired Ilities

Perturbation	Type	Space	Origin	Intent	Nature	Consequence	Effect
Name	Disruption	Design	Internal	Yes	Natural	Positive	Various
	Disturbance	Context	External	No	Artificial	Negative	
	Shift	Needs	Either	Either		Either	

- Survivability, Changeability, Robustness
- Survivability, Changeability, Versatility
- Reliability vs. Outward-Type Ilities
- Underline the importance of Agility and Reactivity
- Changeability, Versatility, Survivability, robustness

- Gather directed and implied ility requests.
- **Trace perturbations to ilities.**
- Use ilities hierarchy.
- Use ilities semantic basis.

2a. Finalize list of potentially useful ilities given mission needs and constraints.

-OR-

1b. Use stakeholder-approved ilities

2b. Finalize list of mission-relevant, stakeholder approved ilities.

© 2014 Massachusetts Institute of Technology

Desired Ilities

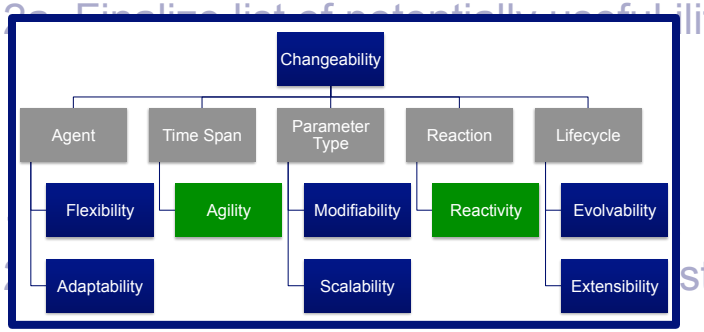
Perturbation	Type	Space	Origin	Intent	Nature	Consequence	Effect
Name	Disruption	Design	Internal	Yes	Natural	Positive	Various
	Disturbance	Context	External	No	Artificial	Negative	
	Shift	Needs	Either	Either		Either	

- Survivability, Changeability, Robustness
- Survivability, Changeability, Versatility
- Reliability vs. Outward-Type Ilities
- Underline the importance of Agility and Reactivity
- Changeability, Versatility, Survivability, robustness

- Trace perturbations to ilities.
- **Use ilities hierarchy.**
- Use ilities semantic basis.

ilities given mission needs and constraints.

-OR-



stakeholder approved ilities.

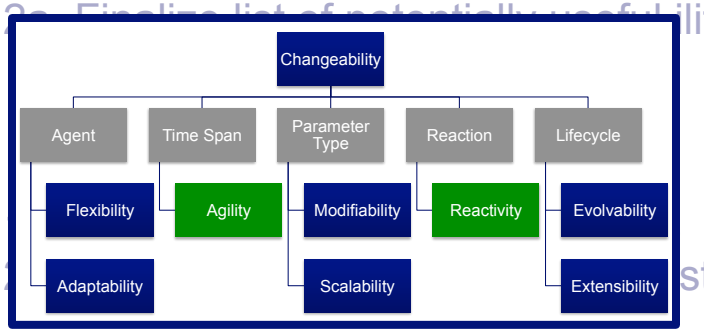
© 2014 Massachusetts Institute of Technology

Desired Ilities

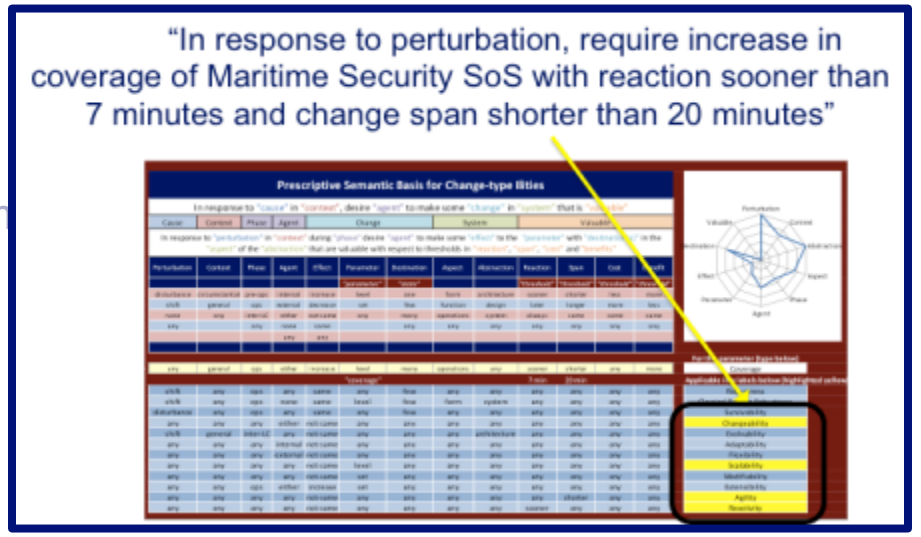
Perturbation	Type	Space	Origin	Intent	Nature	Consequence	Effect
Name	Disruption	Design	Internal	Yes	Natural	Positive	Various
	Disturbance	Context	External	No	Artificial	Negative	
	Shift	Needs	Either	Either		Either	

- Survivability, Changeability, Robustness
- Survivability, Changeability, Versatility
- Reliability vs. Outward-Type Ilities
- Underline the importance of Agility and Reactivity
- Changeability, Versatility, Survivability, robustness

- Trace perturbations to ilities.
- Use ilities hierarchy.
- **Use ilities semantic basis.**

