Experiment Management System for the ASDEX Upgrade Tokamak

G. Raupp[1], G.Neu[1], R. Cole[2], K. Lueddecke[2], W. Treutterer[1], T. Vijverberg[2], D. Zasche[1], T. Zehetbauer[1], and the ASDEX Upgrade Team

[1]*Planck-Institut für Plasmaphysik, Euratom Association, Garching, Germany*
[2]*Unlimited Computer Systems, Iffeldorf, Germany*

Abstract

To run pulsed plasma discharges in a fusion experiment is a complex operational task which requires scientists' and operators' intervention, combined with proper sequencing of actions throughout distributed inhomogeneous systems

For the ASDEX Upgrade's (AUG) new tokamak control system, the experiment management system, XM, is the central entity for automated operation. It synchronizes activities between various systems, and provides central services for configuration, operation, logging, protocolling, and archiving.Multiple, activity specific user interfaces are provided to support the work of the people responsible for operation.

The paper describes the architecture of XM and shows how its services interact with and enable experiment operation.

Introduction

Operating a large fusion device like AUG requires the coordinated action of many people and systems.

To prepare a pulse, the physicists define the discharge programme (DP) containing parameters and waveforms for intended discharge scenarios. Actuator systems are configured by machine control system engineers to match the implicit and explicit requirements of the DP. Diagnostic sensors to provide physical and technical quantities required for plasma control are set up and calibrated accordingly. Once the configuration of sensor and actuator systems is completed, the corresponding systems lock, signal their readiness, and their configuration data, together with the DP, is made available to the discharge control system (DCS). During the pulse, the DCS performs the hard real-time tasks of feedback controlling and monitoring plasma and technical quantities. It uses the timing system to issue synchronization events to diagnostics and actuator systems and has access to the protection systems. State information is displayed to operators and physicists, and DCS process activities logged.After the pulse, protocol data recorded by the DCS, configuration parameters, and log messages are archived for post-discharge analysis by system engineers and physicists.

This sequence of events follows a well-defined schedule readily lending itself to automation in a superior platform: the Experiment Management System.

AUG Operation Environment

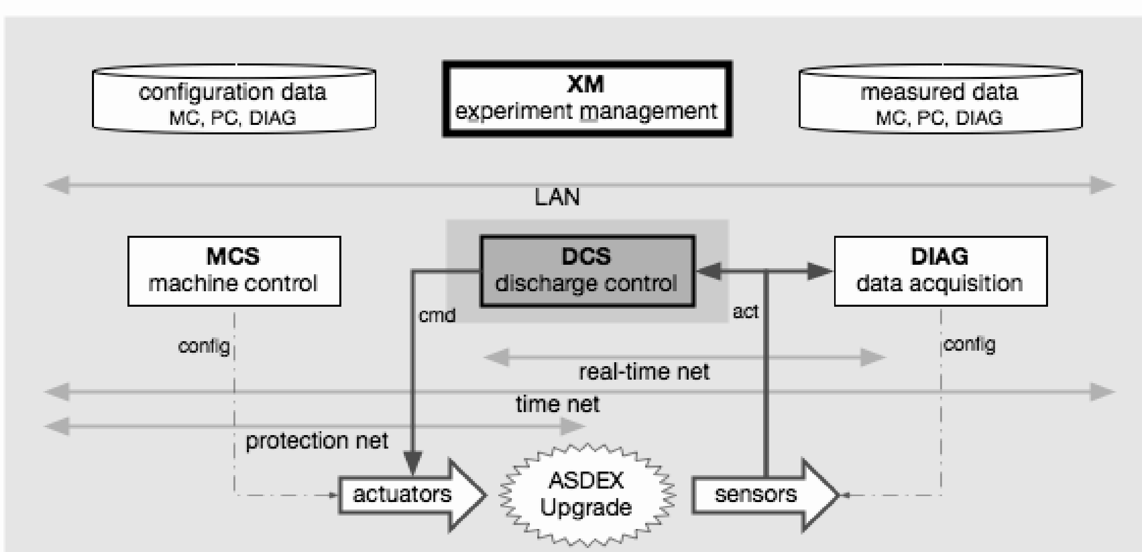

Figure 1: Overview of ASDEX Upgrade's Operation environment.

Systems and Configuration Data

AUG is equipped with a fully digital DCS, consisting of a cluster of VxWorks based PCI low-cost controllers interconnected via a real-time network (RTN) currently implemented as distributed shared memory [1]. A key feature of AUG's DCS is its high configurability: At boot time, allocation of application processes (AP) implementing control tasks to the controllers is read from an XML document: the System Release (SR) file. Further XML documents, accessed via HTTP and parsed at run-time, define the mapping of AP internal quantities to parameters and consumed or produced signals in the system-wide CODAC namespace [1]. A dedicated AP, the reference value injector providing the DCSs real-time reference signals, is configured with the DP, which is also an XML document. Once all APs are configured, the RTN map for all exchanged real-time signals is computed and made available for transparent signal exchange.

