



Topics

- Bash
- Tier-0 Procedures
- Brief Report
- AccessingData
- Multicore Processing
- ToolsGuide
- WorkflowsAndJobs
- GroupsAndPeopleToPing
- Procedimientos
- SLS Alarms
- ChangingOraclePassword
- Questions

- TempReport

- Kibana
- Preguntas
- Tier0 Monitoring
- Tier0 areas on EOS

Content

AFS

Url to enable work area and increase AFS quota

<https://resources.web.cern.ch/resources/Manage/AFS/Settings.aspx>

Machines

VM	Description	DB
vocms001	Dirk's test machine	CMS_TOAST_REPLAY1
vocms015	T0 test machine	CMS_TOAST_REPLAY2
vocms047	T0 test machine	CMS_TOAST_REPLAY3
vocms062	T0 prod machine	CMS_TOAST_1

Useful commands

Comando	Explicacion	Uso comun
ls -lathlr	Ver los archivos en orden	
ls -lh	Ver los archivos en orden cronologico	
grep <regExpr>	Permite filtrar una lista de resultados dada	grep FNAL

	una expresion regular							
tail -<numero> <nombreArchivo>	Muestra los <numero> ultimos archivos							
xrdcp	Copia un archivo							
tar -zxvf <tarball>	Descomprime un tar							
pwd	Me da mi ubicacion actual							
scp	Copia de forma "segura" un archivo							
ps aux	egrep '<pr1> \	<pr2> \	<pr3>'	Permite ver si hay instancias en ejecucion	ps aux	egrep 'wmcore	mysql	couch'
find <dir> -name <searchTerm>	Permite buscar en una carpeta	find ./JobCreator/JobCache -name LogCollect						
sort file.txt	uniq	Muestra una lista de cosa sin repeticiones						
sort file.txt \	uniq -d	Muestra solo las repeticiones de una lista de cosas						
wc -l <filename>	print the line count (note that if the last line does not have \n, it will not be counted)							
wc -c <filename>	print the byte count							
wc -m <filename>	print the character count							
wc -L <filename>	print the length of longest line							
wc -w <filename>	print the word count							



Fallar Jobs / Fail Jobs / Failing Jobs

```
source /data/tier0/admin/env.sh
```

```
$manage execute-agent paused-jobs -f -w PromptReco_Run207231_SingleMu
```

```
$manage execute-agent paused-jobs -f -w <WORKFLOW>
```

```
$manage execute-agent paused-jobs -f -j 9081320
$manage execute-agent paused-jobs -f -j <T0AST_JOB_ID>
```

Global Tag

Especifica las condiciones de procesamiento. Estas se encuentran en CondDB.

Que hay que hacer cuando anuncian un GlobalTag.

Condor & GlideinWMS

condor_q	Ver detalles de la ejecucion de condor en esa maquina
condor_q -w:	Ver mas detalles de la ejecucion de condor en esa maquina
condor_q -analyze	<p>Aplica sobre el collector pool (vocms97 - production pool, vocms097- global pool vocms0?? - T0pool),</p> <p>no para los schedd. Se puede hacer sin sudo, pero hay que cargar primero algo de codigo usando</p> <p>source /data/srv/wmagent/env.sh</p>

Cuando ES T0 Agent

```
source /data/tier0/admin/env.sh
```

|

condor_q -better-analyze	
--------------------------	--

condor_status	
condor_rm	
condor_history -constraint	
condor_history -const 'JobStatus=?=3'	
condor_history -l 2498.0 \	less
condor_ssh_to_job <jobID>	<p>Conecta con el workernode que esta ejecutando un job en particular.</p> <p>Cuando se va a un workernode, normalmente lo que se debe encontrar es:</p> <ul style="list-style-type: none"> • los logs de condor (err & out). Estos son los que, cuando condor termina su ejecucion son devueltos al WMAgent. • El JobPackage.pkl, ese es propiamente "El Job" • Algunos directorios de WMCORE • El directorio /job. En este directorio se encuentran los archivos de la ejecucion del Job. En este a su vez se encontrara <p>+ Logs generales del job + El directorio /WMTaskSpace. Este contiene las tareas que tiene el job. Estos se corresponden con las carpetas y pueden ser: - cmsRun1: Aca se encuentran los archivos de procesamiento de informacion. - stageout - logArch</p>
https://htcondor-wiki.cs.wisc.edu/index.cgi/wiki?p=MagicNumbers	

