ITER Fast Plant System Controller Prototype Based on PXI Platform

M.Ruiz, J.Vega

CIEMAT/UPM team

Presented BY

Arun Veeramani, National Instruments

Universidad Politécnica de Madrid

Asociación Euratom/CIEMAT

Outline

- Project scope and requirements.
- FPSC HW elements.
- FPSC SW elements.
  - Applications running in the controller.
  - Data acquisition. FPSC control using EPICS PVs
  - Streaming/archiving applications
- Conclusions

Project Scope and Requirements: Plant System Controller

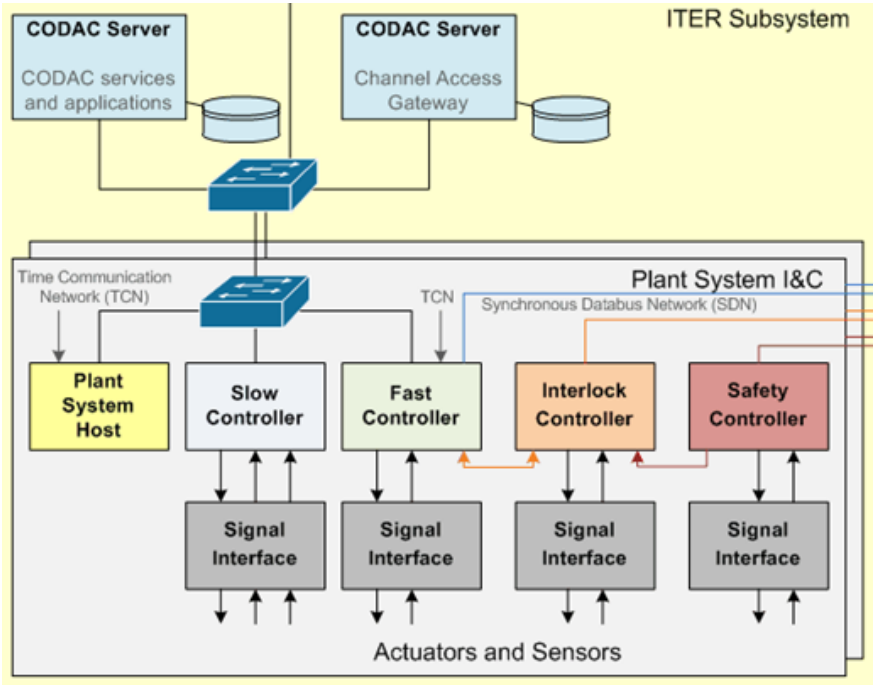
- Plant System Controller (PSC): Plant specific information on
  - Data acquisition
  - Control
  - Monitoring
  - Alarm handling
  - Logging
  - Event handling

- Two types of PSC:
  - Slow PSC
    - Based on industrial automation technology
    - Control loops rates <1 kHz
  - Fast PSC (FPSC)
    - Based on embedded technology
    - Stringent real-time requirements
    - Higher sampling rates

Project Scope and Requirements: FPSC

- Essentials requirements of FPSC:
  - Data acquisition and preprocessing
  - Interfacing with the networks (PON, TCN, SDN, streaming/archiving networks)
  - LINUX OS and EPICS IOC. System setup and operation using process variables.
  - COTS solutions

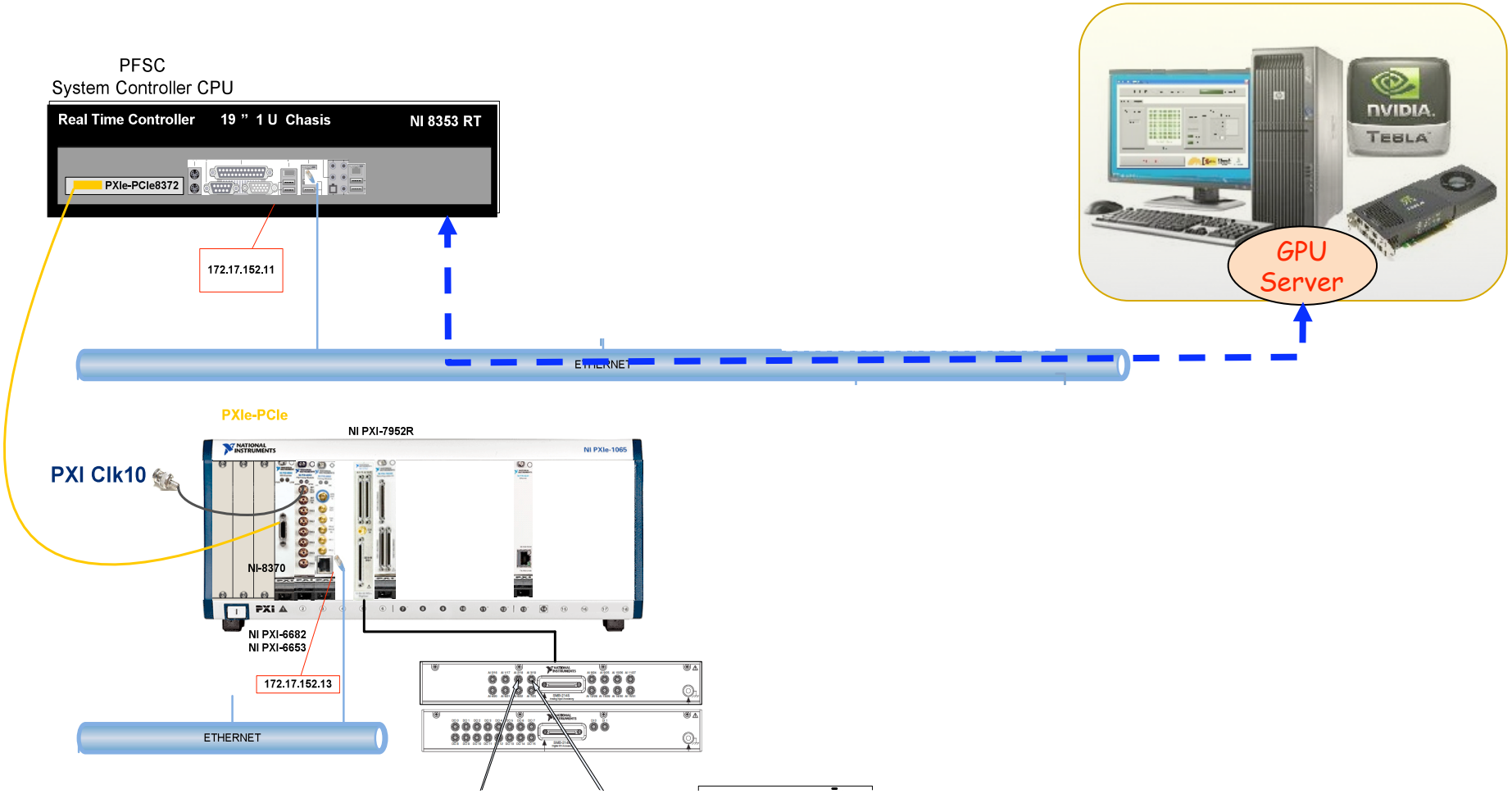
- Developing a prototype FPSC targeting Data Acquisition for ITER IO
  - A two steps approach: Alpha and Beta version.
- Challenges at the start of the project (2010)
  - Drivers (and device support) not available under Linux 64 bits
  - Complicated development to be finished in a limited time



