Webinar

Agile Systems & Processes 105: Operational Awareness – Alert to Threats and Opportunities

04 Dec 2016

(last update)

Rick Dove

Taos County, New Mexico, dove@parshift.com, 575-586-1536
CEO/CTO, Paradigm Shift International
Adjunct Professor, Stevens Institute of Technology
Chair: INCOSE WG on Agile Systems & Systems Engineering

Download 106 webinar slides: Agile System/Process as Risk Management
Download 105 webinar slides: Agile System/Process Operational Awareness
Download 104 webinar slides: Agile System/Process Engagement Quality
Download 103 webinar slides: Agile System/Process Design Principles
Download 102 webinar slides: Agile System/Process Design Requirements
Download 101 webinar slides: Agile System/Process Architecture Pattern
(updated asynchronously from time-to-time)

Agile Systems & Processes 105:
Operational Awareness – Alert to Threats and Opportunities

Abstract: Agility is required in operational environments that are Capricious, Uncertain, Risky, Variable, and Evolving (CURVE). Agility is enabled by common architecture and design principles – a static capability. Agility is facilitated by an agility-sustaining ConOps – a dynamic behavior, driven by operational awareness, proactively alert to internal and external threats and opportunities. This is true for systems of all kinds, from agile development and deployment processes, to agile systems and products that are deployed. This webinar will focus on the facilitating behaviors of operational awareness, and the enabling mechanisms for acting upon that awareness, with real examples.

Presenter Bio: Rick Dove was co-PI on the original work which identified Agility as the next competitive differentiator, in a 3-month industry-collaborative workshop funded by the US Office of the Secretary of Defense in 1991 at Lehigh University. He went on to organize and lead the US DARPA-funded industry collaborative research at Lehigh University's Agility Forum, developing fundamental understandings of what enables and characterizes system's agility. He authored *Response Ability – The Language, Structure, and Culture of the Agile Enterprise*. He has employed these agile concepts in both system architecture and program management for large enterprise IT systems, for rapid manufacturing systems and services, and for self-organizing security strategies. At Stevens Institute of Technology he teaches graduate courses in basic and advanced agile-systems and agile systems-engineering, at client sites. He is CEO/CTO of Paradigm Shift, an applied research firm specializing in agile systems concepts and education. He is an INCOSE Fellow, and chairs the INCOSE working groups on Agile Systems and Systems engineering, and on System Security Engineering.

Goals

Appreciation for:

The need to sustain agility:

- **The environment presents threats and opportunities of unpredictable nature at unpredictable times, which require effective response.**

The means to enable sustainment:

- **Agile Architecture Pattern of SE process and product.**
- **Agility design principles for developing SE process and product.**
- **Embedded responsibilities for awareness, mitigation, and evolution.**

The means to facilitate sustainment:

- **Agile Architecture Pattern of SE process and product.**
- **Agility behavior principles for operating SE process and product.**
- **Embedded ConOps of monitoring, mitigation, and evolution.**

Why This Matters

CURVE

Internal and external environmental forces that impact process and product as systems

Capriciousness: unanticipated system-environment change

Uncertainty: kinetic and potential forces present in the system

Risk: relevance of current system-dynamics understanding

Variation: temporal excursions on existing behavior attractor

Evolution: experimentation and natural selection at work

(CURVE: formerly known as UURVE, Capriciousness = Unpredictability)

Value Proposition for Agility

Faster, lower cost system development?

An appealing argument, but only a side effect (at best).

The value proposition for agility is Risk Management.

Sustainability of process and product at risk.

