

A Framework and Metrics for Addressing an Agile Enterprise

extending the observe-orient-decide-act paradigm into acquisition

Pritchett 'Chett' Harris

The SI Organization, Inc.

pritchett.a.harris@theSlorg.com



- 3 Timescales & Motivations
- 4 Framework
- 5 Gaming
- 6 The O-O-D-A Loop
- 7 Portfolio Management
- 8 Connected Models
- 9 Quanta of Systems of Systems Requirements
- 10 Key Metrics
- 11 Observable & Successful Metrics
- 12 Example
- 13 Questions
- 14 Survey

3 TIMESCALES & MOTIVATIONS.

Define “Enterprise” and the need for “agility”

Enterprise characterized by continuous operations cross multiple scales

At all levels, the enterprise competes with its environment, performing
Boyd’s Observe, Orient, Decide, Act (O-O-D-A)
cycle



Enterprise

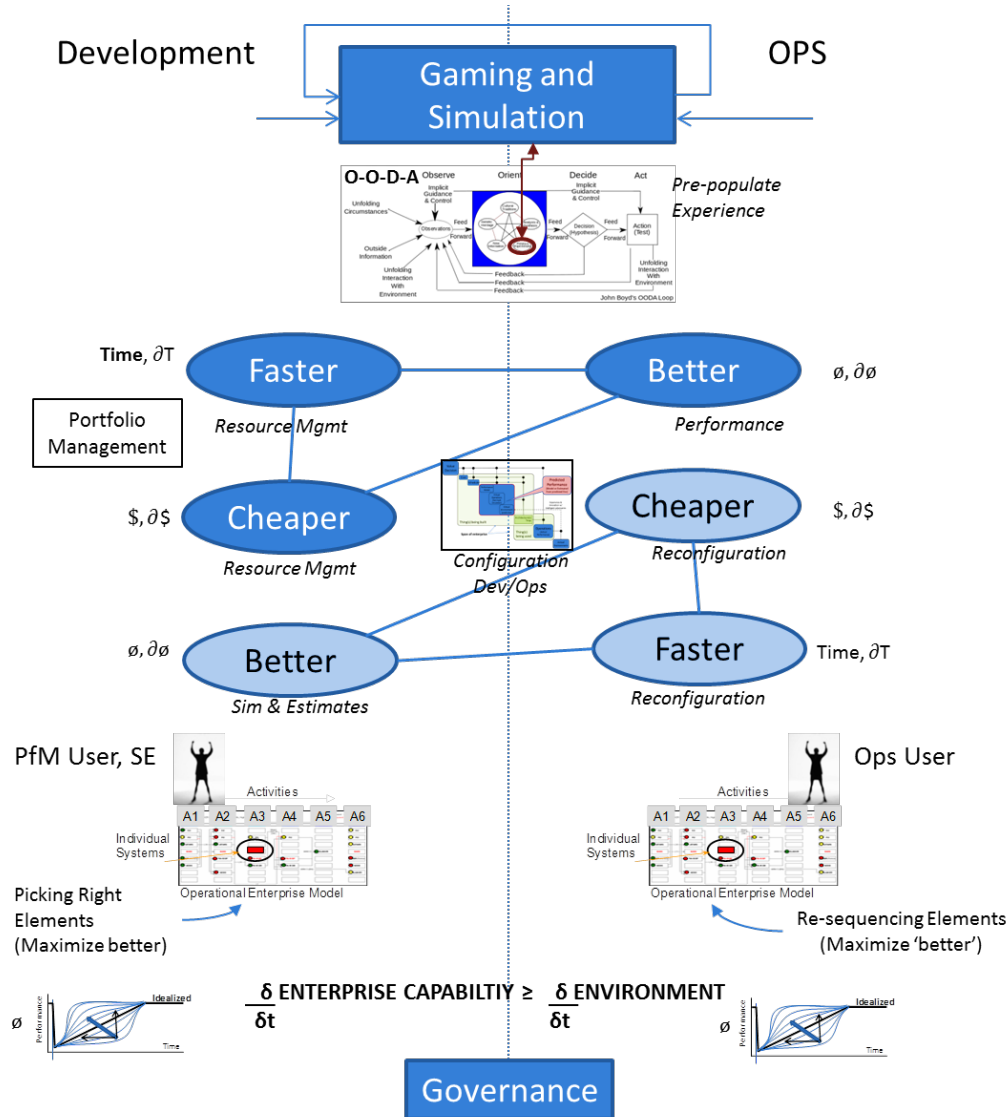


An entity comprised of one or more organizations, engaged in a mission requiring the development, sustainment, and projection of supporting capabilities in a changing environment.

