

## DATABASE USE IN APPLICATION PROGRAMMING AT SNS

J. Galambos<sup>1</sup>, C. Chu<sup>1</sup>, J. Patton<sup>1</sup>, T. Pelaia<sup>1</sup>, A. Shishlo<sup>1</sup>  
<sup>1</sup>SNS, Oak Ridge National Laboratory, Oak Ridge TN

### ABSTRACT

The Spallation Neutron Source (SNS) application programming effort uses database storage in several areas. A primary use is in the configuration of the accelerator hierarchy used in our Java based application programming framework. This task involves considerable effort in data population, but facilitates the ability to write general purpose applications. Configuring the machine definition in this approach for the entire application programming framework also eases code maintenance. Another area of database use is in the storage of measured machine values. This later usage case typically involves working with collected sets of data, and the data population tends to be automatic. Several tools have been written to hide the intricacies of the database connections to programmers. Experience in both of these areas is discussed.

### ACCELERATOR HEIRARCHY

The Java based high level application programming framework at SNS [1, 2] includes a hierarchal view of the accelerator. In this view, shown schematically in Figure 1, the accelerator is composed of sequences, sequences are composed of devices, and devices can have their own characteristics. This is a typical object-oriented view of the accelerator, and in our case the hierarchy is configured from database tables. One advantage of this structure is the ability to write an application once and have it reusable anywhere in the accelerator. Another advantage of this approach is that single point changes in the database configuration are propagated to many applications, easing maintenance. A consequence of this approach is the necessity of careful population and maintenance of the database.

The tables are part of a global SNS relational database [3]. As this global database is used by other groups at SNS, the table structure is not a one-to-one match with the class structures used in XAL. Much of the information needed for the setup of the physics applications is “mined” from tables maintained by engineers. We have written a program that queries the database to configure the XAL structure. The overall organization of these tables is discussed below, as well as the approach used to setup the XAL accelerator hierarchy.

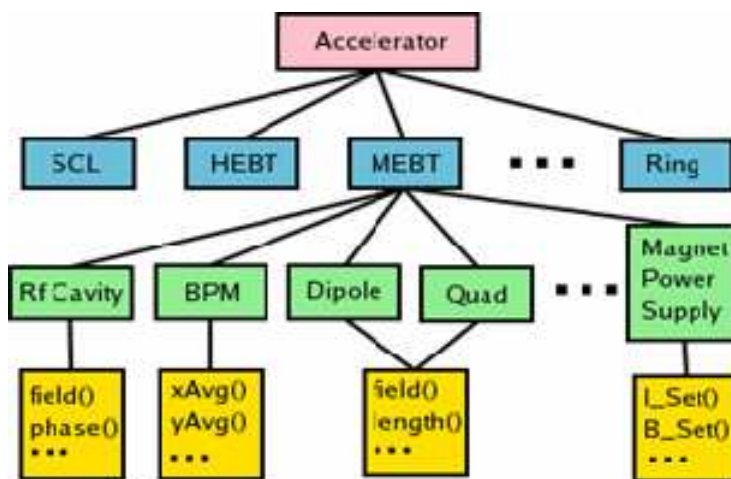


Figure 1: XAL Accelerator hierarchy indicating an accelerator composed of sequences (blue), which are composed of devices (green), which have methods and attributes (yellow).

## *Devices and Equipment*

There is a high level delineation of data storage between devices and equipment in the SNS database tables. Devices are the abstract placeholders for objects that a beamline is composed of (magnets, RF, diagnostics devices etc.). The devices all have prescribed global coordinate positions (used by alignment personnel), a distance along the design beam trajectory (used by beam modellers), and are given names which generally follow a naming convention that allows easy understanding of what and where the device is.

Each device has a specific installed piece of hardware at any given time, which is the equipment. All pieces of equipment use a bar-code number, for their identification. There is an association table that relates which piece of equipment is installed at each device. The information specific to the hardware is associated with the equipment. For example, magnet measurement data is associated with the equipment, not the device. If a magnet fails and is swapped out, a new piece of equipment (i.e. the new magnet) is associated with the device. Much of the data associated with the equipments is held in tables maintained by various groups (magnet measurement, power supply, diagnostics, etc.) but is critical for the setup of the high-level applications. Sometimes information that is characteristic of a class of equipment is stored in tables for an equipment model type (e.g. power supply limits are all the same for individual power supplies of a given type).

