WBS 1.08.06 Integration/Test/ Deployment

L3 Lead - Cosmin Deaconu

Outline

- Presenter Introduction
- Key L3 contributors
- Breakdown of this L3
- Key Requirements
- Interfaces with other L3s
- Technical Design / Scope
- Prototyping/Testing Plans
- Risks and Mitigations
- Conclusion

Presenter Introduction

Name: Cosmin Deaconu

Institution: University of Chicago

Discipline: Particle Astrophysics

Previous experience:

- Postdoc (2015-2020)/Research Scientist (2020-): Development and integration of data acquisition / control / data management software for the ANITA long-duration balloon payload and prototype radio neutrino detectors at the South Pole, the White Mountains of California and Greenland. Also key contributor to analysis/simulation software and data analysis.
- PhD (2009-2015, MIT) on directional dark matter detection instrumentation.

Key Contributors in this L3

Abby Crites (Cornell)

Cosmin Deaconu (Chicago)

Brian Koopman (Yale)

Laura Newburgh (Yale)

Sasha Rahlin (FNAL)

Christopher Weaver (Michigan State)

Nathan Whitehorn (Michigan State)

L4 WBSs within this L3

L4 WBSs will be developed over the coming year for this as the timeline for deployment becomes more clear.

Key Driving Requirements for 1.08.06

CQ1

- Uptime
  - Ensuring that the DAQ is not responsible for more than 0.1% of data loss
- Commonality
  - Ensuring that the DAQ will work in lab setups, South Pole and Chile
- Verified at Scale
  - Having confidence that the DAQ will handle a full observatory before deployment.

Inter-L3 Interfaces within this L2

CQ2

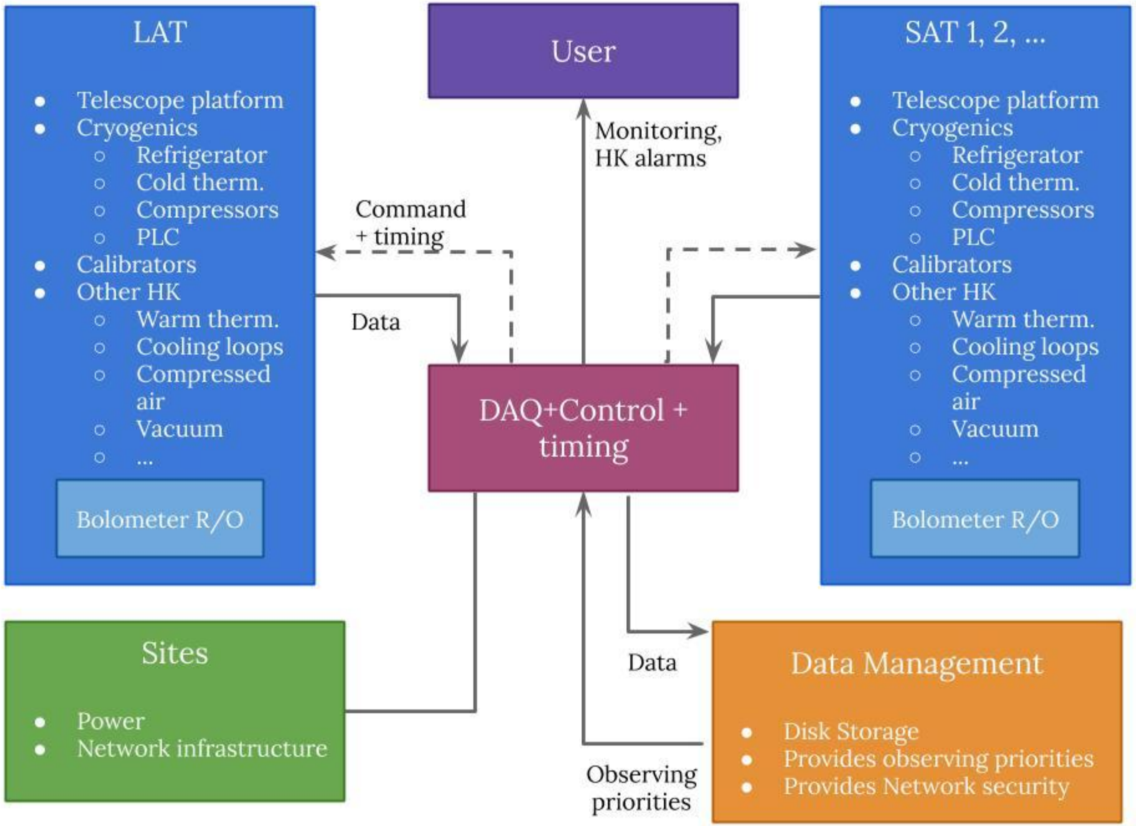
1.08.02 (Observatory Control System), 1.08.03 (Observatory DAQ), and 1.08.04 (Monitoring and Alarms) all have impacts on ITD.

