Functional Requirements for the Hardware Database System

Maxim Potekhin

DUNE DB Meeting
10/16/2019

A bit of history

- FNAL Hardware DB was used in the construction of two subsystems in protoDUNE-SP (Norm, Flor, other exerts)
- System for CE QC/testing at BNL
  - DAQ+DB
- Norm has recently presented the *“DUNE Component/QC/Installation (Hardware) Database Requirements”*
- It's a helpful document with a focus on architecture/organization/SLA etc
- There is also a proposal to review functional requirements for the Hardware DB as suggested by Marco Verzocchi and others
- cf. Marco's talk at the Jan'19 Collaboration meeting: https://indico.fnal.gov/event/16764/session/8/contribution/40/material/slides/0.pdf
- re: Hajime's proposal (see the link above) and recent work (mobile, connected tech etc) https://indico.fnal.gov/event/21445/session/11/contribution/159/material/slides/0.pdf

Why do we need to discuss the Hardware Database?

- Optimally, we should be addressing all aspects of a component and aggregate lifecycle
  - ✓ manufacturing
  - ✓ QC + other testing
  - ✓ assembly
  - ✓ logistics
  - ✓ installation
  - ✓ documentation
  - ✓ calibration
  - ✓ testing "in situ"
  - ✓ maintenance and component replacement
  - ✓ cabling
  - ✓ traceability (i.e. keeping references to relevant docs, technical drawings etc)
  - ✓ ...
- It appears that solutions currently employed have partial coverage of items above
- No common approach across DUNE currently exists

What is optimal?

- It is acknowledged that a well thought-out design with a focus on integration and lifecycle support has the potential to reduce maintenance effort over the projected long life span of the experiment
- It is also acknowledged that this would require substantial investment of effort upfront
- The alternative is to “start local” and worry about integration later, with potentially high maintenance effort
- Current shortage of manpower skews our decision process towards the latter
- Choices need to be made in order to define deliverables and the project timeline

Today's discussion

- Let's take a step back and see whether we can identify functional requirements for the DB to support the component lifecycle
- In doing that, let's look at our recent experience, what works and what doesn't
- Some basic requirements have been summarized in DocDB 16619
  - by no means complete or all-inclusive but mainly to start a discussion
  - expect iteration and broader contributions
- Determine if we can agree on some or all of the requirements
- See what we can do in terms of cross-consortia communication
- What steps forward can be taken

