Evolution of the use of relational and NoSQL databases in the ATLAS experiment

Dario Barberis

(Genoa University/INFN)

On behalf of the ATLAS Collaboration

Topics

- Conditions data: from detector to analysis
- Metadata: data about data
- Database usage by ATLAS in LHC Run2
- Evolution of database technologies, infrastructures and applications

Topics

- Conditions data: from detector to analysis
- Metadata: data about data
- Database usage by ATLAS in LHC Run2
- Evolution of database technologies, infrastructures and applications

Conditions Data

- "Conditions Data" are all non-event data that are useful to reconstruct events:
  - Detector hardware conditions:
    - Temperatures, currents, voltages, gas pressures and mixtures, etc
  - Detector read-out conditions:
    - Trigger and detector read-out configurations
  - Detector calibrations:
    - Energy calibration for calorimeters, time-over-threshold for pixels, R-T relations for drift tubes (TRT and MDT)
  - Detector alignments:
    - Relative and global alignment of sub-detectors
  - Physics calibrations:
    - Jet energy scales and resolutions, jet flavour tagging weights, trigger and reconstruction efficiencies, etc.
- All conditions data have associated intervals of validity and (for derived data) versions
- The traditional way to store and access conditions data is through a relational database
  - The COOL API is used by ATLAS for the conditions database except for the physics calibration data

COOL in 1 slide

DB

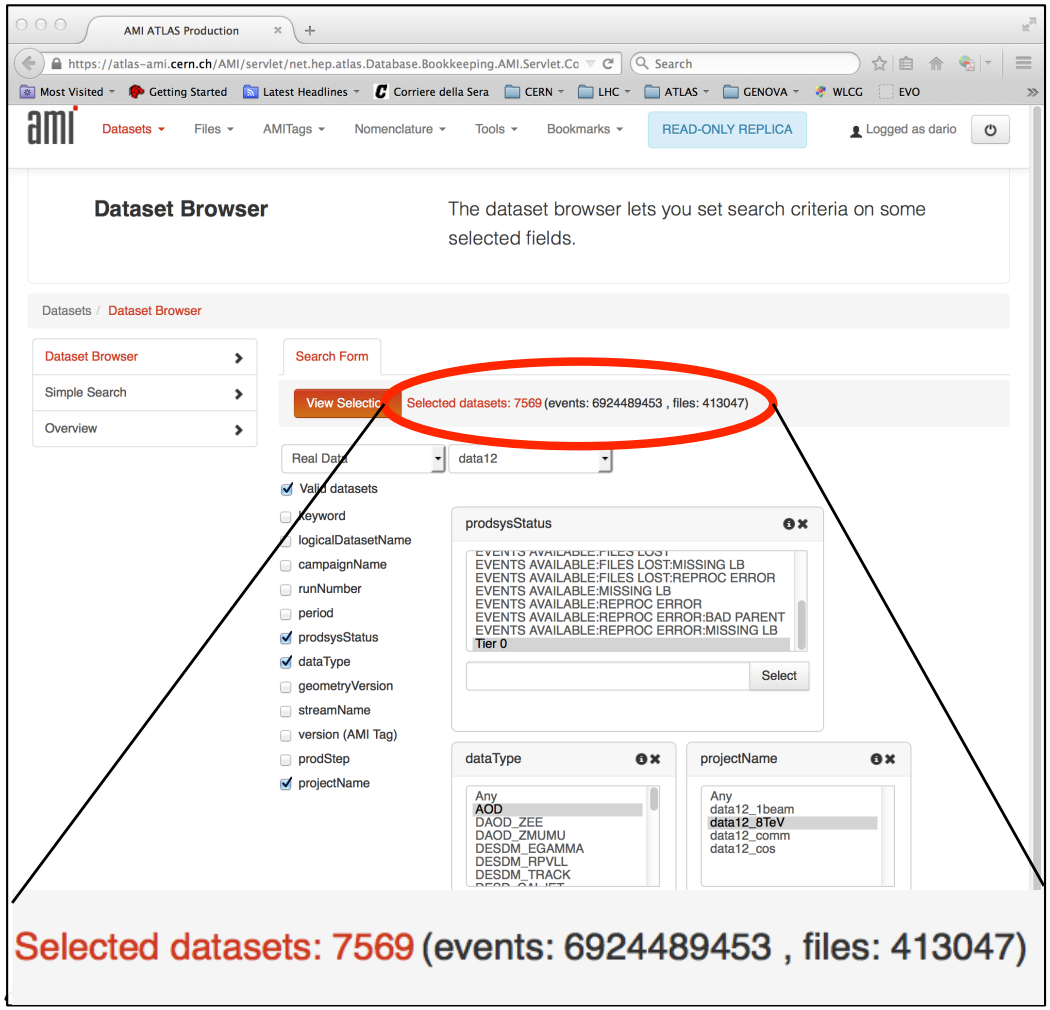
- COOL is the database API used by ATLAS and LHCb to store all time-dependent conditions data
- Data are organised in
  - "Folders" organised in Unix-like directory paths
    - Folders contain one or more conditions which are logically related and are expected to have similar intervals of validity
  - Intervals of Validity (IoV)
    - Each IoV refers to a time range during which the associated data are valid
    - Start and end times can be expressed as timestamps or run-lumiblock
  - Versions
- Versions can be associated an alphanumeric "tag"
- "Global tags" can be associated to groups of folder versions that have to be used together
  - Selecting a given global tag in a job guarantees that the retrieved data will be consistent.
- COOL is written in C++ and uses (through the CORAL interface) an SQL database as data store
  - ATLAS uses Oracle as DB technology
    - The master DB is replicated to RAL, Lyon and TRIUMF for ease of access through the Frontier web services
  - Large data structures can be stored in external files referenced by COOL and made available everywhere though CVMFS

