

The NGS Applications Repository / Grid Portal

Slides selected from a more detailed presentation by
David Meredith
NGS + Grid Technology Group, e-Science Centre, Daresbury
Laboratory, UK

d.j.meredith@dl.ac.uk

NGS Portal

The NGS portal can be used to access and interact with the HPC and Data resources available on the NGS Grid via (Certificates + myproxy):

- Browse for different applications available on a Grid this includes your own personal applications and pre-configured applications (e.g. the NGS is currently publishing applications within the NGS portal to be made easily available for its users).
- Create descriptions of applications and their associated resources (using standard Job Submission Description Language).
- Submit/monitor compute jobs/applications.
- Access and move data.

Portal and Portlets

- The NGS Grid portal extends a JSR-168 compliant portal container which hosts a selection of ‘portlets.’
- Portlets are online-accessible applications that are hosted and managed within the portal container.
- The list of portlets that are deployed to the portal make up the portal’s overall functionality (users may be interested only in a selection of portlets)
- Main Benefit: Portlets facilitate the sharing and re-use of applications (168-compliant 3rd party portlets can be used within a portal as required).
- NGS and SciTech e-Science Centre are developing a collection of portlets designed for the Computational and Data Grid.
- The NGS portal is a current implementation of these Grid portlets.

NGS portal main functionality

1. A **searchable database** of personal and shared JSDL job profile documents (Job Submission Description Language, i.e. 'job recipes/templates').
 - JSDL can be **browsed for, selected, loaded and saved** in order to run applications on the Grid (loaded either 'out-of-the-box' or, more usually loaded, modified/tweaked as required and saved).
 - JSDL can be **searched for** by category of interest in the portal (e.g bioinformatics, chemistry, tutorials/examples).
 - JSDL documents can be **pre-configured and published** by the portlet administrator(s) to be made available to all other users.
2. A **JSDL GUI editor** for graphically authoring, validating, sharing, uploading jobs described in JSDL.
3. An **extensible Grid job submission** and monitoring application for heterogonous Grids (currently, only Globus but more Grid middleware providers are being added, e.g. JDL/gLite).

JSDL – Job Submission Description Language

Ali Anjomshoaa, Fred Brisard, Michel Drescher, Donal K. Fellows, William Lee, An Ly, Steve McGough, Darren Pulsipher, Andreas Savva, Chris Smith

- JSDL 1.0 is an OGF recommendation
- JSDL 1.0 is published as GFD-R-P.56 –
 - Description of JSDL elements and XML Schema,
 - Available at: <http://www.ggf.org/gf/docs/?final>

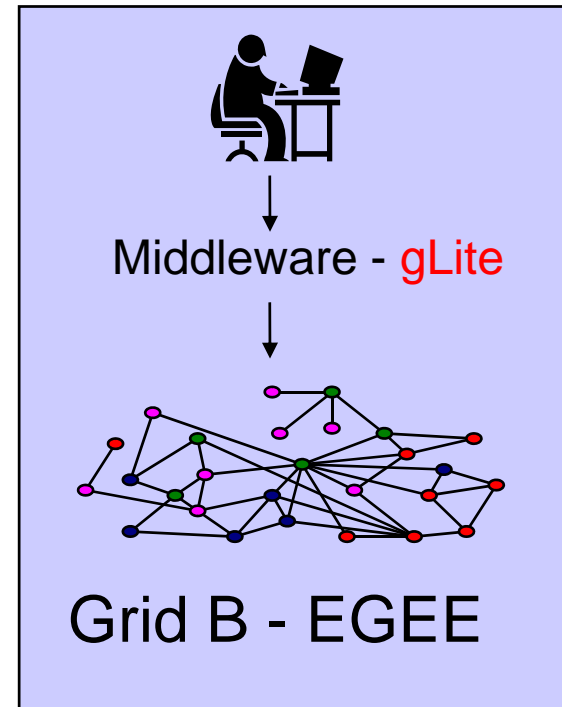
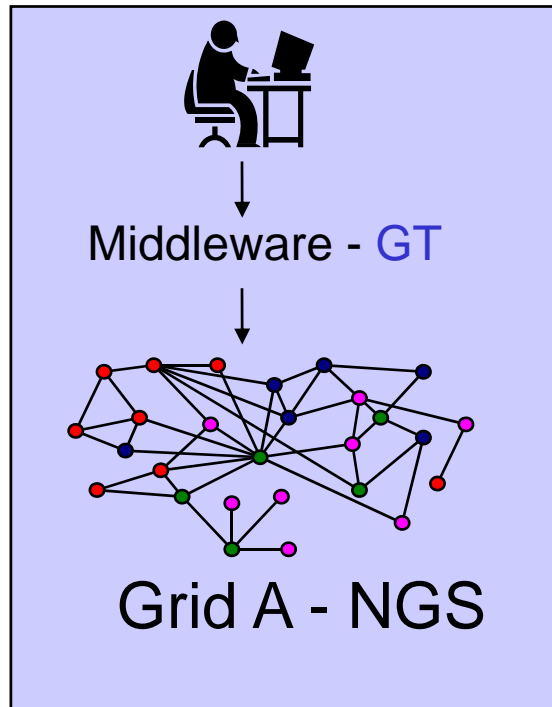
A language for
*describing the
requirements of
computational jobs
for submission to
Grids and other
systems.*

```
<jSDL:Application>
  <jSDL:ApplicationName>gnuplot</jSDL:ApplicationName>
  <jSDL-posix:POSIXApplication>
    <jSDL-posix:Executable>
      /usr/local/bin/gnuplot
    </jSDL-posix:Executable>
    <jSDL-posix:Argument>control.txt</jSDL-posix:Argument>
    <jSDL-posix:Argument>DavesControlFile.txt</jSDL-posix:Argument>
    <jSDL-posix:Input>input.dat</jSDL-posix:Input>
    <jSDL-posix:Output>output1.png</jSDL-posix:Output>
  </jSDL-posix:POSIXApplication>
</jSDL:Application>
<jSDL:Resources>
....
```

Example fragment

Why JSDL

Grids are currently heterogeneous:



- Different middleware implementations adopt different methods for the description of applications and their associated resources, and for their subsequent execution to a Grid.
- A Number of different data storage resources are also relevant for the management and transfer of data, e.g. GsiFTP, SRB, SRM, WebDav and (S)FTP.

