

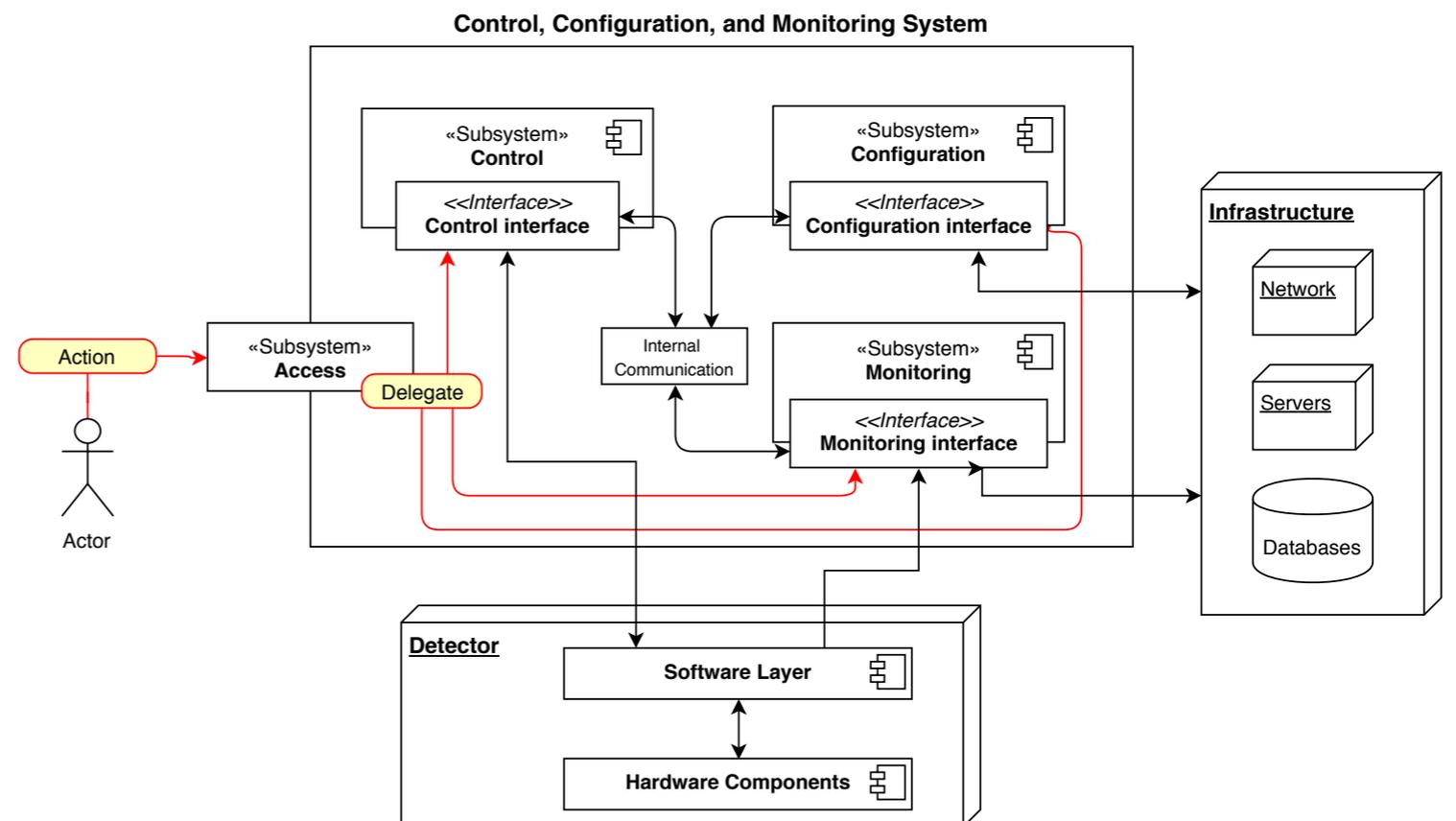
Overview of CCM activities

Alex Tapper

February 15th 2021

Outline

- Logging
- Operational monitoring
- Configuration
- Run Control
- GUI



Logging

- Chose to use the functionality of both ERS and TRACE packages
 - ERS provide highly structured information
 - with time, issue distribution, subscription and archiving, visualisation, searching and analysing issues, also basis for automatic procedures
 - TRACE to provide local logging and fast memory-based debug

