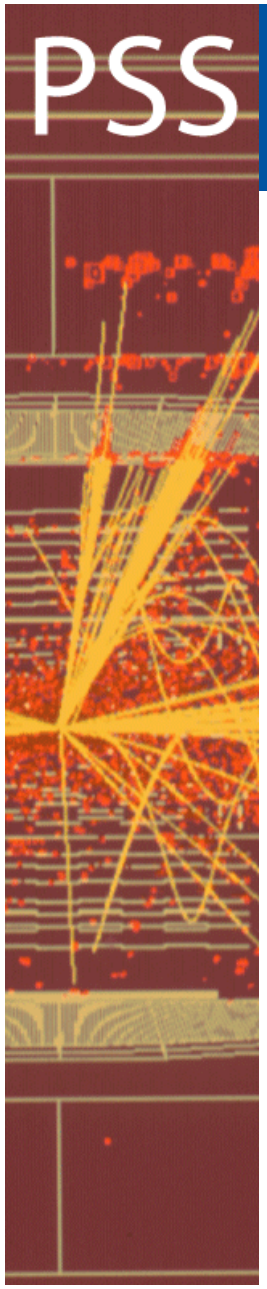


PSS Physics Services Support

CERN IT Department

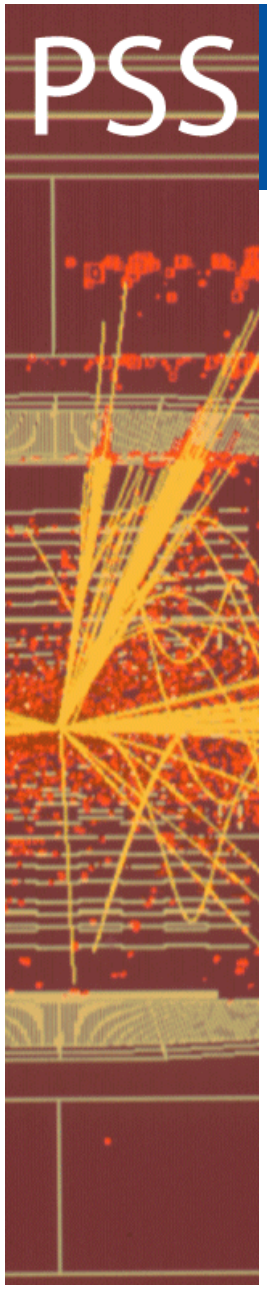
COOL – Status and Plans

(What's new in COOL 2.0.0)

Andrea Valassi (CERN IT-PSS-DP)

Database Workshop, 26 January 2007

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it



COOL development team

CERN IT Department

Active manpower

- **A.V. (CERN - IT/PSS)**
  - 80% FTE since Oct. 2004
  - Project coordination, core development and release mgmt
- **Marco Clemencic (CERN - LHCb)**
  - 20% FTE since Nov. 2004
  - Core development and release mgmt
- **Sven A. Schmidt (Mainz - ATLAS)**
  - 20% FTE since Oct. 2006
  - Previously 80% FTE (Oct. 2004 to June 2006)
  - Core development
- With useful contributions from many other people
  - David Front (IT/LCG since April 2005) - stress tests
  - Richard Hawkings and other ATLAS users, testers and DBAs
  - The CORAL, SEAL/ROOT, SPI and 3D teams

Former collaborators

- **Uli Moosbrugger (Mainz - ATLAS)**
  - 80% FTE (Sep. 2005 to March 2006) – performance optimization
- **Kristoffer Dahl (CERN - IT/PSS summer student 2006)**
  - 60% FTE (July-August 2006) - performance optimization

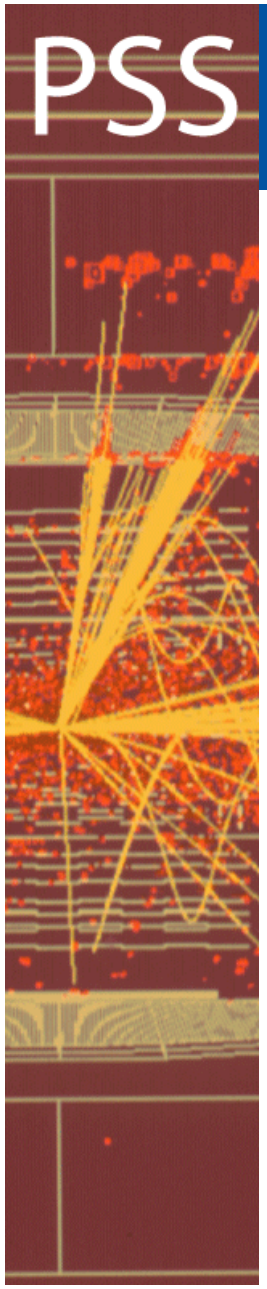
CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

