Software priorities

Sho Uemura

Acting on priorities

- We have consensus on priorities: what next?
- Need to go from "yeah, we should do that" to actually knowing how we get it done
- We're not a top-down organization:
  - Nobody can keep track of everything that needs to be done
  - People closer to the work understand better what's involved
  - An "expert" in one task is a "learner" in another
  - Need direct person-to-person contact
- Due dates, not just priorities

A proposal

- Spin out responsibility for priority items (let's call them "components") to individuals ("component leads")
- The lead is not the person who does all the work; the lead is the person who knows what work needs to be done
- Components have:
  - Defined requirements/deliverables and milestones
  - Simulation studies or system tests
  - Due dates
  - Tasks with assignees
- Leads should know what work is being done for their component
  - Make this part of the "software roundtable?"
  - JIRA

What does this mean for people?

- Leads should feel comfortable assigning tasks, asking for expert help, etc.
  - ▶ Not that others shouldn't! But if you're responsible for something that we need to run HPS, everyone is responsible for helping you
- If you're doing work for a component:
  - ▶ You should be talking to your lead/updating JIRA about what you're doing, even if it's incomplete
  - ▶ Your lead should know how oversubscribed you are with non-software work/other software components

Components, leads (strawman)

- Needed for run:
  - ▶ Trigger tuning/monitoring: Kyle?
  - ▶ Data quality monitoring: Matt?
  - ▶ ECal monitoring/calibration: Stuart?
  - ▶ SVT monitoring/calibration: Omar?
  - ▶ Conditions: Jeremy?
  - ▶ DAQ/ET/EVIO: Jeremy? Sho?
- Needed for analysis:
  - ▶ ECal reconstruction: Holly?
  - ▶ Tracking/alignment: Pelle? Matt?
  - ▶ Event reconstruction and DST: Omar?
  - ▶ Physics analysis/MDC: Matt?
  - ▶ MC: Sho?
  - ▶ Data catalog: ?
- More? We can add things later as they come up

If this is OK, next steps:

- Over e-mail: discuss requirements with leads
- Leads should learn status, identify tasks, talk with other leads and non-software people about dependencies
- At upcoming software meetings, quick presentations on components
  - Emphasis on plans over status