Topics

- Conditions data: from detector to analysis
- Metadata: data about data
- Database usage by ATLAS in LHC Run2
- Evolution of database technologies, infrastructures and applications

Physics metadata (1)

- Metadata are "data about data"
- AMI (ATLAS Metadata Interface) provides ATLAS physicists with a friendly interface to find the datasets they need for their analyses, including lots of accessory information
  - Number of files, file sizes, number of events
  - Data-taking periods and links to the conditions (see later)
- Information is imported from the Tier-0 system, the Grid workload management system (ProdSys/PanDA) and the data management system (Rucio) and presented in a coherent way
- Back-end storage in Oracle
  - Master at CC-IN2P3 in Lyon, read-only copy at CERN
- Information retrieval through the web i/f
  https://ami.in2p3.fr
  or using the python client
  https://ami.in2p3.fr/pyAMI



Dario Barberis:

Physics metadata (2)

- The COMA (Conditions/Configuration Metadata in ATLAS) project began based on the following tenet:
  - to collect a subset of Run and Luminosity Block wise metadata
  - to store that data in a relational database to facilitate its use by dynamic web interfaces
- Usage of COMA information has grown into many areas. It is now the master repository of ATLAS Data Period information.
  - The suite of Data Period interfaces (Reporting, Entry and Services) are described in https://twiki.cern.ch/twiki/bin/view/AtlasProtected/DataPeriods
- COMA tables are populated with data from the best available sources
  - Mostly from subsystem specific databases (Conditions, Trigger, Tier-0/SFO databases) or other ATLAS Metadata Catalogs (AMI)
  - Some information in COMA is collected from non-database sources (TWiki, emails, xml files, ...) at fixed points in time.
- COMA tables are in a Relational Database (in Oracle) to which AMI has easy access
- Upload select conditions for runs of "analysis interest" (NOT all runs and not all conditions) and upload conditions related to Runs in COOL tags (with cross-checks), for example:
  - Luminosity in COMA is collected from the the conditions database using the latest/best luminosity tag version recommended by luminosity experts for each project
  - Conditions DB metadata is collected only for instances which are active in LHC Run 1 or in preparation for Run 2
- COMA tables additionally contain a variety of Refined/Corrected/Derived information to form unique and more effective criteria
  - Information not available in other systems

Technical metadata

- The distributed production/analysis and data management systems produce and need to store a wealth of metadata about the data that are processed and stored
  - Rucio (Distributed Data Management)
    - Dataset contents catalogue: list of files, total size, ownership, provenance, lifetime, status etc.
    - File catalogue: size, checksum, number of events
    - Dataset location catalogue: list of replicas for each dataset
    - Data transfer tools: queue of transfering datasets, status etc.
    - Deletion tools: list of datasets (or replicas) to be deleted, status etc.
    - Storage resource lists, status etc.
  - ProdSys/JEDI/PanDA (Distributed Workload Management)
    - Lists of requested tasks and their input and output datasets, software versions etc.
    - Lists of jobs with status, location etc.
    - Processing resource lists, status etc.
  - Both systems use a combination of quasi-static and rapidly changing information
    - ATLAS runs over 1M jobs/day using 200k job slots and moves 600 TB/day around the world
  - Oracle supports very well both systems if the tables don't grow indefinitely
    - "Old" information is copied to an archive Oracle database and removed from the primary one
    - Accounting information is extracted from the back-up Oracle database and stored in Hadoop for further processing
- Metadata pertaining to event data can also be usefully employed if readily available
  - Which file contains this event, in which format, and which is the internal pointer to retrieve it?
  - The EventIndex uses Hadoop to store this information