We use data from about 25 tables to setup the XAL accelerator hierarchy. This is done via a programmed set of queries which presently takes 10's of seconds, but has not been optimized. Rather than have every application perform this query on start-up, we create an XML file containing the accelerator configuration, in the structure used by XAL. Presently this XML file is over 20,000 lines long for the SNS linac alone. The accelerator configuration does not change often within a beam run, so the occasional updating of the XML configuration file has not been problematic. Also, storage of these files provides an easy way to reconstruct the accelerator's setup for a given time in the past. Importantly, the device has types of signals associated with it, and the signal names which are used by the EPICS control system [4] for the real machine communication are specified in this file. It is this link that allows XAL to provide generic device methods that hide the details of the control system from the user (e.g. a magnet method to get the applied field, of a Beam, Position Monitor (BPM) method to get the measured beam location).

## **MEASURED DATA STORAGE**

The EPICS control system provides tools for archiving and retrieving large amounts of data, and is the primary mechanism used for data archiving at SNS. Mirroring the EPICS control system, the standard method of archiving information is oriented around individual signals. This is useful and works well when the different combinations of signals to be retrieved are not known a priori. However, there are occasions when collection of prescribed collections of signals is anticipated. An example is the collection of all information needed to set up a model from machine settings and compare to measured beam, e.g. the magnet settings, RF settings, and the BPM measurements ("physics" data). While it may be possible to collect the hundreds of signals needed to for this task from the EPICS archiver, this would be tedious. Moreover, there would be no guarantee that all the values retrieved were from the same beam pulse (SNS is a pulsed machine that will run at 60 Hz). A PVLogger tool has been written to facilitate logging groups of data (PV stands for Process Variable in EPICS, which is like a signal). The important distinction between this tool and the standard EPICS archiver, is that this tool collects data as a group, and can guarantee that all data within a group are collected from the same pulse, by requiring signals to have timestamps within a certain range of each other.

This capability is available as either a background service that runs continuously in the background, or as a pushbutton tool. We log the physics data every fifteen minutes in the background service, and other applications that take measurements, such as wire profile measurements and scanning applications take snapshots whenever measurements are taken. There is a separate application to

browse for a group snapshot by time, and to view the value of a logged signal vs. time (from one snapshot to another), and is shown in Figure 2. Also another application provides capability for on-the-fly setup of a new group of signals to be logged, as well as specification of the frequency to log the data .

