

## MhdOrientation

Generated by Doxygen 1.7.6.1

Mon Sep 9 2013 21:40:30



# Contents

<b>1</b>	<b>MhdOrientation</b>	<b>1</b>
<b>2</b>	<b>Namespace Index</b>	<b>3</b>
2.1	Namespace List . . . . .	3
<b>3</b>	<b>Class Index</b>	<b>5</b>
3.1	Class Hierarchy . . . . .	5
<b>4</b>	<b>Class Index</b>	<b>7</b>
4.1	Class List . . . . .	7
<b>5</b>	<b>File Index</b>	<b>9</b>
5.1	File List . . . . .	9
<b>6</b>	<b>Namespace Documentation</b>	<b>11</b>
6.1	Mhd Namespace Reference . . . . .	11
6.1.1	Detailed Description . . . . .	12
6.1.2	Typedef Documentation . . . . .	12
6.1.2.1	MhdBuilder . . . . .	12
6.1.3	Function Documentation . . . . .	12
6.1.3.1	MhdFileReader . . . . .	12
6.1.3.2	operator<< . . . . .	12
<b>7</b>	<b>Class Documentation</b>	<b>13</b>
7.1	Mhd::AIL Class Reference . . . . .	13
7.1.1	Detailed Description . . . . .	13
7.1.2	Constructor & Destructor Documentation . . . . .	13
7.1.2.1	AIL . . . . .	14

7.1.2.2	<a href="#">~AIL</a>	14
7.1.3	<a href="#">Member Function Documentation</a>	14
7.1.3.1	<a href="#">ConvertToRas</a>	14
7.1.3.2	<a href="#">Create</a>	14
7.2	<a href="#">Mhd::ASL Class Reference</a>	14
7.2.1	<a href="#">Constructor &amp; Destructor Documentation</a>	15
7.2.1.1	<a href="#">ASL</a>	15
7.2.1.2	<a href="#">~ASL</a>	15
7.2.2	<a href="#">Member Function Documentation</a>	15
7.2.2.1	<a href="#">ConvertToRas</a>	15
7.2.2.2	<a href="#">Create</a>	15
7.3	<a href="#">Mhd::LAS Class Reference</a>	15
7.3.1	<a href="#">Constructor &amp; Destructor Documentation</a>	16
7.3.1.1	<a href="#">LAS</a>	16
7.3.1.2	<a href="#">~LAS</a>	16
7.3.2	<a href="#">Member Function Documentation</a>	16
7.3.2.1	<a href="#">ConvertToRas</a>	16
7.3.2.2	<a href="#">Create</a>	16
7.4	<a href="#">Mhd::MhdFactory Class Reference</a>	17
7.4.1	<a href="#">Detailed Description</a>	17
7.4.2	<a href="#">Member Typedef Documentation</a>	17
7.4.2.1	<a href="#">Collector</a>	17
7.4.3	<a href="#">Constructor &amp; Destructor Documentation</a>	17
7.4.3.1	<a href="#">~MhdFactory</a>	17
7.4.4	<a href="#">Member Function Documentation</a>	18
7.4.4.1	<a href="#">Get</a>	18
7.4.4.2	<a href="#">Instance</a>	18
7.4.4.3	<a href="#">Register</a>	18
7.4.4.4	<a href="#">Registered</a>	18
7.4.4.5	<a href="#">Unset</a>	19
7.5	<a href="#">Mhd::MhdOrientation Class Reference</a>	19
7.5.1	<a href="#">Detailed Description</a>	20
7.5.2	<a href="#">Member Function Documentation</a>	20
7.5.2.1	<a href="#">AO</a>	20

