GA–A25809

WORLWIDE COLLABORATIVE EFFORTS IN PLASMA CONTROL SOFTWARE DEVELOPMENT

by
B.G. PENAFLOR, J.R. FERRON, M.L. WALKER, D.A. HUMPHREYS,
J.A. LEUER, D.A. PIGLOWSKI, R.D. JOHNSON, B.J. XIAO,
S.H. HAHN, D.A. GATES

MAY 2007

GENERAL ATOMICS

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

GA–A25809

WORLWIDE COLLABORATIVE EFFORTS IN PLASMA CONTROL SOFTWARE DEVELOPMENT

by
B.G. PENAFLOR, J.R. FERRON, M.L. WALKER, D.A. HUMPHREYS,
J.A. LEUER, D.A. PIGLOWSKI, R.D. JOHNSON, B.J. XIAO,*
S.H. HAHN,† D.A. GATES‡

This is a preprint of a paper to be presented at the 6th IAEA Technical Meeting on Control, Data Acquisition, and Remote Participation for Fusion Research, Inuyama, Japan, June 4–8, 2007 and to be published in the *Proceedings.*

*FAR-TECH, Inc., San Diego, California, USA
†National Fusion Research Center, Dae Jeon, Korea
‡Princeton Plasma Physics Laboratory, Princeton, New Jersey

Work supported by
the U.S. Department of Energy
under DE-FC02-04ER54698, DE-FG03-99ER82791
and DE-AC02-76CH03073

GENERAL ATOMICS PROJECT 30200
MAY 2007

GENERAL ATOMICS

ABSTRACT

This presentation will describe the DIII-D collaborations with various tokamak experiments throughout the world which have adapted custom versions of the DIII-D Plasma Control System (PCS) software for their own use. Originally developed by General Atomics for use on the DIII-D tokamak, the PCS has been successfully installed and used for the NSTX experiment in Princeton, the MAST experiment in Culham UK, the EAST experiment in China, and the Pegasus experiment in the University of Wisconsin. In addition to these sites, a version of the PCS is currently being developed for use by the KSTAR tokamak in Korea. A well-defined and robust PCS software infrastructure has been developed to provide a common foundation for implementing the real-time data acquisition and feedback control codes. The PCS infrastructure provides a flexible framework that has allowed the PCS to be easily adapted to fulfill the unique needs of each site. The software has also demonstrated great flexibility in allowing for different computing, data acquisition and real-time networking hardware to be used. A description of the current PCS software architecture will be given along with experiences in developing and supporting the various PCS installations throughout the world.

1. INTRODUCTION

The Plasma Control System (PCS) software has evolved over the years from being a DIII-D-specific tool for specifying plasma shape and position parameters [1] to a much broader and generalized application that has been successfully adapted for use on a number of tokamaks throughout the world. Originally developed to run on i860 VME-based computers, it has since been demonstrated to work on 64 bit Alpha processors and numerous types of Intel-based single and multi-processor PCI form computing systems [2].

The growth of the PCS can be greatly attributed to the many successful and highly productive collaborations between DIII-D and other fusion experiments sharing in the PCS software. A number of tools have been found to be extremely beneficial for many long distance collaborative efforts including the use of video meetings, remote computer logins, terminal sharing utilities in which parties from both sites can see and work in the same screen and the availability of Voice Over IP applications such as Skype [3]. While global networking has made it possible to do much of the software work from afar, the advantages of interacting face to face and making on-site visits has been found to still serve a vital role in productive collaborations.

These collaborations have helped to improve the PCS software in general as numerous ideas and recommendations from the various PCS sites have been received and implemented into the codes and passed along to other users to share.

