

OGSA – DAI

Open Grid Services Architecture – Data Access and Integration

Release Notes

OGSA-DAI Release 3.1

Project Title: OGSA-DAI GridServices
Document Title: Release Notes
Document Identifier: OGSA-DAI-USER-RELEASE-NOTES-v2.9g
Authorship: Ali Anjomshoaa / Neil Chue Hong / Mike Jackson

Document History:

Personnel	Date	Summary	Version
NCH	11/02/2004	GT3.2beta Version	2.9g
NCH	09/02/2004	Added DB2 newline bug workaround	2.9
NCH	06/02/2004	Updated for Release 3.1	2.8
NCH	26/09/2003	Updated for Release 3.0.2	2.7
NCH	30/07/2003	Minor updates and additions	2.6
NCH	25/07/2003	Updated for Phase 2 Release 3.0	2.5
MJJ	05/06/2003	Added references to Globus Toolkit 3 Beta and other bug fixes.	2.4
NCH	02/06/2003	Minor updates and additions	2.3
AA	02/06/2003	Minor Changes to Sections 7 and 8	2.2
AA	30/05/2003	Updated for Phase 2 Release 2.5 (P2R2.5)	2.1
AA	10/04/2003	Updated for Phase 2 Release 2 (P2R2)	2.0
MJJ	25/02/2003	New logo / Alpha 1.5 compatible	1.0
MJJ	04/02/2003	Added known problems and workarounds	0.5
MJJ	16/01/2003	Fourth Draft – post Release 1 minor corrections	0.4
MJJ	15/01/2003	Third Draft	0.3
AA	14/01/2003	Second Draft	0.2
AA	20/12/2002	First Draft	0.1

Contents

Contents.....	2
1 Introduction.....	3
2 Release Overview	3
2.1 Release Namespace	3
3 Platform Dependencies	3
4 Release Limitations and Characteristics	4
4.1 Grid Middleware	4
4.2 Database Support.....	4
4.3 Web Services Container	4
4.4 OGSI Notifications.....	4
4.5 OGSA-DAI Sample Clients	4
4.6 OGSA-DAI Test Framework	5
4.7 Security and Role Mapping	5
4.8 Restrictions on the DAIServiceGroupRegistry (DAISGR)	5
5 Changes Since Previous Release.....	5
5.1 Changes to the DAIServiceGroupRegistry (DAISGR)	5
5.2 Changes to the GridDataServiceFactory (GDSF).....	5
5.3 Changes to the GridDataService (GDS)	5
5.4 Additional Functionality – Client Toolkit	6
5.5 Other changes since R3.0	6
5.6 Summary of changes between R2.5 and R3.0	6
5.7 Summary of changes between R2 and R2.5	7
6 Bug Fixes	7
7 Known Problems and Remedies	9
7.1 Service Access after Tomcat Startup.....	9
7.2 Performing XUpdate queries with Xindice 1.0 and Java 1.4 (JDK 1.4).....	9
7.3 UNIX setenv Scripts.....	10
7.4 Problem With Using Sun’s tar Command	10
7.5 Location of ogsadai.jar	11
7.6 Location of Message Bundle	11
7.7 Oracle JDBC driver renaming	11
7.8 Java Virtual Machine fails.....	11
7.9 Errors when sending perform documents to DB2 database.....	11
7.10 Request Termination time too short – client times out	12
8 Frequently Asked Questions	12
8.1 No GDSFs Found	12
8.2 Configuring Xindice data resources	12
8.3 Synchronous Request problems performing SQL Updates	13
8.4 Setting log4j properties	13
8.5 OGSA-DAI and the Globus Service Container	13
Acknowledgements	14

1 Introduction

This document contains the release notes for the OGSA-DAI Release 3.1 release, from hereon referred to as “R3.1”. This release contains the basic functionalities of the OGSA-DAI software that form its core infrastructure. OGSA-DAI seeks to be a reference implementation of the Grid Database Service Specification (GDSS) from the Database Access and Integration Services Working Group (DAIS-WG) of the GGF.

2 Release Overview

This release improves on previous releases of the OGSA-DAI software, taking account of comments received and adding extra functionality. This release is aimed at previous early adopters of the OGSA-DAI software, plus new e-Science projects who wish to use OGSA-DAI.