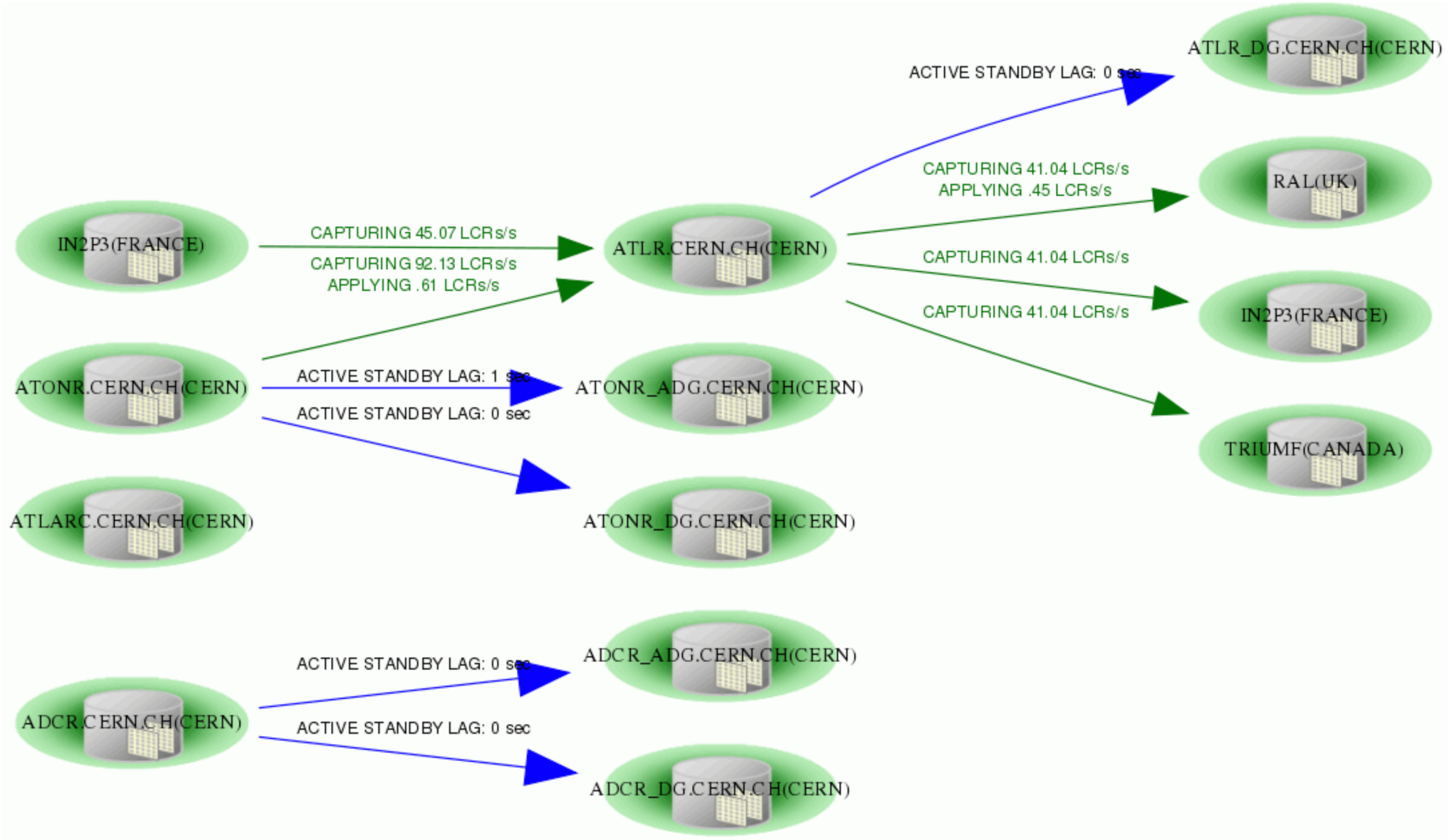
Topics

- Conditions data: from detector to analysis
- Metadata: data about data
- Database usage by ATLAS in LHC Run2
- Evolution of database technologies, infrastructures and applications

