



Enabling Grids for E-scienceE

# Critical services: FTS

*Gavin McCance*

*WLCG Service Reliability Workshop, CERN*

*November 2007*

[www.eu-egee.org](http://www.eu-egee.org)



Information Society  
and Media



- **FTS overview**
- **Understanding service dependencies**
- **Software and HA deployment**
- **Procedures for robustness**
- **Running the Grid service**

- **The File Transfer Service (FTS) is a data movement grid fabric service**
  - (i.e. it copies files around)
  - It is a multi-VO service, used to balance usage of site resources according to VO and site policies
  - The starting assumption is that all these data streams are such high volume that they need to be *managed*
    - Application level prioritisation (between VOs, within a VO)
    - Prevent network overload
    - Prevent storage overload
    - Monitor and understand problems on the service
  - **The focus is on the “service”**
    - The purpose of the FTS is to make it easy to do these things well
    - With a sustainable amount of effort
      - *Even a 0.1% failure is a large amount of data to manually ‘rescue’ when you’re transferring petabytes!*

- **Two aspect to the ‘Transfer service’**
- **Local fabric: the underlying machines, software and fabric-level operations running the service have to be there**
  - **If the base service isn’t reliable, the overall service can’t be either**
- **‘Grid service’ or the 1<sup>st</sup> rule of distributed computing**
  - **We know things (something, somewhere) will always be going wrong in a distributed environment**
  - **The purpose of a ‘grid service’ is to limit the impact to the overall service of any problems with anything it depends on**

- **How do we make local service reliable?**
  - Software
    - Design for HA - talk more on Thursday about this for FTS
  - Hardware and deployment
    - Deployment for HA
    - Critical analysis of service dependencies
  - Procedures and follow-up
    - Operator procedures
    - Operations procedures – critical and planned
    - Intervention plans and follow-up
    - Documentation

- **Understand the internal dependencies**
  - Make and go through the service checklist
- **This was done together with CERN-IT/FIO 18 months ago as we were setting up the initial service for service challenge 3**
  - ....and was reviewed recently as part of the overall drive for reliable WLCG services
- **Main point: Need to keep these things in review!!**
  - A lot of things had changed – things we thought were covered, weren't in fact covered when we looked again!
  - Some things were carefully flagged and then carefully forgotten
  - Some things we didn't initially consider

- The (rather abbreviated) service checklist

- **Hardware:**

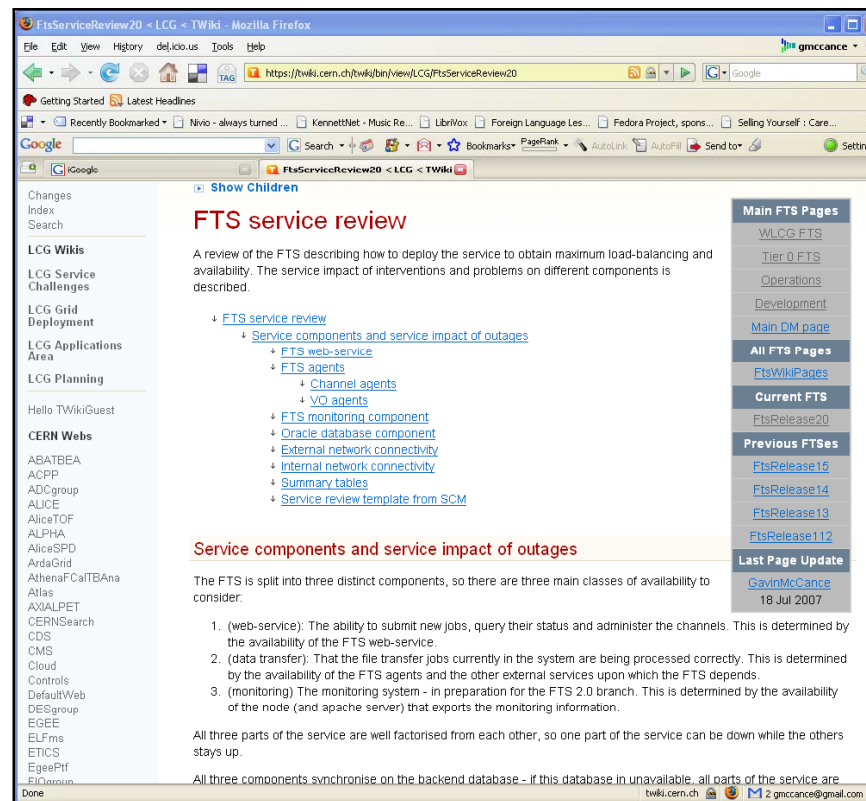
- Decent quality: raid disks,
- Requirements understood (memory, disk use)
- Inventory: what runs where?

- **Operations:**

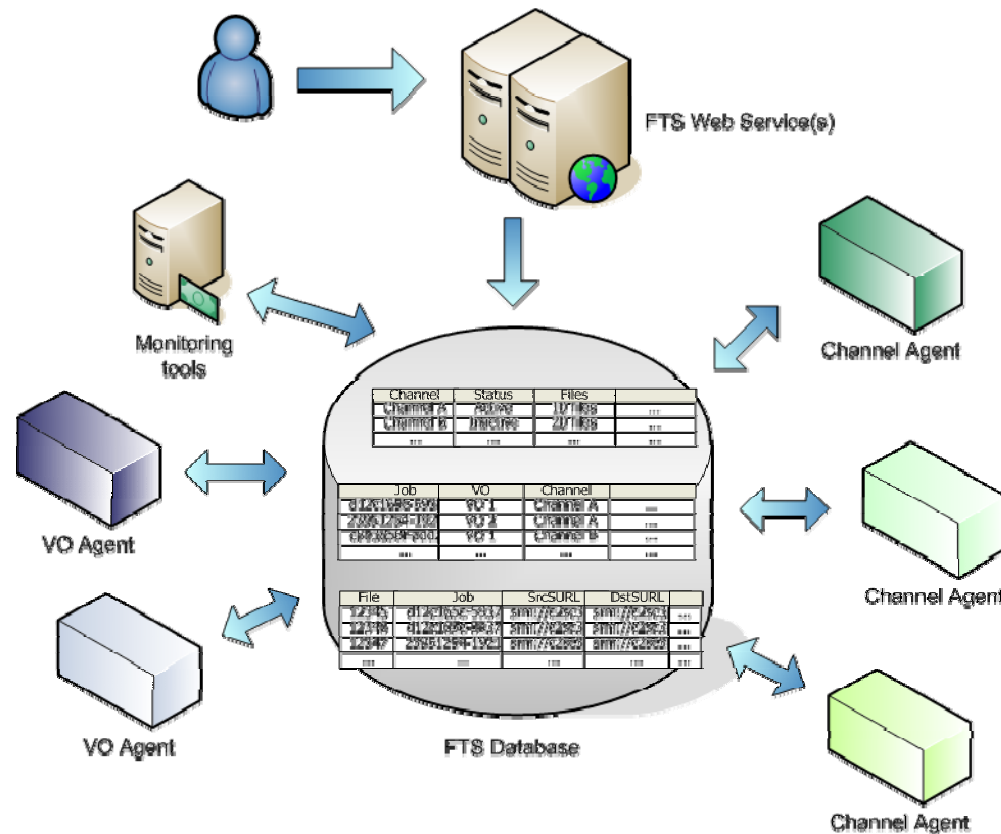
- Sufficient monitoring to catch the most obvious service failures at the operator level (daemon failure, say)
- Operator / sysadmin procedures
  - Basic start/stop/drain documented
- Service manager procedures and problem determination guide
- DB backup procedures defined

