IST-2001-32133
GridLab - A Grid Application Toolkit and Testbed

First GridLabMDS Release

Author(s):	Giovanni Aloisio, Massimo Cafaro, Italo Epicoco, Daniele Lezzi, Maria Mirto, Silvia Mocavero, Serena Pati
Document Filename:	GridLab-10-D.3-0001-1.0_FIRST_MDS_RELEASE
Work package:	WP10 Information Services
Partner(s):	MPG, Germany MU, Czech Republic, SZTAKI, Hungary ISI, USA
Lead Partner:	University of Lecce
Config ID:	GridLab-10-D.3-0001-1.0
Document classification:	Internal

Abstract: This document describes the first release of GridLab MDS. The documentation covers the GridLab MDS schema extension, related information providers and a web service to publish and retrieve information.

Contents

1 Introduction

This document is associated to the first prototype release of the GridLab MDS. The GridLab MDS has been designed as an extension of Globus Project MDS, taking into account the requirements for grid computing suggested by others GridLab WPs . The prototype release includes:

- GridLab MDS schema extension;
- information providers related to the new schema;
- a set of functionalities, developed as a web service, to publish and retrieve information stored in the MDS;

The document is organized as follows: in section 2 we give the instructions needed to install and configure the GridLabMDS, and to build and compile the web service; in section 3 we describe the additional information introduced in this release of the GridLab MDS by means of a schema and finally, in section 4 we present the web service methods to be used to publish and retrieve information.

2 Installation

This package includes a web service and related sample clients. To build and install the web service please modify "Makefile.conf" file according to your environment and follow the installation instructions available in the mds_web_service directory.
The package also includes the first GridLab MDS Schema prototype and related information providers. To build and install the MDS schema extension please follow the instructions available in mds_schema directory.

2.1 Installation of GridLab MDS web service

This package includes a web service and related sample clients. To build and install the web service please follow the instructions below.

Required software:

globus toolkit v2.0
gSOAP toolkit v2.2.2
GSI plugin for gSOAP v1.5
automake 1.7.2
autoconf 2.57

In particular, we require a globus installation with both threaded and non threaded flavors for the Information Services SDK bundle; another requirement is that the *hostname* command must return the fully qualified domain name. Please note that this package includes the linux version of gSOAP toolkit; to install on a different platform, please download the appropriate gSOAP version and replace the files in the src/common directory and src/bin.

1. Modify the ../Makefile.conf file according to your environment

The Makefile.conf file allows the user to modify the following environmental variables:

GLOBUS_FLAVOR_THREADS: the name of the threaded flavor of globus SDK
GLOBUS_FLAVOR: the name of the non threaded flavor of globus SDK
GLOBUS_LOCATION: the globus toolkit installation directory
GLOBUS_INCLUDE_THREADS: the directory containing the header files for the threaded flavor of the globus toolkit
GLOBUS_INCLUDE: the directory containing the header files for the non threaded flavor of the globus toolkit
GLOBUS_LIB: the directory containing the globus toolkit libraries
SED: the pathname of the "sed" application
RANLIB: the pathname of the ranlib command (if needed, otherwise leave it blank)
GRID_MAPFILE: pathname of globus grid-mapfile
CERTIFICATION_AUTHORITY_DIR: pathname of directory containing recognized CAs certificates
GRIDLAB_SPOOL_DIR: the pathname of the directory where the configuration files related to services, software, firewall and Virtual Organization will be stored
GRIDLAB_SERVICES_SPOOL_DIR: services information provider spool directory
GRIDLAB_SOFTWARE_SPOOL_DIR: software information provider spool directory

GRIDLAB_FIREWALL_SPOOL_DIR: firewall information provider spool directory
GRIDLAB_VO_SPOOL_DIR: VOs information provider spool directory
MDS_DEFAULT_PORT: the port number where the LDAP server listens for incoming connections
MDS_DEFAULT_HOSTNAME: the hostname where the LDAP server is running on
DEFAULT_BASE_DN: the Base DN to be used when querying the LDAP server
GRIDLAB_SERVICE_LISTENING_PORT: the port number where the MDS web service listens for incoming connections
GRIDLAB_SERVICE_HOSTNAME: the host name of the machine where the MDS web service is running on
AUTHORIZATION_FILE: pathname of the access control list file
SERVICE_DESCRIPTION_FILE: pathname of the text file whose contents will be returned to the user upon the invocation of the "gridlab_getServiceDescription" method

```
type:
       cd src
       autoreconf --force
       cd ..
       autoreconf --force
```

```
 2. configure --prefix=<your-installation-path>

        ./configure --prefix=/opt/gridlab/mds
```

3. Execute the make command to build the server and client applications.

make

4. install everything as user with read/write permissions to the $prefix directory

make install

2.2 Installation of GridLab MDS

Required software:

perl, at least v5.x
globus toolkit v2.0
automake 1.7.2
autoconf 2.57

1) modify the file ../Makefile.conf according to your environment;

```
type: autoreconf --force
```

2) configure the distribution

./configure

3) Build

make

4) Install the information providers as the user owning the Globus installation (typically the user "globus")

make install

5) modify the ldap server configuration file <GLOBUS_LOCATION>/etc/grid-info-slapd.conf as follows:

insert the following line after all of the include directives
include <GLOBUS_LOCATION>/etc/gridlab.schema

6) append the content of the file gridlab-ldif.conf to <GLOBUS_LOCATION>/etc/grid-info-resource-ldif.conf

```
cd <GLOBUS_LOCATION>/etc
cat gridlab-ldif.conf >> grid-info-resource-ldif.conf
```

7) Configure the <GLOBUS_LOCATION>/etc/grid-info-resource-register.conf setting the timing parameters as follows:

regperiod: time in seconds between outgoing registration messages. A time period too short will generate a lot of useless registration messages; a time period too long is such that the GIIS server will not receive updated registration messages. A tradeoff value for this parameter can be 300.

ttl: time in seconds to keep registration data in the GIIS. Normally it is twice the value of regperiod (600).

timeout: after how long should a client give up when querying the service. The value of this parameter is tightly coupled to the execution time of all of the information providers. To establish the right value on your platform, run the evaluation script:
sbin/eval_providers_execution_time

cachettl: time in seconds for client to cache data. This parameter strongly affects performances.