7.5.2.2	C	21
7.5.2.3	ComputeAngles	21
7.5.2.4	ComputeRotation	21
7.5.2.5	ConvertToRas	21
7.5.2.6	Create	21
7.5.2.7	O	21
7.5.2.8	OrientationReader	22
7.5.2.9	OrientationWriter	22
7.5.2.10	R	22
7.5.3	Friends And Related Function Documentation	22
7.5.3.1	operator<<	22
7.5.4	Member Data Documentation	23
7.5.4.1	AnatomicalOrientation	23
7.5.4.2	Angles	23
7.5.4.3	BinaryData	23
7.5.4.4	BinaryDataByteOrderMSB	23
7.5.4.5	CenterOfRotation	23
7.5.4.6	CompressedData	23
7.5.4.7	CompressedDataSize	23
7.5.4.8	DimSize	23
7.5.4.9	ElementDataFile	23
7.5.4.10	ElementSpacing	23
7.5.4.11	ElementType	23
7.5.4.12	NDims	23
7.5.4.13	ObjectType	23
7.5.4.14	Offset	23
7.5.4.15	TransformMatrix	23
7.6	Mhd::MhdProxy< T > Class Template Reference	23
7.6.1	Detailed Description	24
7.6.2	Constructor & Destructor Documentation	24
7.6.2.1	MhdProxy	24
7.6.2.2	~MhdProxy	24
7.6.3	Member Function Documentation	24
7.6.3.1	Build	24

7.7	Mhd::MhdPythonOrientation Class Reference	25
7.7.1	Detailed Description	25
7.7.2	Member Function Documentation	25
7.7.2.1	AO	25
7.7.2.2	C	26
7.7.2.3	ComputeAngles	26
7.7.2.4	ComputeRotation	26
7.7.2.5	ConvertToRas	26
7.7.2.6	O	26
7.7.2.7	OrientationReader	27
7.7.2.8	OrientationWriter	27
7.7.2.9	R	27
7.8	Mhd::RAI Class Reference	27
7.8.1	Constructor & Destructor Documentation	28
7.8.1.1	RAI	28
7.8.1.2	~RAI	28
7.8.2	Member Function Documentation	28
7.8.2.1	ConvertToRas	28
7.8.2.2	Create	28
<b>8</b>	<b>File Documentation</b>	<b>29</b>
8.1	lib/include/MHD.hxx File Reference	29
8.1.1	Detailed Description	29
8.2	lib/include/MhdFactory.hxx File Reference	29
8.2.1	Detailed Description	30
8.3	lib/include/MhdOrientation.hxx File Reference	30
8.3.1	Detailed Description	31
8.3.2	Define Documentation	31
8.3.2.1	PI	31
8.4	lib/include/MhdOrientationRules.hxx File Reference	31
8.4.1	Detailed Description	32
8.4.2	Define Documentation	32
8.4.2.1	MHDIORIENTATIONRULES_HXX	32
8.5	lib/include/MhdProxy.hxx File Reference	32

8.5.1	Detailed Description	32
8.6	lib/include/MhdPythonOrientation.hxx File Reference	33
8.6.1	Detailed Description	33
8.6.2	Define Documentation	33
8.6.2.1	PI	33
8.7	lib/pymodule/mhd.py File Reference	33
8.7.1	Detailed Description	34
8.8	lib/src/MhdFactory.cxx File Reference	34
8.8.1	Detailed Description	34
8.9	lib/src/MhdFileReader.cxx File Reference	35
8.9.1	Detailed Description	35
8.10	lib/src/MhdOrientation.cxx File Reference	35
8.10.1	Detailed Description	36
8.11	lib/src/MhdOrientationRules.cxx File Reference	36
8.11.1	Detailed Description	36
8.12	lib/src/MhdPythonOrientation.cxx File Reference	37
8.12.1	Detailed Description	37
8.13	lib/src/MhdPythonWrapper.cxx File Reference	37
8.13.1	Detailed Description	38
8.13.2	Function Documentation	38
8.13.2.1	MhdOrientation_AO	38
8.13.2.2	MhdOrientation_C	38
8.13.2.3	MhdOrientation_ComputeAngles	39
8.13.2.4	MhdOrientation_ComputeRotation	39
8.13.2.5	MhdOrientation_ConvertToRas	39
8.13.2.6	MhdOrientation_O	39
8.13.2.7	MhdOrientation_OrientationReader	39
8.13.2.8	MhdOrientation_OrientationWriter	40
8.13.2.9	MhdOrientation_Python	40
8.13.2.10	MhdOrientation_R	40
8.14	README.md File Reference	40





