

Databases in mu3e

"how to and how it's going"

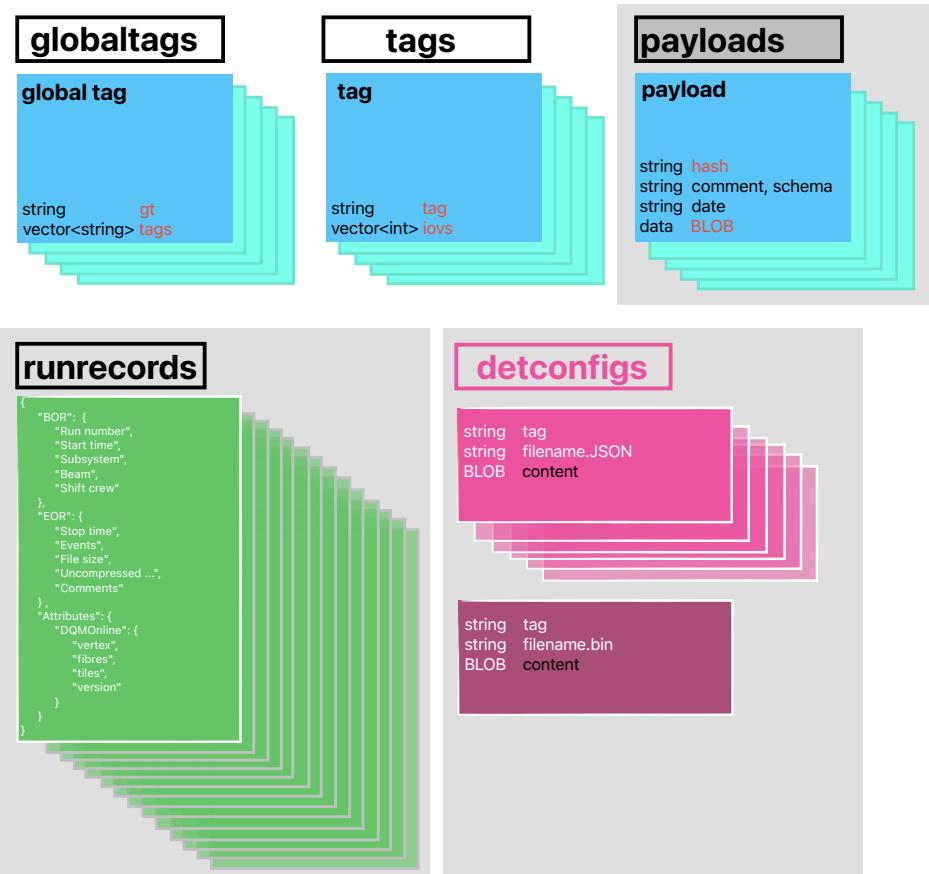
Urs Langenegger

2025/03/18

- CDB
 - ▷ runDB
 - ▷ det configs
 - ▷ conditions
- partsDB

Introduction

- CDB has three "domains" (so far)
- **RDB - Run Database**
 - ▷ information about (all) Mu3e runs (time, duration, comments, DQM, . . .)
 - ▷ records can be inserted/updated via **curl**
 - ▷ web user interface (alternatively)
- **detConfigs**
 - ▷ can contain "anything"
 - mask/TDAC files
 - JSON dumps from ODB/MIDAS
 - . . .
 - ▷ records inserted/extracted via **curl**
- **conditions** for reco/vtx/ana
(online and offline)
 - ▷ alignment constants,
chip status and numbering, . . .
 - ▷ retrieved in offline code
- **Curl** - "command line tool and library for transferring data with URLs"



RDB: run record

- MIDAS writes the (basic) record
 - ▷ currently:

```
moor>cat runlog_007565.json
{
  "BOR": {
    "Run number" : 7565,
    "Start time" : "Fri Jan 19 15:53:11 2024",
    "Subsystems" : 0,
    "Beam" : 0,
    "Shift crew" : "The data challenge crew"
  },
  "EOR": {
    "Stop time" : "Fri Jan 19 15:53:09 2024",
    "Events" : 4403,
    "File size" : 5840820,
    "Uncompressed data size" : 8692397,
    "Comments" : "Test data from the data challenge"
  }
}
```

- ▷ more information should be included
 - active subsystems
 - run type:
cosmics
commissioning
physics
 - B field strength
 - . . .

⇒ automate upload

```
curl -X PUT -H "Content-Type: application/json" --data-binary @/Users/ursl/data/mu3e/json13/runrecords/runlog_007553.json http://pc11740/rdb/runrecords
```

The screenshot shows two windows of the Mu3e Run Database. The top window is titled 'Run 7553' and displays a table of run parameters. The bottom window is titled 'Mu3e Run Database' and shows a list of the last 15 runs.

