THE PEP-II PROJECT-WIDE DATABASE

A. Chan, S. Calish, G. Crane, I. MacGregor, S. Meyer, J. Wong, Stanford Linear Accelerator Center, Stanford University, Stanford, CA 94309 USA

A. Weinstein, Lawrence Livermore National Laboratory, Livermore, CA 94550, U.S.A.

The PEP-II Project Database is a tool for monitoring the technical and documentation aspects of this accelerator construction. It holds the PEP-II design specifications, fabrication and installation data in one integrated system. Key pieces of the database include the machine parameter list, magnet and vacuum fabrication data, CAD drawings, publications and documentation, survey and alignment data, and property control (see Fig. 1). The database can be extended to contain information required for the operations phase of the accelerator and detector. Features such as viewing CAD drawing graphics from the database will be implemented in the future. This central Oracle database on a UNIX server is built using Oracle*Case tools. Users at the three collaborating laboratories (SLAC, LBL, LLNL) can access the data remotely, using various desktop computer platforms and graphical interfaces.

I. GOALS

The PEP-II Project Management during the R&D phase in 1992 had conceived of using a central database as a tool for the three collaborating labs to:

1. track and ensure quality assurance during construction of the project
2. track down problems faster when they develop
3. facilitate the construction process.

Corresponding systems for the *BABAR* detector were included in this vision. These functions required an integrated technical and administrative database. Data that existed then was on multiple, disparate desktop computer systems or on paper, which made data less reliable and harder to share amongst users. Project Management decided to support the design and implementation of a central database based on Oracle. Oracle was chosen because it is a modern, relational database; it runs on the client/server

