On How Agile Systems Gracefully Migrate Across Next-Generation Life Cycle Boundaries

Rick Dove
rick.dove@stevens.edu

Garry Turkington
garry.turkington@gmail.com

Paper: www.parshift.com/Files/PsiDocs/Pap080614GloGift08-LifeCycleMigration.pdf

Continuing research into life cycle migration issues within and across system generations

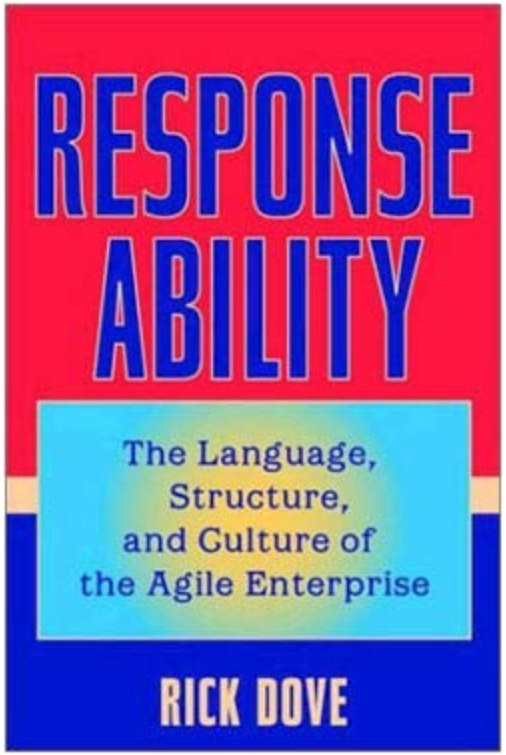
an investigation of Response Ability Principles (RAP) and their ability to illuminate agile system issues and enablers

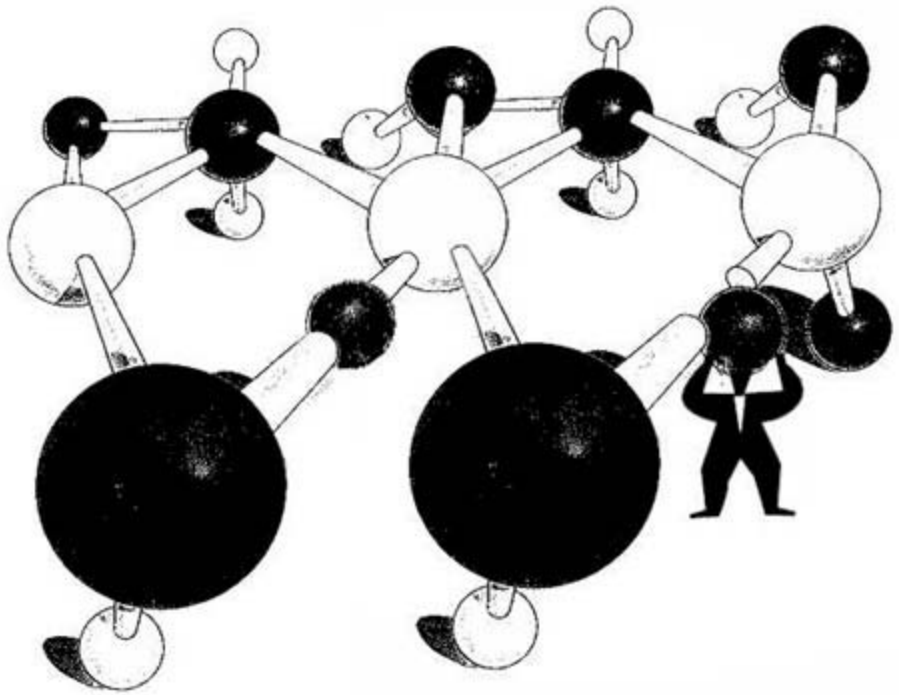
Garry Turkington

GLOGIFT 2008, June 14
Stevens Institute of Technology
Hoboken, New Jersey, USA

Engineering of Agile Systems and Enterprises
Fundamentals of Analysis, Synthesis, and Performance
SDOE 678

1/4 of Graduate Certificate in Agile Systems and Enterprise





Class 1 Agility: Reconfigurable Systems

Based on research funded by OSD/DARPA through Navy/NSF 1991-1997
Purpose: Identify and understand the next competitive paradigm
Lehigh University, Agility Forum, and ~1000 people from ~250 organizations

Purpose: Lay Foundation for Graduate Research Projects
(establishing efficacy of RAP* tools)

This is second in a series focused on emergent *life-cycle / agility* issues:

1. **(CSER-08) Agile Migration across phases within a system life-cycle**
2. **(GLOGIFT-08) Agile Migration across next-generation system boundaries**
3. **(next) Agile Migration impact on enterprise/system life-cycle models**
4. **…?**

This work began with the observation that

Continual Migration Enablement

has emerged as an enterprise/system focus

This paper's investigation:

1) **identifies two systems that exhibit cross-generation graceful migration**
2) **casts these systems in domain independent Response Ability architecture to illuminate nature and source of migration enablement**
3) **notes relationships of these "accidental" successes to two current purposeful attempts: Force Transformation and Service Oriented Architecture (SOA)**

***RAP: Response Ability Principles**

Defining Agility and Migration

Using the term as intended in the 1991 Lehigh study and subsequent research:

- ***Agility*** **is effective response under conditions of uncertainty**

There are at least three components to agility:

- **situational awareness,**
- **decisive choice making and**
- ***the ability to respond***

The latter aspect is what we deal with here

Migration **is the crossing of a change in basic infrastructure, be it technical, organizational or strategic.**

