Databases in ALICE

Costin.Grigoras@cern.ch,
Vasco.Chibante.Barroso@cern.ch,
Peter.Chochula@cern.ch

Main DB usecases in Run3

Grid file catalogue

Conditions database

O2 facility tools

DCS

Grid file catalogue

Central catalogue instance

AliEn schema, organized in 3 namespaces

- LFN: 3.3B rows
- GUID: 3.1B rows
- PFN: 3.7B rows

Blob data: federated storage space accessed via Xrootd protocol

Catalogue DB

- One MySQL server
  - 3TB on-disk footprint
  - 2 RAID controllers, 16 disks
  - 1.5TB of RAM
- Slaves for standby / backup
  - Daily full dumps
  - 5h to dump, ~2 days to restore

DB query rates

Averages (1y):

11500 Hz Selects

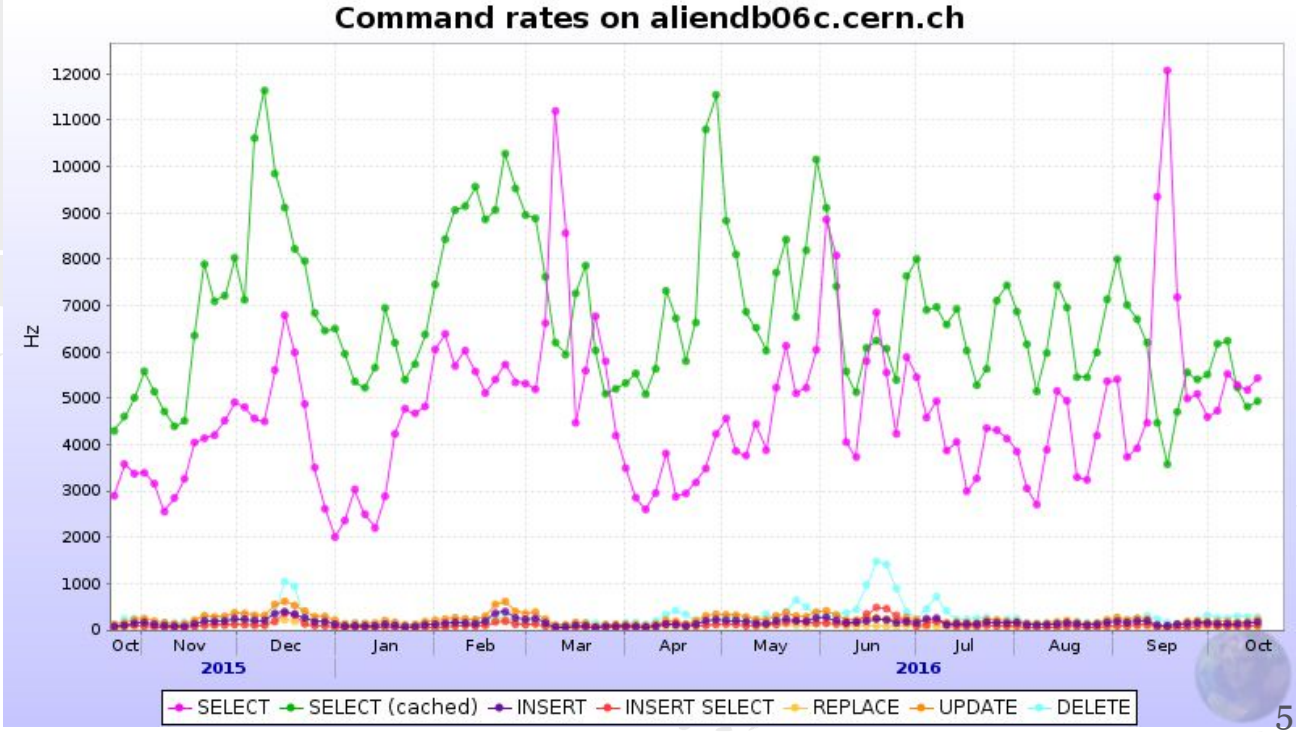
570 Hz Changes

260 Hz Deletes

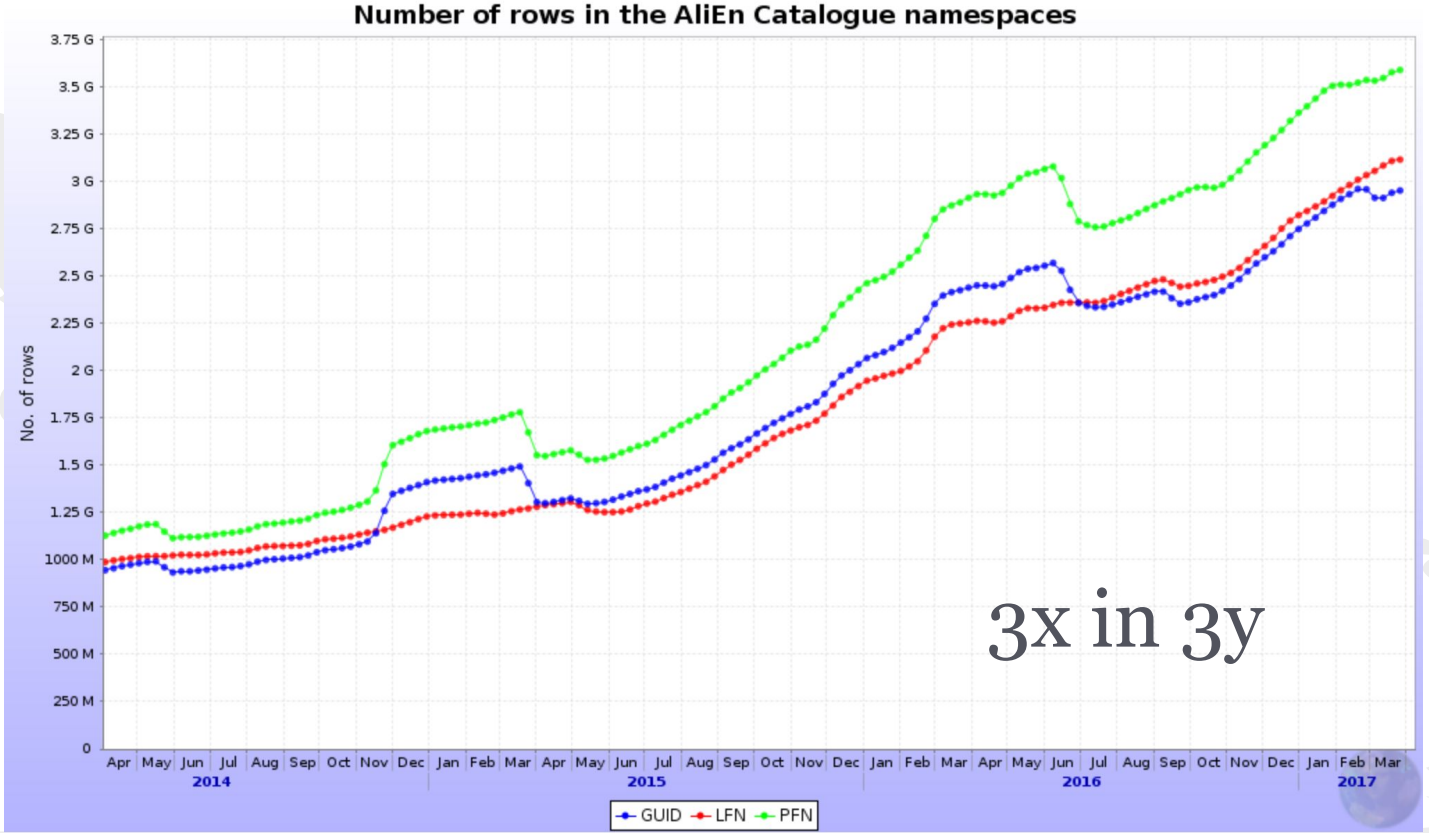
71500 running jobs

20:1 select / change ratio

10:1 read / write data volume



Growth in time



Foreseen growth

In Run3 we will have 5x more computing resources (300K CPUs + 5000GPUs)