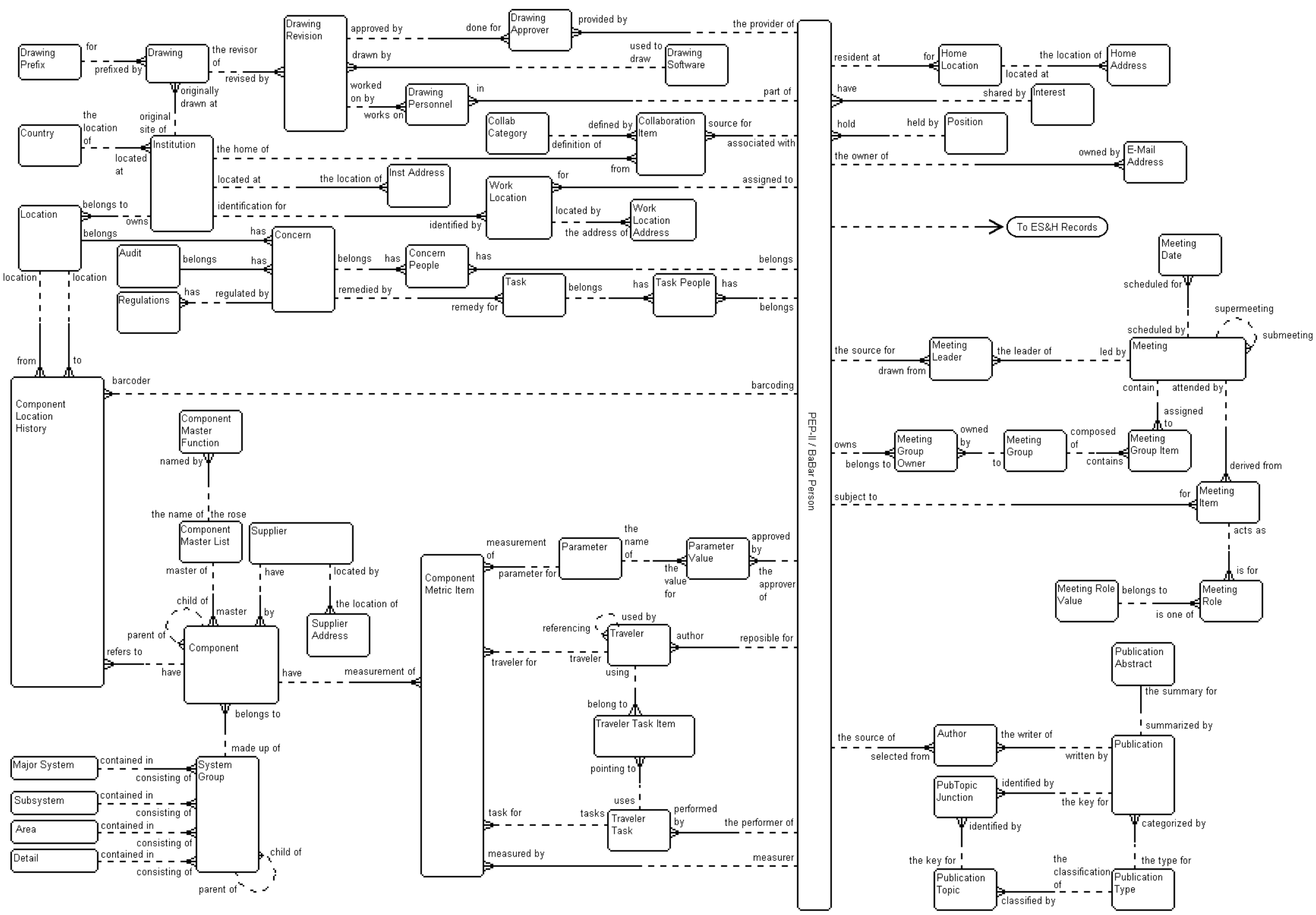


Fig. 1. PEP-II Project-Wide Database Entities and Relationships Diagram generated by Oracle*CASE.

*This work was supported by the U.S. Department of Energy, under contract numbers DE-AC03-76SF00515 and W-7405-Eng-48.

architecture PEP-II is using (clients being Macintosh and IBM-PC, server being UNIX RS6000), it is already site-licensed at SLAC and, in addition, is widely used at other DOE labs.

In July 1992, several Project Managers and programmers visited CERN to learn from their 10-year experience working with Oracle databases. The trip was most useful and we brought away the following key points:

1. have a vision of the overall database from the beginning so hooks are left for further expansion (otherwise data from different systems might not converge later or might be in the wrong format)
2. use Computer Aided System Engineering (CASE) for a structured database analysis, and for faster development and modifications since the database, screens, and reports are one integrated package within CASE rather than loose pieces of code
3. management commitment is crucial
4. involve users in the design from the beginning.

II. IMPLEMENTING KEY PIECES

To deal with the contradiction between the large scope of an enterprise-wide database and the pressing immediate needs of a construction project, we focused on getting key pieces of the skeleton database running right away. Other pieces were tied to management and production needs, as they arose.

Key pieces (see Fig. 2) were:

1. Personnel (most systems in the database have relationships with this entity); this includes a platform-independent e-mail distribution system
2. Drawings (mainly, but not exclusively, CAD)
3. Components

The Components system was the heart of the technical part of the database, and we revised this design many times. In Fig. 1, the Entities and Relationships Diagram shows that for a component to exist in the database it must be entered in the Component Master List entity. The component can have many parameters and their corresponding design values. It can also have a drawing number and a revision

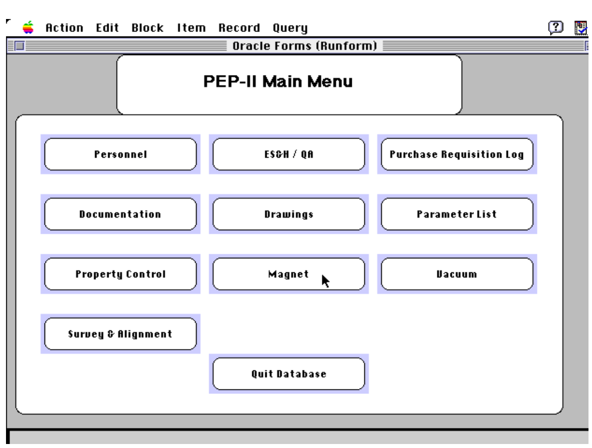


Fig. 2. Main Menu from Oracle Forms4.

number. Each instance of the component is entered into the Component entity. The component instance is produced according to the order of procedures from a traveler which has a traveler number and revision number. A traveler is a set of fabrication instructions and measurements to manufacture that component instance. In the traveler, the fabrication instructions and measurements are identified by task numbers. The measurement values are stored in the Component Metric Item entity. The Component Location History entity records the history of physical locations of the component instance, with the final designation being its destination in the PEP-II tunnel.

