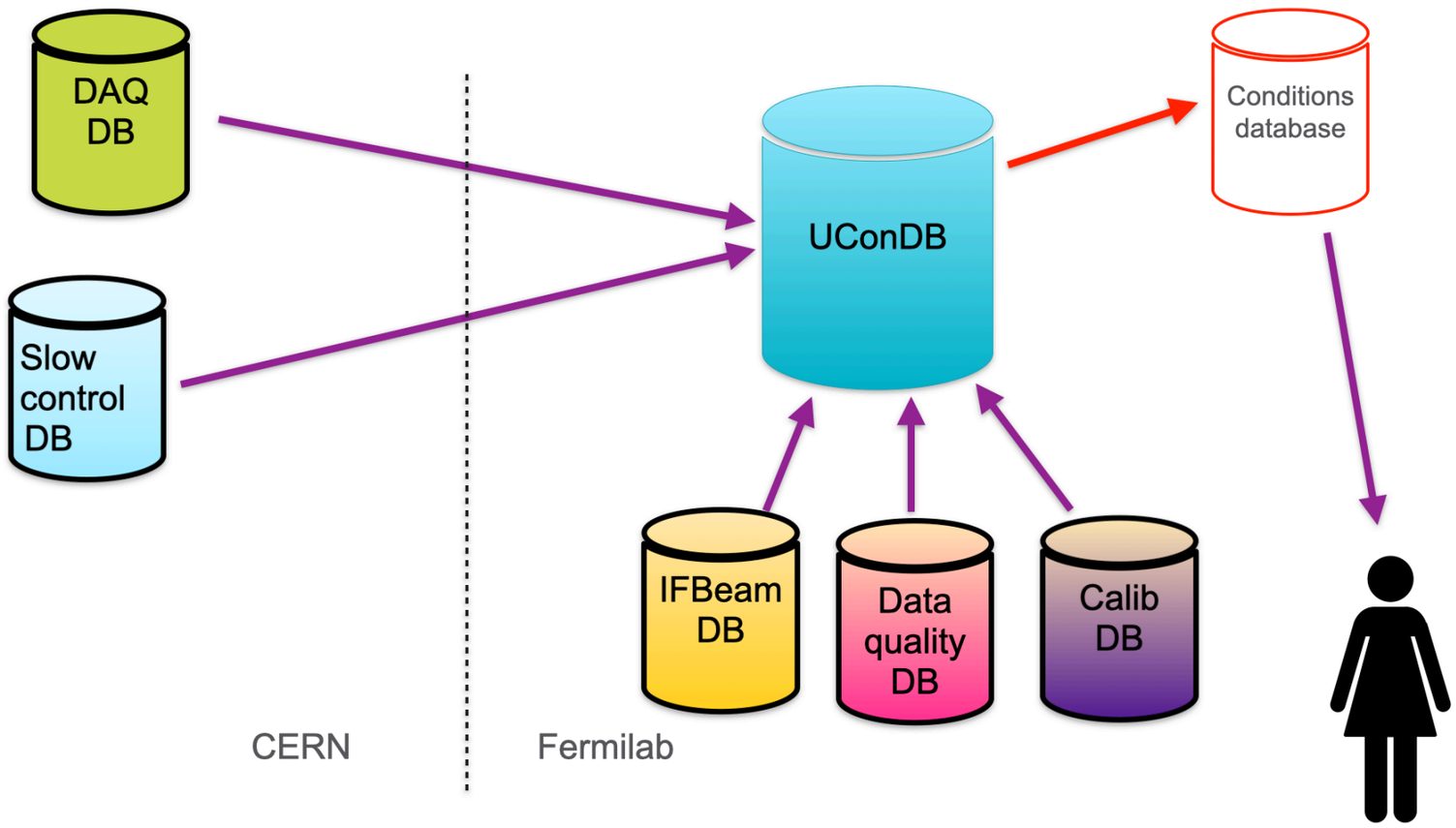
Run Configuration DB

Updates

Ana Paula Vizcaya Hernández

11/30/2022



Colorado State University

Data transfer after run

ProtoDUNE I

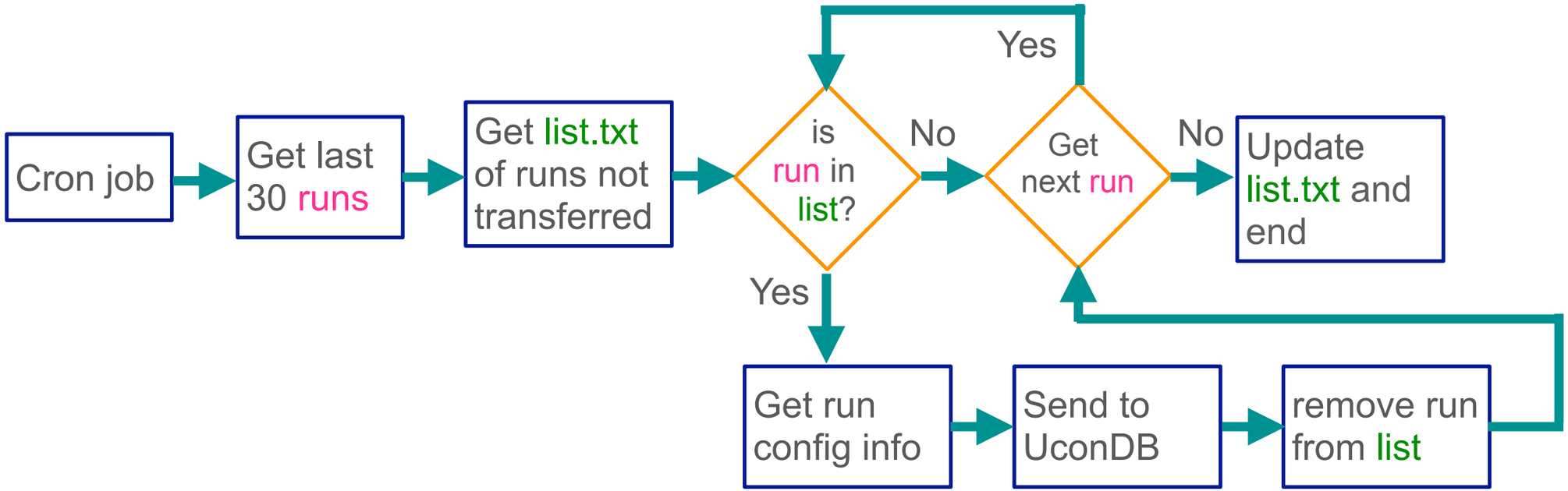
- Cron job that ran every 2 minutes
- Check for end of run:
  - Used the appearance of a stop.txt file on disk
  - Was the stop.txt file created ~ 2 min ago?
  - Look if the run info was transferred

ProtoDUNE Now

- Cron job that will run every ~2 min
- Storing of run registry info is now handled by web service instead of files on disk
- Check for end of run:
  - Extract metadata of last 30 runs using run registry service
  - Check which runs have been transferred

Data transfer after run - how it works

- Consulting with Kurt Biery - finalized defining script parameters
- Things that I had to take into account
  - Runs not necessarily finish in order
  - Incomplete runs - never finish
  - Working with list of runs not transferred to same memory



Data transfer after run - Status

- Consulting with Kurt Biery - help with daq account

Status - testing

- Currently running in .cron folder on np04-srv-024
- Using np04daq account
- It runs in scratch folder
- A log file is saved in /log/configDB_transfer

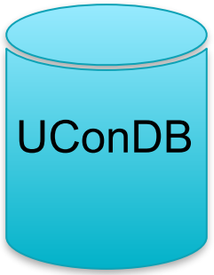
Not permanent, but once the testing is done will talk to system admin to include it.

- Cron jobs on DAQ computers at NP04 are managed by 'ansible'

Conditions data - storage update

The Run Configuration blobs are store in the UconDB

- Created a new folder and object
  - Creating a folder is not as trivial - new table in PostgreSQL so you have to specify permission and passwords.
- First run in new folder/object is: 11902, since it's first run with data after the DAQ changed
- Up-to-date with current runs
- Code in .cron folder and in my home area



- New folder and Objects!
- Not created

Folders:

sp_protodune

….

protodune_conditions

Objects:

configuration _all

configuration _data

Version:

blob-run11902
blob-run11903
….

