SAND2023-05447C

Sandia National Laboratories

Using Scrum for a Scrum Master Team

Justin Lamb, Senior Member of Technical Staff

Sandia National Laboratories

NLIT Summit, Milwaukee, WI

June 29th, 2023 - Session 5: 1:00-1:40

U.S. DEPARTMENT OF ENERGY | NNSA

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Agenda

- Introduction
- Sandia Computational Simulation (CompSim) Group
  - CompSim Codes
  - CompSim Scrum Masters
- Agile Improvement Initiative
- Scrum Team of Scrum Masters
- Conclusion

Introduction

Introduction

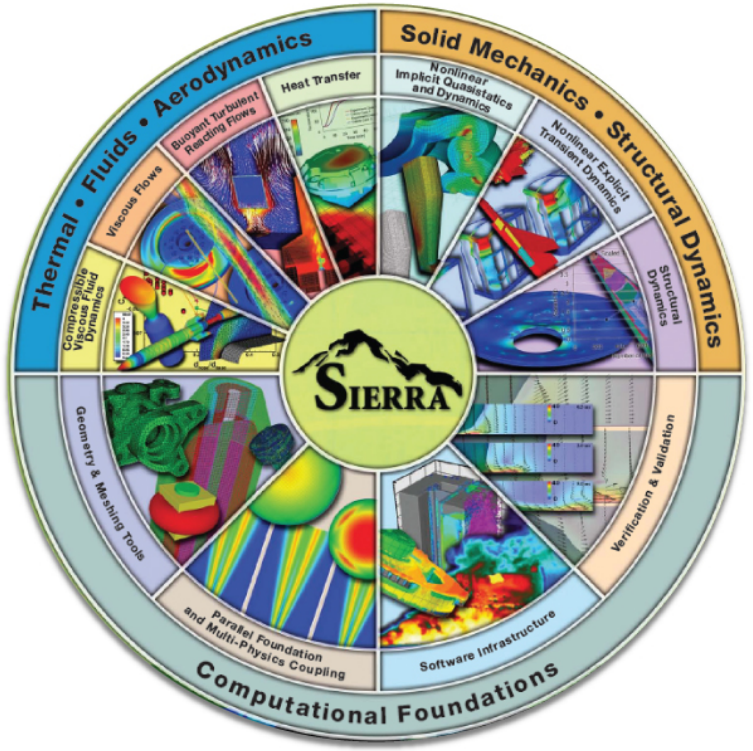
- Personal Background
- Organizational Agile leadership
  - Limited Agile goals/direction
  - Agile improvement initiative
  - Organizational change management can be difficult
  - Scrum Masters need focus and structure
    - Transparency and accountability
- Details, observations, and lessons learned from the early stages of an ongoing effort