CERN

Introduction

- **What's new in COOL 2.0.0**
  - ***Major API changes***
  - ***Major schema changes***
  - Extensive tests and bug fixes
    - Both internal and external (e.g. Frontier, SEAL…)
  - Configuration changes (CMT, MacOS…)
- **Plans after COOL 2.0.0**
  - Optimizations and new features with backward-compatible C++ API and no schema evolution

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it



C++ API changes

- **IRecordSpecification and IRecord**
  - Includes specification and checks on persistent storage precision (e.g. “Int64”, “String16M”)
  - Replaces coral AttributeListSpecification and AttributeList (transient rather than persistent)

- **SEAL-free C++ API**
  - No Time, IntBits, IMessageService in the API
  - Internal implementation still based on SEAL
    - Until component model is moved to CORAL

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

User payload changes

- **Added support for BLOB fields**
- **New constraints on # payload fields**
  - At most 900 fields (Oracle limit: 1000 columns)
  - At most 10 BLOB fields (MySQL limit: 15)
- **New constraints on payload field names**
  - Field names can only contain letters (uppercase and lowercase), numbers or the "_" character
    - Avoid bad SQL when using -.%$ and so on…
- **Empty string '' is considered as NULL**
  - Oracle 'feature': adopt it also for MySQL/SQLite

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Relational schema changes (1)

- **Get ready for schema with fewer tables**
  - Attach schema version to individual folders
    - COOL 2.0.0 cannot read/write folders > 2.0.0
  - Actual new folder schema with fewer tables will be added in a later release (2.1.0 or later)
    - Can be added *without requiring schema evolution*
    - COOL 2.0.0 clients will still be able to read/write 2.0.0 folders, while 2.1.0 clients can also use 2.1.0 folders
  - In 2.0.0 schema, each folder has 5 tables
    - In 2.1.0 schema, folders with the same user payload specification will be able to share the same 5 tables
    - SQL query optimizations also postponed to 2.1.0

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Relational schema changes (2)

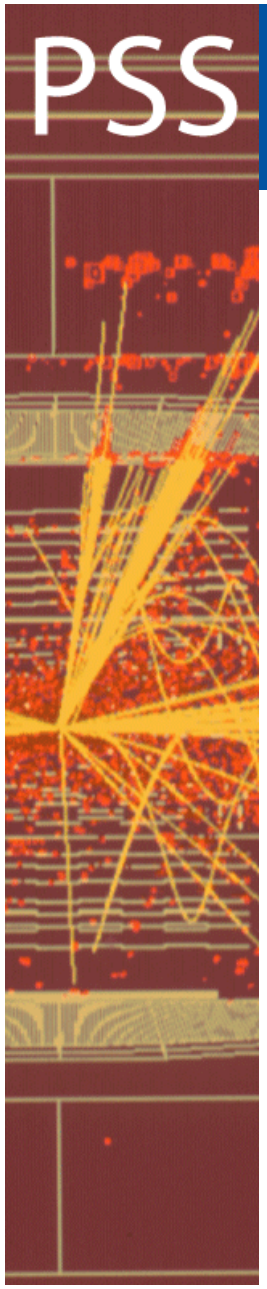
- **Get ready for tag locking functionality**
  - Add ‘tag lock status’ column
    - Implement protection for tags with lock status != 0
  - Actual lockTag/unlockTag API in a later release
    - Can be added *without requiring schema evolution*
- **New channels table for channel metadata**
  - Also needed for multi-channel bulk insertion
  - Now: channel name; later: user-defined?
- **Implement dynamic replication tool**
  - Add several ‘last modification date’ columns
- **Use default SQL types from CORAL**
  - Few minor changes only for MySQL and SQLite

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Platform support

- **AMD64**
  - Fully supported in COOL 2.0.0
    - Thanks to new IRecordSpecification API (ambiguity for 'long' precision in COOL 1.3.0 on 32bit/64bit Linux)
- **Mac**
  - Complete C++ private build – all tests succeed
    - Using private Coral and Oracle installations
  - Pending issues with PyCool – runtime errors
    - Incompatible ROOT/python installations

