

Computing and Services

David Colling
with slides and input from several other people

Imperial College

October 19, 2012



Caveats



Caveats

- Covering a hodgepodge of different items not covered elsewhere



Caveats

- Covering a hodgepodge of different items not covered elsewhere
- Some new and some just never presented here before



Caveats

- Covering a hodgepodge of different items not covered elsewhere
- Some new and some just never presented here before
- Slides from a number of people (Ray Beuselinck, Janusz Martyniak, Antony Wilson, Chris Brew ...). Thanks to all.



Caveats

- Covering a hodgepodge of different items not covered elsewhere
- Some new and some just never presented here before
- Slides from a number of people (Ray Beuselinck, Janusz Martyniak, Antony Wilson, Chris Brew ...). Thanks to all.
- Apologies, only words and no pretty pictures



Caveats

- Covering a hodgepodge of different items not covered elsewhere
- Some new and some just never presented here before
- Slides from a number of people (Ray Beuselinck, Janusz Martyniak, Antony Wilson, Chris Brew ...). Thanks to all.
- Apologies, only words and no pretty pictures
- Second apologies, style caused by me playing around with “Beamer”



Caveats

- Covering a hodgepodge of different items not covered elsewhere
- Some new and some just never presented here before
- Slides from a number of people (Ray Beuselinck, Janusz Martyniak, Antony Wilson, Chris Brew ...). Thanks to all.
- Apologies, only words and no pretty pictures
- Second apologies, style caused by me playing around with “Beamer”
- Dan’s (excellent and insightful) questions allowing, should keep us on track for coffee!



So what is being covered?



So what is being covered?

- Some general news concerning PPD hosted machines



So what is being covered?

- Some general news concerning PPD hosted machines
- A description of the data moving and distribution system



So what is being covered?

- Some general news concerning PPD hosted machines
- A description of the data moving and distribution system
- New grid based reconstruction



General News from PPD hosted (Virtual) machines



General News from PPD hosted (Virtual) machines

- New VM infrastructure going from a few machines using Zen to a new hypervisor with iSCSI disks.



General News from PPD hosted (Virtual) machines

- New VM infrastructure going from a few machines using Zen to a new hypervisor with iSCSI disks.
- few teething problems, but they are understood and expected to be resolved on a timescale of a few weeks.



General News from PPD hosted (Virtual) machines

- New VM infrastructure going from a few machines using Zen to a new hypervisor with iSCSI disks.
- few teething problems, but they are understood and expected to be resolved on a timescale of a few weeks.
- All MICE VMs will be migrated as soon as stable



General News from PPD hosted (Virtual) machines

- New VM infrastructure going from a few machines using Zen to a new hypervisor with iSCSI disks.
- few teething problems, but they are understood and expected to be resolved on a timescale of a few weeks.
- All MICE VMs will be migrated as soon as stable
- Expect to provide more reliable and generally “better” infrastructure.



General News from PPD hosted (Virtual) machines

- New VM infrastructure going from a few machines using Zen to a new hypervisor with iSCSI disks.
- few teething problems, but they are understood and expected to be resolved on a timescale of a few weeks.
- All MICE VMs will be migrated as soon as stable
- Expect to provide more reliable and generally “better” infrastructure.
- Possible external monitoring of services



mousehole and network news



mousehole and network news

- Access to MICENET to be secured so only access via mousehole (and a small number specific machines such as ConfigDB) even from within RAL



mousehole and network news

- Access to MICENET to be secured so only access via mousehole (and a small number specific machines such as ConfigDB) even from within RAL
- (Probably) New network village manager



mousehole and network news

- Access to MICENET to be secured so only access via mousehole (and a small number specific machines such as ConfigDB) even from within RAL
- (Probably) New network village manager
- Mousehole to be changed to help move things in and out of MICENET by addition of scratch space.



