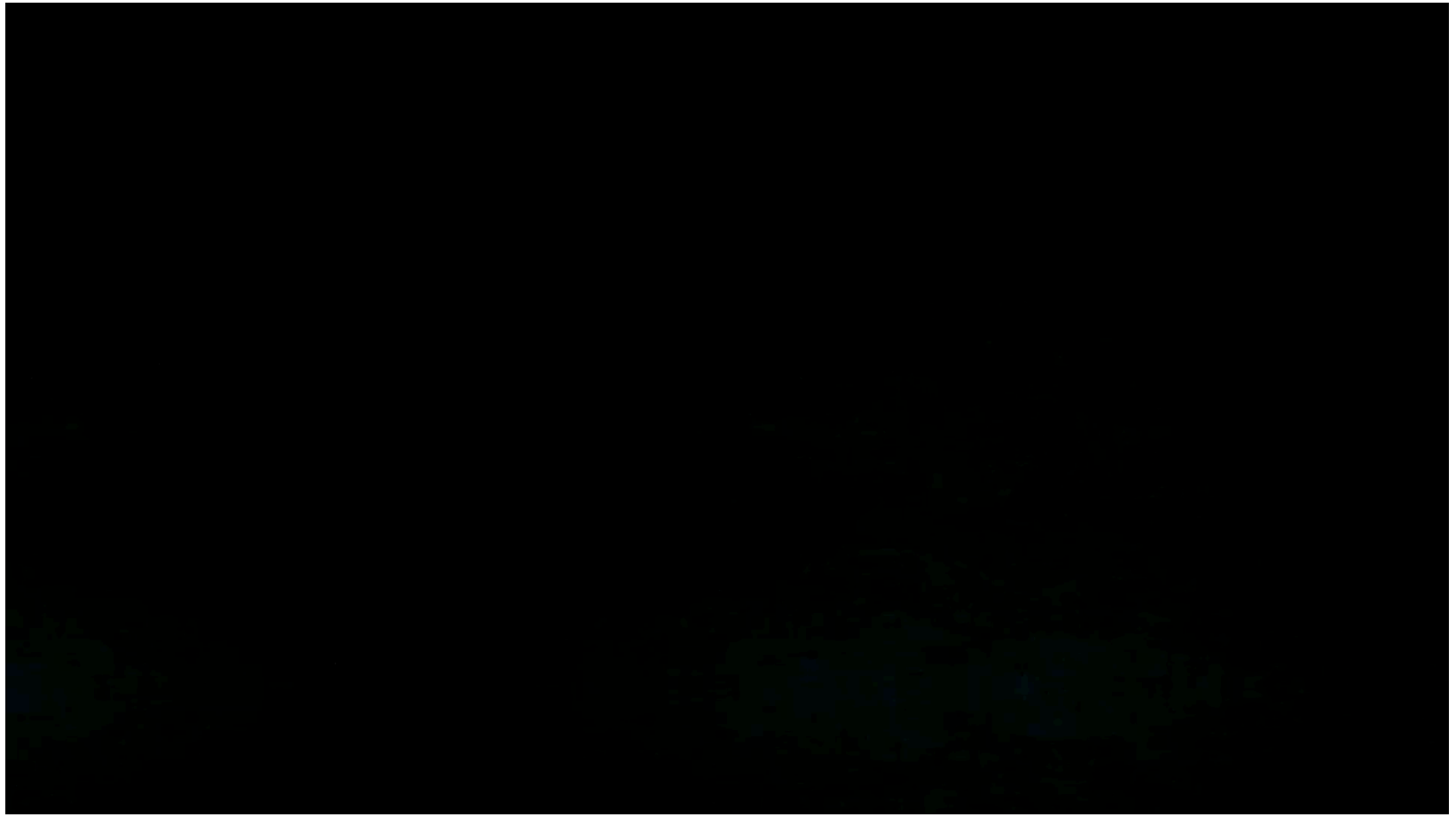
Fault Detection using Advanced Analytics at CERN's Large Hadron Collider

Antonio Romero Marín

Manuel Martin Marquez

What's CERN



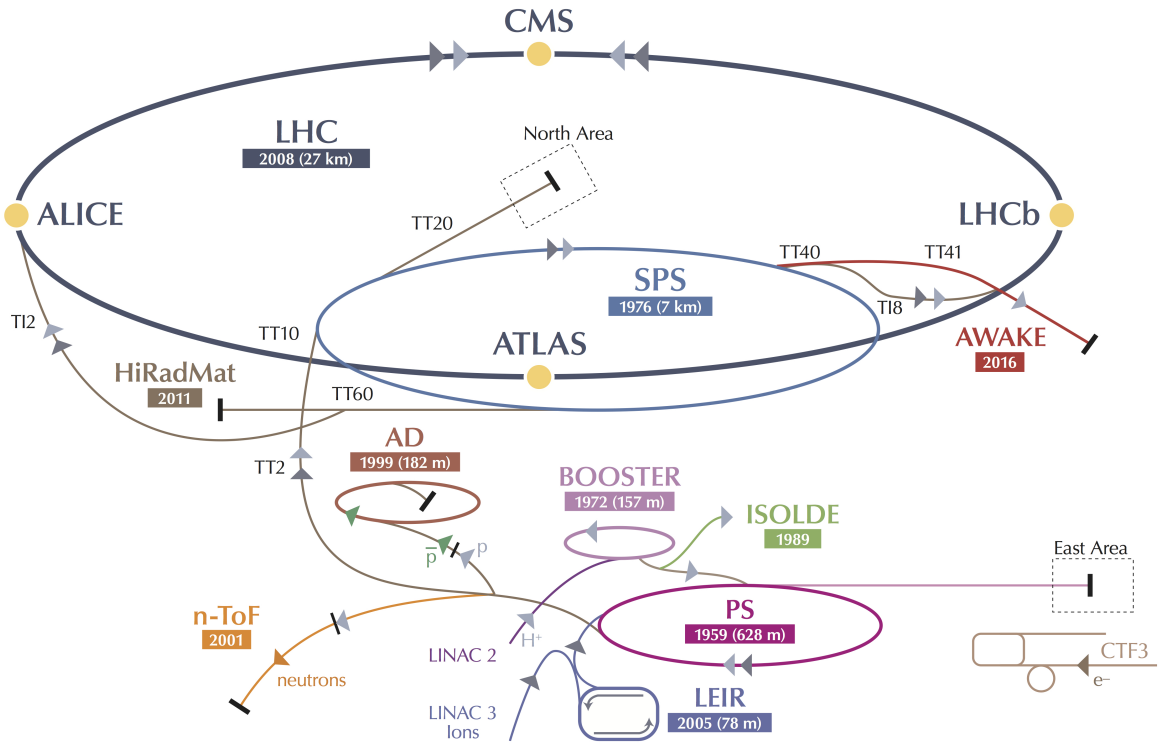
What's CERN

- European Laboratory for Particle Physics
  - Fundamental Research
- Worldwide International Collaboration
  - US is an important contributor to CERN and the experiments
- Education & Training
- Push Frontiers of Technology