Contemporary Context

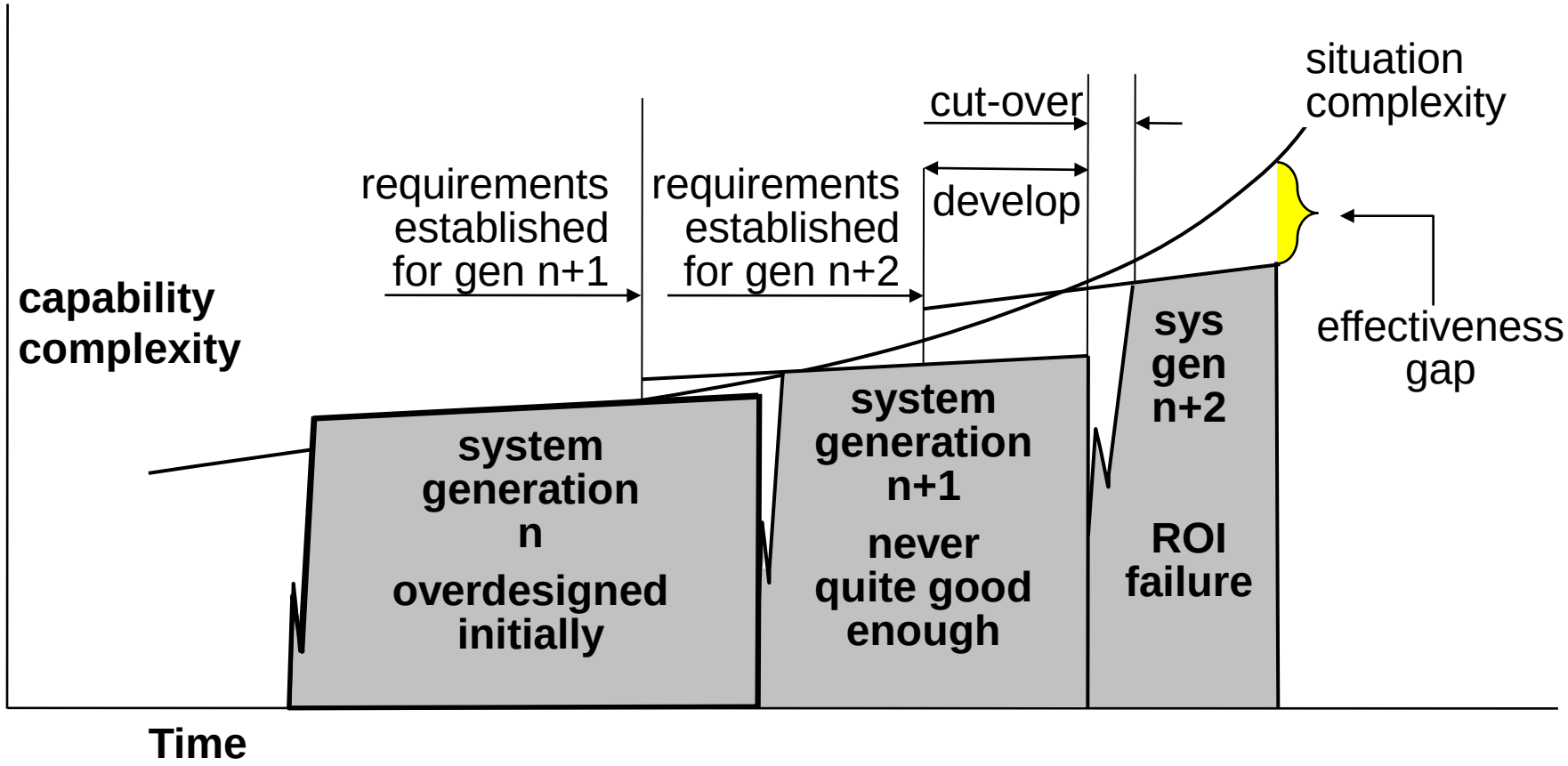
Next-generation challenges are demanding new architectures…

- ***Force Transformation*** **is the U.S. military's response to next-generation warfare**
- ***Service Oriented Architectures*** **is Business' response to next-generation competition**

Significant in both is the objective of *a change that enables future change*

Instead of perpetuating the scrap and replace cycle, an architecture is envisioned that facilitates migration through successive next generations

Increasing Gap Between Need and Capability



Approach

- Identify exemplar systems that migrate successfully across next-generation boundaries
- Determine if these systems can be expressed in domain-independent RAP architecture
- Determine if RAP expression illuminates migration enabling issues and levers
- Explore possibility of RAP to inform design concept when the objective is continual migration

Candidate System Consideration

- **Select from different domains to maximize potential for depicting domain-independent characteristics**
- **3 candidate systems considered:**
  - ***Home Entertainment transformation***
  - ***Internet Protocol transformation***
  - **Company Reinvention of value generation strategy (candidate company proved inappropriate)**

To Note:

- **These systems fit the general signature of plug-and-play (infrastructure) drag-and-drop (components) characteristic of RAP-based agile systems**
- **No attempt was made to find successful cross-generation migration in systems that do not exhibit this characteristic signature, and no suggestion is made here that such systems do not exist**

Home Entertainment

- Emerged roughly in the 40s-50s with speaker and amplifier shared by monaural reel-to-reel tape and record turntable
- Successive forms of local sourcing evolved: vinyl, reel-to-reel, cassette, 8-track, CD, DVD, computer, wii, …
- Successive forms of remote sourcing evolved: radio, TV, satellite, Internet…
- Successive forms of presentation evolved: monaural, stereo, surround sound, video, gaming feedback…
- Old stuff still works
- New stuff can be added anytime, or never
- ***Over 60 years*** --- choosing not to adopt new capability does not degrade existing systems, partial adoption is possible, mix of old with new is possible