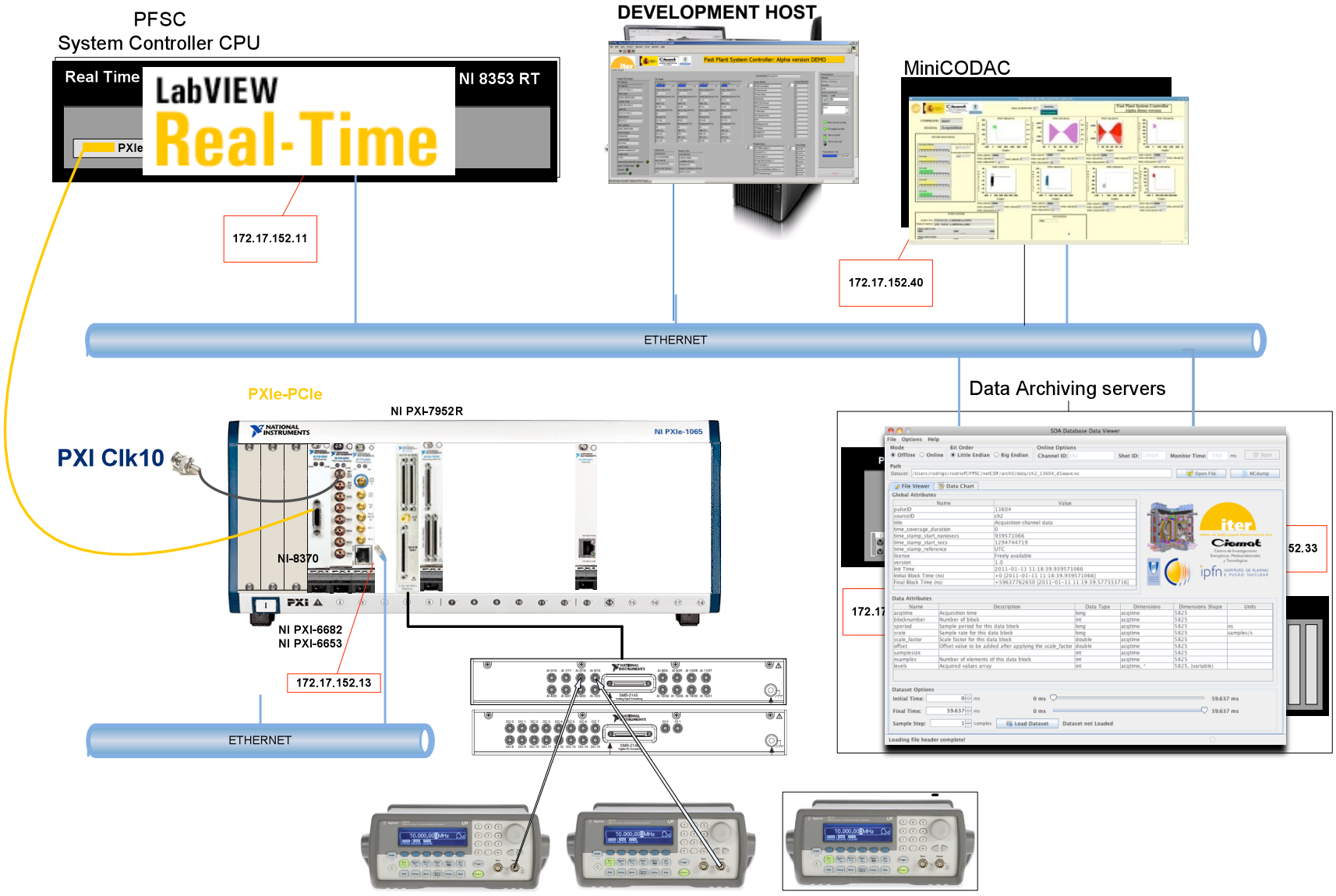
Alpha Version Solution: PXI Form Factor

- PXIe solution using:
  - National Instruments hardware (PXI chassis, timing modules, DAQ using FlexRIO and external controller)
  - LabVIEW RT Module applications running in the controller
  - LabVIEW FPGA for FlexRIO
  - LabVIEW EPICS Channel Access Server for real time targets
- Labview Real Time based
  - For quick prototyping and test system capabilities
  - Gain experience for beta version
- Specific application developed running in external computers for streaming/archiving, data processing with GPUs, and monitoring using ITER CODAC Core System.