Hardware	
Assess	Question to set assessment color if not available
Green	CPU power required for application: mid-range server machine OK
Green	Memory requirement for application: medium-to-high memory requirement for FTS web-service and channel agents (2-4GB), low-memory for VO agents (< 2GB)
Green	Inventory of hardware components defined v: Specified in <a href="#">FtsWlcg</a>
Green	Machines available: fts10[1-9]
Green	Machines installed in appropriate location within data centre (UPS_Network): Installed in LCG network
Green	Database server and disk space allocated: Database backend on LCG Oracle RAC Cluster
<input type="button" value="Edit"/>	
Operations	
Assess	Question to set assessment color if not available
Green	2nd level support organisation defined (who to call when there is a problem with the application or middleware): Support teams defined in GGUS
Green	Mechanism to contact 2nd level organisation: GGUS
Yellow	Response time for 2nd level organisation: Best Effort
Green	List of machines where service is running defined: See <a href="#">FtsWlcg#Overview</a>
Yellow	List of configuration parameters and their values for the software components: still held in AFS rather than ncm. File: <code>/afs/cern.ch/project/gd/SC3/SC4/yaim/agents-split-between-fts101-and-fts102/site-info.def.ncm</code> in progress.
Green	List of processes to monitor: Defined (tomcat5, bdii, glite-transfer agent daemons).
Green	List of file systems and their emergency thresholds for alarms: Defined (standard)
Green	Application status check script requirements defined: Integrated into LEMON
Green	Definition of scheduled processes (e.g. cron): Managed by QUATTOR and YAIM
Green	Test environment defined and available: subcluster machines fts00[1-5] allocated.
Green	Problem determination procedures including how to determine application vs middleware vs database issues: Operator and Sysadmin procedures in OPM
Green	Procedures for start/stop/drain/check status defined: in <a href="#">FtsServerAdmin15</a> and <a href="#">FtsProcedures15</a> .
Green	Automatic monitoring of the application in place: FTS web-service and FTA agents integrated into LEMON.
Green	DB backup procedures defined in <a href="#">BackupPolicyManagers</a>

- An 'FTS installation' has several components to it
- It's good to review all the components and ask 'what's the impact to the overall service if this bit fails'
  - <https://twiki.cern.ch/twiki/bin/view/LCG/FtsServiceReview20>
  - ...and then say (as a developer or service manager) – if the answer is 'the whole thing stops' what can I do to fix that?
- Review some of these here, highlighting the techniques







Experiments interact via web-service

VO agents do VO-specific operations (1 per VO)

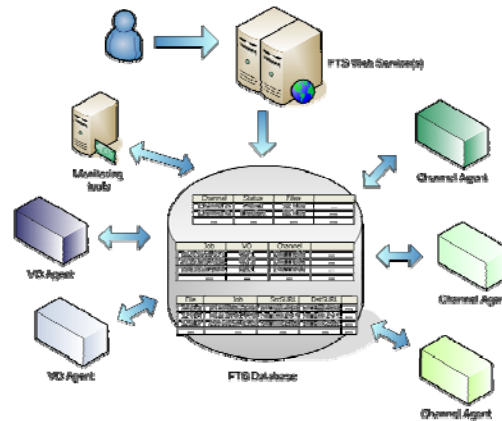
Channel agents to channel specific operation (e.g. the transfers)

Monitoring and statistics can be collected via the DB

- **Key: all components are decoupled from each other**
  - A failure of one ~doesn't impact the others

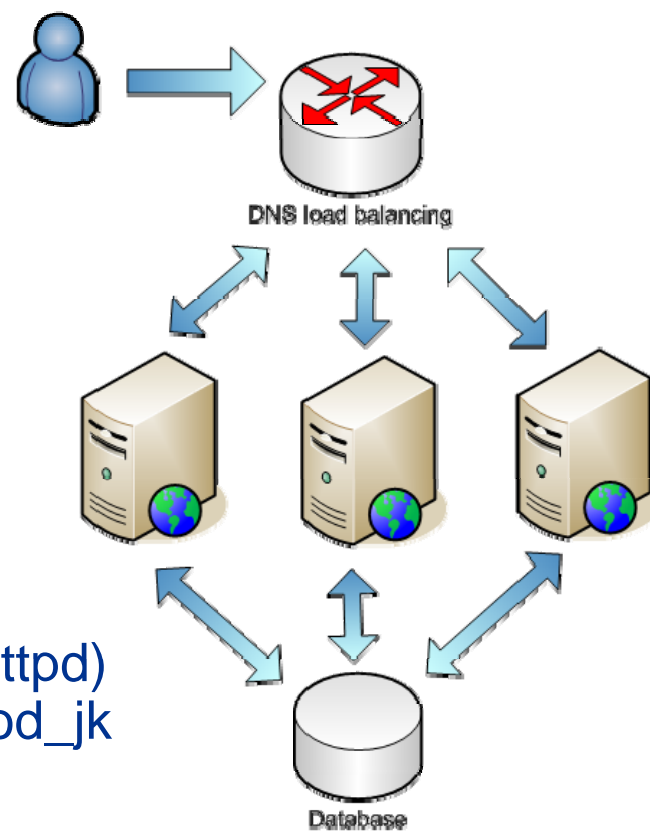
# Deployment architecture: decoupled

- For any serious production-scale system you should match the decoupled nature of the software architecture to its deployment



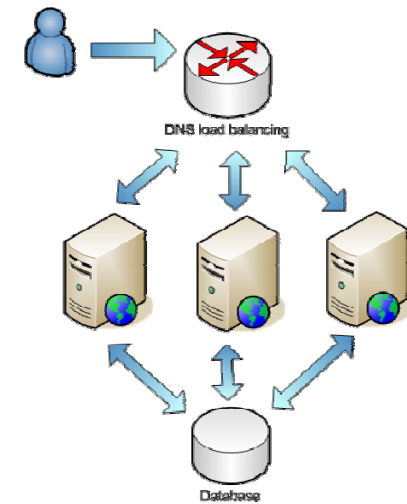
- This limits the impact to the service of individual node failures
  - The web-services are put on separate nodes
  - The VO agents are put together on a separate node(s)
  - The transfer agents are split over multiple nodes
  - The Oracle database should be on a RAC (and on separate nodes!)
  - The monitoring tools and web-server should be on a separate node

- **The web-service is stateless and is designed to be load-balanced for High-Availability**
  - (and to make it scale, as necessary)
  - Currently DNS load-balancing is used at CERN
    - No session state is maintained
    - (currently over 3 mid-range servers for CERN-PROD T0-export)
  - **Advantage: you can intervene on a node without perturbing service since the other nodes are available to handle the requests**
  - Additional possible solutions: apache (httpd) based load-balancing front-tier using mod\_jk
    - handles node interventions better
    - at the expense of an extra 'tier'