CERN openlab

- Public-private partnership between CERN and leading ICT companies and research institutes
- Accelerate cutting-edge solutions for the worldwide LHC community and wider scientific research.
- Designed to create and disseminate knowledge
  - Publication of reports and articles
  - Workshops or seminars
  - CERN openlab Student Programme

CERN Accelerator Complex



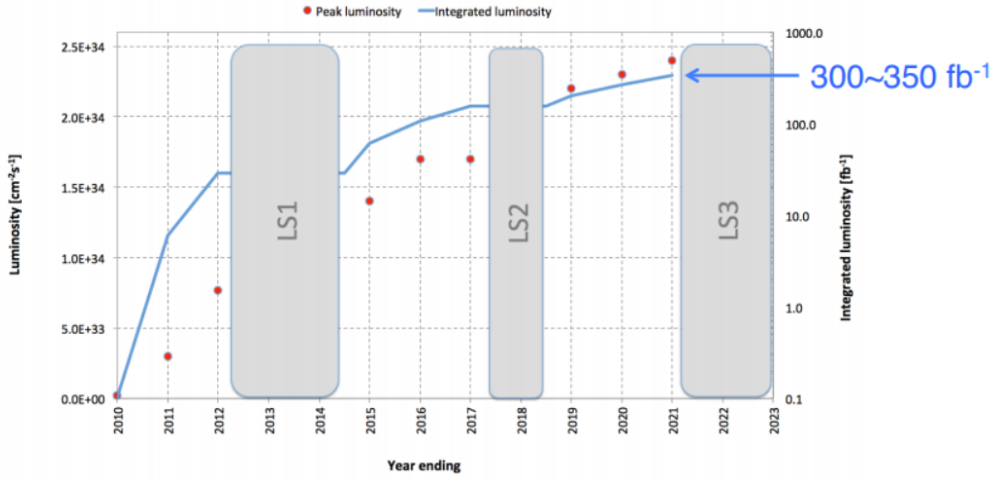
LHC Large Hadron Collider SPS Super Proton Synchrotron PS Proton Synchrotron

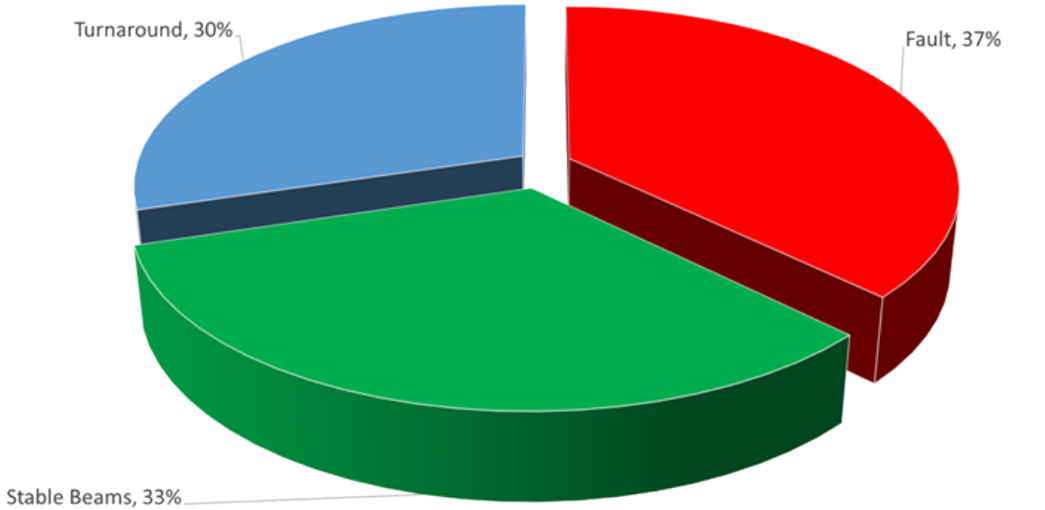
AD Antiproton Decelerator CTF3 Clic Test Facility AWAKE Advanced WAKefield Experiment ISOLDE Isotope Separator OnLine DEvice

LEIR Low Energy Ion Ring LINAC LINear ACcelerator n-ToF Neutrons Time Of Flight HiRadMat High-Radiation to Materials

Data Analytics Challenges

- Some faults cannot be avoided
  - Decrease the availability for running physics
- Preventive maintenance is not enough
  - Does not take into account the condition of the equipment
- LHC upgrades will further increase luminosity
  - Computing resources needs will be higher
  - Data generated will increase drastically

