

Run History DB and MetaCat

- Data discovery

Ana Paula Vizcaya Hernández

11/04/2023

Understanding the data

What data do we want to store?

- Heidi showed a list of data coming from a ProtoDUNE run 1 file
- Data used during ProtoDUNE run 1

I created a spreadsheet with all the data:

https://colostate-my.sharepoint.com/:x:/g/personal/avizcaya_colostate_edu/EeUJJ4LBS_RBnIPqQVHjag4BbvnwaNdHIDoEfJ1Qbfl7cQ?e=nuwngB

	Data	Data type	Examples	Where to find it	Are they necessary?	Comments	Status - In UC	Status - c	Status -
3	run_number	bigint	18000	DAQ metadata file	Yes		y		n
4	dunedaq.start_time	timestamp	2018-10-17T19:45	DAQ metadata file	Yes		y		n
5	dunedaq.end_time	timestamp	2018-10-17T19:56	DAQ metadata file	Yes		y		n
6	dunedaq.run_type	char	prod, test, etc	DAQ metadata file	Yes		y		n
7	dunedaq.detector_ID	char	np04_coldbox, np	DAQ metadata file	Yes		y		n
8	dunedaq.version	char	rc-v3.2.1-1	DAQ metadata file	Yes		y		n
9	DUNE_data.acCouple	decimal	0	DAQ config files	yes	one for each fer	y		n
10	DUNE_data.calibpulsemode	int	32	DAQ config files	Yes		y		n
11	DUNE_data.DAQConfigName	char		DAQ config files	Maybe		y		n
12	DUNE_data.febaselineHigh	int -> float	2	DAQ config files	yes	The number giv	y		n
13	DUNE_data.fegain	int -> float	0	DAQ config files	yes	The number giv	y		n
14	DUNE_data.feleak10x	int -> float	false/0	DAQ config files	yes	The number giv	y-v	some vers	n
15	DUNE_data.feleakHigh (leak)	int -> float	0	DAQ config files	yes	The number giv	y		n
16	DUNE_data.feshapingtime (peak-tim	int -> float	3	DAQ config files	yes		N		n
17	DUNE_data.is_fake_data			DAQ config files?	yes		N		n
18	beam spills	[int]*		IfBeam DB?	Maybe	HMS - This is ac	N		n
19	beam.momentum	decimal	1	Elog or IfBeam DB?	Yes	Need to know t	N		n
20	beam.polarity	char	positive	Elog or IfBeam DB?	Maybe		N		n
21	detector_hv_value	decimal	180	Elog or Slow control	Yes		N		n
22	Wire Bias	[int]*	G: -665V; U: -370	Elog or Slow control	Maybe	Three rows for:	N		n
23	List of raw-data files for this run	[char]*		samweb	Yes	HMS - this is ver	N	Is samweb	n
24	dune-raw-data.timestamp	timestamp	2018-10-17T19:56	samweb get-metadat	Yes		N		n
25	dune-raw-data.version	char		samweb get-metadat	Maybe	hdf5 don't have	N		n
26	dune-raw-data.file_type	char	detector	samweb get-metadat	Yes		N		n
27	dune-raw-data.event_count	bigint	3	samweb get-metadat	Yes	these are pretty	N		n
28	dune-raw-data.fisrt_event	bigint	11463	samweb get-metadat	Yes	Useful if you wa	N		n
29	dune-raw-data.last_event	bigint	11470	samweb get-metadat	Yes	Why is it not fir	N		n
30	dune-raw-data.file_type	char	protodune-sp	samweb get-metadat	Yes		N		n
31	dune-raw-data.file_format	char	root	samweb get-metadat	Yes		N		n
32	dune-raw-data.start_time	timestamp	2018-10-17T19:56	samweb get-metadat	Maybe		N		n
33	dune-raw-data.end_time	timestamp	2018-10-17T19:56	samweb get-metadat	Maybe		N		n
34	artdaq-core.timestamp	timestamp	2018-10-17T19:45	? samweb file.root	Maybe	The run control	N		n
35	artdaq-core.version	char	v3_04_02	? samweb file.root	Yes?	Later hdf5 files	N		n
36	artdaq.timestamp	timestamp	2018-10-17T19:45	? samweb file.root	Maybe		N		n
37	artdaq.version	char	v3_04_02_beta	? samweb file.root	Yes?		N		n
38	data_quality.online_good_run_list			?	Yes	this may be mu	N		n
39	subruns	N/A	N/A	N/A	N/A				
40									
41									
42									
43									
44									

