HIGH AVAILABILITY ON-LINE RELATIONAL DATABASES FOR ACCELERATOR CONTROL AND OPERATION*

D. A. Dohan, L.R. Dalesio, G. Carcassi, BNL, UPTON, NY 11973, U.S.A.

Abstract

The role that relational database (RDB) technology plays in accelerator control and operation continues to grow in such areas as electronic logbooks, machine parameter definitions, and facility infrastructure management. RDBs are increasingly relied upon to provide the official 'master' copy of these data. Whereas the services provided by the RDB have traditionally not been 'mission critical', the availability of modern RDB management systems is now equivalent to that of standard computer file-systems. RDBs can be relied on to supply pseudo real-time response to operator and machine physicist requests. This paper describes recent developments in the IRMIS RDB [1] project. Generic lattice support has been added, serving as the driver for model-based machine control. Abstract physics name service and process variable introspection has been added. Specific emphasis has been placed both on providing fast response time to accelerator operators and modeling code requests, as well as high (24/7) availability of the RDB service.

INTRODUCTION

Relational Database technology has matured to the point where it now routinely relied upon to provide on-line services for accelerator and control. This movement is accompanied with a paradigm shift – where the data and schema are fully exposed, promoting user-driven schema enhancements and application development.

A key requirement in this endeavor is the availability of the RDB as measured by:

Latency. Responses to typical control room requests should be met promptly (~1 sec). Slower queries without the immediacy requirements should be provided by alternative services.

24/7 Availability. The RDB service must be available at all times. Operating system and database software maintenance and upgrades must be applied without loss of RDB service. This is particularly important during shutdown periods, during which the demand for RDB services is often the highest.

With the proven reliability of both hardware and software technology, RDB management systems are now accepted as the official or master repository for control and operational data. This implies tight requirements on data integrity. It also places stringent requirements on the response time for control room and accelerator application database queries.

*Work performed under auspices of the U.S. Department of Energy under Contract No. DE-AC02-98CH10886 with Brookhaven Science Associates, LLC.

RELIABILITY AND AVAILABILITY

Commercial server class hardware, along with reliable open source RDB software systems (MySQL, PostgreSQL, …) are readily affordable solutions to the reliability and availability requirements. Figure 1 shows the master/slave architecture installed at the NSL2 facility under construction at BNL.

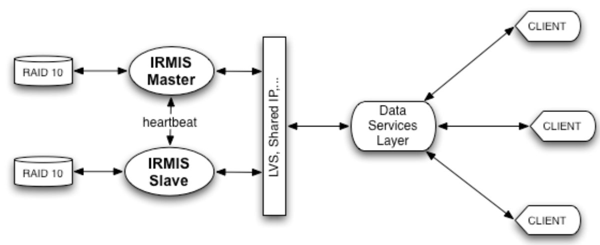

Figure 1: Master/Slave architecture for RDB redundancy and availability.

In this architecture, all database inserts are directed to the active master node; these inserts are replicated in the slave server. The heartbeat signal provides the mechanism for failover in the event of master node failure. The slave node is off-line for operating system and RDBMS maintenance and to provide self-consistent database backups. This is accomplished without interruption to the IRMIS RDB service. The architecture allows for query load balancing and permits direction of slow queries to the slave server, while maximizing the throughput for mission critical queries.

THE IRMIS COMPONENT INSTALLATION MODEL

In the IRMIS hardware schema, a *component* is defined as any unit-replaceable physical entity associated with the accelerator/facility. An IRMIS component belongs to a specific component class or '*component-type'* characterized by the type's name, description, manufacturer and a set of functions that the component can perform [2]. All component types are captured in a single table - there is no classification mechanism to categorize IRMIS component types. As well as recording the existence of each component, its relationships with other components are also captured. Three types of inter-component relationships are defined: these relationships capture how the component is assembled, how it is controlled, and the component's power source. The entire facility is modeled, including infrastructure items such as buildings, rooms, racks and crates.

The component installation information is captured in the IRMIS component INSTALL sub-domain, shown in

Figure 2. This sub-domain is the central feature of the IRMIS hardware model and is the primary mechanism for component fault diagnosis and location. The INSTALL domain, and has strict data consistency requirements, along with the latency and availability requirements discussed earlier.

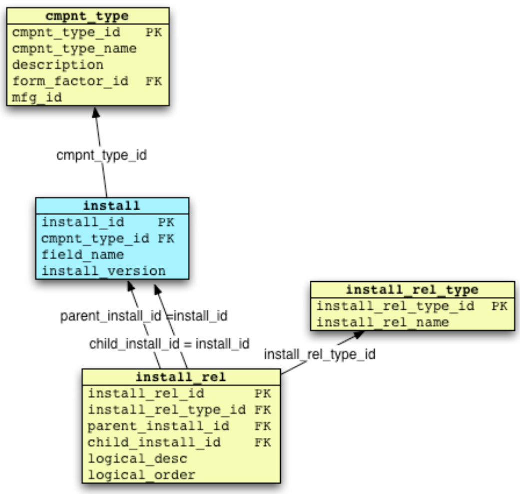

Figure 2: IRMIS component INSTALL domain.

