General session outline

The primary goal of a session is to collect requirements information for the specific topic that will go into a requirements document, the product of the workshop. The information gathered is expected to be in a draft or raw form, capturing relevant things that the community needs to accomplish for LAr TPC processing. The draft document has examples that will help set the level and tone of requirements information that is gathered.

The document is organized with one file per topic. Add feature or areas under the topic, as the examples show. It is expected that features or areas will each have several requirements plus discussion (rationale, goals, perhaps some lower-level details or examples).

Each topic area has a unique tex file:

- Topic I - nonbeam.tex
- Topic II - beam.tex
- Topic III - overarching.tex
- Topic IV - interfaces.tex

The section you should be filling in are:

- Functional requirements
- Nonfunctional requirements

The use cases are in files named after the experiment:

- DUNE - DUNE.tex
- DUNE 35t - DUNE-35t.tex
- Icarus and ArgoNeut - Icarus.tex
- LArIAT - LArIAT.tex
- MicroBooNE - MicroBooNE.tex
- ProtoDUNE - ProtoDune.tex
- SBN - SBN.tex
- SBND - SBND.tex
- WA105 - WA105.tex

The glossary is in glossary.tex. The other document fragments are named after the information they contain.

Timeline

Briefly introduce the session topic and remind everyone of the goal. (~5 minutes) Remember that the goal is to gather information that will be used to formulate requirements. The workshop product is the requirements document.

Discuss use cases and scenarios. (1 hour)
Review the ones currently in the document appendix. Capture new ones that important and critical to your topic. Discuss experiment commonalities and differences.

Move from use cases to requirment statements. (1 hours)
Focus on extrapolating requirements and glossary items from the use cases. Look particlarly for features that must exist in the LArTPC ecosystem to support or allow the use cases (or any other relevent task) to be carried out. Use the roles to help focus the discussion.

Filling in a subsection

Make a new subsection whenever you can identify a cluster of related requirements. People will naturally think in terms of partial solutions and it is okay to capture some of this. We will clean up the wording and text in the days following the workshop.

The subsection ought to have a list of the tasks that people want to do. Use your judgement on when to put in solution information and when to ask for more information about the underlying (or high-level requirement). The "Event-picker" subsection of the Human Interfaces topic is a good example. It directly names a facility. It is reasonable to add things that people want to do with a facility like this, but the better target is discovering what the analyst is doing that calls for such a facliity to exist.

The current examples are seeds for discussion. They have a strong tendency to be written as system requirements and should be migrated to user requirements. They can (and likely should) be heavily modified. Don't try to get the first sentences written right the first time - iterating after people see what is written is typically good.

Subsections can, and probably ought to, have examples and rationale to help clarify and explain the context. It is somethings difficult to remember details after the workshop.

Notes

Things to be explored:

- what do you need to do to achieve the reconconstruction efficiency you expect?
- what inputs to you need to accomplish a specific task?
- what outputs does a specific task need?
- how do we verify that the ecosystem is fulfilling the requirement?

Things to keep in mind:

- requirements need to be testable (fast and easy to use are not good enough by themselves)
- the level of requirements that we are targeting (interactions with parts of the system)
- the topic that you are exploring
- common language should be used, ask for clarification when necessary
- when to move on to another requirement area
- the roles will help focus people's thoughts - identify a role with each requirement
- As people bring up different issues, use cases and common interests should start to appear.
- The subject of a user-level requirement is a type of user, the subject of a system-level requirement is the system.

Examples to help thing move forward:

What difficulties are you encountering when ...

- developing an overall reconstruction stretegy or processing chain?
- developing algorithms or configuring data flows?
- dealing with detector anomolies, problems, and issues that arise?
- handling auxiliary data (calibration data, conditions data, beam data)
- sharing of algorithms or data products?

and what do you want a LArTPC ecosystem to provide for you that will alleviate these difficulties?

Some additional questions:

- Where do you draw the line between production reconstruction and analysis?
- What data do you need to do analysis tasks?