* To have a relational DB these lists will need to be represented as another table int the schema

DAQ payloads

DAQ data and metadata

- DAQ metadata is the same for all runs
- DAQ data has some versions
 - Shape and peak_time should be equivalent
 - baseline_high and baseline also
 - enable_femb_fake_data just available in old runs

DAQ extra data?

Some parameter are not provided by the DAQ config parameters

- list of output files, event numbers
- Should we ask DAQ team if it's possible to include this parameters?

payloads	type
RUN_NUMBER	integer
START_TIME	TIMESTAMPZ
STOP_TIME	TIMESTAMPZ
DETECTOR_ID	text
RUN_TYPE	text
SOFTWARE_VERSION	text
buffer	integer
ac_couple	boolean
pulse_mode	integer
pulse_dac	integer
pulser	boolean
baseline_high	integer
baseline	integer
gain	integer
leak	integer
leak_high	integer
leak_10x	boolean
shape	integer
peak_time	integer
enable_femb_fake_data	boolean
enabled	boolean
test_cap	boolean
daq_config_name	text

Slow controls and IFbeam

Slow control

- The sensor I wanted to use for the hv **does not take data all the time**. So, if we use that sensor, some runs would not have a hv.
 - We are in contact with Xavier Pons (sc expert) to get the correct sensor

IFbeam

Some parameter are not provided by the IFbeam like:

- Beam momentum
- Beam polarity

Where to get the data from?

Other places to get the data

Elogs - not good

- They are filled in by hand
- There are several runs that do not have a corresponding elog
- <https://pdsp-elog.cern.ch/elisa/display>

Spreadsheets - not good

- Same problems, filled by hand
- Data of just the 'preferred runs' is inserted
- https://docs.google.com/spreadsheets/d/1O4o9_q8F-KynQltKDAfmco3e_s1eKltFkzZ4-g_Vls0/edit#gid=0

Showing 166 to 180 of 500 entries

	Date&Time	Author	Subject	Message Type	System Affected	Text
Reset	Search Date&Time	Search Author	Search Subject	Search Message Type	Search System Affected	Search Text
	2022-12-14 10:45	hanjie	hanjie started new run 18262 (PROD) on...	RunControl Message	DAQ	User hanjie started run 18262 of type PROD on np04_hd...
	2022-12-12 20:52	hanjie	hanjie started new run 18254 (PROD) on...	RunControl Message	DAQ	User hanjie started run 18254 of type PROD on np04_hd...
	2022-12-11 20:00	hanjie	hanjie started new run 18237 (PROD) on...	RunControl Message	DAQ	User hanjie started run 18237 of type PROD on np04_hd...
	2022-12-09 18:46	hanjie	hanjie started new run 18236 (PROD) on...	RunControl Message	DAQ	User hanjie started run 18236 of type PROD on np04_hd...
	2022-12-08 20:16	hanjie	hanjie started new run 18231 (PROD) on...	RunControl Message	DAQ	User hanjie started run 18231 of type PROD on np04_hd...