8) as root user, restart gris daemon:

service SXXgris stop
service SXXgris start

3 GridLab MDS schema extension

This section describes the information schema developed for the first MDS release. The schema closely takes into account the requirements for grid computing provided by others GridLab WPs as described in deliverable D10.2. Of course, this is not meant to be static, the schema will evolve and will be extended to support additional information that will be required by the GridLab project. The current security policy adopted for this prototype release of GridLab MDS is to allow anonymous binding to the MDS for reading; however, only authorized users are actually allowed to write data through our MDS web service, and authentication/authorization is based on globus GSI. It is worth noting here that part of the information published inside the MDS is sensitive, so that a better security policy will be in place. In our case, our security policy for the first official release, scheduled for the end of 2003, will be to deny anonymous binding to GridLab MDS and requiring GSI (Globus Security Infrastructure, based on public-key cryptography) binding. We will enforce the security policy by using the authorization service that will be provided by the security WP6, to authorize trusted users to access the MDS for reading/writing. The current prototype release provides information related to:

- Services;
- Software;
- Users;
- Firewalls;
- Virtual Organizations;
- Certification Authorities.

The information providers for users and CA certificates get information directly. Information providers for services, software, firewalls, and VOs get information from an internally managed database, which is initially empty and must be populated through the GridLab MDS web service. This can be done, for instance, using the provided gridlab-*-register-client programs.
We now briefly review the attributes to be inserted inside the GridLab MDS. These attributes comes from a detailed comparison of attributes already available inside the Globus toolkit MDS and attributes needed for GridLab grid computing scenarios.

3.1 Information related to services

During the course of the GridLab project, a number of services will be developed by GridLab WPs. One of the most important requirements for GridLab grid computing scenarios is the ability to discover services dynamically. The MDS will provide GridLab developers with the following functionalities: service registration, service unregistration, service lookup. In this category falls the following attributes:

- GridLab-Mds-Service-name: service name
- GridLab-Mds-Service-port: service port
- GridLab-Mds-Service-type: service protocol
- GridLab-Mds-Service-description: service description
- GridLab-Mds-Service-publisher: service publisher

3.2 Information related to installed software

In order to submit jobs on grid resources, it is of crucial importance to know details related to the software packages that will be used in a run, may be a complex parallel simulation or a simple batch job. In this category falls the following attributes:

- GridLab-Mds-Software-name: name
- GridLab-Mds-Software-version: version
- GridLab-Mds-Software-path: pathname
- GridLab-Mds-Software-totalLicences: number of total software licences
- GridLab-Mds-Software-freeLicence: number of available software licences
- GridLab-Mds-Software-licenceInfo: information about software licence
- GridLab-Mds-Software-startupEnvironment: multi-valued software startup environmental variables
- GridLab-Mds-Software-executable: software executable
- GridLab-Mds-Software-arguments: multi-valued software arguments
- GridLab-Mds-Software-description: software description
- GridLab-Mds-Software-helpURL: software help URL
- GridLab-Mds-Software-usage: software usage

3.3 Information related to users

Information related to users allow complex brokering strategies: for instance, once the set of computing resources available to a user is known to a broker, it is then possible to choose carefully where to submit a user's job. The broker's decision will be based on the information gathered from the MDS and on the job's requirements. In this category falls the following attributes:

- GridLab-Mds-User-ID: user's login name on local resource
- GridLab-Mds-User-Mapped-DN: multi-valued attribute representing the Distinguished name mapped on the user
- GridLab-Mds-User-homedir: user's home directory
- GridLab-Mds-User-shell: user's shell
- GridLab-Mds-User-UID: user's UID
- GridLab-Mds-User-GID: user's GID
- GridLab-Mds-User-comment: a short comment about the user

3.4 Information related to firewalls

This kind of information is strictly related to service information. As a matter of fact, before registering a service, GridLab developers will query the MDS to know dynamically the range of open ports available on a specified computational resource. This is required to allow other people to connect to a service. In this category falls the following attributes:

- GridLab-Mds-Firewall-hostname: firewall hostname
- GridLab-Mds-Firewall-ports: multi-valued attribute representing open ports (range)
- GridLab-Mds-Firewall-validityTime: time frame during which open ports ranges are valid
- GridLab-Mds-Firewall-adminDN: Distinguished name of firewall administrator

3.5 Information related to Virtual Organizations

The GridLab project will span multiple Virtual Organizations. Corresponding information will allow people to know, for instance, how to request an account on a machine belonging to a particular Virtual Organization, or the people to contact in case of trouble. In this category falls the following attributes:

- GridLab-Mds-Vo-name: Virtual Organization to which a specified computational resource belongs to
- GridLab-Mds-Vo-helpDeskPhoneNumber: multi-valued help desk phone number
- GridLab-Mds-Vo-helpDeskURL: URL pointing to a Virtual Organization's web page
- GridLab-Mds-Vo-adminName: administrator name of the VO