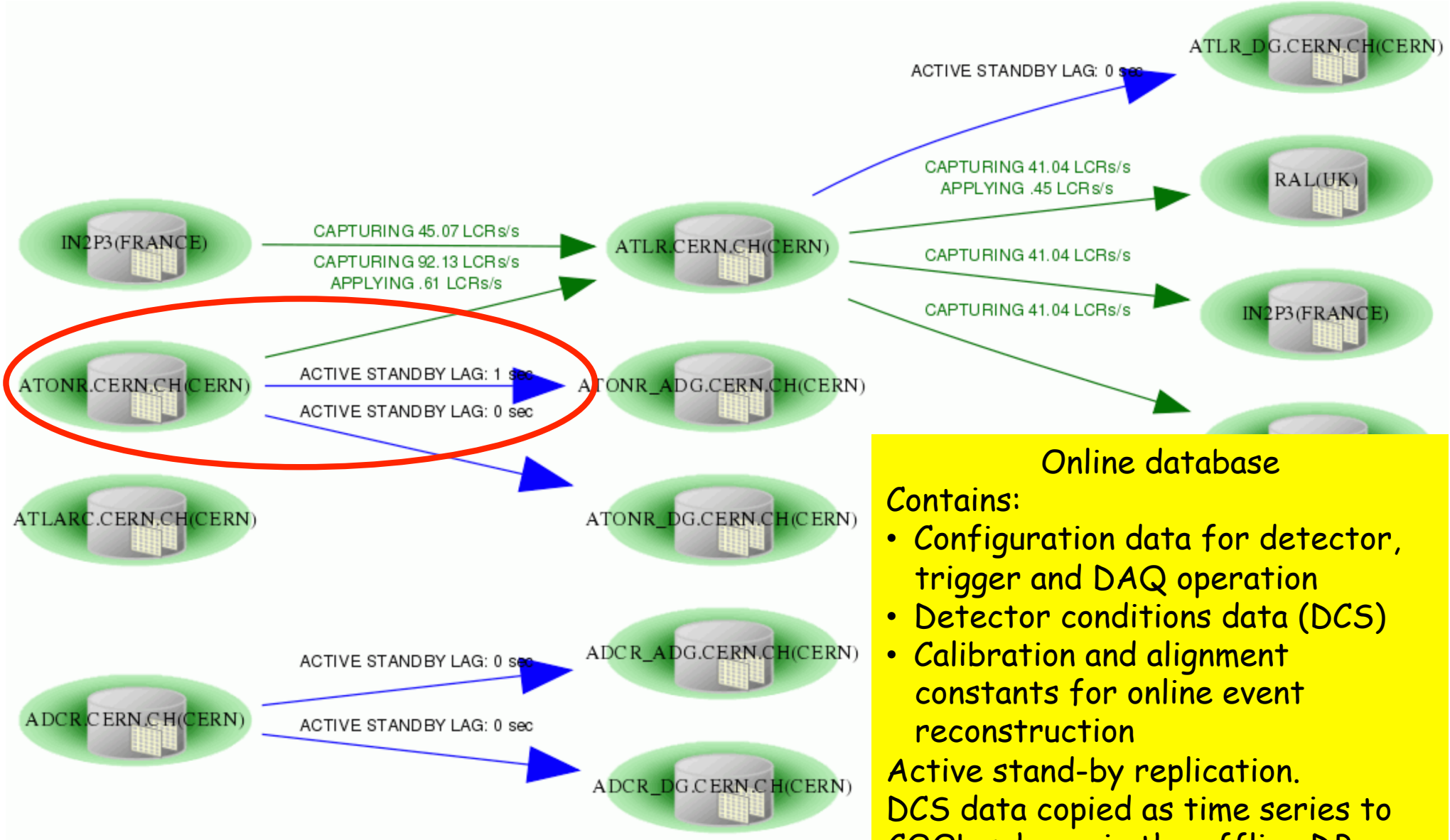
Oracle databases in ATLAS for Run2

- ATLAS relies heavily on Oracle for all major database applications
  - Licence provided by CERN
  - Oracle RACs (Real Application Clusters) provided and supported at the system level by CERN-IT
  - Lots of in-house experience on Oracle application optimisation
    - Two Oracle experts working full time for ATLAS since 2006
- Three main RACs for ATLAS (online, offline, distributed computing) plus an archive DB, all with active stand-by replicas and back-ups
- Selected users and processes have write access. All users have read access.
  - Read access normally through front-end web services (no direct access to Oracle to avoid overloading the servers):
    - Frontier for access from production and analysis jobs
    - DDM and PanDA servers for access to dataset and production/analysis task information
    - The AMI and COMA front-end servers for access to metadata

