Table of Contents

YAIM Values for FTS Version 1.5.

WebServer Configuration

For configuring the FTS WebServer, you need:

```
# 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
```

Where `%FTS_WS_HOSTNAME%` is the name of the host where the FTS Web Server is installed (in case you use dns aliases, put the name of the dns alias here)

In case the FTS WS and the Agents are configure din the same file (usually this is the case) you don't need to provide the DB username and password, since these are taken from the Agents parameters (see below). In case you have separate files for the WS and the Agent, you need to provide these values using the parameters:

```
FTS_DB_TYPE=ORACLE
FTS_DB_USER=...
FTS_DB_PASSWORD=...
```

FTA Configuration

The FTA configuration is slightly more complex. The first thing you have to specify are the hot that will be used by the FTA and what shoudl be the agents that will be installed into these hosts. For example, you can have:

```
FTA_MACHINES="ONE TWO FIVE"

FTA_AGENTS_ONE_HOSTNAME="fts101.cern.ch"
FTA_AGENTS_ONE="CERN-BNL BNL-CERN CERN-INFN INFN-CERN"

FTA_AGENTS_TWO_HOSTNAME="fts102.cern.ch"
FTA_AGENTS_TWO="CERN-FNAL FNAL-CERN CERN-RAL RAL-CERN"

FTA_AGENTS_FIVE_HOSTNAME="fts105.cern.ch"
FTA_AGENTS_FIVE="DTEAM ALICE ATLAS CMS LHCB OPS"
```

In that case, two hosts will be used for the ChannelAgents (fts101.crn.ch and fts102.cern.h) and one for the VOAgents (fts105.cern.ch). Please note that this example is taken from the production FTS at CERN, and doesn't force you to have the agents spread on different boxes (this choice mainly depends on the load you expect on your setup). You have then to specify the type of each agent, like:

```
FTA_CERN_BNL="URLCOPY"
FTA_BNL_CERN="URLCOPY"
FTA_CERN_INFN="URLCOPY"
FTA_INFN_CERN="URLCOPY"
FTA_CERN_FNAL="SRMCOPY"
FTA_FNAL_CERN="URLCOPY"
```

```
FTA_CERN_RAL="URLCOPY"
FTA_RAL_CERN="URLCOPY"

FTA_ATLAS="VOAGENT_PYTHON"
FTA_ALICE="VOAGENT_PYTHON"
FTA_LHCB="VOAGENT_PYTHON"
FTA_DTEAM="VOAGENT_PYTHON"
FTA_CMS="VOAGENT_PYTHON"
FTA_OPS="VOAGENT_PYTHON"
```

The naming convention is quite straightforward: `FTA_%INSTANCE_NAME%` where `%INTANCE_NAME%` is one of the names speficied in the `FTA_AGENTS_*` parameter. Please note that the character `"-"` shoudl be converted in `"_"`. The supported types are:

- Channel Agent types: `URLCOPY` (transfers are excuted using 3rd party gridftp copy), `SRMCOPY` (uses srmcopy)
- VO Agent types: `VOAGENT_PYTHON` (the VOAgent retry logic is provided by a python scrypt, recommeded!), `VOAGENT` (the VOAgent with the basic retry logic)

The only mandatory parameters are the Database type, username, password and connection string:

```
FTA_GLOBAL_DBTYPE=ORACLE
FTA_GLOBAL_DB_CONNECTSTRING=...
FTA_GLOBAL_DB_USER=...
FTA_GLOBAL_DB_PASSWORD=..
```

In addition, please leave the verbosity level of the log files to `INFO`:

```
FTA_GLOBAL_LOG_PRIORITY=INFO
```

The values apply to all the agents. In fact, we defined three diffenet scopes for the configuration parameters:

- `GLOBAL`: the values of the parameter are used for all the agents (VOs and Channels). The parameters mentioned above are example of global parameters. This kind of parameters can also be used to define default values that could be overwritten by more detailed scopes.
- `TYPEDEFAULT_%TYPE%`: the values are used for all the agents of the same type. The supported types are listed above: `URLCOPY`, `SRMCOPY`, `VOAGENT_PYTHON`, `VOAGENT`. Please note that in this context `URLCOPY` and `SRMCOPY` are considered as different types, even if both refer to ChannelAgents. The same concept also apply to `VOAGENT_PYTHON` and `VOAGENT`
- `%INSTANCE_NAME%`: the values are specific to the instance of the agent identified by `%INSTANCE_NAME%` (the name of the VO or the Channel the agent is responsible for).

