Hands-on with Progress Tracking Cards: A Lightweight Method for Improving Your Software Practices

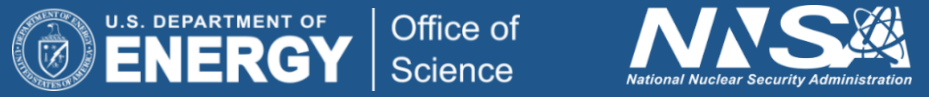
Elsa Gonsiorowski
Lawrence Livermore National Laboratory

Reed Milewicz
Sandia National Laboratories

Elaine M. Raybourn
Sandia National Laboratories

Greg Watson
Oak Ridge National Laboratory

2020 ECP Annual Meeting, Houston, TX



Tutorial outline and objectives

Presentation	10	Overview of PSIP
Activity	30	Summarize Team Practices
Debrief	10	Identify a Goal for Process Improvement
Slides	15	Overview of Progress Tracking Cards
Activity	30	Create a PTC
Debrief	10	Share your card
Conclusion	10	Where to go from here

115 Minutes Total

Demo	*afterwards*	How to move card into GitHub/GitLab/Jira

Focus of this tutorial

Realize process improvements without a disruption to any current development.

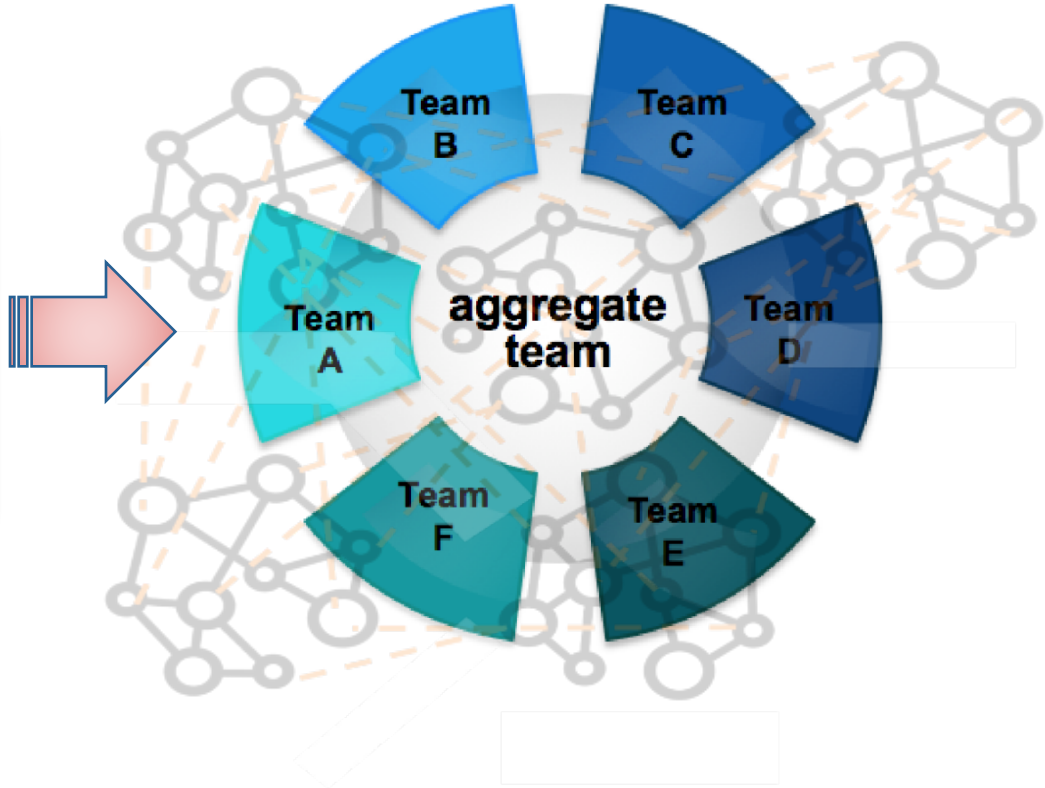
- *Introducing…*
- A practice that can help your team mitigate technical risk and develop software with confidence. (PSIP)
- How to identify topics for improvement by rating your project
- Progress tracking cards (PTC)
- Progress tracking card (PTC) catalog
- Integrating PTCs into your projects