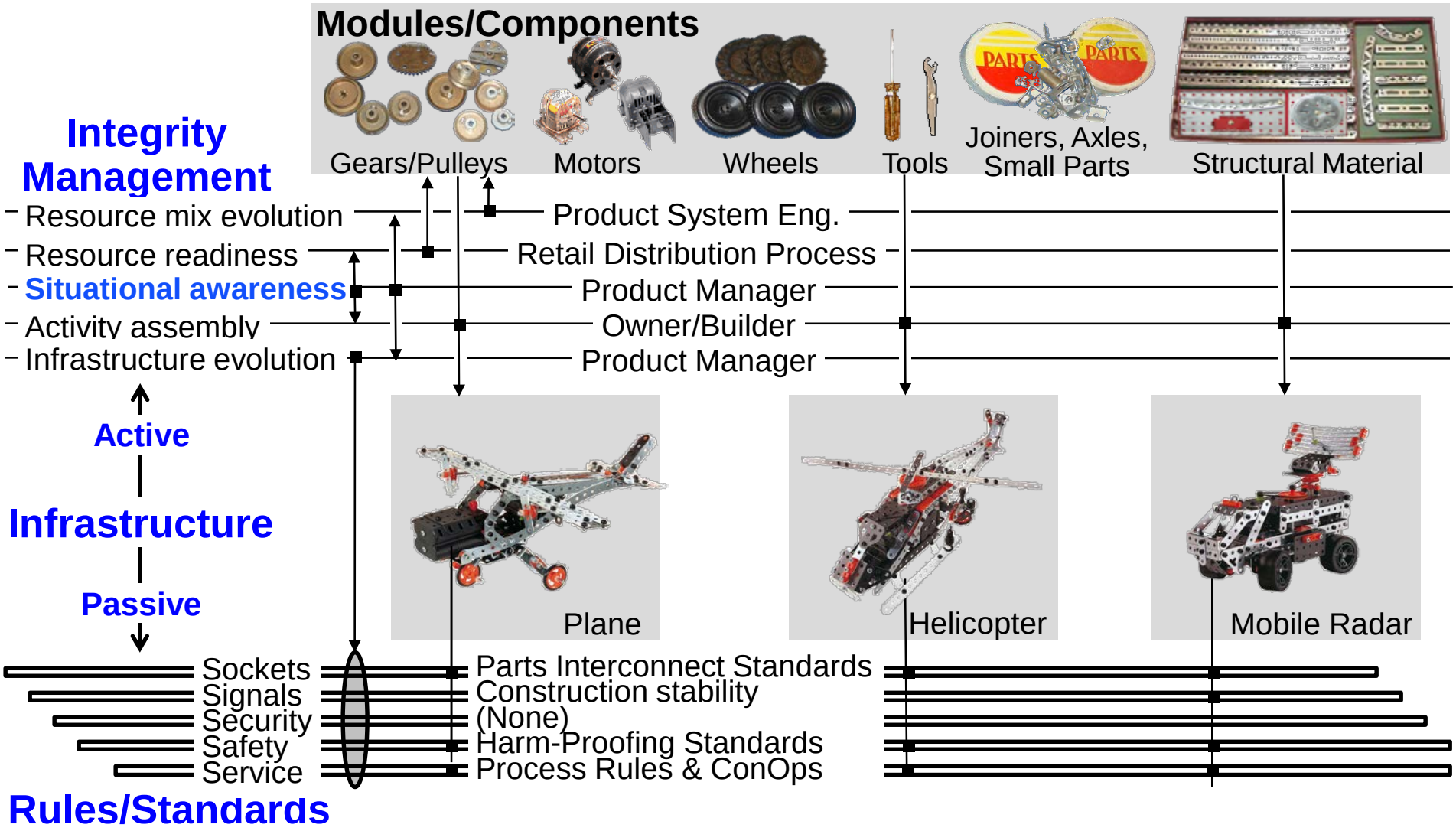
==========

Why is incremental and iterative development useful?

Why are incremental retrospectives useful?

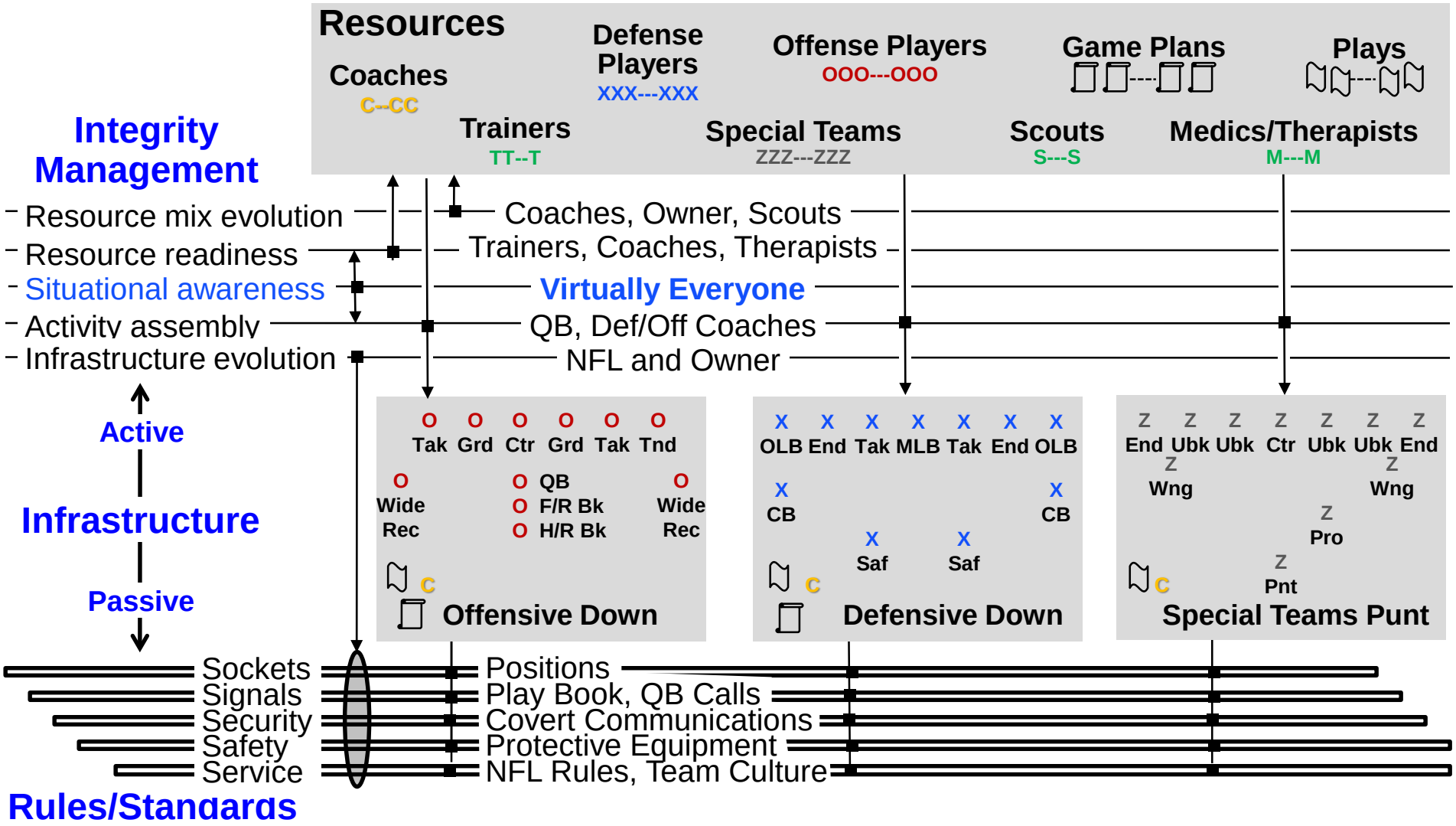
To learn about and mitigate risk affordably.

