Current developments on ASDEX Upgrade data acquisition systems

M. Astrain, M. Michelini, C. Fuchs, G. Raupp, ASDEX Upgrade Team

Outline

- Context
- Data Acquisition at AUG
- Issues or weaknesses
- Introduction of new standards
- Conclusions

Context

- **Current software**
  - The current software for data acquisition has suffered modifications over 30 years.
  - Parts of the codebase attends to specific problems that have arisen during this time.
  - While there are documents and commented code there is no unified documentation.
  - Hard to change because it works. Operation must be maintained at all times.
- **The decision was made to re-think the whole framework from the ground.**
- **There is still an ongoing discussion on the focus of the machine for the next years from the physics standpoint (new requirements).**
- **The data acquisition needs are expected to keep growing. Specially if AI solutions are required in the future.**
- **This is an ideal moment as the people with the know-how is still active and there will be a machine shut-down for the new divertor 2022/2023.**

Data Acquisition at AUG

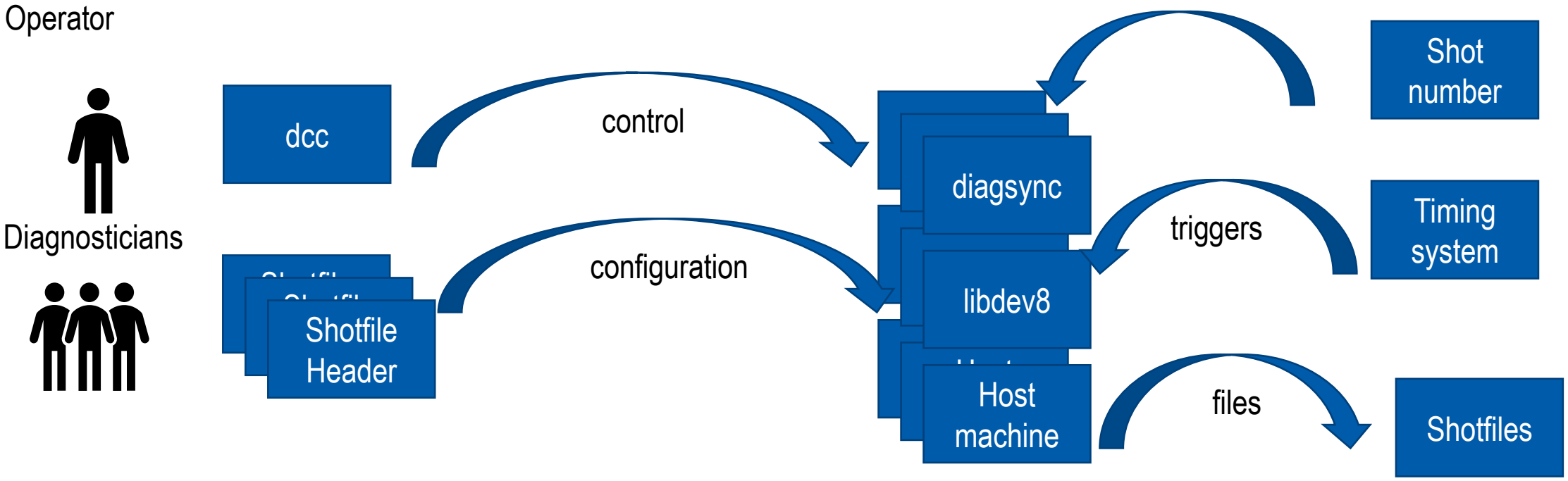
Hardware

- Custom hardware from timing to front end data acquisition with the in-house TDC and SIO systems.
- Very deep integration of all systems TDC,SIO and CAMAC in both SOLARIS and LINUX.
- Features are RT, modular data acquisition system and robust timing and timestamping.
- Still, advancements in the industry continued, and are very compelling.
- May be more efficient to integrate manufacturer solutions into framework.

Data Acquisition at AUG

Software

- **Optimized for operation and very specialized team.**
- **The current design is a monolithic driver design distributed as a library “libdev8“ to Solaris and Linux.**
- **This has the advantage of having a single library to maintain.**
- **The different data acquisition systems are all configured through the usage of binary files “shotfile headers”.**

