CASE Exchange Panel
Incremental/Agile Methods—Fit for Demands of Complex Aerospace Systems?

AIAA Aviation Forum, Denver, CO

6-June-2017, 2:00-5:00pm

Rick Dove

Chair: INCOSE Agile Systems & Systems Engineering Working Group

Background

In The ‘90s we analyzed hundreds of real-world systems that exhibited agility, asking how they did that, and converged on fundamental structural patterns that fit facts.

We are now* analyzing real-world processes that exhibit agility, asking how they do that, and converging on fundamental behavior patterns that fit facts.

***An INCOSE Technical Product project:**
Agile Systems Engineering Life Cycle Model (ASELCM)
(Project details at: www.parshift.com/ASELCM/Home.html)

Is This Your Problem Space?

CURVE

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

Caprice: Unknowable situations.
Unanticipated system-environment change.

Uncertainty: Randomness with unknowable probabilities.
Kinetic and potential forces present in the system.

Risk: Randomness with knowable probabilities.
Relevance of current system-dynamics understanding.

Variation: Knowable variables and associated variance ranges.
Temporal excursions on existing behavior attractor.

Evolution: Gradual successive developments.
Experimentation and natural selection at work.

Incremental/Agile Methods… Fit for Complex Aerospace Systems?

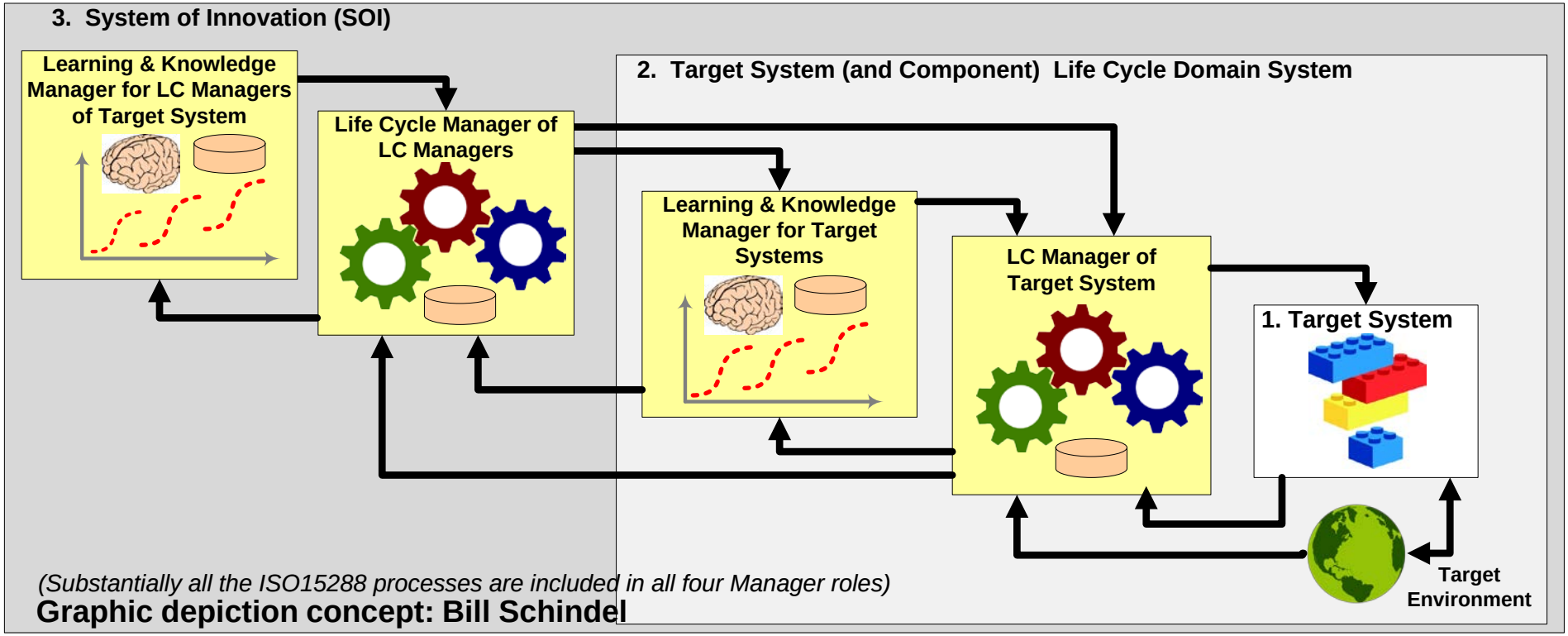
Incremental alone doesn't make a method agile.

Agile software methods:

- **consistent 2-4 week short sprint cadence,**
- **every-sprint deployable features,**
- **dominance of scheduled frequent-increment deliverable dates (at the expense of quality),**
- **iterative feature improvement,**
- **low documentation,**
- **requirement for agile target system (software gets it from O-O development platform),**
- **no recognition of government contract reality and certification time.**

Not compatible with hardware and government contract reality.

But – “underlying” concepts are good:
purposeful learning with facilitated evolution and correction.



S1 product agility: OSA and product-line architectures.

