DOI: 10.17199/NOBUGS2016.36

DonkiOrchestra: a scalable system for data collection and experiment management based on ZeroMQ distributed messaging

Borghes Roberto, Kourousias Georgios
Elettra Sincrotrone Trieste, ITALY

E-mail: roberto.borghes@elettra.eu

Abstract. Synchrotron and Free Electron Laser beamlines consist of a complex network of devices. Such devices can be sensors, detectors, motors, but also computational resources. The setup is not static and is often upgraded. The data acquisition systems are constantly challenged by such continues changes and upgrades, so a constant evolution of software technologies is necessary. DonkiOrchestra is a TANGO based framework developed at Elettra Sincrotrone Trieste that takes full advantage of the ZeroMQ distributed messaging system and supports both data acquisition and experiment control. In the DonkiOrchestra approach, a TANGO device referred to as Director, provides the logical organization of the experiment as a sequential workflow relying on triggers. Each software trigger activates a set of Players that can be hierarchically organized according to different priority levels. This allows for concurrency and map-reduce strategies. Data acquired by the Players is tagged with the trigger number and sent back to the Director which stores it in suitably structured HDF5 archives. The intrinsic asynchronicity of ZeroMQ maximizes the opportunity of performing parallel operations and sensor readouts. This paper describes the software architecture behind DonkiOrchestra, which is fully configurable and scalable, so it can be reused on multiple endstations and facilities. Furthermore, experimental applications, performance results and future developments are presented and discussed.

1. Introduction

The research centre Elettra hosts two advanced particle accelerators: the electron storage ring Elettra and the free-electron laser (FEL) FERMI hosting 34 experimental stations.

Elettra is a third-generation synchrotron light source that provides state-of-the-art techniques to lead experiments in physics, chemistry, biology, life sciences, environmental science, medicine and cultural heritage. Currently 28 beamlines utilize the radiation generated by the Elettra source. It can operate at two different electron energies: 2.0 GeV for enhanced extended ultraviolet performance and spectroscopic applications; 2.4 GeV for enhanced x-ray emission and diffraction applications. A substantial facility upgrade, named Elettra 2.0, is currently planned [1].

FERMI is a seeded free electron laser (FEL) facility operating in the ultraviolet and soft x-ray range. FERMI has been developed to provide fully coherent ultrashort 10-100 femtosecond pulses with a peak brightness ten billion times higher than that made available by third-generation light sources. It is a single-pass FEL that comprises two separate coherent radiation sources: FEL-1 operates in the wavelength range between 100 and 20 nm via a single cascade harmonic generation, while the FEL-2 is designed to operate at shorter wavelengths (20-4 nm) via a double cascade configuration.

The Scientific Computing team manages a set of core services spanning from beamline control and data acquisition systems to Cloud computing and storage resources, plus a set of web based services for e-Science and Scientific Business management. The data acquisition and experiment management system of all the FERMI endstations is based on FermiDAQ [2], a common software framework highly configurable and scalable. The use of a single software infrastructure showed a remarkable enhancement of the team efficiency, but it is deeply dependent on the FERMI pulsed source architecture. So we focused on a new version of the software, reusable on a wider spectrum of scientific endstations and facilities.

The next sections provide the reader with a technical overview of the new framework, called DonkiOrchestra, its experimental applications and future developments.

2. DonkiOrchestra architecture overview

Synchrotron and Free Electron Laser beamlines consist of a complex distributed network of devices, like sensors, detectors, motors, but also computational resources. Each scientific setup is unique, not static and is periodically upgraded with new instrumentation. At the base of the design of the new framework DonkiOrchestra there are few fundamental goals:

- to maximize scalability and customizability;
- to enhance synchronized parallel operations;
- to minimize data communication overheads.

