

THE ROLE OF THE CEBAF ELEMENT DATABASE IN COMMISSIONING THE 12 GeV ACCELERATOR UPGRADE*

T. Larrieu, M. Joyce, M. Keesee, C. Slominski, D. Turner, JLAB, Newport News, VA 23606, U.S.A.

Abstract

The CEBAF Element Database (CED) was first developed in 2010 as a resource to support model-driven configuration of the Jefferson Lab Continuous Electron Beam Accelerator (CEBAF). Since that time, its uniquely flexible schema design, robust programming interface, and support for multiple concurrent versions has permitted it to evolve into a more broadly useful operational and control system tool.

The CED played a critical role before and during the 2013 startup and commissioning of CEBAF following its 18-month long shutdown and upgrade. Information in the CED about hardware components and their relations to one-another facilitated a thorough Hot Checkout process involving more than 18,000 system checks.

New software relies on the CED to generate EDM screens for operators on-demand thereby ensuring that the information on those screens is correct and up-to-date. The CED also continues to fulfil its original mission of supporting model-driven accelerator setup. Using the new ced2elegant and eDT (elegant Download Tool), accelerator physicists have proven able to compute and apply energy-dependent set points with greater efficiency than ever before.

BACKGROUND

The Jefferson Lab CEBAF accelerator is a superconducting recirculating linear accelerator capable of delivering continuous wave electron beams simultaneously to multiple experimental halls. Previously capable of delivering 6GeV, CEBAF recently underwent an extensive multi-year \$338M upgrade project to double its energy capacity to 12GeV and add a fourth experimental hall (fig. 1). The upgraded accelerator was commissioned successfully between November 2013 and May 2014. The CEBAF Element Database (CED) was an indispensable tool during the commissioning.

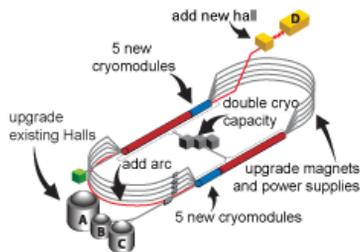


Figure 1: The CEBAF 12GeV Upgrade Project.

* Notice: Authored by Jefferson Science Associates, LLC under U.S. DOE Contract No. DE-AC05-06OR23177. The U.S. Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce this manuscript for U.S. Government purposes.

DATABASE DESIGN

The CED design is based upon a modified Entity-Attribute-Value with Classes and Relationships (EAV/CR) data model [1]. Whereas in a traditional schema adding support for new accelerator hardware would involve adding additional tables and columns to the database, the EAV/CR data model employed by the CED is introspective – defining a new class of accelerator hardware in the CED simply involves adding rows to the already-existing metadata “catalog” tables. Once defined in the catalog, existing software is fully capable of interacting with the new entities after discovering their properties from the metadata tables.

A major benefit of the EAV/CR data model is that its static schema is well suited to integration with Oracle Workspace Manager [2] to provide timestamp-based versioning of table rows, named save points, and multiple independent workspaces (akin to branches in a software version control system).

DATABASE IMPLEMENTATION

Platform

At Jefferson Lab, the CED is implemented using version 11g Standard Edition Oracle Relational Database. The database server runs Redhat Enterprise Linux 6 on x86_64 hardware. The current API library is compiled solely for 32-bit x86 architecture, though prior versions were also compiled for the Solaris 10 Sparc and HP-UX PA-RISC platforms.

Layout

To optimize performance for the different use cases, the CED database instance is distributed among three database users/schemas as shown in Figure 2. The operational schema stores the current machine configuration and is optimized for performance. The historical schema provides efficient storage and access to (read-only) historical save points (snapshots). And the development schema is used to create workspace branches where data can be edited and prepared before being promoted to the operational schema.

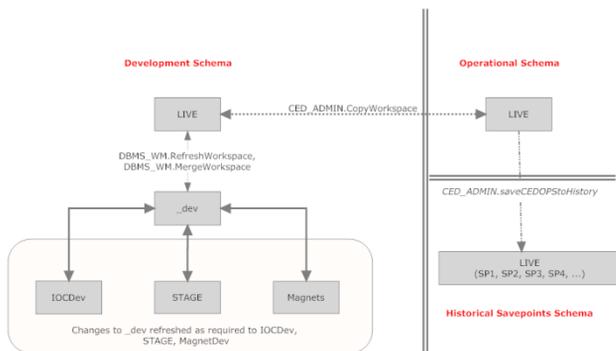


Figure 2: The CED three schema layout.