2. KEY FEATURES OF THE PLASMA CONTROL SYSTEM SOFTWARE

A number of key features promoted the acceptance of the GA-developed PCS software for use on other tokamak experiments. Among the highlights of the system are:

- A modular structure allowing easy addition and modification of control algorithms. This is particularly important in a research environment where the requirements for the PCS change as the research evolves.
- Extensive facilities for diagnosing the control system behavior.
- A graphical user interface for the tokamak operator. Multiple operators can interact with the control system simultaneously, each with his own user interface.
- Coordination of multiple computers for real-time control.
- The ability to handle a mixture of computer architectures (e.g., 32- and 64-bit processors).
- Modular construction of the PCS components with a client/server architecture.
- A structured algorithm programming environment that provides tools and standard methods to minimize control algorithm programming work.
- Automatic distribution of the PCS setup data to multiple real-time processors.
- Archiving of the discharge setup and data acquired during the discharge into the tokamak database.
- Portability of the infrastructure software to most computing platforms.
- Minimized software costs due to reliance on free, open source software for the OS for real-time computers. GA distributes a special customized version of the Linux Operating system providing low latency real-time responsiveness in the control system.
- The ability to incorporate various types of data acquisition, input/output hardware and real-time interconnect hardware into the system.
- Built in facilities for taking advantage of multi-core and multi-cpu computer systems to parallelize real-time processes and tasks.
- Common software routines for implementing real-time equilibrium reconstruction and isoflux control.
- Facilities for performing modeling and simulation using the same software developed to run on the tokamak itself. This capability provides a means of fully testing changes to the software before they are used on live experiments.

3. OVERVIEW OF THE PCS SOFTWARE ARCHITECTURE

The PCS software consists of an infrastructure set of codes that are identical for every site, and installation-specific codes that contain the device-specific and site-specific details for customizing the application. The infrastructure codes are maintained by General Atomics (GA) staff and include a comprehensive set of library routines that are used to build the various user interface, server and real-time processes. A simple demo version of the PCS has been made available to highlight fundamental features and assist in the development of custom versions of the PCS. In addition to this, GA provides a set of common installation source files containing prototype codes for incorporating the real-time equilibrium reconstruction (RT-EFIT) routines and various installation level codes for utilizing certain types of digitizer hardware.

The current PCS software has been designed to run primarily on Linux/Unix-based computing systems. The languages used include C, which makes up the bulk of the software, and IDL, which is used primarily to implement the user interfaces. Source code control for GA-maintained software is accomplished using SCCS. While GA takes responsibility for managing and maintaining the infrastructure and common installations files, all of the site specific installation codes developed for a particular tokamak experiment (with the exception of the DIII-D Tokamak) are maintained outside of GA. Each remote site is free to customize and make changes to their own versions of the installation specific software and employ whatever source code controls they deem appropriate.

PCS components, which are common to each installation, include the user interfaces, the server processes consisting of a waveform server, lock server, and message server, and a set of one or more real-time control processes. What primarily distinguishes one installation of the PCS from another are the chosen computing hardware and the custom controls implemented at each site. The task of creating a customized version of the PCS for a specific tokamak experiment involves identifying the computing hardware and implementing the specific real-time I/O interfaces, real-time networking schemes and the categories and control algorithms required for the specific device. The PCS has been designed with this flexibility in mind to allow developers at each location to be able to use whatever hardware they prefer for real-time data acquisition, outputs to actuators, and real-time computer interprocess communication.

4. INSTALLATIONS THROUGHOUT THE WORLD

The PCS software is currently being used at six different tokamak experiments throughout the world. These include the original implementation for the DIII-D Tokamak, the National Spherical Tokamak Experiment (NSTX) [4] at the Princeton Plasma Physics Laboratory (PPPL) in Princeton New Jersey, the MAST spherical tokamak [5] at Culham in the UK, the EAST tokamak [6] in Hefei China, the KSTAR tokamak [7] in Daejeon, South Korea and the Pegasus spherical tokamak [8] at the University of Wisconsin.

The DIII-D PCS now includes 19 real-time cpus working in parallel to control virtually every aspect of the plasma discharge. It has been operational for the past two decades and has undergone numerous upgrades, which have further extended its capabilities. Among the most significant of these improvements have been the addition of real-time acquisition and analysis of the Charge Exchange Recombination (CER) diagnostic, which has been successfully, used in a number of recent plasma rotation feedback control experiments. Also the ability of the PCS to display virtually any result graphically in real-time (Fig. 1) including information from the RT-EFIT codes has been a highly useful feature to DIII-D researchers. The computing hardware (Fig. 2) is comprised of a mix of Intel single and dual processor equipped PCI form rack mounted computers that communicate using a Myrinet [9] real-time network. Data acquisition is accomplished using D-TACQ PCI and cPCI-based digitizers [10]. Outputs to DIII-D actuators are through VME-based DATEL D/A converters, which are accessed through a real-time VME Intel computer on the Myrinet network.