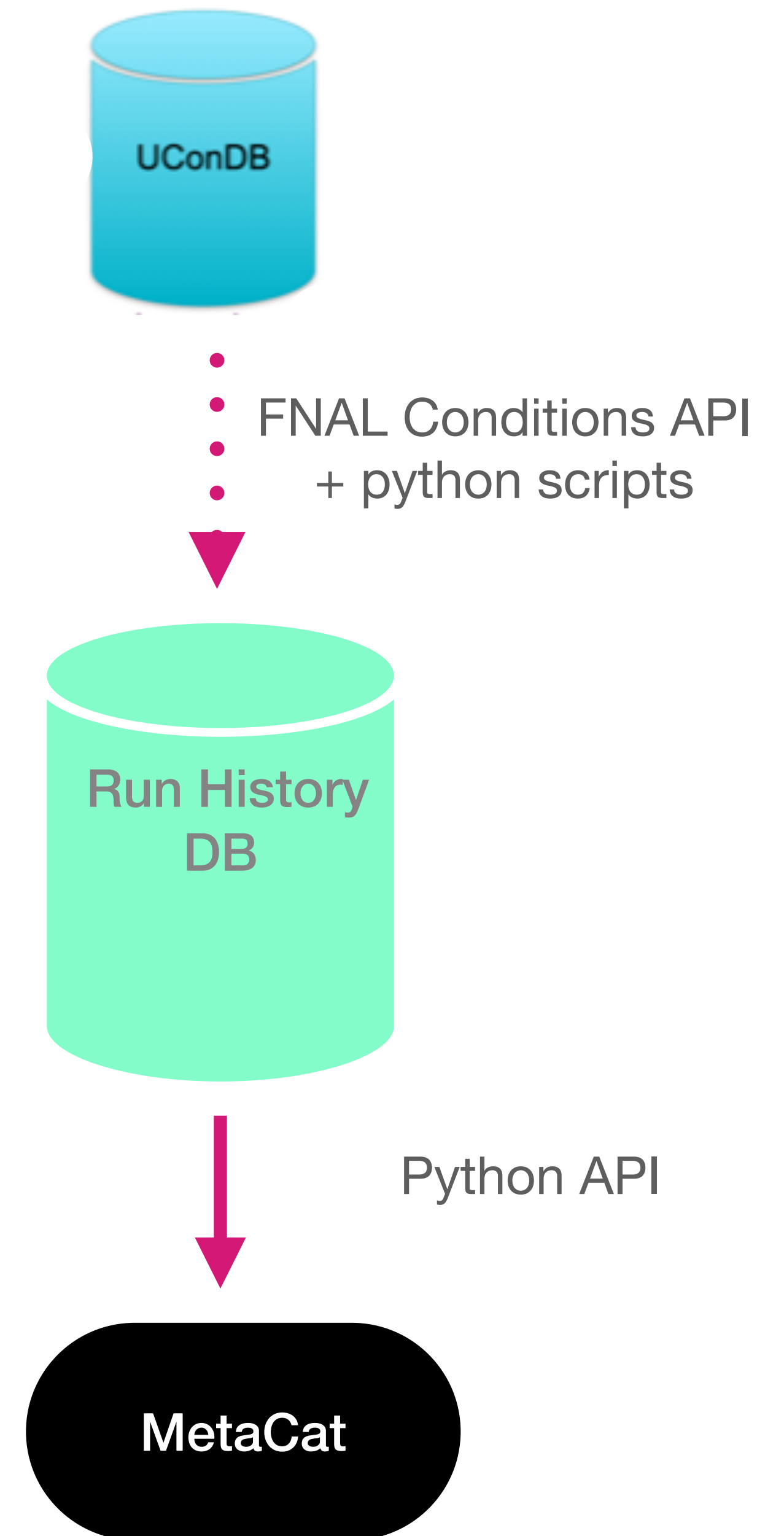
FNAL Conditions DB

Use modified schema where we have one conditions changing table and one versioning table

Updates

1. Created test table which lives in `pdunesp_prod`
2. There is an API that facilitates inserting data, I tried it out and started filling the test table
3. Test table contains DAQ payloads

There already exists a python API to communicate with MetaCat, but it has to be modified to the final version of the payloads.