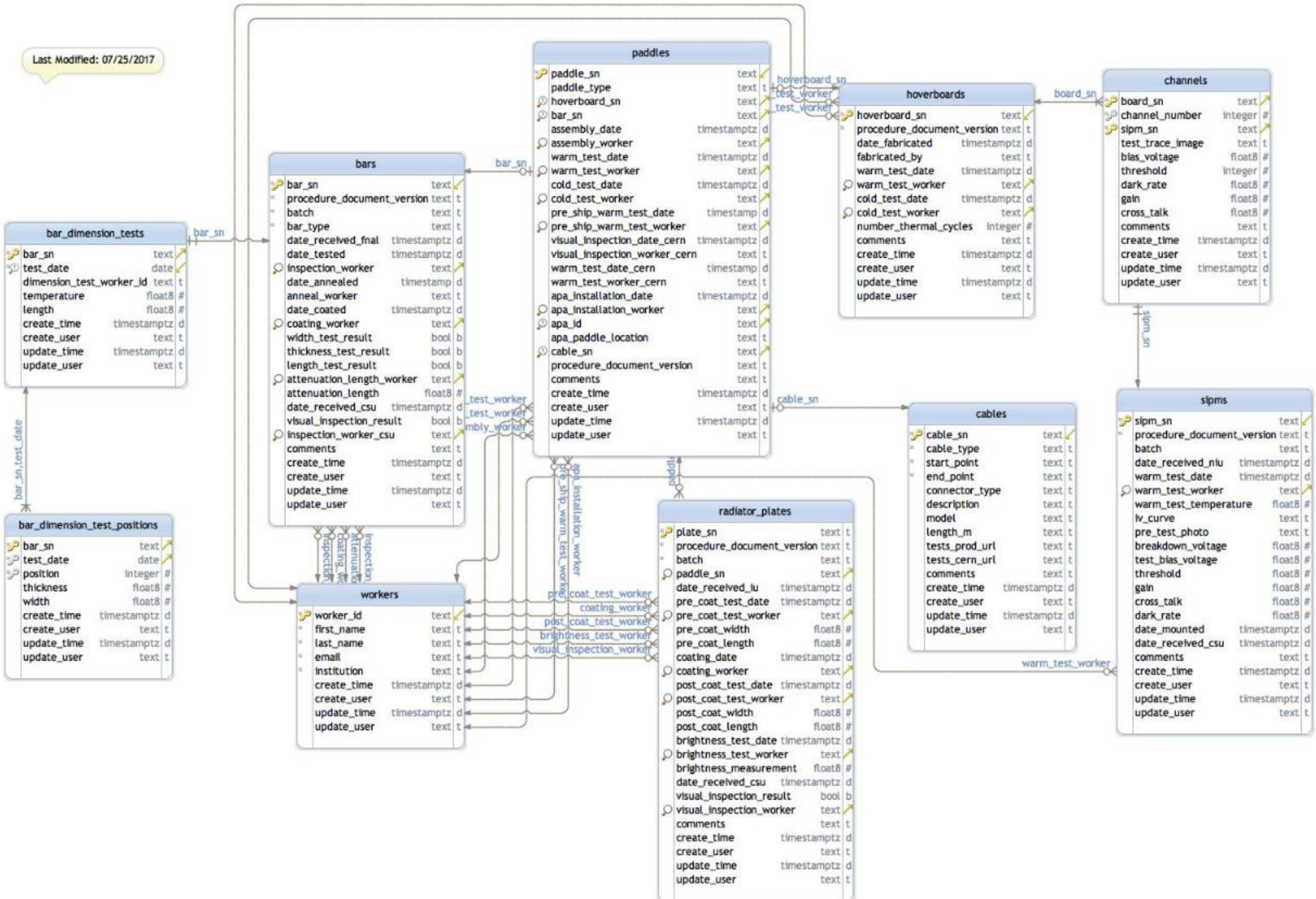
FNAL Hardware DB

- Codifies some very informative use cases, and a good starting point to generalize the requirements
- Used in a few experiments including protoDUNE-SP

FNAL Hardware DB example - PDS paddles - schemas

proto_dune.jpg

Stephen White, 08/23/2017 01:18 PM



FNAL Hardware DB example - PDS paddles - Web UI

protoDune Hardware Database

DEEP UNDERGROUND NEUTRINO EXPERIMENT

Home Permission Mgr Admin Mgr Log out

Table paddles

dump to CSV plot... search data config data <prev next>

Sorting by: paddle_sn desc

	Paddle Sn	Paddle Type	Hoverboard Sn	Bar Sn	Assembly Date	Assembly Worker	Warm Test Date	Warm Test Worker	Cold Test Date	Cold Test Worker	Pre Ship Warm Test Date	Pre Ship Warm Test Worker	Visual Inspection Date Cern	Visual Inspection Worker Cern	Warm Test Date Cern	Warm Test Worker Cern	Apa Installation Date	Apa Installation Worker	Apa Id
view/edit	IU9	IU	HAM-002-0020	IU9	03/16/2018 09:00:00-0500	ap_csu01	03/20/2018 13:23:00-0500	ac_csu01											
view/edit	IU55	IU	HAM-002-0056	IU55	02/14/2018 14:30:00-0600	zr_csu01	03/12/2018 09:07:00-0500	ap_csu01											
view/edit	IU54	IU	HAM-002-0008	IU54	02/28/2018 13:50:00-0600	zr_csu01	03/09/2018 09:54:00-0600	ac_csu01											
view/edit	IU53	IU	403-002-0023	IU53	12/19/2017 13:45:00-0600	zr_csu01	12/20/2017 17:02:00-0600	ap_csu01											
view/edit	IU52	IU	403-002-0042	IU52	11/21/2017 14:00:00-0600	zr_csu01	11/29/2017 13:38:00-0600	ap_csu01	12/05/2017 00:00:00-0600	ap_csu01	12/07/2017 11:21:00	ap_csu01							
view/edit	IU51	IU	403-001-0039	IU51	11/21/2017 16:00:00-0600	zr_csu01	11/29/2017 13:43:00-0600	ap_csu01	12/05/2017 00:00:00-0600	ap_csu01	12/08/2017 12:04:00	ap_csu01							
view/edit	IU50	IU	403-002-0044	IU50	12/01/2017 14:00:00-0600	zr_csu01	12/20/2017 17:07:00-0600	ap_csu01											
view/edit	IU49	IU	403-001-0006	IU49	06/01/2017 00:00:00-0500	jd_csu01	06/12/2017 00:00:00-0500	jd_csu01	07/09/2017 00:00:00-0500	jd_csu01		jd_csu01	08/18/2017 00:00:00-0500	en_fnal01	08/18/2017 00:00:00	en_fnal01	08/26/2017 00:00:00-0500	tr_umn01	1
view/edit	IU48	IU	403-003-0004	IU48	02/12/2018 15:00:00-0600	zr_csu01	03/14/2018 18:01:00-0500	ac_csu01											
view/edit	IU47	IU		IU47	02/26/2018 13:15:00-0600	zr_csu01													
view/edit	IU37	IU	403-003-0011	IU37	02/07/2018 17:30:00-0600	zr_csu01	02/08/2018 14:07:00-0600	ac_csu01											
view/edit	IU35	IU	403-002-0038	IU35		zr_csu01	02/08/2018 15:22:00-0600	ac_csu01											
view/edit	IU31	IU	403-002-0020	IU31		zr_csu01	10/27/2017 10:53:00-0500	ap_csu01	11/01/2017 00:00:00-0500	ac_csu01	11/06/2017 09:12:00	ap_csu01							
view/edit	IU28	IU	403-003-0063	IU28	06/02/2017 00:00:00-0500	jd_csu01	06/12/2017 12:50:00-0500	jd_csu01	07/08/2017 00:00:00-0500	jd_csu01	07/12/2017 00:00:00	jd_csu01	08/22/2017 00:00:00-0500	en_fnal01	08/22/2017 00:00:00	en_fnal01	08/26/2017 00:00:00-0500	tr_umn01	1
view/edit	IU27	IU	403-003-0002	IU27	02/07/2018 13:00:00-0600	zr_csu01	02/08/2018 16:40:00-0600	ac_csu01											

