University of Victoria
Faculty of Engineering
Fall/Winter 2008 Work Term Report

The Differences in Globus 4.2

University of Victoria
Dept. of Physics and Astronomy
Elliot 219
Victoria, BC

Cole Uhlman
0536633
cuhlman@uvic.ca
cuhlman@engr.uvic.ca
Computer Engineering
January 9, 2009

Supervisor's Approval: To be completed by Co-op Employer

I approve the release of this report to the University of Victoria for evaluation purposes only. The report is to be considered

NOT CONFIDENTIAL CONFIDENTIAL (select one)

Signature ____________________ Name (print) ____________________

Position ____________________ E-Mail ____________________

Fax # ____________________ Date ____________________

If a report is deemed CONFIDENTIAL, a non-disclosure form signed by an evaluator will be faxed to the employer. The report will be destroyed following evaluation. If the report is NOT CONFIDENTIAL, it will be returned to the student following evaluation

TABLE OF CONTENTS

SUMMARY

Globus is a software suite for use with computing grids. It facilitates communication between and management of grid resources. The newest version of Globus is version 4.2, an upgrade from version 4.0.

Globus 4.2 has presented a number of changes to its services including new features, tools, updates to software packages and compatibility changes. Globus 4.2 has also changed some of its Java libraries, which will require developers to update their web services code.

The main drawback is that Globus 4.2 is not compatible with Globus 4.0, so upgrading might not be worthwhile. However, some of the new tools could be very useful and would be interesting to explore.

GLOSSARY

Ant – A software tool made by Apache for automating software builds.

C WS Core – The container that facilitates web services written in C.

container – A program to provide a runtime environment for web services. Manages resources and service execution.

GAR – Grid ARchive. A file containing all the files and information needed for the container to deploy a service.

GPT – Grid Packaging Tools.

GRAM – Grid Resource Allocation and Management. A set of web services to manage jobs on a Grid.

IODBC – Database software.

Java WS Core – The container that facilitates web services written in Java.

JVM – Java Virtual Machine.

namespace – A data format that holds multiple pieces of information to disambiguate from a datum with the same name in a different namespace.

node – A computer in the grid, used either for computation or scheduling.

OpensSSL – An open source version of the Security Sockets Layer protocol.

OS – Operating System.

RHEL – Red Hat Enterprise Linux.

RLS – Replica Location Service. A registry to keep track of the mapping between logical names of data and names of replicas kept on a physical storage system.

SimpleCA – Software for obtaining grid certificates from the Canadian grid authority. A certificate is required to run any grid resources.

SOAP – A framework for exchanging XML messages.

sudo – superuser do. A Linux feature that allows normal users to have limited root privilages.

WS – Web Service. Software to provide interaction between computers over a network.

WSDL – Web Services Description Language. An XML based protocol for describing web services.

WSN – Web Services Notification. Tools to allow web services to send messages to each other.

WSRF – Web Services Resource Framework. A specification for resource management in web services.

XML – eXtensible Markup Language. A programming language. It is used for web services

yum – Yellow Dog Update Modifier. An update tool and package installer for Linux.

INTRODUCTION

Some experiments require a large amount of computing power. This is provided by grid computing. A grid consists of several cluster computers communicating over a WAN.

The clusters use a local resource manager, such as Torque, to delegate tasks to its worker nodes. At a higher level, a metascheduler accepts tasks from clients and delegates them to an appropriate cluster. Between each node, a middleware is required to support secure communication. The most common middleware in grid computing is Globus. It provides the means for job submission from the client to the metascheduler as well as communication between the metascheduler and the individual clusters.

Globus version 4.2 has made some important changes from version 4.0. This report will explore these changes and their impact on performance and implementation.

DISCUSSION

1. Installation and Setup

Globus 4.2 brought about a few changes to the setup procedures. These will be highlighted in the next sections.

1.1 Software Prerequisites

There are a few changes to the software requirements of Globus. Firstly, updates to the Java WS Core have changed the Java requirements from a minimum version of 1.4.2 to 1.5.0. Ant version 1.6.2 is the new Ant requirement. Additionally, the SimpleCA certificate system now requires the use of OpenSSL.