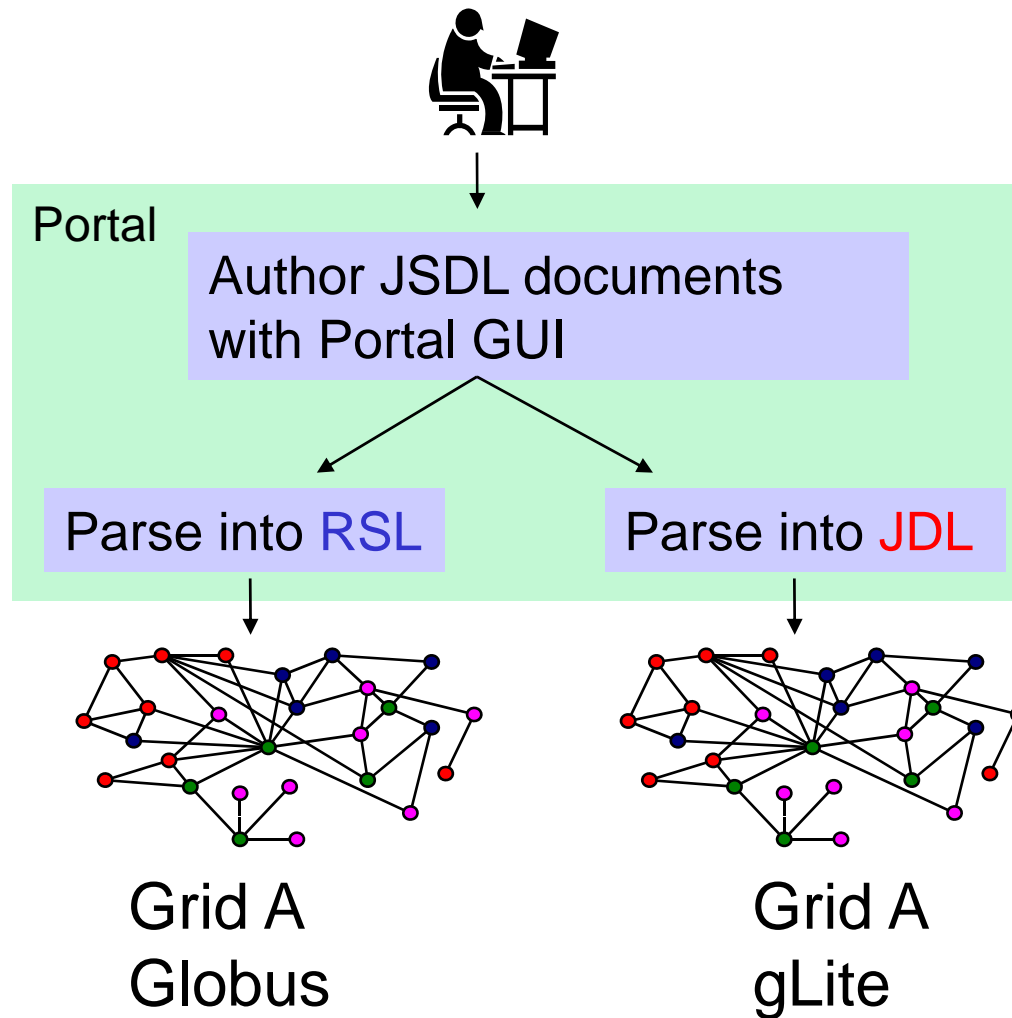
Grid A Globus RSL (Resource Specification Language)

```
&(executable=$(GLOBUSRUN_GASS_URL)/home/ngs0153/cpi) (arguments= 30 fileA)
(jobType=mpi) (environment = (NGSMODULES mpich-gm/1.2.5..10-
intel8.1:intel/fce/9.1.032) (TMP /tmp)) (count = 4) (hostCount = 8) (minMemory = 512)
(maxWallTime = 3) (directory=/home/ngs0153) (stdin=/home/ngs0153/cpi.in)
(stdout=/home/ngs0153/cpi.out) (stderr=/home/ngs0153/cpi.err)
```

Grid B gLite JDL (Job Description Language)