Run 7553 (Top Window):

Field	Value
Run number	7553
Start time	Fri Jan 19 15:44:05 2024
Stop time	Fri Jan 19 15:44:49 2024
Events	2587954
Shift crew	The data challenge crew
Comments	Test data from the data challenge
Subsystems	0
Beam	0
File size	4026005507
Uncompressed data size	5088259203
Online DQM summary	<input checked="" type="checkbox"/> Mu3e <input type="checkbox"/> Beam <input type="checkbox"/> Vertex <input type="checkbox"/> Fibres <input type="checkbox"/> Tiles

Mu3e Run Database (Bottom Window):

Showing the last 15 runs

Run	Start time	End time
7565	Fri Jan 19 15:53:11 2024	Fri Jan 19 15:53:09 2024
7564	Fri Jan 19 15:52:25 2024	Fri Jan 19 15:53:09 2024
7563	Fri Jan 19 15:51:39 2024	Fri Jan 19 15:52:23 2024
7562	Fri Jan 19 15:50:53 2024	Fri Jan 19 15:51:37 2024
7561	Fri Jan 19 15:50:08 2024	Fri Jan 19 15:50:52 2024
7560	Fri Jan 19 15:49:23 2024	Fri Jan 19 15:50:07 2024
7559	Fri Jan 19 15:48:37 2024	Fri Jan 19 15:49:21 2024
7558	Fri Jan 19 15:47:52 2024	Fri Jan 19 15:48:36 2024
7557	Fri Jan 19 15:47:06 2024	Fri Jan 19 15:47:50 2024
7556	Fri Jan 19 15:46:21 2024	Fri Jan 19 15:47:05 2024
7555	Fri Jan 19 15:45:36 2024	Fri Jan 19 15:46:20 2024
7554	Fri Jan 19 15:44:50 2024	Fri Jan 19 15:45:34 2024
7553	Fri Jan 19 15:44:05 2024	Fri Jan 19 15:44:49 2024
7552	Fri Jan 19 15:43:19 2024	Fri Jan 19 15:44:03 2024
7551	Fri Jan 19 15:42:34 2024	Fri Jan 19 15:43:18 2024

Record history (Bottom Window):

Date	Comment
2024/10/30 08:16:52	Database entry inserted
2024/10/30 08:16:56	Added DQMOnline
2024/10/30 08:17:40	DQM status changes: mu3e: 0

RDB: DQM information

- Data Quality Monitoring (DQM) information

- ▷ "online" minalyzer running in hut?
 - must be fast enough (harvesting)
- ▷ "delayed" harvester running in hut?
- ▷ "prompt-reco" running on merlin
 - full statistics
 - delayed, should keep up with data-taking
- ▷ pushed into "Attributes" array

- Currently:

```
[moor>cat dqm10uuu.json
{
  "DQMOnline": {
    "mu3e": "1",
    "beam": "0",
    "vertex": "-1",
    "fibres": "-1",
    "tiles": "-1",
    "version": "unset"
  }
}
```

```
_id: ObjectId('6721dd6432dc8cf86fa7e7fd')
▼ BOR : Object
  Run number : 7553
  Start time : "Fri Jan 19 15:44:05 2024"
  Subsystems : 0
  Beam : 0
  Shift crew : "The data challenge crew"
▼ EOR : Object
  Stop time : "Fri Jan 19 15:44:49 2024"
  Events : 2587954
  File size : 4026005507
  Uncompressed data size : 5088259203
  Comments : "Test data from the data challenge"
▼ History : Array (3)
  ▷ 0: Object
  ▷ 1: Object
  ▷ 2: Object
▼ Attributes : Array (2)
  ▷ 0: Object
    ▷ DQMOnline : Object
      mu3e : "1"
      beam : "0"
      vertex : "-1"
      fibres : "-1"
      tiles : "-1"
      version : "unset"
    ▷ 1: Object
      ▷ DQMOnline : Object
        mu3e : "0"
        beam : "0"
        vertex : "-1"
        fibres : "-1"
        tiles : "-1"
        version : "unset-2"
```

- ▷ of course, this is a template and should be computed for each run

⇒ automate upload (after online and/or prompt reco)

```
curl -X PUT -H "Content-Type: application/json" --data-binary @/Users/ursl/tmp/maskfiles/dqm10uuu.json http://pc11740/rdb/addAttribute/7553
```

RDB: Web user interface

- Single run display
 - ▷ shows all information available
 - ▷ allows updating of record
 - hit "Edit" to allow updating records
 - hit "Save" to store changes to RDB
- Note
 - ▷ you cannot add contents
 - without DQM attribute, you cannot add it
 - that should (and can) be done via curl
 - ▷ if you change a DQM attribute
 - status change will be in history
 - you could/should simultaneously edit "Comments"

The screenshot shows a web browser window titled "Mu3e RDB 7553" displaying a single run record. The URL is "localhost:5050/rdb/7553?runs=7553,...". The page has a header with "Run 7553" and navigation buttons: Back, Previous, Next, Edit, and Save. Below this is a table with "Field" and "Value" columns containing various run parameters. A "Comments" row contains the text "Test data from the data challenge". The "Online DQM summary" section includes checkboxes for Mu3e, Beam, Vertex, Fibres, and Tiles. Below the table is a "Record history" section with a table showing timestamped comments.

