# Table of Contents

FASER Databases.................................................................1
  Database Meetings..............................................................1
  Database Conventions..........................................................1
  Database Keys.................................................................1

Online Configuration..........................................................1
  Trigger and DAQ Configuration..............................................1
  Tracker Configuration.........................................................2
  Calorimeter/Scintillator Configuration....................................2
  Cabling Maps............................................................................2

Online Conditions....................................................................3
  Configuration conditions.....................................................3
  Online monitoring...............................................................3

Offline Databases....................................................................3
  Geometry Database............................................................4
  Magnetic Field.........................................................................4
  Alignment................................................................................4
  Tracking Reconstruction.......................................................4
    Masked strips.......................................................................4
    Noisy strips.........................................................................4
  Calorimeter Reconstruction..................................................5
  Scintillator/Trigger Reconstruction........................................5
  Luminosity and Machine Conditions.........................................5

Metadata Databases................................................................5
  Logging Database...............................................................5
  Database................................................................................5
  Dataset Database.....................................................................5
FASER Databases

This page is intended to collect information and plan the implementation of the databases used in FASER.

Database Meetings

* Kickoff 25 Nov 19

Database Conventions

Database Keys

FASER does not plan to have formal 'Luminosity Blocks' like the ATLAS experiment. FASER will certainly have something akin to run numbers, but only in certain situations will these be suitable to use as keys in databases. The most common keys will be time stamps.

Online Configuration

These databases are used in the online environment, and are primarily used by RunControl, Trigger, or EventBuilder to configure the online system. Need to clarify whether all configuration will come from a database (via one or multiple keys) or whether some of the configuration parameters can be set via GUI or other local file which then needs to be logged into a configuration database to record the actual configuration.

Trigger and DAQ Configuration

Most (all?) trigger configuration will be done via JSON files. These can readily be stored in any DB instance as CLOBs. Whether all configurations are combined into a single configuration object, or whether different components can be configured independently needs to be discussed. MongoDB has been mentioned as a way to manage these configurations. Also need to decide whether all configurations are read from a database, or whether all configuration parameters (which could come from a local JSON file) are logged into a configuration DB instance at the start of each run. Allowing ad-hoc configurations with comprehensive logging is more flexible, particularly during commissioning, but also more prone to errors.

Trigger Logic Board Configuration

- Trigger input timing delays
- Trigger coincidence logic
- Random trigger rate
- Trigger Prescales
- Trigger veto parameters, readout busy length, BCR veto, and rate limiter (are these configurable?)
- Readout delay (time between L1A and request for data from detectors)
- Orbit delay

Clock and Orbit Signal Board Specifications

While it is possible to configure the orbit signal board over USB, this is only for commissioning purposes. Once a stable configuration is found, the configuration will be stored in ROM. There is no routinely configurable parameters.

Run Control Specifications

In addition to managing the configuration of the rest of the DAQ system, the RC specifications document mentions that the RC may itself have configurable parameters, although what these might be isn't very clear. Some possibilities may be participating detectors, logging parameters, process control parameters, and other parameters related to interacting with the other DAQ components.
Event Builder Specifications

Event Buffer Specifications

Tracker Configuration

Chip configuration data following ATLAS folder /SCT/Config/Chip. Includes various chip parameters and a bad strip mask:

```
<channel id="40200013" insertionTime="Fri, 19 Oct 2018 09:35:08 GMT" since="1539941708000000000" until="9223372036854775807">
  <value name="chip">0</value>
  <value name="active">1</value>
  <value name="address">32</value>
  <value name="config">8222</value>
  <value name="mask">2294967295</value>
  <value name="mask1">2294967295</value>
  <value name="mask2">2294967295</value>
  <value name="mask3">2294967295</value>
  <value name="vthr">47</value>
  <value name="wcl">128</value>
  <value name="delay">15</value>
  <value name="preamp">24</value>
  <value name="shaper">25</value>
  <value name="rr_function">3</value>
  <value name="rr_arg">p0 2.240813e+03 pl 1.214277e+01 p2 -1.049019e+03</value>
  <value name="c_factor">1.026</value>
  <value name="target">117.5</value>
  <value name="trim">IIGJDFKKG0JNClD1MKJBEHKIDHG81GJGJLBLLD1G1GGGWFGGBIK1IJKIF1H1I1N1JF1F1GGFGKGBMGKHEIJJPJHFG8H1I11LLJ7G1G1JJ1F8BJLJ</value>
</channel>
```

This chip configuration data comes from an online calibration job, and in ATLAS problems in that calibration are written to two different 'defect' folders: /SCT/DAQ/Calibration/NPtGainDefects and /SCT/DAQ/Calibration/NoiseOccupancyDefects. An example of the /SCT/DAQ/Calibration/NPtGainDefects is given here:

```
<channel id="136320736" insertionTime="Wed, 10 Oct 2018 12:47:35 GMT" since="1560456107917331" until="9223372036854775807">
  <value name="defectParameter">0 0 0 0 2849.15 2622.25 </value>
  <value name="scanNumber">19</value>
  <value name="runNumber">363222</value>
  <value name="serialNumber">22020240108597</value>
  <value name="defectBeginChannel">343 814 841 1062 66 70</value>
  <value name="defectType">1 1 1 1 33 15</value>
  <value name="defectEndChannel">343 814 841 1062 66 70</value>
</channel>
```

For FASER we may be able to simplify the online calibration process, particularly as the tracker isn't used in the trigger, so hot channels are only an issue of data volume.