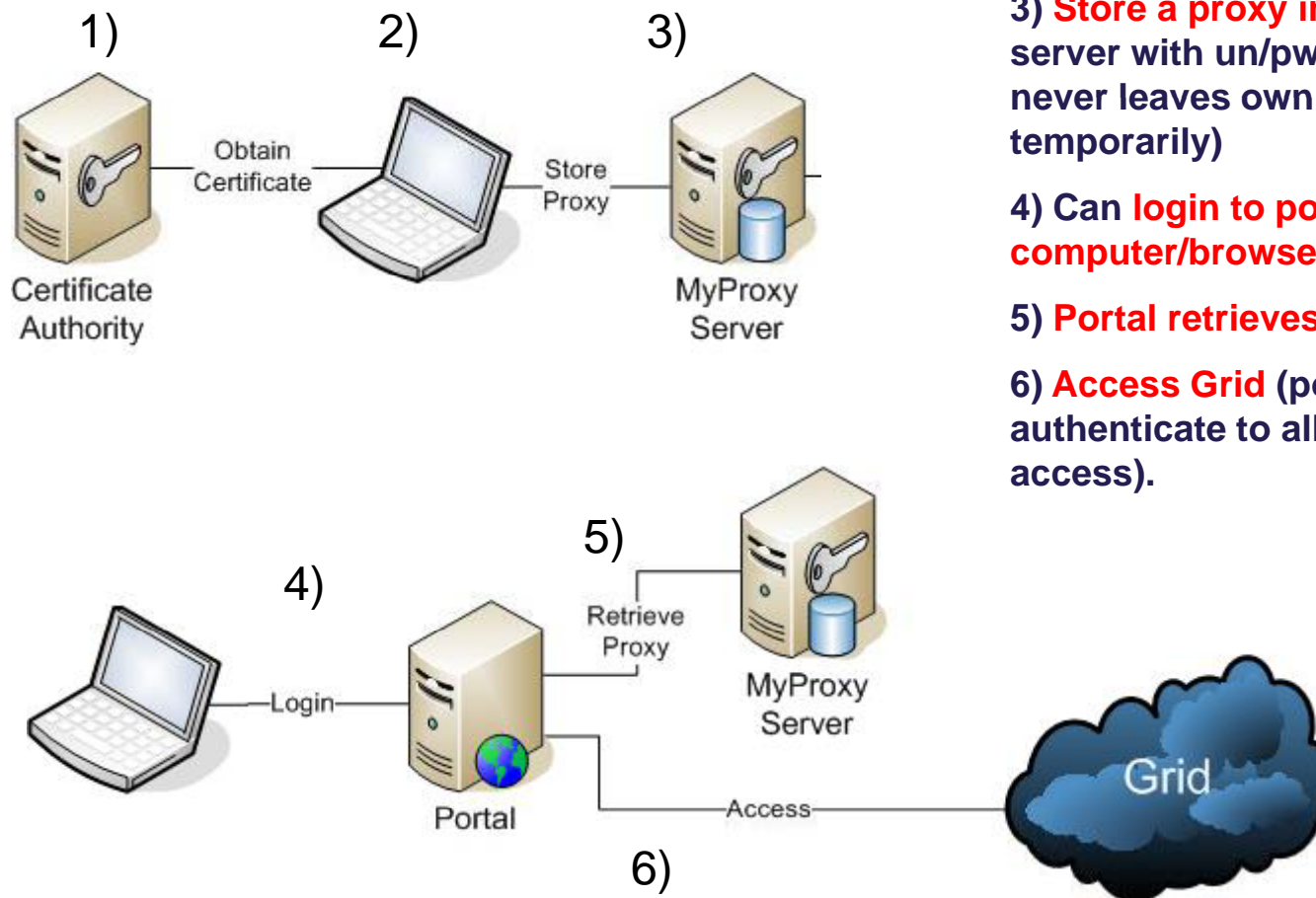
```
Type = "Job";
JobType = "Normal";
RetryCount = 3;
Executable = "/home/ngs0153/cpi";
Arguments = "30 fileA";
VirtualOrganisation = "myGridVOproject";
StdInput = "cpi.in";
StdOutput = "cpi.out";
StdError = "cpi.err";
InputSandbox = { "gsiftp://grid-data.rl.ac.uk:2811/home/ngs0153/cpi", "gsiftp://grid-
data2.dl.ac.uk:2811/myhome/fileA" };
InputSandboxDestFileName = { "cpi", "fileA" };
OutputSandbox = { "cpi.out" };
OutputSandboxDestURI = { "gsiftp://mygridhome.dl.ac.uk:2811/myhome" };
DeleteOnTermination = { "fileA" };
Environment = { "NGSMODULES=mpich-gm/1.2.5..10-intel8.1:intel/fce/9.1.032", "TMP=/tmp" };
Requirements = ( other.GlueCEInfoLRMSType == "PBS" ) && ( member( GlueCEInfoHostName,
{"grid-data.rl.ac.uk:2119" , "mygrid-resource.dl.ac.uk:2119" } ) ) && ( GlueHostProcessorModel ==
"Intel" );
Rank = -other.GlueCEStateEstimatedResponseTime;
```

JSDL and the NGS Grid portal



- One interface definition for job submission
- One job description language
- JSDL is a standard job submission description language (middleware agnostic)

MyProxy Login Procedure



1) **Obtain a certificate** from CA - .p12 / .pfx certificate bundle (one time only)

2) **Create a public/private key pair** - usercert.pem / userkey.pem (one time only)

3) **Store a proxy in UK eScience MyProxy** server with un/pw – delegation, private key never leaves own computer (done temporarily)

4) **Can login to portal from any computer/browser** using same un/pw.

5) **Portal retrieves proxy** using same un/pw.

6) **Access Grid** (portal uses proxy to authenticate to all Grid resources giving SSO access).

JSDL v 1.0

1. XML Schema language for describing compute jobs in a **platform independent** language (XML).
2. Is **agnostic of middleware** - no dependencies on Globus, WSRF, gLite (portal that is **generic and not tied to any particular set of Grid technologies**).
3. OGF (once GGF) **recommendation**.
4. JSDL documents can be **validated** against the JSDL and JSDL POSIX XSD Schema documents to ensure correctness.
5. A JSDL XML **document** describes the requirements of a compute job and its associated resources, e.g. description of required files / data
 - ***What to do, not how to do it***
 - It is an **extensible** template language (elements defined in another XML schema with a different namespace can be added to a JSDL document). This facilitates customisation and nesting of extra information.
 - Is **loosely typed** – does not impose tight controls on the values of elements using advanced XML schema restriction techniques such as `xsd:patterns`
6. **Not included** in JSDL version 1.0
Scheduling, Workflow, Security ...

Edited highlights

- The following few slides show some of the functionality of the Repository – not all!

JSDL Repository / Database

Portal is 'open', are free to browse public JSDL documents without log-in and free to use portal as open JSDL editor.