A broad enterprise approach that “looks into the future” is desired

- Augment a static ‘specified environment’ with a ‘pseudo’ or ‘virtual’ environment in which the system(s) under development are expected to operate and succeed
- Design principals to operate across a range of future cases, rather than tightly specified design rules.

Identifies the conceptual relationships between various artifacts, techniques and metrics

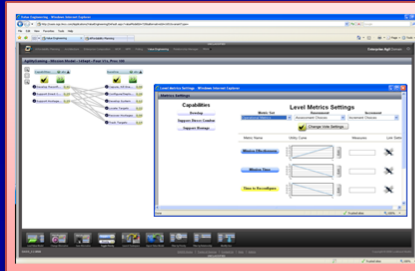


$\emptyset = \text{Performance}$

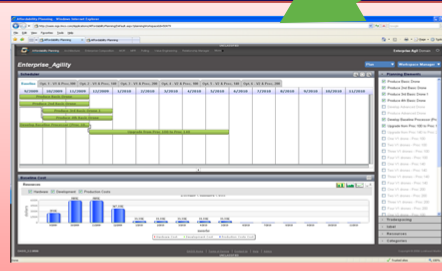
Provide tomorrow's feedback on today's decisions

SE Decision making (portfolio management)

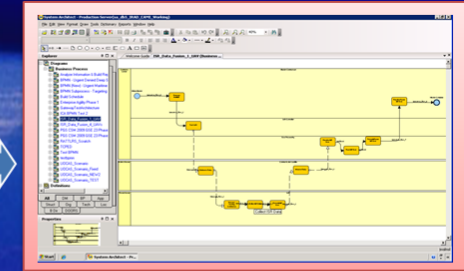
Value Model Validation Decision Scenario



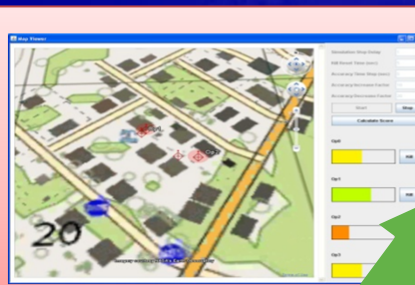
Implement Value Model and Affordability Planning to select highest value scenario (OASIS IRAD)



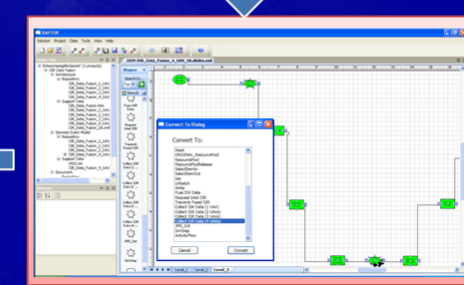
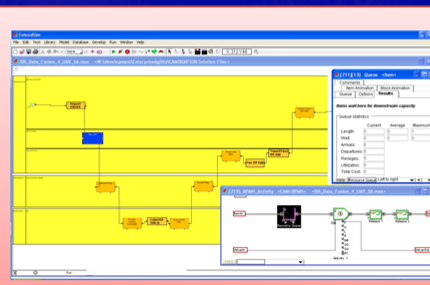
Re-evaluate value model based on collected gaming statistics