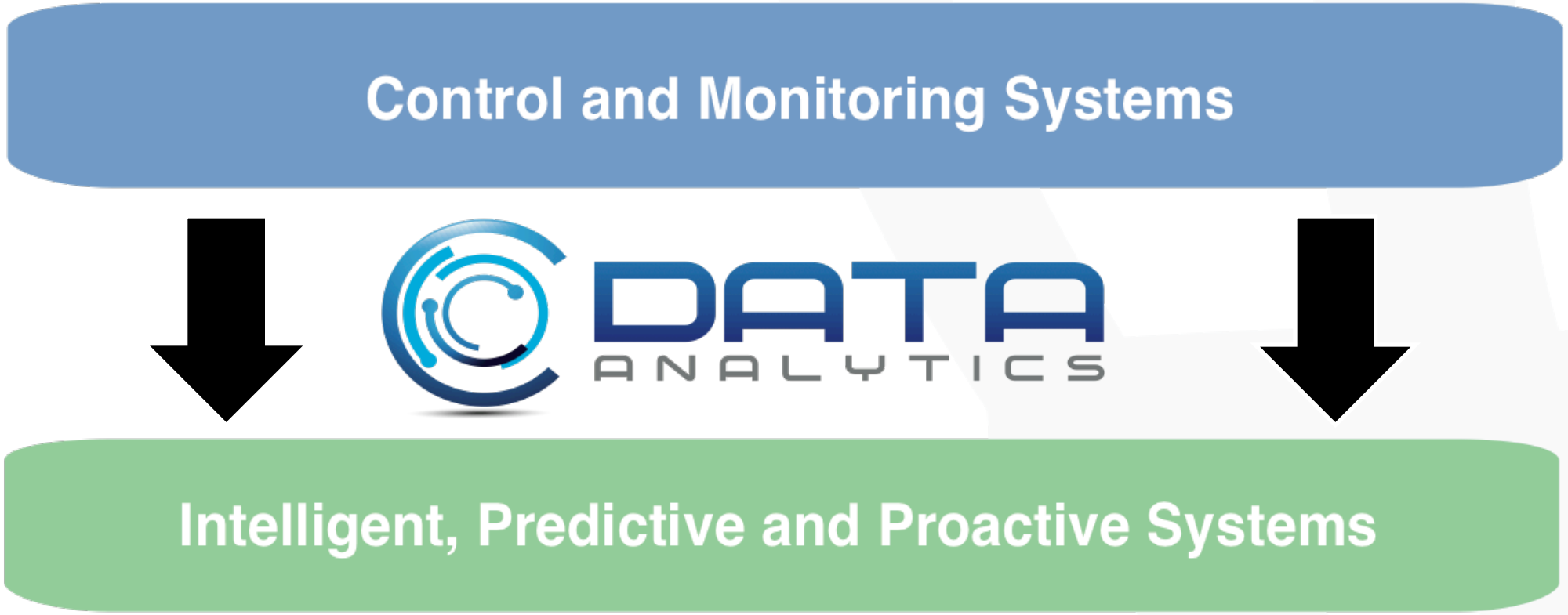




LHC Availability: 63 %

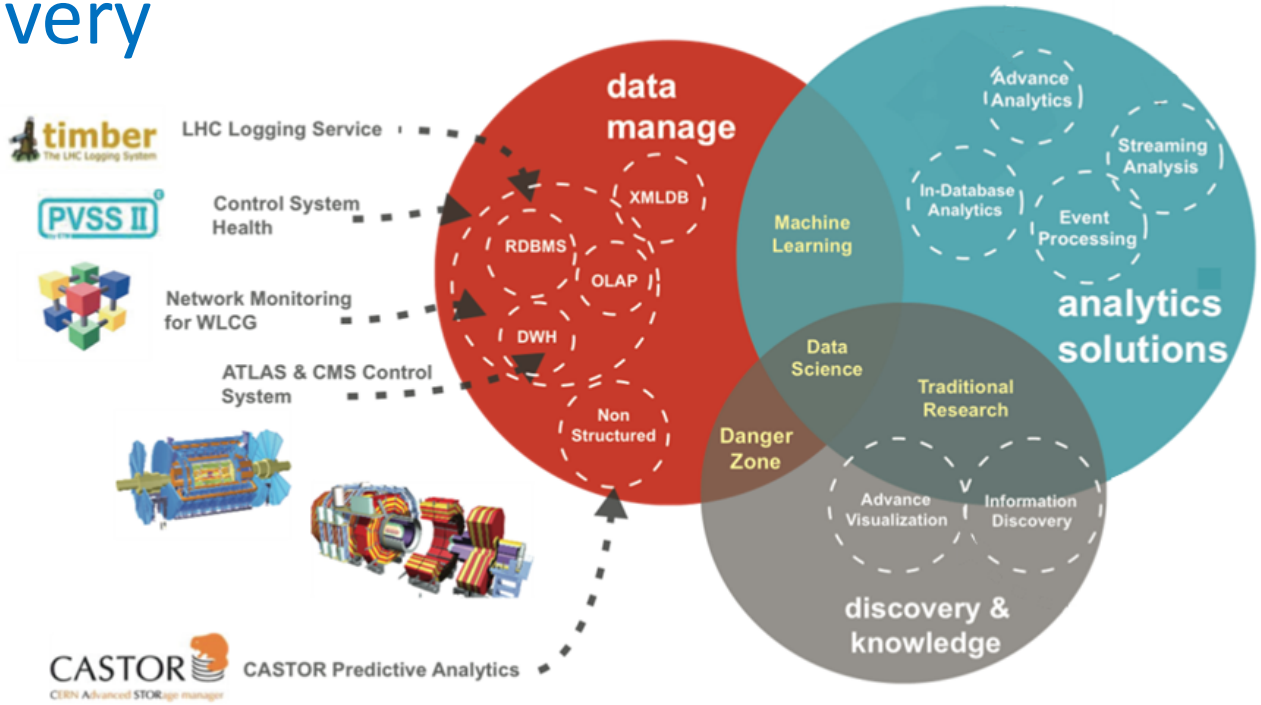
A. Apollonio

Data Analytics Objective



Areas of investigation

- Predictive maintenance and system optimization
- Data extraction, transformation and loading (ETL)
- Data Visualization and Discovery



R for data analytics

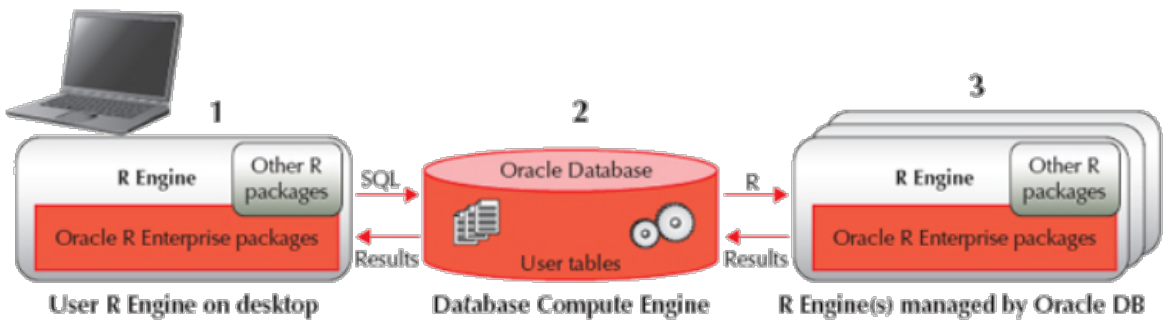
- Language for statistical computing and graphics
- Powerful integrated suite of software facilities
  - Data manipulation
  - Calculation
  - Graphical display, plotting
- Free and Open Source
- Extensible
  - Over 7800 CRAN packages extending base functionality
- Interactive and programmatic environment
- Great user community
  - Active development, frequently updated

Parallel Processing Approaches in R

- Parallel processing with R
  - Foreach
  - Snow
  - Rmpi
  - BatchExperiments package (BatchJobs)
- SparkR
  - Packaged in Spark from 1.4.0
- Oracle R Enterprise (ORE)

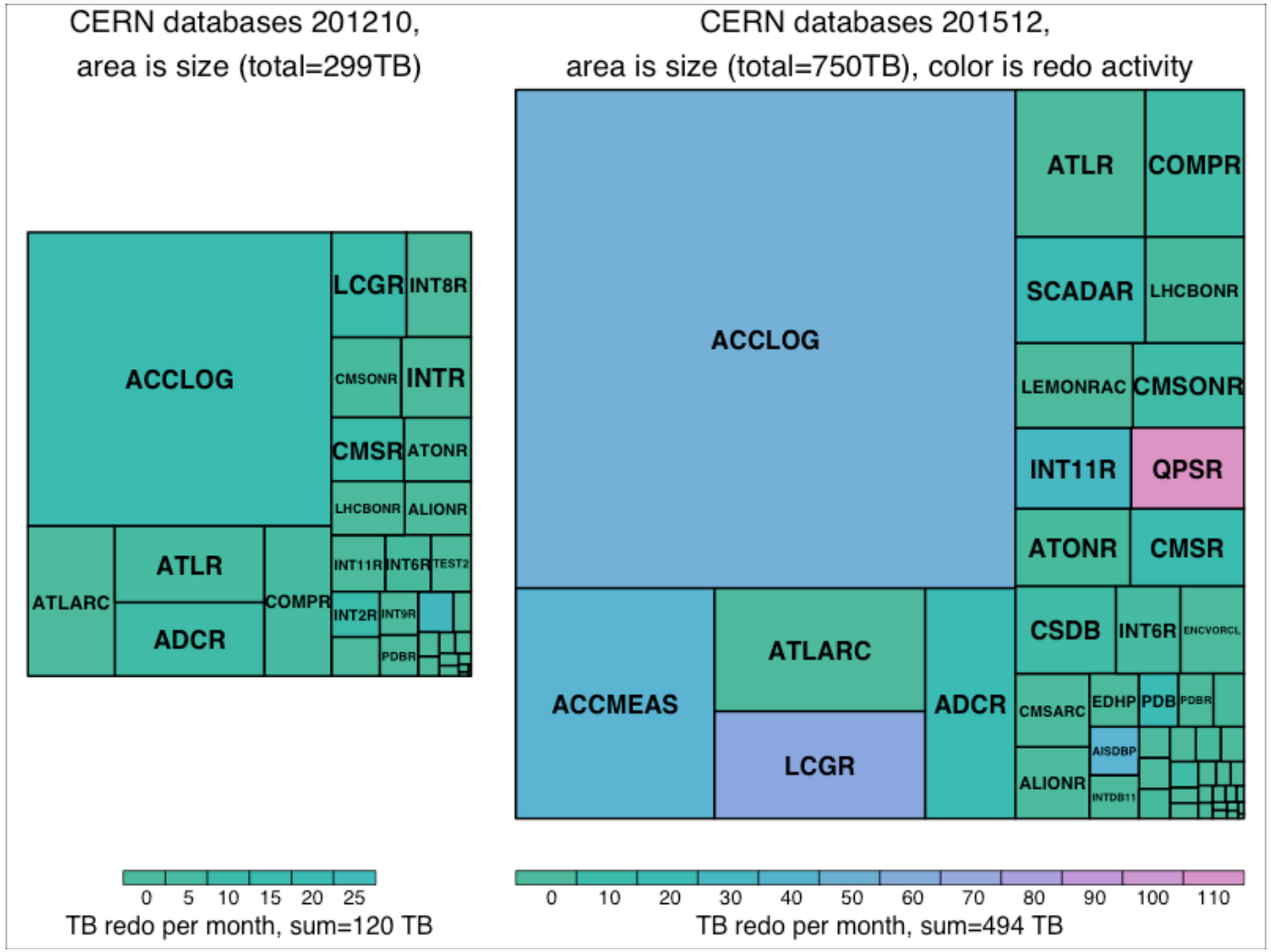
Why ORE? - ORE benefits

- A database-centric environment for analytical processes in R
  - Allows to use the database server to run R scripts (scalability & performance)
  - Eliminate memory constraint of client R engine
  - R working on data directly in the database, no data transfer
  - Integration with the SQL language
  - Allows in-database data analysis
  - Provide data parallelism



- Transparency Layer
  - Transparently analyze and use data in Oracle Database through R
  - No need to know SQL

CERN Database activity

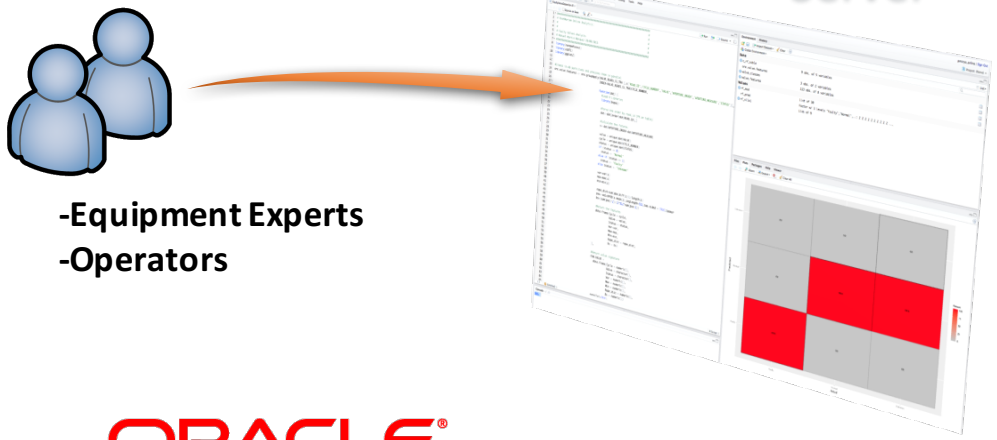


	October 2012	December 2015
Max size	ACCLOG 136TB	ACCLOG 352TB
Max redo	ACCMEAS 27TB / month	QPSR 115TB / month

ORE deployment

ORACLE 12c DATABASE + ORACLE R ENTERPRISE

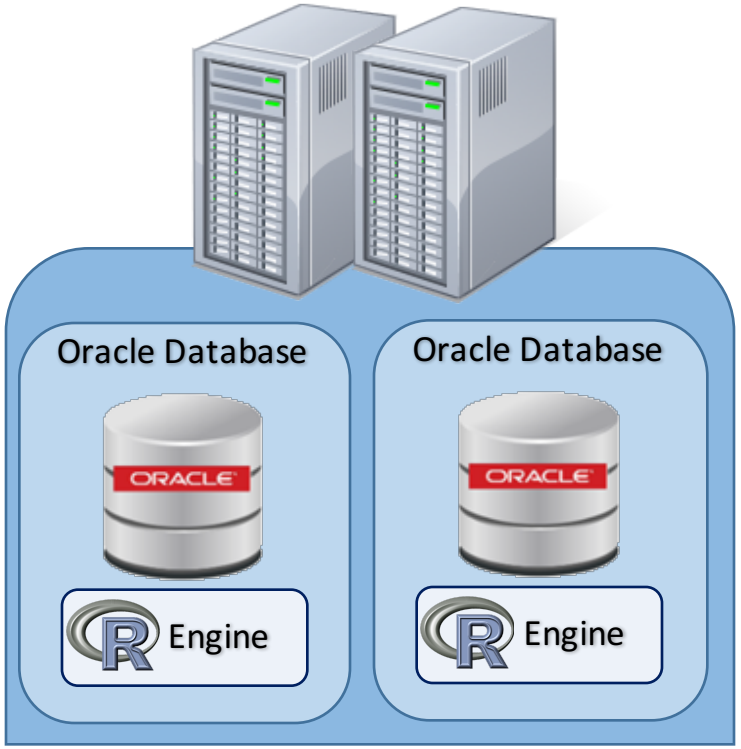
RStudio Server



-Equipment Experts
-Operators

ORACLE EXALYTICS

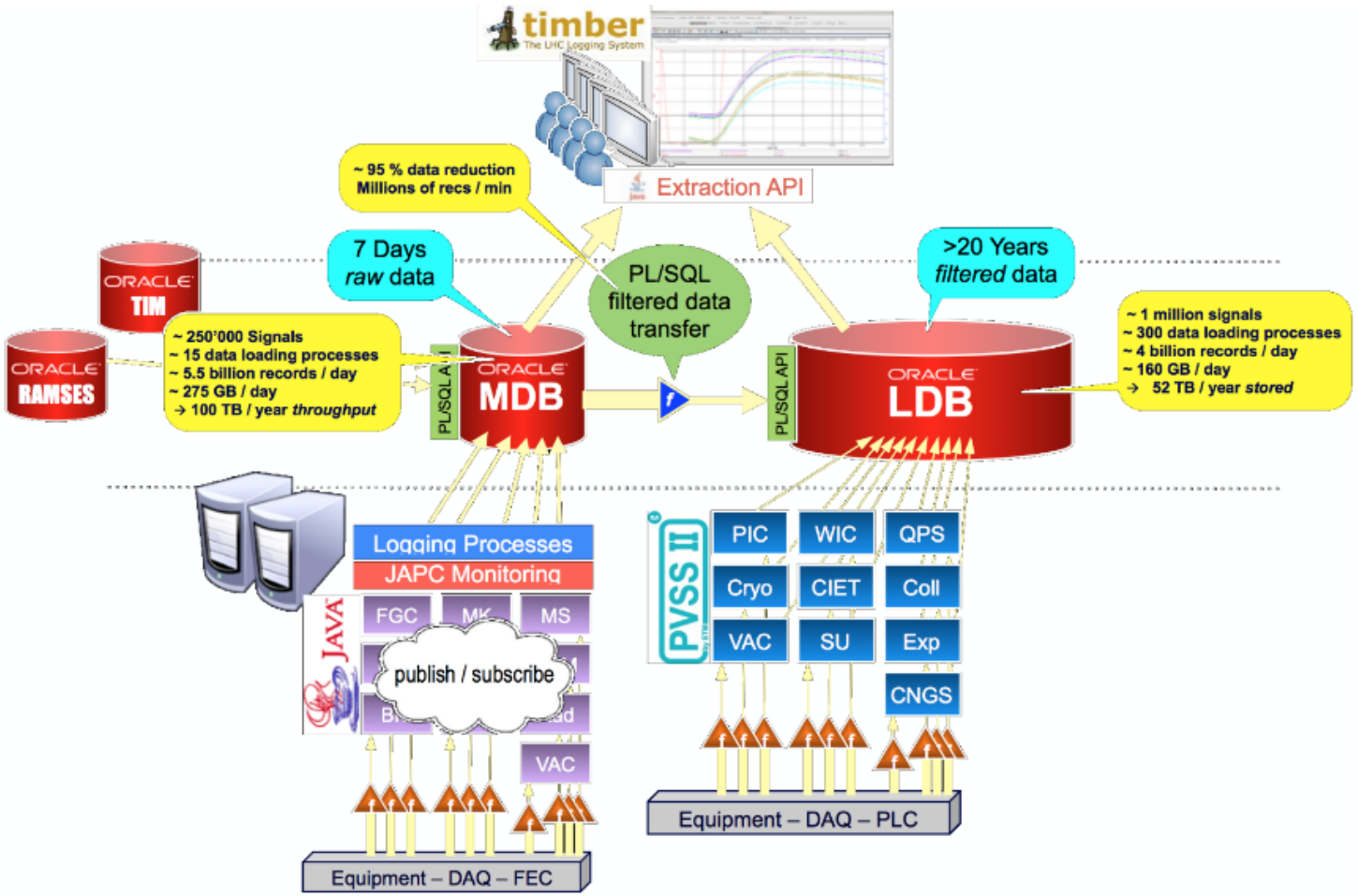
4x Xeon E7-8895 v2 (15 cores each)
2 TB RAM
4.8 TB Flash + 6 x 1.2 TB 10K HDD



RAC – Real Application Cluster

CERN Control Systems

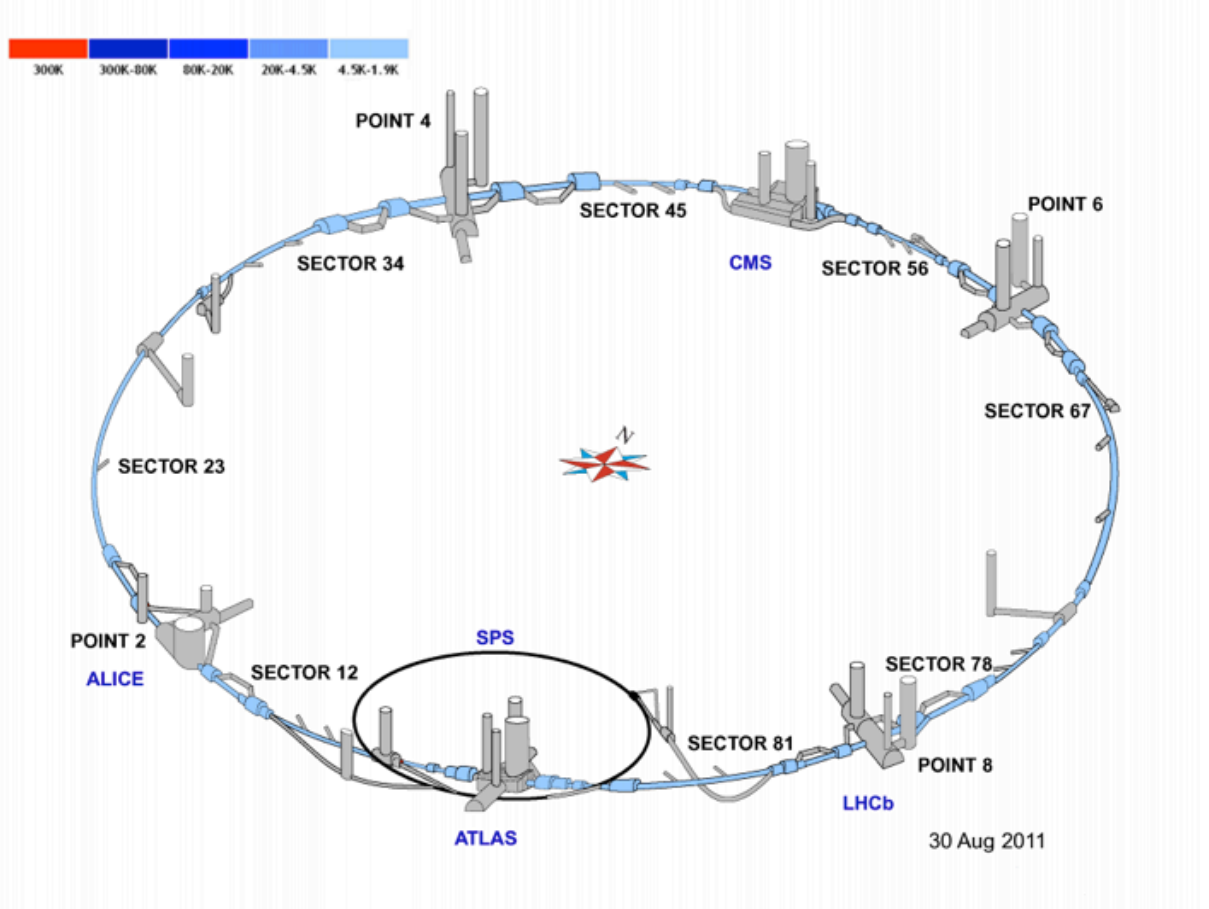
- IoT and Control System
  - **Cryogenics**
  - Vacuum
  - Machine Protection
  - Power Converters
  - QPS
- Accelerator Logging Service
  - ~ 275 GB/day
  - Storing more than 50 TB / year
  - Data acquisition
    - CERN accelerator complex
    - Related subsystems
    - Experiments
  - Around 1 million signals



Largest Cryogenics Installation

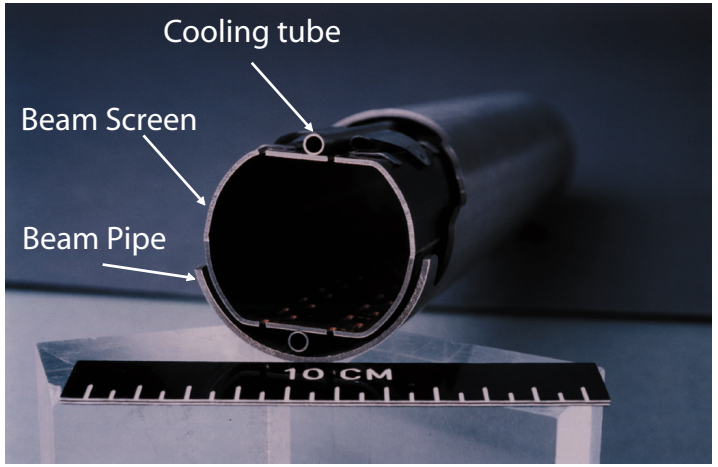
- 50k I/O, 11k actuators, ~5k control loops
- Control:
  - ~100 PLCs (Siemens, Schneider)
  - ~40 FECs (industrial PCs)
- Supervision: 26 SCADA servers

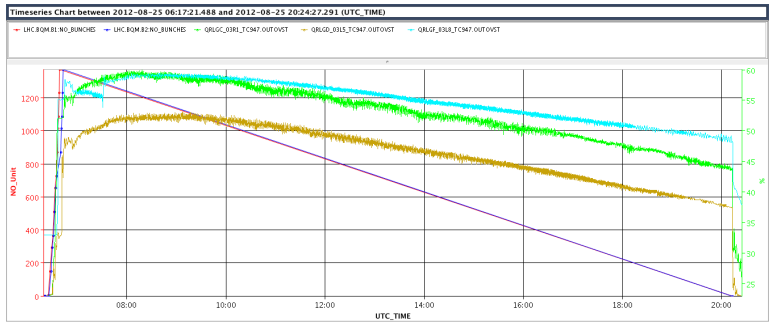
Instrument/Actuators	Total
Temperature [1.6 – 300 K]	10361
Pressure [0 – 20 bar]	2300
Level	923
Flow	2633
Control valves	3692
On/Off valves	1835
Manual valves	1916
Virtual flow meters	325
Controllers (PID)	4833

