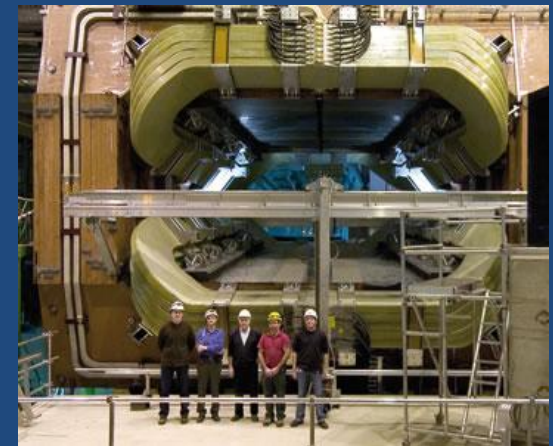
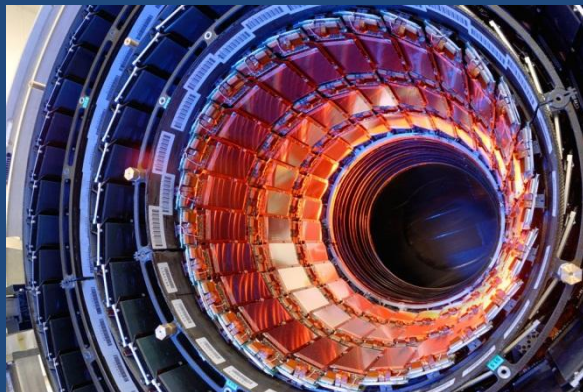


European Organization for Particle Physics

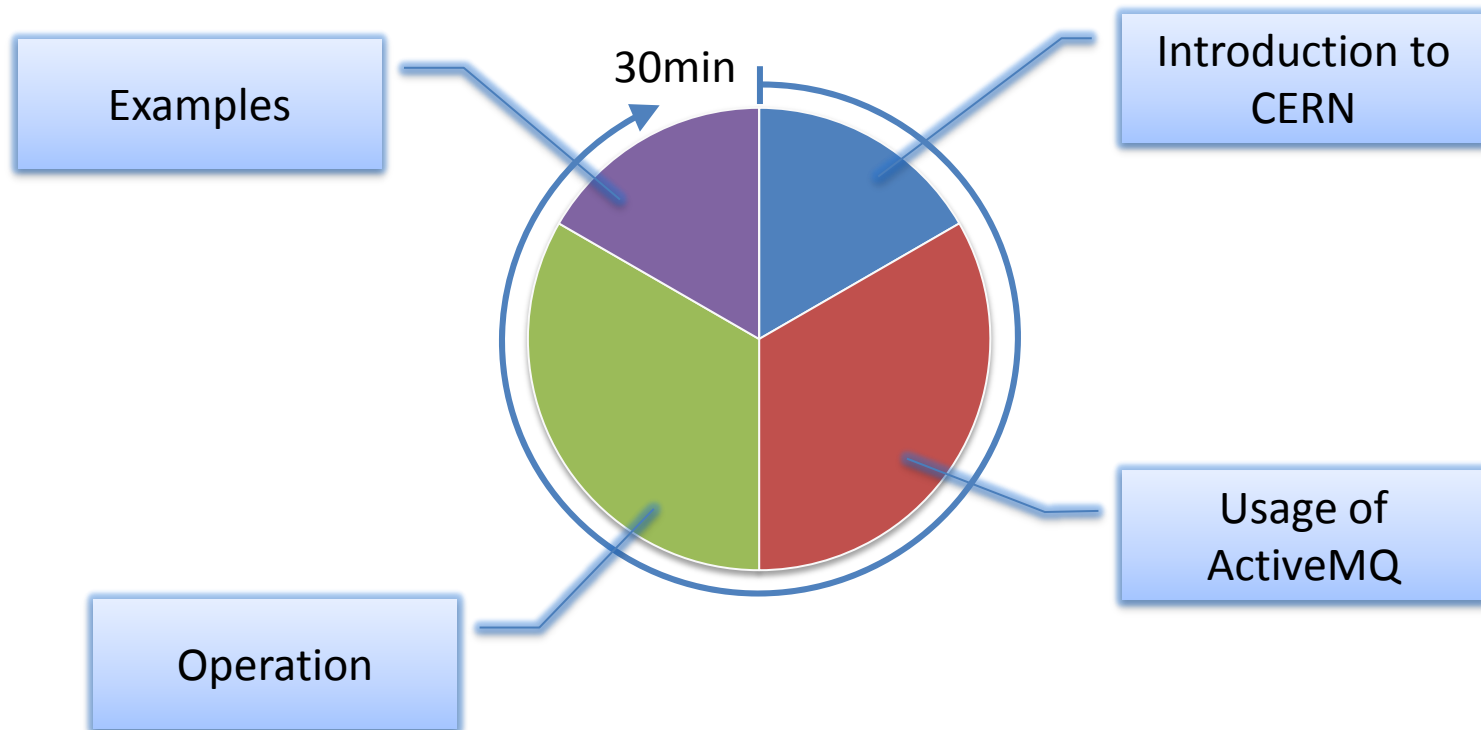
The coolest place on earth



Large Scale Messaging with ActiveMQ for Particle Accelerators at CERN



Overview



About the Speaker

Member of CERN Beams Controls Group:
Responsible for JMS Service Middleware
Developer for Alarm and Monitoring System

Previous Activities:
Large Storage Systems
Grid Environment

felix.ehm@cern.ch



What is CERN ?