Calorimeter/Scintillator Configuration

Aside from DAQ parameters for the waveform digitizer, are there any online configurations needed here?

Cabling Maps

Several people have stressed the need to record cabling maps. It isn't 100% clear to me why this is needed. I don't believe this is configurable in the TLB, and we have such a small number of connections that I would think the chance of changing the cabling over the course of Run3 is very small. Also, there is no way to automatically record this information, so it relies on accurate user input, which isn't very reliable. Perhaps I am missing something here.
Online Conditions

These databases are written by the RunControl, EventBuilder, or EventBuffer online processes to record the conditions during data taking. In general, online conditions do not need to be versioned, as this records the actual conditions at the time the data was taken.

Configuration conditions

While I hope that most of our configuration parameters can be loaded from a DB in the first place, the configuration keys used still need to be logged. In addition, any parameters that can be changed outside of a pre-specified DB entry needs to be logged. Many of these items should be able to be logged with simple run number key. Trigger pre-scales and random trigger rates may actually be changed during a run, so these may need to be logged with timestamp keys.

Online monitoring

This includes both DCS monitoring and DAQ monitoring quantities.

DCS

- All detector voltages and currents
- Temperatures and other environmental monitoring quantities
- Chiller parameters
- Other infrastructure, such as PDU current draw, voltages

The DCS system will use WinCCOA which I believe includes its own logging facilities (to be verified) or an interface to influxdb. The challenge here is to understand which DCS quantities may be needed in other contexts (particularly offline reco, but also offline monitoring and performance studies) and the best way to access information stored in the DCS logging system.

The remaining online monitoring is expected to be stored in influxdb keyed by time. Understanding how to get access to this data from elsewhere is similarly important.

TLB

- Trigger rates (TBP/TAP/TAV counters)
- Deadtime counters

These quantities are sent out from the TLB as a special data type. The EventBuilder will actually log this information.

EventBuilder/EventBuffer It is expected that the EventBuilder process will log standard information such as average event rates, event sizes, error rates, missing fragments, and so on. The EventBuffer will also want to internally keep track of transfer rates and other quantities relevant to monitoring dataflow. See the Metadata section below for other information that needs to be logged outside influxdb for keeping track of data integrity.

Offline Databases

Offline databases are primarily used in the simulation and reconstruction of data in offline processes. These databases need to be accessible from remote locations.
Geometry Database

GeoModel description of detector used in Geant simulations. Currently uses XML files which could just be provided as part of any release. Tools to read/write and visualize this information are available through the ATLAS tools as well as additional work done by Dave Casper's group.

Magnetic Field

Magnetic field maps consist of magnetic field vectors on a grid of cartesian coordinates. At minimum FASER needs one map for simulation and one for the real detector. These maps will likely change very rarely, and using an external file (in some appropriate format for efficient reading into the reco or simulation environments) distributed with the release might be a good solution.

Alignment

A hierarchy of alignment information will need to be provided for both simulation and reconstruction. In general, each element has 6 free parameters with respect to some reference coordinate system. Obvious elements that need to be separately determined are

- Tracking planes - each of 9 frames attached to the strongback that each holds 8 strip modules
- Tracking modules - each double-sided sensor aligned with respect to the frame that holds it
- Tracking sensors - each individual sensor has some alignment (including the stereo angle) compared to the module coordinate
- Calorimeter modules - each of the 4 calorimeter module blocks

It is unlikely that all of these parameters need to be varied in an alignment procedure. In particular, the sensor position on the modules may be very well understood, and the initial alignment accuracy of the modules in each plane may be better than local alignment using sensor overlaps can determine.

One key question here is what determines the global FASER coordinate system?

Tracking Reconstruction

Masked strips

Information about masked strips will need to be read from the online configuration, or an offline replica of that information.

Noisy strips

In ATLAS, noisy strips are identified in the calibration loop and information is written to the /SCT/Derived/Monitoring folder:

```
  &<channel id="130784768" insertionTime="Sat, 10 Sep 2016 11:36:36 GMT" since="132321070420864" until="922337203685477" 99000</value>
  &<value name="BarrelEndCap">-2</value>
  &<value name="Layer">0</value>
  &<value name="Phi">45</value>
  &<value name="Defect/Type">MARG</value>
  &<value name="Threshold">6.015</value>
</channel>
```

This information is used in reconstruction to ignore certain strips.
Calorimeter Reconstruction

The calibration function for each PMT is the only obvious reco condition needed here. This may vary with time, so having this properly in the database from the start is probably a good idea.

Scintillator/Trigger Reconstruction

Not obvious that there are any configurable parameters here.

Luminosity and Machine Conditions

This information will be taken from ATLAS. If we decide we need a replica of this information in the FASER DB (particularly keyed by FASER time) we can arrange to have this copied over. Information about which BCIDs contain colliding bunches will need to be included in the reconstruction process, as the FASER DAQ likely will not know the bunch structure.

Metadata Databases

Metadata databases store information about the data samples themselves.

Logging Database

Database

Dataset Database

-- EricTorrence - 2019-12-06