CMS experience with online and offline Databases

Dr. Andreas Pfeiffer, on behalf of the CMS collaboration
CERN, PH-CMG, CH-1211 Geneve 23, Switzerland
E-mail: andreas.pfeiffer@cern.ch

Abstract. The CMS experiment is made of many detectors which in total sum up to more than 75 million channels. The online database stores the configuration data used to configure the various parts of the detector and bring it in all possible running states. The database also stores the conditions data, detector monitoring parameters of all channels (temperatures, voltages), detector quality information, beam conditions, etc. A subset of the full information, the conditions data, is streamed to another database as it is used in the offline reconstruction, together with alignment and calibrations data for the various detectors. Conditions data sets are accessed by a tag and an interval of validity through the offline reconstruction program CMSSW, written in C++. About 200 types of calibration and alignment exist for the various CMS sub-detectors. Only those data which are crucial for reconstruction are inserted into the offline conditions DB. This guarantees a fast access to conditions during reconstruction and a small size of the conditions DB. The paper presents the experience with the CMS online and offline databases during the 2010 and 2011 data taking periods, showing some of the issues found and lessons learned.

1. Introduction
The primary goal of the Compact Muon Solenoid (CMS) experiment [1] is to explore physics at the TeV energy scale, exploiting the collisions delivered by the Large Hadron Collider (LHC) at the European Organization for Nuclear Research (CERN) in Geneva, Switzerland. The central feature of the CMS apparatus is a superconducting solenoid, of 6 m internal diameter, delivering a field of 3.8 Tesla. Within the field volume are the silicon pixel and strip tracker, the crystal electromagnetic calorimeter (ECAL) and the brass-scintillator hadronic calorimeter (HCAL). Muons are measured in drift tube chambers (DT), resistive plate chambers (RPC), and cathode strip chambers (CSC) embedded in the steel return yoke. All detectors combined comprise more than 75 million channels, most of them to be found in the tracker system. A detailed description of the experimental apparatus can be found elsewhere [1].

Calibration and alignment data are fundamental to maintain the design performance of the experiment. Dedicated, very fast, workflows have been put in place to compute and validate the alignment and calibration sets and insert them in the conditions database before the reconstruction process starts. Some of these sets are produced by analyzing and summarizing the parameters stored in the online database. Others are computed using event data through a special express workflow. A dedicated monitoring system has been put up to monitor these time-critical processes.

A large amount of information is collected by database systems to operate the experiment and check the quality of the data and the performance of the data. All this data is stored securely in relational databases (Oracle), allowing for easy access as requested by the various users. The
information stored encompasses configuration information for the various detectors like their temperatures, high- and low-voltage settings, their status as well as information inferred from calibration algorithms. This data changes with time and is therefore read out and stored in regular intervals into the database. The information is used by the detector experts to study and monitor the performance of the detectors. In addition, the databases collect information from the data acquisition system (DAQ), the level-1 and high-level triggers (L1 and HLT), run, beam and luminosity information. Various processes use the databases to communicate information and support workflows, examples of these are the Tier-0 workflow processing [2], the CMS data distribution service (PhEDEx) [3], the CMS data bookkeeping service (DBS) [4, 5] and similar systems.

Of particular interest for the offline reconstruction and the subsequent physics analysis of the data taken, are the so-called conditions data. These are made of a subset of the conditions data from the detector (e.g. status and calibration information for each channel), plus information gathered from special processing streams which provide alignment informations of the detectors and general information like the detailed position and extension of the interaction region (“beam-spot”) for a given period of time (run). All this information is stored by independent processes into the database and needs to be made available to anyone running a reconstruction job or doing physics analysis in the collaboration.

In the paper we will concentrate on our experience with the usage of the databases in CMS for the conditions. We will start explaining the major issues found and how they are solved, then detail aspects of performance and the evolution of the databases. Finally we will explain the need for rather detailed monitoring of the various services we provide in this context and finish with an outlook into possible future developments.