Build an architecture for the selected scenario



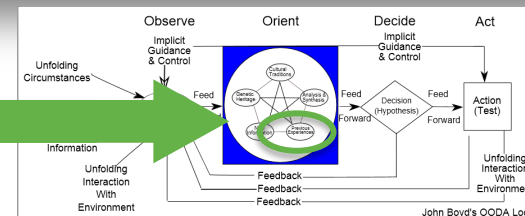
Execute the architecture in an immersive environment (Gaming IRAD)

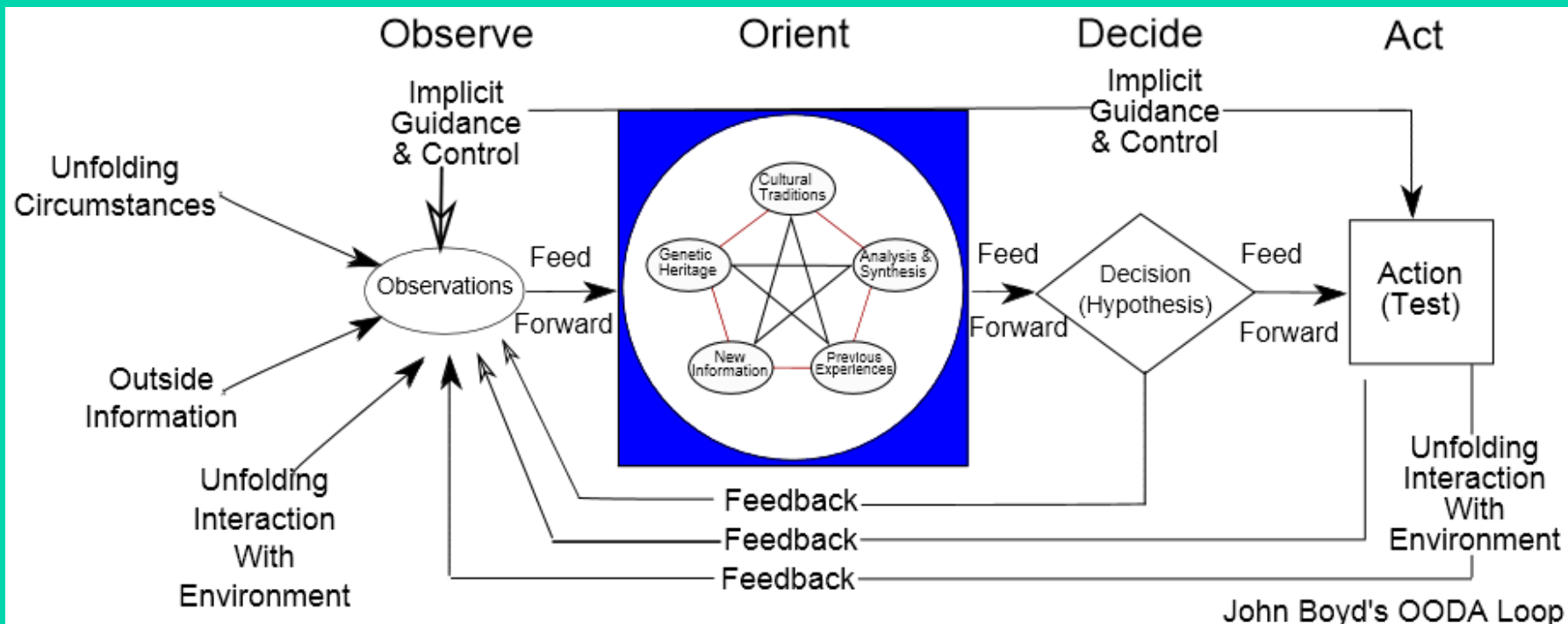


Generate a simulation model matching the architecture (CAMI IRAD)