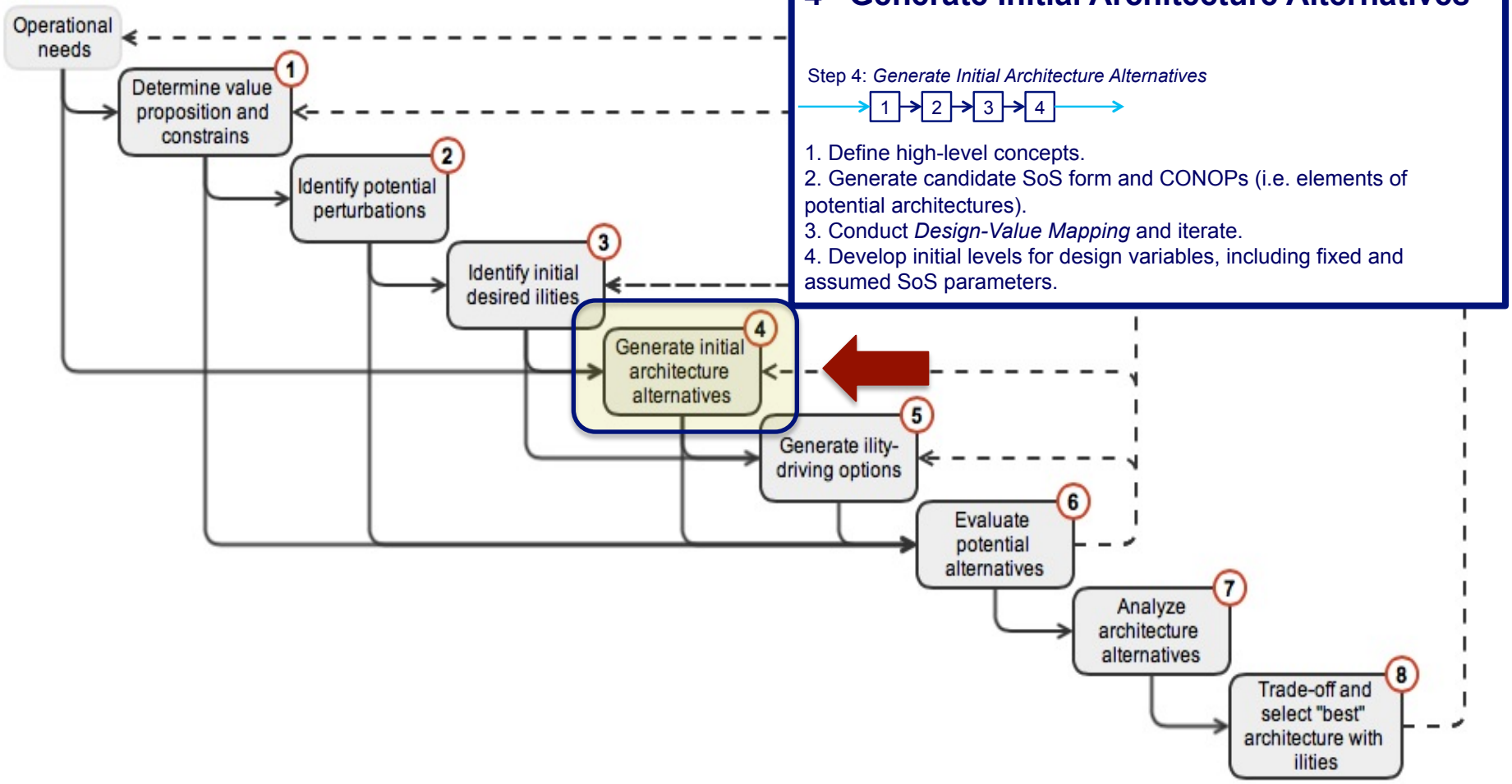


-OR-



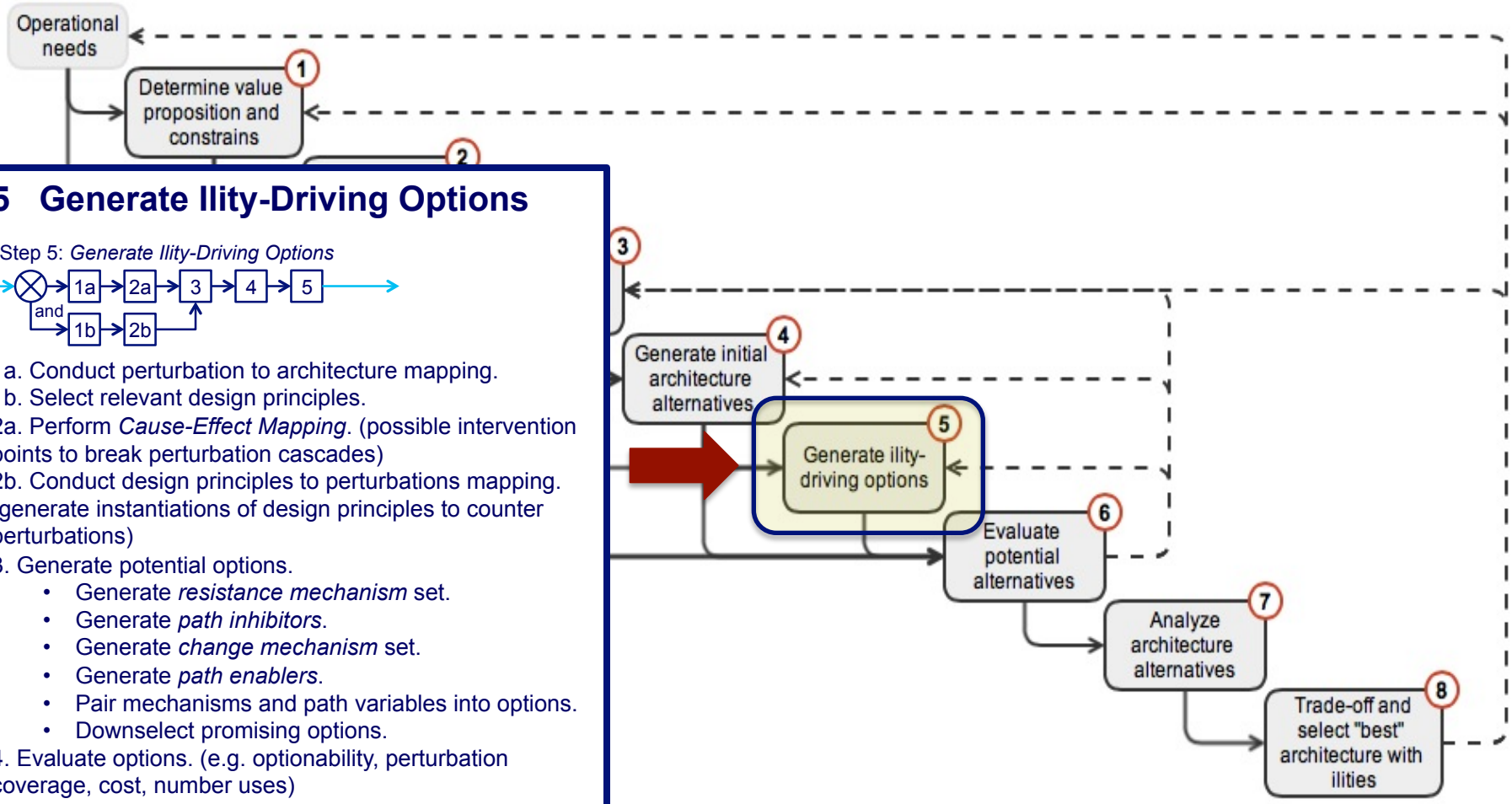
© 2014 Massachusetts Institute of Technology

Step 4



© 2014 Massachusetts Institute of Technology

Step 5



5 Generate Ility-Driving Options

Step 5: *Generate Ility-Driving Options*

1a. Conduct perturbation to architecture mapping.
1b. Select relevant design principles.
2a. Perform *Cause-Effect Mapping*. (possible intervention points to break perturbation cascades)
2b. Conduct design principles to perturbations mapping. (generate instantiations of design principles to counter perturbations)
3. Generate potential options.
- Generate *resistance mechanism* set.
- Generate *path inhibitors*.
- Generate *change mechanism* set.
- Generate *path enablers*.
- Pair mechanisms and path variables into options.
- Downselect promising options.

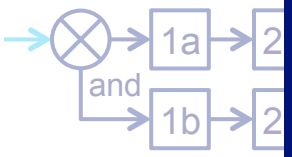
4. Evaluate options. (e.g. optionability, perturbation coverage, cost, number uses)
5. Finalize list of options.

© 2014 Massachusetts Institute of Technology

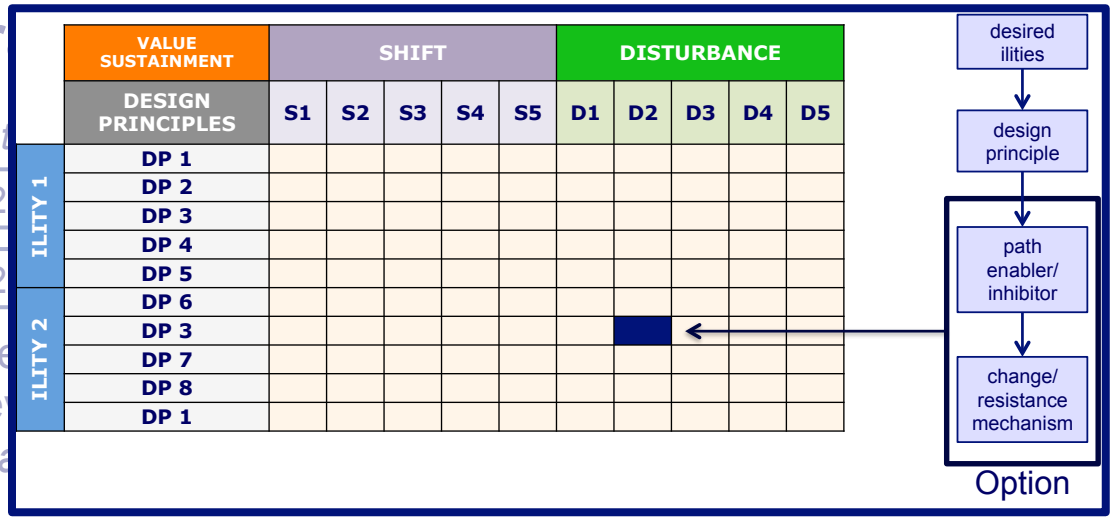
Design Principle to Perturbation Mapping

5 Gener

Step 5: *Generat*

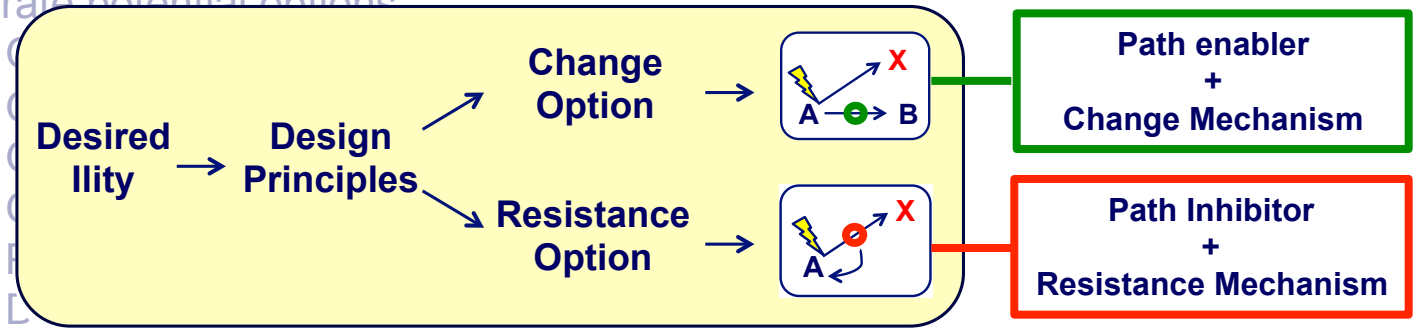


1a. Conduct pe
1b. Select relev
2a. Perform Ca perturbation cascades)



2b. Conduct design principles to perturbations mapping. (generate instantiations of design principles to counter perturbations)

3. Generate potential options



4. Evaluate options. (e.g. optionability, perturbation coverage, cost, number uses)
5. Finalize list of options.

© 2014 Massachusetts Institute of Technology

Options Generation

5 Generate Ility-Driving Options

	CHANGE OPTION	Path Enabler								Latent Path Enabler		
		Extra Interception UAV	Contract with Aircraft Supplier	Extra Cameras	Sea Planes	Pre-Validation Process	Long Range UAV	Workforce Buffer	Dispersed Com Network	Spares	Multi-Role Asset	Central Authority	Satellite Relay
Change Mechanism	Adding Vehicle	C1	C2	-	-	-	-	C3	-	C4	-	-	-
	Change Task Assignment	C5	-	C6	C7	-	-	-	C8	-	C9	-	-
	Change Geographic Segmentation	-	-	-	C10	-	C11	-	C12	-	-	-	C13
	Change Number of Operators per UAV	-	-	-	-	C14	-	C15	-	C16	-	-	-
	Go back to Pre-Validated Set	-	-	-	-	C17	-	-	-	-	C18	-	-
	Change Authority distribution	-	-	-	-	-	-	-	-	-	-	C19	C20
	Add extra features to Asset	-	C21	C22	-	-	-	-	-	-	-	-	-

cascades)

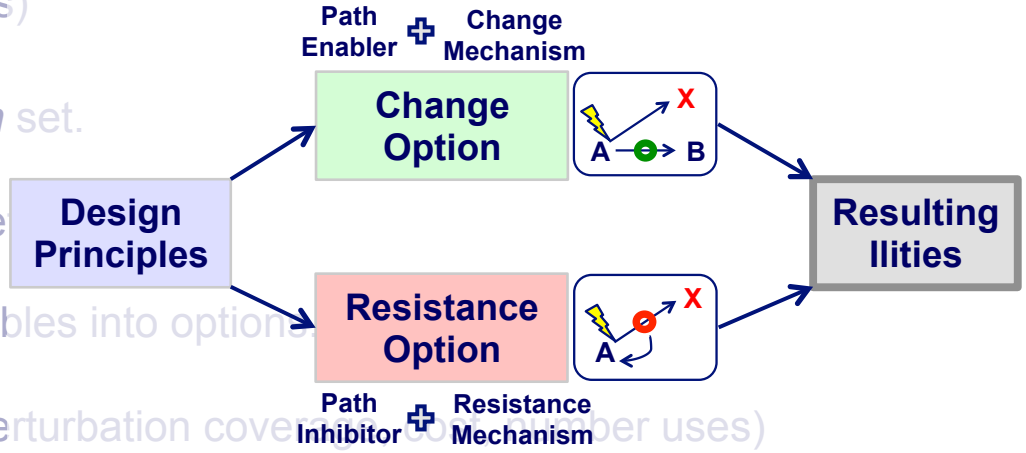
2b. **Conduct design principles to perturbations mapping**. (generate instantiations of design principles to counter perturbations)

3. Generate potential options.

- Generate *resistance mechanism* set.
- Generate *path inhibitors*.
- Generate *change mechanism* se
- Generate *path enablers*.
- Pair mechanisms and path variables into options
- Downselect promising options.

