Belle II Conditions Database

(the very brief version !)

Paul Laycock

BROOKHAVEN NATIONAL LABORATORY

U.S. DEPARTMENT OF ENERGY

Belle II Conditions Model

- Data Model: relational DB

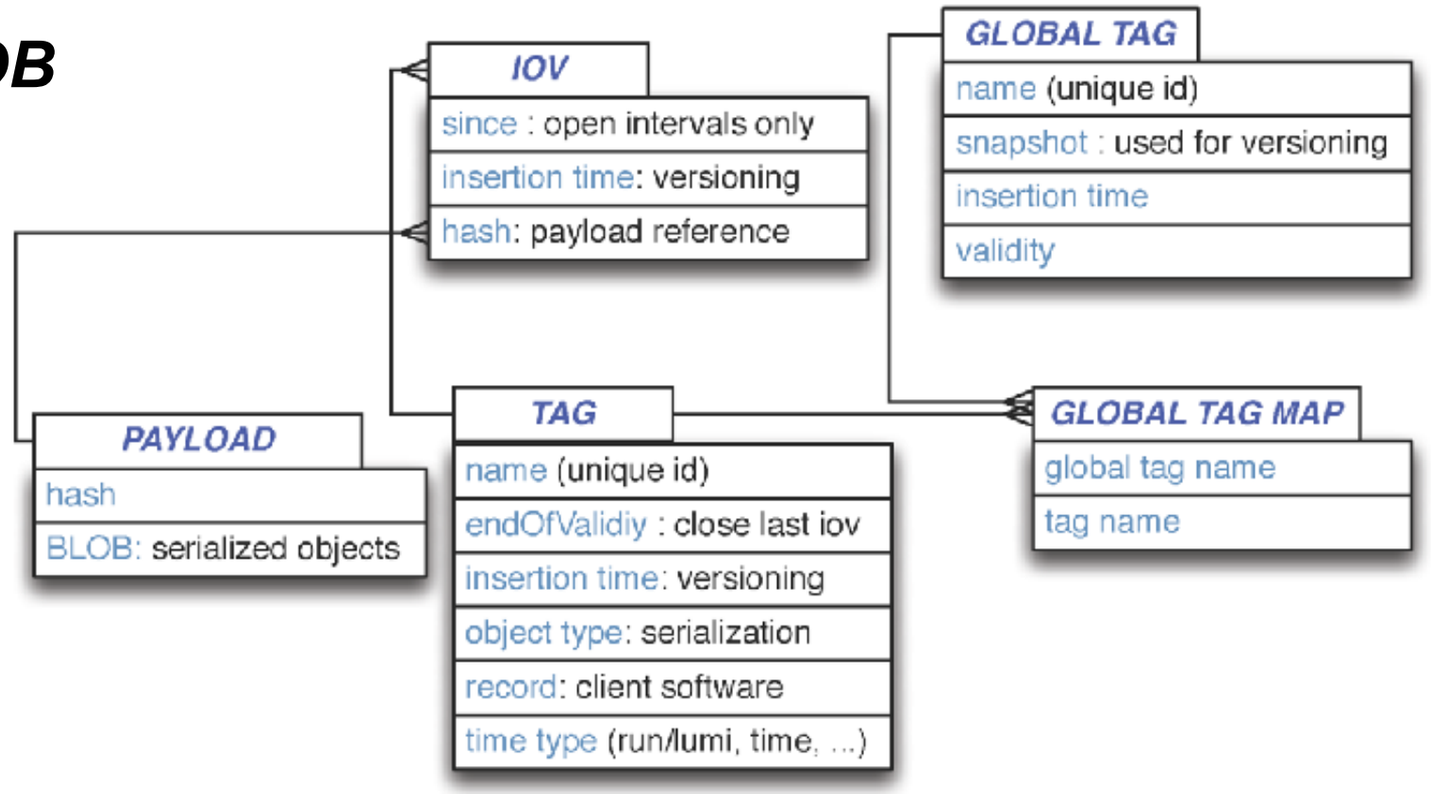
https://belle2db.sdcc.bnl.gov/b2s/rest/v2/iovs/?gtName=B2BII&runNumber=6

