Editors:
S. Newhouse, LeSC
J. MacLaren, eSNW
Auguse, 2003

Resource Usage Service (RUS)

Status of this Memo

This document provides information to the community regarding the specification of the Resource Usage Service (RUS). Distribution of this document is unlimited. This is a DRAFT document and continues to be revised.

Abstract

The Open Grid Services Architecture (OGSA) provides an infrastructure for virtualising resources of many types (compute, storage, software, networking etc.) as Grid Services. The infrastructure for building these basic grid services is being defined elsewhere within the Global Grid Forum. Although mechanisms will exist for defining these services it is unlikely that any sustainable infrastructure will be provided by any non-research organization without financial compensation. For Grid Services to be provided on demand (i.e. to provide the utility infrastructure that has always been the vision of the Grid) organizations will want to be paid for providing these resources.

The purpose of this document is therefore to describe the service data and ports needed to define the Resource Usage Service (RUS) such that is can provide a basic infrastructure to support the auditing and monitoring capability for the resources consumed by OGSA services in the Grid and allows entities within the Grid to extract information from the service on potentially aggregated resource use.

GLOBAL GRID FORUM

office@gridforum.org
www.ggf.org

Full Copyright Notice

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.

Explicit statements about IPR should not be included in the document, because including a specific claim implies that the claim is valid and that the listed claims are exhaustive. Once a document has been published as a final GFD there is no mechanism to effectively update the IPR information. Authors should instead provide the GGF secretariat with any explicit statements or potentially relevant claims.

Contents

1 Introduction

This document focuses on the development of an OGSA compatible Resource Usage Service (RUS) to enable the recording and retrieval of consumed resource information. This service is needed to provide information about service use to a variety of Grid entities:

- The service manager who wishes to examine utilisation across their resources.
- A service that wishes to charge for the use of the consumed resources.
- A virtual organisation that wishes to monitor resource activity within their Grid.

These requirements occur across many Grid projects [SJN:others?] including activity taking place within the UK e-science programme.

Within the UK, RUS is currently being developed to support two projects. The first is too support the UK e-Science Grid where the management team of this particular Virtual Organisation (VO) wishes to monitor the consumed resources (e.g. job activity through the Globus). The second application is as part of the UK e-Science Core programme project – A Market for Computational Services – is to record the consumed resources within OGSA services to generate charging information.

This document will define the Grid Service interface needed to define RUS and forms part of the Resource Usage Service (**RUS**) Working Group within the Global Grid Forum (**GGF**). One other group within the GGF that has particular relevance is the Usage Record (**UR**) Working Group.

2 Overview

2.1 Architecture

The Resource Usage Service's primary requirement is to store records relating to the consumption of resources during the invocation of a Grid Service. This is currently being targeted to support activity within the Grid Economic Services Architecture Working Group in recording consumed resources during service invocation. However, it is also being used to collect resources within a virtual organisation – the UK e-Science Grid. There are therefore two primary functions: the upload of resource usage information into, and the extraction of resource usage data from the service. A possible implementation of this service, but by no means the only approach, is to use a conventional relational database to store the consumed resource information.

There are several possible approaches to exposing the resource usage data contained within the RUS:

- Expose the usage records within the Service Data Elements.
- Expose the underlying database (if this is the implementation mechanism) through mechanisms such as OGSA-DAI.
- Provide operations to explicitly extract usage records from the underlying storage mechanism.

A primary concern within RUS is to maintain confidentiality of the usage information that will be held within the service. It is therefore essential that the mechanism used to expose the data within RUS is able to support the access policy.

There are no mechanisms within the current OGSI draft to specify access control mechanisms to SDE's nor, it could be argued, should there be if the SDE is to represent the public internal state of the Grid Service. If these are introduced SDE provides an alternative mechanism to exposing the RUS data. Exposing the underlying data through OGSA-DAI will therefore rely on the role-based security model within its interface. Within RUS we wish to restrict access to data based on the contents of that data, e.g. user A should only be able to see records written by user A. It is not clear how many RDBMS incorporate ACL at this level of granularity and if this functionality is reflected in OGSA-DAI. In the short term we will pursue the final option allowing access to explicitly defined.