Login to **browse** personal applications and submit jobs.

List jobs, read job descriptions and **load** a job to initialise the 'Active Job.'

Changes to the parameters in the GUI will update the generated JSDL automatically.

Browse and Load the 'Active Job'

1) Select a job category (e.g. Tutorials/Examples) and list the jobs in that category (NGS or personal jobs)
 2) Select and load the ActiveJob from the list (click on the load link).

In Job Category: [\(Create/Edit JobCategories\)](#) **RAL Bioinformatics**
 With Status: **NGS** **Personal**
 Search / List Job Profiles:

Results Found: (9)

Description	Name	Exe	Modified	Status (Check)	Load
View ▶ SIESTA (parallel example)	/apps/siesta-mpi/siesta-2	Jan 17, 2007	<input type="checkbox"/>	load	
View ▶ FASTA (serial example)	/usr/local/applications/bio	Jan 17, 2007	<input type="checkbox"/>	load	
View ▶ NAMD (parallel example)	/usr/local/applications/bio	Jan 17, 2007	<input type="checkbox"/>	load	
View ▶ GROMACS (grompp exam	/usr/local/applications/bio	Jan 17, 2007	<input type="checkbox"/>	load	
View ▶ GROMACS (MD example)	/usr/local/applications/bio	Jan 17, 2007	<input type="checkbox"/>	load	
View ▶ mpiBLAST (blastn exampl	/usr/local/applications/bio	Jan 18, 2007	<input type="checkbox"/>	load	
View ▶ EMBOSS (seqret example)	/usr/local/applications/bio	Jan 18, 2007	<input type="checkbox"/>	load	
View ▶ EMBOSS (wossname exa	/usr/local/applications/bio	Jan 18, 2007	<input type="checkbox"/>	load	
View ▶ BLAST (serial application)	/usr/local/applications/bio	Jan 19, 2007	<input type="checkbox"/>	load	

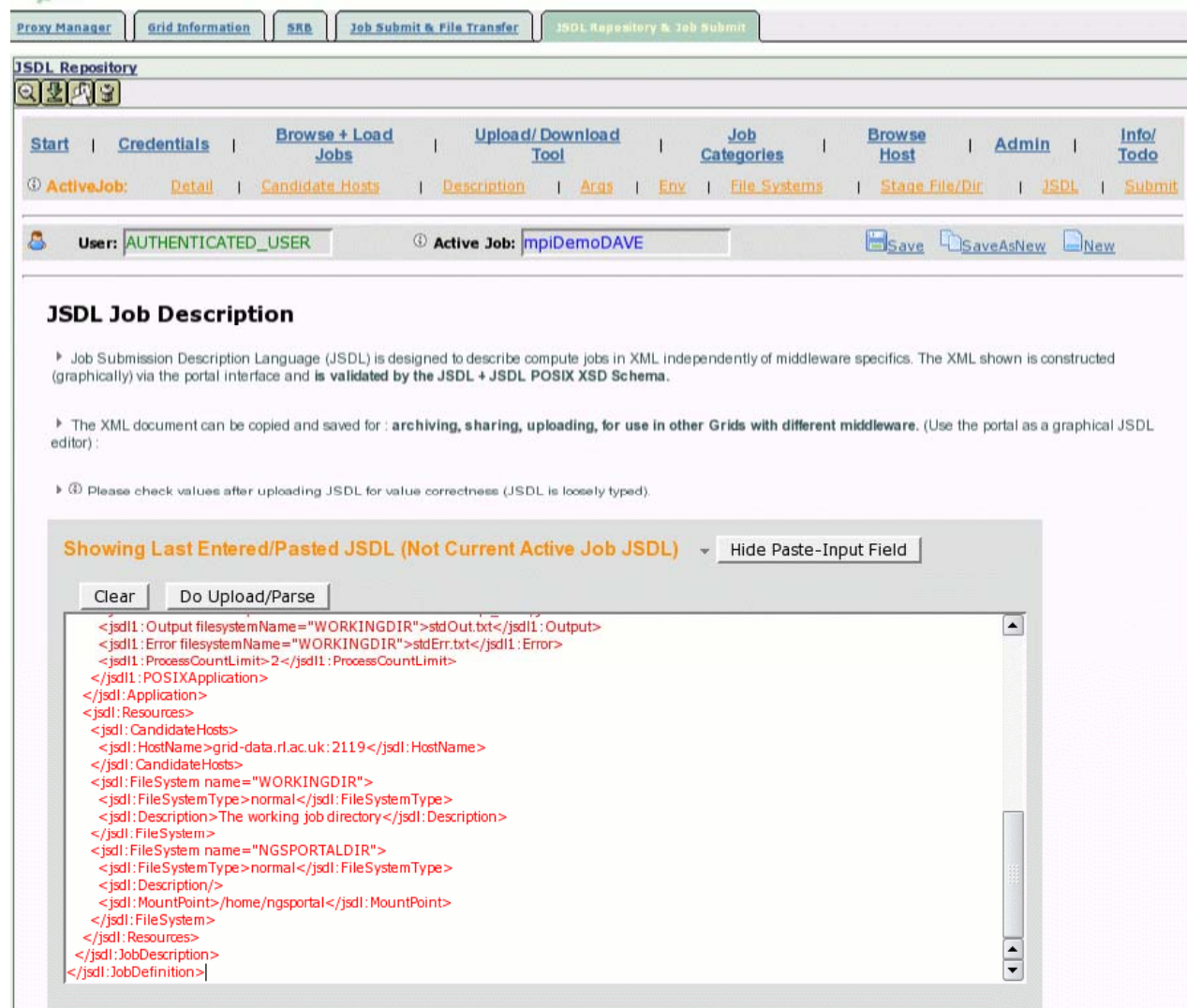
[Reset](#)

Upload / Share JSDL

JSDL documents can be uploaded to the portal in order to initialise the Active Job.