Procedimientos

Problema: Un job lleva corriendo demasiado tiempo

Condor & GlideinWMS


Descripcion: Hay jobs que no deberian duran corriendo tanto tiempo. Cuando hablamos de dias es un problema, (sobre todo en T0).

Procedimiento:

- Acceder a la maquina donde se encuentra el job.
- Como el job esta corriendo, al hacer `condor_q` y/o `condor_q -w` se deberia poder ver la informacion de este.
- Con el ID del job, se puede usar `condor_ssh_to_job` para ir al workernode y revisar que esta pasando. Hay senales de alerta como:
 - + Archivo de log muy grande: Normalmente los archivos de log no son tan grandes. Puede ser que el job este estancado haciendo algo repetidamente.
 - +

Cambiando la configuracion

Hay diferentes archivos de configuracion importantes en la operacion del T0. Se presenta una lista de ellos

Archivo	Ubicacion	Usuario	Componente	Descripcion
ReplayOfflineConfiguration.py	/data/tier0/admin/	cmst1	REPLAY	Permite cambiar configuraciones antes de cambiar una replay
WMAgent.secrets	/data/tier0/admin/	cmst1	REPLAY - PROD	Contiene la informacion necesaria de una maquina para conectarse a T0AST (Oracle), WMSTATS, PHEDEX, DQM, CoachDB, etc. 
config.py	\$config (hacer echo)	cmst1	REPLAY	Permite cambiar configuraciones de un agente, Es necesario hacer source /data/tier0/admin/env.sh

Reiniciar un workflow

Es posible que algunos jobs se pausen, si la causa de esta pausa es incidental *(por ejemplo un certificado que expiro en ese momento pero luego fue renovado) se pueden reiniciar. para ello se usa

```
$manage execute-agent paused-jobs -r -w nombre_del_wf
```

ReplayOfflineConfiguration.py

Correr un replay

1) modificar parches si es el caso en el archivo 00_software

2) Ir a vim `admin/ReplayOfflineConfiguration.py`

Ahi cambiar

```
# Defaults for processing version
```

```
defaultProcVersion = 14
```

```
expressProcVersion = 14
```

```
alcarawProcVersion = 14
```

por

```
# Defaults for processing version
```

```
defaultProcVersion = 15
```

```
expressProcVersion = 15
```

```
alcarawProcVersion = 15
```

y

```
setDefaultScramArch(tier0Config, "slc6_amd64_gcc481")
```

por

```
setDefaultScramArch(tier0Config, "slc6_amd64_gcc491")
```

3)correr 00_software.sh

4)correr 00_deploy.sh (es normal que este tarde un resto)

5)correr 00_start_Agent.sh (si uno modifica este archivo puede modificar el Agent a correr si hubiera varios disponibles)

6)ir a cd replayInject

7)verificar que el archivo resend.txt este apuntando al run correcto (ej, 231135.txt)

```
ln -sf <config_replay_text_file>.txt resend.txt
```

9)correr ./t0_control.sh restart

Este reinicia

Starting Logger/LoggerReceiver

Starting Tier0Injector /Tier0InjectorManager

Starting Tier0Injector /Tier0InjectorWorker

10)correr./t0_resend.sh start

Este inicia la inyecci[on

11)El WMAgent esta desplegado en /data/tier0/srv/wmagent/current/install/tier0/

NOTA: SI SE REQUIERE INICIAR / DETENER UN COMPONENTE PARTICULAR

1. source /data/tier0/admin/env.sh
2. \$manage execute-agent wmcord --shutdown --components=JobSubmitter
3. \$manage execute-agent wmcord --start --components=JobSubmitter
4. \$manage execute-agent wmcord --restart --components=JobSubmitter