The RUS may be deployed within many different usage scenarios. In order to monitor the resources consumed during a service invocation the RUS instance MUST last beyond the lifetime of service it is monitoring. A single RUS instance MAY therefore be instantiated:

- within a service container (if the administrator wishes to only trust local service instances)
- within a real organisation (if the RUS is to collect overall usage within an organisation)
- within a virtual organisation (to capture resource usage across many resources)

2.2 Definitions and Notational Conventions

Throughout this document we will use the term ‘user’ as a generic term for a client to a RUS which may be an interactive client or a service instance interacting with a RUS instance.

The use of the key words “MUST,” “MUST NOT,” “REQUIRED,” “SHALL,” “SHALL NOT,” “SHOULD,” “SHOULD NOT,” “RECOMMENDED,” “MAY,” and “OPTIONAL” in this document are to be interpreted as described in RFC-2119 [RFC 2119].

This specification uses namespace prefixes throughout; they are listed in Table 1. Note that the choice of any namespace prefix is arbitrary and not semantically significant.

Table 1: Prefixes and namespaces used in this specification.

Prefix	Namespace
ogsi	"http://www.gridforum.org/namespaces/2003/03/OGSI"
gwsdl	"http://www.gridforum.org/namespaces/2003/03/gridWSDLExtensions"
sd	"http://www.gridforum.org/namespaces/2003/03/serviceData"
wsdl	“http://schemas.xmlsoap.org/wsdl/”
http	"http://www.w3.org/2002/06/wsdl/http"
xsd	"http://www.w3.org/2001/XMLSchema"
xsi	“http://www.w3.org/2001/XMLSchema-instance”
ur	"http://www.gridforum.org/namespaces/2003/??/UR" ???
rus	"http://www.gridforum.org/namespaces/2003/??/RUS" ???
gesa	"http://www.gridforum.org/namespaces/2003/??/GESA" ???

2.3 Scope

In the remainder of this document we define the structure of the RUS including:

- Service Data Elements: The additional SDE’s that are needed to describe RUS’s capability.
- Service Interface Definition: The operations needed to support interaction with the Grid Service.
- Implementation: Observations on implementing the operations
- Other Issues: Our goal is to build on the OGSI and related standards. In some areas these may need clarification or to be developed further.

3 The Resource Usage Service (RUS)

The RUS is designed to support the upload and extraction of information relating to resource usage.

3.1 Service Data Elements

The SDE's provided by the RUS are in addition to those defined within the GSS.

SDE	Occurrence	Provided By…	Comment
Resource	1	Service Admin	Recorded resources
ResourceManager	0+	Service Admin	Grid users who have the right to view activity on their resources
ServiceManager	0+	Service Admin	Grid users who has the right to view all activity within RUS.

All economic SDE's are contained with a SDE's of type:

```
<serviceData name="rus:ResourceUsageSDE">
   …
</serviceData>
```

3.1.1 Resources

The primary purpose of the RUS is to store information relating to the consumed resources. Not all RUS instances will be capable of storing all resource fields and the fields that are supported are described within the `resources` element and `resourceField` elements.

```
<rus:resources>
  <rus:resourceField name="Jobname" />
  <rus:resourceField name="NodeCount />
</rus:resources>
```

The textual information given in the `name` attribute MUST correspond to one of the elements specified within the Usage Record XML Format [UR-XML].

