Automatization of User analysis Workflow in CMS

Daniele Spiga
CERN
on behalf of CMS offline and computing

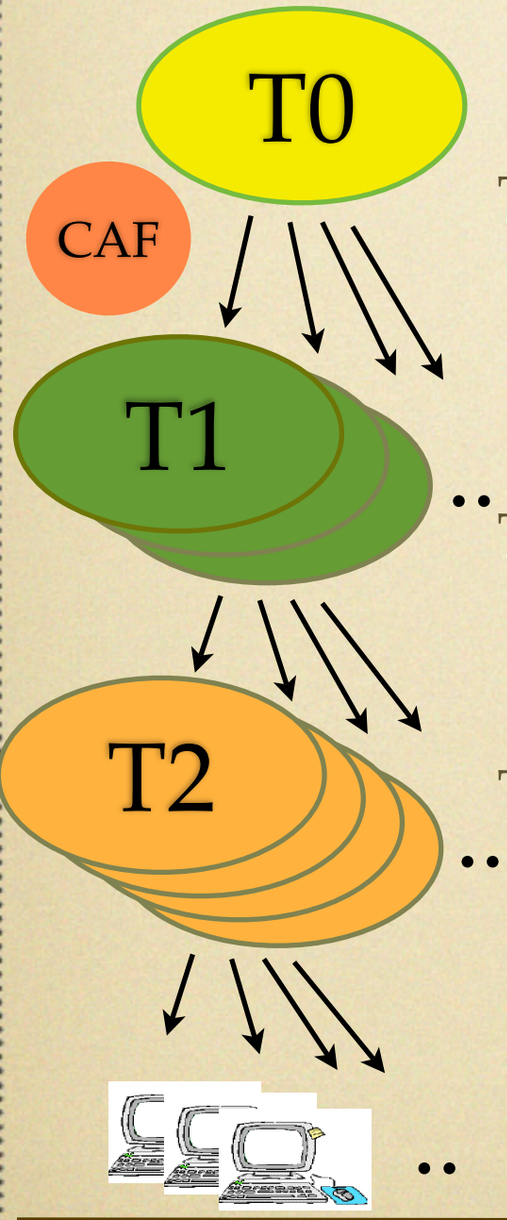
International conference on Computing in High Energy and Nuclear physics
CHEP 2009
Prague 21-27 March 2009

Outline

- Introduction :
  - CMS Computing and Analysis model
  - Distributed analysis features
- Automatization: CMS Remote Analysis Builder
- Results and deployment activity
- Next steps
- Summary

Introduction

The CMS offline computing system is arranged in Tiers geographically distributed.



T0 : only used for organized processing

CAF: (CERN Analysis Facility) supports for "latency critical" activities

T1: perform reprocessing and skimming, mainly organized task

T2: support for more chaotic analysis activities

This model foresees that users performing data analysis must interact with the distributed environment.

Analysis Flow

The CMS model is data location driven: the users analysis runs where data are located.

- User runs interactively on small data sample developing the analysis code.
- User selects large data sample to run the very same code.
- User's analysis code is shipped to the site where sample is located.
- Results are made available to the user for the final plot production.

From the user point of view

The CMS computing model interaction can represent a complex task.

- Data Bookkeeping and Location System (DBS)
  - reading/writing
- Grid Middleware (WLCG/OSG)
  - job submission, tracking, storage system interaction
- CMS Sites information (SiteDB)
  - specific configuration, storage endpoint

CMS Remote Analysis Builder

CMS developed a tool (CRAB) for the transparent usage of the distributed system.

- It provides the user with a simple interface and a lightweight client
- It provides a service platform to automate the user analysis workflow

To increase the automation CRAB is based on an intermediate server.

User interface

Command line python application using an SQLite database for logging purpose.