DonkiOrchestra draws on experience gained with the FEL data acquisition system FermiDAQ. Any experiment at FERMI is based on the acquisition of a train of light pulses with a frequency of 50Hz. Experimental instrumentation is synchronized with the arrival of the light pulses through an electric trigger signal which is tagged with an incremental index. Such a system forces the parallelism of the the control system and acquired data may be easily correlated through the photon pulse index.

This approach of organizing a scientific experiment as a train of independent phases has been inherited by DonkiOrchestra, with the main difference that the synchronization trigger is not a hardware signal but a software event shared through the instrumentation network.

As in a symphonic orchestra the system is composed by a Director and multiple independent Players. Each Player belongs to a Priority group and has a specific task to execute. The Director conducts the experimental sequence by sending a train of software triggers to the Players. For each step of the experimental sequence, a trigger signal is sent to the Players with the highest Priority 0, then to the group of Players with Priority 1 and so on. Each Player executes its task upon the arrival of the trigger and sends back to the Director an acknowledge event. A Player associated to a scientific sensor acquires the data, puts a tag on it with the trigger number and sends it back to the Director for storing it in suitably structured archives. In Figure 1 is schematically shown how a standard 2D scan experiment may be implemented with the described approach.

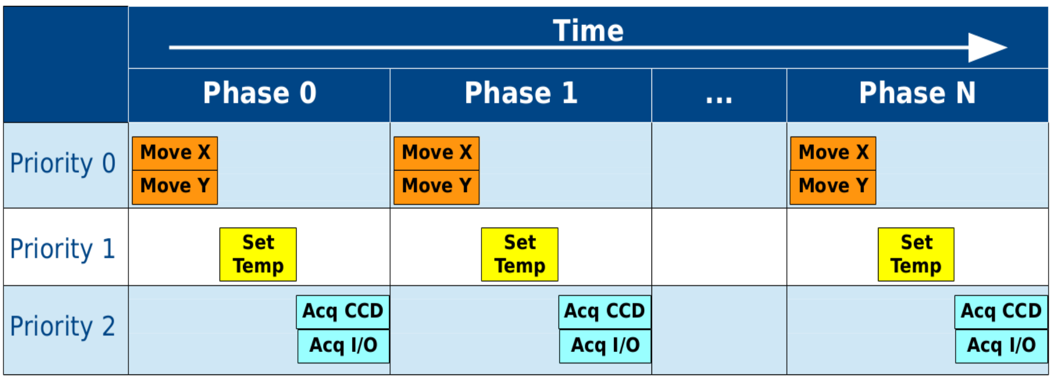

Figure 1. A 2D scan experiment organized as a triggered sequence.

3. DonkiOrchestra technical design

DonkiOrchestra has been written using TANGO [3], a robust and easy to use distributed control system toolkit. TANGO is operating system independent and supports C++, Java and Python for all its components.

DonkiOrchestra Director and Players are independent TANGO devices distributed on different computers connected through an ethernet network. As core language of the new framework we decided to use Python that adds dynamicity and portability. This choice affects only the Director device, since the Player devices may be developed using also the C++ or the Java TANGO binding.

ZeroMQ [4] messaging system has been chosen for its asynchronous I/O model that fits the need of having a scalable distributed application. It has a score of language APIs and runs on most operating systems. Besides this, the TANGO framework recently adopted ZeroMQ for its event system.

One of the project starting points was the use of ZeroMQ messages for scientific data sharing, so we performed a few preliminary reliability tests of the protocol and a set of efficiency tests to evaluate its performance at the increase of the data chunk size. Our particular interest was the measurement of the time spent by a Player for publishing a big chunck of data. Figure 2 shows the results of a series of tests that put the time spent for a ZeroMQ push() operation in relation with the data message size. The comparison of the different curves clearly shows that a TANGO User-Event has the same performance of the basic ZeroMQ API function. The test has been performed using Tango 8.1.2.c, ZMQ 3.2.3 on two x64 Linux machines connected through a 10Gb ethernet connection.