3.6 Information related to recognized Certification Authorities

This is a set of information about the certification authorities allowed to sign the user's certificates that can be used to access and use the resource; it is important to know which CAs must be contacted to obtain a valid certificates. In this category falls the following attributes:

- GridLab-Mds-Certificate-Subj: Distinguished name of the recognized certification authority
- GridLab-Mds-Certificate-version: CA's certificate version
- GridLab-Mds-Certificate-serialNumber: CA's certificate serial number
- GridLab-Mds-Certificate-signatureAlgorithm: CA's certificate signature algorithm
- GridLab-Mds-Certificate-issuer: CA's certificate issuer
- GridLab-Mds-Certificate-validity-from: beginning date of the CA's certificate validity
- GridLab-Mds-Certificate-validity-to: end date of the CA's certificate validity
- GridLab-Mds-Certificate-publicKeyAlgorithm: CA's certificate public key algorithm
- GridLab-Mds-Certificate-RSAPublicKey: CA's certificate RSA public key

3.7 Information model

In this section we provide a detailed explanation of the object classes and attributes belonging to the GridLab information schema.
Object class GridLabServiceName (OID 1.3.6.1.4.1.3536.2.6.3536.9.1.1) is a structural class for information related to services. The class inherits from the Globus schema class Mds the following attributes:

- Mds-validfrom (global time at which the object and its information is first valid)
- Mds-validto (global time at which the object and its information is no longer valid)
- Mds-keepto (global time at which the object and its information should be deleted; the purge time should always be later than the expiration time, if it exists; the existence of a purge time hints that some information in the object may be useful even when the object is invalid, i.e. that the invalid object is better than no information at all)

The class contain the following attributes:

- GridLab-Mds-Service-name (OID 1.3.6.1.4.1.3536.2.6.3536.9.1.1.0.1) is a required single-valued attribute representing the service's name.

Object class GridLabService (OID 1.3.6.1.4.1.3536.2.6.3536.9.1) is an auxiliary class that inherits from Mds and contains the following attributes:

- GridLab-Mds-Service-port (OID 1.3.6.1.4.1.3536.2.6.3536.9.1.0.1) is a required single-valued attribute representing the port used by the service to listen for incoming connections.
- GridLab-Mds-Service-type (OID 1.3.6.1.4.1.3536.2.6.3536.9.1.0.2) is a required single-valued attrubute representing the protocol used by the service, e.g. http, https, httpg etc.
- GridLab-Mds-Service-publisher (OID 1.3.6.1.4.1.3536.2.6.3536.9.1.0.3) is a required single-valued attribute representing the distinguished name of the service's publisher.
- GridLab-Mds-Service-description (OID 1.3.6.1.4.1.3536.2.6.3536.9.1.0.4) is a required single-valued attribute describing information related to the service, for instance: how the service works, its purpose, etc.

Object class GridLabServiceGroup (OID 1.3.6.1.4.1.3536.2.6.3536.9.2.1) is a structural class that inherits from Mds and contains the following attributes:

- GridLab-Mds-Service-total-count (OID 1.3.6.1.4.1.3536.2.6.3536.9.2.1.0.1) is a required single-valued attribute representing the total number of services registered on the local resource.
- GridLab-Mds-Services (OID 1.3.6.1.4.1.3536.2.6.3536.9.2.1.0.2) is a required multi-valued attribute representing the names of all of the services registered on the local resource.
- GridLab-Mds-Services-Group (OID 1.3.6.1.4.1.3536.2.6.3536.9.2.1.0.3) is a required single-valued attribute representing the name of the group

Object class GridLabSoftwareName (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.1) is a structural class that inherits from Mds and contains the following attributes:

- GridLab-Mds-Software-name (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.1.0.1) is a required single-valued attribute representing a software package name.

Object class GridLabSoftware (OID 1.3.6.1.4.1.3536.2.6.3536.9.3) is an auxiliary class that inherits from Mds and contains the following attributes:

- GridLab-Mds-Software-version (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.1) is a required single-valued attribute representing a software package version.
- GridLab-Mds-Software-path (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.2) is a required single-valued attribute representing the pathname (directory) where software is installed.
- GridLab-Mds-Software-totalLicences (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.3) is a required single-valued attribute representing the total number of software package licences.
- GridLab-Mds-Software-freeLicence (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.4) is a required single-valued attribute representing the number of software package free licences.
- GridLab-Mds-Software-licenceInfo (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.5) is a required single-valued attribute describing information related to the software licence.
- GridLab-Mds-Software-startupEnvironment (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.6) is a required multi-valued attribute representing a software package startup environment.
- GridLab-Mds-Software-executable (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.7) is a required single-valued attribute representing a software package executable pathname.
- GridLab-Mds-Software-arguments (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.8) is a required multi-valued attribute describing the software package command line arguments.
- GridLab-Mds-Software-description (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.9) is a required single-

valued attribute representing a software package description.
- GridLab-Mds-Software-helpURL (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.10) is a required single-valued attribute representing a software package URL pointing to a web page describing the software.
- GridLab-Mds-Software-usage (OID 1.3.6.1.4.1.3536.2.6.3536.9.3.0.11) is a required single-valued attribute representing the software usage.