Oracle databases in ATLAS for Run2



Oracle databases in ATLAS: ATONR



Online database

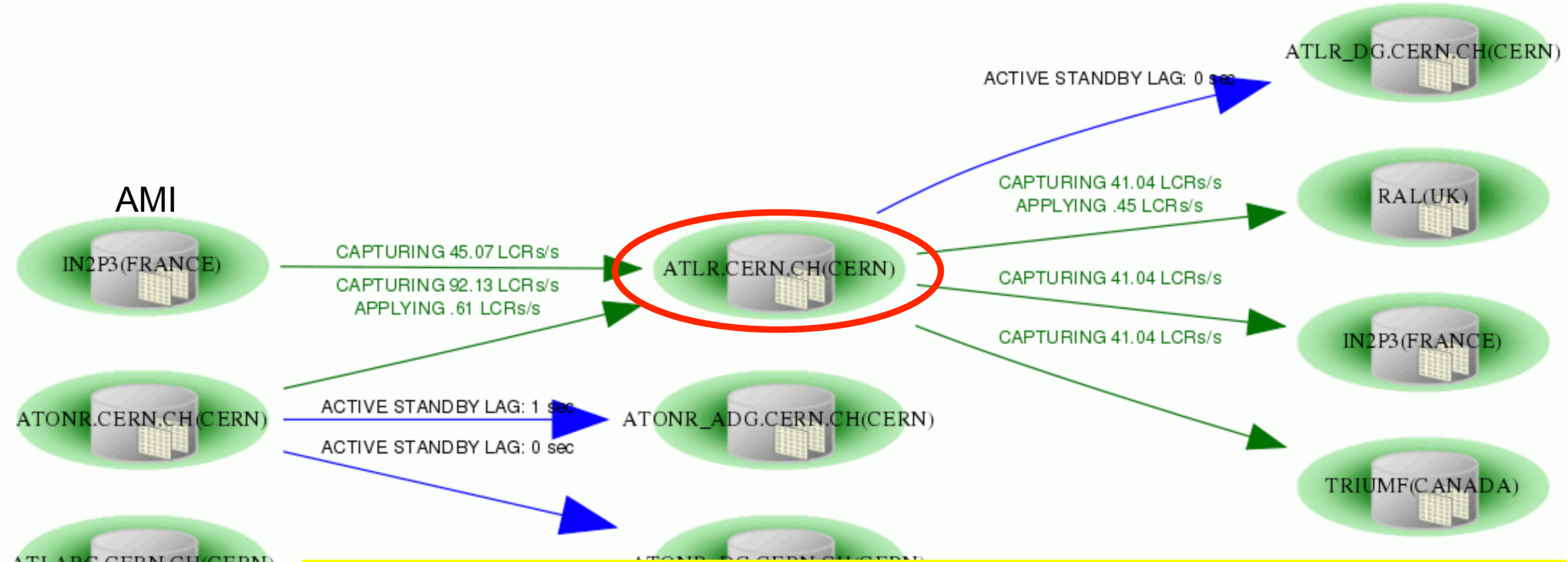
Contains:

- Configuration data for detector, trigger and DAQ operation
- Detector conditions data (DCS)
- Calibration and alignment constants for online event reconstruction

Active stand-by replication.

DCS data copied as time series to COOL schema in the offline DB.

Oracle databases in ATLAS: ATLR



Offline conditions database

Contains:

