Available online at www.sciencedirect.com
SCIENCE DIRECT
Fusion Engineering and Design 81 (2006) 1847–1851
Fusion Engineering and Design
www.elsevier.com/locate/fusengdes

Integrating EPICS and MDSplus

D. Mastrovito *, W. Davis, J. Dong, P. Roney, P. Sichta

Princeton Plasma Physics Laboratory, P.O. Box 451, Princeton, NJ 08543, USA

Abstract

The National Spherical Torus Experiment (NSTX) has been in operation at the Princeton Plasma Physics Laboratory (PPPL) since 1999. Since then, NSTX has made use of the Experimental Physics and Industrial Control System (EPICS) and MDSplus software packages, among others for control and data acquisition. To date, the two products have been integrated using special 'bridging' programs that include client-components for the EPICS and MDSplus servers. Recent improvements in the EPICS software have made it easier to develop a direct interface with MDSplus. This paper will describe the new EPICS extensions developed at PPPL that provide: (1) a direct data interface between EPICS process variables and MDSplus nodes; and (2) an interface between EPICS events and MDSplus events. These extensions have been developed for use with EPICS on Solaris and are currently being modified for use on real-time operating systems. Separately, an XML-RPC client was written to access EPICS 'trended' data, sampled usually once per minute during a 24 h period. The client extracts and writes a day's worth of trended data to a 'daily' MDSplus tree.

Keywords: Plasma control systems; EPICS; MDSplus; NSTX

1. Introduction

The National Spherical Torus Experiment (NSTX) is a low aspect ratio spherical torus ($R = 0.86$ m, $a \leq 0.67$ m, $I_p \leq 1.5$ MA, $B_T \leq 0.6$ T) capable of being operated in single-null, double-null, or center-stack limiter configurations [1]. Since its first plasma, achieved in February 1999, NSTX has made use of the Experimental Physics and Industrial Control System (EPICS) and Model Data System (MDSplus) software packages among others for control and data acquisition [2]. EPICS is used on NSTX for control of engineering systems such as the torus vacuum system, the water cooling system, and the coil power supplies, while MDSplus is used for data management and control of plasma diagnostics. A system diagram is shown in Fig. 1. The new software described in this paper will provide a more heterogeneous computing and control environment.

2. EPICS

EPICS [3] is an open-source collaboratively developed package designed for continuous process con-

* Corresponding author. Tel.: +1 609 243 3064; fax: +1 609 243 3086.
E-mail address: dmastrov@pppl.gov (D. Mastrovito).

0920-3796/$ – see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.fusengdes.2006.04.042

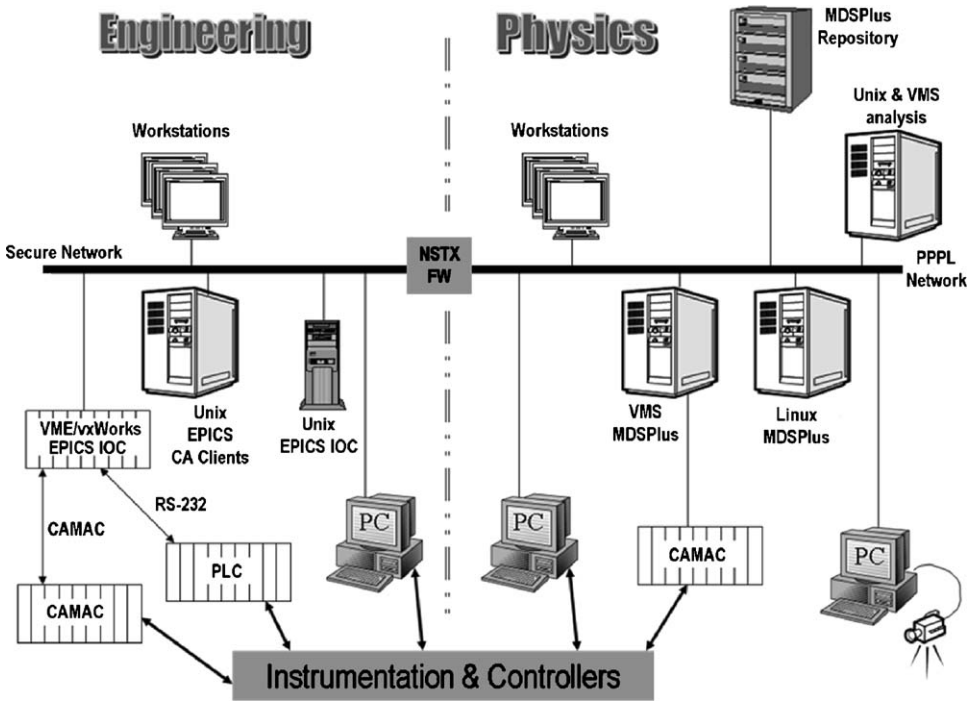

Fig. 1. NSTX data acquisition and control system.

trol applications, which is enhanced and maintained through its community of users. It consists of software components and tools that application developers can use to create a control system. The basic components are the Input/Output Controller (IOC), Operator Interface (OPI), and local area network (LAN).

