Frascati Physics Series Vol. XL (2006), pp. 357–362
Frontier Science 2005, New Frontiers in Subnuclear Physics
Milano, 12-16 September, 2005

CRAB, A TOOL TO ENABLE CMS DISTRIBUTED ANALYSIS

Federica Fanzago, Stefano Lacaprara, Nikolai Smirnov
INFN Padova, Italy
Spiga Daniele
Università of Perugia and INFN, Italy
Giovanni Ciraolo
Università of Firenze and INFN, Italy
Marco Corvo
CERN, Meyrin, Switzerland
Nicola De Filippis
Università of Bari and INFN, Italy
Alessandra Fanfani
Università of Bologna and INFN, Italy

Abstract

The CMS experiment will produce a large amount of data (few PBytes each year) that will be distributed over many computing centres spread in the countries participating to the CMS collaboration. To access and analyse data available at remote sites, physicists will use the grid infrastructure.
CMS is developing a user friendly tool, CRAB (Cms Remote Analysis Builder), whose aim is to simplify the creation and the submission of analysis jobs into the grid environment. Its purpose is to hide the grid infrastructure complexity to final users, so that they can access remote data as easily as in a local environment. Data discovery, resources availability, status monitoring and output retrieval of submitted jobs are fully handled by CRAB.
In this report we will explain how CRAB is interfaced with other CMS/grid services, the experience gained during production and future developments.

1 Introduction

CMS (Compact Muon Solenoid) [1] is one of the four particle physics experiments that will collect data at LHC (Large Hadron Collider) starting in 2007 at CERN, Switzerland.
This experiment will produce a large amount of data (few PBytes each year) that will be stored in many computing centres (Tiers) in the countries partecipating to the CMS collaboration and made available for analysis to world-wide distributed physicists [2].
CMS will use a distributed architecture based on grid infrastructure to ensure remote resources availability and to assure remote data access to authorized user (belonging to CMS Virtual Organization).
Tools for accessing distributed data and resources, provided by the European LHC Computing Grid (LCG) [3] and the American Open Science Grid (OSG) [4], are under evaluation.

2 Distributed analysis chain and CRAB role

Data analysis in a distributed environment is a complex computing task. It assumes to know which data are available, where they are stored and how to access them. It is also needed to understand which resources are able to comply with user analysis job requirements. The CMS collaboration is developing some tools, interfaced with available grid services, to achive these purposes.
They include:

- installation of CMS software via grid on remote resources [5]
- data transfer service to move and manage a large flow of data among Tiers [6]
- data validation system to ensure data consistency and readiness [7]
- data location system to keep track of data available in each site and to allow data discovery. Now it is composed by a central database located at CERN (RefDB) containing information about what kind of data have been produced; a local database (PubDB) in each Tier with information about where data are stored (which local Storage Element - SE) and the

protocol to access them; local POOL catalogs with physical data location on SE [2)]

- A friendly interface to simplify the creation and the submission of analysis jobs to grid environments: **CRAB** (**C**MS **R**emote **A**nalysis **B**uilder). Its purpose is to allow users with no knowledge of grid infrastructure to run their analysis code on data available at remote sites as easily as in a local environment hiding grid infrastructure details [8)]

- job monitoring and logging-bookkeeping system to keep trace of submitted jobs and to identify grid middleware or user application problems

3 CRAB

The aim of CRAB is to simplify the work of final users to create and submit analysis job to the grid. CRAB is written in python and have to be installed to the User Interface (UI), the user access point to the grid. Users develop their analysis code in an interactive environment and decide which data to analyze. They have to provide to CRAB:

- Data parameters: keywords to select a given dataset, as found on RefDB web page, total number of events to be accessed and number of events for each job

- Analysis executable (which is in general user specific) and corresponding parameter cards

- Output file name and how to manage them (return file on UI or store into SE)

Data discovery, resources availability, job creation and submission, status monitoring and output retrieval are fully handled by CRAB.
Specifically, CRAB queries RefDB and local PubDB to discover the location of data to be analyzed and uses these information as job requirements. The grid workload management system match correct resources according to these requirements.
CRAB creates a wrapper of the analysis executable which will be run on remote resources, including CMS environment setup and output management,

like final copy to a grid SE, if required.
CRAB splits the analysis into a number of jobs according to user provided information about events. The user executable is sent to remote resource via input sandbox, together with the job, as tgz archive.
Jobs are submitted using grid workload management command and monitored querying the grid logging and bookkeeping system.

