CMS CR -2010/296

The Compact Muon Solenoid Experiment

Conference Report

Mailing address: CMS CERN, CH-1211 GENEVA 23, Switzerland

09 December 2010

Design and early experience with promoting user-created data in CMS

Manuel Giffels for the CMS Collaboration

Abstract

The Computing Model of the CMS experiment does not address transfering user-created data between different Grid sites. Due to the limited resources of a single site, distribution of individual user-created datasets between sites is crucial to ensure accessibility. In contrast to official datasets, there are no special requirements for user datasets (e.g. concerning data quality). The StoreResults service provides a mechanism to elevate user-created datasets to central bookkeeping ensuring the data quality is the same as an official dataset. This is a prerequisite for further distribution within the CMS dataset infrastructure.

Presented at *CHEP2010: International Conference on Computing in High Energy and Nuclear Physics 2010*

Design and early experience with promoting user-created data in CMS

M Giffels[1] **and E W Vaandering**[2]

[1] III. Physikalisches Institut B, RWTH Aachen
[2] Fermi National Accelerator Laboratory, Batavia, IL 60510, U.S.A

E-mail: `giffels@physik.rwth-aachen.de, ewv@fnal.gov`

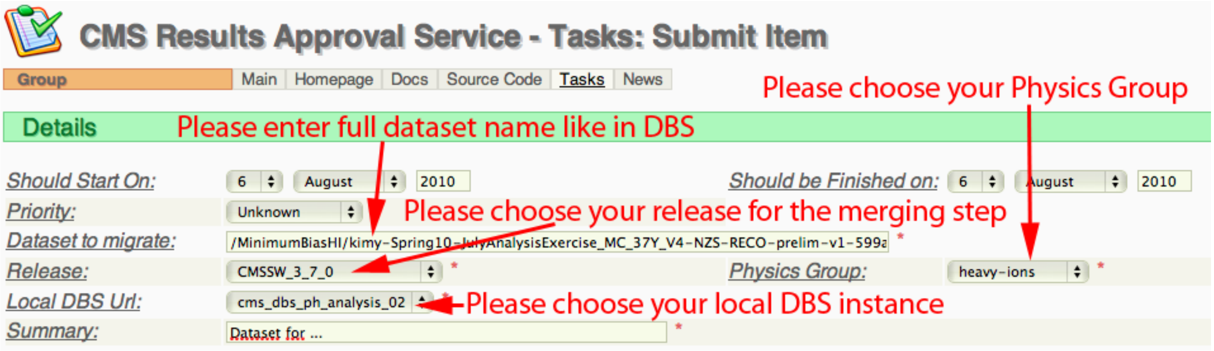
Abstract. The Computing Model of the CMS experiment [1] does not address transfering user-created data between different Grid sites. Due to the limited resources of a single site, distribution of individual user-created datasets between sites is crucial to ensure accessibility. In contrast to official datasets, there are no special requirements for user datasets (e.g. concerning data quality). The StoreResults service provides a mechanism to elevate user-created datasets to central bookkeeping ensuring the data quality is the same as an official dataset. This is a prerequisite for further distribution within the CMS dataset infrastructure.

1. Introduction

In the CMS experiment official datasets and user-created datasets are differentiated. Whereas official datasets are centrally produced by the CMS Data Operations group, users are allowed to produce and store their own datasets containing any kind of data at a Tier 2 Center. In contrast to official datasets, there are no requirements concerning data quality, usefulness or appropriate size to be transfered or stored on tape. User-created data is generally located in the private user space at the their designated Tier 2 and can be registered in a local scope data bookkeeping server, the CMS Dataset Bookkeeping System (DBS) [2]. The provided Grid tools can perform a distributed analysis on user-created data. In principle, this dataset can be analysed by any user of the collaboration, however only at the Tier 2 center hosting the dataset, which naturally has a limited number of job slots. Later the dataset created by the user may become important for many other users or even a whole Analysis Group. To provide better availability it is reasonable to distribute the dataset to additional Tier 2 centers or even to a Tier 1 center for custodial storage on tape. However, the CMS data transfer system (PhEDEx) [3] can only handle official data registered in the central bookkeeping service. Therefore, it is necessary that the user-created dataset becomes an official dataset fitting all the requirements of CMS. The StoreResults service described in this paper provides a mechanism to elevate user-created datasets ensuring all CMS data quality requirements for official datasets.

2. The Design of the StoreResults Service

The current system is ad-hoc based around a Savannah request and problem tracker for approvals and on the CMS ProdAgent production framework for distributed processing [4]. The Savannah interface was chosen to be the interim interface to StoreResults, until the new RequestManager service has been put in place. The RequestManager is designed to handle the data processing requests of CMS, such as StoreResults requests.