Object class GridLabSoftwareGroup (OID 1.3.6.1.4.1.3536.2.6.3536.9.4.1) is a structural class that inherits from Mds and contains the following attributes:

- GridLab-Mds-Software-total-count (OID 1.3.6.1.4.1.3536.2.6.3536.9.4.1.0.1) is a required single-valued attribute representing the total number of software packages registered.
- GridLab-Mds-Software (OID 1.3.6.1.4.1.3536.2.6.3536.9.4.1.0.2) is a required multi-valued attribute representing the software packages name registered.
- GridLab-Mds-Software-Group (OID 1.3.6.1.4.1.3536.2.6.3536.9.4.1.0.3) is a required single-valued attribute representing the name of the group

Object class GridLabUserUniqueID (OID 1.3.6.1.4.1.3536.2.6.3536.9.5.1) is a structural class that inherits from Mds and contains the following attributes:

-GridLab-Mds-User-ID (OID 1.3.6.1.4.1.3536.2.6.3536.9.5.1.0.1) is a required single-valued attribute representing the user login name.

Object class GridLabUser (OID 1.3.6.1.4.1.3536.2.6.3536.9.5) is an auxiliary class that inherits from Mds and contains the following attributes:

-GridLab-Mds-User-Mapped-DN (OID 1.3.6.1.4.1.3536.2.6.3536.9.5.0.1) is a required multi-valued attribute representing the Distinguished Name mapped on the user.
-GridLab-Mds-User-homedir (OID 1.3.6.1.4.1.3536.2.6.3536.9.5.0.2) is a required single-valued representing the user's home directory
-GridLab-Mds-User-shell (OID 1.3.6.1.4.1.3536.2.6.3536.9.5.0.3) is a required single-valued attribute representing the user shell.
-GridLab-Mds-User-UID (OID 1.3.6.1.4.1.3536.2.6.3536.9.5.0.4) is a required single-valued attribute representing the user UID.
-GridLab-Mds-User-GID (OID 1.3.6.1.4.1.3536.2.6.3536.9.5.0.5) is a required single-valued attribute representing the user GID.
-GridLab-Mds-User-comment (OID 1.3.6.1.4.1.3536.2.6.3536.9.5.0.6) is a required single-valued attribute representing the user comment.

Object class GridLabUserGroup (OID 1.3.6.1.4.1.3536.2.6.3536.9.6.1) is a structural class that inherits from Mds and contains the following attributes:

-GridLab-Mds-User-total-count (OID 1.3.6.1.4.1.3536.2.6.3536.9.6.1.0.1) is a required single-valued attribute representing the number of users.
-GridLab-Mds-User (OID 1.3.6.1.4.1.3536.2.6.3536.9.6.1.0.2)is a required multi-valued attribute representing the list of user unique ID.
-GridLab-Mds-User-Group (OID 1.3.6.1.4.1.3536.2.6.3536.9.6.1.0.3) is a required single-valued attribute representing the name of the group

Object class GridLabFirewallHostname (OID 1.3.6.1.4.1.3536.2.6.3536.9.7.1) is a structural class that inherits from Mds and contains the following attributes:

- GridLab-Mds-Firewall-hostname (OID 1.3.6.1.4.1.3536.2.6.3536.9.7.1.0.1) is a required single-valued attribute representing the firewall hostname.

Object class GridLabFirewall (OID 1.3.6.1.4.1.3536.2.6.3536.9.7) is an auxiliary class that inherits from Mds and contains the following attributes:

- GridLab-Mds-Firewall (OID 1.3.6.1.4.1.3536.2.6.3536.9.7.0.1) is a required single-valued attribute representing the firewall.
- GridLab-Mds-Firewall-ports (OID 1.3.6.1.4.1.3536.2.6.3536.9.7.0.2) is a required multi-valued attribute representing open ports ranges.
- GridLab-Mds-Firewall-adminDN (OID 1.3.6.1.4.1.3536.2.6.3536.9.7.0.3) is a required single-valued attribute representing a firewall administrator Distinguished Name.
- GridLab-Mds-Firewall-validityTime (OID 1.3.6.1.4.1.3536.2.6.3536.9.7.0.4) is a required single-valued attribute representing the time frame during which open port ranges are valid.

Object class GridLabVoName (OID 1.3.6.1.4.1.3536.2.6.3536.9.8.1) is a structural class that inherits from Mds and contains the following attributes:

- GridLab-Mds-Vo-name (OID 1.3.6.1.4.1.3536.2.6.3536.9.8.1.0.1) is a required single-valued attribute representing the name of virtual organization to which the computing resource belongs to.

Object class GridLabVo (OID 1.3.6.1.4.1.3536.2.6.3536.9.8.) is an auxiliary class that inherits from Mds and contains the following attributes:

- GridLab-Mds-Vo-helpDeskPhoneNumber (OID 1.3.6.1.4.1.3536.2.6.3536.9.8.0.1) is a required multi-valued attribute representing the help desk phone number related to the virtual organization.
- GridLab-Mds-Vo-helpDeskURL (OID 1.3.6.1.4.1.3536.2.6.3536.9.8.0.2) is a required single-valued attribute representing the URL pointing to a web page describing the Virtual Organization.
- GridLab-Mds-Vo-adminName (OID 1.3.6.1.4.1.3536.2.6.3536.9.8.0.3) is a required single-valued attribute representing the name of the virtual organization administrator.