---

- The expectation isn't perpetual system immortality, but the result has been a surprisingly useful extended life-cycle
- No master designers foresaw this range at the start

Internet Protocol

- **1964: Rand's Paul Baran publishes about packet switching networks: breaking messages into pieces and sending them via various redundant paths to the destination makes them difficult to destroy and hard to intercept**
- **1966: MIT's Larry Roberts publishes first ARPANET plan**
- **1968: ARPA issues RFQ for ARPANET packet switches**
- ***1969: BBN 50kbps packet switches installed at four universities***
- ***1970: First use of Network Control Protocol (NCP) for transport & addressing***
- **Early 80s: managed by NSF for government and university use only**
- ***1983: TCP/IPv4 replaces NCP for transport and addressing – Officially Internet***
- ***1995: NSF opens Internet for public use with Network Access Points***
- **2010: Internet runs out of IPv4 addresses and IPv6 is necessary**
- **IPv4 and IPv6 in concurrent usage now**
- **Devices with old addresses still and will still work**
- **Devices with new addresses and IPv6 protocol can be added anytime**

"Scientists developing networking technology in the 1960's knew that what they were building would be far bigger than themselves; nobody, however, could have predicted the explosion in Internet access and interest in the past several years. The original designers didn't even think email would be something people would want! …the real revolution took place … decades ago -- today's technology just rides on the wave of yesteryear."

[Scott Ruthfield, Internet History and Development: From Wartime Tool to Fish Cam, www.acm.org/crossroads/xrds2-1/inet-history.html]

Other sources: http://www.zakon.org/robert/internet/timeline/; www.livinginternet.com/i/ii_tcpip.htm

RAP: 7 Thought-Guiding Frameworks

Response requirements categories (4 reactive and 4 proactive elements):
Reactive: correction, variation, expansion, reconfiguration
Proactive: creation, improvement, migration, modification

Response performance metrics (4 elements):
Response: cost, time, quality, scope

Response-enabling design principles (10 elements):
Encapsulation, Compatibility, Reusability, Redundancy/Diversity, Scalability, Distributed, Loose, Deferred Commitment, Self-Organizing, Evolving Standards

Design quality principles (3 elements):
Requisite Variety, Parsimony, Harmony

An overarching architectural philosophy (3 elements):
Reusable modules Reconfigurable in a Scalable architecture (RRS)

System integrity responsibilities (4 elements):
Module Inventory, System Re-configuration
Module Evolution, Infrastructure Evolution

An architectural conceptual pattern:
Drag-and drop modules in a plug-and-play infrastructure

RAP: 7 Thought-Guiding Frameworks

Response requirements categories (4 reactive and 4 proactive elements):
Reactive: correction, variation, expansion, reconfiguration
Proactive: creation, improvement, migration, modification

Response performance metrics (4 elements):
Response: cost, time, quality, scope

This paper's focus

Response-enabling design principles (10 elements):
Encapsulation, Compatibility, Reusability, Redundancy/Diversity, Scalability, Distributed, Loose, Deferred Commitment, Self-Organizing, Evolving Standards

Design quality principles (3 elements):
Requisite Variety, Parsimony, Harmony

An overarching architectural philosophy (3 elements):
Reusable modules Reconfigurable in a Scalable architecture (RRS)

System integrity responsibilities (4 elements):
Module Inventory, System Re-configuration
Module Evolution, Infrastructure Evolution

An architectural conceptual pattern:
Drag-and drop modules in a plug-and-play infrastructure

Proactive Change Domains

	Domain	Definition and General Issues	General Characteristics
Proactive	**Creation (and Elimination)**	Make or eliminate something. Issues are generally involved with the development of something new where nothing was before, or the elimination of something in use.	**Proactive** changes are generally triggered internally by the application of new knowledge to generate new value. They are still proactive changes even if the values generated are not positive and even if the knowledge applied is not new – self initiation is the distinguishing feature here. A proactive change is usually one that has effect rather than mere potential; thus, it is an application of knowledge rather than the invention or possession of unapplied knowledge. Proactive change proficiency is the wellspring of leadership and innovative activity.
	Improvement	Incremental improvement. Issues are generally involved with competencies and performance factors, and are often the focus of continual, open-ended campaigns.	
	Migration	Foreseen, eventual, and fundamental change. Issues are generally associated with changes to supporting infrastructure, or transitions to next generation replacements.	
	Modification (Add/Sub Capability)	Addition or subtraction of unique capability. Issues are generally involved with the inclusion of something unlike anything already present, or the removal of something unique.	