Observations

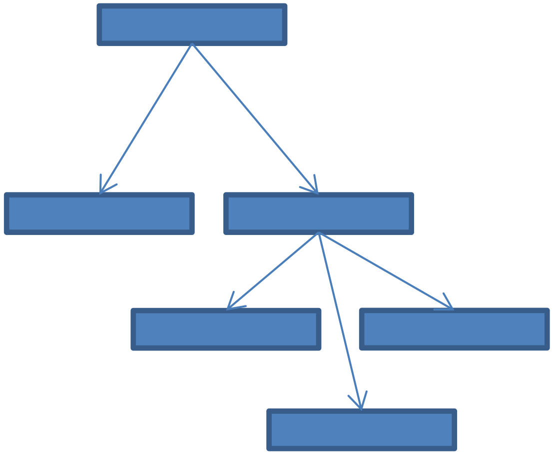
- Disclaimer - please correct me if I didn't get something right!
- Hierarchical model
- Components and Aggregations related via foreign keys
- Tables are tailored for specific types of elements (down to the name)
- Some schemas normalized, however there are cases of schema de-normalization
  - cf. test metrics, manufacturing (down to sequence of procedures), some logistics detail present in same table
- Changes of the process and/or logistics will require schema changes

Generalized Model of the Apparatus

- A few definitions needed to better define the requirements
- The model includes
  - Components and Aggregations
  - States and Transitions

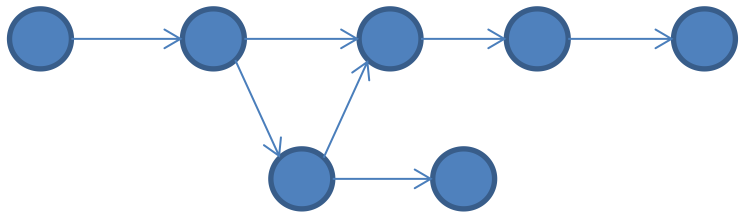
Components and Aggregations

- Consider the apparatus as an hierarchy of elements
  - e.g. model it as a graph
- cf. a FEMB can be modeled as an aggregation which contains various electronics components e.g. ASICS, and can be decsribed as a graph
- Most components are aggregations themselves, although in graph terms, the leaves of the tree are atomic by definition
- In the following we assume that a component may be either atomic or an aggregation of other components subject to the same assumption



States and Transitions

- Each component can exist in a variety of states
- The chain of events starting from design/ordering/manufacture (or procurement) and then on to QC, shipping, installation, testing "in situ" etc is reflected by the changing state of the component as recorded in the database
- The sequence of transitions depends on the type of the specific component
- Each state transition may optionally result in a data product specific to the component and the specific transition
  - *can be recorded as an external reference*
- Each state transition may optionally require a record in the audit trail e.g.the time stamp, the reason for the event any comments required in the process and the entity responsible for changing the state and updating the record



