



Proteus: FRIB Configuration Database

<http://discs.openepics.org>



Vasu Vuppala, Eric Berryman, Sheng Peng

FRIB-NSCL, East Lansing, USA

Bob Dalesio

Brookhaven National Laboratory, Long Island, USA

Introduction

There is need for an integrated information system that manages the data and computational-logic used by an experimental physics facility (EPF) during its design, construction, commissioning, and operation. Such a system can be used to manage design lattices, model them, run what-if scenarios, tune the beams, troubleshoot, manage calibration data, maintenance records, alignment information and quality metrics, and generate reports for funding or regulatory agencies. A critical component of such a system is the configuration database. It manages devices, their layout, measurements, alignment, calibration, signals, and inventory. Proteus is an implementation of such a component. It is being developed and used at Facility for Rare Isotope Beam (FRIB).

Components

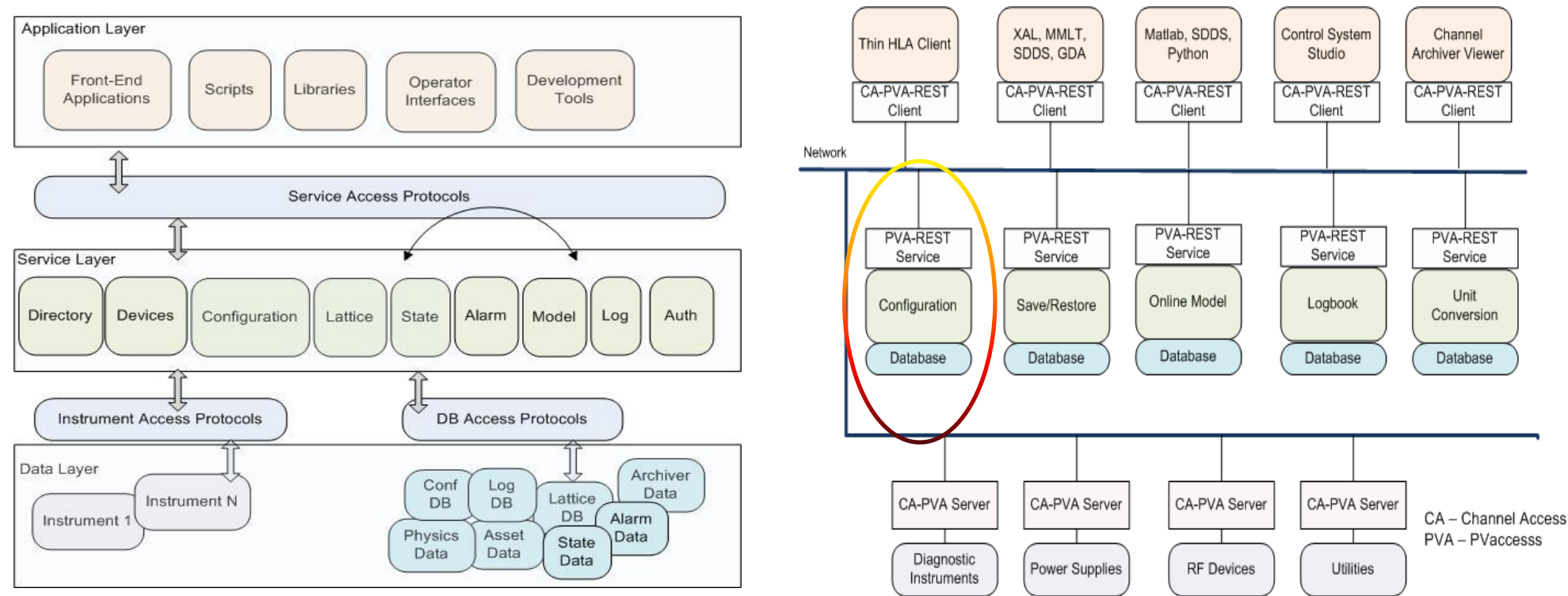
A component is any entity in the accelerator facility's configuration: magnet, power supply, cavity, rack, room, controller etc. Components can be looked at in different ways, and have different kinds of information associated with them: design data, measurements, test data, alignment information, physical characteristics etc. We define two kinds of components:

- **Physical-Component:** This represents physical entities; things that exist in the real world. A physical-component has attributes such as measurements, calibration, traveler data, manufacturer model etc
- **Logical-Component:** It represents the entities that exist on the blueprint or configuration (layout) of the Accelerator facility.