From: *Response Ability – The Language, Structure, and Culture of Agile Enterprise*

4 Integrity Responsibility Elements

The “active” part of the infrastructure

maintaining sufficient inventory of modules ready for use (development people, team leaders, engagement procedures, reusable code modules, reusable test suites, etc),

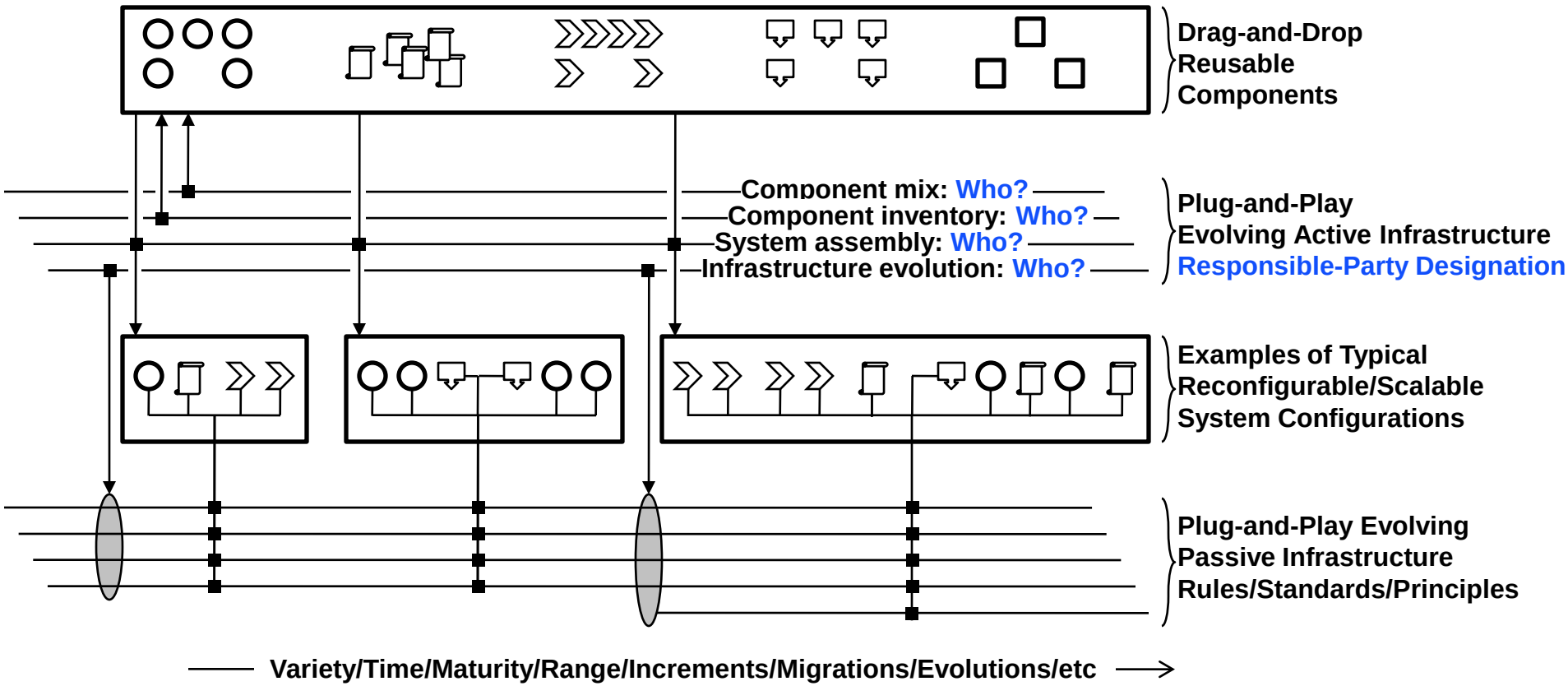
new module addition and upgrade as new capabilities are needed (new developer skills, newly developed code modules, new test suites for new code, new procedures as indicated by a changing situation, user representatives intimate with next stage feature development needs, etc),

infrastructure evolution (improvements to existing rules and standards, new rules and standards, etc), and

assembly of modules into on-demand system configurations suitable for changing response needs (successive iterations in the development process).

