

TTTGW Tango Cpp Class

Contents :

- [Description](#)
- [Properties](#)
- [Commands](#)
 - [State](#)
 - [Status](#)
- [Attributes](#)
- [States](#)

TTTGW Class Identification :

Contact : at esrg.fr - meyer
Class Family : Communication
Platform : Unix Like
Bus : Not Applicable
Manufacturer : none
Manufacturer ref. :

TTTGW Class Inheritance :

- [Tango::DeviceImpl](#)
 - TTTGW

TTTGW Class Description :

The Tine to Tango gateway class maps Tine devices into Tango devices.
The Tine naming schema uses as Tango device names with three fields as:

CONTEXT/SERVER_NAME/DEVICE_NAME

The names can be mapped one to one
from Tine to Tango. A Tine device has a set of properties to read and write data.

These Tine properties can be mapped, in most cases, to Tango attributes.

The basic idea of the Tine to Tango Gateway is based on the one to one name mapping. Every Tango device created for the gateway class **MUST** use a valid Tine device name.

During the `init_device()` the connection to the Tine device will be established with the given device name.

The properties of the Tine device are queried and mapped to Tango attributes. For every Tine property a dynamic attribute will be created under the same name. This is possible for all standard data types, but not for special Tine structure types.

Tine properties with no input and no output data are mapped into Tango commands.

Tine alarms are mapped into an ALARM state of the Tango device. The alarm description is available with the Tango status command.

LIMITATION: Tine properties with different input and output data types are not yet handled.

They must be mapped into Tango commands. Tango commands can be created dynamically, but they are seen always by all devices of the class. This makes sense only when all devices instantiated by the gateway class have the same interface. To insure the same interface, the Tine devices mapped must have the same `SERVER_NAME`.

To access the Tine control system, you have to specify the Tine Equipment Name Server (ENS) to be used. This can be done by setting the `TINE_HOME` environment variable to point to a local `cshosts.csv` file which contains the address of the ENS to be used. For further information please read the [Tine documentation](http://adweb.desy.de/mcs/tine/index.html) under <http://adweb.desy.de/mcs/tine/index.html>

TTGW Properties :

There is no class properties

There is no device properties

TTTGW Class Commands				
Name	Input type	Output type	Level	Description
State	DEV_VOID	DEV_STATE	OPERATOR	This command gets the device state (stored in its <i>device_state</i> data member) and returns it to the caller.
Status	DEV_VOID	CONST_DEV_STRING	OPERATOR	This command gets the device status (stored in its <i>device_status</i> data member) and returns it to the caller.

Command State :

This command gets the device state (stored in its *device_state* data member) and returns it to the caller.

State Definition		
Input Argument	Tango::DEV_VOID	none.
Output Argument	Tango::DEV_STATE	State Code
DisplayLevel	OPERATOR	..
Inherited	true	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

Command Status :

This command gets the device status (stored in its *device_status* data member) and returns it to the caller.

Status Definition		
Input Argument	Tango::DEV_VOID	none.
Output Argument	Tango::CONST_DEV_STRING	Status description
DisplayLevel	OPERATOR	..
Inherited	true	..
Abstract	false	..
Polling Period	Not polled	..
Command allowed for	All states	..

There is no attribute defined.

There is no dynamic attribute defined.

TTGW Class States	
Name	Description

ON	The Tine device is responding correctly on the network
UNKNOWN	No connection to the Tine device. The device is not responding on the network.
ALARM	The Tine device indicates one or more alarms. The alarm description is available with the status command.