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it



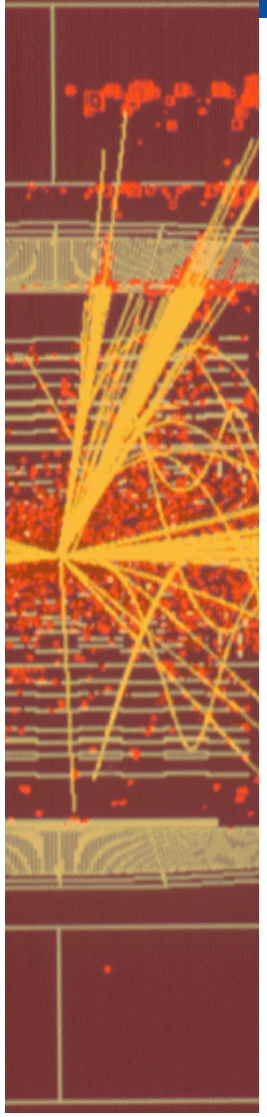
Build infrastructure

CERN IT Department

- **CMT**
  - COOL 2.0.0 will still be released using SCRAM
- **Nightly build system**
  - For shorter time-to-the-experiment
- **QMTEST test infrastructure**
  - Only sqlite is used in the nightlies so far

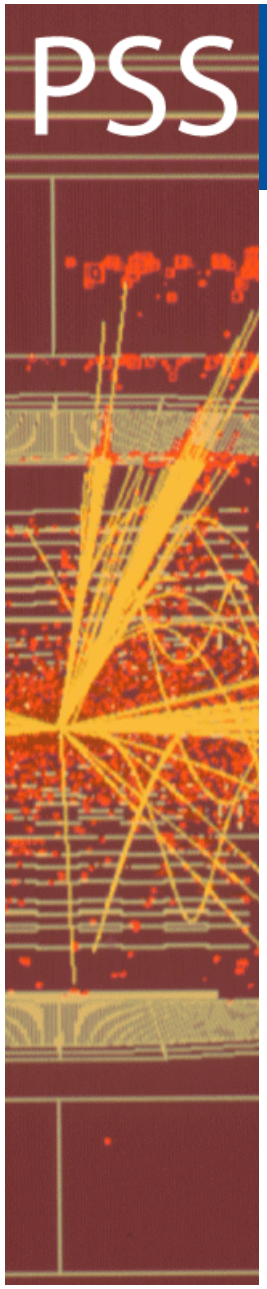
CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

COOL 2.0.0 release status



- **COOL 2.0.0 will be released next week**
  - Using the latest CORAL 1.7.0 (released today)
- **Still missing**
  - A few tables in view of the future 2.1.0 schema
    - Table listing (shared) IOV tables
    - Table listing (shared) channels tables
  - Catch a few errors before UK/PK are violated
  - A few FK in the top-level tables
  - Complete tests against the new CORAL170

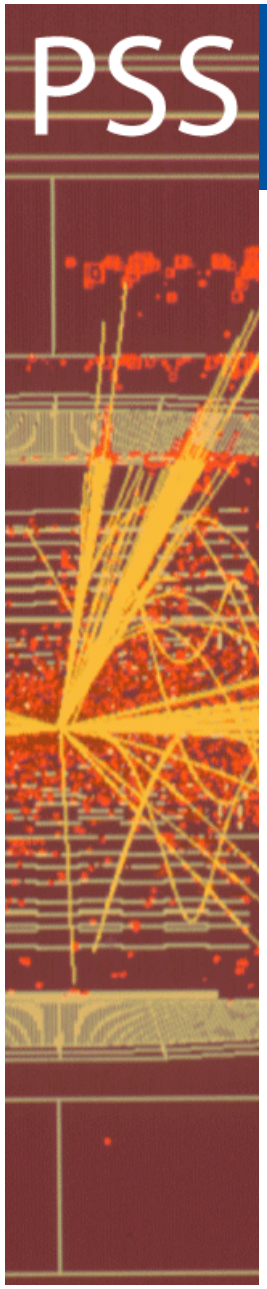
CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it