The implementation of the current release is compliant with the version 1.0 (draft 29, April 5th 2003) of the Open Grid Services Infrastructure Specification (OGSI specification) from the Open Grid Service Infrastructure Working Group (OGSI-WG) of the Global Grid Forum (GGF) (<http://www.gridforum.org/ogsi-wg/>). In addition, it broadly follows the February 16th 2003 version of the Grid Database Service Specification (GDSS) (presented at GGF7, Tokyo, Japan) from the Database Access and Integration Services Working Group (DAIS-WG) of the GGF.

2.1 Release Namespace

The release namespace is `ogsadai.org.uk`.

3 Platform Dependencies

The OGSA-DAI software is implemented in Java and based on the Java 2 platform. It should not, therefore, have any platform dependencies for those that support Java 2.

This release of the OGSA-DAI implementation has been tested on the following platforms:

Operating System	Processor Architecture	Java 2 Platform	Test Status
Sun - Solaris8	SPARC	JDK 1.4.1	Passed
Linux – Redhat 9	X86	JDK 1.4.1	Passed
Windows 2000	X86	JDK 1.4.1	Passed
Windows XP	X86	JDK 1.4.1	Passed

4 Release Limitations and Characteristics

This release has the following limitations and characteristics.

4.1 Grid Middleware

This release uses and needs the Globus Toolkit 3.2 beta release (<http://www-unix.globus.org/toolkit/download.html>), which is the Globus (<http://www.globus.org/>) implementation of the OGSI specification from the OGSI-WG. **This release is not compatible with any previous Globus Toolkit 3.0 releases.**

It may also work with future GT3.2 releases, although this cannot be guaranteed.

4.2 Database Support

The functionality of this release has been tested with the MySQL 3.23 / 4.0, DB/2 v8 and Oracle 9i databases and is consistent with the SQL-92 standard. It has also been tested with the Xindice 1.0 database and supports XPath queries. The release supports regular expression searches of XML documents.

Database management functionality is dependant on the underlying database supporting management via an API such as JDBC. In particular, Oracle does not support the creation or deletion of databases via OGSA-DAI.

4.3 Web Services Container

This release runs under the Apache Tomcat/Axis hosting environment. Please see the Installation Guide (OGSA-DAI-USER-INSTALL) for the relevant versions of the required software and tools.

4.4 OGSI Notifications

There is no OGSA-DAI specific support for the OGSI notification mechanisms defined by the various Grid Service notification portTypes beyond those provided by the Globus Toolkit 3.0 Factory, ServiceGroup, and Grid Service implementations.

4.5 OGSA-DAI Sample Clients

This release contains an end-to-end example client. This client allows a user to work with the DAIServiceGroupRegistry (DAISGR), which has replaced the old GDSR, the GDSF, and the GDS. Example documents are provided for working with the latter two and errors are returned to the client.

This release also includes a GUI client demonstrator which is an example of a more complex client which can be used to visualise the interactions between services, databases and clients.

In addition, sample clients are provided to produce test MySQL, Oracle and Xindice databases.

A preview of a client toolkit is also available for people building clients.

4.6 OGSA-DAI Test Framework

The OGSA-DAI test framework uses the end-to-end command line client provided. This program utilises the first GDSF it discovers from the DAISGR and does not look for any others. When using the test framework, it is important, therefore, to ensure that there is only one GDSF, and that it creates an appropriate GDS for the data resource type to be tested (i.e. MySQL or XMLDB).

4.7 Security and Role Mapping

This release allows an X.509 certificate name to be mapped to a database username, providing basic security and role mapping functionality.

This release also supports Message Level Security as provided by Globus Toolkit 3. Please refer to the user documentation and install notes for details on setting up Secure Grid Data Service Factories.

4.8 Restrictions on the DAIServiceGroupRegistry (DAISGR)

The DAISGR implementation does not provide support for client authorisation. This means that any client/service can register and de-register services. In addition, any client/service can search any DAISGR service for data or information on its registered services.

5 Changes Since Previous Release

The following changes have been made between the Release 3.0 and Release 3.1:

5.1 Changes to the DAIServiceGroupRegistry (DAISGR)

No significant changes.

5.2 Changes to the GridDataServiceFactory (GDSF)

There have been minor changes to the schema for data resource configuration files. A `basePerformDocumentSchema` element has been added which, in conjunction with the `location` element and the `activityMap` elements allow you to use relative or absolute paths to the activity schema files.

In addition, the metadata extraction functionality has been modified so that extracting database structural schema gives more consistent results between the different DBMS

Message level security and Secure GDSF support has been added.

5.3 Changes to the GridDataService (GDS)

The SQL Activity code has been refactored to provide better cross-compatibility with different DBMS. It will also allow non-standard functionality of new DBMS to be supported more easily in the future.