A European Organization for Nuclear Research

originally : **C**onseil **E**uropéen pour la **R**echerche **N**ucléaire

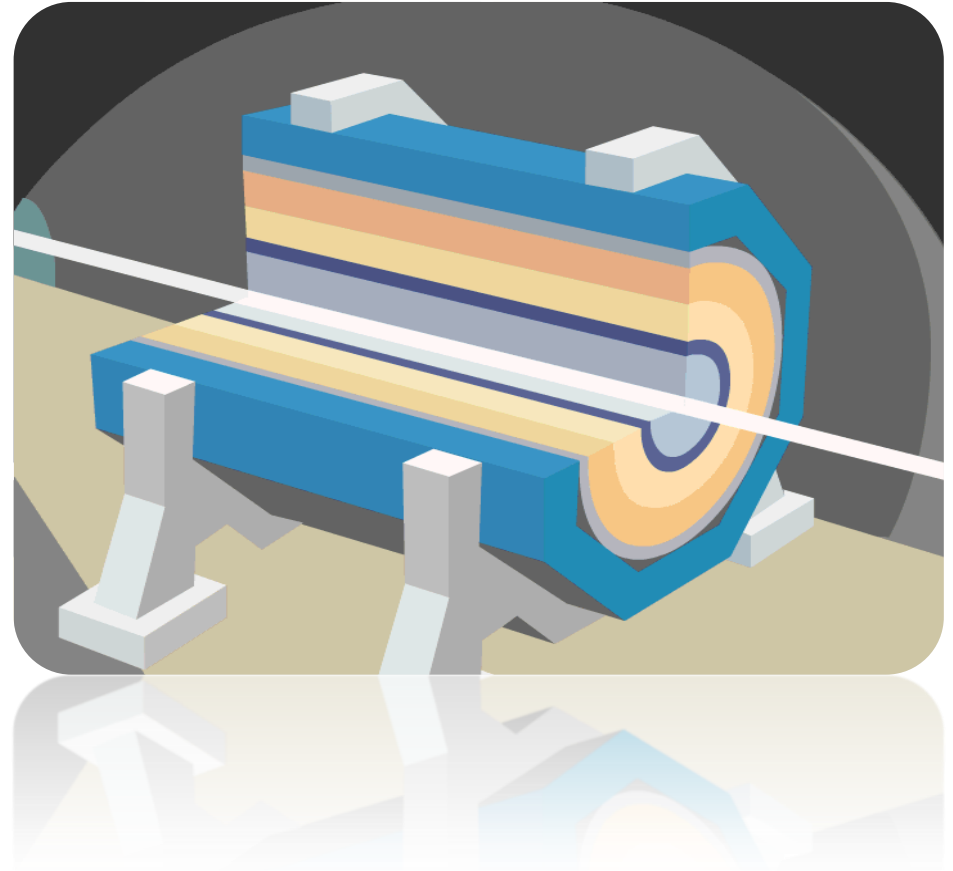
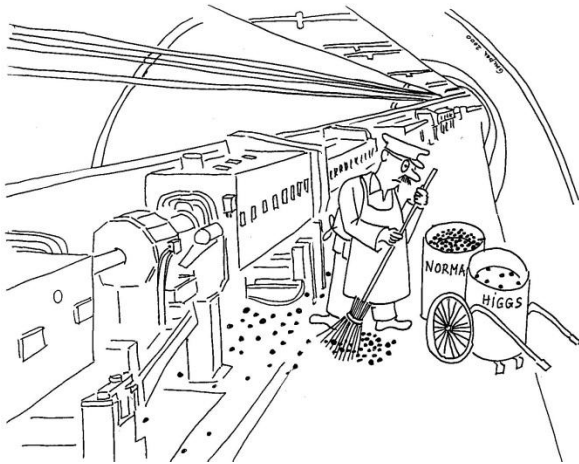
- Founded in 1954
- Based in Geneva, Switzerland
- 3'000 staff members
- 8'000 visiting scientists
- Financed by its member states

Twenty Member States



Large Machines for High Energy Physics

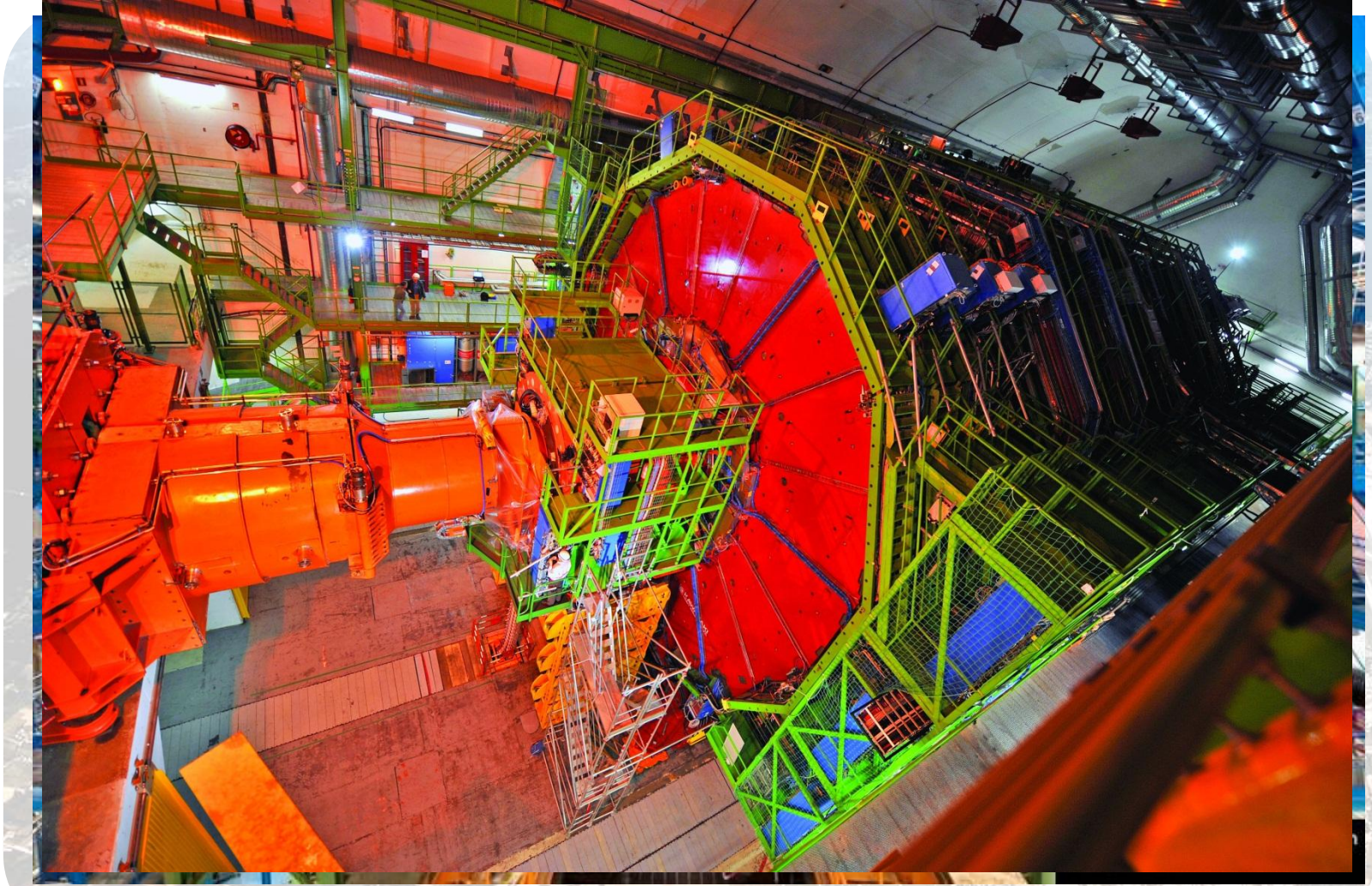
Particle colliders for tracing “fragments”



