

Table of Contents

FTS Configuration model and YAIM variables reference.....	1
Structure of the configuration file.....	2
Web server configuration.....	3
Agents configuration.....	4
Section 1: which agents are on which hosts.....	4
Agents naming conventions.....	4
Section 2: specify the type of each named agent.....	4
Sections 3, 4 and 5: specify agent parameters.....	5
YAIM variables reference.....	7
Notes.....	8
Oracle InstantClient location.....	8
Obsolete parameters.....	8
Migration script: update_channels.py.....	8
Explicitly setting an agent name.....	9
Searching for parameters.....	9

FTS Configuration model and YAIM variables reference.

This document applies to FTS version 2.2

Structure of the configuration file

The YAIM configuration file contains the information to configure the FTS web server (FTS) and the channel and VO agents (FTA).

Web server configuration

FTS web-service YAIM variables are all prefixed with `FTS_`. Any variable with this prefix that the script does not understand will cause it to fail (to catch spelling mis-types).

There is only one parameter which **MUST** be set: `FTS_DBURL`, which specifies the JDBC Oracle connection string. This is usually similar to the agents' `FTA_GLOBAL_DB_CONNECTSTRING` parameter, but not in every case, so it must be specified explicitly. It can be provided by your database administrator.

In case the FTS web service and the agents are configured in the same file, as is usually the case, it is not necessary to provide the DB username and password, since these are taken from the Agents parameters. The database connection parameters may be overridden if desired (for example if the web-service is using a different DB role from the agents).

Parameter name	Default (corresponding agent parameter)
<code>FTS_DB_TYPE</code>	Value of <code>FTA_GLOBAL_DBTYPE</code>
<code>FTS_DB_USER</code>	Value of <code>FTA_GLOBAL_DB_USER</code>
<code>FTS_DB_PASSWORD</code>	Value of <code>FTS_GLOBAL_DB_PASSWORD</code>

```
FTS_DB_TYPE=ORACLE
FTS_DB_USER=lcg_fts_prod_ws_w
FTS_DB_PASSWORD=yyyyyyyyy
```

By default, the web-service will publish its endpoint in BDII using its hostname. If instead, you wish to publish this using a DNS alias, set the parameter `FTS_HOST_ALIAS`.

```
# Node names
[...]
FTS_HOST=%FTS_WS_HOSTNAME%. $MY_DOMAIN
[...]

# BDII/GIP specific settings
[...]
BDII_FTS_URL="ldap://$FTS_HOST:2170/mds-vo-name=resource,o=grid"
[...]

# FTS config file for web-service
FTS_DBURL=... # The JDBC url for connecting to the DB
FTS_HOST_ALIAS=prod-fts-ws.cern.ch
```

WHAT IS FTS_HOST ???

Agents configuration

The agents configuration must provide information about:

1. Which agents are on which hosts
2. Type of each named agent
3. Global parameters, applied to all agents
 - ◆ Logging
 - ◆ DB connection details
4. Type specific parameters, applied to all agents of a given type
 - ◆ e.g. all 3rd party copy agents
5. Specific parameters for individual agents

Section 1: which agents are on which hosts

First, give symbolic names to the machines where the agents will be installed using the variable `FTA_MACHINES`:

```
FTA_MACHINES="ONE TWO"
```

Then, for each of the machines defined, specify the host name using the variable `FTA_AGENTS_${MACHINE}_HOSTNAME` and the list of agents that will run on that machine using the variable `FTA_AGENTS_${MACHINE}`:

```
FTA_AGENTS_ONE_HOSTNAME="fts001.cern.ch"
FTA_AGENTS_ONE="CERN-CERN DTEAM"
FTA_AGENTS_TWO_HOSTNAME="fts002.cern.ch"
FTA_AGENTS_TWO="CERN-RAL CERN-IN2P3"
```

Agents naming conventions

In order to be consistent and to ensure that the FTA channel discovery mechanism works, please observe the following rules when naming channels and VO agents

- The agent names should all be upper case, e.g. DTEAM or CERN-RAL
- For VO agents:
 - ◆ It is preferable if the VO agent names do not contain dashes
- For channel agents:
 - ◆ The channel agent names should specify source and destination separated with a single dash.
 - ◆ The source and destination should be simple site identifiers (they do not need to be GOC DB names) and should contain only alphanumeric characters. The name should match the pattern `[A-Z]*-[A-Z]*`.