Validation errors and messages are displayed in the interface when uploading JSDL.

The portlet allows the sharing of job profiles and JSDL between users and user communities.



The screenshot shows the NCS National Grid Service JSDL Repository interface. The top navigation bar includes links for Proxy Manager, Grid Information, SRB, Job Submit & File Transfer, and JSDL Repository & Job Submit. The JSDL Repository section has a search bar and a set of tabs: Start, Credentials, Browse + Load Jobs, Upload/Download Tool, Job Categories, Browse Host, Admin, and Info/ToDo. Below these tabs are links for ActiveJob, Detail, Candidate Hosts, Description, Args, Env, File Systems, Stage File/Dir, JSDL, and Submit. The user is authenticated as AUTHENTICATED_USER and the active job is mpiDemoDAVE. The main content area is titled "JSDL Job Description" and contains instructions on how to use JSDL. A text area shows the last entered/pasted JSDL code, which is for a job named "WORKINGDIR".

JSDL Job Description

► Job Submission Description Language (JSDL) is designed to describe compute jobs in XML independently of middleware specifics. The XML shown is constructed (graphically) via the portal interface and **is validated by the JSDL + JSDL POSIX XSD Schema**.

► The XML document can be copied and saved for : **archiving, sharing, uploading, for use in other Grids with different middleware**. (Use the portal as a graphical JSDL editor) :

► Please check values after uploading JSDL for value correctness (JSDL is loosely typed).

Showing Last Entered/Pasted JSDL (Not Current Active Job JSDL)

```
<jSDL:Output filesystemName="WORKINGDIR">stdOut.txt</jSDL:Output>
<jSDL:Error filesystemName="WORKINGDIR">stdErr.txt</jSDL:Error>
<jSDL:ProcessCountLimit>2</jSDL:ProcessCountLimit>
</jSDL:POSIXApplication>
</jSDL:Application>
<jSDL:Resources>
<jSDL:CandidateHosts>
<jSDL:HostName>grid-data.rl.ac.uk:2119</jSDL:HostName>
</jSDL:CandidateHosts>
<jSDL:FileSystem name="WORKINGDIR">
<jSDL:FileSystemType>normal</jSDL:FileSystemType>
<jSDL:Description>The working job directory</jSDL:Description>
</jSDL:FileSystem>
<jSDL:FileSystem name="NGSPORTALDIR">
<jSDL:FileSystemType>normal</jSDL:FileSystemType>
<jSDL:Description/>
<jSDL:MountPoint>/home/ngsportal</jSDL:MountPoint>
</jSDL:FileSystem>
</jSDL:Resources>
</jSDL:JobDescription>
</jSDL:JobDefinition>
```


Stage Data

Compile a list of data from across the Grid that should be copied to the consuming system prior to job execution (source URI). Also after job completes (target URI).

JSDL does not mandate the protocol / URI format.

Data is staged relative to named file systems on the consuming system.

The source URI can be either specified manually or, more normally, browsed for in the 'Browse Host' page.

The screenshot shows the 'JSDL Repository' web interface. At the top, there are tabs for 'Proxy Manager', 'Grid Information', 'SRB', 'Job Submit & File Transfer', and 'JSDL Repository & Job Submit'. Below the tabs, there is a navigation bar with links: 'Start', 'Credentials', 'Browse + Load Jobs', 'Upload/ Download Tool', 'Job Categories', and 'Browse Host'. A sub-navigation bar includes 'ActiveJob: Detail', 'Candidate Hosts', 'Description', 'Args', 'Env', 'File Systems', and 'Stage File/Dir'. The main content area is titled 'Stage Data To The Execute Host'. It contains a table with columns: 'Edit', 'File System Name (Add)', 'File/Dir Name', 'Creation', and 'Source URI (Browse Host)'. The table has three rows: 'WORKINGDIR' with 'dave.pl', 'WORKINGDIR' with 'dave.txt', and 'WORKINGDIR' with 'dummy.scala'. Below the table, there is a 'Data Source' section with a 'Source URI' field containing 'gsiftp://grid-data.rl.ac.uk:2811/home/ngs0153/dummy.scala'. A 'Destination on Execute Host' section includes a 'File/Dir Path' field with a 'File System' dropdown set to 'WORKINGDIR' and a 'Name' field containing 'dummy.scala'. There is also a 'Description' field and a 'Creation Option' dropdown set to 'overwrite'. At the bottom, there are buttons for 'View StageLog', 'Validate SourceURIs', 'Validate MountPoints', 'Do Staging', 'Delete Selected', and 'Close Detail'.