Field	Value
Run number	7553
Start time	Fri Jan 19 15:44:05 2024
Stop time	Fri Jan 19 15:44:49 2024
Events	2587954
Shift crew	The data challenge crew
Comments	Test data from the data challenge
Subsystems	0
Beam	0
File size	4026005507
Uncompressed data size	5088259203
Online DQM summary	<input type="checkbox"/> Mu3e <input type="checkbox"/> Beam <input type="checkbox"/> Vertex <input type="checkbox"/> Fibres <input type="checkbox"/> Tiles

Date	Comment
2024/10/30 08:16:52	Database entry inserted
2024/10/30 08:16:56	Added DQMOnline
2024/10/30 08:17:40	DQM status changes: mu3e: 0

detConfigs: many (binary) files

- Possibility to obtain "versioned/keyed" config files (\equiv detConfigs)
 - ▷ mask/TDAC files
 - ▷ JSON dumps from ODB/MIDAS
 - if you want it
 - not replacement of provenance tracking
- No CDB-based versioning (so far)
 - ▷ chose what you want: `mask_408`, `mu3eqc3setup`, `singleChip4`, . . .
- Usage examples:
 - ▷ Upload

```
[moor>foreach file (mask_408_1_*.bin)
foreach? echo $file
foreach? curl -v -F "file=@$file" -F "tag=mask_408" http://localhost:5050/cdb/upload
foreach? end
```

```
[moor>curl -v -F "file=@mask_408_1_9_DS_chip6.bin" -F "file=@mask_408_1_9_DS_chip5.bin" -F "tag=mask_408bis" http://localhost:5050/cdb/uploadMany
```

- ▷ Download

```
[moor>curl -O -J "http://localhost:5050/cdb/downloadTag?tag=mask_408"
% Total    % Received % Xferd  Average Speed   Time   Time   Current
          Dload  Upload   Total Spent    Left  Speed
100  169k    0  169k    0     0  194k      0 --:--:-- --:--:-- 194k
[moor>unzip
unzip   unzipsfx
[moor>unzip mask_408.zip
Archive: mask_408.zip
  inflating: mask_408_1_11_DS_chip4.bin
  inflating: mask_408_1_11_DS_chip5.bin
  inflating: mask_408_1_11_DS_chip6.bin
  inflating: mask_408_1_11_US_chip1.bin
  inflating: mask_408_1_11_US_chip2.bin
```