The NSTX experiment was the first outside of DIII-D to adapt the PCS for use on their device. Developed to run on VME Skybolt computers using Front Panel Data Port (FPDP) digitizers, the NSTX PCS has been in place at PPPL since 1999. A major upgrade to replace the computing hardware with newer and significantly faster multi-processor multi-core Intel-based PCI form computers is close to completion.

The MAST version of the PCS has been in place for approximately four years. The hardware configuration is in many ways similar to the original NSTX implementation utilizing VME-based cpus and FPDP digitizers. Personnel at MAST have been able to incorporate their own configuration tools for defining site-specific application parameters to ease their PCS software development.

The Pegasus version of the PCS developed for the University of Wisconsin demonstrates the scalability and applicability of the software for use on smaller experiments. The Pegasus PCS adapted a year ago utilizes a single Intel Xeon-based, rack-mounted computer to run the real-time processes connected to a cPCI crate containing a single D-TACQ ACQ196 digitizer which provides the I/O interface of up to 96 ADC inputs and 16 DAC outputs.

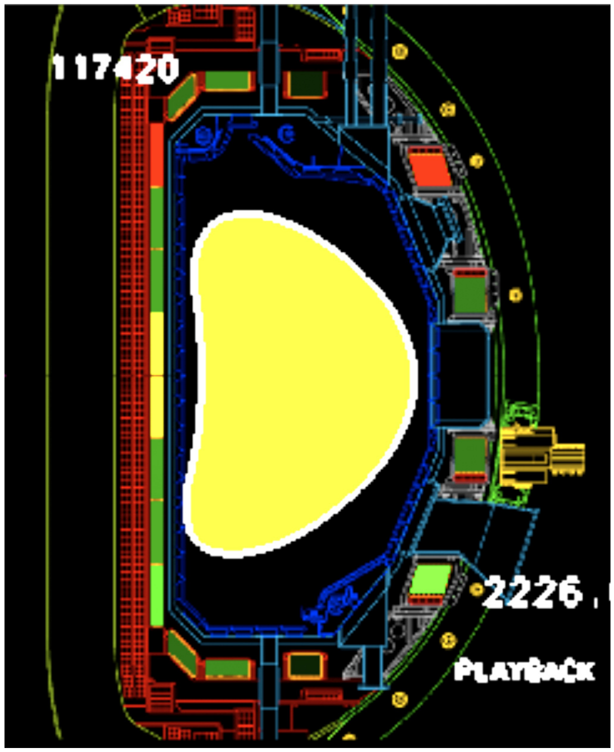

Fig. 1. DIII-D PCS real-time boundary display.

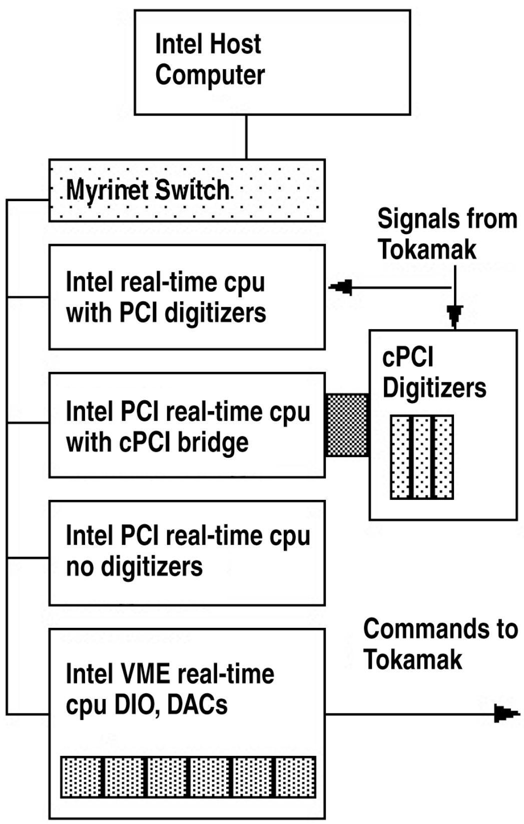

Fig. 2. DIII-D PCS hardware components.

