Conditions Database

Experiments reports at the Database Readiness Workshop

Andrea Valassi

(CERN IT-PSS)

Overview

- **Conditions data**
  - Non-event detector data that vary with time
    - And may exist in several versions (e.g. calibration)
  - Typical query: 'what is the calibration set (computed using the v.215 algorithm) to be used to reconstruct run #347?'

- **Closer to experiment software than to middleware**
  - The conditions data model reflects the event processing model specific to each experiment
  - *Different software choices in the 4 experiments*
    - *Implying different service requirements*

Conditions database software in the 4 experiments

- **ALICE**
  - Alice-specific software for time/version handling
  - ROOT files (via AliROOT)

- **CMS**
  - CMS-specific software for time/version handling
  - Oracle (via POOL-ORA C++ API) with Frontier web cache

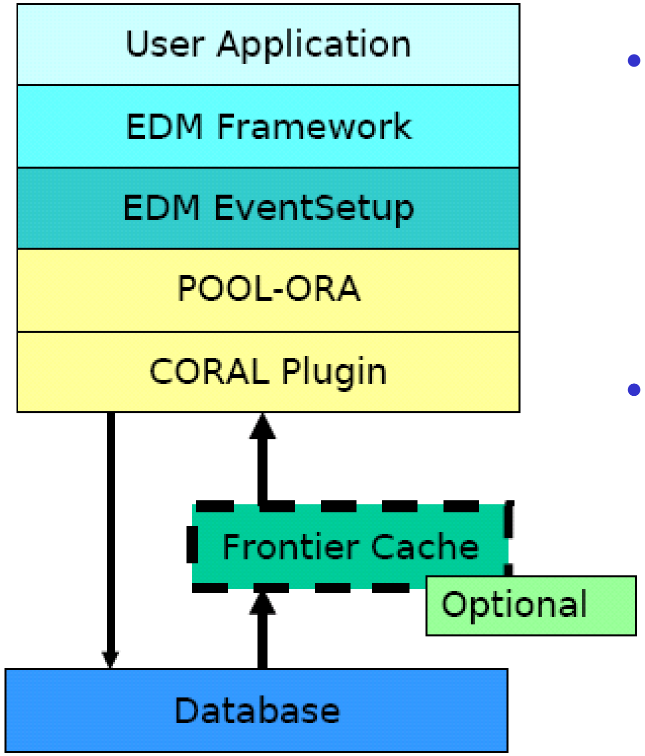
- **ATLAS and LHCb**
  - COOL (LCG AA common software) for time/version handling
  - Oracle, MySQL, SQLite (via COOL C++ API)