[SJN: The elements from the UR XML will require some content, e.g. xsd:string, xsd:int, etc. I can't see how the schema can be re-used without putting dummy values in here or allowing empty elements in the UR schema. This has been raised with UR-WG and if this is allowed then:

```
<rus:resources>
  <ur:Jobname />
  <ur:NodeCount />
</rus:resources>
```

]

3.1.2 Resource Manager

The resource manager has the right to view all activity relating to the resource for which they have managerial responsibility. For Usage Records based around a machine, the `resourceType` attribute will be set to "machine", and the `resourceName` attribute in this element MUST match the 'MachineName' element in the record. This will include any Usage record of type job, reservation, process and maybe license. It is not clear how this would match to the name of a network or license; the example below is a suggestion.

```
<rus:roles>
  <rus:ResourceManager
DN = "/C=UK/O=eScience/OU=Imperial/L=LeSC/CN=steven newhouse"
resourceName = "clusterA" resourceType = "machine" />
  <rus:ResourceManager
DN = "/C=UK/O=eScience/OU=Imperial/L=LeSC/CN=steven newhouse"
resourceName = "*.doc.ic.ac.uk" resourceType = "machine" />
  <rus:ResourceManager
DN = "/C=UK/O=eScience/OU=Manchester/L=eSNW/CN=jon maclaren"
resourceName = "NetNorthWest" resourceType = "network" />
</rus:resources>
```

Standard wild card attributes may also be used to match on hostnames.

3.1.3 Service Manager

The service manager has the right to view all activity stored within the RUS.

```
<rus:roles>
  <rus:ServiceManager
DN= "/C=UK/O=eScience/OU=Imperial/L=LeSC/CN=steven newhouse" />
</rus:resources>
```

3.2 Service Interface Definition

The RUS has two main functionalities to: upload and retrieve information. The information moved into or out of the RUS is contained within an XML document the syntax of which is

3.2.1 RUS::updateUsageRecords

The `updateUsageRecords` is operation that imports the contents of the resource usage document into the RUS.

Input

- *Resource Usage Records*: This XML document describes the resources that have been consumed by the client entity.

Output

- *long*: This is a count of the number of records which could not be updated.

Faults

- As defined in the OGSI specification document.

The `updateUsageRecords` operation differs from the `insertUsageRecords` operation in that it will attempt to overwrite an existing entry in the RUS if such a record already exists. The fields that define uniqueness for this operation are:

- GlobalUserID – **please note that this is now part of an XMLDSIG element**
- GlobalJobID

- MachineName
- SubmitHost

[JM: What about records of type “link”, and maybe “license”?]

3.2.2 RUS::insertUsageRecords

The `insertUsageRecords` is operation that imports the contents of the resource usage document into the RUS.

Input

- *Resource Usage Records*: This XML document describes the resources that have been consumed by the client entity.

Output

- *long*: This is a count of the number of records which could not be inserted.

Faults

- As defined in the OGSI specification document.

The `insertUsage` operation differs from the `updateUsage` operation in that it will insert a new usage record into the RUS.

3.2.3 RUS::extractUsageByGlobalJobID

Enable the client to find all usage records relating to the specified GlobalJobID within the RUS. Access to these records is allowed if the client DN is that of:

- the entity that placed the record in the RUS
- a relevant Resource Manager
- a specified Service Manager
- the User as specified in the GlobalUserID element of the Usage Record

Input

- *GlobalJobID* : All resource usage records relating to this GlobalJobID that meet the specified access rules should be returned to the client.
- *Scope (optional)* : A mechanism to limit the scope of the returned documents e.g. Xpath?

Output

- *Resource Usage Records*: This XML contains the resources usage records contained within RUS for this query.

Faults

As defined in the OGSI specification document.

3.2.4 RUS::extractUsageByGlobalUserID

Enable the client to find all usage records relating to the specified GlobalUserID within the RUS. Access to these records is allowed if the client DN is that of:

- the entity that placed the record in the RUS
- a relevant Resource Manager
- a specified Service Manager
- the User as specified in the GlobalUserID element of the Usage Record

Input

- *GlobalUserID* : All resource usage records relating to this GlobalUserID that meet the specified access rules should be returned to the client.

- *Scope (optional)* : A mechanism to limit the scope of the returned documents e.g. Xpath?

Output

- *Resource Usage Records*: This XML contains the resources usage records contained within RUS for this query.