- **Our recent service review uncovered an unfortunate issue!**

- All our carefully load-balanced web-services were on one switch
- A problem on that switch means the whole service is down



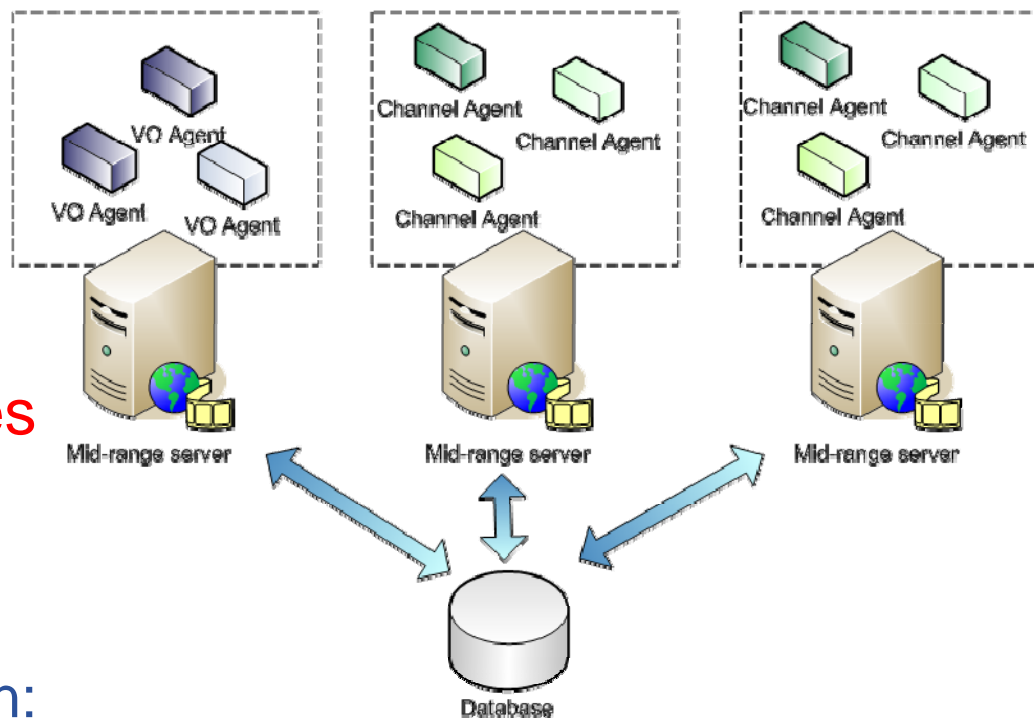
- **The network deployment is a key part of the service review**

- Understand the impact to the service of a network component outage

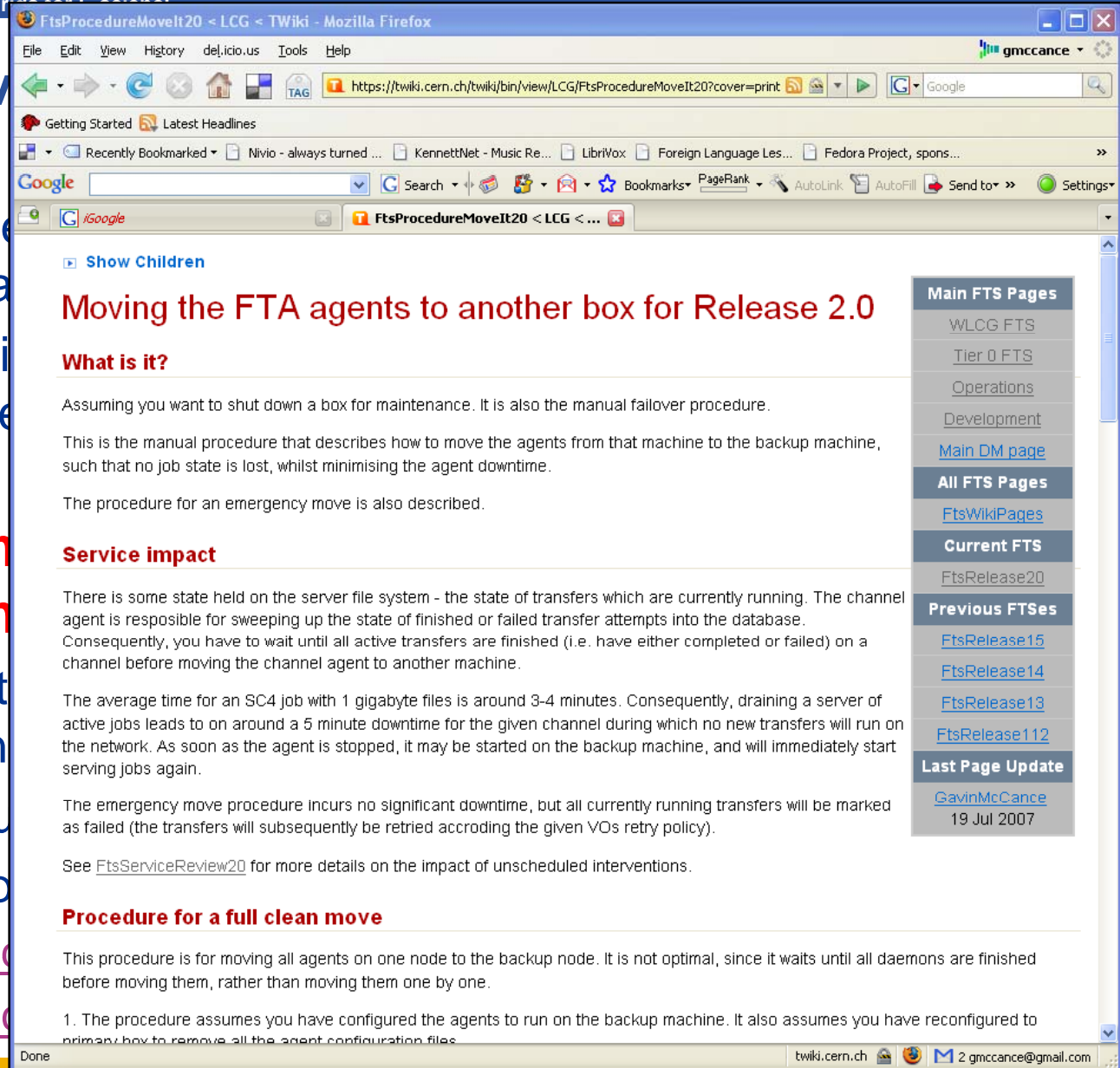
- The configuration allows the agents daemons to be split arbitrarily over server nodes

– Shares load over multiple nodes

- Limit impact of outages
- One agent down: no impact on the others
  - One node goes down: no impact on channels running on other nodes



- **Careful observation of hardware spares'**
  - i.e. the fail-over agents is a major concern
  - This could be improved by better software design, but this is expensive!
- **In the meantime, the software is not cloneable using hardware spares**
  - The configuration is not cloneable using hardware spares
  - Make sure you have a backup of the configuration
  - See for example:
    - <https://twiki.cern.ch/twiki/bin/view/LCG/FTSProcedureMoveIt20?cover=print>
    - <https://twiki.cern.ch/twiki/bin/view/LCG/FTSProcedureMoveIt20?cover=print>



The screenshot shows a Mozilla Firefox browser window with the address bar displaying <https://twiki.cern.ch/twiki/bin/view/LCG/FTSProcedureMoveIt20?cover=print>. The page content includes:

- Show Children**
- Moving the FTA agents to another box for Release 2.0**
- What is it?**

Assuming you want to shut down a box for maintenance. It is also the manual failover procedure.

