

# Control Database Requirements

December 8, 1998

The Control Database will be used in the monitoring and control of the DØ experiment in Run 2.

It should store the essential attributes of hardware devices and provide the Control DAQ (based on EPICS) with the information required to read from or write to a device. In particular it should contain information on detector systems, low and high voltage power supplies, cryogenics, argon purity monitoring and environmental conditioning.

In addition to monitoring and control during the runs, the database should support the Control DAQ in downloading the detector control system, prior to a run start, with appropriate device settings for selected run conditions.

The design of the Control Database should be aided by, but not bound to, the content, scope of, and the experience with the Hardware Database of Run 1. It should also be determined which data, if any could be salvaged and extracted into the new Control Database.

**The Control Database should serve the following needs:**

## **1. generate extractions of EPICS device records**

EPICS has its own database, which resides on a host (OPI), to be downloaded to front end processors (IOC's). There are 2 classes of EPICS database files, usually kept in a text format:

- ".dbd" files with definitions of record types
- ".db" files with record instances; a file like this serves one particular IOC only.

A record contains a number of fields, the set of which depends upon the specific record type. At present, the types, standard and DØ-defined, to be supported are:

Analog Input / Output  
Binary Input / Output  
Waveform (array support)  
VME (bus structure)  
Camac (bus structure)  
RM (Rack Monitor module)  
HV (High Voltage)  
MIL1553  
SVXSeq (sequencer module)

A record type instance must have a unique name, say "device", across all IOC's attached to the same sub-net; its "device.field" entities resemble "device.attribute" entities of Run 1, except that Input (read) and Output (write) functions are disjoint in EPICS.

Records can be linked together to create control algorithms and sequences.

The Control Database should accommodate the record instances (data contained in ".db" files) of the supported record types, and their linkages, in an easily retrievable way. A utility should be written to extract them in a form needed by EPICS.

## **2. accommodate mapping of several record types into more complex objects**

A pattern of record types and links between them (e.g. all the records related to a module in VME crate) should be mapped into a unique object, called a structure device in Run 1 Hardware Database implementation, or a template in EPICS.

It is desirable that instances of the constituents, which are the same for a number of object instances, are amenable to re-using, i.e. they should be entered only once into the Control Database and then referenced by the object instances.

### **3. provide necessary information for the Alarm System**

The design of the Alarm System should influence the implementation of alarms in the database.

Nominal device readings, their limits, alarm severity, action upon alarm, etc, should be adequately described in the database. Detailed information about a device, which has caused an alarm, should be easily retrievable upon the alarm id.

### **4. provide an information for a generic Parameter Page**

#### **GUI interface**

The interface should facilitate entering the data into the Control Database and generating EPICS objects instances including the complex objects based on the templates stored in the Database.

Description of data (i.e. constraints on field values) for menu choices should be based on definitions in EPICS ".dbd" files. The necessary description should therefore be also stored in the Database itself and from there made available for the GUI.