



**TANGO**  
Device  
Server

# Pinhole Camera Attenuator User's Guide

## PHCA attenuator Class

Revision: release\_2\_0\_0 - Author: malik  
Implemented in C++

### Introduction:

Device controlling the attenuator behind the pinhole camera. Gives the top of the transmitted energy spectrum

### Class Inheritance:

- Tango::Device\_3Impl
  - PHCA attenuator

## Properties:

<b>Device Properties</b>		
<b>Property name</b>	<b>Property type</b>	<b>Description</b>
<b>InterpolationMethod</b>	Tango::DEV_STRING	the method of interpolation. must be one of LINEAR, POLYNOMIAL, CUBIC_SPLINE or AKIMA
<b>Plate1Bounds</b>	Array of double	
<b>Plate1Energy</b>	Tango::DEV_DOUBLE	
<b>Plate1SigmaDifValHV</b>	Array of double	
<b>Plate2Bounds</b>	Array of double	
<b>Plate2Energy</b>	Tango::DEV_DOUBLE	
<b>Plate2SigmaDifValHV</b>	Array of double	
<b>PositionAttribute</b>	Tango::DEV_STRING	the name of the attribute where the axis position can be read
<b>PositionDevice</b>	Tango::DEV_STRING	the device name of the attenuator motorized axis
<b>TaperBound</b>	Tango::DEV_DOUBLE	the position limit under which the diffraction is constant (equal to first value of SigmaDiffractionValues)
<b>TaperEnergyValues</b>	Array of double	
<b>TaperRelativeValues</b>	Array of double	table of motor step values (relative to PositionValuesOrigin)
<b>TaperSigmaDifValH</b>	Array of double	value of the PSF for the $15\frac{1}{2}$ µm hole on convertor plane (in meters)
<b>TaperSigmaDifValV</b>	Array of double	

Device Properties Default Values:

Property Name	Default Values
InterpolationMethod	CUBIC_SPLINE
Plate1Bounds	No default value
Plate1Energy	No default value
Plate1SigmaDifValHV	No default value
Plate2Bounds	No default value
Plate2Energy	No default value
Plate2SigmaDifValHV	No default value
PositionAttribute	No default value
PositionDevice	No default value
TaperBound	No default value
TaperEnergyValues	No default value
TaperRelativeValues	No default value
TaperSigmaDifValH	No default value
TaperSigmaDifValV	No default value

**There is no Class properties.**

## Attributes:

<b>Scalar Attributes</b>			
Attribute name	Data Type	R/W Type	Expert
<b>SigmaDiffractionH</b>	DEV_DOUBLE	READ	No
<b>SigmaDiffractionV</b>	DEV_DOUBLE	READ	No
<b>Energy</b>	DEV_DOUBLE	READ	No

<b>Spectrum Attributes</b>			
Attribute name	Data Type	X Data Length	Expert
<b>positions</b>	DEV_DOUBLE	100000	No
<b>sigmaH</b>	DEV_DOUBLE	100000	No
<b>sigmaV</b>	DEV_DOUBLE	100000	No
<b>energyValues</b>	DEV_DOUBLE	100000	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:**

---

**ESRF - Software Engineering Group**



**TANGO**  
Device  
Server

# Pinhole Camera Attenuator User's Guide

## PHCA attenuator Class

Revision: release\_2\_0\_0 - Author: malik  
Implemented in C++

### Introduction:

Device controlling the attenuator behind the pinhole camera. Gives the top of the transmitted energy spectrum

### Class Inheritance:

- Tango::Device\_3Impl
  - PHCA attenuator

## Properties:

<b>Device Properties</b>		
<b>Property name</b>	<b>Property type</b>	<b>Description</b>
<b>InterpolationMethod</b>	Tango::DEV_STRING	the method of interpolation. must be one of LINEAR, POLYNOMIAL, CUBIC_SPLINE or AKIMA
<b>Plate1Bounds</b>	Array of double	
<b>Plate1Energy</b>	Tango::DEV_DOUBLE	
<b>Plate1SigmaDifValHV</b>	Array of double	
<b>Plate2Bounds</b>	Array of double	
<b>Plate2Energy</b>	Tango::DEV_DOUBLE	
<b>Plate2SigmaDifValHV</b>	Array of double	
<b>PositionAttribute</b>	Tango::DEV_STRING	the name of the attribute where the axis position can be read
<b>PositionDevice</b>	Tango::DEV_STRING	the device name of the attenuator motorized axis
<b>TaperBound</b>	Tango::DEV_DOUBLE	the position limit under which the diffraction is constant (equal to first value of SigmaDiffractionValues)
<b>TaperEnergyValues</b>	Array of double	
<b>TaperRelativeValues</b>	Array of double	table of motor step values (relative to PositionValuesOrigin)
<b>TaperSigmaDifValH</b>	Array of double	value of the PSF for the $15\frac{1}{2}$ µm hole on convertor plane (in meters)
<b>TaperSigmaDifValV</b>	Array of double	

Device Properties Default Values:

Property Name	Default Values
InterpolationMethod	CUBIC_SPLINE
Plate1Bounds	No default value
Plate1Energy	No default value
Plate1SigmaDifValHV	No default value
Plate2Bounds	No default value
Plate2Energy	No default value
Plate2SigmaDifValHV	No default value
PositionAttribute	No default value
PositionDevice	No default value
TaperBound	No default value
TaperEnergyValues	No default value
TaperRelativeValues	No default value
TaperSigmaDifValH	No default value
TaperSigmaDifValV	No default value

**There is no Class properties.**

## Attributes:

<b>Scalar Attributes</b>			
Attribute name	Data Type	R/W Type	Expert
<b>SigmaDiffractionH</b>	DEV_DOUBLE	READ	No
<b>SigmaDiffractionV</b>	DEV_DOUBLE	READ	No
<b>Energy</b>	DEV_DOUBLE	READ	No

<b>Spectrum Attributes</b>			
Attribute name	Data Type	X Data Length	Expert
<b>positions</b>	DEV_DOUBLE	100000	No
<b>sigmaH</b>	DEV_DOUBLE	100000	No
<b>sigmaV</b>	DEV_DOUBLE	100000	No
<b>energyValues</b>	DEV_DOUBLE	100000	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:**

---

**ESRF - Software Engineering Group**

# 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

---

# Pinhole Camera Attenuator Device Commands Description PHCA attenuator Class

Revision: release\_2\_0\_0 - Author: malik

## 1 - Init

- **Description:** This command re-initialises a device keeping the same network connection. After an Init command executed on a device, it is not necessary for the client to re-connect to the device.  
This command first calls the device *delete\_device()* method and then executes its *init\_device()* method.  
For C++ device server, all the memory allocated in the *init\_device()* method must be freed in the *delete\_device()* method.  
The language device destructor 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:**
- 

**ESRF - Software Engineering Group**