In order to specify the FTA configuration paremeters, we adoped the following naming convention:

```
FTA_%SCOPE%_%PARAM_NAME%
```

where `%SCOPE%` is one of the values listed above and `%PARAM_NAME%` is the name of the parameter you want to set. For example, in case of `FTA_GLOBAL_LOG_PRIORITY`, `GLOBAL` is the scope and `LOG_PRIORITY` is the confguration paremeter name.

Usually, the paramters have a meaningful default value, but in some circumstances you may want to tune some of these values:

- Parameters related to ChannelAgents (`URLCOPY` or `SRMCOPY`):
    - `GUC_MAXTRANSFERS`: The maximum number of concurrent transfers the agent will process (act as a hard-limit on the number of files specified for a channel). Default is 50.

- GUC_TRANSFERTIMEOUT: The timeout in seconds for completing the transfer. In case of srmcopy transfer, the total timeout is this value multiplied by the number of files speficied in the srmcopy request. Default is 600 for URLCOPY and 0 (no timeout) for SRMCOPY. Recommended value is 1800 for both types.
- GUC_HTTPTIMEOUT: The http timeout for all the SOAP calls. Default is -1 (i.e. the gLite transfer-url-copy default applies: 40 seconds).

In addition, for ChannelAgents, we recommend you to set:

```
FTA_TYPEDEFAULT_%TYPE%_FSM_ENABLEHOLD=false     # since the "Hold" state is a VO policy,
FTA_TYPEDEFAULT_%TYPE%_AGENT_CANCEL_INTERVAL=60 # Check if there are ative transfer to ca
FTA_TYPEDEFAULT_%TYPE%_AGENT_DEFAULTINTERVAL=5  # Execute the ChannelAgent operations (fe
```

where %TYPE% is URLCOPY and SRMCOPY (please set both)

- Parameters related only to URLCOPY ChannelAgents :
  - GUC_STREAMS The maximum number of streams that would be used for a gridftp transfer (act as a hard-limit on the number of streams specifiedfor a channel). Default is 10.
  - GUC_SRMPUTTIMEOUT: The timeout for completing an SrmPut operation and rtriving a valid Turl to be used for the transfer. Default is 60. Recommended value is 180
  - GUC_SRMGETTIMEOUT: The timeout for completing an SrmGet operation and rtriving a valid Turl to be used for the transfer. Default is 60. Recommended value is 180
  - GUC_SRMPUTDONETIMEOUT: The timeout for releasing the Turl returned by the SrmPut call. Default is 60. Recommended value is 180
  - GUC_SRMGETDONETIMEOUT: The timeout for releasing the Turl returned by the SrmGet call. Default is 60. Recommended value is 180
  - GUC_TRANSFERMARKERSTIMEOUT: The timeout between two consequent transfer markers: if the gridftp server is not retruning markers with at least this frequency the transfer is considered stuck and therefore it will be aborted. Default is 120.

- Parameters related only to SRMCOPY ChannelAgents :
  - GUC_SRMCOPYDIRECTION: Identify the direction of the srmcopy transfer. The value can be either pull or push: in pull mode the srmCopy request is sent to the destination SRM, while in push mode is the source SRM that drives the transfer. Default is pull
  - GUC_MAXBULKSIZE: the maximum size for a SrmCopy bulk request. Default is 100
  - AGENT_CHECK_INTERVAL: the frequency for checking the status of active SrmCopy requests. Recommended value is 30.