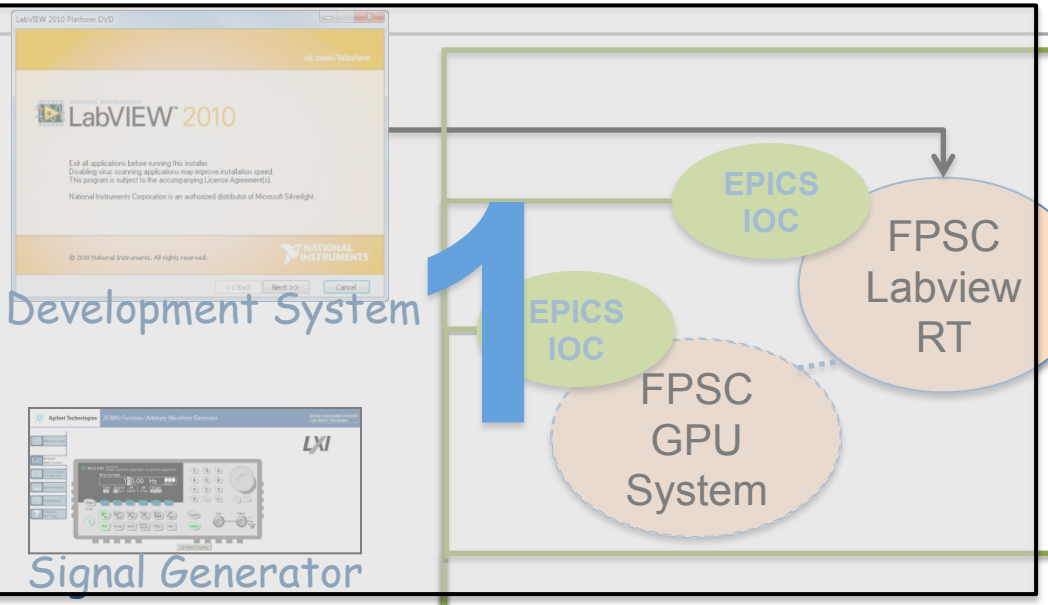
HW Elements: Block Diagram

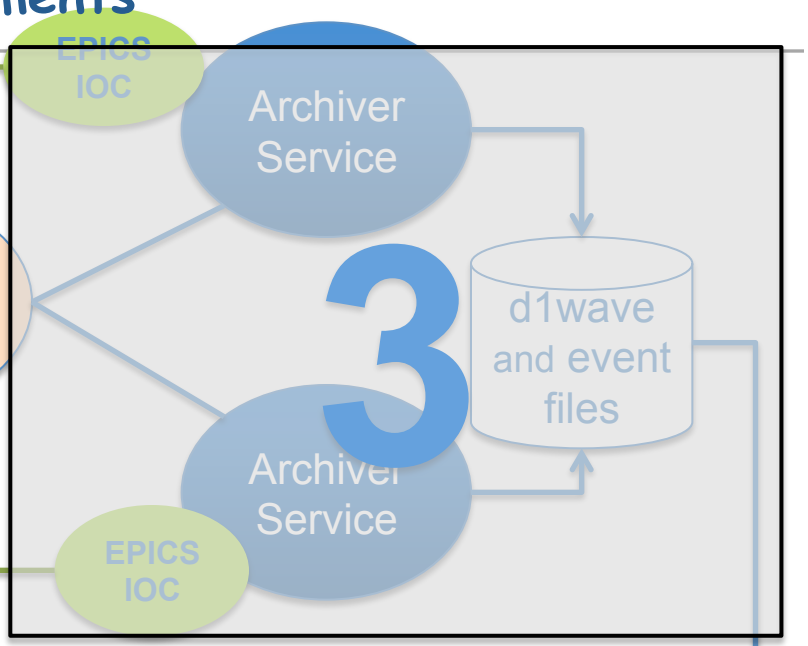


Development tools

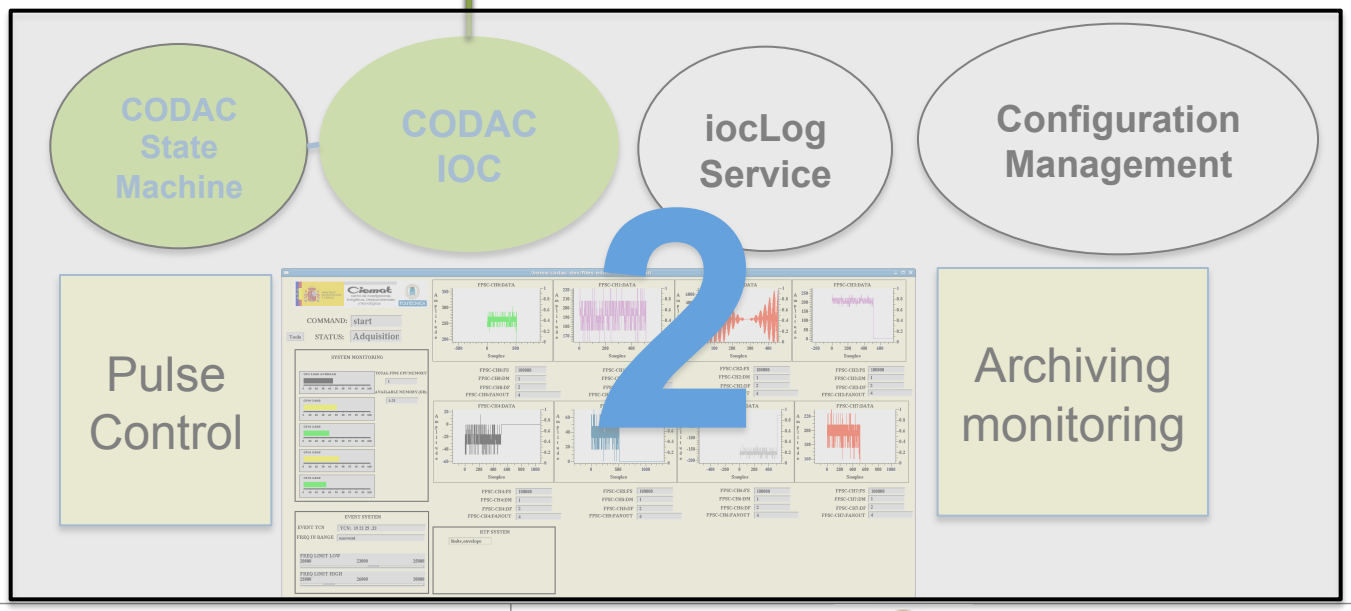


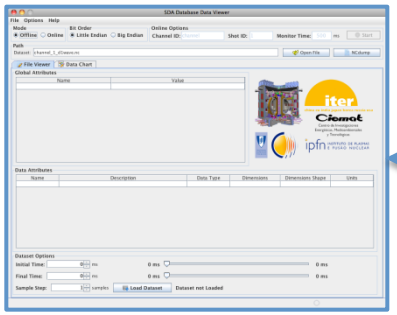
FPSC Software Elements





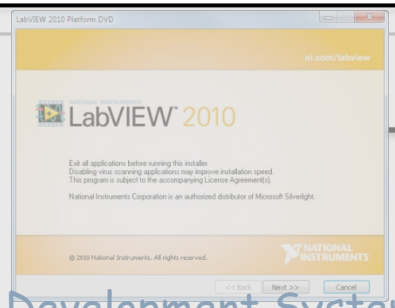
NFS





Archive Viewer

FPSC Software Elements



Development System

Signal Generator

1

EPICS IOC

EPICS IOC

FPSC Labview RT

FPSC GPU System

EPICS IOC

Archiver Service

d1wave and event files

Archiver Service

EPICS IOC

NFS

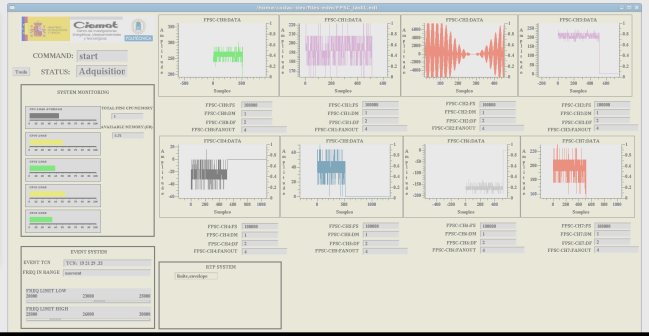
CODAC State Machine

CODAC IOC

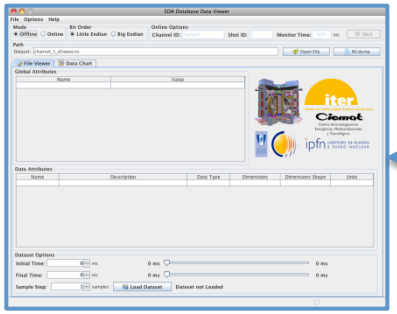
iocLog Service

Configuration Management

Pulse Control



Archiving monitoring



Archive Viewer

FPSC software elements

Development System

Signal Generator

EPICS IOC

EPICS IOC

EPICS IOC

EPICS IOC