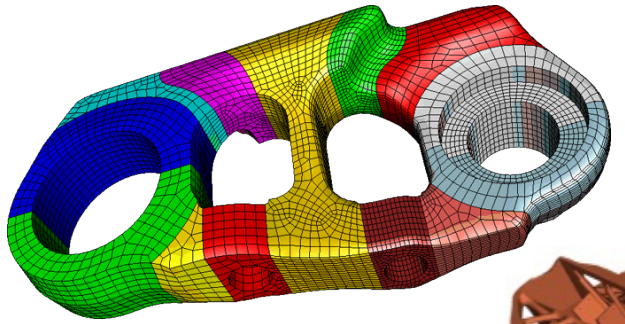
COMPSIM

CompSim Overview

CompSim Teams

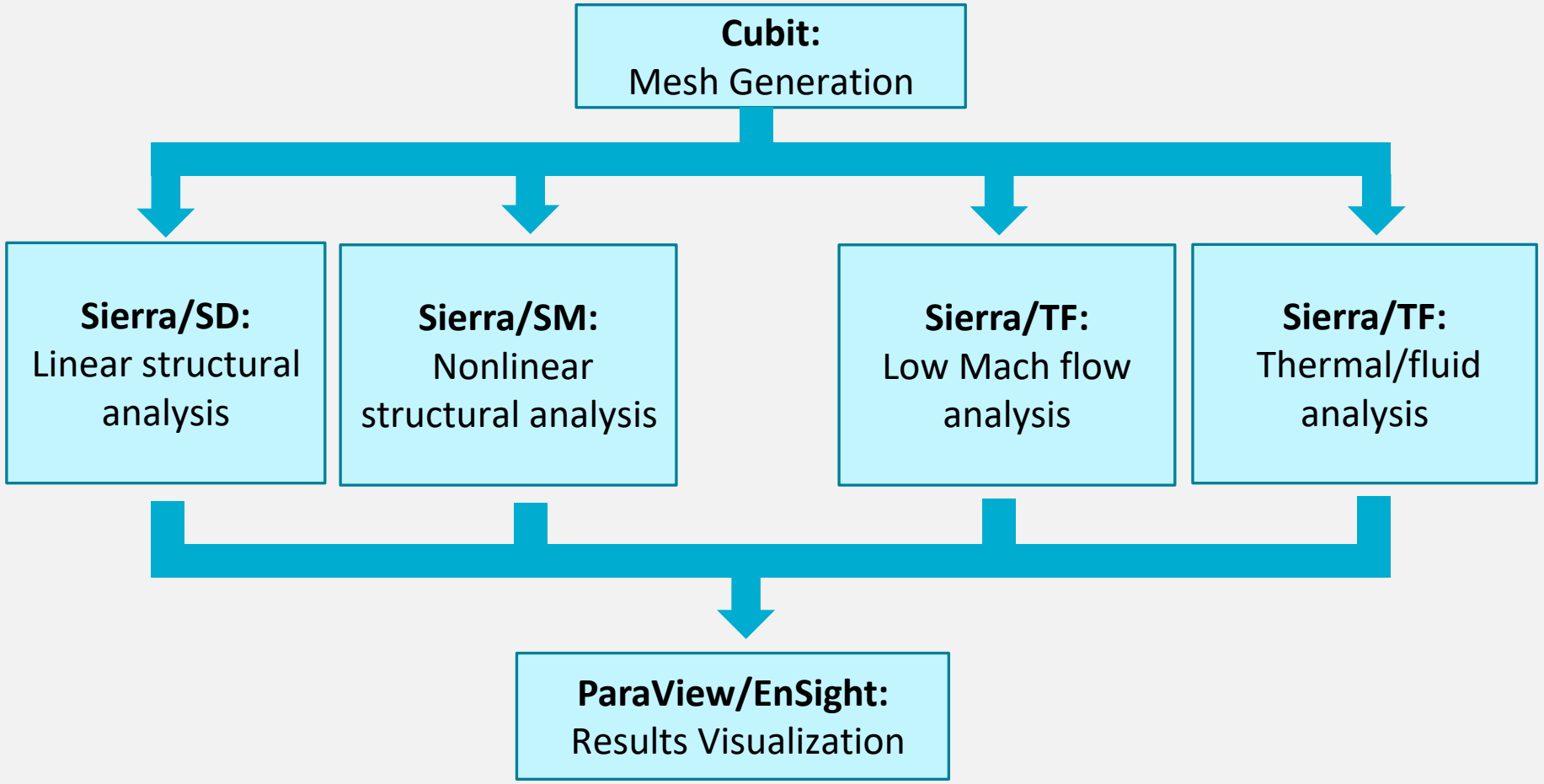
- **Sierra**
  - Sierra is an engineering mechanics simulation code suite that includes coupled simulation capabilities for thermal, fluid, aerodynamics, solid mechanics, and structural dynamics.
  - **Physics applications:** Sierra/TF, Sierra/SD, Sierra/SM, & SPARC
  - **Support applications/tools/utilities:** DevOps & STK
  - **Design codes:** NGS, FuSED, & Plato
- **Cubit**
  - Cubit is a full-featured software toolkit for geometry preparation and robust generation of two-dimensional and three-dimensional finite element meshes (grids)
- **SAW**
  - The Sandia Analysis Workbench (SAW) is a family of software applications that boost productivity and quality by making modeling and simulation easier with capabilities that include workflow & data management, model building, and job submission.