Plans after COOL 2.0.0

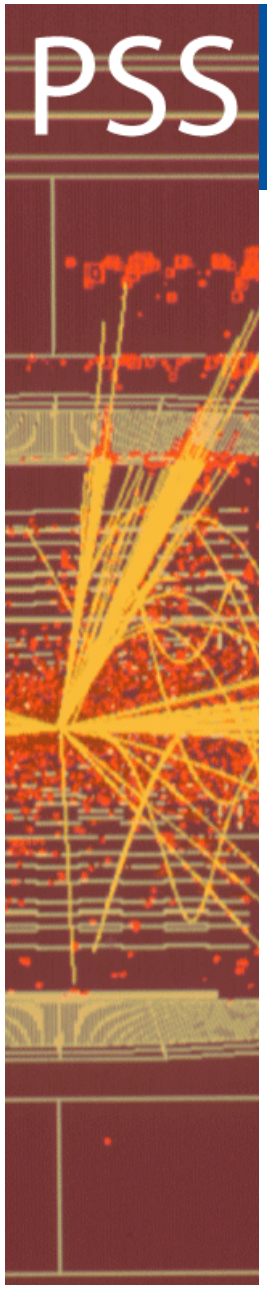
- **Tag locking functionality**
- **New relational schema with fewer tables**
  - Also: SQL query optimizations
  - Also: complete multi-channel bulk insertion
- **CORAL component model (drop SEAL)**
- **Integration with LFCReplicaService**
  - With role-based authentication
- **Payload queries**
  - Based on the new IRecord API

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it



Reserve slides

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it



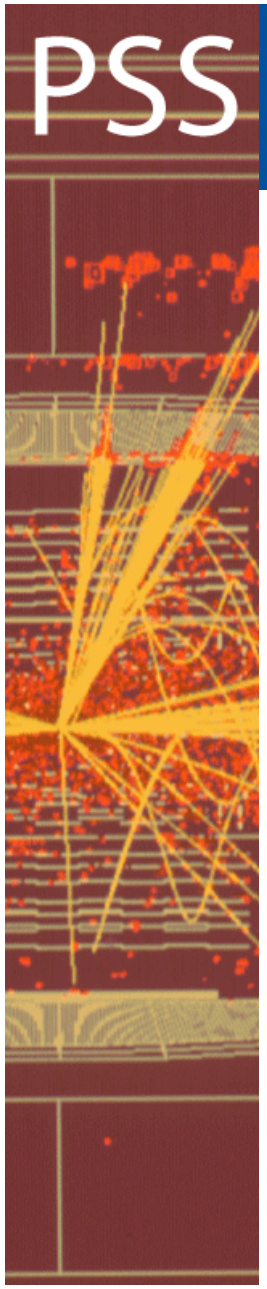
COOL - backend support

CERN IT Department

- **Oracle**
  - From the start (replace old CondDBOracle)
  - *Main focus of development and performance optimization*
- **MySQL**
  - From the start (replace old CondDBMySQL – “Lisbon API”)
- **SQLite**
  - Since COOL 1.2.1 (July 2005)
- **Frontier**
  - Since COOL 1.3.2 (May 2006)
  - Read-only backend - cannot reuse the same test suite
    - A lot of work in the last two months and still in progress
- Same relational schema for all backends
  - A choice (for COOL-independent copies), not a necessity
    - Cross-backend replication (e.g. Octopus) not tested yet

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

CERN



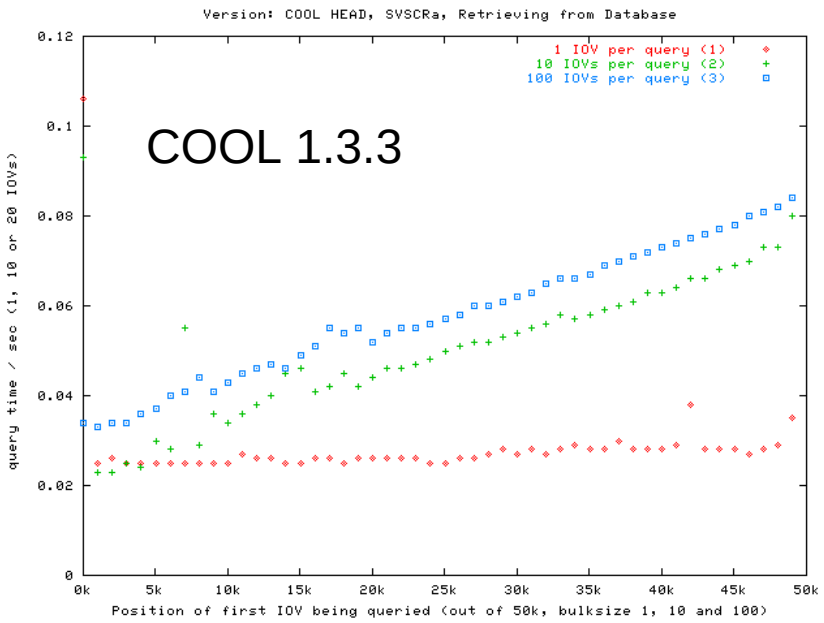
COOL - performance (1)