Enabling Sustainable Agility
Iconic Agile Architecture Pattern (AAP)
System Response-Construction Kit



Operating Process AAP for USA Football

Drag-and-drop resources in a plug-and-play infrastructure



(a concept example, not exhaustive)

Both Process and Product Need AAP

Agile SE processes deal with changing knowledge and environment.

- **They learn and employ that learning during SE process operation.**
- **They modify/augment product-development work-in-process.**
- **The product's AAP enables affordable wip modification/augmentation.**

For the record – but not for discussion here

Agile software development processes (silently) rely on product AAP.

- **Program code development employs an object-oriented AAP development platform (e.g., C++, Java, Eclipse).**
- **Web code development employs a loosely-coupled modular AAP inherent with hyperlinked web-pages.**

Agile hardware development doesn't have off-the-shelf AAP tools.

- **Proprietary product-line-engineering employs AAP.**
- **Proprietary Open System Architecture (OSA) employs AAP.**
- **Proprietary Live-Virtual-Constructive platforms employ AAP.**

Agility-Facilitating Structural Design Principles (RRS)

see INCOSE Webinar Agile 103

Reusable

- Encapsulated modules
- Facilitated interfacing
- Facilitated re-use

Reconfigurable

- Peer-peer interaction
- Deferred commitment
- Distributed control & information
- Self organization

Scalable

- Evolving infrastructure standards
- Redundancy and diversity
- Elastic capacity

Agility-Facilitating Operational Design Principles (MME)

2015 Discoveries of the INCOSE ASELCM Project (WIP)

Monitoring (Observing, Orienting)

- External awareness
- Internal awareness
- Sense making

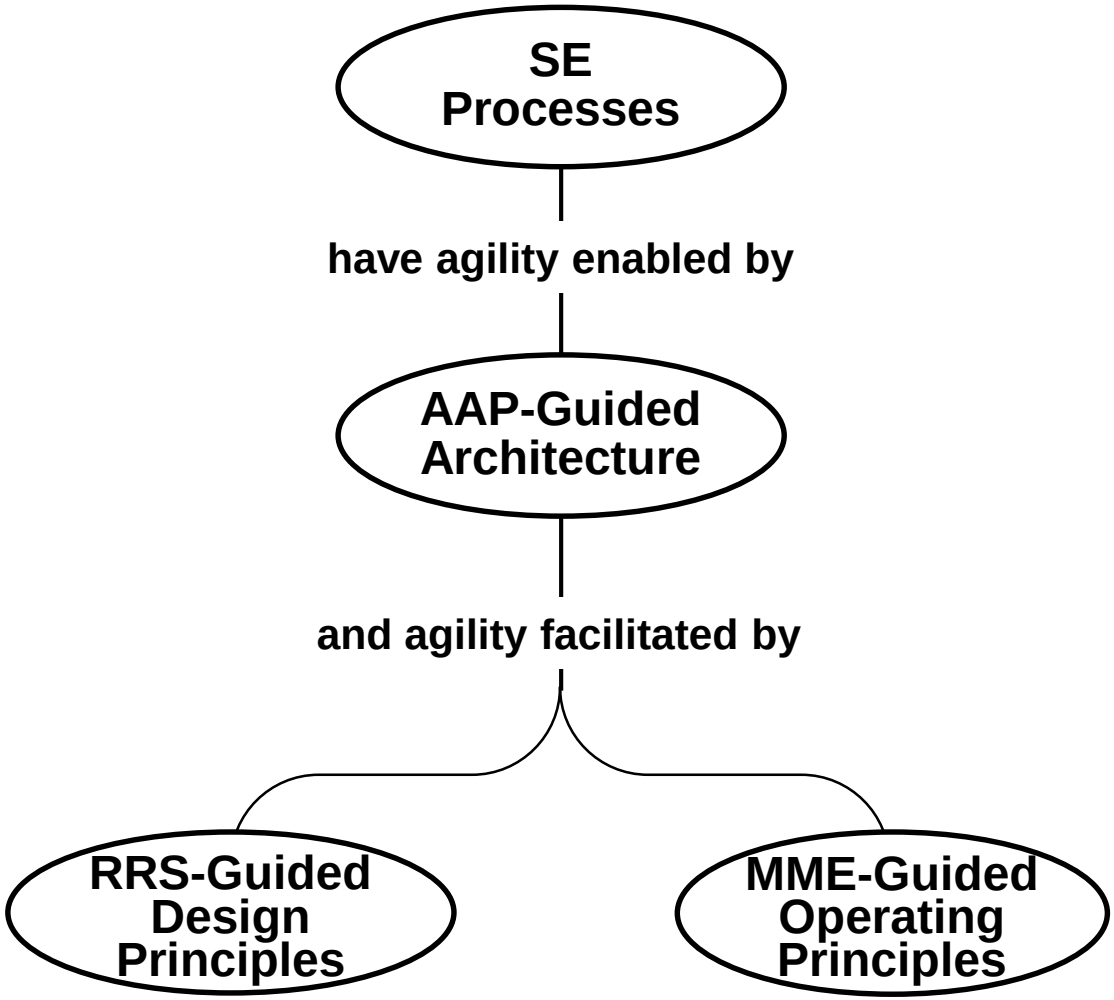
Mitigating (Deciding, Acting)

- Decision making
- Action making
- Action evaluation

Evolving (Improving Above)

- Experimentation
- Evaluation
- Memory

Concept Map


SE Processes
have agility enabled by
AAP-Guided Architecture
and agility facilitated by
RRS-Guided Design Principles
MME-Guided Operating Principles

Examples of Effective Monitoring

Rockwell-Collins:

Product Line Engineering for SW/FW avionics

Lockheed Avionics IFG:

Tailored SAFe for fighter-plane capability evolution

Navy SSC-Pac:

Wave Model for unmanned ground-vehicle technology innovation

Northrop Grumman:

Scrum for evolving joint military-logistics SoS hub

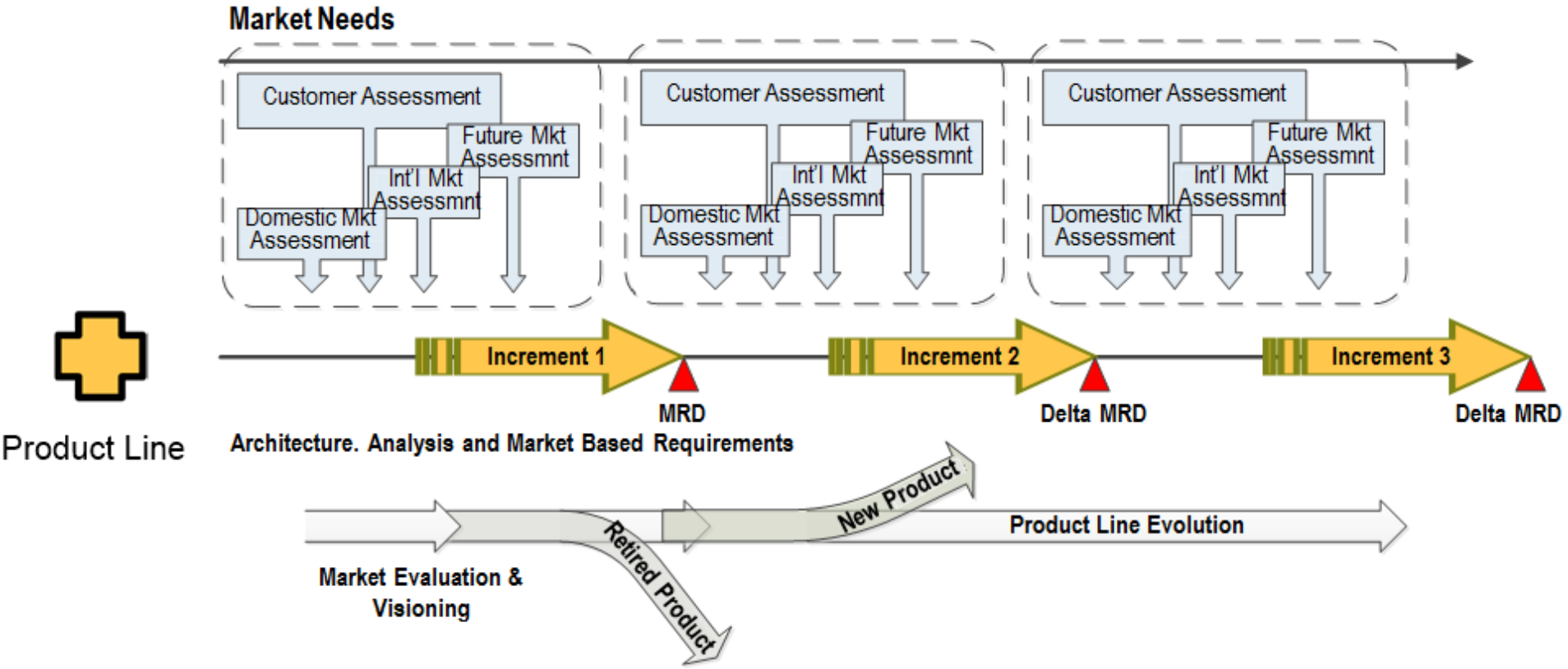
CURVE Environment

- **Capriciousness**
  - **Markets have long/volatile acquisition cycles**
- **Uncertainty**
  - **Feature MIR*s are subjective and not clearly defined**
  - **Ever-moving competitive landscape**
  - **Unknown and Emerging Stakeholders/Users/CONOPs**
- **Risk**
  - **Firmware/Hardware architecture may not be adaptable for future requirements**
  - **Customer expectations exceed technology envelope**
  - **Product development without DoD Sponsorship**
  - **Significant investment with no guarantee for return**
- **Variation**
  - **Market-Based approach tied to evolving industry needs**
- **Evolution**
  - **Customer expectations in SWAP-C* and functionality**

*MIR: Most Important Requirement
*SWAPC: Size, Weight, and Power - Cost

Content: Brendan Getz and Will Hartney, Rockwell-Collins

Rockwell Collins Agile



- Sustained Evaluation of Perceived Market Space – Drives Value to Customers and Shareholders
- Periodically Injects Stakeholder-Accepted Market Requirements into Market Requirements Document (MRD)
- Engineering Review Board Assesses for Impact, Market Value, Cost, and Execution criteria
- Manages Product Line Architectural Tenets: Modularity, Commonality, Scalability, Standardization

Content: Brendan Getz and Will Hartney, Rockwell-Collins

External Awareness: Rockwell-Collins

Market Requirements Document (MRD) Considerations

- Market Evaluation
- Standards
- Surveys
- Site Visits
- Risks
- Commonality
- Opportunities
- Trades

- MRD is roadmap for competitive discrimination.
- Provides awareness of new knowledge/technology needs.
- MRD team owns Product Line roadmap into the future.
- Development team owns current development spec.
- MRD and development proceeds incrementally, but asynchronously.
- SE owner of development spec is member of MRD team.
- Each reviews the other's "plan" for alignment and deferment decisions.
- Generally MRD affects next/future development effort.
- Occasionally MRD has immediate impact on development-in-process.
- MRD is closely held, not shared externally, need-to-know internally.