Technical Design / Scope

CQ2

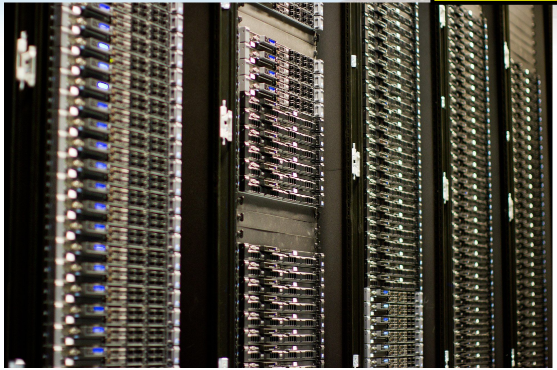
DAQ sits in between all other subsystems and must be integrated with everything at two non-identical sites (Chile/SP).

Same underlying DAQ system will be run at both sites, with differences in configuration.

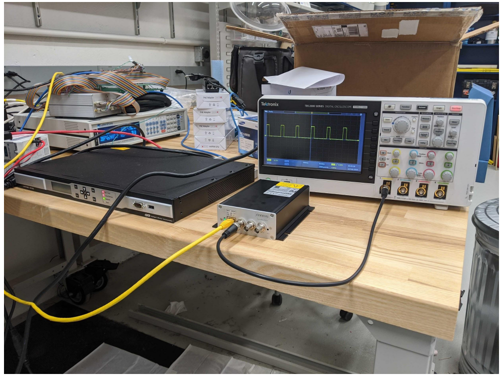


CQ5

DAQ Deployment Hardware



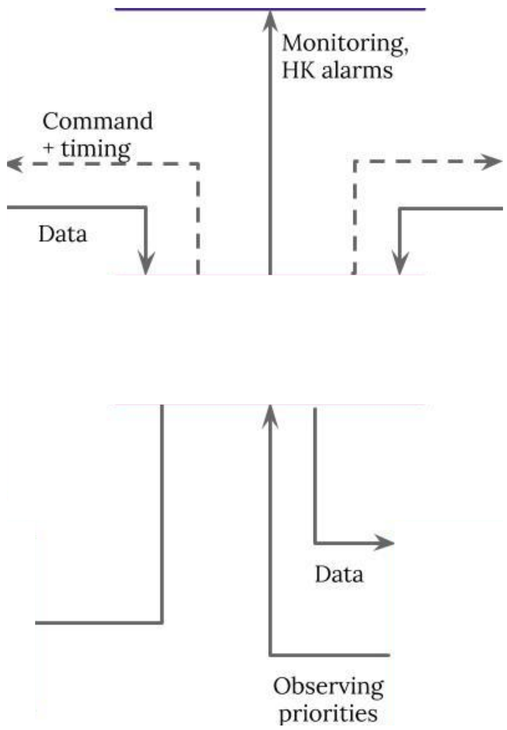
- DAQ software is designed to run on inexpensive, easily-available commodity hardware using standard Linux distros (e.g. Ubuntu LTS, EL7/8).
    - We will coordinate with other L2's to choose a global common target OS for S4.
- Containerization simplifies testing and deployment of DAQ systems, allows for version control and continuous integration of the entire DAQ stack.
- Network-transparent design allows load-sharing across multiple systems.
- Spec at least 10 Gbps networking.
- No custom hardware (COTS timing, switches)



Integration with Detectors/Housekeeping

CQ2

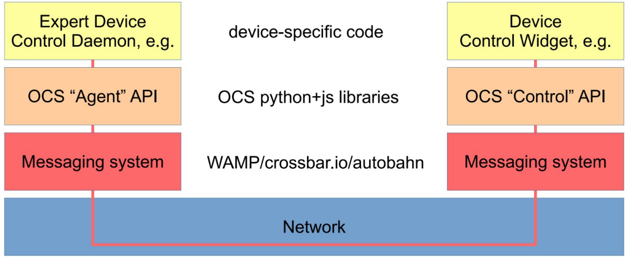
- All communication will be via standard TCP/IP to software agents running on a “computer” (not a microcontroller or an FPGA or ASIC or diode that’s harder to deploy updates to.)
- All data taking is asynchronous (no global clock needs to be distributed) at the DAQ level.
  - PTP will be used to distribute precise timing to systems that need it for after-the-fact synchronization.
  - Many systems will only require NTP (ms) accuracy.
- Outside readout, network load expected to modest.
  - Readout may use a dedicated link for QOS.
- Goal is that DAQ “integration” is essentially plugging in a network cable.
  - All internal communications can operate in the same DMZ, with “gateway systems” for remote access.