ConfigDB update - provided by a Royal Marine somewhere
in a landing craft



ConfigDB update - provided by a Royal Marine somewhere in a landing craft

- Architecture described in a previous CM



ConfigDB update - provided by a Royal Marine somewhere in a landing craft

- Architecture described in a previous CM
- PPD element will be one of the first VMs migrated



ConfigDB update - provided by a Royal Marine somewhere in a landing craft

- Architecture described in a previous CM
- PPD element will be one of the first VMs migrated
- Period of evolution rather than revolution



ConfigDB update - provided by a Royal Marine somewhere in a landing craft

- Architecture described in a previous CM
- PPD element will be one of the first VMs migrated
- Period of evolution rather than revolution
- Example C++ tests with webservices (successful)



ConfigDB update - provided by a Royal Marine somewhere in a landing craft

- Architecture described in a previous CM
- PPD element will be one of the first VMs migrated
- Period of evolution rather than revolution
- Example C++ tests with webservices (successful)
- Half of said Royal Marine is **NOT** enough for smooth and uninterrupted development. Deputy needed



How data goes from MICE to the rest of the World



How data goes from MICE to the rest of the World

- Three components to system



How data goes from MICE to the rest of the World

- Three components to system
- Existed for sometime but as far as I could find out never described at a CM before



How data goes from MICE to the rest of the World

- Three components to system
- Existed for sometime but as far as I could find out never described at a CM before
- Data Mover takes data from MICE and puts it onto the Grid



How data goes from MICE to the rest of the World

- Three components to system
- Existed for sometime but as far as I could find out never described at a CM before
- Data Mover takes data from MICE and puts it onto the Grid
- Grid Download Agent downloads to other sites with Grid infrastructure (at the moment only Imperial unfortunately)



How data goes from MICE to the rest of the World

- Three components to system
- Existed for sometime but as far as I could find out never described at a CM before
- Data Mover takes data from MICE and puts it onto the Grid
- Grid Download Agent downloads to other sites with Grid infrastructure (at the moment only Imperial unfortunately)
- Imperial publishes all data over http at <http://www.hep.ph.ic.ac.uk/micedata/>



How data goes from MICE to the rest of the World

- Three components to system
- Existed for sometime but as far as I could find out never described at a CM before
- Data Mover takes data from MICE and puts it onto the Grid
- Grid Download Agent downloads to other sites with Grid infrastructure (at the moment only Imperial unfortunately)
- Imperial publishes all data over http at <http://www.hep.ph.ic.ac.uk/micedata/>
- Too much detail in some of the slides to come. I have included it for the record but will skip parts of it when talking.



The Data Mover - what it does



The Data Mover - what it does

- Check raw data file (tarball) integrity



The Data Mover - what it does

- Check raw data file (tarball) integrity
- Copy files from MICE control room to Castor at RAL (2 tapes)



The Data Mover - what it does

- Check raw data file (tarball) integrity
- Copy files from MICE control room to Castor at RAL (2 tapes)
- Replicate files from Castor to RAL PPD



Data Mover - Workflow Details



Data Mover - Workflow Details

- Copy a newly created file from /storage/for/MICE to an intermediate /scratch disk.



Data Mover - Workflow Details

- Copy a newly created file from `/storage/for/MICE` to an intermediate `/scratch` disk.
- DAQ has priority, so we obey the presence of the DAQ.writing semaphore.



Data Mover - Workflow Details

- Copy a newly created file from `/storage/for/MICE` to an intermediate `/scratch` disk.
- DAQ has priority, so we obey the presence of the DAQ.writing semaphore.
- Copy is triggered by a `.compact` semaphore set by a compactor (Henry N.)