In addition, for SRMCOPY ChannelAgent, in order to prevent an issue with dCache 1.6.6 we recommend you to set:

```
FTA_TYPEDEFAULT_SRMCOPY_ACTIONS_SURLNORMALIZATION=compact-with-port
```

- Parameters you need to set only for VOAGENT_PYTHON VOAgents :
  - PYTHON_PYTHONPATH: the paths were the python modules and strategies can be loated. Unless you have a setup that differs from te default one, please set this value to: `${GLITE_LOCATION}/lib/python2.2/site-packages:${GLITE_LOCATION}/lib/python/glite/fts`
  - ACTIONS_RETRYMODULE: the name of the python module that provides the retry logic for the VO. We recommend you to set this value to smarter_retry
  - ACTIONS_RETRYPARAMS: the parameter passed to the retry logic. The format of this string depends on the strategy module itself. For the smarter_retry module, this values looks like:

    ```
    "MaxFailures = 3 ; HoldEnabled = false ; OverwriteFailedFiles = true ; OverwriteExis
    ```

Hopefully, the parameters' names are self-explanatory. Please note that in case a VO requires to reduce the retry delay, you may need also to modify the parameter `AGENT_RETRY_INTERVAL`, that by deault is set to 60 seconds. For example, if a VO wants to have a Retry delay of 30 seconds, you may need to specify:

```
FTA_%VO%_AGENT_RETRY_INTERVAL=30
FTA_%VO%_ACTIONS_RETRYPARAMS="MaxFailures = 3 ; HoldEnabled = false ; OverwriteFaile
```

There are many other aspects of the FTA you could configured, but for a production server we suggest you to limit to the configuration parameters illustrated in this page; if these are not sufficient, you can have a look at the FTS documentation or contact fts-support.

Troubleshooting

In case the Yaim `configure_node` script returns an error like:

```
ERROR: The variable FTA_TYPEDEFAULT_%TYPE_%PARAM_NAME% was specified in the configuration file.
This is not used by any of the agents configured in the file.
```

This means that you're setting a property for a type that is not used (it usually happens when some `FTA_TYPEDEFAULT_SRMCOPY_*` properties are set but all the Channel agents are `URLCOPY`). The solution is to simply comment out or remove the lines concerning the unused parameters.

In case you're using diffente types of VOAgents at the same time, you're likely to receive this error:

```
New agent type VOAGENT used. Creating the default generator config files for it.
Writing generator input file for agent type VOAGENT to temporary file:
   /tmp/tmp.iGRUB18291/agenttype.VOAGENT.config.properties
Agent type VOAGENT overrides some defaults with the following variables:
FTA_TYPEDEFAULT_VOAGENT_ACTIONS_MAXFAILURES FTA_TYPEDEFAULT_VOAGENT_PYTHON_ACTIONS_RETRYMODULE

ERROR: The type parameter you have set - FTA_TYPEDEFAULT_VOAGENT_PYTHON_ACTIONS_RETRYMODULE - doe
       the type template file /opt/glite/share/config/glite-data-transfer-agents/glite-transfer-v
       I don't know what to do with this variable, so am aborting.
       Perhaps you mistyped the parameter name?
```

This is due to a bug in the adoped naming convention (please see #18265). In order to prevent this, please use only one type of VOAgent: we recommend to use the `VOAGENT_PYTHON`. In case a VO is not satisfied with the `smarter_retry` logic, you can restore the basic one by setting the property:

```
FTA_%VO%_ACTIONS_RETRYMODULE=basic_retry
```

Searching for parameters

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

```
 cd /opt/glite/share/config/glite-data-transfer-agents/yaim

xsltproc -stringparam Param timeout find-param.xsl ../glite-transfer-channel-agent-urlcopy-oracle

xsltproc -stringparam Prefix FTA_GLOBAL -stringparam Param timeout find-param.xsl ../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.

-- SteveTraylen - 12 Apr 2007

This topic: LCG > FtsYaimValues15
Topic revision: r4 - 2007-10-12 - GavinMcCance

Copyright &© 2008-2019 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.
Ideas, requests, problems regarding TWiki? Send feedback