CMS Results Approval Service - Tasks: Submit Item

Please choose your Physics Group

Please enter full dataset name like in DBS

Should Start On: 6 August 2010

Should be Finished on: 6 August 2010

Priority: Unknown

Please choose your release for the merging step

Dataset to migrate: /MinimumBiasHI/kimy-Spring10-JulyAnalysisExercise_MC_37Y_V4-NZS-RECO-prelim-v1-599a *

Release: CMSSW_3_7_0 *

Physics Group: heavy-ions *

Local DBS Url: cms_dbs_ph_analysis_02 *

Please choose your local DBS instance

Summary: Dataset for ... *

Figure 1. Screenshot of the currently used request interface for the StoreResults service in CMS.

The StoreResults service is implemented as a component of the CMS ProdAgent framework. The structure of its implementation is depicted in figure 2. This ProdAgent component consists of three parts. The RequestQuery part handles the communication with the Request Interface. The requests and their status are periodically synced with the StoreResults table in the ProdAgentDB. New requests are assigned to a physics group Savannah squad according to the given information in the request. These squads are used to notify the physics group conveners or their representatives about new requests awaiting their approval or rejection. Once the request has been approved, a json configuration containing the necessary information is created and the operator is notified that the task is now ready for submission. The actual task submission is manually triggered by the operator and the task is passed to the job handling components of ProdAgent using the ResultsFeeder interface. The bookkeeping, central steering and the monitoring of the processed tasks is done by the StoreResultsAccountant. For this purpose the StoreResultsAccountant periodically queries the internal ProdAgentDB to fetch the current status of the merge jobs processed. In addition, the StoreResultsAccountant also triggers the injection into global DBS and PhEDEx once all merge jobs are successfully done. Afterwards the StoreResultsAccount closes the open request in Savannah by calling a member function of RequestQuery and a notification mail is sent to the requestor, group representatives and the operators.

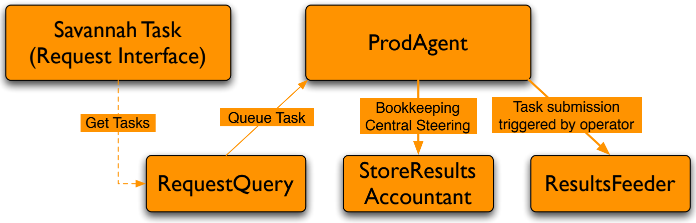


Figure 2. Structure of the current implementation of the StoreResults service as component in the ProdAgent framework.

3. The Process of Promoting User-created Data

The sequence of promoting user-created data is outlined in figure 3. A user belonging to a specific physics group creates a private dataset or skim using the CMS software tool for distributed analysis (CRAB) [5]. The output of the users jobs is either staged-out to `/store/group/<group name>` or to `/store/user/<user name>` at a group or user Tier 2. Afterwards the dataset is registered in the local scope bookkeeping system, so that it is available for distributed analysis using CRAB.

Since the dataset resides only at one Tier 2 site, the number of analysis slots is limited. To make the dataset available for a larger group of users it is essential to distribute it over several Tier 2 sites. The CMS data transfer systems supports only official datasets registered in global bookkeeping. To proceed the elevation to global bookkeeping the user or a group representative can make a StoreResults request via the Savannah interface. To ensure the usefulness of the dataset an approval by the physics group convener or one of its representatives is required. Once the request has been approved the elevation is started.