2.1. EPICS IOC

The IOC is any platform that can support the EPICS run-time components, including the database, and its access routines, device drivers, record types for various input and output and scanning and monitoring functionality.

The EPICS database is the basis of an IOC. The database contains numerous instances of records; there are about 30 standard record types for processing different standard data types, controlling devices and generating events. The records/functions are linked together to perform the desired operation of the system. Some records interface with device support and driver support code. These additional layers are used to bring external information into the database, for example, from an I/O board on the computer's local bus, from a network device, from a PLC, or from a CAMAC module. In addition to the database engine, each IOC runs a Channel Access Server (CAS) to publish information about its records to one or more Channel Access Clients (CAC) using the Channel Access (CA) network protocol. An IOC is also a CAC, so IOCs can communicate for integrated control applications.

EPICS is highly scalable and typically has 100s of IOC and hundreds of thousands of records in the control system. The EPICS control system on NSTX is comprised of four IOCs, 8500 records and 1800 I/O points. In the past, EPICS IOC's have used the vxWorks real-time operating system from Wind River Systems. The most recent major release of EPICS software, version 3.14, has been written to be operating system independent. This has allowed so-called "soft IOCs" to be used with non real-time operating systems such as Windows, Mac OS, and Unix. By choosing an operating system for the IOC that supports an MDSplus client library, it is possible to develop

new EPICS record types that can access MDSplus directly.

2.2. EPICS archiver

The Channel Access protocol is used by all CAC utilities, such as the channel archiver. The channel archiver is an archiving toolset for EPICS. It can take samples of any value that is available via channel access and it stores them for later retrieval in a binary file format. The values to be archived and their sampling frequency are configurable via an Extensible Markup Language (XML) file. Once archived these values and their associated time stamp information are available via a Java archive viewing client, called ArchiveViewer. The values are also made available from a channel archiver web server via XML-RPC (remote procedure call) which uses HTTP as the transport protocol and XML as the encoding language allowing complex data structures to be transmitted on disparate operating systems over the internet. The channel archiver is used on NSTX to trend various machine parameters over time, such as vessel temperatures measured from thermocouples. The flexibility of the XML-RPC architecture makes storing EPICS archived data to MDSplus possible.

3. MDSplus

NSTX data management is based on MDSplus [4], which is a set of software tools that manage a hierarchically organized data repository that is created for each NSTX shot. Both raw and analyzed data from physics diagnostics are written into the 'shot tree'. In addition to data storage, MDSplus provides device control capability and an event system. For example, an MDSplus event can be used to refresh a plot with new data, or to control a CAMAC-interfaced instrument. NSTX currently uses four MDSplus servers, three running on VMS and, one on Linux. MDSplus clients run under VMS, Unix, Windows, and Mac OS X. MDSplus provides functionality for access to its data from standard tools such as data access routines available for LabView, Java, Interactive Data Language (IDL), C, Fortran, Perl, Python, Matlab, as well as a native plotting tool called scope. In addition several web tools have been created for use on NSTX which allow plotting and overlaying of MDSplus signals [5]. Currently, MDSplus stores 140 MB of data per NSTX shot compressed, comprising more than 5500 waveforms and 25,000 parameters [6].

4. Motivation for integration

NSTX has been operating and adding systems for about six years. The need for operating more systems in an integrated fashion has increased; often systems on the engineering side need results computed on the physics side, and physics systems need information from the engineering side, such as the position of a Torus Interface Valve. In addition, it was a requirement that certain values from the engineering subsystems be recorded for each shot. The exchange of information between physics and engineering computing includes both data and events for program synchronization. Until now this has been done using 'C' programs, Unix-scripts, and an IPCS event system [7] running on a Solaris workstation. The 'C' programs used the EPICS CAC library to retrieve information from IOC records, and the MDSplus client library to write values to the nodes. Additionally, setting up devices on EPICS required use of an EPICS utility program. With the advent of EPICS release 3.14, these bridging programs can be eliminated and the MDSplus access made directly from an IOC. In addition, the EPICS event system and the MDSplus event system can now be linked.