- Data discovery
- Packing of user local library and configurations
- Server communication

configuration file:

```
[CRAB]
jobtype                 = cmssw
scheduler               = glite
server_name             = Bari

[CMSSW]
datasetpath             = /HiggsGammaGammaM120/CMSSW_2_0_0_1202115095/RECO
pset                    = myHiggsAnalyzer.py
total_number_of_events  = -1
events_per_job          = 1000

[USER]
copy_data               = 1
storage_name            = T2_IT_Legnaro
remote_dir              = myHiggsAnalysis
```

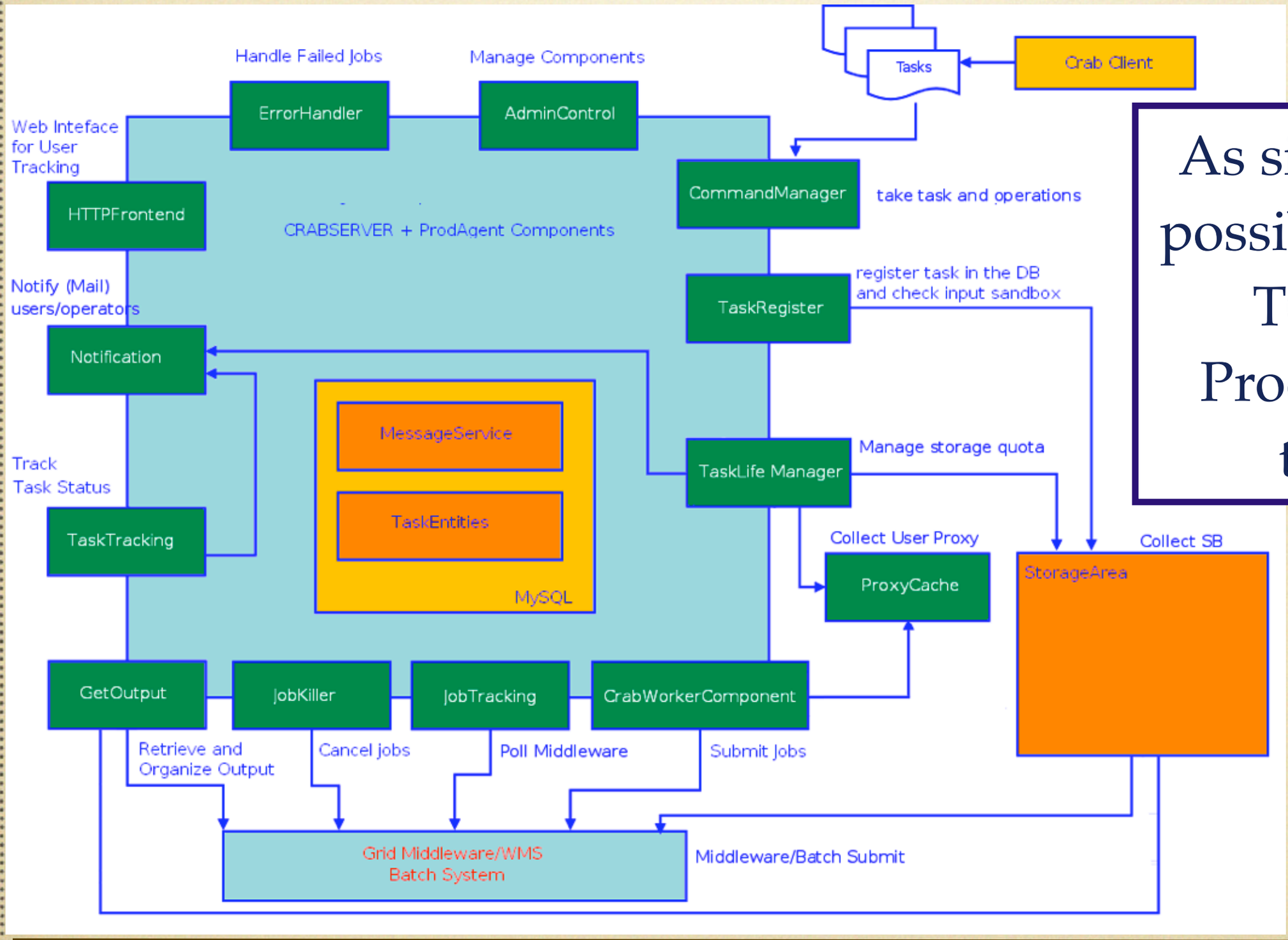
```
[lxplus235] ~/scratch0/WorkOK > crab -submit
crab. crab (version 2.5.0) running on Fri Mar 13 09:48:49 2009

crab. Working options:
  scheduler           glite
  job type            CMSSW
  working directory   /afs/cern.ch/user/s/spiga/scratch0/WorkOK/crab_0_090313_094843/

crab. Registering credential to the server
crab. Registering a valid proxy to the server:
crab. Credential successfully delegated to the server.

crab. Starting sending the project to the storage dot1-prod-2.ba.infn.it...
crab. Task crab_0_090313_094843 successfully submitted to server dot1-prod-2.ba.infn.it


crab. Total of 2 jobs submitted
```