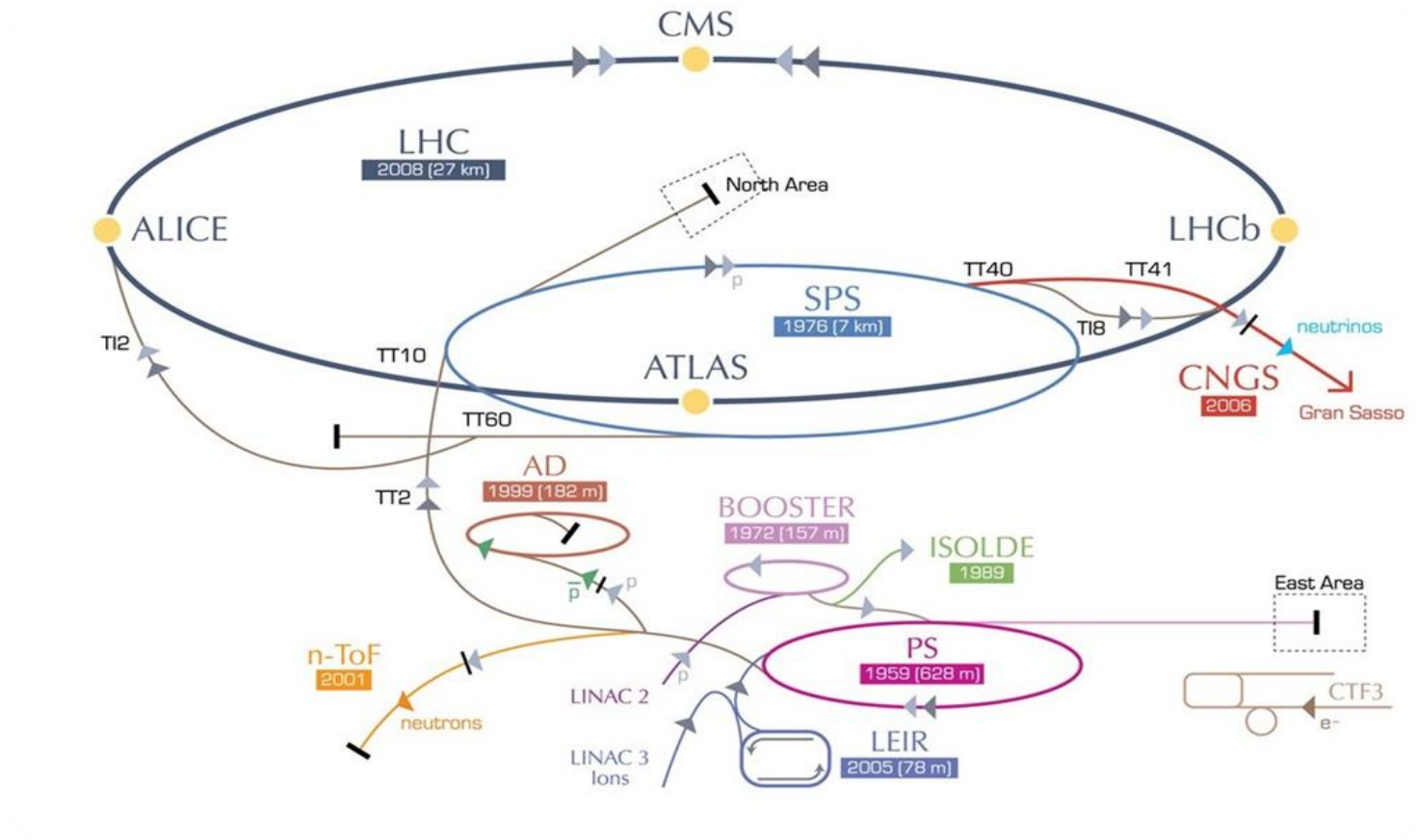
The CERN Campus



The Large Hadron Collider



But the LHC is not the only accelerator



A lot of systems to control

Controls
Computers



85'000 Devices
> 2 Million I/O Endpoints

*Much more when
including subsystems!*

Electricity



Cryogenics



Magnets



Safety



Cooling



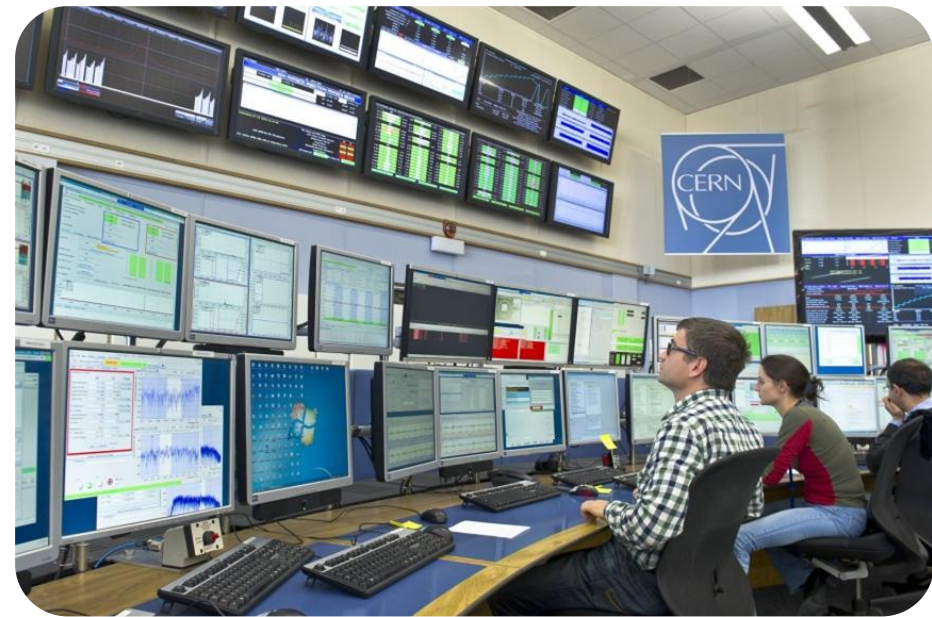
Ventilation



Vacuum



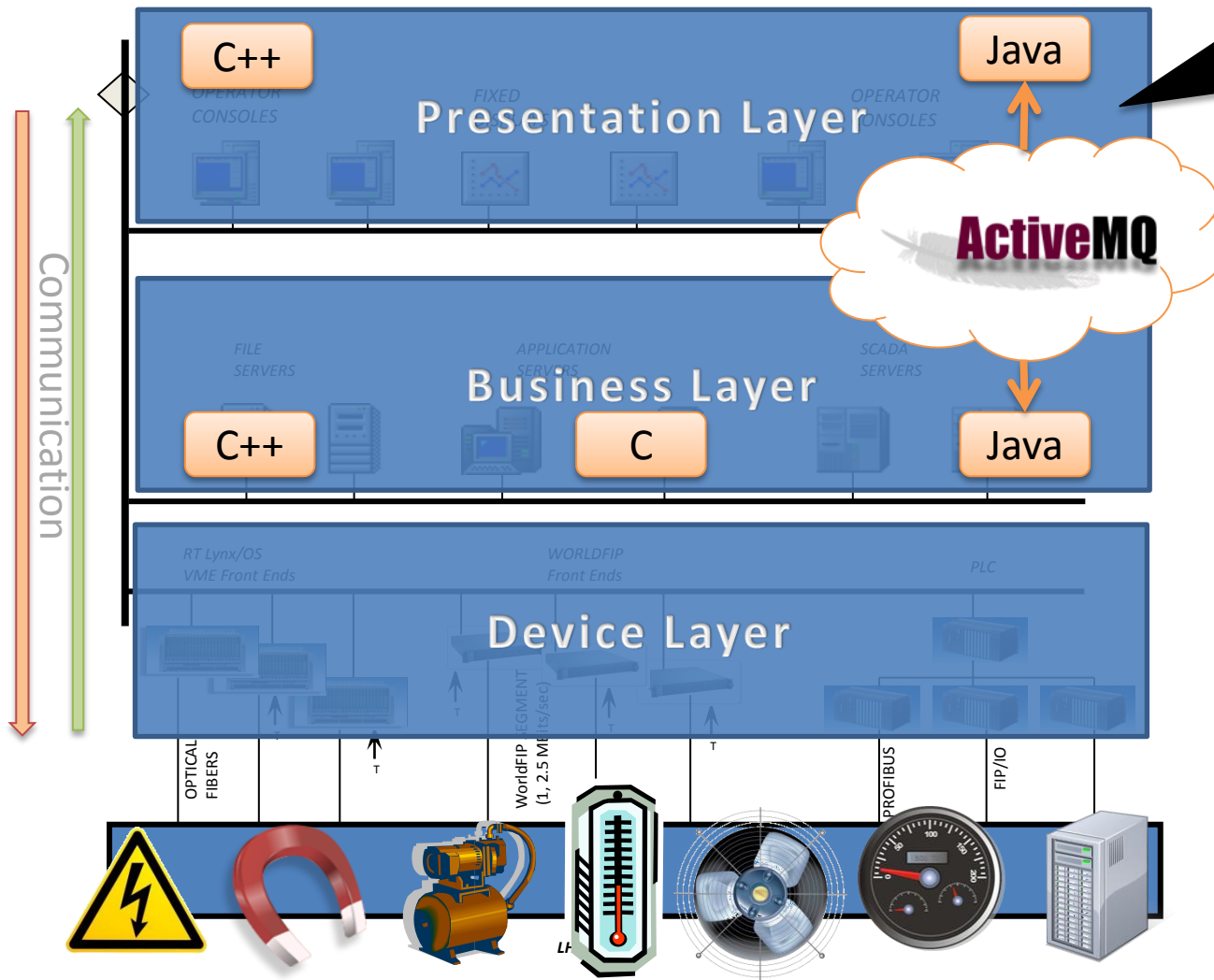
Everything from one central point: The CERN Control Centre



USING ACTIVEMQ FOR ACCELERATOR CONTROLS



Controls Architecture



JMS Purpose:
Reliable and
scalable transport of
data between Java
processes

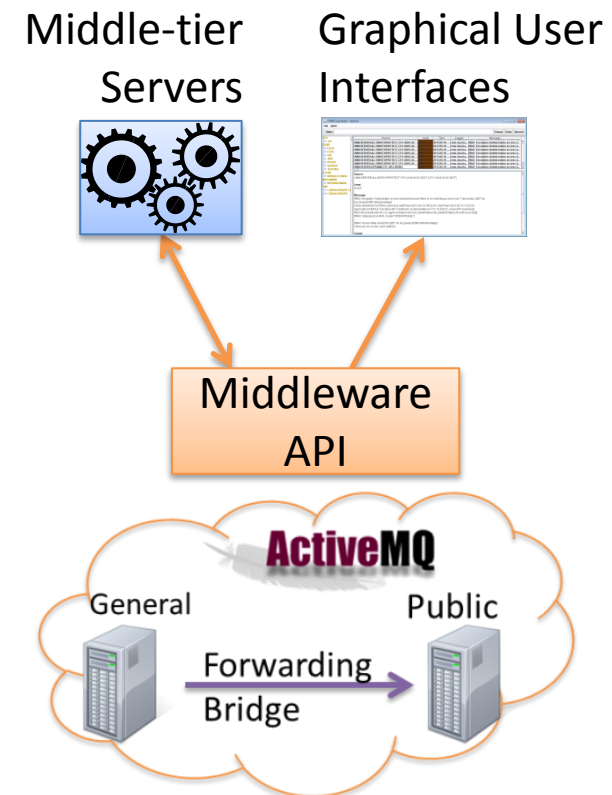
History

- Early use of ActiveMQ already in 2005