Enabling Software Quality

PSIP: Productivity and Sustainability Improvement Planning

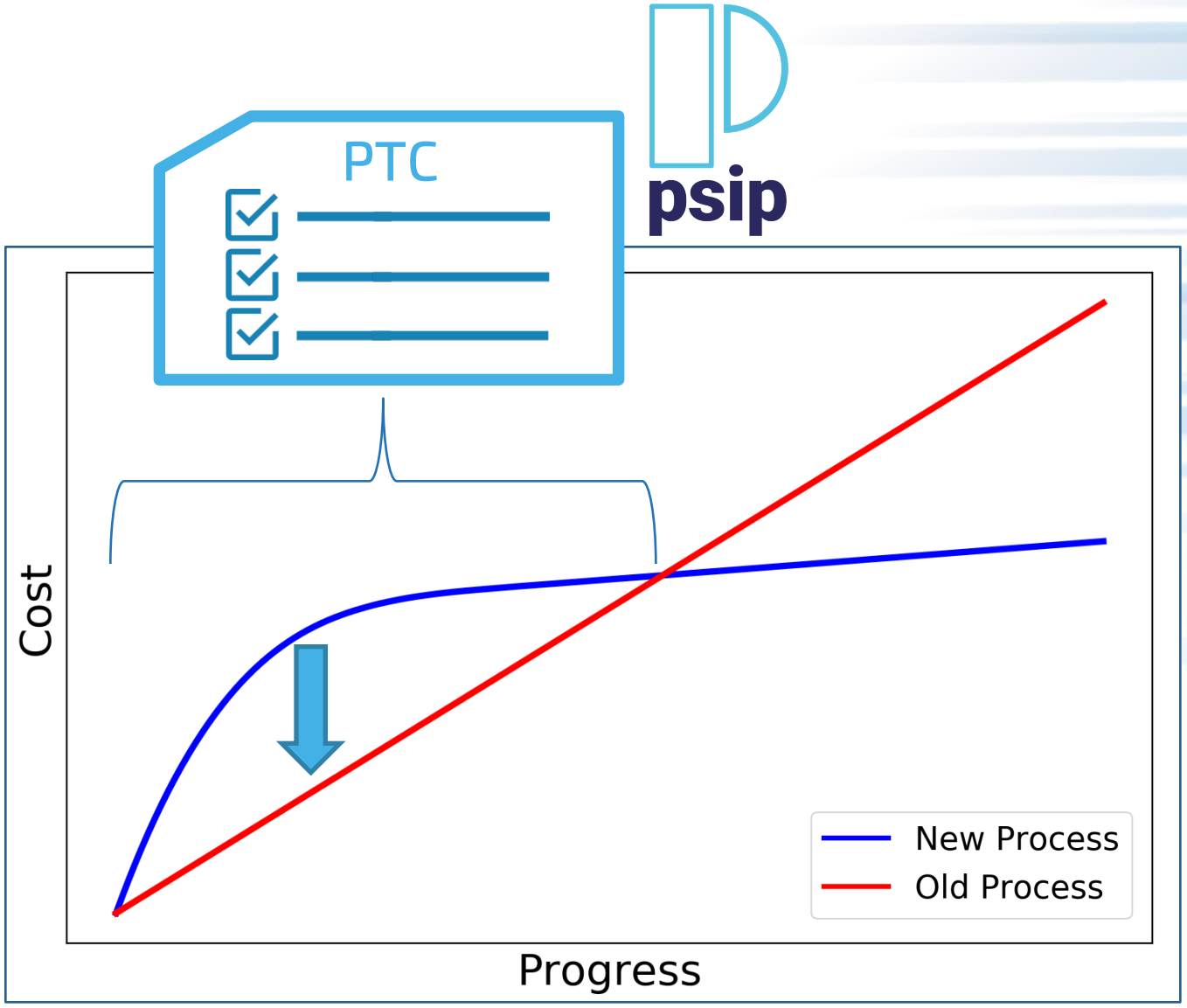
- PSIP is a lightweight workflow that can be used on its own or alongside frameworks you may currently use such as Kanban, Agile, etc.
- You *implement PSIP* by creating and using Progress Tracking Cards (PTCs) to achieve quality goals.