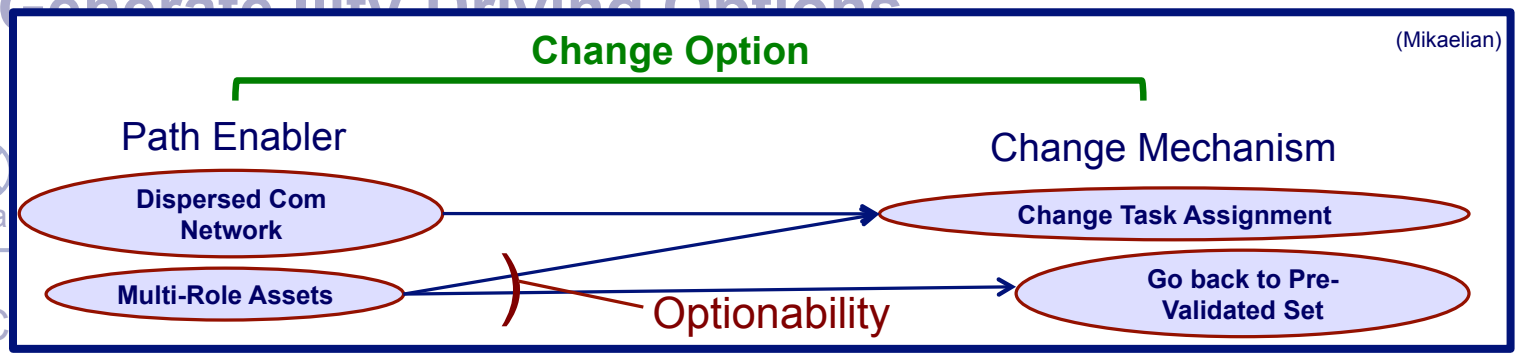
4. Evaluate options. (e.g. optionability, perturbation cove ber uses)

5. Finalize list of options.



© 2014 Massachusetts Institute of Technology

Options Evaluation



For evaluation, four different metrics are considered:

- **OPTIONABILITY**: *the number of options a path enabler/inhibitor is linked to.*
- **NUMBER OF USES**: *how many times can a path enabler/inhibitor be used.*
- **COST**: *the cost of acquiring and using a particular path enabler/inhibitor.*
- **PERTURBATION COVERAGE**: *metric that takes into account impact and probability of perturbations covered by path enabler/inhibitors when paired with change/resistance mechanism*

PERTURBATION			SHIFT			DISTURBANCE			Σ
			S1	S2	S3	D1	D2	D3	
Probability [High-Medium-Low]			low	medium	high	high	low	medium	
Impact [High-Medium-Low]			high	low	medium	low	medium	high	
Change Option Uses [1, N, ∞]	C1	1				1	1	1	3
	C2	∞	1		1	1	1		4
	C3	∞			1	1	1		3
	C4	N			1	1	1	1	4
	C5	1					1		1
Resistance Option Uses [1, N, ∞]	R1	∞		1	1				2
	R2	∞				1	1	1	3
	R3	∞							0
	R4	N				1	1		2
	R5	∞				1			1
Σ			1	1	4	7	7	3	

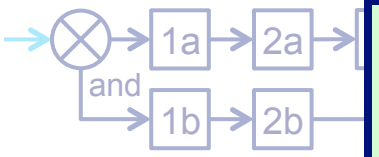
4. Evaluate options. (optionability, perturbation coverage, cost, nu)

© 2014 Massachusetts Institute of Technology

Options Selection

5 Generate Ility-Driving Options

Step 5: *Generate Ility-Driving Options*



1a. Conduct perturb

1b. Select relevant

2a. Perform *Cause* cascades)

2b. **Conduct desig** design principles to

3. Generate poten

- Generate
- Generate
- Generate
- Generate
- Pair mechanisms and path variables into options.
- Downselect promising options.

4. Evaluate options. (e.g. optionability, perturbation coverage, cost, number uses)

5. Finalize list of options.

For evaluation, four different metrics are considered:

- **OPTIONABILITY**: *the number of options a path enabler/inhibitor is linked to.*
- **NUMBER OF USES**: *how many times can a path enabler/inhibitor be used.*
- **COST**: *the cost of acquiring and using a particular path enabler/inhibitor.*
- **PERTURBATION COVERAGE**: *metric that takes into account impact and probability of perturbations covered by path enabler/inhibitors when paired with change/resistance mechanism*

For selection criteria, two tools will be developed:

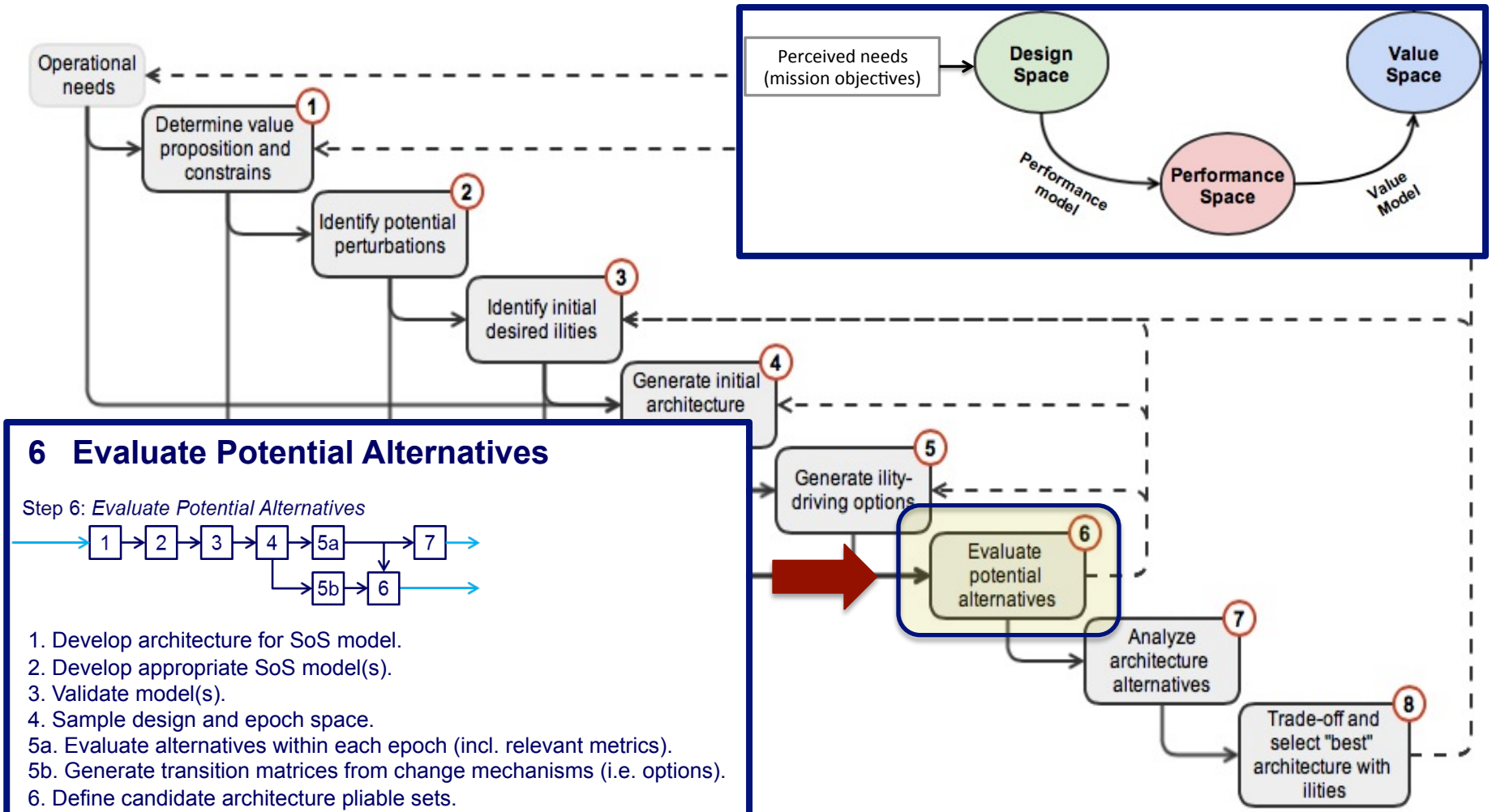
- **VISUALIZATION TOOLS**: *look at different metrics at once.*
- **ANALYTICAL TOOLS**: PC vs. *cost tradespace exploration*

Final list of options to consider for:

1. Direct inclusion into system (if time is a concern)
2. Model and simulation for more detailed analysis and better informed decision