Si se reinicia el PhedexInjector hay que hacer source de WMCore source /data/tier0/srv/wmagent/current/apps/t0/etc/profile.d/init.sh

Revisar el ClassAd de un job

1. condor_history | less (Muestra
2. condor_history -const 'JobStatus=?=3'
3. condor_history -l 2498.0 \ less

Iniciar Agente

- 1) 00_Software
- 2) 00_Deploy

Correr un replay

- 3) REvisar si hay algo que modificar en /data/tier0/admin/replay*config*.py (AL menos subir el processing version tambien a veces se cambia el DQ o la version de CMSSW)
- 4) 00_startAgent
- 5) Ir a /data/tier0/replayinject
.t0_control.sh restart
.t0_resend.sh start

Detener el agente

1. Acceder a /data/tier0 como cmst1
2. 00_StopAgent
 1. condor_rm -all
 2. condor_rm <id del job>
 3. condor_rm <id del que lo mando (por ejemplo cmst1)>
3. chequear con ps aux | egrep 'wmcoremysqlcouch' que no haya nada de eso corriendo

Recolectar Jobs de una run (generar el archivo de la run para despues poder hacer una replay)

- Se debe ir al endpoint del transfer system (en este caso es vocms001.cern.ch:/data/TransferSystem/), **loggeandose como cmsprod**
- Ejecutar `cat Logs/General.log* | grep "'Tier0Inject' => '1'" | grep "'RUNNUMBER' => '224379'" > Run224379.txt.`
 - ◆ Entra a Logs/General.log (como tienen fecha se puede poner una expresion regular o un nombre mas largo para lograr coincidencias mas precisas)
 - ◆ Su salida que es el contenido de el (los) archivos que cumplan con la primera parte, se filtra para los que hayan sido inyectados en el Tier0. La instruccion podria quedar concluida ahi. Si se quisiera obtener solo los archivos de una run particular se procede con la parte 3.
 - ◆ Se hace un filtrado sobre el resultado del filtro anterior con la expresion mostrada mas arriba
- Algunos ejemplos son:
 - ◆ `cat Logs/General.log | grep "'Tier0Inject' => '1'" | grep "'RUNNUMBER' => '224379'" > Run224379.txt.`
 - ◆ `cat Logs/General.log | grep "'Tier0Inject' => '1'"`
 - ◆ `cat Logs/General.log.2015011* | grep "'Tier0Inject' => '1'" > SalidaJ.txt`
- Es normal realizar esta tarea despues de una Global Run o una MWGR. Normalmente los logs de los dias de estar Run son mas pesados y por tanto son facilmente identificables, para realizar esta tarea entonces basta con realizar los pasos antes mencionados y luego copiar los archivos a el area correspondiente /data/tier0/replayinject de cada maquina. **Es necesario conectarse como cmst1 para hacer esta copia.** Despues de realizar la conexion se puede utilizar:
 - ◆ `scp <maquinaOrigen>:<rutaOrigen> <ruta destino>`
- Algunos ejemplos son:
 - ◆ `scp vocms001.cern.ch:/data/TransferSystem/Run232-233mwgr10.txt /data/tier0/replayinject`
 - ◆ `scp vocms001.cern.ch:/data/TransferSystem/Run232miniGR.txt /data/tier0/replayinject`
- Link to get the lastrun <https://cmsweb.cern.ch/t0wmadatasvc/prod/firstconditionsaferun>

Tareas de operacion sobre el T0 WMAgent

- Descripcion general de la arquitectura global
- Descripcion general de los componentes del agente
- Configuracion del agente
- Inicio del agente

- Inicio de la inyeccion
- Revision de logs y componentes del agente

Tareas de operacion sobre los jobs

Cuando una replay esta corriendo normalmente hay diferentes tipos de ejecucion

--++1. HTCondor Se ingresa a la maquina que esta corriendo el replay. En esta se puede ejecutar

- condor_q
- condor_q -w
- condor_q -analyze
- condor_q -betterAnalyze

Estos comandos mostraran los estados de los jobs de condor que estan siendo enviados. Es normal que algunos de estos esten en (I)dle y otros (R)unning. Los jobs pueden durar desde minutos hasta dias. Sin embargo no es normal (ni deseable) que duren demasiado (dependiendo del tipo de job, dias puede empezar a ser preocupante).