The analysis server

Intermediate service responsible for the analysis flow automation: submission/resubmission/error handling/output retrieval....

- Responsible for the middleware interaction
  - fully interoperable supporting: WLCG/OSG/local batch
- Adopts a modular software approach with a MySQL DB as core:
  - independent and multithreaded components (agents)
  - communication through asynchronous and persistent message service
  - external components used for data storage

Architecture

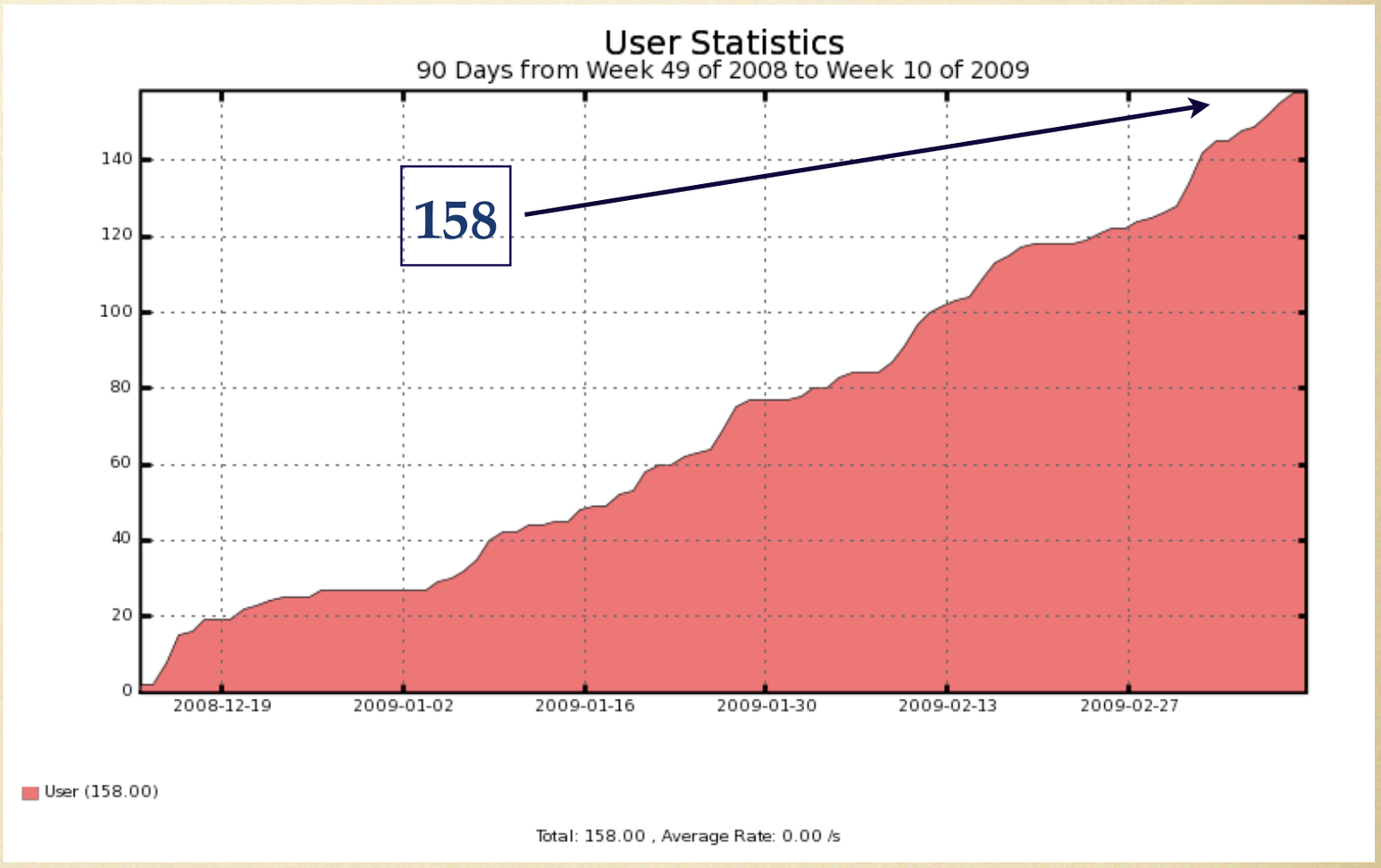


As similar as possible to the T0 and Production tools

Deployment Activity

The server deployment started end of 2008.

- 4 server instances actually in production
  - CERN, Italy, France
- 1 Test instance
  - USA



Cumulative number of distinct users from the past 3 months on a single server instance

Monitoring system

Web monitoring is crucial feature both for users and administrator.

CrabServer Instance: CrabServer@dot1-prod-2.ba.infn.it

Service	Description
Tasks	Tasks entities data in this CrabServer
Jobs	Jobs entities data in this CrabServer
Component Monitor	Component and Sevice status in this CrabServer
User Monitoring	User task and job log information

CrabServer Components and Services Monitoring

Diplay the status of components and active service in this CrabServer:

Status of all components and external services of this CrabServer [Show report]

Allow to access components logs through web:

Show logs for JobTracking [Show logs]

JobTracking
GetOutput
JobKiller
HTTPFrontend
CrabServerWorker
ErrorHandler
AdminControl
CommandManager
Notification
TaskTracking
TaskRegister
TaskLifeManager

Watch message … *component:*

Show message … [Show]

or whatch mes … *all the component:* [Show]

(Work in progress …

Display components CPU usage:

Show CPU plot for All components for last 1 hours [Show Plot]

Display services CPU usage:

Show CPU plot for All services for last 0 hours [Show Plot]

Display resources usage:

Show plot for All resources for last 1 hours [Show Plot]

Admins

CrabServer Tasks: internal server logging

Filling the field with the string resulting from 'crab -printId' allow to check both staus and logging:

Task unique name spiga_crab_0_090308_212332_698esb [Status] [Show]