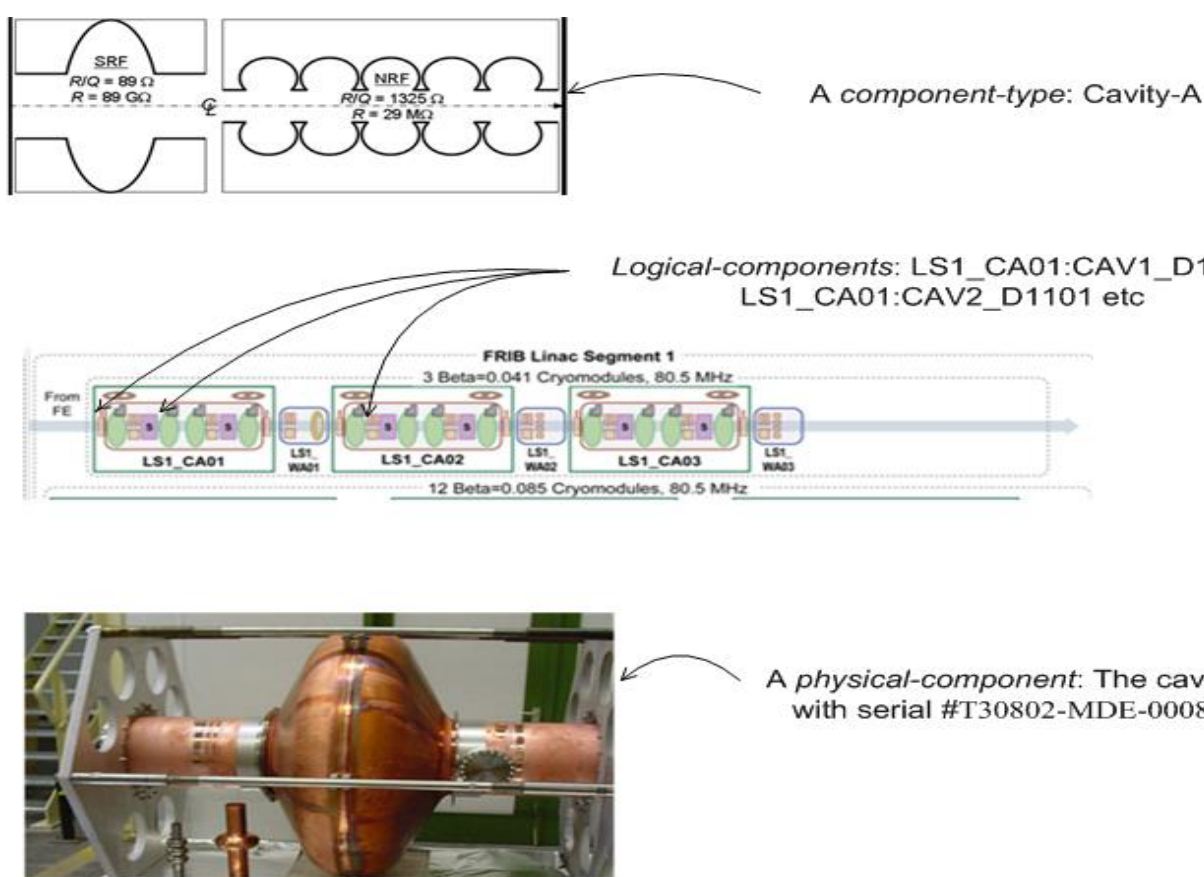
Architecture

The basic architecture for Proteus is shown in the figure to the right. It consists of three layers: Data, Service, and Application. Data Layer represents all the data sources: managed, unmanaged, structured, and unstructured. Service Layer is composed of services. A service is a reusable software component that implements a set of business functions, has a formal and documented interface, and can be located and accessed through standards-based communication mechanisms. In our case, a service can be thought of as a software process that implements controls or physics related logic, and provides high-level data structures to the user through REST and PVaccess protocols.