© 2014 Massachusetts Institute of Technology

Step 6



6 Evaluate Potential Alternatives

Step 6: *Evaluate Potential Alternatives*

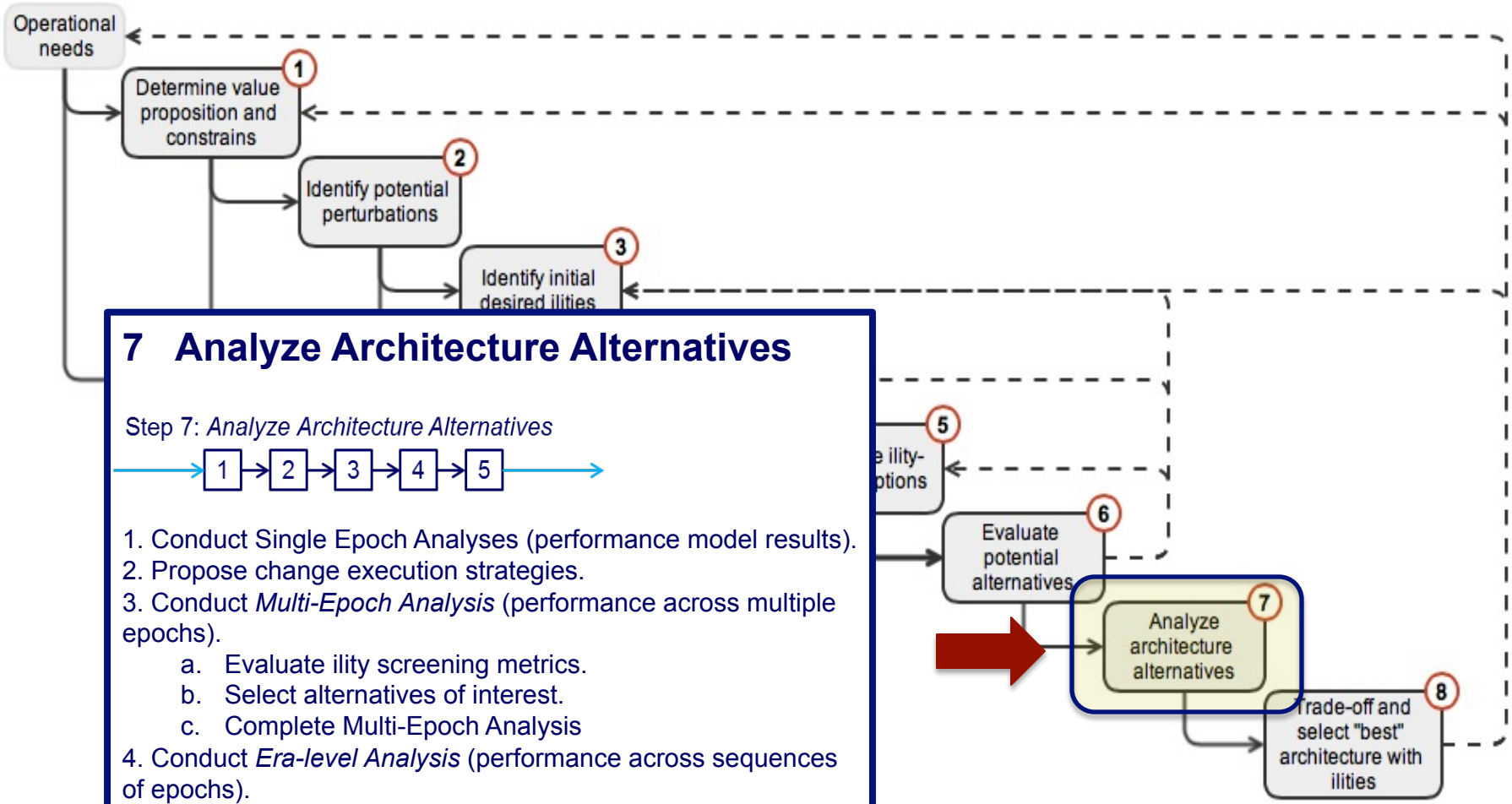
1. Develop architecture for SoS model.
2. Develop appropriate SoS model(s).
3. Validate model(s).
4. Sample design and epoch space.

5a. Evaluate alternatives within each epoch (incl. relevant metrics).
5b. Generate transition matrices from change mechanisms (i.e. options).

6. Define candidate architecture pliable sets.
7. Validate model covers design-value space (*DVM* validation).

© 2014 Massachusetts Institute of Technology

Step 7



7 Analyze Architecture Alternatives

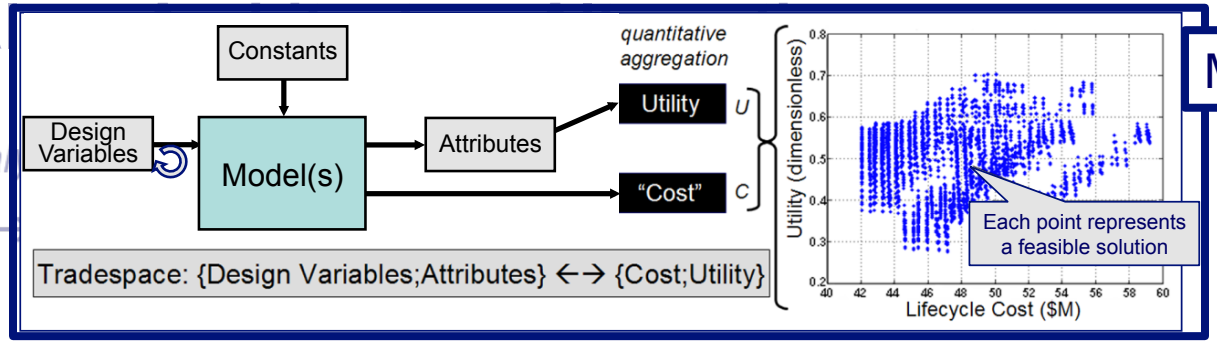
Step 7: *Analyze Architecture Alternatives*

1 → 2 → 3 → 4 → 5

1. Conduct Single Epoch Analyses (performance model results).
2. Propose change execution strategies.
3. Conduct *Multi-Epoch Analysis* (performance across multiple epochs).
    a. Evaluate ility screening metrics.
    b. Select alternatives of interest.
    c. Complete Multi-Epoch Analysis
4. Conduct *Era-level Analysis* (performance across sequences of epochs).
5. Collect set of alternatives of interest with ility metrics.

© 2014 Massachusetts Institute of Technology

Single Epoch Analyses



1. Conduct Single Epoch Analyses (performance model results).

2. Propose change execution strategies.

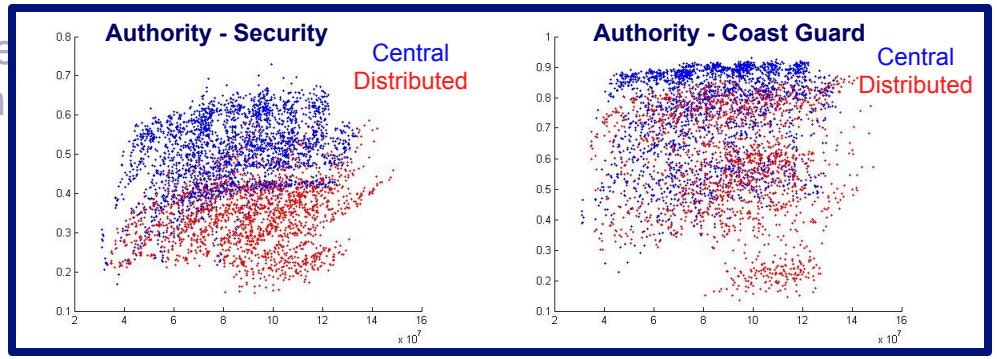
3. Conduct *Multi-Epoch Analysis* (performance across multiple epochs).

a. Evaluate ility screening metrics.

Select alternatives of interest.

EPOCH

Time period with a **fixed context and needs**; characterized by static constraints, concepts, available technologies, and articulated expectations