CURVE Environment

Lockheed Avionic IFG

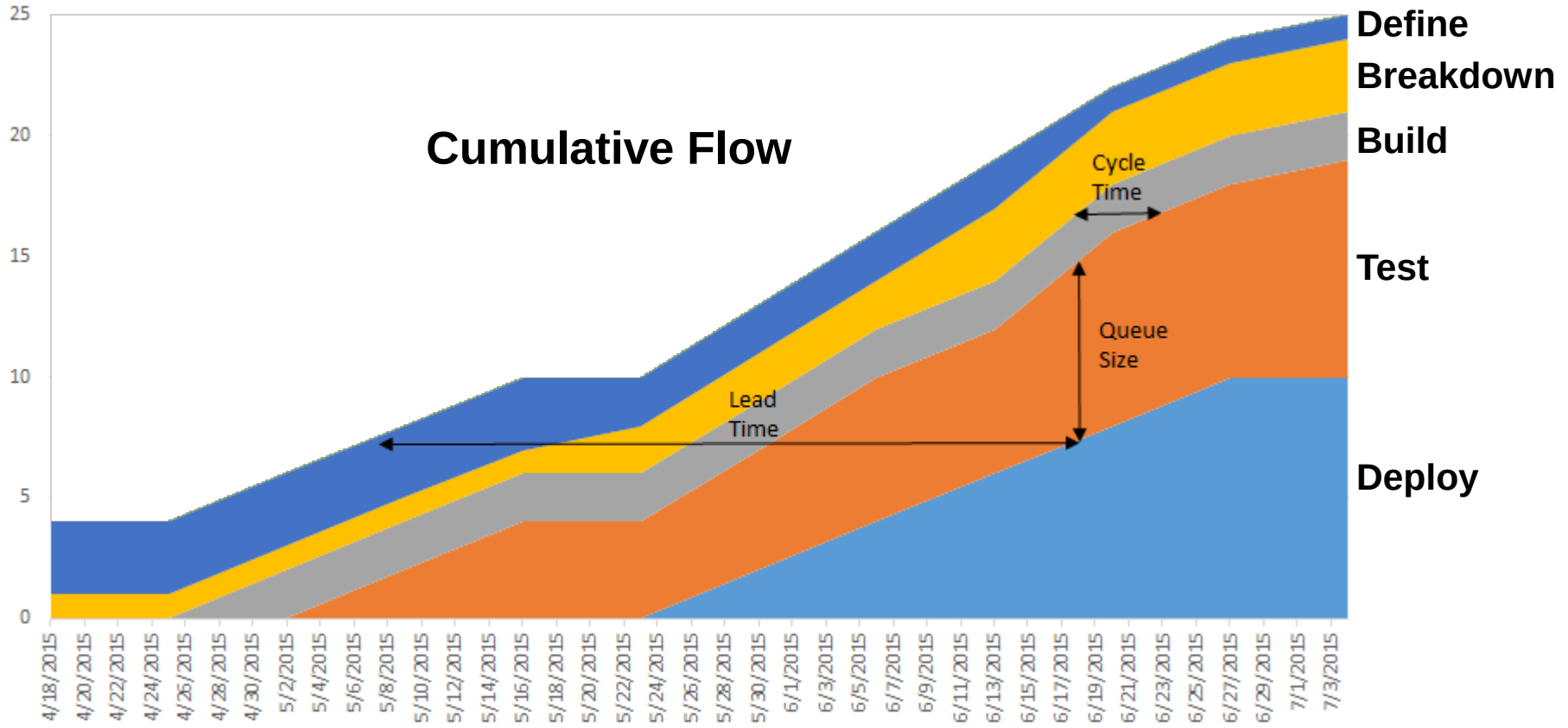
- **Capriciousness: Unknowable Situations**
  - **Urgent Operational Needs**
  - **Diminishing Manufacturing Sources**
- **Uncertainty: Randomness With Unknowable Probabilities**
  - **Funding (e.g. Sequestration)**
  - **Solution Feasibility**
  - **Regression Impacts**
- **Risk: Randomness With Knowable Probabilities**
  - **Competition (e.g. New Trainer) Losses**
  - **Attract/Keep Talent**
  - **Systems Of Systems Requirements Changes**
  - **Schedule/External Stakeholder Timelines (e.g. Certification)**
- **Variation: Knowable Variables And Associated Ranges**
  - **Projects Competing For Bottlenecks (e.g. Ground/Flight Test)**
  - **System Of Systems Integration**
- **Evolution: Gradual Successive Development**
  - **Planned Modernization/Sustainment Increments**
  - **Open Mission Systems Evolution**

Content: Ken Garlington and Gerald Turner, Lockheed Avionics, IFG

Queue Size as Cycle-Time Predictor

Queue size predicts cycle time, providing information faster.
(Don Reinertsen: Principles of Product Development FLOW www.youtube.com/watch?v=rc1MqHsiiKo

Vertical slice on graph measured every two days (shows test example).



