

GWD-R
GGF DAIS Working Group

Editors

Mario Antonioletti, University of Edinburgh
Amy Krause, University of Edinburgh
Shannon Hastings, Ohio State University
Stephen Langella, Ohio State University
Susan Malaika, IBM
James Magowan, IBM
Simon Laws, IBM
Norman W Paton, University of Manchester

Category: INFORMATIONAL

19 September 2003

Grid Data Service Specification: The Relational Realisation

Status of This Memo

This memo provides information to the Grid community regarding the specification of Grid Database Services. The specification is presently a draft for discussion. It does not define any standards or technical recommendations. Distribution is unlimited.

Copyright Notice

Copyright © Global Grid Forum (2003). All Rights Reserved.

Abstract

Data management systems are central to many applications across multiple domains, and play a significant role in many others. Web services provide implementation neutral facilities for describing, invoking and orchestrating collections of networked resources. The Open Grid Services Architecture (OGSA) extends Web Services with consistent interfaces for creating, managing and exchanging information among Grid Services, which are dynamic computational artifacts cast as Web Services. Both Web and Grid service communities stand to benefit from the provision of consistent, agreed service interfaces to data management systems. Such interfaces must support the description and use of data management systems using Web Service standards, taking account of the design conventions and mandatory features of Grid Services. This document presents a specification for a collection of data access interfaces for relational data resources, which extends interfaces defined in the Grid Data Service Specification [GDSS], which in turn is based on the OGSA Data Services proposal [Data Services]. The specification is presented for discussion within the Global Grid Forum (GGF) Database Access and Integration Services (DAIS) Working Group, with a view to the document evolving to become a proposed recommendation. There are several respects in which the current proposal is incomplete, but it is hoped that the material included is sufficient to allow an informed discussion to take place concerning both its form and substance.

Contents

Abstract.....	1
1. Introduction.....	3
2. Notational Conventions.....	3
3. Specification Overview.....	3
3.1 Scope of specification.....	3
3.2 Mapping to Data Service model.....	4
3.3 Relationships with other specifications.....	4
4. DataDescription portTypes.....	5
4.1 RelationalDescription.....	5
4.2 RowsetDescription.....	6
5. DataAccess portTypes.....	6
5.1 SQLAccess.....	6
5.2 RowsetAccess.....	8
5.3 DBOperationAccess.....	8
6. DataFactory portTypes.....	10
6.1 SQLFactory.....	10
6.2 RowsetSelectionFactory.....	11
6.3 DBOperationFactory.....	12
7. Security Considerations.....	13
8. Conclusion.....	13
Editor Information.....	13
Contributor Information.....	14
Acknowledgements.....	14
Intellectual Property Statement.....	15
Full Copyright Notice.....	15
References.....	15

1. Introduction

Data access plays an important role in many Grid applications. In general, data access involves both retrieval and manipulation of data, which may be stored or virtualised using a range of paradigms. This implies the need for a flexible framework for request evaluation, and close integration with functionality for managing and moving data retrieved from, or for insertion into, a data resource.

This document presents a specification for a collection of data access interfaces for relational data resources. A relational data resource is taken to mean a data source/sink, together with any associated management framework, that exhibits capabilities that are characteristic of relational database systems, e.g., can be queried using or updated SQL. The interfaces instantiate the framework provided by the OGSA Data Services proposal [Data Services], in that interfaces are categorized according to the support they provide for data description, data access, data service creation and data management. As such, this document should be read in conjunction with the OGSA Data Services proposal and the generic Grid Data Service Specification [GDSS], which defines various portTypes that are extended in this specification. All of these documents assume some familiarity with the Open Grid Services Infrastructure (OGSI) [OGSI]. The specification does not mandate how the interfaces are composed into services. The proposed interfaces may be used in isolation or in conjunction with others.