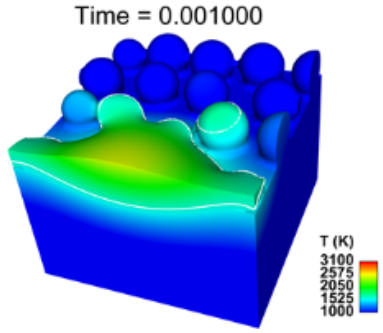
CompSim Workflow

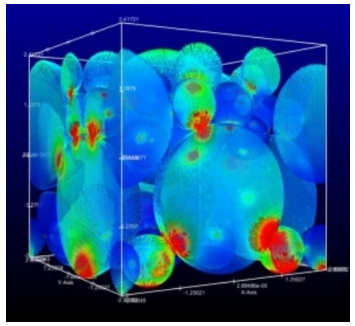
SAW (Sandia Analysis Workbench):
Workflow and User Interface



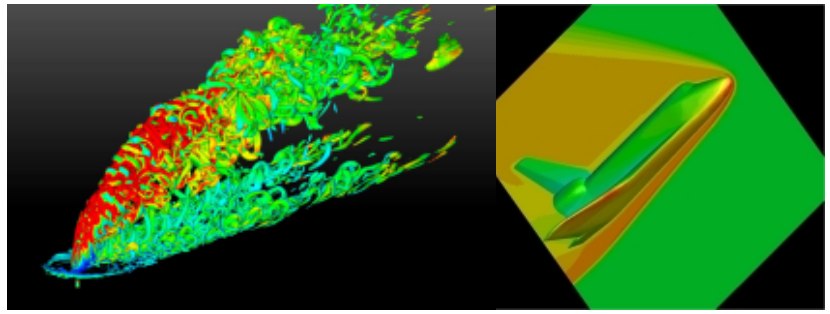
Engineering Analysis Examples

Manufacturing and Fluid Flows

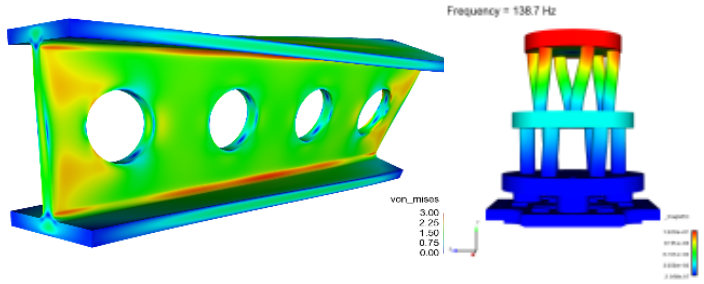




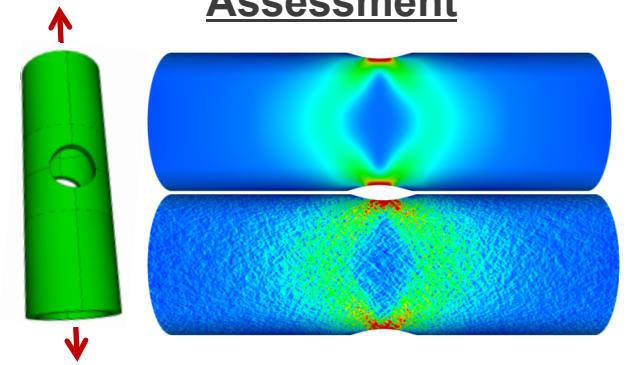
Aeroscience and Flight Phenomena



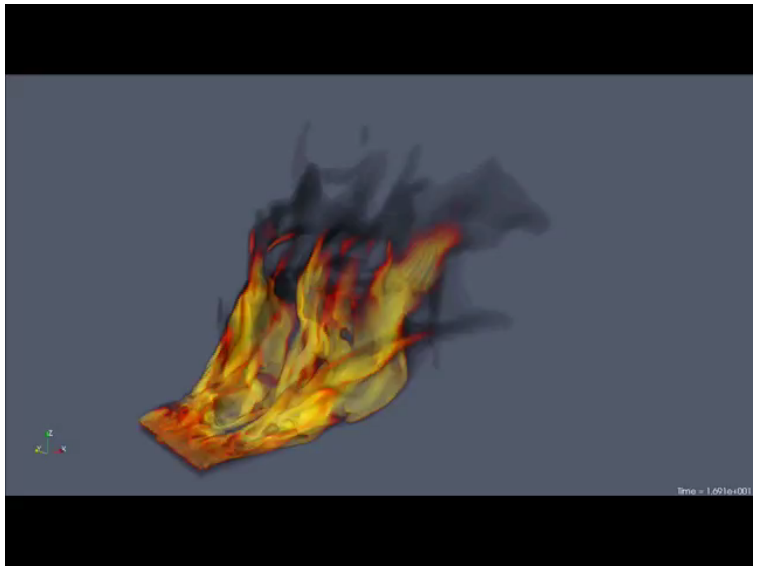
Structural Dynamics



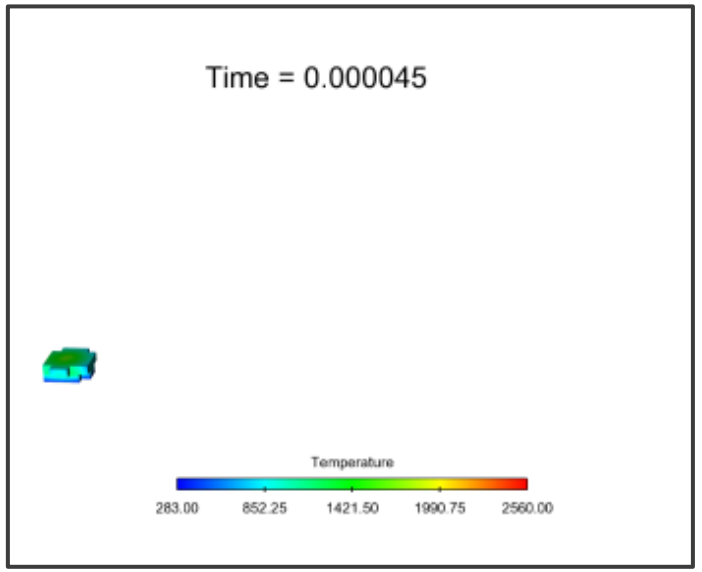
Solid Mechanics and Failure Assessment



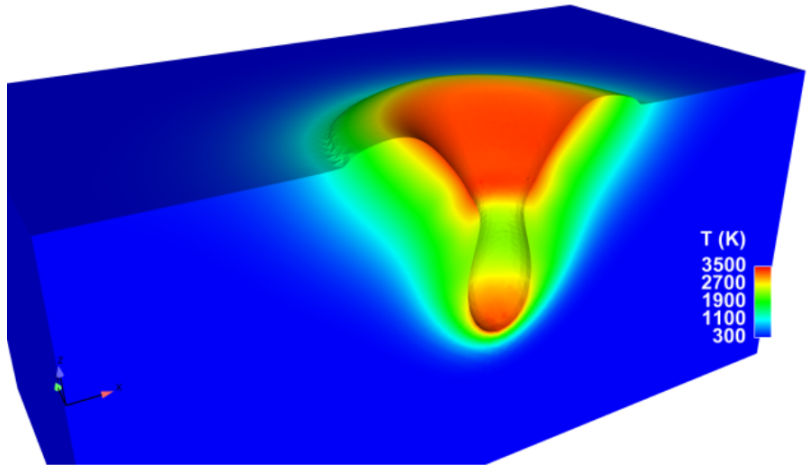
Thermal/Fluid Analysis Examples



Fire simulation



Additive Manufacturing



Laser Heating Melt Pool

Organizational Agile History

- Sierra was established in 1997 with Scrum being adopted around 10 years later with the assistance of a professional Agile coach, Jim York
- Product Owner Leadership Team (POLT) established after ~10 years of using Scrum
  - CompSim Scrum Master
  - CompSim Product Owner
  - CompSim Chief Architect
- With ~90-100 staff spread across 10 teams, we have not adopted any formal Agile frameworks geared towards larger organizations. Informal “Scrum of Scrums”
  - Team independence
  - Overhead costs
- Teams value their autonomy to choose their ideal Agile workflow
  - ‘Scrumban’
  - XP practices