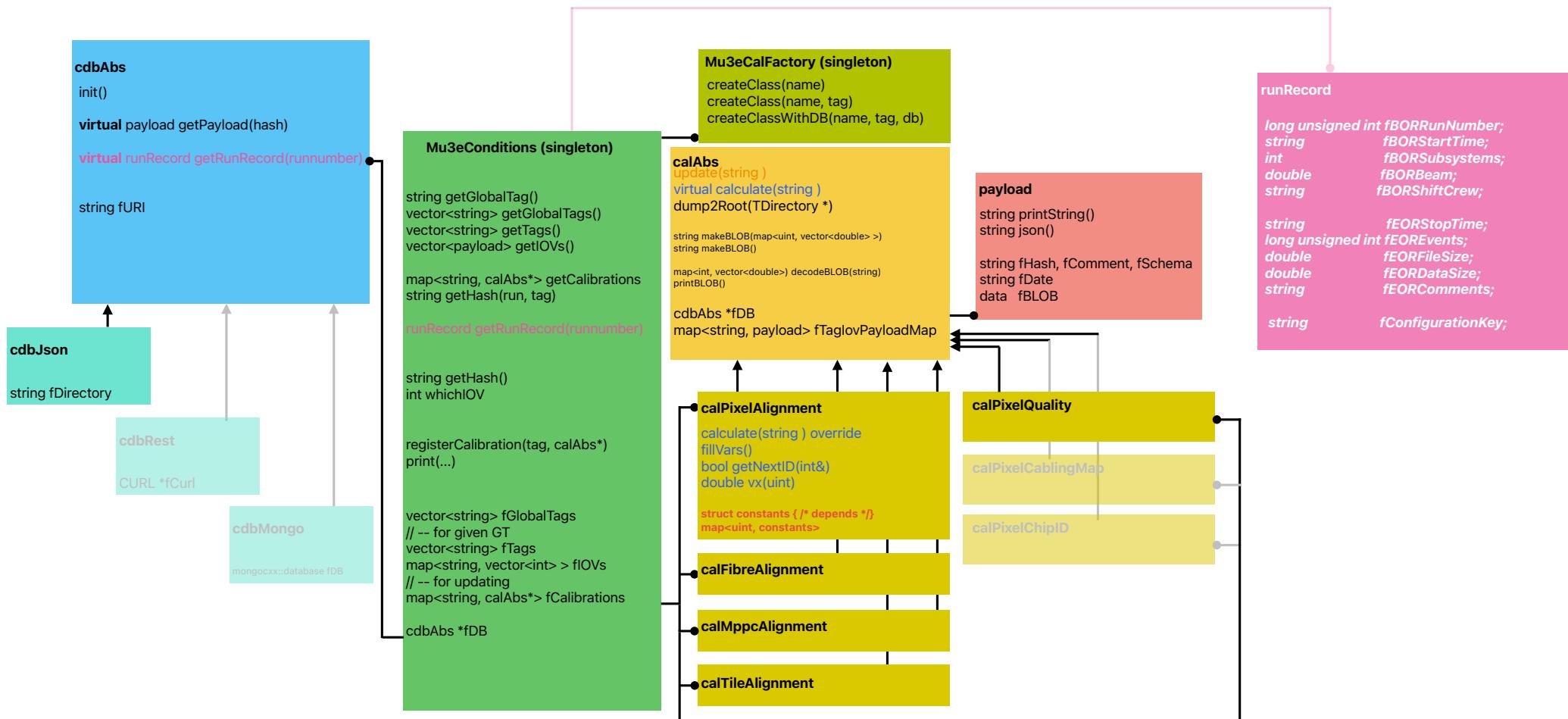
detConfigs: (single) JSON files

- Usage examples:

```
[moor]>ls -l someODBdump.json
-rw-r--r--@ 1 ursl staff 4077410 Oct 30 13:12 someODBdump.json
[moor]>curl -X POST -F "tag=someODBdump" -F "filename=someODBdump.json" -F "file=@someODBdump.json" http://localhost:5050/cdb/uploadJSON
[File uploaded successfully]
[moor]>curl http://localhost:5050/cdb/downloadJSON/someODBdump -o odb.json
% Total    % Received % Xferd  Average Speed   Time     Time      Current
          Dload  Upload   Total Spent    Left Speed
100 1727k  100 1727k    0      0  27.4M      0 --:--:-- --:--:-- --:--:-- 27.6M
[moor]>jq --indent 3 .odb.json > odb-formatted.json
[moor]>diff someODBdump.json odb-formatted.json
124420c124420
< }
\ No newline at end of file
---
```

- ▷ format issue to be solved (and also the filename/file duplicate argument)
- ▷ uploading with same tag will not delete previous entry
you'll get back the "last" uploaded version in case of multiple entries
- If reasonable use case develops, will provide "keyed" access
 - ▷ possibly with "global key"
 - containing multiple "key" for each subsystem
 - ▷ Note "key" somewhat similar to "tag" (for conditions)
 - ▷ Note difference between "tag" and "key"
 - IOVs are part of tag, but key changes for different setup conditions
 - will do this when needed/desired

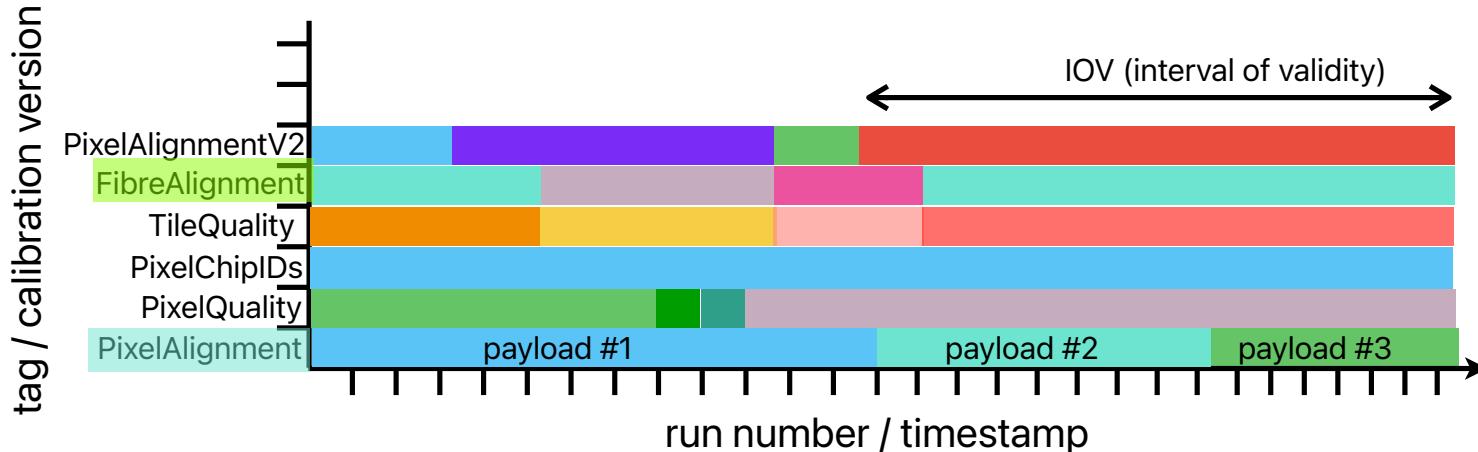
Conditions: code structure



- All coded in terms of abstract classes
 - ▷ DB backend (cdbJson vs cdbRest) can be replaced w/o problems
 - ▷ Constant specifics in one (concrete) place
- One central class (**Mu3eConditions**) to gain conditions access

Conditions: Tags, global tags, . . .

