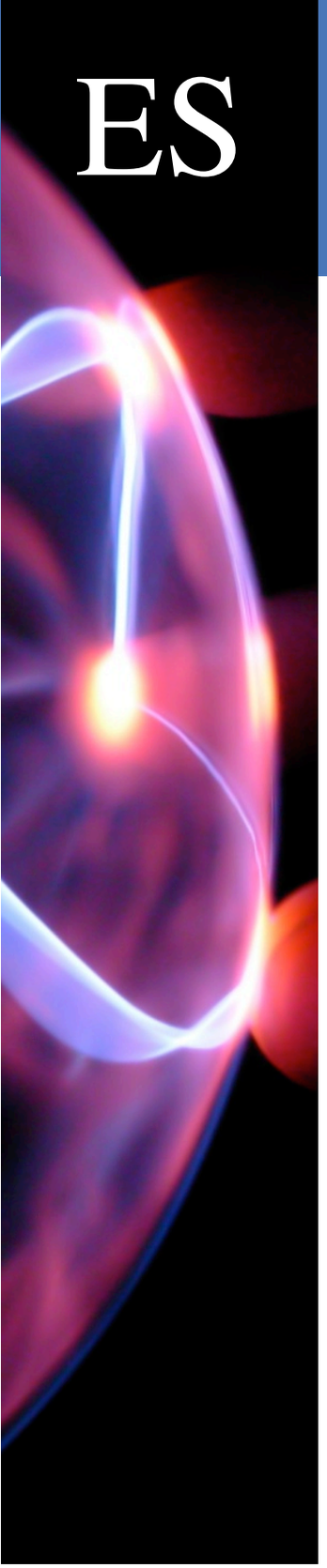
CMS usage of NoSQL databases

Mattia Cinquilli

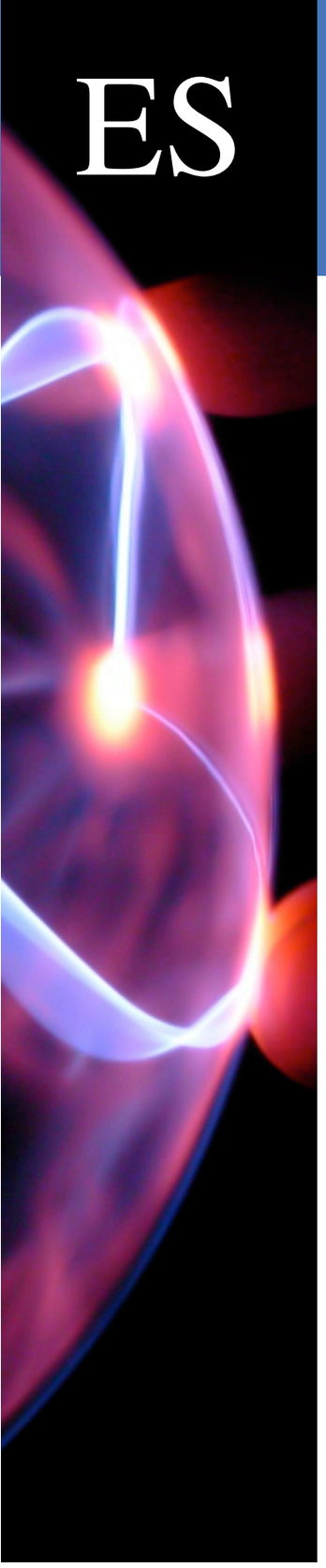
IT-ES-DNG Section Meeting
31-05-2011



Outline

- NoSQL: CouchDB & MongoDB
- DAS
  - Architecture
  - NoSQL motivations
- WMAgent
  - Architecture
  - NoSQL motivations
- Conclusions

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

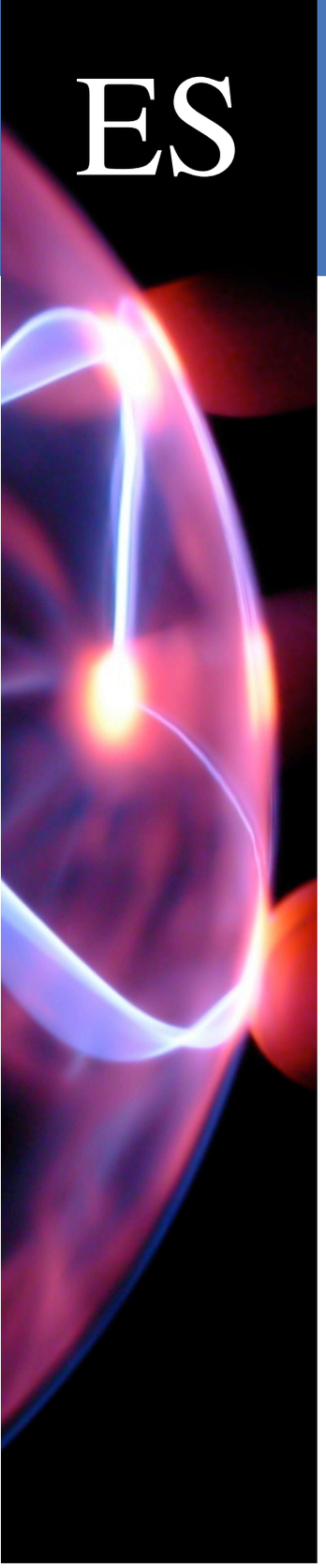


CouchDB

Features:

- Document Store.
- Written in Erlang.
- Manages a collection of JSON documents.
- Exposes a RESTful HTTP API.
- Implements Multi-Version Concurrency Control.
- Distributed Architecture with Replication.

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



MongoDB

mongoDB

Features:

- Document Store.
- Written in C++.
- Manages a collection of JSON documents.
- Exposes a RESTful HTTP API.
- Provides atomic operations on fields.
- Supports dynamic queries with automatic use of indices.