Having MDSplus values available from within an IOC makes it possible to read CAMAC or other device setup parameters from an MDSplus shot tree. Values read by EPICS from digitizers or other devices controlled by EPICS can then be directly placed into MDSplus trees without need of a separate client application. Various engineering control parameters can be stored in MDSplus for use in analysis routines. EPICS processing did not previously have any "knowledge" of MDSplus software events that mark various stages of the NSTX shot cycle. Having access to MDSplus events from EPICS will allow monitoring of EPICS values and processing of EPICS device records to be done in conjunction with MDSplus software events. Traditional programs for viewing MDSplus data or for doing between shot analyses that wait for MDSplus events, can now also receive events from the EPICS event system.

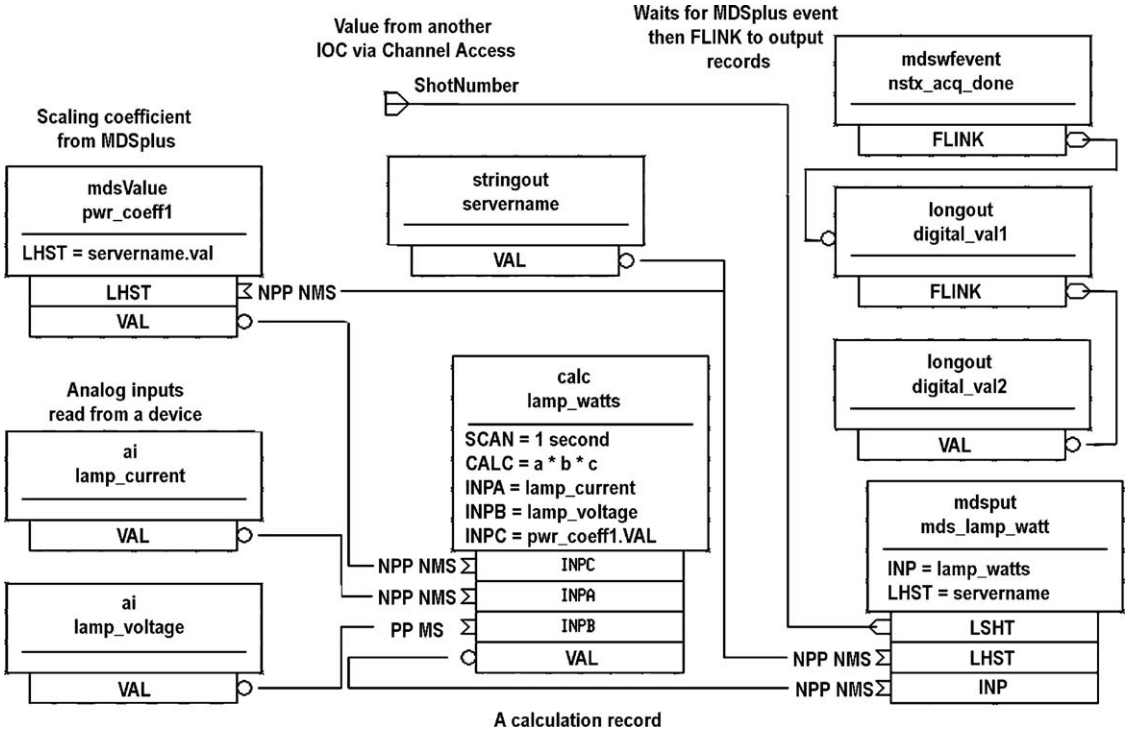

Fig. 2. IOC database record chain, with mdsvalue, mdsput, and mdswfevent records.

5. Interfacing EPICS and MDSplus

5.1. Data interface

New EPICS record types have been developed to facilitate direct communication between EPICS and MDSplus. The "mdsvalue" record reads an MDSplus node, bringing its value into the IOC. The "mdsput" record does the opposite, taking a value from an IOC record and writing it to the MDSplus node. Fig. 2 shows a screen shot of an EPICS tool that allows graphical editing of database records where linking records of different types together defines the functionality of your control system. Included are several examples of record instances where the mds records are used. Both the mdsvalue and mdsput records support multiple data types including, short, unsigned short, long, float, double, char, and unsigned char. The data can be a scalar or an array (1D). The records have several fields for specifying their operation: EPICS native data type, MDSplus host (server:port), tree name, and tag or path to the node, or valid TDI expression. These fields use the EPICS 'IN-LINK' method which means that the specified values can be defined statically in the database definition file, and also at runtime. The fields can be linked to read from other EPICS records or the fields can have their values 'put' there by other records, or a CAC.