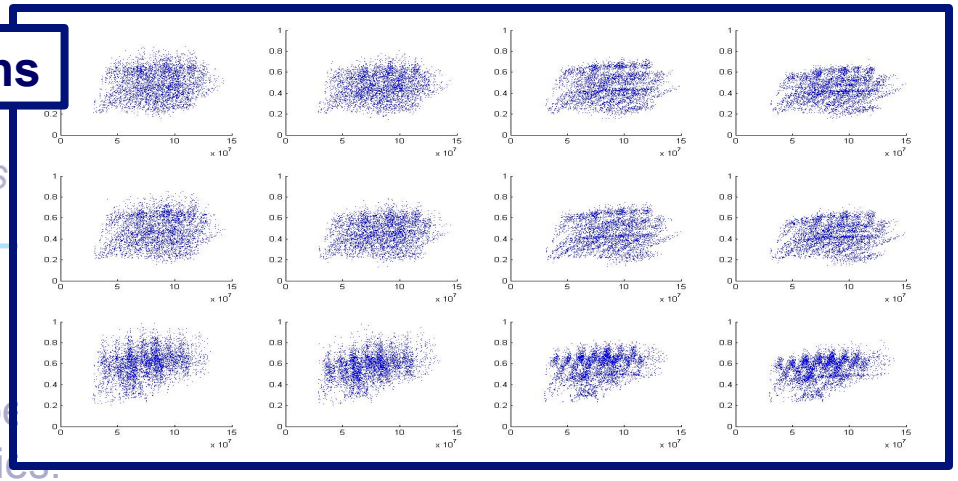


© 2014 Massachusetts Institute of Technology

Multi-Epoch Analysis

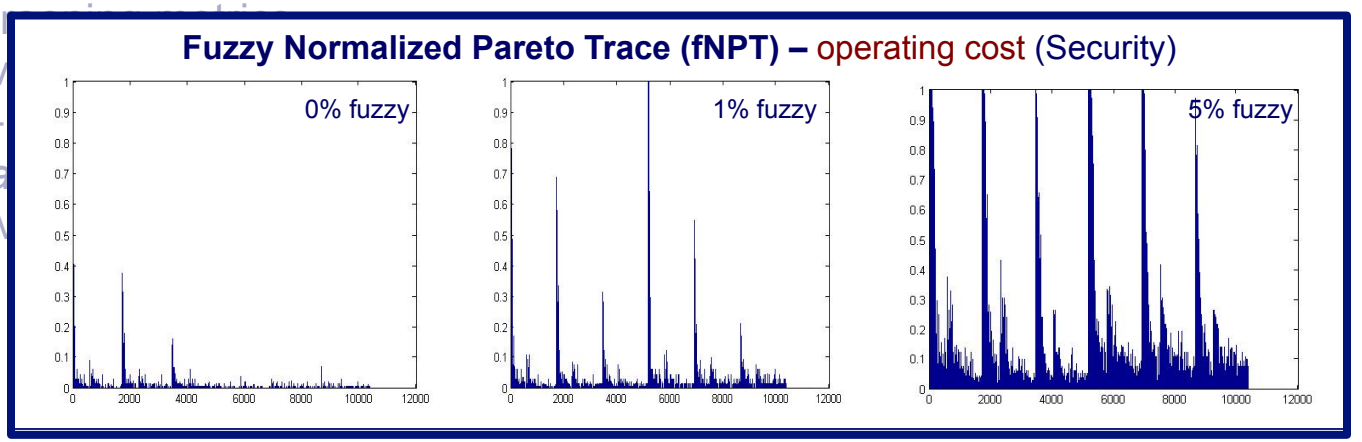
7 Analyz

12 different **epochs**



Step 7: *Analyze Architecture Alternatives*

1. Conduct Single Epoch Analyses (pe
2. Propose change execution strategies.
3. **Conduct *Multi-Epoch Analysis* (performance across multiple epochs).**
   a. Evaluate ility sc
   b. Select alternativ
   c. Complete Multi-
4. Conduct *Era-level Ana*
5. Collect set of alternativ



© 2014 Massachusetts Institute of Technology

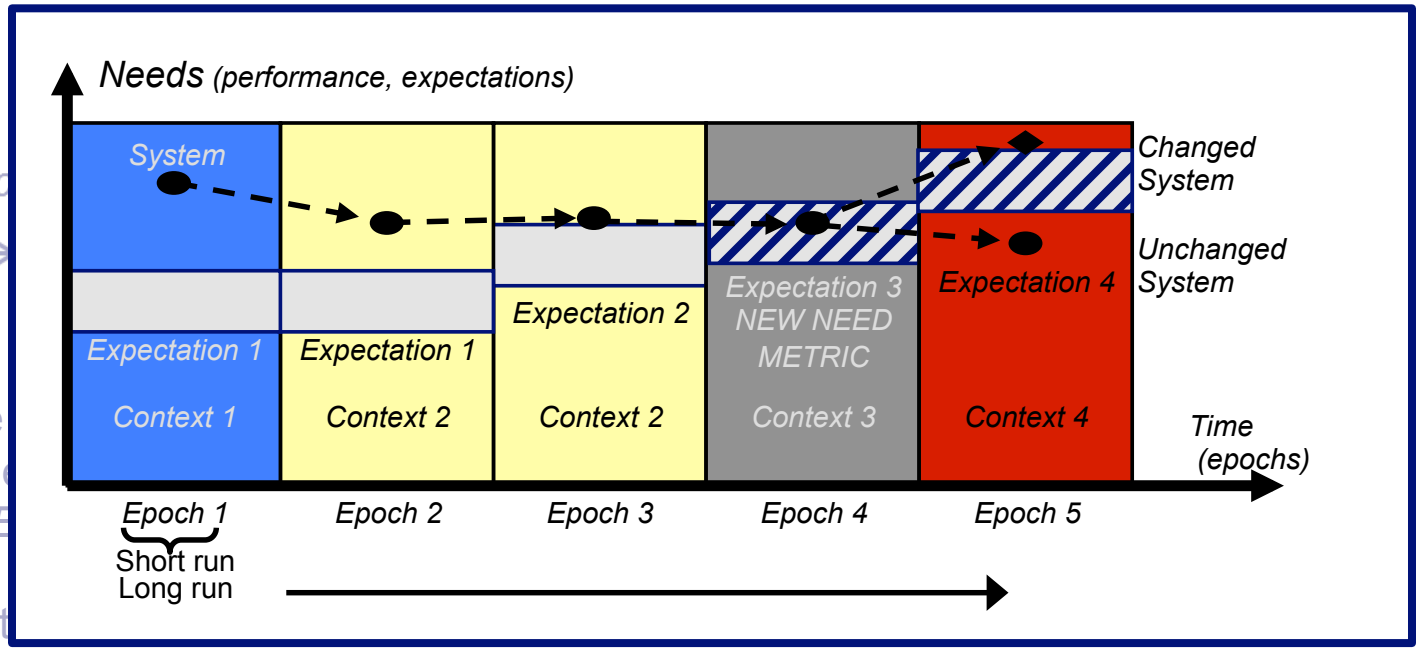
Era Analysis

7 Analyze

Step 7: *Analyze Arc*

1 → 2 →

1. Conduct Single
2. Propose change
3. Conduct *Multi-E*
   a. Evaluate
   b. Select alt
   c. Complete Multi-Epoch Analysis



4. Conduct *Era-level Analysis* (performance across sequences of epochs).

5. Collect set of alternatives of interest with ility metrics.

© 2014 Massachusetts Institute of Technology

Ility Metrics

7 Analyze Architecture A

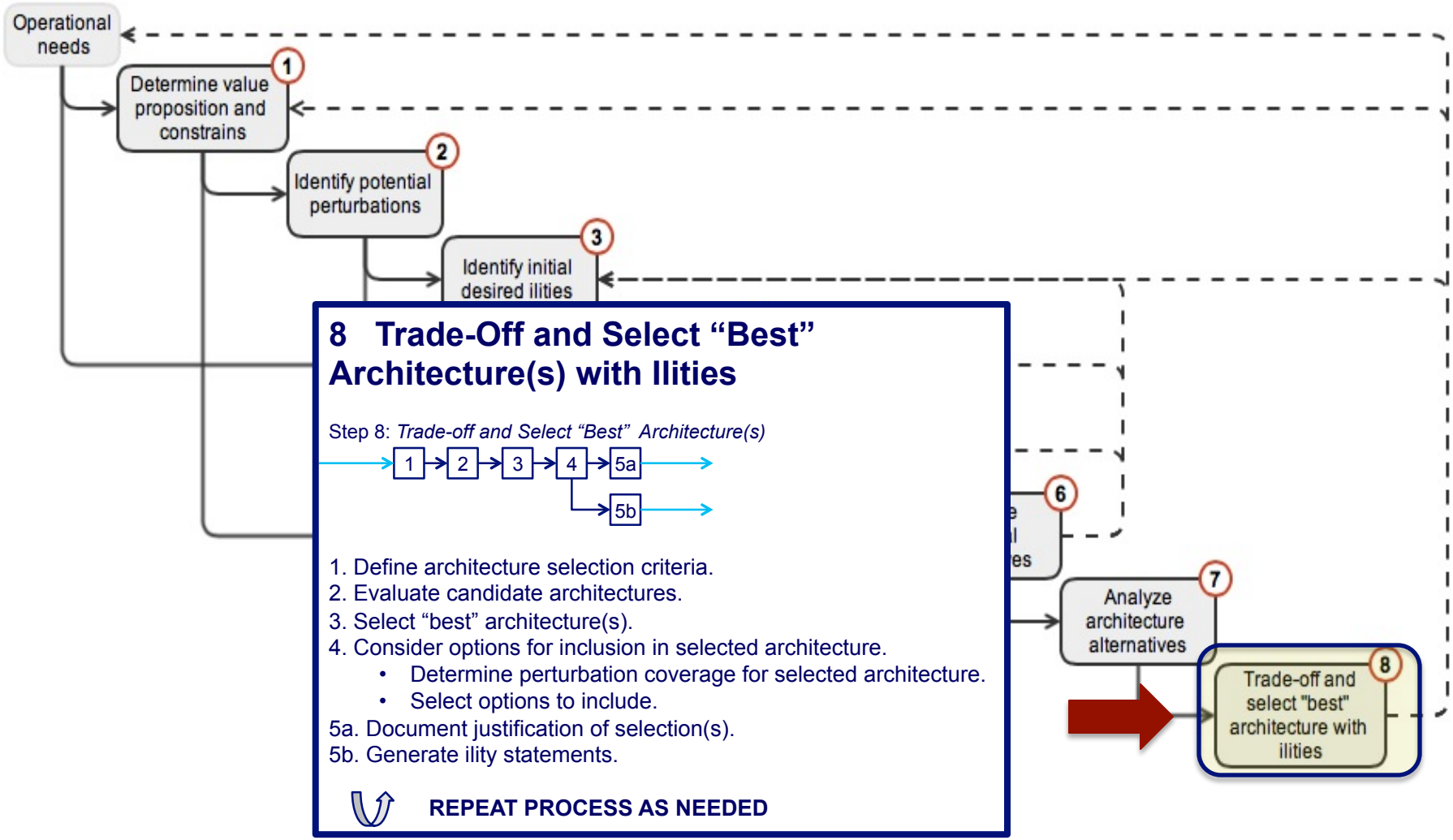
Step 7: *Analyze Architecture Alternatives*

1. Conduct Single Epoch Analyses (perf
2. Propose change execution strategies.
3. Conduct *Multi-Epoch Analysis* (perfor
   a. Evaluate ility screening metrics
   b. Select alternatives of interest.
   c. Complete Multi-Epoch Analysis
4. Conduct *Era-level Analysis* (performance across sequences of epochs).
5. **Collect set of alternatives of interest with ility metrics.**