## Chapter 1

# MhdOrientation

A library used to orient images that works on .mhd header files to convert the anatomical orientation present in the header, into a RAS one. The operation is performed to allow the elaboration of the image using software like VTK and vmtk without losing information about its position when working in the physical space



## Chapter 2

# Namespace Index

### 2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

[Mhd](#)

Namespace [Mhd](#) referred to the classes and methods defined in the  
project [MhdOrientation](#) . . . . . 11



## Chapter 3

# Class Index

### 3.1 Class Hierarchy

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

Mhd::MhdFactory . . . . .	17
Mhd::MhdOrientation . . . . .	19
Mhd::AIL . . . . .	13
Mhd::ASL . . . . .	14
Mhd::LAS . . . . .	15
Mhd::RAI . . . . .	27
mhd.MhdOrientation . . . . .	??
Mhd::MhdProxy< T > . . . . .	23
Mhd::MhdPythonOrientation . . . . .	25



## Chapter 4

# Class Index

### 4.1 Class List

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

<a href="#">Mhd::AIL</a>	
Derived class to perform AIL->RAS conversion . . . . .	13
<a href="#">Mhd::ASL</a> . . . . .	14
<a href="#">Mhd::LAS</a> . . . . .	15
<a href="#">Mhd::MhdFactory</a>	
The factory tha collects different <a href="#">MhdOrientation</a> . . . . .	17
<a href="#">Mhd::MhdOrientation</a>	
Base class that contains methods to perform the RAS conversion . .	19
<a href="#">mhd.MhdOrientation</a>	
Class <a href="#">MhdOrientation</a> imported in Python . . . . .	??
<a href="#">Mhd::MhdProxy&lt; T &gt;</a>	
A proxy used to build an object <a href="#">MhdOrientation</a> and to register it in the factory . . . . .	23
<a href="#">Mhd::MhdPythonOrientation</a>	
The class used for the Python interface . . . . .	25
<a href="#">Mhd::RAI</a> . . . . .	27





## Chapter 5

# File Index

### 5.1 File List

Here is a list of all files with brief descriptions:

<a href="#">README.md</a>	40
<a href="#">lib/include/MHD.hxx</a>	
Header to be included to use the library	29
<a href="#">lib/include/MhdFactory.hxx</a>	
File containing the factory of MhdOrientations	29
<a href="#">lib/include/MhdOrientation.hxx</a>	
File containing the base class MhdOrientation	30
<a href="#">lib/include/MhdOrientationRules.hxx</a>	
File containing the derived classes to perform the orientation starting from the string AnatomicalOrientation stored	31
<a href="#">lib/include/MhdProxy.hxx</a>	
File containing a proxy to build the object MhdOrientation and that manage its automatic registration in the factory	32
<a href="#">lib/include/MhdPythonOrientation.hxx</a>	
Declaration of the class MhdPythonOrientation used for the Python interface	33
<a href="#">lib/pymodule/mhd.py</a>	
Module for the interface with Python using ctypes	33
<a href="#">lib/src/MhdFactory.cxx</a>	
Implementation of the factory	34
<a href="#">lib/src/MhdFileReader.cxx</a>	
File containing the function that reads a .mhd file to get the AnatomicalOrientation parameter	35
<a href="#">lib/src/MhdOrientation.cxx</a>	
Implementation of the base class MhdOrientation	35
<a href="#">lib/src/MhdOrientationRules.cxx</a>	
Implementation of the class derived from MhdOrientation	36

<a href="#">lib/src/MhdPythonOrientation.cxx</a>	
Implementation of MhdPythonOrientation used for the Python inter-	
face . . . . .	37
<a href="#">lib/src/MhdPythonWrapper.cxx</a>	
Implementation of the wrapping in Python . . . . .	37

## Chapter 6

# Namespace Documentation

### 6.1 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 [LAS](#)
- 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)

### 6.1.1 Detailed Description

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

### 6.1.2 Typedef Documentation

6.1.2.1 typedef std::unique\_ptr<[MhdOrientation](#) >(\* [Mhd::MhdBuilder](#))()

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

### 6.1.3 Function Documentation

6.1.3.1 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

6.1.3.2 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 7

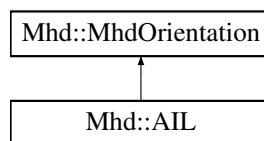
# Class Documentation

### 7.1 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.*

#### 7.1.1 Detailed Description

Derived class to perform AIL->RAS conversion.