 - We were looking for a free JMS solution
 - Apache? Can't be bad!
- Why OpenSource ?
 - Low Costs
 - We can **read and check** the code
 - We can add / fix code

History – First Setup

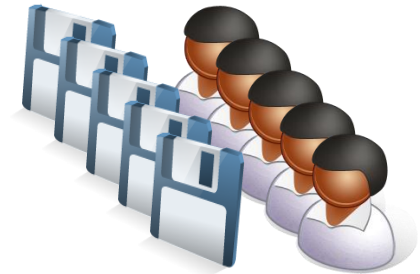
- Clients
 - Java **middle-tier servers** and **GUIs**
 - Data access for Clients via Middleware API
 - Two interconnected Brokers
-

Simple Setup, few projects, little data, easy to use. Did the job!

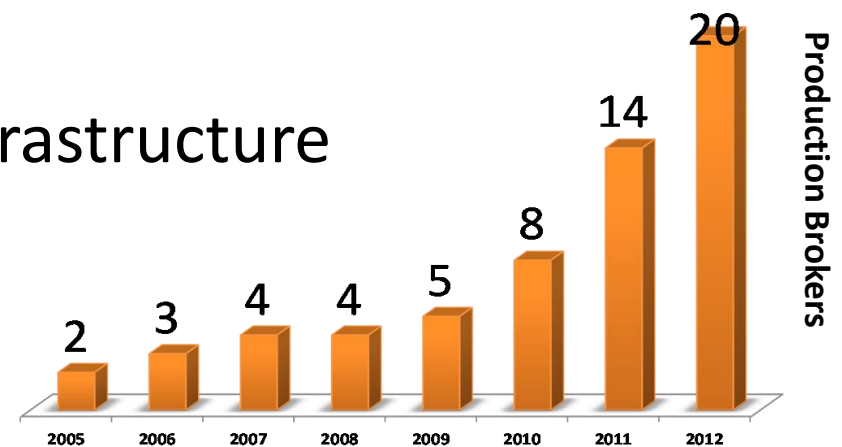


History - Evolution

- But: Service suffered by its own success
 - **More users** and **more data** was sent around
 - Higher **QoS** was requested
 - Redundancy
 - Queues, persistent messaging and global transactions (XA)
 - Support for **non Java** clients