2. Notational Conventions

The key words “MUST,” “MUST NOT,” “REQUIRED,” “SHALL,” “SHALL NOT,” “SHOULD,” “SHOULD NOT,” “RECOMMENDED,” “MAY,” and “OPTIONAL” are to be interpreted as described in RFC-2119 [RFC2199]

The GWSDL for the specification described in this document is available from the DAIS Working Group Web Site

Prefix	Namespace
dais	http://www.ggf.org/namespaces/2003/06/DAIS
gsa	WS-Agreement namespace URI
gwsdl	http://www.ggf.org/namespaces/2003/03/gridWSDLExtensions
http	http://www.w3.org/2002/06/wsdl/http
ogsi	http://www.ggf.org/namespaces/2003/03/OGSI
sd	http://www.ggf.org/namespaces/2003/02/serviceData
sqlxml	http://standards.iso.org/iso/9075/2002/12/ (temporary location)
webr	http://java.sun.com/xml/ns/jdbc/ (contents subject to change)
wsdl	http://schemas.xmlsoap.org/wsdl/
wsp	http://schemas.xmlsoap.org/ws/2002/12/policy
xsd	http://www.w3.org/2001/XMLSchema
xsi	http://www.w3.org/2001/XMLSchema-instance

3. Specification Overview

3.1 Scope of specification

This document extends the interfaces presented in the *Grid Data Service Specification* [GDSS] to allow access to and describe relational data resources and aligns these interfaces with the base types provided in the *OGSA Data Services* [Data Services] Document. The relational data resources are assumed to be composed of databases and tables, which contain rows and columns. In addition, common relational resources such as stored procedures are also described.

3.2 Mapping to Data Service model

3.2.1 DataDescription portTypes

DataDescription portTypes allow a description of virtualized data to be provided via Service Data Elements (SDEs). No operations are defined within these interfaces. The model independent specification for these is given in the *Grid Data Service Specification* [GDSS] document. Here they are extended to provide a description of relational based data resources. These are the main points of extension for relational data resources:

- *RelationalDescription*: provides information about relationalSchemas that describe databases, tables, columns, columnTypes, and keys that a Data Service may virtualize. It also describes indexes, storedProcedures, userDefinedTypes, and userDefinedFunctions. The portType makes available information about these constructs by querying the constructs themselves.
- *RowsetDescription*: provides information about a particular instance of a query result that a Data Service may virtualize. This portType will make available information about the structure representing the query result as well as other relevant data.

These interfaces are described in Section 4.

3.2.2 DataAccess portTypes

DataAccess portTypes allow relational data resources to be modified through insertion or updates or queried through an appropriate query language. Note that a Data Service does not know the nature of the underlying data model so the right interfaces have to be invoked in order to access it correctly. When a Data Service is created the supporting *DataAccess* interfaces may be specified using WS-Agreement [WS Agreement].

- *SQLAccess*
- *RowsetAccess*
- *DBOperationAccess*

These are covered in more detail in Section 5.

3.2.3 DataFactory portTypes

The *DataFactory* portTypes allows data represented in relational data resources, usually as the result of a query or update, to be virtualized (instantiated as services). The specializations in this instance thus deal with the type of expression that can be passed to a *DataFactory* to virtualize the results. The properties and interfaces that will be supported by these virtualizations will be specified using WS-Agreement [WS Agreement]. *DataFactory* specializations are:

- *SQLFactory*
- *RowsetSelectionFactory*
- *DBOperationFactory*

These are covered in more detail in Section 6.

3.3 Relationships with other specifications

DAIS does not propose to provide its own query/update languages for relational based data resources and query result sets (rowsets). Instead, it acts as a conduit for existing relational based query and update languages to be conveyed to the appropriate data resources, in this instance relational based data resources or a data resource that supports relational type queries. As such DAIS relies on existing relational based query and update languages. In the current

version of this document interface support is provided for languages based on the following standards:

- SQL: is an ISO standard defining a language for querying and updating relational data resources in the context of relational tables. There is work underway to define SQL 2003 that includes SQL/XML for manipulating and generating XML from relational data.
- WebRowSet: is one possible dataFormat for response from sqlAccess operations. (See <http://java.sun.com/xml/ns/jdbc/webrowset.xsd>)

4. DataDescription portTypes

The DataDescription portTypes allow metadata for data virtualizations to be made available. DataDescription portTypes are provided for use with relational database virtualizations and for result set virtualizations.