	Ility Metric	Well-performing alternatives (design #)
Changeability	eNPT	5, 9, 49, 1729, 2090, 3505, 4797, 10113
	FOD	5, 9, 49
	FPS	2090, 4797, 10113
Affordability	Accumulated Utility vs Discounted Cost	49, 1729 (ERA 1) 49 (ERA 2) 1729 (ERA 3)
Survivability	TAUL	2090, 4797
Robustness	(f)NPT	5, 9, 49, 1729, 3505
	Pareto Set for Contexts of Interest	4797

© 2014 Massachusetts Institute of Technology

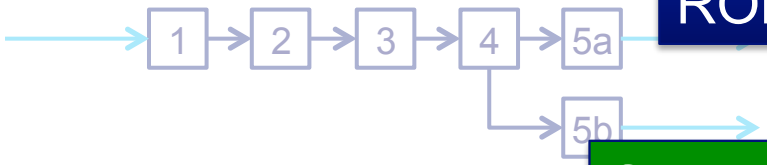
Step 8



© 2014 Massachusetts Institute of Technology

Selection Criteria: Ility Metrics

8 Trade-Off and Select "Best" Architecture(s) with Ilities



ROBUSTNESS

CHANGEABILITY

SURVIVABILITY

AFFORDABILITY

Acronym	Stands For	Definition
NPT	Normalized Pareto Trace	% epochs for which design is Pareto efficient in utility/ cost
fNPT	Fuzzy Normalized Pareto Trace	Above, with margin from Pareto front allowed
eNPT, efNPT	Effective (fuzzy) Normalized Pareto Trace	Above, considering the design's end state after transitioning
FPS	Fuzzy Pareto Shift	Difference in FPN before and after transition
FOD	Filtered Outdegree	Above, considering only arcs below a chosen cost threshold
TAUL	Time Weighted Avg Utility Loss	Integral of utility loss over time
-	Accumulated Utility vs. Discounted Cost	Lifecycle cost benefit

1. Define architecture selection criteria.

2. Evaluate candidate architectures.
3. Select "best" architecture(s).
4. Consider options for inclusion
   - Determine perturbation
   - Select options to inclu
5a. Document justification of selection(s).
5b. Generate ility statements.

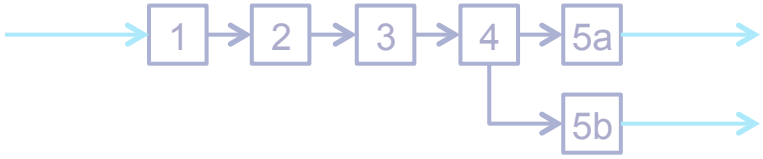
REPEAT PROCESS AS NEEDED

© 2014 Massachusetts Institute of Technology

Architecture Selection

8 Trade-Off and Select "Best" Architecture(s) with Ilities

Step 8: *Trade-off and Select "Best" Architecture(s)*



1. **Define architecture selection criteria.**
2. Evaluate candidate architectures.
3. **Select "best" architecture(s).**
4. Consider options for inclusion in selecte
   - Determine perturbation coverage
   - Select options to include.

5a. Document justification of selection(s).

5b. Generate ility statements.

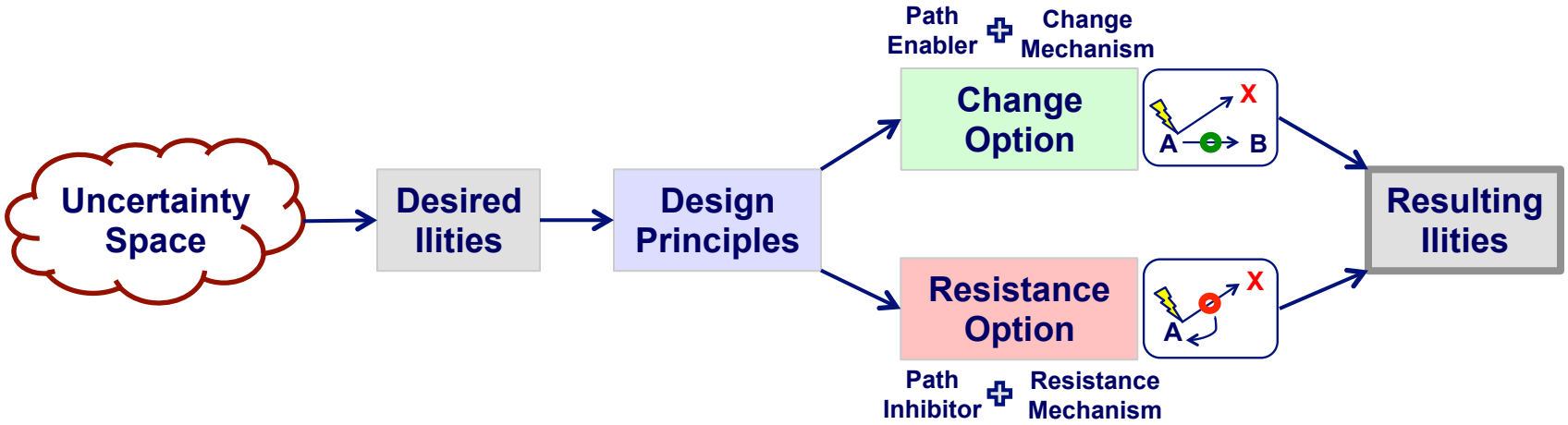
A	1	2	3	5
1	Changeability: 6 Affordability: 4 Survivability: 0 Robustness: 4	-4 -4 0 -4	-4 -4 +1 -4	-2 -4 +1 -2
2		Changeability: 2 Affordability: 0 Survivability: 0 Robustness: 0	0 0 +1 0	+2 0 +1 +2
3			Changeability: 2 Affordability: 0 Survivability: 1 Robustness: 0	+2 0 0 +2
5				Changeability: 4 Affordability: 0 Survivability: 1 Robustness: 2

REPEAT PROCESS AS NEEDED

© 2014 Massachusetts Institute of Technology

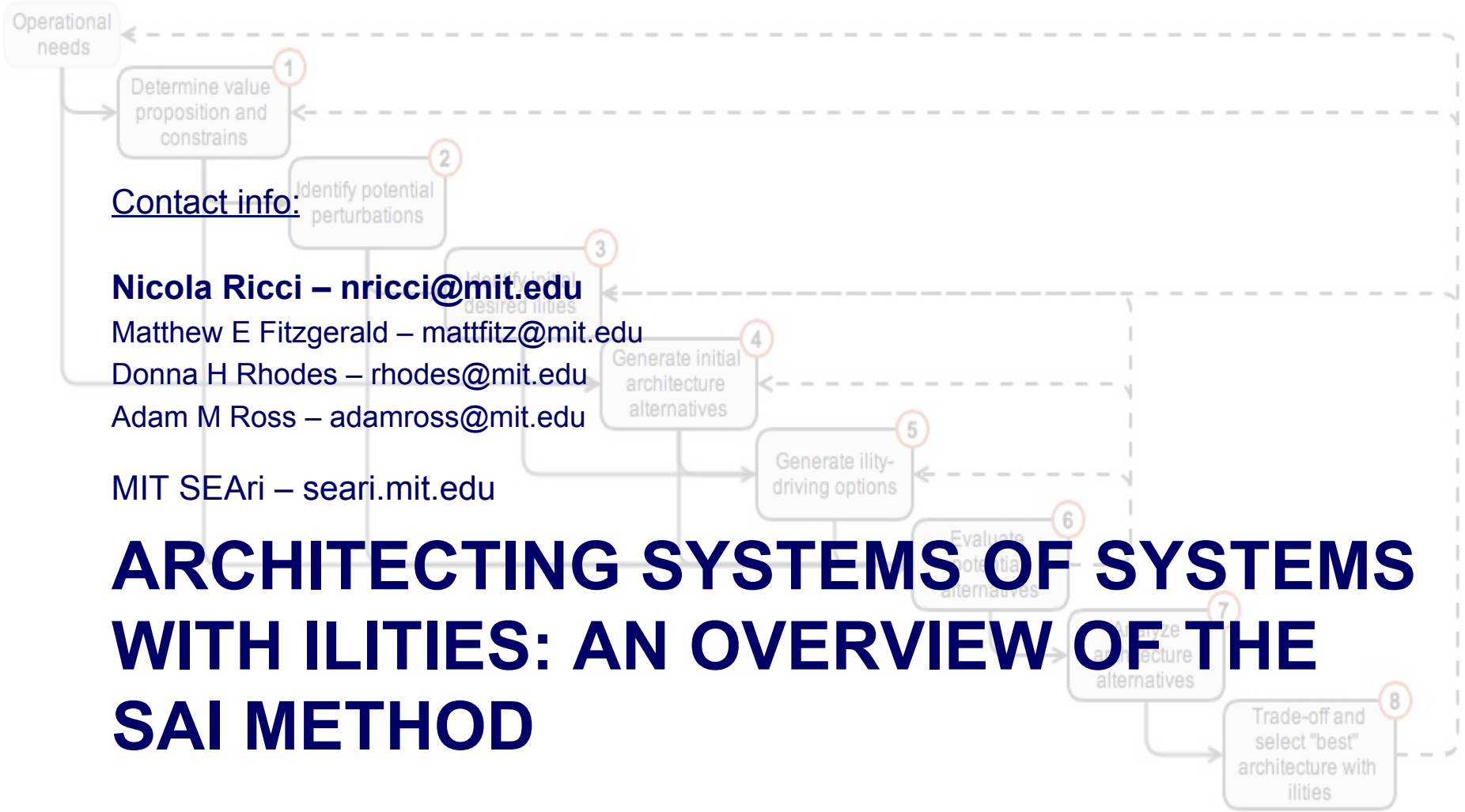
Summary

- Overview of SAI Method
- Method **scalable** in effort
  - Few milestones:
    - Identifying potential value-disrupting **perturbations**
    - **Ilities** of interest
    - Generalized **options**



© 2014 Massachusetts Institute of Technology

QUESTIONS?



