CS-doc-5769-v1

Managed by Fermi Research Alliance, LLC for the U.S. Department of Energy Office of Science

Peer Review: It's not just for Physics Anymore!

Rob Kutschke

LArSoft Usability Workshop

June 23, 2016

Spirit of this talk

- I will point out some things that the HEP community has traditionally done very well
- I will discuss some lessons learned from the broader software development community
- It's time to draw lessons from all of the above and apply them to HEP software development.

Outline

- HEP analysis peer review process
- HEP has a good track record of integration testing
- Lessons learned from HEP construction project reviews.
  - Just the good parts!
- Lessons from the software development community
- Use the specialists
- Summary

HEP Analysis Peer Review

Life Cycle of a Physics Analysis

- HEP Community knows how to review an analysis very well:
  - Genesis of the idea
  - Analysis sub-group
  - Analysis group
  - Formal internal review (aka “god-parenting”)
  - Final review by collaboration
  - Peer review organized by the journal
  - Publication
- Very few outright errors get through the process

Life Cycle of a Physics Analysis

- HEP Community knows how to do this very well:
  - Genesis of the idea
  - Analysis sub-group
  - Analysis group
  - Formal internal review (aka “god-parenting”)
  - Final review by collaboration
  - Peer review organized by the journal
  - Publication
  - Feedback from the HEP community
- Very few outright errors get through the process

Life Cycle of a Physics Analysis

- HEP Community knows how to do this very well:
  - Genesis of the idea
  - Analysis sub-group
  - Analysis group
  - Formal internal review (aka “god-parenting”)
  - Final review by collaboration
  - Peer review organized by the journal
  - Publication
  - Feedback from the HEP community
- Very few outright errors get through the process

Cross-pollination with other groups and subgroups

Analysis Sub-group

- Most of the work is done in these meetings.
- Usually a spirit of “we are all on the same team”.
- Ongoing, weekly or bi-weekly meetings
  - Powerpoint slides; maybe a discussion forum?
  - Eventually a written internal report (or two, or three)
- A continuum of “styles”:
  - The more experienced mentoring the less experienced
  - An on-going community self-education project
  - Any analysis has elements from points along this continuum
- Invite outside experts as appropriate
- Cross-pollination with other groups and sub-groups
- An integral part of the education/mentoring process

Analysis Group … Journal Review

- Presentations in group and collaboration meetings
- Written report by authors
  - Input to the formal internal review process.
  - Many errors and omissions caught by authors at this step
- Sometimes a side working group for a cross-cutting issue.
- Presentations and written report by the formal internal review committee.
- Communication with journal reviewers is logged for everyone to see.

Features of the HEP Analysis Peer Review Process

- It works really well!
- A lot of the value is in the early stages in which review is lightweight and frequent
- A lot of the value is in the preparation:
  - of proper internal note
  - for the formal internal review
- A small group of people are charged with carefully vetting the algorithms and results.
  - But everyone is invited to participate
- External experts invited when appropriate
- Full information is available to all collaborators

Features of the HEP Analysis Peer Review Process

- It works really well!
- A lot of the value is in the early stages in which review is lightweight and frequent
- A lot of the value is in the preparation:
  - of proper internal note
  - for the formal internal review
- A small group of people are charged with carefully vetting the algorithms and results.
  - But everyone is invited to participate
- External experts invited when appropriate
- Full information is available to all collaborators
- An integral part of the education/mentoring process

HEP and Integration Testing

HEP and Integration Testing

- Most mature HEP experiments have broad integration testing suites:
  - A small subset is run as part of the nightly build
  - These days a smaller subset may run in a CI build.
  - The full set is used for release validation.
  - Focus on:
    - Ensure repeatability when it is expected
    - Ensure an overall improvement when it is expected
  - The suite needs to be broad
- Overall we do this well but we usually don't get started as early as we should
  - Mu2e had good integration testing for CD3, but not CD2.

Lessons Learned from DOE Construction Project Reviews

Lessons Learned from Mu2e Reviews

- Mu2e just finished a DOE CD3c review
  - Full DOE 413.3b) is much too heavy-weight for us
  - But there were good things in the process
- Over the past few years, each Mu2e subsystem has had a series of technical reviews
  - Organized by Mu2e
  - Often separate reviews for mechanicals and electronics
  - Reports from these reviews available to CDx reviewers
  - Each subsystem still needs a final Construction Readiness Review before it's funding is final
- Reviews are expensive
  - Money, hours and opportunity cost
  - Costs included in the project budget and plan of work.

Lessons Learned from Mu2e Reviews

- Impressions of most Mu2e people:
  - A lot of value added came from the prep work for the review
  - Each upcoming review motivated us to:
    - Have a second (or third) set of eyes look at everything
    - Track down and tie up loose ends.
- Reviewer comments
  - All in all of mixed value
  - But some were extremely valuable
- On net the reviews were worth their cost

Lessons from the Software Development Community

Lessons From the Software Development Community

- In many successful software companies, code review is a critical part of the development cycle.
- People have studied what works and what does not
- Some references from Marc Mengel
  - FAGAN, M. Design and code inspections to reduce errors in program development, _IBM Systems Journal 15(3) 1976 pp 1820211
    - https://www.cs.umd.edu/class/spring2005/cmsc838p/VandV/fagan.pdf
  - Smartbear: Best Kept Secrets of Code Review
    - http://smartbear.com/SmartBear/media/pdfs/best-kept-secrets-of-peer-code-review.pdf
    - But remember that they are selling their automated tools!

Lessons from the Software Development Community

- I have not yet read these carefully but a few things jump out:
  - Most errors are found by the authors when preparing for the review
    - Marc remembers a number of about 70% from, he thinks, the ”Mythical Man Month”.
  - Many lightweight reviews give better results fewer heavyweight reviews
    - Thinks of this as an analog to the weekly sub-group meeting at which one week’s work is discussed
  - Optimum chunk of code for a lightweight review is 200 to 400 lines.

Use the Specialists

Use the Specialists

- Skills needed to develop a successful algorithm include
  - Physics drivers
  - Detector physics
  - Quirks of this particular detector – often dominant
    - For LArSoft, there are potentially many detectors!
  - What do downstream algorithms and analyzers expect?
  - Software tools
    - Big and getting bigger
- Unreasonable to expect anyone master everything
  - But our community does cover all of the bases
  - Invite relevant experts to participate at appropriate times

Summary

- HEP has a great record of peer review for physics analysis.
  - Integrated QC and education/mentoring
  - In HEP software we don't do this
- Mature HEP experiments do good integration testing
- We know the value in construction project reviews.
- The software community has advice for reviews
  - Many lightweighter better than fewer heavyweight
- All find that much value added is in the prep for the review
  - In our case: having a deadline, profiling, prep presentation
- LArSoft needs to apply these lessons:
  - Reviews have a cost: people, time, opportunity cost
  - SCD and Experiment management must budget for this cost