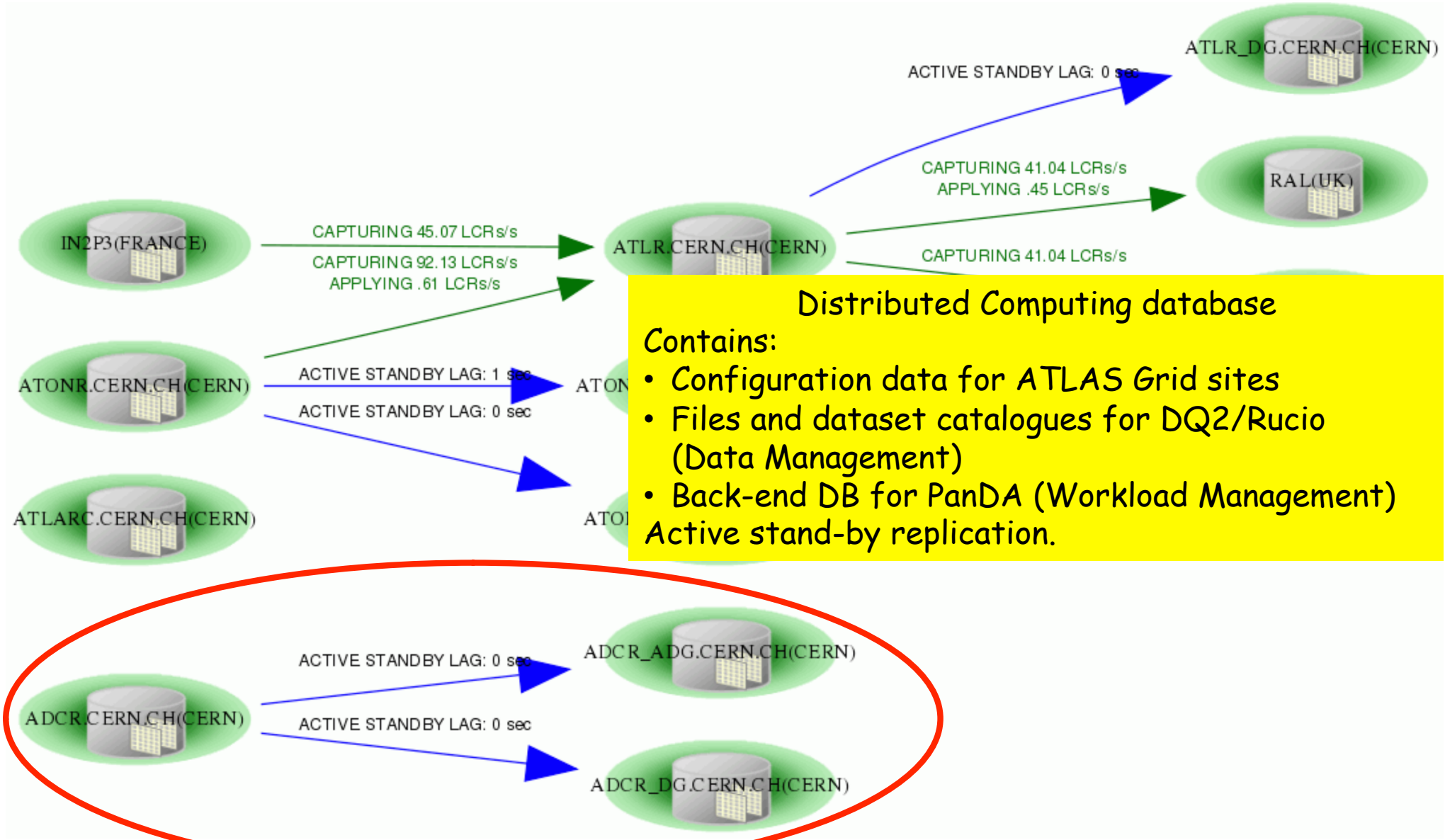
- Detector geometry and trigger DB
- Detector conditions data (DCS)
- Calibration and alignment constants for offline event reconstruction
- COMA metadata database

Active stand-by replication.

COOL data replicated to IN2P3-CC, RAL and TRIUMF for Frontier access.

Gets data from ATONR and AMI master DB in Lyon.

Oracle databases in ATLAS: ADCR



Distributed Computing database

Contains:

- Configuration data for ATLAS Grid sites
- Files and dataset catalogues for DQ2/Rucio (Data Management)
- Back-end DB for PanDA (Workload Management)

