

EPICS APPLICATIONS IN THE CONTROL OF SPES TARGET LABORATORY

M. Giacchini, A. Andrighetto, G. Bassato, N. Conforto, L. Giovannini,
INFN, Laboratori Nazionali di Legnaro, Legnaro (Padova), Italy

Abstract

The project of a new facility for the Selective Production of Exotic Species (SPES) has started at LNL. Radioactive ions will be produced by impinging an UCx target by a 70MeV, 200 μ A proton beam delivered by a commercial cyclotron. Then, the unstable ions will be accelerated by injecting them into the LNL superconducting LINAC. The construction of Target and Ion source prototype (Fig. 1) is at an advanced stage and, after more than two years spent in its construction, preliminary extraction tests were carried out with non-radioactive beams. The control of Target instrumentation is based on EPICS; we describe here the basic choices on hardware and software tools on both IOC and client side and give a brief description of last developments.

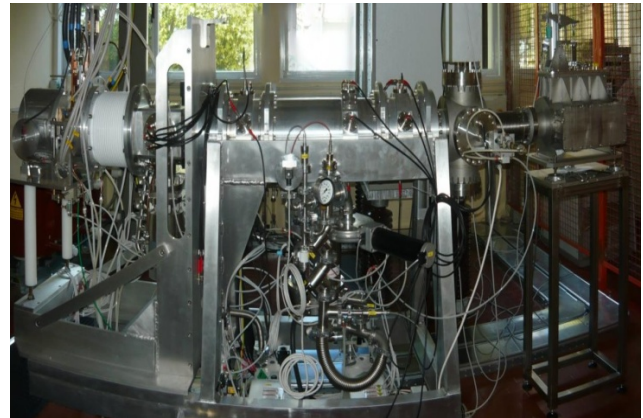


Figure 1: The target chamber and ion source

THE TARGET LABORATORY INSTRUMENTATION

The target instrumentation controls the beam extraction and transport up to a diagnostic station where the physical characteristics of the beam are measured. The beam production is obtained by heating the target to a temperature of about 2000 C, necessary for the optimal extraction of ionized fragments.

The power required for heating is delivered by an array of high current power supplies (LAMBDA GENESYS series) configured in a master-slave chain and providing a current in excess of 1300 A. Other heating methods are foreseen for the future (i.e. using a laser or a microwave source) but currently only the ohmic dissipation has been used. Once extracted, the beam is focused by an electrostatic lens of three quadrupoles fed by a set of HV bipolar power supplies (a special assembly of rack mount units manufactured by Ultravolt).

The target and the power supplies are placed on an insulated platform that is brought at about 60KV from ground by a FUG (HCP series) power supply. The electrical power required by GENESIS modules is transferred to the platform through a 20KW insulation transformer. An Ethernet transceiver from copper (100Base-T) to fiber optic is used to link the control network to the instrumentation placed over the HV platform; a multi-port Ethernet to serial converter (Control Device Master) is then used to connect the devices equipped with a serial interface.

Accelerator Controls

CONTROL DEVICES

The devices used to control the beam production and extraction are Linux-based IOCs. The LAMBDA-GENESYS master unit has a serial RS232 link to the host controller, which is a standard PC running on CentOS Linux. This OS distribution has been chosen because it is open-source, stable and completely compatible with RedHat. The device support is derived, with minimal modifications, from the driver developed at PSI, based on StreamDevice[1]. The HV power supplies (both Ultravolt and FUG) have an analog interface and are controlled by means of three microIOCs manufactured by Cosylab (SI).

These devices are embedded controllers based on a PC104 board and running under Debian Linux (preloaded on a flash disk). Each unit has three I/O boards, providing an adequate number of analog and digital I/O channels. EPICS drivers and debugging utilities come built-in with the controller software.

BEAM DIAGNOSTICS

A diagnostic station has been placed at the output of the electrostatic triplet to measure the beam current and profile. The beam current is measured by means of a faraday cup, while the profile is reconstructed by sampling the currents acquired by a set of horizontal and vertical grids. Stepper motors are used to insert/extract the devices along the beam line. The data acquisition system is implemented in a VME crate and runs under Vxworks.