FPSC Labview RT

FPSC GPU System

Archiver Service

Archiver Service

d1wave and event files

NFS

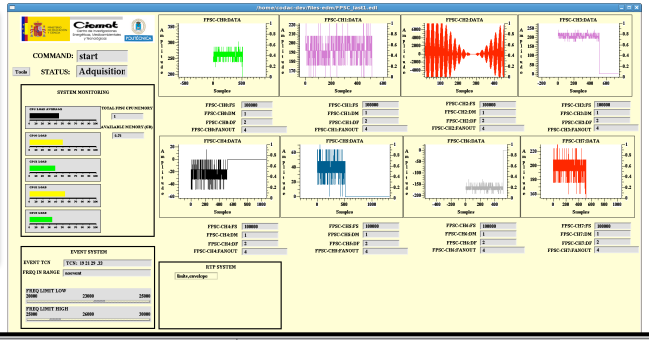
CODAC State Machine

CODAC IOC

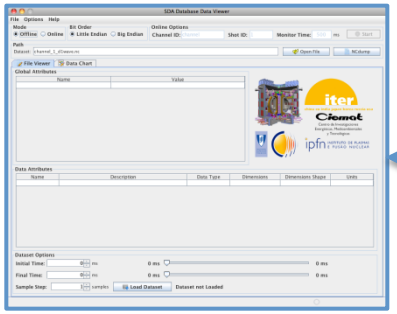
iocLog Service

Configuration Management

Pulse Control



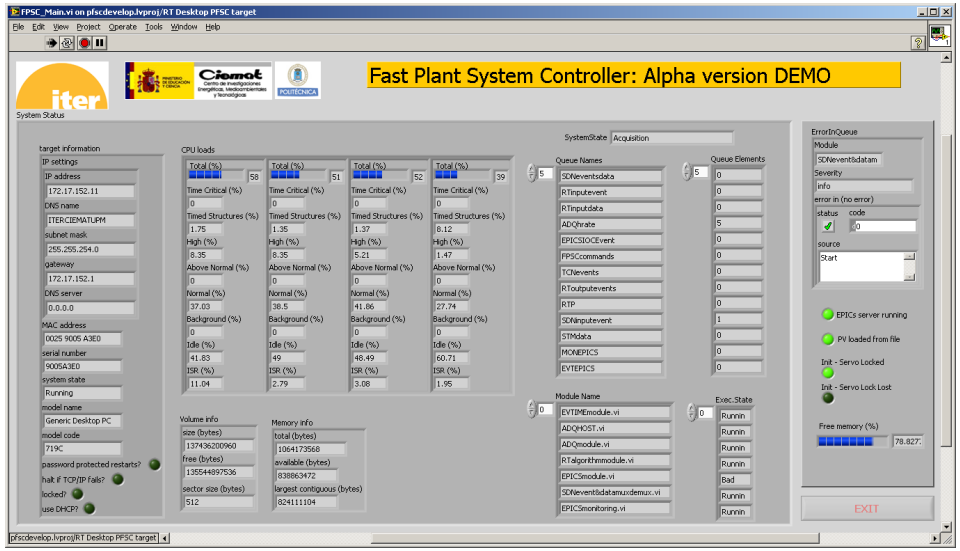
Archiving monitoring

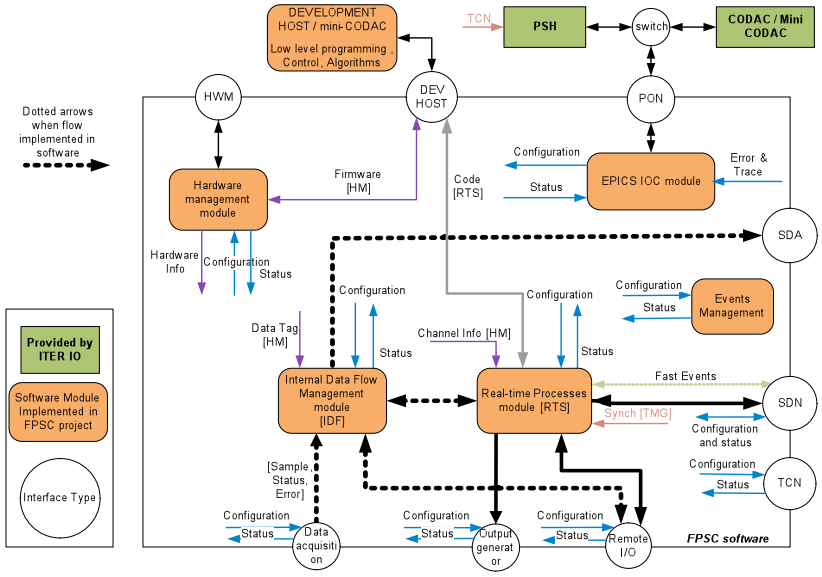


Archive Viewer

FPSC applications running in NI8353 computer

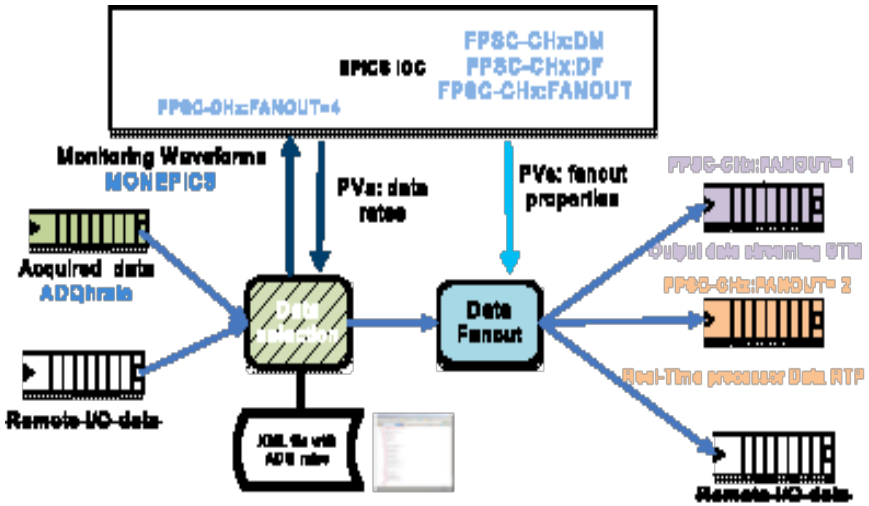
- LabVIEW Modules implemented:
  - CORE: General queues management. Creation, destruction and state machine control.
  - EPICS: Channel access and PVs management.
  - TCN: Management of PXI6653 and PXI6682 for clock generation and event time-stamping.
    - PXI CLK 10MHz is in phase with PXI6682 IEEE1588 clock
  - ACQ: Data acquisition and selection.
  - FPGADAC: Data acquisition application for RIO devices with time-stamping. Also include a signal simulator (inside FPGA) for debugging purposes.
  - EVT: Event management. SDN: Implemented using NI-Time Triggered Variables
  - RTP: Real time processing. Basic algorithms. RTPGPU: GPU management.



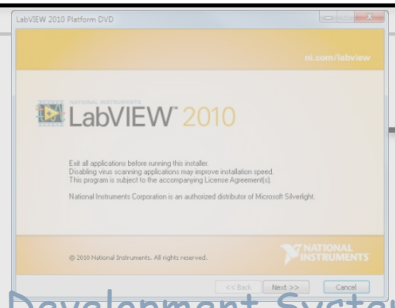


Highlights of FPSC software

- ADQ parameters are controlled & changed using PVs (also during the pulse):
  - Sampling rate and block size for FlexRIO device.
  - Decimation factor and modes (samples and blocks) for EPICS monitoring
- FPSC State machine control and status using PVs: start/stop, memory used, CPU load, etc.
- Acquired data can be sent to streaming, monitoring with EPICS, real time processing and GPU using «FANOUT PVs».
- Preprocessing algorithms can be dynamically selected using PVs.