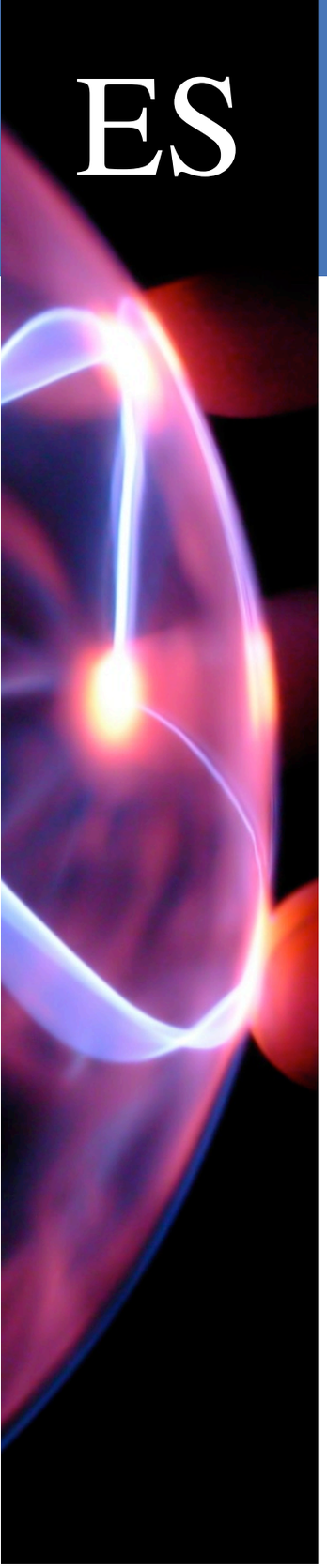
CERN IT Department
CH-1211 Geneva 23
Switzerland
www.cern.ch/it