2. Conditions data in CMS

The CMS conditions data are structured into a set of Intervals Of Validity (IOV) using a time-stamp or run-number to assign/identify the time/run from which the IOV is valid. Each IOV contains a reference to the actual conditions data (usually referred to as “payload”). Each IOV can have a tag to identify and classify it and a consistent set of tags for IOVs is kept in a separate structure (a Global Tag) which is used in reconstruction and analysis. This structure allows the selection of the correct IOV from the list of all IOVs for a given time or run number without having to load all the payloads at the same time. This results in a large performance gain, as only the (with a few tens of MB comparatively small) list of IOVs needs to be retrieved fully, from the payloads, which are much bigger in size (several hundred MB for the data contained in one Global Tag), only the directly needed one(s) need to be loaded. This performant access to the conditions data as C++ objects is a key requirement for the reconstruction and data analysis. More details on the design, implementation and implications of the CMS conditions software is available in [6].

Conditions data enter the database via two different ways: in the online environment, detector experts run jobs using a special application (“PopCon”) to store their updated detector conditions into the online DB. From there they get transferred via Oracle streams to the offline DB; a workflow usually referred to as Online-to-Offline (“O2O”). For the conditions information coming from the special processing jobs running on the Tier-0 (e.g. initial calibrations, beam-spot, ...), a dedicated workflow was set up to handle the strict firewall between the online environment and the offline one, the Offline Dropbox (more information on the details of this workflow can be found in [7]). Figure 1. summarizes these two workflows and shows the main clients writing into and reading from the databases.

An important requirement in the design of the CMS conditions software is the one for consistent and transparent access to conditions via common software using object-relational mapping [6]. One major aspect of this is the focus on data integrity which for example means...
that data (esp. IOVs) are never deleted; data can only be added and updated (if necessary). This ensures the needed consistency while the conditions software transparently selects the latest IOV for the specific request from the user.

3. CMS conditions data - the challenge
The need for the worldwide distribution to all users of the various Tier-N centres in CMS comprises the main challenge for the CMS conditions data. Alone at the Tier-0 and Tier-1 centres, there are typically several tens of thousands of reconstruction jobs running, each of them needs access to the conditions data relevant for the data they are processing. For each job, a few hundred queries to the databases are needed to get all the relevant conditions. As this would imply a large load on the databases, we exploit the fact that conditions data, once written into the database, do not change any more (new data can only be appended, see above). Instead of having each of the jobs querying the database directly, jobs query a service which caches the queries in a hierarchical caching infrastructure, the Frontier squid-based caching service [8]. This way, the number of queries to the database is reduced by several orders of magnitude, while still ensuring that all jobs get the data in a fully transparent way.

Also in the online environment the Frontier squid cache system is used. Here the start of the processes at the high level trigger farm with thousands of jobs at every start of run needs fast access to the conditions. The hierarchical structure of the Frontier cache system allows a fast distribution of the data to all processes, reducing the time to start up the system to the minimum.

A number of web services were developed to allow detector experts and other users to visualize and control the conditions data stored in the database and ease the creation and monitoring of the Global Tags for the conditions.

4. Evolution and Performance of the CMS Databases
The space used in the online and offline CMS databases grows steadily, with a bit of slow-down during the phases when there is no data taking (winter), as detectors are switched off and only a reduced amount of conditions information is stored. On average the growth is about 1.5 TB in each year (since data taking started in 2010) for both, online and offline, databases. At present, each of the databases contains slightly more than 5 TB of data. Figure 2 shows the evolution over time since the start of the data taking. The slightly larger increase at the end of 2010 was caused by the integration of a large new application into the production database (moving its data from the integration test database) and is therefore exceptional.

The offline conditions data is only a small fraction of the total data stored in the databases, at present it amounts to about 300 GB with a growth of about 20 GB/year - mainly during the data-taking period (ca. 2 GB/month during data taking). For the processing of data
reconstruction and Monte-Carlo simulations about 50 Global Tags for the offline conditions are created every month.