PSIP helps software teams to **IDENTIFY** opportunities to iteratively and incrementally **IMPROVE** software team practices and processes.



PSIP: At A High Level

- Why **not** improve?
  - Software process improvement often carries upfront costs.
  - It can introduce uncertainty and risk into a project. You can't control what you can't measure.
- PSIP provides tools and resources to set, measure, and realize improvement goals.



Productivity and Sustainability Improvement Planning (PSIP) Workflow

01 Summarize Current Project Practices

- Write brief practices summary document
- High level description, a few pages

02 Set Goals

- Identify practices ready for improvement
- Select those with near-term payoff

03 Construct Progress Tracking Card (PTC)

- Construct from PTC catalog
- Select only a few items

04 Record Current PTC Values

- Set baseline values for future reference

05 Create Plan For Increasing PTC values

- Define practice improvement steps
- Be specific, track issues

06 Execute Plan

- Increase PTC values by improving selected practices
- Track issues progress

07 Assess Progress

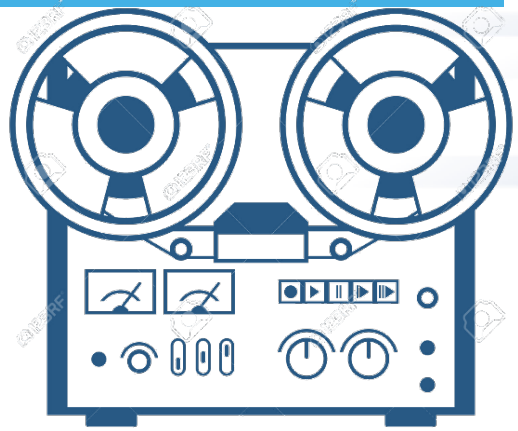
- Track PTC values
- Adjust strategy if needed

Reflect: How does your software deliver value to your users?

Reflect

Q: If the new software requires teaching other how to use it, how does that happen?

A: Our project tasks students and postdocs with formulating their own tutorial for how to build and run the code […]



01 Summarize Current Project Practices

- Write brief practices summary document
- High level description, a few pages.

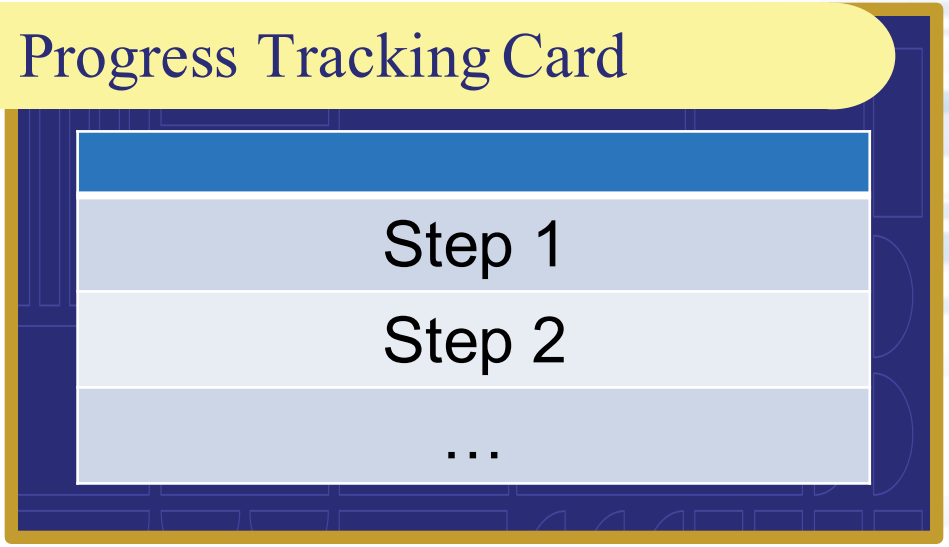
Scope: What are the barriers to quality and efficiency?

