



TANGO Device Server

Diffractometer
User's Guide

Diffractometer Class

Revision: release_4_0_6 - Author: piccaf
Implemented in C++ - CVS repository: tango-ds

Introduction:

This class can be used to compute cristallography computation

Class Inheritance:

- Tango::Device_4Impl
  - Diffractometer

Properties:

Device Properties		
Property name	**Property type**	**Description**
TauConstant	Tango::DEV_DOUBLE	Definition of the Tau constant . Possible values are pi or 1 . Default : 1
RealAxisProxies	Array of string	
LambdaAttributeProxy	Tango::DEV_STRING	The name of the lambda attribute proxy. The diffractometer use it to read the lambda value from another Device
DiffractometerType	Tango::DEV_STRING	The Type of the diffractometer

Device Properties Default Values:

Property Name	**Default Values**
TauConstant	No default value
RealAxisProxies	mu:tests/diffabs/mu komega:tests/diffabs/komega kappa:tests/diffabs/kappa kphi:tests/diffabs/kphi delta:tests/diffabs/delta gamma:tests/diffabs/gamma
LambdaAttributeProxy	No default value
DiffractometerType	No default value

There is no Class properties.

States:

States	
Names	**Descriptions**
FAULT	HKLlibrary not initialised or at least 1 of the motor is in FAULT state
ALARM	At least 1 of the motor is in ALARM state
STANDBY	All motors are in STANDBY state
MOVING	At least one motor is in MOVING state
OFF	At least one motor is in OFF state

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
AnglesIdx: this attribut select the solution used to set the axes after an angle computation.	DEV_SHORT	READ_WRITE	No
Simulated: 0 : Device REALLY moves diffractometers motors 1 : Device only simulates calculations and DOES NOT move any motor	DEV_BOOLEAN	READ_WRITE	No
WaveLength: The wave length used by the diffractometer. Usually connected to an AttributeProxy.	DEV_DOUBLE	READ_WRITE	No
Crystal: The name of the crystal used by the diffractometer. the list of the available crystals is in the attribut crystals	DEV_STRING	READ_WRITE	No
A: crystal lattice a parameter	DEV_DOUBLE	READ	No
B: crystal lattice b parameter	DEV_DOUBLE	READ	No
C: crystal lattice c parameter	DEV_DOUBLE	READ	No
Alpha: crystal lattice alpha parameter	DEV_DOUBLE	READ	No
Beta: crystal lattice beta parameter	DEV_DOUBLE	READ	No
Gamma: crystal lattice gamma parameter	DEV_DOUBLE	READ	No
Ux: The x coordinate of the U matrix	DEV_DOUBLE	READ_WRITE	No
Uy: The y coordinate of the U matrix	DEV_DOUBLE	READ_WRITE	No
Uz: The z coordinate of the U matrix	DEV_DOUBLE	READ_WRITE	No
AStar: The a* crystal parameter	DEV_DOUBLE	READ	No
BStar: The b* crystal parameter	DEV_DOUBLE	READ	No
CStar: The c* crystal parameter	DEV_DOUBLE	READ	No
AlphaStar: The alpha* crystal parameter	DEV_DOUBLE	READ	No
BetaStar: The beta* crystal parameter	DEV_DOUBLE	READ	No
GammaStar: The beta* crystal parameter	DEV_DOUBLE	READ	No

Spectrum Attributes			
Attribute name	**Data Type**	**X Data Length**	**Expert**
CrystalNames	DEV_STRING	100	No
AxesNames: the names of the Dynamic axis attributes	DEV_STRING	100	No
pseudoAxesProxies	DEV_STRING	100	No