blob-run#
blob-run#
…

Conditions data - from blob to json

```
Start of Record
 Run Number: 12000
 Packed on Feb 08 03:57UTC

#######
12000/runMeta.json
#######
[["RUN_NUMBER","START_TIME","STOP_TIME","DETECTOR_ID","RUN_TYPE","SOFTWARE_V
ERSION"],[[12000,"Thu, 04 Nov 2021 19:51:56 GMT","Thu, 04 Nov 2021 19:53:32
GMT","np02_coldbox","PROD","dunedaq-v2.8.1"]]]

#######
12000/tmpmzhogsum/top_config.json
#######
{
    "np02_coldbox_daq": "/nfs/sw/dunedaq/dunedaq-v2.8.1/configurations/np02_
coldbox_hsi",
    "np02_coldbox_wibs": "/nfs/sw/dunedaq/dunedaq-v2.8.1/configurations/np02
_coldbox_wibs"
}
#######
12000/tmpmzhogsum/np02_coldbox/np02_coldbox_wibs/boot.json
#######
{
    "apps": {
        "ctrl_wib401": {
            "exec": "daq_application",
            "host": "host_wibapp",
            "port": 3380
        },
        "ctrl_wib402": {
            "exec": "daq_application",
            "host": "host_wibapp",
            "port": 3381
        },
        "ctrl_wib403": {
            "exec": "daq_application",
            "host": "host_wibapp",
            "port": 3382
        },
        "ctrl_wib404": {
            "exec": "daq_application",
            "host": "host_wibapp",
```

The blobs in the UconDB contain:

- Info from a lot of files
- Info is stored differently depending on the file
- Not user friendly

Subset for Conditions DB

Conditions data

- Just a few parameters
- Stored in json format
- User friendly

Conditions data - json

- The goal is to have a list similar to a run history list

Parameters:

1. Run number
2. Run type
3. Start/Stop time
4. Gain
5. shaping
6. leakage
7. Pulser amplitud

Should we include data from other DBs? like hvfrac?

Summary and outlook

- New cron job was created to automatically transfer the run config blobs to the ucondb
  - Deploy in .cron folder
  - Status - testing
- New folder and object were created in the UConDB
  - First run in there is: 11902
  - Runs up-to-date
- New parameters were suggested to include in the run configuration subset
  - I know where to get the information and how to interpret it
  - I will work on including these parameters
  - Create another cron job for this transfer