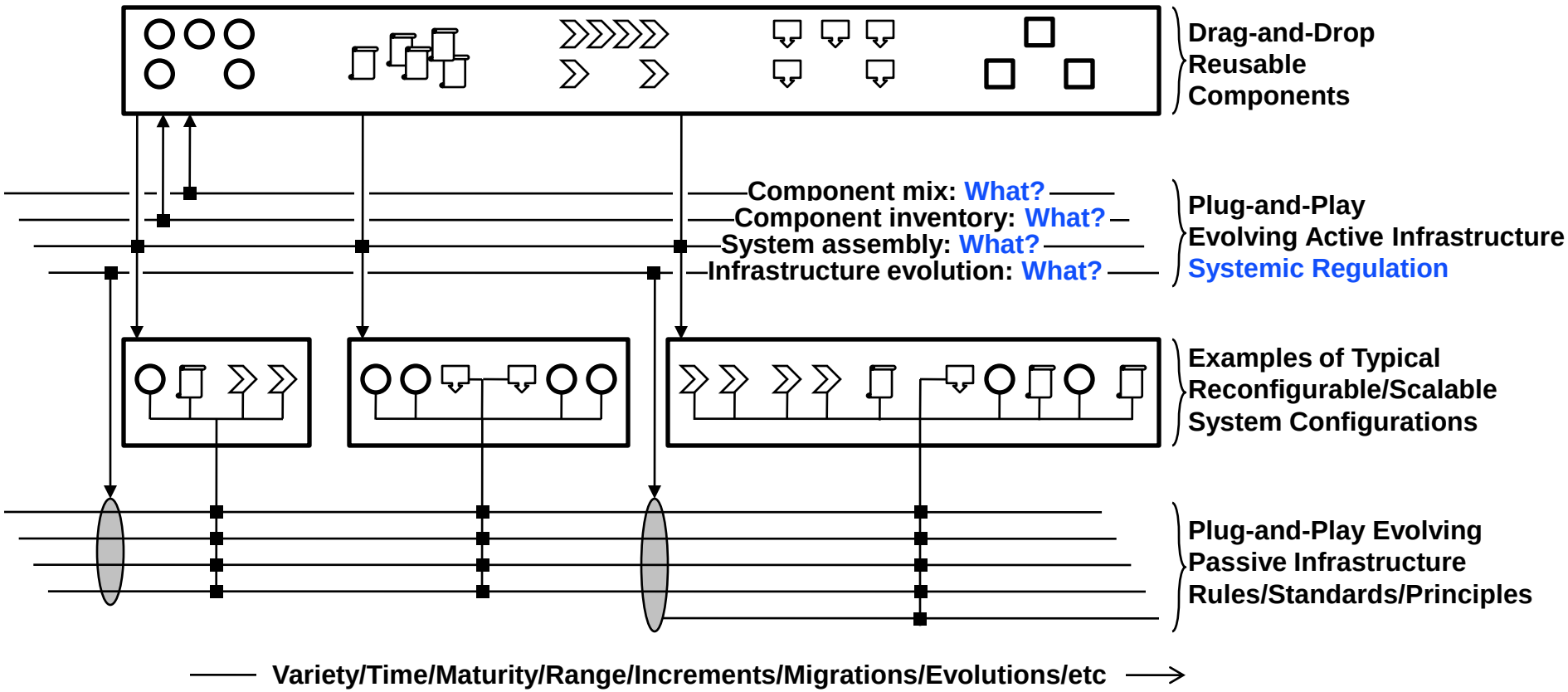
Agile System: Class 1 Reconfigurable

architectural concept pattern: drag-and-drop, plug-and-play



Agile System: Class 2 Reconfiguring

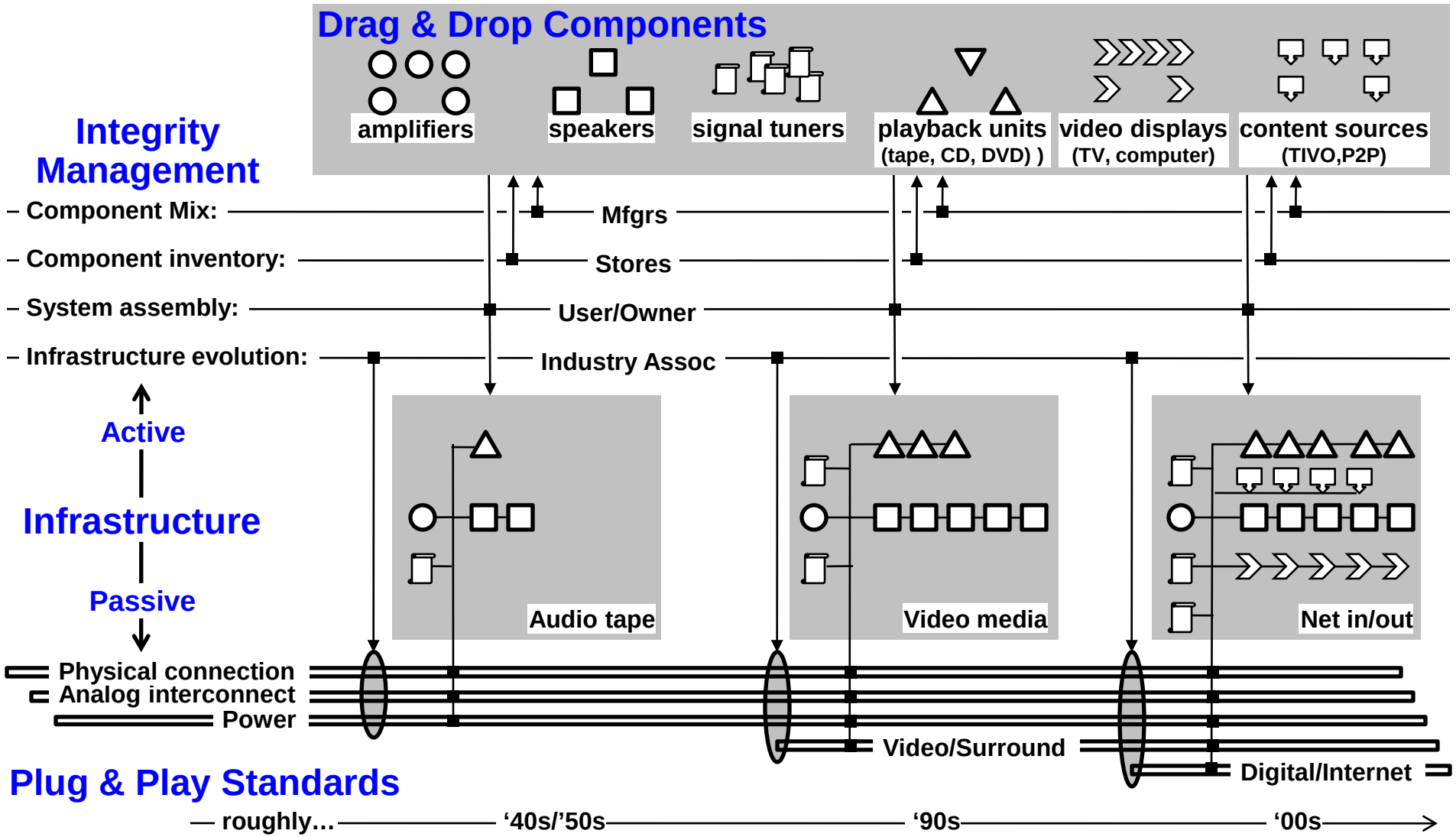
architectural concept pattern: drag-and-drop, plug-and-play