Object class GridLabVoGroup (OID 1.3.6.1.4.1.3536.2.6.3536.9.9.1) is a structural class that inherits from Mds and contains the following attributes:

-GridLab-Mds-Vo-total-count (OID 1.3.6.1.4.1.3536.2.6.3536.9.9.1.0.1) is a required single-valued attribute that represents the total number of Virtual Organization which the resource belongs to
-GridLab-Mds-Vo (OID 1.3.6.1.4.1.3536.2.6.3536.9.9.1.0.2) is a required multi-valued attribute representing the list of all the Virtual Organization which the resource belongs to
-GridLab-Mds-Vo-Group (OID 1.3.6.1.4.1.3536.2.6.3536.9.9.1.0.3) is a required single-valued attribute representing the name of the group

Object class GridLabCertificateSubject (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.1) is a structural class that inherits from Mds and contains the following attributes:

-GridLab-Mds-Certificate-subject (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.1.0.1) is a required multi-valued attribute representing the subject name of all of the certification authority accepted by the computational resource.
-GridLab-Mds-Certificates-Group (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.1.0.2) is a required single-valued attribute representing the name of the group

Object class GridLabCertificate (OID 1.3.6.1.4.1.3536.2.6.3536.9.10) is an auxiliary class that inherits from Mds and contains the following attributes:

- GridLab-Mds-Certificate-version (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.0.1) is a required single-valued attribute representing the version of the certificate.
- GridLab-Mds-Certificate-serialNumber (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.0.2) is a required single-valued attribute representing the serial number of the certificate.
- GridLab-Mds-Certificate-signatureAlgorithm (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.0.3) is a required single-valued attribute representing the signature algorithm of the certificate.
- GridLab-Mds-Certificate-issuer (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.0.4) is a required single-valued attribute representing the issuer of the certificate.
- GridLab-Mds-Certificate-validity-from (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.0.5) is a required single-valued attribute representing the beginning date of the certificate validity.
- GridLab-Mds-Certificate-validity-to (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.0.6) is a required single-valued attribute representing the end date of the certificate validity.
- GridLab-Mds-Certificate-publicKeyAlgorithm (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.0.7) is a required single-valued attribute representing the public key algorithm of the certificate.
- GridLab-Mds-Certificate-RSAPublicKey (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.0.8) is a required single-valued attribute representing the RSA public key of the certification authority.
-GridLab-Mds-Certificate-Subj (OID 1.3.6.1.4.1.3536.2.6.3536.9.10.0.9) is a required single-valued attribute representing the subject name of the CA's certificate

4 GridLab MDS web service

The web service has been developed as a pre-threaded server in order to improve the performances; it provides the user with several methods that can be invoked to search, register, unregister and lookup information.
The namespace used by the web service is urn:mds. The web service's methods are described using Web Service Description Language in Appendix; here, we describe using the C language the data structures and methods:

```
typedef int xsd__int;
typedef char * xsd__string;

struct string_array{
xsd__string *__ptr;
xsd__int __size;
};

struct Attribute {
xsd__string *__ptr; //pointer to the values array
```

```
xsd__int     __size; //size of the values array
};

struct Entry {
struct Attribute *__ptr;  //pointer to the attribute array
xsd__int __size; //size of the attribute array
};

struct mdsInfo {
struct Entry  *__ptr;
xsd__int  __size;
};

enum gridlab__scopes {LDAP_SCOPE_BASE=0,LDAP_SCOPE_SUB=2,LDAP_SCOPE_ONE=1};
```

-gridlab__search(xsd__string mds_hostname, xsd__int port, xsd__string base_dn, enum gridlab__scopes scope, xsd__string attributes, xsd__string filter, struct mdsInfo**result)
this method can be used to query an arbitrary MDS server for specified information

input:

mds_hostname: hostname of the LDAP server to contact
port: port number where the LDAP server is listening on
base_dn: base distinguished name to be used as starting point for searching
scope: the scope of the search. It can be: LDAP_SCOPE_BASE, LDAP_SCOPE_ONE, LDAP_SCOPE_SUB
attributes: space-separated list of attributes
filter: search filter criteria expressed according to RFC 2252

output:

result: query result

-gridlab__getServiceDescription(struct soap *soap, xsd__string *description)
this method can be used to get the description and status of the service
output:

description: the string of description

4.1 Services

-gridlab__register_service(xsd__string name, xsd__int port, xsd__string protocol, xsd__string description, xsd__int *result)
this method allows the user to register a new service.

input:

name: it is the name of the new service to be registered
port: port number where the service will be listening on
protocol: communication protocol to be used to contact the service

description: human readable description of the service

output:

result: set to -1 if an error occurs during the registration, 0 on success.

If the service being registered already exists, the registration will fail. If you want to modify an existing service, you must unregister and register it again.

-gridlab__unregister_service(xsd__string name, xsd__int *result)
this method removes an existing service from the GIS

input:

name: the name of the service to be removed from the GIS

output:

result: set to -1 if an error occurs, 0 on success

If the service does not exist the method returns 0 anyway.

-gridlab__lookup_service(xsd__string name, xsd__string mds_hostname, struct string_array *result)
this method lookups an existing service and returns the information about the services that have been found. It contacts the supplied MDS server on the default port

input:

name: the name of the service to be searched
mds_hostname: the hostname of the MDS server to be contacted

output:

result: it is an array of strings. Each item contains service information. If the service is not registered the array will contain 0 items.

4.2 Software

-gridlab__register_software(xsd__string name, xsd__string version, xsd__string path, xsd__int totallicences, xsd__int freelicence, xsd__string licenceinfo, xsd__string startupenvironment, xsd__string executable, xsd__string arguments, xsd__string description, xsd__string helpURL, xsd__string usage, xsd__int *result)
this method allows the user to register a new software package; all of the input parameters are mandatory to describe and register the software package.

input:

name: software package name
version: package version number
path: pathname of the executable
totallicences: total number of available software licences

freelicence: total number of available free software licences
licenceinfo: human readable information about the licence
startupenvironment: a list, space-separated, of startup environment variables
executable: executable file name
arguments: list of arguments
description: human readable description of the software package
helpURL: link to a help desk URL
usage: a short description about executable's usage

output:

result: set to -1 if an error occurs, 0 on success

If software is already registered the method will fail.