--++2. T0AST Oracle Database Como se ve en la seccion de maquinas, cada una de las maquinas tiene una instancia propia de T0AST con la que esta conectada. Cuando la replay esta corriendo se puede ver el estado de los jobs con las siguientes Querys

SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'created');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'createcooloff');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'cleanout');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'killed');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'complete');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'submitfailed');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'submitcooloff');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'retrydone');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'none');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'submitpaused');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'jobcooloff');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'executing');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'success');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'createpaused');
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'new');


```
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'createfailed');
```

```
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'exhausted');
```

```
SELECT id,name,cache_dir FROM wmbs_job where state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'jobpaused');
```

Sin embargo el nivel de importancia de todos los estados no es el mismo. En particular el estado mas importante es jobpaused. Cuando un job tiene errores por alguna razon este es reintentado tres veces (el valor de tres intentos es configurable en FIXME). Si despues de la tercera vez sigue con errores, el T0 WMAgent lo lleva al estado jobpaused. En ese caso se debe proceder a una revision en detalle de el (los) job(s) problematico(s).

```
SELECT COUNT(*) FROM wmbs_job where (state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'jobpaused') ) OR \ (state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'submitpaused') ) OR \ (state = (SELECT id FROM WMBS_JOB_STATE WHERE name = 'createpaused') );
```

--++3. Logs

Si se detecta una condicion anomala o hay jobs pausados se deberia seguir el siguiente procedimiento: --++3.0 Las consultas en el numeral anterior incluyen en su resultado el cache_dir. Esta es la ubicacion fisica dentro del Schedd (que para T0, se corresponde con la maquina donde se esta ejecutando el T0WMAgent) a donde se transfieren los logs de condor creados durante la ejecucion.

--++3.1 Revisar logs Dado que la direccion es absoluta se puede acceder a ella y consultar los tres tipos de log: log, out, err. Estos logs se deben ordenar y es buena consultar el del ultimo reintento. Son archivos de texto asi que se pueden explorar con vim.

--++3.2 Revisar pkl. El pkl es un archivo que tambien tiene informacion sobre la ejecucion del job. Tal archivo si se trata de revisar es dificil de leer, para poder comprenderlo podemos utilizar un script the python que nos ayude a formatearlo para que sea legible por un humano.

- Primero se debe hacer cargar codigo para poder correr las instrucciones sin problema

```
source /data/tier0/srv/wmagent/current/apps/t0/etc/profile.d/init.sh
```

- Luego Se debe abrir una consola de python, para esto basta ejecutar la instruccion

python

El script python es el siguiente

```
import cPickle
jobHandle = open("Report.pkl", "r")
loadedJob = cPickle.load(jobHandle)
jobHandle.close()
print loadedJob
```

- Esto mostrara en pantalla la informacion dentro del archivo de una manera comprensible

--++3.3 Revisar un Tarball Si no se encontro nada diciendo en los logs ni en los pkl, es una buena idea explorar el tarball. Para ello, tanto en el pkl como en los logs se puede encontrar la direccion fisica (PFN) de cada tarball.

Para explorar un tarball se puede copiar este a lxplus (se recomienda tener una carpeta llamada tarball para trabajar).

