### Table of Contents

DB populating tools (Online-to-Offline)...........................................................................................................1
  Current O2O status and requirements...........................................................................................................1
  ECAL.................................................................................................................................................................1
    Synchronization............................................................................................................................................1
  HCAL...............................................................................................................................................................1
    Synchronization............................................................................................................................................2
  Tracker Strip..................................................................................................................................................2
    Synchronization...........................................................................................................................................2
  Tracker Pixel..................................................................................................................................................2
  CSC...............................................................................................................................................................3
  DT.................................................................................................................................................................3
  HLT...............................................................................................................................................................3
  L1T.................................................................................................................................................................3
  RPC...............................................................................................................................................................3

Architecture proposals and decisions................................................................................................................3
  Meeting 06 Jun 2007......................................................................................................................................3
DB populating tools (Online-to-Offline)

Current O2O status and requirements

ECAL

Current conditions objects utilizing online-offline transformation are

- EcalPedestals
- Laser objects
- Trigger primitive generator objects

The read-back of the objects employs the usage of the complex-multi-join sql query. IOV is obtained via SQL query.

Previously Ricky Egeland's O2O Perl & PL/SQL framework was used to transfer the objects.

Currently CMMSW OCCI application has been developed to provide an interface to the online database.

Synchronization

Synchronization & bookkeeping mechanism has not yet been implemented. Previous method compared last_id from the offline to run iov information from the online database and transferred the objects corresponding to the newer values. Old o2o bookeeping involved offline database table holding information about object name, last iov id and time of the transfer.

Offline DB interface utilizes PoolDBOutputService

Main requirements are:

- Provide a bookkeeping interface
- Provide a solution to effectively synchronize orcon and data sources

HCAL

Online Database

- Contains construction, calibration, configuration and monitoring data.

O2O data (transferred after each run)

- Pedestals
  - calculated by semi-automatic online process (analyzer)
  - before and after each run
  - triggered by Run Control or Shifter

transferred rarely

- QIE response curves (once)
- Gains (several times a year)
- e-map (several times a year)
Object and IOV information can be obtained via materialized views. Interface to the online DB implemented in OCCI.

We’ve heard contradictory information about current Offline DB access - PoolDBOutputService?

Target solution - DBOutputService

**Synchronization**

- no response yet

**Tracker Strip**

Infrastructure based on the CmsRun

To access the online DB a CMMSW library based on Oracle Call Interface is used.

Writing made by EDAnalyzer using PoolDBOutputService

Configuration and execution of the o2o is made using shell scripts

- A script which checks if o2o is needed (monitor)
- Actual o2o script configured with runnumber

IOV management in append mode

O2O is performed only on conditions change. One needs to discover the runs corresponding to the change in the configuration. In the Online DB there’s a field (flag) stating if o2o was done for the given runs, the flag is however not used.

In the online DB the runnumber has nothing to do with the real version. version_id defines the evolution of set of data to be uploaded

**Synchronization**

1. Monitor script gets the list of physics runs
2. For each run check if the configuration in Online is compatible and version has changed
3. For each run the last IOV is compared to the run number

- this determines if the run is within the closed IOV / IOV is old

**Requirements**

- bookkeeping, probably involving the changes in the synchronization model
- Currently there’s no bookkeeping in the offline database, should be stored in the offline DB

**Tracker Pixel**

It is not yet known if online to offline transformation would be required. If so, probably only gains would be transferred to offline database and most probably the implementation would be based on the SiStrip code.
CSC

Current state

CSC has been using Ricky Egeland's O2O framework

Currently 4 conditions objects are subject to o2o

CSCgains CSCNoiseMatrix CSCPedestals CSCCrosstalk

As it turned out, CSC calibration requirements can be fully satisfied with offline software infrastructure using EDAnalyzer, EDProducer and DBOutputService. Therefore there is no need to involve the online database in the process.

Oana has already developed a tool which transfers the data from fake source to the onffline DB using CmsRun infrastructure

DT

Calibrations are computed directly from data so no transfer from OMDS is needed. Configurations are stored as ASCII strings, there is a need to store this data in the ORCON-ORCOFF DB. If the configuration has not changed, only the config. identifier is being transfer for each run, otherwise the actual data are copied. Currently the SQL procedure is being developed to conduct the transfer.

HLT

Under Development. No requirements for the O2O yet.

L1T

There's a need to store the configurations data for trigger emulation. Current setup does not employ the usage of online and offline DB. Theres been a request to provide an example how the online DB should be accessed. Also synchronization questions have been asked

• when the transformation should be run
• how to synchronize ORCON populating software

Final setup will transfer configurations from OMDS to ORCON. Since the payload is not conditions data, some IOV problems have to be resolved.

RPC

Not contacted yet

Architecture proposals and decisions

Meeting 06 Jun 2007

Vincenzo's summary of decisions :

A. The condition offline database (ORCON/ORCOFF) will be populated ONLY using a cmsRun application based on a standard example/template very close to the current DBCondOutputService
example (Zhen please give me the correct name + point me to the example)
  a. we will not support anymore "poolification" of raw RDBMS table
B. The management of the access to the data source (OMDS or other) is responsibility of the detector group
  a. whenever possible the reading code should be shared with other online applications that needs
     access to the very same data
C. data NOT required online (HLT is an offline application in this context) shall not be written to OMDS
   only for the sake of O2O. They shall be written directly to ORCON.
D. Synchronization between data sources and ORCON shall be better understood to guarantee that
   a. no data are omitted
   b. no data are written twice
   c. time corresponds to the correct data
   d. the time sequence is respected

-- MarcinBogusz - 08 Jun 2007