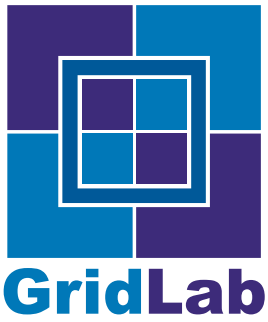


IST-2001-32133
GridLab - A Grid Application Toolkit and Testbed

# Requirements and Technical Specification

| | |
|---|---|
| Author(s): | Giovanni Aloisio, Massimo Cafaro, Italo Epicoco, Silvia Mocavero, Maria Mirto, Serena Pati |
| Document Filename: | GridLab-10-D.2-0001-1.0_Requirements |
| 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.2-0001-1.0 |
| Document classification: | IST |

**A**bstract: This document contains a preliminary version of the specifications and requirements for the information services.

# Contents

# 1 Introduction

The goal of the second task is to develop a new set of Globus GIS extensions taking into account that the grid is inherently dynamic, i.e machine load and availability, network latency and bandwidth change continually, so GIS extensions need to be developed to address grid-aware application requirements. This capabilities will allow new uses of global computational resources. Limitations and capabilities of current state of art GIS, as found by previous task, will be used as a starting point to develop the needed extensions. The activities of this task are strongly dependent on inputs from others WPs; however, as of this writing, we have collected only a few and general information, because the other WPs interested to GIS extensions have not produced their requirements and specification documents. Thus, this deliverable should be considered as a preliminary version of the requirements and specification of the GIS extensions, and the final, revised version will be issued by the end of October as agreed with the project coordinator. The GIS information schema defined by the Globus Project will be used as a starting point. Then additional information providers will be developed and the related information will be specified. Finally, the information schema will be updated. The following sections describe the steps needed to augment the Globus GIS with an information provider; we give here a simple, real example that has been implemented at the University of Lecce. The document is organized as follows: in section 2.1 the Grid Object Specification (GOS) related to the example information provider is given; section 2.2 presents the source code needed to extract the information, and section 2.3 contains the description of the extended information schema. Finally, in section 2.4 we describe how to update the GIS configuration files.

# 2 Simple GIS Extension

We have chosen a simple attribute extension representing the number of processes running on a computational resource as an example of how to update the Globus GIS schema and modify the configuration files of the information server. As a first step for this purpose we need to define the information structure using GOS as the specification language. Then, a specific information provider will be developed to retrieve the information. In our case the information provider is a simple shell script that checks the jobs running on the machine and retrieve the number of processes, returning the information in LDIF (LDAP Data Interchange Format) format. Finally, it is necessary to update the information schema to reflect the additional GOS specification and to update the configuration files related to the information server.

## 2.1 GOS Specification

We have required to the Globus Project a subspace of the MDS OID (Object Identifier); the Globus project has kindly assigned to the GridLab project the OID 1.3.6.1.4.1.3536.2.6.3536.9 and to avoid name clashes we will use the prefix ”GridLab”. The attribute will be inserted into a new object class named ”GridLabWorkLoad” that will collect all of the attributes related to the workload of a machine. The new attribute and the new object class are specified using GOS as formalism for the object specification as shown in the following. The attribute we are going to insert, named ”GridLab-Computer-Processes-number”, is specified as having numeric type and being a single valued attribute. We recall here that in the future GOS will probably be replaced by CIM (Common Information Model), a model for describing overall management information in a network/enterprise environment that is being developed by the Global Grid Forum. CIM is comprised of a Specification and a Schema. The Specification defines the details for integration with other management models while the Schema provides the actual model descriptions.

```
NAMESPACE GridLab {
   DESCRIPTION {
   GridLab Metacomputing directory services
   }
   OBJECTCLASS GridLabWorkLoad {
      OID { 1.3.6.1.4.1.3536.2.6.3536.9.1 }
      DESCRIPTION { Work load of a computer }
      MUST CONTAIN {
         GridLab-Computer-Processes-number ::  single-valued,
numeric,
         { number of active processes}
      }
   }
}

ATTRIBUTE GridLab-Computer-Processes-number {
   OID { 1.3.6.1.4.1.3536.2.6.3536.9.1.0.1}
DESCRIPTION { number of active processes}
}
```