Rev. 1.0.0 of logging package - summary

1. CMakeLists.txt changes:
 - Add: find_package(logging REQUIRED)
 - Remove: find_package(ers REQUIRED) and find_package(TRACE <version> REQUIRED)
 - To DEPENDENCIES, add: logging::logging
2. #include "logging/Logging.hpp"
3. Use ERS DECLARE_* macros, as usual
 - I suspect there may be an "issues package"
4. Use:

```
ers::fatal( ers::Issue & )
ers::error( ers:: Issue & )
ers::warning( ers:: Issue & )
ers::info( ers:: Issue & )
TLOG() << message_or_Issue;
TLOG_DEBUG(lvl) << message_or_Issue
```

note: existing uses of TLOG() may need to be adjusted.
5. Remove any uses of ERS_LOG, ERS_INFO, and ERS_DEBUG

7 3 February 2021 Ron Rechenmacher | Finished Logging Package



R. Rechenmacher: https://indico.fnal.gov/event/47519/contributions/207429/attachments/139699/175431/DUNE_finished_logging_package.pdf

Logging

- Making effective use of logging requires thought and work from developers
- Discussion started a few weeks ago
- Example will be provided alongside documentation

Ways of using ERS Issues

- [ERS documentation](#)
- Each ERS issue is an instance of a class:
 - different information shall be carried by different classes;
 - Don't be lazy at the start, because you will never create the correct issues a posteriori
 - The classname shall be specific and meaningful, e.g.:
 - ConfigurationError is a totally useless Issue, CannotOpenFile is already

- ERS
- Th
- Re

2 Feb 3rd

Throwing or reporting

- When deciding what to do you should be "respectful"
 - Don't throw if the issue cannot be caught: e.g. if you spawn a thread and throw in its run loop your process will crash making any type of reaction (as well as diagnosing) impossible
 - You cannot know and shall not make assumption that your thread has the "right" to destroy the complete workflow of the application
 - In this case report the issue using `ers::warning`, `ers::error`, `ers::fatal`

asked to do a specific thing

... let the caller handle the



Severities

- Severities are chosen based on the impact of the application
- Information
 - Report something that needs to be known but is not indicative of any problem (e.g. ChangedTriggerPrescales)
- Warning
 - Report a specific anomaly that is not affecting the functioning of the application, but needs attention (e.g. EmptyFragmentReceived, DuplicateFragmentReceived, ...)
- Error
 - Report an anomaly that is persistent and will likely not be solved without external intervention (e.g. NoDataFromLink, DiskAlmostFull)
- Fatal
 - Report a major problem that does not allow the application to continue working (e.g. DiskFull, NoMemoryLeft, ...)