Data Mover - Workflow Details

- Copy a newly created file from `/storage/for/MICE` to an intermediate `/scratch` disk.
- DAQ has priority, so we obey the presence of the DAQ.writing semaphore.
- Copy is triggered by a .compact semaphore set by a compactor (Henry N.)
- When on `/scratch` disk, the internal file integrity is checked (the tarball carries a list of checksums of all files present in it)



Data Mover - Workflow Details



Data Mover - Workflow Details

- A file copied to /scratch after integrity verification triggers a Castor Upload Agent



Data Mover - Workflow Details

- A file copied to /scratch after integrity verification triggers a Castor Upload Agent
- The Agent uploads the file to Castor



Data Mover - Workflow Details

- A file copied to /scratch after integrity verification triggers a Castor Upload Agent
- The Agent uploads the file to Castor
- Registers a Castor replica with the LFC



Data Mover - Workflow Details

- A file copied to /scratch after integrity verification triggers a Castor Upload Agent
- The Agent uploads the file to Castor
- Registers a Castor replica with the LFC
- If successful a replication to the PPD follows.



Data Mover - Workflow Details

- A file copied to /scratch after integrity verification triggers a Castor Upload Agent
- The Agent uploads the file to Castor
- Registers a Castor replica with the LFC
- If successful a replication to the PPD follows.
- The Adler checksum is verified at this stage (compared to the original checksum, bypassing Castor)



Data Mover - Workflow Details



Data Mover - Workflow Details

- The PPD replica is registered with the LFC



Data Mover - Workflow Details

- The PPD replica is registered with the LFC
- The file is registered with MICEMETA DB, so it can be used by other tools (GDA, batch reconstruction).



Data Mover - Workflow Details

- The PPD replica is registered with the LFC
- The file is registered with MICEMETA DB, so it can be used by other tools (GDA, batch reconstruction).
- At this stage the files is marked as uploaded and the workflow finishes.



Data Mover - Workflow Details

- The PPD replica is registered with the LFC
- The file is registered with MICEMETA DB, so it can be used by other tools (GDA, batch reconstruction).
- At this stage the files is marked as uploaded and the workflow finishes.
- The system can recover from most upload failures.



Data Mover Runs



Data Mover Runs

- Successfully used so far for all past runs



Data Mover Runs

- Successfully used so far for all past runs
- Discovered only a few corrupted files, some of them have been repaired and uploaded again



Data Mover Runs

- Successfully used so far for all past runs
- Discovered only a few corrupted files, some of them have been repaired and uploaded again
- Uploaded all files up to run 04168



Data Mover Runs

- Successfully used so far for all past runs
- Discovered only a few corrupted files, some of them have been repaired and uploaded again
- Uploaded all files up to run 04168
- Uploaded 1203 GB in total.



Data Mover Runs

- Successfully used so far for all past runs
- Discovered only a few corrupted files, some of them have been repaired and uploaded again
- Uploaded all files up to run 04168
- Uploaded 1203 GB in total.
- Last weekend data will be used to test the batch reconstruction system (see later)



Grid Download Agent (GDA)



Grid Download Agent (GDA)

- GDA is a python script designed as a simple download tool for Grid datasets. It has a simple command line interface and is suitable for running on an ad hoc basis or as a cron job to download newly available data automatically.



Grid Download Agent (GDA)

- GDA is a python script designed as a simple download tool for Grid datasets. It has a simple command line interface and is suitable for running on an ad hoc basis or as a cron job to download newly available data automatically.
- GDA works in conjunction with a separate uploader that copies new files to an initial Grid location, e.g. Castor at RAL, and logs available datasets in a Postgres database.



Grid Download Agent (GDA)

- GDA is a python script designed as a simple download tool for Grid datasets. It has a simple command line interface and is suitable for running on an ad hoc basis or as a cron job to download newly available data automatically.
- GDA works in conjunction with a separate uploader that copies new files to an initial Grid location, e.g. Castor at RAL, and logs available datasets in a Postgres database.
- This tool is not specific to any particular VO. There is a working implementation for the MICE VO.



