WWW-based Accelerator Data Warehouse

A. Chan, G. Crane, I. MacGregor, S. Meyer and R. Sass
Stanford Linear Accelerator Center*, Stanford, CA 94309 U.S.A.

Abstract

In addition to real time data needed for on-line control of an accelerator, there is a need for prompt access to other data required to support and maintain an operating accelerator complex. This includes data on trouble-reporting, spare parts, fabrication, calibration, drawings, etc. This paper focuses on the latter infrastructure data, which is necessary to maintain an operational facility at peak efficiency. We discuss using the World Wide Web (WWW) interface and data repositories to link these data, which are generally segregated by departments and resides on different computer platforms and software. A cornerstone of this is the collaborative effort across departments to conform to a site-wide Formal Device Name, derived from the real-time control system. The disparate databases are linked through this Formal Device Name field.

1 Is there a need for a data warehouse

In an accelerator laboratory we are inundated with data related to the fabrication and maintenance of the machine, such as hardware, personnel, finance [1]--yet it is more often than not quite difficult to obtain such related information in a useful format. To resolve a maintenance problem, one may need to log onto diverse computer systems and run different database software, frequently devoting much time to trying out obscure search strings.

Such compartmentalization of databases of the various accelerator departments evolved because different software was used to suit particular problems and situations. Although there is growing recognition to achieve database implementation under the same database software, we will still be faced to some degree with these differences.

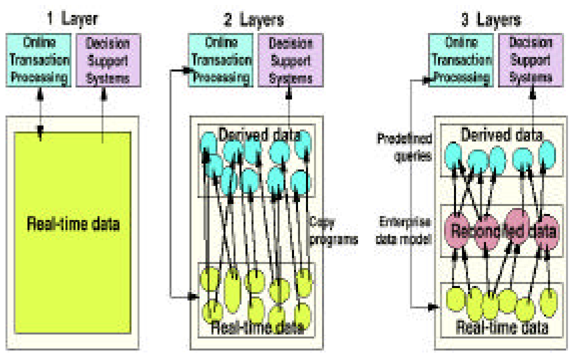

Figure 1 -- Types of implementation of data warehouse

Another reason for this compartmentalization is that databases developed by the departments, of necessity, focus on their area of business functions. This short-term focus can be mitigated by recognizing the need to make linkages to the overall accelerator during the design of the database. However, the implementation of a data warehouse is required for a lab-wide view of decision support information for the accelerator.

2 Data warehouse in an accelerator environment

A data warehouse can be simply defined as [2] ″a single, complete, and consistent store of data obtained from a variety of sources and made available to end users in a way they can understand and use in a business context.″

There are 3 main architectures for a data warehouse (see Fig. 1), although in reality what is implemented is most likely a mixture of these different types. The data layers that Fig. 1 refers to are conceptual, rather than physical.

For the single-layer architecture, its strength is that data is only stored once, which avoids the need to synchronize multiple copies of data. Its weakness is that contention can occur between the online transaction processing systems and the decision support systems.

For the two-layer architecture, its strengths lie in solving the contention between the online transaction processing systems and the decision support systems. It addresses the fact that end-user needs for information are different from what is easily available from real-time data. However, there is a high level of data duplication in the two-layer approach (with a tendency to become ′spaghetti code′).

For the three-layer architecture, its strengths lie in the reconciled layer which is based on enterprise data modeling (i.e., a normalized database). This reconciled layer can support new, unanticipated end-user needs. The derived layer can be used to fill most end user needs, being the equivalent of predefined queries. The enterprise data modeling is a much more committed effort, and needs to be done incrementally.