Content: Ken Garlington and Gerald Turner, Lockheed Avionics, IFG

Internal Awareness: Lockheed Avionics IFG

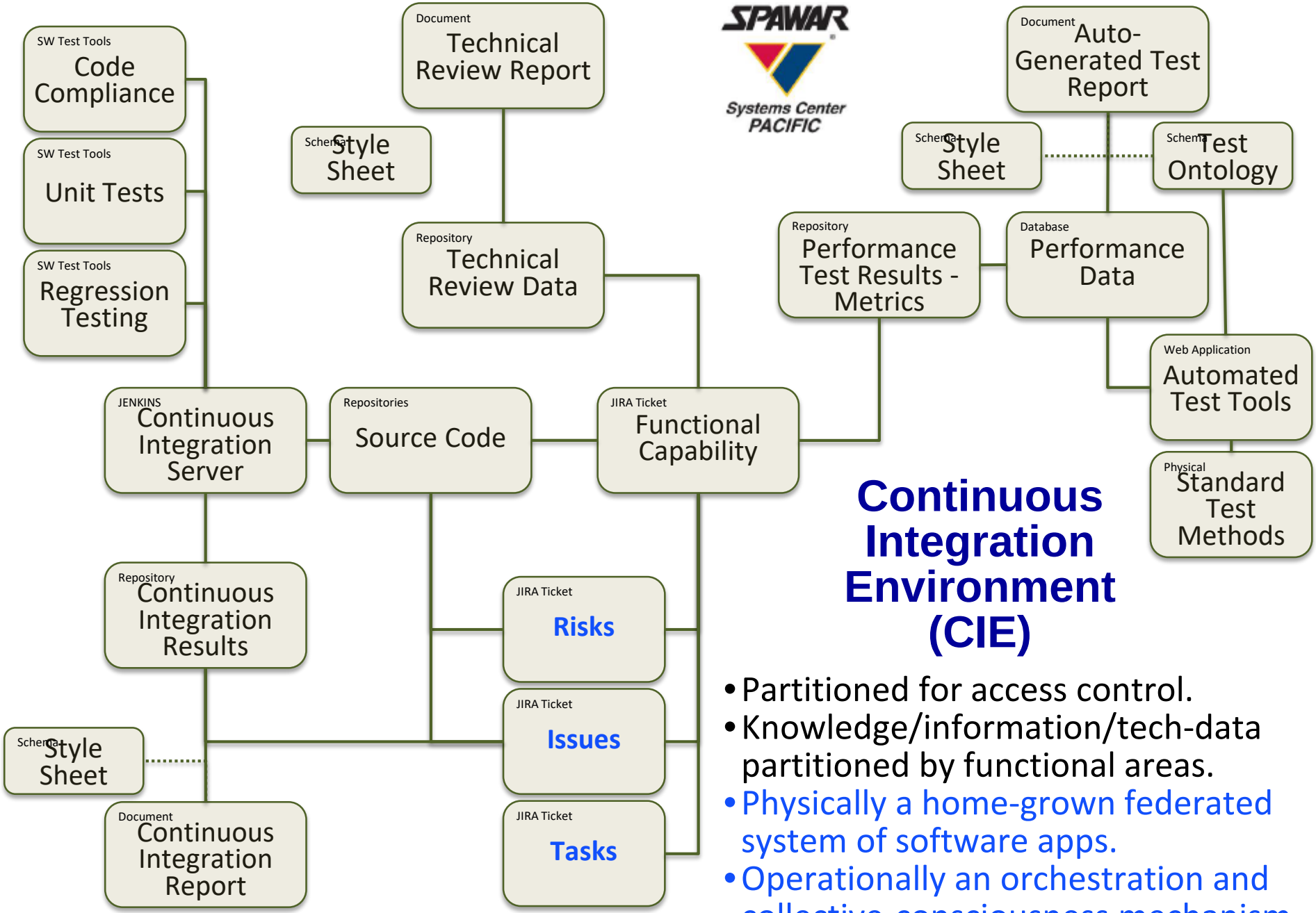
- **Process monitoring instrumentation.**
- **Provides awareness of possible schedule slippage.**
- **Provides scheduling and loading guidance to avoid slippage.**
- **WIP limits can be established for, say, 85% vs 98% of capacity.**
- **Teams don’t get overloaded with work beyond the WIP limit.**
- **Bottlenecks come from building deep queues.**
- **Feedback loop tells the optimum numbers to assign in each band.**

Curve Environment

SPAWAR Systems Center PACIFIC

- **Capriciousness: Unknowable Situations**
  - **Strategic realignment by sponsor**
  - **Engagement and/or availability of personnel & contractors**
- **Uncertainty: Randomness With Unknowable Probabilities**
  - **Feasibility of technical approach and initial designs**
  - **Contracting issues, funding gaps, and budget short falls**
- **Risk: Randomness With Knowable Probabilities**
  - **Failure to meet technical performance measures**
  - **Maturation and integration of required component technologies**
- **Variation: Knowable Variables And Associated Ranges**
  - **Availability of test ranges and test support, and obtaining approvals**
  - **RAM* of vehicle test-beds (vehicle, sensors, computing HW, cables…)**
- **Evolution: Gradual Successive Development**
  - **Technical landscape and insertion of emerging technology**
  - **Programmatic objectives and stakeholder's scope creep**

***RAM: Reliability, Availability, Maintainability**



Continuous Integration Environment (CIE)

- Partitioned for access control.
- Knowledge/information/tech-data partitioned by functional areas.
- Physically a home-grown federated system of software apps.
- Operationally an orchestration and collective-consciousness mechanism.