Outlook

- Finalizing the payloads
 - where to find the data?
 - what is needed?
- Create the Run History table and populate it
- Populate the new Run Conditions DB
- Update python API to communicate with MetaCat

Thank you



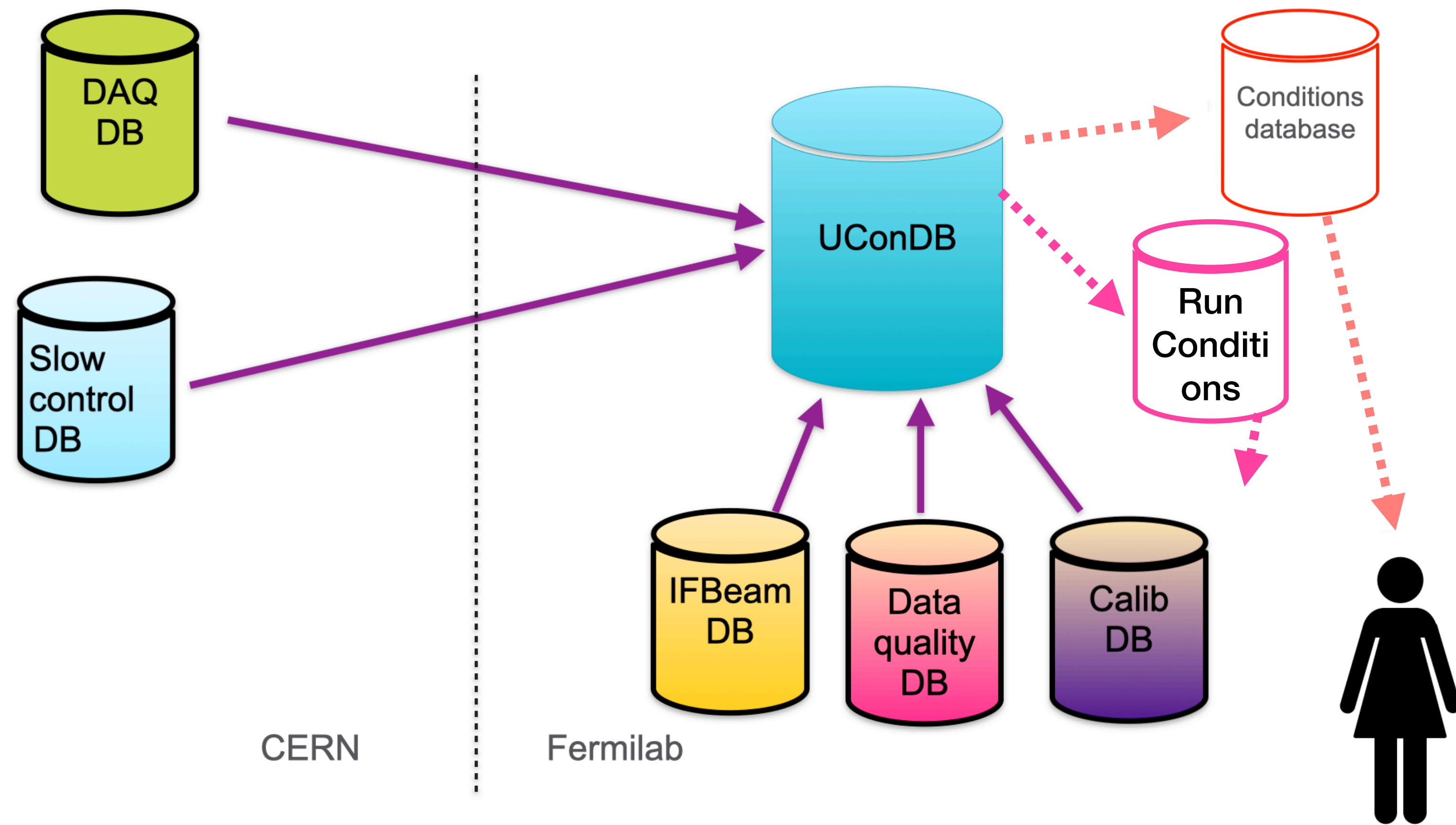
A decorative graphic on the left side of the slide, consisting of a grid of small, dark green dots arranged in 12 rows and 4 columns.