The EAST PCS implementation was the first to be successfully adapted for use on a Superconducting Tokamak (Fig. 3). The system is comprised of three Intel Xeon dual cpu rack-mounted computers connected using Myrinet (Fig. 4). A pair of D-TACQ ACQ196 cPCI digitizers provide the I/O interface with the capability of acquiring up to 192 analog signals and sending 32 analog output commands. In addition to this, the EAST PCS makes use of the DIO capabilities provided by the ACQ196.

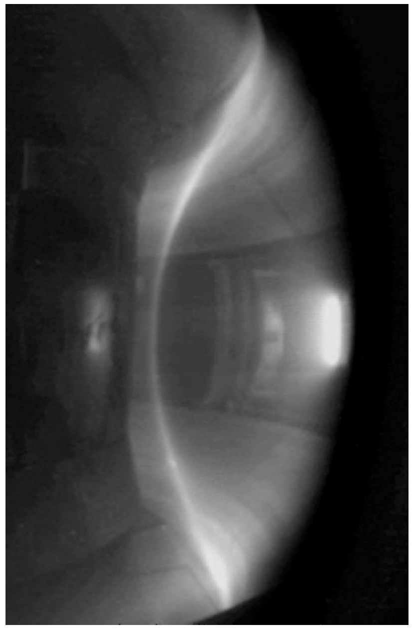

Fig. 3. EAST first plasma using PCS.

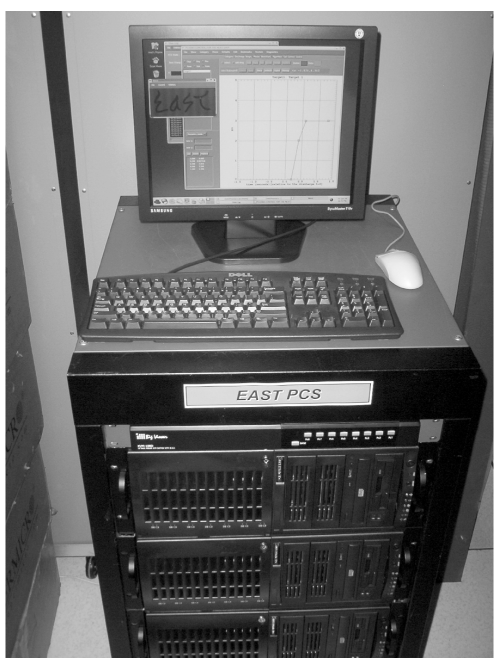

Fig. 4. EAST PCS hardware.

The KSTAR version of the PCS has been instrumental in furthering the capabilities of the PCS software. The KSTAR PCS, which is currently under development at NFRC, is scheduled to be used for coil current feedback control in August 2007. The hardware consists of three Intel Xeon dual cpu computers connected together in a Reflective Memory network (as opposed to Myrinet), which provides access to various external Tokamak sub-systems for obtaining acquired coil current information and controlling KSTAR power supplies. Recent improvements to the GA infrastructure codes to provide better facilities for handling shared memory between real-time processors have come as a direct result of the specific needs stemming from the KSTAR PCS implementation. The most recent version of the KSTAR PCS installed and tested at NFRC also demonstrates the ease at which the software can be adapted and used for longer pulse durations. While the DIII-D version of the PCS is typically used to control pulses lasting up to 10 s, hardware tests using the KSTAR version of the PCS were successfully run for pulse lengths exceeding 90 s.

5. COLLABORATIVE EXPERIENCES

The distribution of effort and responsibilities in PCS collaborations between DIII-D and the different experiments have varied. For the MAST and Pegasus adaptations of the PCS, involvement of DIII-D personnel were minimized to primarily providing access to the source code, supplying documentation and answering questions as they arose. In these instances the local on-site staff accomplished the bulk of the work for installing the PCS. On the other hand, for the first installations of the NSTX, EAST and KSTAR versions of the PCS, DIII-D personnel were actively involved to a very high degree in the software integration and set up of these systems. This involvement included remote collaborations between sites in addition to on-site visits by DIII-D staff to assist in the custom software development, installation and testing at each of these sites. For the EAST PCS, DIII-D staff were charged with delivery of a complete turnkey system and given primary responsibility over selection and assembly of all the required hardware and software.