Current state: 10 individual teams with their own set of practices/guidelines with limited organizational direction on improving scrum and software engineering practices.

Jim York’s Website: https://www.foxhedgeltd.com/

CompSim Scrum Masters

Who are the CompSim Scrum Masters?

- Six Scrum Masters within the CompSim Group
- Diversity in Scrum Master's other roles and organizations at Sandia
  - CompSim technical user support
  - CompSim analyst users
  - Full-time Scrum Masters w/ software engineering backgrounds

Scrum Master Team History

- The organization has a CompSim Scrum Master role
  - Organizational Agile coach and leads the CompSim Scrum Masters
- **CompSim Scrum Master community of practice:** Informal weekly team meetings with no team structure or long-term planning

2017 **Original weekly meeting format**

Weekly meeting: Tactical round-robin discussions with occasional strategic or learning agenda items

2020 **Updated weekly meeting format**

Strategic Meeting: Discuss long-term efforts and goals

Tactical Meeting: Discuss issues facing our teams right now

Learning Meeting: Share knowledge and teach fellow SMs

- **Team Reviews:** Managers and Product Owners invited as stakeholders

Outcomes: Follow-through on long-term efforts was hit or miss. Greater emphasis on short-term tactical needs with the team being effective at addressing them.

Agile Improvement Initiative

Initiative Vision

- My manager presented his vision of the Agile improvement initiative to the CompSim Scrum Masters
  - CompSim staff member
  - Agile/Scrum advocate
- Organizational buy-in crucial at all levels
  - Starts with the Scrum Masters
- Reducing maintenance and support costs by delivering good code
- Producing new capabilities faster

Bad Code Smells

"I wrote this code over the weekend, and it's working." – watch out for "I"

Let's just deploy it, and we'll fix the bugs that are encountered. – crowdsourcing versus TDD?

50% or more of team's capacity is spent on "not new" functionality

Users aren't happy, but they might not vocalize it. If they do, they are labeled as "squeaky wheels". Does your team embrace "squeakiness" or does it make fun of it?

"I am the expert in the field, so I must write the code." – separate solution development and code development

Cost of implementing similar capabilities grows as code base grows. That is, developers were faster with less code.

