CONTROL AND DATA ACQUISITION SYSTEMS FOR THE FERMI@ELETTRA EXPERIMENTAL STATIONS*

R. Borghes, V. Chenda, A. Curri, G. Gaio, G. Kourousias, M. Lonza, G. Passos R. Passuello, L. Pivetta, M. Prica, R. Pugliese, G. Strangolino
Sincrotrone Trieste, Trieste, Italy

Abstract

FERMI@Elettra is a single-pass Free Electron Laser (FEL) user-facility covering the wavelength range from 100 nm to 4 nm. The facility is located in Trieste, Italy, nearby the third-generation synchrotron light source Elettra. Three experimental stations (Fig. 1), dedicated to different scientific areas, have been installed in 2011: Low Density Matter (LDM), Elastic and Inelastic Scattering (EIS) and Diffraction and Projection Imaging (DiProI). The experiment control and data acquisition system is the natural extension of the machine control system. It integrates a shot-by-shot data acquisition framework with a centralized data storage and analysis system. Low-level applications for data acquisition and online processing have been developed using the Tango framework on Linux platforms. High-level experimental applications can be developed on both Linux and Windows platforms using C/C++, Python, LabView, IDL or Matlab. The Elettra scientific computing portal allows remote access to the experiment and to the data storage system.

INTRODUCTION

The newly-built free-electron laser FERMI@Elettra generated its first flashes of coherent light in the far ultraviolet in December 2010. A second phase of commissioning started in January 2011, with the goal of providing optimized FEL light to the experimental chambers [1]. The "first-shots" experimental activities, held in March and in July 2011, were mainly devoted to the commissioning and tuning of the three existing end-stations: LDM, EIS and DiProI.

The FERMI@Elettra fast data acquisition (DAQ) system is used to acquire, store and correlate multiple data sources. Based on the estimated scientific data throughput, the DAQ system has been designed in order to guarantee a data acquisition rate up to 1400 Gbyte/hour. During the experiment, scientific data is saved on a centralized storage system using a 10Gb ethernet connection.

Once stored, experimental data are available for offline data processing using the Elettra high performance computing clusters. Besides this, experimental data and local computing infrastructures can be remotely accessed through the Elettra scientific computing portal.

DATA ACQUISITION FRAMEWORK

FERMI@Elettra DAQ framework inherits from two years of experience gained during the machine commissioning and has been thought as the natural extension of the machine control system [2]. For every single laser pulse, at 50 Hz repetition rate, the DAQ system acquires, stores and links data coming from scientific instrumentation and machine diagnostics.

Control System Hardware Architecture

Dedicated scientific instrumentation apart, a uniform hardware architecture has been adopted for all the FERMI@Elettra experimental stations. According to Fig 2, at top there is a workstation dedicated to application software. Via a private VLAN, it accesses two low-level machines for interfacing instrumentation and a PLC dedicated to protection of the equipment.

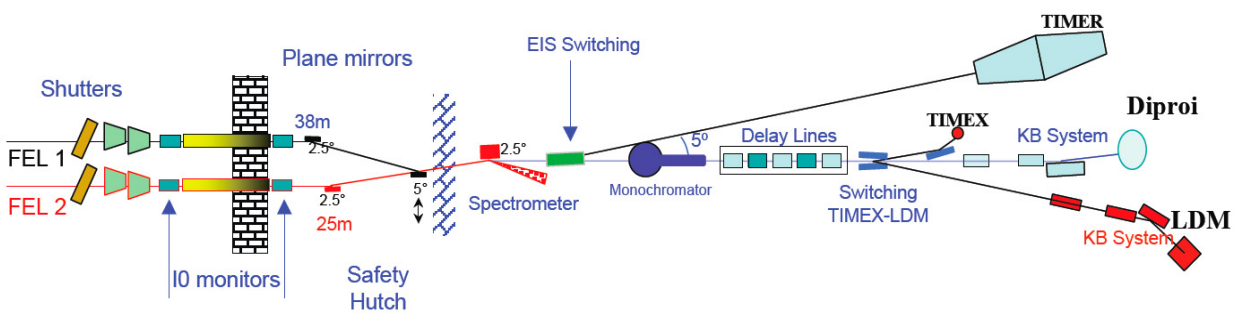

Figure 1: FERMI@Elettra beamlines layout.