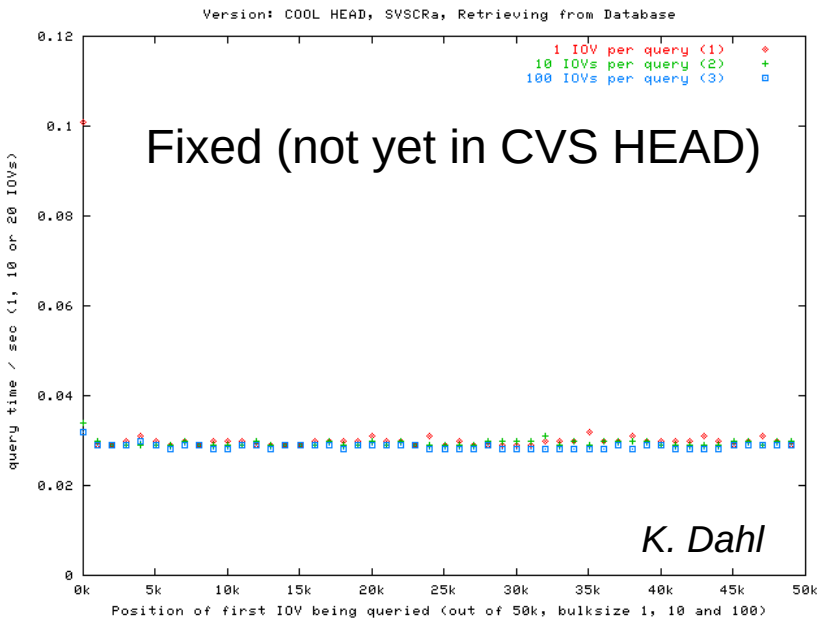
CERN IT Department

- Studied both query optimization and use case validation
  - Focus is Oracle only – not MySQL or SQLite (no manpower…)
- *Scripts to compute table statistics since COOL 1.2.5 (Oct 2005)*
  - Understood tricky features of Oracle execution plans
- Atlas prompt reconstruction validation studies (October 2005)
  - *Achieved 20 MB/s and 20k rows/s sustained read-back rates*
  - See CHEP2006 poster for details
- *Performance report since COOL 1.3.3 (Aug 2006)*
  - Example: report for COOL 1.3.3 linked to COOL web page
  - Includes plots for several pre-defined use cases (e.g. single/multi-version insertion/retrieval from single/multi-channel folders)
    - Most of the issues shown in the COOL 1.3.3 report are pending
    - Some instead are fixed in CVS HEAD or in private test code

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

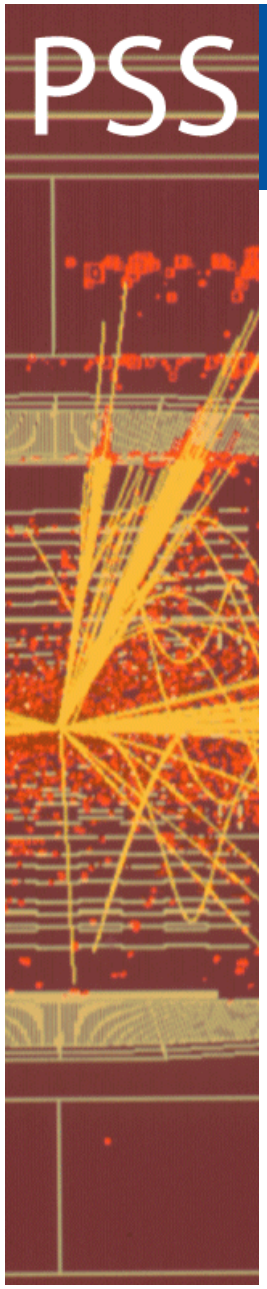
COOL - performance (2)





- Example: query time for single-version single-channel retrieval of all IOVs between two time points t1 and t1+Δt
  - Query time increases with increasing values of t1
  - Fix prototyped in private COOL code, but not yet in CVS
  - Effect previously observed (now fixed) to fetch one IOV at time t1