#### 7.1.2 Constructor & Destructor Documentation

7.1.2.1 **Mhd::AIL::AIL ( )**

7.1.2.2 **Mhd::AIL::~~AIL ( )**

### 7.1.3 Member Function Documentation

7.1.3.1 **void Mhd::AIL::ConvertToRas ( size\_t  $i=1$  )** [virtual]

Perform orientation to RAS.

#### Parameters

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

Implements [Mhd::MhdOrientation](#).

7.1.3.2 **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](#).

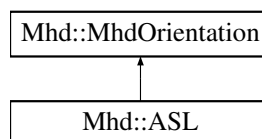
The documentation for this class was generated from the following files:

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

## 7.2 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.*

## 7.2.1 Constructor & Destructor Documentation

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

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

## 7.2.2 Member Function Documentation

7.2.2.1 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](#).

7.2.2.2 [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](#).

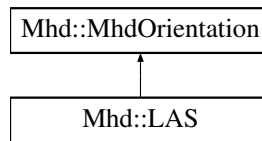
The documentation for this class was generated from the following files:

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

## 7.3 Mhd::LAS Class Reference

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

Inheritance diagram for Mhd::LAS:



## Public Member Functions

- [LAS](#) ()
- [~LAS](#) ()
- 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.*

## 7.3.1 Constructor & Destructor Documentation

7.3.1.1 **Mhd::LAS::LAS ( )**

7.3.1.2 **Mhd::LAS::~~LAS ( )**

## 7.3.2 Member Function Documentation

7.3.2.1 **void Mhd::LAS::ConvertToRas ( size\_t i=1 )** [virtual]

Perform orientation to RAS.

### Parameters

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

Implements [Mhd::MhdOrientation](#).

7.3.2.2 **MhdOrientation \* Mhd::LAS::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](#)



## 7.4 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](#) ()

### Static Public Member Functions

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

#### 7.4.1 Detailed Description

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

#### 7.4.2 Member Typedef Documentation

7.4.2.1 typedef map<string,MhdBuilder> **Mhd::MhdFactory::Collector**

Collector of the orientations.

#### 7.4.3 Constructor & Destructor Documentation

7.4.3.1 **Mhd::MhdFactory::~MhdFactory** ( )

### 7.4.4 Member Function Documentation

7.4.4.1 `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>	
-------------------------	--

7.4.4.2 `MhdFactory & Mhd::MhdFactory::Instance ( ) [static]`

#### Returns

7.4.4.3 `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>	
-------------------------	--

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

List all the orientations contained in the factory.

**Returns**

a vector<string> containing the orientations

**7.4.4.5 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:

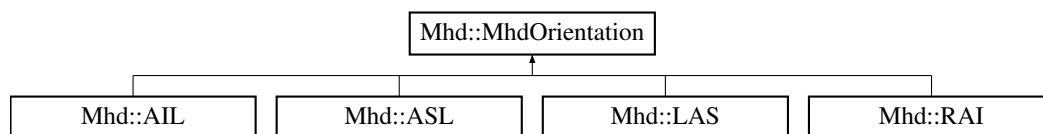
- lib/include/[MhdFactory.hxx](#)
- lib/src/[MhdFactory.cxx](#)

**7.5 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 <<.*

## 7.5.1 Detailed Description

Base class that contains methods to perform the RAS conversion.

## 7.5.2 Member Function Documentation

### 7.5.2.1 const char\* [Mhd::MhdOrientation::AO](#) ( ) [\[inline\]](#)

Returns the stored AnatomicalOrientaion.

#### Returns

the orientation

**7.5.2.2** `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

**7.5.2.3** `void Mhd::MhdOrientation::ComputeAngles ( )`

Compute the rotation angles from the TransformMatrix.

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

Compute from the angle given the TransformMatrix.

