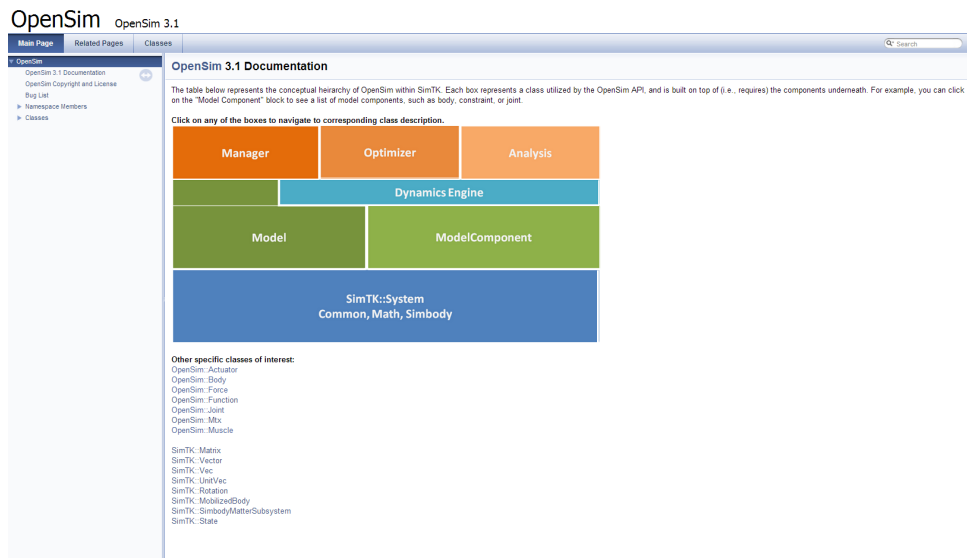


Guide to Using Doxygen

[Doxygen](#) is an automated documentation system for available Classes and methods. This page will introduce you to navigating the online Doxygen pages and orientate you to the structure and language used.

- [Doxygen Navigation](#)
 - [Doxygen Homepage](#)
 - [Class Lists](#)
- [Parts of the Doxygen Page](#)
 - [Class Hierarchy](#)
 - [Constructor Method](#)
 - [Member Functions](#)

Doxygen Navigation

 <p>The screenshot shows the OpenSim 3.1 Doxygen homepage. At the top, there is a navigation bar with 'Main Page', 'Related Pages', and 'Classes'. A search box is located in the top right corner. The main content area is titled 'OpenSim 3.1 Documentation' and contains a conceptual hierarchy diagram. The diagram consists of several colored boxes: 'Manager' (orange), 'Optimizer' (orange), 'Analysis' (orange), 'Dynamics Engine' (blue), 'Model' (green), 'ModelComponent' (green), and 'SimTK::System' (blue) which includes 'Common, Math, Simbody'. Below the diagram, there is a list of 'Other specific classes of interest' including OpenSim::Actuator, OpenSim::Body, OpenSim::Force, OpenSim::Function, OpenSim::Joint, OpenSim::Mtx, OpenSim::Muscle, SimTK::Matrix, SimTK::Vector, SimTK::Vec, SimTK::UnitVec, SimTK::Rotation, SimTK::MobilizeBody, SimTK::SimbodyMatterSubsystem, and SimTK::State.</p>	<h3>Doxygen Homepage</h3> <p>The doxygen homepage gives a top level view of the class groupings and the most popular OpenSim and SimTK classes. From the homepage you can navigate down the Class hierarchy by clicking on blocks that correspond to the Classes that interest you.</p> <p>The search box, located in the top right hand corner, searches through the matching Class names. This is useful if you know the class name but can be limiting if you are trying to discover Classes or methods. If you are trying to discover a method to use we suggest using the OpenSim omniseach box, which searches through all the OpenSim confluence, forum and doxygen content.</p>
---	---

OpenSim 3.1

OpenSim 3.1 Documentation

The table below represents the conceptual hierarchy of OpenSim within SimTK. Each box represents a class utilized by the OpenSim API, and is built on top of (i.e., requires) the components underneath. For example, you can click on the "Model Component" block to see a list of model components, such as body, constraint, or joint.

Click on any of the boxes to navigate to corresponding class description.