S2 process agility: incremental integration & testing, asynchronous alignment of cross-discipline work increments, preliminary SIL for LVC-like component integration and testing, decoupling development from integration, test, and certification.

S3 innovation agility: awareness of the reality and evolution of the process and product operational problem-space environment, and systemic response.

Addressing the Session Questions

- **Q: Are the experiences of the agile software community the only guide?**
  A: They are a misguiding guide. See first reference at end.
- **Q: Compatible or incompatible with Aero?**
  A: Culturally incompatible, but natural selection will sort that out.
- **Q: What relation to systems complexity?**
  A: Requisite variety.
- **Q: Needed by Aero? Has something changed?**
  A: More CURVEs are being thrown.
- **Q: Already practiced by Aero? Old hat or new?**
  A: Ask Elon Musk.
- **Q: What is it? Examples? Successes, Problems?**
  A: See references at end.
- **Q: When a good fit? When not a good fit?**
  A: If you have a CURVE environment. Yes, if not, No.
- **Q: How are these methods different from agile software approaches?**
  A: Recognition of hardware development reality and gov contracts.
- **Q: Other related questions that need increased exposure?**
  A: Acquisition and contract reform, enabling/facilitating infrastructure.

Relevant References

Dove, R. 2001. *Response Ability – The Language, Structure, and Culture of the Agile Enterprise*. Wiley.

Dove, R., R. LaBarge. 2014. Fundamentals of Agile Systems Engineering – Part 1 and Part 2. International Council on Systems Engineering, International Symposium, Las Vegas, NV, 30Jun-3Jul. www.parshift.com/s/140630IS14-AgileSystemsEngineering-Part1&2.pdf

Schindel, W. and R. Dove. 2016. Introduction to the Agile Systems Engineering Life Cycle MBSE Pattern. Proceedings International Symposium. International Council on Systems Engineering. Edinburgh, Scotland, 18-21 July. www.parshift.com/s/160718IS16-IntroToTheAgileSystemsEngineeringLifeCycleMBSEPattern.pdf

Dove, R., W. Schindel, and C. Scrapper. 2016. Agile Systems Engineering Process Features Collective Culture, Consciousness, and Conscience at SSC Pacific Unmanned Systems Group. Proceedings International Symposium. International Council on Systems Engineering. Edinburgh, Scotland, 18-21 July. www.parshift.com/s/ASELCM-01SSCPac.pdf

Dove, R, W. Schindel. 2017. Case study: agile SE process for centralized SoS sustainment at Northrop Grumman. Proceedings International Symposium. International Council on Systems Engineering. Adelaide, Australia, 17-20 July. www.parshift.com/s/ASELCM-03NGC.pdf

Dove, R., W. Schindel, R. Hartney. 2017. Case Study: Agile Hardware/Firmware/Software Product Line Engineering at Rockwell Collins. Proceedings 11th Annual IEEE International Systems Conference. Montreal, Quebec, Canada, 24-27 April. www.parshift.com/s/ASELCM-02RC.pdf

Dove, R., W. Schindel. 2017. Case Study: Transition to Scaled-Agile Systems Engineering at Lockheed Martin's Integrated Fighter Group. Unpublished working paper. www.parshift.com/s/ASELCM-04LMC.pdf

Dove, R. 2017. Agility in Systems Engineering – Findings From Recent Studies. Unpublished working paper, 15-April. www.parshift.com/s/ASELCM170415-AgilityInSE-Findings.pdf

Agile Systems Engineering Life Cycle Fundamentals Project, Documents at: https://connect.incose.org/ProgramsProjects/aselcm/Pages/Home.aspx, alternatively at www.parshift.com/ASELCM/Home.html

Backup

Agile-System Architecture Pattern (AAP)

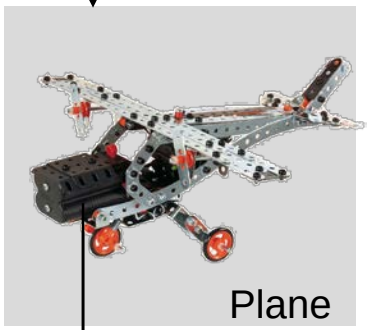
System Response-Construction Kit

Details in www.parshift.com/s/140630IS14-AgileSystemsEngineering-Part1&2.pdf

Integrity Management

- Resource mix evolution
- Resource readiness
- Situational awareness
- Activity assembly
- Infrastructure evolution

Product System Eng.
Retail Distribution Process
Product Manager
Owner/Builder
Product Manager



Active

Infrastructure

Passive

- Sockets — Parts Interconnect Standards
- Signals — Construction Stability
- Security — (None)
- Safety — Harm-Proofing Standards
- Service — Process Rules & ConOps

Rules/Standards

Agility-Enabling Design Principles

Prior Work: see INCOSE Webinar, www.parshift.com/s/AgileSystems-103.pdf

Reusable

- **Encapsulated resources (loosely coupled black-box units)**
- **Facilitated interfacing (easy resource insertion/removal)**
- **Facilitated re-use (support for finding/deploying appropriate resources)**

