



TANGO Device Server

Tango Generic ScanServer User's Guide

ScanServer Class

Revision: release_2_8_36 - Author: langlois
Implemented in C++ - CVS repository: tango-ds

Introduction:

This Device is a generic scan service for the Tango world. it allows to move **Actuators** (ie writable attributes) and to read **Sensors** (ie readable attributes) after a specified **Integration Time**

Class Identification:

- **Contact :** at synchrotron-soleil.fr - langlois
- **Class Family :** Process
- **Platform :** All Platforms
- **Bus :** Not Applicable

Class Inheritance:

- Tango::Device_4Impl
  - ScanServer

Properties:

Device Properties		
Property name	**Property type**	**Description**
CurrentFileBaseName	Tango::DEV_STRING	
CurrentFileExtension	Tango::DEV_STRING	
CurrentFilePath	Tango::DEV_STRING	
DataFileActivation	Tango::DEV_BOOLEAN	enables the ASCII data file
DataFileType	Tango::DEV_LONG	0 : ASCII 1 : Nexus
DataRecorder	Tango::DEV_STRING	device name for the Nexus Data Recorder
DataRecorderWriteMode	Tango::DEV_LONG	0 -> call WriteScanData in the DataRecorder device 1 -> write nexus file directly from ScanServer
DataRecorderSavingPeriod	Tango::DEV_DOUBLE	the period (in seconds) at which the DataRecorder->WriteScanData is called
PluginRootPath	Tango::DEV_STRING	The path to the plugins directory
DataFitter	Tango::DEV_STRING	a data fitter device to use for post scan behavior
SynchronizeData	Tango::DEV_BOOLEAN	Enable/Disable synchronize data coming from timebase and sensor. [default = true]

Device Properties Default Values:

Property Name	**Default Values**
CurrentFileBaseName	No default value
CurrentFileExtension	No default value
CurrentFilePath	No default value
DataFileActivation	false
DataFileType	No default value
DataRecorder	No default value
DataRecorderWriteMode	0
DataRecorderSavingPeriod	10
PluginRootPath	No default value
DataFitter	No default value
SynchronizeData	true

There is no Class properties.

States:

States	
Names	**Descriptions**
ON	Ready to scan
RUNNING	Scanning
STANDBY	Pause State : the scan is in pause mode, and will be wake up by an action Resume
FAULT	Scan is Fault : Maybe stopped anormally

Attributes:

Scalar Attributes			
Attribute name	**Data Type**	**R/W Type**	**Expert**
dataRecorderPartialMode: if true, the scanserver only calls WriteScanData on the DataRecorder, but not StartRecording, ..., EndRecording	DEV_BOOLEAN	READ_WRITE	No
filePath: the path where to save the data	DEV_STRING	READ_WRITE	No
fileBaseName: the name of the file (will be completed by the current hour and the extension)	DEV_STRING	READ_WRITE	No
fileExtension: the extension of the file	DEV_STRING	READ_WRITE	No
runName: a descriptive string for the run (will be recorded in Nexus)	DEV_STRING	READ_WRITE	No
pointNumber: Current number of points performed for the current scan	DEV_LONG	READ_WRITE	No
pointNumber2: Current number of points performed for the current scan	DEV_LONG	READ_WRITE	No
scanNumber: Write -> Number of scan to do Read -> Current number of scan performed for this run	DEV_LONG	READ_WRITE	No
actuatorsDelay: Waiting time after all actuators have finished moving, in seconds.	DEV_DOUBLE	READ_WRITE	No
timebasesDelay: Waiting time after all timebases have finished counting, in seconds.	DEV_DOUBLE	READ_WRITE	No
manualMode: if true, a PAUSE is done at the end of each step of the scan	DEV_BOOLEAN	READ_WRITE	No

automaticDirection	DEV_BOOLEAN	READ_WRITE	No
zigzag: If true, 2D mesh-scan are done in zigzag (even and odd lines are executed in inverse directions)	DEV_BOOLEAN	READ_WRITE	No
onTheFly: If true, do not wait for actuators at each step : instead, send the actuators to their last position and acquire sensor data on the fly during the movement	DEV_BOOLEAN	READ_WRITE	No
hwContinuous	DEV_BOOLEAN	READ_WRITE	No
hwContinuousNbPt	DEV_LONG	READ_WRITE	No
enableScanSpeed	DEV_BOOLEAN	READ_WRITE	No
scanType: 0 -> timescan 1 -> scan 1d 2 -> scan 2d	DEV_LONG	READ	No
contextValidation: The boolean attribute used to check the context validity	DEV_STRING	READ_WRITE	No
contextValidationErrorStrategy: Error Strategy of the Context Validation: 0 -> IGNORE : Scan stays at current point scan and continues with sensors integration 1 -> PAUSE (or WAIT) : WAIT the Validity of the context to be VALID 2 -> ABORT : The Run stops immediately	DEV_LONG	READ_WRITE	No
actuatorsTimeOut: Timeout for the Actuators positioning, in seconds	DEV_DOUBLE	READ_WRITE	No
actuatorsRetryCount: the number of times to try to write actuator value	DEV_LONG	READ_WRITE	No
actuatorsRetryTimeout: the time between two retry	DEV_DOUBLE	READ_WRITE	No
actuatorsErrorStrategy: Error Strategy of the Actuators: 1 -> IGNORE : Scan stays at current step scan and continues with sensors integration 3 -> NEXTPOINT : Scan jumps to the next point of scan 4 -> NEXTSCAN: Run jump to the next scan. 5 -> ABORT : Run stops immediately	DEV_LONG	READ_WRITE	No
timebasesTimeOut: Timeout for the Timebases integration, in seconds	DEV_DOUBLE	READ_WRITE	No
timebasesRetryCount: the number of times to try to start a timebase	DEV_LONG	READ_WRITE	No
timebasesRetryTimeout: the time between two retry	DEV_DOUBLE	READ_WRITE	No
timebasesErrorStrategy: Error Strategy of the Timebases: 1 -> IGNORE : Scan stays at current step scan and continues with sensors integration 2 -> RETRY : Retry to write the actuator 3 -> NEXTPOINT : Scan jumps to the next point of scan 4 -> NEXTSCAN: Run jump to the next scan. 5 -> ABORT : Run stops immediately	DEV_LONG	READ_WRITE	No
sensorsTimeOut: Timeout for the Sensors reading, in seconds	DEV_DOUBLE	READ_WRITE	No
sensorsRetryCount: the number of times to try to read sensor value	DEV_LONG	READ_WRITE	No
sensorsRetryTimeout: the time between two retry	DEV_DOUBLE	READ_WRITE	No

sensorsErrorStrategy: Error Strategy of the Sensors: 1 -> IGNORE : Scan stays at current step scan and continues with sensors integration 2 -> RETRY : Retry to write the actuator 3 -> NEXTPOINT : Scan jumps to the next point of scan 4 -> NEXTSCAN: Run jump to the next scan. 5 -> ABORT : Run stops immediately	DEV_LONG	READ_WRITE	No
hooksTimeOut: Timeout for the HookServers response, in seconds	DEV_DOUBLE	READ_WRITE	No
hooksRetryCount: the number of times to try to execute a hook	DEV_LONG	READ_WRITE	No
hooksRetryTimeout: the time between two retry	DEV_DOUBLE	READ_WRITE	No
hooksErrorStrategy: Error Strategy of the HookServers: 1 -> IGNORE : Scan stays at current step scan and continues with sensors integration 2 -> RETRY : Retry to write the actuator 3 -> NEXTPOINT : Scan jumps to the next point of scan 4 -> NEXTSCAN: Run jump to the next scan. 5 -> ABORT : Run stops immediately	DEV_LONG	READ_WRITE	No
afterRunActionType: 0 -> No action 1 -> First Scan Position 2 -> Prior Scan Position	DEV_LONG	READ_WRITE	No
afterRunActionSensor	DEV_LONG	READ_WRITE	No
afterRunActionActuator	DEV_LONG	READ_WRITE	No
afterRunActionActuatorValue	DEV_DOUBLE	READ_WRITE	No
runStartDate	DEV_STRING	READ	No
scanStartDate	DEV_STRING	READ	No
scanEndDate	DEV_STRING	READ	No
runEndDate	DEV_STRING	READ	No
scanDuration	DEV_STRING	READ	No
runDuration	DEV_STRING	READ	No
scanRemainingTime	DEV_STRING	READ	No
runRemainingTime	DEV_STRING	READ	No
scanElapsed	DEV_STRING	READ	No
runElapsed	DEV_STRING	READ	No
runCompletion: % of completion [0 - 100]	DEV_DOUBLE	READ	No
scanCompletion: % of completion of the current scan [0 - 100]	DEV_DOUBLE	READ	No
deadTime: the total dead time of the current scan	DEV_DOUBLE	READ	No
deadTimePercent: the percentage of dead time of the current scan	DEV_DOUBLE	READ	No
deadTimePerPoint: the dead time of the current scan, per scan point	DEV_DOUBLE	READ	No

Spectrum Attributes

Attribute name	Data Type	X Data Length	Expert
actuators: List of the actuators of the 1st dimension	DEV_STRING	1000	No
actuators2: List of the actuators of the 2nd dimension	DEV_STRING	1000	No
sensors: List of the Sensors	DEV_STRING	1000	No
timebases: List of the Timebases	DEV_STRING	1000	No
integrationTimes: Integration Time in seconds for all points of the trajectory	DEV_DOUBLE	100000	No
scanSpeed	DEV_DOUBLE	1000	No
preRunHooks: List of the Pre-Run Hooks names	DEV_STRING	1000	No
preScanHooks: List of the Pre-Scan Hooks names	DEV_STRING	1000	No
preStepHooks: List of the Pre-Step Hooks names	DEV_STRING	1000	No
postActuatorMoveHooks: List of the Post-Actuator-Move Hooks names	DEV_STRING	1000	No
postIntegrationHooks: List of the Post-Integration Hooks names	DEV_STRING	1000	No
postStepHooks: List of the Post Step Hooks names	DEV_STRING	1000	No
postScanHooks: List of the Post Scan Hooks names	DEV_STRING	1000	No
postRunHooks: List of the Post Run Hooks names	DEV_STRING	1000	No
actuatorsDataList	DEV_STRING	100000	No
actuators2DataList	DEV_STRING	100000	No
sensorsDataList	DEV_STRING	100000	No

Image Attributes

Attribute name	Data Type	X Data Length	Y Data Length	Expert
trajectories: Desired trajectory of the actuators of the 1st dimension. X -> trajectory Y -> actuators	DEV_DOUBLE	1000000000	1000	No
trajectories2: Desired trajectory of the actuators of the 2nd dimension. X -> trajectory Y -> actuators	DEV_DOUBLE	1000000000	1000	No
historic: Log messages of the last scan	DEV_STRING	5	100000	Yes

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
Start	DEV_VOID	DEV_VOID
Abort	DEV_VOID	DEV_VOID
Pause	DEV_VOID	DEV_VOID
Resume	DEV_VOID	DEV_VOID
Clean	DEV_VOID	DEV_VOID
ClearHistoric	DEV_VOID	DEV_VOID
ExecuteAction	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::ON
  - Tango::RUNNING
  - Tango::STANDBY
  - Tango::FAULT

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::ON
  - Tango::RUNNING
  - Tango::STANDBY
  - Tango::FAULT

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::ON
  - Tango::RUNNING
  - Tango::STANDBY
  - Tango::FAULT

4 - Start

- **Description:** Start the Run

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::ON

- Tango::RUNNING
- Tango::STANDBY
- Tango::FAULT

5 - Abort

- **Description:** Abort the Run

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
- Tango::ON
- Tango::RUNNING
- Tango::STANDBY
- Tango::FAULT

6 - Pause

- **Description:** Pause the scan

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
- Tango::ON
- Tango::RUNNING
- Tango::STANDBY
- Tango::FAULT

7 - Resume

- **Description:** Resume the scan when is Pause State

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::ON
  - Tango::RUNNING
  - Tango::STANDBY
  - Tango::FAULT

8 - Clean

- **Description:**

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::ON
  - Tango::RUNNING
  - Tango::STANDBY
  - Tango::FAULT

9 - ClearHistoric

- **Description:** Empty the log historic

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::ON
  - Tango::RUNNING
  - Tango::STANDBY
  - Tango::FAULT

10 - ExecuteAction

- **Description:** Executes the specified AfterRunAction

- **Argin:**
  DEV_VOID :

- **Argout:**
  DEV_VOID :

- **Command allowed for:**
  - Tango::ON
  - Tango::RUNNING
  - Tango::STANDBY
  - Tango::FAULT