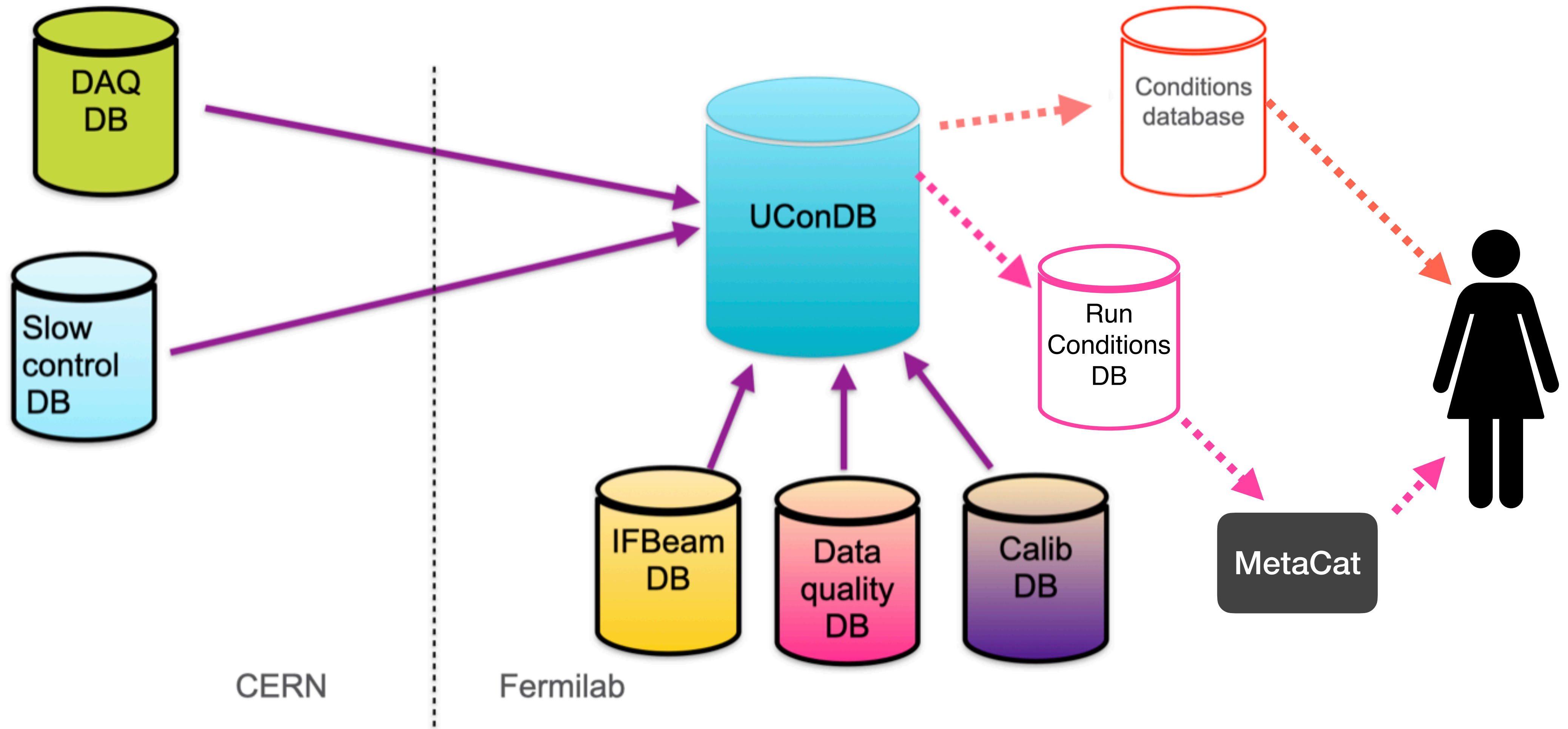
Backup slides

Run History DB / Conditions DB

Run History DB

- It is a relational DB with just a selection of run conditions parameters
- **Purpose:** get runs/files for runs with specific configurations.
- **Goal:** integrate the Run Conditions DB with MetaCat