Internal Awareness: Navy SSC-Pac

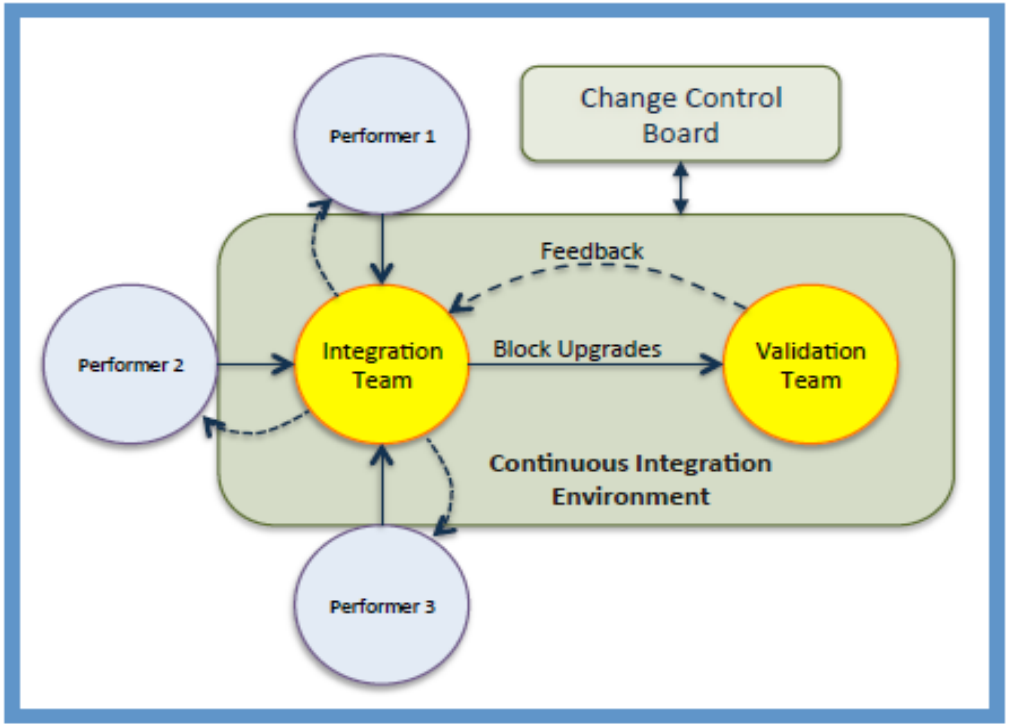
SPAWAR Systems Center PACIFIC

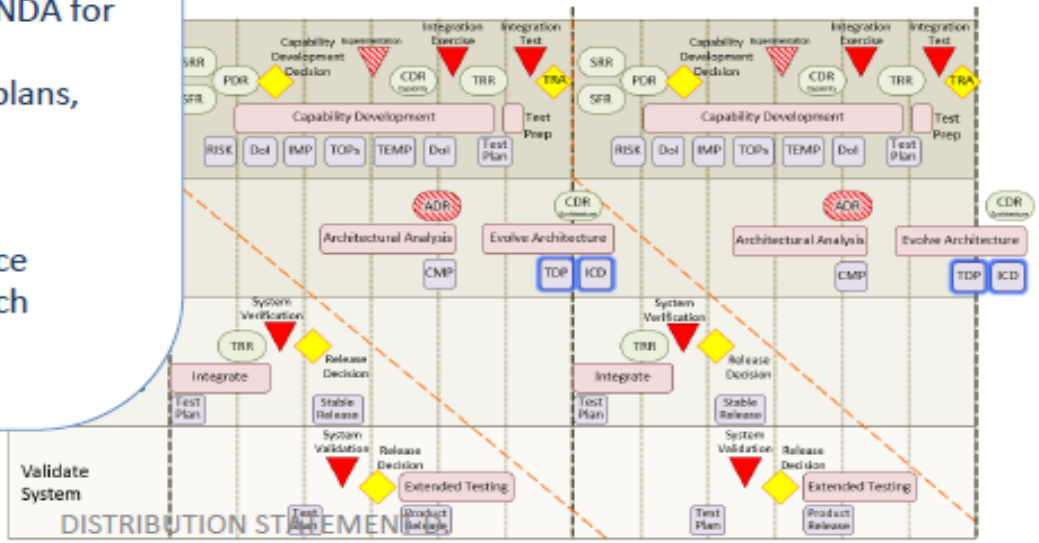
Project Processes:
Information Management Process

Implemented by the Continuous Integration Environment (CIE) that facilitates the continuous integration and feedback between capability developers, integration team, and validation team.

- Best-of-breed collaboration tools (e.g., source code management, issue tracking, and build testing).
- Basis for communication and knowledge transfer between stakeholders.
- Access control and user agreements/NDA for management of Intellectual Property.
- Project documentation (e.g., project plans, bug fixes, change requests, risks, and maintenance reports).

Technical Data Package (TDP) and Interface Control Documentation (ICD) updated each wave.





Content: Chris Scrapper, SSC-Pac

Curve Environment

NORTHROP GRUMMAN

- **Capriciousness: Unknowable Situations**
  - **External data sources change their services**
  - **Number of security vulnerabilities to address varies greatly weekly**
- **Uncertainty: Randomness With Unknowable Probabilities**
  - **Software or Hardware may go end-of-life at any point**
- **Risk: Randomness With Knowable Probabilities**
  - **May not be able to meet 15-day schedule for delivery of security fixes**
- **Variation: Knowable Variables And Associated Ranges**
  - **COTS upgrades deprecate existing interfaces**
- **Evolution: Gradual Successive Development**
  - **As technology changes, the program must port existing capability to new technology**

Areas of Concern

NORTHROP GRUMMAN

- Security
  - Constant barrage of security bulletins.
  - Each bulletin must be assessed for applicability to the program.
  - Patches must be created, tested, and delivered within 15 days. This includes vendor patches as soon as they become available.
  - Patches have to be made to three baselines in most cases: the production baseline, the baseline in test, and the baseline in development.
  - Program allocates velocity each sprint to address security bulletins, but the number of bulletins to address varies.

- Hardware and Software Obsolescence
  - Requires constant monitoring of hardware products for end of life.
  - Requires constant monitoring of COTS/OSS for loss of support/deprecation.
  - Replacement hardware must be on the approved hardware list.
  - Hardware replacement may require architectural changes and requires smoke testing to validate.
  - Analysis of software alternatives also requires adapting the existing code to the replacement product.
  - Requires development velocity and regression testing to validate.

