


TANGO Device Server

BeamLineEnergySwing User's Guide

BeamLineEnergySwing Class

Revision: release_0_0_1 - Author: langlois
Implemented in C++

Introduction:

A high-level device to synchronize the undulator, monochromator and mirror with respect to a given energy.

Class Inheritance:

- Tango::Device_3Impl
  - BeamLineEnergySwing

Properties:

Device Properties		
Property name	**Property type**	**Description**
DefaultCouplingNumber	Tango::DEV_SHORT	Used to parameterize the device when we start it.
UndulatorProxyName	Tango::DEV_STRING	The undulator device name
UndulatorEnergyAttributeName	Tango::DEV_STRING	The name of the energy attribute of the undulator device.
MonochromatorProxyName	Tango::DEV_STRING	
MonochromatorEnergyAttributeName	Tango::DEV_STRING	
MonochromatorExitAngleAttributeName	Tango::DEV_STRING	
MirrorProxyName	Tango::DEV_STRING	
MirrorThetaAngleName	Tango::DEV_STRING	
CommandStateName	Tango::DEV_STRING	
CommandStopName	Tango::DEV_STRING	

Device Properties Default Values:

Property Name	**Default Values**
DefaultCouplingNumber	No default value
UndulatorProxyName	No default value
UndulatorEnergyAttributeName	No default value
MonochromatorProxyName	No default value
MonochromatorEnergyAttributeName	No default value
MonochromatorExitAngleAttributeName	No default value
MirrorProxyName	No default value
MirrorThetaAngleName	No default value
CommandStateName	No default value
CommandStopName	No default value

There is no Class properties.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
energy: The beam line energy	DEV_DOUBLE	READ_WRITE	No
currentCouplingName: The current coupling in use	DEV_STRING	READ	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
Stop	DEV_VOID	DEV_VOID
ChangeCoupling	DEV_SHORT	DEV_VOID

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:**

4 - Stop

- **Description:**
- **Argin:**
  DEV_VOID :
- **Argout:**
  DEV_VOID :
- **Command allowed for:**

5 - ChangeCoupling

- **Description:**
- **Argin:**
  DEV_SHORT :
- **Argout:**
  DEV_VOID :
- **Command allowed for:**




TANGO Device Server

BeamLineEnergySwing User's Guide

BeamLineEnergySwing Class

Revision: release_0_0_1 - Author: langlois
Implemented in C++

Introduction:

A high-level device to synchronize the undulator, monochromator and mirror with respect to a given energy.

Class Inheritance:

- Tango::Device_3Impl
  - BeamLineEnergySwing

Properties:

Device Properties		
Property name	**Property type**	**Description**
DefaultCouplingNumber	Tango::DEV_SHORT	Used to parameterize the device when we start it.
UndulatorProxyName	Tango::DEV_STRING	The undulator device name
UndulatorEnergyAttributeName	Tango::DEV_STRING	The name of the energy attribute of the undulator device.
MonochromatorProxyName	Tango::DEV_STRING	
MonochromatorEnergyAttributeName	Tango::DEV_STRING	
MonochromatorExitAngleAttributeName	Tango::DEV_STRING	
MirrorProxyName	Tango::DEV_STRING	
MirrorThetaAngleName	Tango::DEV_STRING	
CommandStateName	Tango::DEV_STRING	
CommandStopName	Tango::DEV_STRING	

Device Properties Default Values:

Property Name	**Default Values**
DefaultCouplingNumber	No default value
UndulatorProxyName	No default value
UndulatorEnergyAttributeName	No default value
MonochromatorProxyName	No default value
MonochromatorEnergyAttributeName	No default value
MonochromatorExitAngleAttributeName	No default value
MirrorProxyName	No default value
MirrorThetaAngleName	No default value
CommandStateName	No default value
CommandStopName	No default value

There is no Class properties.

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
energy: The beam line energy	DEV_DOUBLE	READ_WRITE	No
currentCouplingName: The current coupling in use	DEV_STRING	READ	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
Stop	DEV_VOID	DEV_VOID
ChangeCoupling	DEV_SHORT	DEV_VOID

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:**

4 - Stop

- **Description:**
- **Argin:**
  DEV_VOID :
- **Argout:**
  DEV_VOID :
- **Command allowed for:**

5 - ChangeCoupling

- **Description:**
- **Argin:**
  DEV_SHORT :
- **Argout:**
  DEV_VOID :
- **Command allowed for:**

Frame Alert

This document is designed to be viewed using the frames feature. If you see this message, you are using a non-frame-capable web client.
Link to Non-frame version.




TANGO Device Server

BeamLineEnergySwing
Device Commands Description
BeamLineEnergySwing Class

Revision: release_0_0_1 - Author: langlois

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:**

4 - Stop

- **Description:**

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**

5 - ChangeCoupling

- **Description:**

- **Argin:**
  DEV_SHORT :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