*Work supported by the Italian Ministry of University and Research under grants FIRB-RBAP045JF2 and FIRB-RBAP06AWK3

The workstation is an Intel Core i7-950 multi-screen desktop computer running either Linux or Windows.

Standard instrumentation is controlled by a VME system equipped with an Emerson MVME-7100 PowerPC board (Equipment Controller – EC). The EC also hosts the Event Receiver (EVR) by Micro-Research, that receives the machine bunch trigger via a fiber optics infrastructure. The EVR board can generate software interrupts and digital signals with a maximum jitter of 20ps. Instruments requiring special hardware interfaces or proprietary software are controlled and acquired by an Intel-based rack-mount server (Pentium XEON Quad Core). Both PowerPc and rack-mount Intel server are directly involved in real-time data acquisition. They are based on Linux (Kubuntu 10.04) with the Xenomai real-time extension [3].

Machine real-time diagnostics data are transparently shared among data acquisition computers using a network software application called Network Reflective Memory (NRM) [4]. The NRM system has been deeply tested during the commissioning period. It works on a fully dedicated, private network that covers the whole machine from the electron gun to the experimental hall.

Finally, a Siemens S7 series 300 PLC is used for the interlock system in order to protect equipment and devices from damage.

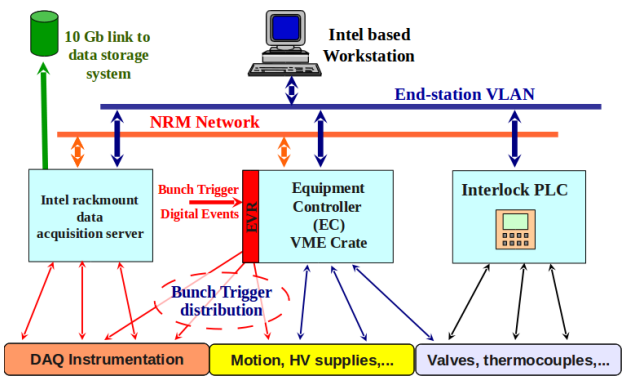

Figure 2: DAQ system hardware architecture.

Control System Software Platform

The DAQ system of the experimental stations is based on the Tango framework [5]. Low-level instrumentation control, scan implementation, data acquisition and analysis is done through Tango device servers. The low level device servers have been developed using C++, while the higher level ones have been written in Python and C++.

Each experimental station control system uses a private Tango database running on a virtual machine in the central system main server. Communication with the machine control system is made possible by special Tango devices that act as gateways. These are configurable servers that export specific attributes and commands between control system running on different VLANs.

As a general rule, automation and complex data acquisition operations are performed at Tango device server level, while client-side GUIs are used only for visualization purposes.

Graphical interfaces can be developed by beamline staff on Linux and Windows platforms using common frameworks like Labview, IDL, Python or Matlab. Applications developed by controls specialists have been written using QTango [6], a Qt based framework developed in house.

SCIENTIFIC DATA PIPELINE

Each laser pulse produced by FERMI@Elettra is unique in terms of spatial position, photon flux and energy spectrum. It is crucial to acquire and store *shot by shot* not only experimental data coming from the beamline detectors but also photon beam diagnostics data.

Every acquisition device server maintains and updates a circular memory buffer in which the acquired data are tagged with an increasing counter named *bunch number*. These low level devices export tagged data to the higher level storage device servers that organize experimental data in HDF5 (Hierarchical Data Format) [7] archives.

A challenging task in terms of data throughput is the acquisition of scientific detectors: until now only 1Mpixel detectors have been used but in the near future these will be upgraded to 4 Mpixel.

Table 1: Estimated Experimental Data Throughput

Endstation	FEL @ 10Hz	FEL @ 50Hz
DIPROI	140 Gbyte/hour	700 Gbyte/hour
LDM	280 Gbyte/hour	1400 Gbyte/hour
EIS	66 Gbyte/hour	330 Gbyte/hour

Data Storage and Analysis Resources

For each experiment a specialized Tango DAQ device performs data collection, online data reduction and storage in the HDF5 format.

