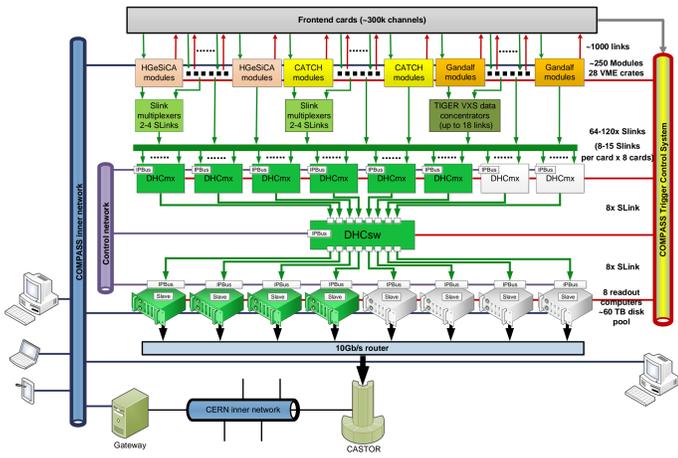


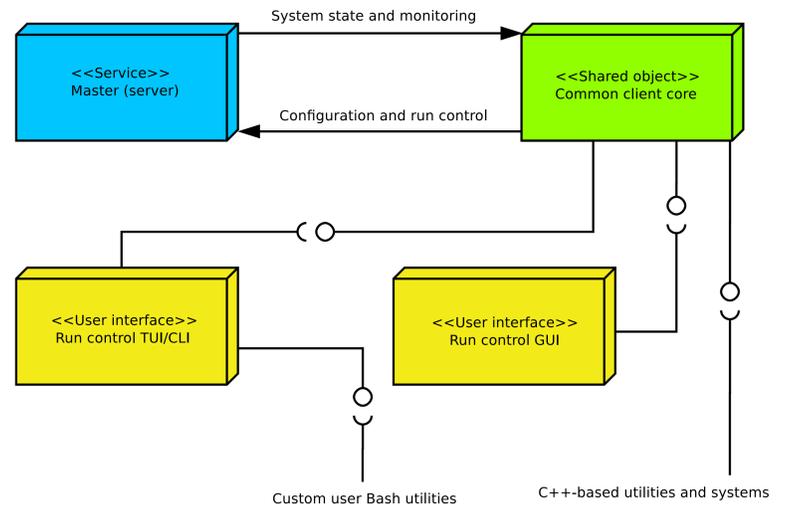


The iFDAQ system of the COMPASS experiment

- Hardware-based EvB
- Data concentrated by 6 (up to 8) DAQ modules with multiplexer firmware
- Distribution to 4 (up to 8) readout servers by the DAQ module switch firmware
- Complete events received by servers
- Consistency check at several layers
- Events verified and converted to DATE data format

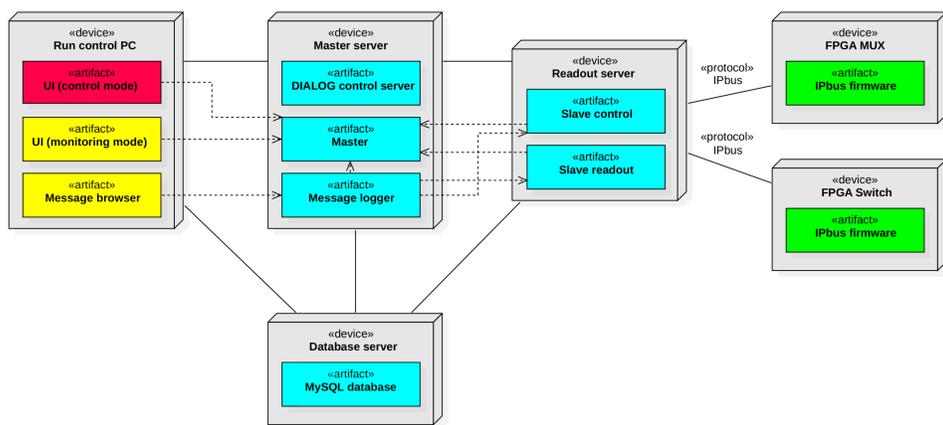


iFDAQ – new interface architecture



The common client core implements representation of the system state and communication with the server. Configuration, run control and monitoring functionality is exposed as an MVC API. The GUI, TUI/CLI and any C++-based utilities/systems build up on this package. The new architecture is planned to be fully deployed in 2020.

