TANGO
Device
Server

VacuumInterlockViewer
User's Guide

VacuumInterlockViewer Class

Revision: release_1_1_0 - Author: coquet
Implemented in C++

Introduction:

displays in status zone as plain text the interlocks generated by vacuum devices. 1 VacuumInterlockViewer per Vacuum PLC

Class Inheritance:

- Tango::Device_3Impl
  - VacuumInterlockViewer

Properties:

Device Properties		
Property name	**Property type**	**Description**
Url	Tango::DEV_STRING	Device name of the PLC Server
DBNumber	Tango::DEV_LONG	Device representative DataBlock number
InputOffset	Tango::DEV_LONG	Input offset of the beginning of the data in the DB IN BYTES
HardwarePollingTime	Tango::DEV_LONG	variables will be refreshed by hardware reading every HardwarePollingTime. given in : 1/10 sec. example : 34 sec gives 3.4 sec between hardware reading accesses Default : 10 (1 second
InterlockTexts	Array of string	interlock text to be displayed in status field : form : BYTE_OFFSET.BIT_OFFSET:text associated with this bit example : .... 0.7:flowmeter ANS_C01/VI/DEB.8 generated outgoing interlock 1.0:pirani gauge ANS_C01/VI/PI.5 generated an outgoing interlock ...
InterlocksPersistence	Tango::DEV_LONG	memorization of interlocks duration given in : sec. example : 3600 sec gives 1 hour persitence for memorized (disapeared) interlocks Default : 86400 equals 24 hours

Device Properties Default Values:

Property Name	**Default Values**
Url	No default value
DBNumber	No default value
InputOffset	No default value
HardwarePollingTime	No default value
InterlockTexts	No default value
InterlocksPersistence	86400

There is no Class properties.

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

VacuumInterlockViewer
User's Guide

VacuumInterlockViewer Class

Revision: release_1_1_0 - Author: coquet
Implemented in C++

Introduction:

displays in status zone as plain text the interlocks generated by vacuum devices. 1 VacuumInterlockViewer per Vacuum PLC

Class Inheritance:

- Tango::Device_3Impl
  - VacuumInterlockViewer

Properties:

Device Properties		
Property name	**Property type**	**Description**
Url	Tango::DEV_STRING	Device name of the PLC Server
DBNumber	Tango::DEV_LONG	Device representative DataBlock number
InputOffset	Tango::DEV_LONG	Input offset of the beginning of the data in the DB IN BYTES
HardwarePollingTime	Tango::DEV_LONG	variables will be refreshed by hardware reading every HardwarePollingTime. given in : 1/10 sec. example : 34 sec gives 3.4 sec between hardware reading accesses Default : 10 (1 second
InterlockTexts	Array of string	interlock text to be displayed in status field : form : BYTE_OFFSET.BIT_OFFSET:text associated with this bit example : .... 0.7:flowmeter ANS_C01/VI/DEB.8 generated outgoing interlock 1.0:pirani gauge ANS_C01/VI/PI.5 generated an outgoing interlock ...
InterlocksPersistence	Tango::DEV_LONG	memorization of interlocks duration given in : sec. example : 3600 sec gives 1 hour persitence for memorized (disapeared) interlocks Default : 86400 equals 24 hours

Device Properties Default Values:

Property Name	**Default Values**
Url	No default value
DBNumber	No default value
InputOffset	No default value
HardwarePollingTime	No default value
InterlockTexts	No default value
InterlocksPersistence	86400

There is no Class properties.

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

VacuumInterlockViewer
Device Commands Description
VacuumInterlockViewer Class

Revision: release_1_1_0 - Author: coquet

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