Tags combine conditions/calibrations with intervals of validity



Global tag collects consistent set of tags

mcidealv5.1	pixelalignment_mcidealv5.1	fibrealignment_mcidealv5.1	mppcalignment_mcidealv5.0	tilealignment_mcidealv5.1
mcidealv5.0	pixelalignment_mcidealv5.0	fibrealignment_mcidealv5.0	mppcalignment_mcidealv5.0	tilealignment_mcidealv5.0

- Notes
 - ▷ minimal coupling between software and conditions
 - ▷ payload name contains class and tag
 - ▷ record/payload/tag naming scheme not cast in stone
- ⇒ Important: $\text{detconf} \neq \text{detconfigs!}$
 - ▷ detconf = minimal detector configuration for offline (target, magnet)

Conditions: usage

```
// get and build mu3e, including make install!
// similar for minalyzer

// simulate and sort a run

./_build/mu3eTrirec/mu3eTrirec \
--cdb.dbconn=/Users/ursl/mu3e/software/241031-tutorial/mu3e/install/cdb/ \
directory/mu3e_sorted_000779.root directory/wcdb-mu3e_trirec_000779.root

./_build/mu3eVertex/mu3eVertexFit \
--cdb.dbconn=/Users/ursl/mu3e/software/241031-tutorial/mu3e/install/cdb/ \
--input=directory/wcdb-mu3e_trirec_000779.root --output=directory/wcdb-mu3e_vertex_000779.root

cd ../../minalyzer
_build/analyizer/minalyzer -e10 /Users/ursl/data/mu3e/run2024/run05378.mid.lz4 \
-- offline cdb.dbconn=/Users/ursl/mu3e/software/241031-tutorial/mu3e/install/cdb/ \
cdb.gt=mcidealv5.1
```

- Note

- ▶ inside PSI network you could use the REST interface: `--cdb.dbconn=rest`
this will connect you to pc11740
- ▶ you could specify another server (but no other server running so far)
- ▶ ASCII/file-based CDB is built in `make install`

Conditions: contents

- Conditions CDB content
 - ▷ `mu3e/mu3e> make install` will create JSON and payload(!) files
 - based on `mu3eUtil/cdb/ascii`

```
[moor]>ls -r /Users/ursl/mu3e/software/241031-tutorial/mu3e/install/cdb/*
/Users/ursl/mu3e/software/241031-tutorial/mu3e/install/cdb/tags:
tilealignment_mcidealv5.1      pixelalignment_mcidealv5.1      mppcalignment_mcidealv5.0      detconfv1_mcidealv5.1
tilealignment_mcidealv5.0      pixelalignment_mcidealv5.0      fibrealignment_mcidealv5.1
pixelalignment_qc2024v1.0      mppcalignment_mcidealv5.1      fibrealignment_mcidealv5.0

/Users/ursl/mu3e/software/241031-tutorial/mu3e/install/cdb/runrecords:
runlog_004001.json

/Users/ursl/mu3e/software/241031-tutorial/mu3e/install/cdb/payloads:
tag_tilealignment_mcidealv5.1 iov_1    tag_pixelalignment_mcidealv5.0 iov_1    tag_fibrealignment_mcidealv5.0 iov_1
tag_tilealignment_mcidealv5.0 iov_1    tag_mppcalignment_mcidealv5.1 iov_1    tag_detconfv1_mcidealv5.1 iov_1
tag_pixelalignment_qc2024v1.0 iov_1    tag_mppcalignment_mcidealv5.0 iov_1
tag_pixelalignment_mcidealv5.1 iov_1    tag_fibrealignment_mcidealv5.1 iov_1

/Users/ursl/mu3e/software/241031-tutorial/mu3e/install/cdb/globaltags:
qc2024v1.0      mcidealv5.1      mcidealv5.0

/Users/ursl/mu3e/software/241031-tutorial/mu3e/install/cdb/configs:
```

- Various tools exist for creating/viewing payload files
 - ▷ `mu3eUtil/cdb/test`
- so far, no (user) web interface to conditions CDB

partsDB - Introduction

- 'All components and materials that are used for the experiment should be registered in the database'
(L. Vigani)
- The partsdb should contain and keep all information on
 - ▷ material components in the experiment
 - ▷ qualification information (QC tests)
 - ▷ external material
- Not the same as the specbook
- Location <https://mu3epartsdb.phys.uni-heidelberg.de/>
 - ▷ contact me if you need an account
- Original specification and design ideas in
 - ▷ [Mu3e-Note-0020-QMproposal](#)
- Project growing over time
 - ▷ code very much historically grown
 - ▷ number of documents in partsdb: ≥ 33436 (was 20048 in June 2024)
 - ▷ size: 12 GB (as of now)