Signals:		Machine Snapshots:		Selected Snapshot:	
Contains: [X]		Timestamp	ID	Signal	Value
<input checked="" type="checkbox"/>	CCL_Diag:BCM102:currentAvg	2005-09-10 11:20:58.0	1322890	CCL_Diag:BCM102:currentAvg	9.86...
<input checked="" type="checkbox"/>	CCL_Diag:BCM102:currentMax	2005-09-10 11:35:58.0	1322921	CCL_Diag:BCM102:currentMax	11.0...
<input checked="" type="checkbox"/>	CCL_Diag:BPM101:amplitudeAvg	2005-09-10 12:05:58.0	1322984	CCL_Diag:BPM101:amplitudeAvg	15.8...
<input checked="" type="checkbox"/>	CCL_Diag:BPM101:phaseAvg	2005-09-10 12:20:58.0	1323017	CCL_Diag:BPM101:phaseAvg	-167...
<input checked="" type="checkbox"/>	CCL_Diag:BPM101:xAvg	2005-09-10 12:35:58.0	1323048	CCL_Diag:BPM101:xAvg	0.53...
<input checked="" type="checkbox"/>	CCL_Diag:BPM101:yAvg	2005-09-10 12:50:58.0	1323080	CCL_Diag:BPM101:yAvg	-1.91...
<input checked="" type="checkbox"/>	CCL_Diag:BPM103:amplitudeAvg	2005-09-10 13:20:58.0	1323144	CCL_Diag:BPM103:amplitudeAvg	15.4...
<input checked="" type="checkbox"/>	CCL_Diag:BPM103:phaseAvg	2005-09-10 13:35:58.0	1323175	CCL_Diag:BPM103:phaseAvg	-8.07...
<input checked="" type="checkbox"/>	CCL_Diag:BPM103:xAvg	2005-09-10 13:50:58.0	1323207	CCL_Diag:BPM103:xAvg	-0.15...
<input checked="" type="checkbox"/>	CCL_Diag:BPM103:yAvg	2005-09-10 14:05:58.0	1323238	CCL_Diag:BPM103:yAvg	-0.67...
<input checked="" type="checkbox"/>	CCL_Diag:BPM112:amplitudeAvg	2005-09-10 14:20:58.0	1323271	CCL_Diag:BPM112:amplitudeAvg	17.9...
<input checked="" type="checkbox"/>	CCL_Diag:BPM112:phaseAvg	2005-09-10 14:35:58.0	1323302	CCL_Diag:BPM112:phaseAvg	36.2...
<input checked="" type="checkbox"/>	CCL_Diag:BPM112:xAvg	2005-09-10 14:50:58.0	1323334	CCL_Diag:BPM112:xAvg	0.07...
<input checked="" type="checkbox"/>	CCL_Diag:BPM112:yAvg	2005-09-10 15:05:58.0	1323365	CCL_Diag:BPM112:yAvg	-0.81...
<input checked="" type="checkbox"/>	CCL_Diag:BPM202:amplitudeAvg	2005-09-10 15:20:58.0	1323398	CCL_Diag:BPM202:amplitudeAvg	16.2...
<input checked="" type="checkbox"/>	CCL_Diag:BPM202:phaseAvg	2005-09-10 15:35:58.0	1323429	CCL_Diag:BPM202:phaseAvg	-148...
<input checked="" type="checkbox"/>	CCL_Diag:BPM202:xAvg	2005-09-10 15:50:58.0	1323461	CCL_Diag:BPM202:xAvg	-0.53...
<input checked="" type="checkbox"/>	CCL_Diag:BPM202:yAvg	2005-09-10 16:05:58.0	1323492	CCL_Diag:BPM202:yAvg	0.79...
<input checked="" type="checkbox"/>	CCL_Diag:BPM212:amplitudeAvg	2005-09-10 16:20:58.0	1323525	CCL_Diag:BPM212:amplitudeAvg	15.8...
<input checked="" type="checkbox"/>	CCL_Diag:BPM212:phaseAvg	2005-09-10 16:35:58.0	1323556	CCL_Diag:BPM212:phaseAvg	-166...
<input checked="" type="checkbox"/>	CCL_Diag:BPM212:xAvg	2005-09-10 16:50:58.0	1323588	CCL_Diag:BPM212:xAvg	1.11...
<input checked="" type="checkbox"/>	CCL_Diag:BPM212:yAvg	2005-09-10 17:05:58.0	1323619	CCL_Diag:BPM212:yAvg	-0.34...
<input checked="" type="checkbox"/>	CCL_Diag:BPM302:amplitudeAvg	2005-09-10 17:20:58.0	1323652	CCL_Diag:BPM302:amplitudeAvg	16.4...
<input checked="" type="checkbox"/>	CCL_Diag:BPM302:phaseAvg	2005-09-10 17:35:58.0	1323683	CCL_Diag:BPM302:phaseAvg	-172...
<input checked="" type="checkbox"/>	CCL_Diag:BPM302:xAvg	2005-09-10 17:50:58.0	1323715	CCL_Diag:BPM302:xAvg	0.13...
<input checked="" type="checkbox"/>	CCL_Diag:BPM302:yAvg	2005-09-10 18:05:58.0	1323746	CCL_Diag:BPM302:yAvg	-0.79...
<input checked="" type="checkbox"/>	CCL_Diag:BPM312:amplitudeAvg	2005-09-10 18:20:58.0	1323779	CCL_Diag:BPM312:amplitudeAvg	16.4...
<input checked="" type="checkbox"/>	CCL_Diag:BPM312:phaseAvg	2005-09-10 18:35:58.0	1323810	CCL_Diag:BPM312:phaseAvg	176...
<input checked="" type="checkbox"/>	CCL_Diag:BPM312:xAvg	2005-09-10 18:50:58.0	1323842	CCL_Diag:BPM312:xAvg	1.67...
<input checked="" type="checkbox"/>	CCL_Diag:BPM312:yAvg	2005-09-10 19:05:58.0	1323873	CCL_Diag:BPM312:yAvg	-0.20...
<input checked="" type="checkbox"/>	CCL_Diag:BPM402:amplitudeAvg	2005-09-10 19:20:58.0	1323906	CCL_Diag:BPM402:amplitudeAvg	19.8...
<input checked="" type="checkbox"/>	CCL_Diag:BPM402:phaseAvg	2005-09-10 19:35:58.0	1323937	CCL_Diag:BPM402:phaseAvg	-12.0...
<input checked="" type="checkbox"/>	CCL_Diag:BPM402:xAvg	2005-09-10 19:50:58.0	1323969	CCL_Diag:BPM402:xAvg	2.25...
<input checked="" type="checkbox"/>	CCL_Diag:BPM402:yAvg	2005-09-10 20:05:58.0	1324000	CCL_Diag:BPM402:yAvg	0.11...
<input checked="" type="checkbox"/>	CCL_Diag:BPM409:amplitudeAvg			CCL_Diag:BPM409:amplitudeAvg	16.6...