Development and application frameworks

The controller is an Emerson (formerly Motorola) MVME3100, the ADC is a XYCom XVME566 board, while the stepper motor controllers are home made devices. A modified version of the diagnostic box has been realized to measure the beam emittance: the main difference in hardware setup consists in a sliding slot moving in front of the grid array and a linear encoder to acquire the slot position. Once the raw data have been acquired, they are transferred to a host computer where the beam emittance is calculated and displayed (Fig.2).

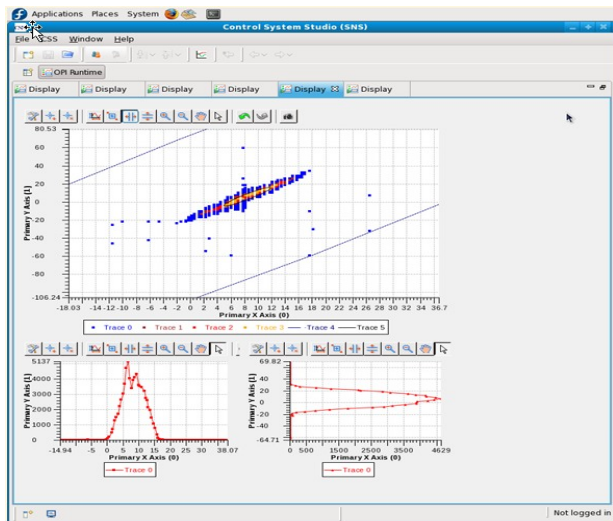


Figure 2: sample of emittance measure in a CSS screen

THE OPERATOR INTERFACE

A long term evaluation has been carried out to select the best tool to create graphic interfaces. First applications (the control of LAMBDA PS) were developed using MEDM that, despite its age, still remains an effective tool for fast prototyping and debugging. It was clear, however, it couldn't be the right choice for a project that will require maintenance over a period of twenty years at least. Then we decided to test the capabilities offered by LabView which provides a great graphic rendering and a big amount of customizable control widgets (Fig. 3).



Figure 3: Control screen of HV power supplies

There are two possible approaches in using LabView as EPICS client: the most traditional is based on the "shared memory" method developed at SNS[2]; the most recent, available since the release 8.6, makes use of NI native "network shared variable" technology. We tested both, with particular attention to the SNS solution. Figure 3 shows a LabView screen of the user interface realized for HV power supplies.

At the end, however, we decided to focus on CSS (Control System Studio)[3], that resulted, by far, the most innovative and promising tool for new developments. CSS is based on Eclipse, a customizable framework, that allows the developer to extend its functionality by adding new control plug-ins. Thanks to this feature, different CSS versions are available; we decided to adopt the SNS version that includes a rich set of graphic widgets (BOY), a new implementation of Alarm Handler and a new interface to Channel Archiver.

THE ARCHIVER

The Archiver is, in an EPICS system, the basic tool for archiving and retrieving the process variables. The Archiver can work using its embedded data base or in conjunction with an external RDB. At LNL we tested both configurations, using, as external data base, the freely distributed software MySQL. Due to the limited number of PVs currently in use, the performances of MySQL are more than acceptable. However, a new project was started, in collaboration with Brookhaven National Laboratory, to study the possibility of using the Archiver in connection with the non-relational data base HyperTable [4], which is based on a novel concept of file system and exhibits very promising performances in terms of speed: this project is HyperArchiver and will be shortly presented in the next paragraph.

The HyperArchiver project

The initial idea was triggered by the observation that the most famous and fast search engine in the world (Google) makes use of a proprietary distributed data base system (BigTable) that allows managing an enormous quantity of data with a surprising efficiency and speed. Most of algorithms used for data searching are property of Google and not published but the concepts underlying the data base structure are known and can be found in other commercial DBs. One of this products, HyperTable (by Zvents), is available either in a professional version and in an open source distribution under GNU public license. We decided to test this latter version and realize a connection to the Archive engine. We compared the store/retrieve speed with the configuration based on MySQL and it resulted that HyperTable is faster of about a factor of three in writing and more than a factor of ten in reading. A collaborative test was carried out at SNS to compare the HyperTable solution to their Archiver implementation, based on a connection with an Oracle server; also in this case it came

out that HyperTable is faster in both reading and writing, with a significant improvement in data retrieving.

