



OpenSceneGraph Version 2.9.6

osgTerrain::

Reference Manual

Contents

1	Main Page	1
2	Directory Documentation	3
2.1	include/ Directory Reference	3
2.2	src/osgTerrain/ Directory Reference	4
2.3	include/osgTerrain/ Directory Reference	5
2.4	src/ Directory Reference	6
3	Namespace Documentation	7
3.1	osgTerrain Namespace Reference	7
3.1.1	Detailed Description	7
3.1.2	Function Documentation	8
3.1.2.1	createCompondSetNameAndFileName	8
3.1.2.2	extractSetNameAndFileName	8
4	Class Documentation	9
4.1	CompositeLayer Class Reference	9
4.1.1	Member Typedef Documentation	11
4.1.1.1	Layers	11
4.1.2	Constructor & Destructor Documentation	11
4.1.2.1	CompositeLayer	11
4.1.2.2	CompositeLayer	11
4.1.2.3	~CompositeLayer	11
4.1.3	Member Function Documentation	11
4.1.3.1	addLayer	11
4.1.3.2	addLayer	11
4.1.3.3	addLayer	11
4.1.3.4	clear	11
4.1.3.5	getCompoundName	11
4.1.3.6	getFileName	11
4.1.3.7	getLayer	11
4.1.3.8	getLayer	11
4.1.3.9	getNumLayers	11
4.1.3.10	getSetName	11
4.1.3.11	getSetName	11
4.1.3.12	META_Object	12
4.1.3.13	removeLayer	12
4.1.3.14	setCompoundName	12

4.1.3.15	setFileName	12
4.1.3.16	setLayer	12
4.1.3.17	setSetName	12
4.1.3.18	setSetName	12
4.1.4	Member Data Documentation	12
4.1.4.1	_layers	12
4.2	CompoundNameLayer Struct Reference	13
4.2.1	Constructor & Destructor Documentation	13
4.2.1.1	CompoundNameLayer	13
4.2.1.2	CompoundNameLayer	13
4.2.1.3	CompoundNameLayer	13
4.2.2	Member Function Documentation	13
4.2.2.1	operator=	13
4.2.3	Member Data Documentation	13
4.2.3.1	filename	13
4.2.3.2	layer	13
4.2.3.3	setname	13
4.3	ContourLayer Class Reference	14
4.3.1	Constructor & Destructor Documentation	15
4.3.1.1	ContourLayer	15
4.3.1.2	ContourLayer	15
4.3.1.3	~ContourLayer	15
4.3.2	Member Function Documentation	15
4.3.2.1	dirty	15
4.3.2.2	getImage	15
4.3.2.3	getImage	15
4.3.2.4	getModifiedCount	15
4.3.2.5	getNumColumns	15
4.3.2.6	getNumRows	16
4.3.2.7	getTransferFunction	16
4.3.2.8	getTransferFunction	16
4.3.2.9	getValue	16
4.3.2.10	getValue	16
4.3.2.11	getValue	16
4.3.2.12	getValue	16
4.3.2.13	META_Object	16
4.3.2.14	setModifiedCount	16
4.3.2.15	setTransferFunction	16
4.3.2.16	transform	16
4.3.3	Member Data Documentation	16
4.3.3.1	_tf	16

4.4	GeometryTechnique Class Reference	17
4.4.1	Member Enumeration Documentation	18
4.4.1.1	FilterType	18
4.4.2	Constructor & Destructor Documentation	18
4.4.2.1	GeometryTechnique	18
4.4.2.2	GeometryTechnique	18
4.4.3	Member Function Documentation	18
4.4.3.1	applyColorLayers	18
4.4.3.2	applyTransparency	18
4.4.3.3	cleanSceneGraph	18
4.4.3.4	computeCenterModel	18
4.4.3.5	computeMasterLocator	18
4.4.3.6	cull	18
4.4.3.7	generateGeometry	19
4.4.3.8	getFilterBias	19
4.4.3.9	getFilterMatrix	19
4.4.3.10	getFilterMatrix	19
4.4.3.11	getFilterWidth	19
4.4.3.12	init	19
4.4.3.13	META_Object	19
4.4.3.14	releaseGLObjects	19
4.4.3.15	setFilterBias	19
4.4.3.16	setFilterMatrix	19
4.4.3.17	setFilterMatrixAs	19
4.4.3.18	setFilterWidth	19
4.4.3.19	smoothGeometry	19
4.4.3.20	traverse	19
4.4.3.21	update	19
4.5	HeightFieldLayer Class Reference	20
4.5.1	Constructor & Destructor Documentation	21
4.5.1.1	HeightFieldLayer	21
4.5.1.2	HeightFieldLayer	21
4.5.1.3	~HeightFieldLayer	21
4.5.2	Member Function Documentation	21
4.5.2.1	dirty	21
4.5.2.2	getFileName	21
4.5.2.3	getHeightField	21
4.5.2.4	getHeightField	21
4.5.2.5	getModifiedCount	21
4.5.2.6	getNumColumns	21
4.5.2.7	getNumRows	22

4.5.2.8	getValue	22
4.5.2.9	getValue	22
4.5.2.10	getValue	22
4.5.2.11	getValue	22
4.5.2.12	META_Object	22
4.5.2.13	setFileName	22
4.5.2.14	setHeightField	22
4.5.2.15	setModifiedCount	22
4.5.2.16	transform	22
4.5.3	Member Data Documentation	22
4.5.3.1	_heightField	22
4.5.3.2	_modifiedCount	22
4.6	ImageLayer Class Reference	23
4.6.1	Constructor & Destructor Documentation	24
4.6.1.1	ImageLayer	24
4.6.1.2	ImageLayer	24
4.6.1.3	~ImageLayer	24
4.6.2	Member Function Documentation	24
4.6.2.1	dirty	24
4.6.2.2	getFileName	24
4.6.2.3	getImage	24
4.6.2.4	getImage	25
4.6.2.5	getModifiedCount	25
4.6.2.6	getNumColumns	25
4.6.2.7	getNumRows	25
4.6.2.8	getValue	25
4.6.2.9	getValue	25
4.6.2.10	getValue	25
4.6.2.11	getValue	25
4.6.2.12	META_Object	25
4.6.2.13	setFileName	25
4.6.2.14	setImage	25
4.6.2.15	setModifiedCount	25
4.6.2.16	transform	25
4.6.3	Member Data Documentation	25
4.6.3.1	_image	25
4.7	Layer Class Reference	26
4.7.1	Constructor & Destructor Documentation	28
4.7.1.1	Layer	28
4.7.1.2	Layer	28
4.7.1.3	~Layer	28

4.7.2	Member Function Documentation	28
4.7.2.1	computeBound	28
4.7.2.2	computeIndices	28
4.7.2.3	dirty	28
4.7.2.4	getCompoundName	28
4.7.2.5	getDefaultValue	28
4.7.2.6	getFileName	28
4.7.2.7	getImage	28
4.7.2.8	getImage	28
4.7.2.9	getInterpolatedValue	28
4.7.2.10	getLocator	28
4.7.2.11	getLocator	28
4.7.2.12	getMagFilter	28
4.7.2.13	getMaxLevel	29
4.7.2.14	getMinFilter	29
4.7.2.15	getMinLevel	29
4.7.2.16	getModifiedCount	29
4.7.2.17	getNumColumns	29
4.7.2.18	getNumRows	29
4.7.2.19	getSetName	29
4.7.2.20	getValidDataOperator	29
4.7.2.21	getValidDataOperator	29
4.7.2.22	getValidValue	29
4.7.2.23	getValidValue	29
4.7.2.24	getValidValue	29
4.7.2.25	getValidValue	29
4.7.2.26	getValue	29
4.7.2.27	getValue	29
4.7.2.28	getValue	29
4.7.2.29	getValue	29
4.7.2.30	META_Object	29
4.7.2.31	setDefaultValue	29
4.7.2.32	setFileName	29
4.7.2.33	setLocator	30
4.7.2.34	setMagFilter	30
4.7.2.35	setMaxLevel	30
4.7.2.36	setMinFilter	30
4.7.2.37	setMinLevel	30
4.7.2.38	setModifiedCount	30
4.7.2.39	setSetName	30
4.7.2.40	setValidDataOperator	30

4.7.2.41	transform	30
4.7.3	Member Data Documentation	30
4.7.3.1	_defaultValue	30
4.7.3.2	_filename	30
4.7.3.3	_locator	30
4.7.3.4	_magFilter	30
4.7.3.5	_maxLevel	30
4.7.3.6	_minFilter	30
4.7.3.7	_minLevel	30
4.7.3.8	_validDataOperator	30
4.8	Locator Class Reference	31
4.8.1	Member Enumeration Documentation	32
4.8.1.1	CoordinateSystemType	32
4.8.2	Constructor & Destructor Documentation	33
4.8.2.1	Locator	33
4.8.2.2	Locator	33
4.8.2.3	~Locator	33
4.8.3	Member Function Documentation	33
4.8.3.1	computeLocalBounds	33
4.8.3.2	convertLocalCoordBetween	33
4.8.3.3	convertLocalToModel	33
4.8.3.4	convertModelToLocal	33
4.8.3.5	getCoordinateSystem	33
4.8.3.6	getCoordinateSystemType	33
4.8.3.7	getDefinedInFile	33
4.8.3.8	getEllipsoidModel	33
4.8.3.9	getEllipsoidModel	33
4.8.3.10	getFormat	33
4.8.3.11	getTransform	33
4.8.3.12	getTransformScaledByResolution	33
4.8.3.13	META_Object	33
4.8.3.14	orientationOpenGL	33
4.8.3.15	setCoordinateSystem	33
4.8.3.16	setCoordinateSystemType	33
4.8.3.17	setDefinedInFile	33
4.8.3.18	setEllipsoidModel	33
4.8.3.19	setFormat	33
4.8.3.20	setTransform	33
4.8.3.21	setTransformAsExtents	34
4.8.3.22	setTransformScaledByResolution	34
4.8.4	Member Data Documentation	34

4.8.4.1	_coordinateSystemType	34
4.8.4.2	_cs	34
4.8.4.3	_definedInFile	34
4.8.4.4	_ellipsoidModel	34
4.8.4.5	_format	34
4.8.4.6	_inverse	34
4.8.4.7	_transform	34
4.8.4.8	_transformScaledByResolution	34
4.9	NoDataValue Struct Reference	35
4.9.1	Constructor & Destructor Documentation	35
4.9.1.1	NoDataValue	35
4.9.2	Member Function Documentation	35
4.9.2.1	getValue	35
4.9.2.2	operator()	35
4.9.2.3	setNoDataValue	35
4.9.3	Member Data Documentation	35
4.9.3.1	_value	35
4.10	ProxyLayer Class Reference	36
4.10.1	Constructor & Destructor Documentation	37
4.10.1.1	ProxyLayer	37
4.10.1.2	ProxyLayer	37
4.10.1.3	~ProxyLayer	37
4.10.2	Member Function Documentation	37
4.10.2.1	computeBound	37
4.10.2.2	dirty	38
4.10.2.3	getFileName	38
4.10.2.4	getImage	38
4.10.2.5	getImage	38
4.10.2.6	getImplementation	38
4.10.2.7	getImplementation	38
4.10.2.8	getModifiedCount	38
4.10.2.9	getNumColumns	38
4.10.2.10	getNumRows	38
4.10.2.11	getValue	38
4.10.2.12	getValue	38
4.10.2.13	getValue	38
4.10.2.14	getValue	38
4.10.2.15	META_Object	38
4.10.2.16	setFileName	38
4.10.2.17	setImplementation	38
4.10.2.18	setModifiedCount	39

4.10.2.19	transform	39
4.10.3	Member Data Documentation	39
4.10.3.1	_implementation	39
4.11	SwitchLayer Class Reference	40
4.11.1	Constructor & Destructor Documentation	41
4.11.1.1	SwitchLayer	41
4.11.1.2	SwitchLayer	41
4.11.1.3	~SwitchLayer	41
4.11.2	Member Function Documentation	41
4.11.2.1	getActiveLayer	41
4.11.2.2	getImage	41
4.11.2.3	getImage	41
4.11.2.4	META_Object	41
4.11.2.5	setActiveLayer	41
4.11.3	Member Data Documentation	41
4.11.3.1	_activeLayer	41
4.12	Terrain Class Reference	42
4.12.1	Detailed Description	43
4.12.2	Member Typedef Documentation	43
4.12.2.1	TerrainTileMap	43
4.12.2.2	TerrainTileSet	43
4.12.3	Constructor & Destructor Documentation	43
4.12.3.1	Terrain	43
4.12.3.2	Terrain	43
4.12.3.3	~Terrain	43
4.12.4	Member Function Documentation	43
4.12.4.1	dirtyRegisteredTiles	43
4.12.4.2	getSampleRatio	43
4.12.4.3	getTerrainTechniquePrototype	43
4.12.4.4	getTerrainTechniquePrototype	43
4.12.4.5	getTile	43
4.12.4.6	getTile	44
4.12.4.7	getVerticalScale	44
4.12.4.8	META_Node	44
4.12.4.9	registerTerrainTile	44
4.12.4.10	setSampleRatio	44
4.12.4.11	setTerrainTechniquePrototype	44
4.12.4.12	setVerticalScale	44
4.12.4.13	traverse	44
4.12.4.14	unregisterTerrainTile	44
4.12.5	Friends And Related Function Documentation	44

4.12.5.1	TerrainTile	44
4.12.6	Member Data Documentation	44
4.12.6.1	_mutex	44
4.12.6.2	_sampleRatio	44
4.12.6.3	_terrainTechnique	44
4.12.6.4	_terrainTileMap	44
4.12.6.5	_terrainTileSet	44
4.12.6.6	_verticalScale	44
4.13	TerrainTechnique Class Reference	45
4.13.1	Constructor & Destructor Documentation	46
4.13.1.1	TerrainTechnique	46
4.13.1.2	TerrainTechnique	46
4.13.1.3	~TerrainTechnique	46
4.13.2	Member Function Documentation	46
4.13.2.1	cleanSceneGraph	46
4.13.2.2	cull	46
4.13.2.3	getTerrainTile	46
4.13.2.4	getTerrainTile	46
4.13.2.5	init	46
4.13.2.6	META_Object	46
4.13.2.7	releaseGLObjects	46
4.13.2.8	setDirty	46
4.13.2.9	traverse	46
4.13.2.10	update	46
4.13.3	Friends And Related Function Documentation	47
4.13.3.1	osgTerrain::TerrainTile	47
4.13.4	Member Data Documentation	47
4.13.4.1	_terrainTile	47
4.14	TerrainTile Class Reference	48
4.14.1	Detailed Description	50
4.14.2	Member Typedef Documentation	50
4.14.2.1	Layers	50
4.14.3	Constructor & Destructor Documentation	50
4.14.3.1	TerrainTile	50
4.14.3.2	TerrainTile	50
4.14.3.3	~TerrainTile	51
4.14.4	Member Function Documentation	51
4.14.4.1	computeBound	51
4.14.4.2	getColorLayer	51
4.14.4.3	getColorLayer	51
4.14.4.4	getDirty	51

4.14.4.5	getElevationLayer	51
4.14.4.6	getElevationLayer	51
4.14.4.7	getLocator	51
4.14.4.8	getLocator	51
4.14.4.9	getNumColorLayers	51
4.14.4.10	getRequiresNormals	51
4.14.4.11	getTerrain	51
4.14.4.12	getTerrain	51
4.14.4.13	getTerrainTechnique	51
4.14.4.14	getTerrainTechnique	51
4.14.4.15	getTileID	51
4.14.4.16	getTileLoadedCallback	51
4.14.4.17	getTreatBoundariesToValidDataAsDefaultValue	51
4.14.4.18	init	51
4.14.4.19	META_Node	52
4.14.4.20	releaseGLObjects	52
4.14.4.21	setColorLayer	52
4.14.4.22	setDirty	52
4.14.4.23	setElevationLayer	52
4.14.4.24	setLocator	52
4.14.4.25	setRequiresNormals	52
4.14.4.26	setTerrain	52
4.14.4.27	setTerrainTechnique	52
4.14.4.28	setTileID	52
4.14.4.29	setTileLoadedCallback	52
4.14.4.30	setTreatBoundariesToValidDataAsDefaultValue	52
4.14.4.31	traverse	53
4.14.5	Friends And Related Function Documentation	53
4.14.5.1	Terrain	53
4.14.6	Member Data Documentation	53
4.14.6.1	_colorLayers	53
4.14.6.2	_dirty	53
4.14.6.3	_elevationLayer	53
4.14.6.4	_hasBeenTraversal	53
4.14.6.5	_locator	53
4.14.6.6	_requiresNormals	53
4.14.6.7	_terrain	53
4.14.6.8	_terrainTechnique	53
4.14.6.9	_tileID	53
4.14.6.10	_treatBoundariesToValidDataAsDefaultValue	53
4.15	TileID Class Reference	54

4.15.1	Constructor & Destructor Documentation	54
4.15.1.1	TileID	54
4.15.1.2	TileID	54
4.15.2	Member Function Documentation	54
4.15.2.1	operator!=	54
4.15.2.2	operator<	54
4.15.2.3	operator==	54
4.15.2.4	valid	54
4.15.3	Member Data Documentation	54
4.15.3.1	level	54
4.15.3.2	x	54
4.15.3.3	y	54
4.16	TileLoadedCallback Struct Reference	55
4.16.1	Detailed Description	55
4.16.2	Member Function Documentation	55
4.16.2.1	deferExternalLayerLoading	55
4.16.2.2	loaded	55
4.17	TransformOperator Struct Reference	56
4.17.1	Constructor & Destructor Documentation	56
4.17.1.1	TransformOperator	56
4.17.2	Member Function Documentation	56
4.17.2.1	operator()	56
4.17.2.2	operator()	56
4.17.2.3	operator()	56
4.17.2.4	operator()	56
4.17.2.5	operator()	56
4.17.2.6	operator()	56
4.17.2.7	operator()	56
4.17.3	Member Data Documentation	56
4.17.3.1	_offset	56
4.17.3.2	_scale	56
4.18	ValidDataOperator Struct Reference	57
4.18.1	Member Function Documentation	57
4.18.1.1	operator()	57
4.18.1.2	operator()	57
4.18.1.3	operator()	57
4.18.1.4	operator()	57
4.19	ValidRange Struct Reference	58
4.19.1	Constructor & Destructor Documentation	58
4.19.1.1	ValidRange	58
4.19.2	Member Function Documentation	58

4.19.2.1	getMaxValue	58
4.19.2.2	getMinValue	58
4.19.2.3	operator()	58
4.19.2.4	setMaxValue	59
4.19.2.5	setMinValue	59
4.19.2.6	setRange	59
4.19.3	Member Data Documentation	59
4.19.3.1	_maxValue	59
4.19.3.2	_minValue	59
4.20	WhiteListTileLoadedCallback Class Reference	60
4.20.1	Detailed Description	61
4.20.2	Member Typedef Documentation	61
4.20.2.1	SetWhiteList	61
4.20.3	Constructor & Destructor Documentation	61
4.20.3.1	WhiteListTileLoadedCallback	61
4.20.3.2	~WhiteListTileLoadedCallback	61
4.20.4	Member Function Documentation	61
4.20.4.1	allow	61
4.20.4.2	deferExternalLayerLoading	61
4.20.4.3	getAllowAll	61
4.20.4.4	getMinimumNumOfLayers	61
4.20.4.5	getReplaceSwitchLayer	61
4.20.4.6	layerAcceptable	61
4.20.4.7	loaded	61
4.20.4.8	readImageLayer	61
4.20.4.9	setAllowAll	61
4.20.4.10	setMinimumNumOfLayers	61
4.20.4.11	setReplaceSwitchLayer	61
4.20.5	Member Data Documentation	61
4.20.5.1	_allowAll	61
4.20.5.2	_mininumNumberOfLayers	61
4.20.5.3	_replaceSwitchLayer	61
4.20.5.4	_setWhiteList	61
5	File Documentation	63
5.1	Export File Reference	63
5.1.1	Define Documentation	63
5.1.1.1	OSGTERRAIN_EXPORT	63
5.1.1.2	OSGTERRAIN_EXPORT_	63
5.2	GeometryTechnique File Reference	64
5.2.1	Define Documentation	64

5.2.1.1	OSGTERRAIN_GEOMETRYTECHNIQUE	64
5.3	GeometryTechnique.cpp File Reference	65
5.3.1	Define Documentation	65
5.3.1.1	NEW_COORD_CODE	65
5.4	Layer File Reference	66
5.4.1	Define Documentation	67
5.4.1.1	MAXIMUM_NUMBER_OF_LEVELS	67
5.4.1.2	OSGTERRAIN_LAYER	67
5.5	Layer.cpp File Reference	68
5.5.1	Function Documentation	68
5.5.1.1	_processRow	68
5.5.1.2	processImage	68
5.5.1.3	processRow	68
5.6	Locator File Reference	69
5.6.1	Define Documentation	69
5.6.1.1	OSGTERRAIN_LOCATOR	69
5.7	Locator.cpp File Reference	70
5.8	mainpage.h File Reference	71
5.8.1	Detailed Description	71
5.9	Terrain File Reference	72
5.9.1	Define Documentation	72
5.9.1.1	OSGTerrain	72
5.10	Terrain.cpp File Reference	73
5.10.1	Variable Documentation	73
5.10.1.1	s_maxNumTiles	73
5.11	TerrainTechnique File Reference	74
5.11.1	Define Documentation	74
5.11.1.1	OSGTERRAIN_TERRAINTECHNIQUE	74
5.12	TerrainTechnique.cpp File Reference	75
5.13	TerrainTile File Reference	76
5.13.1	Define Documentation	77
5.13.1.1	OSGTERRAIN_TERRAINTILE	77
5.14	TerrainTile.cpp File Reference	78
5.15	ValidDataOperator File Reference	79
5.15.1	Define Documentation	79
5.15.1.1	OSGTERRAIN_VALIDDATAOPERATOR	79
5.16	Version File Reference	80
5.16.1	Define Documentation	80
5.16.1.1	OSGTERRAIN_VERSION	80
5.16.2	Function Documentation	80
5.16.2.1	osgTerrainGetLibraryName	80

5.16.2.2	osgTerrainGetVersion	80
5.17	Version.cpp File Reference	81
5.17.1	Function Documentation	81
5.17.1.1	osgTerrainGetLibraryName	81
5.17.1.2	osgTerrainGetVersion	81

Main Page

The OpenSceneGraph exists as a number of modules, each sitting in its own library, enclosed within its own namespace. At the very core lies the osg library. This contains the OpenSceneGraph's central classes and, at the bare minimum, it is all users need to write an OpenSceneGraph program in C++.