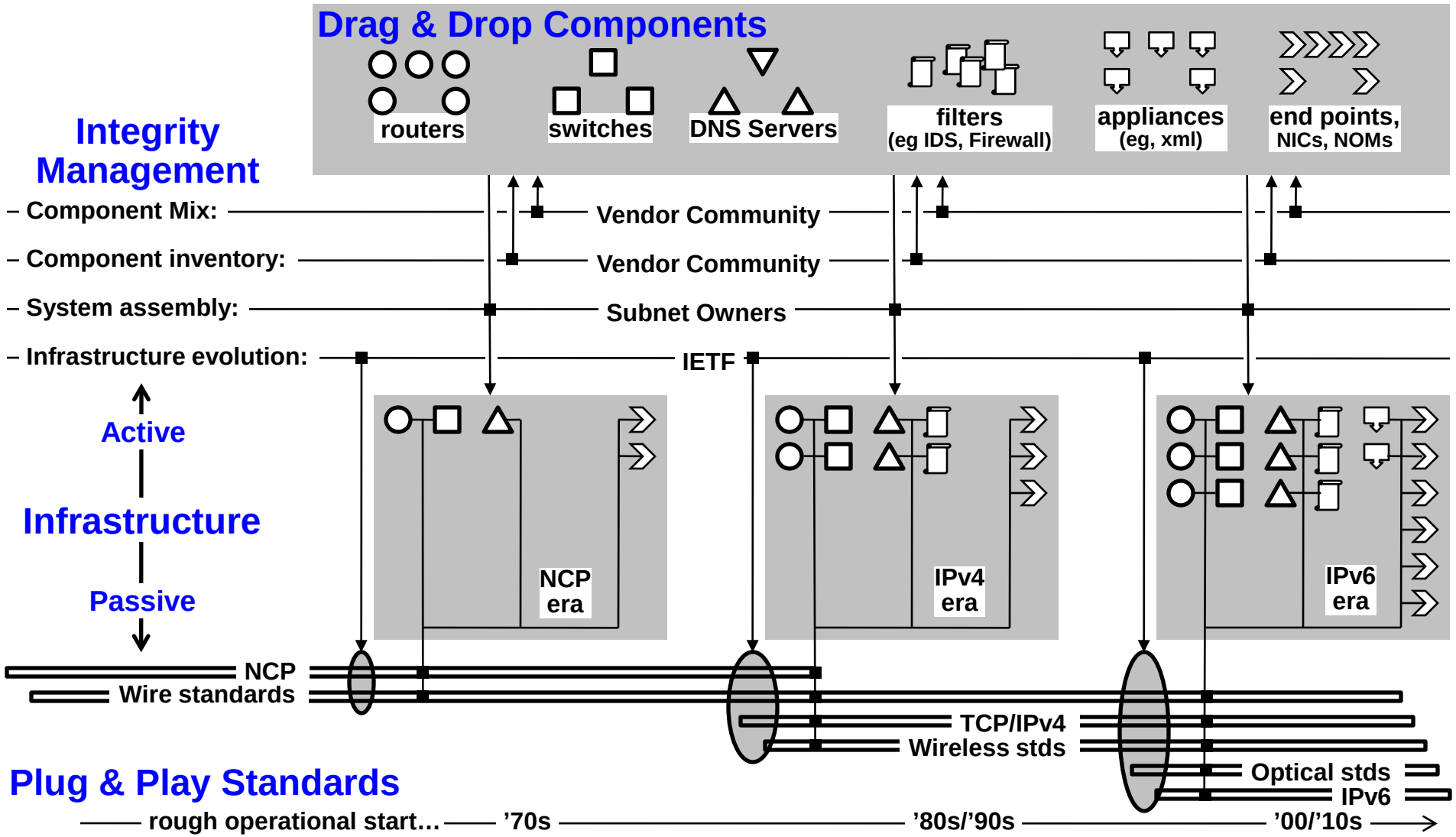
Three Questions

- **Q1: Can RAP tools model and depict the migration of Home Entertainment and Internet Protocol?**
- **Q2: Can RAP casting of these two systems inform the understanding of migration in agile systems?**
- **Q3: Might this RAP viewpoint of agile systems migration inform the in-process developments of SOA and Force Transformation?**

Q1: Can RAP Tools Model and Depict Migration of Home Entertainment



Q1: Can RAP Tools Model and Depict Migration of Internet Protocol



Q2: Can RAP casting of these systems inform the understanding of migration?

- **Both employ strict capabilities-based encapsulation. This is necessary, and facilitates migration by enabling functional swap-out, upgrade, retirement independently and asynchronously.**
- **Both employ a stable passive infrastructure of form-and-content interconnect standards, which is structured to facilitate open-ended augmentation over time with both additional and alternate-option standards. This is necessary, and facilitates migration enabling capability and capacity additions.**
- **Both employ an active infrastructure of stable responsibilities for the evolution of both components and passive infrastructure. This is necessary, and facilitates migration by sustaining controlled evolution.**

Q3: Can this RAP viewpoint inform SOA and Force Transformation?

The difference between a Class 1 and Class 2 RAP-based agile system is centrally-controlled sustainment vs. self organizing sustainment.

In Class 1 systems specific people with centralized sustainment responsibilities can be named, in Class 2 systems sustainment is caused by the equilibrium-seeking self-reorganization of decentralized interactions among autonomous agents.

Home Entertainment fits more a Class 1 profile – the owner that configures systems very centrally controls the system configuration, and has little effect or influence on owners of other Home Entertainment systems.

Internet Protocol fits more a Class 2 profile – there is a greater degree of coupling between the migration-deciding agents. As subnets opt for IPv6 profiles, other interconnected subnets may become shunned for services of lesser security or less optimal interaction.