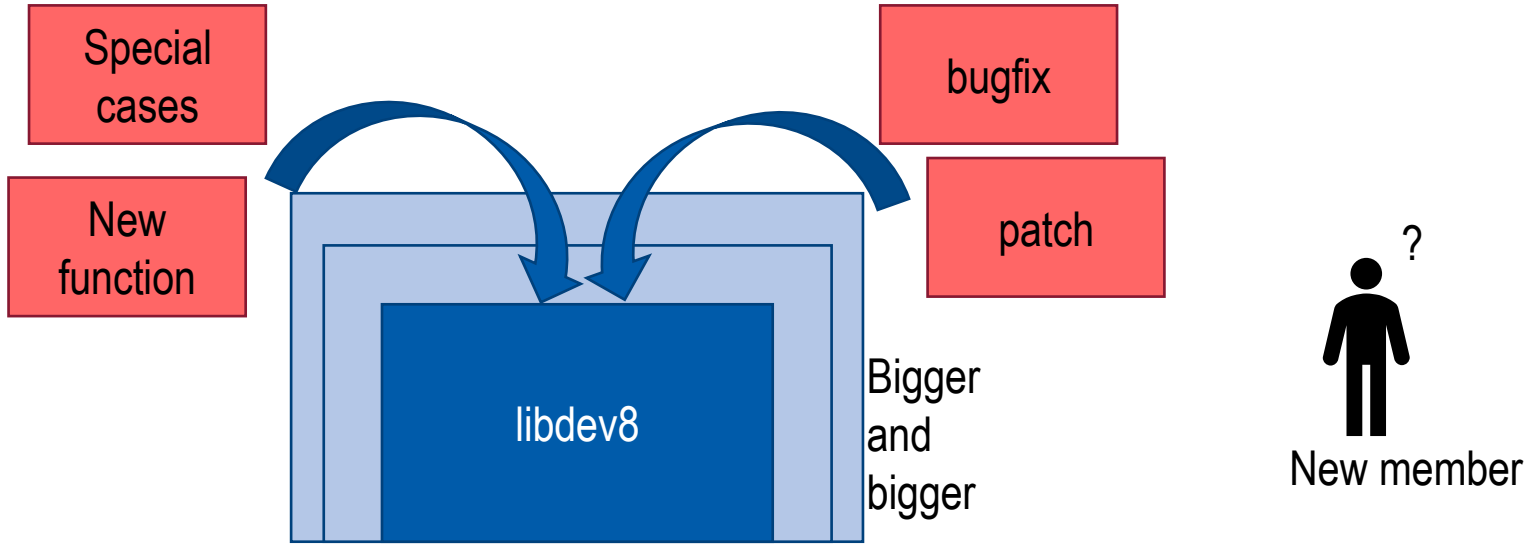


Issues and weaknesses

- **Seems simple enough with a couple of systems…..But there are 465 diagnostic identifying names!**

1. **No clear interfaces to introduce new systems.**
2. **The scale of the project is beyond the team capacity.**
3. **Hard to get new members and train them.**

At a point, the effort to introduce something new is bigger than building it again.



Introduction of new standards

- Clear need to introduce new standards of modern programing.
- Collaborative tools, issue trackers, documentation organization and collaboration.
- Do not re-invent the wheel. Many facilities are facing the same problems.
- Analyze open or available tools and introduce them to the AUG architecture.
- Hardware standard change: MTCA 90%
- Device driver standard introduction: NDS
- Some features from common control systems are nice to have: EPICS, TANGO… but not for now

Introduction of new standards

Hardware

- **Evaluation of in-house (SIO) solutions vs commercial ones**
- **There are many SIO systems that will need to be supported longer**
- **Commercial solutions in MTCA are expected to give best density of channels**
  - Old systems occupy a lot of rack space, this will be reduced severely
  - Modular MTCA also allows sensitive electronics to be produced in-house
  - Separation of backplane to merge old diagnostics into the same system reducing costs
- **PXI is still a compelling platform for rapid deployment of systems**
  - While mtca might be 90% suited, some systems have been replaced by PXI
  - This was required to substitute some systems more rapidly

Introduction of new standards

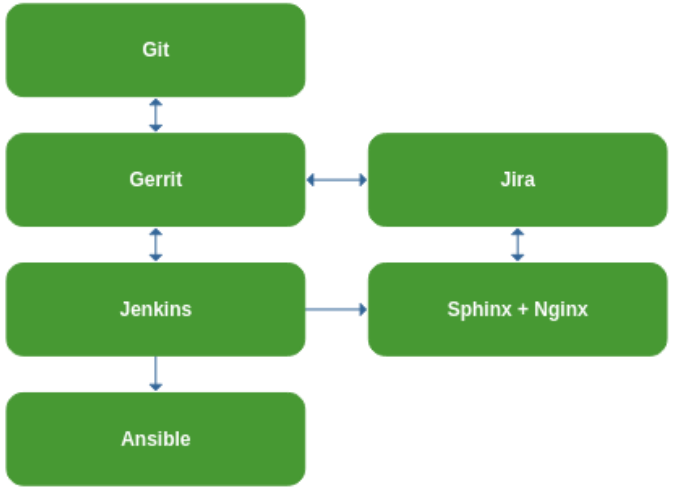
Software

Nominal Device Support v3 (NDS)

- ITER NDS is attractive looking into the future
- Drivers for PXI and MTCA
- Integration with different control systems

CS + Deployment

- The AUG plasma control system (DCS) has integrated many general purpose CS features in the later years
- However, many systems are still managed independent of DCS
- Toolchain modernization changes were already conducted from the DCS team side
- Git, Gerrit, Jira, Jenkins, Docker…
- Need to be integrated into the whole CODAC deployment system