Architecture to Executable

(in)validating decision assumptions and gathering 'previous' experience





Optimizing for ‘something’:

static value at minimum cost or schedule or capability/cost

or $\frac{\partial \text{change}}{\partial T}$

or $\frac{\partial \text{change}}{\partial \$}$

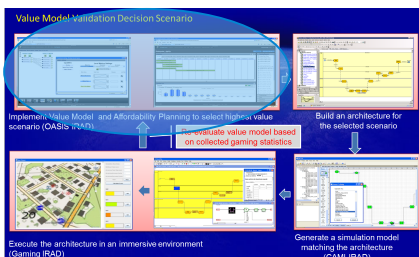
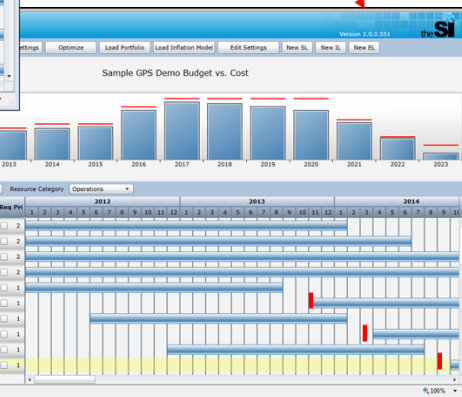
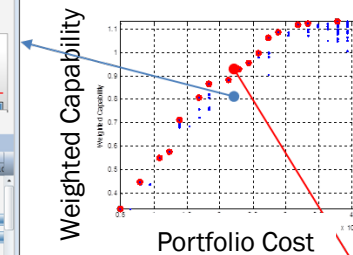
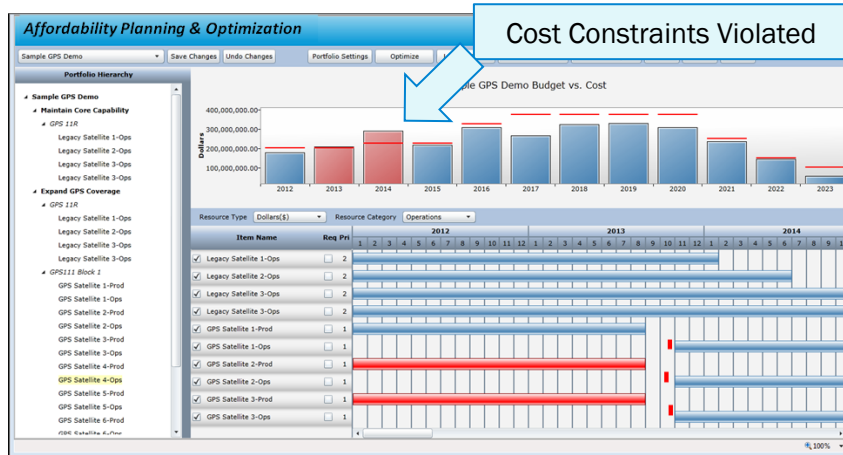
Pareto front of
Resource-constrained solutions
(via Genetic Algorithm)