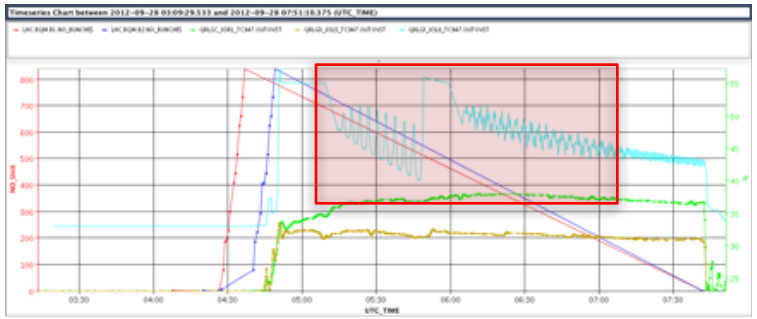


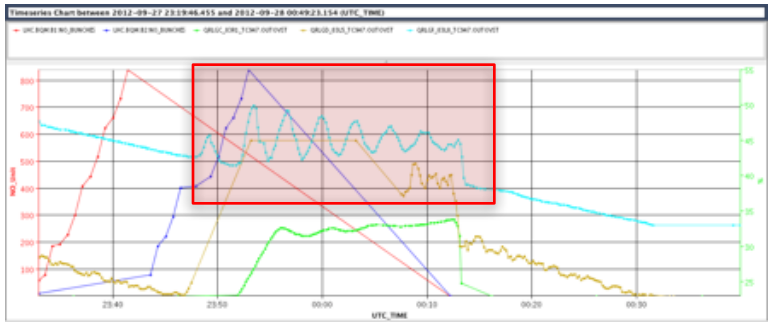
Use Case: Anomaly Detection on Beam Screen

- PID output (time series) segmentation
- Characterization + Feature extraction
- Features based classification



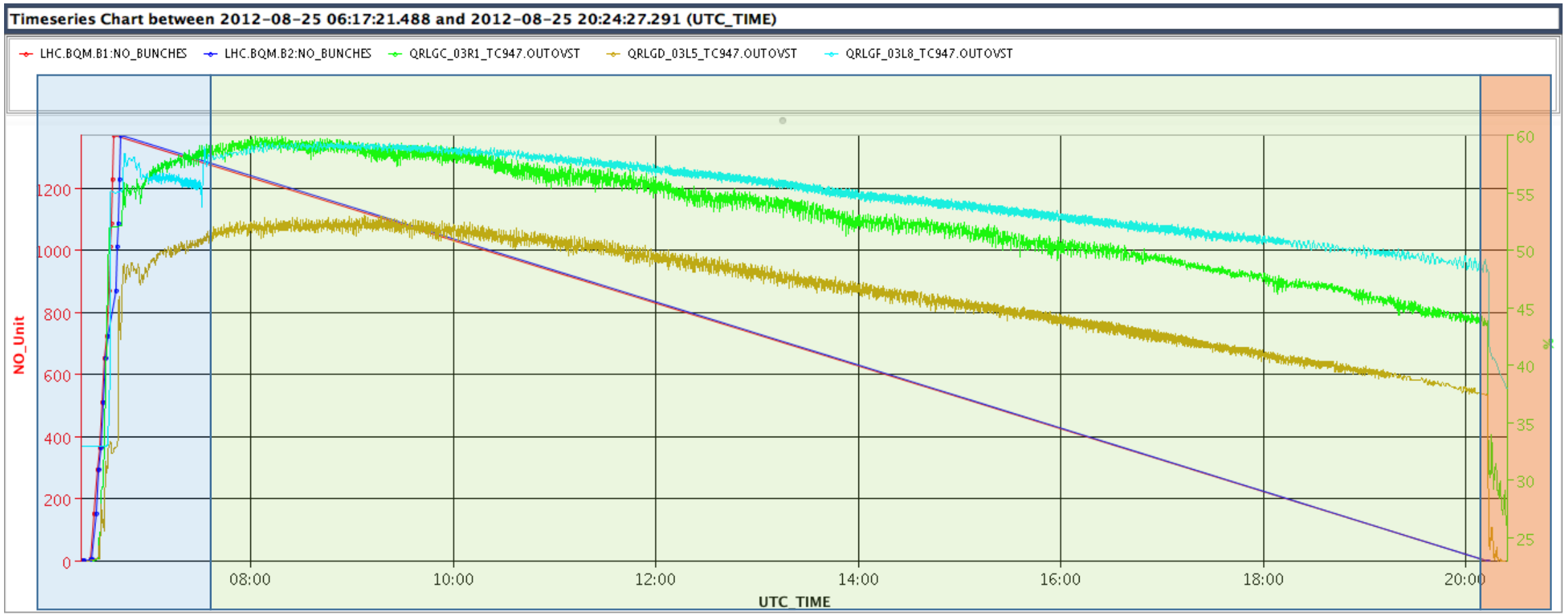






Source: EN-ICE (Benjamin Bradu, Enrique Blanco)