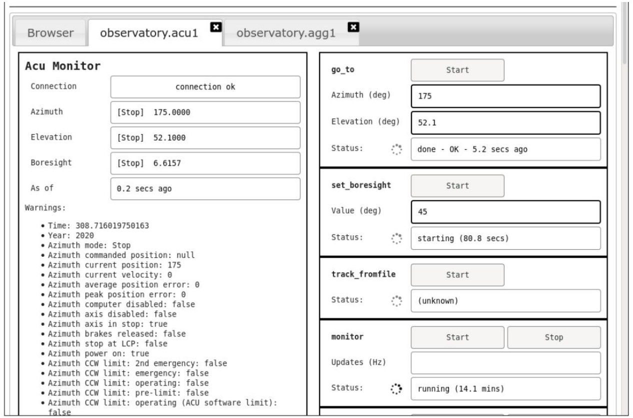


Site Hardware Configuration

See B. Koopman talk

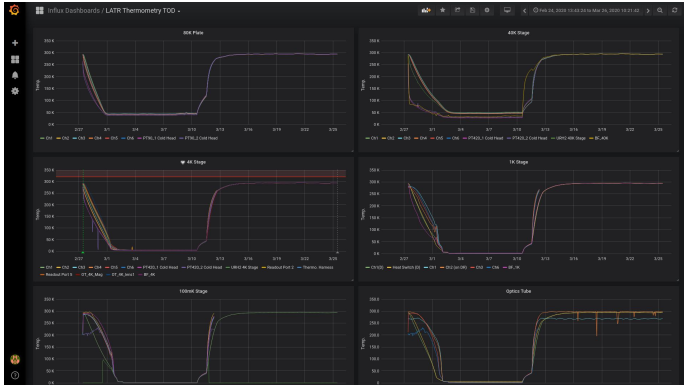
- No hardcoded list of hardware, instead automagic discovery of hardware.
- Hardware “agents” will register themselves to the DAQ, which will be happy to process their data streams, so the two sites can be quite heterogeneous.
    - Therefore pieces can be tested separate from the whole (see next talk).
    - We stress, writing hardware “agents” will documented and simple so that hardware experts will be able to do it for most devices (but we will help!)
- Operation of hardware configurable using DAQ “clients” which control each agent via the knobs it exposes.
    - Such client could be something like a scheduler, or an interactive web page.