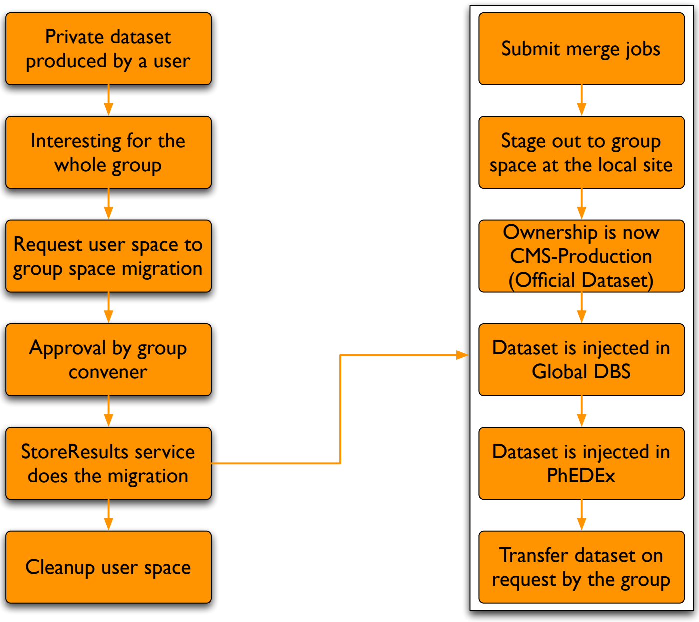

Private dataset produced by a user
Interesting for the whole group
Request user space to group space migration
Approval by group convener
StoreResults service does the migration
Cleanup user space
Submit merge jobs
Stage out to group space at the local site
Ownership is now CMS-Production (Official Dataset)
Dataset is injected in Global DBS
Dataset is injected in PhEDEx
Transfer dataset on request by the group

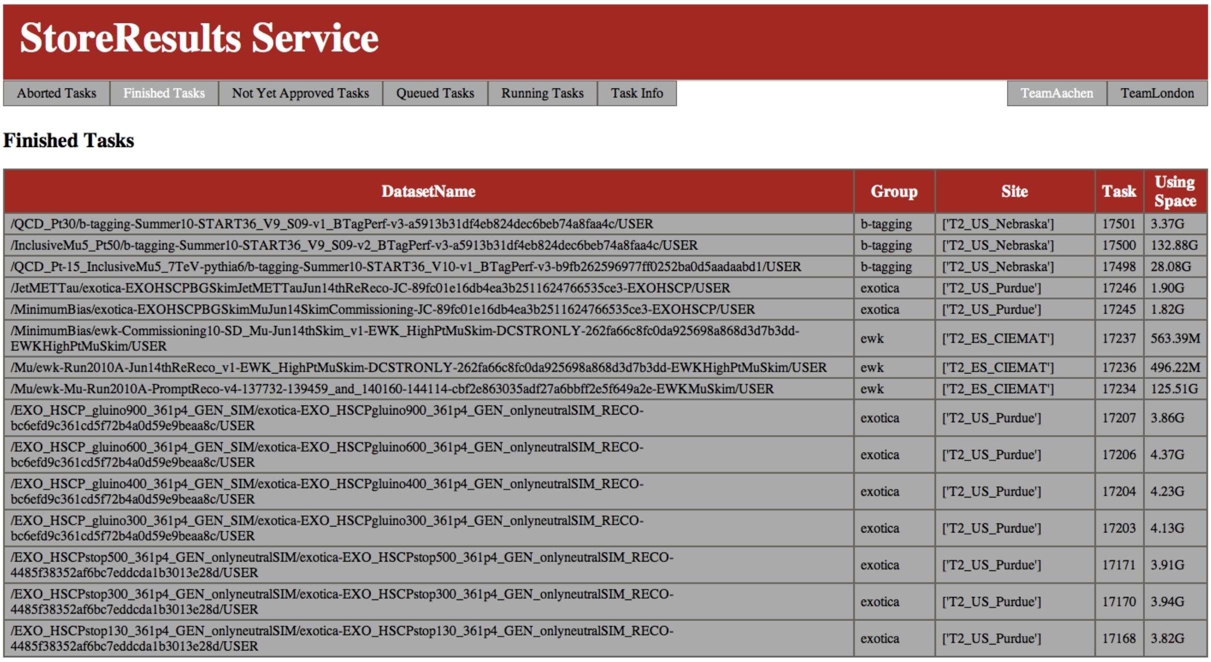
Figure 3. The sequence of promoting user-created data in CMS using the StoreResults service.

For data transfer and possibly custodial storage on tape at a Tier 1 center it is mandatory that the individual files of the migrated dataset have a reasonable size, which is recommended to be at least 2 GB. Therefore, merge jobs are submitted to the Tier 2 center, where the user dataset is located. In addition, merging using the CMS software guarantees that the used dataformat is compatible with the Event Data Model (EDM) of CMS. The output of these merge jobs are staged-out to `/store/results/<group name>` and the dataset becomes the ownership of the requesting group. Since the dataset fulfils now the given requirements concerning file size, usefulness and data quality, it becomes an official CMS dataset and is injected into global bookkeeping and the CMS transfer system PhEDEx. The requesting group is now responsible for initiating data transfers to other locations using PhEDEx and also for the deletion once the dataset becomes obsolete.

The last step of the promotion is the cleanup of the `/store/group/<group name>` or `/store/user/<user name>` directory and the invalidation of the dataset in local scope bookkeeping, which is the responsibility of the requesting physics group or user.