DAS

Data Aggregation System

Mattia Cinquilli - IT-ES-DNG Section meeting - 31 May 2011

DAS

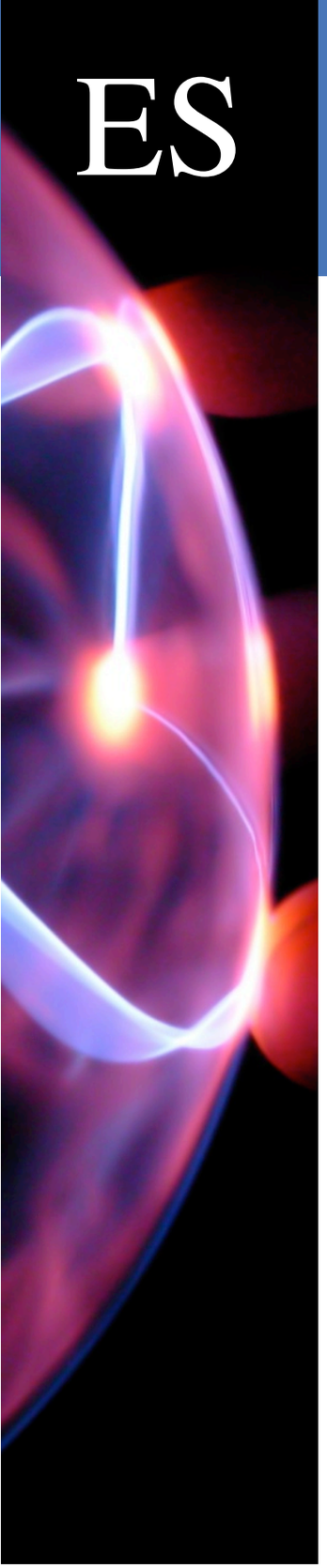


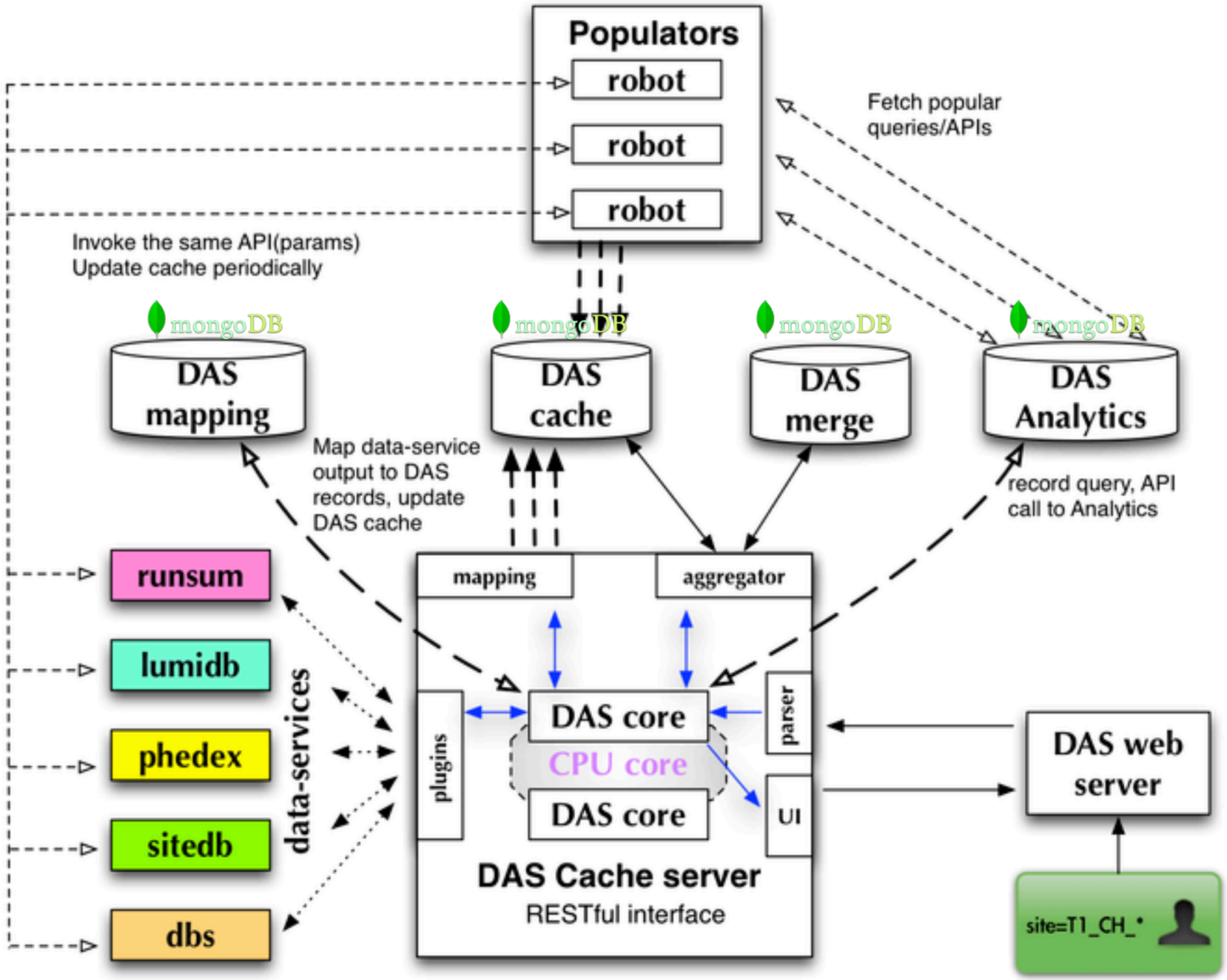
- The Data Aggregation System provides to the users the ability to search and aggregate information across different data-services without knowing the specific data sources.
- (On demand) Fetches and aggregates meta-data information from existing CMS data-services under one single point by using services API and preserving specific security policies.
- Main key points and features:
  - transparent central meta-data repository
  - acts as proxy and caching layer
  - keyword search based system, with conditional operators
  - provides a common data representation

"Data Aggregation System - a system for information retrieval on demand over relational and non-relational distributed data sources"
Presented at CHEP 2010
Ball, G. (Imperial Coll., London); Kuznetsov, V. (Cornell U.); Metson, S. and Evans D.

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

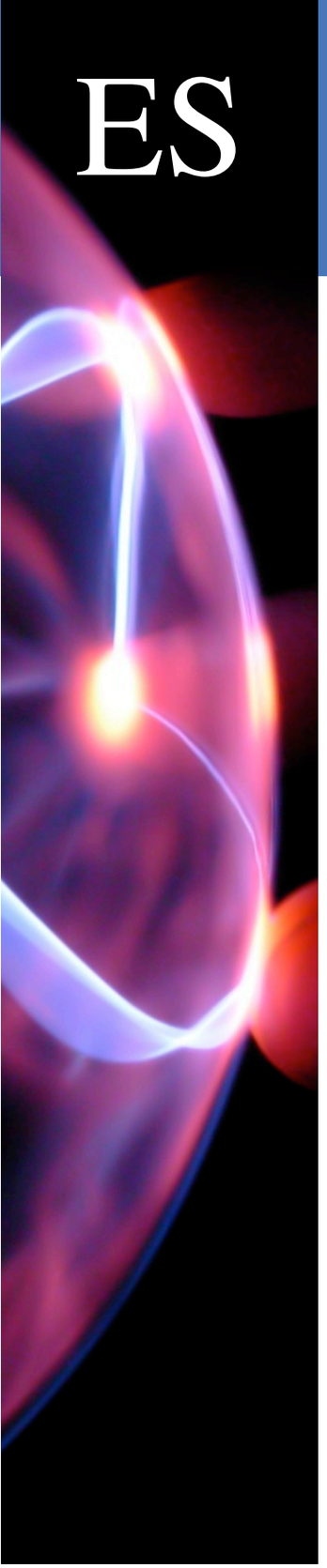
Architecture





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

Motivations of MongoDB

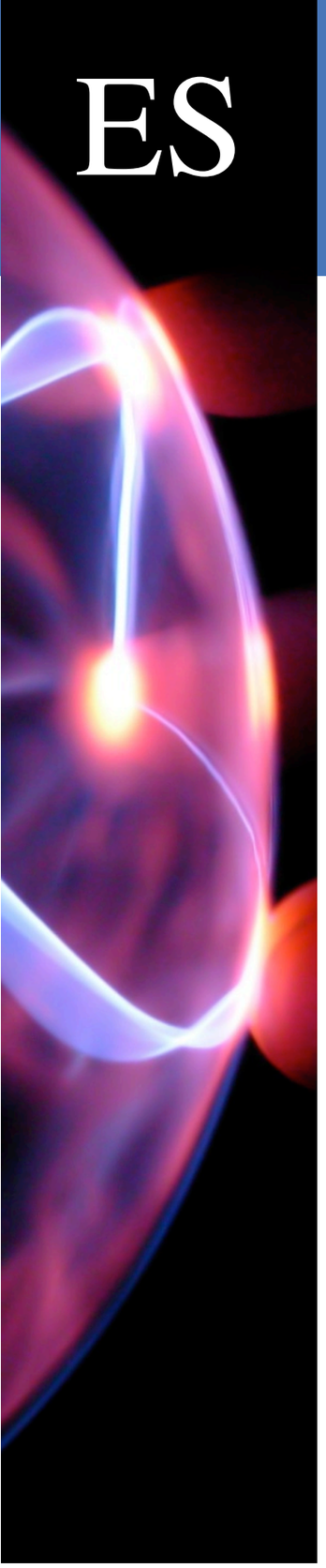


- No needs of relational databases:
  - DAS does not require data preservation and transaction capabilities
  - Dynamic type of stored meta-data objects (no predefined schema)
- Document oriented databases (MongoDB, CouchDB)
  - Schema-less: arbitrary document structure storage
  - Replication and failover features
- MongoDB advantages:
  - Support of dynamic queries (like relational databases)
  - Full-text indexes (index defined on every single word of a field)
  - Auto-sharding