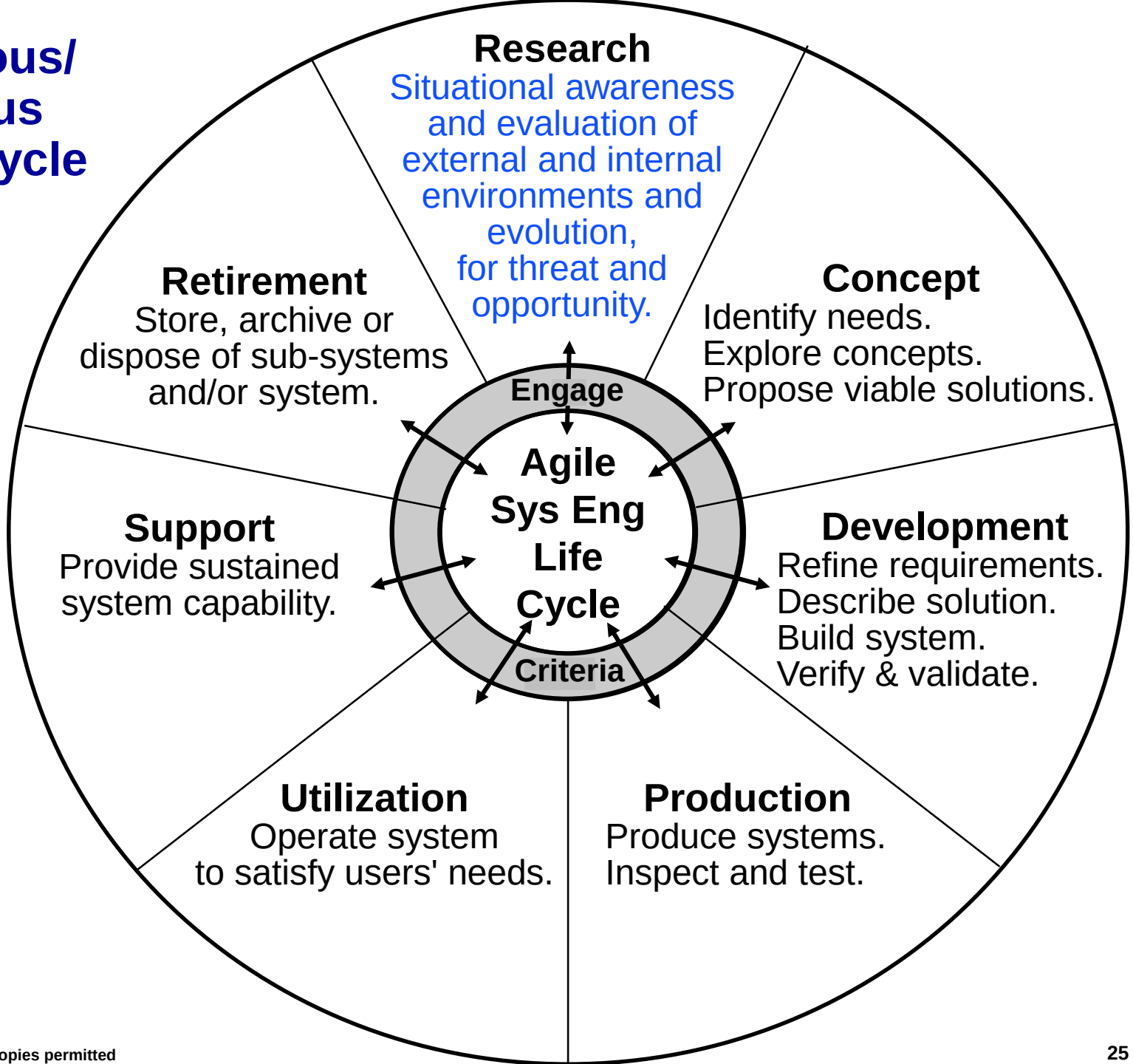
External Awareness: Northrop Grumman

Designated Responsibilities for Situational Awareness

- **Ongoing look ahead for pending security issues.**
  - **Responsibility: Information Security team.**
  - **Lesson-learned to reduce Category-1 (15-day fix) surprise-impact.**
- **Ongoing look ahead for pending COTS and OSS obsolescence**
  - **Responsibility: Systems Engineering team.**
  - **Lesson-learned to eliminated last-minute surprise and churn (each release has 4-7 events to deal with).**
  - **Activity maintains end-of-life spreadsheet for all COTS/OSS, peruses websites for announced upgrade/end-of-life/end-of-support.**
  - **Bi-weekly customer review to advance-plan which release to affect.**

COTS: Common off The Shelf
OSS: Open Systems Software

Asynchronous/ Simultaneous Agile Life-Cycle Framework



Five Necessary Sustaining Responsibilities

- **Resource Mix Evolution – Who (or what process) is responsible for ensuring that existing resources are upgraded, new resources are added, and inadequate resources are removed, in time to satisfy response needs.**
- **Resource Readiness – Who (or what process) is responsible for ensuring that sufficient resources are ready for deployment at unpredictable times.**
- **Situational Awareness – Who (or what process) is responsible for researching, monitoring, and evaluating the operational environment in relationship to situational response ability.**
- **Activity Assembly – Who (or what process) assembles new system configurations when new situations require something different in capability.**
- **Infrastructure Evolution – Who (or what process) is responsible for evolving the passive and active infrastructures as new rules and standards become appropriate to enable next generation capability.**

Download 106 webinar slides: Agile System/Process as Risk Management
Download 105 webinar slides: Agile System/Process Operational Awareness
Download 104 webinar slides: Agile System/Process Engagement Quality
Download 103 webinar slides: Agile System/Process Design Principles
Download 102 webinar slides: Agile System/Process Design Requirements
Download 101 webinar slides: Agile System/Process Architecture Pattern
(updated asynchronously from time-to-time)