

# TangoInterface

Generated by Doxygen 1.5.6

Fri Dec 4 12:49:14 2009



# Contents

<b>1</b>	<b>Class Index</b>	<b>1</b>
1.1	Class List . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Class Documentation</b>	<b>5</b>
3.1	CTangoInterfaceApp Class Reference . . . . .	5
3.1.1	Constructor & Destructor Documentation . . . . .	5
3.1.1.1	CTangoInterfaceApp . . . . .	5
3.1.2	Member Function Documentation . . . . .	5
3.1.2.1	InitInstance . . . . .	5
3.2	TangoInterface Class Reference . . . . .	6
3.2.1	Constructor & Destructor Documentation . . . . .	7
3.2.1.1	TangoInterface . . . . .	7
3.2.1.2	~TangoInterface . . . . .	7
3.2.2	Member Function Documentation . . . . .	7
3.2.2.1	GetErrorMessage . . . . .	7
3.2.2.2	SetMotorSpeed . . . . .	8
3.2.2.3	GetMotorPosition . . . . .	8
3.2.2.4	EnableMotor . . . . .	8
3.2.2.5	JogMotor . . . . .	8
3.2.2.6	GetHomingStatus . . . . .	9
3.2.2.7	Home . . . . .	9
3.2.2.8	MoveMotor . . . . .	9
3.2.2.9	SetVariable . . . . .	10
3.2.2.10	SetVariable . . . . .	10
3.2.2.11	SetVariable . . . . .	10
3.2.2.12	SetVariable . . . . .	10

3.2.2.13	<a href="#">GetVariableFloat</a> . . . . .	11
3.2.2.14	<a href="#">GetVariableInt</a> . . . . .	11
3.2.2.15	<a href="#">GetVariableString</a> . . . . .	11
3.2.2.16	<a href="#">GetVariableBool</a> . . . . .	12
3.2.2.17	<a href="#">GetMotorVariableFloat</a> . . . . .	12
3.2.2.18	<a href="#">GetMotorVariableInt</a> . . . . .	12
3.2.2.19	<a href="#">GetMotorVariableString</a> . . . . .	13
3.2.2.20	<a href="#">GetMotorVariableBool</a> . . . . .	13
3.2.2.21	<a href="#">GetMotorAttributeStrings</a> . . . . .	13
<b>4</b>	<b>File Documentation</b>	<b>15</b>
4.1	<a href="#">/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_-20091201/Resource.h File Reference</a> . . . . .	15
4.2	<a href="#">/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_20091201/stdafx.cpp File Reference</a> . . . . .	16
4.3	<a href="#">/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_20091201/stdafx.h File Reference</a> . . . . .	17
4.3.1	Define Documentation . . . . .	17
4.3.1.1	<a href="#">_ATL_CSTRING_EXPLICIT_CONSTRUCTORS</a> . . . . .	17
4.3.1.2	<a href="#">VC_EXTRALEAN</a> . . . . .	17
4.4	<a href="#">/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_-20091201/TangoInterface.cpp File Reference</a> . . . . .	18
4.4.1	Define Documentation . . . . .	18
4.4.1.1	<a href="#">NUM_STRINGS</a> . . . . .	18
4.5	<a href="#">/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_-20091201/TangoInterface.h File Reference</a> . . . . .	19
4.5.1	Enumeration Type Documentation . . . . .	19
4.5.1.1	<a href="#">TANGO_ATTRIBUTES</a> . . . . .	19
4.6	<a href="#">/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_-20091201/TangoInterfaceApp.cpp File Reference</a> . . . . .	21
4.6.1	Variable Documentation . . . . .	21
4.6.1.1	<a href="#">theApp</a> . . . . .	21
4.7	<a href="#">/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_-20091201/TangoInterfaceApp.h File Reference</a> . . . . .	22
4.8	<a href="#">/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_20091201/targetver.h File Reference</a> . . . . .	23
4.8.1	Define Documentation . . . . .	23
4.8.1.1	<a href="#">_WIN32_IE</a> . . . . .	23
4.8.1.2	<a href="#">_WIN32_WINDOWS</a> . . . . .	23

---

4.8.1.3	<a href="#">_WIN32_WINNT</a> . . . . .	23
4.8.1.4	<a href="#">WINVER</a> . . . . .	23



# Chapter 1

## Class Index

### 1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">CTangoInterfaceApp</a> . . . . .	5
<a href="#">TangoInterface</a> . . . . .	6





## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_20091201/ <a href="#">Resource.h</a> . . .	15
/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_20091201/ <a href="#">stdafx.cpp</a> . . .	16
/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_20091201/ <a href="#">stdafx.h</a> . . . .	17
/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_- 20091201/ <a href="#">TangoInterface.cpp</a> . . . . .	18
/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_- 20091201/ <a href="#">TangoInterface.h</a> . . . . .	19
/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_- 20091201/ <a href="#">TangoInterfaceApp.cpp</a> . . . . .	21
/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_- 20091201/ <a href="#">TangoInterfaceApp.h</a> . . . . .	22
/homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface_20091201/ <a href="#">targetver.h</a> . . .	23



# Chapter 3

## Class Documentation

### 3.1 CTangoInterfaceApp Class Reference

```
#include <TangoInterfaceApp.h>
```

#### Public Member Functions

- [CTangoInterfaceApp \(\)](#)
- virtual BOOL [InitInstance \(\)](#)

#### 3.1.1 Constructor & Destructor Documentation

##### 3.1.1.1 CTangoInterfaceApp::CTangoInterfaceApp ()

#### 3.1.2 Member Function Documentation

##### 3.1.2.1 BOOL CTangoInterfaceApp::InitInstance () [virtual]

The documentation for this class was generated from the following files:

- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_-20091201/[TangoInterfaceApp.h](#)
- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_-20091201/[TangoInterfaceApp.cpp](#)

## 3.2 TangoInterface Class Reference

```
#include <TangoInterface.h>
```

### Public Member Functions

- [TangoInterface](#) (void)  
*Default constructor.*
- [~TangoInterface](#) (void)  
*Destructor.*
- bool [GetErrorMessage](#) (CString &strErrorMsg)  
*Gets an error message.*
- bool [SetMotorSpeed](#) (CString strMotorIdentifier, float fSpeed)  
*Sets the motor speed.*
- bool [GetMotorPosition](#) (CString strMotorIdentifier, float &fPosition)  
*Gets the motor position.*
- bool [EnableMotor](#) (CString strMotorIdentifier, bool bEnable)  
*Enables the motor.*
- bool [JogMotor](#) (CString strMotorIdentifier, bool bMovePositive)  
*Jog motor.*
- bool [GetHomingStatus](#) (CString strMotorIdentifier, int &nHomingStatus)  
*Get the homing status.*
- bool [Home](#) (CString strMotorIdentifier)  
*Issues the Home command.*
- bool [MoveMotor](#) (CString strMotorIdentifier, float fPosition)  
*Move motor.*
- bool [SetVariable](#) (CString strVariableName, CString strValue)  
*Sets a variable.*
- bool [SetVariable](#) (CString strVariableName, int nValue)  
*Sets a variable.*
- bool [SetVariable](#) (CString strVariableName, float fValue)  
*Sets a variable.*
- bool [SetVariable](#) (CString strVariableName, bool bValue)  
*Sets a variable.*
- bool [GetVariableFloat](#) (CString strVariableName, float &fValue)

*Gets a variable (float version).*

- bool [GetVariableInt](#) (CString strVariableName, int &nValue)  
*Gets a variable (integer version).*
- bool [GetVariableString](#) (CString strVariableName, CString &strReturn)  
*Gets a variable (string version).*
- bool [GetVariableBool](#) (CString strVariableName, bool &bValue)  
*Gets a variable (boolean version).*
- bool [GetMotorVariableFloat](#) (CString strMotorIdentifier, CString strVariableName, float &fValue)  
*Get a motor variable (float version).*
- bool [GetMotorVariableInt](#) (CString strMotorIdentifier, CString strVariableName, int &nValue)  
*Get a motor variable (integer version).*
- bool [GetMotorVariableString](#) (CString strMotorIdentifier, CString strVariableName, CString &strReturn)  
*Get a motor variable (string version).*
- bool [GetMotorVariableBool](#) (CString strMotorIdentifier, CString strVariableName, bool &bValue)  
*Get a motor variable (boolean version).*
- int [GetMotorAttributeStrings](#) (CString \*pstrVariableNames, int nNumAttributes)  
*Gets a specified list of motor variable names.*

### 3.2.1 Constructor & Destructor Documentation

#### 3.2.1.1 TangoInterface::TangoInterface (void)

Default constructor.

#### 3.2.1.2 TangoInterface::~~TangoInterface (void)

Destructor.

### 3.2.2 Member Function Documentation

#### 3.2.2.1 bool TangoInterface::GetErrorMessage (CString & strErrorMsg)

Gets an error message.

Will return an error string describing the most recent failure.

##### Parameters:

*strErrorMsg* CString to receive the error message string.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.2 bool TangoInterface::SetMotorSpeed (CString *strMotorIdentifier*, float *fSpeed*)**

Sets the motor speed.

**Parameters:**

*strMotorIdentifier* The motor identifier for the motor of interest.

*fSpeed* The speed value to set in counts per second.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.3 bool TangoInterface::GetMotorPosition (CString *strMotorIdentifier*, float &*fPosition*)**

Gets the motor position.

**Parameters:**

*strMotorIdentifier* The motor identifier for the motor of interest.

*fPosition* The motor position in counts.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.4 bool TangoInterface::EnableMotor (CString *strMotorIdentifier*, bool *bEnable*)**

Enables the motor.

**Parameters:**

*strMotorIdentifier* The motor identifier for the motor of interest.

*bEnable* true to enable it, false to disable it.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.5 bool TangoInterface::JogMotor (CString *strMotorIdentifier*, bool *bMovePositive*)**

Jog motor.

This will cause the motor of interest to jog continuously in the specified direction.

**Parameters:**

*strMotorIdentifier* The motor identifier for the motor of interest.

***bMovePositive*** Direction to Jog. True for Positive, False for negative.

**Returns:**

true if the command succeeds, false if the command fails.

### 3.2.2.6 bool TangoInterface::GetHomingStatus (CString *strMotorIdentifier*, int & *nHomingStatus*)

Get the homing status.

Minimum necessary implementation requires this function to return values 0 and 15. These corresponds to "homing not started" and "homing complete". If the motor does not require or allow homing this function, this function should return 15, home complete. /remarks>

**Parameters:**

***strMotorIdentifier*** The motor identifier for the motor of interest.

***nHomingStatus*** The homing status defined by the values 0-17 with the following definitions: HomingRoutineNotStartedYet = 0, WaitingforOther = 1, MovingToLimit = 2, IsAtLimit = 3, MovingToCoarseHome = 4, MovingToFixedOffset = 5, MovingToFineHome = 6, IsAtHome = 7, MovingToAlignedPos = 8, FollowedoutOnCoarse = 9, FollowedoutOnFixedOffset = 10, FollowedoutOnFineHome = 11, FollowedoutOnAlignedPos = 12, HitLimitOnFineHome = 13, HitLimitOnAlignedPos = 14, HomeComplete = 15, GeneralError = 16, PreconditionNotMet = 17

**Returns:**

true if the command succeeds, false if the command fails.

### 3.2.2.7 bool TangoInterface::Home (CString *strMotorIdentifier*)

Issues the Home command.

This command will cause the specified motor to initiate the homing process. If the motor does not require or allow homing this function, this function should return true.

**Parameters:**

***strMotorIdentifier*** The motor identifier for the motor of interest.

**Returns:**

true if the command succeeds, false if the command fails.

### 3.2.2.8 bool TangoInterface::MoveMotor (CString *strMotorIdentifier*, float *fPosition*)

Move motor.

**Parameters:**

***strMotorIdentifier*** The motor identifier for the motor of interest.

***fPosition*** The ABSOLUTE position in counts.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.9 bool TangoInterface::SetVariable (CString *strVariableName*, CString *strValue*)**

Sets a variable.

Sets the specified variable to the specified value.

**Parameters:**

*strVariableName* The variable name.

*strValue* The value as a string.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.10 bool TangoInterface::SetVariable (CString *strVariableName*, int *nValue*)**

Sets a variable.

Sets the specified variable to the specified value.

**Parameters:**

*strVariableName* The variable name.

*strValue* The value as an integer.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.11 bool TangoInterface::SetVariable (CString *strVariableName*, float *fValue*)**

Sets a variable.

Sets the specified variable to the specified value.

**Parameters:**

*strVariableName* The variable name.

*strValue* The value as a float.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.12 bool TangoInterface::SetVariable (CString *strVariableName*, bool *bValue*)**

Sets a variable.

Sets the specified variable to the specified value.

**Parameters:**

*strVariableName* The variable name.



*strValue* The value as a boolean.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.13 bool TangoInterface::GetVariableFloat (CString strVariableName, float & fValue)**

Gets a variable (float version).

Gets the value of the specified variable.

**Parameters:**

*strVariableName* The variable name.

*fValue* the variable value.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.14 bool TangoInterface::GetVariableInt (CString strVariableName, int & nValue)**

Gets a variable (integer version).

Gets the value of the specified variable.

**Parameters:**

*strVariableName* The variable name.

*nValue* the variable value.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.15 bool TangoInterface::GetVariableString (CString strVariableName, CString & strReturn)**

Gets a variable (string version).

Gets the value of the specified variable.

**Parameters:**

*strVariableName* The variable name.

*strReturn* The variable value.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.16 bool TangoInterface::GetVariableBool (CString *strVariableName*, bool & *bValue*)**

Gets a variable (boolean version).

Gets the value of the specified variable.

**Parameters:**

*strVariableName* The variable name.

*bValue* The variable value.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.17 bool TangoInterface::GetMotorVariableFloat (CString *strMotorIdentifier*, CString *strVariableName*, float & *fValue*)**

Get a motor variable (float version).

Get the value of the specified variable for the specified motor.

**Parameters:**

*strMotorIdentifier* The motor identifier for the motor of interest.

*strVariableName* The variable name.

*fValue* the variable value.

**Returns:**

true if the command succeeds, false if the command fails.

**3.2.2.18 bool TangoInterface::GetMotorVariableInt (CString *strMotorIdentifier*, CString *strVariableName*, int & *nValue*)**

Get a motor variable (integer version).

Get the value of the specified variable for the specified motor.

**Parameters:**

*strMotorIdentifier* The motor identifier for the motor of interest.

*strVariableName* The variable name.

*nValue* the variable value.

**Returns:**

true if the command succeeds, false if the command fails.

### 3.2.2.19 **bool TangoInterface::GetMotorVariableString** (CString *strMotorIdentifier*, CString *strVariableName*, CString & *strReturn*)

Get a motor variable (string version).

Get the value of the specified variable for the specified motor.

#### Parameters:

*strMotorIdentifier* The motor identifier for the motor of interest.

*strVariableName* The variable name.

*strReturn* The variable value.

#### Returns:

true if the command succeeds, false if the command fails.

### 3.2.2.20 **bool TangoInterface::GetMotorVariableBool** (CString *strMotorIdentifier*, CString *strVariableName*, bool & *bValue*)

Get a motor variable (boolean version).

Get the value of the specified variable for the specified motor.

#### Parameters:

*strMotorIdentifier* The motor identifier for the motor of interest.

*strVariableName* The variable name.

*bValue* the variable value.

#### Returns:

true if the command succeeds, false if the command fails.

### 3.2.2.21 **int TangoInterface::GetMotorAttributeStrings** (CString \*& *pstrVariableNames*, int *nNumAttributes*)

Gets a specified list of motor variable names.

This function is used to retrieve the variable names for a specific list of motor attributes. This list is as follows, in order: 0. Negative Limit – Variable is True when motor is at the negative-most limit 1. Positive Limit – Variable is True when motor is at the positive-most limit 2. In Position – Variable is True when the motor has stopped moving and it is within an error bound of its last commanded position 3. Following Error Occurred – Variable is True when the last commanded motion was aborted due to excessive following error 4. Enabled Status – Variable is True when motor is enabled 5. Position – Variable contains the current motor position in counts 6. Set Velocity – Variable contains the current motor target jog velocity

The *pstrVariableNames* parameter should be populated with the variable names for these motor attributes. This allows xradia software to use the GetMotorVariableXXXXX functions to check motor status.

It is an error to return more or less than the specified number of variables names or to return them in the wrong order.

#### Parameters:

*pstrVariableNames* a pointer to an array of CString objects which will receive the expected data.

*nNumAttributes* the size of the allocated array.

**Returns:**

the number of CStrings returned.

The documentation for this class was generated from the following files:

- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_20091201/[TangoInterface.h](#)
- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_20091201/[TangoInterface.cpp](#)

## **Chapter 4**

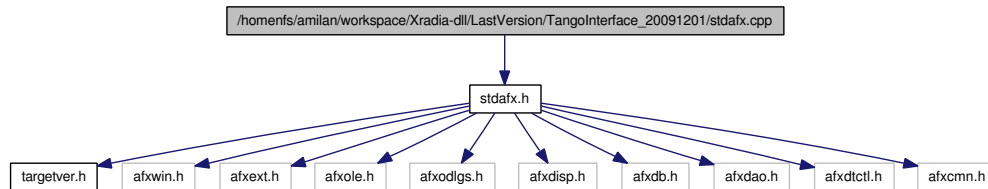
# **File Documentation**

### **4.1 /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_20091201/Resource.h File Reference**

## 4.2 /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_20091201/stdafx.cpp File Reference

```
#include "stdafx.h"
```

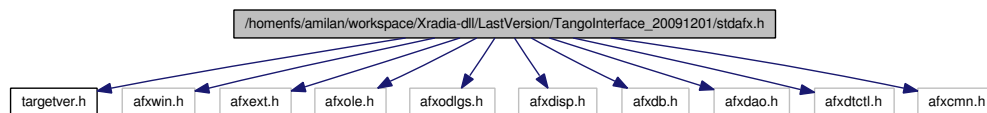
Include dependency graph for stdafx.cpp:



## 4.3 /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_20091201/stdafx.h File Reference

```
#include "targetver.h"
#include <afxwin.h>
#include <afxext.h>
#include <afxole.h>
#include <afxodlgs.h>
#include <afxdisp.h>
#include <afxdb.h>
#include <afxdao.h>
#include <afxdtctl.h>
#include <afxcmn.h>
```

Include dependency graph for stdafx.h:



This graph shows which files directly or indirectly include this file:



## Defines

- `#define VC_EXTRALEAN`
- `#define _ATL_CSTRING_EXPLICIT_CONSTRUCTORS`

### 4.3.1 Define Documentation

#### 4.3.1.1 `#define _ATL_CSTRING_EXPLICIT_CONSTRUCTORS`

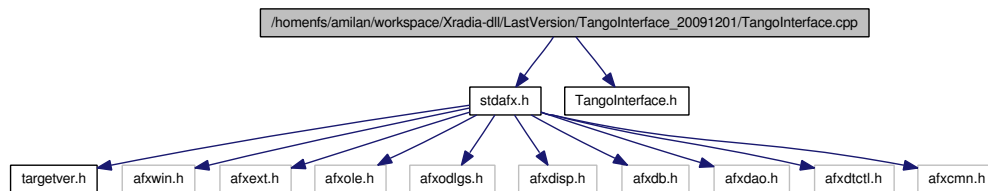
#### 4.3.1.2 `#define VC_EXTRALEAN`

## 4.4 /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_20091201/TangoInterface.cpp File Reference

```
#include "stdafx.h"
```

```
#include "TangoInterface.h"
```

Include dependency graph for TangoInterface.cpp:



### Defines

- `#define NUM_STRINGS 7`

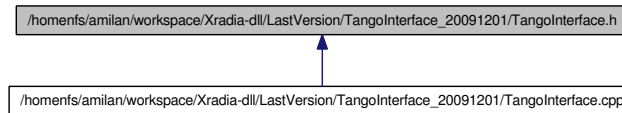
#### 4.4.1 Define Documentation

##### 4.4.1.1 `#define NUM_STRINGS 7`



## 4.5 /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_20091201/TangoInterface.h File Reference

This graph shows which files directly or indirectly include this file:



### Classes

- class [TangoInterface](#)

### Enumerations

- enum [TANGO\\_ATTRIBUTES](#) {  
NEG\_LIMIT = 0, POS\_LIMIT, IN\_POS, FOLLOWING\_ERROR,  
ENABLED, MOTOR\_POSITION, SET\_VELO, CURRENT\_SIZE\_TANGO\_ATTRIBUTES\_-  
ENUM }

*Tango interface.*

#### 4.5.1 Enumeration Type Documentation

##### 4.5.1.1 enum TANGO\_ATTRIBUTES

Tango interface.

One object will be used for contacting all Tango axes. The primary design of this class is to communicate with Tango motors. It also has the facility to communicate with any Tango variable for recording external data.

Units: Assumed in all functions that the units of the passed in values are the same. Xradia software will have a conversion factor that will allow the Xradia software to use Engineering units (ex. mm, deg, eV) independent of the actual hardware. In all descriptions below, the units will be referred to generically as "counts".

Motor Identifier: The motor identifier is a freeform string (configured per motor) that contains all information necessary to connect to a particular hosted motor. The definition of this string is left to the implementor of this interface.

#### Enumerator:

**NEG\_LIMIT**

**POS\_LIMIT**

**IN\_POS**

**FOLLOWING\_ERROR**

**ENABLED**

**MOTOR\_POSITION**

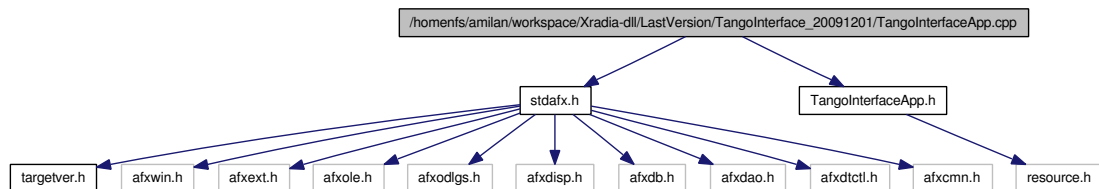
*SET\_VELO*

*CURRENT\_SIZE\_TANGO\_ATTRIBUTES\_ENUM*

```
#include "stdafx.h"
```

```
#include "TangoInterfaceApp.h"
```

Include dependency graph for TangoInterfaceApp.cpp:



## Variables

- [CTangoInterfaceApp theApp](#)

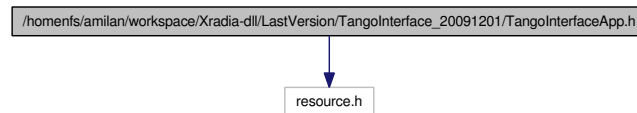
### 4.6.1 Variable Documentation

#### 4.6.1.1 CTangoInterfaceApp theApp

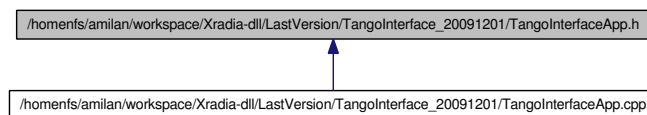
## 4.7 /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_20091201/TangoInterfaceApp.h File Reference

```
#include "resource.h"
```

Include dependency graph for TangoInterfaceApp.h:



This graph shows which files directly or indirectly include this file:

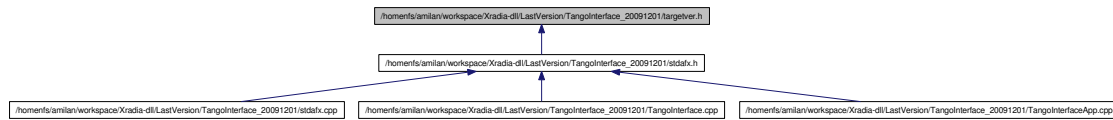


## Classes

- class [CTangoInterfaceApp](#)

## 4.8 /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_20091201/targetver.h File Reference

This graph shows which files directly or indirectly include this file:



### Defines

- #define WINVER 0x0502
- #define \_WIN32\_WINNT 0x0502
- #define \_WIN32\_WINDOWS 0x0410
- #define \_WIN32\_IE 0x0700

### 4.8.1 Define Documentation

4.8.1.1 #define \_WIN32\_IE 0x0700

4.8.1.2 #define \_WIN32\_WINDOWS 0x0410

4.8.1.3 #define \_WIN32\_WINNT 0x0502

4.8.1.4 #define WINVER 0x0502

# Index

- ~TangoInterface
  - TangoInterface, [7](#)
- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_ - 20091201/Resource.h, [15](#)
- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_ - 20091201/TangoInterface.cpp, [18](#)
- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_ - 20091201/TangoInterface.h, [19](#)
- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_ - 20091201/TangoInterfaceApp.cpp, [21](#)
- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_ - 20091201/TangoInterfaceApp.h, [22](#)
- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_ - 20091201/stdafx.cpp, [16](#)
- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_ - 20091201/stdafx.h, [17](#)
- /homenfs/amilan/workspace/Xradia-dll/LastVersion/TangoInterface\_ - 20091201/targetver.h, [23](#)
- \_ATL\_CSTRING\_EXPLICIT\_CONSTRUCTORS  
  stdafx.h, [17](#)
- \_WIN32\_IE  
  targetver.h, [23](#)
- \_WIN32\_WINDOWS  
  targetver.h, [23](#)
- \_WIN32\_WINNT  
  targetver.h, [23](#)
- CTangoInterfaceApp, [5](#)
  - CTangoInterfaceApp, [5](#)
  - InitInstance, [5](#)
- CURRENT\_SIZE\_TANGO\_ATTRIBUTES\_ -  
  ENUM  
  TangoInterface.h, [20](#)
- ENABLED  
  TangoInterface.h, [20](#)
- EnableMotor
  - TangoInterface, [8](#)
- FOLLOWING\_ERROR  
  TangoInterface.h, [20](#)
- GetErrorMessage
  - TangoInterface, [7](#)
- GetHomingStatus
  - TangoInterface, [9](#)
- GetMotorAttributeStrings
  - TangoInterface, [14](#)
- GetMotorPosition
  - TangoInterface, [8](#)
- GetMotorVariableBool
  - TangoInterface, [13](#)
- GetMotorVariableFloat
  - TangoInterface, [12](#)
- GetMotorVariableInt
  - TangoInterface, [13](#)
- GetMotorVariableString
  - TangoInterface, [13](#)
- GetVariableBool
  - TangoInterface, [12](#)
- GetVariableFloat
  - TangoInterface, [11](#)
- GetVariableInt
  - TangoInterface, [11](#)
- GetVariableString
  - TangoInterface, [12](#)
- Home
  - TangoInterface, [9](#)
- IN\_POS
  - TangoInterface.h, [20](#)
- InitInstance
  - CTangoInterfaceApp, [5](#)
- JogMotor
  - TangoInterface, [9](#)
- MOTOR\_POSITION
  - TangoInterface.h, [20](#)
- MoveMotor
  - TangoInterface, [10](#)

- NEG\_LIMIT
  - TangoInterface.h, [19](#)
- NUM\_STRINGS
  - TangoInterface.cpp, [18](#)
- POS\_LIMIT
  - TangoInterface.h, [19](#)
- SET\_VELO
  - TangoInterface.h, [20](#)
- SetMotorSpeed
  - TangoInterface, [8](#)
- SetVariable
  - TangoInterface, [10](#), [11](#)
- stdafx.h
  - \_ATL\_CSTRING\_EXPLICIT\_CONSTRUCTORS, [17](#)
  - VC\_EXTRALEAN, [17](#)
- TANGO\_ATTRIBUTES
  - TangoInterface.h, [19](#)
- TangoInterface, [6](#)
  - ~TangoInterface, [7](#)
  - EnableMotor, [8](#)
  - GetErrorMessage, [7](#)
  - GetHomingStatus, [9](#)
  - GetMotorAttributeStrings, [14](#)
  - GetMotorPosition, [8](#)
  - GetMotorVariableBool, [13](#)
  - GetMotorVariableFloat, [12](#)
  - GetMotorVariableInt, [13](#)
  - GetMotorVariableString, [13](#)
  - GetVariableBool, [12](#)
  - GetVariableFloat, [11](#)
  - GetVariableInt, [11](#)
  - GetVariableString, [12](#)
  - Home, [9](#)
  - JogMotor, [9](#)
  - MoveMotor, [10](#)
  - SetMotorSpeed, [8](#)
  - SetVariable, [10](#), [11](#)
  - TangoInterface, [7](#)
- TangoInterface.cpp
  - NUM\_STRINGS, [18](#)
- TangoInterface.h
  - CURRENT\_SIZE\_TANGO\_ATTRIBUTES\_ENUM, [20](#)
  - ENABLED, [20](#)
  - FOLLOWING\_ERROR, [20](#)
  - IN\_POS, [20](#)
  - MOTOR\_POSITION, [20](#)
  - NEG\_LIMIT, [19](#)
  - POS\_LIMIT, [19](#)
  - SET\_VELO, [20](#)
  - TANGO\_ATTRIBUTES, [19](#)
  - TangoInterfaceApp.cpp
    - theApp, [21](#)
  - targetver.h
    - \_WIN32\_IE, [23](#)
    - \_WIN32\_WINDOWS, [23](#)
    - \_WIN32\_WINNT, [23](#)
    - WINVER, [23](#)
  - theApp
    - TangoInterfaceApp.cpp, [21](#)
  - VC\_EXTRALEAN
    - stdafx.h, [17](#)
  - WINVER
    - targetver.h, [23](#)