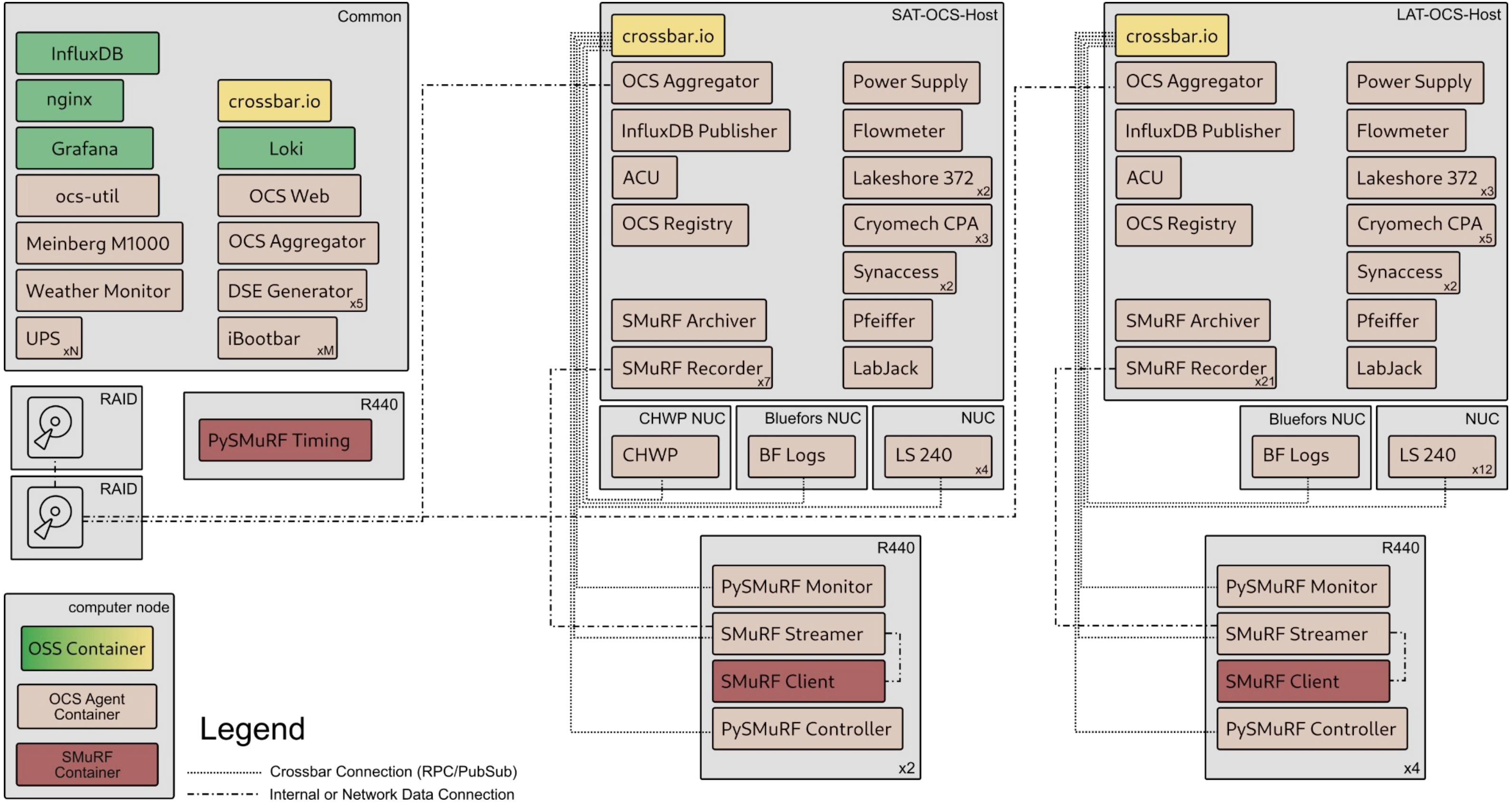


Site Monitoring and Alarm Integration

- Getting data into the DB is not so useful if you can't look at it.
- The planned monitoring system (Grafana) is easily configurable via a discoverable interface to display "dashboards" with the relevant housekeeping quantities (see B. Koopman talk).
- Alarms will also be configurable, at multiple priority levels based on DB values.
    - note: we'll propagate safety alarms on a best-effort basis, but **the DAQ is not a safety system**.
- Each site will have a different specific alarm system (e.g. the South Pole Land Mobile Radio), for which we will develop clients



Deployment “Guinea Pig”: SO



Site Infrastructure Requirements

CQ5

- DAQ design aims to minimize required infrastructure to simplify deployment (everything is hard at our sites).
- All that is required from each site is >=10 Gbps network (copper or fiber) taps, a cooled space to store some server racks, power, and coax to a GPS antenna with a good view of the sky (for timing).
- We do not yet have an estimate of the number of machines required or total power required (expected to be small compared to DM) .
- Deployment time and personnel is expected to be modest due simple interfaces and rigorous testing (next slide) before deployment, minimizing stress on facilities (especially at Pole).
- Due to containerized setup, spare systems easily deployed and provisioned in case of failure.
- **Interface with Integration and Commissioning L2 once requirements fully identified**

Prototyping/Testing Plan

- Simulator agents will be developed for each key piece of hardware
  - Allows testing of the entire system for prior to the delivery of hardware.
  - Simplifies continuous integration of development versions to detect any regressions.
- Mock testing site(s) will be constructed at collaboration institute(s) with required components to simulate real site.
  - Smooth operation of DAQ with actual hardware before deployment to the site.
  - Test any changes before deploying to real sites.
  - Diagnose any issues not “live” on the detector
- Simons Observatory is likely to run into any major showstoppers with OCS, should there be any, before S4.
- Some components (Grafana) already being tested at Pole.

Deployment Risks Identified

CQ4

Risk	Mitigation
Hardware (and/or hardware agents) not ready in time for full integration with test system.	DAQ system testing with simulators only (and work harder on simulators while hardware gets ready). Deploy DAQ experts in case there are problems.
Site differs from test setup in important way. Examples: • network issues, including low bandwidth and availability, firewall issues, DNS resolving issues, latency. • hardware overheats or has higher failure rate due to altitude.	Anticipate issues based on experience and try to incorporate them into test setup. E.g: • Impose site network connectivity/latency into test setup and simulators. • Set up local caches for anything big that may need to be downloaded (e.g. docker images, all development tools!) • Overprovision hardware and derate temperature
Not enough DAQ experts available for deployment (conflicts, PQ, visa, travel delays)	Increase size of DAQ team. Good documentation.
Remote diagnosis of issues at sites (SP) is hard	Good remote test setup, log aggregation, WO docs.
DAQ architectural constraints	Stress testing early, safety margin in throughput.

Deployment Risk Formalization

CQ4

- Formal risk assessment is ongoing, including surveying previous deployment experiences for similar sites.
- We will assign likelihoods and impacts to risks and prioritize effort into mitigations for high-likelihood / high-impact risks.

Conclusions

Deployment is still quite far in the future, but DAQ designed with easy integration, testing, and deployment in mind.

As the schedule and budget becomes clearer, we will develop a firmer deployment plan.

We will soon begin to formulate risks and mitigations.

Backup Slides