Data archives are directly saved through the NFS protocol on a high speed centralized storage via a 10Gb ethernet connection.

The present centralized storage system is based on a 70 TB EMC Storage Area Network (SAN). Its next planned upgrade will provide FERMI users with a 1 PB distributed filesystem that will grant a data throughput of 1600 MB/sec.

Experimental data can be consequently analysed using the Elettra scientific computing infrastructure that currently includes:

- two state-of-the-art High Performance Computing Clusters equipped with InfiniBand and over 250 processor cores;
- an interactive computing cluster with about 40 processor cores;
- a cloud system, with a growing number of cores (presently 30).

The above resources can be remotely accessible through a specialized web portal: the Virtual Control Room (VCR) [8].

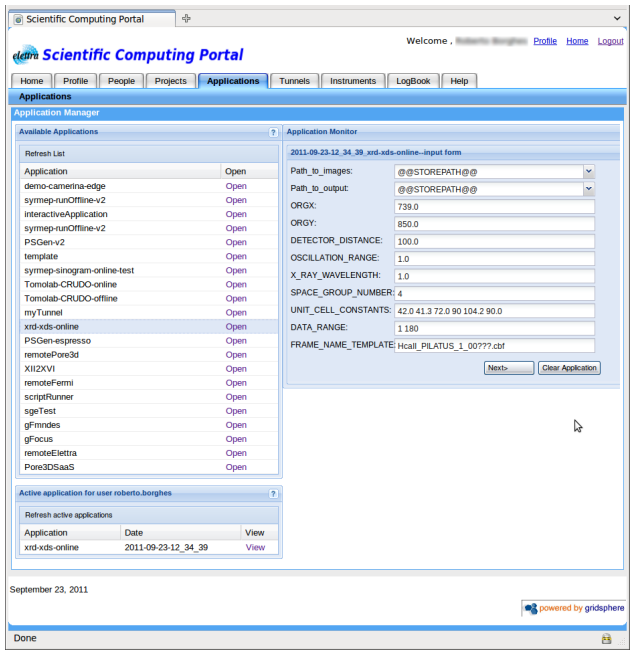

Figure 3: Elettra VCR web portal.

Virtual Control Room

The VCR is an open source web portal based on the GridSphere framework and Web 2.0 technologies. It is a highly configurable tool, designed for easy development and deployment of custom applications with transparent access to the underlying infrastructure. Among its main features are a safe tunnelling tool and an application manager (Fig. 3) that integrates a Jython-based scripting environment. The VCR acts as a front-end to the gLite-based Grids and to grid-enabled remote instrumentation via Instrument Element [9]. A number of tools like e-logbook, help-system and users-browser, make it a collaborative environment.

The portal provides secure remote operations of beamlines and experimental stations. It allows access to raw datasets and enables initiation of data processing during the acquisition phase. As such, the VCR is virtually suitable for any kind of scientific or business application and for the associated user communities.

CONCLUSIONS AND OUTLOOK

The FERMI@Elettra fast DAQ system has been tested during "first-shots" experimental activities held in March and July 2011. Working with a source of 10Hz repetition rate it has proven to be reliable, stable and dynamic enough for an experimental environment.

Further hardware and software upgrades will be carried out in the next months in order to handle a 50Hz repetition rate and 4Mpixel scientific detectors.

The efforts will be in accordance to the PANDATA ODI[10] related activities that include data policy, format standardisation, automatic metadata association and data management.

REFERENCES

[1] S. Di Mitri, "Commissioning and Initial Operation of FERMI@Elettra", IPAC 2011, San Sebastián, September 2011.
[2] M. Lonza et al., "Status report of the FERMI@Elettra control system", these proceedings.
[3] [3] http://www.xenomai.org
[4] L. Pivetta et al., "The FERMI@Elettra distributed real-time framework", these proceedings.
[5] http://www.tango-controls.org
[6] G. Strangolino et al., "QTango: a Qt Based Framework for Tango Graphical Control Panels", ICALEPCS 2009, Kobe, Japan, October 2009.
[7] http://www.hdfgroup.org
[8] http://www.dorii.eu/resources:adaptation:middleware:VCR
[9] http://www.dorii.eu/resources:adaptation:middleware:IE
[10] http://www.pan-data.eu