4.1 RelationalDescription

4.1.1 Service Data Declarations

- *relationalSchema*: Describes the schema of the relational data, such as tables, columns, column types, keys, views, which are specific to the relational environment

```
<sd:serviceData name="relationalSchema"
  type="relationalSchemaType"
  minOccurs="1" maxOccurs="1"
  mutability="mutable"
  modifiable="false"
  nillable="true" />
```

- *indexes*: Describes the names and definitions of any indexes available.

```
<sd:serviceData name="indexes"
  type="IndexType"
  minOccurs="1" maxOccurs="unbounded"
  mutability="mutable"
  modifiable="false"
  nillable="true" />
```

- *storedProcedures*: Describes the name, input and output types of stored procedures available.

```
<sd:serviceData name="storedProcedures"
  type="StoredProcedureType"
  minOccurs="1" maxOccurs="unbounded"
  mutability="mutable"
  modifiable="false"
  nillable="true" />
```

- *userDefinedTypes*: Describes the names and definitions of the user defined types deployed.

```
<sd:serviceData name="userDefinedTypes"
  type="UserDefinedTypesType"
  minOccurs="1" maxOccurs="unbounded"
  mutability="mutable"
  modifiable="false"
  nillable="true" />
```

- *userDefinedFunctions*: Describes the names and definitions of the user defined functions deployed.

```
<sd:serviceData name="userDefinedFunctions"
  type="UserDefinedFunctionsType"
  minOccurs="1" maxOccurs="unbounded"
  mutability="mutable"
  modifiable="false"
  nillable="true"/>
```

- *trigger*: Describes the name and definition of the triggers deployed in the data resource.

```
<sd:serviceData name="trigger"
  type="TriggerType"
  minOccurs="1" maxOccurs="unbounded"
  mutability="mutable"
  modifiable="false"
  nillable="true"/>
```

4.2 RowsetDescription

4.2.1 Service Data Declarations

- *rowSchema*: The schema for representing the RowSet.

```
<sd:serviceData name="rowSchema"
  type="rowSchemaType"
  minOccurs="1" maxOccurs="1"
  mutability="mutable"
  modifiable="false"
  nillable="false"/>
```

- *noOfRows*: The number of rows within the Rowset.

```
<sd:serviceData name="noOfRows"
  type="xsd:int"
  minOccurs="1" maxOccurs="1"
  mutability="mutable"
  modifiable="false"
  nillable="false"/>
```

5. DataAccess portTypes

The dataFormats SDE lists all possible dataFormats for results from Access portType operations.

5.1 SQLAccess

This allows access to the underlying data by means of SQL expressions.

5.1.1 Service Data Declarations

All those defined for a Factory within Grid Data Service Specification.

- *languageCapabilities*: Describes the dialect of SQL supported by this data resource.

```
<sd:serviceData name="languageCapabilities"
  type="LanguageCapabilitiesType"
  minOccurs="1" maxOccurs="unbounded"
  mutability="mutable"
  modifiable="false"
  nillable="false"/>
```

5.1.2 Operations

In all operations below the expression input is a single SQL expression, i.e. only one result arises. New portTypes should be specified to allow block execution.

5.1.2.1 SQLAccess::sqlQuery

Direct an SQL Query to relational virtualizations of the data where the result of the query is returned conforming to the requested dataFormat.

Input

- expression: the SQL query string that is to be run on the data resource.

```
<xsd:element name="sqlExpression" type="sqlExpressionType"/>
```

- dataFormat: the format selected from SDE dataFormats, which the return type will conform to.

```
<xsd:element name="dataFormat" type="dataFormatType"/>
```

Output

- dataSet: output the results in the dataFormat.

Fault(s)

- InvalidQuery: the supplied SQL is syntactically incorrect or fails during evaluation.
- Fault: any other fault.

5.1.2.2 SQLAccess::sqlUpdate

Direct an SQL update to the data.