-gridlab__unregister_software(xsd__string name, xsd__int *result)
this method removes an existing software package from the GIS

input:

name: the name of the software package to be removed

output:

result: set to -1 if an error occurs, 0 on success

If the service does not exist the method sets result to 0 anyway.

- gridlab__lookup_software(xsd__string name, xsd__string mds_hostname, struct string_array *result)
this method lookups an existing software package and returns all the information about the software. It contacts the supplied MDS server on the default port

input:

name: the name of the software package to be searched
mds_hostname: the hostname of the MDS server to be contacted

output:

result: it is an array of strings. Each item contains the information about the software. If the package is not registered the array will contain 0 items.

4.3 Firewall

-gridlab__register_firewall(xsd__string hostname, xsd__string ports, xsd__int validitytime, xsd__int *result)
this method allows the registration of information about a firewall installed on the grid resource.

input:

hostname: host name of the machine where the firewall is installed
ports: space-separated list of open ports
validitytime: time frame during which the open ports are valid

output:

result: set to -1 if an error occurs, 0 on success

-gridlab__unregister_firewall(xsd__string hostname, xsd__int *result)
this method removes information about a registered firewall from the GIS

input:

hostname: host name of the machine where the firewall is installed; the corresponding information will be removed

output:

result: set to -1 if an error occurs, 0 on success

If the firewall information does not exist the method sets result to 0 anyway.

- gridlab__lookup_firewall(xsd__string hostname, xsd__string mds_hostname, struct string_array *result)
this method lookups firewall information. It contacts the supplied MDS server on the default port

input:

hostname: host name of the machine where the firewall is installed
mds_hostname: the host name of the MDS server to be contacted

output:

result: it is an array of strings. Each item contains information about the firewall. If firewall information is not registered the array will contain 0 items.

4.4 Virtual Organization

-gridlab__register_vo(xsd__string name, xsd__string helpDeskPN, xsd__string helpDeskURL, xsd__string adminname, xsd__int *result) this method allows the user to register the virtual organization to which the grid resource belongs.

input:

name: name of the VO
helpDeskPN: the Help Desk Phone number of the Virtual Organization
helpDeskURL: the URL of the help Desk
adminname: Distinguished name of the administrator.

output:

result: set to -1 if an error occurs, 0 on success

-gridlab__unregister_vo(xsd__string name, xsd__int *result)
this method removes information about a Virtual organization from the GIS

input:

name: name of the virtual organization whose information must be removed

output:

result: set to -1 if an error occurs, 0 on success

If the VO information does not exist the method sets result to 0 anyway.

-gridlab__lookup_vo(xsd__string name, xsd__string mds_hostname, struct string_array *result) this method lookups information about a specified virtual organization. It contacts the supplied MDS server on the default port

input:

name: name of the VO to be searched
mds_hostname: the host name of the MDS server to be contacted

output:

result: it is an array of strings. Each item contains information about a VO. If no VO is found the array will contain 0 items.

4.5 User

-gridlab__lookup_user(xsd__string name, xsd__string mds_hostname, struct string_array *result) this method lookups the information about a specified user. It contacts the supplied MDS server on the default port

input:

dname: DN mapped on the user
mds_hostname: the host name of the MDS server to be contacted

output:

result: it is an array of strings. Each item contains information about a user. If no user is found the array will contain 0 items.

4.6 Certification Authority

-gridlab__lookup_cert(xsd__string name, xsd__string mds_hostname, struct string_array *result) this method lookups the information about a specified certification authority. It contacts the supplied MDS server on the default port

input:

name: CA's subject name
mds_hostname: the host name of the MDS server to be contacted

output:

result: it is an array of strings. Each item contains information about recognized CA. If no CA is found the array will contain 0 items.

Appendix

We show here the WSDL document related to the GridLab MDS web service. Please note that this WDSL document is just a reference.

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="Service"
 xmlns="http://schemas.xmlsoap.org/wsdl/"
 xmlns:SOAP="http://schemas.xmlsoap.org/wsdl/soap/"
 xmlns:WSDL="http://schemas.xmlsoap.org/wsdl/"
 targetNamespace="http://location/Service.wsdl"
 xmlns:tns="http://location/Service.wsdl"
 xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns:gridlab="http://tempuri.org">

<types>
 <schema targetNamespace="http://tempuri.org"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:gridlab="http://tempuri.org"
  xmlns="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="unqualified"
  attributeFormDefault="unqualified">

  <simpleType name="scopes">
   <restriction base="xsd:string">
    <enumeration value="LDAP-SCOPE-BASE"/>
    <enumeration value="LDAP-SCOPE-SUB"/>
    <enumeration value="LDAP-SCOPE-ONE"/>
   </restriction>
  </simpleType>

  <complexType name="ArrayOfstring">
   <complexContent>
    <restriction base="SOAP-ENC:Array">
     <sequence>
      <element name="item" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
     </sequence>
     <attribute ref="SOAP-ENC:arrayType" WSDL:arrayType="xsd:string[]"/>
    </restriction>
   </complexContent>
  </complexType>
```

```
  <complexType name="ArrayOfstring">
   <complexContent>
    <restriction base="SOAP-ENC:Array">
     <sequence>
      <element name="item" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
     </sequence>
     <attribute ref="SOAP-ENC:arrayType" WSDL:arrayType="xsd:string[]"/>
    </restriction>
   </complexContent>
  </complexType>

  <complexType name="ArrayOfArrayOfstring">
   <complexContent>
    <restriction base="SOAP-ENC:Array">
     <sequence>
      <element name="item" type="gridlab:ArrayOfstring" minOccurs="0" maxOccurs="unbounded"/
     </sequence>
     <attribute ref="SOAP-ENC:arrayType" WSDL:arrayType="gridlab:ArrayOfstring[]"/>
    </restriction>
   </complexContent>
  </complexType>