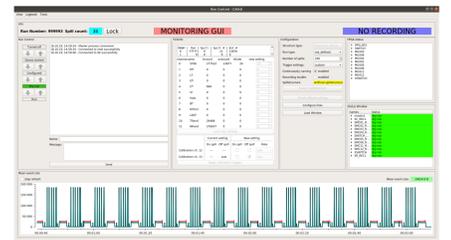
iFDAQ Architecture – Software Structure



- **DIALOG** is a custom alternative to DIM used in the iFDAQ control system – its control server is equivalent to a DIM nameserver in terms of functionality
- **Master** is the main control process which controls the slave processes by forwarding configuration and run control commands to them; it also collects and exposes monitoring information
- **Slave-readout** reads out and verifies data
- **Slave-control** monitors and controls the FPGA cards using IPbus
- **UI** is a general client which can take on two modes: **Monitoring**, which subscribes to the monitoring service of the Master process, and **Control** (only one such instance can exist at a time), which can send configuration and run control commands to the Master process – development of this client is discussed further in this poster

Run control GUI

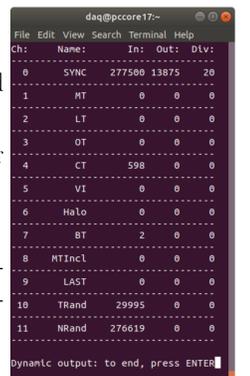
- Provides a graphical representation of the configuration and state of the system
- Configuration and run control is exposed to the user in the form of GUI elements
- Remote access is partially possible with a dedicated X2Go/Xpra server



Run control TUI/CLI

Terminal mode:

- Provides a dynamic textual representation of the configuration and state of the system
- Configuration and run control functionality is exposed to the user in the form of interactive commands
- Functionally equivalent to the GUI
- Solves the remote access question in the cases when the aforementioned X-forwarding alternatives are not feasible (e.g., on smartphones)



Command-line mode:

- Exposes configuration, run control and monitoring functionality as a Bash API
- Control commands re-used from terminal mode
- Non-dynamic, machine-readable alternative provided for monitoring commands

Example use case:

In 2019, the Bash API's ability to automate run control has been used to measure the software readout performance of the *Slave readout* process – including transformation to the DATE format, the process has been measured to be able to read out as much as 700MB/s per readout server at an event size of 250KB.

iFDAQ – interface requirements

Ideally, as for most control systems in HEP, the following interfaces should be present:

- **Desktop GUI** – to be used in the control room; emphasis should be placed on user experience, as shifts are traditionally used to train new members of the collaboration
- **Remote access UI** – to be used by experts to solve problems remotely; normally takes form of a web interface or an X-forwarded on-site desktop GUI
- **Programming API** – to expose control and monitoring functionality to other systems
- **Scripting API** – to allow for quick and easy creation of custom user utilities

Achieving maximum manpower efficiency

Having separate interfaces is labor-intensive and not achievable in practice with a small team, as the existing design showed. Therefore, when designing the new interface architecture of the iFDAQ, emphasis was placed on reusability of the interfaces for different purposes. A combination of a GUI and a TUI has been chosen as the optimal combination in terms of reusability – a web interface has been determined as inefficient in terms of workload.

Acknowledgements

This work has been supported in part by the grant LTT 17018 of the Ministry of Education, Youth and Sports of the Czech Republic.



Presented at



Nov. 4 – 8, 2019 Adelaide, Australia