This is the manual procedure that describes how to move the agents from that machine to the backup machine, such that no job state is lost, whilst minimising the agent downtime.

The procedure for an emergency move is also described.
- Service impact**

There is some state held on the server file system - the state of transfers which are currently running. The channel agent is responsible for sweeping up the state of finished or failed transfer attempts into the database. Consequently, you have to wait until all active transfers are finished (i.e. have either completed or failed) on a channel before moving the channel agent to another machine.

The average time for an SC4 job with 1 gigabyte files is around 3-4 minutes. Consequently, draining a server of active jobs leads to on around a 5 minute downtime for the given channel during which no new transfers will run on the network. As soon as the agent is stopped, it may be started on the backup machine, and will immediately start serving jobs again.

The emergency move procedure incurs no significant downtime, but all currently running transfers will be marked as failed (the transfers will subsequently be retried according to the given VOs retry policy).

See [FtsServiceReview20](#) for more details on the impact of unscheduled interventions.
- Procedure for a full clean move**

This procedure is for moving all agents on one node to the backup node. It is not optimal, since it waits until all daemons are finished before moving them, rather than moving them one by one.

  1. The procedure assumes you have configured the agents to run on the backup machine. It also assumes you have reconfigured to primary box to remove all the agent configuration files.

On the right side of the page, there is a sidebar with navigation links:

- Main FTS Pages**
  - WLCG FTS
  - Tier 0 FTS
  - Operations
  - Development
  - [Main DM page](#)
- All FTS Pages**
  - [FtsWikiPages](#)
- Current FTS**
  - [FtsRelease20](#)
- Previous FTSES**
  - [FtsRelease15](#)
  - [FtsRelease14](#)
  - [FtsRelease13](#)
  - [FtsRelease112](#)
- Last Page Update**
  - [GavinMcCance](#)
  - 19 Jul 2007

- **Procedures are the critical in operations**
- **Different types of procedure**
- **24/7 operator procedures (e.g.)**
  - These react to alarms and have
    - FTA\_WRONG: detected one or more failures
    - Procedure: restart them using the appropriate procedure
    - Action: if the alarm doesn't go away, open a standard ticket
- **Service manager procedures**
  - Incident response – see previous one (WhatToDoWhen)
  - Planned procedures
    - Scheduled hardware moves, kernel upgrades
  - Procedures should always try to use software / deployment features to minimise the impact to the service

```

NO_CONTACT
SWAP_FULL
TMP_FULL
VAR_FULL
ROOT_FS_FULL

    • Open a standard ticket

GRID_BDII_WRONG

    • Log onto the node as root
    • Restart the BDII daemon: /sbin/service bdii restart
    • If the alarm does not clear in 10 minutes or if the restart fails, open a standard ticket.
    • If the alarm does clear OK, make a log-only entry.

TOMCAT_WRONG

    • Log onto the node as root
    • Restart the Tomcat daemon: /sbin/service tomcat5 restart
    • If the alarm does not clear in 10 minutes or if the restart fails, open a standard ticket.
    • If the alarm does clear OK, make a log-only entry.
    • Regardless of whether the alarm cleared or not, always send mail to grid-cern-prod-dms@cern.ch

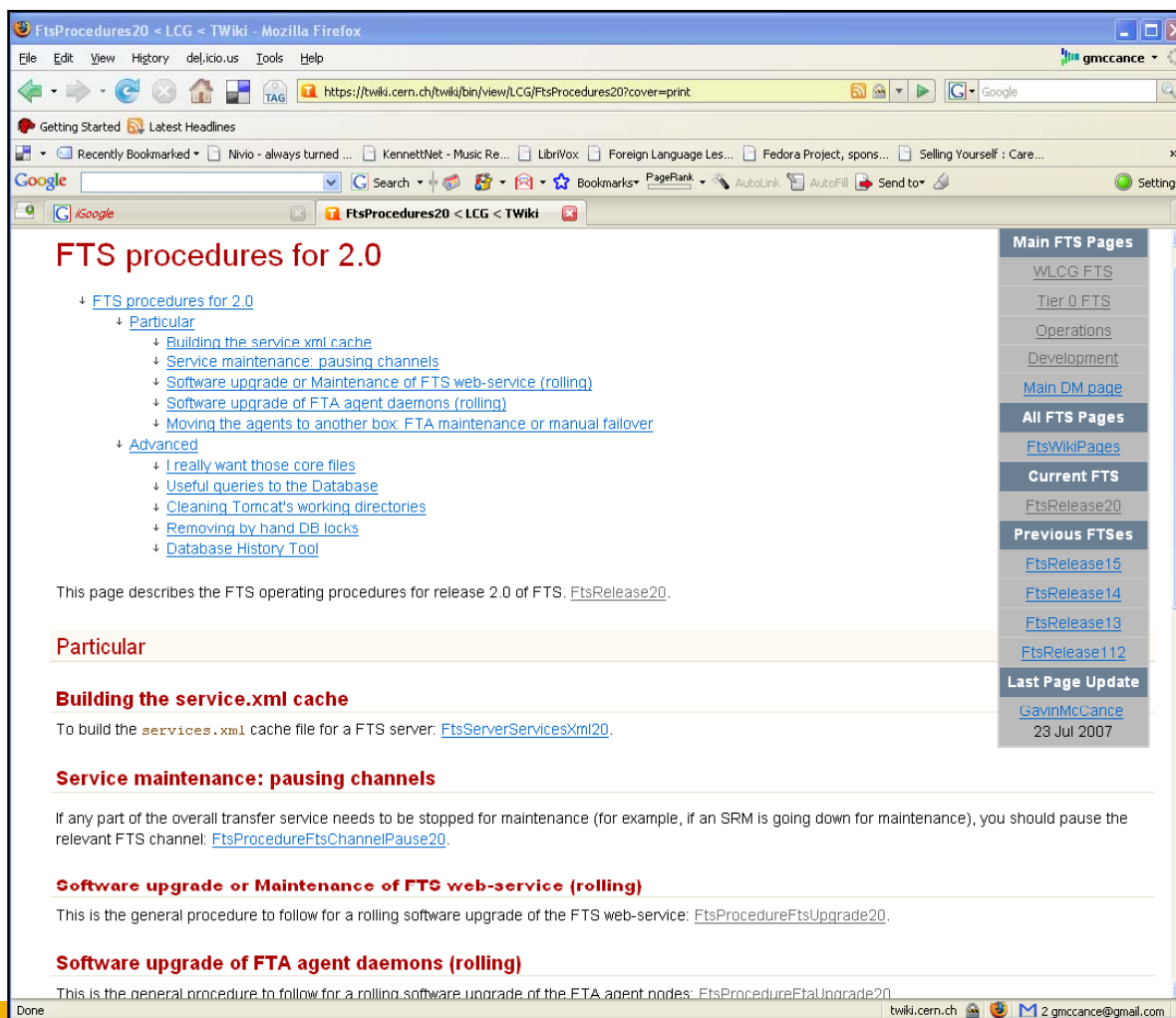
FTS_STUCK

    • Log onto the node as root
    • Restart the Tomcat daemon: /sbin/service tomcat5 restart
    • If the alarm does not clear in 10 minutes or if the restart fails, open a standard ticket.
    • If the alarm does clear OK, make a log-only entry.
    • Regardless of whether the alarm cleared or not, always send mail to grid-cern-prod-dms@cern.ch

FTA_WRONG

    • Log onto the node as root
    • The alarm monitors multiple daemons. Determine which one is wrong:
      o cat /var/lib/fts-agent-alarm/fts-agent.alarm
    • If the file is empty, and the alarm has not cleared, open a standard ticket.
    • Otherwise, you should see a line or lines like "glite-transfer-agent-name BAD" where
      "glite-transfer-agent-name" is the name of the daemon which is down.
    • For every BAD line listed, attempt to restart that daemon, using the full name of it as the instance. The
      exact name depends on the agent type, but will always be as it is printed in the BAD line. For example:
      o service transfer-agents start --instance
  