  <complexType name="ArrayOfArrayOfArrayOfstring">
   <complexContent>
    <restriction base="SOAP-ENC:Array">
     <sequence>
      <element name="item" type="gridlab:ArrayOfArrayOfstring" minOccurs="0" maxOccurs="unbo
     </sequence>
     <attribute ref="SOAP-ENC:arrayType" WSDL:arrayType="gridlab:ArrayOfArrayOfstring[]"/>
    </restriction>
   </complexContent>
  </complexType>

 </schema>
</types>

<message name="register-serviceRequest">
 <part name="name" type="xsd:string"/>
 <part name="port" type="xsd:int"/>
 <part name="protocol" type="xsd:string"/>
 <part name="description" type="xsd:string"/>
</message>

<message name="register-serviceResponse">
 <part name="result" type="xsd:int"/>
</message>

<message name="unregister-serviceRequest">
 <part name="name" type="xsd:string"/>
```

```
</message>

<message name="unregister-serviceResponse">
 <part name="result" type="xsd:int"/>
</message>

<message name="lookup-serviceRequest">
 <part name="name" type="xsd:string"/>
 <part name="mds-hostname" type="xsd:string"/>
</message>

<message name="lookup-serviceResponse">
 <part name="result" type="gridlab:ArrayOfstring"/>
</message>

<message name="searchRequest">
 <part name="mds-hostname" type="xsd:string"/>
 <part name="port" type="xsd:int"/>
 <part name="base-dn" type="xsd:string"/>
 <part name="scope" type="gridlab:scopes"/>
 <part name="attributes" type="xsd:string"/>
 <part name="filter" type="xsd:string"/>
</message>

<message name="searchResponse">
 <part name="result" type="gridlab:ArrayOfArrayOfArrayOfstring"/>
</message>

<message name="register-softwareRequest">
 <part name="name" type="xsd:string"/>
 <part name="version" type="xsd:string"/>
 <part name="path" type="xsd:string"/>
 <part name="totallicence" type="xsd:int"/>
 <part name="freelicence" type="xsd:int"/>
 <part name="licenceinfo" type="xsd:string"/>
 <part name="startupenvironment" type="xsd:string"/>
 <part name="executable" type="xsd:string"/>
 <part name="arguments" type="xsd:string"/>
 <part name="description" type="xsd:string"/>
 <part name="helpURL" type="xsd:string"/>
 <part name="usage" type="xsd:string"/>
</message>

<message name="register-softwareResponse">
 <part name="result" type="xsd:int"/>
</message>

<message name="unregister-softwareRequest">
 <part name="name" type="xsd:string"/>
```

```
</message>

<message name="unregister-softwareResponse">
 <part name="result" type="xsd:int"/>
</message>

<message name="register-firewallRequest">
 <part name="hostname" type="xsd:string"/>
 <part name="ports" type="xsd:string"/>
 <part name="validity-time" type="xsd:string"/>
</message>

<message name="register-firewallResponse">
 <part name="result" type="xsd:int"/>
</message>

<message name="lookup-softwareRequest">
 <part name="name" type="xsd:string"/>
 <part name="mds-hostname" type="xsd:string"/>
</message>

<message name="lookup-softwareResponse">
 <part name="result" type="gridlab:ArrayOfstring"/>
</message>

<message name="unregister-firewallRequest">
 <part name="hostname" type="xsd:string"/>
</message>

<message name="unregister-firewallResponse">
 <part name="result" type="xsd:int"/>
</message>

<message name="lookup-firewallRequest">
 <part name="hostname" type="xsd:string"/>
 <part name="mds-hostname" type="xsd:string"/>
</message>

<message name="lookup-firewallResponse">
 <part name="result" type="gridlab:ArrayOfstring"/>
</message>

<message name="register-voRequest">
 <part name="name" type="xsd:string"/>
 <part name="helpDeskPN" type="xsd:string"/>
 <part name="helpDeskURL" type="xsd:string"/>
 <part name="adminname" type="xsd:string"/>
</message>
```

```
<message name="register-voResponse">
 <part name="result" type="xsd:int"/>
</message>

<message name="unregister-voRequest">
 <part name="name" type="xsd:string"/>
</message>

<message name="unregister-voResponse">
 <part name="result" type="xsd:int"/>
</message>

<message name="lookup-voRequest">
 <part name="name" type="xsd:string"/>
 <part name="mds-name" type="xsd:string"/>
</message>

<message name="lookup-voResponse">
 <part name="result" type="gridlab:ArrayOfstring"/>
</message>

<message name="lookup-userRequest">
 <part name="name" type="xsd:string"/>
 <part name="mds-name" type="xsd:string"/>
</message>

<message name="lookup-userResponse">
 <part name="result" type="gridlab:ArrayOfstring"/>
</message>

<message name="lookup-certRequest">
 <part name="name" type="xsd:string"/>
 <part name="mds-name" type="xsd:string"/>
</message>

<message name="lookup-certResponse">
 <part name="result" type="gridlab:ArrayOfstring"/>
</message>

<message name="getServiceDescriptionRequest">
</message>

<message name="getServiceDescriptionResponse">
 <part name="result" type="xsd:string"/>
</message>

<portType name="ServicePortType">
 <operation name="register-service">
  <documentation>Service definition of function gridlab__register_service</documentation>
```