El comando de copia es lcg-cp

Un ejemplo de este comando es:

lcg-cp

root://cmseos.fnal.gov//lustre/unmerged/data/logs/prod/2015/1/29/PromptReco_Run231228_Cosmics/Reco/RecoMerge

Al descomprimir un tarball podemos encontrar mas informacion que nos puede ayudar a encontrar la causa de los problemas en la ejecucion del job.

```
source /data/tier0/srv/wmagent/current/apps/t0/etc/profile.d/init.sh
```

Configuracion para arrancar el agent

Archivo de secrets

Revision del LoggerSender.pl en el endpoint de transferencia

```
/data/TransferSystem/T0/src/Logger
```

Scripts SLS

Generacion de graficas para seguimiento (Weekly Report) con un ChroneJob

PhEDEx consulta de archivos y solicitud de transferencia/borrado de archivos

Cuando se corren replays, la mayoría de las veces no se necesitan los archivos resultantes, a menos que esto sea especificado en el ELog de la replay correspondiente. Cada determinado tiempo (CUANTO?) es necesario solicitar que los archivos no necesitados se borren. Para esto es necesario:

1. Revisar los archivos que se pueden borrar. Desde el último ELOG. Aquellos que no pueden ser borrados tendrán un mensaje especificándolo.

A continuación el registro de los últimos borrados:

Fecha Revisión	Reviewer	Link ELOG
02/04/2015	John	https://cms-logbook.cern.ch/elog/Tier-0+processing/12573
08/02/2015	Luis / John	https://cms-logbook.cern.ch/elog/Tier-0+processing/12294
02/10/2014	Luis	https://cms-logbook.cern.ch/elog/Tier-0+processing/11955

2. Después de tener claros los logs a borrar, se debe acceder a PhEDEx / Requests / Create Requests (<https://cmsweb.cern.ch/phedex/prod/Request::Create>). Ahí se encontrará un cuadro de búsqueda. Se pueden usar expresiones regulares, Algunos ejemplos de expresiones usadas

- `/*Tier0_Test_SUPERBUNNIES_vocms015*v2/*`
- `/*Tier0_Test_SUPERBUNNIES_vocms001*v33/*`
- `/*Tier0_Test_SUPERBUNNIES_vocms047*v2/*`

3. Se deben seleccionar los

4.

Cambiar la version de CMSSW en Produccion

1. Ir al archivo de configuracion /data/tier0/admin ProductionOfflineConfiguration.py y cambiar alla la linea correspondiente a la version de CMSSW con la nueva version.
2. Ejercutar los siguientes scripts, (se debe cambiar la parte final del query con la version conveniente) para buscar a partir de cual Run aplicara el cambio

```
select RECO_CONFIG.RUN_ID, CMSSW_VERSION.NAME from RECO_CONFIG
inner join CMSSW_VERSION on RECO_CONFIG.CMSSW_ID = CMSSW_VERSION.ID
where name = 'CMSSW_7_3_2_patch3';
```

```
select EXPRESS_CONFIG.RUN_ID, CMSSW_VERSION.NAME from EXPRESS_CONFIG
inner join CMSSW_VERSION on EXPRESS_CONFIG.RECO_CMSSW_ID = CMSSW_VERSION.ID
where name = 'CMSSW_7_3_2_patch3';
```

3. Anunciar en HN el cambio de la version y la informacion de la run a la cual aplicara

Conectar un T0 WMAgent a WMStats

1. Se accede al archivo de configuracion de T0 (VER ARCHIVOS DE CONFIGURACION) 1. En este archivo (config.py) se cambian las ocurrencias por la URL que se requiera (si se quiere conectar con el WMStats TESTBED o con PRODUCCION) 1. Si el agente NO ESTA CORRIENDO se debe cambiar tambien el archivo de secrets (VER ARCHIVOS DE CONFIGURACION - WMAgent.secrets) para que el atributo WMSTATS_URL tambien apunte a la URL del WMStats que se requiera (TESTBED o PRODUCCION) 1. Deploy / Redeploy del agent

Name	URL	Description
Production WMStats		
Testbed WMStats		

Consulta de Datasets, Runs, Files, etc.