<jSDL:DataStaging>

<jSDL:FileName>dave.txt</jSDL:FileName>

<jSDL:FilesystemName>WORKINGDIR</jSDL:FilesystemName>

<jSDL:CreationFlag>overwrite</jSDL:CreationFlag>

<jSDL>DeleteOnTermination>true</jSDL>DeleteOnTermination>

<jSDL:Source>

<jSDL:URI>gsiftp://grid-data2.dl.ac.uk:2811/home/ngs0153/dave.txt</jSDL:URI>

</jSDL:Source>

</jSDL:DataStaging>

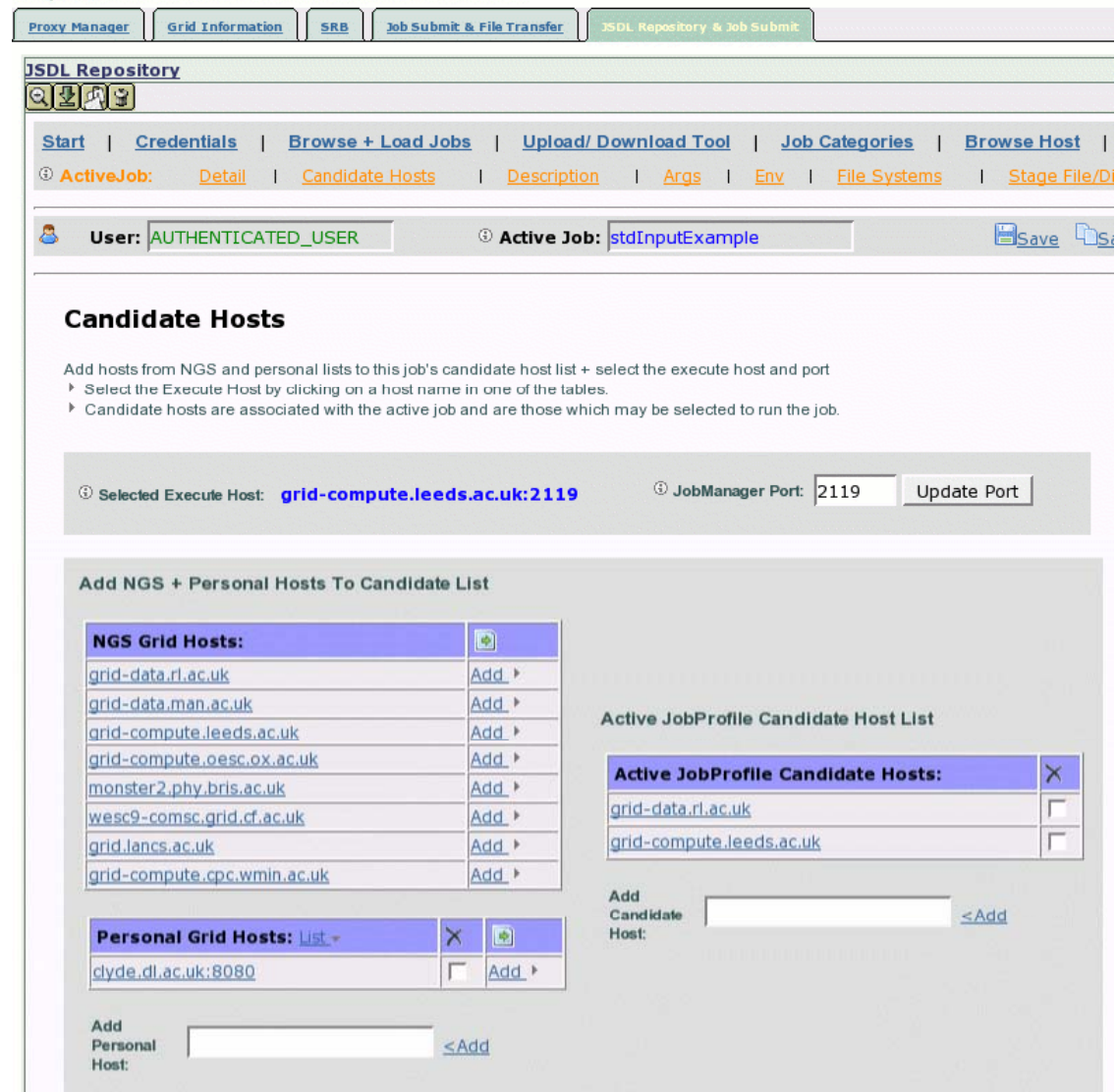
Candidate Hosts

Candidate hosts are consuming systems (execute hosts) that can be to run the Active Job.

The candidate host list can be compiled from a personal host list and from a default host list (available to all users).

In future, a RB matchmaking will be used to auto-compile the candidate host list.

```
<jSDL:CandidateHosts>
  <jSDL:HostName>
    grid-data.rl.ac.uk:2119
  </jSDL:HostName>
  <jSDL:HostName>
    grid-compute.leeds.ac.uk:2119
  </jSDL:HostName>
</jSDL:CandidateHosts>
```



The screenshot shows the 'JSDL Repository' web interface. The top navigation bar includes links for Proxy Manager, Grid Information, SRB, Job Submit & File Transfer, and JSDL Repository & Job Submit. The 'JSDL Repository' section has a sub-navigation bar with links for Start, Credentials, Browse + Load Jobs, Upload/Download Tool, Job Categories, and Browse Host. Below this, there's a status bar showing 'ActiveJob: Detail', 'Candidate Hosts', 'Description', 'Args', 'Env', 'File Systems', and 'Stage File/D'. The user is logged in as 'AUTHENTICATED_USER' and the active job is 'stdInputExample'. The 'Candidate Hosts' section is titled 'Candidate Hosts' and contains instructions: 'Add hosts from NGS and personal lists to this job's candidate host list + select the execute host and port'. It also provides a 'Selected Execute Host' of 'grid-compute.leeds.ac.uk:2119' and a 'JobManager Port' of '2119'. Below this, there's a section 'Add NGS + Personal Hosts To Candidate List' which contains two tables. The 'NGS Grid Hosts' table lists several hosts with 'Add' buttons. The 'Personal Grid Hosts' table lists 'clyde.dl.ac.uk:8080' with an 'Add' button. To the right, the 'Active JobProfile Candidate Host List' table shows the selected hosts: 'grid-data.rl.ac.uk' and 'grid-compute.leeds.ac.uk'.

Candidate Hosts

Add hosts from NGS and personal lists to this job's candidate host list + select the execute host and port

- ▶ Select the Execute Host by clicking on a host name in one of the tables.
- ▶ Candidate hosts are associated with the active job and are those which may be selected to run the job.