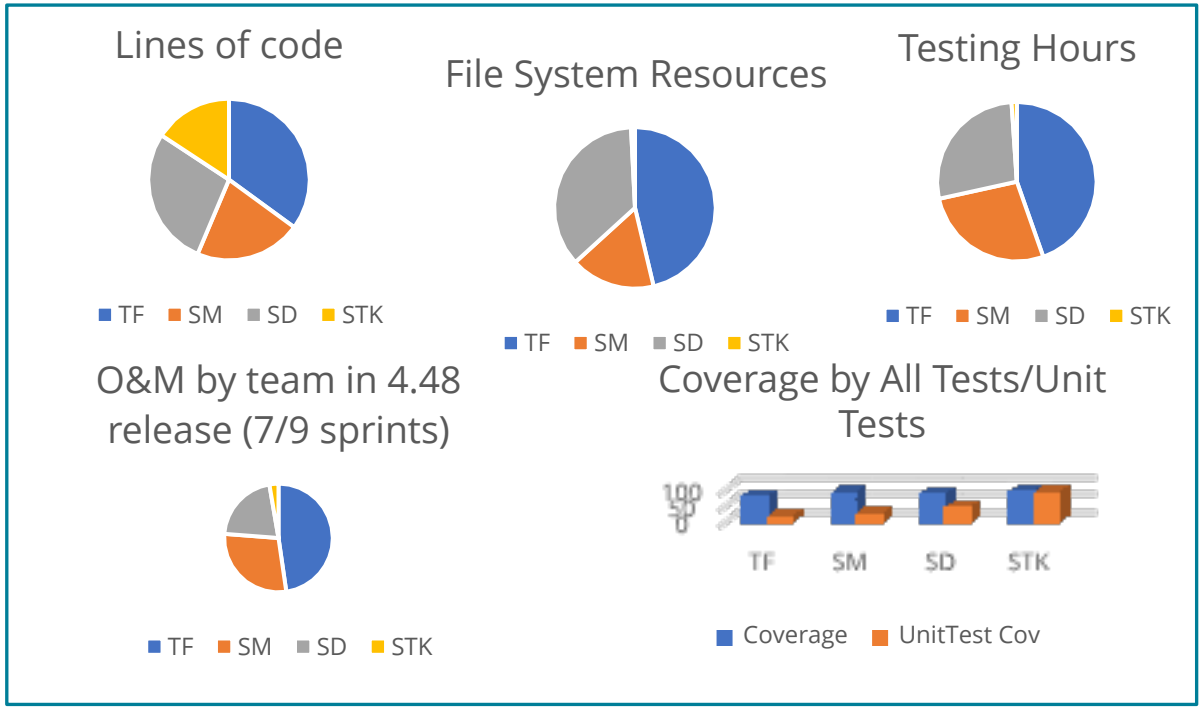
Trying to produce many "ok" capabilities versus working on less, but more solid, capabilities

Out of control spiral: debt is so large that we can't afford to pay down debt because we'll produce less. – Long term sustainability or short term success

Watch out for elegant code. – ever notice a developer saying "You should see my code. So simple."

Case Study – Sierra STK Team

- The STK Team learned about clean code practices
  - Implemented clean code practices in the team's definition of done (and coding standards)
- What happened after several years?
  - Code was higher quality, less support costs. Adding new **good** code became faster!



What can we do to Enable and Promote Good Code?

- Incorporate technical practices into definition of done
- Test-Driven Development (TDD)
  - If legacy, refactor to be unit testable
  - Monitor unit test coverage
- Pair programming and code reviews
- Measure Maintenance and Support (M&S) costs over time
- Readability is important. Reading to writing ratio when developing code is 10:1.
- Visible team practices and standards
  - Six line functions
  - No more than 3 parameters per function
  - …

What is the Impact?

Substantially increase the investment towards Sandia's core mission, annually

Initiative Kickoff Retreat

Full Day Kickoff Retreat

- Jim York led planning event
  - Get CompSim Scrum Masters' buy-in for the initiative and get team momentum on the initiative
- **Team building activity:** Five Dysfunctions of a Team
  1. Avoidance of Accountability
  2. Team cohesion
- Brainstorm initiative objectives
  - Follow-on user story to finalize objectives using the newly adopted Scrum framework

Objective #1: Modsim Enables credible program decision making

Objective #2: ModSim analysis cycle time is reduced

Objective #3: CompSim sustainably deploys new capabilities for production use

Implementing Scrum

How can we best provide organizational Agile leadership?

- Delivering on the Agile improvement initiative will likely be difficult. The Scrum Masters need to collaborate effectively to ensure success.
- The Scrum Masters undertook a self-evaluation to gauge our current capacity to deliver on the initiative. We asked ourselves several questions that would help us move forward as a group:
  1. Can we be accountable to each other within an informal, unstructured community of practice?
  2. Does the community of practice have an adequate track record of following through on long-term efforts with our current working arrangement?
  3. Will the initiative roadmap be simple enough that any Scrum Master will be confident on the next items to work on? Can we each independently define the work and deliver on it?
  4. Will each of us be able to commit to working on organizational efforts that don't have immediate impacts to the teams we serve?

Conclusion: We should utilize Scrum!