Input

- expression: the SQL update string that is to be run on the data resource.

```
<xsd:element name="sqlExpression" type="sqlExpressionType"/>
```

- dataFormat: the format selected from SDE dataFormats, which the return type will conform to.

```
<xsd:element name="dataFormat" type="dataFormatType"/>
```

Output

- *updateCount*: the updateCount returned from the sqlUpdate.

Fault(s)

- *InvalidQuery*: the SQL supplied to do the update is incorrect or an SQL query statement is being supplied to this operation.
- *Fault*: any other fault.

5.1.2.3 sqlAccess::sqlExecute

Direct an SQL statement to the data, which can be an update or a query.

Input

- expression: the SQL string that is to be run on the data resource.

```
<xsd:element name="sqlExpression" type="sqlExpressionType"/>
```

- *dataFormat*: the format selected from SDE *dataFormats*, which the return type will conform to.

```
<xsd:element name="dataFormat" type="dataFormatType"/>
```

Output

- *dataSet.output* the results in the *dataFormat*

Fault(s)

- *InvalidQuery*: the SQL supplied to do the update is incorrect or an SQL query statement is being supplied to this operation.
- *Fault*: any other fault.

5.2 RowsetAccess

This allows access to the underlying data by means of rows.

5.2.1 Service Data Declarations

- *nTupleFormat*: The format and data types of the tuples

```
<sd:serviceData name="nTupleFormat"
  type="nTupleFormatType"
  minOccurs="1" maxOccurs="unbounded"
  mutability="mutable"
  modifiable="false"
  nillable="false"/>
```

5.2.2 Operations

5.2.2.1 RowsetAccess::getNextNTuples

Return a specified number of tuples from a service that virtualizes a result set

Input

- *startPosition* the position of the first tuple to be returned (1st tuple is position 1)

```
<xsd:element name="startPosition" type="xsd:int"/>
```

- *count*: the number of tuples.

```
<xsd:element name="count" type="xsd:int"/>
```

Output

- *tuples*: the result of the update operation.

Fault(s)

- *InvalidCount*: Cannot return that number of tuples.
- *Fault*: any other fault.

5.3 DBOperationAccess

This allows access to the underlying data by means of executing a Database Operation. Database operations include the execution of a stored procedure or a stand-alone user defined function.

5.3.1 Service Data Declarations

All those defined for a Factory within Grid Data Service Specification. No additional SDE's required. There are SDE's within the RelationalDescription portType that allow the discovery of available DB Operations.

5.3.2 Operations

5.3.2.1 DBOperationAccess::getReturnValue

Get the return value of the DBOperation. The DBOperation is evaluated when getReturnValue is requested.

Input

- DBOperation: the Database operation to be performed

```
<xsd:element name="DBOperation" type="DBOperationType"/>
```

- operationParameters: the parameters required to complete the Database Operation.

```
<xsd:element name="operationParameters" type="operationParametersType"/>
```

Output

- dataSet: output the results.

Fault(s)

- InvalidParameters: the supplied parameters are invalid for this Operation.
- Fault: any other fault.

5.3.2.2 DBOperationAccess::getResultSets

Retrieve any result sets arising from DBOperation that has already executed such as a stored procedure.

Input

- DBOperation: the Database operation to be performed

```
<xsd:element name="DBOperation" type="DBOperationType"/>
```

- operationParameters: the parameters required to complete the Database Operation.

```
<xsd:element name="operationParameters" type="operationParametersType"/>
```

Output

- *resultSets*: the resultSets returned by operation.

Fault(s)

- InvalidParameters: the supplied parameters are invalid for this Operation.
- *Fault*: any other fault.

5.3.2.3 DBOperationAccess::getOutputParameter

Retrieve an output parameter arising from a DBOperation that has already executed, e.g., a stored procedure or a user-defined function.

