



TANGO
Device
Server

MultiChannelEmittance User's Guide

MultiChannelEmittance Class

Revision: - Author: vedder
Implemented in C++ - CVS repository: ESRF

Introduction:

This class compute an average emittance based on several local emittance device server.

Class Inheritance:

- Tango::Device_4Impl
 - MultiChannelEmittance

Properties:

Device Properties		
Property name	Property type	Description
XemittanceDeviceList	Array of string	This is the list of Device of Xemittance that will be read for average XEmittance computation. This property is mapped on the XDeviceList and is updated on XDeviceList write.
ZemittanceDeviceList	Array of string	This is the list of Device of Zemittance that will be read for average ZEmittance computation. This property is mapped on the ZDeviceList and is updated on ZDeviceList write.
FullXemittanceDeviceList	Array of string	This is the exhaustive list of device from which X emittances could be read. This is not the list of device which are used to calculate average emittance (XEmittanceDeviceList is). This property is mapped on FullZDeviceList attribute.
FullZemittanceDeviceList	Array of string	This is the exhaustive list of device from which Z emittances could be read. This is not the list of device which are used to calculate average emittance (ZEmittanceDeviceList is). This property is mapped on FullZDeviceList attribute.

Device Properties Default Values:

Property Name	Default Values
XemittanceDeviceList	No default value
ZemittanceDeviceList	No default value
FullXemittanceDeviceList	No default value
FullZemittanceDeviceList	No default value

There is no Class properties.

Attributes:

Scalar Attributes			
Attribute name	Data Type	R/W Type	Expert
Xemittance	DEV_DOUBLE	READ	No
Zemittance	DEV_DOUBLE	READ	No
SdX : Standard deviation of X Values	DEV_DOUBLE	READ	No
SdZ : Standard deviation of Z Values	DEV_DOUBLE	READ	No

Spectrum Attributes			
Attribute name	Data Type	X Data Length	Expert
XDeviceList: This attribute is the list of XEmittance devices used to calculate average XEmittance. This attribute is linked to XEmittanceDeviceList property. Any write to this attribute will modify the property.	DEV_STRING	256	No
ZDeviceList: This attribute is the list of ZEmittance devices used to calculate average ZEmittance. This attribute is linked to ZEmittanceDeviceList property. Any write to this attribute will modify the property.	DEV_STRING	256	No
FullXdeviceList: This is the exhaustive list of device from which X emittances can be read. This is not the list of device which are used to calculate average emittance (XDeviceList is). This Attribute is only a mapping of the device property FullXemittanceDeviceList.	DEV_STRING	256	No
FullZdeviceList: This is the exhaustive list of device from which Z emittances can be read. This is not the list of device which are used to calculate average emittance (ZDeviceList is). This Attribute is only a mapping of the device property FullZemittanceDeviceList.	DEV_STRING	256	No

Commands:

More Details on commands....

Device Commands for Operator Level		
Command name	Argument In	Argument Out
Init	DEV_VOID	DEV_VOID
State	DEV_VOID	DEV_STATE
Status	DEV_VOID	CONST_DEV_STRING

1 - Init

- Description:** This commands re-initialise a device keeping the same network connection. After an Init command executed on a device, it is not necessary for client to re-connect to the device. This command first calls the device *delete_device()* method and then execute its *init_device()* method. For C++ device server, all the memory allocated in the *nit_device()* method must be freed in the *delete_device()* method. The language device desctructor automatically calls the *delete_device()* method.
- Argin:**
DEV_VOID : none.

- **Argout:**
DEV_VOID : none.
- **Command allowed for:**

2 - State

- **Description:** This command gets the device state (stored in its *device_state* data member) and returns it to the caller.
- **Argin:**
DEV_VOID : none.
- **Argout:**
DEV_STATE : State Code
- **Command allowed for:**

3 - Status

- **Description:** This command gets the device status (stored in its *device_status* data member) and returns it to the caller.
- **Argin:**
DEV_VOID : none.
- **Argout:**
CONST_DEV_STRING : Status description
- **Command allowed for:**

TANGO is an open source project hosted by :
SOURCEFORGE.NET®

Core and Tools : CVS repository on tango-cs project
Device Servers : CVS repository on tango-ds project