Numerous changes have been made to the Activity and Engine framework which the GDS uses to process Perform Documents to improve the performance and scalability. In particular:

- The request context is now viewed through an ActivityContext interface from within an Activity implementation. This provides more straightforward retrieval of inputs, outputs, and data resource implementation.
- Request status now uses the observable pattern allowing one or more observers to be attached. The GDS uses an observer to update the request status service data element whenever the request status changes.
- Threading has been optimised so that additional threads are not spawned for simple requests.
- The response document includes a <request> element that specifies the request status. If there has been an error processing the request, a "cause" attribute is included specifying the name of the activity from which the error spawned.

In addition the following prototype functionality has been added:

- A prototype CSV file activity has been implemented which allows a subset of SQL queries to be run on CSV files
- A prototype BinX file activity has been implemented which allows XPath queries to be run on BinX described files

These are not production features and are liable change significantly in future releases.

5.4 Additional Functionality – Client Toolkit

A technical preview of a Client Toolkit API has been included. This allows applications to be developed without requiring the developer or user to write perform documents – these are built for them by the toolkit. It also provides a number of useful features to control security and data input and output.

5.5 Other changes since R3.0

- The major bug where multiple perform documents requests to a GDS caused it to hang has been fixed. (Release 3.0.1)
- Full Message Level Security support has been added (Release 3.0.2)
- DeliverToStream reports additional status information (Release 3.0.2)
- XMLTransformActivity can now use inline stylesheets (Release 3.0.2)
- EngineImpl checks if currently executing request is due to finish in short order and blocks request rather than throwing an exception (Release 3.0.2)

5.6 Summary of changes between R2.5 and R3.0

The following changes were made between the R2.5 release and the R3.0 release:

- Perform documents have been changed significantly to make them easier to understand
 - the request, execute and terminate elements have been phased out. Termination of requests is now carried out by destroying the GDS
 - deliverToResponse has been removed. Use webRowSetStream to achieve equivalent results. DeliverToStream has been added
 - DataStore, item, and itemCursor elements have been added to support the DataStore activity
- Data is now streamed through the Zip / GZip transformation activities
- A deliverToStream activity has been added along with a DeliverToStreamServlet
- A DataStore activity has been added to cache parameters and results
- If the DBMS and JDBC driver support setReadOnly on a connection the query will not allow an Update to occur

- Stored Procedure Activities can be specified with either webRowSetStream (query) or resultStream (update).
- Relational Resource Management Schema has been changed to be more consistent with other SQL Activity schema.
- SQLParameters and expressions can be provided by input streams
- Collection and Resource Management activities for XML dbs now let you list collections and resources
- GDSF configuration files have been consolidated into two files
- Schema for data resource configuration changed to allow database product types and non-standard database connection URIs to be used
- Minor changes to service data elements returned by GDSF
- DBMS metadata is now specified in config files, dynamic metadata is generated by dataresource implementations
- DAISGR lists all OGSA-DAI services registered
- DAISGR allows any service implementing a DAI portType to register
- Database schema can be extracted from a GDS.
- An internationalisation framework has been added .
- A GUI client demonstrator has been added.
- The system test framework has been updated.
- Improved performance and general bug fixes.
- Improved installation procedure.
- Improved end-to-end example client.
- Refactored document schema extensively.
- Refactored namespace extensively (including date versioning).
- The directory structure of the release has been changed slightly.
- Compatible with the Globus Toolkit 3.0 release. Not compatible with previous releases.
- Example xml documents have been updated and made consistent.
- Errors notified in the User Documentation, in particular, the Installation Guide, have been addressed.
- Sample clients are provided to produce test MySQL, Oracle and Xindice databases.
- New example xml document sets are included. These are in ogsadai/examples.

5.7 Summary of changes between R2 and R2.5

The following changes have been made between the R2 release and the R2.5 release:

- GDSR replaced by DAISGR
- GDSF rewritten and configuration documents refactored
- Data Resource Implementations introduced to manage access to data resources
- Engine streams data transparently
- Engine has new perform/response
- Support for parameter activities added
- Support for XSLT transform activities added
- Support for zip and gzip compression activities added
- Support for basic XMLDB collection and resource management added
- SQL **UPDATE** and **INSERT** commands now return a result of form "**<X> rows changed**" to a client.
- Example scripts have been corrected and made consistent.
- Errors in the User Documentation, in particular, the Installation Guide, have been addressed.
- Sample clients are provided to produce test MySQL and Xindice databases.