Grid Download Agent (GDA)

- GDA is a python script designed as a simple download tool for Grid datasets. It has a simple command line interface and is suitable for running on an ad hoc basis or as a cron job to download newly available data automatically.
- GDA works in conjunction with a separate uploader that copies new files to an initial Grid location, e.g. Castor at RAL, and logs available datasets in a Postgres database.
- This tool is not specific to any particular VO. There is a working implementation for the MICE VO.
- In use at Imperial since July 2010.



GDA System Requirements



GDA System Requirements

- Python (at least version 2.4.3)



GDA System Requirements

- Python (at least version 2.4.3)
- Standard UI installation (grid commands available)



GDA System Requirements

- Python (at least version 2.4.3)
- Standard UI installation (grid commands available)
- Python modules pg and sqlite for database access



GDA System Requirements

- Python (at least version 2.4.3)
- Standard UI installation (grid commands available)
- Python modules pg and sqlite for database access
- An account with suitable voms membership (e.g. mice)



GDA System Requirements

- Python (at least version 2.4.3)
- Standard UI installation (grid commands available)
- Python modules pg and sqlite for database access
- An account with suitable voms membership (e.g. mice)
- An FTS channel between your selected remote and local SE



GDA System Requirements

- Python (at least version 2.4.3)
- Standard UI installation (grid commands available)
- Python modules pg and sqlite for database access
- An account with suitable voms membership (e.g. mice)
- An FTS channel between your selected remote and local SE
- Files are transferred asynchronously using glite-transfer-submit



GDA Features



GDA Features

- Customised by a simple ASCII config file, e.g. specify source and destination endpoints location of database files and local (non-grid) cache etc.



GDA Features

- Customised by a simple ASCII config file, e.g. specify source and destination endpoints location of database files and local (non-grid) cache etc.
- Downloaded files are catalogued in a local sqlite database and LFC.



GDA Features

- Customised by a simple ASCII config file, e.g. specify source and destination endpoints location of database files and local (non-grid) cache etc.
- Downloaded files are catalogued in a local sqlite database and LFC.
- Standard log file generation with variable logging levels



GDA Features

- Customised by a simple ASCII config file, e.g. specify source and destination endpoints location of database files and local (non-grid) cache etc.
- Downloaded files are catalogued in a local sqlite database and LFC.
- Standard log file generation with variable logging levels
- Standard option to download all newly available files



GDA Features

- Customised by a simple ASCII config file, e.g. specify source and destination endpoints location of database files and local (non-grid) cache etc.
- Downloaded files are catalogued in a local sqlite database and LFC.
- Standard log file generation with variable logging levels
- Standard option to download all newly available files
- Uses PostgreSQL db filled by front-end transfer tool to determine what is available



GDA Features

- Customised by a simple ASCII config file, e.g. specify source and destination endpoints location of database files and local (non-grid) cache etc.
- Downloaded files are catalogued in a local sqlite database and LFC.
- Standard log file generation with variable logging levels
- Standard option to download all newly available files
- Uses PostgreSQL db filled by front-end transfer tool to determine what is available
- Automatic renewal of voms proxy for unattended running as a cron job for up to a month



GDA Features

- Customised by a simple ASCII config file, e.g. specify source and destination endpoints location of database files and local (non-grid) cache etc.
- Downloaded files are catalogued in a local sqlite database and LFC.
- Standard log file generation with variable logging levels
- Standard option to download all newly available files
- Uses PostgreSQL db filled by front-end transfer tool to determine what is available
- Automatic renewal of voms proxy for unattended running as a cron job for up to a month
- User selectable file downloads, e.g. selection by run number(s)



GDA Features

- Customised by a simple ASCII config file, e.g. specify source and destination endpoints location of database files and local (non-grid) cache etc.
- Downloaded files are catalogued in a local sqlite database and LFC.
- Standard log file generation with variable logging levels
- Standard option to download all newly available files
- Uses PostgreSQL db filled by front-end transfer tool to determine what is available
- Automatic renewal of voms proxy for unattended running as a cron job for up to a month
- User selectable file downloads, e.g. selection by run number(s)
- Ability to copy downloaded files from local SE to a non-grid disk cache for local analysis via lcg-cp commands.