Section 2: specify the type of each named agent

For each agent defined, specify its type. The type is either:

- URLCOPY for a 3rd party copy channel agent.
- SRMCOPY for an SRM copy channel agent.
- VOAGENT for a VO agent.

Construct the YAIM variable to use as `FTA_${agent-name}`. Since the underlying shell of YAIM is bash, agent names with a dash '-' should substitute an underscore instead, '_'. Continuing the previous example:

```

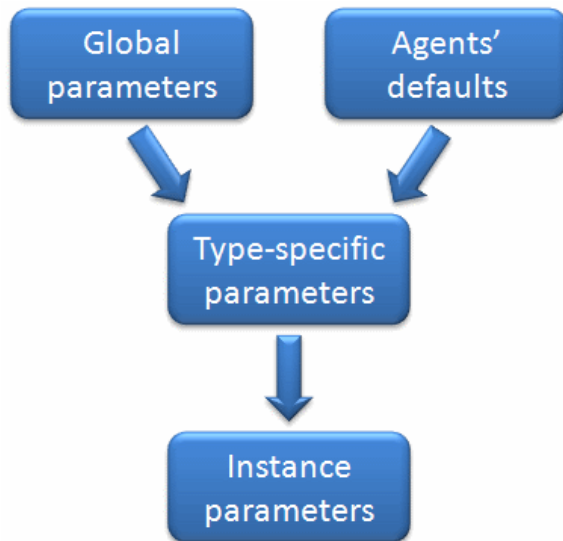
FTA_CERN_CERN="URLCOPY"
FTA_DTEAM="VOAGENT"
FTA_CERN_RAL="URLCOPY"
FTA_CERN_IN2P3="SRMCOPY"

```

Sections 3, 4 and 5: specify agent parameters

Most of the agents' configuration parameters may be specified at different levels:

- GLOBAL level
- TYPE level
- INSTANCE level



Agents come with reasonable defaults for most of the parameters, and for the vast majority of parameters the default should be fine.

Global parameters are applied to all agents and override the defaults.

Type-specific parameters are applied to all agents of a given type, e.g. all URLCOPY agents. These parameters override the default and global ones; typically, most of the parameters that need a non-default value are changed at this level.

Instance-specific parameters change the value of a variable for a single agent, and override default values, global and type-specific parameters. Such parameters should (ideally) be used rarely.

The general format of a YAIM variable is: `FTA_${SCOPE}_${PARAMNAME}`

SCOPE is either GLOBAL, TYPEDEFAULT_\${TYPE}, or the name of the agent (replacing dashes with underscores).

Example:

```

# All agents:
FTA_GLOBAL_LOG_PRIORITY="INFO"
# All URLCOPY channel agents:
FTA_TYPEDEFAULT_URLCOPY_LOG_PRIORITY="DEBUG"
# the named CERN-RAL agent (note the '-' converted to '_')
FTA_CERN_RAL_LOG_PRIORITY="INFO"

```

FtsConfiguration < EGEE < TWiki

Please note that DB connection parameters have no defaults, and should be specified at global level:

```
FTA_GLOBAL_DB_CONNECTSTRING=" (DESCRIPTION= (LOAD_BALANCE=no) (ADDRESS= (PROTOCOL=TCP) (HOST=lcgtestdb
FTA_GLOBAL_DB_USER=lcg_fts_prod_w
FTA_GLOBAL_DB_PASSWORD=xxxxxxxx
```

YAIM variables reference

FTS Yaim reference has been moved to [Trac](#).