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

WMAgent (WM = Workload Management)

Mattia Cinquilli - IT-ES-DNG Section meeting - 31 May 2011



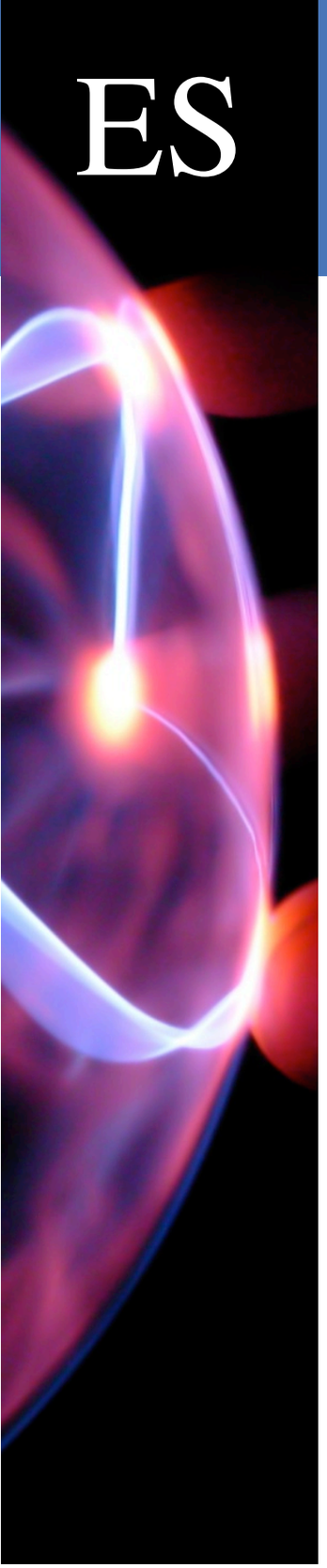
WMAgent

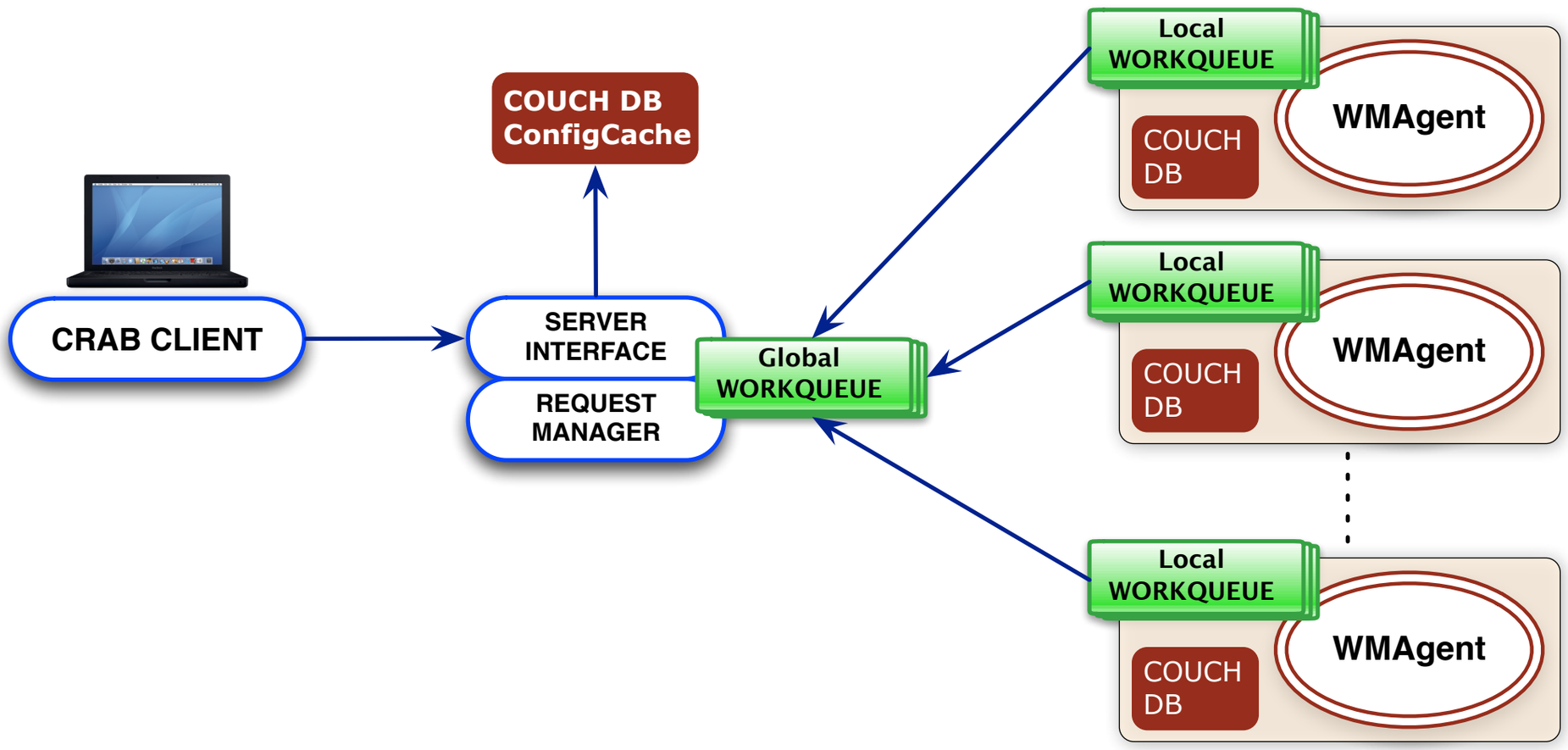
- WMAgent will replace all the three CMS workload management tools (Tier-0, ProdAgent, CRAB) with a completely new tool that will improve known bottlenecks and issues.
- It aims to provide a common infrastructure where the different workflows (analysis, (re)processing, ...) will run by using the very same code (a part of configurations and plug-ins), saving development man power, improving support with a common operational effort.
- CMS has just started real reprocessing workflow with WMAgent, while real user analysis workflows will start to run by the end of the year.