Contact info:

Nicola Ricci – nricci@mit.edu
Matthew E Fitzgerald – mattfitz@mit.edu
Donna H Rhodes – rhodes@mit.edu
Adam M Ross – adamross@mit.edu

MIT SEAri – seari.mit.edu

ARCHITECTING SYSTEMS OF SYSTEMS WITH ILITIES: AN OVERVIEW OF THE SAI METHOD

© 2014 Massachusetts Institute of Technology

References

1. Maier MW. Architecting principles for systems of systems. In *Systems Engineering, volume 1, issue 4*: pp. 267-284, 1998.

2. Dahmann J, Rebovich G, Lowry R, Lane J, Baldwin K. An implementers' view of systems engineering for systems of systems. *Systems Conference (SysCon)*, 2011 IEEE International, pages 212-217.

3. Rhodes DH, Ross AM. Five aspects of engineering complex systems: emerging constructs and methods. *4th Annual IEEE Systems Conference*, San Diego, CA, April 2010.

4. Neches R, Madni AM. Towards affordably adaptable and effective systems. *Systems Engineering Journal*. Article first published online: 19 Oct. 2012. DOI: 10.1002/sys.21234.

5. Ross AM, McManus HL, Rhodes DH, Hastings DE, Long AM. Responsive systems comparison method: dynamic insights into designing a satellite radar system. *AIAA Space* 2009, Pasadena, CA, September 2009.

6. de Weck OL, Ross AM, Rhodes DH. Investigating relationships and semantic sets amongst system lifecycle properties (ilities). *3rd International Conference on Engineering Systems*, TU Delft, the Netherlands, June 2012.

7. Ricci N, Ross AM, Rhodes DH. A generalized options-based approach to mitigate perturbations in a maritime security systems-of-systems. *11th Conference on Systems Engineering Research*, Atlanta, GA, March 2013.

8. Fitzgerald ME, Ross AM, Rhodes DH. Assessing uncertain benefits: a valuation approach for strategic changeability (VASC). *INCOSE International Symposium 2012*, Rome, Italy, July 2012.

9. Keeney RL, Raiffa H. *Decisions with multiple objectives: preference and value tradeoffs*. Published by John Wiley & Sons, Inc., Ch. 2. (1976)

10. Keeney RL. *Value-focused thinking: a path to creative decisionmaking*. Harvard University Press (February 1, 1996)

© 2014 Massachusetts Institute of Technology

References

11. Bergey JK, Blanchette Jr S, Clements PC, Gagliardi MJ, Klein J, Wojcik R, Wood B. *U.S. Army Workshop on Exploring Enterprise, System of Systems, System, and Software Architectures*. Technical Report, CMU/SEI-2009-TR-008, ESC-TR-2009-008, SEI Administrative Agent. Copyright 2009 Carnegie Mellon University.

12. Feng W, Crawley EF. *Stakeholder value network analysis for large oil and gas projects*. Research Report, Engineering Systems Division. Cambridge, MA: Massachusetts Institute of Technology (2008)

13. Saaty TL. Decision making – the analytic hierarchy and network process (AHP/ANP). *Jurnl of Systems Science and Systems Engineering*, Vol. 13, No.1, 2004, pp.1-35.

14. Rader AA, Ross AM, Rhodes DH. A methodological comparison of Monte Carlo methods and epoch-era analysis for system assessment in uncertain environments. *4th Annual IEEE Systems Conference*, San Diego, CA, 5-8 April, 2010.

15. Beesemyer JC. Empirically characterizing evolvability and changeability in engineering systems. *Master of Science Thesis*, Aeronautics and Astronautics, MIT, June 2012.

16. Mekdeci B, Ross AM, Rhodes DH, Hastings DE. A taxonomy of perturbations: determining the ways that systems lose value. *6th Annual IEEE Systems Conference*, Vancouver, Canada, March 2012.

17. Ross AM, Beesemyer JC, Rhodes DH. A prescriptive semantic basis for system lifecycle properties. Working Paper 2011-2-2, http://seari.mit.edu/papers.php [cited 1-23-2014]. (2011)

18. Mekdeci B, Ross AM, Rhodes DH, Hastings DE. Investigating alternative concepts of operations for a maritime security system of systems. *INCOSE International Symposium 2012*, Rome, Italy, July 2012.

19. Wasson CS. *Systems analysis, design and development*. Published by John Wiley and Sons, Inc., Hoboken, New Jersey, 2006

20. Mekdeci B. Managing the impact of change through survivability and pliability to achieve viable systems of systems. *Doctor of Philosophy Dissertation*, Engineering Systems Division, MIT, February 2013.

© 2014 Massachusetts Institute of Technology

References

21. Mikaelian T, Hastings DE, Rhodes DH, Nightingale DJ. Model-based estimation of flexibility and optionability in an integrated real options framework. *3rd Annual IEEE Systems Conference*, Vancouver, Canada, March 2009.

22. Wilcox K. Design space Exploration. 16.888 *Multidisciplinary System Design Optimization lecture*. Massachusetts Institute of Technology, Cambridge, MA (2012, February 23rd).

23. Mekdeci B, Ross AM, Rhodes DH, Hastings DE. Controlling change within complex systems through pliability. *3rd International Conference on Engineering Systems*, TU Delft, the Netherlands, June 2012.

24. Fulcoly DO, Ross AM, Rhodes DH. Evaluating system change options and timing using the epoch syncopation framework. *10th Conference on Systems Engineering Research*, St. Louis, MO, March 2012.

25. Richards MG, Ross AM, Hastings DE, Rhodes DH. Multi-attribute tradespace exploration for survivability. *7th Conference on Systems Engineering Research*, Loughborough University, UK, April 2009.

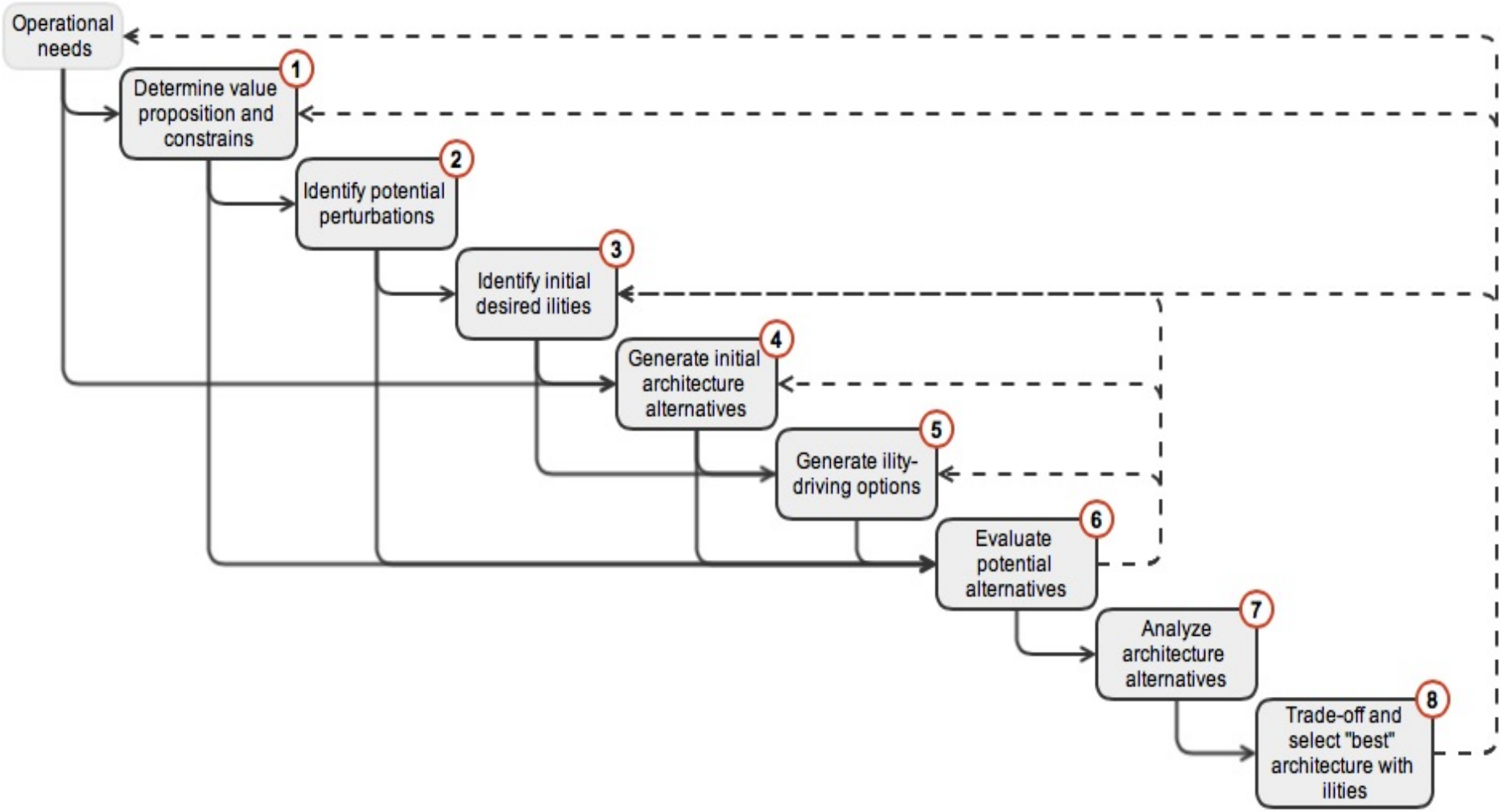
26. Ricci N, Ross AM, Rhodes DH, Fitzgerald ME. Considering alternative strategies for value sustainment in systems-of-systems. *7th Annual IEEE Systems Conference*, Orlando, FL, April 2013.