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it



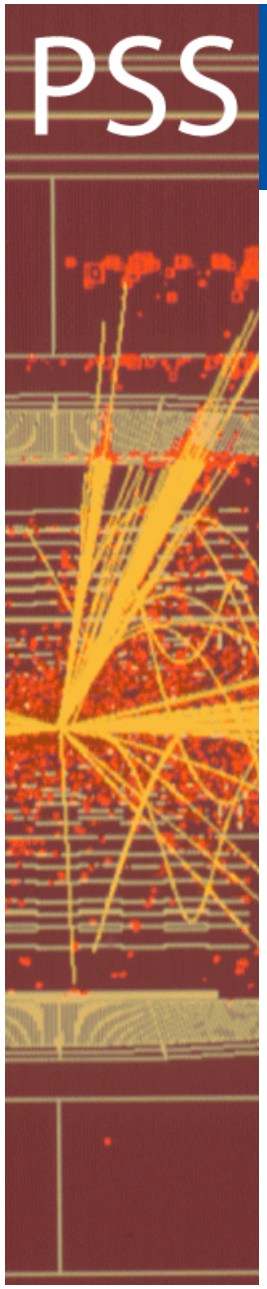
COOL - data distribution

CERN IT Department

- Replication at the database backend level
  - **Oracle Streams** technology (tested and deployed within the 3D project) - see slides about Atlas and LHCb
  - Cross-technology replication is possible in principle (same schema for all backends), not really considered/tested yet
- Oracle remote access via **Frontier**
  - Intermediate Squid web caches
  - *Work in progress (functional/performance tests with Squid, cache optimization for specific use cases…) – Atlas tests*
- Replication using tools based on the COOL API
  - Data slicing/selection is also possible
  - Cross-technology replication (e.g. to **SQLite files**)
  - *Dynamic replication tools as of COOL 2.0.0*

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

CERN



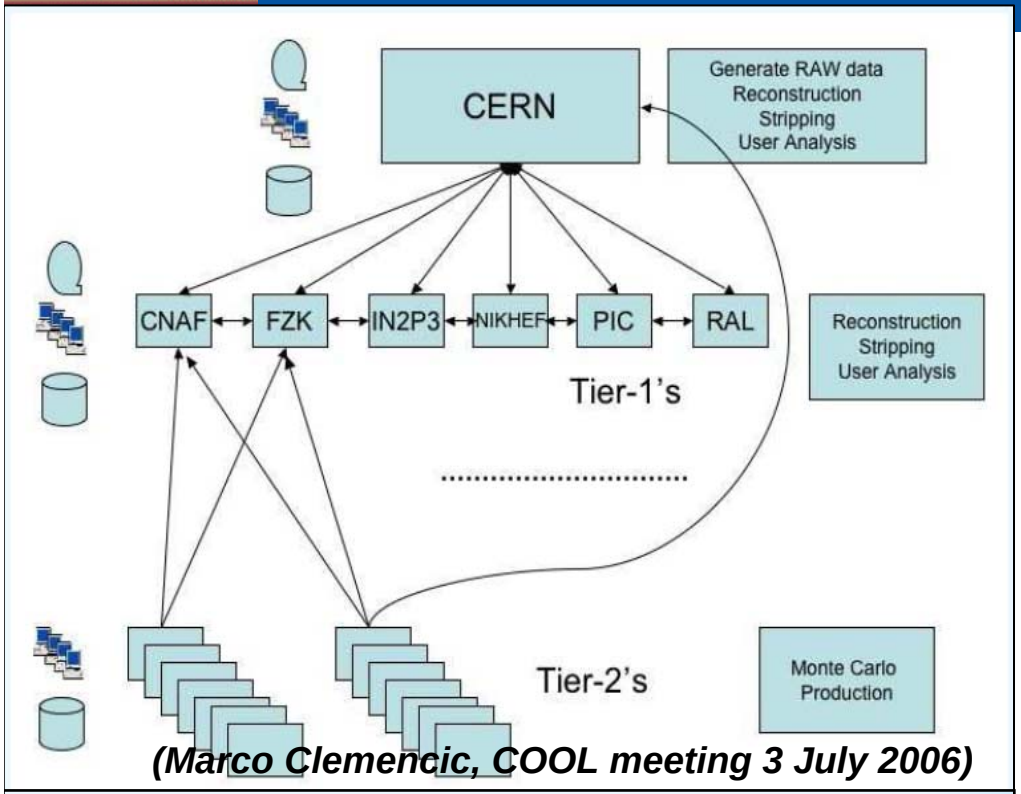
Conditions database choices

- **ALICE**
  - *Alice-specific software* for time/version handling
  - ROOT files (via AliROOT)
- **CMS**
  - *CMS-specific software* for time/version handling
  - Oracle with Frontier cache (via POOL-ORA API)
- ***ATLAS and LHCb***
  - *COOL common software for time/version handling*
  - *Oracle, MySQL, SQLite, Frontier (via COOL API)*