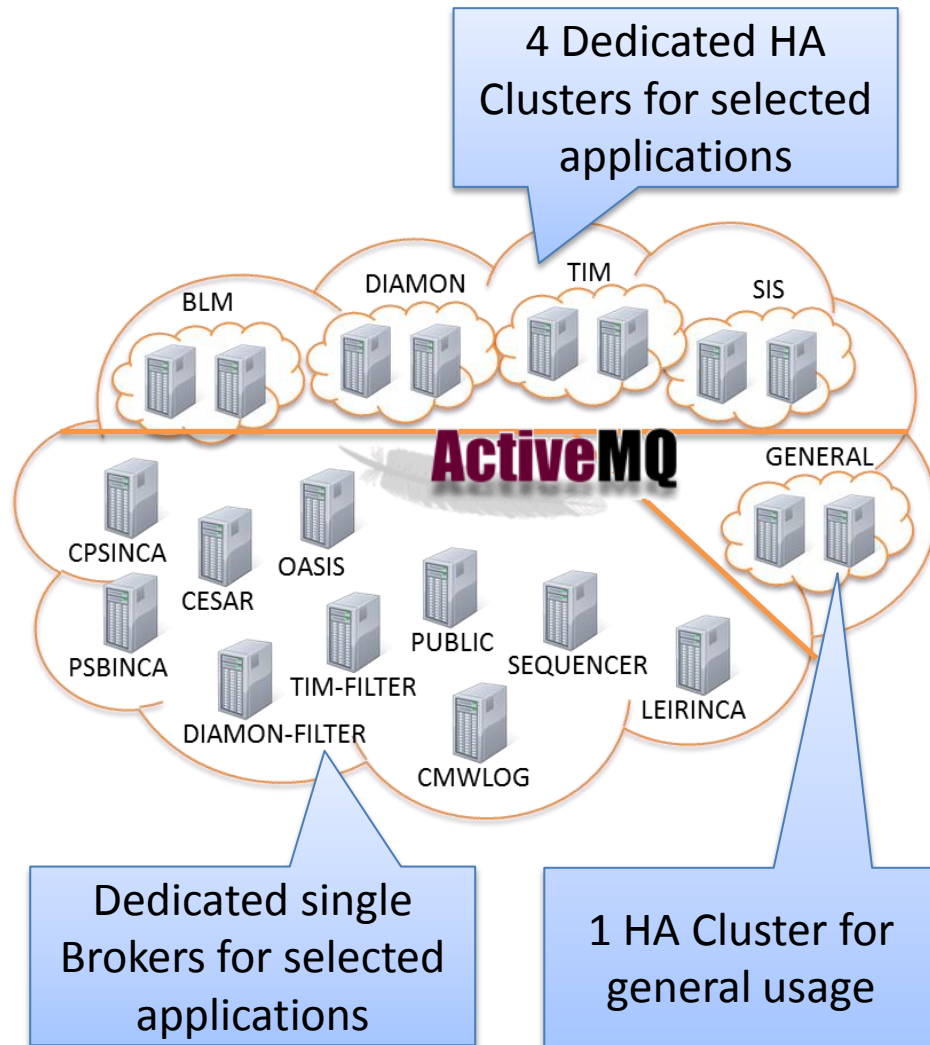
- We needed to adapt the infrastructure and needed to **SCALE**



Deployment Today

- 20 Production Brokers
 - 10 single Brokers
 - 5 HA Clusters
- No one large cluster but **manageable** entities
- **Vital part** of beam instrumentation and operation

**No JMS –
No Particle Physics !**



Deployment Today - Setup



- HA Clusters
 - 2 equivalent broker members, no master-slave
 - Two (real) machines with **separate network links**
- Single Broker
 - For projects which do not require HA
 - Deployed on same (real) machine as middle-tier server
- Local secured network together with Producers and Consumers
- Fuse ActiveMQ Distribution since 2010
 - Issues solved faster in service packs

Deployment Today

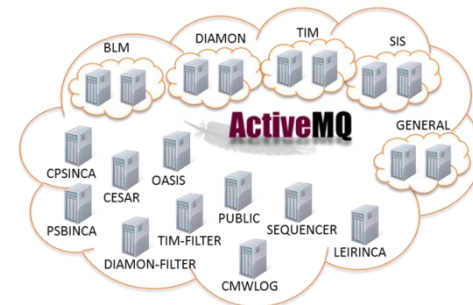
Some numbers :

- *300 Applications*
- *4'400 Connections*
- *40'000 Subscriptions*
- *85'000 Topics*
- *68'000 Consumers*
- *8 Million msg/h IN,
3.5M msg/h OUT*

Archived Uptime
in 2011: 99.98%

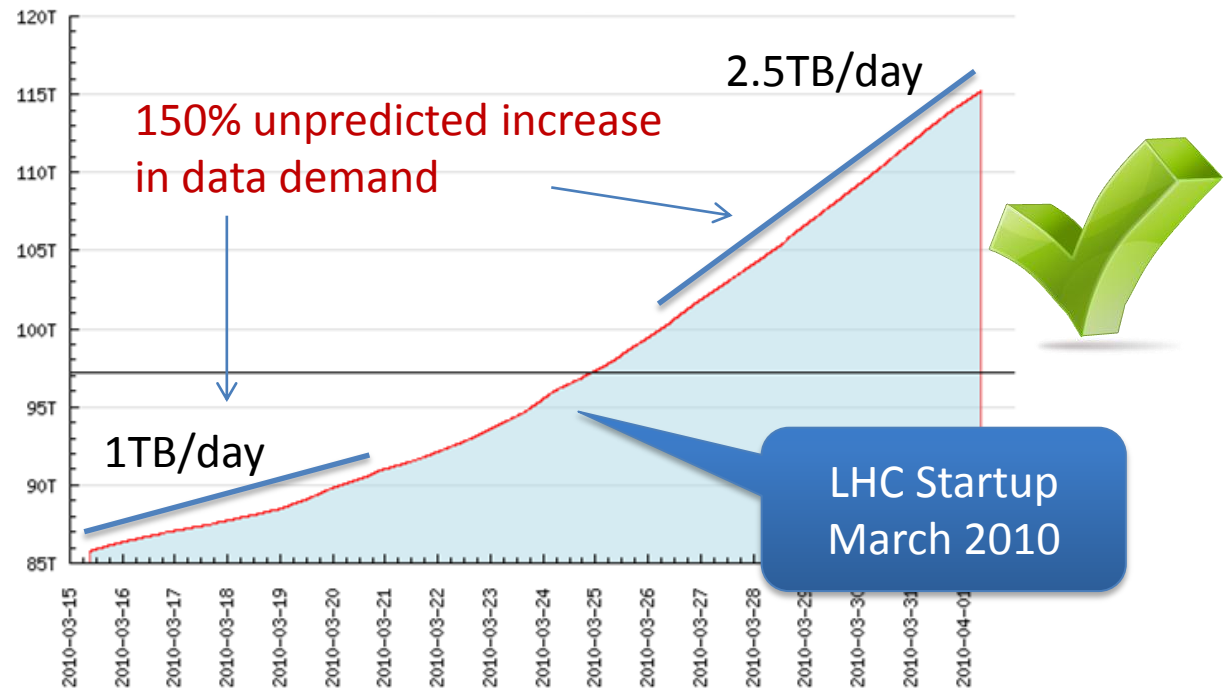


Not all data which
is produced is
consumed



Example of Data Handling during LHC Startup 2010

Output Data Handling for 1 Broker of the General Broker Cluster



Usage Cases

Usage Case 1

- **Payload 2MByte**
- 1 msg/sec
- 1 Topic
- 20-30 Java Clients

Usage Case 2

- Payload 500Bytes-1KBytes
- 30-200 msg/sec
- 120 Queues
- **4 Million XAS/day**
- 2 Clients