8 Feb 3rd, 2021 GLM I ERS



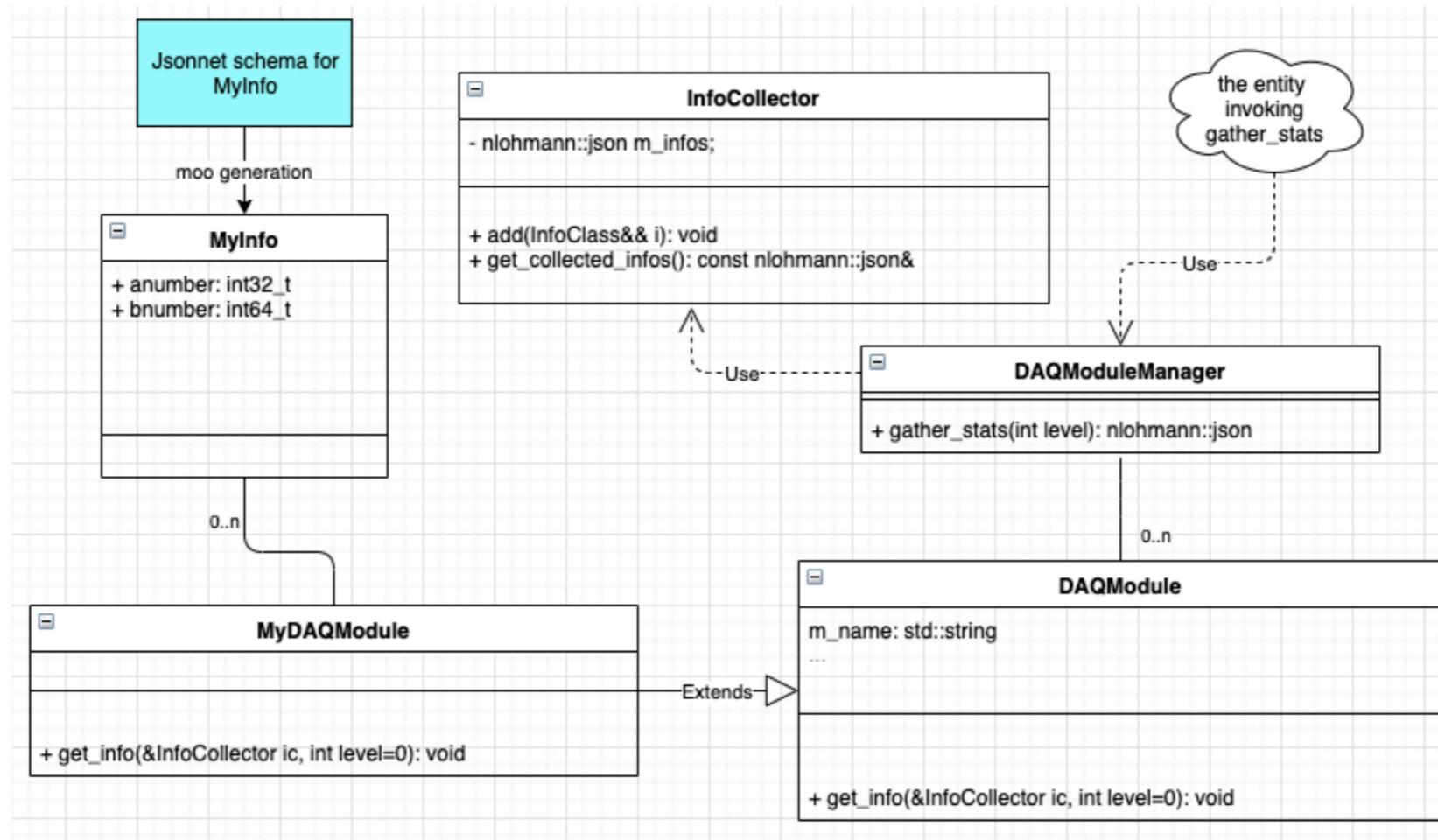
G. Lehmann: <https://indico.fnal.gov/event/47590/contributions/207746/attachments/139701/175428/ERSConventions.pptx>

Operational monitoring

- Previously presented operational monitoring prototype:
 - Register independent (atomic) variables to some service and let it worry about collecting, processing, packaging, etc
- New approach now being pursued:
 - Define information structures via schema that are filled by the user code on request and returned as structured data
 - Data structure forms a contract for what the application will provide as information and what listener should expect

L. Stankovic: <https://indico.fnal.gov/event/45866/contributions/199349/attachments/135926/168961/Operational-monitoring-roadmap.pdf>

Operational monitoring



- Implementing user side of the diagram using timing service as the guinea pig — backend also being implemented