Reconfigurable

- **Peer-peer interaction (direct communication w/o intermediaries)**
- **Deferred commitment (decisions & fixed bindings at last-responsible-moment)**
- **Distributed control and information (decisions at point of maximum knowledge)**
- **Self organization (relationships and interactions negotiable)**

Scalable

- **Evolving infrastructure standards (resource interface and interaction change)**
- **Redundancy and diversity (duplicate and diverse resource populations)**
- **Elastic capacity (resource populations and functional capacity is variable)**

Agility-Facilitating Operational Principles

Monitoring (observe, orient)

- External awareness (proactive alertness)
- Internal awareness (proactive alertness)
- Sense making (risk & opportunity analysis, trade space analysis)

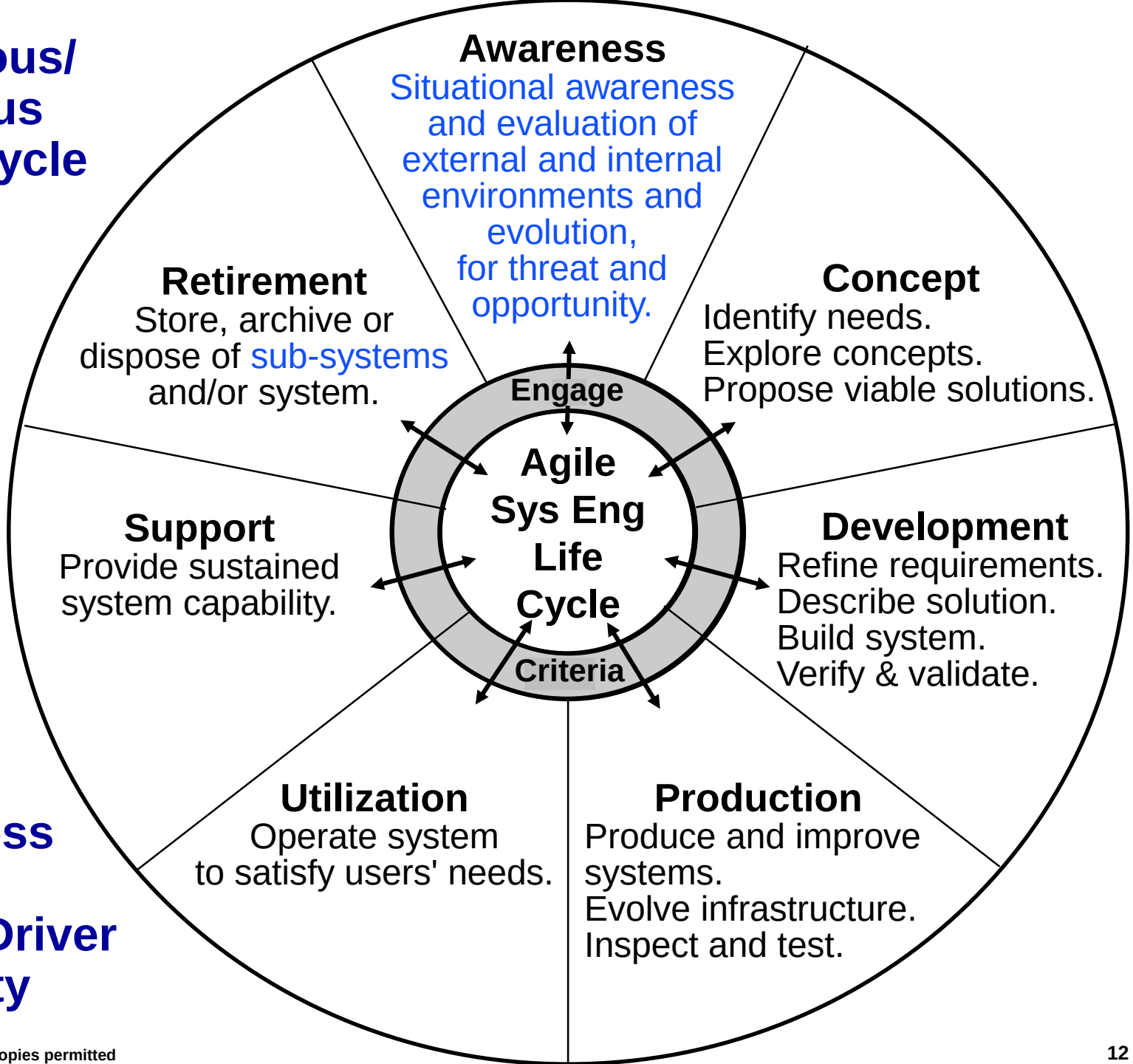
Mitigating (decide, act)

- Decision making (timely, informed)
- Action making (invoke/configure process activity for the situation)
- Action evaluation (validation & verification)

Evolving (improve above with more knowledge and better capability)

- Experimentation (variations on process ConOps)
- Evaluation (internal and external judgement)
- Memory (evolving process ConOps)

Asynchronous/ Simultaneous Agile Life-Cycle Framework



Awareness Stage as Critical Driver of Agility