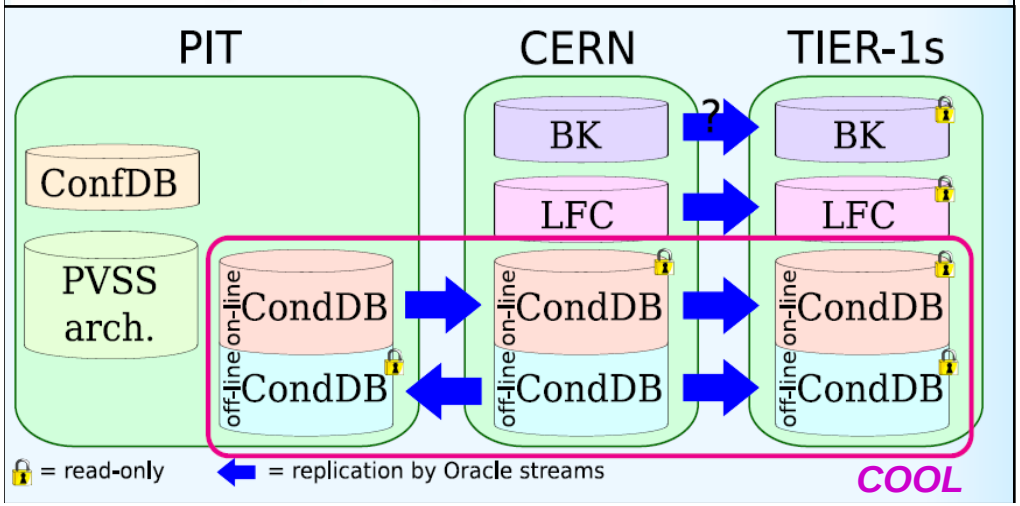
CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