- Single tables for payload, tags, IOVs
  - ***Payloads can be separated completely from metadata, only need a reference in the DB***
  - *Largely experiment agonistic*

- *List of IOVs and payloads retrieval are independent*

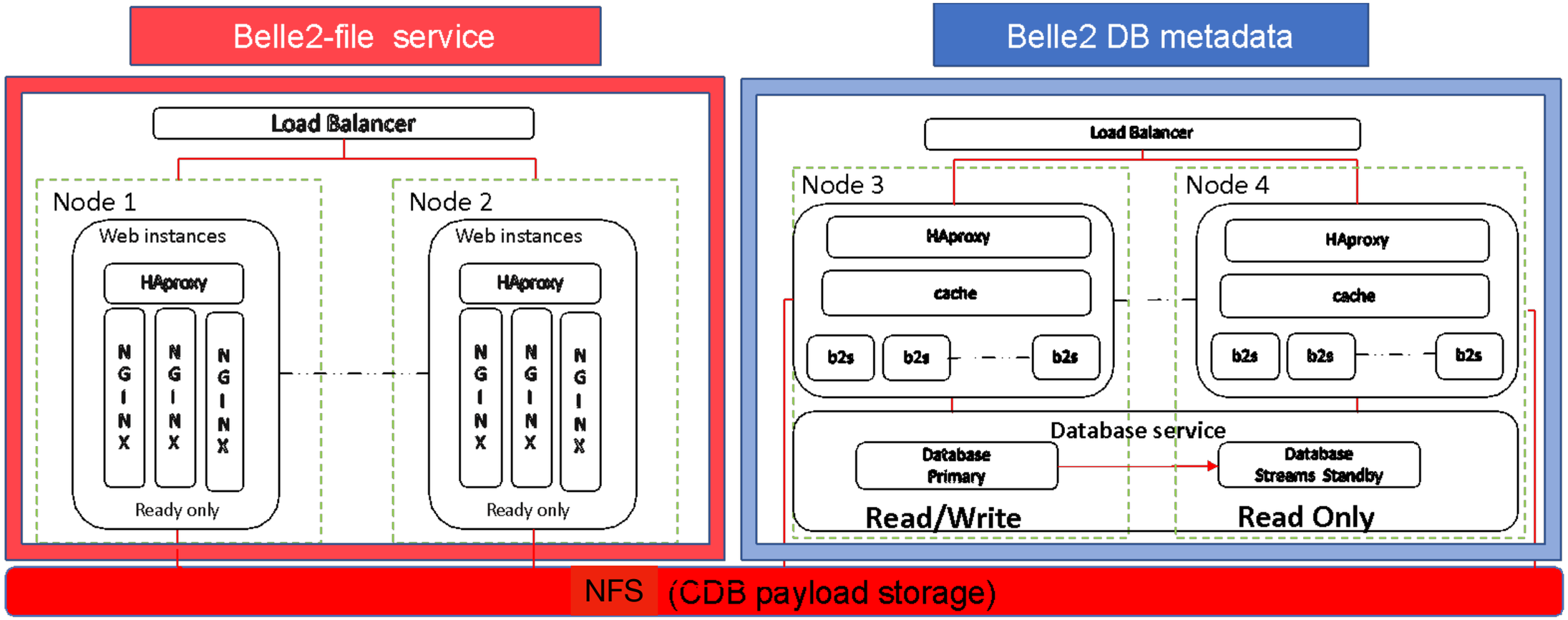
- ***Cache-friendly design***

- Largely follows best practice principles in **HSF CWP** paper:
  - ***https://arxiv.org/abs/1901.05429***