Around and alongside this sit other supporting libraries, such as osgUtil (containing visitors for app traversals, cull traversals, scene graph optimizers and so on), osgDB (for handling plug-ins, shared library loading, database reading and writing and the like), osgText, osgParticle, etc.

Extensive online documentation is available from the OSG Support section to help in using Open Scene Graph.

The project's original reference guides generated by Doxygen from the source code may be downloaded as a single file from the OSG Reference Guides section.

To download source code, binaries, dependencies and sample datasets visit the OSG Download page.

For more about dependencies see the OSG Dependencies page.

The documentation you are looking at can be downloaded from www.3draum.ch.

Enjoy!

Directory Documentation

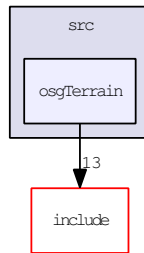
2.1 include/ Directory Reference



Directories

- directory **osgTerrain**

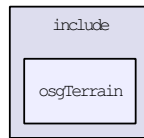
2.2 src/osgTerrain/ Directory Reference



Files

- file **GeometryTechnique.cpp**
- file **Layer.cpp**
- file **Locator.cpp**
- file **Terrain.cpp**
- file **TerrainTechnique.cpp**
- file **TerrainTile.cpp**
- file **Version.cpp**

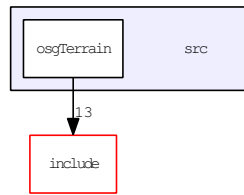
2.3 include/osgTerrain/ Directory Reference



Files

- file **Export**
- file **GeometryTechnique**
- file **Layer**
- file **Locator**
- file **mainpage.h**
- file **Terrain**
- file **TerrainTechnique**
- file **TerrainTile**
- file **ValidDataOperator**
- file **Version**

2.4 src/ Directory Reference



Directories

- directory **osgTerrain**

Namespace Documentation

3.1 osgTerrain Namespace Reference

The **osgTerrain** (p. 7) library is a NodeKit that provides geospecific terrain rendering support.

Classes

- class **CompositeLayer**
- class **ContourLayer**
- class **GeometryTechnique**
- class **HeightFieldLayer**
- class **ImageLayer**
- class **Layer**
- class **Locator**
- struct **NoDataValue**
- class **ProxyLayer**
- class **SwitchLayer**
- class **Terrain**

Terrain (p. 42) provides a framework for loosely coupling height field data with height rendering algorithms.

- class **TerrainTechnique**
- class **TerrainTile**

Terrain (p. 42) provides a framework for loosely coupling height field data with height rendering algorithms.

- class **TileID**
- struct **ValidDataOperator**
- struct **ValidRange**
- class **WhiteListTileLoadedCallback**

Helper callback for managing optional sets of layers, that loading of is deferred to this callback, with this callback working out which layers to load, and how to create fallback versions of the layers.

Functions

- OSGTERRAIN_EXPORT std::string **createCompoundSetNameAndFileName** (const std::string &setname, const std::string &filename)
- Create a compound string in the form set:setname:filename, or just filename if setname is "".*
- OSGTERRAIN_EXPORT void **extractSetNameAndFileName** (const std::string &compoundstring, std::string &setname, std::string &filename)
- Extract the setname and filename from a compound string in the form set:setname:filename".*

3.1.1 Detailed Description

The **osgTerrain** (p. 7) library is a NodeKit that provides geospecific terrain rendering support.

3.1.2 Function Documentation

3.1.2.1 `std::string createCompondSetNameAndFileName (const std::string & setname, const std::string & filename)`

Create a compound string in the form `set:setname:filename`, or just `filename` if `setname` is `""`.

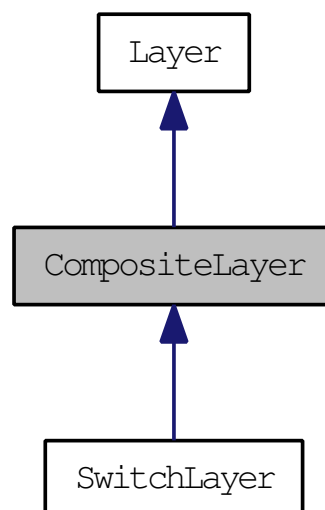
3.1.2.2 `void extractSetNameAndFileName (const std::string & compoundstring, std::string & setname, std::string & filename)`

Extact the `setname` and `filename` from a compound string in the from `set:setname:filename`". Returns a `setname` of `""` when non `set:setname:` entry is present.

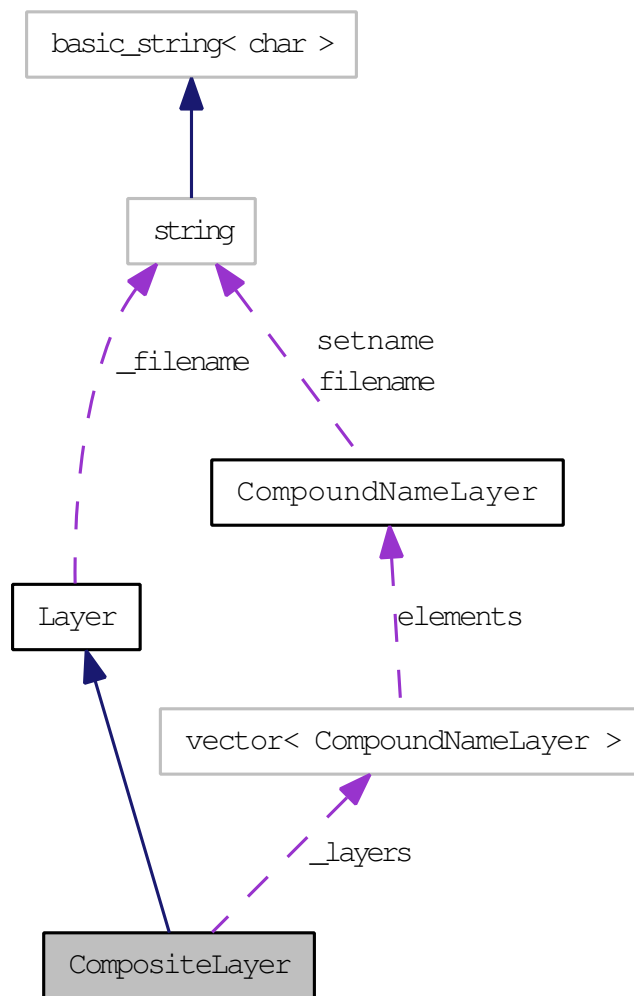
Class Documentation

4.1 CompositeLayer Class Reference

Inheritance diagram for CompositeLayer:



Collaboration diagram for CompositeLayer:



Classes

- struct **CompoundNameLayer**

Public Member Functions

- **CompositeLayer** (const **CompositeLayer** &compositeLayer, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
Copy constructor using CopyOp to manage deep vs shallow copy.
- **CompositeLayer** ()
- void **addLayer** (**Layer** *layer)
- void **addLayer** (const std::string &setname, const std::string &filename)
- void **addLayer** (const std::string &compoundname)
- void **clear** ()
- std::string **getCompoundName** (unsigned int i) const
- const std::string & **getFileName** (unsigned int i) const
- const **Layer** * **getLayer** (unsigned int i) const
- **Layer** * **getLayer** (unsigned int i)
- unsigned int **getNumLayers** () const
- const std::string & **getSetName** (unsigned int i) const
- const std::string & **getSetName** () const
- **META_Object** (osgTerrain, **CompositeLayer**)

- void **removeLayer** (unsigned int *i*)
- void **setCompoundName** (unsigned int *i*, const std::string &compoundname)
- void **setFileName** (unsigned int *i*, const std::string &filename)
- void **setLayer** (unsigned int *i*, **Layer** *layer)
- void **setSetName** (unsigned int *i*, const std::string &setname)
- void **setSetName** (const std::string &setname)

Protected Types

- typedef std::vector< **CompoundNameLayer** > **Layers**

Protected Member Functions

- virtual ~**CompositeLayer** ()

Protected Attributes

- **Layers** _layers

4.1.1 Member Typedef Documentation

4.1.1.1 typedef std::vector< **CompoundNameLayer** > **Layers** [protected]

4.1.2 Constructor & Destructor Documentation

4.1.2.1 **CompositeLayer** ()

4.1.2.2 **CompositeLayer** (const **CompositeLayer** & *compositeLayer*, const osg::CopyOp & *copyop* = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.1.2.3 virtual ~**CompositeLayer** () [inline, protected, virtual]

4.1.3 Member Function Documentation

4.1.3.1 void **addLayer** (**Layer** * *layer*) [inline]

4.1.3.2 void **addLayer** (const std::string & *setname*, const std::string & *filename*)

4.1.3.3 void **addLayer** (const std::string & *compoundname*)

4.1.3.4 void **clear** ()

4.1.3.5 std::string **getCompoundName** (unsigned int *i*) const

4.1.3.6 const std::string& **getFileName** (unsigned int *i*) const [inline]

4.1.3.7 const **Layer*** **getLayer** (unsigned int *i*) const [inline]

4.1.3.8 **Layer*** **getLayer** (unsigned int *i*) [inline]

4.1.3.9 unsigned int **getNumLayers** () const [inline]

4.1.3.10 const std::string& **getSetName** (unsigned int *i*) const [inline]

4.1.3.11 const std::string& **getSetName** () const [inline]

Reimplemented from **Layer** (p. 29).

4.1.3.12 **META_Object** (osgTerrain, CompositeLayer)

4.1.3.13 **void removeLayer** (unsigned int *i*) [inline]

4.1.3.14 **void setCompoundName** (unsigned int *i*, const std::string & *compoundname*)

4.1.3.15 **void setFileName** (unsigned int *i*, const std::string & *filename*) [inline]

4.1.3.16 **void setLayer** (unsigned int *i*, Layer * *layer*) [inline]

4.1.3.17 **void setSetName** (unsigned int *i*, const std::string & *setname*) [inline]

4.1.3.18 **void setSetName** (const std::string & *setname*) [inline]

Reimplemented from **Layer** (p. 30).

4.1.4 Member Data Documentation

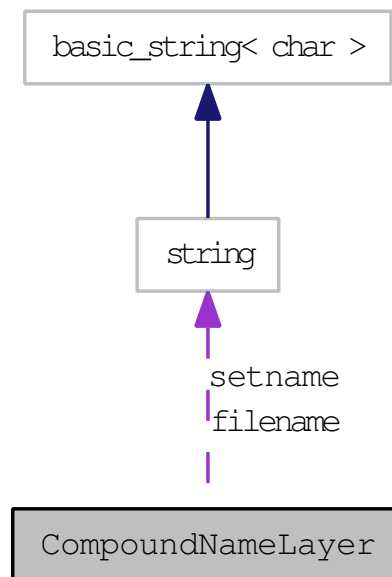
4.1.4.1 **Layers_layers** [protected]

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

- **Layer**
- **Layer.cpp**

4.2 CompoundNameLayer Struct Reference

Collaboration diagram for CompoundNameLayer:



Public Member Functions

- **CompoundNameLayer** (const std::string &sn, const std::string &fn, **Layer** *l)
- **CompoundNameLayer** (const **CompoundNameLayer** &cnl)
- **CompoundNameLayer** ()
- **CompoundNameLayer** & **operator=** (const **CompoundNameLayer** &cnl)

Public Attributes

- std::string **filename**
- osg::ref_ptr< **Layer** > **layer**
- std::string **setname**

4.2.1 Constructor & Destructor Documentation

4.2.1.1 **CompoundNameLayer** () [inline]

4.2.1.2 **CompoundNameLayer** (const **CompoundNameLayer** & *cnl*) [inline]

4.2.1.3 **CompoundNameLayer** (const std::string & *sn*, const std::string & *fn*, **Layer** * *l*) [inline]

4.2.2 Member Function Documentation

4.2.2.1 **CompoundNameLayer**& **operator=** (const **CompoundNameLayer** & *cnl*) [inline]

4.2.3 Member Data Documentation

4.2.3.1 std::string **filename**

4.2.3.2 osg::ref_ptr<**Layer**> **layer**

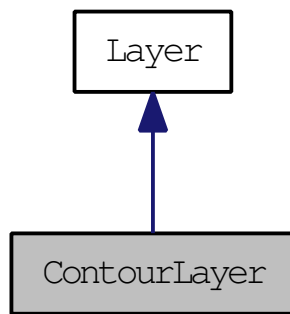
4.2.3.3 std::string **setname**

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

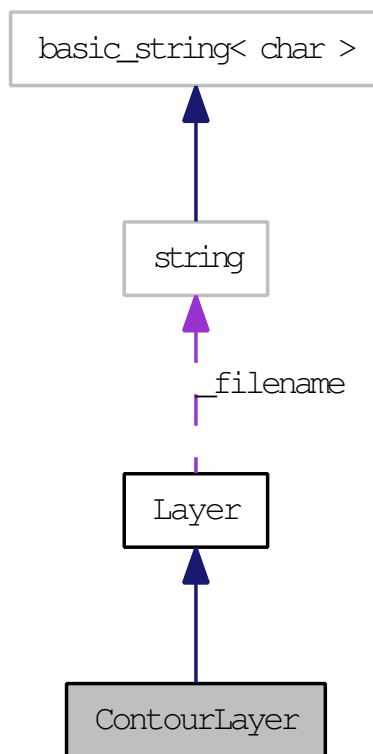
- **Layer**

4.3 ContourLayer Class Reference

Inheritance diagram for ContourLayer:



Collaboration diagram for ContourLayer:



Public Member Functions

- **ContourLayer** (const **ContourLayer** &tfLayer, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
 - Copy constructor using CopyOp to manage deep vs shallow copy.*
- **ContourLayer** (osg::TransferFunction1D *tf=0)
- virtual void **dirty** ()
 - increment the modified count.*
- virtual const osg::Image * **getImage** () const
 - Return const image associated with layer.*
- virtual osg::Image * **getImage** ()

Return image associated with layer.

- virtual unsigned int **getModifiedCount** () const
Get modified count value.
- virtual unsigned int **getNumColumns** () const
- virtual unsigned int **getNumRows** () const
- const osg::TransferFunction1D * **getTransferFunction** () const
- osg::TransferFunction1D * **getTransferFunction** ()
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec4 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec3 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec2 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, float &value) const
- **META_Object** (osgTerrain, **ContourLayer**)
- virtual void **setModifiedCount** (unsigned int value)
Set the modified count value.
- void **setTransferFunction** (osg::TransferFunction1D *tf)
- virtual bool **transform** (float offset, float scale)

Protected Member Functions

- virtual ~**ContourLayer** ()

Protected Attributes

- osg::ref_ptr< osg::TransferFunction1D > **_tf**

4.3.1 Constructor & Destructor Documentation

4.3.1.1 ContourLayer (osg::TransferFunction1D * tf = 0)

4.3.1.2 ContourLayer (const ContourLayer & tfLayer, const osg::CopyOp & copyop = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.3.1.3 virtual ~ContourLayer () [inline, protected, virtual]

4.3.2 Member Function Documentation

4.3.2.1 void dirty () [virtual]

increment the modified count. "

Reimplemented from **Layer** (p. 28).

4.3.2.2 virtual const osg::Image* getImage () const [inline, virtual]

Return const image associated with layer.

Reimplemented from **Layer** (p. 28).

4.3.2.3 virtual osg::Image* getImage () [inline, virtual]

Return image associated with layer.

Reimplemented from **Layer** (p. 28).

4.3.2.4 unsigned int getModifiedCount () const [virtual]

Get modified count value.

Reimplemented from **Layer** (p. 29).

4.3.2.5 virtual unsigned int getNumColumns () const [inline, virtual]

Reimplemented from **Layer** (p. 29).

4.3.2.6 virtual unsigned int getNumRows () const [inline, virtual]

Reimplemented from **Layer** (p. 29).

4.3.2.7 const osg::TransferFunction1D* getTransferFunction () const [inline]

4.3.2.8 osg::TransferFunction1D* getTransferFunction () [inline]

4.3.2.9 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec4 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.3.2.10 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec3 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.3.2.11 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec2 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.3.2.12 bool getValue (unsigned int *i*, unsigned int *j*, float & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.3.2.13 META_Object (osgTerrain, ContourLayer)

4.3.2.14 void setModifiedCount (unsigned int) [virtual]

Set the modified count value.

Reimplemented from **Layer** (p. 30).

4.3.2.15 void setTransferFunction (osg::TransferFunction1D * *tf*)

4.3.2.16 bool transform (float *offset*, float *scale*) [virtual]

Reimplemented from **Layer** (p. 30).

4.3.3 Member Data Documentation

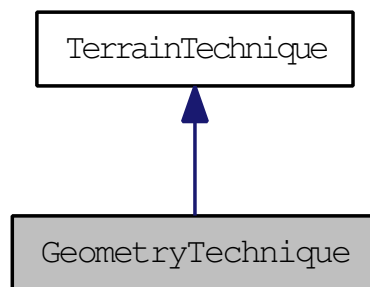
4.3.3.1 osg::ref_ptr<osg::TransferFunction1D> _tf [protected]

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

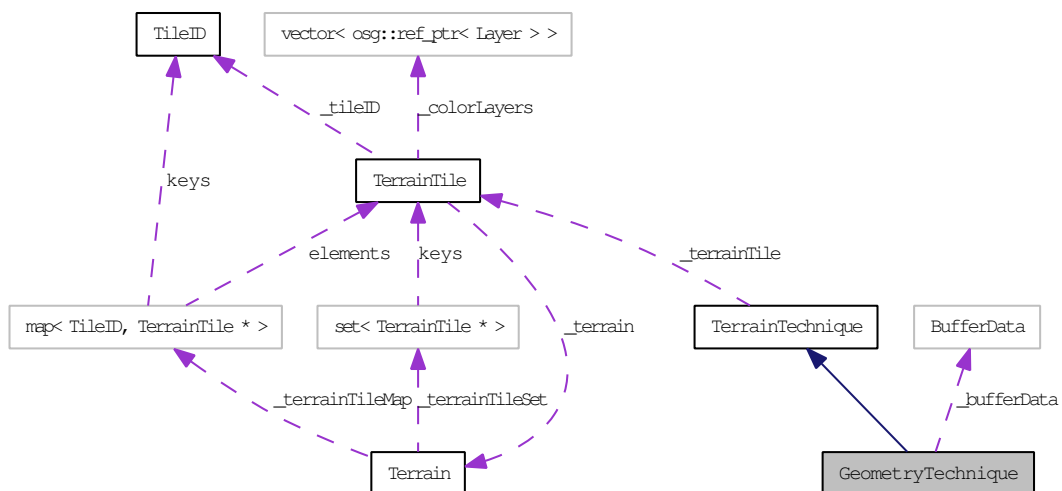
- **Layer**
- **Layer.cpp**

4.4 GeometryTechnique Class Reference

Inheritance diagram for GeometryTechnique:



Collaboration diagram for GeometryTechnique:



Classes

- struct **BufferData**

Public Types

- enum **FilterType** { **GAUSSIAN**, **SMOOTH**, **SHARPEN** }

Public Member Functions

- **GeometryTechnique** (const **GeometryTechnique** &, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

- **GeometryTechnique** ()
- virtual void **applyColorLayers** ()
- virtual void **applyTransparency** ()
- virtual void **cleanSceneGraph** ()

Clean scene graph from any terrain technique specific nodes.

- virtual osg::Vec3d **computeCenterModel** (Locator *masterLocator)
- virtual Locator * **computeMasterLocator** ()
- virtual void **cull** (osgUtil::CullVisitor *nv)

- virtual void **generateGeometry** (**Locator** *masterLocator, const osg::Vec3d ¢erModel)
- float **getFilterBias** () const
- const osg::Matrix3 & **getFilterMatrix** () const
- osg::Matrix3 & **getFilterMatrix** ()
- float **getFilterWidth** () const
- virtual void **init** ()
- **META_Object** (osgTerrain, **GeometryTechnique**)
- virtual void **releaseGLObjets** (osg::State *s=0) const
If State is non-zero, this function releases any associated OpenGL objects for the specified graphics context.
- void **setFilterBias** (float filterBias)
- void **setFilterMatrix** (const osg::Matrix3 &matrix)
- void **setFilterMatrixAs** (**FilterType** filterType)
- void **setFilterWidth** (float filterWidth)
- virtual void **smoothGeometry** ()
- virtual void **traverse** (osg::NodeVisitor &nv)
Traverse the terrain subgraph.
- virtual void **update** (osgUtil::UpdateVisitor *nv)

4.4.1 Member Enumeration Documentation

4.4.1.1 enum FilterType

Enumerator:

GAUSSIAN
SMOOTH
SHARPEN

4.4.2 Constructor & Destructor Documentation

4.4.2.1 GeometryTechnique ()

4.4.2.2 GeometryTechnique (const GeometryTechnique & gt, const osg::CopyOp & copyop = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.4.3 Member Function Documentation

4.4.3.1 void applyColorLayers () [virtual]

4.4.3.2 void applyTransparency () [virtual]

4.4.3.3 void cleanSceneGraph () [virtual]

Clean scene graph from any terrain technique specific nodes.

Reimplemented from **TerrainTechnique** (p. 46).

4.4.3.4 osg::Vec3d computeCenterModel (Locator * masterLocator) [virtual]

4.4.3.5 Locator * computeMasterLocator () [virtual]

4.4.3.6 void cull (osgUtil::CullVisitor * nv) [virtual]

Reimplemented from **TerrainTechnique** (p. 46).

4.4.3.7 void generateGeometry (Locator * *masterLocator*, const osg::Vec3d & *centerModel*) [virtual]

4.4.3.8 float getFilterBias () const [inline]

4.4.3.9 const osg::Matrix3& getFilterMatrix () const [inline]

4.4.3.10 osg::Matrix3& getFilterMatrix () [inline]

4.4.3.11 float getFilterWidth () const [inline]

4.4.3.12 void init () [virtual]

Reimplemented from **TerrainTechnique** (p. 46).

4.4.3.13 META_Object (osgTerrain, GeometryTechnique)

4.4.3.14 void releaseGLObjets (osg::State * *state* = 0) const [virtual]

If State is non-zero, this function releases any associated OpenGL objects for the specified graphics context. Otherwise, releases OpenGL objects for all graphics contexts.

Reimplemented from **TerrainTechnique** (p. 46).

4.4.3.15 void setFilterBias (float *filterBias*)

4.4.3.16 void setFilterMatrix (const osg::Matrix3 & *matrix*)

4.4.3.17 void setFilterMatrixAs (FilterType *filterType*)

4.4.3.18 void setFilterWidth (float *filterWidth*)

4.4.3.19 void smoothGeometry () [virtual]

4.4.3.20 void traverse (osg::NodeVisitor & *nv*) [virtual]

Traverse the terrain subgraph.