Selected Execute Host: **grid-compute.leeds.ac.uk:2119** JobManager Port:

Add NGS + Personal Hosts To Candidate List

NGS Grid Hosts:	
grid-data.rl.ac.uk	<input type="button" value="Add"/>
grid-data.man.ac.uk	<input type="button" value="Add"/>
grid-compute.leeds.ac.uk	<input type="button" value="Add"/>
grid-compute.oesc.ox.ac.uk	<input type="button" value="Add"/>
monster2.phy.bris.ac.uk	<input type="button" value="Add"/>
wesc9-comsc.grid.cf.ac.uk	<input type="button" value="Add"/>
grid.lancs.ac.uk	<input type="button" value="Add"/>
grid-compute.cpc.wmin.ac.uk	<input type="button" value="Add"/>

Personal Grid Hosts:	
clyde.dl.ac.uk:8080	<input type="button" value="Add"/>

Add Personal Host:

Active JobProfile Candidate Host List

Active JobProfile Candidate Hosts:	
grid-data.rl.ac.uk	<input type="checkbox"/>
grid-compute.leeds.ac.uk	<input type="checkbox"/>

Add Candidate Host:

Browse Hosts

Browse remote
Grid hosts for
data required for
staging.

Select files and
directories that
should be copied
to the consuming
system via GsiFtp
(more protocols to
be supported inc
srb, webdav).

The screenshot shows the 'JSDL Repository' web interface. At the top, there are navigation tabs: 'Proxy Manager', 'Grid Information', 'SRB', 'Job Submit & File Transfer', and 'JSDL Repository & Job Submit'. Below these, there are icons and a set of links: 'Start', 'Credentials', 'Browse + Load Jobs', 'Upload/ Download Tool', 'Job Categories', and 'Browse Host'. A secondary set of links includes 'ActiveJob: Detail', 'Candidate Hosts', 'Description', 'Args', 'Env', 'File Systems', and 'Stage File/Di'. Below this is a user/active job section with 'User: AUTHENTICATED_USER' and 'Active Job: JobProfile name', along with 'Save' and 'Se' buttons.

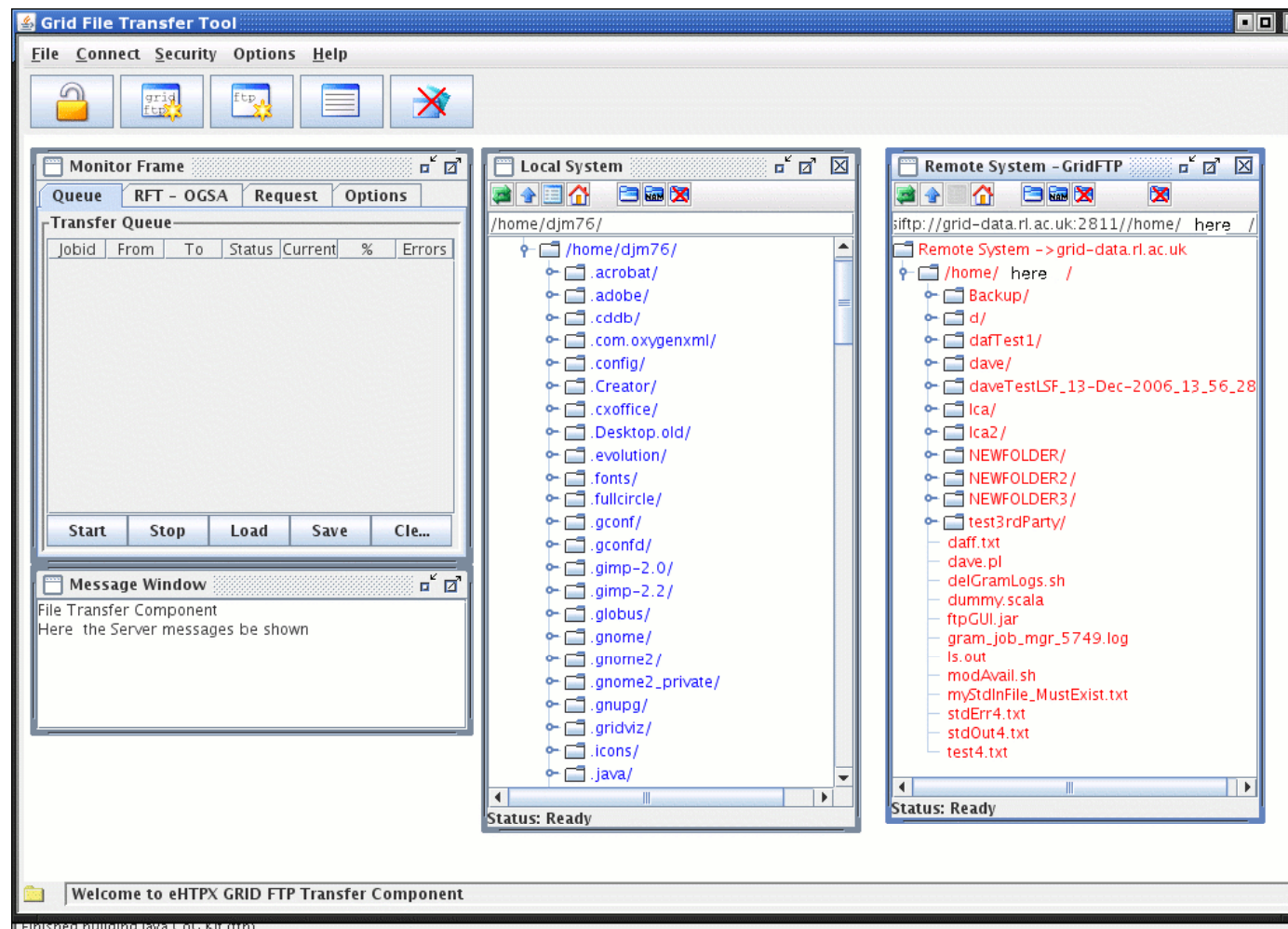
The main section is titled 'Browse Host'. It contains two input fields: 'Browse Host/Port:' with a dropdown menu showing '(grid-data.rl.ac.uk) grid-data.rl.a' and a text box with '2811', and 'ExecuteHost: (Update)' with a dropdown menu showing '()'. Below these are two links: 'Update WORKINGDIR Val' and 'Update WORKINGDIR Val'.