**Parameters**

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

**7.5.2.5** `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::LAS](#), [Mhd::RAI](#), [Mhd::ASL](#), and [Mhd::AIL](#).

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

Virtual constructor.

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

**7.5.2.7** `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

### 7.5.2.8 void Mhd::MhdOrientation::OrientationReader ( char \* *InputFile* )

Read from a .mhd files the parameters.

## Parameters

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

### 7.5.2.9 void Mhd::MhdOrientation::OrientationWriter ( char \* *OutputFile* )

Write on a file the parameters in .mhd format.

## Parameters

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

### 7.5.2.10 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

## 7.5.3 Friends And Related Function Documentation

### 7.5.3.1 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

## 7.5.4 Member Data Documentation

7.5.4.1 `string Mhd::MhdOrientation::AnatomicalOrientation` [protected]

7.5.4.2 `pair< vector<float>, vector<float> > Mhd::MhdOrientation::Angles`  
[protected]

7.5.4.3 `string Mhd::MhdOrientation::BinaryData` [protected]

7.5.4.4 `string Mhd::MhdOrientation::BinaryDataByteOrderMSB` [protected]

7.5.4.5 `vector<float> Mhd::MhdOrientation::CenterOfRotation` [protected]

7.5.4.6 `string Mhd::MhdOrientation::CompressedData` [protected]

7.5.4.7 `size_t Mhd::MhdOrientation::CompressedDataSize` [protected]

7.5.4.8 `vector<size_t> Mhd::MhdOrientation::DimSize` [protected]

7.5.4.9 `string Mhd::MhdOrientation::ElementDataFile` [protected]

7.5.4.10 `vector<float> Mhd::MhdOrientation::ElementSpacing` [protected]

7.5.4.11 `string Mhd::MhdOrientation::ElementType` [protected]

7.5.4.12 `size_t Mhd::MhdOrientation::NDims` [protected]

7.5.4.13 `string Mhd::MhdOrientation::ObjectType` [protected]

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

7.5.4.15 `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](#)

## 7.6 Mhd::MhdProxy&lt; T &gt; 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.*

### 7.6.1 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 thr orientation rule to be build and to be registered
----------	---

### 7.6.2 Constructor & Destructor Documentation

7.6.2.1 `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
-------------	--

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

### 7.6.3 Member Function Documentation

7.6.3.1 `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

**7.7 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.*

**7.7.1 Detailed Description**

The class used for the Python interface.

**7.7.2 Member Function Documentation**

**7.7.2.1** const char\* [Mhd::MhdPythonOrientation::AO](#) ( ) `[inline]`

Returns the stored AnatomicalOrientaion.

**Returns**

the orientation

**7.7.2.2 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

**7.7.2.3 void Mhd::MhdPythonOrientation::ComputeAngles ( )**

Compute the rotation angles from the TransformMatrix.

**7.7.2.4 void Mhd::MhdPythonOrientation::ComputeRotation ( float \* *angles* )**

Compute from the angle given the TransformMatrix.

**Parameters**

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

**7.7.2.5 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
----------	---

**7.7.2.6 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

**7.7.2.7 void Mhd::MhdPythonOrientation::OrientationReader ( char \* *InputFile* )**

Read from a .mhd file the parameters.

**Parameters**

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

**7.7.2.8 void Mhd::MhdPythonOrientation::OrientationWriter ( char \* *OutputFile* )**

Write on a file the parameters in .mhd format.

**Parameters**

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

**7.7.2.9 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

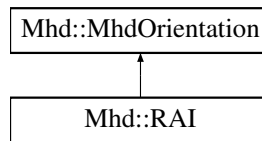
The documentation for this class was generated from the following files:

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

**7.8 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.*

## 7.8.1 Constructor & Destructor Documentation

7.8.1.1 `Mhd::RAI::RAI ( )`

7.8.1.2 `Mhd::RAI::~~RAI ( )`

## 7.8.2 Member Function Documentation

7.8.2.1 `void Mhd::RAI::ConvertToRas ( size_t i = 1 )` [virtual]

Perform orientation to RAS.

### Parameters

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

Implements [Mhd::MhdOrientation](#).