Use Case: Anomaly Detection on Beam Screen



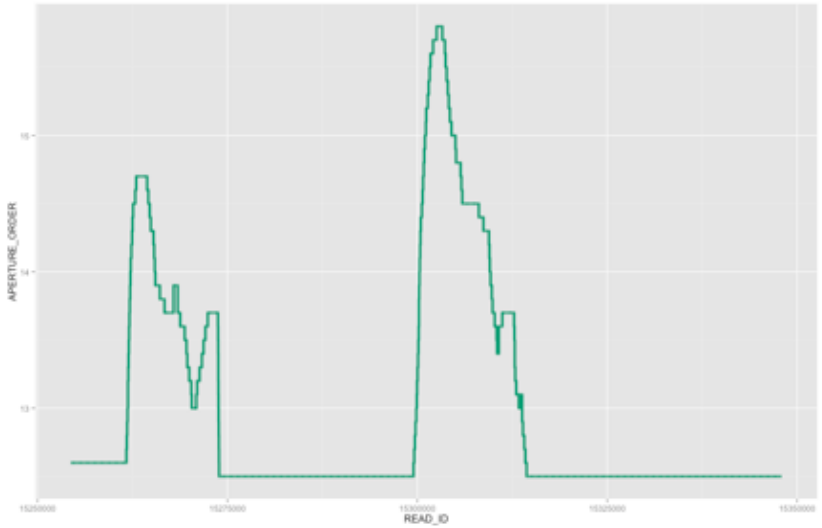
SETUP
INJECT BEAM

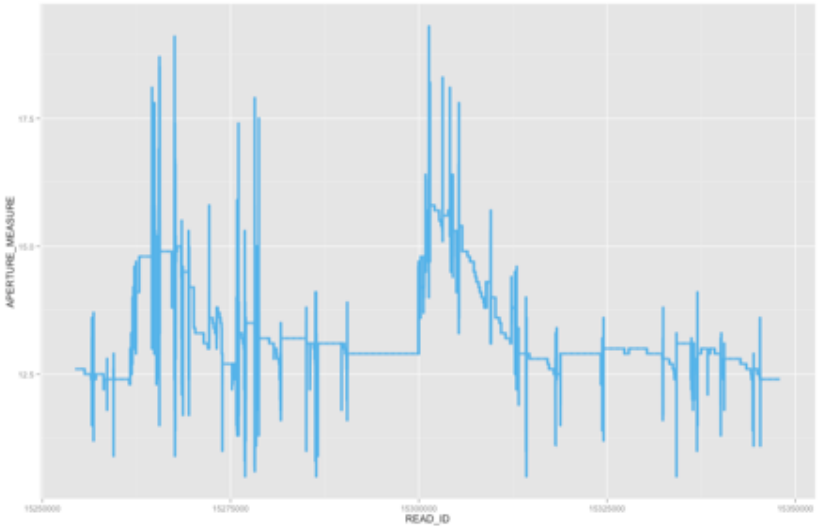
STABLE BEAM

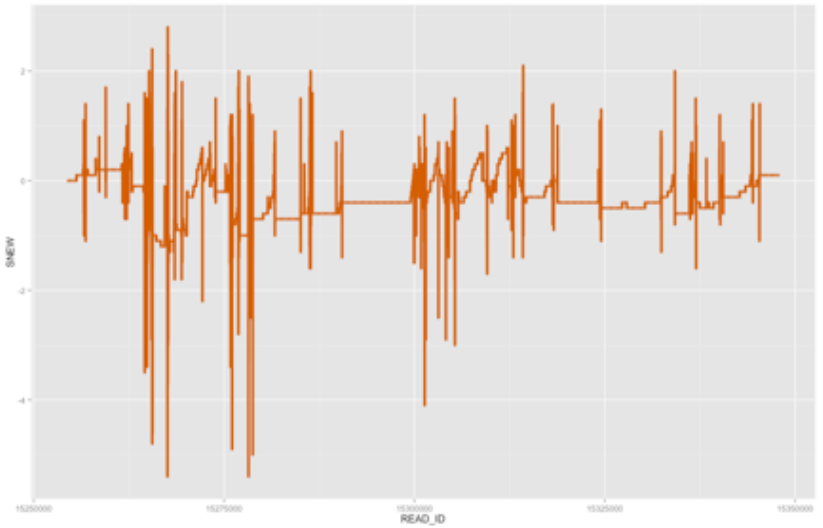
BEAMDUMP

Use Case: Faulty Cryogenics Valve Detection

- What is the objective?
  - Predict faulty valves before they actually fail
- How?
  - Valve receive an aperture order value (**aperture order**)
  - Effective aperture realized by the valve (**aperture measured**)
  - Analyzing the difference between both (**S** = **aperture order - aperture measured**)







Use Case: Faulty Cryogenics Valve Detection

- Signals used:
  - S = aperture order - aperture measured
- Features extractions based on S
  - Variance
  - Percentile 99.9
  - Rope distance – R(S)
  - Noise Band – B(S)
- **Automatic Faulty Valves Detection System**
  - SVM - Support Vector Machine
- The learning set 44 valves
  - aperture order (%)
  - aperture measured (%)

```
f <- function(dat) {

  #Force the order by read_id (PK on table)
  dat<-dat[order(dat$READ_ID),]

  #Calculate the fatures
  s<-dat$APERTURE_ORDER-dat$APERTURE_MEASURE

  valve = unique(dat$VALVE)
  cycle = unique(dat$CYCLE_NUMBER)
  status = unique(dat$STATUS)
  if (status == 0)
    status <- "Normal"
  else if (status == 1)
    status <- "Faulty"
  else status <- "Unknown"

  var=var(s)
  max=max(s)
  min=min(s)

  rope_dist=sum(abs(diff(s)))/length(s)
  pxx<-welchPSD(s-mean(s),seglength=512,two.sided = TRUE)$power
  bs=(sum(pxx)^2)/(2*512*sum(pxx^2))

  #Return the features
  data.frame(Cycle = cycle,
             Valve = valve,
             Status = status,
             Var=var,
             Max=max,
             Min=min,
             Rope_dist = rope_dist,
             Bs = bs)
}
```

Cryo Valves – Extract Features in ORE

```
#Create in-db partitions and proccess them in parallel
ore.valve.features <- ore.groupApply(VALVE_READS_CL_TRA [,c('READ_ID','CYCLE_NUMBER','VALVE','APERTURE_ORDER','APERTURE_MEASURE','STATUS')],
                                     INDEX=VALVE_READS_CL_TRA$CYCLE_NUMBER, FUN = f,
                                     #Return value signature
                                     FUN.VALUE =
                                       data.frame(Cycle = numeric(),
                                                  Valve = character(),
                                                  Status = character(),
                                                  Var = numeric(),
                                                  Max = numeric(),
                                                  Min = numeric(),
                                                  Rope_dist = numeric(),
                                                  Bs = numeric())
                                     ,parallel=TRUE)
```

- **Valves dataset with 6 features**
  - **10.5 M rows**
  - **42 M rows**
  - **168 M rows**

Our experience

- Move the data is very expensive
  - It is faster to process it in the DB with ORE using the appropriate degree of parallelism
- DB nodes already prepared for the workload
  - Memory is a problem for small clients
  - Simplifies the infrastructure
- Write/adapt R code is straight forward
  - Thanks to transparency layer and embedded R execution

Next steps

- Test with DB in-memory features
- Benefit from the hybrid ecosystem
  - Get the best from RDBMS + Hadoop
    - Don’t forget about current application layer working with the DB
  - Deploy in streaming
- Predictive Model Markup Language (PMML)
  - Compatible across multiple languages (R, Python, Spark, Java, etc.)

Conclusions

- R ecosystem is very good for Data Analytics
- ORE offers a good solution to get more from your DB deployment
  - No pain adapting code
  - Follows R standards
- Avoid moving data when possible
- Hybrid systems look quite promising to cover multiple scenarios

www.cern.ch