The performance of the databases was excellent during the years, the overall availability for the online database was 99.88 % (a total of 10.5 hours of downtime) during the whole year 2011, the offline database was available 99.64 % (a total of 30.7 hours of downtime) in 2011. The downtimes here include planned interventions (full reboots for some security updates) as well as unplanned downtimes (e.g. due to power-cuts). The SQL query time in both databases has been very stable and is usually in the range of a few msec, fast enough to not cause any problems for the applications.

The good performance of the production databases was clearly helped by the strict implementation of the policy to always test any application in an integration database for an extended period of time and with close monitoring before moving the application to the production database.

5. Monitoring of the database services

It is essential for the proper functioning of the CMS databases and their associated services to closely monitor a number of parameters, ranging from hardware related numbers like disk read and write rates, SQL query times, and CPU usage over infrastructure related numbers like network rates to the top-level checking of availability of the various services proper. To achieve this, we have set up a Nagios [9] based monitoring and alarming system with a number of customized plugins specific to the services monitored. This system covers the basic services and a set of high-level views for some of the more critical applications like the O2O and the Offline Dropbox [10]. This monitoring and alarming system is extending the more general monitoring for the DB services at CERN and allows the main stakeholders in CMS to perform detailed checks on their specific domains: the condition DB experts can control the workflows involved in populating the databases, while the detector experts get information about the status of their submitted requests to update conditions. The active notifications of problems to the experts as set up with Nagios allows these experts to intervene at a very early stage before the problem becomes severe.
6. Summary and Outlook
Taking advantage of the technical stop of the LHC accelerator in winter 2011/2012, the database systems in CMS were upgraded to replace the aging hardware with a new, more performant one and at the same time improve the overall redundancy of the systems. In addition, the version of the Oracle database system was upgraded from 10.2 to 11g in order to profit from the new technologies, like the replacement of the Oracle Streams by the new Oracle Active Data Guard (ADG) technology to replicate the whole offline database in a read-only manner at the CERN computer centre. In the first few months after the upgrade our experience with using the new system was overall positive, small problems were quickly fixed by the CERN team of database administrators.

The basic structure of the conditions as a set of key-value pairs suggests that they are ideally suited to be stored using the recent key-value stores (a.k.a. “NoSQL” databases). In the medium to long term we are planning to evaluate this option for the CMS conditions and will look into studies of their performance as well as deployment, maintenance and migration of applications to use the new stores. Clearly the design and use of a central software layer between the applications and the database will help the latter.

Acknowledgments
We would like to thank the team of database administrators in the CERN IT department for their excellent work in keeping the performance of the CMS databases at a very high level and for their continued support for the developers of the CMS applications.

References
[1] CMS Collaboration 2008 The CMS experiment at the CERN LHC JINST 3 S08004, doi:10.1088/1748-0221/3/08/S08004
[2] Hufnagel D et al 2011 The architecture and operation of the CMS Tier-0, J. Phys.: Conf. Ser. 331 032017
[3] Egeland R, Wildish T and Metson S 2008 Data transfer infrastructure for CMS data taking, XII Advanced Computing and Analysis Techniques in Physics Research (Erice, Italy: Proceedings of Science)
[4] Afaq A et al 2008 The CMS Dataset Bookkeeping Service, J. Phys.: Conf. Ser. 119 072001, http://iopscience.iop.org/1742-6596/119/7/072001/
[5] Giffels M et al 2012 Data Bookkeeping Service 3 - A new event data catalog for CMS, paper [187] at this conference
[6] Govi G et al 2012 Handling of time-critical Conditions Data in the CMS experiment - Experience of the first year of data taking, paper [351] at this conference
[7] Cavallari F et al 2011 Time-critical Database Condition Data Handling in the CMS Experiment During the First Data Taking Period J. Phys.: Conf. Ser. 331 042007
[8] Dykstra D and Lueking L 2010 Greatly improved cache update times for conditions data with Frontier/Squid J. Phys.: Conf. Ser. 219 072034
[9] Barth W 2008 Nagios: System and Network Monitoring, 2nd edition (San Francisco, No Starch Press) ISBN 1-59327-179-4, http://www.nagios.org
[10] Di Guida S et al 2012 Monitor and alarm system for time-critical conditions data handling, paper [202] at this conference