SOA and Home Entertainment environments share a characteristic that may be useful in guiding SOA adoption plans. Both occur in relative isolation to their greater communities, and resemble a Class 1 agile system.

Force Transformation, on the other hand, has an environmental profile more like the Internet Protocol model. Both have sizable sub-groups with interdependent couplings – looking somewhat like an ecological system in the large.

SOA Adoption

- Adoption and subsequent migratory evolution of SOA within an enterprise is largely a local (enterprise) decision, with little interdependence on when and what other enterprises choose to do.
- Though enterprises are increasingly networked to each other electronically as well as strategically, SOA is largely an internal infrastructure for enterprise IT support of business practices. Perimeter gateways of various types are standard methods for reconciling inter company transactions.
- The nature of the SOA infrastructure nevertheless must conform to greater community common/universal standards if maximum and sustainable access to component services of benefit are to be realized.
- This raises a cautionary flag on brand-unique infrastructure employment, as well as enterprise- or brand-unique service interfaces.

Force Transformation

Force Transformation is a massive undertaking, on many functional fronts within each military force as well as across the many independent but interdependent military forces of Army, Navy, Air Force, Marines, and Coast Guard.

Force Transformation is predicated on developing far more intimate interoperability than currently exists.

The magnitude of the effort necessarily requires an asynchronous adoption for economic, cultural and technological reasons as a minimum – without any disruption of capability.

The military has a tradition of controlled mandated actions that may not serve well in either the initial adoption or the subsequent continual evolution intended.

The model of Internet Protocol migration that relies on pulling self-organized adoption with enticing benefit, rather than forcing a change that may be incompatible with the reality of the status quo, might well provide both economic and speed-of-adoption advantages.

Concluding Remarks

A broader application of RAP tools appears to be a promising path

for analyzing and guiding
structure and strategy
of extended life-cycle systems

like Force Transformation and SOA Adoption

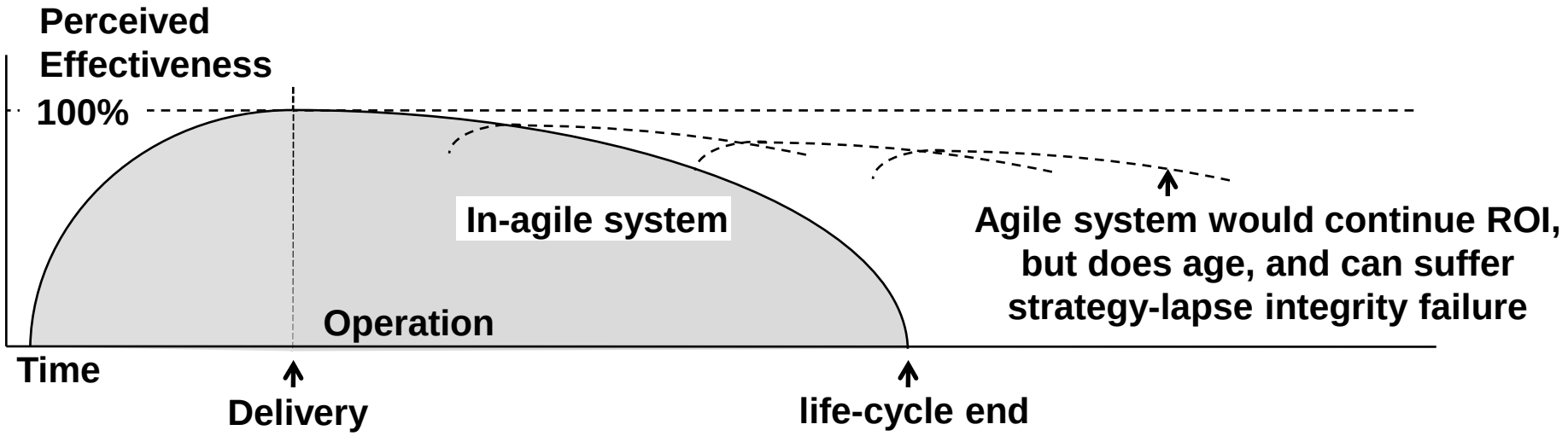
For Example:

A graduate project completed earlier this week (June 10, 2008)
built a comprehensive RAP characterization (all frameworks used)
of SOA and Web Services
to illuminate fundamental and necessary enabling principles

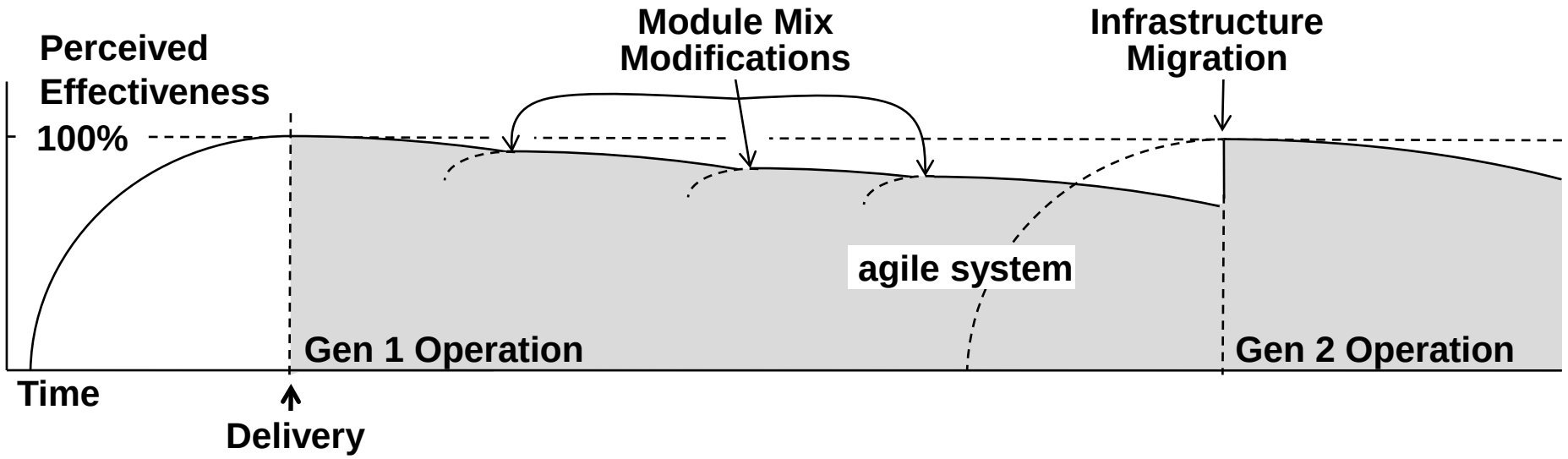
as a result of this paper's encouragement.