6 Bug Fixes

Bugs fixed in earlier releases:

1. The “repeated results” bug has been fixed. Results are now cleared from the Context once they have been delivered to the client.
2. The “logging bug” in SQL results has been fixed to reduce memory usage when handling large result sets.
3. The intermittent loss of the penultimate set of 20 blocks from a result has been fixed.
4. The memory leak in service destruction has been fixed.
5. The **NullPointerException** thrown when delivering to an input stream via the Grid DataTransportPort (GDT) portType has been fixed.
6. The loss of the penultimate 20 results for a JDBC query during high demand queries has been fixed.
7. The **OutOfMemoryException** after instantiating more than 300 GDS service instances has been fixed
8. The **InvalidDataConversion** error when retrieving **CHAR**-type data from DB2 has been fixed.
9. The “Unsupported CLOB type” error arising in in JDBC-to-Sun Rowset XML conversion has been fixed – “CLOB type” is now supported.
10. The **ClassCastException** occurring during **SQLUpdateActivity** when a parameter passes a byte array into the SQL statement has been fixed. Byte arrays are now supported by all **SQL*Activities**.
11. In the GDSF-Config file (for example **gdsf_config.xml**) you can now specify **dataManager productType** as **'DB2'** (rather than **'DB'**).
12. You no longer have to specify a port number in the physicalLocation in GDSF-Config files (for example **gdsf_config.xml**).

Bugs fixed since R2.5:

13. Support for namespace bindings in XPath queries has been fixed (broken in P2R2.5).
14. The DAISGR now lists all OGSADAI services registered and allows services implementing any OGSADAI portType to register.
15. The schema has been extended to allow any database productType to be specified in the data resource configuration file.
16. Non-standard database connection URIs (e.g. Oracle, or without port number) are now supported.

Bugs fixed since R3.0:

17. File Lock not released when exception is raised is fixed
18. NullPointerException thrown when MySQL server not started now fixed
19. Repeated execution of perform documents with same delivery output now works without causing the GDS to hang.
20. Many multiple requests to a GDS no longer causes a memory leak.
21. Various synchronisation and threading issues in Activity.java, SynchronisedPipe.java and RunAheadHandler have been resolved
22. XMLRowsetInputStream now uses new SynchronisedPipe implementation
23. XMLUtilities no longer wraps long lines which was causing failures with DB2
24. Documents with no outputs now execute properly
25. The demonstrator client now extracts SDE names properly
26. SDEs for GDSF are now consistent with GDS
27. Fixed exception caused when uploading null timestamp values using BulkLoad
28. XMLUtilities.xmlFileToDOM no longer deletes the xml file if it is invalid
29. XMLRowsetOutputStream now converts strings to boolean values correctly
30. A more informative error is logged if an incorrect database URI is used
31. Database schema extraction in Oracle no longer throws an SQLException

7 Known Problems and Remedies

The known problems and remedies in Release 3.1 are as follows.

7.1 Service Access after Tomcat Startup

Sometimes an error will occur if attempting to communicate with the DAISGR or the GDSF immediately after Apache Tomcat has been re-started. If using a command-line client then this error will can be manifested in a way similar to the following:

```
[07/30/2003 10:16:13:663 ]
org.globus.ogsa.impl.core.handle.HandleHelper
[getGSR:124] ERROR: [org.apache.axis.AxisFault] AxisFault
...
Error: org.gridforum.ogsi.InvalidHandleFaultType:
http://localhost:8080/ogsa/services/ogsadai/DAIServiceGroupRegistry
```

In the Tomcat logs such an error is manifested in a way similar to the following:

```
[07/30/2003 09:27:18:772 ] org.apache.axis.Message [writeTo:443]
ERROR: java.io.IOException:
java.net.SocketException: Socket closed

[07/30/2003 09:27:08:649 ]
org.globus.ogsa.impl.core.handle.HandleHelper
[getGSR:124] ERROR: [org.apache.axis.AxisFault] AxisFault
faultCode:
{http://schemas.xmlsoap.org/soap/envelope/}Server.userException
```

This problem can be avoided if after starting up Tomcat you visit the `ogsa/services` page using a Web browser (see OGSA-DAI-USER-INSTALL) and select to view either the DAISGR or GDSF WSDL document. The client will then work fine.

