CMS distributed analysis infrastructure and operations: experience with the first LHC data

E W Vaandering
Fermi National Accelerator Laboratory, Batavia, IL 60510, U.S.A.
E-mail: ewv@fnal.gov

Abstract. The CMS distributed analysis infrastructure represents a heterogeneous pool of resources distributed across several continents. The resources are harnessed using glite and glidein-based work load management systems (WMS). We provide the operational experience of the analysis workflows using CRAB-based servers interfaced with the underlying WMS. The automatized interaction of the server with the WMS provides a successful analysis workflow. We present the operational experience as well as methods used in CMS to analyze the LHC data. The interaction with CMS Run-registry for Run and luminosity block selections via CRAB is discussed. The variations of different workflows during the LHC data-taking period and the lessons drawn from this experience are also outlined.

1. Introduction
In this paper, we describe the analysis experiences of CMS with the first LHC data, taken in 2010. CMS has adopted the MONARC computing model [1]. The CERN Tier0 and the seven regional Tier1 sites are responsible for official data reprocessing and custodial storage of the data. This means that all user analysis must take place at Tier2 and Tier3 centers, which is necessarily more chaotic. Since the beginning of data taking, this system has performed at the level expected and its operation has exposed issues that needed to be resolved.

2. Tier2 sites and analysis infrastructure
The CMS Tier2 sites are the primary location for analysis work. Fifty percent of the CPU resources are dedicated to user analysis with the other half being dedicated to central Monte Carlo production. To support user analysis, high demand datasets are subscribed to Tier2 sites on disk resources dedicated for this purpose. Since the datasets can be repopulated on the Tier2 sites quickly (often within hours), the Tier2 storage is often viewed as “cache,” repopulating from custodial copies at the Tier1 when needed.

Each physics group in CMS has several assigned Tier2 sites which are tasked with storing data needed by those groups. Additionally, each user in CMS has an assigned Tier2 where they have a terabyte of personal storage space. These assignments are based on geography, so a user may not have personal space at the Tier2 where the official data he or she is using resides.

Much of the infrastructure for users’ analysis is shared with the CMS production tools; two pieces of the architecture are unique to user analysis workflows: CRAB, the CMS Remote Analysis Builder, and StoreResults, which is used to promote useful user data to official.
data suitable for custodial storage. StoreResults is the subject of another paper in these proceedings [2]. CRAB will be described in the next section.

3. CRAB usage in CMS

CMS uses CRAB to hide the intricacies of submitting analysis jobs to the Grid from the user. CRAB operates in two modes, client-server and stand-alone, with the same user interface. Client-server mode is now the dominant mode of submission.

In client-server mode, the client has the responsibility of generating the user’s proxy, packaging up the pre-compiled user code, and submitting the job package to the CRAB server. As the jobs submitted by the server run and complete, the client communicates with the server to check on the status of the jobs and to retrieve the output. The server has the responsibility of submitting the jobs to the chosen Grid middleware, resubmitting jobs as needed, and caching the users’ output on the CRAB server. The current CRABServer code is a hybrid of the current CMS production workflow management software, ProdAgent, and custom written code. The split between these code sources is approximately 50/50.

In stand-alone mode, there is no server and the CRAB client is responsible for communicating with the Grid or local scheduler middleware. Any resubmission of jobs is at the discretion and control of the user. This code only shares some lower level libraries with the production system.

The server code supports submission to both the gLite and Glidein Workload Management Systems (WMS). The stand-alone mode of submission supports direct submission to local schedulers such as condor and LSF as well as direct submission to Grid resources (CondorG, gLite, and ARC).

CRAB supports automated submission of some alignment and calibration high-priority workflows on the CERN CAF. This functionality is described in another submission to these proceedings [3].

4. CRAB usage statistics

CRAB usage has remained fairly constant since the beginning of data taking, as shown in Figure 1. There are approximately 400 unique users of CRAB per week with peaks prior to conferences (ICHEP) and during our analysis stress test (OctX) in October 2009. There are approximately 800 unique users of CRAB per month.

The number of jobs run with CRAB is close to what CMS has been planning for. Figure 2 shows the number of analysis jobs per week which averages 595 000 per week or about 85 000 per day. A dip in late July is visible after the completion of analyses for the ICHEP conference. Figure 3 shows our Tier2 utilization. The average number of simultaneously running jobs on all Tier2 sites is about 7000.

These 85 000 jobs/day are serviced by multiple CRAB servers. Each server can manage about 50 000 jobs at a time (running and waiting). With spare capacity for peak loads and maintenance downtime, our current installation of four servers is barely adequate, so the installation of a fifth server is planned for late 2010. Of these four (or five) server installations, all of them can submit to the gLite WMS while only two can currently use the Glidein WMS.