An optional prerequisite for Globus 4.0 was IODBC to enable the use of RLS. Globus 4.2 includes a built-in IODBC driver, so this is no longer required. Also, in Globus 4.0, the sudo prerequisite for GRAM services was optional, whereas in Globus 4.2 it is required.

1.2 Install Issues

When installing Globus version 4.2.1 there is an error in the default install. When running configure, you will need to specify the "--prefix" argument with the actual installation directory. Installation will otherwise fail because the default wasn't updated from version 4.2.0 and was left as "/usr/local/globus-4.2.0/" rather than "/usr/local/globus-4.2.1.1/".

Another noteworthy problem is that despite it not being listed in the Globus documentation as a prerequisite, perl-XML-Parser is another requirement to install Globus.

This is not a large problem, since the error message is clear. The package can be installed using yum on RHEL-based linux distributions.

2. Java WS Core

The Java WS Core is the container that facilitates Java-based web services. There have been a few changes to its implementation in version 4.2 of Globus that will be described in the next few sections.

2.1 New Command Line Tools

Globus 4.2 introduced some new command line tools and options.

globus-validate-descriptors is a diagnostic tool that validates all the configuration files for Globus' web services. ***globus-check-environment*** is a debug tool that displays version information for OS, JVM, Java, Globus libraries, and 3rd party libraries. These tools can be useful to determine if an error is caused by misconfiguration or unfulfilled version requirements.

globus-update-client-config is used to merge the “client-config.wsdd” configuration files for installed services into a global configuration file. This is mainly meant to aid in adding and removing typemappings. Typemappings are exceptions in the service configuration to allow it to use non-standard XML variable types. Merging the configuration files puts all the services' typemappings in one place.

globus-reload-container allows the Globus container to be updated with new configuration without being restarted. ***globus-remote-deploy-gar*** and ***globus-remote-undeploy-gar*** are used to deploy and undeploy GARs. These three tools are part of the dynamic deployment feature (see section 2.3).

WS-Enumerate is a protocol for transferring large amounts of XML data in an enumerated form. Globus 4.2 makes use of this protocol with three new command line tools. ***ws-enumerate-start*** opens an "enumeration context" (effectively establishing a connection), ***ws-enumerate*** retrieves a group of enumeration data, and ***ws-enumerate-end*** closes the enumeration context.

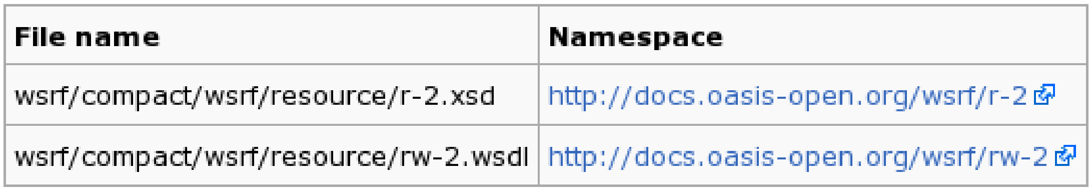
Xpath is a proprietary protocol for querying resource properties. ***globus-xpath-query*** queries the resource properties of a resource with a new query dialect called "TargetedXPath".

2.2 Developer's Changes

Some changes to the Java container that were made will affect developers' web services code. One of these changes is to fault handling. In earlier versions, faults were structured as arrays of fault types. The new specification uses a chain of fault causes.