Python Plugin

```
972 condor_q 973 ps aux | grep condor 974 export condor_config=/etc/condor/config.d 975 echo
$condor_config 976 condor_q 977 cd /data/srv/ 978 ls 979 export
CONDOR_CONFIG=/etc/condor/condor_config 980 cd /etc/condor/ 981 ls 982 condor_q 983 cd
/etc/profile.d 984 ls 985 less ~/.bashrc 986 acrontab -l 987 whoami 988 sh ~/.bashrc 989 echo
$CONDOR_CONFIG 990 sudo -u cmst1 /bin/bash --init-file ~cmst1/.bashrc 991 pwd 992 echo
$CONDOR_CONFIG 993 ls 994 ls .bash/ 995 ls .bash_history 996 cat .bash_history 997 echo
$CONDOR_CONFIG 998 grep -ir "condor" * 999 grep -ir "CONDOR_CONFIG" * 1000 history
```

Configuracion

135	KIT	PIC	CNAF	RAL	IN2P3	FNAL
Cosmics	X					
MiniBias		X				
HcalNzs			X			
NoBPTX				X		

228	KIT	PIC	CNAF	RAL	IN2P3	FNAL
Cosmics						X
MiniBias					X	

HcalNZS			X			
NoBPTX				X		

A que se debio el cambio?

El cambio se debio a que, como ya se habia llevado a cabo esa replay, se sabia en cuales nodos iba a ser ejecutado

Quando Los Logs estan vacios y hay una cosa de un REMOVE en el PKL o en el LOG

```
<a n="Reason"><s>The system macro SYSTEM_PERIODIC_REMOVE expression '((((NumJobStarts>9)=?=True
condor_history -b -m 1 490.0 -format '%s ' DiskUsage -format '%s ' NumShadowStarts -format '%s\n'
```

How To Resume a Job

Por Sitio

```
$manage execute-agent paused-jobs -r -s T2_CH_CERN_T0
```

Por Workflow

```
-w Workflow
```

Por Tarea

```
-t Repack
-t Express
-t Merge
-t Processing
-t Cleanup
-t LogCollect
```

Si se corre una run de 2012 se deben utilizar estos escenarios

```
# Configure scenarios
cosmicsScenario = "cosmics"
ppScenario = "pp"
hcalnzsScenario = "hcalnzs"
```

Error codes Log

65 - Indexes out of bounds (Reconstruction Step)

Error message

```
Begin processing the 8th record. Run 244492, Event 104971682, LumiSection 51 at 17-May-2015 04:34:27 CEST
%MSG-e TooManyErrors: SiStripRawToDigiModule:siStripDigis 17-May-2015 04:34:27 CEST Run: 244492
Total number of errors = 35390
%MSG
----- Begin Fatal Exception 17-May-2015 04:34:27 CEST-----
An exception of category 'InvalidDetId' occurred while
  [0] Processing run: 244492 lumi: 51 event: 104971682
  [1] Running path 'reconstruction_step'
  [2] Calling event method for module EcalRecHitProducer/'ecalRecHit'
Exception Message:
EBDetId: Cannot create object. Indexes out of bounds
```

```
eta = -81 phi = 361
----- End Fatal Exception -----
```

Procedure

- It was reported to Reconstruction Development HN
- The related jobs were failed

66 - Std exception while doing dqmoffline_step

Error message

```
----- Begin Fatal Exception 14-May-2015 14:05:54 CEST-----
An exception of category 'StdException' occurred while

[0] Processing run: 244163 lumi: 1 event: 20728
[1] Running path 'dqmoffline_step'
[2] Calling event method for module JetAnalyzer/'jetDQMANalyzerAk4CaloCleaned'
Exception Message:
A std::exception was thrown.
vector<bool>::_M_range_check: __n (which is 0) &gt;= this-&gt;size() (which is 0)
----- End Fatal Exception -----
```