Evolution of Aggregate Components

- Experience shows that the hardware may need to be swapped, updated
- Consider the use case where a particular component within an aggregation needs to be replaced, such as a failed ASIC installed on a board. The existing database ID can be preserved and the version number updated, or the the object can be retired and recreated under new ID
- The choice will depend on the situation or resolved later in the design stage
- This approach allows to handle an important use case whereby a component is redesigned in terms of its construction and composition while retaining its functionality and place in the overall hierarchy.

Component Location and Ownership

- We will use the concept of the *functional position* introduced in ATLAS, which refers to a physical position of a component specific to its function and which cannot be move
- It is complementary to the component *geographic location*
- Ownership refers to the entity who is assigned custody of the item and is authorized to perform specific actions relate to the item
- The geographical location, functional position and ownership of an item are an integral part of its state

Functional Requirements (abridged, see the paper for details)

- The Unique ID Requirement (not identical to the manufacturer's and other ad hoc IDs)
- Traceability - identify component in technical documentation/drawings etc
- Completeness - support the full lifecycle
- Record of state transitions
  - History
  - Periods of vailidity
  - Evolving documentation and metrics after deployment
- Record of Connectivity
  - physical and electrical connections
  - History
- Interfaces
  - Network and Web
  - Component Lifecycle Management e.g. links to the Conditions/Slow Controls etc
  - Role-based Auth/Auth
  - Industrial ID Systems and Mobile Platforms

Reusability

- It is foreseen that any design or adaptation of a system that meets the functional requirements listed above will require a substantial investment of effort and resources
- We propose to state a collateral requirement that the system used in DUNE should be experiment-agnostic in terms of its architecture, and be adaptable to other projects
- This will help to assure a more efficient use of resources and may serve to attract collaboration from other experiments as a part of a cross-cutting effort

The Timeline

- Near and Medium Term
  - Preparations for the protoDUNE-2 experiment, with manufacturing of components to commence in 2020, and installation scheduled for 2021
  - Early manufacturing of certain components of the DUNE experiment
- Longer term, including manufacturing and installation of the DUNE detector, commissioning and operation
- We can't create an optimal solution complete with lifecycle management right away
- “Bridge the gap”:
  - Survey and catalog the interim systems (e.g.existing and in use at participating sites) which are likely to be used in near and medium term
    - not just at FNAL but elsewhere
  - Identify additional attributes (if any) to be added to object models in these systems and modest changes in functionality to allow for eventual migration to the final system
  - Define formats for data export and ingestion necessary for such migration as well as future I/O and data exchange protocols

Options

- We won't go into discussions about design specifics today
- In terms of reuse let's look at the non-exclusive list of existing options (perhaps as candidates for additional development)

FNAL Hardware DB

- Pros:
  - Existing, proven system, experience in DUNE
  - Supports many of the existing requirements as stated above
- Cons:
  - A somewhat rigid schema not well suited for evolution
  - Doesn't meet all requirements, does not support the component lifecycle
- Can be the basis for future development and added features

ATLAS Equipment Database

- Pros:
  - A full featured system covering many complex use cases
  - Likely supports most and perhaps all of the DUNE requirements
  - Proven during construction, installation and operation of a very complex detector
  - Integration of a number of separate databases (using the Glance system)
- Cons:
  - Substantially based on a proprietory commercial system, entailing the licensing and lifecycle issues
  - Has many hooks into the vast CERN information infrastructure which would be hard to disentangle in the reuse scenario
  - Considerable amount of work was required to tool the database for specific ATLAS needs, this part won’t be immediately reusable in any case
- Need to take a further look

Proposed Action Items

- Conduct a survey of the DUNE Consortia and computing groups in order to document use cases and existing systems and practices
- Determine what can be done in the interim to bridge the timeline gap
- Conduct a survey of the DUNE Consortia and computing groups in order to agree on a common set of requirements for the database
- Evaluate design and reuse options
- Begin discussions with Lab and university experts of an appropriate system for the full life-cycle of DUNE
- Motivations:
  - to avoid duplication of effort
  - to prevent divergent and/or incompatible designs in each sybsystem area and avoid operational risks associated with that
  - lifecycle support for the DUNE components (cf. QC, logistics and installation)