FPSC Software Elements



Development System

Signal Generator

EPICS IOC

FPSC Labview RT

EPICS IOC

FPSC GPU System

EPICS IOC

Archiver Service

d1wave and event files

Archiver Service

EPICS IOC

NFS

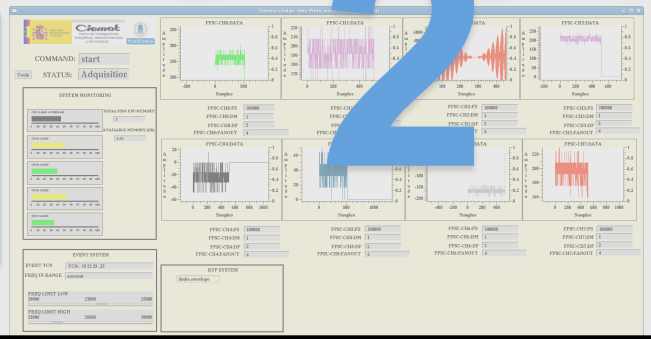
CODAC State Machine

CODAC IOC

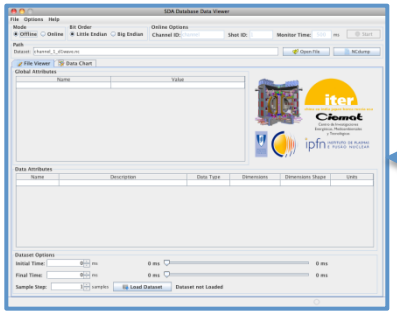
iocLog Service

Configuration Management

Pulse Control

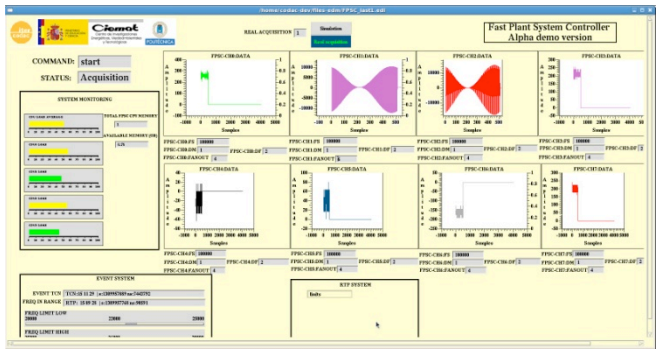


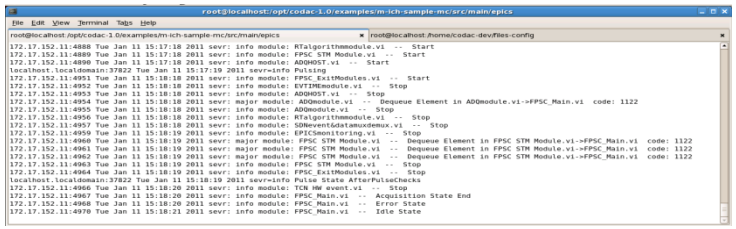