Procedure

- It was reported to DQM Development HN (i.e. <https://hypernews.cern.ch/HyperNews/CMS/get/dqmDevel/2247.html>)
- After they ack, the jobs were failed

84 - Error opening Streamer Input File, file does not exist

Example Report

<https://cms-logbook.cern.ch/elog/Tier-0+processing/12771>

Error Message

```
18-May-2015 16:44:14 CEST Initiating request to open file
root://eoscms.cern.ch//store/t0streamer/Data/ExpressCosmics/000/244/807/run244807_ls0128_streamEx
----- Begin Fatal Exception 18-May-2015 16:44:14 CEST-----
An exception of category 'FileOpenError' occurred while
  [0] Constructing the EventProcessor
  [1] Constructing input source of type NewEventStreamFileReader
Exception Message:
StreamerInputFile::openStreamerFile Error Opening Streamer Input File, file does not exist:
root://eoscms.cern.ch//store/t0streamer/Data/ExpressCosmics/000/244/807/run244807_ls0128_streamEx
```

Procedure

- Checking the file exists in EOS and retrieving it manually from the agent.
- As it was, resuming a paused jobs, ssh_to_job to the corresponding worker node and retrieving it manually.
- As it was, we checked certificates, they looked ok.
- A file not found problem can hide other problems, i.e. the file can exist and be present, but it is not possible to read it, so the file was downloaded and the job was run locally. It worked with no problems.
- The problem affected runs running a new version of CMSSW so it could be a SW problem.

85 - Unknown code in event header (corrupted file)

Example Report

<https://cms-logbook.cern.ch/elog/Tier-0+processing/12698>

Error message

```
6-May-2015 14:21:09 CEST Successfully opened file root://eoscms.cern.ch//store/t0streamer/Data/C
----- Begin Fatal Exception 16-May-2015 14:21:10 CEST-----
An exception of category 'FileReadError' occurred while
  [0] Calling InputSource::getNextItemType
Exception Message:
StreamerInputFile::readEventMessage Failed reading streamer file, unknown code in event header
code = 0
----- End Fatal Exception -----
```

Procedure used

- Copying the local file to check if it exists and is accessible
- Comparing file size in the Transfer System and EOS.
- Reporting problem to Storage Manager Operations HN (SMOps)
- They report back they were changing the procedure to check files consistency
- In the meantime the only thing we could do was:
 - ◆ Tracing the problematic files
 - ◆ Reporting them to SMOps
 - ◆ Failing the related jobs

90 - Unable to find requested service (DQM / DQMIO)

Example Report

<https://cms-logbook.cern.ch/elog/Tier-0+processing/12755>

Error message

```
14-May-2015 13:45:40 CEST Successfully opened file root://eoscms.cern.ch//store/data/Commissioni
%MSG-w XrdAdaptor: (NoModuleName) 14-May-2015 13:45:49 CEST pre-events
Data is now served from cern.ch instead of previous
%MSG
----- Begin Fatal Exception 14-May-2015 13:45:54 CEST-----
An exception of category 'NotFound' occurred while
  [0] Calling beginJob for module DQMRootOutputModule/'write_DQMIO'
Exception Message:
Service Request unable to find requested service with compiler type name ' 8DQMStore'.
----- End Fatal Exception -----
```

Procedure used

- We reported to Data Quality Monitoring Development (DQMDev).
- The AlcaReco matrix was updated a couple of days before, the problem was tracked down to a specific Alca Producer.
- We disabled PromptReco for AlcaLumipixels.
- We failed the problematic jobs.

