

Each group should appoint a leader, a scribe, and a reporter (who might be the leader or the scribe or a third person). The leader's job is to keep the conversation focused and making progress, and to make sure everyone has an opportunity to contribute. The scribe's job is to talk notes, using one of the following google docs. And the reporter's job is to get ready to report back on the discussion in the next session, in 2-3 minutes.

Each group might want to start by having everyone briefly introduce themselves. They should then talk about software sustainability in the context of HEP.

The output of each group's discussion should be 3 specific actions that the HEP community could take to make software more sustainable. At least 2 should be things that could be accomplished within 2 years, or at least that would make significant progress that would lead to a measurable difference within 2 years.

A suggestion is to talk generally for 35 minutes, and then to focus on the specific outputs in the last 10 minutes, based on the previous discussion, but this is up to each group to decide.

Leader: Gordon Watts
Reporter: David Bruhwiler
Scribe: Robert Quick

Others in the room 5: David Lange, Martin Ritter, Paul Laycock

Action 1: Workshop to explore common software and services.

- Libraries,
- data and computational services
- Gateways (VREs) and analysis services

Action 2: Define common development environments. Work toward standardization of development environments. Create standard development environments by providing a package to new developers in the community. The HEP software development kit !

Action 3: We ran out of time.

General Ideas:

- David Bruhwiler:
 - Use of application containers -- Docker locally, with automatic conversion to facility-dependent HPC container technology -- to facilitate sustainability. The idea requires an adoption of standards, such as the Linux flavor and version that is maintained inside the container.

- Gordon Watts: Use preexisting open software tools to generate project templates including everything: testing/documentation/etc..
- Paul Laycock:
 - Workshop to identify places for common software. Data management, tracking software. Systematic review of code bases.
- Martin Ritter:
 - skip
- David Lange:
 - Identify opportunities for services needed by HEP that experiments or other groups are interested in co-developing (to eventually reduce cost)
- Rob Quick:
 - Professionalizing the software development in academic environment
 - Gateway services, infrastructure as a service