5.2. Event interface

In addition to exchanging data between the EPICS and MDSplus systems, it is useful to integrate their event systems. MDSplus events are broadcast by a server to any number of 'listening' event clients. The MDSplus events are identified as a character string. The EPICS event system is local to each IOC. The events are numerically identified in the range 1–255. IOC's can 'know' of events on other IOC's through CA. Several record types can post an event, and most records can be triggered to process in response to an event.

Two record types have been developed to perform this event integration, "mdssetevent" and the "mdswfevent". When the mdswfevent record is pro-

cessed, it will wait for the specified MDSplus event to occur. When the event is received the record's processing will complete and optionally post an EPICS event and process out-links (causes other EPICS records to process). The mdssetevent record is used to generate an MDSplus event. Both records have special fields for specifying the MDSplus host and the MDSplus event name. These special fields use the EPICS 'IN-LINK' method, with the capabilities as described in the previous section.

5.3. Trended data

Trended data is data compiled by the EPICS archiver over an extended time, rather than accumulated in bursts during each plasma pulse. This data is stored with timestamp information and is therefore easily separated into daily data. On NSTX we have created a special data tree for daily "trended" data. This allows us to track how values change over the course of a day and to store this information in a familiar format that is easily accessible. Since both the EPICS archiver web server and MDSplus support access via the Java programming language, an XML-RPC client was written in Java which communicates with the archiver web server, retrieves trended data for channels listed in a local configuration file and puts them in the desired MDSplus nodes in the "daily" tree.

6. Future work

The initial round of code development was to demonstrate the basic functionality of the proposed records. However, the processing time of these new records is currently too long. To use these new records in a typical IOC environment, where soft real-time operation is required, additional work is necessary. This will most likely be implemented using an EPICS device layer which will add asynchronous processing to new records. The initial implementation was done on a Unix/Solaris IOC. Additional work is required to build the records for use with IOC's running on other EPICS supported operating systems.

7. Summary

As the number of operating systems, IOCs, and other control components on NSTX increase, there is a greater need for integration between these systems. The most recent major release of EPICS software, version 3.14, has been written to be operating system independent, allowing IOC's to be built with non-real-time operating systems. By choosing an operating system for the IOC that supports an MDSplus client library, it is possible to develop new EPICS record types that can access MDSplus directly. Having the capability to communicate with MDSplus from within an EPICS IOC makes reading device setup parameters from MDSplus and writing of data from devices into MDSplus very convenient, thus making EPICS data readily available for viewing from more standard applications. Integrating the EPICS and MDSplus event systems allows synchronization of processing via software events that occur in either of the two systems. These new records have been implemented for use on Solaris, however additional work is required to build the records for use on real-time operating systems.

Acknowledgement

This work was supported by DOE Contract DE-AC02-76CH03073.

References

[1] S. Kaye, et al., The physics design of the national spherical torus experiment, Fusion Technol. 36 (1999) 16, http://nstx.pppl.gov/index.shtml.
[2] P. Sichta, J. Dong, Status of the experimental physics and industrial control system at NSTX, in: 19th IEEE/NPSS Symposium on Fusion Engineering, January 22–25, 2002, pp. 245–247.
[3] R. Lange, EPICS: recent developments and future perspectives, in: Ninth International Conference on Accelerator and Large Experimental Physics Control Systems, October 13–17, 2003.
[4] J.A. Stillerman, T.W. Fredian, K.A. Klare, G. Manduchi, MDSplus data acquisition system, Rev. Sci. Instrum. 68 (1) (1997) 939, http://www.MDSplus.org.
[5] W. Davis, P. Roney, T. Carroll, T. Gibney, D. Mastrovito, Web interfaces to MDSplus data, in: The Proceedings of the 19th IEEE/NPSS Symposium on Fusion Engineering, January 22–25, 2002, pp. 176–197.
[6] W. Davis, P. Roney, T. Carroll, T. Gibney, D. Mastrovito, The use of MDSplus on NSTX at PPPL, Fusion Eng. Des. 60 (3) (2002) 261.
[7] G.G. Preckshot, D.N. Butner, A simple and efficient interprocess communication system for actually using a laboratory computer network, IEEE Trans. Nucl. Sci. 34 (1987) 858.