The autonomous machine control system (MCS), continuously operating the actuator systems and configuring them for the DCS, is implemented using SIMATIC programmable logic controllers (PLC) running a proprietary communication protocol. One of the PLCs, the SIMATIC Leitsteuerung (SLS) is equipped with a communication processor translating the proprietary protocol to TCP/IP for external access. All system settings of the MCS are reflected on the SLS. The configuration of the actuator systems is done either remotely, by selecting recipes in the SLS, and distributing them to the PLCs, or locally on a PLC console by the system engineer. The SLS also holds state information of all PLCs, e.g. indicating the readiness of the system for configuration or operation.

A subset of the available diagnostic sensors is required by the input processes of the DCS. The sensor characteristics are defined in the headers of diagnostics' proprietary description file, the shotfile (SFH). Other diagnostics, the real-time diagnostics offer processed sensor data directly as real-time signals. Data acquisition is pulse oriented, the pulse number being distributed by the DCS, and triggered by timer events from the DCS.

Low-level protection systems provide protection for personnel and equipment. An autonomously operated time system provides unique experiment time as a measurable quantity [2].

Operation Modes

AUG has various operation modes which are reflected in the MCS and DCS. The standard operation mode, the so called shot mode, requires access to actuators and sensors. In test mode, actuators will not accept set values from the DCS. The DCS may either use input from sensors or, in the replay submode, simulate input by using data from a previously recorded shot. Replay data is additional configuration information for the input processes.

Experiment Management

The operation environment described above is sufficiently complex and depends on so many interactions and interconnections that human operators should be aided in running the experiment, both to minimize the risk of costly errors, and to increase the efficiency of experiment operations. Therefore, an automation and management layer was introduced to coordinate as many tasks as possible.

Tasks of the Experiment Management System

The experiment cycle of ASDEX Upgrade has five major phases, controlled by XM, involving the activities of various users synchronization, and data exchange between systems.

Boot phase:

In this phase a control system engineer defines a system release (real-time code, allocation to DCS controllers, priorities) to be downloaded to the controllers of the DCS by selecting and possibly editing a single system release file.

Preparation

The preparation phase defines the requirements for configuration: the experiment leader selects and possibly edits the DP using a dedicated discharge programme editor and selects the actuators and diagnostics required. According to the requirements of the experiment leaders, the control system engineer defines the APs connectivity (real-time signals) and parameter references.

Physicists and engineers may also specify signals for online visualization during the pulse.

Configuration

In this phase, the experiment operator instructs all systems to prepare themselves for the pulse: According to the requirements of the DP, the MCS operator selects the recipes and instructs MCS engineers to configure their systems [3]. The current state of the MCS is continuously displayed on the SLS console.

When the configuration is completed, MCS systems lock themselves and signal their readiness for the pulse.

Diagnostic physicists select or edit the SFH to match the requirements of the DP. Diagnostics signal their readiness by attaching to a server processes and waiting for the distribution of the next pulse number.

Upon request by the APs, parameters for their configuration files are computed from DP, i/o database, MCS and diagnostic settings. The parameter values are inserted into the configuration files and downloaded to the DCS.

A mapping for the real-time signals onto the RTN is computed and downloaded to dedicated RTN communication processes of the DCS, based on requirements of the APs and the visualization clients. At this point all systems are fully configured.

Execution

Before starting pulse execution, the XM operator checks status of protection systems, and the readiness of the actuator systems. The next pulse number is distributed to the diagnostics. Timers of the timing network are reset. Then, the DCS takes over control and starts executing the pulse performing its control cycles. During the pulse, XM offers real-time services, such as logging and visualization of system states and real-time signals. After the pulse, focus of control is passed back to XM.

Post-Pulse Activities

This is the only operation phase that is not initiated by the XM operator. Post-pulse activities involve analyzing protocols and logs and archiving them, together with the configuration data for later reference.