Input

- DBOperation: the Database operation to be performed

```
<xsd:element name="DBOperation" type="DBOperationType"/>
```

- `operationParameters`: the parameters required to complete the Database Operation.

```
<xsd:element name="operationParameters" type="operationParametersType"/>
```

Output

- *output*: The output parameter from the operation

Fault(s)

- `InvalidParameters`: the supplied parameters are invalid for this Operation.
- *Fault*: any other fault.

6. DataFactory portTypes

6.1 SQLFactory

A WS-Agreement document holding an SQL Query, the interfaces to be created and behavior specification is passed to `SQLFactory::createService`. The factory will create a service fulfilling the desired behavior, exposing the desired portTypes and representing the results of the SQL Query.

6.1.1 Service Data Declarations

All those defined for a Factory within Grid Data Service Specification.

- *languageCapabilities*: Describes the dialect of SQL supported by this data resource.

```
<sd:serviceData name="languageCapabilities"
  type="LanguageCapabilitiesType"
  minOccurs="1" maxOccurs="unbounded"
  mutability="mutable"
  modifiable="false"
  nillable="false"/>
```

In addition the `SQLFactory` portType defines the following initial set of service data value elements:

```
<sd:staticServiceDataValues>
  <dais:proposedAgreement>
    <dais:SQLFactoryAgreement>
      <wsp:OneOrMore wsp:Usage="wsp:Required">
        <dais:readable gsa:Negotiability="gsa:Fixed" .../>
        <dais:updateable gsa:Negotiability="gsa:Fixed" .../>
      </wsp:OneOrMore/>
      <wsp:OneOrMore wsp:Usage="wsp:Required">
        <dais:supportsInterface dais:qname="DataDescription"
          sa:Negotiability="gsa:Fixed" .../>
        <dais:supportsInterface dais:qname="DataAccess"
          sa:Negotiability="gsa:Fixed" .../>
        <dais:supportsInterface dais:qname="DataFactory"
          sa:Negotiability="gsa:Fixed" .../>
        <dais:supportsInterface dais:qname="DataManagement"
          sa:Negotiability="gsa:Fixed" .../>
        <dais:supportsInterface dais:qname="RowsetDescription"
          sa:Negotiability="gsa:Fixed" .../>
      </wsp:OneOrMore/>
    </dais:SQLFactoryAgreement>
  </dais:proposedAgreement>
</sd:staticServiceDataValues>
```

```

        <dais:supportsInterface dais:qname="RowsetAccess"
                               sa:Negotiability="gsa:Fixed" .../>
        <wsp:OneOrMore/>
    </dais:SQLFactoryAgreement>
</dais:proposedAgreement>
</sd:staticServiceDataValues>

<xsd:complexType name="SQLFactoryAgreementType">
  <xsd:complexContent>
    <xsd:extension base="dais:DAISBaseAgreementType">
      <xsd:sequence>
        <!-- SQL query used to populate the Data Service -->
        <xsd:element name="sqlExpression" type="sqlExpressionType"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:element name="sqlFactoryAgreement" type="sqlFactoryAgreementType"/>

```

The createServiceExtensibility SDE of the OGSF Factory portType and the supportedAgreement SDE of the WS-Agreement AgreementFactory portType should be populated with the information required to support the use of WS-Agreement documents of type SQLFactoryAgreementType.

6.1.2 Operations

6.1.2.1 SQLFactory::createService

Create a new Data Service that corresponds to the results of an SQL Query.

6.2 RowsetSelectionFactory

A WS-Agreement document holding the RowSet index, the interfaces to be created and behavior specification is passed to RowsetSelectionFactory::createService. The factory will create a service fulfilling the desired behavior, exposing the desired portTypes and representing the requested rows.

6.2.1 Service Data Declarations

All those defined for a Factory within Grid Data Service Specification. No additional SDE's required. However, the RowsetSelectionFactory portType defines the following initial set of service data value elements:

```

<sd:staticServiceDataValues>

  <dais:proposedAgreement>
    <dais:RowsetSelectionFactoryAgreement>
      <wsp:OneOrMore wsp:Usage="wsp:Required">
        <dais:readable gsa:Negotiability="gsa:Fixed" .../>
        <dais:updateable gsa:Negotiability="gsa:Fixed" .../>
      <wsp:OneOrMore/>

      <wsp:OneOrMore wsp:Usage="wsp:Required">
        <dais:supportsInterface dais:qname="DataDescription"
                               sa:Negotiability="gsa:Fixed" .../>
        <dais:supportsInterface dais:qname="DataAccess"
                               sa:Negotiability="gsa:Fixed" .../>
        <dais:supportsInterface dais:qname="DataFactory"
                               sa:Negotiability="gsa:Fixed" .../>
        <dais:supportsInterface dais:qname="DataManagement"
                               sa:Negotiability="gsa:Fixed" .../>
      </wsp:OneOrMore>
    </dais:RowsetSelectionFactoryAgreement>
  </dais:proposedAgreement>
</sd:staticServiceDataValues>

```

```

        <dais:supportsInterface dais:qname="RowsetDescription"
                               sa:Negotiability="gsa:Fixed" .../>

        <dais:supportsInterface dais:qname="rowsetAccess"
                               sa:Negotiability="gsa:Fixed" .../>

        <wsp:OneOrMore/>
    </ dais:RowsetSelectionFactoryAgreement>
</dais:proposedAgreement>
</sd:staticServiceDataValues>

<xsd:complexType name="RowsetSelectionFactoryAgreementType">
  <xsd:complexContent>
    <xsd:extension base="dais:DAISBaseAgreementType">
      <xsd:sequence>
        <!-- The start position for
        <xsd:element name="startPosition" type="xsd:int"/>
        <xsd:element name="count" type="xsd:int"/>

      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:element name="rowsetSelectionFactoryAgreement"
              type="RowsetSelectionFactoryAgreementType"/>

```

6.2.2 Operations

6.2.2.1 rowsetSelectionFactory::createService

Create a new Data Service, which corresponds to the RowSet selected.

6.3 DBOperationFactory

A WS-Agreement document holding the Database Operation desired, the interfaces to be created and behavior specification is passed to DBOperationFactory::createService. The factory will create a service fulfilling the desired behavior, exposing the desired portTypes and representing the requested rows.

6.3.1 Service Data Declarations

All those defined for a Factory within Grid Data Service Specification. No additional SDE's required. However, the DBOperationFactory portType defines the following initial set of service data value elements:

```

<sd:staticServiceDataValues>

  <dais:proposedAgreement>
    <dais:DBOperationFactoryAgreement>
      <wsp:OneOrMore wsp:Usage="wsp:Required">
        <dais:readable gsa:Negotiability="gsa:Fixed" .../>
        <dais:updateable gsa:Negotiability="gsa:Fixed" .../>
      <wsp:OneOrMore/>

      <wsp:OneOrMore wsp:Usage="wsp:Required">
        <dais:supportsInterface dais:qname="DataDescription"
                               sa:Negotiability="gsa:Fixed" .../>
        <dais:supportsInterface dais:qname="DataAccess"
                               sa:Negotiability="gsa:Fixed" .../>
        <dais:supportsInterface dais:qname="DataFactory"
                               sa:Negotiability="gsa:Fixed" .../>
        <dais:supportsInterface dais:qname="DataManagement"

```

```

                sa:Negotiability="gsa:Fixed".../>

        <dais:supportsInterface dais:qname="RowsetDescription"
                sa:Negotiability="gsa:Fixed".../>

        <dais:supportsInterface dais:qname="RowsetAccess"
                sa:Negotiability="gsa:Fixed".../>

        <wsp:OneOrMore/>
    </ dais:DBOperationFactoryAgreement>
</dais:proposedAgreement>
</sd:staticServiceDataValues>

<xsd:complexType name="DBOperationFactoryAgreementType">
  <xsd:complexContent>
    <xsd:extension base="dais:DAISBaseAgreementType">
      <xsd:sequence>
        <!-- The DB operations -->
        <xsd:element name="DBOperation" type="DBOperationType"/>
        <xsd:element name="operationParameters" type="operationParametersType"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:element name="dbOperationFactoryAgreement"
  type="DBOperationFactoryAgreementType"/>

```

6.3.2 Operations

6.3.2.1 DBOperationFactory::createService

Create a new Data Service, which corresponds to a Database Operation such as Stored Procedure or User Defined Function, see *Grid Data Service Specification* [GDSS].