4. Monitoring of the StoreResults Service

To monitor the progress of the elevation a Web Service using CherryPy has been developed. The Web Service is running on each instance of the StoreResults service. The access has been restricted to group representatives for performance and security reasons. The authentication is done by using X.509 grid certificates. A screenshot of the StoreResults monitoring website is shown in figure 4. On one hand the Web Service allows the physics groups to monitor the progress of their requests in real time. On the other hand, it also very useful for the operators of the StoreResults service, to track the handled requests. Since the Standard Output and Standard Error from retrieved jobs are also available through the Web Service, debugging of failed jobs is much simpler.



StoreResults Service

Finished Tasks

DatasetName	Group	Site	Task	Using Space
/QCD_Pt30/b-tagging-Summer10-START36_V9_S09-v1_BTagPerf-v3-a5913b31df4eb824dec6beb74a8faa4c/USER	b-tagging	['T2_US_Nebraska']	17501	3.37G
/InclusiveMu5_Pt50/b-tagging-Summer10-START36_V9_S09-v2_BTagPerf-v3-a5913b31df4eb824dec6beb74a8faa4c/USER	b-tagging	['T2_US_Nebraska']	17500	132.88G
/QCD_Pt-15_InclusiveMu5_7TeV-pythia6/b-tagging-Summer10-START36_V10-v1_BTagPerf-v3-b9fb262596977ff0252ba0d5aadaabd1/USER	b-tagging	['T2_US_Nebraska']	17498	28.08G
/JetMETTau/exotica-EXOHSCPBGSkimJetMETTauJun14thReReco-JC-89fc01e16db4ea3b2511624766535ce3-EXOHSCP/USER	exotica	['T2_US_Purdue']	17246	1.90G
/MinimumBias/exotica-EXOHSCPBGSkimMuJun14SkimCommissioning-JC-89fc01e16db4ea3b2511624766535ce3-EXOHSCP/USER	exotica	['T2_US_Purdue']	17245	1.82G
/MinimumBias/ewk-Commissioning10-SD_Mu-Jun14thSkim_v1-EWK_HighPtMuSkim-DCSTRONLY-262fa66c8fc0da925698a868d3d7b3dd-EWKHighPtMuSkim/USER	ewk	['T2_ES_CIEMAT']	17237	563.39M
/Mu/ewk-Run2010A-Jun14thReReco_v1-EWK_HighPtMuSkim-DCSTRONLY-262fa66c8fc0da925698a868d3d7b3dd-EWKHighPtMuSkim/USER	ewk	['T2_ES_CIEMAT']	17236	496.22M
/Mu/ewk-Mu-Run2010A-PromptReco-v4-137732-139459_and_140160-144114-cbf2e863035adf27a6bbff2e5f649a2e-EWKMuSkim/USER	ewk	['T2_ES_CIEMAT']	17234	125.51G
/EXO_HSCP_gluino900_361p4_GEN_SIM/exotica-EXO_HSCPgluino900_361p4_GEN_onlyneutralSIM_RECO-bc6efd9c361cd5f72b4a0d59e9beaa8c/USER	exotica	['T2_US_Purdue']	17207	3.86G
/EXO_HSCP_gluino600_361p4_GEN_SIM/exotica-EXO_HSCPgluino600_361p4_GEN_onlyneutralSIM_RECO-bc6efd9c361cd5f72b4a0d59e9beaa8c/USER	exotica	['T2_US_Purdue']	17206	4.37G
/EXO_HSCP_gluino400_361p4_GEN_SIM/exotica-EXO_HSCPgluino400_361p4_GEN_onlyneutralSIM_RECO-bc6efd9c361cd5f72b4a0d59e9beaa8c/USER	exotica	['T2_US_Purdue']	17204	4.23G
/EXO_HSCP_gluino300_361p4_GEN_SIM/exotica-EXO_HSCPgluino300_361p4_GEN_onlyneutralSIM_RECO-bc6efd9c361cd5f72b4a0d59e9beaa8c/USER	exotica	['T2_US_Purdue']	17203	4.13G
/EXO_HSCPstop500_361p4_GEN_onlyneutralSIM/exotica-EXO_HSCPstop500_361p4_GEN_onlyneutralSIM_RECO-4485f38352af6bc7eddcda1b3013e28d/USER	exotica	['T2_US_Purdue']	17171	3.91G
/EXO_HSCPstop300_361p4_GEN_onlyneutralSIM/exotica-EXO_HSCPstop300_361p4_GEN_onlyneutralSIM_RECO-4485f38352af6bc7eddcda1b3013e28d/USER	exotica	['T2_US_Purdue']	17170	3.94G
/EXO_HSCPstop130_361p4_GEN_onlyneutralSIM/exotica-EXO_HSCPstop130_361p4_GEN_onlyneutralSIM_RECO-4485f38352af6bc7eddcda1b3013e28d/USER	exotica	['T2_US_Purdue']	17168	3.82G

Figure 4. Screenshot of the StoreResults monitoring website.