Archiving monitoring

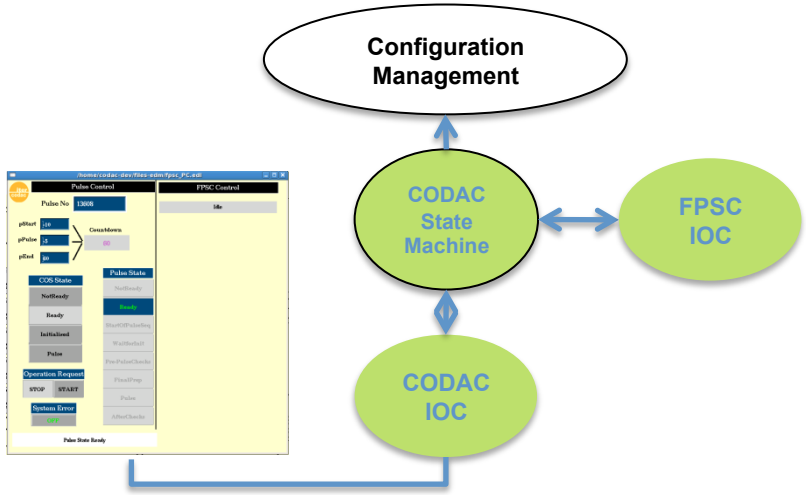


Archive Viewer

GUI using EDM (EPICS), LOG and states machine

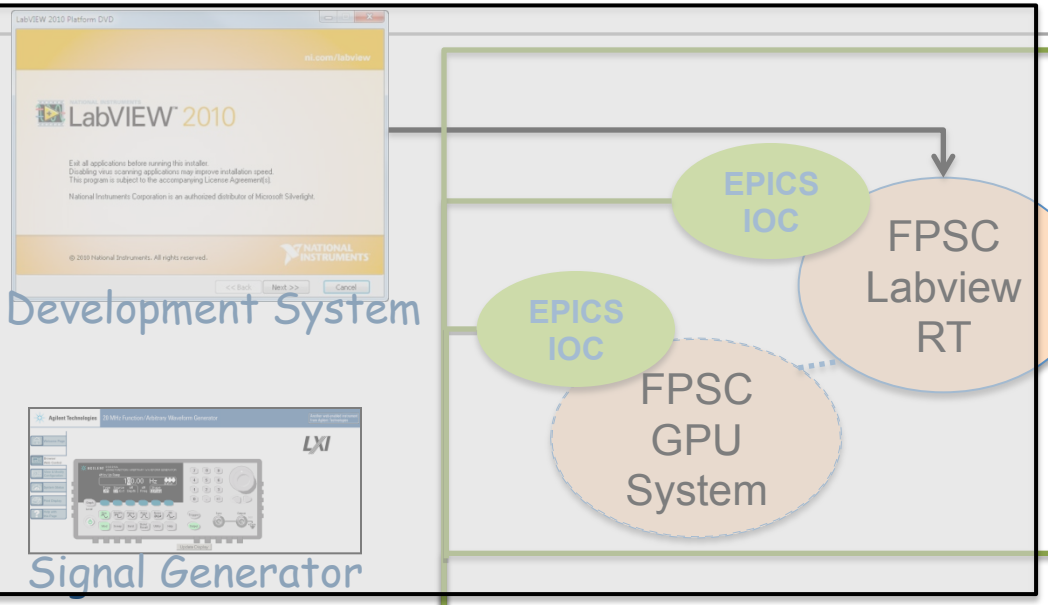


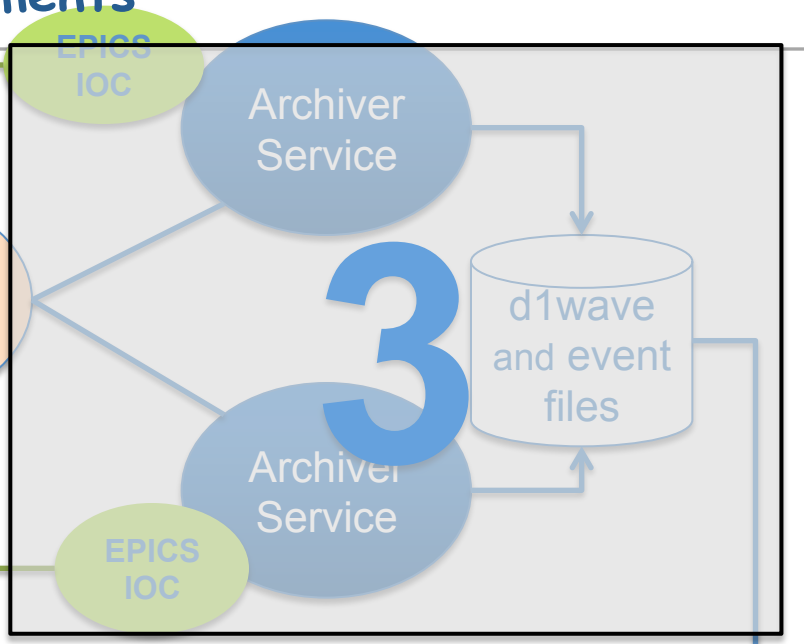




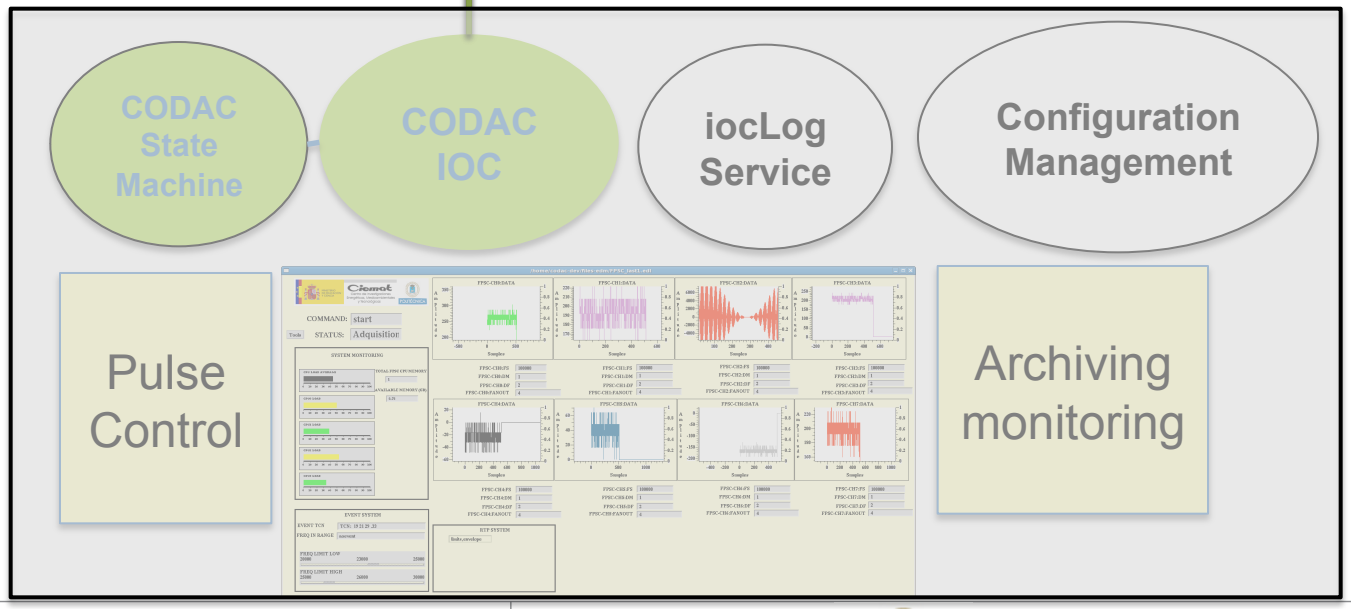
- Manual start/stop of FPSC
- Basic control of PVs during the pulse.
- Implementation of IocLog client in LabVIEW
- IOC with the pulse states machine and configuration management (XML files)