Faults

- As defined in the OGSI specification document.

3.2.5 RUS::extractUsageByMachineName

Enable the client to find all usage records relating to the specified MachineName within the RUS. Access to these records is allowed if the client DN is that of:

- the entity that placed the record in the RUS
- a relevant Resource Manager
- a specified Service Manager
- the User as specified in the GlobalUserID element of the Usage Record

Input

- *MachineName* : All resource usage records relating to this MachineName that meet the specified access rules should be returned to the client.
- *Scope (optional)* : A mechanism to limit the scope of the returned documents e.g. Xpath?

Output

- *Resource Usage Records*: This XML contains the resources usage records contained within RUS for this query.

Faults

- As defined in the OGSI specification document.

3.2.6 RUS::extractUsageBySubmitHost

Enable the client to find all usage records relating to the specified SubmitHost within the RUS. Access to these records is allowed if the client DN is that of:

- the entity that placed the record in the RUS
- a relevant Resource Manager
- a specified Service Manager
- the User as specified in the GlobalUserID element of the Usage Record

Input

- *MachineName* : All resource usage records relating to this MachineName that meet the specified access rules should be returned to the client.
- *Scope (optional)* : A mechanism to limit the scope of the returned documents e.g. Xpath?

Output

- *Resource Usage Records*: This XML contains the resources usage records contained within RUS for this query.

Faults

- As defined in the OGSI specification document.

3.2.7 RUS::extractRecords

Enable the client to find all usage records relating to a more complicated set of requirements. Access to these records is allowed if the client DN is that of:

- the entity that placed the record in the RUS
- a relevant Resource Manager
- a specified Service Manager
- the User as specified in the GlobalUserID element of the Usage Record

Input

- *searchTerm (1)*: All resource usage records relating to the search criteria that meet the specified access rules should be returned to the client. The search term should be specified as part of an XPath/XQuery string. More precisely, the query string is prefixed with "/urwg:UsageRecords/ur:wg/UsageRecord[" and suffixed with "]", plus extra terms to implement security. The strings "urwg" and "ds" are set to correspond to the usage record and digital signature namespaces for all queries. At the moment, there is no way to specify any extra namespaces.

Output

- *Resource Usage Records*: This XML contains the resources usage records contained within RUS for this query.

Faults

- As defined in the OGSI specification document.

3.2.8 RUS::deleteRecords

Enable the client to delete all usage records that match the specified criteria. Access to this operation is allowed if the client DN is that of:

- a specified Service Manager

Input

- *searchTerm (1)*: All resource usage records relating to the search criteria that meet the specified access rules should be returned to the client. Defined in the same way as *searchTerm* in extractRecords, Section 3.2.6.

Output

- *recordCount (1):* The number of records removed.

Faults

- As defined in the OGSI specification document.

3.3 Other Issues

There is a general requirement to define a mechanism to specify and support access control policies within SDEs & ports.
The operations for extracting information from RUS need to be refined.

4 The Resource Usage Record

The RUS uploads (and provides) record of resource usage through an XML document using a format developed by the Usage Record Working Group (UR-WG) in their 'Usage Record XML Format' document [In draft] and should be consulted in reading this section. Several issues need to be clarified:

- Introduction of a UsageRecords element that encapsulates multiple UsageRecord elements
- The ability to declare the presence of an element (i.e. empty body)

For a detailed description of the terms the reader is referred to [UR-WG website]

The 'Resource Usage Record' XML document references in earlier sections has the general form:

```
<ur:UsageRecords>
  <ur:UsageRecord>
    <UserIdentity>
      <ur:LocalUserId>sjn5</ur:LocalUserId>
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <X509Data>
       <X509SubjectName>/C=UK/O=eScience/OU=Imperial/L=LeSC/CN=steven
newhouse</X509SubjectName>
        </X509Data>
      </ds:KeyInfo>
    </UserIdentity>
    <JobIdentity>
      <GlobalJobID>The job ID</GlobalJobID>
    <JobIdentity>
    <MachineName>eric.mvc.mcc.ac.uk</MachineName>
    <SubmitHost>eric.mvc.mcc.ac.uk</SubmitHost>
    <Processors>4</Processors>
    <cputime units="MB" metric="MAX">30</cputime>
    <memory units="GB" metric="TOTAL">2</memory>
  </ur:UsageRecord>
</ur:UsageRecords>
```

This record indicates that the user sjn5 has a global identity based on their distinguished name used 4 processors for an average CPU time of 30 minutes over all the processors and a total of 2GB memory.

5 Security Considerations

This document assumes the availability of the security provisions from the OGSI. There is a clear need within this service to specify fine-grained access control to an operation and potentially to control access to the contents of the SDE.

6 Editor Information

Steven Newhouse
London e-Science Centre
...
Email:

Jon MacLaren
e-Science NorthWest
Kilburn Building
University of Manchester
Manchester M13 9PL
United Kingdom
Email: jon.maclaren@man.ac.uk

7 Contributors

We gratefully acknowledge the contributions made to this specification by the following people:

8 Acknowledgements

The creation of this document has been supported by:

- The Computational Markets project funded under the UK e-Science Core programme by the Department of Trade and Industry and the Engineering and Physical Sciences Research Council (http://www.lesc.ic.ac.uk/markets/)

The editors would like to acknowledge the contribution of Anthony Mayer (LeSC) and Asif Saleem (LeSC) for discussions relating to this document.

9 References

9.1 Normative References

[RFC 2119]
Key words for use in RFCs to Indicate Requirement Levels, S. Bradner, Author. Internet Engineering Task Force, RFC 2119, March 1997. Available at http://www.ietf.org/rfc/rfc2119.txt

[WSDL 1.2]
Web Services Description Language (WSDL) Version 1.2, Published W3C Working Draft, World Wide Web Consortium. Available at http://www.w3.org/TR/wsdl12/

[WSDL 1.2 DRAFT]

Web Services Description Language (WSDL) Version 1.2, W3C Working Draft 3 March 2003, World Wide Web Consortium. Available at http://www.w3.org/TR/2003/WD-wsdl12-20030303

[OGSI 1.2 DRAFT]
Open Grid Services Infrastructure (OGSI) Version 1.0, GGF Working Draft 33 June 2003, Global Grid Forum. Available from: http://forge.gridforum.org/projects/ogsi-wg

9.2 Informative References

[Globus Overview]
Globus: A Toolkit-Based Grid Architecture, I. Foster, C. Kesselman. In [Grid Book], 259-278.

[Grid Anatomy]
The Anatomy of the Grid: Enabling Scalable Virtual Organizations, I. Foster, C. Kesselman, S. Tuecke. International Journal of High Performance Computing Applications, 15 (3). 200-222. 2001. Available at http://www.globus.org/research/papers/anatomy.pdf

[Grid Book]
The Grid: Blueprint for a New Computing Infrastructure, I. Foster, C. Kesselman, eds. Morgan Kaufmann, 1999.

[Grid Physiology]
The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, I. Foster, C. Kesselman, J. Nick, S. Tuecke. Globus Project, 2002. Available at http://www.globus.org/research/papers/ogsa.pdf

[JAX-RPC]
Java™ API for XML-Based RPC (JAX-RPC)**.**
http://java.sun.com/xml/jaxrpc/docs.html

[Web Services Book]
Building Web Services with Java: Making Sense of XML, SOAP, WSDL, and UDDI, s. Graham, S. Simeonov, T. Boubez, G. Daniels, D. Davis, Y. Nakamura, R. Neyama. Sams, 2001.

[WSIF]
Welcome to WSIF: Web Services Invocation Framework, http://www.apache.org/wsif/

10 Normative XSD and WSDL Specifications

This section contains the full normative XSD and WSDL definitions for everything described in this document. The definitions in this section MUST be considered normative, if there are any discrepancies between the definitions in this section and those portions described in other sections above.