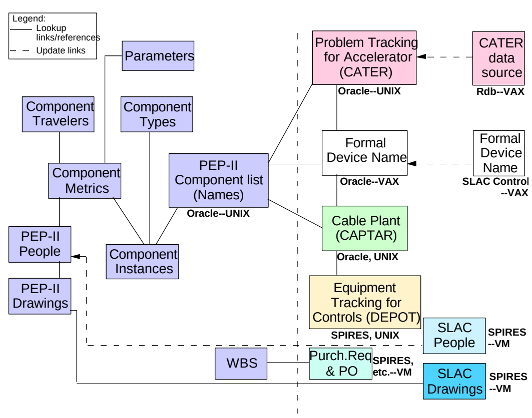

Figure 2 -- Linked SLAC accelerator databases

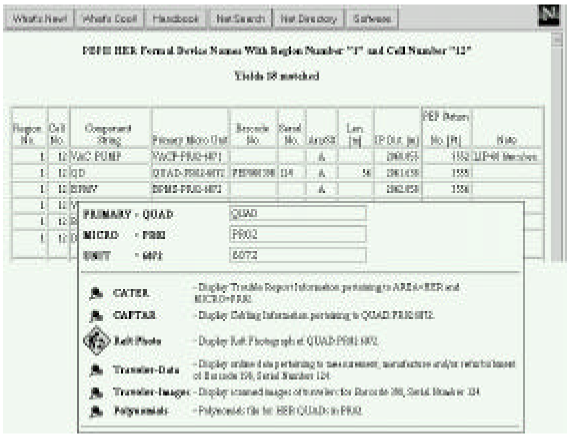

Figure 3 -- WWW screens of Formal Device Names, and menu choices for each formal device identified by 'Primary, Micro, Unit'

At SLAC, we have mostly implemented the single layer and two-layer approaches. We recognize the benefits of the three-layer data warehouse which will take some future commitment in effort.

The World Wide Web (WWW) technology has made it possible to access and link disparate databases, allowing the data warehouse to be a reality for us, and at a fraction of the programming effort in today′s resource-scarce environment.

3 Site-wide device names

The key to our ability to provide a lab-wide view of the data is a consistent set of site-wide device names (Formal Device Names). We took the Formal Device Names from the accelerator on-line control system, with its 37,000 records, as the master list. We used the Formal Device Names as the link to the other disparate off-line accelerator databases.

The real-time database used by the accelerator control system is rigidly structured and designed for high-speed access by the many different control system computers. As such, it is not in a format suitable for access outside of the control system. To bridge the gap, an Oracle Device Database is updated whenever there are any structural changes to the real-time database. This Device Database is then available for remote access like any other Oracle database.

Similar to other sites, we encountered two problems in our ability to join disparate accelerator databases

- where the Formal Device Name fields existed in other databases, the data may not have been validated
- other databases may define their device names differently, leading to imprecise mapping

We find that the effort for implementing data integrity in these disparate databases is an iterative one. When the user community can utilize some of the information provided by such a linked system, we are able to obtain more support from management for the extra effort in requiring data validation in the different departments.

Table 1. Linkage of data elements.

Formal Device Name	*Primary* QUAD	*Micro* PR02	*Unit* 6072
Problem Tracking for Accelerator (CATER)	*Area* HER (not required) LER	*Micro* PR02	*Primary Unit*
Cable Plant (CAPTAR)	*System Function* QUAD:PR02,6072		
	Micro LI09	*Crate* CR02	*Model* 233-002-00
Equipment Tracking for Controls (DEPOT)	*Location* LI09/CR02		*Model* 233-002

4 Linking the lab databases

Figure 2 is an overview of the lab′s major accelerator-related databases that we have linked for the accelerator data warehouse. These systems are:

- the The Formal Device Names. This is integrated into the PEP-II project-wide database [3] which is in Oracle, containing component fabrication history and calibrations.
- the problem tracking database for the accelerator (CATER), which is in VAX Rdb and replicated into Oracle tables.
- the cable plant database (CAPTAR) in Oracle.
- the equipment tracking database for Controls (DEPOT) in SPIRES.

Via WWW Common Gateway Interface (CGI) scripts, these databases are linked together using common data elements (i.e., variations of formal device names shown in Table 1). The WWW interface gives excellent ′drill-down′ capabilities. Since linkages of the different databases are done by the CGI scripts, huge savings in programming time are realized as we do not have to create physical tables to join the databases.

Users can query the Formal Device Names List on WWW, and the search results will contain hypertext references that request information from the other database systems. In Fig. 3, among the search results returned for components in Region 1 Cell 12 is the Primary/Micro/Unit QUAD PR02 6072 (the three fields comprising the formal device name), which has a hypertext link that produces a menu with additional hypertext links to:

⇒ the CATER database summary and detail screens for Area HER or LER, Micro PR02 (Fig. 4).

⇒ the CAPTAR database of cables linked to QUAD PR02 6072. From the Micro field, one can obtain crate profiles which are database reports in WWW tables (Fig. 5)

⇒ the DEPOT database with summary screen of the history of modules in this crate location, and detail screens of the maintenance records of each module is linked from the CAPTAR crate profile. In Fig. 5, this link is shown for Micro LI09 Crate CR02 Model 233-002.