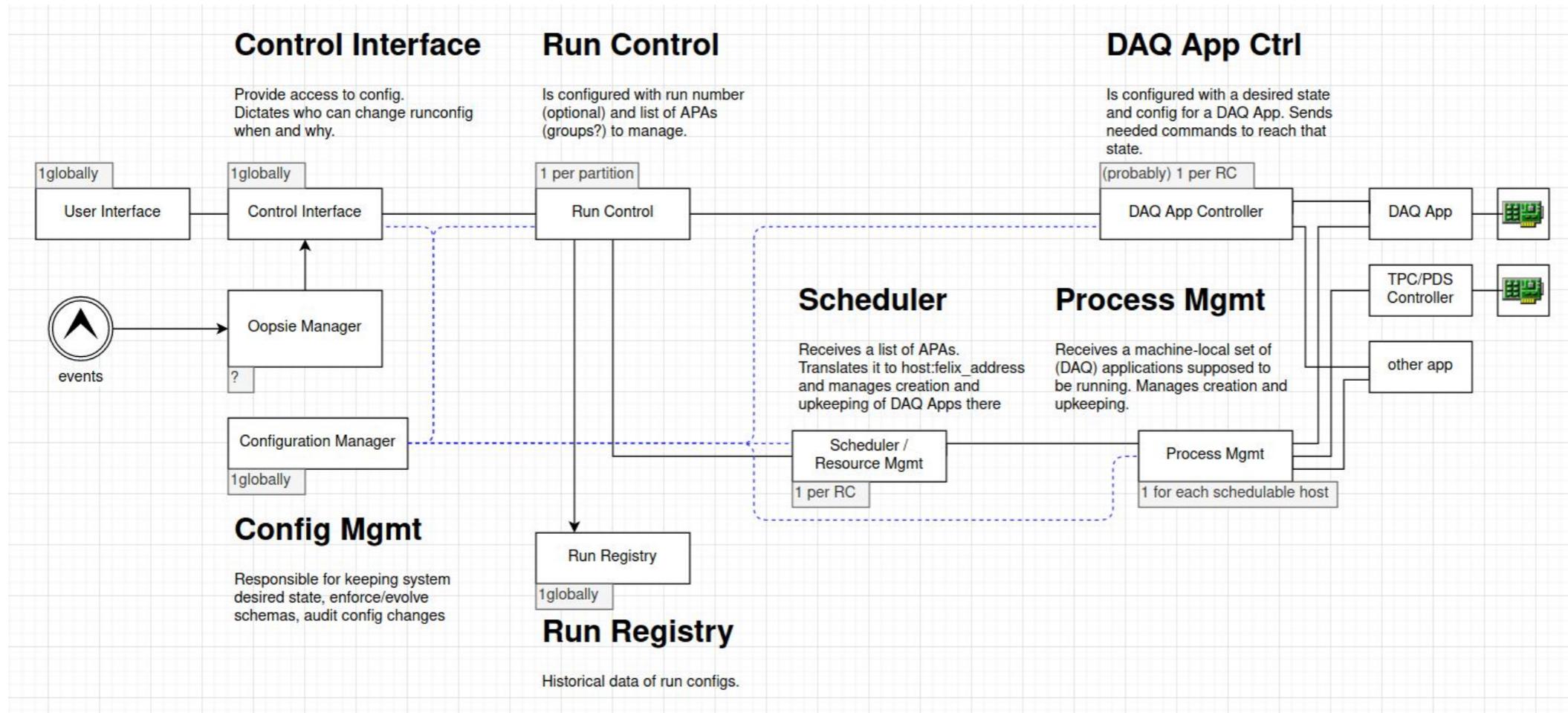
G. Lehmann et al.: https://indico.fnal.gov/event/47519/contributions/207430/attachments/139703/175430/022021_opmon.pptx

Configuration

- Several presentations on code-gen and configuration
 - No time/need for a reminder here ...
- Configuration data is sent to modules from Run Control in json format according to a defined schema
- Schema are specified in jsonnet which uses moo to generate json, C++ and python corresponding to schema
- Currently discussing configuration generation
 - Factorise configuration data into logical chunks, act as templates which can be reused, edited etc.
 - For example: process information (host, env etc.), init information (modules, queues), commands
 - Likely to be separate files for upcoming release for ease of use

Run Control

- Just started to think about concrete implementation



- For upcoming release focus on DAQ App Ctrl

G. Dirkx: https://indico.fnal.gov/event/47774/contributions/208291/attachments/139919/175753/10-feb-21_run_control_discussion.pdf

Run Control

- Just started to think about concrete implementation

DAQ Application Manager

Supplements Process Manager

- Provides DAQ-specific process management (the DAQ state machine)

Goals: given a (list of) application(s) and their desired state

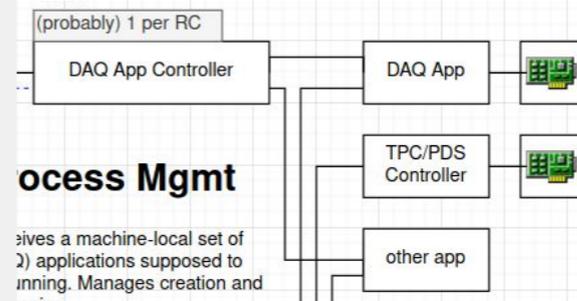
- perform needed state transitions to achieve desired state.