"Job life cycle management libraries for CMS workflow management projects"
Journal of Physics: Conference Series Volume 219 Part 4
Frank van Lingen et al 2010 J. Phys.: Conf. Ser. 219 042024

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

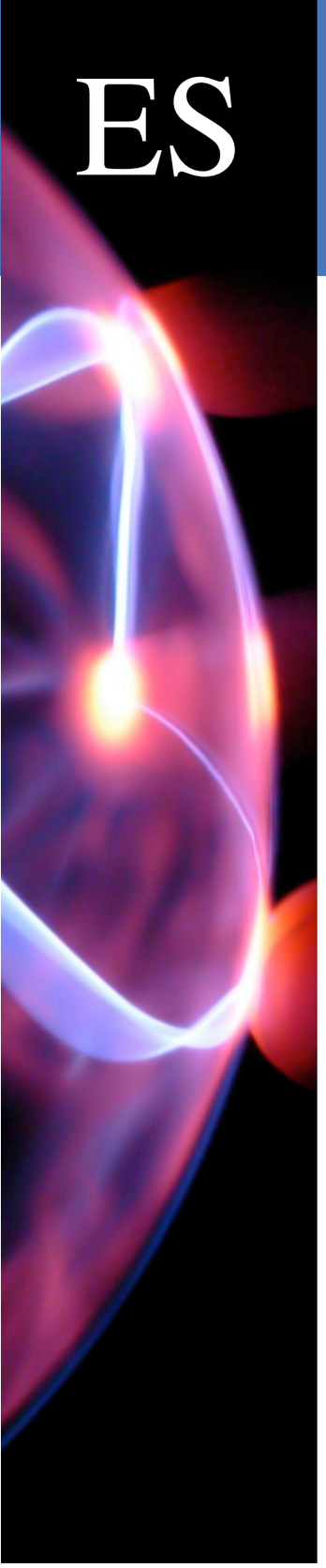
Architecture





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

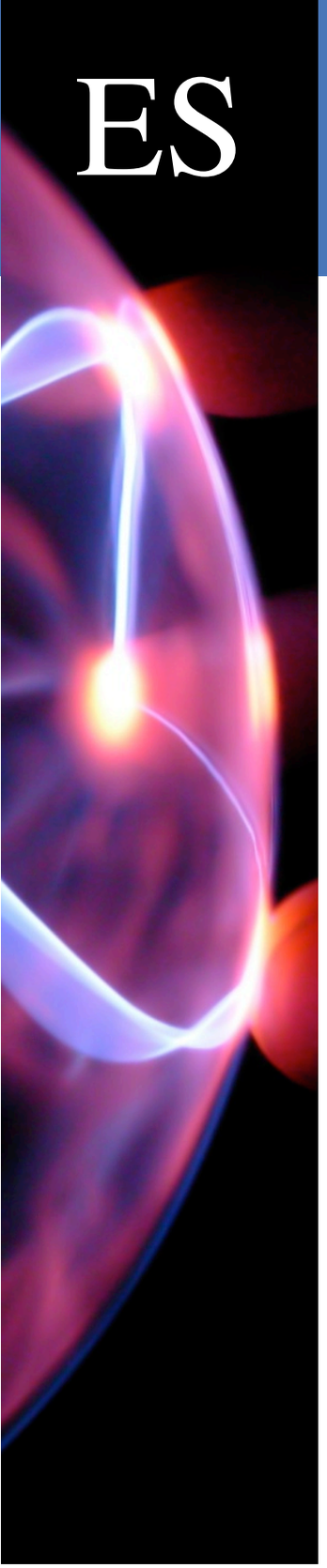
Motivations of CouchDB



- Some reasons to adopt CouchDB:
  - data format changes very often and in many cases data does not have meta-data (thinking about the job reports and different formats of errors)
  - remote access of information and data replication between different endpoints, allowing an easy way to show monitoring information without impact on the system
- WMAgent intensively uses a local relational database (MySQL/Oracle) to handle transaction and status changes between different components inside the agent itself.
- CouchDB has been coupled with each WMAgent instance to:
  - reduce and decouple from the core database the load generated by monitoring needs
  - providing an easy point where to show/propagate/replicate information (job history, workflow reports, ...) and to provide API to build other applications
- Example: the agent itself pushes both job status information and job framework job reports into CouchDB, which becomes a storage where task/job information can be easily browsed through web pages