Reimplemented from **TerrainTechnique** (p. 46).

4.4.3.21 void update (osgUtil::UpdateVisitor * *uv*) [virtual]

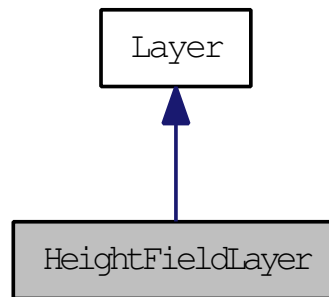
Reimplemented from **TerrainTechnique** (p. 46).

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

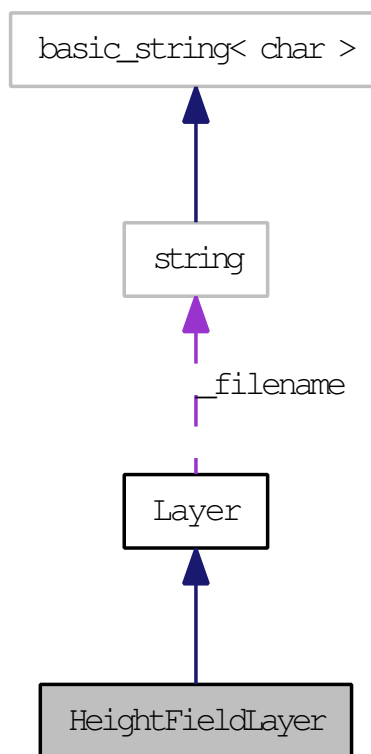
- **GeometryTechnique**
- **GeometryTechnique.cpp**

4.5 HeightFieldLayer Class Reference

Inheritance diagram for HeightFieldLayer:



Collaboration diagram for HeightFieldLayer:



Public Member Functions

- **HeightFieldLayer** (const **HeightFieldLayer** &hfLayer, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
Copy constructor using CopyOp to manage deep vs shallow copy.
- **HeightFieldLayer** (osg::HeightField *hf=0)
- virtual void **dirty** ()
increment the modified count.
- virtual const std::string & **getFileName** () const
Get the file name of the layer.
- const osg::HeightField * **getHeightField** () const

- osg::HeightField * **getHeightField** ()
- virtual unsigned int **getModifiedCount** () const
Get modified count value.
- virtual unsigned int **getNumColumns** () const
- virtual unsigned int **getNumRows** () const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec4 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec3 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec2 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, float &value) const
- **META_Object** (osgTerrain, **HeightFieldLayer**)
- void **setFileName** (const std::string &filename)
Set the file name of the data associated with this layer.
- void **setHeightField** (osg::HeightField *hf)
- virtual void **setModifiedCount** (unsigned int value)
Set the modified count value.
- virtual bool **transform** (float offset, float scale)

Protected Member Functions

- virtual ~**HeightFieldLayer** ()

Protected Attributes

- osg::ref_ptr< osg::HeightField > **_heightField**
- unsigned int **_modifiedCount**

4.5.1 Constructor & Destructor Documentation

4.5.1.1 HeightFieldLayer (osg::HeightField * hf = 0)

4.5.1.2 HeightFieldLayer (const HeightFieldLayer & hfLayer, const osg::CopyOp & copyop = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.5.1.3 virtual ~HeightFieldLayer () [inline, protected, virtual]

4.5.2 Member Function Documentation

4.5.2.1 void dirty () [virtual]

increment the modified count. "

Reimplemented from **Layer** (p. 28).

4.5.2.2 virtual const std::string& getFileName () const [inline, virtual]

Get the file name of the layer.

Reimplemented from **Layer** (p. 28).

4.5.2.3 const osg::HeightField* getHeightField () const [inline]

4.5.2.4 osg::HeightField* getHeightField () [inline]

4.5.2.5 unsigned int getModifiedCount () const [virtual]

Get modified count value.

Reimplemented from **Layer** (p. 29).

4.5.2.6 virtual unsigned int getNumColumns () const [inline, virtual]

Reimplemented from **Layer** (p. 29).

4.5.2.7 virtual unsigned int getNumRows () const [inline, virtual]

Reimplemented from **Layer** (p. 29).

4.5.2.8 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec4 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.5.2.9 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec3 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.5.2.10 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec2 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.5.2.11 bool getValue (unsigned int *i*, unsigned int *j*, float & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.5.2.12 META_Object (osgTerrain, HeightFieldLayer)**4.5.2.13 void setFileName (const std::string & *filename*) [inline, virtual]**

Set the file name of the data associated with this layer.

Reimplemented from **Layer** (p. 29).

4.5.2.14 void setHeightField (osg::HeightField * *hf*)**4.5.2.15 void setModifiedCount (unsigned int) [virtual]**

Set the modified count value.

Reimplemented from **Layer** (p. 30).

4.5.2.16 bool transform (float *offset*, float *scale*) [virtual]

Reimplemented from **Layer** (p. 30).

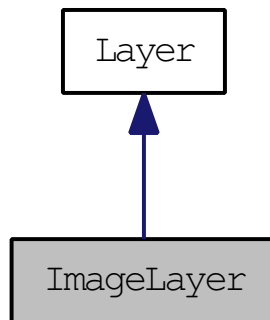
4.5.3 Member Data Documentation**4.5.3.1 osg::ref_ptr<osg::HeightField> _heightField [protected]****4.5.3.2 unsigned int _modifiedCount [protected]**

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

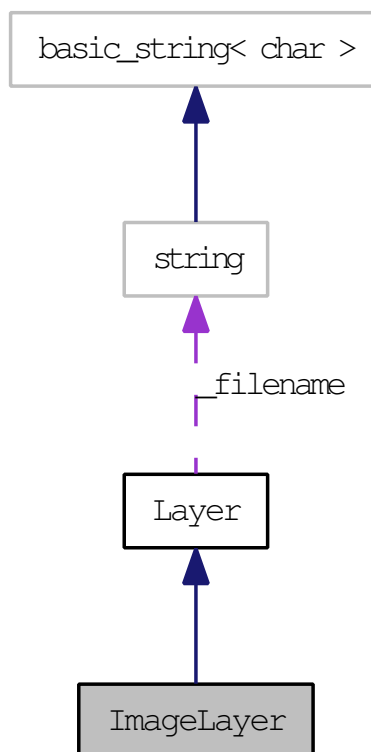
- **Layer**
- **Layer.cpp**

4.6 ImageLayer Class Reference

Inheritance diagram for ImageLayer:



Collaboration diagram for ImageLayer:



Public Member Functions

- **ImageLayer** (const **ImageLayer** &imageLayer, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
Copy constructor using CopyOp to manage deep vs shallow copy.
- **ImageLayer** (osg::Image *image=0)
- virtual void **dirty** ()
increment the modified count.
- virtual const std::string & **getFileName** () const
Get the file name of the layer.

- virtual const osg::Image * **getImage** () const
Return const image associated with layer.
- virtual osg::Image * **getImage** ()
Return image associated with layer.
- virtual unsigned int **getModifiedCount** () const
Get modified count value.
- virtual unsigned int **getNumColumns** () const
- virtual unsigned int **getNumRows** () const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec4 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec3 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec2 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, float &value) const
- **META_Object** (osgTerrain, **ImageLayer**)
- void **setFileName** (const std::string &filename)
Set the file name of the data associated with this layer.
- void **setImage** (osg::Image *image)
- virtual void **setModifiedCount** (unsigned int value)
Set the modified count value.
- virtual bool **transform** (float offset, float scale)

Protected Member Functions

- virtual ~**ImageLayer** ()

Protected Attributes

- osg::ref_ptr< osg::Image > **_image**

4.6.1 Constructor & Destructor Documentation

4.6.1.1 ImageLayer (osg::Image * image = 0)

4.6.1.2 ImageLayer (const ImageLayer & imageLayer, const osg::CopyOp & copyop = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.6.1.3 virtual ~ImageLayer () [inline, protected, virtual]

4.6.2 Member Function Documentation

4.6.2.1 void dirty () [virtual]

increment the modified count. "

Reimplemented from **Layer** (p. 28).

4.6.2.2 virtual const std::string& getFileName () const [inline, virtual]

Get the file name of the layer.

Reimplemented from **Layer** (p. 28).

4.6.2.3 virtual const osg::Image* getImage () const [inline, virtual]

Return const image associated with layer.

Reimplemented from **Layer** (p. 28).

4.6.2.4 virtual osg::Image* getImage () [inline, virtual]

Return image associated with layer.

Reimplemented from **Layer** (p. 28).

4.6.2.5 unsigned int getModifiedCount () const [virtual]

Get modified count value.

Reimplemented from **Layer** (p. 29).

4.6.2.6 virtual unsigned int getNumColumns () const [inline, virtual]

Reimplemented from **Layer** (p. 29).

4.6.2.7 virtual unsigned int getNumRows () const [inline, virtual]

Reimplemented from **Layer** (p. 29).

4.6.2.8 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec4 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.6.2.9 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec3 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.6.2.10 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec2 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.6.2.11 bool getValue (unsigned int *i*, unsigned int *j*, float & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.6.2.12 META_Object (osgTerrain, ImageLayer)**4.6.2.13 void setFileName (const std::string & *filename*) [inline, virtual]**

Set the file name of the data associated with this layer.

Reimplemented from **Layer** (p. 29).

4.6.2.14 void setImage (osg::Image * *image*)**4.6.2.15 void setModifiedCount (unsigned int) [virtual]**

Set the modified count value.

Reimplemented from **Layer** (p. 30).

4.6.2.16 bool transform (float *offset*, float *scale*) [virtual]

Reimplemented from **Layer** (p. 30).

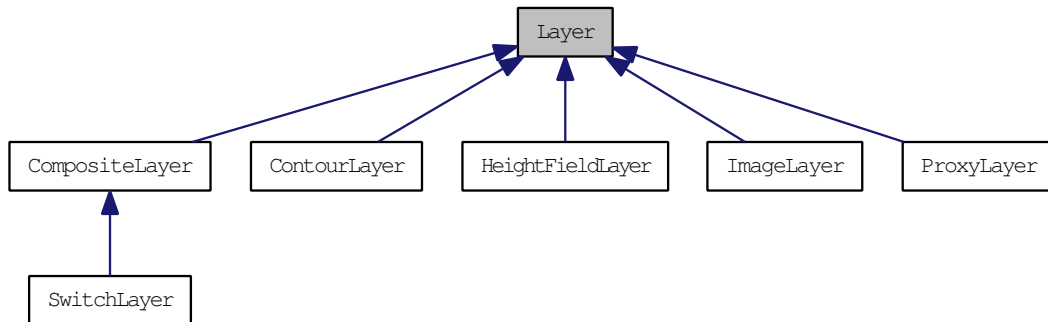
4.6.3 Member Data Documentation**4.6.3.1 osg::ref_ptr<osg::Image> _image [protected]**

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

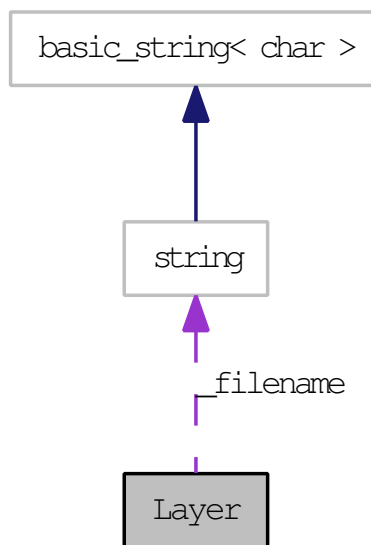
- **Layer**
- **Layer.cpp**

4.7 Layer Class Reference

Inheritance diagram for Layer:



Collaboration diagram for Layer:



Public Member Functions

- **Layer** (const **Layer** &, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
Copy constructor using CopyOp to manage deep vs shallow copy.
- **Layer** ()
- virtual osg::BoundingSphere **computeBound** (bool treatAsElevationLayer) const
- void **computeIndices** (double ndc_x, double ndc_y, unsigned int &i, unsigned int &j, double &ir, double &jr)
- virtual void **dirty** ()
increment the modified count.
- std::string **getCompoundName** () const
Return the compound name of the layer in the form set::name::filename string.
- const osg::Vec4 & **getDefaultValue** () const
- virtual const std::string & **getFileName** () const
Get the file name of the layer.
- virtual const osg::Image * **getImage** () const
Return const image associated with layer if supported.

- virtual osg::Image * **getImage** ()
Return image associated with layer if supported.
- bool **getInterpolatedValue** (double ndc_x, double ndc_y, float &value)
- const Locator * **getLocator** () const
- Locator * **getLocator** ()
- osg::Texture::FilterMode **getMagFilter** () const
Get the magnification texture filter to use when do texture associated with this layer.
- unsigned int **getMaxLevel** () const
- osg::Texture::FilterMode **getMinFilter** () const
Get the minification texture filter to use when do texture associated with this layer.
- unsigned int **getMinLevel** () const
- virtual unsigned int **getModifiedCount** () const
Get modified count value.
- virtual unsigned int **getNumColumns** () const
- virtual unsigned int **getNumRows** () const
- const std::string & **getSetName** () const
- const ValidDataOperator * **getValidDataOperator** () const
- ValidDataOperator * **getValidDataOperator** ()
- bool **getValidValue** (unsigned int i, unsigned int j, osg::Vec4 &value) const
- bool **getValidValue** (unsigned int i, unsigned int j, osg::Vec3 &value) const
- bool **getValidValue** (unsigned int i, unsigned int j, osg::Vec2 &value) const
- bool **getValidValue** (unsigned int i, unsigned int j, float &value) const
- virtual bool **getValue** (unsigned int, unsigned int, osg::Vec4 &) const
- virtual bool **getValue** (unsigned int, unsigned int, osg::Vec3 &) const
- virtual bool **getValue** (unsigned int, unsigned int, osg::Vec2 &) const
- virtual bool **getValue** (unsigned int, unsigned int, float &) const
- META_Object (osgTerrain, Layer)
- void **setDefaultValue** (const osg::Vec4 &value)
- virtual void **setFileName** (const std::string &filename)
Set the file name of the data associated with this layer.
- void **setLocator** (Locator *locator)
- void **setMagFilter** (osg::Texture::FilterMode filter)
Set the magnification texture filter to use when do texture associated with this layer.
- void **setMaxLevel** (unsigned int maxLevel)
- void **setMinFilter** (osg::Texture::FilterMode filter)
Set the minification texture filter to use when do texture associated with this layer.
- void **setMinLevel** (unsigned int minLevel)
- virtual void **setModifiedCount** (unsigned int)
Set the modified count value.
- void **setSetName** (const std::string &setname)
- void **setValidDataOperator** (ValidDataOperator *validDataOp)
- virtual bool **transform** (float, float)

Protected Member Functions

- virtual ~Layer ()

Protected Attributes

- osg::Vec4 **_defaultValue**
- std::string **_filename**
- osg::ref_ptr< **Locator** > **_locator**
- osg::Texture::FilterMode **_magFilter**
- unsigned int **_maxLevel**
- osg::Texture::FilterMode **_minFilter**
- unsigned int **_minLevel**
- osg::ref_ptr< **ValidDataOperator** > **_validDataOperator**

4.7.1 Constructor & Destructor Documentation

4.7.1.1 Layer ()

4.7.1.2 Layer (const Layer & layer, const osg::CopyOp & copyop = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.7.1.3 ~Layer () [protected, virtual]

4.7.2 Member Function Documentation

4.7.2.1 osg::BoundingSphere computeBound (bool treatAsElevationLayer) const [virtual]

Reimplemented in **ProxyLayer** (p. 37).

4.7.2.2 void computeIndices (double ndc_x, double ndc_y, unsigned int & i, unsigned int & j, double & ir, double & jr) [inline]

4.7.2.3 virtual void dirty () [inline, virtual]

increment the modified count. "

Reimplemented in **ImageLayer** (p. 24), **ContourLayer** (p. 15), **HeightFieldLayer** (p. 21), and **ProxyLayer** (p. 38).

4.7.2.4 std::string getCompoundName () const [inline]

Return the compound name of the layer in the form set::name::filename string.

4.7.2.5 const osg::Vec4& getDefaultValue () const [inline]

4.7.2.6 virtual const std::string& getFileName () const [inline, virtual]

Get the file name of the layer.

Reimplemented in **ImageLayer** (p. 24), **HeightFieldLayer** (p. 21), and **ProxyLayer** (p. 38).

4.7.2.7 virtual const osg::Image* getImage () const [inline, virtual]

Return const image associated with layer if supported.

Reimplemented in **ImageLayer** (p. 24), **ContourLayer** (p. 15), **ProxyLayer** (p. 38), and **SwitchLayer** (p. 41).

4.7.2.8 virtual osg::Image* getImage () [inline, virtual]

Return image associated with layer if supported.

Reimplemented in **ImageLayer** (p. 25), **ContourLayer** (p. 15), **ProxyLayer** (p. 38), and **SwitchLayer** (p. 41).

4.7.2.9 bool getInterpolatedValue (double ndc_x, double ndc_y, float & value) [inline]

4.7.2.10 const Locator* getLocator () const [inline]

4.7.2.11 Locator* getLocator () [inline]

4.7.2.12 osg::Texture::FilterMode getMagFilter () const [inline]

Get the magnification texture filter to use when do texture associated with this layer.

4.7.2.13 `unsigned int getMaxLevel () const [inline]`

4.7.2.14 `osg::Texture::FilterMode getMinFilter () const [inline]`

Get the minification texture filter to use when do texture associated with this layer.

4.7.2.15 `unsigned int getMinLevel () const [inline]`

4.7.2.16 `virtual unsigned int getModifiedCount () const [inline, virtual]`

Get modified count value.

Reimplemented in **ImageLayer** (p. 25), **ContourLayer** (p. 15), **HeightFieldLayer** (p. 21), and **ProxyLayer** (p. 38).

4.7.2.17 `virtual unsigned int getNumColumns () const [inline, virtual]`

Reimplemented in **ImageLayer** (p. 25), **ContourLayer** (p. 15), **HeightFieldLayer** (p. 21), and **ProxyLayer** (p. 38).

4.7.2.18 `virtual unsigned int getNumRows () const [inline, virtual]`

Reimplemented in **ImageLayer** (p. 25), **ContourLayer** (p. 16), **HeightFieldLayer** (p. 22), and **ProxyLayer** (p. 38).

4.7.2.19 `const std::string& getSetName () const [inline]`

Reimplemented in **CompositeLayer** (p. 11).

4.7.2.20 `const ValidDataOperator* getValidDataOperator () const [inline]`

4.7.2.21 `ValidDataOperator* getValidDataOperator () [inline]`

4.7.2.22 `bool getValidValue (unsigned int i, unsigned int j, osg::Vec4 & value) const [inline]`

4.7.2.23 `bool getValidValue (unsigned int i, unsigned int j, osg::Vec3 & value) const [inline]`

4.7.2.24 `bool getValidValue (unsigned int i, unsigned int j, osg::Vec2 & value) const [inline]`

4.7.2.25 `bool getValidValue (unsigned int i, unsigned int j, float & value) const [inline]`

4.7.2.26 `virtual bool getValue (unsigned int, unsigned int, osg::Vec4 &) const [inline, virtual]`

Reimplemented in **ImageLayer** (p. 25), **ContourLayer** (p. 16), **HeightFieldLayer** (p. 22), and **ProxyLayer** (p. 38).

4.7.2.27 `virtual bool getValue (unsigned int, unsigned int, osg::Vec3 &) const [inline, virtual]`

Reimplemented in **ImageLayer** (p. 25), **ContourLayer** (p. 16), **HeightFieldLayer** (p. 22), and **ProxyLayer** (p. 38).

4.7.2.28 `virtual bool getValue (unsigned int, unsigned int, osg::Vec2 &) const [inline, virtual]`

Reimplemented in **ImageLayer** (p. 25), **ContourLayer** (p. 16), **HeightFieldLayer** (p. 22), and **ProxyLayer** (p. 38).

4.7.2.29 `virtual bool getValue (unsigned int, unsigned int, float &) const [inline, virtual]`

Reimplemented in **ImageLayer** (p. 25), **ContourLayer** (p. 16), **HeightFieldLayer** (p. 22), and **ProxyLayer** (p. 38).

4.7.2.30 `META_Object (osgTerrain, Layer)`

4.7.2.31 `void setDefaultValue (const osg::Vec4 & value) [inline]`

4.7.2.32 `virtual void setFileName (const std::string & filename) [inline, virtual]`

Set the file name of the data associated with this layer.

Reimplemented in **ImageLayer** (p. 25), **HeightFieldLayer** (p. 22), and **ProxyLayer** (p. 38).

4.7.2.33 void setLocator (Locator * *locator*) [inline]

4.7.2.34 void setMagFilter (osg::Texture::FilterMode *filter*) [inline]

Set the magnification texture filter to use when do texture associated with this layer.

4.7.2.35 void setMaxLevel (unsigned int *maxLevel*) [inline]

4.7.2.36 void setMinFilter (osg::Texture::FilterMode *filter*) [inline]

Set the minification texture filter to use when do texture associated with this layer.

4.7.2.37 void setMinLevel (unsigned int *minLevel*) [inline]

4.7.2.38 virtual void setModifiedCount (unsigned int) [inline, virtual]

Set the modified count value.

Reimplemented in **ImageLayer** (p. 25), **ContourLayer** (p. 16), **HeightFieldLayer** (p. 22), and **ProxyLayer** (p. 39).

4.7.2.39 void setSetName (const std::string & *setname*) [inline]

Reimplemented in **CompositeLayer** (p. 12).

4.7.2.40 void setValidDataOperator (ValidDataOperator * *validDataOp*) [inline]

4.7.2.41 virtual bool transform (float, float) [inline, virtual]

Reimplemented in **ImageLayer** (p. 25), **ContourLayer** (p. 16), **HeightFieldLayer** (p. 22), and **ProxyLayer** (p. 39).