Weighted Capability

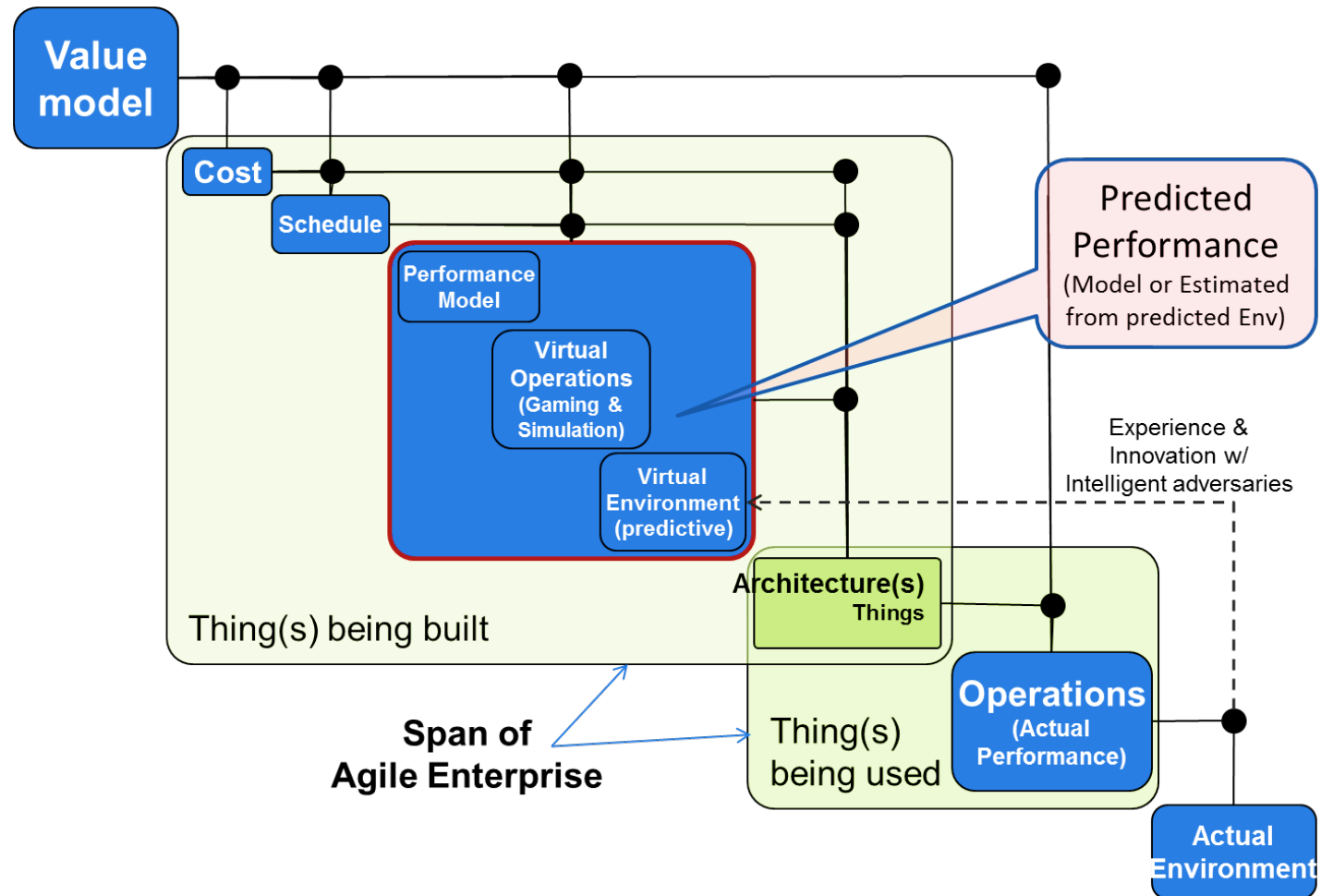
Portfolio Cost

This essentially
gives you...
partial
derivative w.r.t.
cost

$\frac{\partial \text{change}}{\partial \$}$

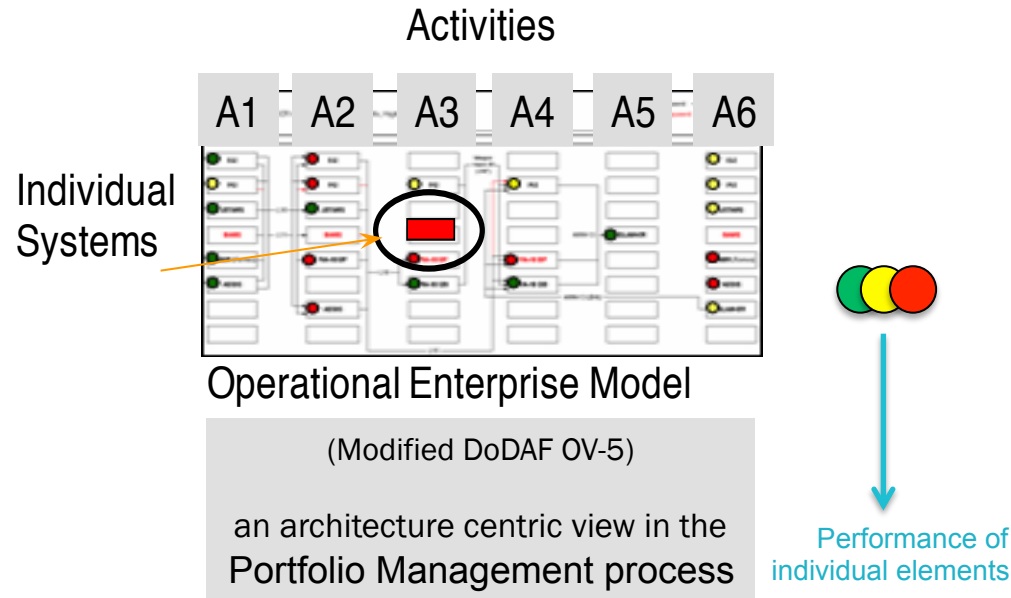


Link models that represent different points in time.



Q: How quickly can this go to “equilibrium”?

understand the units or quanta of change



How quickly can this change?

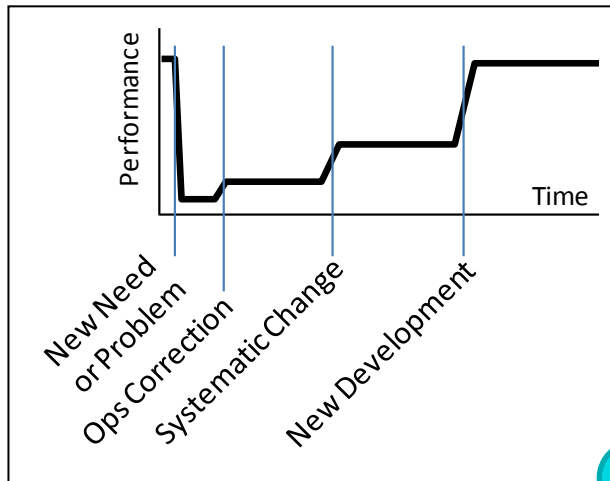
- In Operations (based on engineered solutions)
- In Development or SE space (governance)
- Combined?
- What aspect of DOTMLPF*?

How many ways can it be changed?

- What externals can it deal with?

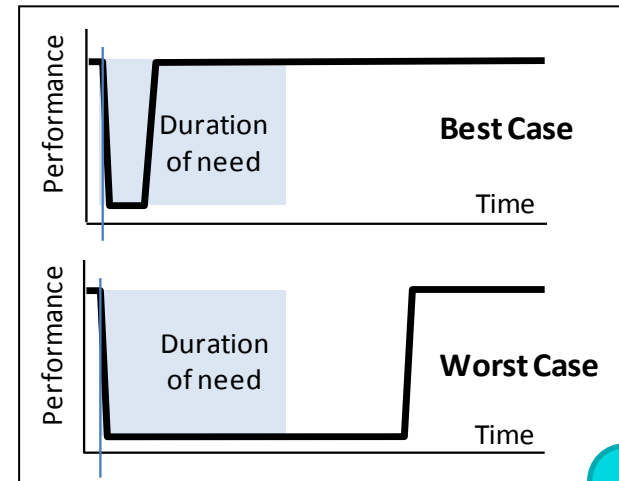
*DOTMLPF = doctrine, organization, training, materiel, leadership and education, personnel and facilities