Experiment Management System Overview

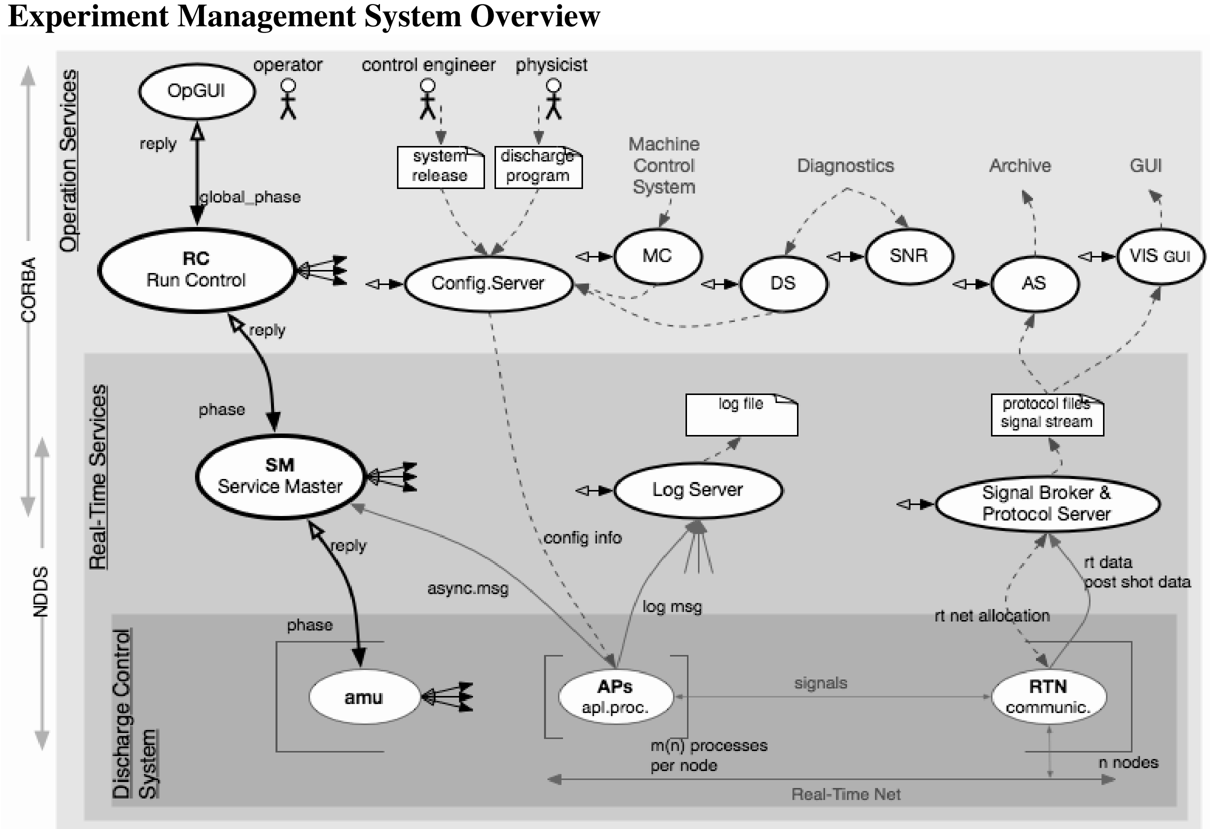

Figure 2: Overview of XM processes, user interfaces and data bases

Operation Services Layer

Processes of this layer deal with tasks outside the execution phase. Their focus lies on flow control, the communication with external systems, and the configuration activities.

The **OpGUI** is the central interface of XM and implements a view-controller for the Run Control Process (RC) described below. It contains command elements for setting operation modes, for the selection of a DP, and issuing the phase commands described in the previous section. It also has visualizing elements for displaying phase relevant information, alarms, logs, MCS and protection system states and parameters.

The **Run Control Process** (RC) implements a state machine for the operation phases. It receives the high-level phase commands for booting, preparation, configuration, and execution from the OpGUI and breaks them down into elementary commands for the Services Master of the Real-Time Services Layer. At the end of the pulse it autonomously triggers postshot activities. At start-up, RC instructs the processes of the RTS layer to start NDDS and establish their connections among each other and with the DCS.

The **Visualizer GUI** (VisGUI) allows to select signals for on-line visualization during the pulse and to define max. sampling frequencies used to update the plots. VisGUI connects to the Signal Broker and Protocol Process of the RTS layer but is otherwise autonomous. Several VisGUIs may run simultaneously.

The **Archive Server Process** (AS) converts the XML log file into HTML for viewing in a web browser. It also tags the DP with the pulse number and bundles, compresses, and archives the configuration, protocol, and log files.

The **Machine Control Server** (MS) provides the single interface for communication with the MCS. It is connected to the SLS and has functions for reading and writing named MCS parameters and state information through a CORBA interface. It holds a database for mapping the parameters from the fixed MCS address space, into a dedicated namespace. The same database contains a history of parameter values which have been accessed. MS is queried by RC for MCS readiness before the configuration phase and, once more, before pulse execution to check that the MCS settings have remained unchanged.