- 10x more disk and tape storage
- So … ~10x more files to manage

The goal is to sustain

- ~200kHz queries (stable)
- ~1MHz queries (peaks)

(Some of these will be cached in service memory…)

MySQL experience

Worked well until recently (5.7)

Now random segfaults,table corruptions,…

Doesn’t scale out of the box

- a lot of headaches…

As such we started looking for alternatives

- and found Cassandra

MySQL to Cassandra

Pros

Scalable solution

No SPF

Less pressure for very reliable hardware

Append-only mode

- Trash bin, undo…

Cons

A radically different schema

Have to rewrite the framework

- An opportunity in fact …

MySQL to Cassandra

Pros

Scalable solution

No SPF

Less pressure for very reliable hardware

Append-only mode

Trash bin, undo...

Cons

A radically different schema

Have to rewrite the framework

An opportunity in disguise

...

Many thanks to Apple for sharing their experience with C* !

Cassandra test setup

7 high-end machines

- 2.4TB RAM total
- 256 CPU cores (HT on)

Replication factor 3, Quorum 2

Sustained rates of 100 kHz read and write operations

Other ideas

- Explore Intel's 3D XPoint for storing the catalogue database "in memory"
  - Hopefully transparent via the DB layer
- ScyllaDB as drop-in alternative to Cassandra
  - A bit green at the moment

Conditions database

Currently part of the AliEn/Grid file catalogue

File metadata-based queries:

"Latest version of object X for run range [r1,r2]"

In runs 1 and 2 this was only used in Offline environment

Per-run snapshots prepared in advance to reduce the impact on the central catalogue

Conditions database, Runs 1&2

24000 data runs

- O(hours) for each data run
- 10x more test & calibration runs

2.5M objects in total, 47TB

200 objects / run

- some objects span multiple runs

250MB / run (snapshot size)