Usage Case 3

- Payload 200Byte - 10KByte
- **50-4500 msg/sec**
- **10'000 Topics**
- 2-5 Clients

Usage Case 4

- Payload <200 Bytes
- <10 msg/hour
- 1 Queue
- **< 10 STOMP Clients**

ActiveMQ

Scalability

Reliability

Versatility

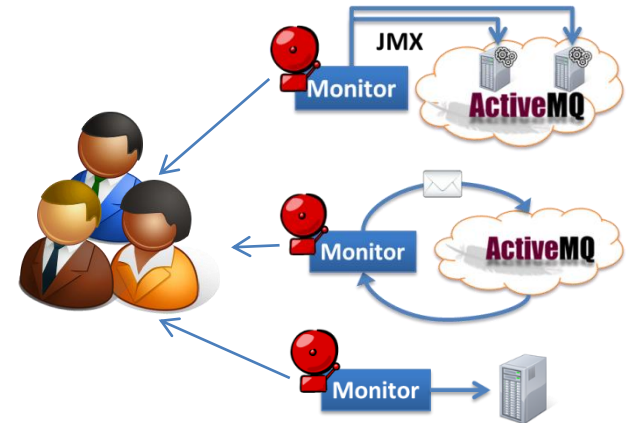
Flexibility

OPERATION



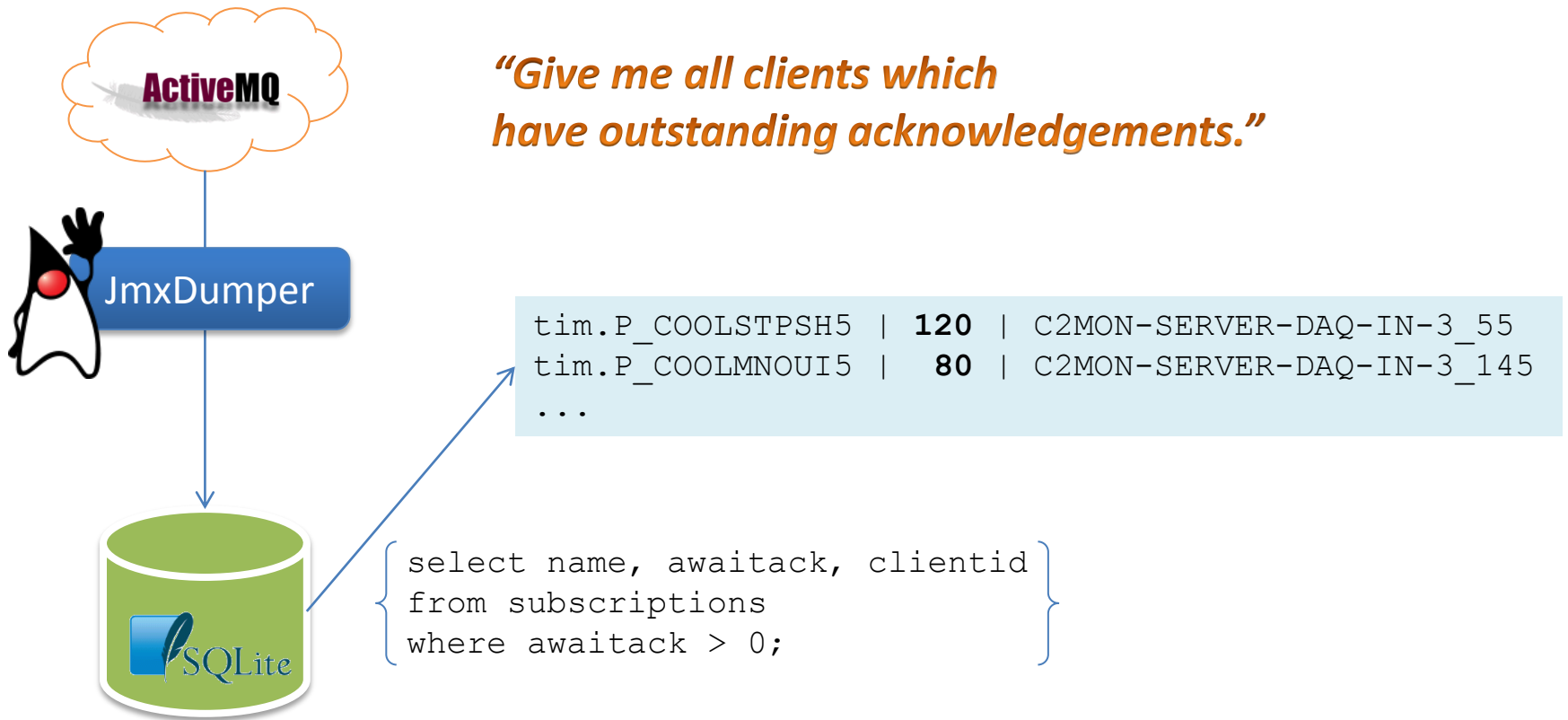
Management of Brokers

- Monitoring
 - via JMX
 - Submission of test message
 - Host machine monitoring
- Deployment
 - Rollout scripts for deploying/rollback
 - Configuration changes tracked in SVN



Diagnostic Tools

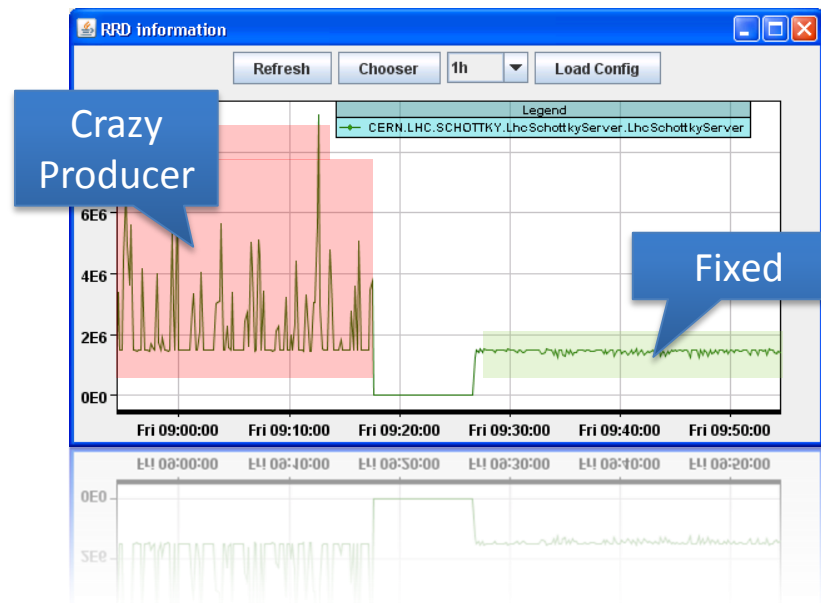
Dump JMX information into SQLite database for **fast, easy** and **intuitive** access.