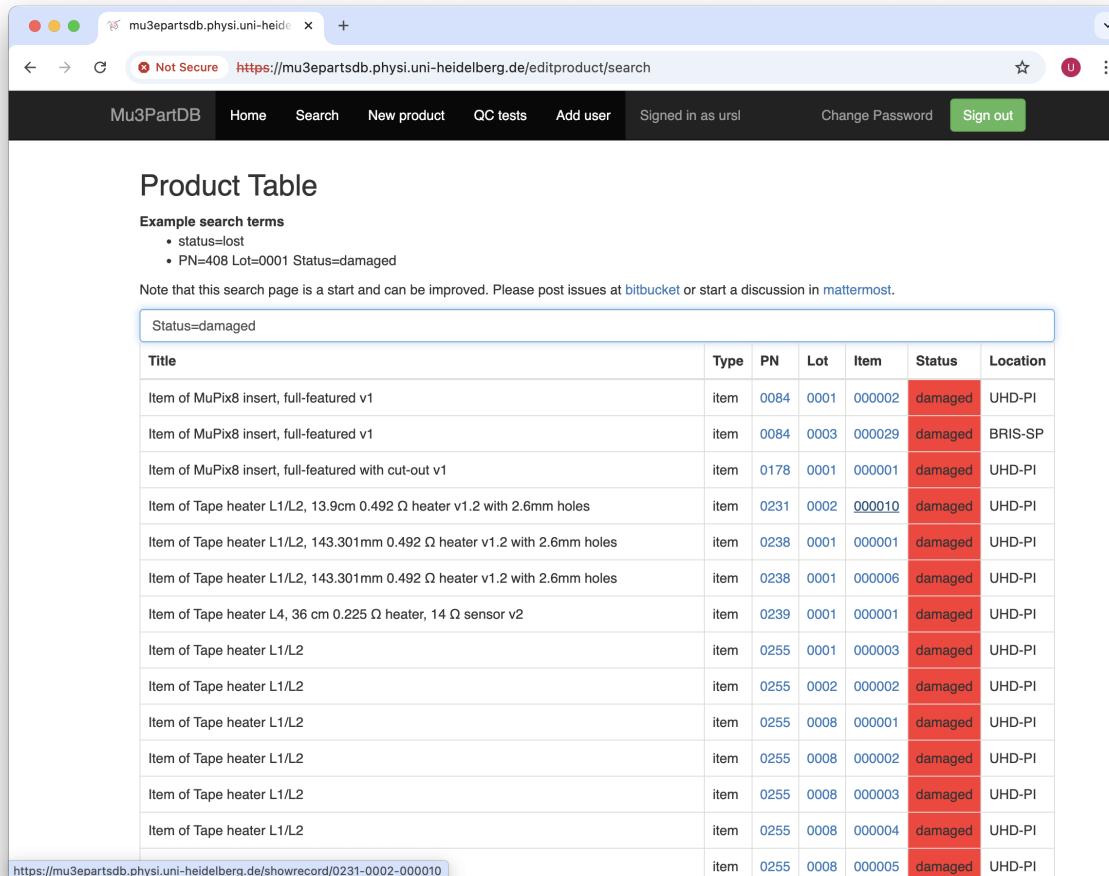
Parts DB GUI

The screenshot shows a web-based application titled "Mu3PartDB" with a dark-themed header bar. The header includes links for "Home", "Search", "New product", "QC tests", "Add user", and "Signed in as ursl". There are also buttons for "Change Password" and "Sign out". The main content area is a table listing 24 parts, each with a unique ID (e.g., 0564, 0565, ..., 0587), name, status (e.g., approved, preliminary), type (e.g., stdp, part, assy), supplier (e.g., fibre24, PCB Electronics, PCBElectronics), and notes (e.g., FOPC-F2-O3-MT12..., DAQ, fi, ladder, SciTile, SciTile, SciTile, FEB, F, Sense, pixel, v, pixel, v).

ID	Name	Status	Type	Supplier	Notes
0564	MTP12 Male to MTP12 Female OM3 cable ...	approved	stdp	fibre24	FOPC-F2-O3-MT12... DAQ, fi
0565	Interposer flex ladder L3/4 US (PCBElectron...	preliminary	part	PCB Electronics	
0566	Interposer flex ladder L3/4 DS (PCBElectron...	preliminary	part	PCBElectronics	
0567	Carbon-fibre ladder stiffener L4	approved	part		ladder
0568	Carbon-fibre ladder stiffener L3	approved	part		ladder
0569	Outer Pixel Ladder L4	approved	assy		ladder,
0570	Single chip card (SCC) PCB board	approved	assy		
0571	MTP bundle full assmbly, type 3, US	approved	assy		
0572	MTP bundle full assmbly, type 3, DS	approved	assy		
0573	MTP bundle full assembly, type 2, US	approved	assy		
0574	MTP bundle full assmbly, type 2, DS	approved	assy		
0575	SciTile Endring SSW	approved	part		SciTile
0576	SciTile Endring TGT	approved	part		SciTile
0577	Tile cooling system	approved	assy		SciTile,
0578	Front-end Crate (installed assembly)	approved	assy		FEB, F
0579	Sense Wire Board	approved	part	Multi-CB, KPH	Sense,
0580	Pixel Module L2 (Vertex Detector) - 50µm, 3...	approved	assy		pixel, v
0581	Pixel Module L1 (Vertex Detector) - 50µm, 8...	approved	assy		pixel, v
0582	Pixel Module L2 (Vertex Detector) - 50µm, 8...	approved	assy		pixel, v
0583	SciFi DAB 2.3	approved	assy	ETH Zurich	
0584	HDI (LTU) Layer 3 - three-layer flex	preliminary	part	LTU	
0585	HDI (LTU) Layer 4 production	approved	part	LTU	
0586	9-Chip test board PCB (upstream)	preliminary	assy		pixel, F
0587	Outer Pixel Ladder L3	approved	assy		ladder,

Search page

- A first step . . .
 - ▷ not all working as expected, missing features
 - ▷ computationally expensive (probably)



The screenshot shows a web browser window for the Mu3PartDB search page. The URL is <https://mu3epartsdb.physi.uni-heidelberg.de/editproduct/search>. The page title is "Product Table". A search bar contains the term "Status=damaged". The main content is a table with the following data:

Title	Type	PN	Lot	Item	Status	Location
Item of MuPix8 insert, full-featured v1	item	0084	0001	000002	damaged	UHD-PI
Item of MuPix8 insert, full-featured v1	item	0084	0003	000029	damaged	BRIS-SP
Item of MuPix8 insert, full-featured with cut-out v1	item	0178	0001	000001	damaged	UHD-PI
Item of Tape heater L1/L2, 13.9cm 0.492 Ω heater v1.2 with 2.6mm holes	item	0231	0002	000010	damaged	UHD-PI
Item of Tape heater L1/L2, 143.301mm 0.492 Ω heater v1.2 with 2.6mm holes	item	0238	0001	000001	damaged	UHD-PI
Item of Tape heater L1/L2, 143.301mm 0.492 Ω heater v1.2 with 2.6mm holes	item	0238	0001	000006	damaged	UHD-PI
Item of Tape heater L4, 36 cm 0.225 Ω heater, 14 Ω sensor v2	item	0239	0001	000001	damaged	UHD-PI
Item of Tape heater L1/L2	item	0255	0001	000003	damaged	UHD-PI
Item of Tape heater L1/L2	item	0255	0002	000002	damaged	UHD-PI
Item of Tape heater L1/L2	item	0255	0008	000001	damaged	UHD-PI
Item of Tape heater L1/L2	item	0255	0008	000002	damaged	UHD-PI
Item of Tape heater L1/L2	item	0255	0008	000003	damaged	UHD-PI
Item of Tape heater L1/L2	item	0255	0008	000004	damaged	UHD-PI
	item	0255	0008	000005	damaged	UHD-PI

At the bottom of the table, there is a link: <https://mu3epartsdb.physi.uni-heidelberg.de/showrecord/0231-0002-000010>

⇒ waiting for issues or complaints :-)

Terminology (product/lot/item/"part")

- **Product:** A type of something 4-digit identifier (will [have to] stay like that)
 - ▷ e.g. MuPix11, Terasic DE5A-NET ARRIA 10 GX, SciTile Module v2, . . .
 - ▷ Types of products
 - "part": "Single item, not an assembly"
 - "assy": "Assembly, consists of parts"
 - "rawm": "Raw material, used to fabricate things"
 - "stdp": "Standard catalog part from a supplier"
 - "eqpm": "Piece of equipment, usually from a supplier"
 - ▷ Status possibilities
 - "preliminary": "Preliminary part, documentation not yet signed off."
 - "approved": "Approved part. Good for production."
 - "obsolete": "Obsolete part, do not use anymore."
 - "special": "Special part, not good for production but for testing/studies."
- **Lot:** Group of units of a product 4-digit identifier (will [have to] stay like that)
 - ▷ always with a **(fixed!)** lot size. Starting item number can be chosen.
 - ▷ with or without items → "consumables" (can be added in bulk to assemblies)
- **Item:** Single unit of a product 6-digit identifier (will [have to] stay like that)
 - ▷ just a number. Not the index (slot number) returned by some utility :-)

Document contents (couchdb)

```
{ "_id": "cf3b94fff45519009a3b90fb68808877", "pn": "0581", "lot": "0001", "item": "000001", "mfgSN": "", "status": [ { "status": "inproduction", "datetime": "2025-01-16T15:36:44.562Z", "user": "thomi", "comment": "" }, { "status": "planned", "datetime": "2025-01-16T15:31:36.896Z", "user": "thomi", "comment": "Initial entry" } ], "qcstatus": [ { "status": "quarantine", "datetime": "2025-01-16T15:31:36.896Z", "user": "thomi", "comment": "Initial entry" } ], "consistsof": [ { "pn": [ "0397", "0237", "0425" ], "amount": "8", "isconsumable": false, "unit": "pcs", "madeof": [ {}, .. snip .. etc .. ] }, { "pn": [ "0364", "0365" ], "amount": "1", "isconsumable": false, "unit": "pc", "madeof": [ { "pn": "0364", "lot": "0001", "item": "000008" } ] }, { "pn": [ "0365" ], "amount": "1", "isconsumable": false, "unit": "pc", "madeof": [ { "pn": "0365", "lot": "0001", "item": "000007" } ] }, { "pn": [ "0240" ], "amount": "2", "isconsumable": false, "unit": "pcs", "madeof": [ { "pn": "0240", "lot": "0002", "item": "000001" } ] } ], "usedby": {}, "owner": "UHD-PI", "location": [ { "location": "PSI", "datetime": "2025-01-16T15:31:36.896Z", "user": "thomi", "comment": "Initial entry" } ], "Comments": [] }
```

0397 = "Interposer 7x12"
0237 = "Interposer bracket"
0425 = "Screw M1.2x3 brass A2 DIN 84"

Urs Langenegger

Databases in mu3e (2025/03/18)

Tracking a part/item (not a product)

- Keep record of the history and status of the parts
 - ▷ where is it? who had it last? why is/was it not used?
- Status lifecycle (originally: strictly sequential), normally
 - ▷ planned
 - ▷ ordered
 - ▷ inproduction
 - ▷ intransit
 - ▷ delivered
 - ▷ tested
 - ▷ available
 - ▷ used
- With dead ends
 - ▷ cancelled, damaged, lost
- QC states
 - ▷ approved, quarantine, rejected
 - ▷ originally with restrictions