Scope

www.github.com/bssw-psip/ptc-catalog

02 Set Goals

- Identify practices ready for improvement.
- Select those with near-term payoff.



03 Construct Progress Tracking Card (PTC)

- Construct from PTC catalog.
- Select only a few items.

Plan: What actions can you take in a predictable span of time?

Plan

04 Record Current PTC Values

- Set baseline values for future reference.

05 Create Plan For Increasing PTC Values

- Define practice improvement steps.
- Be specific, track issues.

Enact: Can you execute on the plan? Are these changes impactful?

Enact

06 Execute Plan

- Increase PTC values by improving selected practices.
- Track issues progress.

07 Assess Progress

- Track PTC values.
- Adjust strategy if needed.

Activity: Summarize team practices by rating the current state of your project

- Software engineering is a systematic approach to the design, development, and maintenance of a software system
- By adopting a systematic approach to software development, it is possible to mitigate technical risk and develop software more effectively
- Quality in software is no accident; it is the result of deliberate practices
  - We often track the external qualities of software such as correctness or performance
  - But we can give the same care and attention to internal qualities such as maintainability, reusability, or testability
- Rate your projects' software practices: https://bssw-psip.github.io/ptc-catalog/survey.html

Debriefing: PSIP Goal Setting

- Rate your project – how did you do?
  - What would you like to improve based on your evaluation?
- When setting improvement goals, look for improvements that *can be realized in a few weeks or months*, at least in early PSIP iterations. Larger goals are fine, if they can factored into smaller increments with delivered value.
  - For example, if your team wants to have a goal of comprehensive testing, you will want to specify incremental goals in making your plan toward the comprehensive goal.

Work through improvement goal setting on your own, or with a PSIP facilitator.
bssw.io/psip

Debriefing: PSIP Goal Setting, continued

- As you set your goals, consider these questions, engaging external expertise as needed:

What consumes most of our time?

then Where are we inefficient?

then Are there new ways to improve our efficiency?

Where do we have quality problems?

then What are the sources of poor quality?

then How can we improve quality?

Is our product sustainable for its expected lifespan?

then What is limiting sustainability?

then How can we improve sustainability?

Can I construct a goal with deliverable value in manageable increments?

then Can each step be completed within a few weeks?

then Will accomplishing the goal provide tangible payoffs after completion?