Image Attributes				
Attribute name	**Data Type**	**X Data Length**	**Y Data Length**	**Expert**
Angles: containe all the hkl computed values. simulated (0/1) h, k, l, axies angles	DEV_DOUBLE	20	1000	No
AnglesDegenerated: containe all the hkl computed values. simulated (0/1) h, k, l, axies angles	DEV_BOOLEAN	20	1000	No
AnglesRangeCheck: containe all the hkl computed values. simulated (0/1) h, k, l, axies angles	DEV_BOOLEAN	20	1000	No
Reflections	DEV_DOUBLE	12	50	No
ReflectionsAngles: matrix with the theoretical (lower triangular part) and mesured angles (upper triangular part) between reflections.	DEV_DOUBLE	2000	2000	No
UB: The UB matrix of the current crystal	DEV_DOUBLE	3	3	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
Abort	DEV_VOID	DEV_VOID
AddNewCrystal	DEV_STRING	DEV_VOID
AddReflection	DEVVAR_DOUBLEARRAY	DEV_VOID
AffineCrystal	DEV_STRING	DEV_DOUBLE
ComputeU	DEVVAR_LONGARRAY	DEV_VOID
ConfigureCrystal	DEVVAR_DOUBLEARRAY	DEV_VOID
CopyCurrentCrystalAs	DEV_STRING	DEV_VOID
CopyReflectionTo	DEVVAR_DOUBLESTRINGARRAY	DEV_VOID
DeleteCurrentCrystal	DEV_VOID	DEV_VOID
GetCrystalParameterValues	DEV_STRING	DEVVAR_DOUBLESTRINGARRAY
RemoveReflection	DEV_SHORT	DEV_VOID
SetCrystalParameterValues	DEVVAR_DOUBLESTRINGARRAY	DEV_VOID

Device Commands for Expert Level Only		
Command name	**Argument In**	**Argument Out**
Load	DEV_VOID	DEV_VOID
Save	DEV_VOID	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:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

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:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

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::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

4 - Abort

- **Description:** Stops the movement of all axis driven by the Diffractometer

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

5 - AddNewCrystal

- **Description:** Create a new crystal

- **Argin:**
  DEV_STRING : Name of the new crystal

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

6 - AddReflection

- **Description:** This commands requires the definition of 1 reflection defined by (h,k,l) coordinates. This coodinates are associated with the current angles configuration. TODO: dï¿½taillï¿½ le commentaire reflections i.e : h, k, l, relevance, enable/disable

- **Argin:**
  DEVVAR_DOUBLEARRAY : reflections i.e : h, k, l, relevance, enable/disable

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

7 - AffineCrystal

- **Description:** Compute affinement for the current crystal with available list of reflection.

- **Argin:**
  DEV_STRING : name of the crystal to fit

- **Argout:**
  DEV_DOUBLE : the fitness of the crystal after the affinement.

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

8 - ComputeU

- **Description:** Compute Matrix U from two reflections

- **Argin:**
  DEVVAR_LONGARRAY : a two elements array with the index of reflection to use

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

9 - ConfigureCrystal

- **Description:** Defines crystal lattice parameters. This information is mandatory for angles calculations. Angles units are degrees

- **Argin:**
  DEVVAR_DOUBLEARRAY : Crystal parameters : alpha,beta,gamma, A,B,C

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

10 - CopyCurrentCrystalAs

- **Description:** Copy the current crytal as another name

- **Argin:**
  DEV_STRING :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

11 - CopyReflectionTo

- **Description:** Copy one reflection from the current crystal to another crystal

- **Argin:**
  DEVVAR_DOUBLESTRINGARRAY : Double: Reflection number of the current crystal String : Name of the crystal where copy this reflection

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

12 - DeleteCurrentCrystal

- **Description:** Delete the current from this device

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

13 - GetCrystalParameterValues

- **Description:** Return all values of a parameter for the current crystal

- **Argin:**
  DEV_STRING : Name of parameter (see getParametersNames for the complete list of parameters)

- **Argout:**
  DEVVAR_DOUBLESTRINGARRAY : In the order : minimum value, maximum value, affinement enable

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

14 - Load (for expert only)

- **Description:** Load all crystals

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

15 - RemoveReflection

- **Description:** This commands removes reflection from the current cristal.

- **Argin:**
  DEV_SHORT : index of reflection to remove

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF

16 - Save (for expert only)

- **Description:** Save all crystals

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT

- Tango::ALARM
- Tango::STANDBY
- Tango::MOVING
- Tango::OFF

17 - SetCrystalParameterValues

- **Description:** Double : min value, max value, affinement enable

- **Argin:**
  DEVVAR_DOUBLESTRINGARRAY : String : Name of parameter ; Double : min value, max value, affinement enable

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::FAULT
  - Tango::ALARM
  - Tango::STANDBY
  - Tango::MOVING
  - Tango::OFF