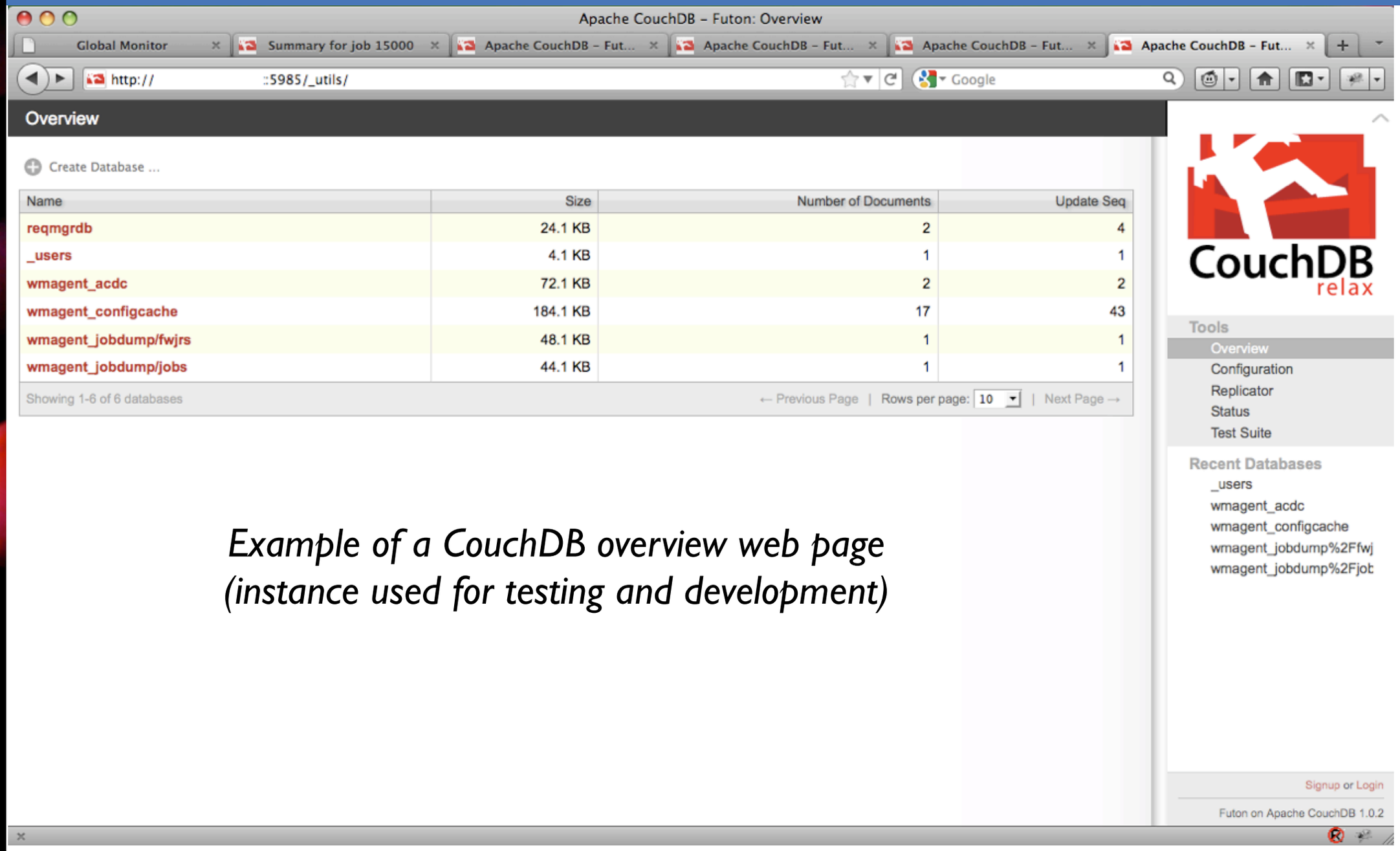
CERN IT Department
CH-1211 Geneva 23
Switzerland
www.cern.ch/it