⇒ the scanned paper fabrication and alignment travelers, online calibration measurements, drawings, photos, polynomials files, etc. of the PEP-II Project Database.

5 Lessons and issues

This accelerator-wide view of the information delivered through the WWW interface has been popular with users, enabling more efficient work.

There are a few key issues that have enabled us to get to this stage for an accelerator data warehouse

- We ′web-ified′ both new or existing main disparate databases at the lab. This provided immediate benefits to the departments, and they were very willing to work with us.
- We approached each system with a lab-wide view of the information-not a departmental view.
- We focused on linking the existing departmental databases-not on rebuilding systems. Re-engineering is a major effort that our scarce programming resources cannot easily undertake, so we have left that effort to the departments-though, hopefully, there is coordination amongst related database work.
- We built incrementally, for each increment giving quick turnaround and visible benefits to the users and the departments.
- We relied on experience database programmer(s) for data modeling.

With the ability now to link databases via a few common data fields, the integrity of the data stored in these fields need to be enhanced by cleansing the data and/or applying better database constraints. But this effort becomes easier when the benefits are now apparent to the users (even if it may not benefit their own department immediately).

For PEP-II, EPICS has been used for several major subsystems. Unfortunately, these devices are contained within a separate database and are presently not reflected in the Oracle Device Database. Future work on the Device Database should include integrating the EPICS database as well as other devices not specifically accessed by the control system.

In the future, we would like to rely more on commercial WWW-database tools (such as Oracle Designer2000, Remedy, etc.), and less on CGI scripts which are time consuming and less secure.

The URL for the PEP-II Project Database is:

http://www.slac.stanford.edu/accel/pepii/db.htm

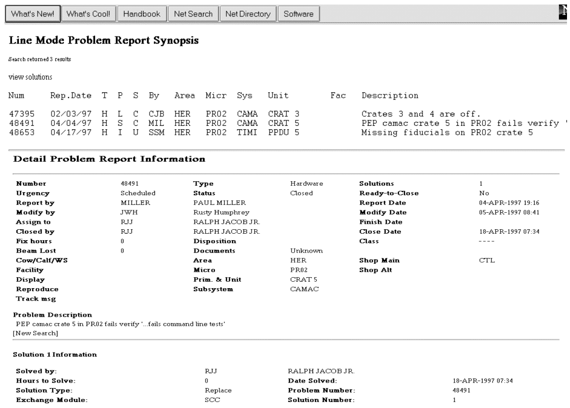

Figure 4 -- Summary and detail WWW screens of accelerator problem report database (CATER) linked from ‘Area HER, Micro PR02’ in figure 3

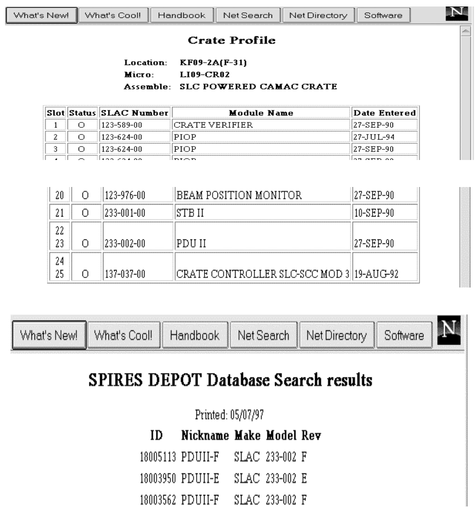

Figure 5 -- WWW screen of crate LI09/CR02 -- crate profile from cable plant database (CAPTAR), and linked history of modules in the Equipement Tracking for Controls database (DEPOT) which leads to maintenance history screens

Acknowledgments

We thank the people in PEP-II and the various Technical Division departments who manage the different databases, and who helped us analyze the linkages of these different systems. We have drawn many lessons from discussions with members of the International Accelerator Database Group (IADBG).

References

[1] J. Poole, ″Databases for Accelerator Control An Operations Viewpoint″, Proceedings of 1995 Particle Accelerator Conference, Dallas, May 1-5, 1995, p.2157.

[2] Barry Devlin, ″Data Warehouse: from Architecture to Implementation″, Addison-Wesley, 1997.

[3] A. Chan, G. Crane, I. MacGregor, S. Meyer, ″The PEP-II/*BaBar* Project-Wide Database Using World Wide Web and Oracle*CASE″, Proceedings of 1995 International Accelerator Database Group Workshop, Argonne National Lab, November 6-8, 1995.