10 KEY METRICS.

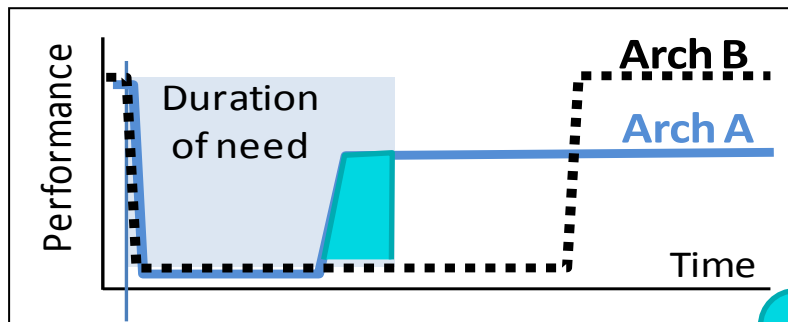


1

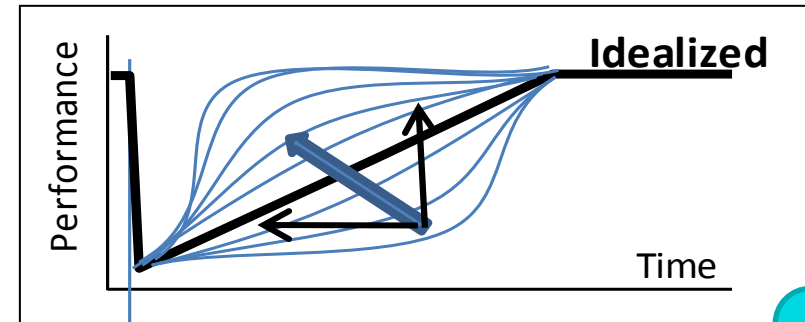
Note: the pre- and post- problem performance axis might be different



2



3

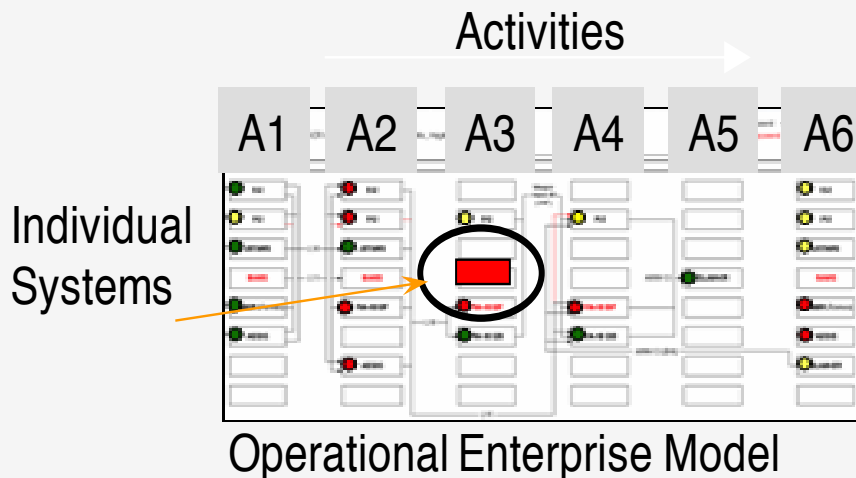


4

Improving this metric is a 'design' goal - when the problem is not known *a priori* the only agile feature which can be identified is 'reaction time' (time to change or recover).

Capability

(and the supporting portfolio of assets)



**Can your enterprise adapt faster
than the environment can
change?**

If you use the DoDAF OV-5 figure to
define your capability, can it be
reconfigured, or added-to within the
rate of change of the environment?

...or are you going to fall behind?

Is your enterprise keeping up?