5. Running on certified data

With the advent of data taking, efficiently running over just the good data has become of utmost importance. Once a week (or more often in the run up to a conference), CMS certifies a list of runs and “luminosity sections” (LS). Each LS represents 23 seconds of data and is the atomic unit of data in CMS. Currently the certified LS are those in which all available detector elements are working; this information is extracted from the CMS run registry and the list of good LS is given in a JSON formatted file. This file is passed to CRAB which associates these LS with files using the Data Bookkeeping System. Jobs are then submitted to run only over files which
Figure 1. Number of CRAB users/week. The average number of users is about 400/week with peaks seen just before ICHEP and during our OctX stress test and a fall-off near the end of the year.

Figure 2. Number of CRAB jobs/week. This is the sum over all the Tier2 sites (different colors) and the FNAL Tier3, which runs the largest number of jobs. The average is about 85 000 jobs per day.

contain good data. A further restriction is made in the CMS runtime application to restrict jobs to running on only good data in the case where a file contains both good and bad data. While CMS supplies one file describing all the good data, users are able to generate their own lists (e.g. if a detector they don’t use is off).

6. Operations and error rates
Prior to our October scale exercise, CMS formed a new group “Analysis Operations” tasked with supporting the Grid analysis tools users need. Their responsibilities include placing data on the Tier2 sites, operating the CRAB servers and the StoreResults service, and user support for CRAB. User support is by far the biggest share of the effort. Many errors that are reported
are actually errors in the users’ applications. However, there are also a number of Grid error scenarios that impact users as well as failures in the CRABServer software.

Figure 4 shows the same 595,000 jobs/week categorized by their success or failure status. The majority of jobs are successful. Of the unsuccessful jobs that this plot shows, most are user errors, although certain types of site problems are also reported in the “Application-Failed” category. The number of failures of the Grid infrastructure is comparatively low at about 10%.

One large component of the “GRID-failed” jobs has traditionally been failures of transfers from one Tier2 to another Tier2. As explained above, the data a user is analyzing may be on one Tier2 and his or her personal space for output may be on another Tier2. In that case, output must be transferred from one Tier2 to another, possibly on a different continent. A recent campaign to certify all the links between the 40 high-availability CMS Tier2s has paid substantial dividends, greatly reducing the fraction of problems attributed to Tier2-Tier2 transfers. This campaign is the subject of another contribution [4].

7. Future plans
CMS is currently in the process of replacing its production job submission system. This next generation tool is named WMAgent (compared to ProdAgent in the past). This set of tools has more robust failure mitigation. CRAB will be part of this migration, so the existing CRAB tools are in feature freeze with only bug fixes being applied. WMAgent is more flexible allowing new workflows to be implemented more easily and will have natural support for workflow subscriptions, allowing new jobs to be submitted on the user’s behalf as soon as new data is available.

We plan to have a prototype of CRAB implemented in this new framework by early 2011. Much of the remainder of 2011 will likely be a transition period. This new user analysis framework will be client-server only with a very thin client and a server that is almost entirely shared code with the other production tools. (This should be compared with the current situation in which about half of the code is unique to the analysis case.) Client and server will communicate with a RESTful communication protocol.

The time and effort spent developing and supporting the existing CRAB and CRABServer software will benefit us in driving this transition in the right direction. Additionally, we expect
Figure 4. Analysis jobs by status. The largest portion of jobs are successful. The next largest group, labeled “Application-Failed”, are typically a result of user errors. The infrastructure failure rate (“Grid-failed”) is about 10%.

that we will be able to implement the backlog of development requests that users have made in the last year. These requests were not deemed important enough to expend development effort in the existing CRAB infrastructure.

Finally, StoreResults is also being reimplemented in the WMAgent framework. As the workflow is much simpler, this is a comparatively easy task.

8. Conclusions
In conclusion, CRAB is widely used within CMS and our level of Tier2 usage is approximately what we projected in our data model. The failure rate is acceptable (about 10%) and we have made considerable effort to drive this lower. Our operational experience has been instrumental in this effort. Our planned 2010–11 reimplementation of CRAB, StoreResults, and WMAgent in a common framework should further improve this situation allowing us to simultaneously deliver new features that users would like and more robust error handling to shield them from problems with the Grid.

References


[3] Riahi H Large scale and low latency analysis facilities for the CMS experiment: Development and operational aspects in these proceedings, PS31-5-200

[4] Letts J Large scale commissioning and operational experience with Tier-2 to Tier-2 data transfer links in CMS in these proceedings, PS48-2-187