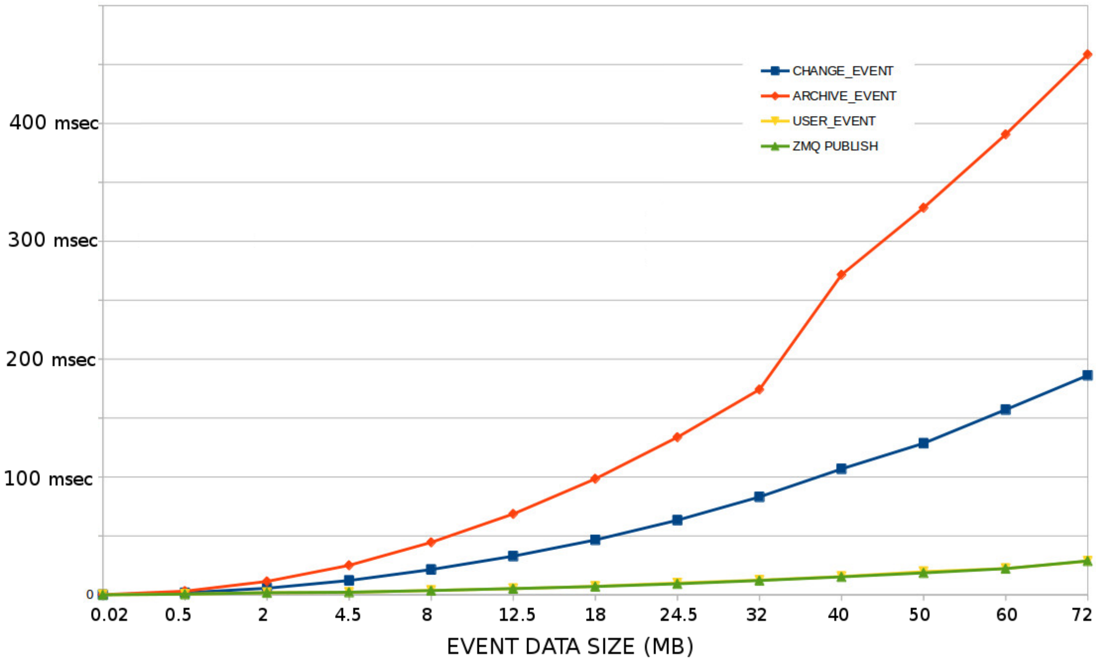

Figure 2. Time spent for a ZeroMQ publish() function vs message data size.

The Director device has been designed with a configurable, scalable and reliable structure. Schematically, an experiment sequence is composed by three consecutive phases shown in Figure 3.

- In the planning phase the Director reads the address of the Player devices from an XML file, retrieves their priorities and sends them a preparation signal. During this preliminary phase each Player sets up the sequence of operations to be performed upon the arrival of the event triggers, e.g. a motion player may prepare the sequence of target positions for a scan.

- During the collection phase the Director sets up and conducts the train of ZeroMQ trigger events. Each trigger message is composed by an index number and a priority tag. For each main experimental phase, the Director sends a trigger to the group of Players with the highest priority, waits for all the acknowledge messages, then proceed with the next priority group.
- In the closeout phase the Director sends to the Players a specific signal in order to perform any needed closing procedure.