Diagnostic Tools

Traffic Monitoring Tool

- Collecting information by **listening** on Topic
 - Message size & speed
- Allows **history view** on
 - Average Message throughput
 - Average Message size



Lessons learnt - Operation

- Separate usage cases from each other
 - Easier than implementing per destination policies
 - Broker restart does not affect others
- More clients, more memory
- Broker does not die when hitting memory limits
- Check if you need all features
 - Maintenance overhead vs. failure probability & effect

Lessons learnt - Configuration

- Disable DedicatedTaskRunner: takes many threads
- Reduce the PrefetchLimit on clients side
- Couldn't get Broker discarding messages
 - ConstantPrefetchLimit unset + brain-dead client
= Out of Memory
- Memory is important
 - But GC takes longer
 - New Java GC options help

```
-XX:+UseConcMarkSweepGC  
-XX:+CMSIncrementalMode  
-XX:+CMSIncrementalPacing  
-XX:CMSIncrementalDutyCycleMin=0  
-XX:CMSIncrementalDutyCycle=10  
-XX:+HeapDumpOnOutOfMemoryError
```

EXAMPLES



LHC Status Displays

LHC Page1 Fill: 1059 E: 450 GeV 25-04-2010 21:48:48

PROTON PHYSICS: INJECTION PROBE BEAM

BCT TI2: 0.00e+00 **I(B1):** 1.39e+10 **BCT TI8:** 0.00e+00 **I(B2):** 1.94e+10

TED TI2 position: **BEAM** TDI P2 gaps/mm up: 9.00 down: 8.93

TED TI8 position: **BEAM** TDI P8 gaps/mm up: 8.76 down: 8.77

FBCT Intensity Updated: 21:48:48

Comments 25-04-2010 20:50:54 :

injected one bunch per beam
correcting machine

BIS status and SMP flags

	B1	B2
Link Status of Beam Permits	false	false
Global Beam Permit	true	true
Setup Beam	true	true
Beam Presence	true	true
Moveable Devices Allowed In	false	false

LHC Operation in CCC : 77600, 70480

<http://imgtfy.com/?q=cern+vistars>

Remote Control for Video Viewer

Video Viewer

Access Request



"Show Cam 1"

