ProShell - The MedAustron Accelerator Control Procedure Framework

R. Moser1, A. B. Brett1, J. Dedić3, J. Gutleber2, M. Marchhart1, C. Torcato De Matos1, S. Sah3
1EBG MedAustron, Wiener Neustadt, Austria
2CERN, Geneva, Switzerland
3Cosylab, Ljubljana, Slovenia

ABSTRACT
MedAustron is an ion therapy and research centre presently under construction in Wiener Neustadt, Austria. It features a synchrotron particle accelerator for protons and carbon-ion beams. This paper presents the architecture and concepts for implementing an object-oriented framework called ProShell. Procedures to automate high level control and analysis tasks need to be defined at an object-oriented single state-driven device interface. Components can extend these device interfaces following an object-oriented single state-driven device interface that provides a minimal set of DPEs for monitoring.

INTRODUCTION
MedAustron is an ion therapy and research centre presently under construction in Wiener Neustadt, Austria. The facility features a synchrotron particle accelerator for up to 5 ion sources for protons, carbon ions and possibly other light ions. It will provide ion beams with energies up to 800MeV to 5 beam lines, one of which is a rotating proton gantry. The Procedure Shell Execution Framework (ProShell) is a C# application to automate high level control and analysis tasks for commissioning and during operation. Each task called a procedure implements a standardized procedure interface and is deployed as NET assembly (shared objects).

ARCHITECTURE
ProShell is a framework to dynamically load and execute procedures implemented as C# classes. It provides access to system, software and physical devices for monitoring and control purposes. These services are accessible from ProShell through service-specific Driver objects that are used internally and are not directly accessible from the procedures.

PROCEDURE CONTEXT
Each procedure is executed in a separate Procedure Context that acts as a container to provide coordinated access to devices and control system services and manages the procedure lifecycle with contexts and resources.

RESOURCES
Resources are made available through adapter objects that shield procedures from the underlying addressing and communication specifics. Each adapter owns a set of fields that implement the IElement interface. Classes implementing the IElement interface interact with drivers to communicate with the front-end controllers and (1) implement data type conversion and (2) perform client-side validity checks. They implement one of the following interfaces:

• BasicDevice is a state-less front-end device interface that provides a minimal set of DPEs for monitoring.
• StateDrivenDevice (SDD) is a state-driven front-end device interface that extends the BasicDevice to provide mechanisms for sending commands to front-end controllers and for login to gain exclusive access to a device. ProShell also provides resource adapters to control a set of SDDs concurrently through a single virtual device:
  • Working Set (WS) is a virtual device that implements the SDD interface and controls a set of SDDs.
  • Virtual Accelerator (VAA) controls a set of Working Sets and subsequently a set of devices that are connected.

BEAM SPECTRUM ANALYSIS
The ion source beam spectrum analysis procedure detects the particles types generated by a specific ion source. Therefore a current is applied to the bending magnet. The generated field deflects the particles in an angle depending on the particle mass and the applied field. The energy of the particles that hit the following faraday cup is measured. Due to correlation of current to particle type, the generated particles can be detected with peak detection algorithm on the energy over current plot (Figure 5).

REFERENCES