7. Security Considerations

The Relational Realization of a Grid Data Service will use standard Grid Security mechanisms as specified by OGSA Security working group combined with standard ways of relating Grid credentials and authorities to resource access rights. The assumption is that these standards will also indicate how to make information related to authentication, authorization security etc available.

8. Conclusion

This document has discussed a specialization of the portTypes defined in the *Grid Data Service Services* [GDSS] Specification and the additional capabilities required to properly address relational based data resources. This is work in progress and feedback is welcomed on this document.

Editor Information

Mario Antonioletti,
 EPCC,
 University of Edinburgh,
 James Clerk Maxwell Building,
 Mayfield Road,
 Edinburgh EH9 3JZ,
 United Kingdom.

Shannon Hastings,
Ohio State University,
333 W. Tenth Ave.,
Columbus OH, 43210,
USA.

Amy Krause,
EPCC,
University of Edinburgh,
James Clerk Maxwell Building,
Mayfield Road,
Edinburgh EH9 3JZ,
United Kingdom.

Stephen Langella,
Ohio State University,
333 W. Tenth Ave.,
Columbus OH, 43210,
USA.

Simon Laws,
IBM United Kingdom Limited,
Hursley Park,
Winchester,
Hampshire, SO21 2JN,
United Kingdom.

Susan Malaika,
IBM Corporation,
Silicon Valley Laboratory,
555 Bailey Avenue,
San Jose, CA 95141,
USA.

James Magowan,
IBM United Kingdom Limited,
Hursley Park,
Winchester,
Hampshire, SO21 2JN,
United Kingdom.

Norman W. Paton,
Department of Computer Science,
University of Manchester,
Oxford Road,
Manchester M13 9PL,
United Kingdom.

Contributor Information

Vijay Dialani, University of Southampton.
Greg Riccardi, Florida State University.

Acknowledgements

The DAIS Working Group of the Global Grid Forum is active, and many people have contributed to discussions within the group in recent months, including but not limited to: Bill Allcock, Vijay

Dialani , Dieter Gawlick, Allen Luniewski , Sastry Malladi, Inderpal Narang, Steve Tuecke , Jay Unger, Paul Watson and Martin Westhead.

Intellectual Property Statement

The GGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the GGF Secretariat.

The GGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights, which may cover technology that may be required to practice this recommendation. Please address the information to the GGF Executive Director.

Full Copyright Notice

Copyright (C) Global Grid Forum (2003). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the GGF or other organizations, except as needed for the purpose of developing Grid Recommendations in which case the procedures for copyrights defined in the GGF Document process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the GGF or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE GLOBAL GRID FORUM DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE."

References

[Data Services]

I.Foster, S.Tuecke, J.Unger. *OGSA Data Services*, August 14, 2003. See: https://forge.gridforum.org/docman2/ViewProperties.php?group_id=42&document_content_id=733

[GDSS]

M. Antonioletti, M. Atkinson, S. Malaika, S. Laws, N. W. Paton D. Pearson and G. Riccardi. *Grid Data Service Specification*. DAIS-WG Informational Draft, 9th Global Grid Forum, 19th September 2003.

[OGSI]

S. Tuecke, K. Czajkowski, I. Foster, J. Frey, S. Graham, C. Kesselman, D. Snelling, P. Vanderpilt, Open Grid Services Infrastructure, Version 1.0, <http://www.gridforum.org/ogsi-wg>, March 13, 2003.

[RFC2199]

S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*, Internet Engineering Task Force, RFC 2119, <http://www.ietf.org/rfc/rfc2119.txt>, March 1997.

[WS-Agreement]

K.Czajkowski, A.Dan, J.Rofrano, S.Tuecke, M.Xu. *Agreement-based Grid Service Management*, Version 0, June 12, 2003