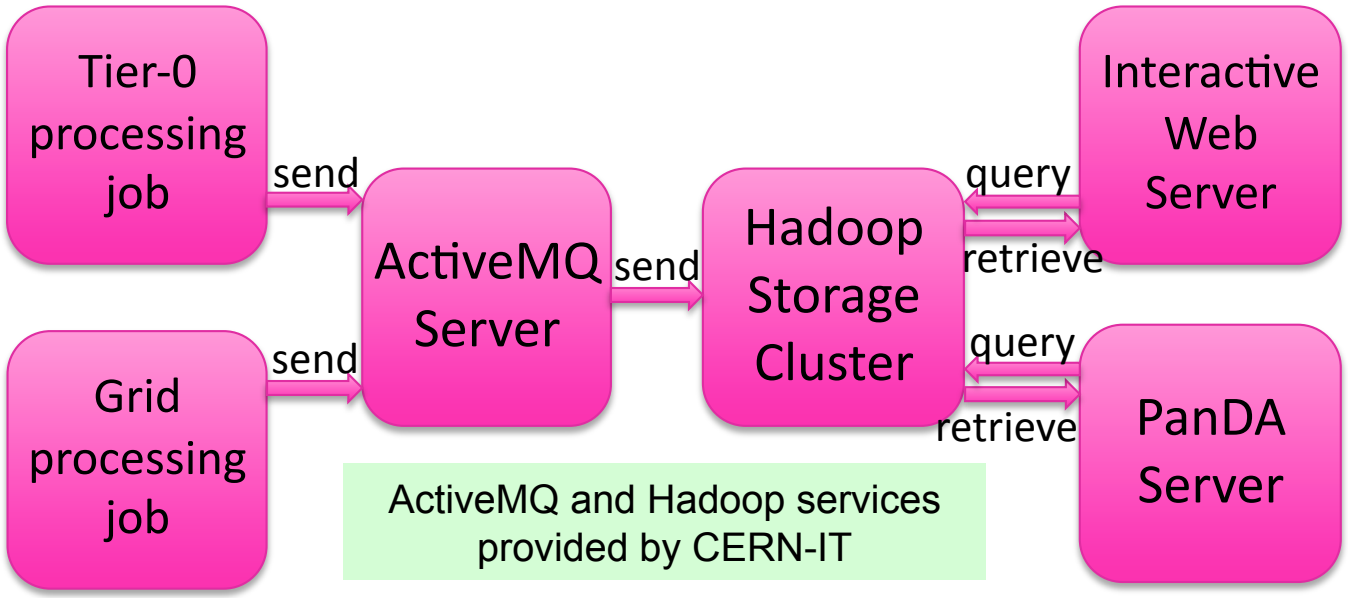
Active stand-by replication.

A NoSQL example: EventIndex

- A complete catalogue of ATLAS events
  - All events, real and simulated data
  - All processing stages
- Contents
  - Event identifiers
  - Online trigger patterns
  - References (pointers) to the events at each processing stage (RAW, ESD, (x)AOD, DAOD) in all permanent files on storage
- Use cases:
  - Event picking
    - Give me the reference (pointer) to "this" event in "that" format for a given processing cycle
  - Production consistency checks
    - Technical checks that processing cycles are complete
  - Event service
    - Give me the references for this list of events (to be distributed to HPC or cloud clusters for processing)
      - Technically the same as event picking
  - Event skimming with trigger selections
    - Give me the list of events passing "this" selection and their references

EventIndex Project Breakdown

- Four major work areas (or tasks):
  - Data collection and storage
  - Core storage
  - Query services
  - Functional testing and operation; system monitoring
- More details in Andrea Favareto's talk on Friday



Topics

- Conditions data: from detector to analysis
- Metadata: data about data
- Database usage by ATLAS in LHC Run2
- Evolution of database technologies, infrastructures and applications

Evolution of Databases in ATLAS

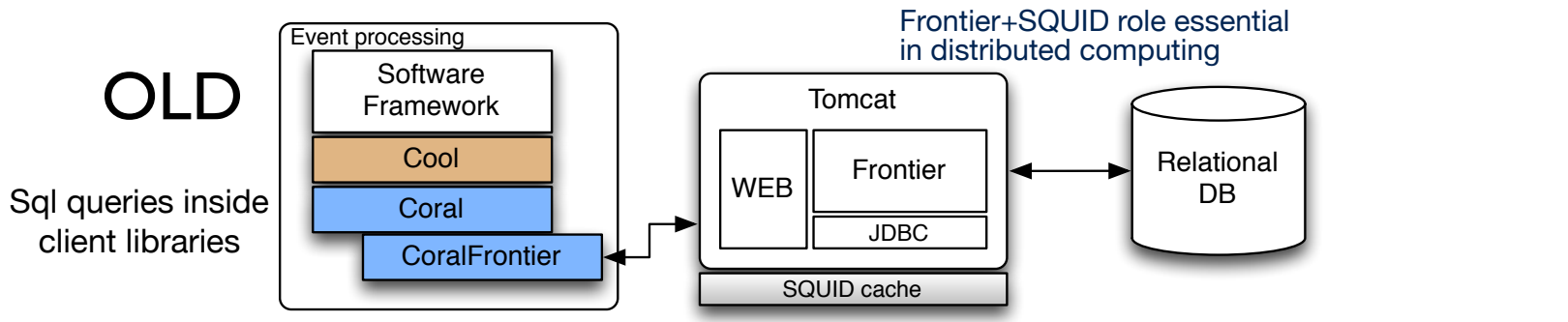
- Oracle is OK for the time being but we are warned by CERN that the licence conditions may change at the end of the current agreement (2018)
  - Diversification may be needed
- Some types of data and metadata fit naturally into the relational DB model, other data much less
  - Large amounts of useful but static data on DDM datasets (accounting), completed PanDA production and analysis tasks, event metadata
- Modern structured storage systems ("NoSQL databases") are now in use in addition to Oracle to store large chunks of read-only data:
  - Hadoop in operation for DDM accounting and EventIndex (talk by A.Favareto)
  - Hadoop under study for Distributed Computing activities monitoring
  - Cassandra under study for PanDA production and analysis tasks archive (talk by M. Grigorieva)
- As long as access to the data is done through an interface server, the user won't actually see the underlying storage technology
- Keeping only the "live" data in Oracle means that at some point in the future we could change technology for the SQL DB without too much trouble (only in case of need of course)
  - See an example in the following slides

