Grid Resource Specification

Authors:
Savas Parastatidis
Paul Watson
Jim Webber

North East Regional e-Science Centre
School of Computing Science
University of Newcastle, Newcastle-upon-Tyne, NE1 7RU
United Kingdom

Grid-Resource Specification
Document version: 1.0
Date published: 01 October 2003

The latest version of this document and supporting material can be found at
http://www.neresc.ac.uk/ws-gaf/grid-resource

Abstract

This specification defines the *Grid Resource Identifier* for globally naming resources and the *Grid Resource Metadata* document which may contain metadata information for those resources.

Author contact information:

Savas Parastatidis (Savas.Parastatidis@newcastle.ac.uk)[1]
Paul Watson (Paul.Watson@newcastle.ac.uk)[1]
Jim Webber (Jim.Webber@arjuna.com)[2]

[1] School of Computing Science
University of Newcastle upon Tyne
Newcastle upon Tyne, NE1 7RU
United Kingdom

[2] Arjuna Technologies Ltd.
Nanotechnology Centre, Herschel Building
Newcastle upon Tyne, NE1 7RU
United Kingdom

Grid-Resource Specification

Motivation

One of the main distinguishing characteristics of the Grid application domain is the need to share resources [1] (e.g., data, instruments, computers, humans, etc.) between organisations. For a resource to be shared it must be uniquely identified and, in many cases, to have associated metadata that describes it. The Web Services community does not address the Grid's requirement for resource identification and associated metadata beyond an organisation's boundary since this is requirement is particular to the Grid, and is not part of the lower-level Web Services infrastructure.

This specification defines the *Grid Resource Identifier* for globally naming resources and the *Grid Resource Metadata* document which may contain metadata information for those resources. The specification does not address the means by which such documents are populated. Furthermore, it is left to the Grid application designers to define the semantics of the metadata information in Grid Resource Metadata documents.

Grid Resource Identifier

Once the decision has been made to globally identify a resource, it is important that the resource is given a unique name with which it can be identified throughout the Grid application domain. There have been proposals for naming and uniformly providing access to resources, like the REpresentational State Transfer (REST) [2] model. However, since REST depends on HTTP it is protocol specific and hence unsuitable for heterogeneous systems like the Grid.

The Uniform Resource Name (URN) proposal [3] is a technology-agnostic scheme for identifying resources. The Grid Resource specification utilises URNs as follows:

> **A Grid Resource Identifier (GRI) is a URN that globally, uniquely, and everlastingly identifies a resource. A resource cannot have more than one GRI.**

Since the GRI is unique and everlasting, it can be stored in databases and printed in journals, in the safe knowledge that it could be used, at any time in the future, to locate any services offering that resource. In this, it fulfils a similar role to an LSID [4] (and indeed, an LSID could be used as a GRI).

Please note that although a resource may have a finite lifetime, its GRI is everlasting, which is an attribute derived directly from the URN scheme.

Examples of GRIs:

```
urn:dais:dataset:b4136aa4-2d11-42bd-aa61-8e8aa5223211
urn:instruments:telescope:nasa:hubble
urn:physics:colliders:cern
urn:lsid:pdb.org:1AFT:1
```

Grid Resource Metadata Document

In addition to the identity of a Grid resource, it may also be desirable to expose metadata information about it. A Grid Resource Metadata (GRM) document is used as the means to capture such metadata information. A GRM provides an extensible mechanism for the addition of application-specific metadata, and so this specification provides only a minimal set of metadata information elements.

Often in the Grid it is desirable to know about the lifetime of a shared resource. Also, consumers wishing to access or operate on a resource may need to locate services that may be aware of this resource. For example, services may allow a data resource to be

returned in a message or may provide the means to extract information from a resource (e.g. through a query). Consequently, the lifetime of a resource and the service endpoints, that may be aware of it, can be part of that resource's GRM document.

The following elements should or may be included in a GRM document:

- **gri**. (mandatory) Its value is the GRI of the resource with which the metadata is related.
- **type**. (optional) In situations where the identified resource could be represented by, or is, an XML document, this element might be used to provide information about the XML type of that document (e.g., the "http://www.dais.org/dataset" complexType for a DAIS dataset).
- **lifetime.** (optional) It provides a hint about the lifetime of the identified resource.
- **endpoint.** (optional) It provides information about the endpoint of a service that may be aware of the identified resource. The semantics of what it means for a service to "be aware of a resource" are application-specific. There may be zero to many endpoint elements in a GRM document. An endpoint element consists of the following sub-elements:
  - **wsa:EndpointReference**. A WS-Address [5] to the service.
  - **lifetime.** A hint about the period of time when the addressed service will be aware of the resource.