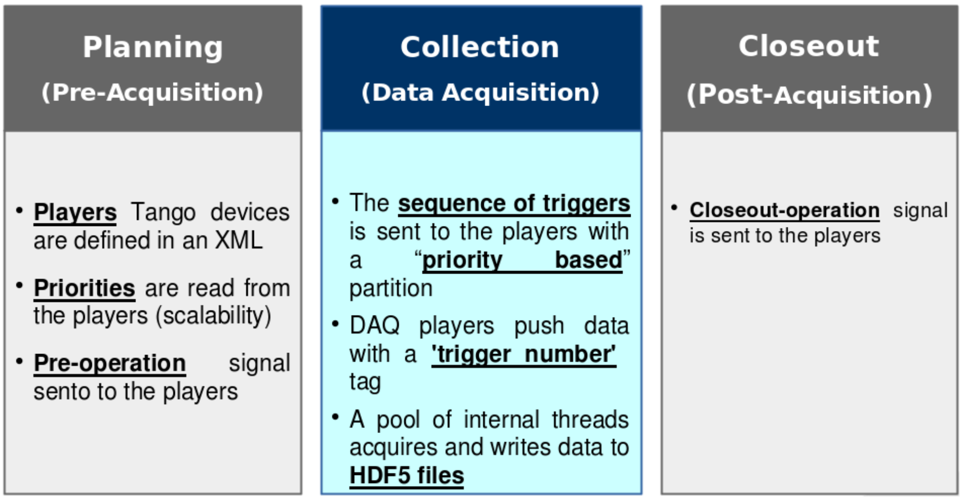

Figure 3. DonkiDirector experiment phases.

Concurrently to the experiment execution, a pool of internal threads acquires and saves asynchronous data messages coming from the acquisition Players. The structure of a data message is composed by the experimental data and a trigger index information, so data messages coming from different sources may be easily correlated. The Director device stores scientific data in HDF5 [5] binary archives. Experimental data may be integrated with environment information by defining some metadata sources in the Director XML configuration file. A simplified schema of the explained architecture is shown in Figure 4.

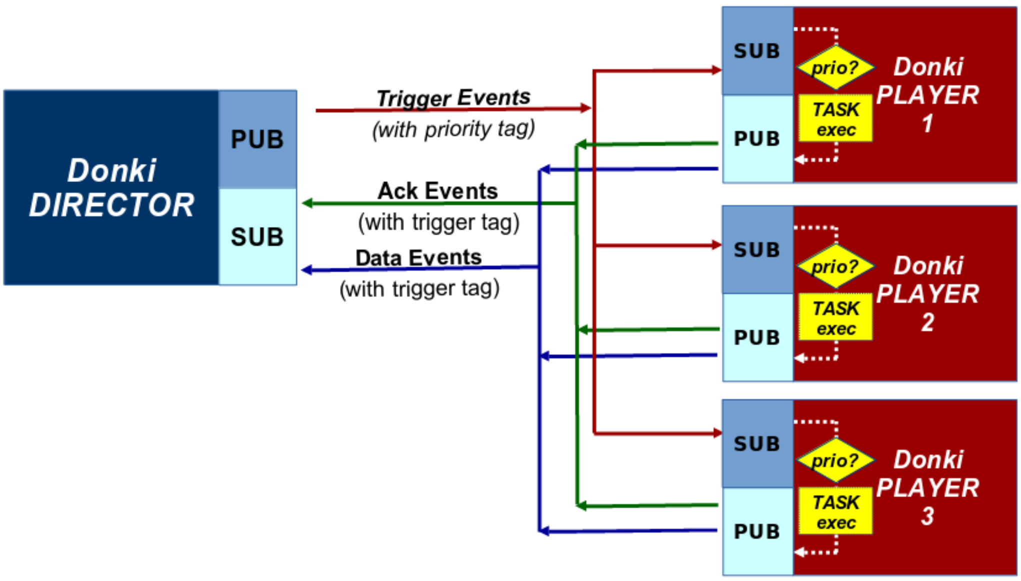

Figure 4. DonkiOrchestra schematic architecture.

4. DonkiOrchestra at Twinmic beamline

European scientists with highest expertise in X-ray microscopy, diffractive X-ray optics, X-ray contrast technologies and detection, started in 2001 to integrate the advantages of complementary scanning and full-field imaging modes into a *single* instrument, which they named 'TwinMic'. Since its opening to the users in late 2007, the TwinMic beamline at the Elettra synchrotron has been attracting the interest of a broad life science community. Using simultaneously the x-ray transmission and emission detection systems it is possible to perform elemental, absorption and phase contrast imaging that provides complementary chemical and morphological information for the sample under investigation.