WMAgent & CouchDB



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

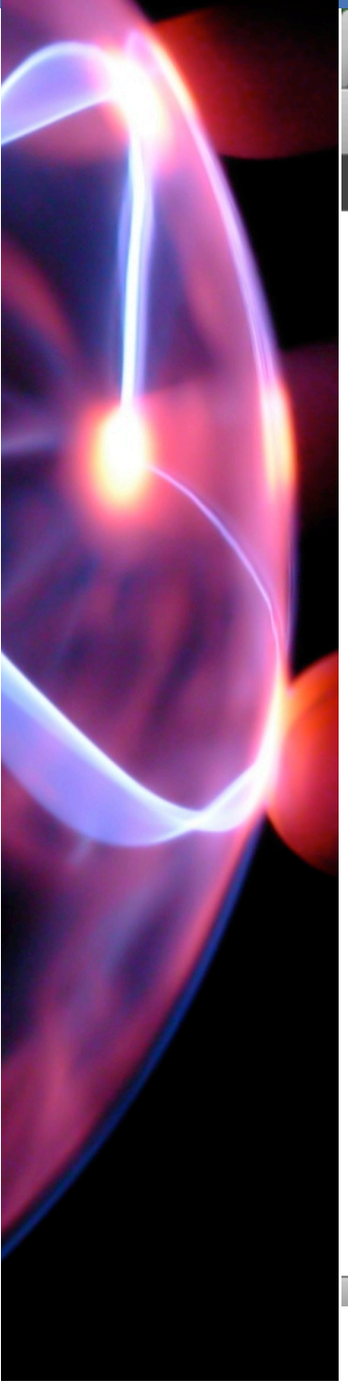
WMAgent & CouchDB

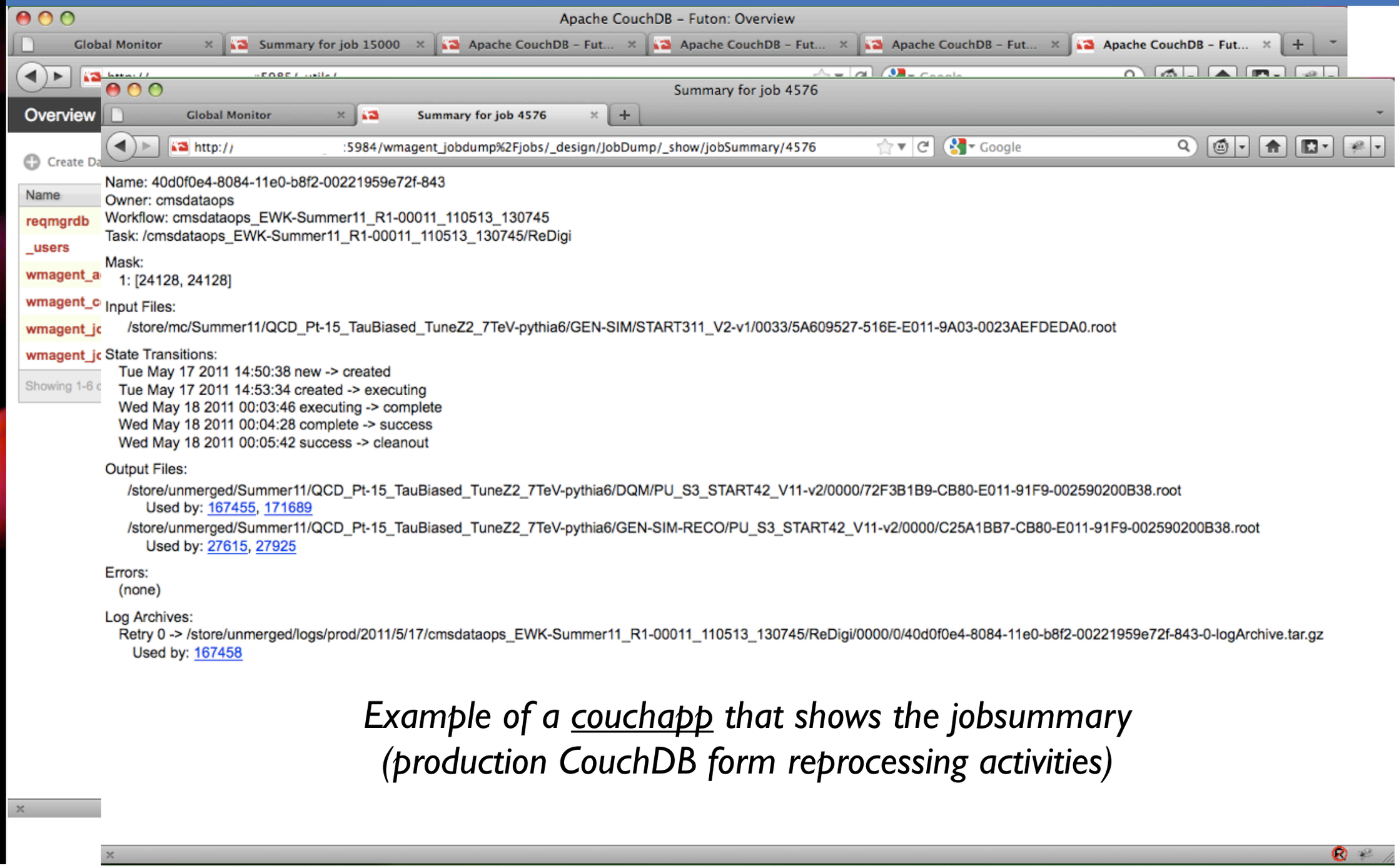


Example of a CouchDB overview web page
(instance used for testing and development)