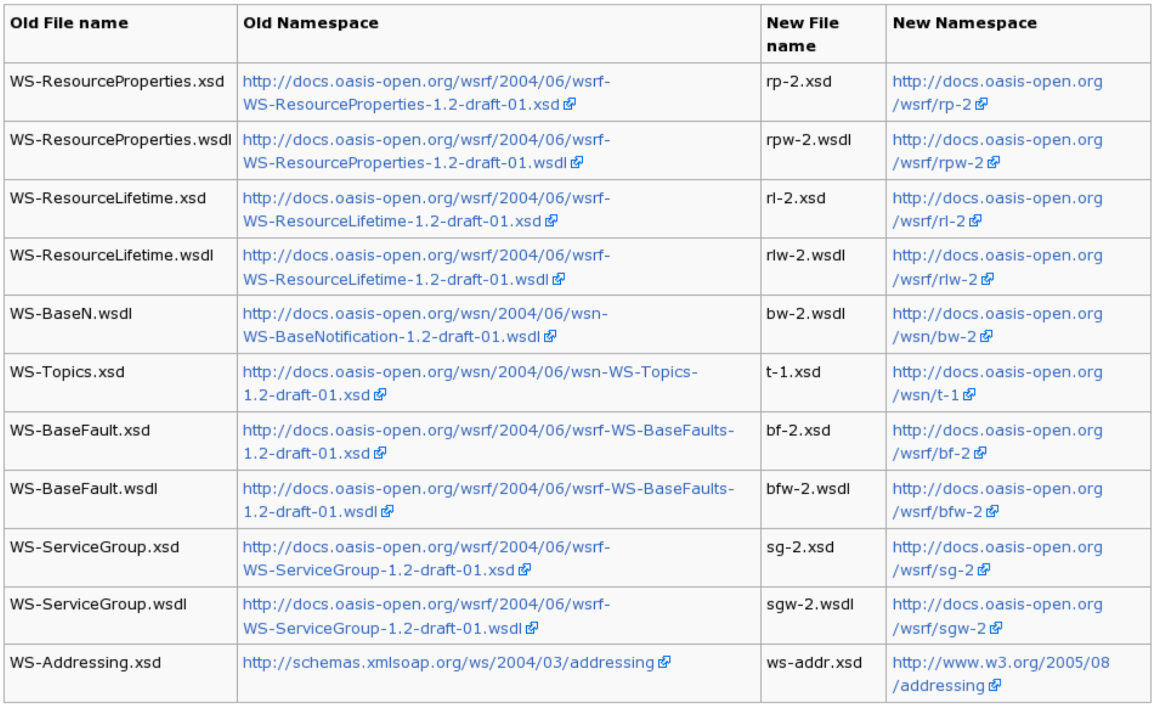
Another change has to do with task lifetime. A change to specification has made "xsd:duration" (relative timeout) available in addition to "xsd:dateTime" (absolute timeout). However, while duration is a seemingly legitimate argument, it is incompatible with the Java WS Core.

There have been changes to the address scheme for WSDL. Several of the namespaces and file names were changed to an abbreviated format. There is also a new specification for web service resource files. The details are listed in the following tables.



File name	Namespace
wsrf/compact/wsrf/resource/r-2.xsd	http://docs.oasis-open.org/wsrf/r-2
wsrf/compact/wsrf/resource/rw-2.wsdl	http://docs.oasis-open.org/wsrf/rw-2

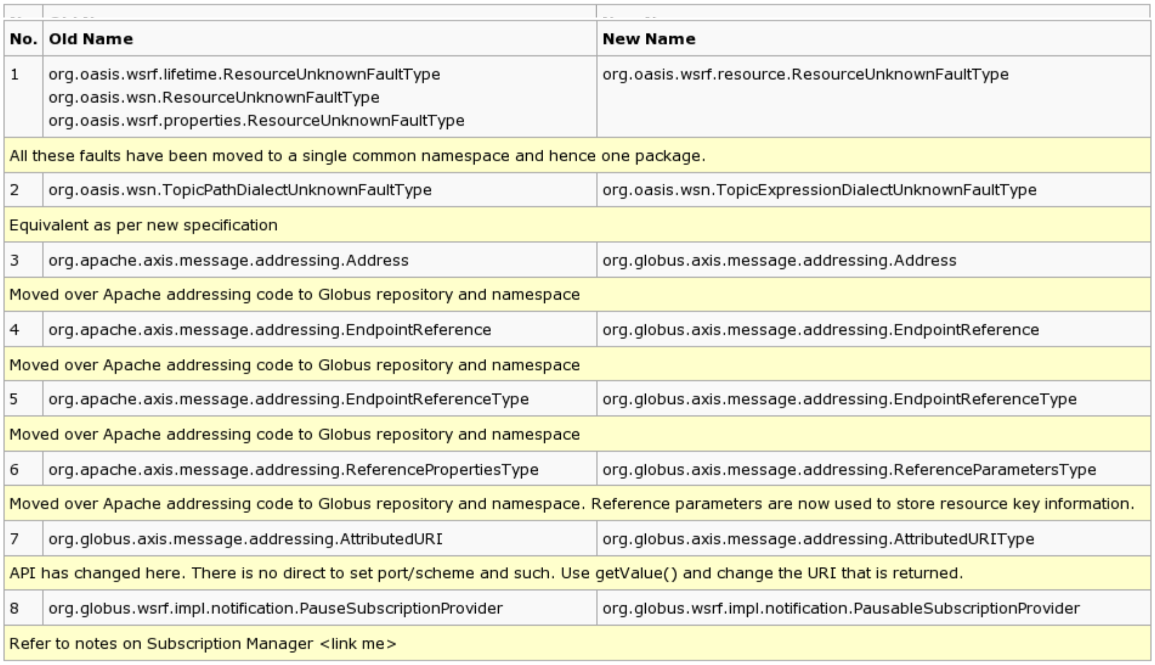
Table 1: New WSDL Files and namespace [1]