Data model from CWP paper

Belle II Conditions Service



- Payloads are retrieved from the file service (left), but are preferentially taken from a local cache (if already used) or from ***cvmfs*** - for Belle II this means there isn't much traffic for the file service, may be different for **sPHENIX**
- The metadata service is a relational DB but as it is ***only*** the metadata it is small, < 3 GB for Belle II which includes all Belle data as well
- These features for Belle II led to the SDCC proposal to use VMs
  - Carlos has tested that also the VM deployment copes with 100s Hz rates, Belle II compute scale is order ***40k*** parallel jobs

Belle II Conditions Framework Service

ACAT2017 IOP Publishing
IOP Conf. Series: Journal of Physics: Conf. Series **1085** (2018) 032032 doi:10.1088/1742-6596/1085/3/032032

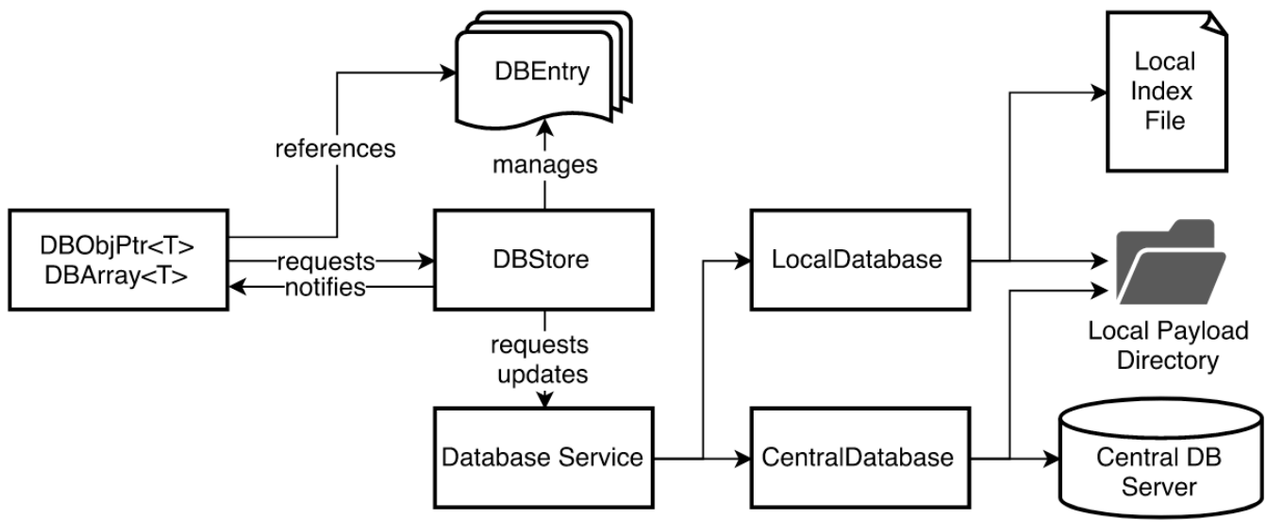
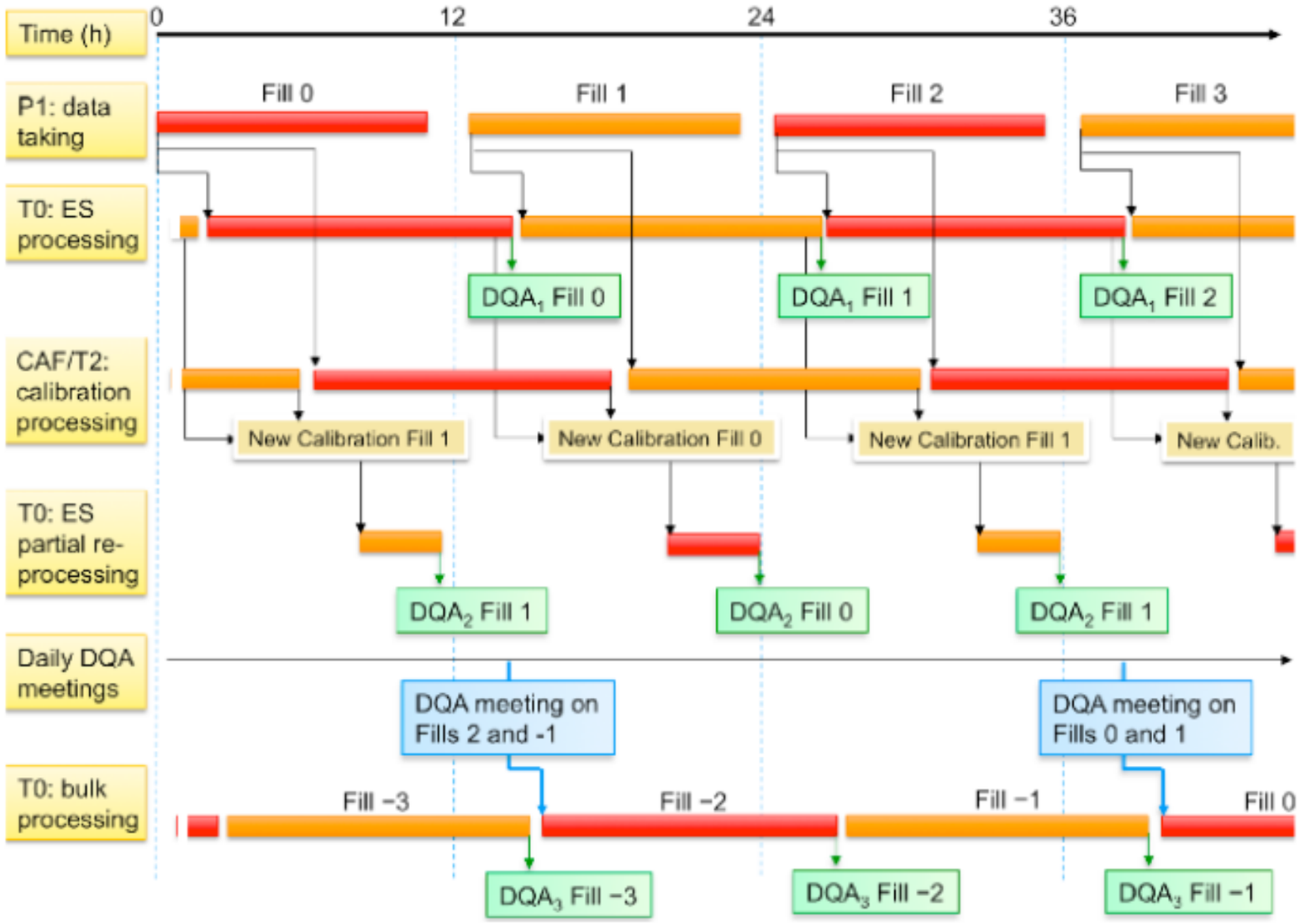


Figure 3. Relationships for the Conditions Database Interface. The user only interacts with the `DBObjPtr` and `DBArray` classes, everything else is handled transparently and can be configured independently.

- On the client (framework) side, the user interacts with DBObjPtr/DBObjArray
- Multiple instances point to the DBEntries kept in the DBStore
- So… global tag for configuration: **GlobalTag = “FinalFinalBestCalibration2040”**
  - resolves to payload-type tags for every payload used in reco
- Reco module asks a ConditionsService for payload-type and gives a timestamp read from the event
  - **cdbSvc->Get(“MyCalibrationType”, Run_number)**

Conditions management



- Organising the conditions to be used in offline processing can be a real bottleneck to delivering quality physics, if you ***may*** need versioning then build it in and don't assume you won't!
- Running global tags are also really crucial!!