7.2 Performing XUpdate queries with Xindice 1.0 and Java 1.4 (JDK 1.4)

On Windows 2000/NT/XP, the Xindice 1.0 XUpdate service does not operate correctly under JDK1.4. Whenever a query matches some existing data, and an update should be performed, a **QueryProcessingError** is thrown. This problem can be resolved by modifying one line of the Xindice 1.0 source code and re-building, as described below:

Edit the file:

```
<XINDICE_HOME>\java\src\org\apache\xindice\core\xupdate\XObjectImpl.j
ava
```

Change the code at line 105 from,

```
return (NodeList) _xobj.nodeset();
```

to,

```
return (NodeList) _xobj.nodelist();
```

Re-build Xindice by running 'build.bat' (assuming you have set the JAVA_HOME environment variable within the batch file) from the Xindice home directory. This will create the Xindice.jar again in the lib folder.

7.3 UNIX `setenv` Scripts

The software used to unpack the release may introduce problems into the `<OGSA-DAI>/setenv.sh` and `<OGSA-DAI>/setenv.csh` scripts which means they will not run without modification.

The problem is evident under `sh` if you see the output analogous to the following:

```
$ source setenv.sh
sh:
: command not found
sh: setenv.sh: line 5: syntax error near unexpected token `do
'
sh: setenv.sh: line 5: `do
```

The problem is evident under `csh` if you see the following:

```
$ source setenv.csh
set: Syntax error
```

Cause

This problem can arise if using WinZip to download the OGSA-DAI distribution and then unpacking and running under UNIX or Linux, so beware – either use a UNIX-based or Linux-based unzip/untar product – for example gnutar – or make modifications to the files as described.

Solutions – To correct the `<OGSA-DAI>/setenv.sh` file:

1. Open the `<OGSA-DAI>/setenv.sh` file within an editor.
2. Remove all the carriage-return characters `^M` and replace with standard newlines.
3. Save the file.

Solutions – To correct the `<OGSA-DAI>/setenv.csh` file:

1. Open the `<OGSA-DAI>/setenv.sh` file within an editor.
2. Remove all the carriage-return characters `^M` and replace with standard newlines.
3. Save the file.

7.4 Problem With Using Sun's `tar` Command

On Sun's Solaris you get a "TAR file corrupted" error when trying to untar using Sun's version of `tar`.

Cause

It is known that Sun's `tar` cannot handle long pathnames, which are often present in Java package structures

Solution

Use `gnutar` instead.

7.5 Location of ogsadai.jar

Under a manual installation of OGSADAI, the `ogsadai.jar` file must be in

```
<TOMCAT>/webapps/ogsa/WEB-INF/lib
```

and not

```
<TOMCAT>/common/lib
```

Putting the jar file in a different directory will lead to a number of non-obvious errors.

7.6 Location of Message Bundle

The base english message bundle used to map error codes to messages is packaged in `ogsadai.jar`. On some installations, you may experience problems where this file is not being located properly.

This can be resolved by using a suitable zipfile utility, e.g. WinZip on windows or `unzip` on unix, to extract the file `uk.org.ogsadai.common.ogsadai_messages_en.properties`, which should be copied to `<TOMCAT>/webapps/ogsa/WEB-INF/etc`

7.7 Oracle JDBC driver renaming

Some versions of Oracle come with a JDBC driver as a set of classes contained in a Zip file, e.g. `classes12.zip`. This must be renamed to have a `Jar` extension and placed in the `<TOMCAT>/webapps/ogsa/WEB-INF/lib` directory, e.g. `oracle-jdbc.jar`.

7.8 Java Virtual Machine fails

If using very large result sets, the Java Virtual Machine may run out of memory. This can be alleviated by increasing the memory heap sizes for the VM using the `-Xms<size>` flag.

7.9 Errors when sending perform documents to DB2 database

DB2 cannot handle expressions which contain newline characters, so the following example will generate an error:

```
<expression>
  select * from littleblackbook where id > ? and id <= ?
</expression>
```

To work around this, make sure the perform document is changed so the expression is written as:

```
<expression>select * from littleblackbook where id=10</expression>
```

This will affect various activities including queries, parameters and prepared statements.

7.10 Request Termination time too short – client times out

The end-to-end client may timeout before an asynchronous request has had time to complete. It is possible to alter the amount of time before the request times out.

Add the following lines before the call to `griddataservice.perform` in `Client.java`:

```
org.apache.axis.client.Stub s = (org.apache.axis.client.Stub)
                                griddataservice;
s.setTimeout(time);
```

where time is in milliseconds, or 0 for infinity.