Old File name	Old Namespace	New File name	New Namespace
WS-ResourceProperties.xsd	http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceProperties-1.2-draft-01.xsd	rp-2.xsd	http://docs.oasis-open.org/wsrf/rp-2
WS-ResourceProperties.wsdl	http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceProperties-1.2-draft-01.wsdl	rpw-2.wsdl	http://docs.oasis-open.org/wsrf/rpw-2
WS-ResourceLifetime.xsd	http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-1.2-draft-01.xsd	rl-2.xsd	http://docs.oasis-open.org/wsrf/rl-2
WS-ResourceLifetime.wsdl	http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ResourceLifetime-1.2-draft-01.wsdl	rlw-2.wsdl	http://docs.oasis-open.org/wsrf/rlw-2
WS-BaseN.wsdl	http://docs.oasis-open.org/wsn/2004/06/wsn-WS-BaseNotification-1.2-draft-01.wsdl	bw-2.wsdl	http://docs.oasis-open.org/wsn/bw-2
WS-Topics.xsd	http://docs.oasis-open.org/wsn/2004/06/wsn-WS-Topics-1.2-draft-01.xsd	t-1.xsd	http://docs.oasis-open.org/wsn/t-1
WS-BaseFault.xsd	http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-BaseFaults-1.2-draft-01.xsd	bf-2.xsd	http://docs.oasis-open.org/wsrf/bf-2
WS-BaseFault.wsdl	http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-BaseFaults-1.2-draft-01.wsdl	bfw-2.wsdl	http://docs.oasis-open.org/wsrf/bfw-2
WS-ServiceGroup.xsd	http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ServiceGroup-1.2-draft-01.xsd	sg-2.xsd	http://docs.oasis-open.org/wsrf/sg-2
WS-ServiceGroup.wsdl	http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ServiceGroup-1.2-draft-01.wsdl	sgw-2.wsdl	http://docs.oasis-open.org/wsrf/sgw-2
WS-Addressing.xsd	http://schemas.xmlsoap.org/ws/2004/03/addressing	ws-addr.xsd	http://www.w3.org/2005/08/addressing

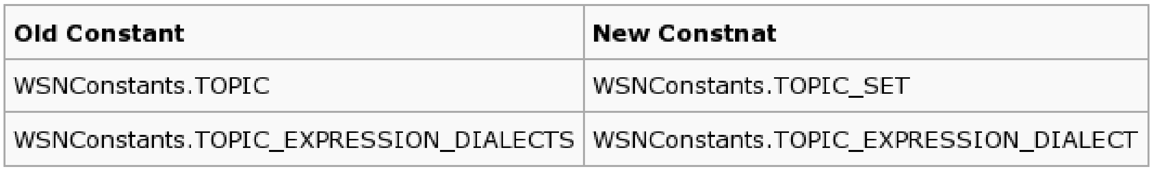
Table 2: WSDL Files and namespace mapping [1]

There have also been changes to some of Globus' Java library classes and constants, as described in the following tables.