Below the input fields is a table listing files and directories. The table has columns: 'Name', 'State', 'Mode', 'Size (bytes)', 'Modified', and a checkbox. The table contains the following entries:

Name	State	Mode	Size (bytes)	Modified	
Backup2	<input type="checkbox"/>	755	3864	Aug 13	<input type="checkbox"/>
d	<input type="checkbox"/>	755	3864	Dec 17	<input type="checkbox"/>
daff.txt	<input type="checkbox"/>	644	21	Dec 17	<input type="checkbox"/>
daffTest1	<input type="checkbox"/>	755	3864	Jul 25	<input type="checkbox"/>
dave	<input type="checkbox"/>	755	3864	Nov 20	<input type="checkbox"/>
daveName	<input type="checkbox"/>	755	3864	Jan 8	<input type="checkbox"/>
dave.pl	<input type="checkbox"/>	755	3319	Aug 22	<input type="checkbox"/>
daveTestLSF 13-Dec-2006 13 56 28	<input type="checkbox"/>	755	3864	Dec 13	<input type="checkbox"/>
dave.txt	<input type="checkbox"/>	664	10	Dec 19	<input type="checkbox"/>
delGramLogs.sh	<input type="checkbox"/>	644	22	Nov 12	<input type="checkbox"/>
dummy.scala	<input type="checkbox"/>	644	5813	Nov 26	<input type="checkbox"/>
ftpGUI.jar	<input type="checkbox"/>	644	13028780	Feb 14	<input type="checkbox"/>
gram job_mgr_9260.log	<input type="checkbox"/>	644	8340	Jan 4	<input type="checkbox"/>
hello.java	<input type="checkbox"/>	644	124	Jan 4	<input type="checkbox"/>
lca	<input type="checkbox"/>	775	3864	Dec 14	<input type="checkbox"/>

GridFtp Upload / Download / File Transfer Tool

Transfer files to/from your desktop and a remote Grid ftp server via 'drag-n-drop' – akin to Windows explorer for the Grid (in serious need of update – currently using depreciated libs).



Summary

1. Open / free to use JSDL GUI editor, browse the list of applications hosted to date. **Please contact NGS to request more hosted applications** (currently an ongoing task, more will appear soon).
2. The application is designed to be **generic and not tied to any particular set of Grid technologies** (facilitated using JSDL). Middleware dependencies added at **job-submit/monitor time when the specifics of the middleware have to be accommodated** (e.g. parsing the JSDL into RSL, adding mw-specific parameters, e.g. RSL JobType).
3. Application can be **deployed as a JSR-168 portlet or as a standalone Web application**. Portal can therefore be hosted in project specific portals.
4. Currently, application only supports Globus, but GridSAM and gLite JDL (NGS RB) are scheduled for addition (more on GridSAM next few days).

TODO / Future

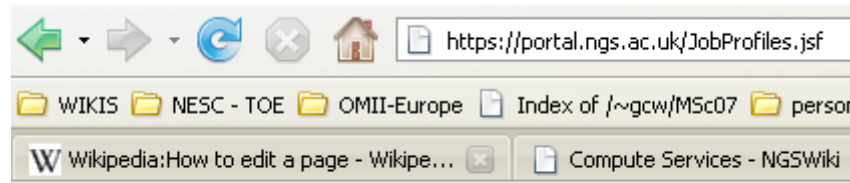
- **Extend portal to accommodate more/new middleware**, e.g. NGS gLite resource broker and JSDL conversion to JDL.
- **Parametric jobs** (depends partly on parametric schema extensions for JSDL and subsequent middleware support).
- Extend application to support **staging from more Data Grid + Web protocols (SRB)**. This will involve browsing / interacting with different data storage resources within the interface (e.g. browse SRB). Performing staging across different protocols adds some complexity.
- Growing list of improvements / suggestions to refine interface
- **Release the application** for use in other projects / Grids (scheduled for OMII open source release).
- Grid will probably move to a more SOA style architecture – more use of temporary ‘sandboxes’ for job execution rather than provision of permanent user accounts on compute resources (staging become more important).

*** A portal is only as good as the underlying deployed infrastructure.... portal development often involves debugging the underlying consuming systems and middleware ***

Software Stack

1. **JSF (Java Server Faces)** for server application interface development: GUI interface components + MVC controller (Http session and request scope data). Vanilla JSF is JSR-168 compliant so can host applications as Web app or portlet.
2. **Spring v2.0 for managing objects in an n-tier server** application (recommended in a lightweight non J2EE application server, e.g. Tomcat/Jetty), e.g. managing singleton + prototype object graphs, declarative transaction demarcation, data source management, propagation of persistence context across long running transactions.
3. **C3p0** db connection pooling.
4. **JPA (Java persistence API) for ORM (object relational mapping)**. Hibernate 3.2 for domain model (could use Kodo, Toplink, apache openJPA)
5. **CogKit** for Globus API from Globus Alliance.
6. **Xml-to-object data binding framework** – Apache XMLBeans (or JAX-B).

Go to <https://portal.ngs.ac.uk/>



- Authenticate using MyProxy

NGS National Grid Service
JSDL Application Repository / Job Submission Portal

My Grid Identity: None

Authenticate

Applications Repository

My Job : JobProfile na

Authenticate using MyProxy Server

Upload Credentials: [Launch Upload Tool](#) | [Launch New Upload Tool \(requires](#)

MyProxy Server: myproxy.grid-support.ac.uk:7512

Myproxy Username: user00

Myproxy Password: [masked]

Lifetime (hours): 2

Login

- Then in another browser window, follow link on agenda page