Agile Systems Engineering Life Cycle Model for Mixed Discipline Engineering

Rick Dove
Paradigm Shift International
dove@parshift.com

Bill Schindel
ICTT System Sciences
schindel@ictt.com
Agile Systems Engineering Life Cycle Model (ASELCM)

An INCOSE Project to...

- Discover generic principles/patterns that are necessary for effective agile systems engineering of SW/FW/HW projects
- Publish informative case studies
- Build evidence-based generic agile-SE life cycle model to inform effective implementation – as an INCOSE Product

And ...

- Provide material for next INCOSE Handbook revision
- Influence published standards evolution
Value Proposition for Agile Systems Engineering

Faster, lower cost system development?
An appealing argument, at the business level.

But to achieve this,
a different value proposition is needed at the engineering level:

Minimization of project risk and rework.
Agile Architecture Pattern (AAP) Enables Agility
Notional Concept: System Response-Construction Kit


- Resource mix evolution
- Resource readiness
- Situational awareness
- Activity assembly
- Infrastructure evolution
Sustaining Agility Requires ...

• Proactive awareness of situations needing responses
• Effective options appropriate for responses
• Assembly of timely responses

Five Agility-Sustaining Responsibilities:
1. Resource Mix Evolution
2. Resource Readiness
3. Situational Awareness
4. Response Assembly
5. Infrastructure Evolution
Two different systems with synergistic dependencies (a first principle)

You can’t have an agile engineering process if it doesn’t engineer an agile product (and vice versa)
ASELCM Project Findings

The IS19 paper discusses:
1. Agile SE Life Cycle Model Framework
2. CURVE Framework Characterizing the Problem Space
3. Operational Principles
4. ASELCM Pattern of Three Concurrent Systems
5. Concept of Information Debt
6. General Agile SE Response Requirements

Above covered in the IS19 paper

Here we add a 7th finding:
7. Continuous Integration Platform
Agile SE
Life Cycle Model
Framework

Asynchronous/Concurrent Stages.
Consistent with
ISO/IEC/IEEE 24748-1:2018

Situational Awareness
Engages System
Evolution Stages/Tasks

Concept
Identify needs.
Explore concepts.
Propose viable solutions.

Development
Refine requirements.
Describe solution.
Build agile system.
Verify & validate.

Production
Produce and evolve systems.
Inspect and test.

Retirement
Store, archive or dispose of sub-systems and/or system.

Support
Provide sustained system capability.

Utilization
Operate system to satisfy users' needs.

Entry/Exit

Criteria

Situational Awareness

Engages System Evolution Stages/Tasks

Asynchronous/Concurrent Stages.
Consistent with ISO/IEC/IEEE 24748-1:2018

Situational Awareness
Engages System Evolution Stages/Tasks
CURVE Framework for Characterizing the Problem Space

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

**Caprice:** unanticipated system-environment change (randomness among unknowable possibilities)

**Uncertainty:** kinetic and potential forces present in the system (randomness among known possibilities with unknowable probabilities)

**Risk:** relevance of current system-dynamics understanding (randomness among known possibilities with knowable probabilities)

**Variation:** temporal excursions on existing behavior attractor (randomness among knowable variables and knowable variance ranges)

**Evolution:** experimentation and natural selection at work (relatively gradual successive developments)
Operational Principles

Sensing (observe, orient)
• External awareness (proactive alertness)
• Internal awareness (proactive alertness)
• Sense making (risk & opportunity analysis, trade space analysis)

Responding (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 culture, response capabilities, and process ConOps)
The enablement of agility

- System-1 is the target system under development.
- System-2 includes the basic systems engineering development and maintenance processes, and their operational domain that produces System-1.
- System-3 is the process improvement system, called the system of innovation that learns, configures, and matures System-2.

The Innovation System is responsible for situational awareness and evolution, the provider of operational agility.
Future costs of a project become committed early by SE decisions. One of the traditional arguments for early stage SE investment.