4.7.3 Member Data Documentation

4.7.3.1 osg::Vec4 _defaultValue [protected]

4.7.3.2 std::string _filename [protected]

4.7.3.3 osg::ref_ptr<Locator> _locator [protected]

4.7.3.4 osg::Texture::FilterMode _magFilter [protected]

4.7.3.5 unsigned int _maxLevel [protected]

4.7.3.6 osg::Texture::FilterMode _minFilter [protected]

4.7.3.7 unsigned int _minLevel [protected]

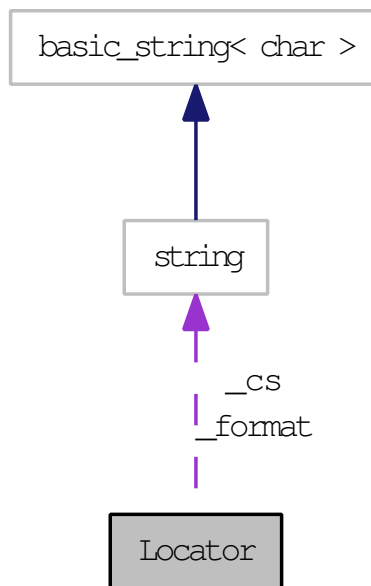
4.7.3.8 osg::ref_ptr<ValidDataOperator> _validDataOperator [protected]

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

- Layer
- Layer.cpp

4.8 Locator Class Reference

Collaboration diagram for Locator:



Public Types

- enum **CoordinateSystemType** { **GEOCENTRIC**, **GEOGRAPHIC**, **PROJECTED** }
CoordinateSystemType provides the classification of the type coordinate system represented.

Public Member Functions

- **Locator** (const **Locator** &, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
Copy constructor using CopyOp to manage deep vs shallow copy.
- **Locator** ()
- bool **computeLocalBounds** (**Locator** &source, osg::Vec3d &bottomLeft, osg::Vec3d &topRight) const
- virtual bool **convertLocalToModel** (const osg::Vec3d &local, osg::Vec3d &world) const
- virtual bool **convertModelToLocal** (const osg::Vec3d &world, osg::Vec3d &local) const
- const std::string & **getCoordinateSystem** () const
Get the CoordinateSystem reference string.
- **CoordinateSystemType** **getCoordinateSystemType** () const
Get the CoordinatesSystemType.
- bool **getDefinedInFile** () const
- const osg::EllipsoidModel * **getEllipsoidModel** () const
Get the const EllipsoidModel.
- osg::EllipsoidModel * **getEllipsoidModel** ()
Get the EllipsoidModel.
- const std::string & **getFormat** () const
Get the coordinate system format string.
- const osg::Matrixd & **getTransform** () const
Set the transformation from local coordinates to model coordinates.

- bool **getTransformScaledByResolution** () const
- **META_Object** (osgTerrain, **Locator**)
- virtual bool **orientationOpenGL** () const
- void **setCoordinateSystem** (const std::string &cs)
Set the CoordinateSystem reference string, should be stored in a form consistent with the Format.
- void **setCoordinateSystemType** (**CoordinateSystemType** type)
Set the CoordinatesSystemType.
- void **setDefinedInFile** (bool flag)
- void **setEllipsoidModel** (osg::EllipsoidModel *ellipsoid)
Set EllipsoidModel to describe the model used to map lat, long and height into geocentric XYZ and back.
- void **setFormat** (const std::string &format)
Set the coordinate system format string.
- void **setTransform** (const osg::Matrixd &transform)
Set the transformation from local coordinates to model coordinates.
- void **setTransformAsExtents** (double minX, double minY, double maxX, double maxY)
Set the extents of the local coords.
- void **setTransformScaledByResolution** (bool scaledByResolution)

Static Public Member Functions

- static bool **convertLocalCoordBetween** (const **Locator** &source, const osg::Vec3d &sourceNDC, const **Locator** &destination, osg::Vec3d &destinationNDC)

Protected Member Functions

- virtual ~**Locator** ()

Protected Attributes

- **CoordinateSystemType _coordinateSystemType**
- std::string **_cs**
- bool **_definedInFile**
- osg::ref_ptr< osg::EllipsoidModel > **_ellipsoidModel**
- std::string **_format**
- osg::Matrixd **_inverse**
- osg::Matrixd **_transform**
- bool **_transformScaledByResolution**

4.8.1 Member Enumeration Documentation

4.8.1.1 enum CoordinateSystemType

CoordinateSystemType provides the classification of the type coordinate system represented.

Enumerator:

GEOCENTRIC GEOCENTRIC coordinate systems are ones mapped to the around the ellipsoid, i.e. whole earth.

GEOGRAPHIC GEOGRAPHIC coordinate systems are ones mapped to latitude and longitude.

PROJECTED PROJECTED coordinate systems are ones projected to a local projected coordinate system i.e. UTM.

4.8.2 Constructor & Destructor Documentation

4.8.2.1 Locator ()

4.8.2.2 Locator (const Locator & *locator*, const osg::CopyOp & *copyop* = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.8.2.3 ~Locator () [protected, virtual]

4.8.3 Member Function Documentation

4.8.3.1 bool computeLocalBounds (Locator & *source*, osg::Vec3d & *bottomLeft*, osg::Vec3d & *topRight*) const

4.8.3.2 static bool convertLocalCoordBetween (const Locator & *source*, const osg::Vec3d & *sourceNDC*, const Locator & *destination*, osg::Vec3d & *destinationNDC*) [inline, static]

4.8.3.3 bool convertLocalToModel (const osg::Vec3d & *local*, osg::Vec3d & *world*) const [virtual]

4.8.3.4 bool convertModelToLocal (const osg::Vec3d & *world*, osg::Vec3d & *local*) const [virtual]

4.8.3.5 const std::string& getCoordinateSystem () const [inline]

Get the CoordinateSystem reference string.

4.8.3.6 CoordinateSystemType getCoordinateSystemType () const [inline]

Get the CoordinatesSystemType.

4.8.3.7 bool getDefinedInFile () const [inline]

4.8.3.8 const osg::EllipsoidModel* getEllipsoidModel () const [inline]

Get the const EllipsoidModel.

4.8.3.9 osg::EllipsoidModel* getEllipsoidModel () [inline]

Get the EllipsoidModel.

4.8.3.10 const std::string& getFormat () const [inline]

Get the coordinate system format string.

4.8.3.11 const osg::Matrixd& getTransform () const [inline]

Set the transformation from local coordinates to model coordinates.

4.8.3.12 bool getTransformScaledByResolution () const [inline]

4.8.3.13 META_Object (osgTerrain, Locator)

4.8.3.14 bool orientationOpenGL () const [virtual]

4.8.3.15 void setCoordinateSystem (const std::string & *cs*) [inline]

Set the CoordinateSystem reference string, should be stored in a form consistent with the Format.

4.8.3.16 void setCoordinateSystemType (CoordinateSystemType *type*) [inline]

Set the CoordinatesSystemType. Note, the user must keep the CoordinateSystemString consistent with the type of the CoordinateSystem.

4.8.3.17 void setDefinedInFile (bool *flag*) [inline]

4.8.3.18 void setEllipsoidModel (osg::EllipsoidModel * *ellipsode*) [inline]

Set EllipsoidModel to describe the model used to map lat, long and height into geocentric XYZ and back.

4.8.3.19 void setFormat (const std::string & *format*) [inline]

Set the coordinate system format string. Typical values would be WKT, PROJ4, USGS etc.

4.8.3.20 void setTransform (const osg::Matrixd & *transform*) [inline]

Set the transformation from local coordinates to model coordinates.

4.8.3.21 void setTransformAsExtents (double *minX*, double *minY*, double *maxX*, double *maxY*)

Set the extents of the local coords.

4.8.3.22 void setTransformScaledByResolution (bool *scaledByResolution*) [inline]

4.8.4 Member Data Documentation

4.8.4.1 CoordinateSystemType _coordinateSystemType [protected]

4.8.4.2 std::string _cs [protected]

4.8.4.3 bool _definedInFile [protected]

4.8.4.4 osg::ref_ptr<osg::EllipsoidModel> _ellipsoidModel [protected]

4.8.4.5 std::string _format [protected]

4.8.4.6 osg::Matrixd _inverse [protected]

4.8.4.7 osg::Matrixd _transform [protected]

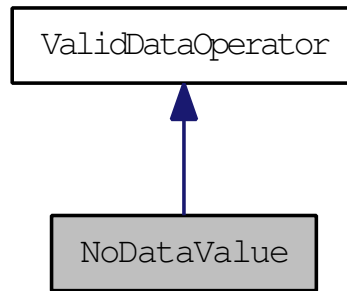
4.8.4.8 bool _transformScaledByResolution [protected]

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

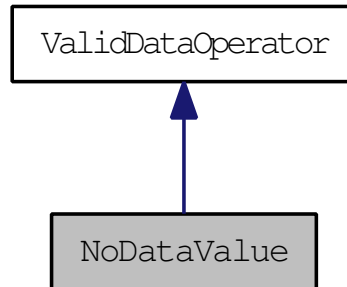
- **Locator**
- **Locator.cpp**

4.9 NoDataValue Struct Reference

Inheritance diagram for NoDataValue:



Collaboration diagram for NoDataValue:



Public Member Functions

- **NoDataValue** (float value)
- float **getValue** () const
- virtual bool **operator()** (float value) const
- void **setNoDataValue** (float value)

Public Attributes

- float **_value**

4.9.1 Constructor & Destructor Documentation

4.9.1.1 **NoDataValue** (float *value*) [inline]

4.9.2 Member Function Documentation

4.9.2.1 float **getValue** () const [inline]

4.9.2.2 virtual bool **operator()** (float *value*) const [inline, virtual]

Reimplemented from **ValidDataOperator** (p. 57).

4.9.2.3 void **setNoDataValue** (float *value*) [inline]

4.9.3 Member Data Documentation

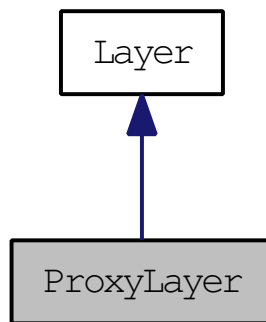
4.9.3.1 float **_value**

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

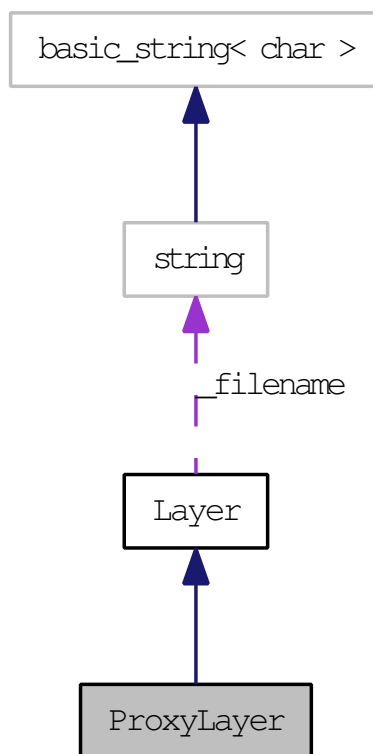
- **ValidDataOperator**

4.10 ProxyLayer Class Reference

Inheritance diagram for ProxyLayer:



Collaboration diagram for ProxyLayer:



Public Member Functions

- **ProxyLayer** (const **ProxyLayer** &proxyLayer, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
 - Copy constructor using CopyOp to manage deep vs shallow copy.*
- **ProxyLayer** ()
- virtual osg::BoundingSphere **computeBound** (bool treatAsElevationLayer) const
- virtual void **dirty** ()
 - increment the modified count.*
- virtual const std::string & **getFileName** () const
 - Get the file name of the layer.*

- virtual const osg::Image * **getImage** () const
Return const image associated with layer if supported.
- virtual osg::Image * **getImage** ()
Return image associated with layer if supported.
- const Layer * **getImplementation** () const
Get the const implementation layer that does the actual work.
- Layer * **getImplementation** ()
Get the implementation layer that does the actual work.
- virtual unsigned int **getModifiedCount** () const
Get modified count value.
- virtual unsigned int **getNumColumns** () const
- virtual unsigned int **getNumRows** () const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec4 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec3 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, osg::Vec2 &value) const
- virtual bool **getValue** (unsigned int i, unsigned int j, float &value) const
- META_Object (osgTerrain, ProxyLayer)
- virtual void **setFileName** (const std::string &filename)
Set the file name of the data associated with this layer.
- void **setImplementation** (Layer *layer)
Set the implementation layer that does the actual work.
- virtual void **setModifiedCount** (unsigned int value)
Set the modified count value.
- virtual bool **transform** (float offset, float scale)

Protected Member Functions

- virtual ~ProxyLayer ()

Protected Attributes

- osg::ref_ptr< Layer > **_implementation**

4.10.1 Constructor & Destructor Documentation

4.10.1.1 ProxyLayer ()

4.10.1.2 ProxyLayer (const ProxyLayer & proxyLayer, const osg::CopyOp & copyop = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.10.1.3 ~ProxyLayer () [protected, virtual]

4.10.2 Member Function Documentation

4.10.2.1 osg::BoundingSphere computeBound (bool treatAsElevationLayer) const [virtual]

Reimplemented from Layer (p. 28).

4.10.2.2 void dirty () [virtual]

increment the modified count. "

Reimplemented from **Layer** (p. 28).

4.10.2.3 virtual const std::string& getFileName () const [inline, virtual]

Get the file name of the layer.

Reimplemented from **Layer** (p. 28).

4.10.2.4 virtual const osg::Image* getImage () const [inline, virtual]

Return const image associated with layer if supported.

Reimplemented from **Layer** (p. 28).

4.10.2.5 virtual osg::Image* getImage () [inline, virtual]

Return image associated with layer if supported.

Reimplemented from **Layer** (p. 28).

4.10.2.6 const Layer* getImplementation () const [inline]

Get the const implementation layer that does the actual work.

4.10.2.7 Layer* getImplementation () [inline]

Get the implementation layer that does the actual work.

4.10.2.8 unsigned int getModifiedCount () const [virtual]

Get modified count value.

Reimplemented from **Layer** (p. 29).

4.10.2.9 unsigned int getNumColumns () const [virtual]

Reimplemented from **Layer** (p. 29).

4.10.2.10 unsigned int getNumRows () const [virtual]

Reimplemented from **Layer** (p. 29).

4.10.2.11 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec4 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.10.2.12 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec3 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.10.2.13 bool getValue (unsigned int *i*, unsigned int *j*, osg::Vec2 & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.10.2.14 bool getValue (unsigned int *i*, unsigned int *j*, float & *value*) const [virtual]

Reimplemented from **Layer** (p. 29).

4.10.2.15 META_Object (osgTerrain, ProxyLayer)**4.10.2.16 void setFileName (const std::string & *filename*) [virtual]**

Set the file name of the data associated with this layer.

Reimplemented from **Layer** (p. 29).

4.10.2.17 void setImplementation (Layer * *layer*) [inline]

Set the implementation layer that does the actual work.

4.10.2.18 void setModifiedCount (unsigned int) [virtual]

Set the modified count value.

Reimplemented from **Layer** (p. 30).

4.10.2.19 bool transform (float *offset*, float *scale*) [virtual]

Reimplemented from **Layer** (p. 30).

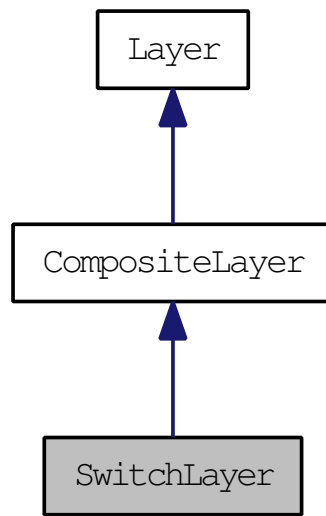
4.10.3 Member Data Documentation**4.10.3.1 osg::ref_ptr<Layer> _implementation [protected]**

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

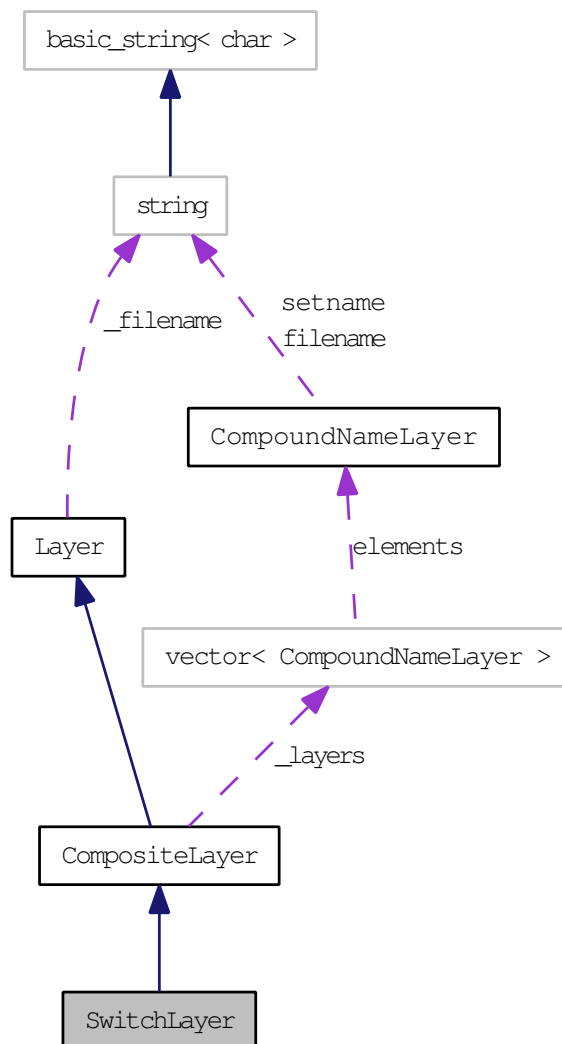
- **Layer**
- **Layer.cpp**

4.11 SwitchLayer Class Reference

Inheritance diagram for SwitchLayer:



Collaboration diagram for SwitchLayer:



Public Member Functions

- **SwitchLayer** (const **SwitchLayer** &switchLayer, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
 - Copy constructor using CopyOp to manage deep vs shallow copy.*
- **SwitchLayer** ()
- int **getActiveLayer** () const
- virtual const osg::Image * **getImage** () const
 - Return const image associated with layer if supported.*
- virtual osg::Image * **getImage** ()
 - Return image associated with layer if supported.*
- **META_Object** (osgTerrain, **SwitchLayer**)
- void **setActiveLayer** (int i)

Protected Member Functions

- virtual ~**SwitchLayer** ()

Protected Attributes

- int **_activeLayer**

4.11.1 Constructor & Destructor Documentation**4.11.1.1 SwitchLayer ()****4.11.1.2 SwitchLayer (const SwitchLayer & switchLayer, const osg::CopyOp & copyop = osg::CopyOp::SHALLOW_COPY)**

Copy constructor using CopyOp to manage deep vs shallow copy.

4.11.1.3 virtual ~SwitchLayer () [inline, protected, virtual]**4.11.2 Member Function Documentation****4.11.2.1 int getActiveLayer () const [inline]****4.11.2.2 virtual const osg::Image* getImage () const [inline, virtual]**

Return const image associated with layer if supported.

Reimplemented from **Layer** (p. 28).

4.11.2.3 virtual osg::Image* getImage () [inline, virtual]

Return image associated with layer if supported.

Reimplemented from **Layer** (p. 28).

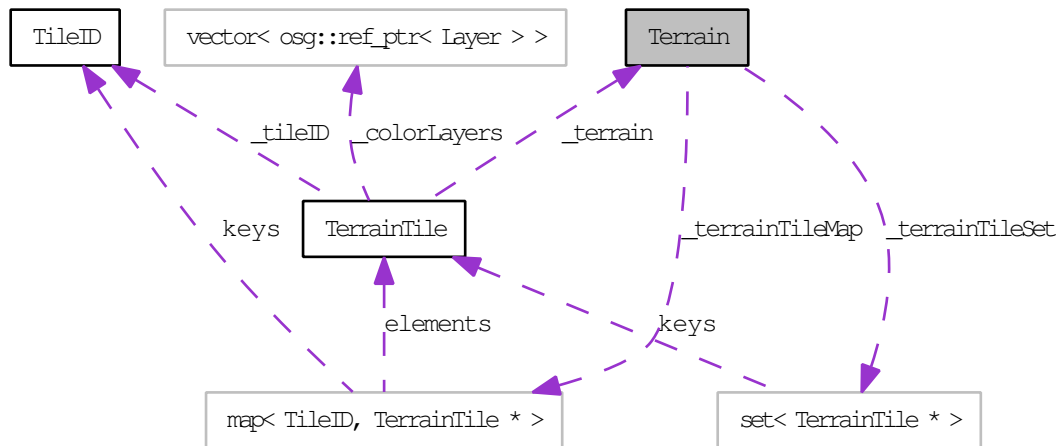
4.11.2.4 META_Object (osgTerrain, SwitchLayer)**4.11.2.5 void setActiveLayer (int i) [inline]****4.11.3 Member Data Documentation****4.11.3.1 int _activeLayer [protected]**

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

- **Layer**
- **Layer.cpp**

4.12 Terrain Class Reference

Terrain (p. 42) provides a framework for loosely coupling height field data with height rendering algorithms. Collaboration diagram for Terrain:



Public Member Functions

- **Terrain** (const **Terrain** &, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
Copy constructor using CopyOp to manage deep vs shallow copy.
- **Terrain** ()
- float **getSampleRatio** () const
Get the sample ratio hint.
- const **TerrainTechnique** * **getTerrainTechniquePrototype** () const
*Get the const **TerrainTechnique** (p. 45) prototype.*
- **TerrainTechnique** * **getTerrainTechniquePrototype** ()
*Get the **TerrainTechnique** (p. 45) prototype.*
- const **TerrainTile** * **getTile** (const **TileID** &tileID) const
*Get the const **TerrainTile** (p. 48) for a given **TileID** (p. 54).*
- **TerrainTile** * **getTile** (const **TileID** &tileID)
*Get the **TerrainTile** (p. 48) for a given **TileID** (p. 54).*
- float **getVerticalScale** () const
Get the vertical scale hint.
- **META_Node** (osgTerrain, **Terrain**)
- void **setSampleRatio** (float ratio)
*Set the sample ratio hint that **TerrainTile** (p. 48) should use when building geometry.*
- void **setTerrainTechniquePrototype** (**TerrainTechnique** *technique)
*Set the **TerrainTechnique** (p. 45) prototype from which TerrainTiles can clone the techniques from.*
- void **setVerticalScale** (float scale)
Set the vertical scale hint.
- virtual void **traverse** (osg::NodeVisitor &nv)

Protected Types

- typedef std::map< **TileID**, **TerrainTile** * > **TerrainTileMap**
- typedef std::set< **TerrainTile** * > **TerrainTileSet**

Protected Member Functions

- virtual **~Terrain** ()
- void **dirtyRegisteredTiles** ()
- void **registerTerrainTile** (**TerrainTile** *tile)
- void **unregisterTerrainTile** (**TerrainTile** *tile)

Protected Attributes

- OpenThreads::Mutex **_mutex**
- float **_sampleRatio**
- osg::ref_ptr< **TerrainTechnique** > **_terrainTechnique**
- **TerrainTileMap** **_terrainTileMap**
- **TerrainTileSet** **_terrainTileSet**
- float **_verticalScale**

Friends

- class **TerrainTile**

4.12.1 Detailed Description

Terrain (p. 42) provides a framework for loosely coupling height field data with height rendering algorithms. This allows **TerrainTechniques** to be plugged in at runtime.