The **Diagnostic Server** (DS), a process not yet implemented, will provide functions for accessing sensor settings required for DCS ouput process configuration. Currently this data is stored in XML files.

The **Configuration Server** (CS) provides the single access point for all configuration data requests by APs and other XM processes. To satisfy these requests it accesses data from various sources (databases, XML files, MC), processes them, and converts them into the desired form. A CORBA interface allows to set its operation mode from the OpGUI, e.g. to define alternate values for MCS parameters for testing purposes.

The **Shotnumber Daemon** (SNR) is a small process which distributes the next shotnumber to the diagnostics just before the pulse.

Real-Time Services (RTS) Layer.

These are the processes that closely cooperate with the DCS. They perform specialized taks during pulse preparation and execution. RTS layer processes communicate with the DCS through NDDS, a networking middleware from RTI, specially designed for applications that need to process real-time data. All RTS layer processes also have a CORBA interface through which RC can send requests to start and stop their NDDS communication channels.

The **Service Master Process** (SM) is a stateless process that issues low level flow control commands to the DCS and the processes of the XM real-time layer. During its initialization phase SM reads a list of services (dedicated processes of the DCS and real-time layer XM processes), defined in the System Release file, which need to receive phase commands for pulse preparation and execution. Communication with the services is via NDDS. SM broadcasts the commands it receives from RC to the services, and the application management units (AMU) on the DCS controllers and collects their replies in its own reply to the command. SM can also receive asynchronous messages from services and APs and forward them to clients which have registered for them.

The **Signal Broker and Protocol Process** (SBP) collects real-time and postshot signal protocols, converts them into XML and writes them into files and/or streams them to The Visualizer GUIs. Before the pulse, it receives lists of produced and requested real-time signals from DCS, and visualization clients. During DCS configuration, it produces the signal mapping for the RTN. Optionally, SBP writes the memory map to a file. APs are informed which of the signals they produce are actually requested. Communication processes [1] on the controllers use the memory map to read and write real-signals produced or requested by APs.

The **Log Server** (LS) provides an NDDS interface for real-time log messages from DCS applications and a CORBA interface for log messages from XM processes. It writes them into a large ring buffer and converts them into XML and and stores them in a file upon request. Log messages have a set of fixed attributes (timestamp, source, severity) which allow filter functions to be applied to them. LS also streams log data to an arbitrary number of registered clients, e.g. user interfaces, via CORBA, allowing them to define filters upon registration. It accepts commands through a CORBA interface for opening and closing the log file and defining its name.

Summary and Outlook

ASDEX Upgrade's Experiment Management System provides the binding agent between most of the many independent and heterogeneous components of the tokamak experiment. A majority of its parts have been successfully employed already during this year's experiment campaign, whereas the final, missing bits are being commissioned for operation during the coming campaign. So far, the Management System has helped a great deal in allowing shorter experiment cycles, simplifying experiment operation, and hiding unnecessary or annoying details from users by automatically performing repetitive tasks and transparently handling protocol conversions between the different controller families. Organizing logging, warning and error messages from the various participating services into machine-readable XML documents has proven highly useful for analyzing system behaviour on many levels. The main user interface as a central access point for real-time services is capable of presenting a wealth of information for the system engineers, as well as offering a very compact command console to operators.

To further enhance the experiment management environment, additional services are in development.

For the technical preparation phase of a shot, we need tools for cross-validation of MCS and PCS configuration sources; physicists find a flight simulator [4] useful which simulates the planned discharge in a slightly simplified physics model using reference trajectories from the DP and machine parameters from MCS settings. Evaluation of DP trajectories in conjunction with MCS settings allows on-the-fly calculation of other MCS parameters, such as the predicted energy consumption of the toroidal field magnet.

After a shot, analysis of technical and physics performance will be facilitated by the confluence of protocolled measurements, logged control system performance and configuration data, all of which are present as XML documents and as such easily processed.

References

[1] W. Treutterer et al., "Software Structure and Real-time Signal Exchange for the ASDEX Upgrade Tokamak Control System", paper TH111, ICALEDCS '03, Gyeongju, Korea

[2] G. Raupp, et al. : A “"Universal Time" System for ASDEX Upgrade, Proc. 22nd Symposium on Fusion Technology, Helsinki (SF), 2002, published in Fusion Engineering and Design 66-68 (2003), p.947ff

[3] G. Neu et al., "ASDEX Upgrade Tokamak Control Parameter Organization and Configuration Management", paper WP527, ICALEPCS '03, Gyeongju, Korea

[4] W. Suttrop, L. Höllt: "Predictive simulation of tokamak discharge behaviour based on simple scalings", to be published in http://eps2005.ciemat.es/, Paper Nr. P4.076.