## 2.2 Information Provider

An information provider is an executable module that extracts the information from the resource and returns it to the information server in LDIF format, it is platform dependent and thus it may differs depending on the operating system. In our simple case the information provider must extract the number of processes running on the machine, and has been developed on the linux/intel platform as a shell script. To determine the number of processes we have used the shell command
"ps -aux | wc -l"

```
#!  /bin/bash
# default libexecdir used to bootstrap scripts
libexecdir=$GLOBUS_LOCATION/libexec
# load GRIS common code and initialization
.  $libexecdir/grid-info-common
###

emit_resource_platform () {
cat < < EOF
${_line_class_dn}dn:  ${_suffix}
${_line_class_oc}objectclass:  GridLabWorkLoad
${_line_class_av}GridLab-Computer-Processes-number:
${_processes_number}
EOF }

probe_resource_platform ()
{
   _processes_number=‘ps aux |wc -l‘

   # run this in initialized subshell environment
```

```
    emit_resource_platform
}
```

The script extracts the number of processes and returns it to the information server in LDIF format as shown in the following

dn: GridLab-Workload=processes,Mds-Host-hn=lupiae.unile.it,Mds-Vo-name=local, o=grid
objectclass: GridLabWorkLoad
GridLab-Computer-Processes-number: 59

## 2.3 Schema Definition

The information Schema provides a unique naming schema for the specified object classes, a specification of the attributes that an object class must or may contain, a data type for each attribute and the information needed to match a filter against the attribute found in an entry. The object class is characterized by an OID that uniquely identifies it. The OID number is assigned by the Grid Forum. In our case we have used our reserved OID 1.3.6.1.4.1.3536.2.6.3536.9 as base definition and we have choosen an unique ID (1) for our object class. The object class is also specified by:

NAME: is the name of the object class

SUP: is the name of the parent object class

MUST: is the list of attribute types that are required in the object class

The attribute is characterized by an OID. It is composed of its class object OID and an unique ID (0.1) The attribute is also specified by:

NAME: is the name of the attribute

DESC: gives a short description of meaning of the attribute. It typically includes a concise explanation or a statement of purpose of the attribute and its usage

EQUALITY: represents the typology of comparison

SYNTAX: represents an elementary data type such as numeric, cis (case insensitive string), boolean, ces (case exact string). The attribute syntaxes are described in more detail in RFC 2252. In our case the attribute syntax is numeric.

SINGLE-VALUE: denotes that the attribute may appear only once in an entity

```
objectclass ( 1.3.6.1.4.1.3536.2.6.3536.9.1
   NAME 'GridLabWorkLoad'
   SUP 'Mds'
   MUST GridLab-Computer-Processes-number
)

attributetype ( 1.3.6.1.4.1.3536.2.6.3536.9.1.0.1
   NAME 'GridLab-Computer-Processes-number'
   DESC 'Number of processes'
   EQUALITY integerMatch
   SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
   SINGLE-VALUE
)
```

## 2.4 Server Configuration

The GIS structure is based on LDAP servers, so we need to update the configuration files of the LDAP server in order to activate the new information provider. In the configuration file is present an entry for each information provider. The entry indicates the distinguished name of the directory that will contain the information published by the provider, the location of the information provider module, the arguments to be passed, and the time validity for the cached value.

```
# generate Processes number
dn:  GridLab-Workload=processes, Mds-Host-hn=lupiae.unile.it,
Mds-Vo-name=local, o=grid
objectclass:  GlobusTop
objectclass:  GlobusActiveObject
objectclass:  GlobusActiveSearch
type:  exec
path:  /usr/local/globus/libexec
base:  grid-info-processes-number
args:  -dn GridLab-Workload=processes,Mds-Host-
hn=lupiae.unile.it,Mds-Vo-name=local,o=grid
cachetime:  10
timelimit:  20
sizelimit:  1
```