

# Parts Identifier and the Hardware DB

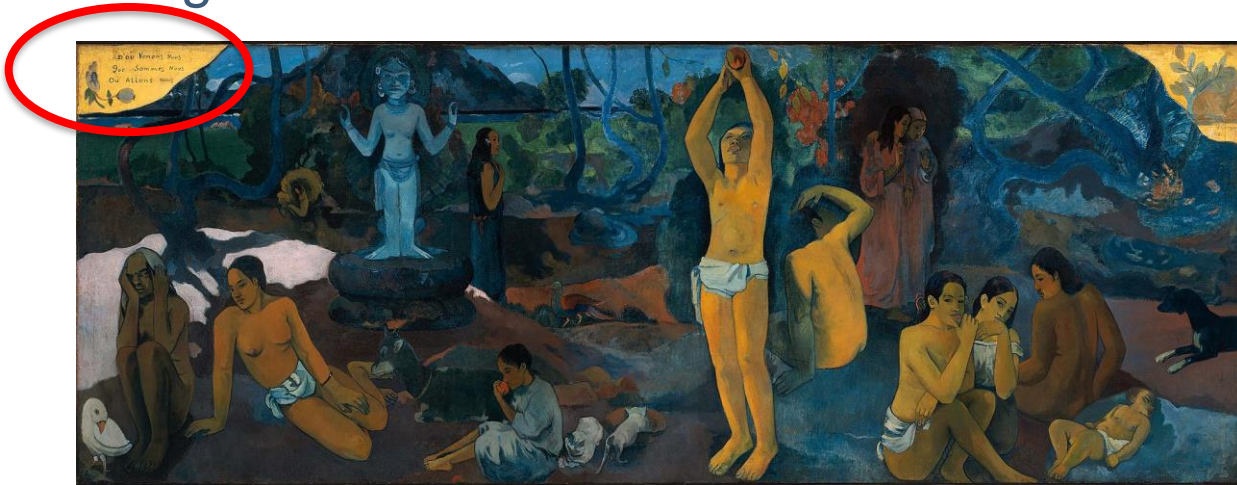
Marco Verzocchi

Fermilab

Date

# Outline

- Reminder of the parts identifier definition / DB interface
- Need for a physicist who will take over the responsibility for the parts identifier
- Interface between hardware DB and parts identifier DB
- Where are we ?
- Where do we go ?



# The parts identifier (i)

- It took us ~1 year to come to an agreement on a unique parts identifier for LBNF/DUNE
- Documented in [EDMS 2505353](#)

D/I/L/P	01-99	001-999	0001-FFFF	-	0001-FFFF	-	AA-ZZ	001-999	-	00-99	00-99	001-999
Project	System ID	Subsystem ID	Item Type ID	Dash	Item Number	Dash	Country of Origin	Responsible Institution ID	Dash	Detector ID	Final Destination	Intermediate Destination
F	F	F	F		F		F	F		M	M	M

- First five fields are non-mutable, make up unique identifier used in the HWDB
- Document has not been updated since March
- Current interface at <https://dune-pbs-id-gen.web.cern.ch/dune-pbs-id-gen/index.php>
- Parts identifier needs to be a team effort involving dedicated personnel plus the consortia

# The parts identifier (ii)

- Ownership of various fields
  - Project – Joint Project Office / Project Director
  - System ID – “Physicist Responsible for Parts Identifier”
  - Subsystem ID, Item Type ID, Item Number – group / consortium
  - Mutable fields – “Physicist Responsible for Parts Identifier”
- Current problems
  - Who is the “Physicist Responsible for Parts Identifier” ?
  - HV consortium requested more System IDs, making sure we cover FD2 (and ND)
  - No entries in the database at CERN (also, I do not know whether the problems in previous implementation that were pointed out earlier this year have been fixed)

# Physicist in charge of Parts ID (i)

- The definition of three fields (subsystem ID, item type ID, item number) are a responsibility of the individual consortia. The consortia are responsible for
  - maintaining the definition of the fields (including the documentation)
  - providing instructions to the computing expert(s) on how to change the programs that encode/decode the parts identifier
- Job description for scientist in charge of the parts identifier (partially included also in the parts identifier document):
  - be the liaison with the database group on one side and the detector consortia, technical coordination, I&I teams on the other side
  - has the unique responsibility for defining the following fields in the parts identifier:
    - project
    - system ID
    - country of origin (using ISO codes)
    - responsible institution
    - detector ID
    - final destination
    - intermediate destination
  - direct work of computing expert working on interface of parts identifier DB (encoding/decoding information)

# Physicist in charge of Parts ID (ii)

- Job description for scientist in charge of the parts identifier (continues from previous slide):
  - review / approve the changes in the fields that are a responsibility of the consortia (ensure that rules / standards are enforced when using the parts identifier.)
  - collaborate with the consortia to spread the adoption of the parts identifier, inform the consortia of best practices and common pitfalls
  - understand / anticipate the needs of different detector consortia
- The parts identifier will also be used for some elements of the facility (infrastructure in the cavern, cryogenics)
  - We may need one person for the detector and one person for the facility or a single person that works with both the detector consortia and the facility/cryogenics/I&I
- Technical coordination is trying to identify the “physicist in charge” (Jim Stewart thinks we need a group...)