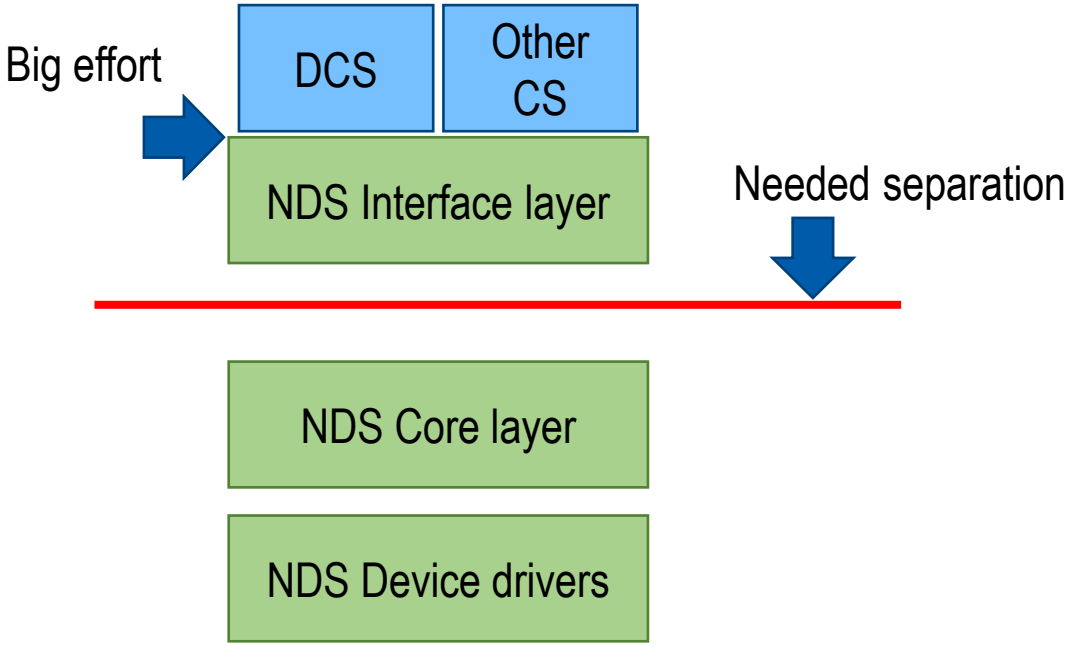


Introduction of new standards

Nominal Device Support v3 (NDS) drivers

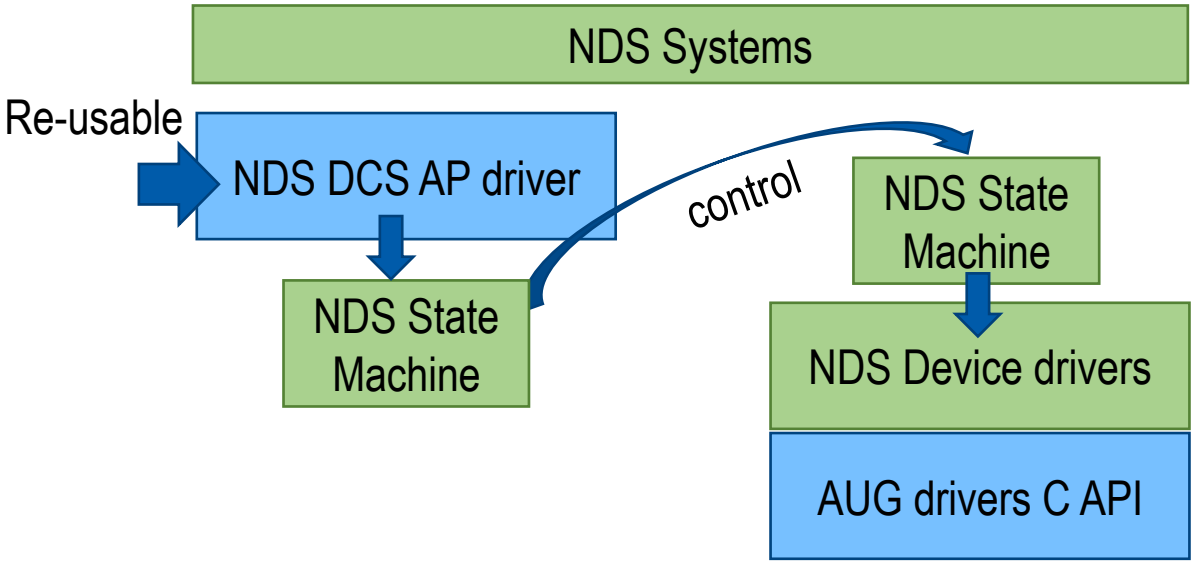
- Modular pure C++ drivers no old library dependencies
- Control System agnostic
- Project supported by ITER is gaining some mass
- Needed to develop cmake build system

- **Two ways of controlling other drivers**



DCS AP

A process controlled remotely by DCS with a simple interface



Conclusions

- There are different commercial solutions that may fit our needs
- Keeping in-house developments for sensitive electronics would be interesting
- Is the perfect time to have a wish list for future systems Physics going “beyond AUG”
- Looking into solutions for AI, hardware agnostic drivers
- Prototyping and brainstorm stage
- NDS introduced into cmake build system
- Prototype machine using CentOS 7 (still awaiting possible RHEL switch)
- Looking forward to share experiences with other facilities. Contact us!

Thank you