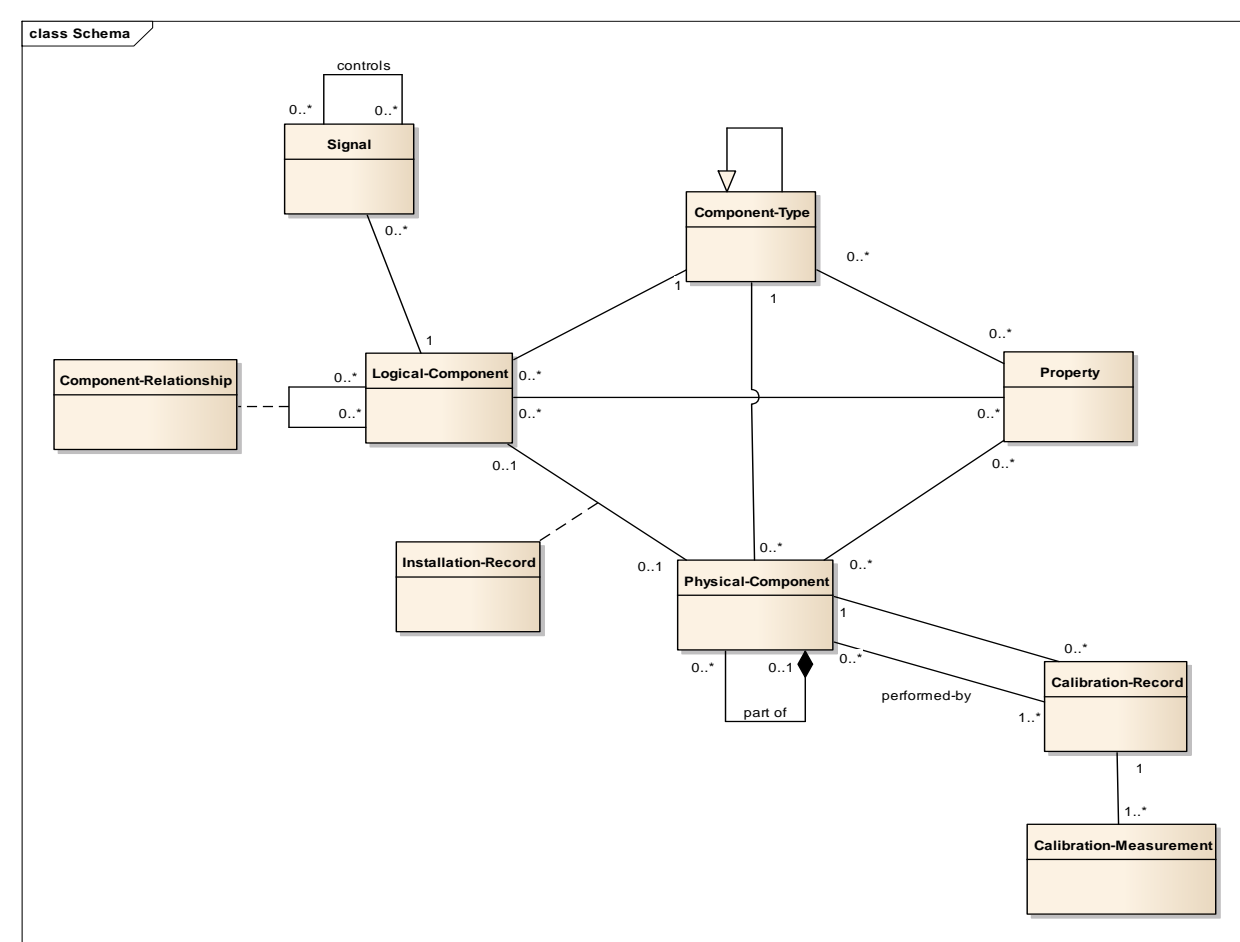
EPICS V4



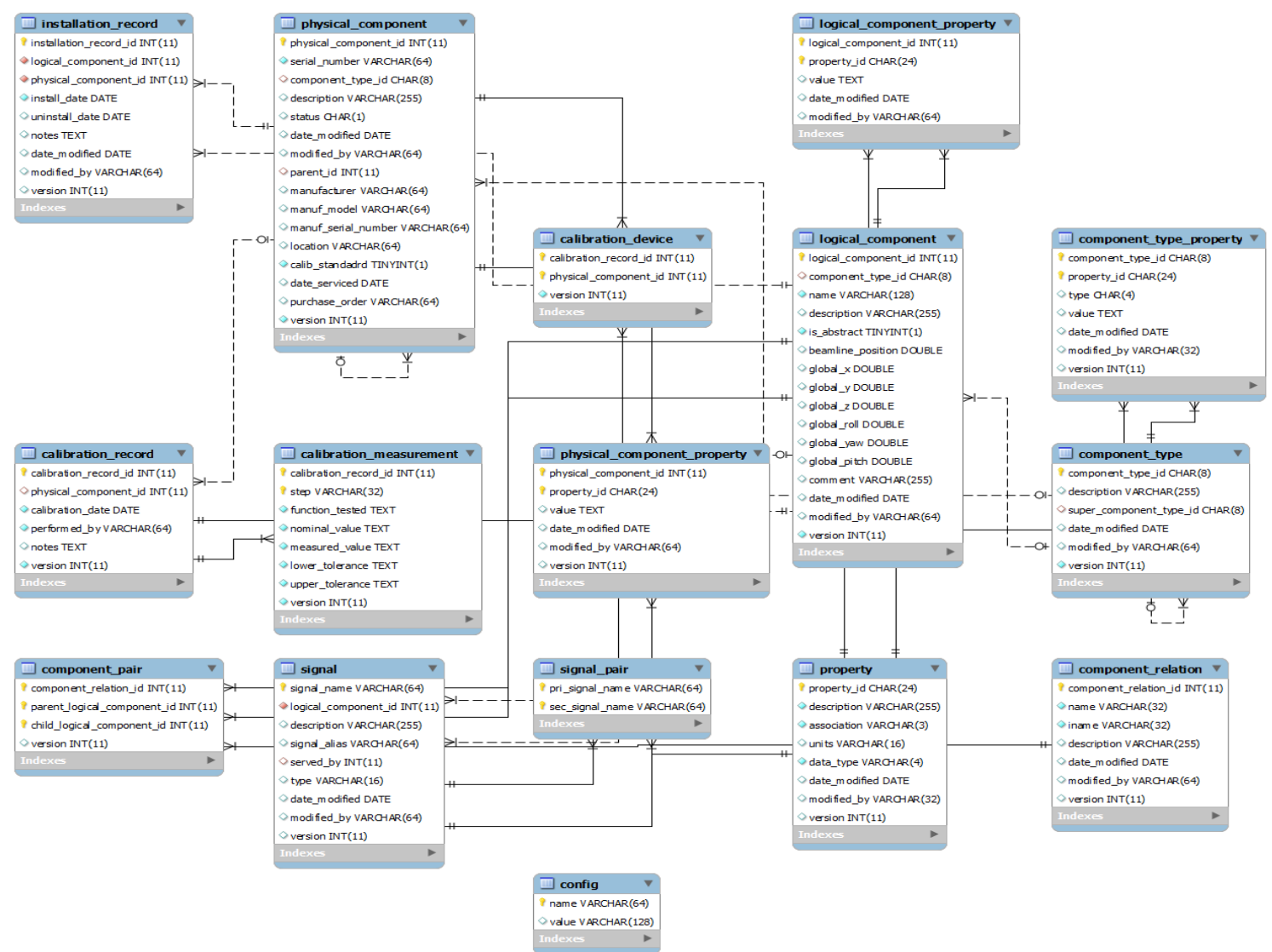