Thank you

Colorado State University

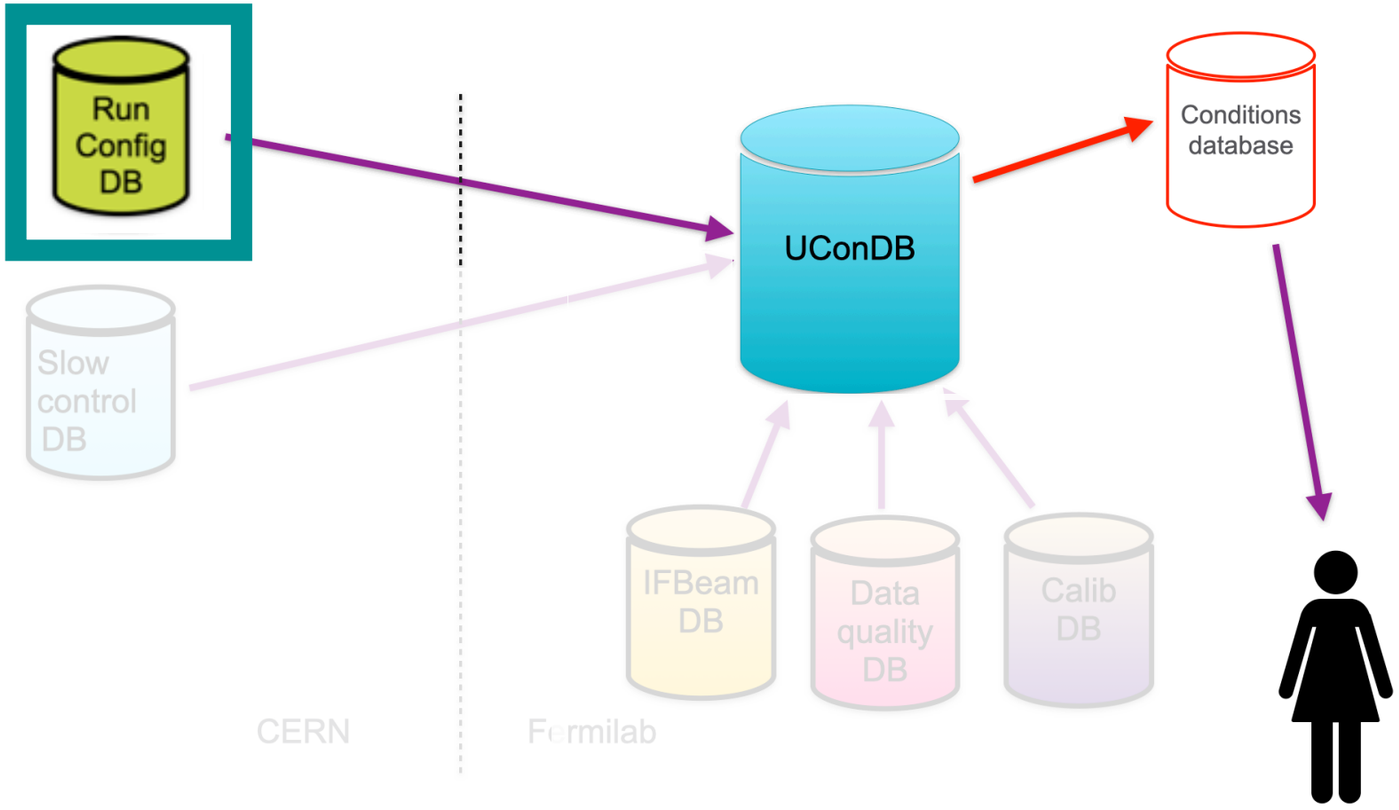
Backup slides

Run Configuration DB

- New method to extract the metadata from the run configuration DB and send it to the UConDB
- Subset of data is then send to another database with a more useful format

Updates

- Transfer data automatically after runs
- Select more data for the ‘conditions DB’



Conditions data - more parameters

David Adams pointed out some parameters that are needed as conditions data, and that can/will be found in the run configuration blobs:

1. Run number
2. APAs: will eventually be available, once the APAs are in the cryostat.
3. Gain
4. Shaping
5. Leakage
6. hvfrac, Slow control setting
7. Pulser Amplitud
8. Pulser source, not yet available but just one has been used

- Available in run config
- Not available in run config

I will add the parameters that come from the run config DB, but I don't think we should add info from a spread sheet that is filled by hand

Run Config blobs sent to UConDB

The new Run Config - UConDB blobs contain:

- Run number and record of creation
- Metadata information
- Name of config files with path:
  - Front end electronics configuration files (wibs files)
  - DAQ run configuration files (DAQ files)

```
Start of Record
 Run Number: 12000
 Packed on Feb 08 03:57UTC

#######
12000/runMeta.json
#######
[["RUN_NUMBER","START_TIME","STOP_TIME","DETECTOR_ID","RUN_TYPE","SOFTWARE_V
ERSION"],[[12000,"Thu, 04 Nov 2021 19:51:56 GMT","Thu, 04 Nov 2021 19:53:32
GMT","np02_coldbox","PROD","dunedaq-v2.8.1"]]]

#######
12000/tmpmzhogsum/top_config.json
#######
{
    "np02_coldbox_daq": "/nfs/sw/dunedaq/dunedaq-v2.8.1/configurations/np02_
coldbox_hsi",
    "np02_coldbox_wibs": "/nfs/sw/dunedaq/dunedaq-v2.8.1/configurations/np02
_coldbox_wibs"
}
#######
12000/tmpmzhogsum/np02_coldbox/np02_coldbox_wibs/boot.json
#######
{
    "apps": {
        "ctrl_wib401": {
            "exec": "daq_application",
            "host": "host_wibapp",
            "port": 3380
        },
        "ctrl_wib402": {
            "exec": "daq_application",
            "host": "host_wibapp",
            "port": 3381
        },
        "ctrl_wib403": {
            "exec": "daq_application",
            "host": "host_wibapp",
            "port": 3382
        },
        "ctrl_wib404": {
            "exec": "daq_application",
            "host": "host_wibapp",
```

Conditions data - more parameters

1. Run number
2. APAs: will eventually be available, once the APAs are in the cryostat.
3. Gain
4. Shaping
5. Leakage
6. hvfrac, Slow control setting
7. Pulser Amplitud
8. Pulser source, not yet available but just one has been used

The correct interpretation of each parameter can be found in:

/nfs/home/alyankel/gen_coldbox_configs/gen_coldbox_configs.py