4.12.2 Member Typedef Documentation

4.12.2.1 typedef std::map< **TileID**, **TerrainTile*** > **TerrainTileMap** [protected]

4.12.2.2 typedef std::set< **TerrainTile*** > **TerrainTileSet** [protected]

4.12.3 Constructor & Destructor Documentation

4.12.3.1 **Terrain** ()

4.12.3.2 **Terrain** (const **Terrain** & *ts*, const osg::CopyOp & *copyop* = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.12.3.3 **~Terrain** () [protected, virtual]

4.12.4 Member Function Documentation

4.12.4.1 void **dirtyRegisteredTiles** () [protected]

4.12.4.2 float **getSampleRatio** () const [inline]

Get the sample ratio hint.

4.12.4.3 const **TerrainTechnique*** **getTerrainTechniquePrototype** () const [inline]

Get the const **TerrainTechnique** (p. 45) prototype.

4.12.4.4 **TerrainTechnique*** **getTerrainTechniquePrototype** () [inline]

Get the **TerrainTechnique** (p. 45) prototype.

4.12.4.5 const **TerrainTile** * **getTile** (const **TileID** & *tileID*) const

Get the const **TerrainTile** (p. 48) for a given **TileID** (p. 54).

4.12.4.6 TerrainTile * getTile (const TileID & *tileID*)

Get the **TerrainTile** (p. 48) for a given **TileID** (p. 54).

4.12.4.7 float getVerticalScale () const [inline]

Get the vertical scale hint.

4.12.4.8 META_Node (osgTerrain, Terrain)

4.12.4.9 void registerTerrainTile (TerrainTile * *tile*) [protected]

4.12.4.10 void setSampleRatio (float *ratio*) [inline]

Set the sample ratio hint that **TerrainTile** (p. 48) should use when building geometry. Defaults to 1.0, which means use all original sample points.

4.12.4.11 void setTerrainTechniquePrototype (TerrainTechnique * *technique*) [inline]

Set the **TerrainTechnique** (p. 45) prototype from which TerrainTiles can clone the techniques from.

4.12.4.12 void setVerticalScale (float *scale*) [inline]

Set the vertical scale hint.

4.12.4.13 void traverse (osg::NodeVisitor & *nv*) [virtual]

4.12.4.14 void unregisterTerrainTile (TerrainTile * *tile*) [protected]

4.12.5 Friends And Related Function Documentation

4.12.5.1 friend class TerrainTile [friend]

4.12.6 Member Data Documentation

4.12.6.1 OpenThreads::Mutex _mutex [mutable, protected]

4.12.6.2 float _sampleRatio [protected]

4.12.6.3 osg::ref_ptr<TerrainTechnique> _terrainTechnique [protected]

4.12.6.4 TerrainTileMap _terrainTileMap [protected]

4.12.6.5 TerrainTileSet _terrainTileSet [protected]

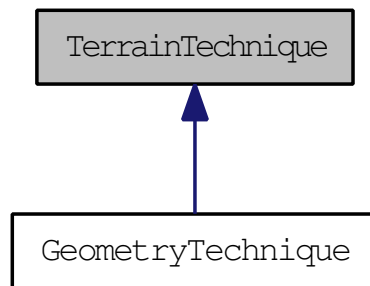
4.12.6.6 float _verticalScale [protected]

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

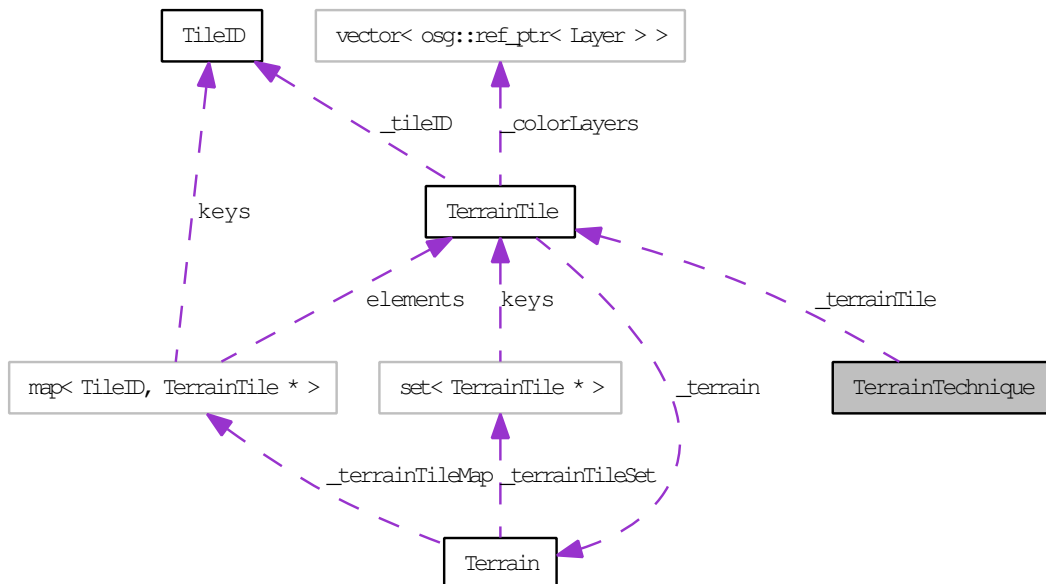
- **Terrain**
- **Terrain.cpp**

4.13 TerrainTechnique Class Reference

Inheritance diagram for TerrainTechnique:



Collaboration diagram for TerrainTechnique:



Public Member Functions

- **TerrainTechnique** (const **TerrainTechnique** &, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
 - Copy constructor using CopyOp to manage deep vs shallow copy.*
- **TerrainTechnique** ()
- virtual void **cleanSceneGraph** ()
 - Clean scene graph from any terrain technique specific nodes.*
- virtual void **cull** (osgUtil::CullVisitor *nv)
- const **TerrainTile** * **getTerrainTile** () const
- **TerrainTile** * **getTerrainTile** ()
- virtual void **init** ()
- **META_Object** (osgTerrain, **TerrainTechnique**)
- virtual void **releaseGLObjets** (osg::State *s=0) const
 - If State is non-zero, this function releases any associated OpenGL objects for the specified graphics context.*
- virtual void **traverse** (osg::NodeVisitor &nv)

Traverse the terrain subgraph.

- virtual void **update** (osgUtil::UpdateVisitor *nv)

Protected Member Functions

- virtual ~**TerrainTechnique** ()
- void **setDirty** (bool dirty)

Protected Attributes

- **TerrainTile** * **_terrainTile**

Friends

- class **osgTerrain::TerrainTile**

4.13.1 Constructor & Destructor Documentation

4.13.1.1 TerrainTechnique ()

4.13.1.2 TerrainTechnique (const TerrainTechnique & *TerrainTechnique*, const osg::CopyOp & *copyop* = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.13.1.3 ~TerrainTechnique () [protected, virtual]

4.13.2 Member Function Documentation

4.13.2.1 void cleanSceneGraph () [virtual]

Clean scene graph from any terrain technique specific nodes.

Reimplemented in **GeometryTechnique** (p. 18).

4.13.2.2 void cull (osgUtil::CullVisitor * *nv*) [virtual]

Reimplemented in **GeometryTechnique** (p. 18).

4.13.2.3 const TerrainTile* getTerrainTile () const [inline]

4.13.2.4 TerrainTile* getTerrainTile () [inline]

4.13.2.5 void init () [virtual]

Reimplemented in **GeometryTechnique** (p. 19).

4.13.2.6 META_Object (osgTerrain, TerrainTechnique)

4.13.2.7 virtual void releaseGLObjects (osg::State * = 0) const [inline, virtual]

If State is non-zero, this function releases any associated OpenGL objects for the specified graphics context. Otherwise, releases OpenGL objects for all graphics contexts.

Reimplemented in **GeometryTechnique** (p. 19).

4.13.2.8 void setDirty (bool *dirty*) [protected]

4.13.2.9 void traverse (osg::NodeVisitor & *nv*) [virtual]

Traverse the terrain subgraph.

Reimplemented in **GeometryTechnique** (p. 19).

4.13.2.10 void update (osgUtil::UpdateVisitor * *nv*) [virtual]

Reimplemented in **GeometryTechnique** (p. 19).

4.13.3 Friends And Related Function Documentation

4.13.3.1 friend class osgTerrain::TerrainTile [friend]

4.13.4 Member Data Documentation

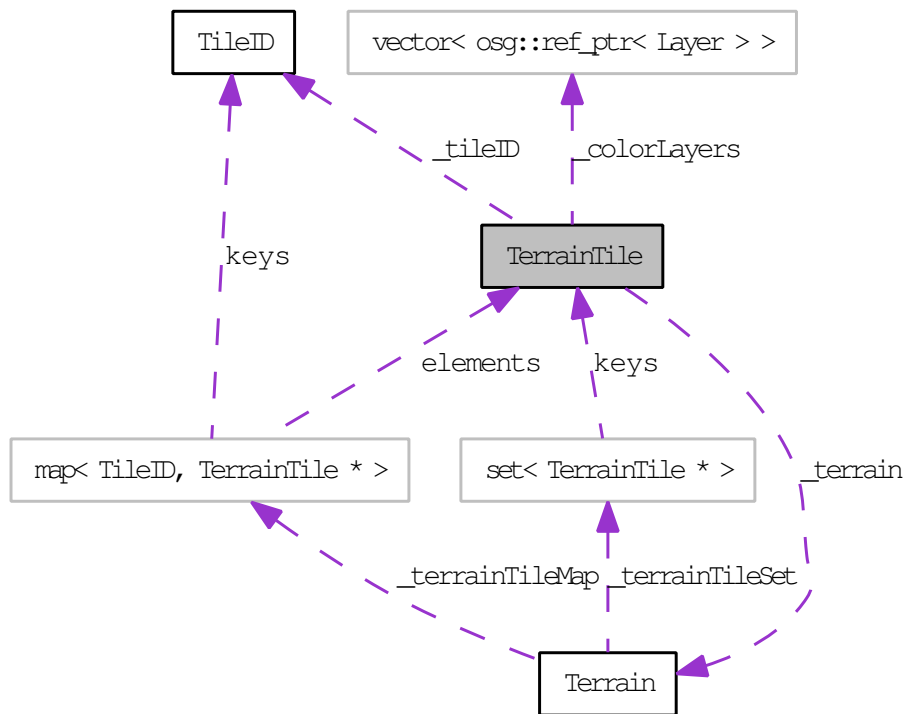
4.13.4.1 TerrainTile*_terrainTile [protected]

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

- TerrainTechnique
- TerrainTechnique.cpp

4.14 TerrainTile Class Reference

Terrain (p. 42) provides a framework for loosely coupling height field data with height rendering algorithms. Collaboration diagram for TerrainTile:



Classes

- struct **TileLoadedCallback**
*Callback for post processing loaded **TerrainTile** (p. 48), and for filling in missing elements such as external external imagery.*

Public Member Functions

- **TerrainTile** (const **TerrainTile** &, const osg::CopyOp ©op=osg::CopyOp::SHALLOW_COPY)
Copy constructor using CopyOp to manage deep vs shallow copy.
- **TerrainTile** ()
- virtual osg::BoundingSphere **computeBound** () const
Compute the bounding volume of the terrain by computing the union of the bounding volumes of all layers.
- const **Layer** * **getColorLayer** (unsigned int i) const
Set const color layer with specified layer number.
- **Layer** * **getColorLayer** (unsigned int i)
Get color layer with specified layer number.
- bool **getDirty** () const
return true if the tile is dirty and needs to be updated,
- const **Layer** * **getElevationLayer** () const
Get the const layer to use to define the elevations of the terrain.
- **Layer** * **getElevationLayer** ()

Get the layer to use to define the elevations of the terrain.

- **const Locator * getLocator () const**
Get the const coordinate frame locator of the terrain node.
- **Locator * getLocator ()**
Get the coordinate frame locator of the terrain node.
- **unsigned int getNumColorLayers () const**
Get the number of colour layers.
- **bool getRequiresNormals () const**
*Get whether the **TerrainTechnique** (p. 45) should create per vertex normals for lighting purposes.*
- **const Terrain * getTerrain () const**
*Get the const **Terrain** (p. 42) that this **Terrain** (p. 42) tile is a member of.*
- **Terrain * getTerrain ()**
*Get the **Terrain** (p. 42) that this **Terrain** (p. 42) tile is a member of.*
- **const TerrainTechnique * getTerrainTechnique () const**
*Get the const **TerrainTechnique** (p. 45).*
- **TerrainTechnique * getTerrainTechnique ()**
*Get the **TerrainTechnique** (p. 45).*
- **const TileID & getTileID () const**
*Get the **TileID** (p. 54) (layer, x,y) of the **TerrainTile** (p. 48).*
- **bool getTreatBoundariesToValidDataAsDefaultValue () const**
*Get whether the **TreatBoundariesToValidDataAsDefaultValue** hint.*
- **void init ()**
*Call init on any attached **TerrainTechnique** (p. 45).*
- **META_Node (osgTerrain, TerrainTile)**
- **virtual void releaseGLObjects (osg::State *s=0) const**
If State is non-zero, this function releases any associated OpenGL objects for the specified graphics context.
- **void setColorLayer (unsigned int i, Layer *layer)**
Set a color layer with specified layer number.
- **void setDirty (bool dirty)**
Set the dirty flag on/off.
- **void setElevationLayer (Layer *layer)**
Set the layer to use to define the elevations of the terrain.
- **void setLocator (Locator *locator)**
Set the coordinate frame locator of the terrain node.
- **void setRequiresNormals (bool flag)**
*Set hint to whether the **TerrainTechnique** (p. 45) should create per vertex normals for lighting purposes.*
- **void setTerrain (Terrain *ts)**
*Set the **Terrain** (p. 42) that this **Terrain** (p. 42) tile is a member of.*

- void **setTerrainTechnique** (**TerrainTechnique** *terrainTechnique)
Set the **TerrainTechnique** (p. 45).
- void **setTileID** (const **TileID** &tileID)
Set the **TileID** (p. 54) (layer, x,y) of the **TerrainTile** (p. 48).
- void **setTreatBoundariesToValidDataAsDefaultValue** (bool flag)
Set the hint to whether the **TerrainTechnique** (p. 45) should treat the invalid **Layer** (p. 26) entries that are neighbours to valid entries with the default value.
- virtual void **traverse** (osg::NodeVisitor &nv)

Static Public Member Functions

- static osg::ref_ptr< **TileLoadedCallback** > & **getTileLoadedCallback** ()
- static void **setTileLoadedCallback** (**TileLoadedCallback** *lc)

Protected Types

- typedef std::vector< osg::ref_ptr< **Layer** > > **Layers**

Protected Member Functions

- virtual ~**TerrainTile** ()

Protected Attributes

- **Layers** _colorLayers
- bool _dirty
- osg::ref_ptr< **Layer** > _elevationLayer
- bool _hasBeenTraversal
- osg::ref_ptr< **Locator** > _locator
- bool _requiresNormals
- **Terrain** * _terrain
- osg::ref_ptr< **TerrainTechnique** > _terrainTechnique
- **TileID** _tileID
- bool _treatBoundariesToValidDataAsDefaultValue

Friends

- class **Terrain**

4.14.1 Detailed Description

Terrain (p. 42) provides a framework for loosely coupling height field data with height rendering algorithms. This allows **TerrainTechnique**'s to be plugged in at runtime.

4.14.2 Member Typedef Documentation

4.14.2.1 typedef std::vector< osg::ref_ptr<**Layer**> > **Layers** [protected]

4.14.3 Constructor & Destructor Documentation

4.14.3.1 **TerrainTile** ()

4.14.3.2 **TerrainTile** (const **TerrainTile** & *terrain*, const osg::CopyOp & *copyop* = osg::CopyOp::SHALLOW_COPY)

Copy constructor using CopyOp to manage deep vs shallow copy.

4.14.3.3 `~TerrainTile () [protected, virtual]`

4.14.4 Member Function Documentation

4.14.4.1 `osg::BoundingSphere computeBound () const [virtual]`

Compute the bounding volume of the terrain by computing the union of the bounding volumes of all layers.

4.14.4.2 `const Layer* getColorLayer (unsigned int l) const [inline]`

Set const color layer with specified layer number.

4.14.4.3 `Layer* getColorLayer (unsigned int l) [inline]`

Get color layer with specified layer number.

4.14.4.4 `bool getDirty () const [inline]`

return true if the tile is dirty and needs to be updated,

4.14.4.5 `const Layer* getElevationLayer () const [inline]`

Get the const layer to use to define the elevations of the terrain.

4.14.4.6 `Layer* getElevationLayer () [inline]`

Get the layer to use to define the elevations of the terrain.

4.14.4.7 `const Locator* getLocator () const [inline]`

Get the const coordinate frame locator of the terrain node.

4.14.4.8 `Locator* getLocator () [inline]`

Get the coordinate frame locator of the terrain node.

4.14.4.9 `unsigned int getNumColorLayers () const [inline]`

Get the number of colour layers.

4.14.4.10 `bool getRequiresNormals () const [inline]`

Get whether the **TerrainTechnique** (p. 45) should create per vertex normals for lighting purposes.

4.14.4.11 `const Terrain* getTerrain () const [inline]`

Get the const **Terrain** (p. 42) that this **Terrain** (p. 42) tile is a member of.

4.14.4.12 `Terrain* getTerrain () [inline]`

Get the **Terrain** (p. 42) that this **Terrain** (p. 42) tile is a member of.

4.14.4.13 `const TerrainTechnique* getTerrainTechnique () const [inline]`

Get the const **TerrainTechnique** (p. 45).

4.14.4.14 `TerrainTechnique* getTerrainTechnique () [inline]`

Get the **TerrainTechnique** (p. 45).

4.14.4.15 `const TileID& getTileID () const [inline]`

Get the **TileID** (p. 54) (layer, x,y) of the **TerrainTile** (p. 48).

4.14.4.16 `osg::ref_ptr< TerrainTile::TileLoadedCallback > & getTileLoadedCallback () [static]`

4.14.4.17 `bool getTreatBoundariesToValidDataAsDefaultValue () const [inline]`

Get whether the **TreatBoundariesToValidDataAsDefaultValue** hint.

4.14.4.18 `void init ()`

Call **init** on any attached **TerrainTechnique** (p. 45).

4.14.4.19 META_Node (osgTerrain, TerrainTile)**4.14.4.20 void releaseGLObjets (osg::State * *state* = 0) const [virtual]**

If State is non-zero, this function releases any associated OpenGL objects for the specified graphics context. Otherwise, releases OpenGL objects for all graphics contexts.

4.14.4.21 void setColorLayer (unsigned int *i*, Layer * *layer*)

Set a color layer with specified layer number.

4.14.4.22 void setDirty (bool *dirty*)

Set the dirty flag on/off.

4.14.4.23 void setElevationLayer (Layer * *layer*)

Set the layer to use to define the elevations of the terrain.

4.14.4.24 void setLocator (Locator * *locator*) [inline]

Set the coordinate frame locator of the terrain node. The locator takes non-dimensional s,t coordinates into the X,Y,Z world coords and back.

4.14.4.25 void setRequiresNormals (bool *flag*) [inline]

Set hint to whether the **TerrainTechnique** (p. 45) should create per vertex normals for lighting purposes.

4.14.4.26 void setTerrain (Terrain * *ts*)

Set the **Terrain** (p. 42) that this **Terrain** (p. 42) tile is a member of.

4.14.4.27 void setTerrainTechnique (TerrainTechnique * *terrainTechnique*)

Set the **TerrainTechnique** (p. 45).

4.14.4.28 void setTileID (const TileID & *tileID*)

Set the **TileID** (p. 54) (layer, x,y) of the **TerrainTile** (p. 48). The **TileID** (p. 54) is used so it can be located by its neighbours via the enclosing **Terrain** (p. 42) node that manages a map of **TileID** (p. 54) to TerrainTiles.

4.14.4.29 void setTileLoadedCallback (TerrainTile::TileLoadedCallback * *lc*) [static]**4.14.4.30 void setTreatBoundariesToValidDataAsDefaultValue (bool *flag*) [inline]**

Set the hint to whether the **TerrainTechnique** (p. 45) should treat the invalid **Layer** (p. 26) entries that are neighbours to valid entries with the default value.

4.14.4.31 void traverse (osg::NodeVisitor & *nv*) [virtual]

4.14.5 Friends And Related Function Documentation

4.14.5.1 friend class Terrain [friend]

4.14.6 Member Data Documentation

4.14.6.1 Layers_colorLayers [protected]

4.14.6.2 bool_dirty [protected]

4.14.6.3 osg::ref_ptr<Layer>_elevationLayer [protected]

4.14.6.4 bool_hasBeenTraversal [protected]

4.14.6.5 osg::ref_ptr<Locator>_locator [protected]

4.14.6.6 bool_requiresNormals [protected]

4.14.6.7 Terrain*_terrain [protected]

4.14.6.8 osg::ref_ptr<TerrainTechnique>_terrainTechnique [protected]

4.14.6.9 TileID_tileID [protected]

4.14.6.10 bool_treatBoundariesToValidDataAsDefaultValue [protected]

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

- TerrainTile
- TerrainTile.cpp

4.15 TileID Class Reference

Public Member Functions

- **TileID** (int *in_level*, int *in_x*, int *in_y*)
- **TileID** ()
- bool **operator!=** (const **TileID** &*rhs*) const
- bool **operator<** (const **TileID** &*rhs*) const
- bool **operator==** (const **TileID** &*rhs*) const
- bool **valid** () const

Public Attributes

- int **level**
- int **x**
- int **y**

4.15.1 Constructor & Destructor Documentation

4.15.1.1 TileID ()

4.15.1.2 TileID (int *in_level*, int *in_x*, int *in_y*)

4.15.2 Member Function Documentation

4.15.2.1 bool operator!= (const TileID & *rhs*) const [inline]

4.15.2.2 bool operator< (const TileID & *rhs*) const [inline]

4.15.2.3 bool operator== (const TileID & *rhs*) const [inline]

4.15.2.4 bool valid () const [inline]

4.15.3 Member Data Documentation

4.15.3.1 int level

4.15.3.2 int x

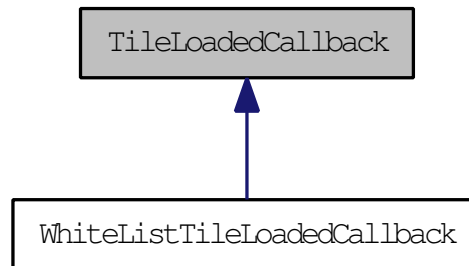
4.15.3.3 int y

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

- **TerrainTile**
- **TerrainTile.cpp**

4.16 TileLoadedCallback Struct Reference

Callback for post processing loaded **TerrainTile** (p. 48), and for filling in missing elements such as external external imagery. Inheritance diagram for TileLoadedCallback:



Public Member Functions

- virtual bool **deferExternalLayerLoading** () const =0
- virtual void **loaded** (osgTerrain::TerrainTile *tile, const osgDB::ReaderWriter::Options *options) const =0

4.16.1 Detailed Description

Callback for post processing loaded **TerrainTile** (p. 48), and for filling in missing elements such as external external imagery.