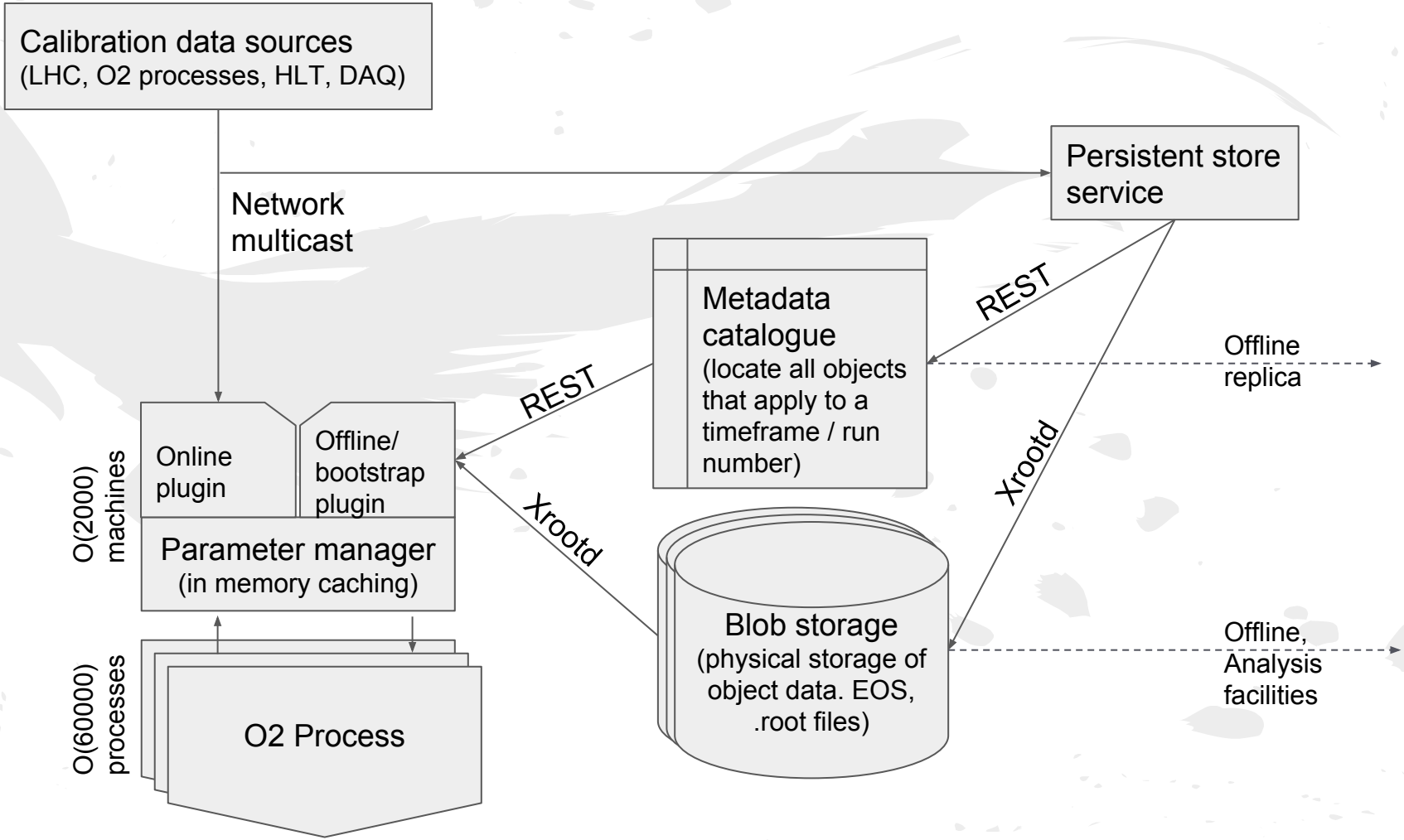
CCDB in Run 3 / 4

Timeframe granularity (O(20ms)) -> 10^5 increase

Used in both Online and Offline (O^2) spaces

- Real-time calibration process
- Applied immediately to the reconstruction
- One shot to get it right (too much data to store for later processing)

Leverage current experience: Cassandra for metadata queries / EOS for blob storage and replication to external facilities


Calibration data sources
(LHC, O2 processes, HLT, DAQ)
Network
multicast
Persistent store
service
Metadata
catalogue
(locate all objects
that apply to a
timeframe / run
number)
REST
REST
Offline
replica
Xrootd
Xrootd
Online
plugin
Offline/
bootstrap
plugin
Parameter manager
(in memory caching)
O(2000)
machines
O(60000)
processes
O2 Process
Blob storage
(physical storage of
object data. EOS,
.root files)
Offline,
Analysis
facilities

O² facility tools

- O² = Big dual personality farm in Run3
  - Online data taking, reconstruction, calibration
  - General processing facility while not taking data
- Investigating best mix of tools for
  - Monitoring
  - Configuration
  - Quality control
  - Logging

O² Monitoring

Database for historical record

Large repository, high writes, low reads

Current top candidates (bundled with monitoring solution)

- MySQL (Zabbix)
- PostgreSQL (MonALISA)
- InfluxDB

O² Configuration

Based on distributed key-value stores

Small repository, low writes, low reads

Current top candidates

- etcd
- Consul

O² Quality control

Need some database for monitoring objects store (medium/large blobs)

Medium sized repository, medium/high writes, low reads

Current prototype based on MySQL

Alternative solution under study

- ElasticSearch

O² Logging

Large repository, high writes, medium reads

Current top candidate solution based on MySQL

The DCS data flow

Two main data storage mechanism are implemented in the DCS:

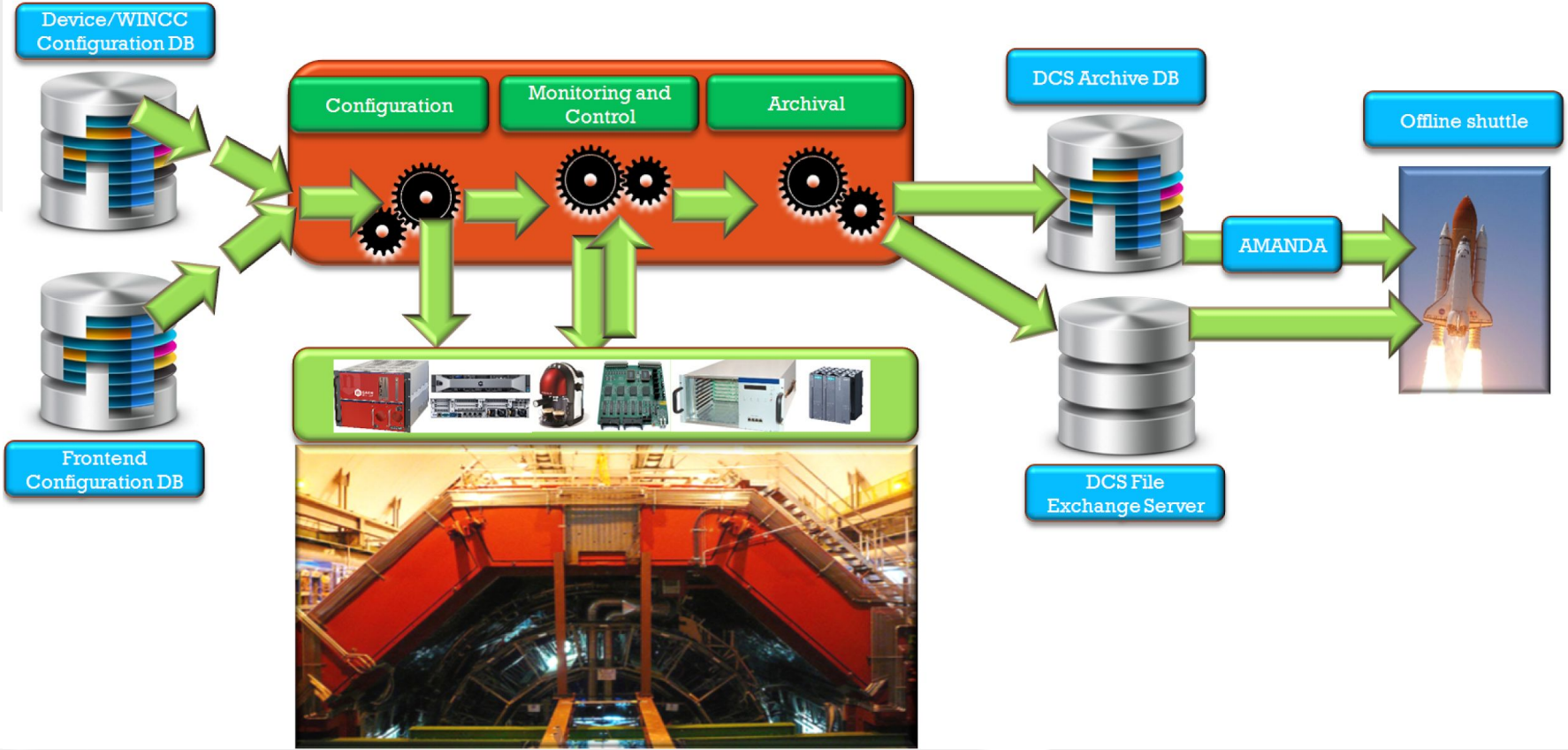
- The DCS ARCHIVE , a relational database based on ORACLE
- The File Exchange Server (FXS or FES)

The DCS provides the infrastructure (hardware, tools, network…)

The DCS stores all the data provided by the clients, but the storage is not manipulating this data, for example:

- If a temperature sensor shows due to a wrong calibration a value of 4000 C (and not 20 C as expected), this will be recorded to Oracle
- If is a role of the individual control system to act (interlock, alert, correction…) as needed, the storage remains passive in this respect