Select the user name to see all his tasks on the server

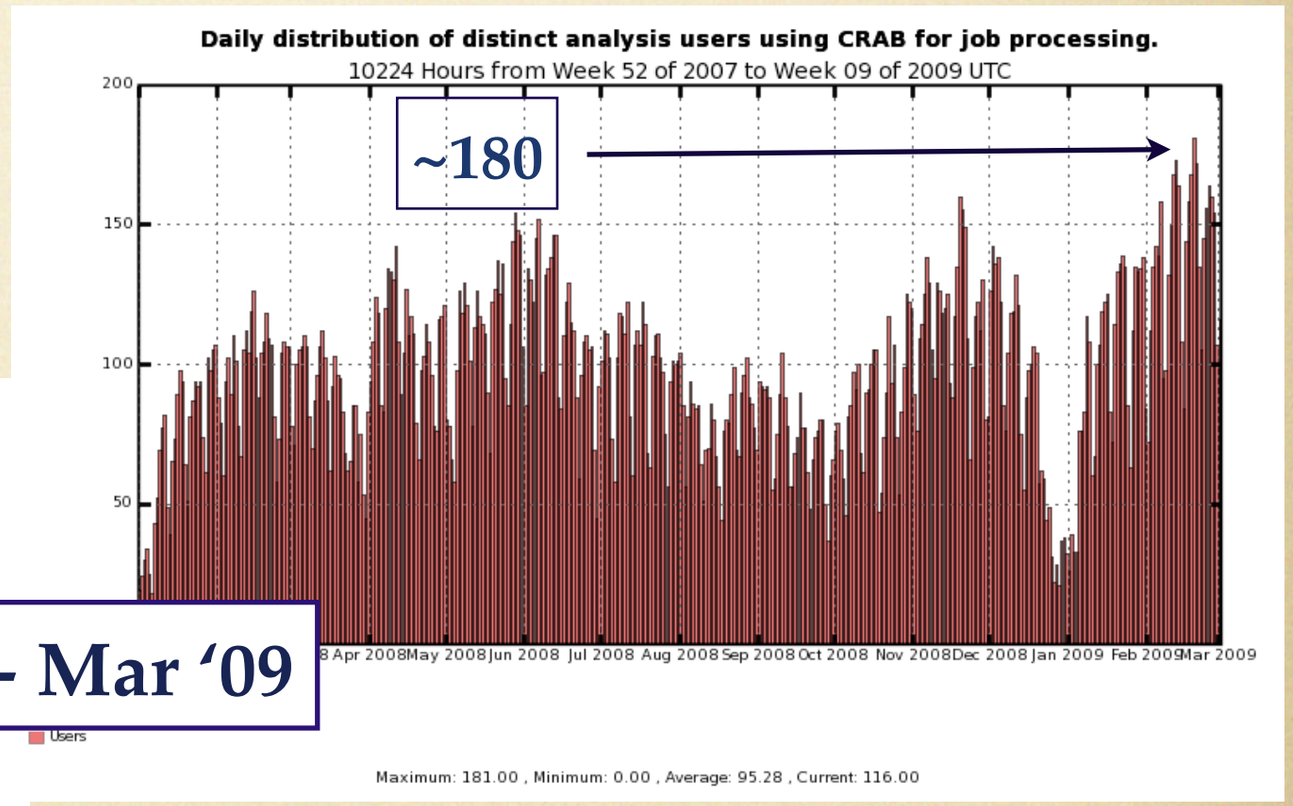
User All status of All tasks for last 100 hours [Show Tasks]

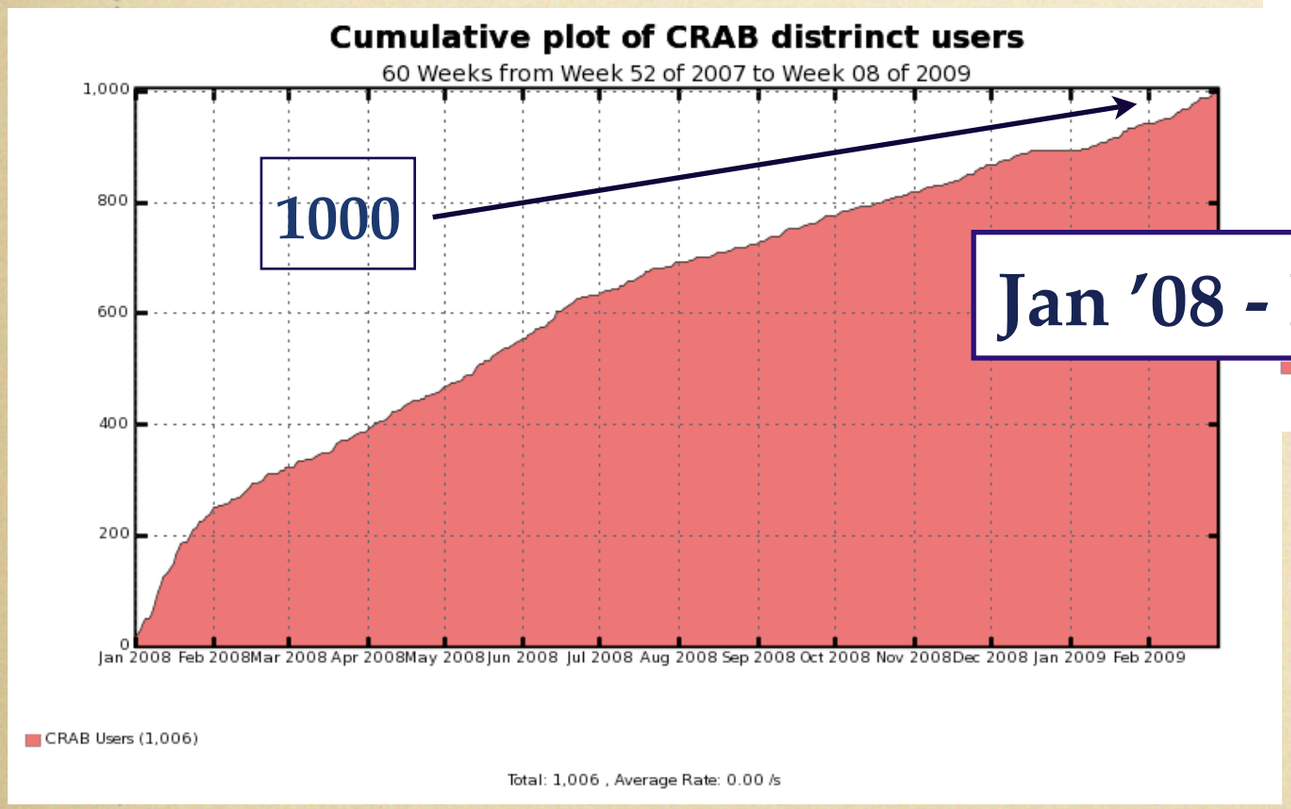
Farida Fassi
Federica Fanzago
Ferhat Ozok
Finn O'Neill Rebassoo 238694
Francesco Fiori
Francesco Pandolfi
Frederic Ronga
Gautier Hamel De Monchenault
Gavril Adrian Giurgiu 134236
Geng-Yuan Jeng
Georgia Karapostoli
Georgios Anagnostou
Gianluca Cerminara
Giovanni Petrucciani
Giuseppe Cerati
Giuseppe Codispoti
Gordon Kaussen
Guofan Hu 86181
Hammad Gregory
Harold Nguyen 924852