Run History DB outlook

- Make a new runs-test folder
- Start with a short list of parameters from the DAQ DB to demonstrate that all the infrastructure is working for data challenge
- Decide on a new folder or a new database
- Make a more complete selection of data
- Work with SAM team to include this database into their infrastructure

Calibration DBs data

Data

- It contains 22 parameters
- Divided in 4 databases
- The correction DBs give a run number

Question

- The data will be transferred to the UConDB, Should we include them in the run history DB? since the databases are not automatically filled this may take some time

Database	Parameter
Electron lifetime	lifetime_TPCC
	lifetime_TPCL
	lifetime_TPCH
	Timestamp
dQ/dx YZ correction	channel
	Run Number
	y
	dy
	z
	dz
	corr
	corr_err
dQ/dx X correction	channel
	Run Number
	x
	dx
	shape
	shape_err
dQ/dx normalization correction	channel
	Run Number
	norm
	norm_err

Adding IFBeam data

Status

- Created executable program (instead of ART module) to transfer data from the IFBeam DB to the UConDB
- Big thanks to Marc Menguel, who is back from extended leave

To - do

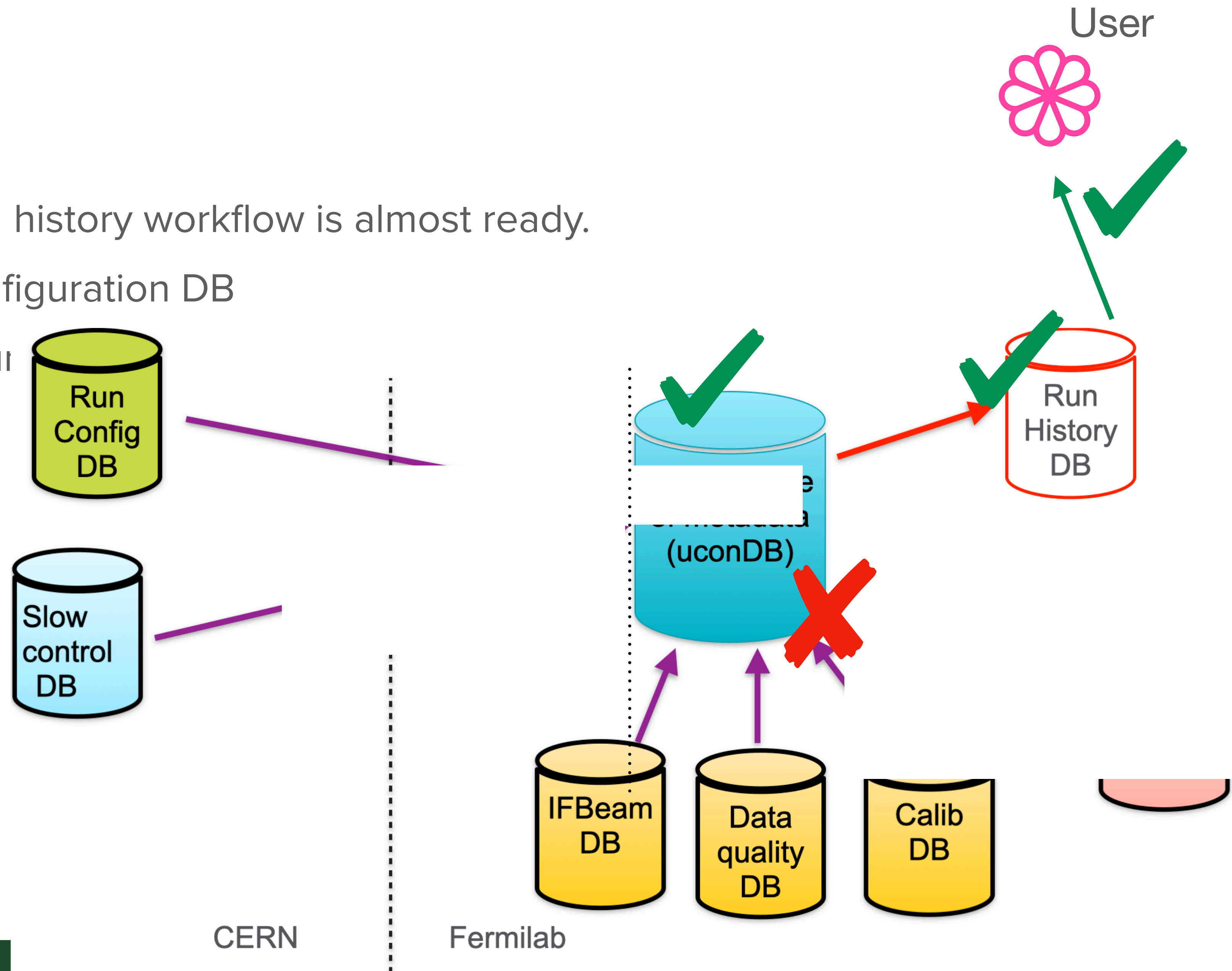
- For the data challenge, add function to transfer data from the IFbeam table in the UConDB to the run history
 - Using the devices 35 devices from Beam Event analysis
 - Take mean/std of each run
 - Suggestions?