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

WMAgent & CouchDB

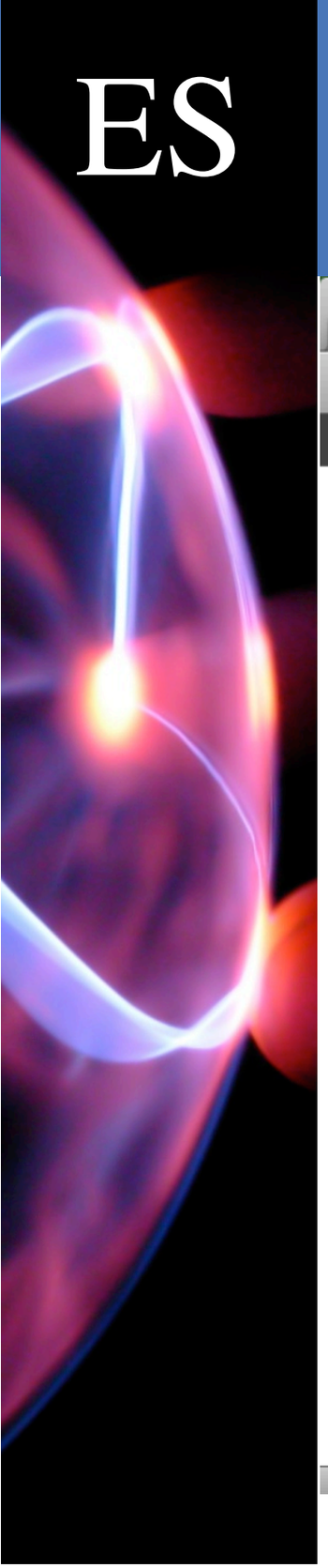


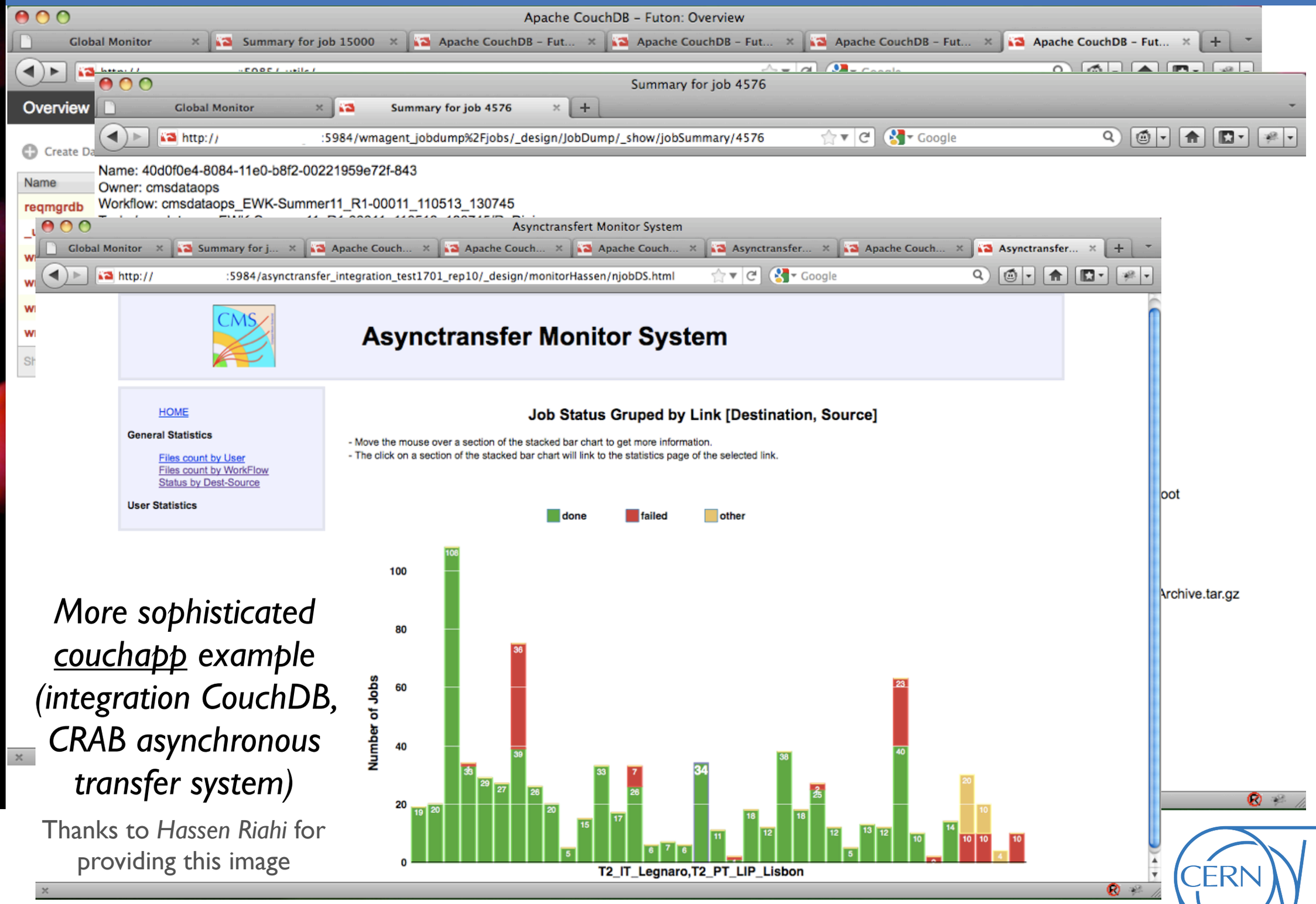


Example of a couchapp that shows the jobsummary (production CouchDB form reprocessing activities)

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

WMAgent & CouchDB

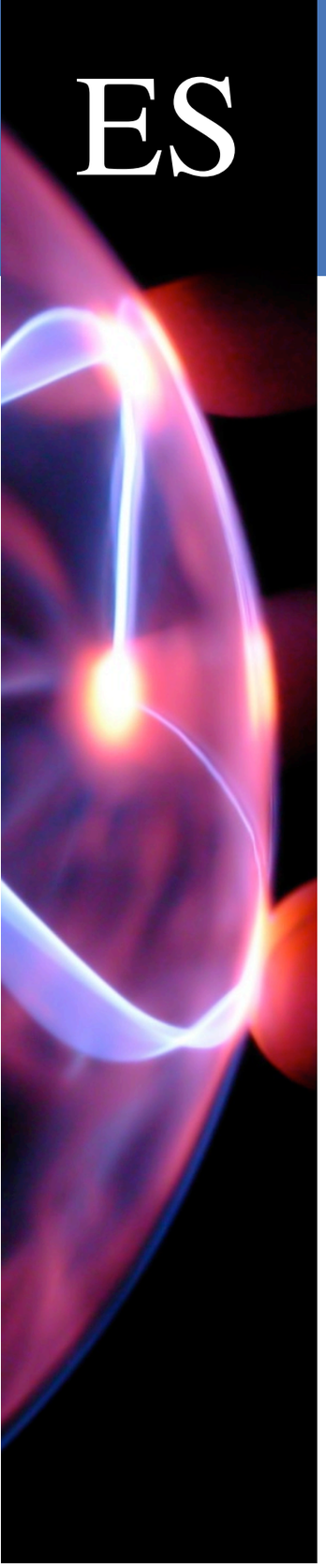




More sophisticated couchapp example (integration CouchDB, CRAB asynchronous transfer system)

Thanks to *Hassen Riahi* for providing this image

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



Conclusions

- NoSQL has certainly many advantages coming from a newer technology with many features (map/reduce queries, REST interfaces, sharding, ...) and CMS has already tested and developed tools that take advantages of these...
  - ...but issues disclosure on production services has just started in CMS and first experience with real running services is being currently done.
- Very well known issue: an SQL expert developer takes some time to start efficiently with not relational database.
  (Often this time is recovered due to the shorter development time taken by NoSQL database)
- When developing new things or need to rewrite applications, it is worth that dashboard team evaluates the usage of not relational database solutions (even on top of already existing systems)
- Next week there will be a 2 days workshop @CERN about "Database Futures Workshop", with some contribution related to NoSQL usage and evaluation:
  https://indico.cern.ch/contributionListDisplay.py?confId=130874

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