Conceptual Model



Logical Model

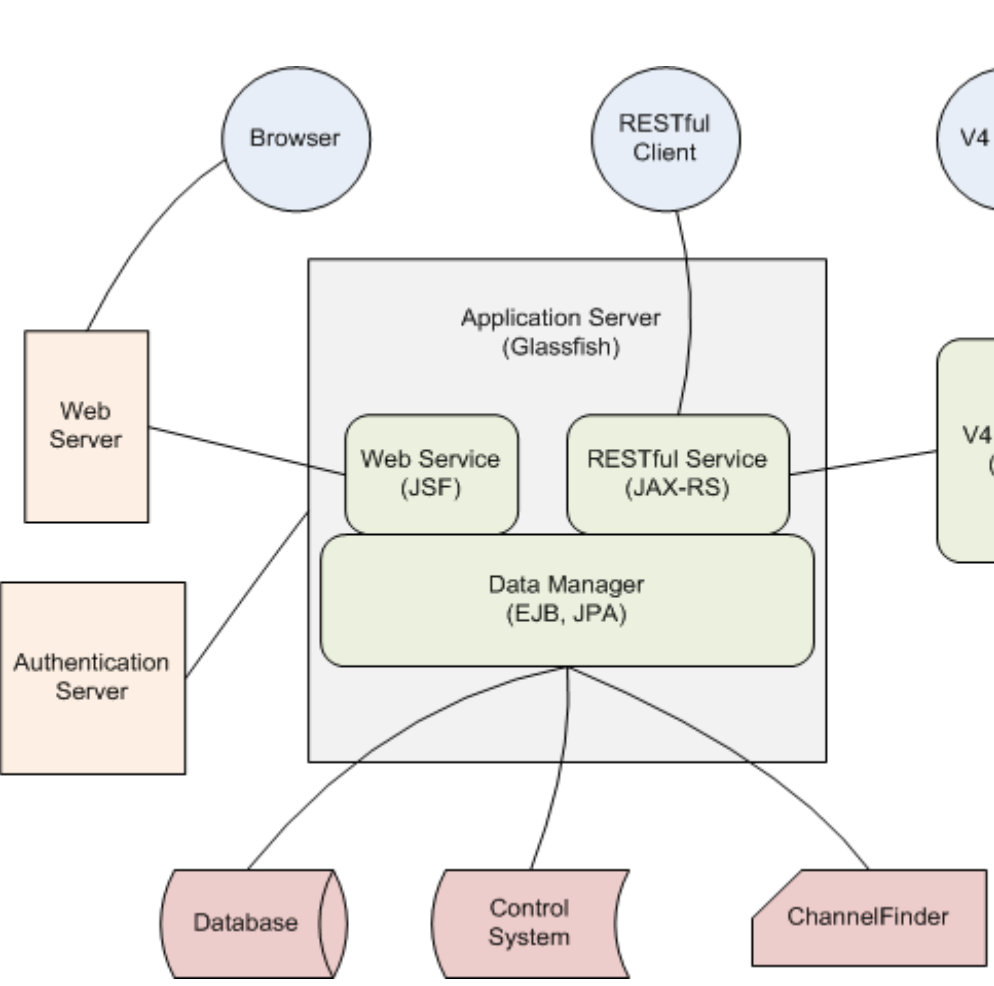