Figure 2: Browser application for retrieving PVLogger snapshots. The group to select (filterable) is shown in the left panel, The available snapshots in the specified time-range are shown in the middle panel, and the specific snapshot is shown in the right panel.

By far the most important application of the logged data is configuration of the online model [5] of the accelerator. The SNS online model is a simple envelope model used for transverse and longitudinal lattice calculations. It is fast enough to be used in the control room. While the basic lattice geometry is fixed and can be configured from the database as described above, specific setup of a large accelerator changes, especially during commissioning periods as with the SNS. For example, an operator may tweak a quadrupole. In order to analyze data it is important to know the machine setup when the data was collected, and the PVLogger provides this capability. As mentioned, applications that take measurements typically also request a PVLogger snapshot, and store an identification number of this snapshot with the measured data. The online model has a PVLogger data-source which sets adjustable model input (e.g. magnetic field levels) to those of a given PVLogger snapshot. This capability is shared by several applications that use the online model for data analysis and machine setup, as well as single-use scripts.

## SAVE-COMPARE-RESTORE (SCORE)

All large accelerator systems have some sort of save, restore capability for tuning the machine. It is always desirable to be able to easily recover a good machine setup, after (perhaps unsuccessfully) attempting to improve things. At SNS an application called SCORE provides this capability, and it uses a database for all data storage (see Figure 3). Collections of data that are desired to be viewed together are setup as a group. Examples of groups that have been used so far are the general operator setup of the machine, diagnostic device configuration, and some RF equipment configuration. The groups consist of defining sets of setpoint and readback signals, each with a system and sub-system

identifier. These identifications can be used to filter query results. A collection (snapshot) of each group can be taken, and is delineated by the timestamp of the snapshot.