From the technical point of view, the StoreResults monitoring obtains its information about the processed and queued requests by querying the local ProdAgentDB and the bookkeeping database DBS.

5. Operation of the StoreResults Service

To handle the StoreResults requests, currently two StoreResults instances have been deployed. One is located at RWTH Aachen and the other one at Imperial College London. Since the task submission is done manually, operator actions are required. The StoreResults service is currently operated by three part-time operators. Two people at RWTH Aachen and one at Imperial College London. Furthermore, there are two part-time developers, one at Fermi National Accelerator Laboratory (FNAL) and one at RWTH Aachen.

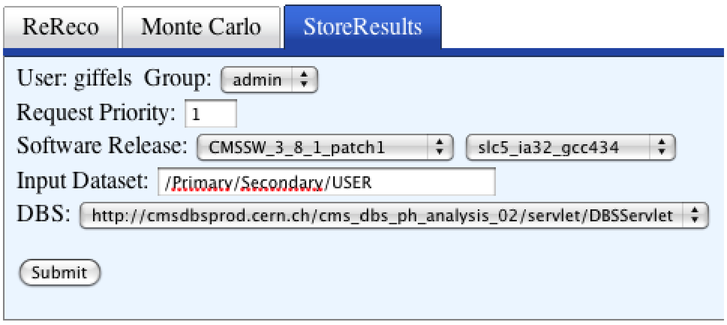
The primary concern of the physics groups is: “How long does the elevation take?” This question cannot be answered in general, since the time to complete the process depends on various factors. The parent dataset migration from local scope bookkeeping to global bookkeeping takes usually 10 to 60 minutes. This step is necessary to keep the parentage information in the database. A usual task comprises 1-1000 merge jobs. Each jobs lasts between two hours and two days. The elevation task is finished only once the last job is done. However, the merge jobs of StoreResults have a higher priority, since their are running with the CMS Production-Role. Due to recurring Grid errors, each merge job will be resubmitted up to ten times, before it is declared as failed and additional operator action is required. Once all jobs are done, the PhEDEx subscription to a site has to be approved by the operator or the data manager of the Grid site, even if the data resides already at the given site. On average it takes 1–2 days to complete all the steps in elevating a dataset.

On average the StoreResults service deals with 1-2 requests per day. Usually, those requests emerge in clusters of 2-10. Both instances handled 333 requests in 2010 so far. About 98% of the request were successfully elevated and the average time is around 46 hours per request.

6. Future Development

The CMS experiment is going to change its production framework for distributed processing on the Grid to WMAgent, which is an evolution of the currently used ProdAgent system. That means the StoreResults service is going to be completely reimplemented in the WMAgent framework and will also natively support growing datasets.

Accompanying to the transition to WMAgent, the CMS experiment will use the RequestManager to handle data processing requests. The RequestManager combines the possibility of making requests, approvals by authorized people as well as a tracking of their status. Therefore, the currently used Savannah interface will be replaced in the future by an interface integrated in the RequestManager to allow a more convenient handling of StoreResults requests by a unified interface for all data processing requests.



User: giffels Group: admin

Request Priority: 1

Software Release: CMSSW_3_8_1_patch1 slc5_ia32_gcc434

Input Dataset: /Primary/Secondary/USER

DBS: http://cmsdbsprod.cern.ch/cms_dbs_ph_analysis_02/servlet/DBSServlet

Figure 5. Screenshot of the future RequestManager interface for the StoreResults service.

References

[1] The CMS Collaboration, The CMS Computing Model, CERN LHCC 2004-035

[2] Anzar Afaq, The CMS Dataset Bookkeeping Service, J. Phys.: Conf. Ser. 119 072001

[3] Nicolo Magini, Improving CMS Data Transfers among its Distributed Computing Facilities in these proceedings, PS 23-3-204

[4] Evans D et al., The CMS Monte Carlo Production System: Development and Design, 18th Hadron Collider Physics Symposium 2007 (HCP 2007) 20-26 May 2007, La Biodola, Isola d'Elba, Italy. Published in Nuclear Physics B - Proceedings Supplements,Volumes 177-178 (2008) 285-286

[5] Eric Vaandering, CMS distributed analysis infrastructure and operations: experience with the first LHC data in these proceedings, PS 25-2-212