```

- Service manager procedures
- <https://twiki.cern.ch/twiki/bin/view/LCG/FtsProcedures20>



**FtsProcedures20 < LCG < TWiki** - Mozilla Firefox

File Edit View History del.icio.us Tools Help

gmccance

https://twiki.cern.ch/twiki/bin/view/LCG/FtsProcedures20?cover=print

Getting Started Latest Headlines

Recently Bookmarked Nivio - always turned ... KennettNet - Music Re... LibriVox Foreign Language Les... Fedora Project, spons... Selling Yourself : Care...

Google Search

**FtsProcedures20 < LCG < TWiki**

## Fts procedures for 2.0

- ↓ [Fts procedures for 2.0](#)
  - ↓ [Particular](#)
    - ↓ [Building the service.xml cache](#)
    - ↓ [Service maintenance: pausing channels](#)
    - ↓ [Software upgrade or Maintenance of FTS web-service \(rolling\)](#)
    - ↓ [Software upgrade of FTA agent daemons \(rolling\)](#)
    - ↓ [Moving the agents to another box: FTA maintenance or manual failover](#)
  - ↓ [Advanced](#)
    - ↓ [I really want those core files](#)
    - ↓ [Useful queries to the Database](#)
    - ↓ [Cleaning Tomcat's working directories](#)
    - ↓ [Removing by hand DB locks](#)
    - ↓ [Database History Tool](#)

This page describes the FTS operating procedures for release 2.0 of FTS: [FtsRelease20](#).

### Particular

#### Building the service.xml cache

To build the `services.xml` cache file for a FTS server: [FtsServerServicesXml20](#).

#### Service maintenance: pausing channels

If any part of the overall transfer service needs to be stopped for maintenance (for example, if an SRM is going down for maintenance), you should pause the relevant FTS channel: [FtsProcedureFtsChannelPause20](#).

#### Software upgrade or Maintenance of FTS web-service (rolling)

This is the general procedure to follow for a rolling software upgrade of the FTS web-service: [FtsProcedureFtsUpgrade20](#).

#### Software upgrade of FTA agent daemons (rolling)

This is the general procedure to follow for a rolling software upgrade of the FTA agent nodes: [FtsProcedureFtaUpgrade20](#).

**Main FTS Pages**

- [WLCG FTS](#)
- [Tier 0 FTS](#)
- [Operations](#)
- [Development](#)
- [Main DM page](#)

**All FTS Pages**

- [FtsWikiPages](#)

**Current FTS**

- [FtsRelease20](#)

**Previous FTses**

- [FtsRelease15](#)
- [FtsRelease14](#)
- [FtsRelease13](#)
- [FtsRelease112](#)

**Last Page Update**

- [GavinMcCance](#)
- 23 Jul 2007

Done twiki.cern.ch gmccance@gmail.com



- We always have
- Some examples
  - <https://twiki.cern.ch>
- Need to understand
  - Work out best possible
  - Know people manager, our
  - We appoint a
  - Include the a
  - Document the
    - e.g. [FTS] to your DE so you can

## Upgrade of production tier-0 export to FTS 2.0

### Scope:

- The production T1 export service and production T2<->T1 service
- The tier-2 production service will not be upgraded at this point.
- The pilot service is already running FTS 2.0.

### Intervention announcement...

### Scope:

- Production tier-0 export service
- Production tier-2 service

### Preparation steps:

- Verify that the FTA agent actuator is disabled when the nodes are in maintenance. **VERIFIED**
- Only two CDB templates need updating `pro_system_gridfts` and `pro_type_gridfts_slc3`. These are now in `~straylen/fts-upgrade` level.
- The primary schema upgrade script is in the `transfer-fts` FTS 2.0 RPM: `/opt/glite/etc/glite-data-transfer-fts/schema/oracle/oracle-upgrade_2.2.1-3.0.0.sql`
- The history schema upgrade script is in `/afs/cern.ch/user/m/mccance/public/fts20-upgrade-intervention/fts_history`

### Migration steps:

- Switch all channels to inactive. **DONE**
- Go to coffee while they drain currently running transfers. **DONE**
- Put all production nodes in maintenance. **DONE**
- There are three DBMS user jobs running: stop them (SQL\*Plus on `lcg_fts_prod`):
  - `exec fts_stats.stop_hourly_job;` **DONE**
  - `exec fts_history.stop_job;` **DONE**
  - `exec fts_statcount.stop_job;` **DONE**
  - Verify that `select * from user_jobs;` returns no rows. **DONE**
- Stop the web-services (`fts101, fts114, fts115`). **DONE**
- Stop the agent daemons (`fts110, fts111, fts112, fts113`). **DONE**
- Stop the multitude of little scripts running on the FTS monitoring node (`fts102`). **DONE** Move to `/cron.d/`
- Ask DB team (contact Miguel Anjo) to copy the partial schema to the backup account. This should take around 20 minutes. **DONE**
- ... [upgrade software] **DONE**
- ... [upgrade CDB yaim configuration for FTS2.0]. Backup the old one. **DONE**
- BACKOUT 1
- Upgrade the main schema (this should take around 2 minutes) **DONE**
- Upgrade the history schema (this should take around 20 minutes) **DONE**
- Load the delegation schema (YAIM will insist anyway). **DONE**
- Run the writer account script to build new synonyms and make the appropriate grants: [FtsServer20WriterAccount](#). **DONE**
- BACKOUT 2