This panel is in testing mode

From: Aug 31, 2005 11:18:15 To: Sep 24, 2005 11:18:15

Date	Comment
2005-09-02 04:20:32.0	Tuned up thru cavity 22d. ~909 MeV. Full energy quad settings. Orbit corrected...
2005-09-02 06:32:28.0	Better orbit correction through SCL29 but beyond is not optimized. Also DTL5-...
2005-09-03 19:06:57.0	magnet settings used in SCLHigh -> linac dump orbit difference studies.
2005-09-03 20:33:18.0	Correct trajectory through linac dump. Good warm + scl fingerprints.
2005-09-04 11:39:46.0	Snapshot of conditions prior to SLACS reproducibility scans.
2005-09-04 19:58:13.0	Score for the cavity 11 SLAC statistics studies, where the cavity 11 energy manag...
2005-09-06 23:50:47.0	Correct trajectory through linac dump. Good warm + scl fingerprints.
2005-09-07 01:15:48.0	Machine state for the linac dump wire profile measurements that will be used to ...
2005-09-10 10:41:20.0	tuned SCL up to CM15 and will load quad law CM15_08_29.enman
2005-09-10 10:43:10.0	tuned SCL up to CM15 and will load quad law CM15_08_29.enman
2005-09-10 10:44:47.0	tuned SCL up to CM15 and loaded quad law CM15_08_29.enman
2005-09-10 10:58:33.0	tuned SCL up to CM15 and loaded quad law CM15_08_29.enman
2005-09-10 16:18:29.0	tuned SCL up to SCL 2.1c and will load quad law FullEnergy_08_29.enman
2005-09-10 16:19:13.0	tuned SCL up to SCL 2.1c and loaded quad law FullEnergy_08_29.enman
2005-09-10 16:23:33.0	tuned SCL up to SCL 2.1c and loaded quad law FullEnergy_08_29.enman, LDmp ...
2005-09-11 02:26:48.0	LW12/15wpg
2005-09-11 05:44:52.0	Sept10, new SCL tuneup 901 MeV
2005-09-11 06:46:58.0	SCL Full Energy 902 MeV, and smaller beam losses
2005-09-11 11:37:23.0	SCL cavities set according to SCL80percent_tuneup_2005_09_10-new.sxc. Linac ...
2005-09-11 11:58:07.0	SCL cavities set according to SCL80percent_tuneup_2005_09_10-new.sxc. Linac ...
2005-09-12 09:01:26.0	Snapshot of machine for fault study with losses CM23, ~387 MeV.
2005-09-12 10:02:53.0	Snapshot of machine for fault study with losses at CM20, 387 MeV (actually clos...
2005-09-12 12:31:55.0	Machine state for the linac dump wire profile measurements that will be used to ...
2005-09-12 12:34:17.0	Machine parameters for spill in vicinity of CM16.
2005-09-12 14:41:53.0	Machine parameters for 187 MeV spill in vicinity of CM23.
2005-09-13 18:50:48.0	As found beam conditions with linac dump quad setpoints not right for 900 MeV

Fetch selected snapshot    Fetch golden snapshot

No snapshot data

(a)

Type	Setpoint name	SP Save Val	SP live Val	Readback Name	RB Save Val	RB live Val
CCL	CCL_Mag:ShntC_QTV109:I...	14.6826	14.6826	CCL_Mag:ShntC_QTV109:I...	14.5769	-0.0101
CF	CCL_Mag:ShntC_QTV111:I...	20.3881	20.3881	CCL_Mag:ShntC_QTV111:I...	20.2579	0.0010
DTL	CCL_Mag:ShntC_QTV201:I...	3.9945	3.9945	CCL_Mag:ShntC_QTV201:I...	3.9383	-0.0050
HEBT	CCL_Mag:ShntC_QTV203:I...	9.5871	9.5871	CCL_Mag:ShntC_QTV203:I...	9.5037	-0.0129
LDmp	CCL_Mag:ShntC_QTV205:I...	15.0836	15.0836	CCL_Mag:ShntC_QTV205:I...	15.1758	0.0160
LEBT	CCL_Mag:ShntC_QTV207:I...	20.5663	20.5663	CCL_Mag:ShntC_QTV207:I...	20.5627	0.0264
MEBT	CCL_Mag:ShntC_QTV209:I...	3.9924	3.9924	CCL_Mag:ShntC_QTV209:I...	3.8520	0.0164
RFO	CCL_Mag:ShntC_QTV211:I...	9.3836	9.3836	CCL_Mag:ShntC_QTV211:I...	9.3388	0.0086
	CCL_Mag:ShntC_QTV301:I...	14.6784	14.6784	CCL_Mag:ShntC_QTV301:I...	14.5175	0.0247
CU	CCL_Mag:ShntC_QTV303:I...	19.8666	19.8666	CCL_Mag:ShntC_QTV303:I...	19.7936	0.0568
HPRF	CCL_Mag:ShntC_QTV305:I...	3.3934	3.3934	CCL_Mag:ShntC_QTV305:I...	3.2548	0.0009
MPS	CCL_Mag:ShntC_QTV307:I...	9.1830	9.1830	CCL_Mag:ShntC_QTV307:I...	9.1241	0.0839
Mag	CCL_Mag:ShntC_QTV309:I...	14.9727	14.9727	CCL_Mag:ShntC_QTV309:I...	14.9235	-0.0128
RF	CCL_Mag:ShntC_QTV311:I...	20.6641	20.6641	CCL_Mag:ShntC_QTV311:I...	20.6526	0.0311
Setup	CCL_Mag:ShntC_QTV401:I...	3.2946	3.2946	CCL_Mag:ShntC_QTV401:I...	3.2648	0.0418
chop	CCL_Mag:ShntC_QTV403:I...	8.8840	8.8840	CCL_Mag:ShntC_QTV403:I...	8.8050	0.0246
volts	CCL_Mag:ShntC_QTV405:I...	14.3726	14.3726	CCL_Mag:ShntC_QTV405:I...	14.3756	0.0156
	CCL_Mag:ShntC_QTV407:I...	19.8710	19.8710	CCL_Mag:ShntC_QTV407:I...	19.8407	0.0374
RF	CCL_LLRF:FCM1:CtlAmpSet	0.7768	0.000E0	CCL_LLRF:FCM1:cavAmpAvg	0.7758	0.0005
	CCL_LLRF:FCM1:CtlPhaseSet	-86.2720	-86.2720	CCL_LLRF:FCM1:cavPhase...	-85.9036	28.0647
	CCL_LLRF:FCM2:CtlAmpSet	0.7251	0.000E0	CCL_LLRF:FCM2:cavAmpAvg	0.7228	0.0007
	CCL_LLRF:FCM2:CtlPhaseSet	-52.5000	-52.5000	CCL_LLRF:FCM2:cavPhase...	-52.5409	7.8642
	CCL_LLRF:FCM3:CtlAmpSet	0.6620	0.000E0	CCL_LLRF:FCM3:cavAmpAvg	0.6598	0.0015
	CCL_LLRF:FCM3:CtlPhaseSet	22.0000	22.0000	CCL_LLRF:FCM3:cavPhase...	22.0108	57.0622
	CCL_LLRF:FCM4:CtlAmpSet	0.8202	0.000E0	CCL_LLRF:FCM4:cavAmpAvg	0.8180	0.0007
	CCL_LLRF:FCM4:CtlPhaseSet	12.4000	12.4000	CCL_LLRF:FCM4:cavPhase...	12.5071	-5.0894
				CCL_LLRF:FCM1:cavW	24.9282	0.0502
				CCL_LLRF:FCM2:cavW	31.3045	0.0722
				CCL_LLRF:FCM3:cavW	30.5774	0.0539
				CCL_LLRF:FCM4:cavW	2.3000	0.0036