Manages

- DAQ-specific lifecycle
- First-Aid failure procedures
- Basic health checks

DAQ App Ctrl

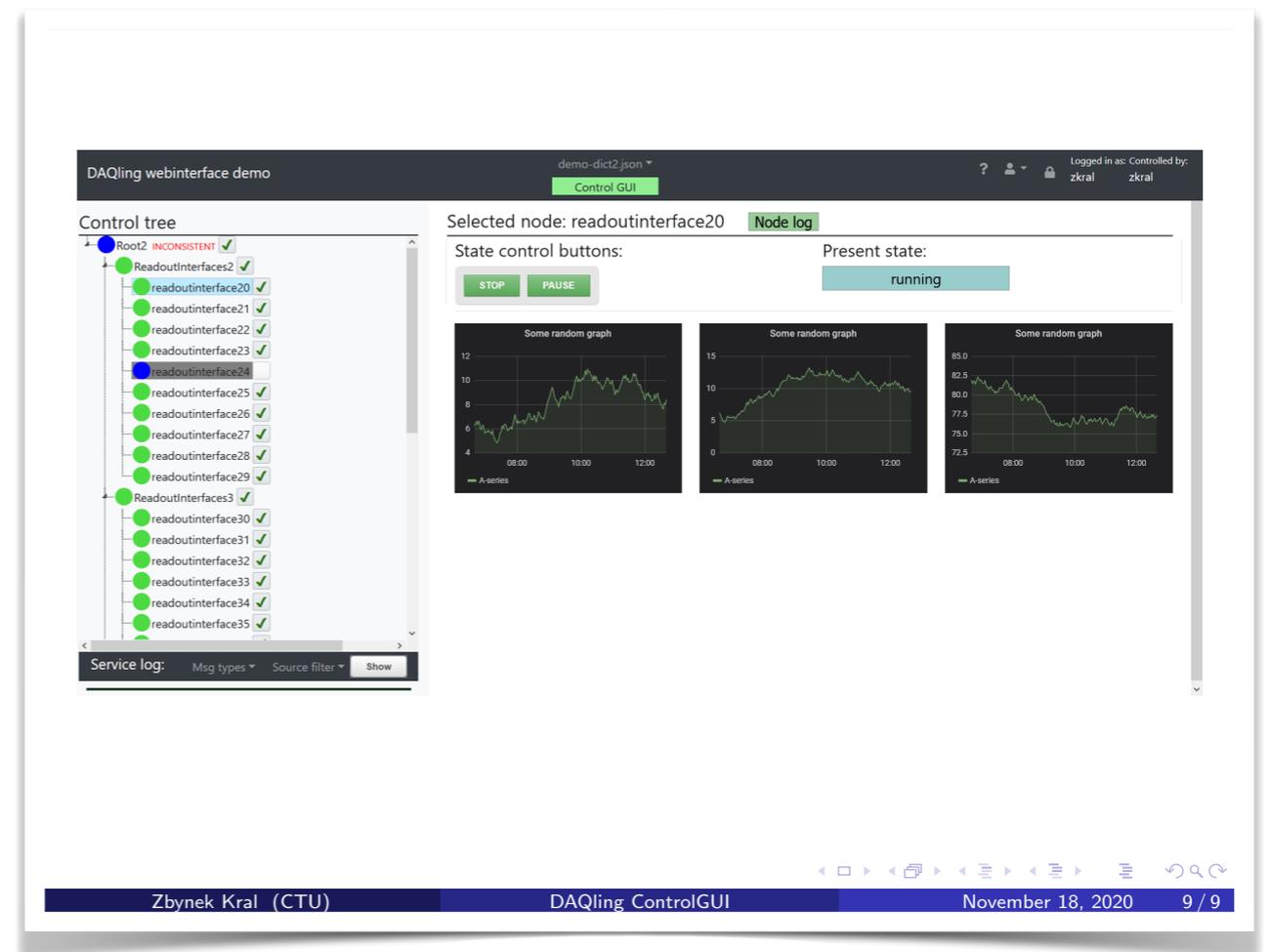
Is configured with a desired state and config for a DAQ App. Sends needed commands to reach that state.