7.8.2.2 `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 8

# File Documentation

### 8.1 lib/include/MHD.hxx File Reference

Header to be included to use the library.

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

#### 8.1.1 Detailed Description

Header to be included to use the library.

**Author**

Matteo Manica

**Date**

2013-09-08

### 8.2 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.*

### 8.2.1 Detailed Description

File containing the factory of MhdOrientations.

#### Author

Matteo Manica

#### Date

2013-09-08

## 8.3 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.*

### 8.3.1 Detailed Description

File containing the base class MhdOrientation.

#### Author

Matteo Manica

#### Date

2013-09-08

### 8.3.2 Define Documentation

8.3.2.1 `#define PI 3.14159265`

## 8.4 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::LAS](#)

## Namespaces

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

## Defines

- `#define MHDORIENTATIONRULES_HXX 1`

### 8.4.1 Detailed Description

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

**Author**

Matteo Manica

**Date**

2013-09-08

### 8.4.2 Define Documentation

8.4.2.1 `#define MHDORIENTATIONRULES_HXX 1`

## 8.5 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](#).*

### 8.5.1 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



## 8.6 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

#### 8.6.1 Detailed Description

Declaration of the class MhdPythonOrientation used for the Python interface.

#### Author

Matteo Manica

#### Date

2013-09-08

#### 8.6.2 Define Documentation

##### 8.6.2.1 #define [PI](#) 3.14159265

## 8.7 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)

### 8.7.1 Detailed Description

Module for the interface with Python using ctypes.

#### Author

Matteo Manica

#### Date

2013-09-08

## 8.8 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](#).*

### 8.8.1 Detailed Description

Implementation of the factory.

**Author**

Matteo Manica

**Date**

2013-09-08

## 8.9 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 [MhdOrientation](#).*

**Functions**

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

*Read from a file the AnatomicalOrientation parameter.*

### 8.9.1 Detailed Description

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

**Author**

Matteo Manica

**Date**

2013-09-08

## 8.10 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)

### 8.10.1 Detailed Description

Implementation of the base class MhdOrientation.

#### Author

Matteo Manica

#### Date

2013-09-08

## 8.11 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](#).*

### 8.11.1 Detailed Description

Implementation of the class derived from MhdOrientation.

#### Author

Matteo Manica

#### Date

2013-09-08

## 8.12 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](#).*

### 8.12.1 Detailed Description

Implementation of MhdPythonOrientation used for the Python interface.

#### Author

Matteo Manica

#### Date

2013-09-08

## 8.13 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.*

### 8.13.1 Detailed Description

Implementation of the wrapping in Python.

#### Author

Matteo Manica

#### Date

2013-09-08

### 8.13.2 Function Documentation

#### 8.13.2.1 void [MhdOrientation\\_AO](#) ( [Mhd::MhdPythonOrientation](#) \* mo )

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

##### Parameters

<i>mo</i>	Object <a href="#">MhdPythonOrientation</a>
-----------	---

#### 8.13.2.2 void [MhdOrientation\\_C](#) ( [Mhd::MhdPythonOrientation](#) \* mo, size\_t i )

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

##### Parameters

<i>mo</i>	Object <a href="#">MhdPythonOrientation</a>
<i>i</i>	i-th element of <a href="#">CenterOfRotation</a>

**8.13.2.3** void MhdOrientation\_ComputeAngles ( Mhd::MhdPythonOrientation \* *mo* )

MhdPythonOrientation::ComputeAngles in Python.

#### Parameters

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

**8.13.2.4** 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

**8.13.2.5** 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

**8.13.2.6** 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

**8.13.2.7** void MhdOrientation\_OrientationReader ( Mhd::MhdPythonOrientation \* *mo*, char \* *InputFile* )

MhdPythonOrientation::OrientationReader in Python.

## Parameters

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

**8.13.2.8** 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

**8.13.2.9** Mhd::MhdPythonOrientation\* **MhdOrientation\_Python** ( )

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

Constructor of MhdPythonOrientation

## Returns

Pointer to MhdPythonOrientation

**8.13.2.10** 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

## 8.14 README.md File Reference