Physical Model

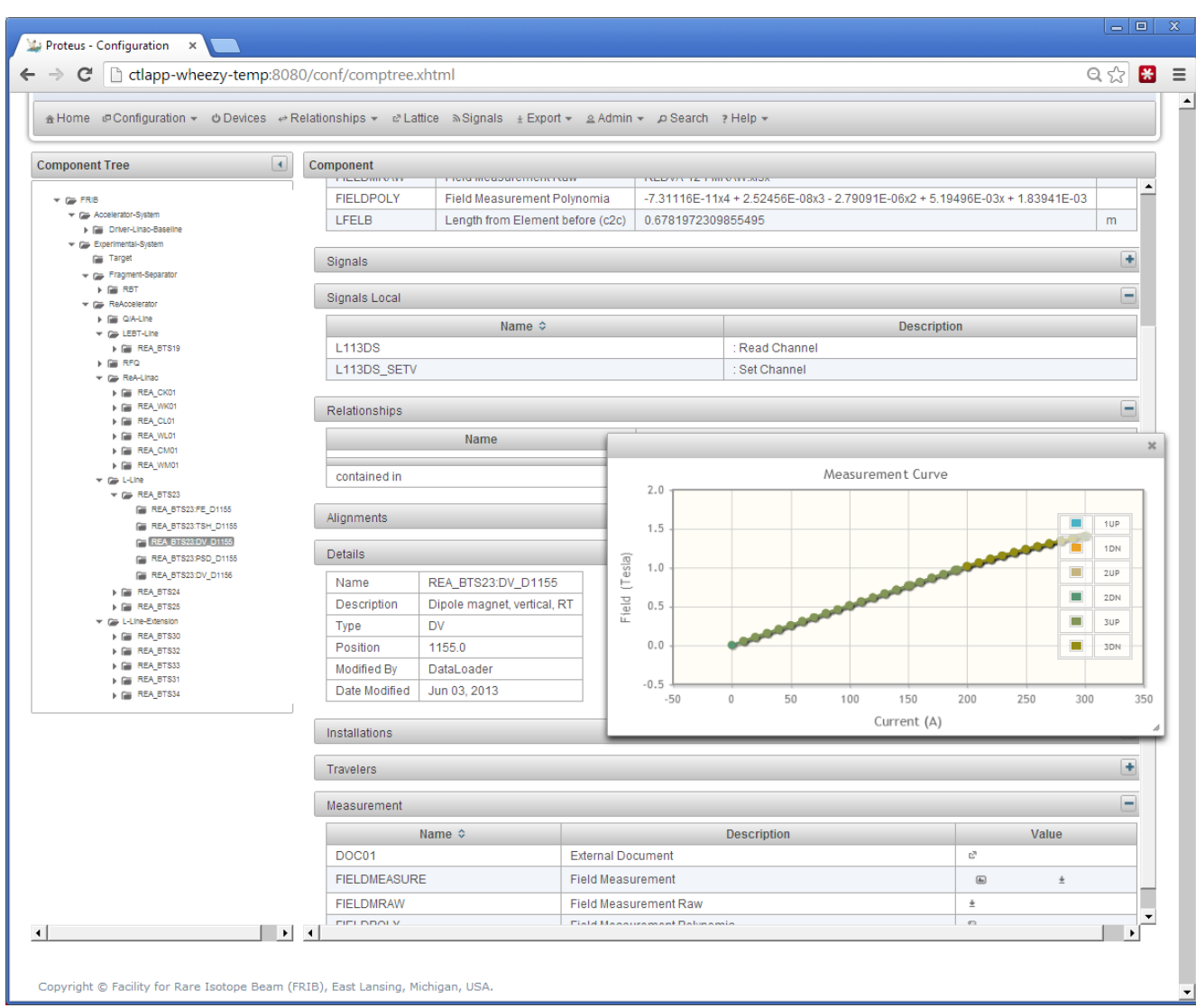


Implementation