Except from the **gri** element, all the other elements are optional.

The services for which endpoints are included in a GRM document do not all need to have the same interface. Some applications that already utilise services with a specific WSDL interface may not be able to take advantage of the availability of new services with different interfaces. However, various approaches may be implemented for choosing which of the advertised services can be used. An application may be statically linked to a specific service endpoint at design time, or bound to it at runtime. In more advanced schemes, an application could be made aware of multiple WSDL interfaces at design time, and choose to bind to the service endpoint which offers one of them at run-time.

Populating a GRM Document

This specification does not define the means through which the metadata in a GRM document may be populated. It is expected that this would become possible through appropriate registries, peer-to-peer systems, or other higher-level services (e.g., OGSAI-DAI [6]).

Intended Use

Although this specification focuses on the identification of resources for the Grid and their related metadata, it is noted that exposing resources to the Grid should be avoided where possible. Exposing many resources to the Grid may lead to a great number of interdependencies that cross organisation boundaries, and lead to tight coupling and poor scalability. The Web Services Architecture (WSA) [7] approach to inter-organisation integration through the exchange of rich, coarse-grained messages should, we believe, be the preferred way to building Grid applications [8]. A Grid Resource Metadata document should only be used when it is absolutely necessary to expose metadata about globally identified resources, and where common Web Services practices are not sufficient to meet Grid application requirements (e.g., extremely large datasets for which "pass-by-reference" is appropriate, telescopes that must be uniquely identified, globally unique and shared gene sequences, etc.).

Grid Resource Metadata Document XML Schema

The XML Schema for a GRM document is shown in Listing 1.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
      elementFormDefault="qualified"
      attributeFormDefault="unqualified"
      targetNamespace="http://www.neresc.ac.uk/2003/10/grm"
      xmlns:grm="http://www.neresc.ac.uk/2003/10/grm"
      xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing">
  <xs:import
  namespace="http://schemas.xmlsoap.org/ws/2003/03/addressing"
  schemaLocation="http://schemas.xmlsoap.org/ws/2003/03/addressing"/>

  <xs:element name="lifetime">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="valid-from" type="xs:dateTime"
            minOccurs="0" maxOccurs="1"/>
        <xs:element name="valid-until" type="xs:dateTime"
            minOccurs="0" maxOccurs="1"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:complexType name="Endpoint">
    <xs:sequence>
      <xs:element ref="wsa:EndpointReference"
            minOccurs="1" maxOccurs="1"/>
      <xs:element ref="grm:lifetime"
            minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="GRM">
    <xs:sequence>
      <xs:element name="gri" type="xs:anyURI"
          minOccurs="1" maxOccurs="1"/>
      <xs:element name="type" type="xs:QName"
          minOccurs="0" maxOccurs="1"/>
      <xs:element ref="grm:lifetime"
          minOccurs="0" maxOccurs="1"/>
      <xs:element name="endpoint" type="grm:Endpoint"
          minOccurs="0" maxOccurs="unbounded"/>

      <xs:any namespace="##other" processContents="lax"
          minOccurs="0" maxOccurs="unbounded" />

    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

Listing 1: XML Schema for a WS-Resource

Acknowledgements

We would like to thank Dave Ingham (Arjuna Technologies) for his feedback.

References

1. Foster, I., et al., *The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration*. 2002, Global Grid Forum.
2. Fielding, R.T., *Architectural Styles and the Design of Network-based Software Architectures*. 2000, University of California, Irvine.
3. Moats, R., *RFC4121*, in *URN Syntax*. 1997: http://www.ietf.org/rfc/rfc2141.txt.
4. *I3C Life Sciences Identifiers (LSIDs)*: http://www.i3c.org/wgr/ta/resources/lsid/docs/index.htm.
5. *Web Services Addressing (WS-Addressing)*: http://msdn.microsoft.com/ws/2003/03/ws-addressing.
6. *Open Grid Services Architecture - Data Access and Integration (OGSA-DAI)*: http://www.ogsa-dai.org.uk.
7. *Web Services Architecture*: http://www.w3.org/TR/ws-arch.
8. Parastatidis, S., et al., *A Grid Application Framework based on Web Services Specifications and Practices*. 2003: http://www.neresc.ac.uk/projects/gaf.