



TANGO
Device
Server

Linac User's Guide

Linac Class

Revision: Linac-Release_2_1 - Author: vedder
Implemented in C++ - CVS repository: ESRF

Introduction:

This class will manage the high level object controlling the linac. This object will allow the executions of sequences to put the linac in ON,OFF, STANDBY or LOW HEATING STATE.

Class Inheritance:

- Tango::Device_4Impl
 - Linac

Class Description:

Properties:

Device Properties		
Property name	Property type	Description
Dirname	Tango::DEV_STRING	The operation files for the linac will be load and saved in that directory and its subfolders.
Startfile	Tango::DEV_STRING	Name of the config file to be loaded at startup.
ICT_device	Tango::DEV_STRING	ICT device on which we could read the current used for beamstop functionality.
PCT_device	Tango::DEV_STRING	PCT device on which we could read the current used for beamstop functionality only if we can't read the taco resource sr/d-ct/1/name during the init. If we can read this taco resource, we will use this resource as the PCT device to be read.
Sequencer	Tango::DEV_STRING	Sequencer device name.
ICT_limit	Tango::DEV_DOUBLE	Value used at startup as SRCT limit when ct_device = ICT.
PCT_limit	Tango::DEV_DOUBLE	Value used at startup as SRCT limit when ct_device = PCT.
Daresbury	Tango::DEV_STRING	Name of the daresbury device.
Pss_interlocks_list	Array of double	list of pss interlocks.
Reset_devices_list	Array of string	List of devices to be reset when the command reset is called

Device Properties Default Values:

Property Name	Default Values
Dirname	No default value
Startfile	No default value
ICT_device	No default value
PCT_device	No default value
Sequencer	No default value
ICT_limit	4.60
PCT_limit	200.0
Daresbury	elin/intlk/0
Pss_interlocks_list	3.3, 5.2, 17.4
Reset_devices_list	elin/master/seq elin/intlk/0 elin/r-mod/reset elin/focus/klystron1 elin/focus/klystron2 elin/focus/beam1 elin/focus/beam2 elin/steer/1

There is no Class properties.

States:

States	
Names	Descriptions
ON	The Linac is ON.
STANDBY	*
INIT	Low Heating State
OFF	The linac is off
MOVING	*
FAULT	*
UNKNOWN	*
DISABLE	When a sequence failed with errors, the state will be set to DISABLE.
ALARM	There was an error during the execution of the last sequence. The linac is probably in an intermediate state.

Attributes:

Scalar Attributes			
Attribute name	Data Type	R/W Type	Expert
File: Configuration file	DEV_STRING	READ	No
SRCT_Limit: The set value of this attribute corresponds to the maximum allowed threshold for the current in the SR. If the SR current is greater than the set value of this attribute, the linac is put in Standby state and cannot be switched ON. The read value of this attribute is the present SR current value.	DEV_DOUBLE	READ_WRITE	No
CT_device: Current Transformer on which we read the current used for the beam_stop functionality. Possible values are PCT or ICT.	DEV_STRING	READ_WRITE	No
CT_Current	DEV_DOUBLE	READ	No
ShortStatus: A short description of the status of the linac.	DEV_STRING	READ	No
PCT_device: Name of the PCT device read via the property sr/d-ct/1/CurrentDeviceName	DEV_STRING	READ	No
ICT_device: Name of the PCT device read via the property sr/d-ct/2/CurrentDeviceName	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
On	DEV_VOID	DEV_VOID
Off	DEV_VOID	DEV_VOID
Standby	DEV_VOID	DEV_VOID
LowHeating	DEV_VOID	DEV_VOID
Abort	DEV_VOID	DEV_VOID
Reset	DEV_VOID	DEV_VOID
ReadDirName	DEV_VOID	DEV_STRING
LoadFile	DEV_STRING	DEV_VOID
SaveFile	DEV_STRING	DEV_VOID
ForceSave	DEV_STRING	DEV_VOID
GetPCTDeviceName	DEV_VOID	DEV_STRING
GetICTDeviceName	DEV_VOID	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:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING

- Tango::FAULT
- Tango::UNKNOWN
- Tango::DISABLE
- Tango::ALARM

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::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

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::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN

- Tango::DISABLE
- Tango::ALARM

4 - On

- **Description:**
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

5 - Off

- **Description:**
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

6 - Standby

- **Description:**
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

7 - LowHeating

- **Description:**
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

8 - Abort

- **Description:**
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

9 - Reset

- **Description:**
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

10 - ReadDirName

- **Description:** Command to retrieve the name of the directory under which the Linac configuration files are load and saved.
- **Argin:**
DEV_VOID :
- **Argout:**
DEV_STRING : The name of the directory under which the files are load and saved.
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

11 - LoadFile

- **Description:** Load the configuration file given in argument.
- **Argin:**
DEV_STRING : File to load
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

12 - SaveFile

- **Description:** This command will save all the present settings in a file. The argin is the name of the file. It's a relative pathname. For example, if 'test/16bunch' is given in argin, the file will be saved in dirname/test/16bunch. dirname is the dirname property of the server. You can use the command readDirName to know the value of dirname. These settings will be applied by the server from now.
- **Argin:**
DEV_STRING : Name of the file (relative to dirname property)
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

13 - ForceSave

- **Description:** This command will force the save of all the present settings in a file even if there was a problem when getting all the current set points. If a set point could not be read, the set point given in the last loaded or saved file will be used. The argin is the name of the file. It's a relative pathname. For example, if 'test/16bunch' is given in argin, the file will be saved in dirname/test/16bunch. dirname is the dirname property of the server. You can use the command readDirName to know the value of dirname. These settings will be applied by the server from now.
- **Argin:**
DEV_STRING : Name of the file (relative to dirname property)
- **Argout:**
DEV_VOID :
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN

- Tango::DISABLE
- Tango::ALARM

14 - GetPCTDeviceName

- **Description:** This command will return the name of the PCT device to be read. This name is obtained through the Tango database by reading the property sr/d-ct/1/CurrentDeviceName. After the call to this function, the PCT device used in the Linac server will be the one given by this property.
- **Argin:**
DEV_VOID : none
- **Argout:**
DEV_STRING : PCT device name
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN
 - Tango::DISABLE
 - Tango::ALARM

15 - GetICTDeviceName

- **Description:** This command will return the name of the ICT device to be read. This name is obtained through the Tango database by reading the property sr/d-ct/2/CurrentDeviceName. After the call to this function, the ICT device used in the Linac server will be the one given by this property.
- **Argin:**
DEV_VOID : none
- **Argout:**
DEV_STRING : PCT device name
- **Command allowed for:**
 - Tango::ON
 - Tango::STANDBY
 - Tango::INIT
 - Tango::OFF
 - Tango::MOVING
 - Tango::FAULT
 - Tango::UNKNOWN

- Tango::DISABLE
- Tango::ALARM

TANGO is an open source project hosted by :
SOURCEFORGE.NET®

Core and Tools : CVS repository on tango-cs project
Device Servers : CVS repository on tango-ds project