\$manage execute-agent wmagent-unregister-wmstats vocms062.cern.ch:9999

Helpful commands for AFS:

Command:	Effect:	Arguments:
tokens	show all AFS tokens the user currently holds	
klog	to obtain an AFS tokens different from the one established upon login	[] : username to be used for token
unlog	destroy all AFS tokens	
fs whereis	show on which afs server a file/directory is saved on	[] or [] : file or directory to search on AFS or \$HOME : position of home directory on AFS
fs checks	check which afs servers are running	
fs listquota	show your quota on AFS	
fs listacl	show AFS Access Control List (ACL)	[] : show ACL of a directory
fs setacl	set AFS Access Control List (ACL)	-dir : set ACL of path -acl : set access right for login access rights: r : reading of files w : writing to files k : locking of files (flock() system call) l : the right to search for files in the directory, i.e. list which files it contains i : creation of new files or directories d : deletion of files a : the right to change the ACL all : r+w+k+l+i+d+a none : delete entry from ACL read : r+l write : r+w+k+l+i+d

Helpful commands for LSF:

Command:	Effect:	Arguments:
bqueues	show list and usage of all LSF queues	
bjobs	show status of all submitted LSF jobs	[-1] <jobid> : job id number of the LSF job
bsub	submit a batch job to LSF	[-q <queuename>] : start job in specific LSF queue [-J <jobname>;>] : give job specific name [-o <file>] : use specific output file <command> [<arguments>] : program/script to be started on LSF with arguments
bkill	kill a LSF job	jobid : job id number of the LSF job (see bjobs)
bpeek	displays the stdout and stderr output of an unfinished LSF job	[-f] : use command <i>tail -f</i> to show output [-q <queuename>] : show output of most recently submitted LSF job in the specified queue [-m <hostname>] : show output of most recently submitted LSF job executed on the specified host [-J <jobname>] : show output of most recently submitted LSF job with specified job name

[<jobid>] : operates on the specified job

Helpful commands for CASTOR[↗]:

Command:	Effect:	Arguments:
nsls	list CASTOR file or directory	[<file>] or [<directory>] : file or directory on CASTOR to list or \$CASTOR_HOME : list users CASTOR home directory [--class [<classid>]] : to show file class [-R] : recursively [-l] : long listing [-T] : show tape number of file [--deleted] : show deleted files
nslstclass	show attributes for CASTOR classes	[--id <classid>] : CASTOR file class id
nsmkdir	create a new CASTOR directory	<directory> : file or directory on CASTOR to create
rfdir	list CASTOR file or directory	<file> or <directory> : file or directory on CASTOR to list or \$CASTOR_HOME : list users CASTOR home directory [-R] : recursively
rfrm	remove CASTOR file or directory	<file> or <directory> : file or directory on CASTOR to delete [-r] : recursively
rfrename	rename CASTOR file or directory	<oldfilename> <newfilename> : rename file from oldfilename to newfilename or <olddirectoryname> <newdirectoryname> : rename directory from olddirectoryname to newdirectoryname
rfchmod	set CASTOR file or directory permissions	<accessright> : access right to set <i>access rights:</i> + : add access righth - : remove access right u : user who owns the file g : other users in file's group o : other users not in the file's group a : all users r : reading of files or directories w : writing to files or directories x : executing of files or accessing of directories <file> or <directory>: file or directory name to change access rights
rfcp	copy local file to CASTOR or from CASTOR to local file	<localfilename> <castorfilename> : copy file from localfilename to CASTOR or <castorfilename> <localfilename> : copy file from CASTOR to localfilename
stager_get	stage file(s) to disk	-M <castorfilename> [-M <castofilename> ...] : file(s) on CASTOR to stage [--nowait] : do not wait till staging finished
stager_qry	view status of file on staging disk	-M <castorfilename> : file on CASTOR staging disk

Helpful linux commands:

Command:	Effect:	Arguments:
<code>find . -exec grep -H <pattern> {} \;</code>	searches a pattern in all files in the directory and all its subdirectories	<code><pattern></code> : pattern to search
See folder permissions as a number	<code>stat -c %a /path/to/file</code>	



This topic: [Sandbox > JohnHarveyCasallasLeonSandbox](#)

Topic revision: r58 - 2017-07-10 - JohnHarveyCasallasLeon



Copyright &© 2008-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

or Ideas, requests, problems regarding TWiki? use [Discourse](#) or [Send feedback](#)