Now largely open: user decides next status

The screenshot shows a web-based application for tracking part history. At the top, there's a navigation bar with links for Home, Search, New product, QC tests, Add user, and a signed-in user section. Below the navigation, there are fields for Mfg. lot and Mfg. SN. Under the 'Expiration date' field, it says 'never'. The 'Status' section shows a history of states:

Date	Status	User	Comment
21-Aug-2023 16:54	available	thomi	
21-Aug-2023 16:54	tested	thomi	
21-Aug-2023 16:54	delivered	thomi	
21-Aug-2023 16:53	inproduction	thomi	
21-Aug-2023 16:42	planned	thomi	Initial entry

Below this, there are sections for 'QC test ID' (QC test not defined) and 'QC status' (approved). The 'Owner' is listed as PSI. The 'Location' section shows a history of locations:

Date	Location	User	Comment
02-Dec-2024 18:56	PSI	thomi	PIA 4th floor - module assembly station
21-Aug-2023 16:42	PSI	thomi	Initial entry

At the bottom, there are fields for 'Used in making of' (empty) and 'Not used for prototyping' (empty), followed by a blue 'Make prototype' button.

Item location

The screenshot shows a web browser window titled "Lot detail" with the URL <https://mu3epartsdb.physi.uni-heidelberg.de/showrecord/0365-0001>. The page is part of the Mu3PartDB system. At the top, there is a navigation bar with links for Home, Search, New product, QC tests, Add user, Signed in as ursl, Change Password, and Sign out. Below the navigation bar, there are two log entries:

21-Aug-2023 17:19	Labels generated for all items, format: TZe9mmVert	thomi
21-Aug-2023 16:42	labeled as AS103B-0	thomi

Below these entries, the heading "Items in this lot" is displayed, followed by a table listing 13 items:

Item	Mfg. SN	Status	QC status	QC result	Location	Owner
000001		used	approved	Not available	PSI (unclear location)	PSI
000002		available	approved	Not available	PSI (PIA 4th floor - module assembly station)	PSI
000003		available	approved	Not available	PSI (PIA 4th floor - module assembly station)	PSI
000004		available	approved	Not available	PSI (PIA 4th floor - module assembly station)	PSI
000005		used	approved	Not available	PSI (PIA 4th floor - module assembly station)	PSI
000006		available	approved	Not available	PSI (PIA 4th floor - module assembly station)	PSI
000007		used	approved	Not available	PSI (PIA 4th floor - module assembly station)	PSI
000008		available	approved	Not available	PSI (PIA 4th floor - module assembly station)	PSI
000009		available	approved	Not available	PSI (PIA 4th floor - module assembly station)	PSI
000010		used	approved	Not available	PSI (PIA 4th floor - module assembly station)	PSI
000011		used	approved	Not available	PSI (PIA 4th floor - module assembly station)	PSI
000012		used	approved	Not available	PSI (unclear location)	PSI
000013		used	approved	Not available	PSI (unclear location)	PSI

Documentation

- Wiki with links to
 - ▷ Luigi's user instructions
 - ▷ videos
 - Assemblies must have lots and items when you create the product (of the assembly)
 - Fill all positions in an assembly with "items" of a lot-only product
 - ▷ create local instance of parts DB
 - ▷ rationale

The screenshot shows a Bitbucket repository page for 'mu3epartdb'. The top navigation bar includes links for Bitbucket, Your work, Pull requests, Repositories, Projects, People, More, and a Create button. The left sidebar contains links for Source, Commits, Branches, Pull requests, Pipelines, Deployments, Issues, Jira issues, Security, Wiki (which is highlighted in blue), and Downloads. The main content area displays the repository's homepage with the title 'mu3epartdb / Home' and a 'Create page' button. A list of links is shown, which matches the list provided in the slide text:

- Luigi's user instructions
- Tutorial videos
- Create a local instance of the partsdb
- Discussion on what to put into the partsdb (for reference)

Below this list, it says 'Updated 15 seconds ago'.

- User support
 - ▷ mattermost
 - ▷ bitbucket issues

Summary

- CDB has three domains (so far)
 - ▷ RDB - Run Database
 - ▷ detConfigs - detector configuration data
 - ▷ conditions - for offline reco/vtx/ana
- Access to CDB
 - ▷ **conditions**
 - ASCII files created during **make install** (in mu3e/mu3e or mu3e/mu3eUtil)
 - `cdb.dbconn=/Users/ursl/mu3e/software/241031-tutorial/mu3e/install/cdb/`
beware of differences (--) between minalyzer and offline executables
 - `cdb.dbconn=rest`
REST api works within PSI network (VPN)
 - ▷ **RDB** with curl or browser <http://pc11740/rdb>
 - ▷ **detConfigs** with curl (only)
- partsDB
 - ▷ relevant components of experiment should be registered
 - ▷ Location <https://mu3epartsdb.phys.uni-heidelberg.de/>
 - ▷ contact me if you made a mistake, *e.g.*, wrong lot size :-(