### Cleanup:

- Restart the web-services (`fts114, fts115`). **DONE**
  - Test a few commands. **DONE**
- Restart the agent daemons (`fts110, fts111, fts112, fts113`). **DONE**
- Restart the monitoring scripts on `fts102`.
- Re-enable jobs:

plete

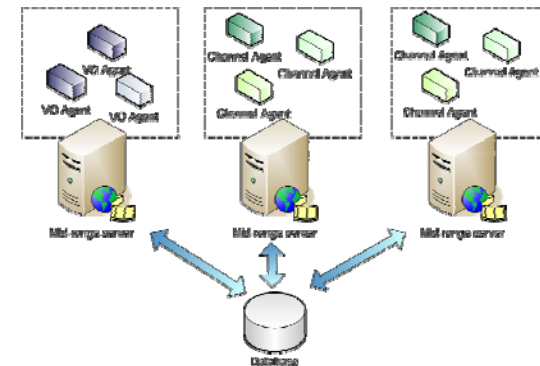
ho

rice

Talk

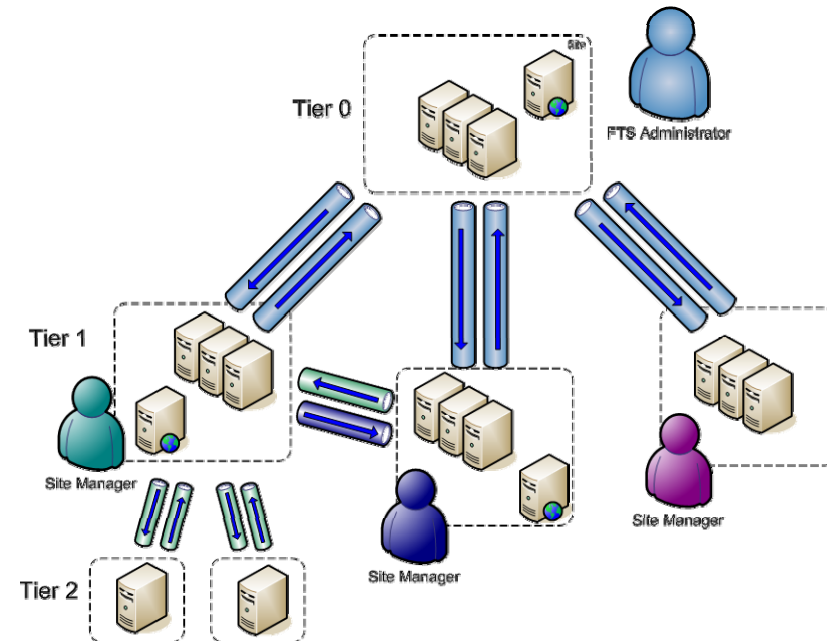
de

- We're working on our plan for SL4 upgrade on CERN-PROD
  - [ Pre-certification software being piloted (btw) ]
- **The plan is to do this with zero user-visible service downtime and minimal downtime on the FTS channels**
- Schema upgrade shouldn't be needed (we already did it in FTS2.0)
- Pre-prepare (and test!) elfms Quattor configuration
- Take each FTS web-service node out of DNS in turn and upgrade it
  - No user-visible downtime to the service
- For each agent node, in turn:
  - Swap template, pause the channels on that node, take node down and rebuild it
  - The other nodes and channels will continue to run
  - The new node should come back up running
  - Restart channels
  - Each channel will experience a ~15 minute interruption



- **Finally, we always try to follow-up any interventions and incidents**
- **We do this on CERN-PROD within our regular operations meetings**
  - **We look for things that could have gone better – there’s usually something. e.g.**
    - We sometimes pick up channel configuration problems
    - Forgotten ‘workarounds’ (aka hacks) that bit you during the upgrade
    - The (open) issue of schema fragmentation was found during one of our interventions because it badly affected it
  - **We look for feedback to the developers**
    - “It’d be so much better if you...”

- **How do we make it reliable?**



- **Understand Grid service dependencies**
- **Distributed procedures to add value**
  - “The whole is greater than the sum of its parts”
  - The purpose of a ‘grid service’ is to limit the impact to the overall service of any problems with anything it depends on

[iGoogle](#)
[FtsServiceDependencies < LCG ...](#)

[Show Children](#)

## FTS service dependencies

This describes the service dependencies of FTS upon other WLCG services. It tries to describe the impact upon the FTS service of any problems in its dependent services.

Service	Activities	Ramification of service interruption	Criticality
Oracle database services	All components of the FTS synchronise on the database.	If the local DB is unavailable, all components of the FTS service are down.	Critical
SRM	The FTS transfers files between SRMs	If an SRM is down, the FTS will fail when attempting to transfer the file. FTS retries, but extended SRM downtime will be visible to the user as 'Failed' jobs.	Critical to overall service.
MyProxy server in <i>authorised retriever</i> mode ('special FTS' MyProxy setup)	Required for FTS 1.5 and FTS 2.0 (in legacy mode) to retrieve the user proxy needed for the transfer	A user proxy is retrieved only once (and retrieved again once it expires 12 hours later). Users already running jobs will have a proxy on the FTS system and can survive until that local proxy expires. New users' jobs will fail immediately.	Critical for FTS 1.5 and for FTS 2.0 running in legacy mode
MyProxy server in <i>authorised renewer</i> mode (MyProxy as used by WMS)	Used by FTS 2.0 (in the new delegation mode) to renew an expired credential. However, the original credential is refreshed (by delegation) on every new job submission (if the credential is due to expire), so it is expected that the FTS 2.0 server will contact MyProxy only infrequently.	Users who are not continually submitting into the system may have some of their jobs fail if the credential expires (while the job in in the queue) and cannot be renewed.	Not critical
VOMS	Used by FTS 2.0 at the same point as MyProxy (upon renewal of an expired credential). Same issues apply: the credential is refreshed whenever a new job is submitted into the system (if needed)	The same as MyProxy server	Not critical
BDII information service - FTS server	Not used directly by the server. The SRM endpoints are currently cached by the <code>make-services.xml</code> tool.	None	Not critical
BDII information service - FTS client	Discovery tool uses BDII to find the relevant FTS server.	Minimum: the correct server is defined by WLCG policy. Most experiments hardcode the relevant endpoint.	Not critical

-- [GavinMcCance](#) - 02 Jul 2007

Resisting all temptations for a Grid Architecture diagram with lots of boxes and arrows

- **Some things to do to improve the overall service delivered to the users**
- **Procedures:**
  - e.g. Manage the channels that involve your site
    - ...even if the FTS server itself is on another site
- **Understanding overall service failures and how to detect these**
  - Failures of the FTS service (already talked about)
  - Failures of the underlying SRM / Storage services
    - Monitoring!

- **For tier-1 and tier-2 sites**
  - Bear in mind that a lot of the transfers will be coming from an FTS at another site
  - Know who the (other) FTS service administrators are
  - **Know which channels have been setup which involve your site**
  - Make sure that that your local admin(s) have been set as channel managers for these channels
- **You can pause these channels if needed**
  - **Set Inactive:** FTS jobs will be queued and resumed when you set the channel Active again
  - Avoids “queue” draining
  - It’s not possible for all channels involving your site (e.g. for the catch-all “\*” channels), but you can do it for a lot of them...

- **Scheduled: “I’m fixing / upgrading / etc my SRM”**

## **How to stop all transfers:**

- Know in advance which of your tier-1 channels affects you (and which you are able to control)
  - 10 to 15 minutes before the intervention:
    - Inform your tier-1 that you are pausing the channel(s)
    - `glite-transfer-channel-set -S Inactive CHANNELNAME`
  - After the intervention:
    - Inform your tier-1 that you are restarting the channel(s)
    - `glite-transfer-channel-set -S Active CHANNELNAME`
- **Unscheduled : “My SRM has just gone down”**
    - The same procedure as a scheduled intervention. If the SRM is down, set the channels that involve you directly to **Inactive**.



- **Degradation or failure of a service we depend on**
  - SRM / Storage problems
- **If the storage servers (SRM) go bad, the transfers will likely fail or be degraded**
  - This is by far the most common thing that causes transfer jobs to fail
  - Fabric-level HA for storage services is critical
- **The focus for the Grid FTS operations should be on the overall end-to-end service**
  - **Since the FTS measures all the transfers going through it, we use it as a debugging tool for storage related problems**
  - That's really what it's for...
  - Focus is now on monitoring for this purpose
    - See session on FTS service debugging tomorrow...



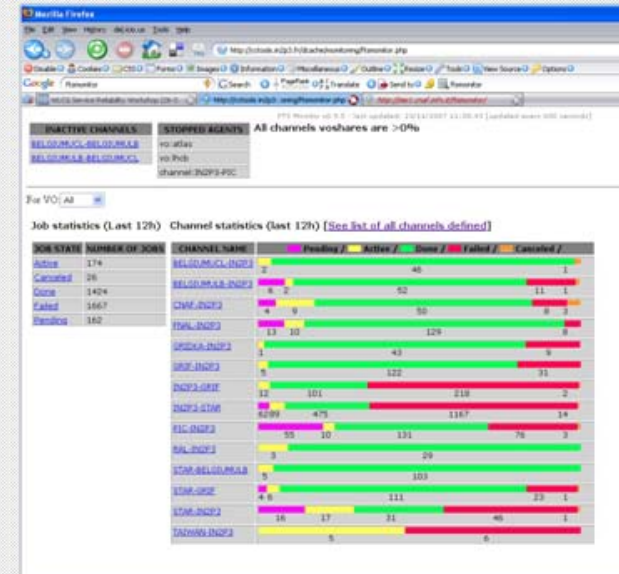
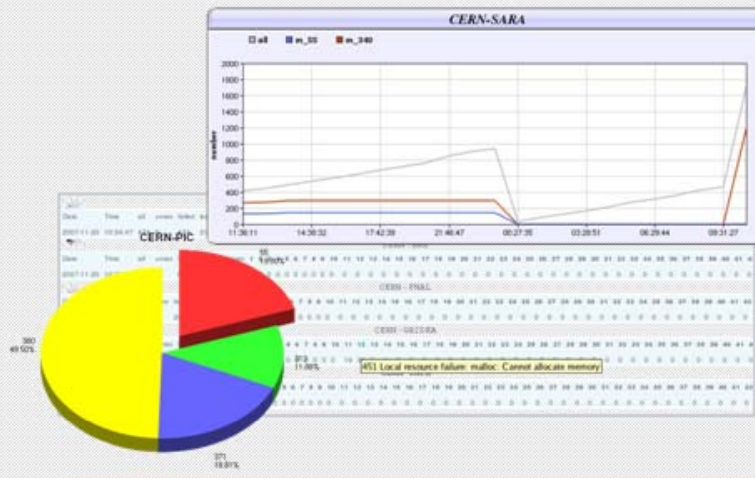
□ We  
un  
wh  
me

□ ...a  
to  
da

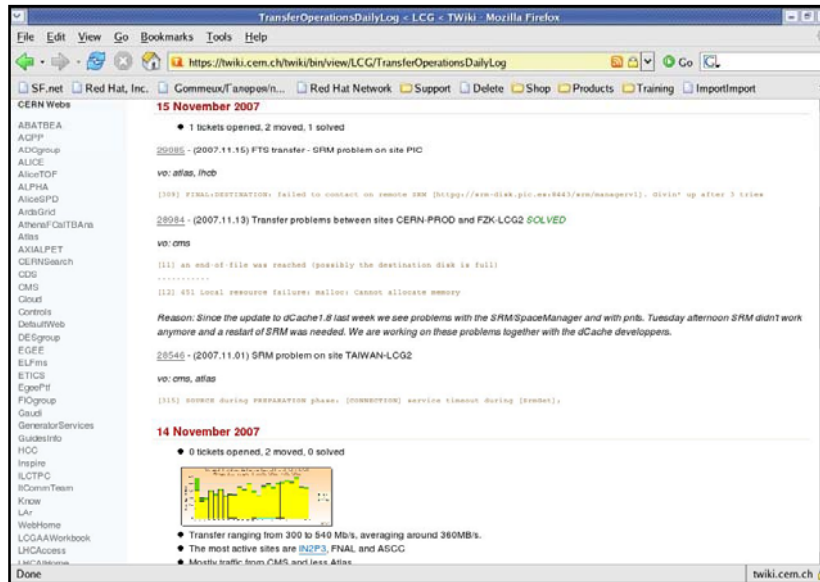


## Prototype tools

- And we have a set of tools developed (CERN + T1 sites) that are being used to run the operations currently



1. Daily log – tracking of current problems and open issues - GGUS
2. Daily log Archive – history of situation on channels from February '07 till present time
3. Weekly Report – summary report for the Joint Operations Meeting
4. Weekly Tier-0 – summary of issues noticed on the Castor Tier-0 service



TransferOperationsDailyLog - LCG - TWiki - Mozilla Firefox

https://twiki.cern.ch/twiki/bin/view/LCG/TransferOperationsDailyLog

**15 November 2007**

- 1 tickets opened, 2 moved, 1 solved

20085 - (2007.11.15) FTG transfer - SRM problem on site PIC  
 vv: atlas, thob

[309] FINAL-DESTINATION failed to contact on remote SRM [http://www-disk.pic.cern.ch/~srm/manager/]: Givin' up after 3 tries

20084 - (2007.11.13) Transfer problems between sites CERN-PROD and FZK-LCG2 SOLVED  
 vv: oms

[11] an end-of-file was reached (possibly the destination disk is full)  
 .....  
 [12] 491 Local resource failure: malloc: Cannot allocate memory

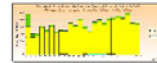
Reason: Since the update to uCache 1.8 last week we see problems with the SRM@spaceManager and with prb. Tuesday afternoon SRM didn't work anymore and a restart of SRM was needed. We are working on these problems together with the dCache developers.