A Conditions Data Service for Run3

PCD WS

Proposal for a new architecture

- Review software architecture
- Expand the role of the intermediate server
  - usage of a **multi-tier model architecture** providing all business methods for DB handling at the level of the central server
  - use a light software client (*curl*-like) during event processing : profit more of the REST web access and increase the multi language support
  - use standards from IT industry in order to support multiple DB platforms in a transparent manner

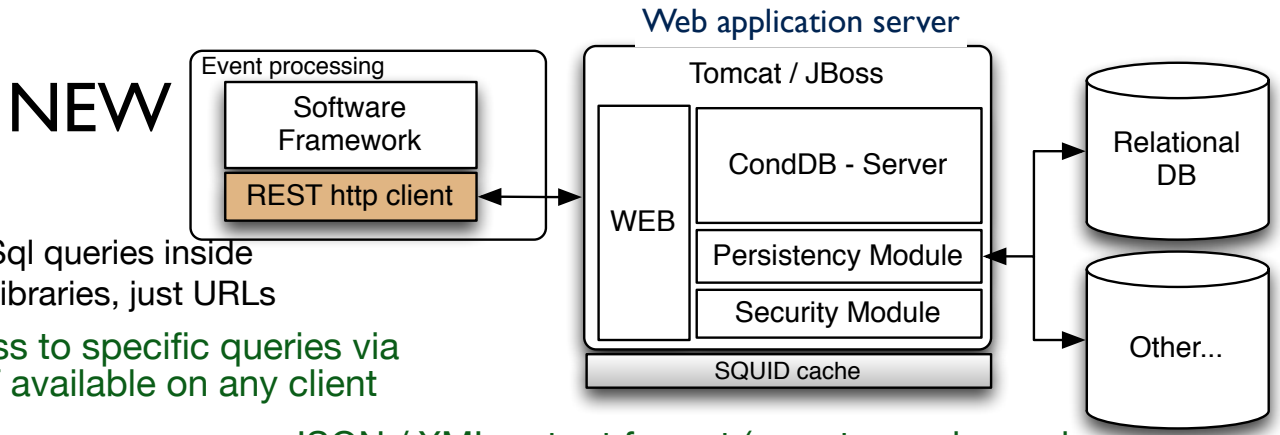


A Conditions Data Service for Run3

PCD WS

Proposal for a new architecture

- Review software architecture
- Expand the role of the intermediate server
  - usage of a **multi-tier model architecture** providing all business methods for DB handling at the level of the central server
  - use a light software client (*curl*-like) during event processing : profit more of the REST web access and increase the multi language support
  - use standards from IT industry in order to support multiple DB platforms in a transparent manner



ex: JSON / XML output format (easy to use in any language on user machine)

Closing Remarks

- The Database group in ATLAS is small but runs a number of operation and development activities
- Diversification is the word of the day: we should use the technologies that best fit the applications we have
  - Some applications are well matched to relational SQL databases
  - Others are more data-intensive and need a flexible data store and search engine rather than fixed schemas and transaction management
- ATLAS supports centrally developers of SQL (Oracle) and NoSQL (Hadoop) applications on clusters provided by CERN-IT
  - Oracle support through 2 DBAs since 2006
  - Hadoop support structure being discussed but necessary for successful operations
- The EventIndex and the new Physics Conditions Data Web Service are examples of "small" projects that can go from ideas to development, deployment and operations in 2-3 years with a reduced number of people
  - Using of course the technologies that match the problem and the knowledge of our developers!