Business Logic

ActiveMQ

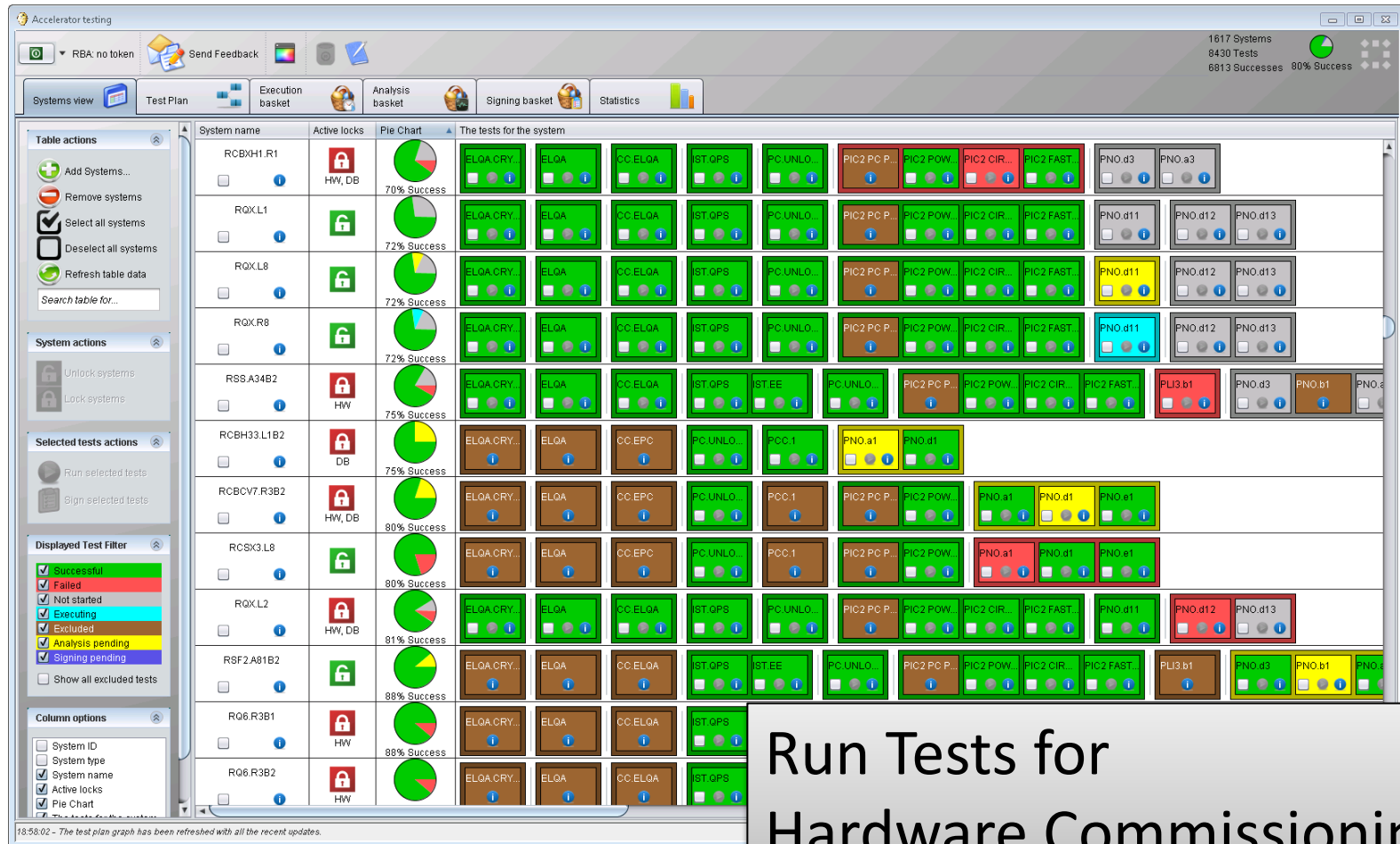


Button Panel

Powered by VLC

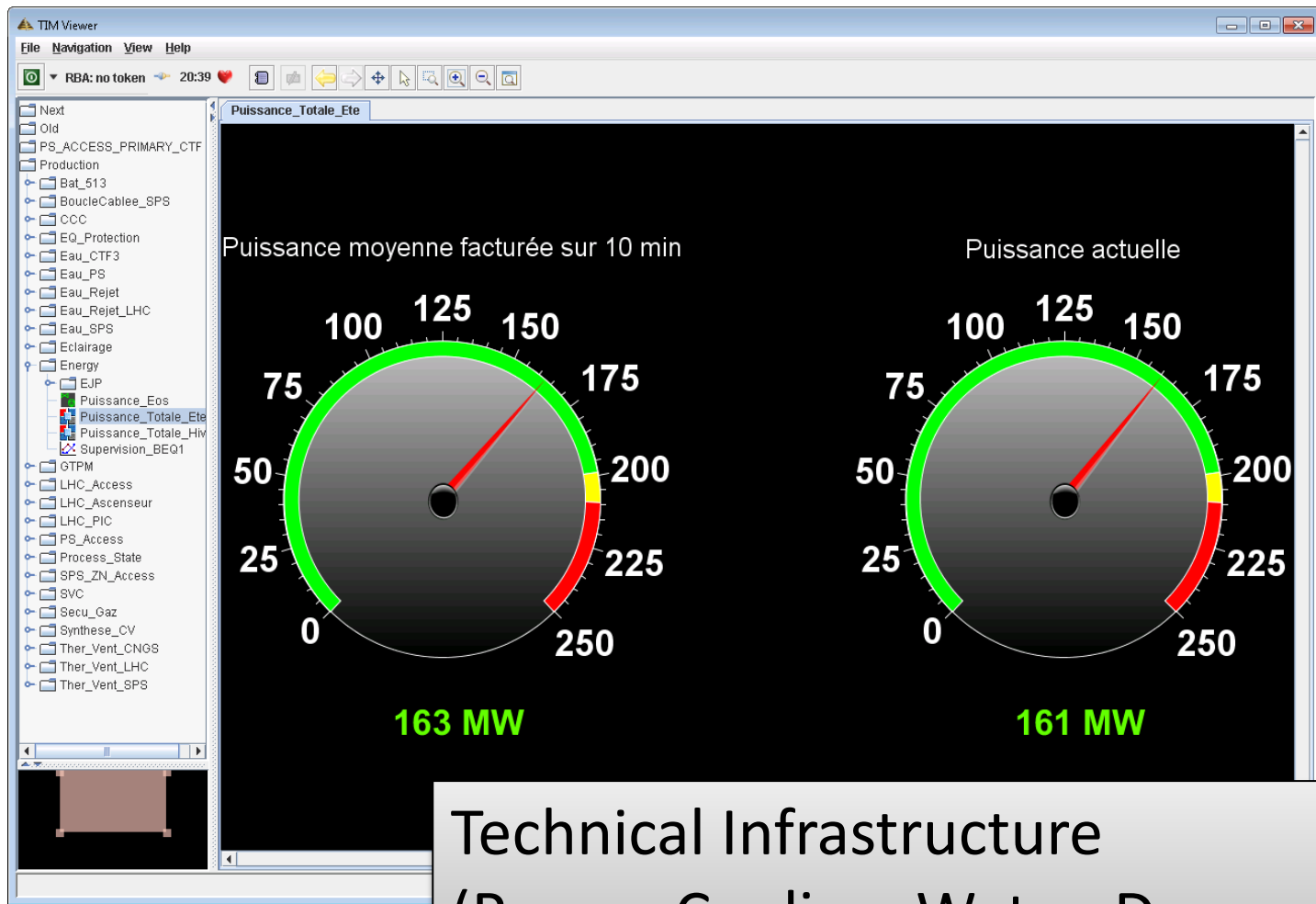


“Bamboo” for Accelerators



Run Tests for
Hardware Commissioning

Animate Synoptic Displays



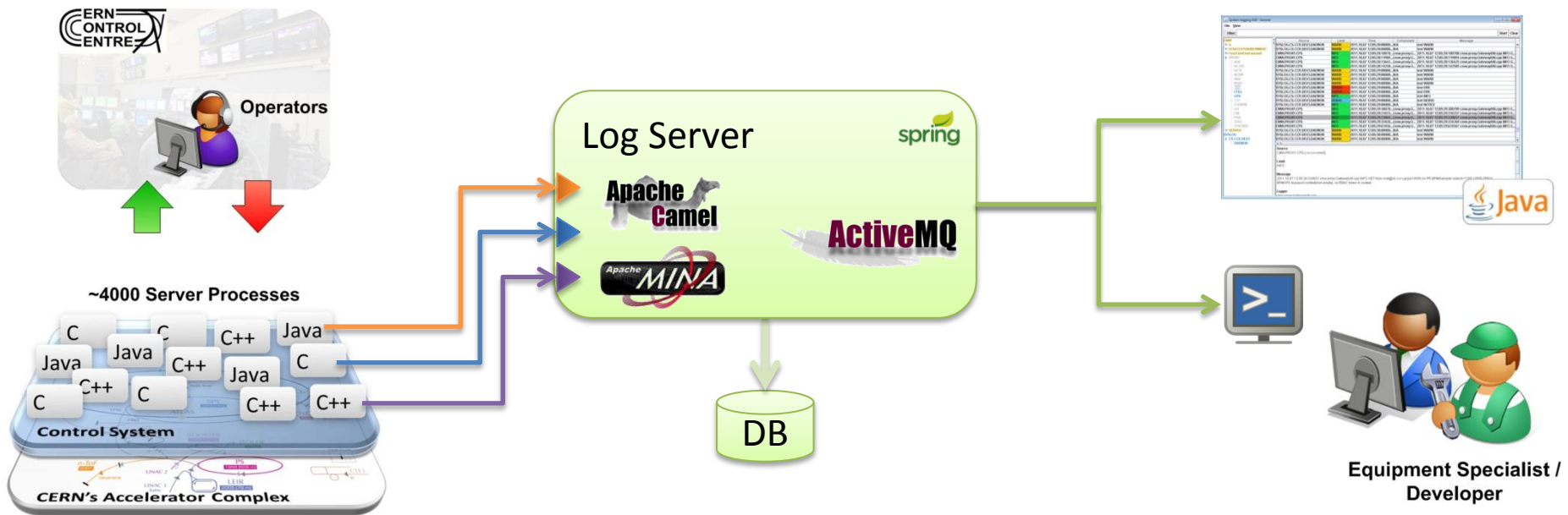
Technical Infrastructure
(Power, Cooling, Water, Doors,...)

Camel for Unifying Log Events

Finding/Debugging a problem becomes cumbersome!

Collecting and **unifying** log messages in one **central** place

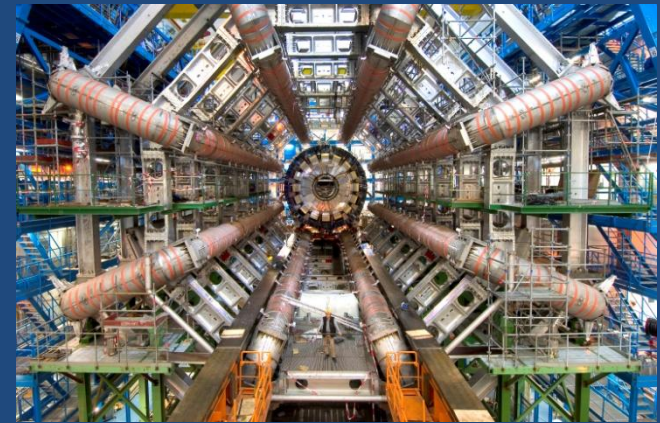
Easy **correlation** of events among **many** services



Finally

A great thanks to all developers !





European Organization for Particle Physics
Take part!