```
 <input message="tns:register-serviceRequest"/>
 <output message="tns:register-serviceResponse"/>
</operation>
<operation name="unregister-service">
 <documentation>Service definition of function gridlab__unregister_service</documentation>
 <input message="tns:unregister-serviceRequest"/>
 <output message="tns:unregister-serviceResponse"/>
</operation>
<operation name="lookup-service">
 <documentation>Service definition of function gridlab__lookup_service</documentation>
 <input message="tns:lookup-serviceRequest"/>
 <output message="tns:lookup-serviceResponse"/>
</operation>
<operation name="search">
 <documentation>Service definition of function gridlab__search</documentation>
 <input message="tns:searchRequest"/>
 <output message="tns:searchResponse"/>
</operation>
<operation name="register-software">
 <documentation>Service definition of function gridlab__register_software</documentation>
 <input message="tns:register-softwareRequest"/>
 <output message="tns:register-softwareResponse"/>
</operation>
<operation name="unregister-software">
 <documentation>Service definition of function gridlab__unregister_software</documentation>
 <input message="tns:unregister-softwareRequest"/>
 <output message="tns:unregister-softwareResponse"/>
</operation>
<operation name="register-firewall">
 <documentation>Service definition of function gridlab__register_firewall</documentation>
 <input message="tns:register-firewallRequest"/>
 <output message="tns:register-firewallResponse"/>
</operation>
<operation name="lookup-software">
 <documentation>Service definition of function gridlab__lookup_software</documentation>
 <input message="tns:lookup-softwareRequest"/>
 <output message="tns:lookup-softwareResponse"/>
</operation>
<operation name="unregister-firewall">
 <documentation>Service definition of function gridlab__unregister_firewall</documentation>
 <input message="tns:unregister-firewallRequest"/>
 <output message="tns:unregister-firewallResponse"/>
</operation>
<operation name="lookup-firewall">
 <documentation>Service definition of function gridlab__lookup_firewall</documentation>
 <input message="tns:lookup-firewallRequest"/>
 <output message="tns:lookup-firewallResponse"/>
</operation>
<operation name="register-vo">
```

```
  <documentation>Service definition of function gridlab__register_vo</documentation>
  <input message="tns:register-voRequest"/>
  <output message="tns:register-voResponse"/>
 </operation>
 <operation name="unregister-vo">
  <documentation>Service definition of function gridlab__unregister_vo</documentation>
  <input message="tns:unregister-voRequest"/>
  <output message="tns:unregister-voResponse"/>
 </operation>
 <operation name="lookup-vo">
  <documentation>Service definition of function gridlab__lookup_vo</documentation>
  <input message="tns:lookup-voRequest"/>
  <output message="tns:lookup-voResponse"/>
 </operation>
 <operation name="lookup-user">
  <documentation>Service definition of function gridlab__lookup_user</documentation>
  <input message="tns:lookup-userRequest"/>
  <output message="tns:lookup-userResponse"/>
 </operation>
 <operation name="lookup-cert">
  <documentation>Service definition of function gridlab__lookup_cert</documentation>
  <input message="tns:lookup-certRequest"/>
  <output message="tns:lookup-certResponse"/>
 </operation>
 <operation name="getServiceDescription">
  <documentation>Service definition of function gridlab__getServiceDescription</documentatio
  <input message="tns:getServiceDescriptionRequest"/>
  <output message="tns:getServiceDescriptionResponse"/>
 </operation>
</portType>

<binding name="ServiceBinding" type="tns:ServicePortType">
 <SOAP:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
 <operation name="register-service">
  <SOAP:operation soapAction=""/>
  <input>
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
  </input>
  <output>
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
  </output>
 </operation>
 <operation name="unregister-service">
  <SOAP:operation soapAction=""/>
  <input>
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
  </input>
  <output>
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
```

```
 </output>
</operation>
<operation name="lookup-service">
 <SOAP:operation soapAction=""/>
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </output>
</operation>
<operation name="search">
 <SOAP:operation soapAction=""/>
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </output>
</operation>
<operation name="register-software">
 <SOAP:operation soapAction=""/>
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </output>
</operation>
<operation name="unregister-software">
 <SOAP:operation soapAction=""/>
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </output>
</operation>
<operation name="register-firewall">
 <SOAP:operation soapAction=""/>
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </output>
</operation>
<operation name="lookup-software">
 <SOAP:operation soapAction=""/>
```

```
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </output>
</operation>
<operation name="unregister-firewall">
 <SOAP:operation soapAction=""/>
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </output>
</operation>
<operation name="lookup-firewall">
 <SOAP:operation soapAction=""/>
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </output>
</operation>
<operation name="register-vo">
 <SOAP:operation soapAction=""/>
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </output>
</operation>
<operation name="unregister-vo">
 <SOAP:operation soapAction=""/>
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </output>
</operation>
<operation name="lookup-vo">
 <SOAP:operation soapAction=""/>
 <input>
  <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
 </input>
 <output>
```

```
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
  </output>
 </operation>
 <operation name="lookup-user">
  <SOAP:operation soapAction=""/>
  <input>
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
  </input>
  <output>
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
  </output>
 </operation>
 <operation name="lookup-cert">
  <SOAP:operation soapAction=""/>
  <input>
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
  </input>
  <output>
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
  </output>
 </operation>
 <operation name="getServiceDescription">
  <SOAP:operation soapAction=""/>
  <input>
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
  </input>
  <output>
   <SOAP:body use="encoded" namespace="http://tempuri.org" encodingStyle="http://schemas.xml
  </output>
 </operation>
</binding>

<service name="Service">
 <documentation>gSOAP 2.2.2 generated service definition</documentation>
 <port name="Service" binding="tns:ServiceBinding">
  <SOAP:address location="http://location"/>
 </port>
</service>

</definitions>
```