Conclusion

- The basic infrastructure of the run history workflow is almost ready.
 - Contains data from the run configuration DB
 - Using a test table in the new run history DB

To - do

- Add function to transfer data from Slow control DB
- Web interface for the run history?



Run history DB for HD

Location

- It will be a table in the database: [pdunehd_prod](#)
- [Special permission](#) is needed [to access](#) the Db (just for DB experts! - no users)
 - Request access to the DB, create a ticket, and Olga will probably handle it
- Host = ifdbprod.fnal.gov, Port: 5451, and dbname=pdunehd_prod

Amenities

- Development database: [pdunehd_dev](#)
 - Used for testing
 - Host = ifdb07.fnal.gov, Port: 5448

New Run History DB for protoDUNE-HD

Tables and data

- For a test, the data was loaded to the public schema, but it will be modified in the future
- There is one table with data: test_runs `[('test_runs',)]`

- Data that can be uploaded:

```
run_number  
start_time  
stop_time  
detector_id  
run_type  
software_version
```

How to access the Run History?

For now

- The data is accessible via Query Engine, which is widely used in protoDUNE

```
-bash-4.2$ curl https://dbdata0vm.fnal.gov:9443/QE/protodune_prod/app/query?t=test_runs
run_number,start_time,stop_time,detector_id,run_type,software_version
12006,2021-11-05 11:31:22-05:00,2021-11-05 11:34:32-05:00,np02_coldbox,PROD,dunedaq-v2.8.1
126,2021-11-05 17:31:22-05:00,2021-11-05 17:34:32-05:00,np02_coldbox,PROD,dunedaq-v2.8.1
1206,2021-11-05 11:31:22-05:00,2021-11-05 11:34:32-05:00,np02_coldbox,PROD,dunedaq-v2.8.1
106,2021-11-05 11:31:22-05:00,2021-11-05 11:34:32-05:00,np02_coldbox,PROD,dunedaq-v2.8.1
```

In the future

- SAM and/or Metacat
- Web interface to see the table with the run history parameters?

New Run History DB for protoDUNE-HD

Location

- It will be a table in the database: [pdunehd_prod](#)
- [Special permission](#) is needed to access the Db (just for DB experts! - no users)
 - Request access to the DB, create a ticket, and Olga will probably handle it
- Host = ifdbprod.fnal.gov, Port: 5451, and dbname=pdunehd_prod

Amenities

- Development database: [pdunehd_dev](#)
 - Used for testing
 - Host = ifdb07.fnal.gov, Port: 5448