ALICE

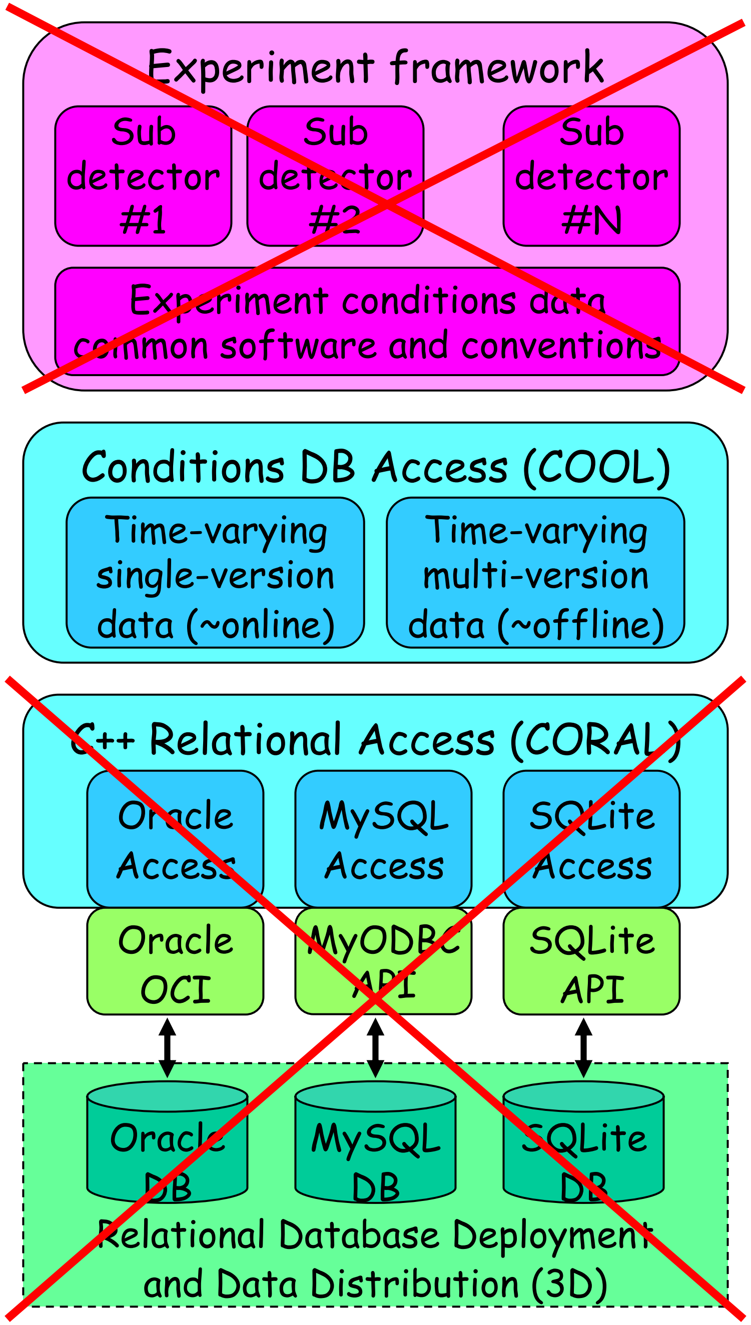
- **Conditions data stored as read-only ROOT files**
- **ROOT files registered in** *Grid Distributed File Catalogue*
  - *No need for distributed DBMS* in a traditional sense and with all accompanying problems (access, replication, authentication)
  - Drawback: replication of information – potentially large parts of 'external' DBs have to be put in the offline conditions DB
- **AliROOT implementation**
  - Classes to put/get conditions ready and tested by all detector groups
  - GRID file and metadata storage (AliEn GRID File catalogue) deployed
  - Conditions data dump in local files (expert/disconnected use) deployed
- **Tests in 2006 – with real data and as part of PDC06/SC4**

CMS

- **Conditions database applications: alignment/calibration**
  - Test in May 2006 for MTCC (Magnet/Cosmics), not in SC4
  - Oracle service needed at P5 and T0 (with Oracle streams in between)
  - Frontier deployment at T1



- **Both metadata and payload stored in Oracle via POOL-ORA**
  - POOL object-relational access layer
  - Internally using CORAL Oracle plugin
- **Remote r/o access via Frontier web cache**
  - Frontier server (close to DB server) reads data from Oracle and encodes it as http
  - Frontier clients send SQL queries as http requests, which can be cached (Squid)
  - Frontier (CMS/FNAL) is now a CORAL plugin


Experiment framework
Sub detector #1
Sub detector #2
Sub detector #N
Experiment conditions data common software and conventions
Conditions DB Access (COOL)
Time-varying single-version data (~online)
Time-varying multi-version data (~offline)
C++ Relational Access (CORAL)
Oracle Access
MySQL Access
SQLite Access
Oracle OCI
MyODBC API
SQLite API
Oracle DB
MySQL DB
SQLite DB
Relational Database Deployment and Data Distribution (3D)

COOL *common* software

NOT the problems specific to one experiment or one data type *(handled by each experiment)*

Software for time-varying and versioned data: a common component with a well-defined task

(RDBMS implementation of technology-neutral API)

NOT the generic C++ access to relational data *(handled by CORAL)*

NOT the generic deployment of relational services and distribution of relational data *(handled by 3D – at CERN by IT-PSS)*

COOL software overview

- **New software merging ideas of previous packages**
  - Separate Oracle/MySQL implementations of similar APIs
- **Single implementation for many relational backends**
  - Encapsulated behind C++ API (no direct SQL user access)
  - Support for Oracle, MySQL and SQLite via CORAL
    - Attention to Oracle performance (bulk operations, bind variables…)
- **Maximize integration with other LCG projects**
  - Reduce duplication of effort with SPI, SEAL, POOL…
  - *Excellent and mutually beneficial collaboration with CORAL*