Machine data saved at 2005-09-11 11:58:07.0

(b)

Figure 3: Score application screenshots. (a) The panel used to select a group, and snapshot to fetch. (b) The compare feature for a specific system. Items in red indicate live values more than a specified amount different than the saved value. Individual rows can be selected restoring saved values.

Originally this application used an XML file based system for the storage of snapshots. Transition to the use a database has offered several benefits. First, the maintenance of the many files taken during a given run is eliminated. Also, previous snapshot retrieval was based on file browsing (using a file name protocol to identify when the data was taken). Use of a database to store information provides flexibility of offering the user more than just the timestamp of a machine snapshot for data retrieval, e.g. now we also offer a short text description in the snapshot browsing feature. To do this previously would have required opening and parsing of many files.

## COMMENTS

The high level applications programming effort at SNS is using a database as the primary means of machine setup configuration. This design choice was made early on, and has had several ramifications. One impact is that a significant effort has gone into data-population. This effort has paid dividends already, in the ability to deploy general purpose applications. Also, critical machine information is stored in a central repository, instead of personal spreadsheets, or files on different computers. Regarding the use of a database to store measured machine data, this also has been a positive experience. In addition to the flexibility of using database storage of information, maintenance is easier – files in multiple directories no longer have to be managed. We anticipate increasing the database use with the high level applications.

## ACKNOWLEDGEMENT

SNS is managed by UT-Battelle, LLC, under contract DE-AC05-00OR22725 for the U.S. Department of Energy. SNS is a partnership of six national laboratories: Argonne, Brookhaven, Jefferson, Lawrence Berkeley, Los Alamos, and Oak Ridge.

## REFERENCES

- [1] <http://www.sns.gov/APGroup/appProg/xal/xal.htm>.
- [2] J. Galambos, *et al.*, “XAL Application Programming Structure”, Particle Accelerator Conference, Knoxville, Tennessee, USA, May 2005.
- [3] <http://www-ssrl.slac.stanford.edu/lcls/epics/agenda.php>
- [4] <http://www.aps.anl.gov/epics/>
- [5] C. Allen, *et al.*, “A Novel Online Simulator for High-Level Control Applications Requiring A Model Reference”, ICALEPCS’2003, Gyeongju, Korea, October 2003.