For example, a Component Master List entry of PEP-II High Energy Ring (HER) dipole magnet will have design parameters like Bdl measured at 650 A. An instance of a component is the HER dipole magnet with serial number 148 (see Fig. 3). It is refurbished according to instructions on traveler number 1, revision number 4. In this traveler, the measurement for Bdl at 650 A is task number 1000032. The Bdl at 650 A (relative to the reference magnet) for magnet serial number 148 is 0.99167.

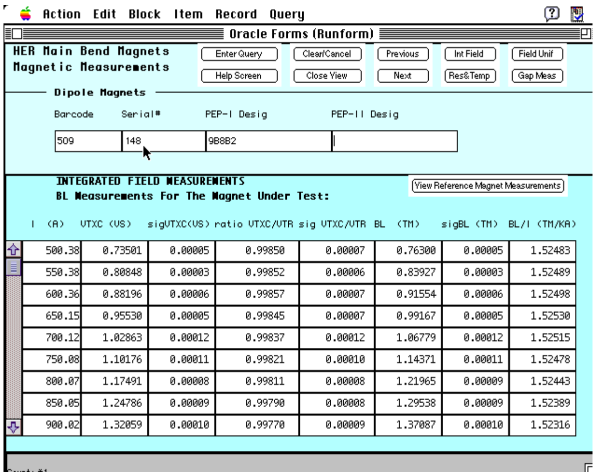
We have tested this Components design with PEP-II magnet and vacuum systems, and the *BABAR* calorimeter crystals system. Through these normalized tables relating to the Components system, we are able to retrieve fabrication and measurement data by many criteria. The users can query against views that we have built which join these normalized, but fragmented, tables together. Data can also be dumped from the database by the Clear Access program into many other software packages preferred by the users, like Microsoft Excel (see Fig. 4).

Important data that are under change control, like changes in parameter values, are journaled by the database. Historical data for the refurbished PEP components are also stored in the Components system.

III. USER INTERFACE ISSUES

A Graphical User Interface (GUI) that is intuitive and easy to use is key to the PEP-II Project Database being widely adopted by the Project Management and the general user community. In fact, while the database means solid design of entity relationships and table definitions to the programmers, to the users the screens and reports interface *is* the database. Therefore, we have tried to quickly give key pieces of the database to the users so that they can test the interface and the data. CASE has been an important part in shortening this development cycle. Although using CASE requires a greater investment in startup time (more efforts at design and populating common tables), it is easier later to add modules to the enterprise-wide database and respond to user modifications.

Since a construction project like PEP-II has a tight schedule, generally we do not have the luxury of having sufficient analysis time with the engineers and physicists to gather their requirements. We had to rely on very short



Action Edit Block Item Record Query

Oracle Forms (Runform)

HER Main Bend Magnets
Magnetic Measurements

Enter Query | Clear/Cancel | Previous | Int Field | Field Unif
Help Screen | Close View | Next | Res&Temp | Gap Meas

Dipole Magnets

Barcode	Serial#	PEP-I Desig	PEP-II Desig
509	148	9B8B2	

INTEGRATED FIELD MEASUREMENTS
BL Measurements For The Magnet Under Test:

View Reference Magnet Measurements

I (A)	VTXC (VS)	sigVTXC(VS)	ratio VTXC/VTR	sig VTXC/VTR	BL (TM)	sigBL (TM)	BL/I (TM/KA)
500.38	0.73501	0.00005	0.99850	0.00007	0.76300	0.00005	1.52483
550.38	0.80848	0.00003	0.99852	0.00006	0.83927	0.00003	1.52489
600.36	0.88196	0.00006	0.99857	0.00007	0.91554	0.00006	1.52498
650.15	0.95530	0.00005	0.99845	0.00007	0.99167	0.00005	1.52530
700.12	1.02863	0.00012	0.99837	0.00012	1.06779	0.00012	1.52515
750.08	1.10176	0.00011	0.99821	0.00010	1.14371	0.00011	1.52478
800.07	1.17491	0.00008	0.99811	0.00008	1.21965	0.00009	1.52443
850.05	1.24786	0.00009	0.99790	0.00008	1.29538	0.00009	1.52389
900.02	1.32059	0.00010	0.99770	0.00009	1.37087	0.00010	1.52316

Fig. 3. Bdl measurements for HER Dipole Magnet serial number 148 from Oracle Forms4

meetings for analysis, followed by developing the screens and reports based on what we hope was close to 80% correct in the relationships and functions. Time and again, the users told us that they themselves have only a vague idea of what they want. They need to play with the interface in order to know what an enterprise database will do, and to be able to give us feedback to correct the initial design. So in our situation, the textbook case of heavy up front investment to get as close as possible to 100% correct analysis is not practical, unlike the situation for mission-critical databases like accelerator controls or banking.