- For upcoming release focus on DAQ App Ctrl
 - Send commands with appropriate configuration where necessary to multiple Apps
 - Drive applications through state transitions

GUI

- Discussions on Graphical User Interface
- Potential to build on work done for DAQLing →
- Possibly start with configuration generation

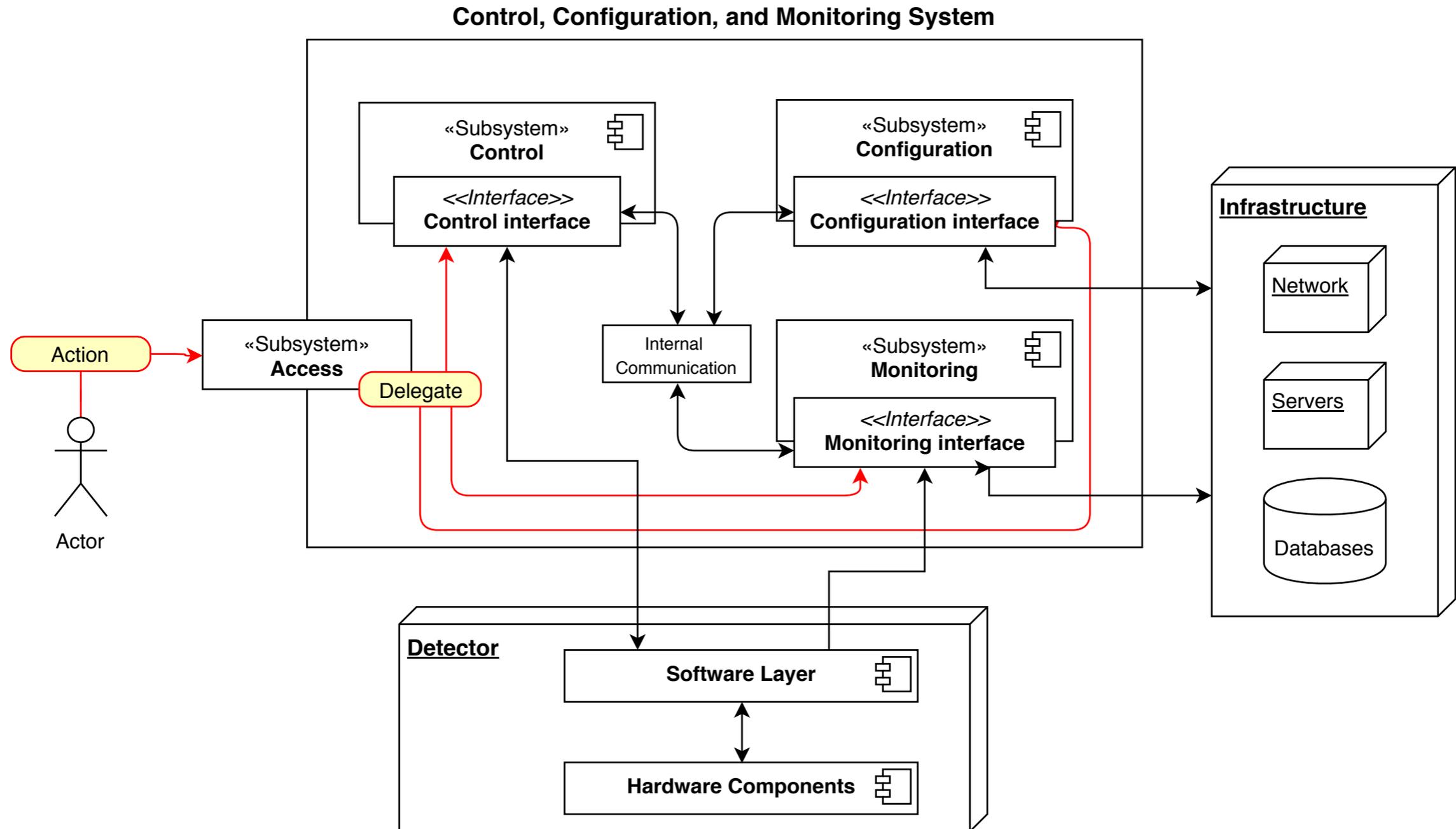


Z. Kral: https://indico.fnal.gov/event/46474/contributions/202129/attachments/137517/171473/DAQLing_ControlGUI_11182020.pdf