Notes

Oracle InstantClient location

The site-info.def file should contain the ORACLE_LOCATION variable for Oracle InstantClient. For example, for the recommended 10.2.0.3 InstantClient:

```
ORACLE_LOCATION=/usr/lib/oracle/10.2.0.3
```

Obsolete parameters

The following parameters are now obsolete, and trying to set them through the YAIM template will result in an error. The table lists the parameter name, the corresponding field of the t_channel table in the database and the flag of the command line interface tool `glite-transfer-channel-set` that should be used to set the value.

Parameter name	Database field	glite-transfer-channel-set flag
GUC_TRANSFERTIMEOUT	urlcopy_tx_to (urlcopy channels) srmcopy_to (srmcopy channels)	--urlcopy-tx-to --srmcopy-to
GUC_HTTPTIMEOUT	http_to	--http-timeout
GUC_STREAMS	nostreams	-T
GUC_TCPBUFFERSIZE	tcp_buffer_size	-u
GUC_TCPBLOCKSIZE	blocksize	--block-size
GUC_SRMPUTTIMEOUT	urlcopy_put_to	--urlcopy-put-to
GUC_SRMGETTIMEOUT	urlcopy_get_to	--urlcopy-get-to
GUC_SRMPUTDONETIMEOUT	urlcopy_putdone_to	--urlcopy-putdone-to
GUC_SRMGETDONETIMEOUT	urlcopy_getdone_to	--urlcopy-getdone-to
GUC_TRANSFERMARKERTIMEOUT	urlcopy_txmarks_to	--urlcopy-tx-marks-to
GUC_SRMCOPYDIRECTION	srmcopy_direction	--srmcopy-direction
GUC_LOGLEVEL	tx_loglevel	--tx-log-level
GUC_SRMSTATUSTIMEOUT	srmcopy_refresh_to	--srmcopy-refresh-to

Migration script: update_channels.py

IMPORTANT NOTE: if you are upgrading from FTS 2.1, this script should not be necessary.

The `update_channels.py` script copies in the database the parameters defined in the channel agents configuration files. It is located under the same directory as the agents' configuration files,

```
${GLITE_LOCATION}/etc/glite-data-transfer-agents.d.
```

In FTS 2.1 it was not yet possible to set all channel parameters properly through the command line interface, so this script was run by YAIM whenever a channel was updated. With FTS 2.2 this is no longer necessary since the CLI allows setting all channel parameters, and YAIM will produce an error in case one of the obsolete parameters is set. Your database should be synchronized with the agents' configuration files if you are upgrading from FTS 2.1, and running this script should not be necessary, but it is still provided in case you want to run it manually before upgrading.

The script requires no parameters. You can run it simply with: `python update_channels.py` The script will:

- locate and read the channel agents configuration files.
- generate an sql update script for the t_channel table.

- run the sql script.

The following output files are produced:

- update_channels.log: the log file of the script (same contents as standard output).
- update_channels.oracle.log: the output of the sqlplus command
- update_channels.sql: the sql update script

In case an error occurs, please mail FtsSupport including the log files; the log files should contain no sensible information (database connection parameters), but you might want to check before sending them.

Explicitly setting an agent name

The mandatory Name parameter of the agents configuration is already set with a sensible default by the YAIM scripts. This parameter is used by the channel agents as the name of the managed channel and by the VO agents as the name of the managed VO.

In case you should set it explicitly, you should use the parameter `FTA_%AGENT%_AGENT_NAME`.

Searching for parameters

A prototype tool exists for searching for parameters and determining the correct YAIM variable:

```
cd /opt/glite/share/config/glite-data-transfer-agents/yaim
xsltproc -stringparam Param timeout find-param.xml ../glite-transfer-channel-agent-urlcopy-oracle
xsltproc -stringparam Prefix FTA_GLOBAL -stringparam Param timeout find-param.xml ../glite-transf
```

where "timeout" is the string to search for. The variable names and descriptions are searched.

The `Prefix` option allows you to get the full yaim variable with the appropriate scope prefix.

Last edit: UnknownUser on 2010-05-11 - 17:23

Number of topics: 1

Maintainer: AkosFrohner

This topic: EGEE > FtsConfiguration

Topic revision: r8 - 2010-05-11 - unknown



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

Ideas, requests, problems regarding TWiki? Ask a support question or Send feedback