SNS GLOBAL DATABASE USE IN APPLICATION PROGRAMMING *

J. Galambos, C. Chu, E. Danilova, A. Shishlo, J. Patton, T. Pelaia, ORNL, Oak Ridge TN, USA,
W.-D. Klotz, ESRF, Grenoble France

Abstract

A global relational database is being assembled to track accelerator components for the Spallation Neutron Source (SNS). As part of this activity, beamline element information is stored for use in high level application programs. A hierarchal accelerator framework is generated from the database and used for initialization of a Java based programming infrastructure. From within this framework input files for beam simulation codes can be generated using either live machine values or design values. The database also includes global coordinates for beamline element alignment, and magnet measurement data. An overview of the table schema and relationships to tables used in other parts of the project are discussed.

INTRODUCTION

The approach of the Spallation Neutron Source (SNS) application programming effort [1] is to write general purpose software which can be applied to any part of the SNS accelerator. To facilitate this, the SNS global database plays an important role as the primary repository for the static data used to configuring applications for a specific parts of the accelerator. In addition to application program configuration, the beamline device information stored in the database is used for other purposes, such as storage of the global coordinates defining the ideal and actual lattice elements positions, design values for magnets and RF, and for external lattice generation.

This paper discusses the general architecture of the beamline device portion of the database. The following sections describe the organization of the beamline device schema, describe some of the ways data is extracted, and show some example applications. There are many other parts of the global database not covered here, particularly related to the equipment aspects. These areas include equipment tracking, maintenance, and calibration data. More detailed information about the SNS global database may be found in reference [2].

DATABASE ORGANIZATION

An important aspect of the SNS global database is its organization around two distinct conceptual components: devices and equipment. A device instance is an abstract component with perhaps some design values. An equipment instance is an actual piece of hardware with a barcode. Generic information (e.g. design values or position along the lattice) is stored with the device and equipment specific information (e.g. magnet measurement) is stored with the equipment. For example, every magnet has a device record containing its design field, lattice position, etc. There is also a bar-coded piece of equipment that occupies each magnet position. The equipment portion of the magnet database contains the magnet measurement information. A device-equipment association table relates which piece of equipment is installed in a particular device. If a piece of equipment breaks, and is replaced by another, none of the device information changes; only the device to equipment association needs to be modified.

Beamline Devices

Devices are required to be identified by four different predefined pieces: system, subsystem, type and instance. Concatenation of these pieces is the core of the SNS naming convention. The “system” is similar to a primary lattice sequence (e.g. Drift Tube Linac (DTL), Ring, etc.); subsystems include magnets, diagnostic, power supply, etc.; type includes Quad, BPM, RF, etc., and for example “instance” is 1,2,3, etc. Devices include components throughout the accelerator complex, including racks, computer controllers etc. Subsets of the devices are indicated as beamline devices – i.e. components that would normally be found in a modeling lattice file. It is the beamline devices that the application programming is most directly concerned with, and the rest of this paper primarily addresses. But we emphasis that there are no hard boundaries in the global database schema – the application programming configuration also has to access some equipment data. Population and use of the database crosses group administrative boundaries.

Each item that would be treated as an independent element in a typical beam-tracking model is handled as an independent beamline device. For example, in the SNS there are single hardware pieces, each consisting of a quadrupole, a dipole corrector and a BPM. Each of these are treated as three separate beamline devices, all at the same beamline position. Also, each magnetic pole of a magnet assembly is treated as an independent beamline device.

* 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.