COOL development milestones

- **Nov 2004: start of COOL software development**
  - Brand new code, initially ~1.3 FTE for development
- **Apr 2005: first COOL production release 1.0.0**
  - Support for Oracle and MySQL through POOL RAL
  - Basic insertion/retrieval (single/multi-version, single/bulk)
- **Oct 2005: Atlas use case performance validation**
  - One job every 5s, each retrieving 100 MB in 100k rows
- **Latest of many releases: COOL 1.2.8 (Jan 2006)**
  - SQLite (July), CLOB, PyCool, multi-channel bulk ops (Aug.), performance tests (Oct.), data copy (Nov.), CORAL (Jan.)
  - Team is now ~2 FTE for development plus ~1.5 FTE for tests (all from IT/LCG, Atlas, LHCb)

COOL - ATLAS deployment

- COOL fully integrated into Athena since mid-2005
  - Small payload stored 'inline', complex payload as POOL refs
  - Development priorities: CORAL API, schema evolution, hierarchical tags

- Still data/software based on Lisbon MySQL implementation
  - Transition phase: complete migration to COOL by mid-2006

- Priorities for tests and deployment in 2006 ("CSC" and SC4)
  - Full software chain including calib/align and condition data distribution
    - T0 "closed loop" tests (Mar-Sep): reconstruct, improve calibration, re-reconstruct
    - Distribution to T1 (June) using 3D Oracle streams and then to T2 (October)
  - Data volume estimates for calibration/alignment not yet finalized
  - Model includes upload into T0 master of calib/align computed at T1/T2

- Replication strategies
  - COOL API-level static replication (Oracle->SQLite) available and used now
  - Explore T0-T1 dynamic replication via Oracle streams in 3D whenever ready
  - COOL API-level 'dynamic' replication (updating replicas) needed from July
  - Evaluate Frontier replication from February
  - *Experience in Q1-Q2 will guide replication choices for T2 tests in Q3-Q4*

COOL - LHCb deployment

- **COOL fully integrated into LHCb Gaudi framework**
  - Ongoing developments for subdetectors' concrete data model definition
  - COOL team requests: improve tagging (functionality and performance)
- **Conditions DB (COOL) one of many databases**
  - COOL holds conditions data for reconstruction/analysis
  - Other data in PVSS, LFC, Bookkeeping, Configuration DBs
- **Deployment model (online and offline)**
  - Masters (r/w) at the pit and CERN T0; Replicas (r/o) at T1
  - Oracle replication via Oracle streams (tests underway CERN <-> RAL)
- **Data challenge plans in 2006**
  - Alignment/calibration challenge in Oct (with all T1 sites)
  - SC4: only LFC will be tested, not COOL

COOL development plans

- **Improved functionalities and new features**
  - Major API and schema changes in 1.3.0 (~Mar-Apr 2006)
  - Improved tagging (user tags, hierarchical tags…)

- **Deployment-oriented new developments**
  - Continue T0 performance tests and SQL optimizations
  - Add CORAL connection monitoring, retrial, authentication
  - Improved replication strategies (Atlas 'dynamic' replication)
    - Prototype Frontier plugin?
  - *Collaboration with 3D and IT-PSS service teams is crucial*

- **COOL development is far from finished**

Summary

- **Conditions data software is different in the 4 experiments**
  - ALICE – ROOT files
  - CMS – Oracle with Frontier Web cache
  - ATLAS and LHCb – Oracle/MySQL/SQLite via COOL

- **2006 (and/or SC4) plans involving conditions data**
  - ALICE – PDC06/SC4 Grid tests using Alien file catalogue
    - No database service requests for these tests
  - CMS – Magnet-Test/Cosmic-Challenge (MTCC) in May, not in SC4
    - *Oracle service at P5 and T0 (with streams in between), Frontier at T1*
  - ATLAS – T0 closed loop tests, then distribution to T1/T2
    - *Oracle service at T0; streams/Frontier under evaluation (outside SC4)*
  - LHCb – No conditions database tests in the context of SC4
    - Self-contained functionality and performance tests ongoing