Users

Results

More than 40% of the CMS Collaboration make use of the distributed infrastructure.



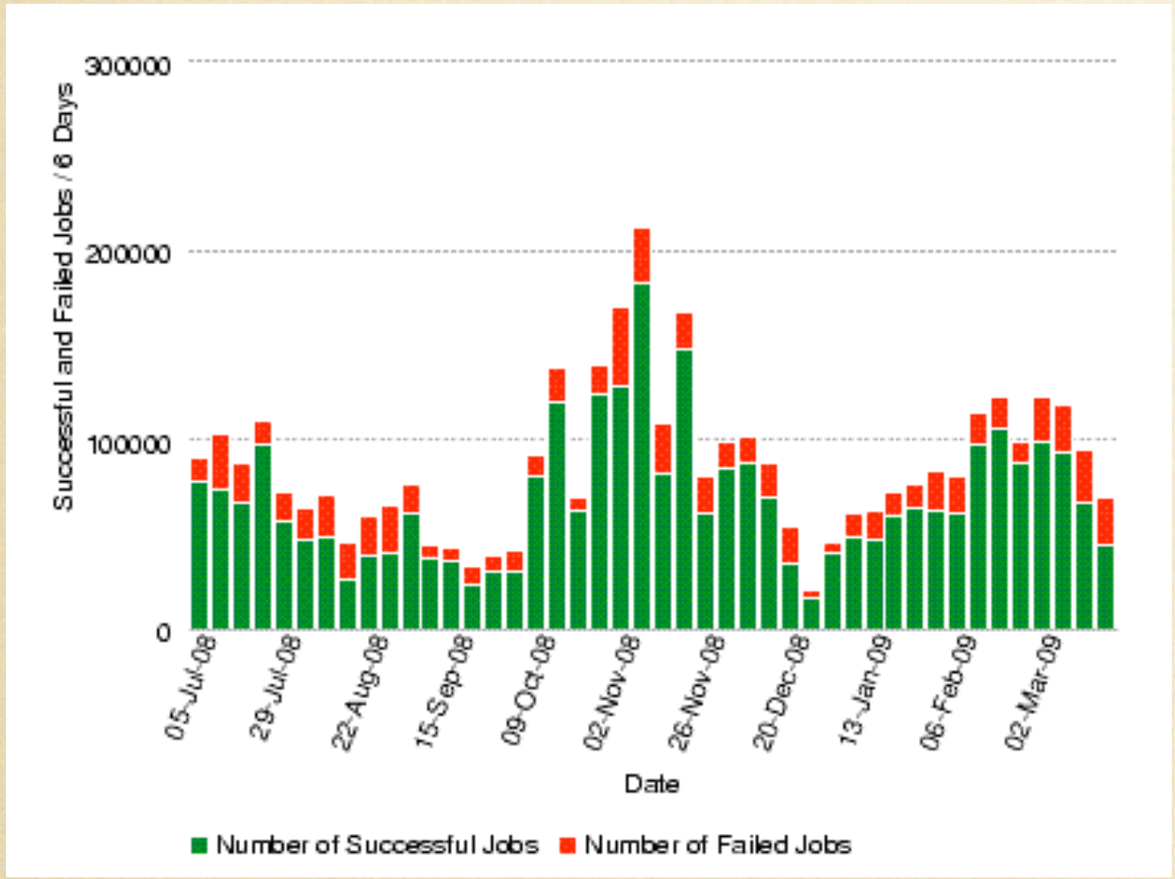


Jan '08 - Mar '09

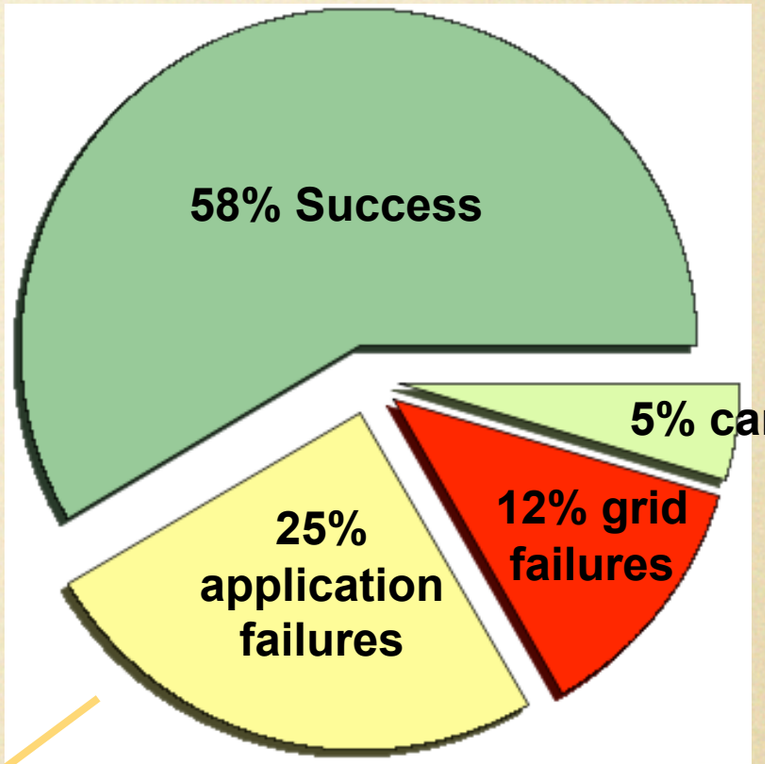
Data from CMS Dashboard

Analysis job Efficiency

From July 08 to March 09



see J.Letts Talk ID 207



Data from CMS Dashboard

User configurations errors
Remote stage out issues
Few % of failures reading data at site

Lessons

The huge CRAB user community generates a lot of feedback. In view of the LHC start-up:

- Analysis job efficiency needs to be improved
- New submission logic based on trigger system (submit jobs once data became available at T2 sites)
- New features for user interaction with distributed storage system needed

The designed architecture offers a lot of handles allowing us to reach these goal.

Next steps

- Improve the interface presented to the user
  - new functionality focused on the physics domain
- Add intelligence to the server for more advanced use cases
- Converge on cross project common library (WMCore) to improve the CMS WorkLoad Management maintenance....

see F.W.Lingen Talk ID 288

Summary

- The most crucial design metrics have been demonstrated also during the 2008 challenges : User analysis automation can be supported

see A.Fanfani poster ID 213

- Deployment activity started with success

- Need to spend effort on:
  - User interface optimization
  - Job efficiency improvements
  - CMS Workload Management common core migration