We also designed the necessary plug-in to retrieve data and display them into CSS (Fig.4). The interface to CSS works but causes a significant slow-down in data reading that still has to be fixed. The collaboration on HyperTable continues and our goal is to reach a stable configuration to be used as the default Archiver installation for SPES project.

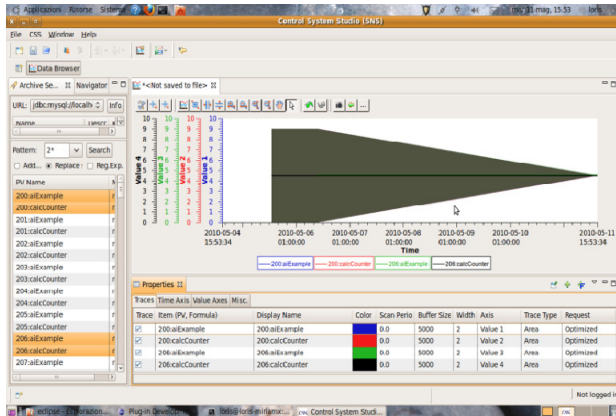


Figure 4: Snapshot of PV retrieval (simulated ramp of analog values) in CSS.

THE CONTROL NETWORK

Special care has been dedicated to the design of the control network. The following services were implemented:

- Gateway, to provide access to external services together with isolation from LNL network
- DHCP server, to manage IP addresses of control computers.
- Firewall, to protect the network from unauthorized accesses.
- Backup server, based on a Network Attached Storage (NAS) device, to allow full or incremental backup of control machines.
- Nagios [5] server, whose function is monitoring the operation of all installed IOCs and dispatching alarms in case of malfunction.
- CVS and Wiki servers: the CVS repository keeps trace of code versions and NamingConvention updating, while the Wiki server is very useful to maintain the documentation on team activity.
- PXE boot server for Vxworks and automatic reinstallation of operating system plus Epics development environment for Linux computers.

CONCLUSION

The control system of Target Laboratory has been a test bench for hardware and software technologies that will be used for SPES facility. Some technical options have been investigated enough to lead to strategic choices (i.e. using CSS for the development of user interface). Other key points must still be tested. A very important one is the integration of PLCs used for safety applications into the EPICS network. A solution based on dedicated drivers is possible for many families of PLCs, but we are strongly oriented to focus on the usage of an OPC server[6]. This approach has the considerable advantage of being independent from the PLC brand.

We also plan to continue the development of HyperArchiver, encouraged by the great interest shown for its possible application in large projects [7] where the capability of managing a huge amount of PV data in a fast way is of extreme relevance.

ACKNOWLEDGEMENTS

We sincerely thank Bob Dalesio and Robert Petkus

for supporting the HyperArchiver project and for the fruitful discussions during the stage of one of authors (M. Giacchini) at Brookhaven National Laboratory. We are also grateful to Michael Davidsaver and Daron Chabot (BNL) for their valuable work in customizing the XVME566 driver [8] and writing the device support for our stepping motor controllers.

REFERENCES

- [1] <http://epics.web.psi.ch/software/streamdevice/>
- [2] http://neutrons.ornl.gov/diagnostics/documents/epics/LabVI/SNS_LabVIEWEPICS.html
- [3] <http://ics-web.sns.ornl.gov/css/index.html>
- [4] <http://www.lnl.infn.it/~epics/Hypertable.pdf>
- [5] <http://www.lnl.infn.it/~epics/NAL.html>
- [6] <http://www-csr.bessy.de/control/SoftDist/OPCsupport>
- [7] <http://www-arch.iter.org/sites/epics2010/slides/>
- [8] <http://epics.hg.sourceforge.net/hgweb/epics/xycomioc>