The World Wide Web (WWW) interface to the Project Database that we built in recent months has been very popular. Besides being easy to use and easy to access, WWW can retrieve and join data from Oracle to legacy databases such as SPIRES (see Fig. 5). We see WWW as the preferred interface for systems in our database that have been well tested and have stabilized.

The users at present capture data in a myriad of ways—in Oracle Forms4 screens, in spreadsheets, flat files, non-relational databases, barcode readers—and will continue to

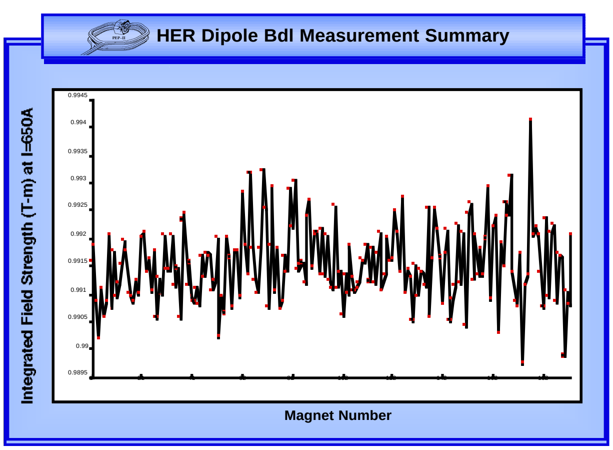


Fig. 4. Graph of Bdl measurements at 650 A for all HER Dipole Magnets from data retrieved by Clear Access and charted in Microsoft Excel.

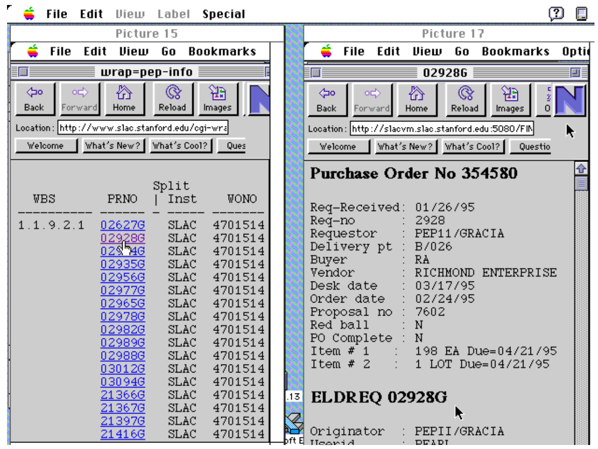


Fig. 5. World Wide Web interface to PEP-II Purchase Requisitions joining Oracle and legacy SPIRES data

do so. Often we work with them to record the data in this intermediate format until our Oracle tools catch up with the analysis and development so that we can reverse-engineer these sub-systems into the Project Database. The use of barcodes and barcode readers in facilitating data capture in PEP-II (from barcoding components to assigning barcodes to buildings) is spreading. In retrieving records from the database, similarly, a variety of interfaces are used and supported. An example of graphing data through Clear Access/Excel is shown in Fig. 4. Other tools used have been WWW, Oracle Forms4 and Oracle Reports.

VII. SUMMARY

During the last three years with a small team averaging two and a half full time employees per year, we have developed nine modules in the PEP-II Project Database. This short development time would not have been possible without the reliance on Oracle*CASE tools. The database is used by many of the ~300 PEP-II (and ~450 *BABAR*) collaborators from multiple computer platforms and sites. It is a useful tool to facilitate management and coordination of the three labs. It is also helping to maintain quality assurance during the construction of PEP-II. This integration of administrative and technical data is an innovative use within the accelerator community of a central project-wide database.

VIII. ACKNOWLEDGMENTS

We are grateful for the support and vision that Jonathan Dorfan, David Hitlin and Michael Zisman gave us in 1992 which launched this exciting enterprise-wide effort. We also thank the present PEP-II Project Management for their continued support, the Data Administrators of the different systems for helping with the design and for maintaining the integrity of their thousands of records, and our colleagues at CERN from whom we learned much.