Basically ...



Basically ...

- **Basically, if you have a reasonable, Grid-enabled site, and you want a copy of MICE data then this is how you should be downloading them**



Basically ...

- **Basically, if you have a reasonable, Grid-enabled site, and you want a copy of MICE data then this is how you should be downloading them**
- This is scalable in a way that downloading from the website shouldn't be



Basically ...

- **Basically, if you have a reasonable, Grid-enabled site, and you want a copy of MICE data then this is how you should be downloading them**
- This is scalable in a way that downloading from the website shouldn't be
- Website should be for a few files here or there or for none
Grid-enabled sites.



GDA command line options

Usage: `./GDA.py [-amhpcldr | --keyword] [arguments]`

Options:

- `-a, --all` : Download all new files available
- `-m, --monitor` : Monitor and update the status of existing transfers
- `-h, --help` : print this help.
- `-p, --proxy` : check for valid proxy.
- `-c, --check` : check local copies are catalogued in LFC
- `-l, --local` : copy files from local SE to non-grid cache
- `-d, --detach` : run copies to non-grid cache as a detached process
- `-r, --runs=runlist` : download selected runs (e.g. 10,12,23-30)

`--loglevel=level` (one of ['DEBUG', 'INFO', 'WARNING', 'ERROR', 'CRITICAL']
default is 'INFO')

`--logfile=filename` (logfile, default is stderr)

`--force` : Force local database update if transfer 'FinishedDirty'

Arguments: There are none defined right now.

Typical use in a cron job:

```
./GDA.py --all --logfile=download.log
```



Example configuration file

```
# This is an example config file for GDA.py
#
# There are currently four sections defined. Others may be added in future
# versions.
#
# Edit this file as indicated below to produce a version for your local site.

[endpoints]
# The following two lines are required for copying from Castor at RAL
fts = https://lcf.fts.gridpp.rl.ac.uk:8443/glite-data-transfer-fts/services/FileTransfer
remote = srm://srm-mice.gridpp.rl.ac.uk/castor/ads.rl.ac.uk/prod/mice/raw/MICE/

## The following line specifies the location of the Imperial College SE.
## Do not use this one!
## You need to create an appropriate declaration for your local SE.
##
## local = srm://gfe02.grid.hep.ph.ic.ac.uk/pnfs/hep.ph.ic.ac.uk/data/mice/

## This is an example of a local non-grid cache. Change as appropriate.
userdisk = /srv/localstage/mice/

[databases]
# The next four lines specify the location of the Global Postgres db.
remote_db = micemeta
remote_user = mice_rotester
remote_host = mice00.hep.ph.ic.ac.uk
remote_port = 5432

# change the following line as appropriate for your site.
local_db = /path_to/my_local_sqlite.db

[credentials]
# change the next line as appropriate for you.
long_proxy = /location_of/my_long_term.proxy
vo = mice
min_hours = 2

[transfers]
# specify a sensible limit on the maximum number of files to request in a single
# FTS transfer operation.
limit = 100
```



MICE Reconstruction on the Grid

Requirements:



MICE Reconstruction on the Grid

Requirements:

- Install MAUS reconstruction program on the Grid



MICE Reconstruction on the Grid

Requirements:

- Install MAUS reconstruction program on the Grid
- Reconstruction process should be triggered by raw data availability on the Grid (Castor is an obvious choice)



MICE Reconstruction on the Grid

Requirements:

- Install MAUS reconstruction program on the Grid
- Reconstruction process should be triggered by raw data availability on the Grid (Castor is an obvious choice)
- Reco files should be uploaded to Castor



MICE Reconstruction on the Grid - Design



MICE Reconstruction on the Grid - Design

- The reconstruction process is to be triggered by new file records appearing in the MICEMETA database.