The details of the three schema layout are transparent to non-administrative users who need simply include an optional workspace or savepoint name when interacting with the database.

Configuration Control

As a rule, users do not directly edit data in the operational schema. Instead, updates are prepared in a development workspace and merged into OPS upon request. A database administrator receiving a merge request will audit the proposed changes for validity and then promote the changes to the operational schema. Administrators make use of a web-based management tool to automate the auditing and merging process as well as the creation of new savepoints.

API

The generalized nature of the EAV/CR data model employed by CED requires a robust API to interpret the contents of the metadata tables and translate the abstract database storage into recognizable attributes useful to programs and users. The API is also responsible for enforcing much of the data validation and domain logic embodied in the metadata catalog.

For the CED, a shared library was written in C++ to be the sole intermediary for applications that will access its information. Native versions of the API are available not only to C++ programs, but also to scripts written in Perl, PHP, and Tcl. The script language versions are generated automatically from the original library via the open source SWIG (Simplified Wrapper and Interface Generator) tool [3,4].

On top of the core API, two general purpose user interfaces to the CED have also been built around the C++ library and its scripting language derivatives: a RESTful web interface and a full-featured command-line tool usable from JLAB Linux workstations.

CED Event Server

The CED Event Server provides a means to link control system actions to database updates. Whenever rows are modified in the operational schema, a database trigger notifies the event server of the changes. Based upon rules in its configuration file, the event server can take actions including: telling long-running daemon processes to refresh their configurations, sending email to system

owners, making logbook entries, or calling the ced2Epics utility to update pvs in the control system. The CED Event Server will even notify users if an auto-generated screen on their desktop needs to be refreshed because of a database update.

HOT CHECKOUT

Readiness

The immense scope of the work completed during the 18 months the lab was shut down necessitated a thorough system whereby all new, refurbished, and even supposedly unmodified systems could be verified as ready to be operated safely.

The CED was used as the underpinning for a web-based Hot Checkout (HCO) tool [5]. This tool tracked HCO readiness status on a daily basis in the months leading up commencement of beam operations. Geographic information from the CED granted users of the HCO tool an ability to determine readiness status region by region.

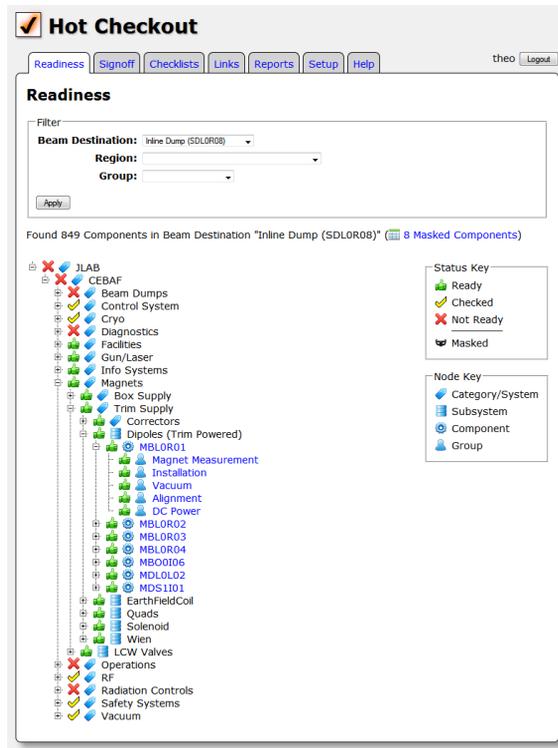


Figure 3: Web-based Hot Checkout application.

CED-linked Hardware Labels

New hardware labelling was devised that utilizes the CED to provide detail that can't fit on the label itself. In addition to the key details such name, weight, power supply, etc., the new labels contain a web QR code that can be scanned with camera-equipped tablets or smart phones to call a CED-linked web page. For example, the web page accessed from a Dipole label (see figure 4) would allow the user to find out which other nearby

hardware shares the same cooling water valves or power supply.



Figure 4: Hardware label with QR code linked to the CED.

EDM SCREEN GENERATION

The display manager used in the CEBAF control room is EDM (Extensible Display Manager). For many large hardware systems, including magnets and vacuum, the process of building and maintaining screens by hand has been replaced by programmatic screen generation. A large percentage of the screens used by CEBAF operators no longer exist as static *.edl* files, but are generated on-the-fly using data from the CED. This process liberates EPICS programmers from the burden of creating screens for new device instances and also ensures that the screens always match the current operational configuration.

In addition to the screens that are placed on menus for operators, a command line tool permits users to create task specific on-the-fly screens for many systems based on CED filters. For instance users may request screens that show non-contiguous regions or that group elements in a non-standard way (e.g. by power supply, by rack, by building, by controlling IOC, etc.).

ELEGANT DOWNLOAD TOOL

The ced2Elegant and Elegant Download Tool (eDT) software tools were developed at JLab as the mechanism whereby the design set points stored in the CED can be used to perform accelerator set up. First, the ced2Elegant utility is executed to retrieve design BdL values from the CED and produce an Elegant [6] lattice. Then eDT is rung to calculate new energy-specific BdL values by using Elegant to compute the momentum at each element in the lattice. At the end of the process, a BURT "snap file" containing the BdL values to be sent to the control system is written out. This process is faster and more robust than the previous method of manual scaling of an historical machine save followed by substantial operator tuning in order to achieve a stable orbit. The database-driven approach also lends itself to ongoing improvement – for example, a recent enhancement to eDT compensates for synchrotron radiation losses that occur at higher energy configurations.

ISBN 978-3-95450-146-5

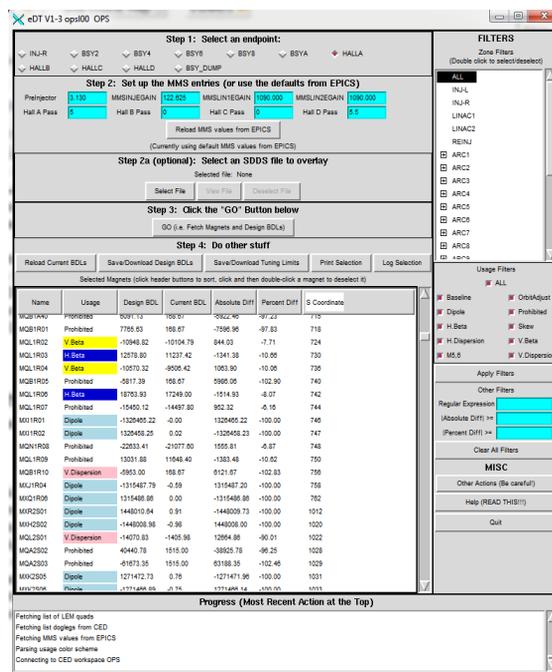


Figure 5: The Elegant Download Tool uses design data from the CED to generate downloadable energy-specific machine settings.

CONCLUSION

The CED has proven its utility during installation and commissioning of the CEBAF 12GeV upgrade project. Its flexibility and the growing suite of tools that have been built upon it ensure that it will play a central role in efficient and safe future operation of the upgraded accelerator.

REFERENCES

- [1] Nadkarni, MD, Prakash M.; Marengo, MD, Luis; Chen, MD, Roland; Skoufos, PhD, Emmanouil; Shepherd, MD, DPhil, Gordon; Miller, MD, PhD, Perry (1999), "Organization of Heterogeneous Scientific Data Using the EAV/CR Representation", Journal of the American Medical Informatics Association 6 (6): 478–493, PMID 10579606.
- [2] <http://www.oracle.com/technetwork/database/twp-appdev-workspace-manager-11g-128289.pdf>
- [3] Beazley, David M (1996), "SWIG : An Easy to Use Tool For Integrating Scripting Languages with C and C++", Proceedings of the Fourth Annual Tcl/Tk workshop, Monterey, CA July 6-10.
- [4] <http://www.swig.org/>.
- [5] Baggett, K. (2014, October), "New and Improved": The JLab (state-of-the-art) HCO System, WAO2014, Workshop on Accelerator Operations in Mainz, Germany, 2014.
- [6] http://www.aps.anl.gov/Accelerator_Systems_Division/Accelerator_Operations_Physics/elegant.html