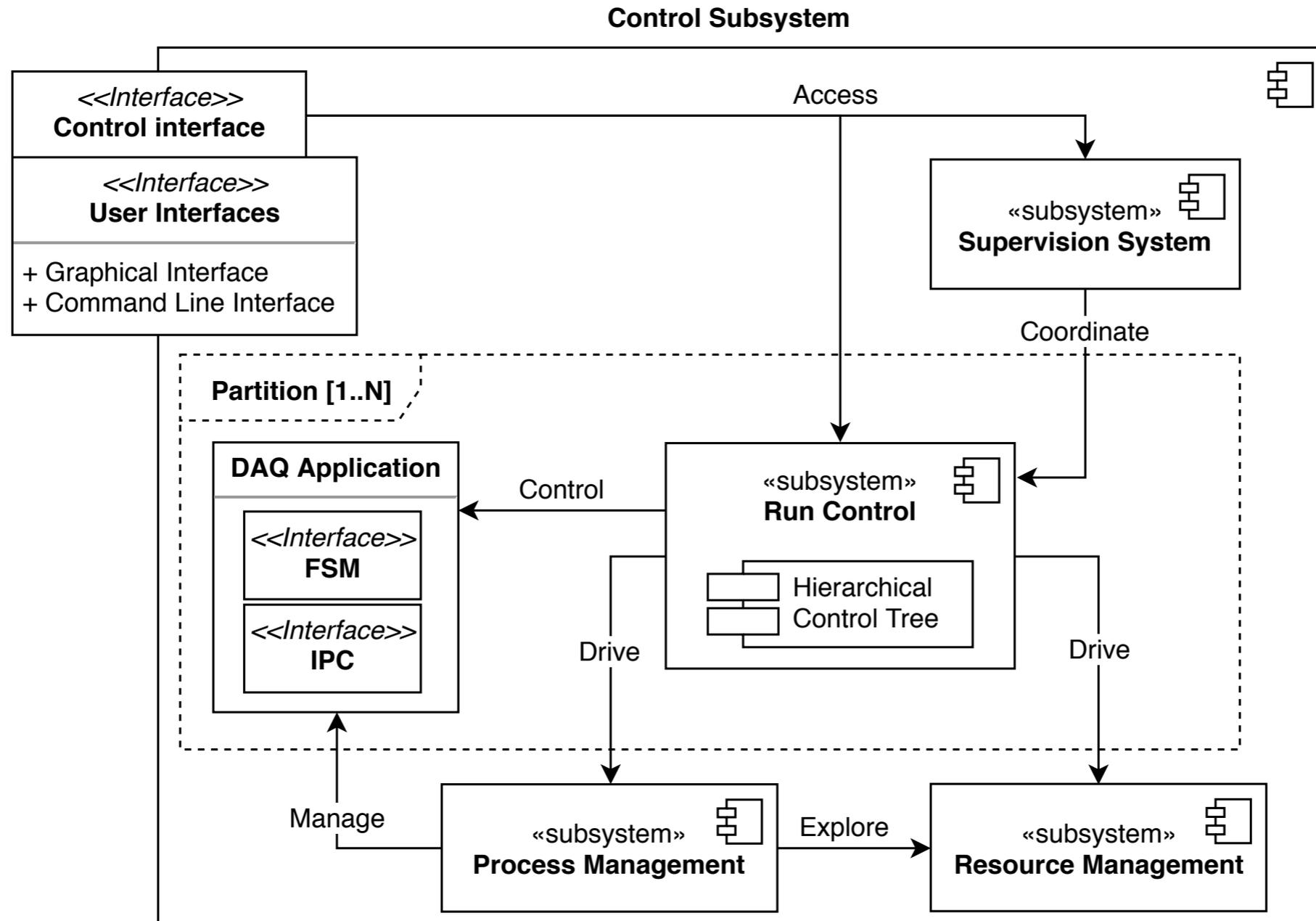
Summary

- Making progress on key items for DAQ release v2.3
- (small) team coming together — new contributions welcome!