4.16.2 Member Function Documentation

4.16.2.1 virtual bool deferExternalLayerLoading () const [pure virtual]

Implemented in **WhiteListTileLoadedCallback** (p. 61).

4.16.2.2 virtual void loaded (osgTerrain::TerrainTile * *tile*, const osgDB::ReaderWriter::Options * *options*) const [pure virtual]

Implemented in **WhiteListTileLoadedCallback** (p. 61).

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

- **TerrainTile**

4.17 TransformOperator Struct Reference

Public Member Functions

- **TransformOperator** (float offset, float scale)
- void **operator()** (float &v) const
- void **operator()** (int &v) const
- void **operator()** (short &v) const
- void **operator()** (char &v) const
- void **operator()** (unsigned int &v) const
- void **operator()** (unsigned short &v) const
- void **operator()** (unsigned char &v) const

Public Attributes

- float **_offset**
- float **_scale**

4.17.1 Constructor & Destructor Documentation

4.17.1.1 **TransformOperator** (float *offset*, float *scale*) [`inline`]

4.17.2 Member Function Documentation

4.17.2.1 **void operator()** (float & *v*) const [`inline`]

4.17.2.2 **void operator()** (int & *v*) const [`inline`]

4.17.2.3 **void operator()** (short & *v*) const [`inline`]

4.17.2.4 **void operator()** (char & *v*) const [`inline`]

4.17.2.5 **void operator()** (unsigned int & *v*) const [`inline`]

4.17.2.6 **void operator()** (unsigned short & *v*) const [`inline`]

4.17.2.7 **void operator()** (unsigned char & *v*) const [`inline`]

4.17.3 Member Data Documentation

4.17.3.1 **float _offset**

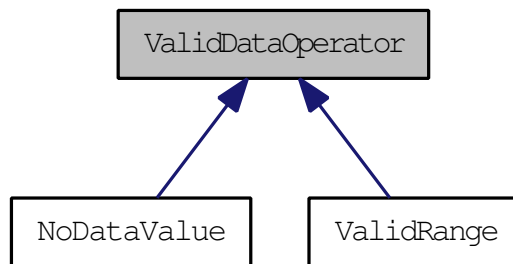
4.17.3.2 **float _scale**

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

- **Layer.cpp**

4.18 ValidDataOperator Struct Reference

Inheritance diagram for ValidDataOperator:



Public Member Functions

- virtual bool **operator()** (const osg::Vec4 &value) const
- virtual bool **operator()** (const osg::Vec3 &value) const
- virtual bool **operator()** (const osg::Vec2 &value) const
- virtual bool **operator()** (float) const

4.18.1 Member Function Documentation

4.18.1.1 virtual bool **operator()** (const osg::Vec4 & *value*) const [inline, virtual]

4.18.1.2 virtual bool **operator()** (const osg::Vec3 & *value*) const [inline, virtual]

4.18.1.3 virtual bool **operator()** (const osg::Vec2 & *value*) const [inline, virtual]

4.18.1.4 virtual bool **operator()** (float) const [inline, virtual]

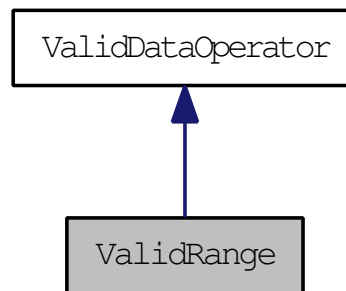
Reimplemented in **ValidRange** (p. 58), and **NoDataValue** (p. 35).

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

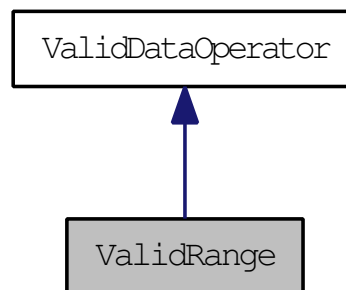
- **ValidDataOperator**

4.19 ValidRange Struct Reference

Inheritance diagram for ValidRange:



Collaboration diagram for ValidRange:



Public Member Functions

- **ValidRange** (float minValue, float maxValue)
- float **getMaxValue** () const
- float **getMinValue** () const
- virtual bool **operator()** (float value) const
- void **setMaxValue** (float maxValue)
- void **setMinValue** (float minValue)
- void **setRange** (float minValue, float maxValue)

Public Attributes

- float **_maxValue**
- float **_minValue**

4.19.1 Constructor & Destructor Documentation

4.19.1.1 **ValidRange** (float *minValue*, float *maxValue*) [inline]

4.19.2 Member Function Documentation

4.19.2.1 float **getMaxValue** () const [inline]

4.19.2.2 float **getMinValue** () const [inline]

4.19.2.3 virtual bool **operator()** (float *value*) const [inline, virtual]

Reimplemented from **ValidDataOperator** (p. 57).

4.19.2.4 void setMaxValue (float *maxValue*) [inline]

4.19.2.5 void setMinValue (float *minValue*) [inline]

4.19.2.6 void setRange (float *minValue*, float *maxValue*) [inline]

4.19.3 Member Data Documentation

4.19.3.1 float *_maxValue*

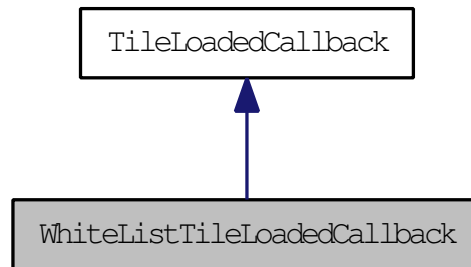
4.19.3.2 float *_minValue*

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

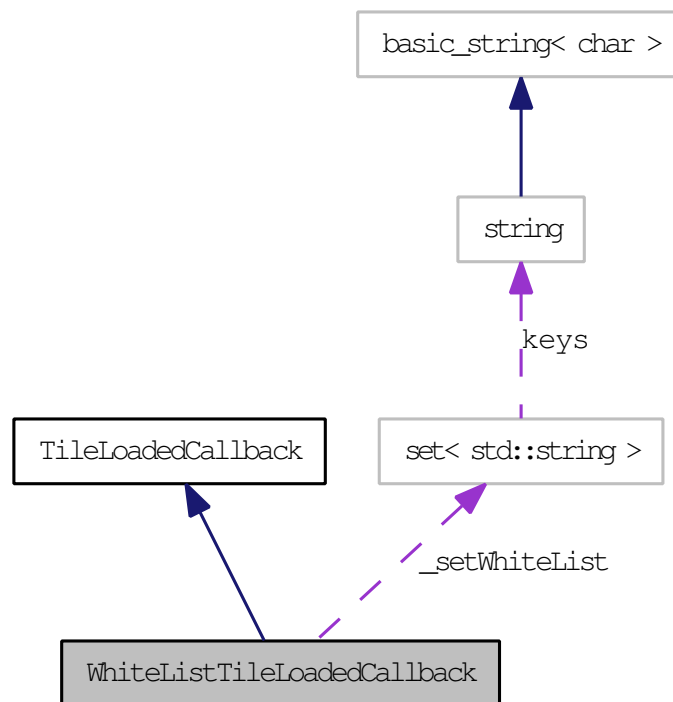
- ValidDataOperator

4.20 WhiteListTileLoadedCallback Class Reference

Helper callback for managing optional sets of layers, that loading of is deferred to this callback, with this callback working out which layers to load, and how to create fallback versions of the layers. Inheritance diagram for WhiteListTileLoadedCallback:



Collaboration diagram for WhiteListTileLoadedCallback:



Public Member Functions

- **WhiteListTileLoadedCallback** ()
- void **allow** (const std::string &setname)
- virtual bool **deferExternalLayerLoading** () const
- bool **getAllowAll** () const
- unsigned int **getMinimumNumOfLayers** () const
- bool **getReplaceSwitchLayer** () const
- bool **layerAcceptable** (const std::string &setname) const
- virtual void **loaded** (osgTerrain::TerrainTile *tile, const osgDB::ReaderWriter::Options *options) const
- bool **readImageLayer** (osgTerrain::ImageLayer *imageLayer, const osgDB::ReaderWriter::Options *options) const
- void **setAllowAll** (bool allowAll)
- void **setMinimumNumOfLayers** (unsigned int numLayers)
- void **setReplaceSwitchLayer** (bool replaceSwitchLayer)

Protected Types

- typedef std::set< std::string > **SetWhiteList**

Protected Member Functions

- virtual ~WhiteListTileLoadedCallback ()

Protected Attributes

- bool **_allowAll**
- unsigned int **_minumumNumberOfLayers**
- bool **_replaceSwitchLayer**
- **SetWhiteList _setWhiteList**

4.20.1 Detailed Description

Helper callback for managing optional sets of layers, that loading of is deferred to this callback, with this callback working out which layers to load, and how to create fallback versions of the layers.

4.20.2 Member Typedef Documentation

4.20.2.1 typedef std::set<std::string> **SetWhiteList** [protected]

4.20.3 Constructor & Destructor Documentation

4.20.3.1 WhiteListTileLoadedCallback ()

4.20.3.2 ~WhiteListTileLoadedCallback () [protected, virtual]

4.20.4 Member Function Documentation

4.20.4.1 void allow (const std::string & *setname*) [inline]

4.20.4.2 bool deferExternalLayerLoading () const [virtual]

Implements **TileLoadedCallback** (p. 55).

4.20.4.3 bool getAllowAll () const [inline]

4.20.4.4 unsigned int getMinimumNumOfLayers () const [inline]

4.20.4.5 bool getReplaceSwitchLayer () const [inline]

4.20.4.6 bool layerAcceptable (const std::string & *setname*) const

4.20.4.7 void loaded (osgTerrain::TerrainTile * *tile*, const osgDB::ReaderWriter::Options * *options*) const [virtual]

Implements **TileLoadedCallback** (p. 55).

4.20.4.8 bool readImageLayer (osgTerrain::ImageLayer * *imageLayer*, const osgDB::ReaderWriter::Options * *options*) const

4.20.4.9 void setAllowAll (bool *allowAll*) [inline]

4.20.4.10 void setMinimumNumOfLayers (unsigned int *numLayers*) [inline]

4.20.4.11 void setReplaceSwitchLayer (bool *replaceSwitchLayer*) [inline]

4.20.5 Member Data Documentation

4.20.5.1 bool **_allowAll** [protected]

4.20.5.2 unsigned int **_minumumNumberOfLayers** [protected]

4.20.5.3 bool **_replaceSwitchLayer** [protected]

4.20.5.4 **SetWhiteList _setWhiteList** [protected]

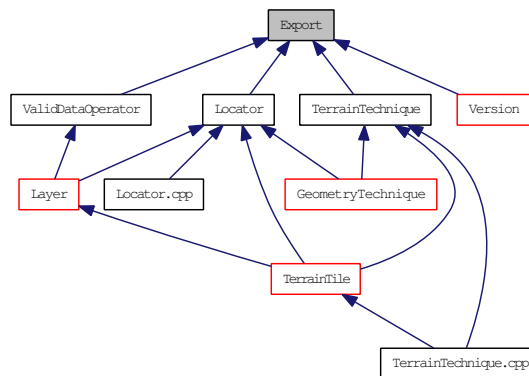
The documentation for this class was generated from the following files:

- [TerrainTile](#)
- [TerrainTile.cpp](#)

File Documentation

5.1 Export File Reference

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



Namespaces

- namespace **osgTerrain**

The *osgTerrain* (p. 7) library is a NodeKit that provides geospecific terrain rendering support.

Defines

- #define **OSGTERRAIN_EXPORT**
- #define **OSGTERRAIN_EXPORT_1**

5.1.1 Define Documentation

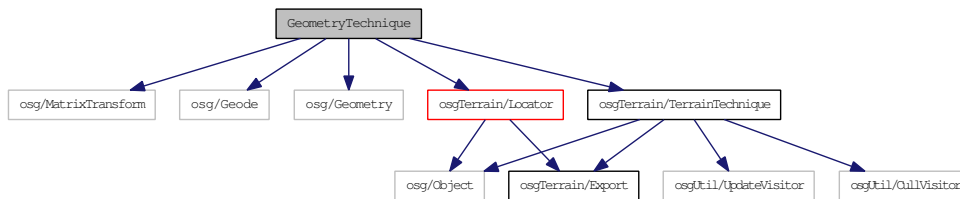
5.1.1.1 #define **OSGTERRAIN_EXPORT**

5.1.1.2 #define **OSGTERRAIN_EXPORT_1**

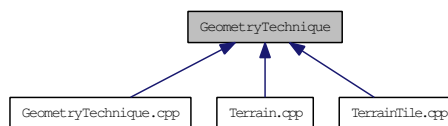
5.2 GeometryTechnique File Reference

```
#include <osg/MatrixTransform>
#include <osg/Geode>
#include <osg/Geometry>
#include <osgTerrain/TerrainTechnique>
#include <osgTerrain/Locator>
```

Include dependency graph for GeometryTechnique:



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



Classes

- struct **BufferData**
- class **GeometryTechnique**

Namespaces

- namespace **osgTerrain**

*The **osgTerrain** (p. 7) library is a NodeKit that provides geospecific terrain rendering support.*

Defines

- #define **OSGTERRAIN_GEOMETRYTECHNIQUE** 1

5.2.1 Define Documentation

5.2.1.1 #define OSGTERRAIN_GEOMETRYTECHNIQUE 1

5.3 GeometryTechnique.cpp File Reference

```
#include <osgTerrain/GeometryTechnique>
#include <osgTerrain/TerrainTile>
#include <osgTerrain/Terrain>
#include <osgUtil/SmoothingVisitor>
#include <osgDB/FileUtils>
#include <osg/io_utils>
#include <osg/Texture2D>
#include <osg/Texture1D>
#include <osg/TexEnvCombine>
#include <osg/Program>
#include <osg/Math>
#include <osg/Timer>
```

Defines

- #define **NEW_COORD_CODE**

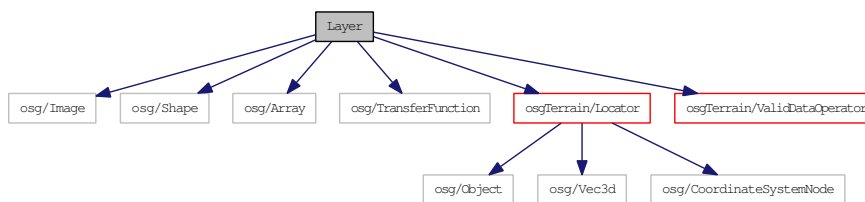
5.3.1 Define Documentation

5.3.1.1 #define NEW_COORD_CODE

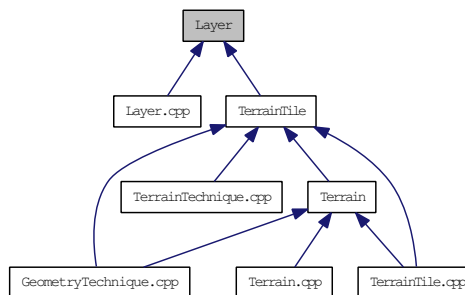
5.4 Layer File Reference

```
#include <osg/Image>
#include <osg/Shape>
#include <osg/Array>
#include <osg/TransferFunction>
#include <osgTerrain/Locator>
#include <osgTerrain/ValidDataOperator>
```

Include dependency graph for Layer:



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



Classes

- class **CompositeLayer**
- struct **CompoundNameLayer**
- class **ContourLayer**
- class **HeightFieldLayer**
- class **ImageLayer**
- class **Layer**
- class **ProxyLayer**
- class **SwitchLayer**

Namespaces

- namespace **osgTerrain**

The *osgTerrain* (p. 7) library is a NodeKit that provides geospecific terrain rendering support.

Defines

- #define **MAXIMUM_NUMBER_OF_LEVELS** 30
- #define **OSGTERRAIN_LAYER** 1

Functions

- OSGTERRAIN_EXPORT std::string **createCompondSetNameAndFileName** (const std::string &setname, const std::string &filename)
Create a compound string in the form set:setname:filename, or just filename if setname is "".
- OSGTERRAIN_EXPORT void **extractSetNameAndFileName** (const std::string &compoundstring, std::string &setname, std::string &filename)
Extact the setname and filename from a compound string in the from set:setname:filename".

5.4.1 Define Documentation

5.4.1.1 **#define** MAXIMUM_NUMBER_OF_LEVELS 30

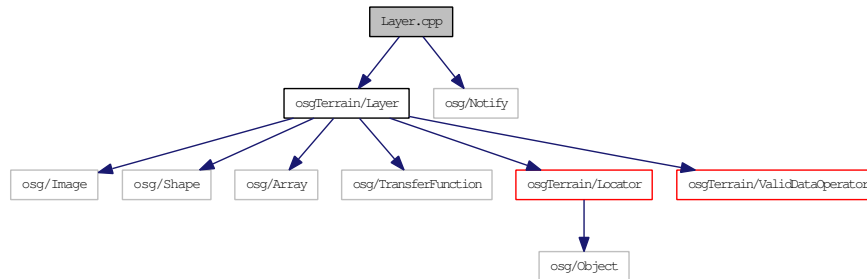
5.4.1.2 **#define** OSGTERRAIN_LAYER 1

5.5 Layer.cpp File Reference

```
#include <osgTerrain/Layer>
```

```
#include <osg/Notify>
```

Include dependency graph for Layer.cpp:



Classes

- struct **TransformOperator**

Functions

- template<typename T , class O >
void **_processRow** (unsigned int num, GLenum pixelFormat, T *data, const O &operation)
- template<class O >
void **processImage** (osg::Image *image, const O &operation)
- template<class O >
void **processRow** (unsigned int num, GLenum pixelFormat, GLenum dataType, unsigned char *data, const O &operation)

5.5.1 Function Documentation

5.5.1.1 void **_processRow** (unsigned int *num*, GLenum *pixelFormat*, T * *data*, const O & *operation*) [inline]

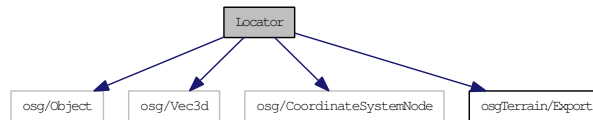
5.5.1.2 void **processImage** (osg::Image * *image*, const O & *operation*) [inline]

5.5.1.3 void **processRow** (unsigned int *num*, GLenum *pixelFormat*, GLenum *dataType*, unsigned char * *data*, const O & *operation*) [inline]

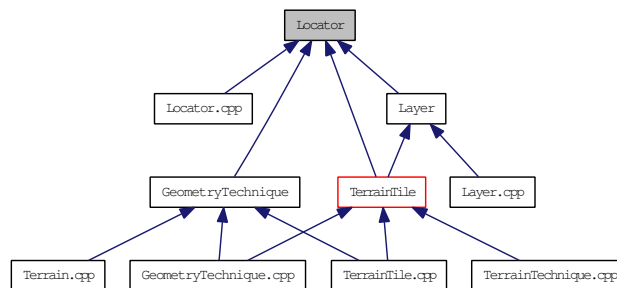
5.6 Locator File Reference

```
#include <osg/Object>
#include <osg/Vec3d>
#include <osg/CoordinateSystemNode>
#include <osgTerrain/Export>
```

Include dependency graph for Locator:



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



Classes

- class **Locator**

Namespaces

- namespace **osgTerrain**

*The **osgTerrain** (p. 7) library is a NodeKit that provides geospecific terrain rendering support.*

Defines

- #define **OSGTERRAIN_LOCATOR 1**

5.6.1 Define Documentation

5.6.1.1 #define OSGTERRAIN_LOCATOR 1

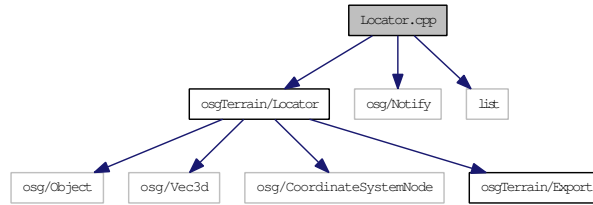
5.7 Locator.cpp File Reference

```
#include <osgTerrain/Locator>
```

```
#include <osg/Notify>
```

```
#include <list>
```

Include dependency graph for Locator.cpp:



5.8 mainpage.h File Reference

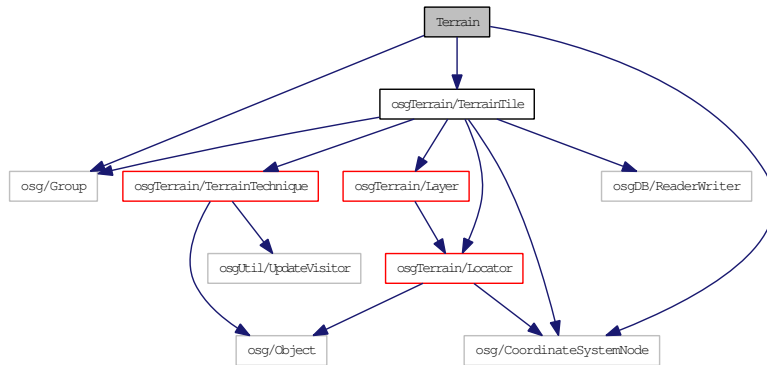
5.8.1 Detailed Description

This file contains doxygen special commands and text for the **Main Page** (p. ??) and some other minor aspects of this documentation. It is not part of the OSG.

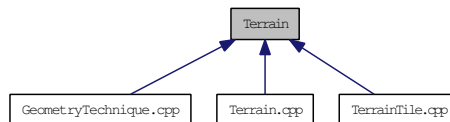
5.9 Terrain File Reference

```
#include <osg/Group>
#include <osg/CoordinateSystemNode>
#include <osgTerrain/TerrainTile>
```

Include dependency graph for Terrain:



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



Classes

- class **Terrain**
Terrain (p. 42) provides a framework for loosely coupling height field data with height rendering algorithms.