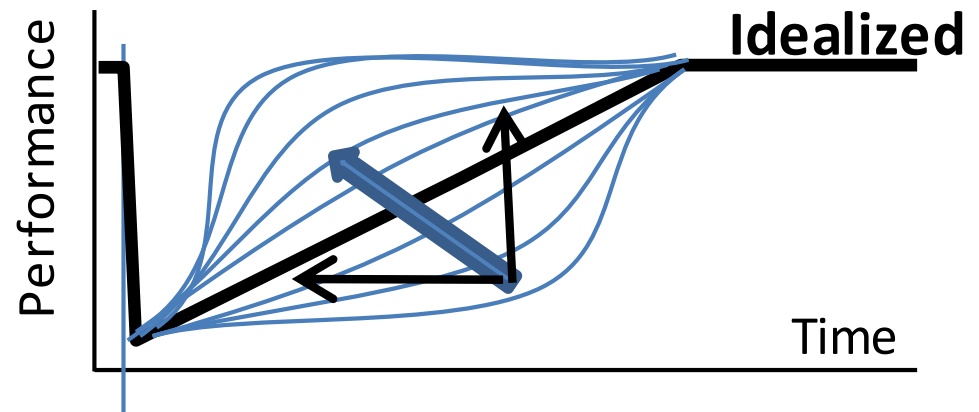
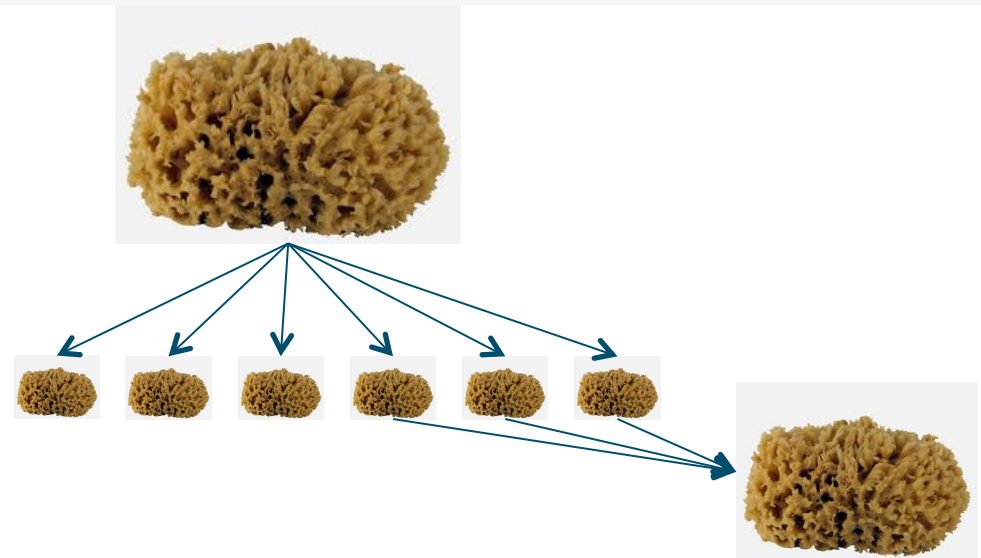
Is it *capable* of keeping up?

$$\frac{\delta \text{ ENTERPRISE CAPABILITY}}{\delta t} > \frac{\delta \text{ ENVIRONMENT}}{\delta t}$$

Damage tolerant agility

“Be recursively self-similar (fractal), regardless of scale”

Perform auto-aggregation and auto-recovery from disaggregation



This increases the “area under the curve”

thank you.

Future effort to improve the framework may include:

- Using genetic algorithms which now only adjust cost and schedule to adjust capabilities (system features or entire architectures) while still fitting in cost and schedule constraints (i.e. capability level portfolio management).
- Introducing budget and schedule uncertainty into the above process to drive genetic selection of 'budget agile' enterprises which can sustain capability deployment in funding risk environment.



Please take the time to rate this presentation by
submitting the web survey found @:

www.incose.org/symp2013/survey