© 2014 Massachusetts Institute of Technology

Back-up Slides

© 2014 Massachusetts Institute of Technology

SoS Architecting with Ilities (SAI) Method



© 2014 Massachusetts Institute of Technology

Motivation

> **"Complex DoD systems tend to be designed to deliver optimal performance within a narrow set of initial requirements and operating conditions at the time of design. This usually results in the delivery of point-solution systems that fail to meet emergent requirements throughout their lifecycles, that cannot easily adapt to new threats, that too rapidly become technologically obsolete, or that cannot provide quick responses to changes in mission and operating conditions."**
>
> **- Office of the Secretary of Defense (SERC RT-18 Task Description, Sept 2010)**

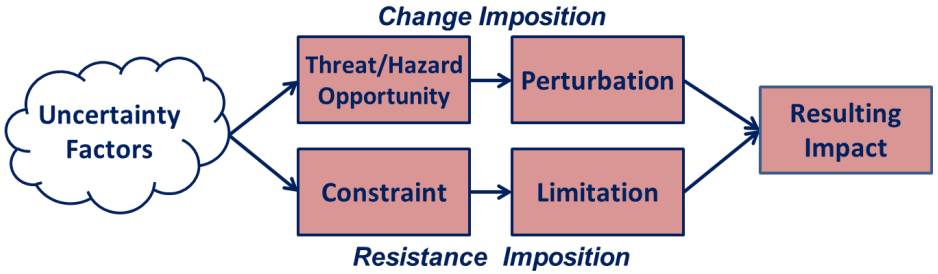
Optimization is weak to uncertainty…

Engineering design must move beyond optimization of "first use" considerations in order to create complex systems that are able to sustain value delivery in the face of uncertainty

© 2014 Massachusetts Institute of Technology

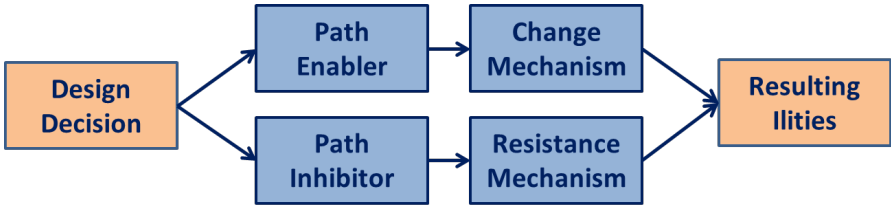
Ilities as Responses to Uncertainties

Uncertainties

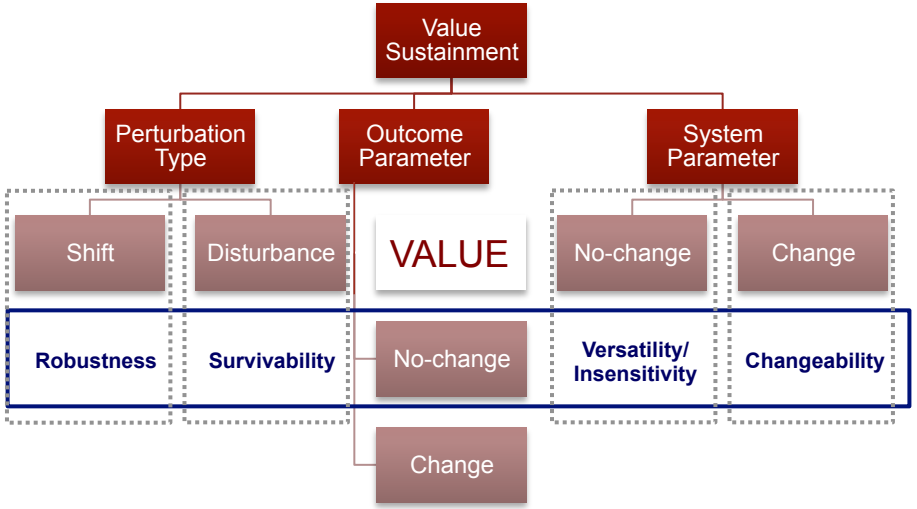


Perturbations and limitations impact value

Responses

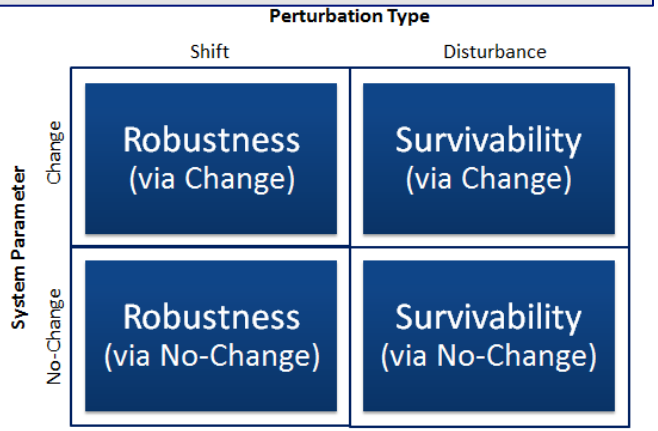


Changes and resistances maintain value



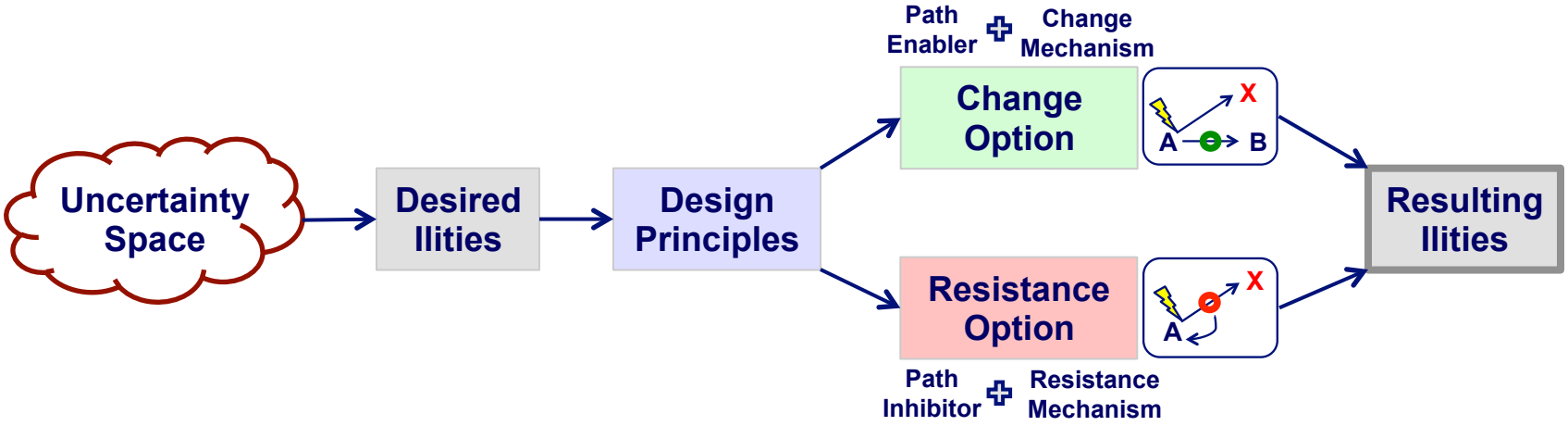
Suppose we want to maintain value
(i.e. no-change in outcome parameter *value*)

There are four high level ility responses



Further info: *Beesemyer, J.C., Empirically Characterizing Evolvability and Changeability in Engineering Systems, Master of Science Thesis, Aeronautics and Astronautics, MIT, June 2012.*

© 2014 Massachusetts Institute of Technology



© 2014 Massachusetts Institute of Technology

Options

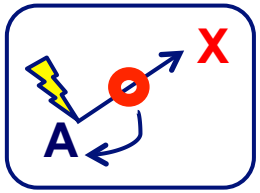
Problem	Perturbations introduce the **RISK** of interruption of value delivery
Solution	In the design of the system, include **"options"** that reduce the risk of interruption of value delivery

An **option**, in general, is *the ability to execute a design feature that will change or prevent change to the system in order to respond to perturbations and avoid interruption of value delivery*

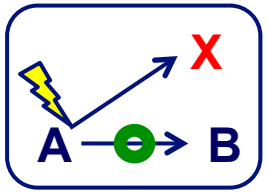
GENERALIZED OPTION

- **Resistance Option (RO)**
  - **Path Inhibitor (PI)**
  - **Resistance Mechanism (RM)**
- **Change Option (CO)**
  - **Path Enabler (PE)**
  - **Change Mechanism (CM)**

Having _____ allows you to _____
(path variable) (mechanism)



RO



CO

X = bad state
A,B = acceptable state

© 2014 Massachusetts Institute of Technology

Step 8.5b: Generate Ility Statements

- Based on the four evaluated transition rules and the decision to add the option of spares, we can generate the following ility statements:

Change Mechanism 1

In response to a decrease in available workforce, it is required that the SoS manager should have the ability to decrease the levels of the "number of vehicles" variables in the form of the design, to deliver value with respect to the need for a reduced manpower solution and with a reaction time of under 2 months. [i.e. scalability in vehicles as a response to workforce shortage]

Change Mechanism 2

In response to any change in context, it is required that the SoS manager should have the ability to change the level of the "operators per UAV" and "task assignment" CONOPs in the operation of the design, in order to deliver more benefits and with a reaction time of under 1 month. [i.e. modifiability in task assignment and operators per UAV as a response to changes in context]

Change Mechanism 4

In response to a system rearchitecting, it is required that the SoS manager should have the ability to increase the level of the "number of vehicles" variables in the form of the design, in order to deliver more benefits and with a span of at least 1 year. [i.e. evolvability in number of vehicles]

© 2014 Massachusetts Institute of Technology