20546 - (2007.11.01) SRM problem on site TAIWAN-LCG2  
 vv: oms, atlas

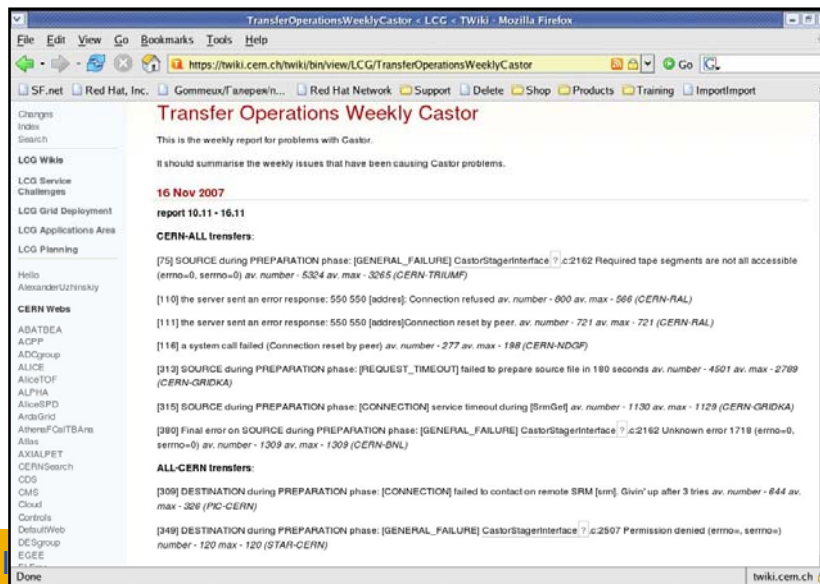
[315] SOURCE during PREPARATION phase: [CONNECTION] service timeout during [SRMGET]

**14 November 2007**

- 0 tickets opened, 2 moved, 0 solved



- Transfer ranging from 300 to 540 M/s, averaging around 380MB/s.
- The most active sites are IN2P3, FNAL and ASGC
- Mostly traffic from CMS and less Atlas.



TransferOperationsWeeklyCastor - LCG - TWiki - Mozilla Firefox

https://twiki.cern.ch/twiki/bin/view/LCG/TransferOperationsWeeklyCastor

**Transfer Operations Weekly Castor**

This is the weekly report for problems with Castor.

It should summarise the weekly issues that have been causing Castor problems.

**16 Nov 2007**

report 10.11 - 16.11

**CERN-ALL transfers:**

[75] SOURCE during PREPARATION phase: [GENERAL\_FAILURE] CastorStagerInterface?;c:2162 Required tape segments are not all accessible (errno=0, sermo=0) av. number - 5324 av. max - 3265 (CERN-TRIUMF)

[110] the server sent an error response: 550 550 [address] Connection refused av. number - 800 av. max - 566 (CERN-RAL)

[111] the server sent an error response: 550 550 [address] Connection reset by peer av. number - 721 av. max - 721 (CERN-RAL)

[118] a system call failed (Connection reset by peer) av. number - 277 av. max - 198 (CERN-NDGF)

[313] SOURCE during PREPARATION phase: [REQUEST\_TIMEOUT] failed to prepare source file in 180 seconds av. number - 4501 av. max - 2769 (CERN-GRIDKA)

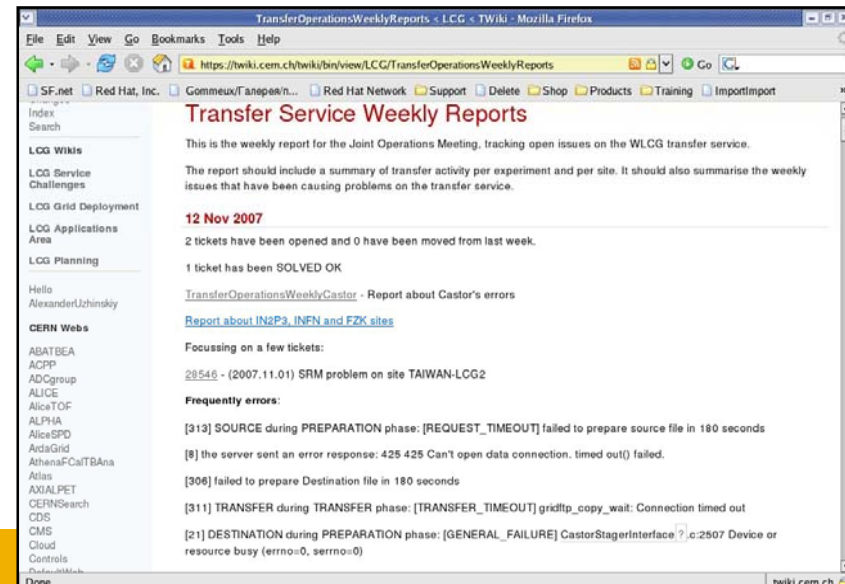
[315] SOURCE during PREPARATION phase: [CONNECTION] service timeout during [SRMGET] av. number - 1130 av. max - 1129 (CERN-GRIDKA)

[380] Final error on SOURCE during PREPARATION phase: [GENERAL\_FAILURE] CastorStagerInterface?;c:2162 Unknown error 1718 (errno=0, sermo=0) av. number - 1309 av. max - 1309 (CERN-DNL)

**ALL-CERN transfers:**

[309] DESTINATION during PREPARATION phase: [CONNECTION] failed to contact on remote SRM [rm]. Givin' up after 3 tries av. number - 844 av. max - 226 (PIC-CERN)

[349] DESTINATION during PREPARATION phase: [GENERAL\_FAILURE] CastorStagerInterface?;c:2507 Permission denied (errno=, sermo=) number - 120 max - 120 (STAR-CERN)



TransferOperationsWeeklyReports - LCG - TWiki - Mozilla Firefox

https://twiki.cern.ch/twiki/bin/view/LCG/TransferOperationsWeeklyReports

**Transfer Service Weekly Reports**

This is the weekly report for the Joint Operations Meeting, tracking open issues on the WLCG transfer service.

The report should include a summary of transfer activity per experiment and per site. It should also summarise the weekly issues that have been causing problems on the transfer service.

**12 Nov 2007**

2 tickets have been opened and 0 have been moved from last week.

1 ticket has been SOLVED OK

[TransferOperationsWeeklyCastor](#) - Report about Castor's errors

[Report about IN2P3, INFN and FZK sites](#)

Focussing on a few tickets:

20546 - (2007.11.01) SRM problem on site TAIWAN-LCG2

**Frequently errors:**

[313] SOURCE during PREPARATION phase: [REQUEST\_TIMEOUT] failed to prepare source file in 180 seconds

[8] the server sent an error response: 425 425 Can't open data connection, timed out() failed.

[306] failed to prepare Destination file in 180 seconds

[311] TRANSFER during TRANSFER phase: [TRANSFER\_TIMEOUT] gridftp\_copy\_wait: Connection timed out

[21] DESTINATION during PREPARATION phase: [GENERAL\_FAILURE] CastorStagerInterface?;c:2507 Device or resource busy (errno=0, sermo=0)

- **Two (inter-related) aspects to the FTS operations**
  - Fabric operations and Grid operations
  - ...but it's the same story
- **Understand the dependencies**
  - Ask what-if?
  - Affects the software design and service deployment
    - Given the software, match it in the deployment
- **Have robust operations procedures**
  - And review them regularly
  - Critical procedures and planned ones
  - Intervention plans: make use of the features to minimise the impact of an intervention
  - Grid (i.e. distributed) procedures can help a lot with the overall service quality