PSS COOL - deployment in LHCb

CERN IT Department

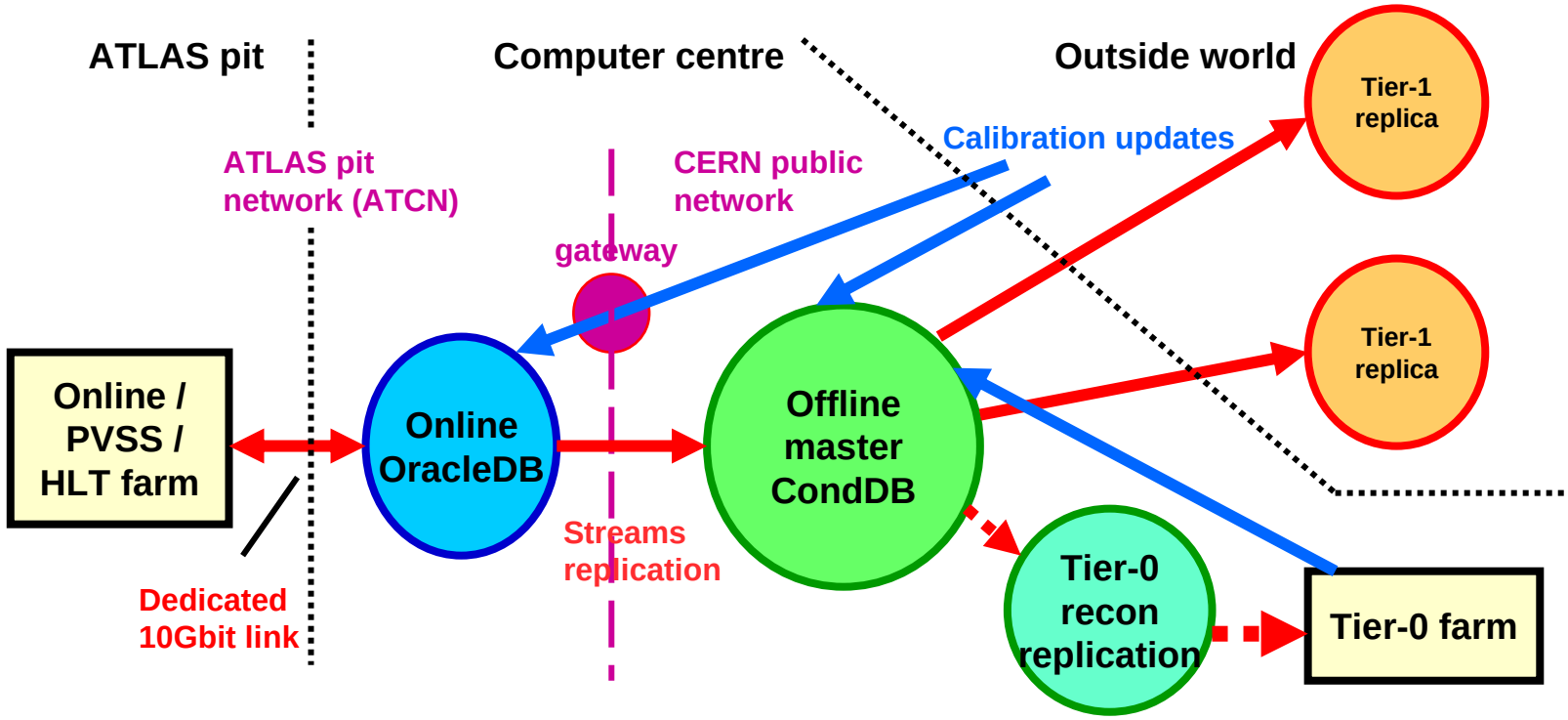


(Marco Clemencic, COOL meeting 3 July 2006)



- Computing model
  - Reconstruction at Tier0/1
  - Only MC prod at Tier2
- **COOL stores only conditions data for event reconstruction**
  - ***Oracle*** *at PIT, Tier0, Tier1 with replication via* ***Streams***
  - Geometry and conditions for MC sent to Tier2 as SQLite file
- **Online db master at PIT**
  - Replicated forward to Tier0 and Tier1 via Streams
  - Data from PVSS processes
- **Offline db master at Tier0**
  - Replicated back to PIT and forward to Tier1 via Streams
  - Data computed in offline calibration/alignment jobs

COOL - deployment in Atlas



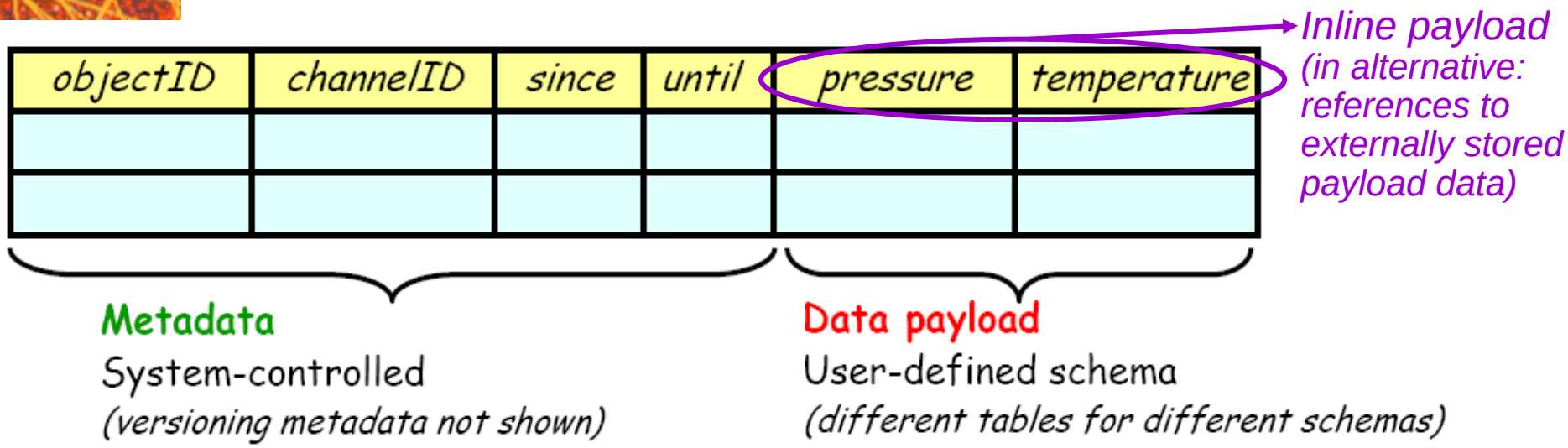
(Sasha Vaniachine and Richard Hawkings, 3D workshop 14 Sep 2006)

- Replication to Tier2's
  - COOL 'dynamic replication', e.g. to MySQL – as of COOL 2.0.0
  - Evaluating COOL Frontier backend (performance, cache consistency…)

CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

COOL - relational implementation

- Modeling of condition data “objects”
  - System-managed common **“metadata”**
    - Data items: many tables, each with many “channels”
    - Interval of validity - IOV: since, until
    - Versioning information with handling of interval overlaps
  - User-defined schema for **“data payload”**
    - Support for simple C++ types as CORAL “AttributeList”



CERN - IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it