The DCS data flow



The DCS database services

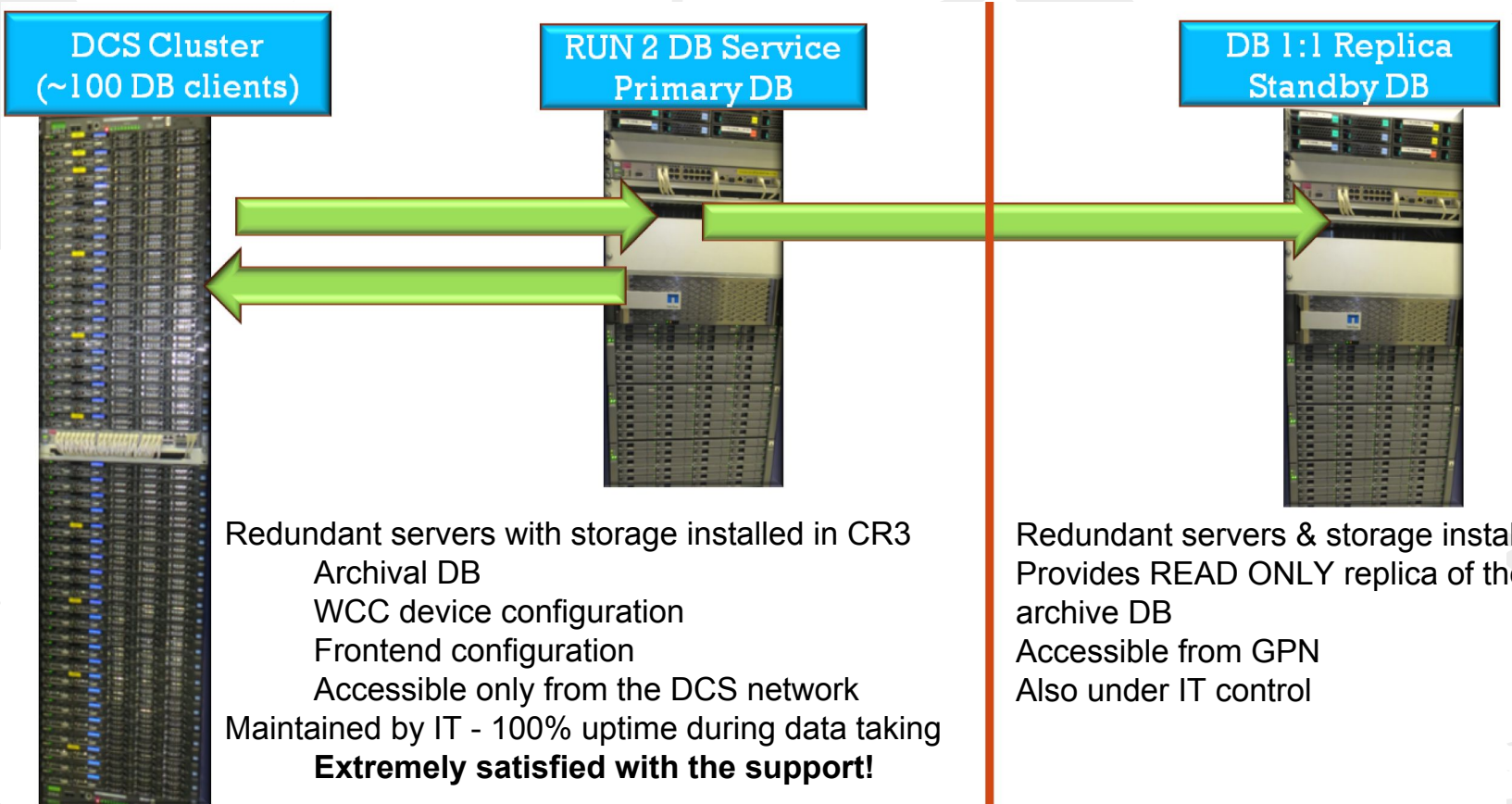
RDB service based on ORACLE

The DCS hosts 3 different databases:

- The WCC device configuration database
  power supply setting, various configuration parameters
  all managed by the JCOP framework tools
- The frontend configuration database
  free format, detector-specific data
  Managed by detector tools
- The DCS ARCHIVE
  The main DCS storage
  Contains all measured data
  Main source of DCS data for OFFLINE analysis

These 3 databases are fully independent and are managed and accessed by completely different tools and methods

DCS database service



Redundant servers with storage installed in CR3
- Archival DB
- WCC device configuration
- Frontend configuration
- Accessible only from the DCS network

Maintained by IT - 100% uptime during data taking
- **Extremely satisfied with the support!**

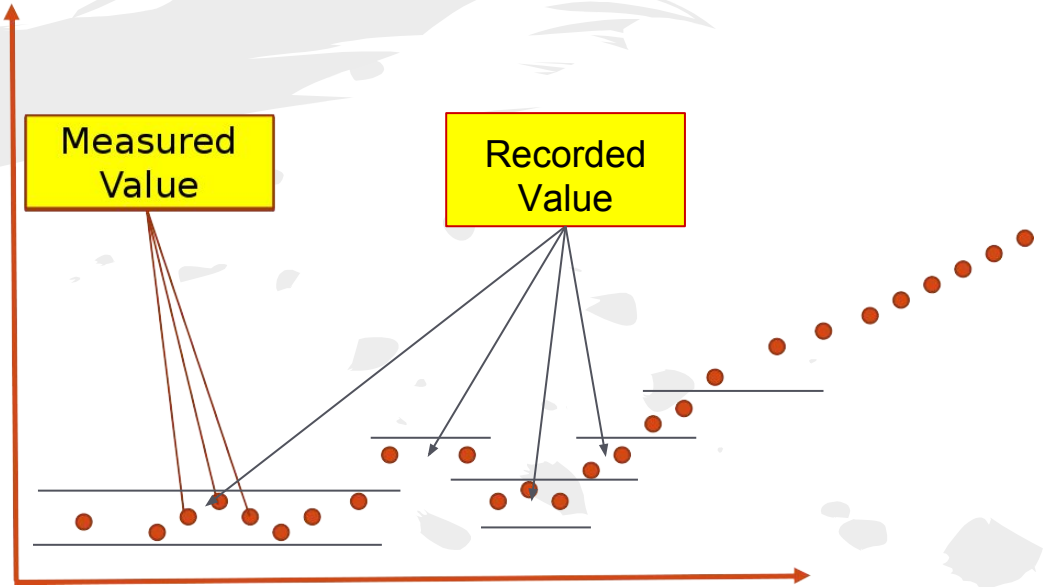
Redundant servers & storage installed in CC
Provides READ ONLY replica of the DCS archive DB
Accessible from GPN
Also under IT control

Smoothing principle

The amount of data written the DB varies from detector to detector, for example:

- TPC is configured to archive 39000 parameters
  5,914,472,823 values are written to Archive per year
  ~200 values/s
- EMC is configured to archive 3098 parameters
  1,291,732,693 values are written to Archive per year (at present a bit more ~4 billion)

Without smoothing it would be 3 orders of magnitude more data



The scale of the DCS db service

The service installed in ALICE P2 consists of 4 servers with redundant storage

100 (mostly WINCC) clients archive:

- 145 000 parameters (out of 1 million parameters defined in the DCS)
- 14 billion values/year
- Steady data insertion rate ~400 values/s

The archival technology is defined by the use of the SIEMENS SCADA system WINCC OA.

- ALICE will follow the evolution of the Siemens software
- the scale of the service will grow by ~20-30%

Summary

Our DB zoo keeps growing :)

In many cases the DB is simply a dependency of the deployed application

We will keep using most of the current solutions

Promising results from NoSQL databases

- Not everything will be ported to them…
- DCS in particular will continue using RDBs