Today's session will be recorded.

An archive of today's talk will be available at: www.sercuarc.org/serc-talks/

Use the Q&A box to queue questions, reserving the chat box for comments, and questions will be answered during the last 5-10 minutes of the session.

If you are connected via the dial-in information only, please email questions or comments to SERCTalks@stevens.edu.

Any issues? Use the chat feature for any technical difficulties or other comments, or email SERCTalks@stevens.edu.
The Systems Engineering Research Center (SERC) is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology.

Any views, opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense, ASD(R&E), nor the SERC.

No Warranty. This Stevens Institute of Technology Material is furnished on an “as-is” basis. Stevens Institute of Technology makes no warranties of any kind, either expressed or implied, as to any matter including, but not limited to, warranty of fitness for purpose or merchantability, exclusivity, or results obtained from use of the material. Stevens Institute of Technology does not make any warranty of any kind with respect to freedom from patent, trademark, or copyright infringement.

This material has been approved for public release and unlimited distribution.
Can DevOps Practices Be Applied to Cyber-Physical Systems Development?

Dr. Steve Mayner
SAFe Fellow and Principal Consultant
Scaled Agile, Inc.
steve.mayner@scaledagile.com
The world’s leading framework for enterprise agility.

SAFe® for Lean Enterprises is a knowledge base of proven, integrated principles, practices, and competencies for Lean, Agile, and DevOps.

http://www.scaledagileframework.com/
Agencies are using SAFe for Lean-Agile transformation
Can Agile work with major cyber-physical systems?

The question is... how do you do “continuous iterative development” with physical systems in government?
Process guidance for building systems of systems
Ten practices for building big things using Lean and Agile

- Visualize and organize around the value stream
- Apply multiple horizons of planning
- Base decisions on objective evidence of system state and performance
- Architect for scale, modularity, and serviceability
- Manage the supply chain with systems-of-systems thinking
- Iterate and reduce batch size
- Establish cadence and synchronization
- Employ “continu-ish integration
- Be test driven
- Continually address compliance concerns
Visualize and organize around the value stream
Instead of organizing around vertical silos with infinite hand-offs, organize in cross-domain teams-of-teams that have all the disciplines needed to deliver value.

The solution can be a fully deployable capability (portable tactical radio) or a component to a larger solution (communications module on an F-22)
ARTs are long-lived, cross-functional teams of agile teams

- Team-of-teams (50-125 people) that create a component or capability
- Can independently explore, specify, build, integrate, test, deploy, and (possibly) release
Large programs require multiple ARTs and Suppliers

- Consumer solutions
- Fleet management
- Parcel solutions
- Back-end financial
- Autonomous vehicle platform
- Autonomous sensors

- Order mgmt
- Customer app
- Fleet mgmt
- Vendor mgmt
- Finance
- Autonomous Driving
- Sensors/cameras

© Scaled Agile, Inc.
Apply multiple horizons of planning
Planning occurs at multiple levels

- Outer levels less defined, only high level commitments
- Inner levels more understood, detailed, commitments more specific
Define intent and roadmap to move from as-is to to-be

- Evolve the intent and roadmap based on new learning

Specifications → Roadmap → TO-BE → AS-IS → Solution

New knowledge
- Mission changes
- Technical discoveries
- User/operator feedback

Increment goals
- New Features
- Experiments

Evolve together
Base decisions on objective evidence of system state and performance
Deliver working features every Program Increment

- Working features provide better evidence of progress
- Enables fast feedback throughout the lifecycle
- Faults can be found and corrected in small batches while the cost of change is low

\[ \dot{H} = \sum_{n=1}^{N} \frac{\hat{p}_n^2}{2m_n} + V(x_1, x_2, \ldots, x_N) \]
\[ = -\frac{\hbar^2}{2} \sum_{n=1}^{N} \frac{1}{m_n} \frac{\partial^2}{\partial x_n^2} + V(x_1, x_2, \ldots, x_N) \]
Apply objective Milestones

Program Increment (PI) System Demos are orchestrated to deliver objective progress, product, and process metrics.
Architect for scale, modularity, release-ability, and serviceability
Architect to facilitate the CD Pipeline

Continuous Delivery Pipeline

- Architect for rapid learning
- Engage in MVP / MMF definition
- Support innovation accounting
- Champion Design Thinking and Lean UX behaviors

- Architect for ‘continuish’ integration
  - Automate compliance
    - V&V, traceability
    - Static analysis & scanning
    - Automated approvals