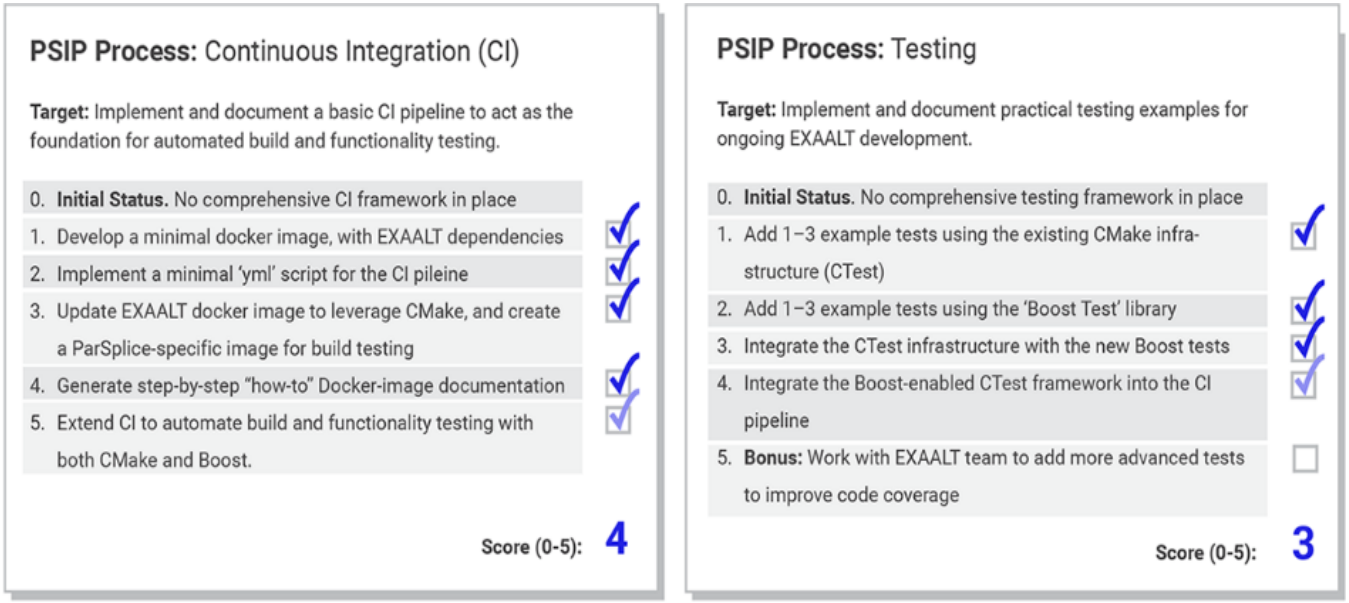
Overview: Progress Tracking Cards (PTC)

What is a **Progress Tracking Card (PTC)**?

- A step-by-step list of activities or outcomes that incrementally lead to improvements in team effectiveness and efficiency
- Created by the team during the PSIP process
- Contains the target, or goal of the planning activity
- Each goal should map to one PTC. Larger goals can be split over several consecutive PTCs
- Teams may select PTCs from the PTC catalog, modify PTCs found in the catalog, or define their own PTC

EXAALT PSIP: Continuous integration (CI) testing

BSSw blog article: Adopting Continuous Integration for Long Timescale Materials Simulation, Rick Zamora (Sept 2018)



PSIP Process: Continuous Integration (CI)

Target: Implement and document a basic CI pipeline to act as the foundation for automated build and functionality testing.

0. **Initial Status.** No comprehensive CI framework in place
1. Develop a minimal docker image, with EXAALT dependencies ☑
2. Implement a minimal 'yml' script for the CI pileine ☑
3. Update EXAALT docker image to leverage CMake, and create a ParSplice-specific image for build testing ☑
4. Generate step-by-step "how-to" Docker-image documentation ☑
5. Extend CI to automate build and functionality testing with both CMake and Boost. ☑

Score (0-5): 4

PSIP Process: Testing

Target: Implement and document practical testing examples for ongoing EXAALT development.

0. **Initial Status.** No comprehensive testing framework in place
1. Add 1–3 example tests using the existing CMake infra-structure (CTest) ☑
2. Add 1–3 example tests using the 'Boost Test' library ☑
3. Integrate the CTest infrastructure with the new Boost tests ☑
4. Integrate the Boost-enabled CTest framework into the CI pipeline ☑
5. **Bonus:** Work with EXAALT team to add more advanced tests to improve code coverage ☐

Score (0-5): 3

Activity: Create a PTC

- Hands-on example of how to go through the PSIP process to select existing or create new PTCs
- PSIP and PTC GitHub repos https://github.com/bssw-psip
- PTC catalog https://github.com/bssw-psip/ptc-catalog