8 Frequently Asked Questions

There follows some frequently asked questions.

8.1 No GDSFs Found

If running the OGSA-DAI client you get an exception analogous to the following:

```
Error: java.lang.Exception: No factories registered at the GDSR at
http://localhost:8080/ogsa/services/ogsadai/DAIRegistry
```

And you can see both the `DAIServiceGroupRegistry` and `GridDataServiceFactory` in the list of services available at `http://localhost:8080/ogsa/services`.

Then check that in the `<CONTAINER>/webapps/ogsa/WEB-INF/etc/registrationList.xml` file, a `gdsfRegistration` element exists which has an attribute `gsh` with value `(http://localhost:8080/ogsa/services/ogsadai/DAIServiceGroupRegistry)`

8.2 Configuring Xindice data resources

In the GDSF configuration and database role map files, make sure you omit the preceding slash from the database name, i.e. for a collection called `/db/littleblackbook` use:

ExampleDatabasesRoles.xml

```
<DatabaseRoles>
  <Database name="db/littleblackbook">
    <User dn="No Certificate Provided" userid="" password="" />
  </Database>
</DatabaseRoles>
```

GDSFConfig.xml

```
<gdsf:location name="myXMLDBMS">
  <gdsf:dataManager name="xmlDBMS" productType="Xindice"/>
  <gdsf:physicalLocation>localhost:4080</gdsf:physicalLocation>
  <gdsf:roleMap
configuration="doc/examples/RoleMap/ExampleDatabaseRoles.xml"
class="uk.org.ogsadai.common.rolemap.SimpleFileRoleMapper"/>
  <gdsf:path name="myXMLDB">
```

```
<gdsf:data name="db/littleblackbook"/>
</gdsf:path>
<gdsf:validSchemes>
  <gdsf:validScheme>xindice</gdsf:validScheme>
</gdsf:validSchemes>
</gdsf:location>
```

8.3 Synchronous Request problems performing SQL Updates

If you get an error such as “GDS Engine – Problem while constructing sqlUpdateStatement; WebRowSetStream not specified and no alternative given” make sure you have defined a `<resultStream>` (not `<webRowSetStream>`) within your `<sqlUpdateStatement>` in your GDS Perform document.

8.4 Setting log4j properties

You can change the log4j logging options by specifying a log4j.properties file using the CATALINA_OPTS variable before restarting Tomcat, e.g. on UNIX:

```
export CATALINA_OPTS=
"-Dlog4j.properties=file:///path/to/my/log4j.properties"
```

8.5 OGSA-DAI and the Globus Service Container

You can attempt to install OGSA-DAI to work under the built-in Globus Service Container by:

- Following the OGSA-DAI installation instructions (OGSA-DAI-USER-INSTALL).
- But, setting the **CATALINA_HOME** environment to the Globus Toolkit 3 Alpha 4 / Beta distribution directory.

Acknowledgements

The following people have contributed to or helped the OGSA-DAI Project:

EPCC: Ali Anjomshoaa, Mario Antonioletti, Rob Baxter, Stephen Booth, Daragh Byrne, Neil Chue Hong, Matthew Egbert, Ally Hume, Mike Jackson, Amy Krause, Jeremy Nowell, Charaka Palansuriya, Kira Smyllie, Tom Sugden, Martin Westhead.

IBM UK: Andrew Borley, Michael Coleman, Brian Collins, Jonathan Davies, Neil Hardman, George Hicken, Alan Knox, Simon Laws, James Magowan, Manfred Oevers, Paul Taylor.

IBM US: Susan Malaika, Inderpal Narang.

NeSC: Malcolm Atkinson, Mark Cavanagh, Alastair Knowles, Gill Maddy, Lee McLeod, David McNicol.

Oracle UK: Ian Carney, Dave Pearson, Andrew Whaley.

University of Manchester: Nedim Alpdemir, Alvaro Fernandes, Desmond Fitzgerald, Tasos Gounaris, Norman Paton.

University of Newcastle: Arijit Mukherjee, Paul Watson.

Greg Riccardi (Florida State University), Juha Marjomaa (University of Oulu), Beth Plale (Indiana University), Deepti Kodeboyina (Indiana University), Ewa Deelman (ISI), Gurmeet Singh (ISI), Vijay Dialani (University of Southampton), Ed Zaluska (University of Southampton), Guy Rixon (University of Cambridge).

Apologies if there are any omissions or mistakes in the above list.

This work is supported by the UK e-Science Core Programme, whose support we are pleased to acknowledge.