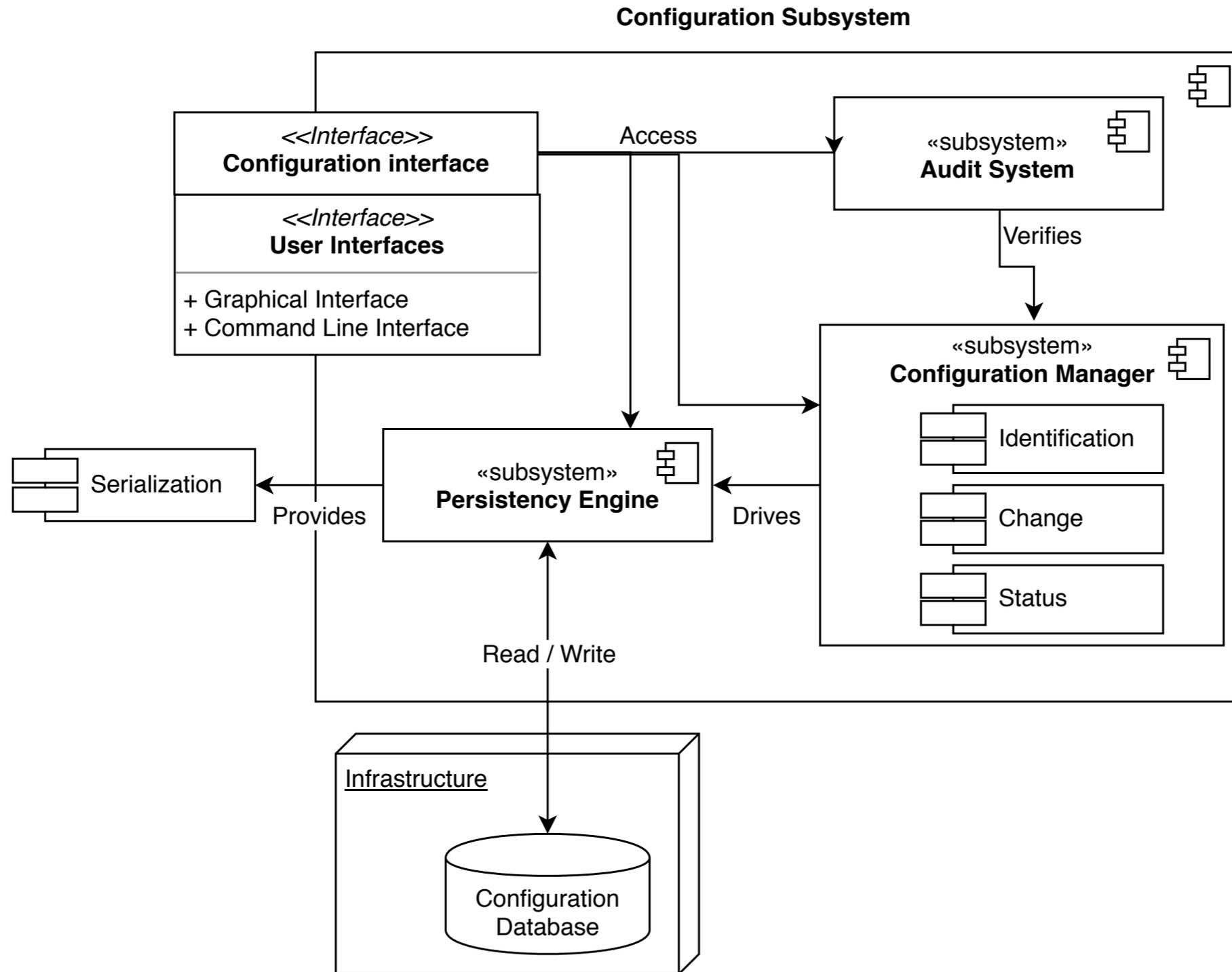
Overview of CCM



Overview of Control



Overview of Configuration



Overview of Monitoring