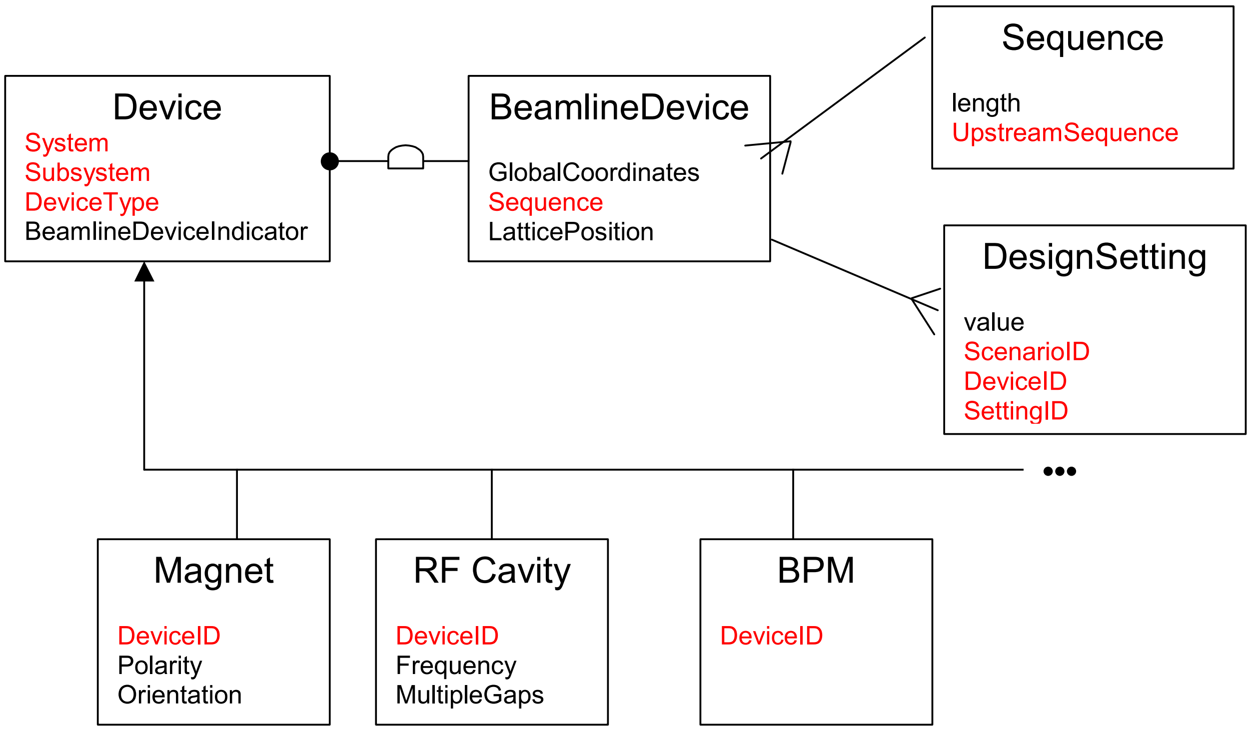

Figure 1. Simplified schematic of the Beamline-device related tables in the SNS global database. Items in red are foreign keys to other table records.

Relationship to XAL Classes

The application-programming framework [2] is Java based and has a class hierarchy describing the accelerator. The SNS database is a relational database (not object oriented), but the table structure is organized to simplify the beamline mapping to the XAL class structure. As shown in Figure 1, beamline device elements belong to a sequence. Information is also stored regarding the beamline device placement within a sequence and which sequences may be concatenated with others. Also, each beamline device has a specified type (e.g. BPM, Magnet, etc.). By using the type identification, proper tables can be found to obtain additional information specific to that device. Also there are tables to hold design values (e.g. baseline field setting for a magnet). Design values have a scenario ID, allowing multiple sets of design values to be stored (but at present, only a single "baseline" set of design values is used).

Connection to the Control System

An important aspect of the application programming framework initialization is relating the beamline devices to the appropriate control system signals. SNS uses the EPICS [3] control system which uses "Process Variables" or PVs to communicate between clients and Front-End computers. The EPICS PVs are stored in separate tables and used to initialize EPICS setup files (or EPICS "database" files). In the XAL setup procedure, a prescribed subset of the available PVs for each device type are selected and assigned a "handle" that describes their function. For instance we assign the PV that supplies the average horizontal position from a BPM the handle "xAvg". The handle can be used in general actions over a set of devices of the same type.

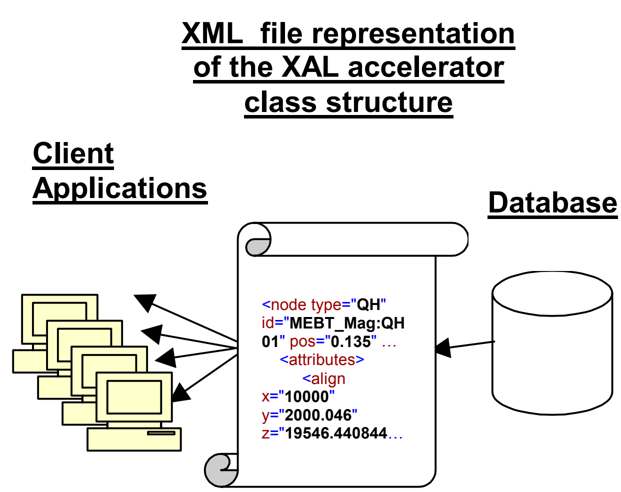

Figure 2. Data flow path from the database to the application programs.