4 CRAB current usage

In the last month (15 August-12 September 2005) have been submitted about 50000 jobs whose weekly rate increased from 6000 to 15000 jobs.
About 210 different datasets are used at least once and the most accessed was required by 15000 jobs. Jobs have been submitted from 25 UI using 40 remote sites storing data.
The job success rate is about 75%, where success means that jobs arrive to remote sites and produce outputs, while the remnant 25% aborts due to site setup problem or grid services failure.

5 CRAB future development

5.1 CRAB job splitting

Up to now the criteria for job splitting is based on events. A user decides the total number of events to analyze and how many events for each jobs and CRAB creates a number of jobs according to this information. By policy, only complete dataset are stored in a site.
In the future the core splitting mechanism will be based on "event-collection" (today this correspond to a "run"), the smallest unit of events that a user is able to select. A job will be configured with the proper set of event-collections contained in given file-blocks, where file-blocks are the packaging units used in data transfer that can contain more than one event-collection.

5.2 CRAB data location and discovery system

Today the data location is based on RefDB, local PubDB and local POOL catalog, as explained in section 2. In the future will be used a "three catalogues system" [2]:

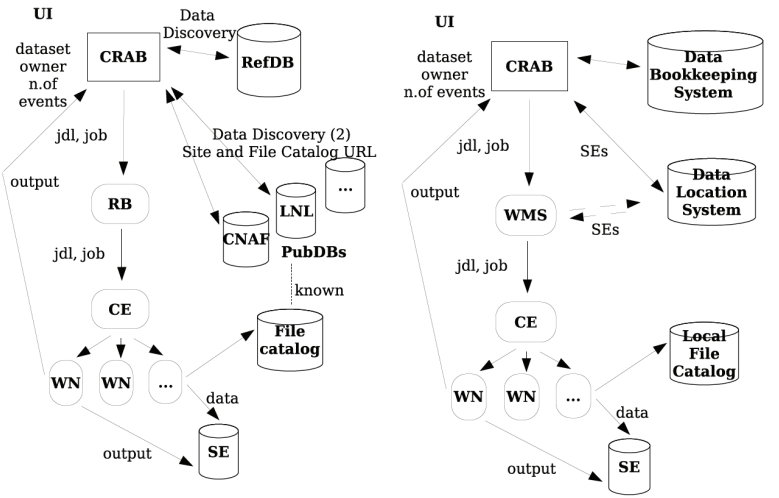

Figure 1: *The current and future data discovery system*

- Dataset Bookkeeping System (DBS) : which data exists. It contains CMS-specific description of event data.
- Data Location Service (DLS): where data are located. It allows to find replicas of a given set of data (file-block) in the distributed grid computing system. It maps file-blocks to Storage Elements where the data are located.
- Local File Catalogue: physical location of data on the SE

CRAB will be modified to use the new catalog and splitting model.

6 Conclusion

The CMS collaboration is developing CRAB, a tool to simplify the creation and the submission of analysis jobs in a distributed environment, based on grid infrastructure. A big effort has been done to understand user needs in order to realize a friendly tool and to use in the best way services provided by grid. The tool has been successfully used by many CMS collaborators to analyze

remote data otherwise not accessible. As CMS analysis framework and grid middleware evolve, CRAB has adapt to cope with these changes and always guarantee its usability and thus remote data access to users.

References

1. The CMS experiment: http://cmsdoc.cern.ch
2. C-TDR - Computing Technical Design Report, CERN-LHCC-2005-023, June 2005
3. LCG Project: http://lcg.web.cern.chLCG and LCG Technical Design Report, CERN-LHCC-2005-024, June 2005
4. OSG project: http://www.opensciencegrid.org
5. XCMSI project: http://cern.ch/cms-xcmsi
6. Tim Barras *et al*, Software agents in data and workload management, presented at CHEP2004, http://www.chep2004.org
7. N. De Filippis, Data Validation tools for CMS analysis, CMS Note 2005/XXX in preparation
8. CRAB project: http://cmsdoc.cern.ch/cms/ccs/wm/www/Crab