LOOSE COUPLING AND DOMAIN SEPARATION.

The IRMIS install domain is the base structure around which other relational sub-domains are modeled and constructed.

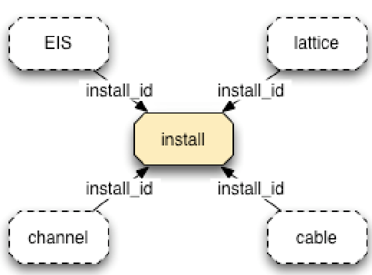

Figure 3: IRMIS loose coupled sub-domains.

These ancillary sub-domains are essentially disjoint from the main INSTALL domain, and follow separate schema and application prototyping and development cycles. Loose coupling between each sub-domain to the install table is achieved with a single foreign key, defined within the attached sub-domain. The functionality provided by these sub-domains is described below:

EIS. The Equipment Inventory System captures component history information. This includes historical data concerning calibration, installation, repair, purchase record, safety certifications, etc. Asset information related to the purchase/warranty of the physical component instance is also captured. The EIS database schema and application is prototyped separately from the INSTALL database. Third party or locally developed inventory management systems may used to provide this functionality, with the simple requirement that the foreign key relating the asset item to the INSTALL table be implemented.

Cable. The CABLE sub-domain models cables as component port-to-port connections. The cable sub-domain retains foreign keys between the CABLE domain connected ports and the INSTALL domain components, otherwise it retains a separate development and prototyping cycle independent of the stable INSTALL domain.

Lattice. The lattice database provides a single, centralized location of the machine description from a machine physics point of view. At NSLS2, a model-based control strategy is under development to address the highly non-linear nature of the NSLS2 storage ring beam dynamics implies. The IRMIS lattice provides the lattice description and is the driver for creating TRACY and Elegant modeling code input files. The lattice schema introduces a 4th hierarchy to the IRMIS model – the (abstract) accelerator hierarchy. A generic lattice consists of a set of nested ‘sequence’ elements, culminating in the leaf nodes consisting of the installed beam manipulation elements. Lattice element properties are stored in an extensible key/value attribute table.

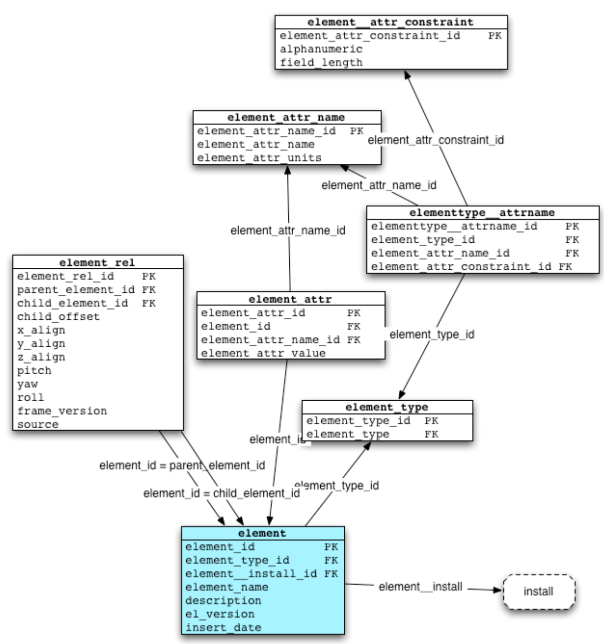

Figure 4: The LATTICE sub-domain.

Channel. The component type concept has been extended to include the list of I/O streams or “channels” that a particular component type exports. These are the leaf devices in the control hierarchy. When a hardware item is installed, the associated channel list is added to the available list of channels at the disposal of the EPICS developer. For block oriented components, a separate table captures the "Interface Control Document" – the contract between the EPICS developer and the PLC developer.

SUMMARY

Relational databases are an established, mature technology for the commissioning and operation of modern accelerators. They provide powerful tools for managing the complexity of modern accelerators, while addressing the constantly changing nature of data.

The loose coupling architecture in the IRMIS schema has proven to be very useful in providing a stable environment (the INSTALL domain) for permanent data entry, while providing the capability for the implementation of new application capabilities.

REFERENCES

[1] D.A.Dohan. Component/Connection/Signal Modeling of Accelerator Systems, PAC'05, Knoxville, May 2005, 707.

[2] D. A. Dohan. IRMIS Universal Component-Type Model, ICALEPCS 2007, Knoxville, TOPA03.

[3] Guobao Shen. A Software Architecture For High Level Applications, this conference.