APPLICATIONS

For application programs, the primary use of the database is to acquire the accelerator hierarchy class structure initialization. This is represented schematically in Fig. 2. An XML file is generated by a java-based query, which is subsequently read by client applications on startup. A direct database query from the applications is also possible, but we use the intermediate XML file for reasons of speed and flexibility for quick changes. In the future, a service is planned that will query the database and dynamically provide clients the latest accelerator structure.

Device and Lattice views

Only information about complete physical devices is stored in the database. This view of the accelerator is referred to as the accelerator "data-graph", and is adequate by itself for some applications (e.g. displaying the BPM orbit vs. position). However, for modeling purposes devices must be subdivided and drift spaces between physical elements must be accounted for. This different way of viewing the accelerator is depicted in Fig. 3. The database contains only information about device positions and lengths. All drift space calculation and element splitting is done within the XAL framework. Also depicted in Fig. 3 are views of the "Devices" and "Lattice elements" in XML formats which are used within XAL. This lattice view is used in the XAL online model [4], as well as for creating external model lattice files.

Magnet Families

Another important consideration in configuring the application framework is knowledge of magnet families (on common power supplies) and magnet excitation curve measurements. The mapping of magnets to power supplies is included in the beamline devices tables. Field mapping data is also stored in special magnet measurement tables. This information is associated with specific equipment (as opposed to devices). The implementation of current to field transformations is to be done at the Front End Controller level (i.e. field information will be available via EPICS). However, the database is being used to provide the necessary field mapping and magnet family information for Front End Controller initialization.

EXPERIENCE

Organization and population of the database has proved to be more labor intensive than originally anticipated. Part of this is due to increasing schema complexity driven by the overlapping needs of diverse groups using the database. At present the beamline elements are roughly halfway populated and the database is being used as the configuration base for the application programs (the SNS accelerator is much less than halfway commissioned now). Even in the initial commissioning stages use of the database to configure applications has proved useful. For example, modification of only a single point to accommodate multiple applications realization that BPM electronics have moved from one device to another. We anticipate the flexibility offered by the use of a database to reap further dividends as SNS moves towards operations.

Acknowledgements

Many thanks to D. O Jeon, E. Tanke, S. Henderson, J. Wang for helping gather data for populating the tables.

REFERENCES

[1] C. Chu, J. Galambos, T. Pelaia, A. Shishlo, C. Allen, N. Pattengale, "Application Programming Structure and Physics Applications", these proceedings.
[2] http://it.sns.ornl.gov/mis/MIS_IT.htm
[3] http://www.aps.anl.gov/epics/
[4] C.K. Allen, C. Chu, J. Galambos, T. A. Pelaia, N. D. Pattengale, "A Modular, Online Simulator for Model Reference Control of Charged Particle Beams", these proceedings.

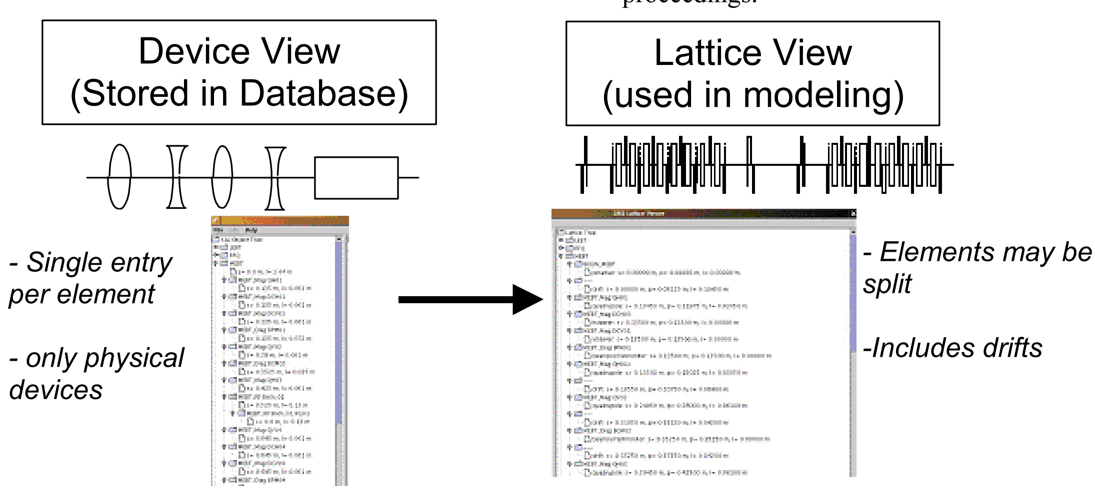

Fig. 3 Schematic of the device view of data abstraction used in the database, and the lattice view abstraction used in setting up modeling sets. The lattice view is constructed within the XAL framework.