MICE Reconstruction on the Grid - Design

- The reconstruction process is to be triggered by new file records appearing in the MICEMETA database.
- A local (to the submitter) database keeps track of submitted jobs, reconstructed files, files being reconstructed (1 file = 1 job)



MICE Reconstruction on the Grid - Design

- The reconstruction process is to be triggered by new file records appearing in the MICEMETA database.
- A local (to the submitter) database keeps track of submitted jobs, reconstructed files, files being reconstructed (1 file = 1 job)
- When a failed file/job record is removed from the local DB, the reconstruction process will be automatically repeated for that file.



MICE Reconstruction on the Grid - Design



MICE Reconstruction on the Grid - Design

- The reconstruction will be mainly done at RAL LCG2 site since a record in MICEMETA guaranties raw data availability at the site and simplifies data upload to Castor (lcg-cr to a close SE).



MICE Reconstruction on the Grid - Design

- The reconstruction will be mainly done at RAL LCG2 site since a record in MICEMETA guaranties raw data availability at the site and simplifies data upload to Castor (lcg-cr to a close SE).
- After a successful reconstruction a Grid job calls a Web Service (the Controller) based at IC to add a new record to MICEMETA.



MICE Reconstruction on the Grid - Design

- The reconstruction will be mainly done at RAL LCG2 site since a record in MICEMETA guaranties raw data availability at the site and simplifies data upload to Castor (lcg-cr to a close SE).
- After a successful reconstruction a Grid job calls a Web Service (the Controller) based at IC to add a new record to MICEMETA.
- In case a job runs at another site the Controller will also perform data movement to Castor using FTS (sites close SE to Castor).



MICE Reconstruction on the Grid - Tests



MICE Reconstruction on the Grid - Tests

- Only limited test have been performed so far (reconstructing a few files in one go)



MICE Reconstruction on the Grid - Tests

- Only limited test have been performed so far (reconstructing a few files in one go)
- A complete workflow has been tested both for RAL and IC.



MICE Reconstruction on the Grid - Tests

- Only limited test have been performed so far (reconstructing a few files in one go)
- A complete workflow has been tested both for RAL and IC.
- A site selection based on data availability has still to be implemented either based on the LFC or a build in gLite mechanism.



Conclusions



Conclusions

- The PPD VM infrastructure is being updated and we will benefit from this



Conclusions

- The PPD VM infrastructure is being updated and we will benefit from this
- Gradually and constantly improving computer security - but need to keep a workable environment e.g. changes to mousehole.



Conclusions

- The PPD VM infrastructure is being updated and we will benefit from this
- Gradually and constantly improving computer security - but need to keep a workable environment e.g. changes to mousehole.
- **Need backup for Antony - sometimes causes delays in development**



Conclusions

- The PPD VM infrastructure is being updated and we will benefit from this
- Gradually and constantly improving computer security - but need to keep a workable environment e.g. changes to mousehole.
- **Need backup for Antony - sometimes causes delays in development**
- Data moving and distribution system in place and working



Conclusions

- The PPD VM infrastructure is being updated and we will benefit from this
- Gradually and constantly improving computer security - but need to keep a workable environment e.g. changes to mousehole.
- **Need backup for Antony - sometimes causes delays in development**
- Data moving and distribution system in place and working
 - If you are self respecting Grid site that wants the data you should be using the GDA



Conclusions

- The PPD VM infrastructure is being updated and we will benefit from this
- Gradually and constantly improving computer security - but need to keep a workable environment e.g. changes to mousehole.
- **Need backup for Antony - sometimes causes delays in development**
- Data moving and distribution system in place and working
 - If you are self respecting Grid site that wants the data you should be using the GDA
- Grid based automatic reconstruction should be here “real soon”