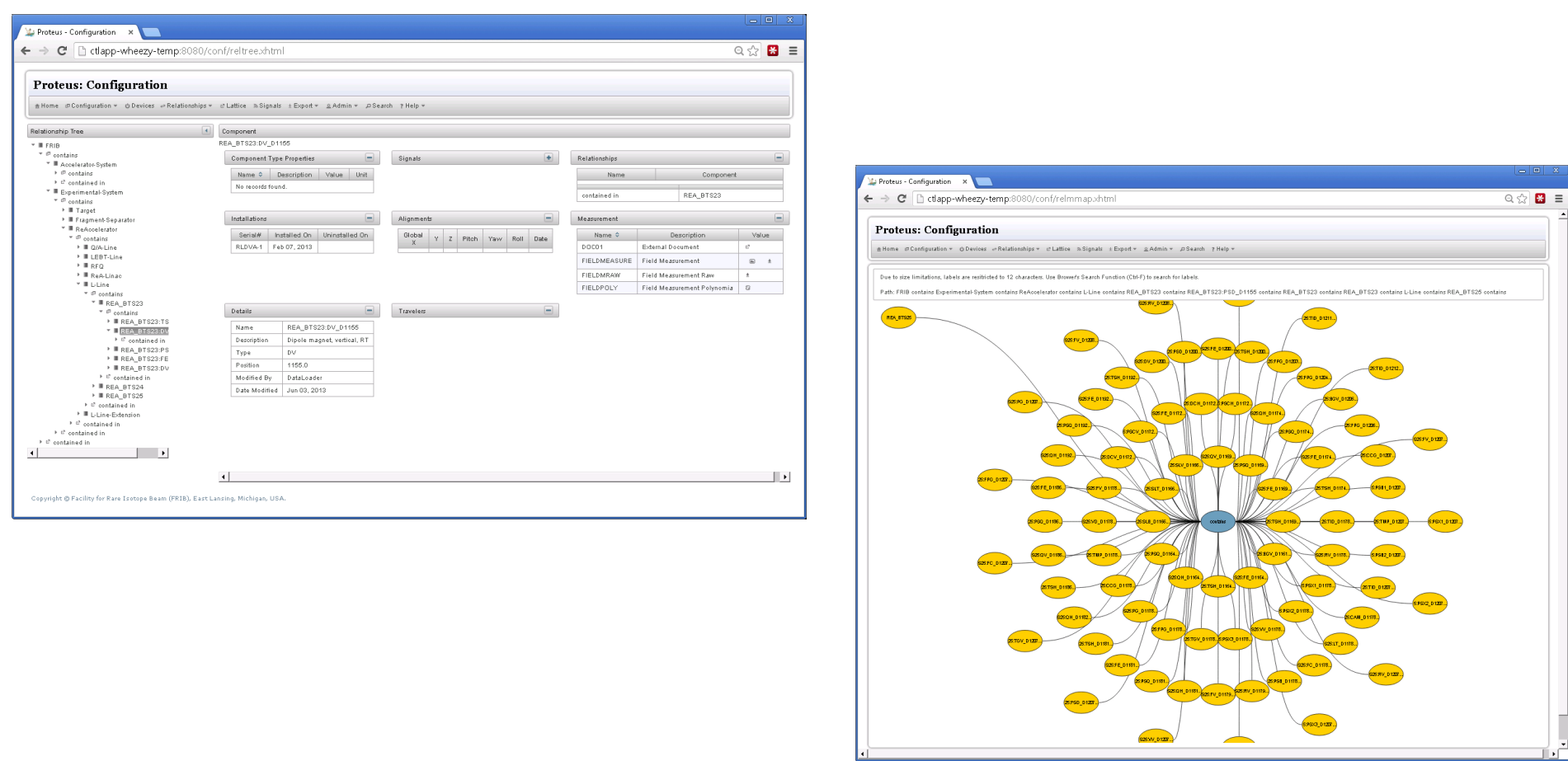
- DBMS: MySQL
- Framework: Java EE
- Service: EPICS V4 and RESTful
- GUI: JSF and Primefaces
- App Server: Glassfish