A number of software tools were found to be extremely beneficial and in many cases essential to successful remote collaborative efforts and software support for abroad. With the PCS software running primarily on Linux/Unix-based computing systems comes access to powerful pre-installed tools including X windows, ssh, sftp and scp. These tools provide the capabilities needed for remote user interface display, logins and file transfers. Using the remote ssh login and X display capabilities, software development with the PCS user interfaces running from a remote machine located halfway around the world and appearing on the local displays became possible. In addition to this, the ability to invoke a local copy of the PCS user interface client on a GA computer and have it communicate directly with PCS server processes running at the remote sites was demonstrated with even greater speed and performance than when starting the interfaces from the remote computers.

Other tools including “talk”, a simple UNIX terminal chat tool, were very helpful particularly in cases where language differences presented communication challenges. Typed communications using “talk” were often easier for both sides to follow and understand than conversations over the phone. In addition to this another UNIX-based tool called “kibitz,” which provides a means for two remote users to share control of a single terminal session, proved to be very helpful in reviewing and debugging source code collaboratively and explaining the PCS software. The remote collaborative experience also benefited from the maturity of Voice/Video Over IP tools such as Skype. Multi-participant video conferences between GA staff and remote sites [11] using systems such as H.323 [12] have also proved essential in these collaborations.

From a technical standpoint, the use of the aforementioned tools demonstrated the feasibility of performing most if not all of the software development and support from GA. However, certain practical considerations including network security and time zone differences introduced a few hindrances and limitations to what could be fully accomplished by the local GA support staff. Tight network security often meant that the PCS computer systems located at the remote sites could not be directly accessed from computer systems local to GA. Support for these remote installations required first satisfying the unique security requirements of each site. This tended to involve multiple steps to be followed including the use of RSA keys and logging into one or more portal computers before access to the remote PCS computers could be obtained. In addition to this, the inability to access the PCS computers directly prevented GA staff from taking advantage of the significant performance advantages which were demonstrated by running the PCS user interface clients locally and having them connect directly to the remote PCS servers. Time zone differences also limited the window of opportunity for personnel from both ends to be involved in live correspondences which are essential to coordinating use of computer systems and performing PCS testing on the remote hardware.

6. CONCLUSION

The PCS software has provided excellent opportunities for worldwide collaborations in the field of fusion research. A flexible and well-designed software architecture supplied the basis for numerous successful adaptations of the PCS utilizing various types of computing, real-time I/O and interconnect hardware. The collaboration opportunities provided through the PCS have helped to improve the software for all users and have also enhanced communications and the sharing of ideas among those participating in PCS development.

REFERENCES

[1] J.R. Ferron, A.G. Kellman, G.R. McKee, et al., An advanced plasma control system for the DIII–D tokamak, Proc. 14th IEEE/NPSS Symp. on Fusion Engineering, San Diego, 1991 (Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey, 1992) p. 761.

[2] B.G. Penaflor, et al., Current status of the upgraded DIII-D real-time digital plasma control system, Fusion Engin. Design 71 (2004) 47.

[3] www.skype.com

[4] D.M. Mastrovito, Plasma control system upgrade for NSTX, P1-12, this meeting.

[5] G.J. McCardle, Adaptation of the MAST passive current simulation model for real-time plasma control, this meeting.

[6] B.J. Xiao, EAST plasma control system, 01-3, this meeting.

[7] S.H. Seo, Performance test of the reflective memory based plasma control system, P1-41, this meeting.

[8] M.W. Bongard, Development of an active plasma control system for Pegasus, Bull. Am. Phys. Soc. 50 (2005) 329.

[9] www.myri.com

[10] www.d-tacq.com

[11] G.C. Abla, A remote control room at DIII-D, this meeting.

[12] www.packetizer.com/voip/h323

ACKNOWLEDGMENT

This work supported by the U.S. Department of Energy under DE-FC02-04ER54698, DE-FG03-99ER82791 and DE-AC02-76CH03073.