Assigning Roles

- **Stakeholders?**
  - CompSim Scrum Master as the primary stakeholder
  - Product Owners & managers
- **Product Owner?**
  - I took on the role of the Scrum Master Team Product Owner
  - Product Owner comes from my manager's department
  - Years of experience within the CompSim Group
- **Scrum Master?**
  - Debated within the team on making this a rotating position
  - Selected Scrum Master who is passionate about this idea
  - Always has time and availability to commit to this role
  - Years of experience within the CompSim Group

Product Backlog

- The team had a few ideas for how we would structure our product backlog:
  1. The entire scope of the Scrum Masters roles and responsibilities would be captured and tracked in the team's product backlog on Jira
  2. The product backlog would only include work items that came out of the Agile improvement initiative
  3. The team would capture the work items in the following epics: CompSim Agile needs, tactical improvements, organizational impediments, and Agile improvement initiative
- The team decided to go with the third option. The goal of the third option was to have a product backlog that had some level of scope beyond just an individual team

Adjustments for a Unique Scrum Team

Sprint Review: Invited audience to sprint reviews with attendees having direct interest in the completed user stories.

- Small sprint backlog with some sprints having limited relevance to a wider audience
- Maintain quarterly reviews to socialize a summary of our efforts to the larger community

Daily Scrum: Held twice per week due to a limited team capacity and the corresponding smaller sprint backlog.

Retrospective: Preparation and facilitation of this meeting on a rotating basis among Scrum Master Team members.

Planning: Short planning meeting with fewer items on the sprint backlog.

Sprint Length: Three weeks sprints to match the CompSim teams.

- Considered longer sprints to boost the team capacity for any given sprint
- Offset from the starting/ending week of the CompSim teams we serve

- Concerns with increased overhead with using Scrum affected our adjustments for the daily scrum, retrospective, and planning.

Using Scrum

Observations & Benefits

- First sprint planning meeting
  - Discovered quickly that more detail was needed in the user stories to reduce ambiguity on how to complete the work
- Daily Scrum parking lots
  - Much of the initial work came from lengthy parking lot discussions where a majority of the team worked through user stories together
- Scrum provided persistent focus for the team's work
- Productive team meetings
  - More effective discussions
  - Targeted discussions on most important work
- Valuable feedback from stakeholders during sprint review meetings
  - This level of targeted feedback was missing in the past
- Improved team capability of onboarding new Scrum Masters

Challenges and Lessons Learned

- Measuring success
  - How are we going to measure the value we provide to our teams through this new Scrum arrangement? Is it worth the time commitment?
- Stakeholder engagement
  - Balancing competing priorities and needs between our initial stakeholders
- Unclear team direction/product goal
  - Abstract nature of product backlog items
  - Roadmap adjustments in the middle of the sprint after working on sprint user stories
- Loss of tactical focus
  - Feedback during team's first retrospective
  - Re-introduce weekly tactical meetings to team member's schedules.
- Tertiary priority for most Scrum Masters
  - Tough to work outside of team meetings/parking lots
  - Attendance at scrum ceremonies could be lacking at times
- A team made up of Scrum Masters introduced unique challenges. We have to check our behaviors and take off our 'Scrum Master hats'.

What is our Current Focus?

- Determine the appropriate level of detail to work towards for our initiative. Make progress on user stories to fully define the initiative
- Balance our commitment on this effort with our other priorities
- Obtain broader organizational buy-in for the Scrum improvement initiative

Conclusion

Conclusion

- CompSim has limited organizational direction on what to strive for within the Agile/Scrum sphere. The organization needs the Scrum Masters to step up and provide additional leadership
- Scrum Master Team has had variable success in the past with long-term strategic efforts. Therefore, we implemented Scrum as a way improve focus on the group's long term needs
- Scrum can be difficult to implement even for Scrum Masters; however, we are well equipped and willing to make rapid adjustments
- Commitment and accountability could be improved
  - These qualities should improve as the team gains more clarity on the long-term details for the Agile improvement initiative and the work they commit to becomes less abstract
- Promising early results on transparency and long term follow-through

Thank you for your time and attention!

Questions?