Component Tree



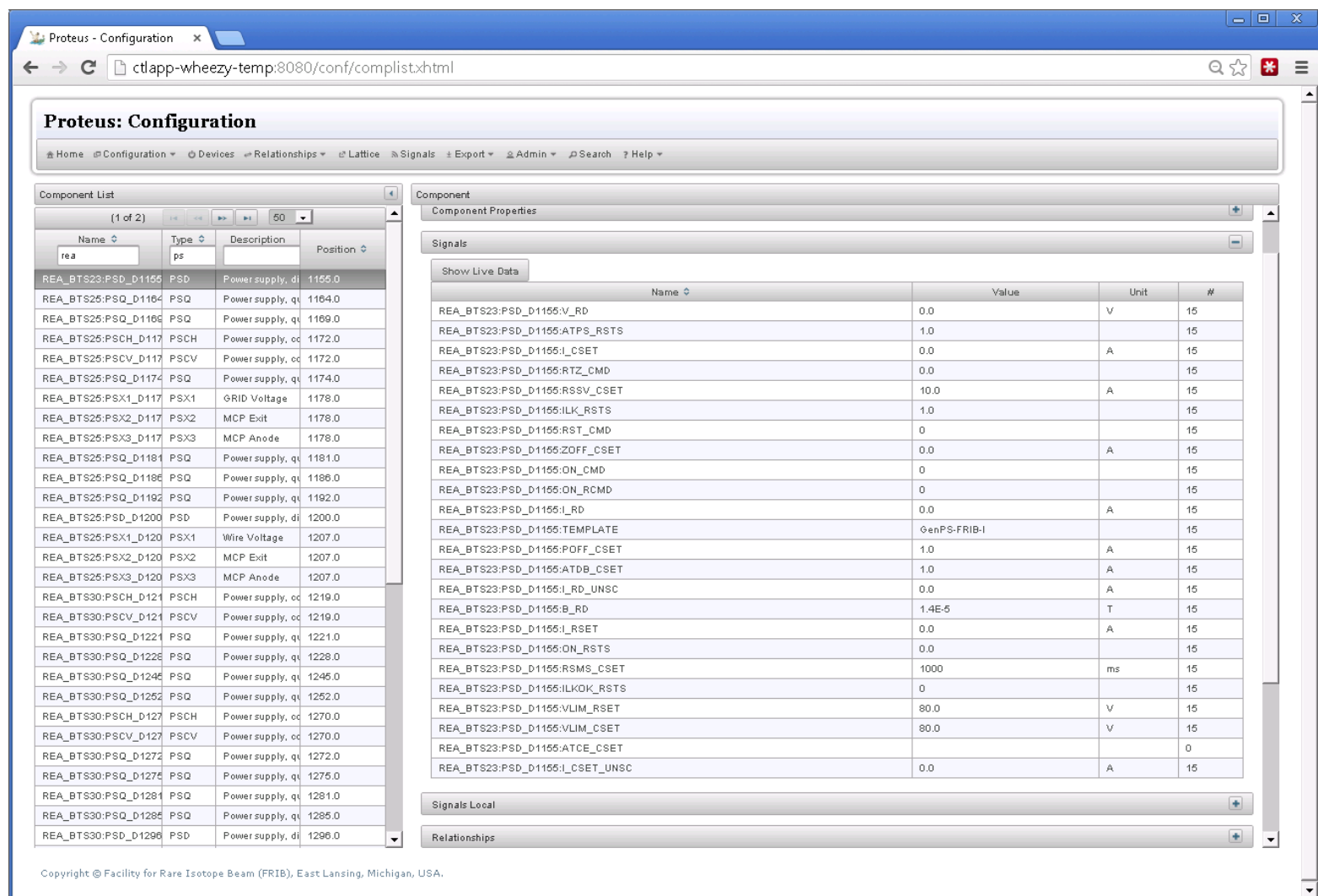
Component Relationships



EPICS V4 Service



Live Control Signals



This material is based upon work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661, the State of Michigan and Michigan State University. Michigan State University designs and establishes FRIB as a DOE Office of Science National User Facility in support of the mission of the Office of Nuclear Physics.