No.	Old Name	New Name
1	org.oasis.wsrf.lifetime.ResourceUnknownFaultType org.oasis.wsn.ResourceUnknownFaultType org.oasis.wsrf.properties.ResourceUnknownFaultType	org.oasis.wsrf.resource.ResourceUnknownFaultType
All these faults have been moved to a single common namespace and hence one package.		
2	org.oasis.wsn.TopicPathDialectUnknownFaultType	org.oasis.wsn.TopicExpressionDialectUnknownFaultType
Equivalent as per new specification		
3	org.apache.axis.message.addressing.Address	org.globus.axis.message.addressing.Address
Moved over Apache addressing code to Globus repository and namespace		
4	org.apache.axis.message.addressing.EndpointReference	org.globus.axis.message.addressing.EndpointReference
Moved over Apache addressing code to Globus repository and namespace		
5	org.apache.axis.message.addressing.EndpointReferenceType	org.globus.axis.message.addressing.EndpointReferenceType
Moved over Apache addressing code to Globus repository and namespace		
6	org.apache.axis.message.addressing.ReferencePropertiesType	org.globus.axis.message.addressing.ReferenceParametersType
Moved over Apache addressing code to Globus repository and namespace. Reference parameters are now used to store resource key information.		
7	org.globus.axis.message.addressing.AttributedURI	org.globus.axis.message.addressing.AttributedURIType
API has changed here. There is no direct to set port/scheme and such. Use getValue() and change the URI that is returned.		
8	org.globus.wsrf.impl.notification.PauseSubscriptionProvider	org.globus.wsrf.impl.notification.PausableSubscriptionProvider
Refer to notes on Subscription Manager <link me>		

Table 3: Package names and class name mappings [1]



Old Constant	New Constnat
WSNConstants.TOPIC	WSNConstants.TOPIC_SET
WSNConstants.TOPIC_EXPRESSION_DIALECTS	WSNConstants.TOPIC_EXPRESSION_DIALECT

Table 4: Package names and class name mappings [1]

2.3 Dynamic Deployment

A new Globus 4.2 feature is dynamic deployment. The Java WS Core uses a 'common' classloader and a 'service' classloader to load Java libraries to the container. The 'common' classloader loads the libraries needed to run the container and cannot be reloaded. The 'service' classloader loads libraries for specific services and can be reloaded.

Reloading the 'service' classloader is the process of loading the required libraries without having to restart the container outright. Running jobs finish before reloading starts (so they aren't lost) and new jobs are accepted but not run during reloading.

Reloading must occur in order to deploy or undeploy a GAR (Grid ARchive) file. A GAR file contains all the required information for a grid service, so deploying the GAR is effectively deploying the service.

Dynamic deployment is an effective new tool for deploying services. It removes the need to restart the container and therefore affects service minimally.

3. C WS Core

The C WS Core is the container for C based web services.

3.1 New Command Line Tools

Several new command line tools have been implemented in the C container. ***globus-wsc-container*** is a standalone SOAP server to host C based web services. ***globus-wsrf-cgen*** generates a GPT source package from WSDL and XML Schema files.

globus-wsrf-destroy destroys a WSRF resource. ***globus-wsrf-set-termination-time*** will set a termination time for a WSRF resource.

There are new tools to obtain and edit WSRF resource's properties. ***globus-wsrf-query*** will retrieve a Resource Property document. ***globus-wsrf-get-property*** gets a single resource property's value and ***globus-wsrf-get-properties*** will get multiple. ***globus-wsrf-insert-property***, ***globus-wsrf-update-property***, and ***globus-wsrf-delete-property*** all have obvious uses.

globus-wsn-get-current-message, ***globus-wsn-pause-subscription***, ***globus-wsn-resume-subscription***, and ***globus-wsn-subscribe*** are new tools to deal with Web Services Notification (WSN) subscriptions.

4. Backward Compatibility

The Java WS Core and C WS Core have both had changes to their frameworks. This has made some features inoperable with machines running Globus 4.0. Two such features are the Delegation Service and Community Authorization Service (CAS). These security services are not strictly necessary, but losing security features in a grid is not desirable.

The Index Service, for collecting grid statistics, also no longer works with Globus 4.0. Neither does the Trigger Service, which can run a script when a condition is met in the grid statistics. Losing these services greatly reduces functionality, but they are not strictly necessary.