Namespaces

- namespace **osgTerrain**
The osgTerrain (p. 7) library is a NodeKit that provides geospecific terrain rendering support.

Defines

- #define **OSGTerrain 1**

5.9.1 Define Documentation

5.9.1.1 #define OSGTerrain 1

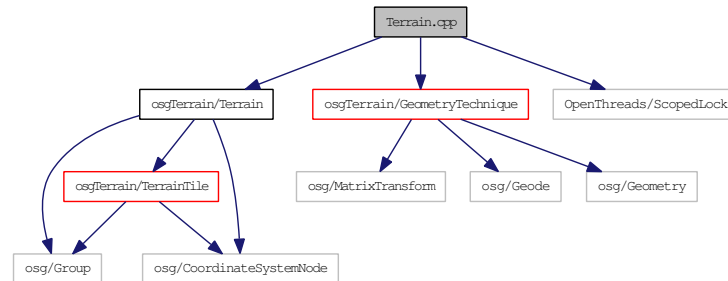
5.10 Terrain.cpp File Reference

```
#include <osgTerrain/Terrain>
```

```
#include <osgTerrain/GeometryTechnique>
```

```
#include <OpenThreads/ScopedLock>
```

Include dependency graph for Terrain.cpp:



Variables

- static unsigned int **s_maxNumTiles** = 0

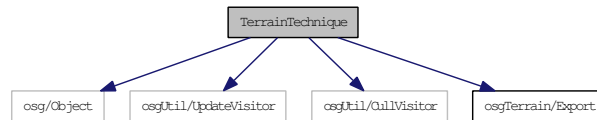
5.10.1 Variable Documentation

5.10.1.1 unsigned int **s_maxNumTiles** = 0 [static]

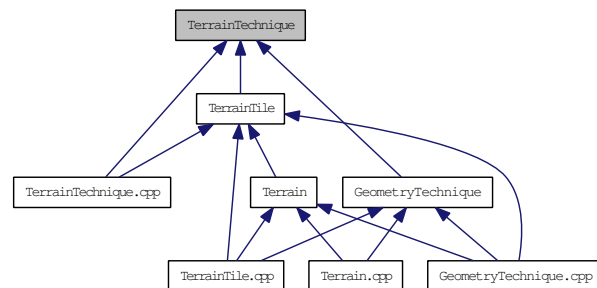
5.11 TerrainTechnique File Reference

```
#include <osg/Object>
#include <osgUtil/UpdateVisitor>
#include <osgUtil/CullVisitor>
#include <osgTerrain/Export>
```

Include dependency graph for TerrainTechnique:



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



Classes

- class **TerrainTechnique**

Namespaces

- namespace **osgTerrain**

*The **osgTerrain** (p. 7) library is a NodeKit that provides geospecific terrain rendering support.*

Defines

- #define **OSGTERRAIN_TERRAINTECHNIQUE 1**

5.11.1 Define Documentation

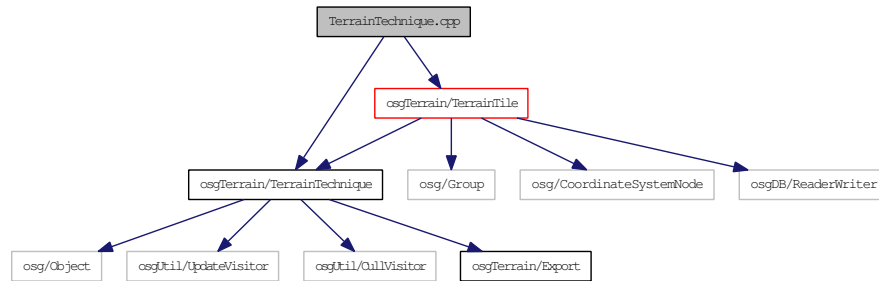
5.11.1.1 #define OSGTERRAIN_TERRAINTECHNIQUE 1

5.12 TerrainTechnique.cpp File Reference

```
#include <osgTerrain/TerrainTechnique>
```

```
#include <osgTerrain/TerrainTile>
```

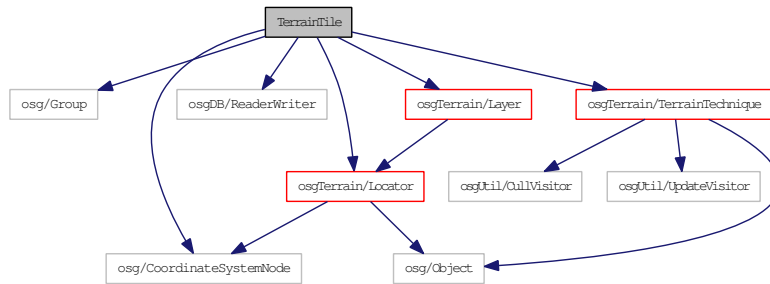
Include dependency graph for TerrainTechnique.cpp:



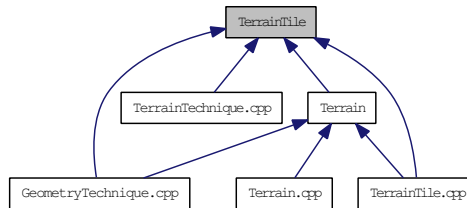
5.13 TerrainTile File Reference

```
#include <osg/Group>
#include <osg/CoordinateSystemNode>
#include <osgDB/ReaderWriter>
#include <osgTerrain/TerrainTechnique>
#include <osgTerrain/Layer>
#include <osgTerrain/Locator>
```

Include dependency graph for TerrainTile:



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



Classes

- class **TerrainTile**
Terrain (p. 42) provides a framework for loosely coupling height field data with height rendering algorithms.
- class **TileID**
- struct **TileLoadedCallback**
Callback for post processing loaded TerrainTile (p. 48), and for filling in missing elements such as external external imagery.
- class **WhiteListTileLoadedCallback**
Helper callback for managing optional sets of layers, that loading of is deferred to this callback, with this callback working out which layers to load, and how to create fallback versions of the layers.

Namespaces

- namespace **osgTerrain**
The osgTerrain (p. 7) library is a NodeKit that provides geospecific terrain rendering support.

Defines

- #define **OSGTERRAIN_TERRAINTILE 1**

5.13.1 Define Documentation

5.13.1.1 `#define OSGTERRAIN_TERRAINTILE 1`

5.14 TerrainTile.cpp File Reference

```
#include <osgTerrain/TerrainTile>
```

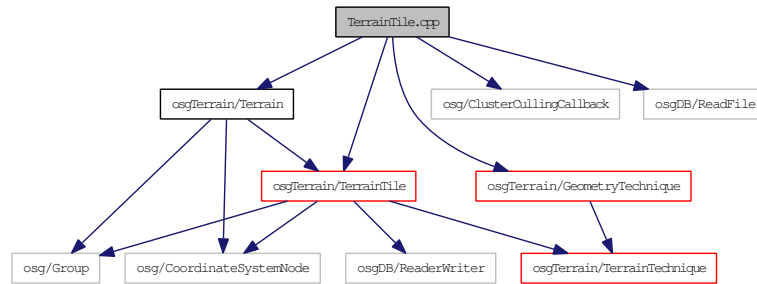
```
#include <osgTerrain/Terrain>
```

```
#include <osgTerrain/GeometryTechnique>
```

```
#include <osg/ClusterCullingCallback>
```

```
#include <osgDB/ReadFile>
```

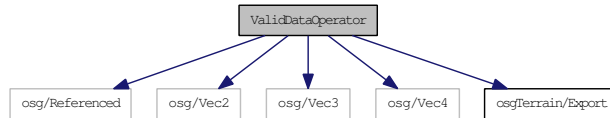
Include dependency graph for TerrainTile.cpp:



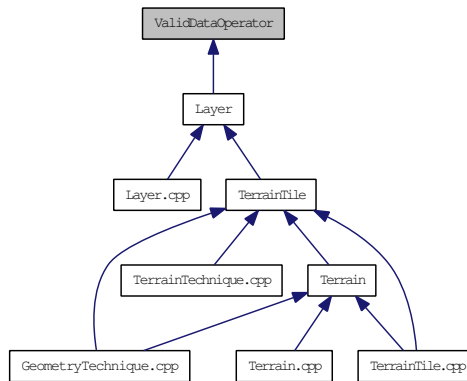
5.15 ValidDataOperator File Reference

```
#include <osg/Referenced>
#include <osg/Vec2>
#include <osg/Vec3>
#include <osg/Vec4>
#include <osgTerrain/Export>
```

Include dependency graph for ValidDataOperator:



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



Classes

- struct **NoDataValue**
- struct **ValidDataOperator**
- struct **ValidRange**

Namespaces

- namespace **osgTerrain**

The *osgTerrain* (p. 7) library is a NodeKit that provides geospecific terrain rendering support.

Defines

- #define **OSGTERRAIN_VALIDDATAOPERATOR 1**

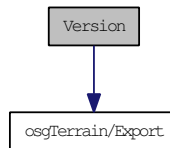
5.15.1 Define Documentation

5.15.1.1 #define OSGTERRAIN_VALIDDATAOPERATOR 1

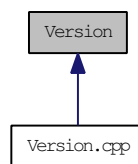
5.16 Version File Reference

```
#include <osgTerrain/Export>
```

Include dependency graph for Version:



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



Defines

- #define **OSGTERRAIN_VERSION** 1

Functions

- OSGTERRAIN_EXPORT const char * **osgTerrainGetLibraryName** ()
osgTerrainGetLibraryName() (p. 81) returns the library name in human friendly form.
- OSGTERRAIN_EXPORT const char * **osgTerrainGetVersion** ()
osgTerrainGetVersion() (p. 81) returns the library version number.

5.16.1 Define Documentation

5.16.1.1 #define OSGTERRAIN_VERSION 1

5.16.2 Function Documentation

5.16.2.1 OSGTERRAIN_EXPORT const char* osgTerrainGetLibraryName ()

osgTerrainGetLibraryName() (p. 81) returns the library name in human friendly form.

5.16.2.2 OSGTERRAIN_EXPORT const char* osgTerrainGetVersion ()

osgTerrainGetVersion() (p. 81) returns the library version number. Numbering convention : OpenSceneGraph-1.0 will return 1.0 from **osgTerrainGetVersion**.

This C function can be also used to check for the existence of the OpenSceneGraph library using autoconf and its m4 macro **AC_CHECK_LIB**.

Here is the code to add to your configure.in:

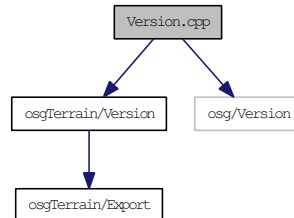
```
#
# Check for the OpenSceneGraph (OSG) Terrain library
#
AC_CHECK_LIB(osg, osgTerrainGetVersion, ,
  [AC_MSG_ERROR(OpenSceneGraph Terrain library not found. See http://www.openscenegraph.org)],)
```

5.17 Version.cpp File Reference

```
#include <osgTerrain/Version>
```

```
#include <osg/Version>
```

Include dependency graph for Version.cpp:



Functions

- const char * **osgTerrainGetLibraryName** ()
osgTerrainGetLibraryName() (p. 81) returns the library name in human friendly form.
- const char * **osgTerrainGetVersion** ()
osgTerrainGetVersion() (p. 81) returns the library version number.

5.17.1 Function Documentation

5.17.1.1 const char* osgTerrainGetLibraryName ()

osgTerrainGetLibraryName() (p. 81) returns the library name in human friendly form.

5.17.1.2 const char* osgTerrainGetVersion ()

osgTerrainGetVersion() (p. 81) returns the library version number. Numbering convention : OpenSceneGraph-1.0 will return 1.0 from *osgTerrainGetVersion*.

This C function can be also used to check for the existence of the OpenSceneGraph library using autoconf and its m4 macro `AC_CHECK_LIB`.

Here is the code to add to your configure.in:

```

#
# Check for the OpenSceneGraph (OSG) Terrain library
#
AC_CHECK_LIB(osg, osgTerrainGetVersion, ,
  [AC_MSG_ERROR(OpenSceneGraph Terrain library not found. See http://www.openscenegraph.org)],)

```

Index

- Symbols -

- ~CompositeLayer
 - osgTerrain::CompositeLayer, 11
- ~ContourLayer
 - osgTerrain::ContourLayer, 15
- ~HeightFieldLayer
 - osgTerrain::HeightFieldLayer, 21
- ~ImageLayer
 - osgTerrain::ImageLayer, 24
- ~Layer
 - osgTerrain::Layer, 28
- ~Locator
 - osgTerrain::Locator, 33
- ~ProxyLayer
 - osgTerrain::ProxyLayer, 37
- ~SwitchLayer
 - osgTerrain::SwitchLayer, 41
- ~Terrain
 - osgTerrain::Terrain, 43
- ~TerrainTechnique
 - osgTerrain::TerrainTechnique, 46
- ~TerrainTile
 - osgTerrain::TerrainTile, 50
- ~WhiteListTileLoadedCallback
 - osgTerrain::WhiteListTileLoadedCallback, 61
- _activeLayer
 - osgTerrain::SwitchLayer, 41
- _allowAll
 - osgTerrain::WhiteListTileLoadedCallback, 61
- _colorLayers
 - osgTerrain::TerrainTile, 53
- _coordinateSystemType
 - osgTerrain::Locator, 34
- _cs
 - osgTerrain::Locator, 34
- _defaultValue
 - osgTerrain::Layer, 30
- _definedInFile
 - osgTerrain::Locator, 34
- _dirty
 - osgTerrain::TerrainTile, 53
- _elevationLayer
 - osgTerrain::TerrainTile, 53
- _ellipsoidModel
 - osgTerrain::Locator, 34
- _filename
 - osgTerrain::Layer, 30
- _format
 - osgTerrain::Locator, 34
- _hasBeenTraversal
 - osgTerrain::TerrainTile, 53
- _heightField
 - osgTerrain::HeightFieldLayer, 22
- _image
 - osgTerrain::ImageLayer, 25
- _implementation
 - osgTerrain::ProxyLayer, 39
- _inverse
 - osgTerrain::Locator, 34
- _layers
 - osgTerrain::CompositeLayer, 12
 - _locator
 - osgTerrain::Layer, 30
 - osgTerrain::TerrainTile, 53
 - _magFilter
 - osgTerrain::Layer, 30
 - _maxLevel
 - osgTerrain::Layer, 30
 - _maxValue
 - osgTerrain::ValidRange, 59
 - _minFilter
 - osgTerrain::Layer, 30
 - _minLevel
 - osgTerrain::Layer, 30
 - _minValue
 - osgTerrain::ValidRange, 59
 - _minimumNumberOfLayers
 - osgTerrain::WhiteListTileLoadedCallback, 61
 - _modifiedCount
 - osgTerrain::HeightFieldLayer, 22
 - _mutex
 - osgTerrain::Terrain, 44
 - _offset
 - TransformOperator, 56
 - _processRow
 - Layer.cpp, 68
 - _replaceSwitchLayer
 - osgTerrain::WhiteListTileLoadedCallback, 61
 - _requiresNormals
 - osgTerrain::TerrainTile, 53
 - _sampleRatio
 - osgTerrain::Terrain, 44
 - _scale
 - TransformOperator, 56
 - _setWhiteList
 - osgTerrain::WhiteListTileLoadedCallback, 61
 - _terrain
 - osgTerrain::TerrainTile, 53
 - _terrainTechnique
 - osgTerrain::Terrain, 44
 - osgTerrain::TerrainTile, 53
 - _terrainTile
 - osgTerrain::TerrainTechnique, 47
 - _terrainTileMap
 - osgTerrain::Terrain, 44
 - _terrainTileSet
 - osgTerrain::Terrain, 44
 - _tf
 - osgTerrain::ContourLayer, 16
 - _tileID
 - osgTerrain::TerrainTile, 53
 - _transform
 - osgTerrain::Locator, 34
 - _transformScaledByResolution
 - osgTerrain::Locator, 34
 - _treatBoundariesToValidDataAsDefaultValue
 - osgTerrain::TerrainTile, 53
 - _validDataOperator
 - osgTerrain::Layer, 30
 - _value

- osgTerrain::NoDataValue, 35
- _verticalScale
 - osgTerrain::Terrain, 44
- A -**
- addLayer
 - osgTerrain::CompositeLayer, 11
- allow
 - osgTerrain::WhiteListTileLoadedCallback, 61
- applyColorLayers
 - osgTerrain::GeometryTechnique, 18
- applyTransparency
 - osgTerrain::GeometryTechnique, 18
- C -**
- cleanSceneGraph
 - osgTerrain::GeometryTechnique, 18
 - osgTerrain::TerrainTechnique, 46
- clear
 - osgTerrain::CompositeLayer, 11
- CompositeLayer
 - osgTerrain::CompositeLayer, 11
- CompoundNameLayer
 - osgTerrain::CompositeLayer::CompoundNameLayer, 13
- computeBound
 - osgTerrain::Layer, 28
 - osgTerrain::ProxyLayer, 37
 - osgTerrain::TerrainTile, 51
- computeCenterModel
 - osgTerrain::GeometryTechnique, 18
- computeIndices
 - osgTerrain::Layer, 28
- computeLocalBounds
 - osgTerrain::Locator, 33
- computeMasterLocator
 - osgTerrain::GeometryTechnique, 18
- ContourLayer
 - osgTerrain::ContourLayer, 15
- convertLocalCoordBetween
 - osgTerrain::Locator, 33
- convertLocalToModel
 - osgTerrain::Locator, 33
- convertModelToLocal
 - osgTerrain::Locator, 33
- CoordinateSystemType
 - osgTerrain::Locator, 32
- createCompondSetNameAndFileName
 - osgTerrain, 8
- cull
 - osgTerrain::GeometryTechnique, 18
 - osgTerrain::TerrainTechnique, 46
- D -**
- deferExternalLayerLoading
 - osgTerrain::TerrainTile::TileLoadedCallback, 55
 - osgTerrain::WhiteListTileLoadedCallback, 61
- dirty
 - osgTerrain::ContourLayer, 15
 - osgTerrain::HeightFieldLayer, 21
 - osgTerrain::ImageLayer, 24
 - osgTerrain::Layer, 28
 - osgTerrain::ProxyLayer, 37
- dirtyRegisteredTiles
 - osgTerrain::Terrain, 43
- E -**
- Export, 63
 - OSGTERRAIN_EXPORT, 63
 - OSGTERRAIN_EXPORT_, 63
- extractSetNameAndFileName
 - osgTerrain, 8
- F -**
- filename
 - osgTerrain::CompositeLayer::CompoundNameLayer, 13
- FilterType
 - osgTerrain::GeometryTechnique, 18
- G -**
- GAUSSIAN
 - osgTerrain::GeometryTechnique, 18
- generateGeometry
 - osgTerrain::GeometryTechnique, 18
- GEOCENTRIC
 - osgTerrain::Locator, 32
- GEOGRAPHIC
 - osgTerrain::Locator, 32
- GeometryTechnique, 64
 - osgTerrain::GeometryTechnique, 18
 - OSGTERRAIN_GEOMETRYTECHNIQUE, 64
- GeometryTechnique.cpp, 65
 - NEW_COORD_CODE, 65
- getActiveLayer
 - osgTerrain::SwitchLayer, 41
- getAllowAll
 - osgTerrain::WhiteListTileLoadedCallback, 61
- getColorLayer
 - osgTerrain::TerrainTile, 51
- getCompoundName
 - osgTerrain::CompositeLayer, 11
 - osgTerrain::Layer, 28
- getCoordinateSystem
 - osgTerrain::Locator, 33
- getCoordinateSystemType
 - osgTerrain::Locator, 33
- getDefaultValue
 - osgTerrain::Layer, 28
- getDefinedInFile
 - osgTerrain::Locator, 33
- getDirty
 - osgTerrain::TerrainTile, 51
- getElevationLayer
 - osgTerrain::TerrainTile, 51
- getEllipsoidModel
 - osgTerrain::Locator, 33
- getFileName
 - osgTerrain::CompositeLayer, 11
 - osgTerrain::HeightFieldLayer, 21
 - osgTerrain::ImageLayer, 24
 - osgTerrain::Layer, 28
 - osgTerrain::ProxyLayer, 38
- getFilterBias
 - osgTerrain::GeometryTechnique, 19
- getFilterMatrix
 - osgTerrain::GeometryTechnique, 19
- getFilterWidth
 - osgTerrain::GeometryTechnique, 19
- getFormat