The extreme versatility of the Twinmic beamline allows to plan a wide range of different scientific experiments. TwinMic beamline is a prefect field test for DonkiOrchestra, so we installed and adapted it in order to control the TwinMic instrumentation:

- a Physics Instruments piezoelectric sample stage;
- a Princeton X-ray CCD 1300x1340;
- an Andor IXON DV860 camera;
- a XGLab low-energy X-ray fluorescence detector.

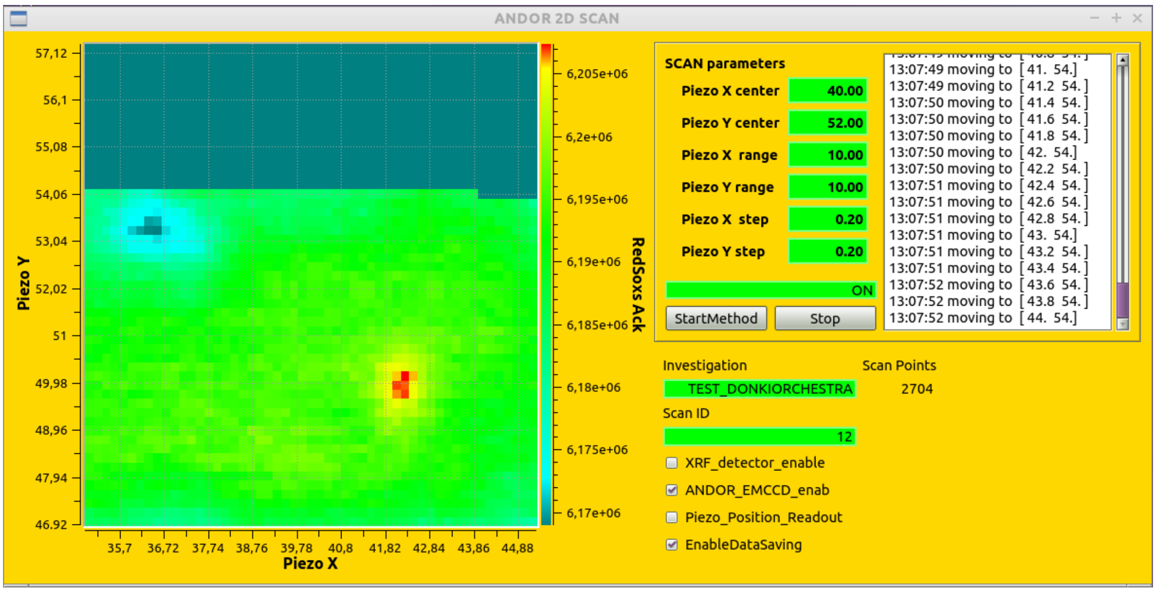

Figure 5. Two dimensional scan performed with DonkiOrchestra at Twinmic beamline.

As first experiment, DonkiOrchestra has been used to implement a two dimensional raster scan with concurrent acquisition of CCD images and XRF spectrums. For such application we have developed a dedicated graphical user interface (Figure 5) that uses the event-based infrastructure of DonkiOrchestra. Similarly to the Director, the GUI receives the ZeroMQ data messages from the Players. Each message contains experimental data (stage positions and image intensity) and a trigger index that is used to correlate the data and create the scan plot .

The results of the scan are stored in a HDF5 archive (Figure 6) that contains the datasets of stage positions, CCD images, XRF spectrums and some environment metadata. Data belonging to different datasets may be easily correlated by a post-processing analysis software in order to produce scientific results.