- Full pipeline automation – CI/CD
  - Decoupled deploy and release
  - Pervasive telemetry (about the system)
  - Architect for recoverability, fault tolerance

- Enable continuous release
  - Feature toggles / Canary Releases
  - Decoupled releases / streamlets
  - Designed for operations/servicing
  - Value-centric telemetry (about the usage)
Decouple components of large solutions

- Define independent modules that communicate through well-defined interfaces
- Enable teams and ARTs to independently build, test, deploy, and even release

Decoupling considerations

- What parts will be serviced separately?
- How will the parts be manufactured, integrated?
- How are manufacturing costs impacted?
- What is the communication strategy?
Decoupling enables release-ability for large systems

- Different parts of the solution require different release strategies
- Architect the solution to enable the various strategies and to shift them over time based on business demand
- Isolate structural complexity
Manage the supply chain with system-of-systems thinking
Lean-Agile supply chain model at scale

- Entire program aligned on a common cadence
- Supports massive continuous integration and alignment across internal and external suppliers
Solution requires continuous delivery across entire supply chain

Aggregate Solution Intents to support development and compliance

Align everyone on a common cadence
Iterate and reduce batch size
Fast feedback and learning on working features
The importance of small batches

- Large batch sizes increase variability
- High utilization increases variability
- Severe project slippage is the most likely result

- Small batches go through the system faster, with lower variability
- The most important batch is the handoff batch

Principles of Product Development Flow, Reinertsen, Don
Implementing Lean Software Development, Poppendieck, Mary
Establish cadence and synchronization
Cadence and synchronization

**Cadence**
- Converts unpredictable events into predictable ones and lowers cost
- Makes waiting times for new work predictable
- Supports regular planning and cross-functional coordination
- Limits batch sizes to a single interval
- Controls injection of new work
- Provides scheduled integration points

**Synchronization**
- Causes multiple events to happen at the same time
- Facilitates cross-functional tradeoffs
- Provides routine dependency management
- Supports full system and integration and assessment
- Provides multiple feedback perspectives

*Note: Delivering on cadence requires scope or capacity margin*

*Note: To work effectively, design cycles must be synchronized*

---

*Principles of Product Development Flow, Don Reinertsen*
*The Lean Machine, Dantar Ootserwall*
Synchronize to assure delivery

This system is iterating

Allows alignment, integration, and dependency management at multiple levels of aggregation
Apply ‘continuish integration’
Support frequent integration and testing in large systems

- Frequent integration and testing is to provide frequent feedback
- Do not let small changes sit idle, find a way to integrate with others
- Trade-offs are inevitable in terms of:
  - Frequency of integration
  - Depth of integration
  - Fidelity of feedback
Use proxies to provide early, end-to-end system integration

- Solution integration is realized as a combination of real subsystems and proxies of various complexity
- Interfaces allows components to mature independently
- Bring in manufacturing early; validate with actual production facilities
- The cost of integrating an actual subsystem can be too high, yet substituting a proxy may provide no useful information – it’s a trade-off
Align functional and physical roadmaps

- Hardware teams already create proxies for their learning – coordinate them

- Strive for early, end-to-end solution mockup that matures in fidelity over time

- Hardware teams responsible for supporting incremental demonstrations
Be test driven
Traditional testing (V-Model) delays feedback
Shift testing left for fast and continuous feedback

Build hardware proxies early on and mature them over time (ex. 3D printed parts)
Continuously address compliance concerns
Continuous compliance and a Lean QMS

Build in compliance incrementally

Build in compliance w/ automation

Include compliance in the value stream

Apply continuous verification & validation

- Functional test
- Security test
- Performance test
- QA test
- Compliance test

- Functional test
- Security test
- Performance test
- QA test
- Compliance test

Continuous verification & validation and traceability integration for efficient and effective compliance management.
Wrap-up
Thank you!

steve.mayner@scaledagile.com
@stevemayner
linkedin.com/in/stevemayner
UPCOMING TALKS:
“Distributed Autonomy” Series

“What’s Really Distributed in Distributed Autonomy?”
August 7, 2019 | 1:00 PM ET
Dr. Azad Madni, University of Southern California
REGISTER NOW

CONTACT
Editor-in-Chief: Dr. Barry Boehm, University of Southern California – boehm@usc.edu
Webinar Coordinator: Ms. Mimi Marcus, Stevens Institute of Technology – mmarcus@stevens.edu

Please visit the SERC Talks page to register and for more information and updates.
Thank you for joining us!

Please check back on the SERC website for today's recording and future SERC Talks information!

Subscribe and follow SERC on our social channels: