

MhdOrientation

Generated by Doxygen 1.7.6.1

Sun Sep 8 2013 22:20:33

Contents

1	Namespace Index	1
1.1	Namespace List	1
2	Class Index	3
2.1	Class Hierarchy	3
3	Class Index	5
3.1	Class List	5
4	File Index	7
4.1	File List	7
5	Namespace Documentation	9
5.1	Mhd Namespace Reference	9
5.1.1	Detailed Description	10
5.1.2	Typedef Documentation	10
5.1.2.1	MhdBuilder	10
5.1.3	Function Documentation	10
5.1.3.1	MhdFileReader	10
5.1.3.2	operator<<	10
6	Class Documentation	11
6.1	Mhd::AIL Class Reference	11
6.1.1	Detailed Description	11
6.1.2	Constructor & Destructor Documentation	11
6.1.2.1	AIL	12
6.1.2.2	~AIL	12

6.1.3	Member Function Documentation	12
6.1.3.1	ConvertToRas	12
6.1.3.2	Create	12
6.2	Mhd::ASL Class Reference	12
6.2.1	Constructor & Destructor Documentation	13
6.2.1.1	ASL	13
6.2.1.2	~ASL	13
6.2.2	Member Function Documentation	13
6.2.2.1	ConvertToRas	13
6.2.2.2	Create	13
6.3	Mhd::LSA Class Reference	13
6.3.1	Constructor & Destructor Documentation	14
6.3.1.1	LSA	14
6.3.1.2	~LSA	14
6.3.2	Member Function Documentation	14
6.3.2.1	ConvertToRas	14
6.3.2.2	Create	14
6.4	Mhd::MhdFactory Class Reference	15
6.4.1	Detailed Description	15
6.4.2	Member Typedef Documentation	15
6.4.2.1	Collector	15
6.4.3	Constructor & Destructor Documentation	16
6.4.3.1	~MhdFactory	16
6.4.3.2	MhdFactory	16
6.4.4	Member Function Documentation	16
6.4.4.1	Get	16
6.4.4.2	Instance	16
6.4.4.3	Register	16
6.4.4.4	Registered	17
6.4.4.5	Unset	17
6.5	Mhd::MhdOrientation Class Reference	17
6.5.1	Detailed Description	18
6.5.2	Member Function Documentation	19
6.5.2.1	AO	19

6.5.2.2	C	19
6.5.2.3	ComputeAngles	19
6.5.2.4	ComputeRotation	19
6.5.2.5	ConvertToRas	19
6.5.2.6	Create	20
6.5.2.7	O	20
6.5.2.8	OrientationReader	20
6.5.2.9	OrientationWriter	20
6.5.2.10	R	20
6.5.3	Friends And Related Function Documentation	21
6.5.3.1	operator<<	21
6.5.4	Member Data Documentation	21
6.5.4.1	AnatomicalOrientation	21
6.5.4.2	Angles	21
6.5.4.3	BinaryData	21
6.5.4.4	BinaryDataByteOrderMSB	21
6.5.4.5	CenterOfRotation	21
6.5.4.6	CompressedData	21
6.5.4.7	CompressedDataSize	21
6.5.4.8	DimSize	21
6.5.4.9	ElementDataFile	21
6.5.4.10	ElementSpacing	21
6.5.4.11	ElementType	21
6.5.4.12	NDims	21
6.5.4.13	ObjectType	21
6.5.4.14	Offset	22
6.5.4.15	TransformMatrix	22
6.6	Mhd::MhdProxy< T > Class Template Reference	22
6.6.1	Detailed Description	22
6.6.2	Constructor & Destructor Documentation	22
6.6.2.1	MhdProxy	23
6.6.2.2	~MhdProxy	23
6.6.3	Member Function Documentation	23
6.6.3.1	Build	23

6.7	Mhd::MhdPythonOrientation Class Reference	23
6.7.1	Detailed Description	24
6.7.2	Member Function Documentation	24
6.7.2.1	AO	24
6.7.2.2	C	24
6.7.2.3	ComputeAngles	24
6.7.2.4	ComputeRotation	24
6.7.2.5	ConvertToRas	25
6.7.2.6	O	25
6.7.2.7	OrientationReader	25
6.7.2.8	OrientationWriter	25
6.7.2.9	R	25
6.8	Mhd::RAI Class Reference	26
6.8.1	Constructor & Destructor Documentation	26
6.8.1.1	RAI	26
6.8.1.2	~RAI	26
6.8.2	Member Function Documentation	26
6.8.2.1	ConvertToRas	26
6.8.2.2	Create	27
7	File Documentation	29
7.1	lib/include/MHD.hxx File Reference	29
7.1.1	Detailed Description	29
7.2	lib/include/MhdFactory.hxx File Reference	29
7.2.1	Detailed Description	30
7.3	lib/include/MhdOrientation.hxx File Reference	30
7.3.1	Detailed Description	31
7.3.2	Define Documentation	31
7.3.2.1	PI	31
7.4	lib/include/MhdOrientationRules.hxx File Reference	31
7.4.1	Detailed Description	32
7.4.2	Define Documentation	32
7.4.2.1	MHDIORIENTATIONRULES_HXX	32
7.5	lib/include/MhdProxy.hxx File Reference	32

7.5.1	Detailed Description	32
7.6	lib/include/MhdPythonOrientation.hxx File Reference	33
7.6.1	Detailed Description	33
7.6.2	Define Documentation	33
7.6.2.1	PI	33
7.7	lib/pymodule/mhd.py File Reference	33
7.7.1	Detailed Description	34
7.8	lib/src/MhdFactory.cxx File Reference	34
7.8.1	Detailed Description	34
7.9	lib/src/MhdFileReader.cxx File Reference	35
7.9.1	Detailed Description	35
7.10	lib/src/MhdOrientation.cxx File Reference	35
7.10.1	Detailed Description	36
7.11	lib/src/MhdOrientationRules.cxx File Reference	36
7.11.1	Detailed Description	36
7.12	lib/src/MhdPythonOrientation.cxx File Reference	37
7.12.1	Detailed Description	37
7.13	lib/src/MhdPythonWrapper.cxx File Reference	37
7.13.1	Detailed Description	38
7.13.2	Function Documentation	38
7.13.2.1	MhdOrientation_AO	38
7.13.2.2	MhdOrientation_C	38
7.13.2.3	MhdOrientation_ComputeAngles	39
7.13.2.4	MhdOrientation_ComputeRotation	39
7.13.2.5	MhdOrientation_ConvertToRas	39
7.13.2.6	MhdOrientation_O	39
7.13.2.7	MhdOrientation_OrientationReader	39
7.13.2.8	MhdOrientation_OrientationWriter	40
7.13.2.9	MhdOrientation_Python	40
7.13.2.10	MhdOrientation_R	40

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

mhd	??
Mhd	Namespace Mhd referred to the classes and methods defined in the	
	project MhdOrientation 9

Chapter 2

Class Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Mhd::MhdFactory	15
Mhd::MhdOrientation	17
Mhd::AIL	11
Mhd::ASL	12
Mhd::LSA	13
Mhd::RAI	26
mhd.MhdOrientation	??
Mhd::MhdProxy< T >	22
Mhd::MhdPythonOrientation	23

Chapter 3

Class Index

3.1 Class List

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

Mhd::AIL	
Derived class to perform AIL->RAS conversion	11
Mhd::ASL	12
Mhd::LSA	13
Mhd::MhdFactory	
The factory tha collects different MhdOrientation	15
Mhd::MhdOrientation	
Base class that contains methods to perform the RAS conversion	17
mhd.MhdOrientation	
Class MhdOrientation imported in Python	??
Mhd::MhdProxy< T >	
A proxy used to build an object MhdOrientation and to register it in the factory	22
Mhd::MhdPythonOrientation	
The class used for the Python interface	23
Mhd::RAI	26

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

lib/include/ MHD.hxx	
Header to be included to use the library	29
lib/include/ MhdFactory.hxx	
File containing the factory of MhdOrientations	29
lib/include/ MhdOrientation.hxx	
File containing the base class MhdOrientation	30
lib/include/ MhdOrientationRules.hxx	
File containing the derived classes to perform the orientation starting from the string AnatomicalOrientation stored	31
lib/include/ MhdProxy.hxx	
File containing a proxy to build the object MhdOrientation and that manage its automatic registration in the factory	32
lib/include/ MhdPythonOrientation.hxx	
Declaration of the class MhdPythonOrientation used for the Python interface	33
lib/pymodule/ mhd.py	
Module for the interface with Python using ctypes	33
lib/src/ MhdFactory.cxx	
Implementation of the factory	34
lib/src/ MhdFileReader.cxx	
File containing the function that reads a .mhd file to get the - AnatomicalOrientation parameter	35
lib/src/ MhdOrientation.cxx	
Implementation of the base class MhdOrientation	35
lib/src/ MhdOrientationRules.cxx	
Implementation of the class derived from MhdOrientation	36
lib/src/ MhdPythonOrientation.cxx	
Implementation of MhdPythonOrientation used for the Python inter- face	37

lib/src/MhdPythonWrapper.cxx	
Implementation of the wrapping in Python	37

Chapter 5

Namespace Documentation

5.1 Mhd Namespace Reference

Namespace [Mhd](#) referred to the classes and methods defined in the project [Mhd-Orientation](#).

Classes

- class [MhdFactory](#)
The factory tha collects different [MhdOrientation](#).
- class [MhdOrientation](#)
Base class that contains methods to perform the RAS conversion.
- class [AIL](#)
Derived class to perform AIL->RAS conversion.
- class [ASL](#)
- class [RAI](#)
- class [LSA](#)
- class [MhdProxy](#)
A proxy used to build an object [MhdOrientation](#) and to register it in the factory.
- class [MhdPythonOrientation](#)
The class used for the Python interface.

Typedefs

- typedef std::unique_ptr < [MhdOrientation](#) >(* [MhdBuilder](#))()
A typedef to the builder that returns a unique_ptr to a [MhdOrientation](#) object.

Functions

- `char * MhdFileReader (char *InputFile)`
Read from a file the AnatomicalOrientation parameter.
- `ostream & operator<< (ostream &out, const MhdOrientation &K)`

5.1.1 Detailed Description

Namespace `Mhd` referred to the classes and methods defined in the project `MhdOrientation`.

5.1.2 Typedef Documentation

5.1.2.1 `typedef std::unique_ptr<MhdOrientation> (* Mhd::MhdBuilder)()`

A typedef to the builder that returns a `unique_ptr` to a `MhdOrientation` object.

5.1.3 Function Documentation

5.1.3.1 `char * Mhd::MhdFileReader (char * InputFile)`

Read from a file the AnatomicalOrientation parameter.

Parameters

<i>InputFile</i>	Input .mhd file
------------------	-----------------

Returns

a string containing the orientation

5.1.3.2 `ostream& Mhd::operator<< (ostream & out, const MhdOrientation & K)`

Parameters

<i>out</i>	Ostream for .mhd file writing
<i>K</i>	The object used to write the .mhd file

Returns

ofstream to write the object

Chapter 6

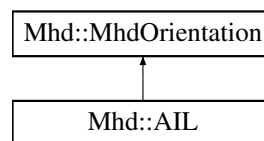
Class Documentation

6.1 Mhd::AIL Class Reference

Derived class to perform AIL->RAS conversion.

```
#include <MhdOrientationRules.hxx>
```

Inheritance diagram for Mhd::AIL:



Public Member Functions

- [AIL](#) ()
- [~AIL](#) ()
- void [ConvertToRas](#) (size_t i=1)
Perform orientation to RAS.
- virtual [MhdOrientation](#) * [Create](#) () const
Construction of the object returning a pointer to the base class.

6.1.1 Detailed Description

Derived class to perform AIL->RAS conversion.

6.1.2 Constructor & Destructor Documentation

6.1.2.1 Mhd::AIL::AIL ()

6.1.2.2 Mhd::AIL::~~AIL ()

6.1.3 Member Function Documentation

6.1.3.1 void Mhd::AIL::ConvertToRas (size_t $i = 1$) [virtual]

Perform orientation to RAS.

Parameters

i	i -th angle of rotation
-----	---------------------------

Implements [Mhd::MhdOrientation](#).

6.1.3.2 MhdOrientation * Mhd::AIL::Create () const [virtual]

Construction of the object returning a pointer to the base class.

Returns

Pointer to the base class

Implements [Mhd::MhdOrientation](#).

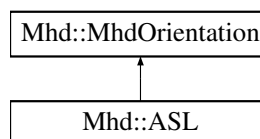
The documentation for this class was generated from the following files:

- [lib/include/MhdOrientationRules.hxx](#)
- [lib/src/MhdOrientationRules.cxx](#)

6.2 Mhd::ASL Class Reference

```
#include <MhdOrientationRules.hxx>
```

Inheritance diagram for Mhd::ASL:



Public Member Functions

- [ASL \(\)](#)
- [~ASL \(\)](#)

- void [ConvertToRas](#) (size_t i=1)
Perform orientation to RAS.
- virtual [MhdOrientation](#) * [Create](#) () const
Construction of the object returning a pointer to the base class.

6.2.1 Constructor & Destructor Documentation

6.2.1.1 [Mhd::ASL::ASL](#) ()

6.2.1.2 [Mhd::ASL::~~ASL](#) ()

6.2.2 Member Function Documentation

6.2.2.1 void [Mhd::ASL::ConvertToRas](#) (size_t i = 1) [virtual]

Perform orientation to RAS.

Parameters

<i>i</i>	i-th angle of rotation
----------	------------------------

Implements [Mhd::MhdOrientation](#).

6.2.2.2 [MhdOrientation](#) * [Mhd::ASL::Create](#) () const [virtual]

Construction of the object returning a pointer to the base class.

Returns

Pointer to the base class

Implements [Mhd::MhdOrientation](#).

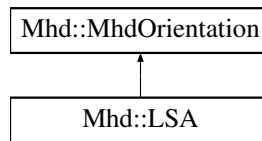
The documentation for this class was generated from the following files:

- lib/include/[MhdOrientationRules.hxx](#)
- lib/src/[MhdOrientationRules.cxx](#)

6.3 Mhd::LSA Class Reference

```
#include <MhdOrientationRules.hxx>
```

Inheritance diagram for Mhd::LSA:



Public Member Functions

- [LSA](#) ()
- [~LSA](#) ()
- void [ConvertToRas](#) (size_t i=1)
Perform orientation to RAS.
- virtual [MhdOrientation](#) * [Create](#) () const
Construction of the object returning a pointer to the base class.

6.3.1 Constructor & Destructor Documentation

6.3.1.1 **Mhd::LSA::LSA** ()

6.3.1.2 **Mhd::LSA::~~LSA** ()

6.3.2 Member Function Documentation

6.3.2.1 void **Mhd::LSA::ConvertToRas** (size_t i=1) [virtual]

Perform orientation to RAS.

Parameters

<i>i</i>	<i>i</i> -th angle of rotation
----------	--------------------------------

Implements [Mhd::MhdOrientation](#).

6.3.2.2 **MhdOrientation** * **Mhd::LSA::Create** () const [virtual]

Construction of the object returning a pointer to the base class.

Returns

Pointer to the base class

Implements [Mhd::MhdOrientation](#).

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

- lib/include/[MhdOrientationRules.hxx](#)
- lib/src/[MhdOrientationRules.cxx](#)

6.4 Mhd::MhdFactory Class Reference

The factory tha collects different [MhdOrientation](#).

```
#include <MhdFactory.hxx>
```

Public Types

- typedef map< string, [MhdBuilder](#) > [Collector](#)
Collector of the orientations.

Public Member Functions

- unique_ptr< [MhdOrientation](#) > [Get](#) (string const &Name) const
Get an object of the factory.
- void [Register](#) (string const &Name, [MhdBuilder](#) const &Func) throw (invalid_ - argument)
Registers in the factory the orientation given.
- vector< string > [Registered](#) () const
List all the orientations contained in the factory.
- void [Unset](#) (string const &Name)
Remove the given orientation from the factory.
- ~[MhdFactory](#) ()
- [MhdFactory](#) ()

Static Public Member Functions

- static [MhdFactory](#) & [Instance](#) ()

6.4.1 Detailed Description

The factory tha collects different [MhdOrientation](#).

6.4.2 Member Typedef Documentation

6.4.2.1 typedef map<string,MhdBuilder> Mhd::MhdFactory::Collector

Collector of the orientations.

6.4.3 Constructor & Destructor Documentation

6.4.3.1 **Mhd::MhdFactory::~~MhdFactory ()**

6.4.3.2 **Mhd::MhdFactory::MhdFactory ()**

6.4.4 Member Function Documentation

6.4.4.1 **unique_ptr< MhdOrientation > Mhd::MhdFactory::Get (string const & *Name*) const**

Get an object of the factory.

Parameters

<i>Name</i>	Name of the orientation selected
-------------	----------------------------------

Returns

The orientation with the Name chosen

Exceptions

<i>invalid_argument</i>	
-------------------------	--

6.4.4.2 **MhdFactory & Mhd::MhdFactory::Instance () [static]**

Returns

6.4.4.3 **void Mhd::MhdFactory::Register (string const & *Name*, MhdBuilder const & *Func*) throw (invalid_argument)**

Registers in the factory the orientation given.

Parameters

<i>Name</i>	Name of the orientation to be registered
<i>Func</i>	The builder used

Exceptions

<i>invalid_argument</i>	
-------------------------	--

6.4.4.4 `vector< string > Mhd::MhdFactory::Registered () const`

List all the orientations contained in the factory.

Returns

a `vector<string>` containing the orientations

6.4.4.5 `void Mhd::MhdFactory::Unset (string const & Name)`

Remove the given orientation from the factory.

Parameters

<i>Name</i>	Name of the orientation to be removed
-------------	---------------------------------------

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

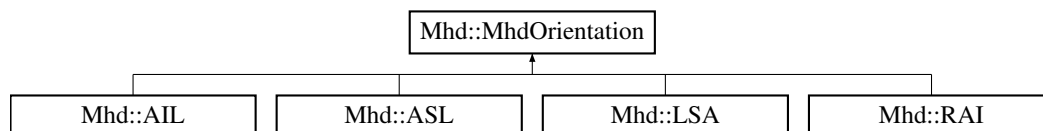
- [lib/include/MhdFactory.hxx](#)
- [lib/src/MhdFactory.cxx](#)

6.5 Mhd::MhdOrientation Class Reference

Base class that contains methods to perform the RAS conversion.

```
#include <MhdOrientation.hxx>
```

Inheritance diagram for Mhd::MhdOrientation:



Public Member Functions

- void [OrientationReader](#) (char *InputFile)
Read from a .mhd files the parameters.
- void [OrientationWriter](#) (char *OutputFile)
Write on a file the parameters in .mhd format.
- virtual void [ConvertToRas](#) (size_t i=1)=0
Virtual declaration of the method used to convert the orientation to RAS orientation.
- virtual [MhdOrientation](#) * [Create](#) () const =0
Virtual constructor.

- void [ComputeAngles](#) ()
Compute the rotation angles from the TransformMatrix.
- void [ComputeRotation](#) (float *angles)
Compute from the angle given the TransformMatrix.
- float [R](#) (size_t i, size_t j)
Returns the element (i,j) of the TransformMatrix.
- float [O](#) (size_t i)
Returns i-th element of Offset.
- float [C](#) (size_t i)
Returns i-th element of CenterOfRotation.
- const char * [AO](#) ()
Returns the stored AnatomicalOrientaion.

Protected Attributes

- vector< float > [TransformMatrix](#)
- vector< float > [Offset](#)
- vector< float > [CenterOfRotation](#)
- string [AnatomicalOrientation](#)
- string [ObjectType](#)
- size_t [NDims](#)
- size_t [CompressedDataSize](#)
- string [BinaryData](#)
- string [BinaryDataByteOrderMSB](#)
- string [CompressedData](#)
- vector< float > [ElementSpacing](#)
- vector< size_t > [DimSize](#)
- string [ElementType](#)
- string [ElementDataFile](#)
- pair< vector< float >, vector< float > > [Angles](#)

Friends

- ostream & [operator<<](#) (ostream &out, const [MhdOrientation](#) &K)
Overloading of the operator <<.

6.5.1 Detailed Description

Base class that contains methods to perform the RAS conversion.

6.5.2 Member Function Documentation

6.5.2.1 `const char* Mhd::MhdOrientation::AO () [inline]`

Returns the stored AnatomicalOrientaion.

Returns

the orientation

6.5.2.2 `float Mhd::MhdOrientation::C (size_t i) [inline]`

Returns i-th element of CenterOfRotation.

Parameters

<i>i</i>	The i-th element of CenterOfRotation
----------	--------------------------------------

Returns

float element of the CenterOfRotation

6.5.2.3 `void Mhd::MhdOrientation::ComputeAngles ()`

Compute the rotation angles from the TransformMatrix.

6.5.2.4 `void Mhd::MhdOrientation::ComputeRotation (float * angles)`

Compute from the angle given the TransformMatrix.

Parameters

<i>angles</i>	The angle used to compute the TransformMatrix
---------------	---

6.5.2.5 `virtual void Mhd::MhdOrientation::ConvertToRas (size_t i = 1) [pure virtual]`

Virtual declaration of the method used to convert the orientation to RAS orientation.

Parameters

<i>i</i>	Select the i-th angle to perform the conversion
----------	---

Implemented in [Mhd::LSA](#), [Mhd::RAI](#), [Mhd::ASL](#), and [Mhd::AIL](#).

6.5.2.6 `virtual MhdOrientation* Mhd::MhdOrientation::Create () const` [pure virtual]

Virtual constructor.

Implemented in [Mhd::LSA](#), [Mhd::RAI](#), [Mhd::ASL](#), and [Mhd::AIL](#).

6.5.2.7 `float Mhd::MhdOrientation::O (size_t i)` [inline]

Returns i-th element of Offset.

Parameters

<i>i</i>	The i-th element of Offset
----------	----------------------------

Returns

float element of the Offset

6.5.2.8 `void Mhd::MhdOrientation::OrientationReader (char * InputFile)`

Read from a .mhd files the parameters.

Parameters

<i>InputFile</i>	Input .mhd file
------------------	-----------------

6.5.2.9 `void Mhd::MhdOrientation::OrientationWriter (char * OutputFile)`

Write on a file the parameters in .mhd format.

Parameters

<i>OutputFile</i>	Output .mhd file
-------------------	------------------

6.5.2.10 `float Mhd::MhdOrientation::R (size_t i, size_t j)` [inline]

Returns the element (i,j) of the TransformMatrix.

Parameters

<i>i</i>	Row index of the TransformMatrix
<i>j</i>	Column index of the TransformMatrix

Returns

float element of the TransformMatrix

6.5.3 Friends And Related Function Documentation**6.5.3.1 ostream& operator<< (ostream & out, const MhdOrientation & K) [friend]**

Overloading of the operator <<.

Parameters

<i>out</i>	Ostream for .mhd file writing
<i>K</i>	The object used to write the .mhd file

Returns

ofstream to write the object

6.5.4 Member Data Documentation**6.5.4.1 string Mhd::MhdOrientation::AnatomicalOrientation [protected]****6.5.4.2 pair< vector<float>, vector<float> > Mhd::MhdOrientation::Angles [protected]****6.5.4.3 string Mhd::MhdOrientation::BinaryData [protected]****6.5.4.4 string Mhd::MhdOrientation::BinaryDataByteOrderMSB [protected]****6.5.4.5 vector<float> Mhd::MhdOrientation::CenterOfRotation [protected]****6.5.4.6 string Mhd::MhdOrientation::CompressedData [protected]****6.5.4.7 size_t Mhd::MhdOrientation::CompressedDataSize [protected]****6.5.4.8 vector<size_t> Mhd::MhdOrientation::DimSize [protected]****6.5.4.9 string Mhd::MhdOrientation::ElementDataFile [protected]****6.5.4.10 vector<float> Mhd::MhdOrientation::ElementSpacing [protected]****6.5.4.11 string Mhd::MhdOrientation::ElementType [protected]****6.5.4.12 size_t Mhd::MhdOrientation::NDims [protected]****6.5.4.13 string Mhd::MhdOrientation::ObjectType [protected]**

6.5.4.14 `vector<float> Mhd::MhdOrientation::Offset` [protected]

6.5.4.15 `vector<float> Mhd::MhdOrientation::TransformMatrix` [protected]

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

- [lib/include/MhdOrientation.hxx](#)
- [lib/src/MhdOrientation.cxx](#)

6.6 Mhd::MhdProxy< T > Class Template Reference

A proxy used to build an object [MhdOrientation](#) and to register it in the factory.

```
#include <MhdProxy.hxx>
```

Public Member Functions

- [MhdProxy](#) (char const *const &Name)
Constructor of the [MhdProxy](#) that perform the registration of the [MhdOrientation](#) in the class.
- [~MhdProxy](#) ()

Static Public Member Functions

- static `unique_ptr< MhdOrientation > Build` ()
The builder of the object.

6.6.1 Detailed Description

```
template<typename T>class Mhd::MhdProxy< T >
```

A proxy used to build an object [MhdOrientation](#) and to register it in the factory.

Template Parameters

<i>T</i>	The string indicating the orientation rule to be build and to be registered
----------	---

6.6.2 Constructor & Destructor Documentation

6.6.2.1 `template<typename T> Mhd::MhdProxy< T>::MhdProxy (char const *const & Name)`

Constructor of the [MhdProxy](#) that perform the registration of the [MhdOrientation](#) in the class.

Parameters

<i>Name</i>	String containing the name of the orientation to be registered
-------------	--

6.6.2.2 `template<typename T> Mhd::MhdProxy< T>::~~MhdProxy () [inline]`

6.6.3 Member Function Documentation

6.6.3.1 `template<typename T> unique_ptr< MhdOrientation > Mhd::MhdProxy< T>::Build () [static]`

The builder of the object.

Returns

A static `unique_ptr<MhdOrientation>`

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

- `lib/include/MhdProxy.hxx`

6.7 Mhd::MhdPythonOrientation Class Reference

The class used for the Python interface.

```
#include <MhdPythonOrientation.hxx>
```

Public Member Functions

- void [OrientationReader](#) (char *InputFile)
Read from a .mhd file the parameters.
- void [OrientationWriter](#) (char *OutputFile)
Write on a file the parameters in .mhd format.
- void [ConvertToRas](#) (size_t i=1)
Method used to convert the orientation to RAS orientation.
- void [ComputeAngles](#) ()
Compute the rotation angles from the TransformMatrix.
- void [ComputeRotation](#) (float *angles)
Compute from the angle given the TransformMatrix.

- float **R** (size_t i, size_t j)
Returns the element (i,j) of the TransformMatrix.
- float **O** (size_t i)
Returns i-th element of Offset.
- float **C** (size_t i)
Returns i-th element of CenterOfRotation.
- const char * **AO** ()
Returns the stored AnatomicalOrientaion.

6.7.1 Detailed Description

The class used for the Python interface.

6.7.2 Member Function Documentation

6.7.2.1 const char* Mhd::MhdPythonOrientation::AO () `[inline]`

Returns the stored AnatomicalOrientaion.

Returns

the orientation

6.7.2.2 float Mhd::MhdPythonOrientation::C (size_t i) `[inline]`

Returns i-th element of CenterOfRotation.

Parameters

<i>i</i>	The i-th element of CenterOfRotation
----------	--------------------------------------

Returns

float element of the CenterOfRotation

6.7.2.3 void Mhd::MhdPythonOrientation::ComputeAngles ()

Compute the rotation angles from the TransformMatrix.

6.7.2.4 void Mhd::MhdPythonOrientation::ComputeRotation (float * angles)

Compute from the angle given the TransformMatrix.

Parameters

<i>angles</i>	The angle used to compute the TransformMatrix
---------------	---

6.7.2.5 void Mhd::MhdPythonOrientation::ConvertToRas (size_t i = 1)

Method used to convert the orientation to RAS orientation.

Parameters

<i>i</i>	Select the i-th to perform the conversion
----------	---

6.7.2.6 float Mhd::MhdPythonOrientation::O (size_t i) [inline]

Returns i-th element of Offset.

Parameters

<i>i</i>	The i-th element of Offset
----------	----------------------------

Returns

float element of the Offset

6.7.2.7 void Mhd::MhdPythonOrientation::OrientationReader (char * InputFile)

Read from a .mhd file the parameters.

Parameters

<i>InputFile</i>	Input .mhd file
------------------	-----------------

6.7.2.8 void Mhd::MhdPythonOrientation::OrientationWriter (char * OutputFile)

Write on a file the parameters in .mhd format.

Parameters

<i>OutputFile</i>	
-------------------	--

6.7.2.9 float Mhd::MhdPythonOrientation::R (size_t i, size_t j) [inline]

Returns the element (i,j) of the TransformMatrix.

Parameters

<i>i</i>	Row index of the TransformMatrix
<i>j</i>	Column index of the TransformMatrix

Returns

float element of the TransformMatrix

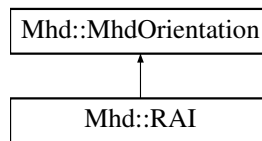
The documentation for this class was generated from the following files:

- lib/include/[MhdPythonOrientation.hxx](#)
- lib/src/[MhdPythonOrientation.cxx](#)

6.8 Mhd::RAI Class Reference

```
#include <MhdOrientationRules.hxx>
```

Inheritance diagram for Mhd::RAI:



Public Member Functions

- [RAI](#) ()
- [~RAI](#) ()
- void [ConvertToRas](#) (size_t i=1)
Perform orientation to RAS.
- virtual [MhdOrientation](#) * [Create](#) () const
Construction of the object returning a pointer to the base class.

6.8.1 Constructor & Destructor Documentation

6.8.1.1 **Mhd::RAI::RAI** ()

6.8.1.2 **Mhd::RAI::~~RAI** ()

6.8.2 Member Function Documentation

6.8.2.1 **void Mhd::RAI::ConvertToRas** (size_t i=1) [virtual]

Perform orientation to RAS.

Parameters

i	i -th angle of rotation
-----	---------------------------

Implements [Mhd::MhdOrientation](#).

6.8.2.2 MhdOrientation * Mhd::RAI::Create () const [virtual]

Construction of the object returning a pointer to the base class.

Returns

Pointer to the base class

Implements [Mhd::MhdOrientation](#).

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

- lib/include/[MhdOrientationRules.hxx](#)
- lib/src/[MhdOrientationRules.cxx](#)

Chapter 7

File Documentation

7.1 lib/include/MHD.hxx File Reference

Header to be included to use the library.

```
#include "MhdOrientation.hxx"    #include "MhdOrientation-  
Rules.hxx"
```

7.1.1 Detailed Description

Header to be included to use the library.

Author

Matteo Manica

Date

2013-09-08

7.2 lib/include/MhdFactory.hxx File Reference

File containing the factory of MhdOrientations.

```
#include "MhdOrientation.hxx"    #include <map>    #include  
<stdexcept>    #include <memory>    #include <algorithm> ×  
#include <iterator>
```

Classes

- class [Mhd::MhdFactory](#)
The factory tha collects different [MhdOrientation](#).

Namespaces

- namespace [Mhd](#)

Namespace [Mhd](#) referred to the classes and methods defined in the project [Mhd-Orientation](#).

Typedefs

- typedef std::unique_ptr < MhdOrientation > (* [Mhd::MhdBuilder](#))()

A typedef to the builder that returns a unique_ptr to a [MhdOrientation](#) object.

7.2.1 Detailed Description

File containing the factory of MhdOrientations.

Author

Matteo Manica

Date

2013-09-08

7.3 lib/include/MhdOrientation.hxx File Reference

File containing the base class MhdOrientation.

```
#include <fstream> #include <iostream> #include <vector> ×  
#include <cstdlib> #include <string> #include <cstring>  
#include <cmath> #include <utility>
```

Classes

- class [Mhd::MhdOrientation](#)

Base class that contains methods to perform the RAS conversion.

Namespaces

- namespace [Mhd](#)

Namespace [Mhd](#) referred to the classes and methods defined in the project [Mhd-Orientation](#).

Defines

- #define [PI](#) 3.14159265

Functions

- char * [Mhd::MhdFileReader](#) (char *InputFile)
Read from a file the AnatomicalOrientation parameter.

7.3.1 Detailed Description

File containing the base class MhdOrientation.

Author

Matteo Manica

Date

2013-09-08

7.3.2 Define Documentation

7.3.2.1 `#define PI 3.14159265`

7.4 lib/include/MhdOrientationRules.hxx File Reference

File containing the derived classes to perform the orientation starting from the string AnatomicalOrientation stored.

```
#include "MhdOrientation.hxx" #include "MhdProxy.hxx"
```

Classes

- class [Mhd::AIL](#)
Derived class to perform AIL->RAS conversion.
- class [Mhd::ASL](#)
- class [Mhd::RAI](#)
- class [Mhd::LSA](#)

Namespaces

- namespace [Mhd](#)
Namespace [Mhd](#) referred to the classes and methods defined in the project [Mhd-Orientation](#).

Defines

- `#define MHDORIENTATIONRULES_HXX 1`

7.4.1 Detailed Description

File containing the derived classes to perform the orientation starting from the string AnatomicalOrientation stored.

Author

Matteo Manica

Date

2013-09-08

7.4.2 Define Documentation

7.4.2.1 `#define MHDORIENTATIONRULES_HXX 1`

7.5 lib/include/MhdProxy.hxx File Reference

File containing a proxy to build the object MhdOrientation and that manage its automatic registration in the factory.

```
#include "MhdFactory.hxx" #include "MhdOrientation.hxx" ×  
#include <typeinfo>
```

Classes

- class [Mhd::MhdProxy< T >](#)
A proxy used to build an object [MhdOrientation](#) and to register it in the factory.

Namespaces

- namespace [Mhd](#)
Namespace [Mhd](#) referred to the classes and methods defined in the project [MhdOrientation](#).

7.5.1 Detailed Description

File containing a proxy to build the object MhdOrientation and that manage its automatic registration in the factory.

Author

Matteo Manica

Date

2013-09-08

7.6 lib/include/MhdPythonOrientation.hxx File Reference

Declaration of the class MhdPythonOrientation used for the Python interface.

```
#include <fstream> #include <iostream> #include <vector> ×  
#include <cstdlib> #include <string> #include <cstring>  
#include <cmath> #include <utility>
```

Classes

- class [Mhd::MhdPythonOrientation](#)
The class used for the Python interface.

Namespaces

- namespace [Mhd](#)
Namespace [Mhd](#) referred to the classes and methods defined in the project [Mhd-Orientation](#).

Defines

- #define [PI](#) 3.14159265

7.6.1 Detailed Description

Declaration of the class MhdPythonOrientation used for the Python interface.

Author

Matteo Manica

Date

2013-09-08

7.6.2 Define Documentation

7.6.2.1 #define [PI](#) 3.14159265

7.7 lib/pymodule/mhd.py File Reference

Module for the interface with Python using ctypes.

Classes

- class [mhd.MhdOrientation](#)
Class [MhdOrientation](#) imported in Python.

Namespaces

- namespace [mhd](#)

Variables

- tuple [mhd.lib](#) = ctypes.CDLL('./libMhdOrientation.so',mode=ctypes.RTLD_GLOBAL)

7.7.1 Detailed Description

Module for the interface with Python using ctypes.

Author

Matteo Manica

Date

2013-09-08

7.8 lib/src/MhdFactory.cxx File Reference

Implementation of the factory.

```
#include "MhdFactory.hxx"
```

Namespaces

- namespace [Mhd](#)
Namespace [Mhd](#) referred to the classes and methods defined in the project [MhdOrientation](#).

7.8.1 Detailed Description

Implementation of the factory.

Author

Matteo Manica

Date

2013-09-08

7.9 lib/src/MhdFileReader.cxx File Reference

File containing the function that reads a .mhd file to get the AnatomicalOrientation parameter.

```
#include "MhdOrientation.hxx"
```

Namespaces

- namespace [Mhd](#)

Namespace [Mhd](#) referred to the classes and methods defined in the project [Mhd-Orientation](#).

Functions

- char * [Mhd::MhdFileReader](#) (char *InputFile)

Read from a file the AnatomicalOrientation parameter.

7.9.1 Detailed Description

File containing the function that reads a .mhd file to get the AnatomicalOrientation parameter.

Author

Matteo Manica

Date

2013-09-08

7.10 lib/src/MhdOrientation.cxx File Reference

Implementation of the base class MhdOrientation.

```
#include "MhdOrientation.hxx"
```

Namespaces

- namespace [Mhd](#)

Namespace [Mhd](#) referred to the classes and methods defined in the project [Mhd-Orientation](#).

Functions

- ostream & [Mhd::operator<<](#) (ostream &out, const MhdOrientation &K)

7.10.1 Detailed Description

Implementation of the base class MhdOrientation.

Author

Matteo Manica

Date

2013-09-08

7.11 lib/src/MhdOrientationRules.cxx File Reference

Implementation of the class derived from MhdOrientation.

```
#include "MhdOrientationRules.hxx"
```

Namespaces

- namespace [Mhd](#)

Namespace [Mhd](#) referred to the classes and methods defined in the project [Mhd-Orientation](#).

7.11.1 Detailed Description

Implementation of the class derived from MhdOrientation.

Author

Matteo Manica

Date

2013-09-08

7.12 lib/src/MhdPythonOrientation.cxx File Reference

Implementation of MhdPythonOrientation used for the Python interface.

```
#include "MhdPythonOrientation.hxx"
```

Namespaces

- namespace [Mhd](#)
Namespace [Mhd](#) referred to the classes and methods defined in the project [Mhd-Orientation](#).

7.12.1 Detailed Description

Implementation of MhdPythonOrientation used for the Python interface.

Author

Matteo Manica

Date

2013-09-08

7.13 lib/src/MhdPythonWrapper.cxx File Reference

Implementation of the wrapping in Python.

```
#include "MhdPythonOrientation.hxx"
```

Functions

- [Mhd::MhdPythonOrientation * MhdOrientation_Python \(\)](#)
Declaration of the functions that will be exported in Python using ctypes.
- void [MhdOrientation_OrientationReader](#) ([Mhd::MhdPythonOrientation](#) *mo, char *InputFile)
MhdPythonOrientation::OrientationReader in Python.
- void [MhdOrientation_OrientationWriter](#) ([Mhd::MhdPythonOrientation](#) *mo, char *OutputFile)
MhdPythonOrientation::OrientatioWriter in Python.
- void [MhdOrientation_ConvertToRas](#) ([Mhd::MhdPythonOrientation](#) *mo, size_t i)
MhdPythonOrientation::ConvertToRas in Python.
- void [MhdOrientation_ComputeAngles](#) ([Mhd::MhdPythonOrientation](#) *mo)
MhdPythonOrientation::ComputeAngles in Python.

- void [MhdOrientation_ComputeRotation](#) ([Mhd::MhdPythonOrientation](#) *mo, float *angles)
MhdPythonOrientation::ComputeRotation in Python.
- void [MhdOrientation_R](#) ([Mhd::MhdPythonOrientation](#) *mo, size_t i, size_t j)
MhdPythonOrientation::R in Python.
- void [MhdOrientation_O](#) ([Mhd::MhdPythonOrientation](#) *mo, size_t i)
MhdPythonOrientation::O in Python.
- void [MhdOrientation_C](#) ([Mhd::MhdPythonOrientation](#) *mo, size_t i)
MhdPythonOrientation::C in Python.
- void [MhdOrientation_AO](#) ([Mhd::MhdPythonOrientation](#) *mo)
MhdPythonOrientation::AO in Python.

7.13.1 Detailed Description

Implementation of the wrapping in Python.

Author

Matteo Manica

Date

2013-09-08

7.13.2 Function Documentation

7.13.2.1 void [MhdOrientation_AO](#) ([Mhd::MhdPythonOrientation](#) * mo)

[MhdPythonOrientation::AO](#) in Python.

Parameters

<i>mo</i>	Object MhdPythonOrientation
-----------	---

7.13.2.2 void [MhdOrientation_C](#) ([Mhd::MhdPythonOrientation](#) * mo, size_t i)

[MhdPythonOrientation::C](#) in Python.

Parameters

<i>mo</i>	Object MhdPythonOrientation
<i>i</i>	i-th element of CenterOfRotation

7.13.2.3 `void MhdOrientation_ComputeAngles (Mhd::MhdPythonOrientation * mo)`

MhdPythonOrientation::ComputeAngles in Python.

Parameters

<i>mo</i>	Object MhdPythonOrientation
-----------	-----------------------------

7.13.2.4 `void MhdOrientation_ComputeRotation (Mhd::MhdPythonOrientation * mo, float * angles)`

MhdPythonOrientation::ComputeRotation in Python.

Parameters

<i>mo</i>	Object MhdPythonOrientation
<i>angles</i>	Angle used to compute the rotation

7.13.2.5 `void MhdOrientation_ConvertToRas (Mhd::MhdPythonOrientation * mo, size_t i)`

MhdPythonOrientation::ConvertToRas in Python.

Parameters

<i>mo</i>	Object MhdPythonOrientation
<i>i</i>	Angle selected for the conversion

7.13.2.6 `void MhdOrientation_O (Mhd::MhdPythonOrientation * mo, size_t i)`

MhdPythonOrientation::O in Python.

Parameters

<i>mo</i>	Object MhdPythonOrientation
<i>i</i>	i-th element of Offset

7.13.2.7 `void MhdOrientation_OrientationReader (Mhd::MhdPythonOrientation * mo, char * InputFile)`

MhdPythonOrientation::OrientationReader in Python.

Parameters

<i>mo</i>	Object MhdPythonOrientation
<i>InputFile</i>	Input .mhd file

**7.13.2.8 void MhdOrientation_OrientationWriter (Mhd::MhdPythonOrientation *
mo, char * *OutputFile*)**

MhdPythonOrientation::OrientatioWriter in Python.

Parameters

<i>mo</i>	Object MhdPythonOrientation
<i>OutputFile</i>	Output in .mhd format

7.13.2.9 Mhd::MhdPythonOrientation* MhdOrientation_Python ()

Declaration of the functions that will be exported in Python using ctypes.

Constructor of MhdPythonOrientation

Returns

Pointer to MhdPythonOrientation

**7.13.2.10 void MhdOrientation_R (Mhd::MhdPythonOrientation * *mo*, size_t *i*, size_t *j*
)**

MhdPythonOrientation::R in Python.

Parameters

<i>mo</i>	Object MhdPythonOrientation
<i>i</i>	i-th row of the TransformMatrix
<i>j</i>	j-th column of the TransformMatrix