More will follow.

CSER Paper (April 2008)



GLOGIFT Paper (June 2008)



4-Course Graduate Certificate

GRADUATE CERTIFICATE IN

AGILE SYSTEMS & ENTERPRISES

LEADING TO A MASTER'S DEGREE IN

SYSTEMS ENGINEERING

An offering of the SDOE Program at Stevens Institute of Technology

More and more, systems today must show new agile capabilities for effective response in the face of continual change. Enterprise thrives only while it mirrors the environment it serves. Products lead only while they fit new demands better than others. Process serves only while it delivers new performance as needed. Integration works only while it meets new expectations.

Real-time responsiveness characterizes systems at the forefront of competition, enterprise, strategy, warfare, governance, innovation, engineering, information, integration, and virtually anything designed today for purpose. These and other systems are the focus of case studies, exercises, and course projects configured in response to participant needs and interests.

Agile systems dance with the environments they serve and live in; leading sometimes, following other times, but always dancing. Agile systems don't skip a beat when the music changes, when the tempo changes, when the dance-of-the-day changes, when the venue changes. And they are the partner of choice – they dance with élan, they are the source of energy and synergy.

www.stevens.edu/sdoe

STEVENS Institute of Technology

COURSES

Class 1

SDOE 675

Systems "Thinking" Integrating Paradox, Perspectives, and People

Agile systems and enterprises are the product of thinking that envisions systems holistically – in harmony with their environments, in existence across time, in service to a purpose, and in useful tension with the changing forces of reality. This module provides a fundamental underpinning for organized thought, drawing the practitioner toward insightful vision and strategy; building a solid conceptual foundation for defining, conceiving, and realizing agile systems and enterprises of all kinds.

SDOE 678

Engineering of Agile Systems and Enterprises: Fundamentals of Analysis, Synthesis, and Performance

Agile systems and enterprises are created and enabled by architectures, principles, and operational practices that facilitate responsive configuration and reconfiguration in the continual face of changing needs. This module introduces tools for analyzing and establishing response requirements and performance metrics; and engineering principles for synthesiz-

SDOE 679

Architecting the Extended Enterprise: Human-Activity Systems of Systems

Agility enables the extended enterprise, where purpose is achieved when multiple independent units respond effectively to common opportunity and common threat. This module explores the process and purpose of architecting the extended enterprises, from pick-up teams to alliance networks, from business process to corporate strategy, from local issues to global concerns.

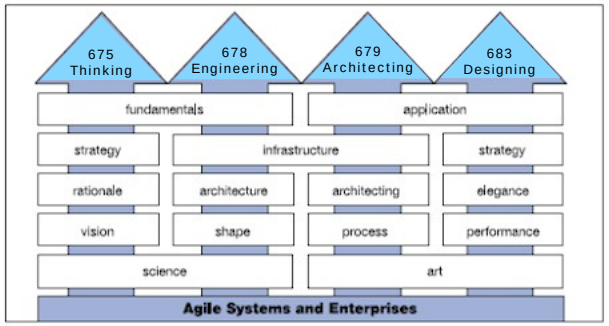
Class 2

SDOE 683

Design of Agile Systems and Enterprise: Design Quality and Self Organizing Systems

Esthetic quality in systems and enterprises makes the difference between enforced compliance and embraced experience; and determines the positive or negative vectors of self organization and emergence. This modules explores the value and nature of esthetic design quality, principles and architectures for harnessing self organized extended enterprise, agility as risk manage-

This certificate in Agile Systems and Enterprise integrates four complimentary courses. One common theme throughout defines enterprise as a human activity system. Another defines agile systems as those responding effectively to unpredicted situations, at all times, within mission. These common themes facilitate a study of agility across a seemingly wide variety of interesting system types, with the lines of difference blurred as each informs the other. The frontier of systems engineering today seeks new levels of system capability and behavior, and expects to find those benefits in higher forms of systems that elude traditional control and creation concepts.



Audience

This graduate certificate is relevant to engineers, managers, and decision makers in commercial, healthcare, financial and insurance, and defense domains working with systems that must thrive in a dynamic unpredictable environment - especially if they are system of systems or enterprise systems. The graduate certificate and the constituent courses first build a theoretical and philosophical basis for understanding and formulating the interactive and interdependent problem and solution spaces, and then suggest pragmatic and executable approaches to realize the enterprise potential.

Please visit our website at: **www.stevens.edu/sdoe**