Other specific classes of interest:

- OpenSim: Actuator
- OpenSim: Body
- OpenSim: Force
- OpenSim: Function
- OpenSim: Joint
- OpenSim: Mtx
- OpenSim: Muscle
- SimTK: Matrix
- SimTK: Vector
- SimTK: Vec
- SimTK: UnitVec
- SimTK: Rotation
- SimTK: MobilizedBody
- SimTK: SimbodyMatterSubsystem
- SimTK: State

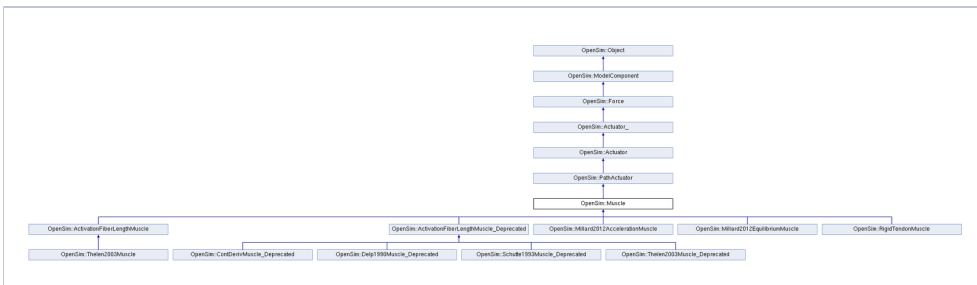
Generated on Tue Jul 15 2013 14:10:11 for OpenSim by doxygen 1.8.1.2

Class Lists

The left hand navigation panel is used to move through Class lists, Class Hierarchy and Class Members. The flat and hierarchical Class lists show all of the API classes and give you the opportunity to discover new Classes.

Parts of the Doxygen Page

Once you navigate to a Class of interest, you will find several sections.



Class Hierarchy

Class Hierarchy diagrams display the inheritance of methods from abstract to concrete Classes. For example, a Muscle() is a PathActuator() and inherits all the functionality of a PathActuator() while the Thelen2003Muscle() is a Muscle() and inherits all the functionality of the Muscle() Class.

Constructor & Destructor Documentation

OpenSim::Thelen2003Muscle::Thelen2003Muscle ()

```

OpenSim::Thelen2003Muscle::Thelen2003Muscle ( const std::string & aName,
double aMaxIsometricForce,
double aOptimalFiberLength,
double aTendonSlackLength,
double aPennationAngle
)
  
```

Constructor Method

Methods and inputs to build an instance of a class. When given no inputs, indicated by empty brackets, a default instance of the class is created.

Public Member Functions

```
Thelen2003Muscle ()
Thelen2003Muscle (const std::string &aName, double aMaxIsometricForce, double aOptimalFiberLength, double aTendonStackLength, double aPennationAngle)
double getActivationTimeConstant () const
double getMinimumActivation () const
double getDeactivationTimeConstant () const
double getFmaxTendonStrain () const
double getFmaxMuscleStrain () const
double getKshapeActive () const
double getKshapePassive () const
double getAf () const
double getFlen () const
double getForceVelocityExtrapolationThreshold () const
double getMinimumFiberLength () const
double getMaximumPennationAngle () const
bool setActivationTimeConstant (double aActivationTimeConstant)
bool setDeactivationTimeConstant (double aDeactivationTimeConstant)
bool setMinimumActivation (double aActivationMinValue)
bool setFmaxTendonStrain (double aFmaxTendonStrain)
bool setFmaxFiberStrain (double aFmaxMuscleStrain)
bool setKshapeActive (double aKshapeActive)
bool setKshapePassive (double aKshapePassive)
bool setAf (double aAf)
bool setFlen (double aFlen)
bool setForceVelocityExtrapolationThreshold (double aFvThresh)
const
MuscleFirstOrderActivationDynamicModel & getActivationModel () const
const
MuscleFixedWidthPennationModel & getPennationModel () const
void printCurveToCSVFile (const CurveType ctype, const std::string &path)
virtual double computeActuation (const SimTK::State &s) const override
Actuator interface for a muscle computes the tension in the muscle and applied by the tendon to bones (i.e. More...
void computeInitialFiberEquilibrium (SimTK::State &s) const override
Compute initial fiber length (velocity) such that muscle fiber and tendon are in static equilibrium and update the state. More...
```

Member Functions

Public member functions list all the available methods. Public member functions are easily accessible, especially in Matlab and python. Protected member functions are generally not accessible by users and are rather used in development of new classes

Next: [Scripting](#)

Previous: [The OpenSim API](#)

Home: [Scripting and Development](#)