Debriefing: Share your PTC

PSIP Process: Onboarding Process

Target: Implement a technical onboarding process to facilitate integration of new team members

0	**Initial Status**: No training process in place.
1	Understand MPICH current onboarding training practices and define training categories
2	Review and gather resources for training in atleast two categories
3	Design website and integrate content for two categories
4	Solicit feedback, improve content, design processes for external contributions and updating of website

Score (0-4) = **3**

EXAALT PSIP: Continuous integration (CI) testing

BSSw blog article: Adopting Continuous Integration for Long Timescale Materials Simulation, Rick Zamora (Sept 2018)

PSIP Process: Continuous Integration (CI)

Target: Implement and document a basic CI pipeline to act as the foundation for automated build and functionality testing.

0. **Initial Status.** No comprehensive CI framework in place
1. Develop a minimal docker image, with EXAALT dependencies
2. Implement a minimal 'yml' script for the CI pileine
3. Update EXAALT docker image to leverage CMake, and create a ParSplice-specific image for build testing
4. Generate step-by-step "how-to" Docker-image documentation
5. Extend CI to automate build and functionality testing with both CMake and Boost.

Score (0-5): **4**

PSIP Process: Testing

Target: Implement and document practical testing examples for ongoing EXAALT development.

0. **Initial Status.** No comprehensive testing framework in place
1. Add 1–3 example tests using the existing CMake infra-structure (CTest)
2. Add 1–3 example tests using the 'Boost Test' library
3. Integrate the CTest infrastructure with the new Boost tests
4. Integrate the Boost-enabled CTest framework into the CI pipeline
5. **Bonus:** Work with EXAALT team to add more advanced tests to improve code coverage

Score (0-5): **3**

Conclusion: In summary, what is PSIP?

- PSIP: Incremental, integrated into the primary feature development process, lightweight, iterative and informal.
- PSIP embodies the spirit of the **Plan-Do-Check-Act** (PDCA) management cycle.
- Aims to captures the tacit, more subjective aspects of team collaboration, workflow planning, and progress tracking.
- Bootstrap aggregate team capabilities into best practices.
- Encourage teams to adopt a culture of process improvement.
- **PSIP is easy to learn, especially for scientists who cannot dedicate time and resources to more formal or heavyweight approaches.**

Conclusion: Where do we go from here? PSIP and PTC resources

- PSIP landing page on bssw.io https://bssw.io/psip
- PSIP and PTC GitHub repos https://github.com/bssw-psip
- PTC catalog https://github.com/bssw-psip/ptc-catalog
- PSIP research paper https://www.osti.gov/biblio/1574620
- PSIP and PTC help https://gitter.im/bssw-psip/community
- Rate your project's practices here: https://bssw-psip.github.io/ptc-catalog/survey.html
- Examples of PTC integration in GitHub
  - https://github.com/bssw-psip/example-ptc-projects
  - https://github.com/bssw-psip/example-ptc-issues
  - https://github.com/bssw-psip/example-ptc-milestones

Thank you for your participation!

@gonsie

@rmmilewi

@elaineraybourn

https://bssw.io

ideasproductivity.org

https://bssw.io/psip

License and acknowledgements

License

Citations

- ***Requested citation: Gonsiorowski, E., Milewicz, R., Raybourn, E.M., Watson, G.*** **Hands-on with Progress Tracking Cards: A Lightweight Method for Improving Your Software Practices.** ***2020 ECP Annual Meeting, Houston, TX, February 3-7.***

Acknowledgements

- **Special thanks to Benjamin Sims, Jim Willenbring, and members of IDEAS-ECP.**
- This work was supported by the U.S. Department of Energy Office of Science, Office of Advanced Scientific Computing Research (ASCR), and by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.
- Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and 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-NA-0003525. Images used by permission.