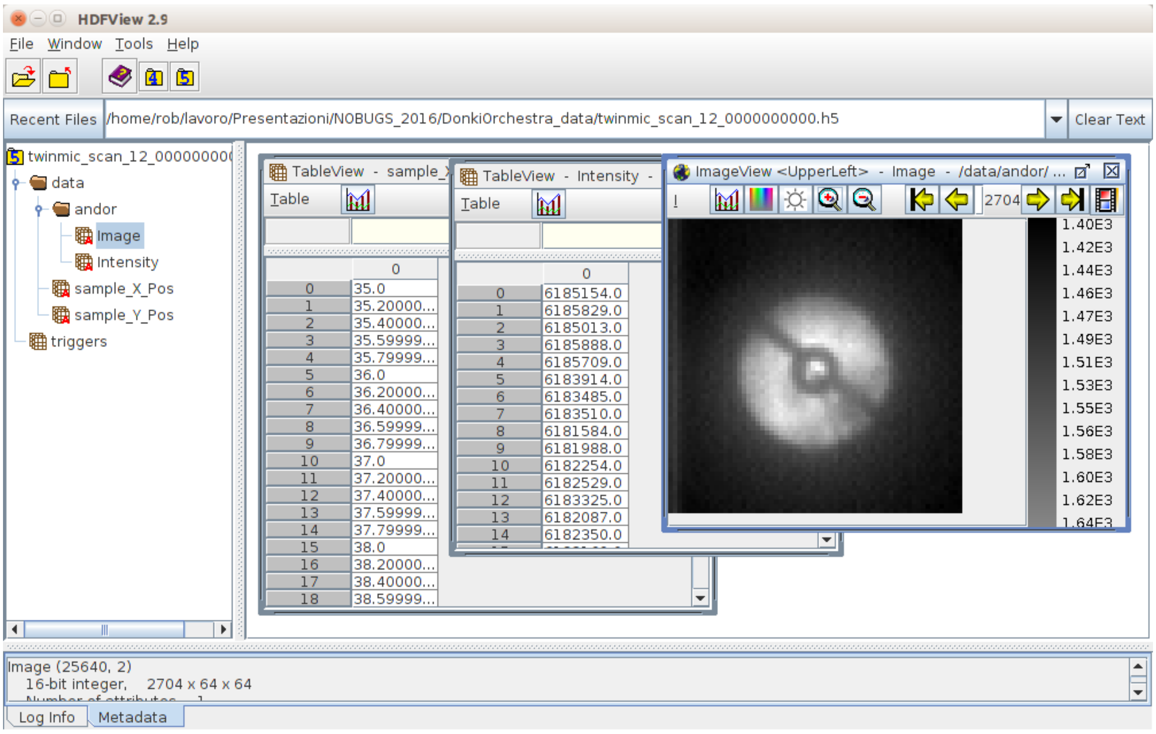

Figure 6. TwinMic HDF5 archive produced with DonkiOrchestra.

5. Conclusions

The present paper describes the software architecture behind DonkiOrchestra, a framework which supports both data acquisition and experiment control. The DonkiOrchestra approach is fully configurable and scalable, so it can be reused on different endstations and facilities. The strength of this software product resides in its design choices: a powerful messaging system like ZeroMQ that maximizes the opportunity of performing parallel operations and sensor readouts, a reliable distributed control system like TANGO, a dynamic and portable language like Python and a fully configurable structure that permits a high degree of customization.

Acknowledgement

The authors, members of the Scientific Computing team of Elettra, thank the IT Group and the Twinmic beamline staff of Elettra Sincrotrone Trieste. Moreover they acknowledge the importance of advanced technologies and open source software like that of the distributed control system TANGO and ZeroMQ messaging system.

[1] E Karantzoulis E, “Elettra 2.0 – The next machine”,Proceedings of IPAC2015, Richmond, VA, USA

[2] R Borghes, V Chenda, A Curri, G Kourousias, M Lonza, M Prica, and R Pugliese. “A Common Software Framework for FEL Data Acquisition and Experiment Management at FERMI.” Proceedings of ICALEPCS2013, San Francisco, CA, USA, 2013, 6–11

[3] www.tango-controls.org/

[4] http://zeromq.org/

[5] https://support.hdfgroup.org/HDF5/