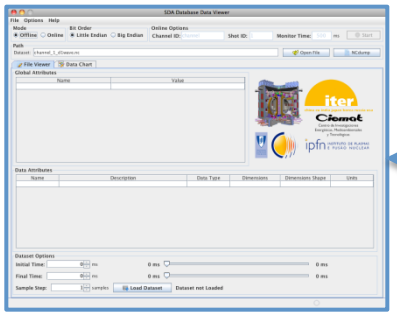
FPSC Software Elements





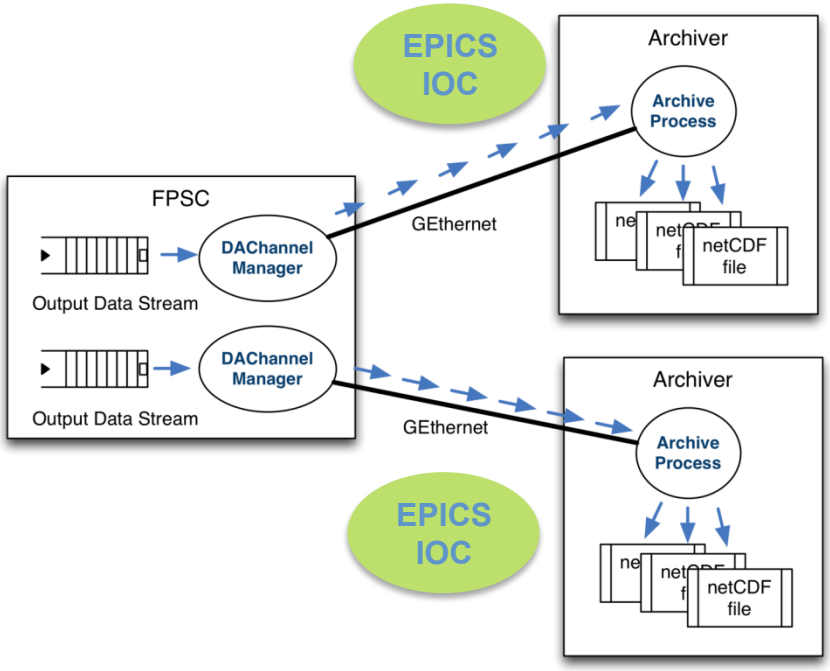
NFS





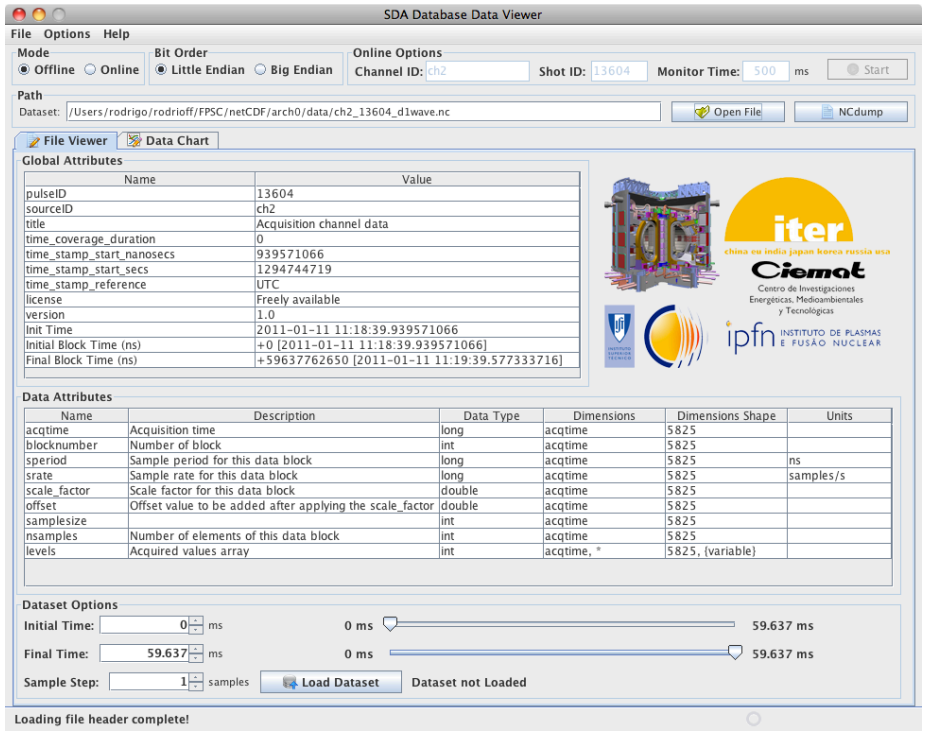
Archive Viewer

Archiving System



- Data sources can be assigned to data archivers
- netCDF file is the fundamental storage unit
- A file per data source (signal) and pulse
- Two types of data are currently implemented: "d1wave" and "event".
- EPICS IOC currently used for monitoring

Archiving Viewer and monitoring



- "Online" and "Offline" mode
- On remote via NFS (Network File System)
- Time slice positioning
- Self Description data visualization
- Flexible plotter
  - Zooms
  - Export options
- Completely based on EPICS channel access
  - Every archiver implements its own EPICS IOC
- System variables:
  - CPU load
  - Memory Usage
- Archiving system performance
  - Receiving data rate per channel
  - Total received data rate
  - Storing data rate per channel
  - Total saved data rate

Conclusions

- Implementation of a basic FPSC devoted to data acquisition following essential ITER requirements:
  - "Intelligent data acquisition" using FPGA DAQ devices with IEEE-1588 time-stamping.
  - System DAQ parameters controlled by EPICS' PVs (changed dynamically during the PULSE)
  - Streaming capabilities.
  - Preprocessing algorithms using local processor and GPU (controlled with EPICS PVs).
  - Integration with EPICS CODAC system (v1.1).
  - 100kS/s per channel with streaming, time-stamping, EPICS monitoring, and 2 channels preprocessing
- LabVIEW based tools (RT/FPGA) have been a good choice for quick prototyping in a short period of time (3 months).
  - Graphical oriented design simplifies: the definition of complex software models , the debugging of the different applications, and the test of complex hardware setups.

ITER Fast Plant System Controller Prototype Based on PXI Platform

Thank you for your attention!!

CIEMAT: J. Vega, R. Castro

ITER: N. Utzel, P. Makijarvi.

Technical University of Madrid: M. Ruiz, J.M. López, E. Barrera, G. Arcas, D. Sanz & J. Nieto

Thank you to NI for the strong support in the development of this project