The most important backward incompatible service is GRAM4. This is the resource manager that distributes jobs on the grid. This is a core service of Globus. GRAM2 is still provided as a legacy service from Globus 2, but using GRAM2 would make for an awkward setup. The incompatibility of GRAM4 means that updating to Globus 4.2 must be done on entire clusters at a time. Since a Globus 4.2 metascheduler can't delegate to Globus 4.0 clusters, the entire grid would need to be updated to use Globus 4.2 on a metascheduler.

5. Other Changes

The update to Globus 4.2 does more than add new features. There were several other subtle changes. Many of the known bugs from Globus 4.0 were resolved in 4.2 (35 from Java WS Core and 7 from C WS Core). New features to the Java WS Core that were not described in this document include WS-Enumeration support, SOAP attachment support, and Persistent HTTP connection support. The Java container has also been updated to support WSA version 1.0, WSRF version 1.2, and WSN version 1.3.

Aside from this, there are several small changes to configuration of some web services and updates to 3rd party libraries. There have also been several small improvements to the operation of the C container.

CONCLUSIONS

There are several small changes to Globus in the new version 4.2. This makes them hard to keep track of and makes updating from version 4.0 troublesome. Some useful new features have been added along with several new command line tools.

The changes to installation have been minimal. There are several changes to the prerequisite software, but this has a small impact. Besides that, installation complications are a matter of misdocumentation.

The Java WS Core has been changed quite a lot. It has several new command line tools which improve its ease of use. The changes to coding libraries force updates to web services, which is a large drawback. However these changes will facilitate cleaner coding in the future. The dynamic deployment feature is an extremely useful tool, since it allows for GAR deployment without restarting the container. This is a good tool to help speed up the development of web services.

The C WS Core has had fewer changes. The most apparent changes are the several new command line tools. They greatly improve the accessibility of services in the container.

The lack of backward compatibility with Globus 4.0 is a problem. Specifically, the incompatibility of GRAM4 makes it infeasible to interface Globus 4.2 with Globus 4.0. This makes it difficult to update a grid, since it must be done whole clusters at a time.

RECOMMENDATIONS

Globus 4.2 makes several improvements over Globus 4.0. The two biggest drawbacks are backward incompatibility and the need to update code. A grid may need to interface with a Globus 4.0 (or older) machine. The scale of updating Globus installations and web services code may also be deterrents to the upgrade. The practicality of specific features should be more deeply investigated before upgrading a grid.

REFERENCES

Cited References:

[1] "Java WS Core/Final WSRF Migration Guide" Globus, [Globus Wiki Page], Available at: http://dev.globus.org/wiki/Java_WS_Core/Final_WSRF_Migration_Guide

General References:

"Globus Toolkit 4.2.0 Release Manuals" Globus, [Online Manuals], Available at: http://www.globus.org/toolkit/docs/4.2/4.2.0/

"GT4 Java WS Core Container Dynamic Deployment Design" Globus, [Design Document], (2006), Available at: http://www.globus.org/toolkit/docs/4.2/4.2.0/common/javawscore/developer/HotDeployDesign.pdf

"GT 4.2.0 Glossary" Globus, [Online Glossary], Available at: http://www.globus.org/toolkit/docs/4.2/4.2.0/glossary.html

"GT 4.2.0 Release Notes" Globus, [Release Notes], Available at: http://www.globus.org/toolkit/docs/4.2/4.2.0/rn/release_notes.html

"GT 4.2.0 Release Notes: C WS Core" Globus, [Release Notes], Available at: http://www.globus.org/toolkit/docs/4.2/4.2.0/common/cwscore/rn/index.html

"GT 4.2.0 Release Notes: Java WS Core" Globus, [Release Notes], Available at: http://www.globus.org/toolkit/docs/4.2/4.2.0/common/javawscore/rn/index.html

INSTALL files, Globus, [4.0 and 4.2 Installation Instructions]

"Java WS Core : Developer's Guide" Globus, [Online Manual], Available at: http://www.globus.org/toolkit/docs/4.2/4.2.0/common/javawscore/developer/

"Java WS Core WS-Enumeration Design" Globus, [Design Document], (2006), Available at: http://www.globus.org/toolkit/docs/4.2/4.2.0/common/javawscore/developer/WSEnumDesign.pdf

"Web Services Enumeration (WS-Enumeration)" w3, [Online Specification], (2006), Available at: http://www.w3.org/Submission/WS-Enumeration/