Will project end with outstanding information debt: a “working system” but an interest penalty caused by shortage of needed information?

SE information must be generated (e.g., reqs, architectures, risk assessments, etc.) early enough in the project.
## General Agile SE Response Requirements

<table>
<thead>
<tr>
<th>Domain</th>
<th>Response Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Proactive</strong></td>
<td></td>
</tr>
</tbody>
</table>
| Creation                | • Opportunity & risk awareness  
                          • Response actions/options                                                |
|                         | • Acculturated memory  
                          • Decisions to act                                                       |
| Improvement             | • Awareness/Sensing  
                          • Memory in culture, options, ConOps                                       |
|                         | • Action/options effectiveness                                                      |
| Migration               | • New fundamentally-different types of opportunities and risks                       |
| Modification (Capability)| • Actions appropriate for needs  
                          • Personnel appropriate for actions                                        |
| **Reactive**            |                                                                                        |
| Correction              | • Insufficient awareness  
                          • Ineffective actions/options                                                |
|                         | • Wrong decisions                                                               |
| Variation               | • Effectiveness of actions/options  
                          • Effectiveness of evaluation                                              |
| Expansion (Capacity)    | • Capacity to handle 1-? actions simultaneously                                     |
| Reconfiguration         | • Elements of an action  
                          • Response managers/engineers                                                |
Continuous Integration Platforms - Context

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.

Integration Platforms for Agile SE employ/enforce AAP Structure

Agile software development processes (silently) rely on AAP platforms.
• 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 platforms.
• Proprietary Product-Line-Engineering employs AAP.
• Proprietary Open System Architecture (OSA) employs AAP.
• Proprietary Live-Virtual-Constructive platforms employ AAP.
Agile Systems Engineering Goals

produce an innovative result,
produce a “success-assured” result,
produce a sustainable result,
rapidly.

Rework is the bane of Rapid.
Continuous Integration Platform

**Need:** Minimize rework (common value across all disciplines).

**Intent:** An agile Continuous Integration Platform (CIP), that enables and facilitates...

- An asynchronous continuous test capability (less rework).
- Early detection of integration issues (less rework).
- WIP feedback demos to users/customers/management (less rework).
- DevOps/DevSecOps collaborative development interaction (less rework).
- Alternative/prototype experimentation (less rework).
- A set-based knowledge-development test stand (less rework).

Less rework is a value common to all engineering disciplines.
Continuous Integration Platform Examples

SpaWar Case Study – two+ unmanned ground vehicles with continuously evolving devices and device wip for multiple simultaneous projects.

Rockwell Case Study – every project has an Integrated Computing Platform – a Rockwell-built scalable circuit card rack with supporting power and cabling that can accommodate multiple evolving circuit boards (FPGA dev boards, prior developed boards, wip boards), and interface with external devices and computers for evolving software and firmware.

Lockheed IFG Case Study – Agile Non-Target Environment (ANTE). Conceptually similar to a Live, Virtual, Constructive (LVC) environment, used to compose an integrated system early. ANTE integrates simulated devices, real devices, lower fidelity COTS proxy devices, IFG software work-in-process, and operators. Subcontractors are required to provide device simulations to ANTE specs.

Northrop Grumman Case Study – a software SoS hub developed with a DevOps, Scrum and SAFe-like operational model on an Eclipse platform, in two-week development and test sprints that produce a user demonstrable wip capability.

VSILs (Virtual System Integration Labs) – so called because they employ a mixed simulation and real device integrated wip system, and/or employ internet connected remote devices and simulations at different physical locations.
INCOSE ASELCM “Product” in Process

First draft for review targeted for end of 2019.

Reviewers will be invited from an international cross section of INCOSE-member organizations.

Principle review questions:
1. is this useful to your organization,
2. what parts are most useful,
3. what would improve usefulness.

Final draft for INCOSE publication targeted for end of 2020.