# Interface between partsID / hardwareDB

From the E-mail of Steve White (25 Feb)

The hardware DB expects the user to enter a unique identifier for each new component at the initial point of data entry

Two ways of doing this

- User first requests list of identifiers from parts identifier DB and then uses these identifiers for each component as they are entered in the hardware DB
- The hardware DB requests the identifier from the parts identifier DB when the user enters a new item

We should plan for both, but it's likely that the first one will be used more often for data insertion, while the second one will be more often for data extraction

- Note: I do not think that the parts identifier interface has a method to provide the user with the “next unused item number” of a given part

# Interface between partsID / hardwareDB

First method: the hardware DB does not need to know anything about the parts ID

- But then if you want to ask for “give me this part”, you have to specify the part by its identifier and not with a logic method (i.e. Dxxsssiiii-nnnn instead of TPC electronics FEMBs number nnnn)

If we want to support the second method (hardware DB requests the next identifier from the parts identifier):

- need to be able to use the fields for [project ID, system ID, sub-system ID, item type ID] to describe the parts into the hardware DB interface
- this requires either that data are exported from the parts identifier DB to the hardware DB, or that appropriate interfaces are made available



# Need more APIs from parts ID DB

Access to the next item of a given type

Access to the map (it's really a tree of maps) between identifiers and system ID, sub-system ID, item type ID

Operation on many IDs at once

Steve White pointed out issues with this model

- What do you do if hardware DB is available and parts ID DB is not responding ?
  - You need both of them to be operational for the 2<sup>nd</sup> option
- What do you do if somebody uses a parts identifier to enter an object in the hardware DB, but this has not been allocated / reserved in the parts identifier DB ?
  - Very easy to use the same entry multiple times (twice)

# Where is the parts identifier DB ?

The parts identifier DB sits at CERN because the initial developer was at CERN

I believe that the underlying implementation is based on MySQL

The user interface is rather nice (PHP ?)

I don't know whether there is any description of the implemented APIs, nor whether there is any description of the DB schema (the one in the EDMS document is insufficient / probably wrong (I don't remember if it was updated since mistakes were pointed out)

I don't know whether there is any documentation available

# Where do we go from here ?

We need to understand as soon as possible what is the current status of the parts identifier DB and its web interface

Understand how long it takes to implement missing features in API

Develop a plan to go forward including maintenance of the parts identifier DB / interfaces for XX years and interfaces with other databases

This includes answering the hard question: does the parts identifier DB need to be separate from the hardware DB ?

# Short term solutions ?

Part construction for ProtoDUNE-II has already started

APA consortium is already using sietch QR codes on its boards for the APAs

etching QR codes

30W Fiber Laser

input and  
QR code before



But if I scan this bar code with my cell phone I get a “502 Bad Gateway” error (even <https://dev.sietch.xyz> gives the same error)

<https://sietch.xyz> doesn't know this component....

From outside the APA consortium, it is not clear whether the data is already considered “production quality”. Some of the data is already production quality and may need to be migrated to a final DB later

# Short term solutions ?

Another possibility:

Any consortium can start allocating parts identifier numbers provided we follow a schema and each one of us is responsible for avoiding duplications

you don't even need a computer program for this

But then, we need to be able to flag the already used identifiers in the parts identifier DB

# Discussion

It should be clear to everybody that the parts identifier is a problem that needs to be solved as quickly as possible (it should have been solved a while ago)

- Identify scientist that takes responsibility for the system
- Identify computing expert(s) that take over from George
- Develop plan to move forward with interface between hardware DB (and other DBs...) and parts identifier
- Document existing API and understand what is missing from the API to support all kind of use cases (including interface to other DBs)
- Develop plan for long term support (including moving DB from CERN to Fermilab, better integration with other DBs ?)

If these issues are not solved quickly (next 2 weeks) there is a very important risk that we will see the rise of “homegrown solutions” that will require significant effort at a later time for migrating the data into the final database(s)