- osgTerrain::Locator, 33
 - getHeightField
 - osgTerrain::HeightFieldLayer, 21
 - getImage
 - osgTerrain::ContourLayer, 15
 - osgTerrain::ImageLayer, 24
 - osgTerrain::Layer, 28
 - osgTerrain::ProxyLayer, 38
 - osgTerrain::SwitchLayer, 41
 - getImplementation
 - osgTerrain::ProxyLayer, 38
 - getInterpolatedValue
 - osgTerrain::Layer, 28
 - getLayer
 - osgTerrain::CompositeLayer, 11
 - getLocator
 - osgTerrain::Layer, 28
 - osgTerrain::TerrainTile, 51
 - getMagFilter
 - osgTerrain::Layer, 28
 - getMaxLevel
 - osgTerrain::Layer, 28
 - getMaxValue
 - osgTerrain::ValidRange, 58
 - getMinFilter
 - osgTerrain::Layer, 29
 - getMinimumNumOfLayers
 - osgTerrain::WhiteListTileLoadedCallback, 61
 - getMinLevel
 - osgTerrain::Layer, 29
 - getMinValue
 - osgTerrain::ValidRange, 58
 - getModifiedCount
 - osgTerrain::ContourLayer, 15
 - osgTerrain::HeightFieldLayer, 21
 - osgTerrain::ImageLayer, 25
 - osgTerrain::Layer, 29
 - osgTerrain::ProxyLayer, 38
 - getNumColorLayers
 - osgTerrain::TerrainTile, 51
 - getNumColumns
 - osgTerrain::ContourLayer, 15
 - osgTerrain::HeightFieldLayer, 21
 - osgTerrain::ImageLayer, 25
 - osgTerrain::Layer, 29
 - osgTerrain::ProxyLayer, 38
 - getNumLayers
 - osgTerrain::CompositeLayer, 11
 - getNumRows
 - osgTerrain::ContourLayer, 15
 - osgTerrain::HeightFieldLayer, 21
 - osgTerrain::ImageLayer, 25
 - osgTerrain::Layer, 29
 - osgTerrain::ProxyLayer, 38
 - getReplaceSwitchLayer
 - osgTerrain::WhiteListTileLoadedCallback, 61
 - getRequiresNormals
 - osgTerrain::TerrainTile, 51
 - getSampleRatio
 - osgTerrain::Terrain, 43
 - getSetName
 - osgTerrain::CompositeLayer, 11
 - osgTerrain::Layer, 29
 - getTerrain
 - osgTerrain::TerrainTile, 51
 - getTerrainTechnique
 - osgTerrain::TerrainTile, 51
 - getTerrainTechniquePrototype
 - osgTerrain::Terrain, 43
 - getTerrainTile
 - osgTerrain::TerrainTechnique, 46
 - getTile
 - osgTerrain::Terrain, 43
 - getTileID
 - osgTerrain::TerrainTile, 51
 - getTileLoadedCallback
 - osgTerrain::TerrainTile, 51
 - getTransferFunction
 - osgTerrain::ContourLayer, 16
 - getTransform
 - osgTerrain::Locator, 33
 - getTransformScaledByResolution
 - osgTerrain::Locator, 33
 - getTreatBoundariesToValidDataAsDefaultValue
 - osgTerrain::TerrainTile, 51
 - getValidDataOperator
 - osgTerrain::Layer, 29
 - getValidValue
 - osgTerrain::Layer, 29
 - getValue
 - osgTerrain::ContourLayer, 16
 - osgTerrain::HeightFieldLayer, 22
 - osgTerrain::ImageLayer, 25
 - osgTerrain::Layer, 29
 - osgTerrain::NoDataValue, 35
 - osgTerrain::ProxyLayer, 38
 - getVerticalScale
 - osgTerrain::Terrain, 44
- H -**
- HeightFieldLayer
 - osgTerrain::HeightFieldLayer, 21
- I -**
- ImageLayer
 - osgTerrain::ImageLayer, 24
 - include/ Directory Reference, 3
 - include/osgTerrain/ Directory Reference, 5
 - init
 - osgTerrain::GeometryTechnique, 19
 - osgTerrain::TerrainTechnique, 46
 - osgTerrain::TerrainTile, 51
- L -**
- Layer, 66
 - MAXIMUM_NUMBER_OF_LEVELS, 67
 - osgTerrain::Layer, 28
 - OSGTERRAIN_LAYER, 67
 - layer
 - osgTerrain::CompositeLayer::CompoundNameLayer, 13
 - Layer.cpp, 68
 - _processRow, 68
 - processImage, 68
 - processRow, 68
 - layerAcceptable
 - osgTerrain::WhiteListTileLoadedCallback, 61
 - Layers
 - osgTerrain::CompositeLayer, 11
 - osgTerrain::TerrainTile, 50
 - level
 - osgTerrain::TileID, 54

- loaded
 - osgTerrain::TerrainTile::TileLoadedCallback, 55
 - osgTerrain::WhiteListTileLoadedCallback, 61
- Locator, 69
 - osgTerrain::Locator, 33
 - OSGTERRAIN_LOCATOR, 69
- Locator.cpp, 70
- M -**
- mainpage.h, 71
- MAXIMUM_NUMBER_OF_LEVELS
 - Layer, 67
- META_Node
 - osgTerrain::Terrain, 44
 - osgTerrain::TerrainTile, 51
- META_Object
 - osgTerrain::CompositeLayer, 11
 - osgTerrain::ContourLayer, 16
 - osgTerrain::GeometryTechnique, 19
 - osgTerrain::HeightFieldLayer, 22
 - osgTerrain::ImageLayer, 25
 - osgTerrain::Layer, 29
 - osgTerrain::Locator, 33
 - osgTerrain::ProxyLayer, 38
 - osgTerrain::SwitchLayer, 41
 - osgTerrain::TerrainTechnique, 46
- N -**
- NEW_COORD_CODE
 - GeometryTechnique.cpp, 65
- NoDataValue
 - osgTerrain::NoDataValue, 35
- O -**
- operator<
 - osgTerrain::TileID, 54
- operator()
 - osgTerrain::NoDataValue, 35
 - osgTerrain::ValidDataOperator, 57
 - osgTerrain::ValidRange, 58
 - TransformOperator, 56
- operator=
 - osgTerrain::CompositeLayer::CompoundNameLayer, 13
- operator==
 - osgTerrain::TileID, 54
- orientationOpenGL
 - osgTerrain::Locator, 33
- OSGTerrain
 - Terrain, 72
- osgTerrain, 7
 - createCompoundSetNameAndFileName, 8
 - extractSetNameAndFileName, 8
- osgTerrain::CompositeLayer, 9
 - ~CompositeLayer, 11
 - _layers, 12
 - addLayer, 11
 - clear, 11
 - CompositeLayer, 11
 - getCompoundName, 11
 - getFileName, 11
 - getLayer, 11
 - getNumLayers, 11
 - getSetName, 11
 - Layers, 11
 - META_Object, 11
 - removeLayer, 12
 - setCompoundName, 12
 - setFileName, 12
 - setLayer, 12
 - setSetName, 12
- osgTerrain::CompositeLayer::CompoundNameLayer, 13
 - CompoundNameLayer, 13
 - filename, 13
 - layer, 13
 - operator=, 13
 - setname, 13
- osgTerrain::ContourLayer, 14
 - ~ContourLayer, 15
 - _tf, 16
 - ContourLayer, 15
 - dirty, 15
 - getImage, 15
 - getModifiedCount, 15
 - getNumColumns, 15
 - getNumRows, 15
 - getTransferFunction, 16
 - getValue, 16
 - META_Object, 16
 - setModifiedCount, 16
 - setTransferFunction, 16
 - transform, 16
- osgTerrain::GeometryTechnique, 17
 - applyColorLayers, 18
 - applyTransparency, 18
 - cleanSceneGraph, 18
 - computeCenterModel, 18
 - computeMasterLocator, 18
 - cull, 18
 - FilterType, 18
 - GAUSSIAN, 18
 - generateGeometry, 18
 - GeometryTechnique, 18
 - getFilterBias, 19
 - getFilterMatrix, 19
 - getFilterWidth, 19
 - init, 19
 - META_Object, 19
 - releaseGLObjects, 19
 - setFilterBias, 19
 - setFilterMatrix, 19
 - setFilterMatrixAs, 19
 - setFilterWidth, 19
 - SHARPEN, 18
 - SMOOTH, 18
 - smoothGeometry, 19
 - traverse, 19
 - update, 19
- osgTerrain::HeightFieldLayer, 20
 - ~HeightFieldLayer, 21
 - _heightField, 22
 - _modifiedCount, 22
 - dirty, 21
 - getFileName, 21
 - getHeightField, 21
 - getModifiedCount, 21
 - getNumColumns, 21
 - getNumRows, 21
 - getValue, 22
 - HeightFieldLayer, 21
 - META_Object, 22
 - setFileName, 22

- setHeightField, 22
- setModifiedCount, 22
- transform, 22
- osgTerrain::ImageLayer, 23
 - ~ImageLayer, 24
 - _image, 25
 - dirty, 24
 - getFileName, 24
 - getImage, 24
 - getModifiedCount, 25
 - getNumColumns, 25
 - getNumRows, 25
 - getValue, 25
 - ImageLayer, 24
 - META_Object, 25
 - setFileName, 25
 - setImage, 25
 - setModifiedCount, 25
 - transform, 25
- osgTerrain::Layer, 26
 - ~Layer, 28
 - _defaultValue, 30
 - _filename, 30
 - _locator, 30
 - _magFilter, 30
 - _maxLevel, 30
 - _minFilter, 30
 - _minLevel, 30
 - _validDataOperator, 30
 - computeBound, 28
 - computeIndices, 28
 - dirty, 28
 - getCompoundName, 28
 - getDefaultValue, 28
 - getFileName, 28
 - getImage, 28
 - getInterpolatedValue, 28
 - getLocator, 28
 - getMagFilter, 28
 - getMaxLevel, 28
 - getMinFilter, 29
 - getMinLevel, 29
 - getModifiedCount, 29
 - getNumColumns, 29
 - getNumRows, 29
 - getSetName, 29
 - getValidDataOperator, 29
 - getValidValue, 29
 - getValue, 29
 - Layer, 28
 - META_Object, 29
 - setDefaultValue, 29
 - setFileName, 29
 - setLocator, 29
 - setMagFilter, 30
 - setMaxLevel, 30
 - setMinFilter, 30
 - setMinLevel, 30
 - setModifiedCount, 30
 - setSetName, 30
 - setValidDataOperator, 30
 - transform, 30
- osgTerrain::Locator, 31
 - ~Locator, 33
 - _coordinateSystemType, 34
 - _cs, 34
 - _definedInFile, 34
 - _ellipsoidModel, 34
 - _format, 34
 - _inverse, 34
 - _transform, 34
 - _transformScaledByResolution, 34
 - computeLocalBounds, 33
 - convertLocalCoordBetween, 33
 - convertLocalToModel, 33
 - convertModelToLocal, 33
 - CoordinateSystemType, 32
 - GEOCENTRIC, 32
 - GEOGRAPHIC, 32
 - getCoordinateSystem, 33
 - getCoordinateSystemType, 33
 - getDefinedInFile, 33
 - getEllipsoidModel, 33
 - getFormat, 33
 - getTransform, 33
 - getTransformScaledByResolution, 33
 - Locator, 33
 - META_Object, 33
 - orientationOpenGL, 33
 - PROJECTED, 32
 - setCoordinateSystem, 33
 - setCoordinateSystemType, 33
 - setDefinedInFile, 33
 - setEllipsoidModel, 33
 - setFormat, 33
 - setTransform, 33
 - setTransformAsExtents, 33
 - setTransformScaledByResolution, 34
- osgTerrain::NoDataValue, 35
 - _value, 35
 - getValue, 35
 - NoDataValue, 35
 - operator(), 35
 - setNoDataValue, 35
- osgTerrain::ProxyLayer, 36
 - ~ProxyLayer, 37
 - _implementation, 39
 - computeBound, 37
 - dirty, 37
 - getFileName, 38
 - getImage, 38
 - getImplementation, 38
 - getModifiedCount, 38
 - getNumColumns, 38
 - getNumRows, 38
 - getValue, 38
 - META_Object, 38
 - ProxyLayer, 37
 - setFileName, 38
 - setImplementation, 38
 - setModifiedCount, 38
 - transform, 39
- osgTerrain::SwitchLayer, 40
 - ~SwitchLayer, 41
 - _activeLayer, 41
 - getActiveLayer, 41
 - getImage, 41
 - META_Object, 41
 - setActiveLayer, 41
 - SwitchLayer, 41
- osgTerrain::Terrain, 42
 - ~Terrain, 43

- _mutex, 44
 - _sampleRatio, 44
 - _terrainTechnique, 44
 - _terrainTileMap, 44
 - _terrainTileSet, 44
 - _verticalScale, 44
- dirtyRegisteredTiles, 43
- getSampleRatio, 43
- getTerrainTechniquePrototype, 43
- getTile, 43
- getVerticalScale, 44
- META_Node, 44
- registerTerrainTile, 44
- setSampleRatio, 44
- setTerrainTechniquePrototype, 44
- setVerticalScale, 44
- Terrain, 43
- TerrainTile, 44
- TerrainTileMap, 43
- TerrainTileSet, 43
- traverse, 44
- unregisterTerrainTile, 44
- osgTerrain::TerrainTechnique, 45
 - ~TerrainTechnique, 46
 - _terrainTile, 47
 - cleanSceneGraph, 46
 - cull, 46
 - getTerrainTile, 46
 - init, 46
 - META_Object, 46
 - osgTerrain::TerrainTile, 47
 - releaseGLObjects, 46
 - setDirty, 46
 - TerrainTechnique, 46
 - traverse, 46
 - update, 46
- osgTerrain::TerrainTile, 48
 - ~TerrainTile, 50
 - _colorLayers, 53
 - _dirty, 53
 - _elevationLayer, 53
 - _hasBeenTraversal, 53
 - _locator, 53
 - _requiresNormals, 53
 - _terrain, 53
 - _terrainTechnique, 53
 - _tileID, 53
 - _treatBoundariesToValidDataAsDefaultValue, 53
 - computeBound, 51
 - getColorLayer, 51
 - getDirty, 51
 - getElevationLayer, 51
 - getLocator, 51
 - getNumColorLayers, 51
 - getRequiresNormals, 51
 - getTerrain, 51
 - getTerrainTechnique, 51
 - getTileID, 51
 - getTileLoadedCallback, 51
 - getTreatBoundariesToValidDataAsDefaultValue, 51
 - init, 51
 - Layers, 50
 - META_Node, 51
 - osgTerrain::TerrainTechnique, 47
 - releaseGLObjects, 52
 - setColorLayer, 52
 - setDirty, 52
 - setElevationLayer, 52
 - setLocator, 52
 - setRequiresNormals, 52
 - setTerrain, 52
 - setTerrainTechnique, 52
 - setTileID, 52
 - setTileLoadedCallback, 52
 - setTreatBoundariesToValidDataAsDefaultValue, 52
 - Terrain, 53
 - TerrainTile, 50
 - traverse, 52
- osgTerrain::TerrainTile::TileLoadedCallback, 55
 - deferExternalLayerLoading, 55
 - loaded, 55
- osgTerrain::TileID, 54
 - level, 54
 - operator<, 54
 - operator==, 54
 - TileID, 54
 - valid, 54
 - x, 54
 - y, 54
- osgTerrain::ValidDataOperator, 57
 - operator(), 57
- osgTerrain::ValidRange, 58
 - _maxValue, 59
 - _minValue, 59
 - getMaxValue, 58
 - getMinValue, 58
 - operator(), 58
 - setMaxValue, 58
 - setMinValue, 59
 - setRange, 59
 - ValidRange, 58
- osgTerrain::WhiteListTileLoadedCallback, 60
 - ~WhiteListTileLoadedCallback, 61
 - _allowAll, 61
 - _minumumNumberOfLayers, 61
 - _replaceSwitchLayer, 61
 - _setWhiteList, 61
 - allow, 61
 - deferExternalLayerLoading, 61
 - getAllowAll, 61
 - getMinimumNumOfLayers, 61
 - getReplaceSwitchLayer, 61
 - layerAcceptable, 61
 - loaded, 61
 - readImageLayer, 61
 - setAllowAll, 61
 - setMinimumNumOfLayers, 61
 - setReplaceSwitchLayer, 61
 - SetWhiteList, 61
 - WhiteListTileLoadedCallback, 61
- OSGTERRAIN_EXPORT
 - Export, 63
- OSGTERRAIN_EXPORT_
 - Export, 63
- OSGTERRAIN_GEOMETRYTECHNIQUE
 - GeometryTechnique, 64
- OSGTERRAIN_LAYER
 - Layer, 67
- OSGTERRAIN_LOCATOR
 - Locator, 69
- OSGTERRAIN_TERRAINTECHNIQUE
 - TerrainTechnique, 74

- OSGTERRAIN_TERRAINTILE
 - TerrainTile, 77
- OSGTERRAIN_VALIDDATAOPERATOR
 - ValidDataOperator, 79
- OSGTERRAIN_VERSION
 - Version, 80
- osgTerrainGetLibraryName
 - Version, 80
 - Version.cpp, 81
- osgTerrainGetVersion
 - Version, 80
 - Version.cpp, 81
- P -**
- processImage
 - Layer.cpp, 68
- processRow
 - Layer.cpp, 68
- PROJECTED
 - osgTerrain::Locator, 32
- ProxyLayer
 - osgTerrain::ProxyLayer, 37
- R -**
- readImageLayer
 - osgTerrain::WhiteListTileLoadedCallback, 61
- registerTerrainTile
 - osgTerrain::Terrain, 44
- releaseGLObjects
 - osgTerrain::GeometryTechnique, 19
 - osgTerrain::TerrainTechnique, 46
 - osgTerrain::TerrainTile, 52
- removeLayer
 - osgTerrain::CompositeLayer, 12
- S -**
- s_maxNumTiles
 - Terrain.cpp, 73
- setActiveLayer
 - osgTerrain::SwitchLayer, 41
- setAllowAll
 - osgTerrain::WhiteListTileLoadedCallback, 61
- setColorLayer
 - osgTerrain::TerrainTile, 52
- setCompoundName
 - osgTerrain::CompositeLayer, 12
- setCoordinateSystem
 - osgTerrain::Locator, 33
- setCoordinateSystemType
 - osgTerrain::Locator, 33
- setDefaultValue
 - osgTerrain::Layer, 29
- setDefinedInFile
 - osgTerrain::Locator, 33
- setDirty
 - osgTerrain::TerrainTechnique, 46
 - osgTerrain::TerrainTile, 52
- setElevationLayer
 - osgTerrain::TerrainTile, 52
- setEllipsoidModel
 - osgTerrain::Locator, 33
- setFileName
 - osgTerrain::CompositeLayer, 12
 - osgTerrain::HeightFieldLayer, 22
 - osgTerrain::ImageLayer, 25
 - osgTerrain::Layer, 29
 - osgTerrain::ProxyLayer, 38
- setFilterBias
 - osgTerrain::GeometryTechnique, 19
- setFilterMatrix
 - osgTerrain::GeometryTechnique, 19
- setFilterMatrixAs
 - osgTerrain::GeometryTechnique, 19
- setFilterWidth
 - osgTerrain::GeometryTechnique, 19
- setFormat
 - osgTerrain::Locator, 33
- setHeightField
 - osgTerrain::HeightFieldLayer, 22
- setImage
 - osgTerrain::ImageLayer, 25
- setImplementation
 - osgTerrain::ProxyLayer, 38
- setLayer
 - osgTerrain::CompositeLayer, 12
- setLocator
 - osgTerrain::Layer, 29
 - osgTerrain::TerrainTile, 52
- setMagFilter
 - osgTerrain::Layer, 30
- setMaxLevel
 - osgTerrain::Layer, 30
- setMaxValue
 - osgTerrain::ValidRange, 58
- setMinFilter
 - osgTerrain::Layer, 30
- setMinimumNumOfLayers
 - osgTerrain::WhiteListTileLoadedCallback, 61
- setMinLevel
 - osgTerrain::Layer, 30
- setMinValue
 - osgTerrain::ValidRange, 59
- setModifiedCount
 - osgTerrain::ContourLayer, 16
 - osgTerrain::HeightFieldLayer, 22
 - osgTerrain::ImageLayer, 25
 - osgTerrain::Layer, 30
 - osgTerrain::ProxyLayer, 38
- setname
 - osgTerrain::CompositeLayer::CompoundNameLayer, 13
- setNoDataValue
 - osgTerrain::NoDataValue, 35
- setRange
 - osgTerrain::ValidRange, 59
- setReplaceSwitchLayer
 - osgTerrain::WhiteListTileLoadedCallback, 61
- setRequiresNormals
 - osgTerrain::TerrainTile, 52
- setSampleRatio
 - osgTerrain::Terrain, 44
- setSetName
 - osgTerrain::CompositeLayer, 12
 - osgTerrain::Layer, 30
- setTerrain
 - osgTerrain::TerrainTile, 52
- setTerrainTechnique
 - osgTerrain::TerrainTile, 52
- setTerrainTechniquePrototype
 - osgTerrain::Terrain, 44
- setTileD
 - osgTerrain::TerrainTile, 52

- setTileLoadedCallback
 - osgTerrain::TerrainTile, 52
- setTransferFunction
 - osgTerrain::ContourLayer, 16
- setTransform
 - osgTerrain::Locator, 33
- setTransformAsExtents
 - osgTerrain::Locator, 33
- setTransformScaledByResolution
 - osgTerrain::Locator, 34
- setTreatBoundariesToValidDataAsDefaultValue
 - osgTerrain::TerrainTile, 52
- setValidDataOperator
 - osgTerrain::Layer, 30
- setVerticalScale
 - osgTerrain::Terrain, 44
- SetWhiteList
 - osgTerrain::WhiteListTileLoadedCallback, 61
- SHARPEN
 - osgTerrain::GeometryTechnique, 18
- SMOOTH
 - osgTerrain::GeometryTechnique, 18
- smoothGeometry
 - osgTerrain::GeometryTechnique, 19
- src/ Directory Reference, 6
- src/osgTerrain/ Directory Reference, 4
- SwitchLayer
 - osgTerrain::SwitchLayer, 41
- T -**
- Terrain, 72
 - OSGTerrain, 72
 - osgTerrain::Terrain, 43
 - osgTerrain::TerrainTile, 53
- Terrain.cpp, 73
 - s_maxNumTiles, 73
- TerrainTechnique, 74
 - osgTerrain::TerrainTechnique, 46
 - OSGTERRAIN_TERRAINTECHNIQUE, 74
- TerrainTechnique.cpp, 75
- TerrainTile, 76
 - osgTerrain::Terrain, 44
 - osgTerrain::TerrainTile, 50
 - OSGTERRAIN_TERRAINTILE, 77
- TerrainTile.cpp, 78
- TerrainTileMap
 - osgTerrain::Terrain, 43
- TerrainTileSet
 - osgTerrain::Terrain, 43
- TileID
 - osgTerrain::TileID, 54
- transform
 - osgTerrain::ContourLayer, 16
 - osgTerrain::HeightFieldLayer, 22
 - osgTerrain::ImageLayer, 25
 - osgTerrain::Layer, 30
 - osgTerrain::ProxyLayer, 39
- TransformOperator, 56
 - _offset, 56
 - _scale, 56
 - operator(), 56
 - TransformOperator, 56
- traverse
 - osgTerrain::GeometryTechnique, 19
 - osgTerrain::Terrain, 44
 - osgTerrain::TerrainTechnique, 46
- osgTerrain::TerrainTile, 52
- U -**
- unregisterTerrainTile
 - osgTerrain::Terrain, 44
- update
 - osgTerrain::GeometryTechnique, 19
 - osgTerrain::TerrainTechnique, 46
- V -**
- valid
 - osgTerrain::TileID, 54
- ValidDataOperator, 79
 - OSGTERRAIN_VALIDDATAOPERATOR, 79
- ValidRange
 - osgTerrain::ValidRange, 58
- Version, 80
 - OSGTERRAIN_VERSION, 80
 - osgTerrainGetLibraryName, 80
 - osgTerrainGetVersion, 80
- Version.cpp, 81
 - osgTerrainGetLibraryName, 81
 - osgTerrainGetVersion, 81
- W -**
- WhiteListTileLoadedCallback
 - osgTerrain::WhiteListTileLoadedCallback, 61
- X -**
- x
 - osgTerrain::TileID, 54
- Y -**
- y
 - osgTerrain::TileID, 54