

EUROPEAN MIDDLEWARE INITIATIVE

EMI YAIM CORE v. 5.1.1

Document Version:	5.1.1
EMI Component Version:	5.1.1
Date:	14.02.2013

Table of Contents

YAIM in EMI Functional Description.....	1
Introduction.....	1
YAIM directory structure.....	1
Configuration flow in YAIM.....	1
YAIM configuration variables.....	1
How to build your own YAIM module.....	2
References.....	3

YAIM in EMI Functional Description

Introduction

YAIM (Yet Another Installation Method) is a software to configure grid services. The aim of YAIM is to provide simple configuration methods that can be used to set up uniform grid sites but can also be adapted and extended to meet the needs of larger sites. To ensure that local administrators can adapt YAIM, it has been implemented as a set of bash scripts. To support the component based release model within the EGEE project, YAIM has been modularized and a YAIM core is supplemented by component specific scripts distributed as RPMs.

YAIM directory structure

When a YAIM module is installed, the following directory structure is created under `/opt/glite/yaim`: *node-info.d* contains a set of files per node type with the format name `glite-node_type`. These files contain a list of functions to configure the corresponding node type. *functions* contains the functions that configure each node type. They are all bash scripts. They have the format `configure_function_name`. *functions/local* contains functions defined by the site administrators. These functions can override the ones provided by YAIM if the same function name is used. *defaults* contains `.pre` and `.post` files to be sourced before and after the files that contain the configuration variables. *bin* contains the main `yaim` executable. *log* contains the `yaim` log file called `yaimlog`. *examples* contains examples of configuration files and other files like `users.conf` and `groups.conf` needed in the configuration of the node types.

Configuration flow in YAIM

When configuring a node type, the different configuration files are sourced in a specific order. In the following example we are configuring a node type called LFC, in the host `lxb001.cern.ch` and using the VO `atlas`. Our `siteinfo` folder is under `/root/siteinfo`:

1. `/opt/glite/yaim/defaults/site-info.pre`
2. `/opt/glite/yaim/defaults/glite-lfc.pre`
3. `/root/siteinfo/site-info.def`
4. `/root/siteinfo/services/glite-lfc`
5. `/opt/glite/yaim/defaults/site-info.post`
6. `/opt/glite/yaim/defaults/glite-lfc.post`
7. `/root/siteinfo/nodes/lxb001.cern.ch`
8. `/root/siteinfo/vo.d/atlas`
9. `/opt/glite/yaim/node-info.d/glite-lfc`

Files in red are provided by the `yaim` core module. Files in blue are provided by the `yaim lfc` module. All of them are mandatory. The remaining files are optional and can be added by the system administrator. Moreover, the system administrator can also put `/root/siteinfo/services/glite-lfc` variables in `site-info.def` to use one central file.

YAIM configuration variables

In the `/opt/glite/yaim/examples/siteinfo` directory a set of examples for the needed configuration files can be found. This location is world readable and therefore site administrators should find a safer path for the configuration files but always under the same folder. The configuration folder should contain:

- *site-info.def*: contains a list of configuration variables in the format of key-value pairs. It's a

mandatory file and it's a parameter passed to the YAIM command. It can contain a complete list of all the variables needed to configure a site or a minimum set of variables common to all node types.

Optionally, the configuration folder can contain:

- *services*: contains a file per node type with the format `glite-node-type`. The file contains a list of configuration variables specific to that node type. It can be used together with the main `site-info.def` to structure the configuration variables in per node basis.
- *nodes*: contains a file per host with the format `hostname.domain_name`. The file contains host specific variables that are different from one host to another in a certain site.
- *vo.d*: contains a file per VO with the format `vo_name`.

The optional folders are meant to help system administrators to organise the configuration in a more structured way but it's up to them to keep the site configuration in `site-info.def`.

There is also an example of `users.conf` and `groups.conf` files under `/opt/glite/yaim/examples/`. These files can be placed at any location since their path is specified in the variables `USERS_CONF` and `GROUPS_CONF` in `site-info.def`.

How to build your own YAIM module

In order to create your own YAIM module you have to follow these steps:

- create the node type function list under `node-info.d/`
- define the corresponding functions under `functions/`
- define node type specific configuration files under `services/`

YAIM command

```
Usage: /opt/glite/yaim/bin/yaim <action> <parameters>
Actions:
-c | --configure      : Configure already installed services.
                       Compulsory parameters: -s, -n
-r | --runfunction   : Execute a configuration function.
                       Compulsory parameters: -s, -f
                       Optional parameters : -n
-v | --verify        : Checks that the necessary variables
                       are all defined in site-info.def.
                       Compulsory parameters: -s n
-h | --help          : This is help
```

Parameters:

```
-s | --siteinfo      : Location of the site-info.def file
-n | --nodetype      : Name of the node type(s) to configure
-f | --function      : Name of the functions(s) to execute
```

Examples:

- Configuration:

```
/opt/glite/yaim/bin/yaim -c -s /root/siteinfo/site-info.def -n SE_dpm_mysql
```

- Running a function:

```
/opt/glite/yaim/bin/yaim -r -s /root/siteinfo/site-info.def -n creamCE -f config_users
```

- Verify your site-info.def:

```
/opt/glite/yaim/bin/yaim -v -s /root/siteinfo/site-info.def -n SE_storm_backend
```

References

- YAIM leaflet - <https://edms.cern.ch/file/865512/1/YAIMleaflet.pdf>
-

This topic: EMI > YaimEmiFunctionalDesc

Topic revision: r4 - 04-Mar-2013 - DoinaCristinaAiftimiei

Copyright &© by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback