OpenSceneGraph Version 2.9.6

osgManipulator::

Reference Manual

Generated for OpenSceneGraph by Doxygen, Graphviz and MikTex

Wed Jan 6 20:33:30 2010

Contents

1 Main Page **1**

2 Directory Documentation **3**

2.1 include/ Directory Reference . . . 3

2.2 src/osgManipulator/ Directory Reference . . . 4

2.3 include/osgManipulator/ Directory Reference . . . 5

2.4 src/ Directory Reference . . . 6

3 Namespace Documentation **7**

3.1 anonymous_namespace{AntiSquish.cpp} Namespace Reference . . . 7

3.2 anonymous_namespace{Constraint.cpp} Namespace Reference . . . 8

3.2.1 Function Documentation . . . 8

3.2.1.1 round_to_nearest_int . . . 8

3.2.1.2 snap_point_to_grid . . . 8

3.3 anonymous_namespace{Projector.cpp} Namespace Reference . . . 9

3.3.1 Function Documentation . . . 9

3.3.1.1 computeClosestPointOnLine . . . 9

3.3.1.2 computeClosestPoints . . . 9

3.3.1.3 computePlaneParallelToAxisAndOrientedToEye . . . 9

3.3.1.4 computePlaneThruPointAndOrientedToEye . . . 9

3.3.1.5 getCylinderLineIntersection . . . 9

3.3.1.6 getLocalEyeDirection . . . 9

3.3.1.7 getPlaneLineIntersection . . . 9

3.3.1.8 getSphereLineIntersection . . . 9

3.3.1.9 getUnitCylinderLineIntersection . . . 9

3.4 anonymous_namespace{Scale1DDragger.cpp} Namespace Reference . . . 10

3.4.1 Function Documentation . . . 10

3.4.1.1 computeScale . . . 10

3.5 anonymous_namespace{Scale2DDragger.cpp} Namespace Reference . . . 11

3.5.1 Function Documentation . . . 11

3.5.1.1 computeScale . . . 11

3.6 anonymous_namespace{TabPlaneDragger.cpp} Namespace Reference . . . 12

3.6.1 Function Documentation . . . 12

3.6.1.1 createCornerScaleDraggerGeometry . . . 12

3.6.1.2 createEdgeScaleDraggerGeometry . . . 12

3.6.1.3 createHandleNode . . . 12

3.6.1.4 createHandleScene . . . 12

3.6.1.5 createTranslateDraggerGeometry . . . 12
3.7 anonymous_namespace{TrackballDragger.cpp} Namespace Reference . . . 13
3.7.1 Function Documentation . . . 13
3.7.1.1 createCircleGeometry . . . 13
3.8 osgManipulator Namespace Reference . . . 14
3.8.1 Detailed Description . . . 16
3.8.2 Typedef Documentation . . . 16
3.8.2.1 Selection . . . 16
3.8.3 Function Documentation . . . 16
3.8.3.1 computeNodePathToRoot . . . 16
3.8.3.2 setDrawableToAlwaysCull . . . 16
3.8.3.3 setMaterialColor . . . 16

4 Class Documentation **17**
4.1 AntiSquish Class Reference . . . 17
4.1.1 Detailed Description . . . 17
4.1.2 Constructor & Destructor Documentation . . . 18
4.1.2.1 AntiSquish . . . 18
4.1.2.2 AntiSquish . . . 18
4.1.2.3 AntiSquish . . . 18
4.1.2.4 AntiSquish . . . 18
4.1.2.5 ∼AntiSquish . . . 18
4.1.3 Member Function Documentation . . . 18
4.1.3.1 clone . . . 18
4.1.3.2 cloneType . . . 18
4.1.3.3 computeUnSquishedMatrix . . . 18
4.1.3.4 getPivot . . . 18
4.1.3.5 getPosition . . . 18
4.1.3.6 isSameKindAs . . . 18
4.1.3.7 setPivot . . . 18
4.1.3.8 setPosition . . . 18
4.1.4 Member Data Documentation . . . 18
4.1.4.1 _asqCallback . . . 18
4.1.4.2 _cachedLocalToWorld . . . 18
4.1.4.3 _dirty . . . 18
4.1.4.4 _pivot . . . 18
4.1.4.5 _position . . . 18
4.1.4.6 _usePivot . . . 18
4.1.4.7 _usePosition . . . 18
4.2 AntiSquishCallback Class Reference . . . 19
4.2.1 Constructor & Destructor Documentation . . . 19

4.2.1.1 AntiSquishCallback . . . 19
4.2.1.2 ∼AntiSquishCallback . . . 19
4.2.2 Member Function Documentation . . . 19
4.2.2.1 operator() . . . 19
4.2.3 Member Data Documentation . . . 19
4.2.3.1 _antiSquish . . . 19
4.3 CommandManager Class Reference . . . 20
4.3.1 Detailed Description . . . 20
4.3.2 Member Typedef Documentation . . . 20
4.3.2.1 Selections . . . 20
4.3.3 Constructor & Destructor Documentation . . . 20
4.3.3.1 CommandManager . . . 20
4.3.3.2 ∼CommandManager . . . 20
4.3.4 Member Function Documentation . . . 20
4.3.4.1 connect . . . 20
4.3.4.2 connect . . . 20
4.3.4.3 disconnect . . . 20
4.3.4.4 getConnectedSelections . . . 20
4.4 CompositeDragger Class Reference . . . 21
4.4.1 Detailed Description . . . 22
4.4.2 Constructor & Destructor Documentation . . . 22
4.4.2.1 CompositeDragger . . . 22
4.4.2.2 CompositeDragger . . . 22
4.4.2.3 ∼CompositeDragger . . . 22
4.4.3 Member Function Documentation . . . 22
4.4.3.1 addDragger . . . 22
4.4.3.2 containsDragger . . . 22
4.4.3.3 findDragger . . . 22
4.4.3.4 getComposite . . . 22
4.4.3.5 getComposite . . . 22
4.4.3.6 getDragger . . . 22
4.4.3.7 getDragger . . . 22
4.4.3.8 getNumDraggers . . . 22
4.4.3.9 handle . . . 22
4.4.3.10 removeDragger . . . 22
4.4.3.11 setParentDragger . . . 22
4.4.4 Member Data Documentation . . . 22
4.4.4.1 _draggerList . . . 22
4.5 Constraint Class Reference . . . 23
4.5.1 Constructor & Destructor Documentation . . . 23
4.5.1.1 Constraint . . . 23

4.5.1.2 ~Constraint . . . 23
4.5.2 Member Function Documentation . . . 23
4.5.2.1 computeLocalToWorldAndWorldToLocal . . . 23
4.5.2.2 constrain . . . 23
4.5.2.3 constrain . . . 23
4.5.2.4 constrain . . . 23
4.5.2.5 constrain . . . 23
4.5.2.6 constrain . . . 23
4.5.2.7 constrain . . . 24
4.5.2.8 getLocalToWorld . . . 24
4.5.2.9 getReferenceNode . . . 24
4.5.2.10 getReferenceNode . . . 24
4.5.2.11 getWorldToLocal . . . 24
4.6 CylinderPlaneProjector Class Reference . . . 25
4.6.1 Detailed Description . . . 26
4.6.2 Constructor & Destructor Documentation . . . 26
4.6.2.1 CylinderPlaneProjector . . . 26
4.6.2.2 CylinderPlaneProjector . . . 26
4.6.2.3 ~CylinderPlaneProjector . . . 26
4.6.3 Member Function Documentation . . . 26
4.6.3.1 getRotation . . . 26
4.6.3.2 isProjectionOnCylinder . . . 26
4.6.3.3 project . . . 26
4.6.4 Member Data Documentation . . . 26
4.6.4.1 _onCylinder . . . 26
4.6.4.2 _plane . . . 26
4.6.4.3 _planeLineEnd . . . 26
4.6.4.4 _planeLineStart . . . 26
4.7 CylinderProjector Class Reference . . . 27
4.7.1 Detailed Description . . . 28
4.7.2 Constructor & Destructor Documentation . . . 28
4.7.2.1 CylinderProjector . . . 28
4.7.2.2 CylinderProjector . . . 28
4.7.2.3 ~CylinderProjector . . . 28
4.7.3 Member Function Documentation . . . 28
4.7.3.1 getCylinder . . . 28
4.7.3.2 isPointInFront . . . 28
4.7.3.3 project . . . 28
4.7.3.4 setCylinder . . . 28
4.7.3.5 setFront . . . 28
4.7.4 Member Data Documentation . . . 28

4.7.4.1 _cylinder . . . 28
4.7.4.2 _cylinderAxis . . . 28
4.7.4.3 _front . . . 28
4.8 Dragger Class Reference . . . 29
4.8.1 Detailed Description . . . 30
4.8.2 Member Typedef Documentation . . . 31
4.8.2.1 Constraints . . . 31
4.8.2.2 DraggerCallbacks . . . 31
4.8.3 Constructor & Destructor Documentation . . . 31
4.8.3.1 Dragger . . . 31
4.8.3.2 Dragger . . . 31
4.8.3.3 ∼Dragger . . . 31
4.8.4 Member Function Documentation . . . 31
4.8.4.1 addConstraint . . . 31
4.8.4.2 addDraggerCallback . . . 31
4.8.4.3 addTransformUpdating . . . 31
4.8.4.4 dispatch . . . 31
4.8.4.5 getActivationKeyEvent . . . 31
4.8.4.6 getActivationModKeyMask . . . 31
4.8.4.7 getComposite . . . 31
4.8.4.8 getComposite . . . 31
4.8.4.9 getConstraints . . . 31
4.8.4.10 getConstraints . . . 31
4.8.4.11 getDraggerActive . . . 31
4.8.4.12 getDraggerCallbacks . . . 31
4.8.4.13 getDraggerCallbacks . . . 31
4.8.4.14 getHandleEvents . . . 31
4.8.4.15 getParentDragger . . . 31
4.8.4.16 getParentDragger . . . 31
4.8.4.17 handle . . . 31
4.8.4.18 handle . . . 31
4.8.4.19 META_Node . . . 31
4.8.4.20 receive . . . 32
4.8.4.21 removeConstraint . . . 32
4.8.4.22 removeDraggerCallback . . . 32
4.8.4.23 removeTransformUpdating . . . 32
4.8.4.24 setActivationKeyEvent . . . 32
4.8.4.25 setActivationModKeyMask . . . 32
4.8.4.26 setDraggerActive . . . 32
4.8.4.27 setHandleEvents . . . 32
4.8.4.28 setupDefaultGeometry . . . 32

4.8.4.29 traverse . . . 32
4.8.5 Member Data Documentation . . . 32
4.8.5.1 _activationKeyEvent . . . 32
4.8.5.2 _activationModKeyMask . . . 32
4.8.5.3 _activationPermittedByKeyEvent . . . 32
4.8.5.4 _activationPermittedByModKeyMask . . . 32
4.8.5.5 _constraints . . . 32
4.8.5.6 _draggerActive . . . 32
4.8.5.7 _draggerCallbacks . . . 32
4.8.5.8 _handleEvents . . . 32
4.8.5.9 _parentDragger . . . 32
4.8.5.10 _pointer . . . 32
4.8.5.11 _selfUpdater . . . 32
4.9 DraggerCallback Class Reference . . . 33
4.9.1 Constructor & Destructor Documentation . . . 33
4.9.1.1 DraggerCallback . . . 33
4.9.1.2 DraggerCallback . . . 33
4.9.2 Member Function Documentation . . . 33
4.9.2.1 META_Object . . . 33
4.9.2.2 receive . . . 33
4.9.2.3 receive . . . 33
4.9.2.4 receive . . . 33
4.9.2.5 receive . . . 33
4.9.2.6 receive . . . 33
4.9.2.7 receive . . . 33
4.9.2.8 receive . . . 33
4.10 DraggerTransformCallback Class Reference . . . 34
4.10.1 Constructor & Destructor Documentation . . . 34
4.10.1.1 DraggerTransformCallback . . . 34
4.10.2 Member Function Documentation . . . 34
4.10.2.1 getTransform . . . 34
4.10.2.2 getTransform . . . 34
4.10.2.3 receive . . . 34
4.10.3 Member Data Documentation . . . 35
4.10.3.1 _localToWorld . . . 35
4.10.3.2 _startMotionMatrix . . . 35
4.10.3.3 _transform . . . 35
4.10.3.4 _worldToLocal . . . 35
4.11 ForceCullCallback Class Reference . . . 36
4.11.1 Member Function Documentation . . . 36
4.11.1.1 cull . . . 36

4.12 GridConstraint Class Reference . . . 37
4.12.1 Detailed Description . . . 37
4.12.2 Constructor & Destructor Documentation . . . 37
4.12.2.1 GridConstraint . . . 37
4.12.2.2 ∼GridConstraint . . . 37
4.12.3 Member Function Documentation . . . 37
4.12.3.1 constrain . . . 37
4.12.3.2 constrain . . . 38
4.12.3.3 constrain . . . 38
4.12.3.4 constrain . . . 38
4.12.3.5 constrain . . . 38
4.12.3.6 getOrigin . . . 38
4.12.3.7 getSpacing . . . 38
4.12.3.8 setOrigin . . . 38
4.12.3.9 setSpacing . . . 38
4.13 LineProjector Class Reference . . . 39
4.13.1 Detailed Description . . . 39
4.13.2 Constructor & Destructor Documentation . . . 40
4.13.2.1 LineProjector . . . 40
4.13.2.2 LineProjector . . . 40
4.13.2.3 ∼LineProjector . . . 40
4.13.3 Member Function Documentation . . . 40
4.13.3.1 getLineEnd . . . 40
4.13.3.2 getLineEnd . . . 40
4.13.3.3 getLineStart . . . 40
4.13.3.4 getLineStart . . . 40
4.13.3.5 project . . . 40
4.13.3.6 setLine . . . 40
4.13.4 Member Data Documentation . . . 40
4.13.4.1 _line . . . 40
4.14 MotionCommand Class Reference . . . 41
4.14.1 Detailed Description . . . 42
4.14.2 Member Enumeration Documentation . . . 42
4.14.2.1 Stage . . . 42
4.14.3 Constructor & Destructor Documentation . . . 42
4.14.3.1 MotionCommand . . . 42
4.14.3.2 ∼MotionCommand . . . 42
4.14.4 Member Function Documentation . . . 42
4.14.4.1 createCommandInverse . . . 42
4.14.4.2 getLocalToWorld . . . 42
4.14.4.3 getMotionMatrix . . . 42

4.14.4.4 getStage . . . 42
4.14.4.5 getWorldToLocal . . . 42
4.14.4.6 setLocalToWorldAndWorldToLocal . . . 42
4.14.4.7 setStage . . . 42
4.15 PlaneProjector Class Reference . . . 43
4.15.1 Detailed Description . . . 43
4.15.2 Constructor & Destructor Documentation . . . 44
4.15.2.1 PlaneProjector . . . 44
4.15.2.2 PlaneProjector . . . 44
4.15.2.3 ∼PlaneProjector . . . 44
4.15.3 Member Function Documentation . . . 44
4.15.3.1 getPlane . . . 44
4.15.3.2 project . . . 44
4.15.3.3 setPlane . . . 44
4.15.4 Member Data Documentation . . . 44
4.15.4.1 _plane . . . 44
4.16 PointerInfo Class Reference . . . 45
4.16.1 Member Typedef Documentation . . . 46
4.16.1.1 IntersectionList . . . 46
4.16.1.2 NodePathIntersectionPair . . . 46
4.16.2 Constructor & Destructor Documentation . . . 46
4.16.2.1 PointerInfo . . . 46
4.16.2.2 PointerInfo . . . 46
4.16.3 Member Function Documentation . . . 46
4.16.3.1 addIntersection . . . 46
4.16.3.2 completed . . . 46
4.16.3.3 contains . . . 46
4.16.3.4 getEyeDir . . . 46
4.16.3.5 getLocalIntersectPoint . . . 46
4.16.3.6 getNearFarPoints . . . 46
4.16.3.7 next . . . 46
4.16.3.8 projectWindowXYIntoObject . . . 46
4.16.3.9 reset . . . 46
4.16.3.10 setCamera . . . 46
4.16.3.11 setMousePosition . . . 46
4.16.3.12 setNearFarPoints . . . 46
4.16.4 Member Data Documentation . . . 46
4.16.4.1 _eyeDir . . . 46
4.16.4.2 _farPoint . . . 46
4.16.4.3 _hitIter . . . 46
4.16.4.4 _hitList . . . 46

4.16.4.5 _inverseMVPW . . . 46
4.16.4.6 _MVPW . . . 46
4.16.4.7 _nearPoint . . . 46
4.17 Projector Class Reference . . . 47
4.17.1 Detailed Description . . . 47
4.17.2 Constructor & Destructor Documentation . . . 47
4.17.2.1 Projector . . . 47
4.17.2.2 ∼Projector . . . 47
4.17.3 Member Function Documentation . . . 47
4.17.3.1 getLocalToWorld . . . 47
4.17.3.2 getWorldToLocal . . . 48
4.17.3.3 project . . . 48
4.17.3.4 setLocalToWorld . . . 48
4.17.4 Member Data Documentation . . . 48
4.17.4.1 _localToWorld . . . 48
4.17.4.2 _worldToLocal . . . 48
4.17.4.3 _worldToLocalDirty . . . 48
4.18 Rotate3DCommand Class Reference . . . 49
4.18.1 Detailed Description . . . 49
4.18.2 Constructor & Destructor Documentation . . . 49
4.18.2.1 Rotate3DCommand . . . 49
4.18.2.2 ∼Rotate3DCommand . . . 49
4.18.3 Member Function Documentation . . . 49
4.18.3.1 createCommandInverse . . . 49
4.18.3.2 getMotionMatrix . . . 50
4.18.3.3 getRotation . . . 50
4.18.3.4 setRotation . . . 50
4.19 RotateCylinderDragger Class Reference . . . 51
4.19.1 Detailed Description . . . 52
4.19.2 Constructor & Destructor Documentation . . . 52
4.19.2.1 RotateCylinderDragger . . . 52
4.19.2.2 ∼RotateCylinderDragger . . . 52
4.19.3 Member Function Documentation . . . 52
4.19.3.1 getColor . . . 52
4.19.3.2 getPickColor . . . 52
4.19.3.3 META_OSGMANIPULATOR_Object . . . 52
4.19.3.4 setColor . . . 52
4.19.3.5 setPickColor . . . 52
4.19.3.6 setupDefaultGeometry . . . 52
4.19.4 Member Data Documentation . . . 52
4.19.4.1 _color . . . 52

4.19.4.2 _pickColor . . . 52
4.19.4.3 _prevPtOnCylinder . . . 52
4.19.4.4 _prevRotation . . . 52
4.19.4.5 _prevWorldProjPt . . . 52
4.19.4.6 _projector . . . 52
4.19.4.7 _startLocalToWorld . . . 52
4.19.4.8 _startWorldToLocal . . . 52
4.19.4.9 ea . . . 52
4.19.4.10 us . . . 52
4.20 RotateSphereDragger Class Reference . . . 53
4.20.1 Detailed Description . . . 54
4.20.2 Constructor & Destructor Documentation . . . 54
4.20.2.1 RotateSphereDragger . . . 54
4.20.2.2 ∼RotateSphereDragger . . . 54
4.20.3 Member Function Documentation . . . 54
4.20.3.1 getColor . . . 54
4.20.3.2 getPickColor . . . 54
4.20.3.3 META_OSGMANIPULATOR_Object . . . 54
4.20.3.4 setColor . . . 54
4.20.3.5 setPickColor . . . 54
4.20.3.6 setupDefaultGeometry . . . 54
4.20.4 Member Data Documentation . . . 54
4.20.4.1 _color . . . 54
4.20.4.2 _pickColor . . . 54
4.20.4.3 _prevPtOnSphere . . . 54
4.20.4.4 _prevRotation . . . 54
4.20.4.5 _prevWorldProjPt . . . 54
4.20.4.6 _projector . . . 54
4.20.4.7 _startLocalToWorld . . . 54
4.20.4.8 _startWorldToLocal . . . 54
4.20.4.9 ea . . . 54
4.20.4.10 us . . . 54
4.21 Scale1DCommand Class Reference . . . 55
4.21.1 Detailed Description . . . 55
4.21.2 Constructor & Destructor Documentation . . . 56
4.21.2.1 Scale1DCommand . . . 56
4.21.2.2 ∼Scale1DCommand . . . 56
4.21.3 Member Function Documentation . . . 56
4.21.3.1 createCommandInverse . . . 56
4.21.3.2 getMinScale . . . 56
4.21.3.3 getMotionMatrix . . . 56

4.21.3.4 getReferencePoint . . . 56
4.21.3.5 getScale . . . 56
4.21.3.6 getScaleCenter . . . 56
4.21.3.7 setMinScale . . . 56
4.21.3.8 setReferencePoint . . . 56
4.21.3.9 setScale . . . 56
4.21.3.10 setScaleCenter . . . 56
4.22 Scale1DDragger Class Reference . . . 57
4.22.1 Detailed Description . . . 58
4.22.2 Member Enumeration Documentation . . . 58
4.22.2.1 ScaleMode . . . 58
4.22.3 Constructor & Destructor Documentation . . . 58
4.22.3.1 Scale1DDragger . . . 58
4.22.3.2 ∼Scale1DDragger . . . 58
4.22.4 Member Function Documentation . . . 58
4.22.4.1 getColor . . . 58
4.22.4.2 getLeftHandleNode . . . 58
4.22.4.3 getLeftHandlePosition . . . 58
4.22.4.4 getMinScale . . . 58
4.22.4.5 getPickColor . . . 58
4.22.4.6 getRightHandleNode . . . 58
4.22.4.7 getRightHandlePosition . . . 58
4.22.4.8 META_OSGMANIPULATOR_Object . . . 58
4.22.4.9 setColor . . . 59
4.22.4.10 setLeftHandleNode . . . 59
4.22.4.11 setLeftHandlePosition . . . 59
4.22.4.12 setMinScale . . . 59
4.22.4.13 setPickColor . . . 59
4.22.4.14 setRightHandleNode . . . 59
4.22.4.15 setRightHandlePosition . . . 59
4.22.4.16 setupDefaultGeometry . . . 59
4.22.5 Member Data Documentation . . . 59
4.22.5.1 _color . . . 59
4.22.5.2 _leftHandleNode . . . 59
4.22.5.3 _minScale . . . 59
4.22.5.4 _pickColor . . . 59
4.22.5.5 _projector . . . 59
4.22.5.6 _rightHandleNode . . . 59
4.22.5.7 _scaleCenter . . . 59
4.22.5.8 _scaleMode . . . 59
4.22.5.9 _startProjectedPoint . . . 59

4.22.5.10 ea . . . 59
4.22.5.11 us . . . 59
4.23 Scale2DCommand Class Reference . . . 60
4.23.1 Detailed Description . . . 60
4.23.2 Constructor & Destructor Documentation . . . 61
4.23.2.1 Scale2DCommand . . . 61
4.23.2.2 ∼Scale2DCommand . . . 61
4.23.3 Member Function Documentation . . . 61
4.23.3.1 createCommandInverse . . . 61
4.23.3.2 getMinScale . . . 61
4.23.3.3 getMotionMatrix . . . 61
4.23.3.4 getReferencePoint . . . 61
4.23.3.5 getScale . . . 61
4.23.3.6 getScaleCenter . . . 61
4.23.3.7 setMinScale . . . 61
4.23.3.8 setReferencePoint . . . 61
4.23.3.9 setScale . . . 61
4.23.3.10 setScaleCenter . . . 61
4.24 Scale2DDragger Class Reference . . . 62
4.24.1 Detailed Description . . . 63
4.24.2 Member Enumeration Documentation . . . 63
4.24.2.1 ScaleMode . . . 63
4.24.3 Constructor & Destructor Documentation . . . 64
4.24.3.1 Scale2DDragger . . . 64
4.24.3.2 ∼Scale2DDragger . . . 64
4.24.4 Member Function Documentation . . . 64
4.24.4.1 getBottomLeftHandleNode . . . 64
4.24.4.2 getBottomLeftHandlePosition . . . 64
4.24.4.3 getBottomRightHandleNode . . . 64
4.24.4.4 getBottomRightHandlePosition . . . 64
4.24.4.5 getColor . . . 64
4.24.4.6 getMinScale . . . 64
4.24.4.7 getPickColor . . . 64
4.24.4.8 getTopLeftHandleNode . . . 64
4.24.4.9 getTopLeftHandlePosition . . . 64
4.24.4.10 getTopRightHandleNode . . . 64
4.24.4.11 getTopRightHandlePosition . . . 64
4.24.4.12 META_OSGMANIPULATOR_Object . . . 64
4.24.4.13 setBottomLeftHandleNode . . . 64
4.24.4.14 setBottomLeftHandlePosition . . . 64
4.24.4.15 setBottomRightHandleNode . . . 64

4.24.4.16 setBottomRightHandlePosition . . . 64
4.24.4.17 setColor . . . 64
4.24.4.18 setMinScale . . . 64
4.24.4.19 setPickColor . . . 64
4.24.4.20 setTopLeftHandleNode . . . 64
4.24.4.21 setTopLeftHandlePosition . . . 64
4.24.4.22 setTopRightHandleNode . . . 64
4.24.4.23 setTopRightHandlePosition . . . 64
4.24.4.24 setupDefaultGeometry . . . 64
4.24.5 Member Data Documentation . . . 65
4.24.5.1 _bottomLeftHandleNode . . . 65
4.24.5.2 _bottomLeftHandlePosition . . . 65
4.24.5.3 _bottomRightHandleNode . . . 65
4.24.5.4 _bottomRightHandlePosition . . . 65
4.24.5.5 _color . . . 65
4.24.5.6 _minScale . . . 65
4.24.5.7 _pickColor . . . 65
4.24.5.8 _projector . . . 65
4.24.5.9 _referencePoint . . . 65
4.24.5.10 _scaleCenter . . . 65
4.24.5.11 _scaleMode . . . 65
4.24.5.12 _startProjectedPoint . . . 65
4.24.5.13 _topLeftHandleNode . . . 65
4.24.5.14 _topLeftHandlePosition . . . 65
4.24.5.15 _topRightHandleNode . . . 65
4.24.5.16 _topRightHandlePosition . . . 65
4.24.5.17 ea . . . 65
4.24.5.18 us . . . 65
4.25 ScaleAxisDragger Class Reference . . . 66
4.25.1 Detailed Description . . . 66
4.25.2 Constructor & Destructor Documentation . . . 66
4.25.2.1 ScaleAxisDragger . . . 66
4.25.2.2 ∼ScaleAxisDragger . . . 66
4.25.3 Member Function Documentation . . . 66
4.25.3.1 META_OSGMANIPULATOR_Object . . . 66
4.25.4 Member Data Documentation . . . 67
4.25.4.1 _xDragger . . . 67
4.25.4.2 _yDragger . . . 67
4.25.4.3 _zDragger . . . 67
4.26 ScaleUniformCommand Class Reference . . . 68
4.26.1 Detailed Description . . . 68

4.26.2 Constructor & Destructor Documentation . . . 69
4.26.2.1 ScaleUniformCommand . . . 69
4.26.2.2 ∼ScaleUniformCommand . . . 69
4.26.3 Member Function Documentation . . . 69
4.26.3.1 createCommandInverse . . . 69
4.26.3.2 getMotionMatrix . . . 69
4.26.3.3 getScale . . . 69
4.26.3.4 getScaleCenter . . . 69
4.26.3.5 setScale . . . 69
4.26.3.6 setScaleCenter . . . 69
4.27 SpherePlaneProjector Class Reference . . . 70
4.27.1 Detailed Description . . . 71
4.27.2 Constructor & Destructor Documentation . . . 71
4.27.2.1 SpherePlaneProjector . . . 71
4.27.2.2 SpherePlaneProjector . . . 71
4.27.2.3 ∼SpherePlaneProjector . . . 71
4.27.3 Member Function Documentation . . . 71
4.27.3.1 getRotation . . . 71
4.27.3.2 isProjectionOnSphere . . . 71
4.27.3.3 project . . . 71
4.27.4 Member Data Documentation . . . 71
4.27.4.1 _onSphere . . . 71
4.27.4.2 _plane . . . 71
4.28 SphereProjector Class Reference . . . 72
4.28.1 Detailed Description . . . 73
4.28.2 Constructor & Destructor Documentation . . . 73
4.28.2.1 SphereProjector . . . 73
4.28.2.2 SphereProjector . . . 73
4.28.2.3 ∼SphereProjector . . . 73
4.28.3 Member Function Documentation . . . 73
4.28.3.1 getSphere . . . 73
4.28.3.2 isPointInFront . . . 73
4.28.3.3 project . . . 73
4.28.3.4 setFront . . . 73
4.28.3.5 setSphere . . . 73
4.28.4 Member Data Documentation . . . 73
4.28.4.1 _front . . . 73
4.28.4.2 _sphere . . . 73
4.29 TabBoxDragger Class Reference . . . 74
4.29.1 Detailed Description . . . 74
4.29.2 Constructor & Destructor Documentation . . . 74

4.29.2.1 TabBoxDragger . . . 74
4.29.2.2 ~TabBoxDragger . . . 74
4.29.3 Member Function Documentation . . . 74
4.29.3.1 META_OSGMANIPULATOR_Object . . . 74
4.29.3.2 setPlaneColor . . . 75
4.29.4 Member Data Documentation . . . 75
4.29.4.1 _planeDraggers . . . 75
4.30 TabBoxTrackballDragger Class Reference . . . 76
4.30.1 Detailed Description . . . 76
4.30.2 Constructor & Destructor Documentation . . . 76
4.30.2.1 TabBoxTrackballDragger . . . 76
4.30.2.2 ~TabBoxTrackballDragger . . . 76
4.30.3 Member Function Documentation . . . 76
4.30.3.1 META_OSGMANIPULATOR_Object . . . 76
4.30.4 Member Data Documentation . . . 77
4.30.4.1 _tabBoxDragger . . . 77
4.30.4.2 _trackballDragger . . . 77
4.31 TabPlaneDragger Class Reference . . . 78
4.31.1 Detailed Description . . . 79
4.31.2 Constructor & Destructor Documentation . . . 79
4.31.2.1 TabPlaneDragger . . . 79
4.31.2.2 ~TabPlaneDragger . . . 79
4.31.3 Member Function Documentation . . . 79
4.31.3.1 META_OSGMANIPULATOR_Object . . . 79
4.31.3.2 setPlaneColor . . . 79
4.31.3.3 setupDefaultGeometry . . . 79
4.31.4 Member Data Documentation . . . 79
4.31.4.1 _cornerScaleDragger . . . 79
4.31.4.2 _handleScaleFactor . . . 79
4.31.4.3 _horzEdgeScaleDragger . . . 79
4.31.4.4 _translateDragger . . . 79
4.31.4.5 _vertEdgeScaleDragger . . . 79
4.31.4.6 ea . . . 79
4.31.4.7 us . . . 79
4.32 TabPlaneTrackballDragger Class Reference . . . 80
4.32.1 Detailed Description . . . 80
4.32.2 Constructor & Destructor Documentation . . . 80
4.32.2.1 TabPlaneTrackballDragger . . . 80
4.32.2.2 ~TabPlaneTrackballDragger . . . 80
4.32.3 Member Function Documentation . . . 80
4.32.3.1 META_OSGMANIPULATOR_Object . . . 80

4.32.3.2 setPlaneColor . . . 81
4.32.4 Member Data Documentation . . . 81
4.32.4.1 _tabPlaneDragger . . . 81
4.32.4.2 _trackballDragger . . . 81
4.33 TrackballDragger Class Reference . . . 82
4.33.1 Detailed Description . . . 82
4.33.2 Constructor & Destructor Documentation . . . 83
4.33.2.1 TrackballDragger . . . 83
4.33.2.2 ∼TrackballDragger . . . 83
4.33.3 Member Function Documentation . . . 83
4.33.3.1 META_OSGMANIPULATOR_Object . . . 83
4.33.4 Member Data Documentation . . . 83
4.33.4.1 _xDragger . . . 83
4.33.4.2 _xyzDragger . . . 83
4.33.4.3 _yDragger . . . 83
4.33.4.4 _zDragger . . . 83
4.34 Translate1DDragger Class Reference . . . 84
4.34.1 Detailed Description . . . 84
4.34.2 Constructor & Destructor Documentation . . . 85
4.34.2.1 Translate1DDragger . . . 85
4.34.2.2 ∼Translate1DDragger . . . 85
4.34.3 Member Function Documentation . . . 85
4.34.3.1 getColor . . . 85
4.34.3.2 getPickColor . . . 85
4.34.3.3 setCheckForNodeInNodePath . . . 85
4.34.3.4 setColor . . . 85
4.34.3.5 setPickColor . . . 85
4.34.3.6 setupDefaultGeometry . . . 85
4.34.4 Member Data Documentation . . . 85
4.34.4.1 _checkForNodeInNodePath . . . 85
4.34.4.2 _color . . . 85
4.34.4.3 _pickColor . . . 85
4.34.4.4 _projector . . . 85
4.34.4.5 _startProjectedPoint . . . 85
4.35 Translate2DDragger Class Reference . . . 86
4.35.1 Detailed Description . . . 87
4.35.2 Constructor & Destructor Documentation . . . 87
4.35.2.1 Translate2DDragger . . . 87
4.35.2.2 Translate2DDragger . . . 87
4.35.2.3 ∼Translate2DDragger . . . 87
4.35.3 Member Function Documentation . . . 87

4.35.3.1 getColor . . . 87
4.35.3.2 getPickColor . . . 87
4.35.3.3 META_OSGMANIPULATOR_Object . . . 87
4.35.3.4 setColor . . . 87
4.35.3.5 setPickColor . . . 87
4.35.3.6 setupDefaultGeometry . . . 87
4.35.4 Member Data Documentation . . . 87
4.35.4.1 _color . . . 87
4.35.4.2 _pickColor . . . 87
4.35.4.3 _polygonOffset . . . 87
4.35.4.4 _projector . . . 87
4.35.4.5 _startProjectedPoint . . . 87
4.35.4.6 ea . . . 87
4.35.4.7 us . . . 87
4.36 TranslateAxisDragger Class Reference . . . 88
4.36.1 Detailed Description . . . 88
4.36.2 Constructor & Destructor Documentation . . . 88
4.36.2.1 TranslateAxisDragger . . . 88
4.36.2.2 ∼TranslateAxisDragger . . . 88
4.36.3 Member Function Documentation . . . 88
4.36.3.1 META_OSGMANIPULATOR_Object . . . 88
4.36.4 Member Data Documentation . . . 89
4.36.4.1 _xDragger . . . 89
4.36.4.2 _yDragger . . . 89
4.36.4.3 _zDragger . . . 89
4.37 TranslateInLineCommand Class Reference . . . 90
4.37.1 Detailed Description . . . 90
4.37.2 Constructor & Destructor Documentation . . . 91
4.37.2.1 TranslateInLineCommand . . . 91
4.37.2.2 TranslateInLineCommand . . . 91
4.37.2.3 ∼TranslateInLineCommand . . . 91
4.37.3 Member Function Documentation . . . 91
4.37.3.1 createCommandInverse . . . 91
4.37.3.2 getLineEnd . . . 91
4.37.3.3 getLineStart . . . 91
4.37.3.4 getMotionMatrix . . . 91
4.37.3.5 getTranslation . . . 91
4.37.3.6 setLine . . . 91
4.37.3.7 setTranslation . . . 91
4.38 TranslateInPlaneCommand Class Reference . . . 92
4.38.1 Detailed Description . . . 92

4.38.2 Constructor & Destructor Documentation . . . 93
4.38.2.1 TranslateInPlaneCommand . . . 93
4.38.2.2 TranslateInPlaneCommand . . . 93
4.38.2.3 ∼TranslateInPlaneCommand . . . 93
4.38.3 Member Function Documentation . . . 93
4.38.3.1 createCommandInverse . . . 93
4.38.3.2 getMotionMatrix . . . 93
4.38.3.3 getPlane . . . 93
4.38.3.4 getReferencePoint . . . 93
4.38.3.5 getTranslation . . . 93
4.38.3.6 setPlane . . . 93
4.38.3.7 setReferencePoint . . . 93
4.38.3.8 setTranslation . . . 93
4.39 TranslatePlaneDragger Class Reference . . . 94
4.39.1 Detailed Description . . . 95
4.39.2 Constructor & Destructor Documentation . . . 95
4.39.2.1 TranslatePlaneDragger . . . 95
4.39.2.2 ∼TranslatePlaneDragger . . . 95
4.39.3 Member Function Documentation . . . 95
4.39.3.1 getTranslate1DDragger . . . 95
4.39.3.2 getTranslate2DDragger . . . 95
4.39.3.3 META_OSGMANIPULATOR_Object . . . 95
4.39.3.4 setColor . . . 95
4.39.3.5 setupDefaultGeometry . . . 95
4.39.4 Member Data Documentation . . . 95
4.39.4.1 _translate1DDragger . . . 95
4.39.4.2 _translate2DDragger . . . 95
4.39.4.3 _usingTranslate1DDragger . . . 95
4.39.4.4 ea . . . 95
4.39.4.5 us . . . 95

5 File Documentation **97**

5.1 AntiSquish File Reference . . . 97
5.1.1 Define Documentation . . . 98
5.1.1.1 _OSG_ANTISQUISH_ . . . 98
5.2 AntiSquish.cpp File Reference . . . 99
5.3 Command File Reference . . . 100
5.3.1 Define Documentation . . . 100
5.3.1.1 OSGMANIPULATOR_COMMAND . . . 100
5.4 Command.cpp File Reference . . . 101
5.5 CommandManager File Reference . . . 102

5.5.1 Define Documentation . . . 102
5.5.1.1 OSGMANIPULATOR_COMMANDMANAGER . . . 102
5.6 Constraint File Reference . . . 103
5.6.1 Define Documentation . . . 103
5.6.1.1 OSGMANIPULATOR_CONSTRAINT . . . 103
5.7 Constraint.cpp File Reference . . . 104
5.8 Dragger File Reference . . . 105
5.8.1 Define Documentation . . . 105
5.8.1.1 OSGMANIPULATOR_DRAGGER . . . 105
5.9 Dragger.cpp File Reference . . . 106
5.10 Export File Reference . . . 107
5.10.1 Define Documentation . . . 107
5.10.1.1 META_OSGMANIPULATOR_Object . . . 107
5.10.1.2 OSGMANIPULATOR_EXPORT . . . 107
5.10.1.3 OSGMANIPULATOR_EXPORT_ . . . 107
5.11 mainpage.h File Reference . . . 108
5.11.1 Detailed Description . . . 108
5.12 Projector File Reference . . . 109
5.12.1 Define Documentation . . . 110
5.12.1.1 OSGMANIPULATOR_PROJECTOR . . . 110
5.13 Projector.cpp File Reference . . . 111
5.14 RotateCylinderDragger File Reference . . . 112
5.14.1 Define Documentation . . . 112
5.14.1.1 OSGMANIPULATOR_ROTATECYLINDERDRAGGER . . . 112
5.15 RotateCylinderDragger.cpp File Reference . . . 113
5.16 RotateSphereDragger File Reference . . . 114
5.16.1 Define Documentation . . . 114
5.16.1.1 OSGMANIPULATOR_ROTATESPHEREDRAGGER . . . 114
5.17 RotateSphereDragger.cpp File Reference . . . 115
5.18 Scale1DDragger File Reference . . . 116
5.18.1 Define Documentation . . . 116
5.18.1.1 OSGMANIPULATOR_SCALE1DDRAGGER . . . 116
5.19 Scale1DDragger.cpp File Reference . . . 117
5.20 Scale2DDragger File Reference . . . 118
5.20.1 Define Documentation . . . 118
5.20.1.1 OSGMANIPULATOR_SCALE2DDRAGGER . . . 118
5.21 Scale2DDragger.cpp File Reference . . . 119
5.22 ScaleAxisDragger File Reference . . . 120
5.22.1 Define Documentation . . . 120
5.22.1.1 OSGMANIPULATOR_SCALEAXISDRAGGER . . . 120
5.23 ScaleAxisDragger.cpp File Reference . . . 121

5.24 Selection File Reference . . . 122
5.24.1 Define Documentation . . . 122
5.24.1.1 OSGMANIPULATOR_SELECTION . . . 122
5.25 TabBoxDragger File Reference . . . 123
5.25.1 Define Documentation . . . 123
5.25.1.1 OSGMANIPULATOR_TABBOXDRAGGER . . . 123
5.26 TabBoxDragger.cpp File Reference . . . 124
5.27 TabBoxTrackballDragger File Reference . . . 125
5.27.1 Define Documentation . . . 125
5.27.1.1 OSGMANIPULATOR_TABBOXTRACKBALLDRAGGER . . . 125
5.28 TabBoxTrackballDragger.cpp File Reference . . . 126
5.29 TabPlaneDragger File Reference . . . 127
5.29.1 Define Documentation . . . 127
5.29.1.1 OSGMANIPULATOR_TABPLANEDRAGGER . . . 127
5.30 TabPlaneDragger.cpp File Reference . . . 128
5.31 TabPlaneTrackballDragger File Reference . . . 129
5.31.1 Define Documentation . . . 129
5.31.1.1 OSGMANIPULATOR_TABPLANETRACKBALLDRAGGER . . . 129
5.32 TabPlaneTrackballDragger.cpp File Reference . . . 130
5.33 TrackballDragger File Reference . . . 131
5.33.1 Define Documentation . . . 131
5.33.1.1 OSGMANIPULATOR_TRACKBALLDRAGGER . . . 131
5.34 TrackballDragger.cpp File Reference . . . 132
5.35 Translate1DDragger File Reference . . . 133
5.35.1 Define Documentation . . . 133
5.35.1.1 OSGMANIPULATOR_TRANSLATE1DDRAGGER . . . 133
5.36 Translate1DDragger.cpp File Reference . . . 134
5.37 Translate2DDragger File Reference . . . 135
5.37.1 Define Documentation . . . 135
5.37.1.1 OSGMANIPULATOR_TRANSLATE2DDRAGGER . . . 135
5.38 Translate2DDragger.cpp File Reference . . . 136
5.39 TranslateAxisDragger File Reference . . . 137
5.39.1 Define Documentation . . . 137
5.39.1.1 OSGMANIPULATOR_TRANSLATEAXISDRAGGER . . . 137
5.40 TranslateAxisDragger.cpp File Reference . . . 138
5.41 TranslatePlaneDragger File Reference . . . 139
5.41.1 Define Documentation . . . 139
5.41.1.1 OSGMANIPULATOR_TRANSLATEPLANEDRAGGER . . . 139
5.42 TranslatePlaneDragger.cpp File Reference . . . 140
5.43 Version File Reference . . . 141
5.43.1 Define Documentation . . . 141

5.43.1.1 OSGMANIPULATOR_VERSION 141
5.43.2 Function Documentation 141
5.43.2.1 osgManipulatorGetLibraryName 141
5.43.2.2 osgManipulatorGetVersion 141
5.44 Version.cpp File Reference 142
5.44.1 Function Documentation 142
5.44.1.1 osgManipulatorGetLibraryName 142
5.44.1.2 osgManipulatorGetVersion 142

Chapter 1

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.

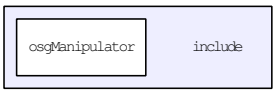
The documentation you are looking at can be downloaded from www.3draum.ch.

Enjoy!

Chapter 2

Directory Documentation

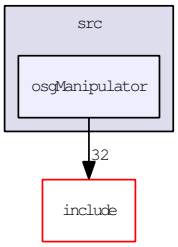
2.1 include/ Directory Reference



Directories

- directory **osgManipulator**

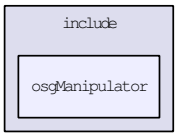
2.2 src/osgManipulator/ Directory Reference



Files

- file **AntiSquish.cpp**
- file **Command.cpp**
- file **Constraint.cpp**
- file **Dragger.cpp**
- file **Projector.cpp**
- file **RotateCylinderDragger.cpp**
- file **RotateSphereDragger.cpp**
- file **Scale1DDragger.cpp**
- file **Scale2DDragger.cpp**
- file **ScaleAxisDragger.cpp**
- file **TabBoxDragger.cpp**
- file **TabBoxTrackballDragger.cpp**
- file **TabPlaneDragger.cpp**
- file **TabPlaneTrackballDragger.cpp**
- file **TrackballDragger.cpp**
- file **Translate1DDragger.cpp**
- file **Translate2DDragger.cpp**
- file **TranslateAxisDragger.cpp**
- file **TranslatePlaneDragger.cpp**
- file **Version.cpp**

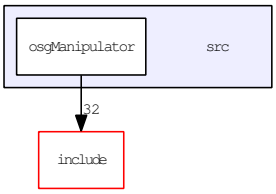
2.3 include/osgManipulator/ Directory Reference



Files

- file **AntiSquish**
- file **Command**
- file **CommandManager**
- file **Constraint**
- file **Dragger**
- file **Export**
- file **mainpage.h**
- file **Projector**
- file **RotateCylinderDragger**
- file **RotateSphereDragger**
- file **Scale1DDragger**
- file **Scale2DDragger**
- file **ScaleAxisDragger**
- file **Selection**
- file **TabBoxDragger**
- file **TabBoxTrackballDragger**
- file **TabPlaneDragger**
- file **TabPlaneTrackballDragger**
- file **TrackballDragger**
- file **Translate1DDragger**
- file **Translate2DDragger**
- file **TranslateAxisDragger**
- file **TranslatePlaneDragger**
- file **Version**

2.4 src/ Directory Reference



Directories

- directory **osgManipulator**

Chapter 3

Namespace Documentation

3.1 anonymous_namespace{AntiSquish.cpp} Namespace Reference

Classes

- class **AntiSquishCallback**

3.2 anonymous_namespace{Constraint.cpp} Namespace Reference

Functions

- double **round_to_nearest_int** (double x)
- osg::Vec3d **snap_point_to_grid** (const osg::Vec3d &point, const osg::Vec3d &origin, const osg::Vec3d &spacing)

3.2.1 Function Documentation

3.2.1.1 double anonymous_namespace{Constraint.cpp}::round_to_nearest_int (double *x*)

3.2.1.2 osg::Vec3d anonymous_namespace{Constraint.cpp}::snap_point_to_grid (const osg::Vec3d & *point*, const osg::Vec3d & *origin*, const osg::Vec3d & *spacing*)

3.3 anonymous_namespace{Projector.cpp} Namespace Reference

Functions

- bool **computeClosestPointOnLine** (const osg::Vec3d &lineStart, const osg::Vec3d &lineEnd, const osg::Vec3d &fromPoint, osg::Vec3d &closestPoint)
- bool **computeClosestPoints** (const osg::LineSegment &l1, const osg::LineSegment &l2, osg::Vec3d &p1, osg::Vec3d &p2)
- osg::Plane **computePlaneParallelToAxisAndOrientedToEye** (const osg::Vec3d &eyeDir, const osg::Matrix &localToWorld, const osg::Vec3d &axisDir, double radius, osg::Vec3d &planeLineStart, osg::Vec3d &planeLineEnd, bool front)
- osg::Plane **computePlaneThruPointAndOrientedToEye** (const osg::Vec3d &eyeDir, const osg::Matrix &localToWorld, const osg::Vec3d &point, bool front)
- bool **getCylinderLineIntersection** (const osg::Cylinder &cylinder, const osg::Vec3d &lineStart, const osg::Vec3d &lineEnd, osg::Vec3d &isectFront, osg::Vec3d &isectBack)
- osg::Vec3d **getLocalEyeDirection** (const osg::Vec3d &eyeDir, const osg::Matrix &localToWorld)
- bool **getPlaneLineIntersection** (const osg::Vec4d &plane, const osg::Vec3d &lineStart, const osg::Vec3d &lineEnd, osg::Vec3d &isect)
- bool **getSphereLineIntersection** (const osg::Sphere &sphere, const osg::Vec3d &lineStart, const osg::Vec3d &lineEnd, osg::Vec3d &frontISect, osg::Vec3d &backISect)
- bool **getUnitCylinderLineIntersection** (const osg::Vec3d &lineStart, const osg::Vec3d &lineEnd, osg::Vec3d &isectFront, osg::Vec3d &isectBack)

3.3.1 Function Documentation

3.3.1.1 bool anonymous_namespace{Projector.cpp}::computeClosestPointOnLine (const osg::Vec3d & *lineStart*, const osg::Vec3d & *lineEnd*, const osg::Vec3d & *fromPoint*, osg::Vec3d & *closestPoint*)

3.3.1.2 bool anonymous_namespace{Projector.cpp}::computeClosestPoints (const osg::LineSegment & *l1*, const osg::LineSegment & *l2*, osg::Vec3d & *p1*, osg::Vec3d & *p2*)

3.3.1.3 osg::Plane anonymous_namespace{Projector.cpp}::computePlaneParallelToAxisAndOrientedToEye (const osg::Vec3d & *eyeDir*, const osg::Matrix & *localToWorld*, const osg::Vec3d & *axisDir*, double *radius*, osg::Vec3d & *planeLineStart*, osg::Vec3d & *planeLineEnd*, bool *front*)

3.3.1.4 osg::Plane anonymous_namespace{Projector.cpp}::computePlaneThruPointAndOrientedToEye (const osg::Vec3d & *eyeDir*, const osg::Matrix & *localToWorld*, const osg::Vec3d & *point*, bool *front*)

3.3.1.5 bool anonymous_namespace{Projector.cpp}::getCylinderLineIntersection (const osg::Cylinder & *cylinder*, const osg::Vec3d & *lineStart*, const osg::Vec3d & *lineEnd*, osg::Vec3d & *isectFront*, osg::Vec3d & *isectBack*)

3.3.1.6 osg::Vec3d anonymous_namespace{Projector.cpp}::getLocalEyeDirection (const osg::Vec3d & *eyeDir*, const osg::Matrix & *localToWorld*)

3.3.1.7 bool anonymous_namespace{Projector.cpp}::getPlaneLineIntersection (const osg::Vec4d & *plane*, const osg::Vec3d & *lineStart*, const osg::Vec3d & *lineEnd*, osg::Vec3d & *isect*)

3.3.1.8 bool anonymous_namespace{Projector.cpp}::getSphereLineIntersection (const osg::Sphere & *sphere*, const osg::Vec3d & *lineStart*, const osg::Vec3d & *lineEnd*, osg::Vec3d & *frontISect*, osg::Vec3d & *backISect*)

3.3.1.9 bool anonymous_namespace{Projector.cpp}::getUnitCylinderLineIntersection (const osg::Vec3d & *lineStart*, const osg::Vec3d & *lineEnd*, osg::Vec3d & *isectFront*, osg::Vec3d & *isectBack*)

3.4 anonymous_namespace{Scale1DDragger.cpp} Namespace Reference

Functions

- double **computeScale** (const osg::Vec3d &startProjectedPoint, const osg::Vec3d &projectedPoint, double scaleCenter)

3.4.1 Function Documentation

3.4.1.1 double anonymous_namespace{Scale1DDragger.cpp}::computeScale (const osg::Vec3d & *startProjectedPoint*, const osg::Vec3d & *projectedPoint*, double *scaleCenter*)

3.5 anonymous_namespace{Scale2DDragger.cpp} Namespace Reference

Functions

- osg::Vec2d **computeScale** (const osg::Vec3d &startProjectedPoint, const osg::Vec3d &projectedPoint, const osg::Vec2d &scaleCenter)

3.5.1 Function Documentation

3.5.1.1 osg::Vec2d anonymous_namespace{Scale2DDragger.cpp}::computeScale (const osg::Vec3d & *startProjectedPoint*, const osg::Vec3d & *projectedPoint*, const osg::Vec2d & *scaleCenter*)

3.6 anonymous_namespace{TabPlaneDragger.cpp} Namespace Reference

Functions

- void **createCornerScaleDraggerGeometry** (**Scale2DDragger** ∗cornerScaleDragger, osg::Node ∗handleNode, float handleScaleFactor)
- void **createEdgeScaleDraggerGeometry** (**Scale1DDragger** ∗horzEdgeScaleDragger, **Scale1DDragger** ∗vertEdgeScaleDragger, osg::Node ∗handleNode, float handleScaleFactor)
- osg::Node ∗ **createHandleNode** (**Scale2DDragger** ∗cornerScaleDragger, float handleScaleFactor, bool twosided)
- osg::Node ∗ **createHandleScene** (const osg::Vec3 &pos, osg::Node ∗handleNode, float handleScaleFactor)
- void **createTranslateDraggerGeometry** (**Scale2DDragger** ∗cornerScaleDragger, **TranslatePlaneDragger** ∗translateDragger)

3.6.1 Function Documentation

3.6.1.1 void anonymous_namespace{TabPlaneDragger.cpp}::createCornerScaleDraggerGeometry (Scale2DDragger ∗ *cornerScaleDragger*, osg::Node ∗ *handleNode*, float *handleScaleFactor*)

3.6.1.2 void anonymous_namespace{TabPlaneDragger.cpp}::createEdgeScaleDraggerGeometry (Scale1DDragger ∗ *horzEdgeScaleDragger*, Scale1DDragger ∗ *vertEdgeScaleDragger*, osg::Node ∗ *handleNode*, float *handleScaleFactor*)

3.6.1.3 osg::Node∗ anonymous_namespace{TabPlaneDragger.cpp}::createHandleNode (Scale2DDragger ∗ *cornerScaleDragger*, float *handleScaleFactor*, bool *twosided*)

3.6.1.4 osg::Node∗ anonymous_namespace{TabPlaneDragger.cpp}::createHandleScene (const osg::Vec3 & *pos*, osg::Node ∗ *handleNode*, float *handleScaleFactor*)

3.6.1.5 void anonymous_namespace{TabPlaneDragger.cpp}::createTranslateDraggerGeometry (Scale2DDragger ∗ *cornerScaleDragger*, TranslatePlaneDragger ∗ *translateDragger*)

3.7 anonymous_namespace{TrackballDragger.cpp} Namespace Reference

Functions

- osg::Geometry * **createCircleGeometry** (float radius, unsigned int numSegments)

3.7.1 Function Documentation

3.7.1.1 osg::Geometry* anonymous_namespace{TrackballDragger.cpp}::createCircleGeometry (float *radius*, unsigned int *numSegments*)

3.8 osgManipulator Namespace Reference

The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.

Classes

- class **AntiSquish**

  Class that performs the Anti Squish by making the scaling uniform along all axes.

- class **CommandManager**

  Deprecated.

- class **CompositeDragger**

  CompositeDragger *(*p. *21) allows to create complex draggers that are composed of a hierarchy of Draggers.*

- class **Constraint**
- class **CylinderPlaneProjector**

  CylinderPlaneProjector *(*p. *25) projects points onto the given cylinder.*

- class **CylinderProjector**

  CylinderProjector *(*p. *27) projects points onto the given cylinder.*

- class **Dragger**

  Base class for draggers.

- class **DraggerCallback**
- class **DraggerTransformCallback**
- class **GridConstraint**

  Constraint *(*p. *23) to snap motion commands to a sugar cube grid.*

- class **LineProjector**

  LineProjector *(*p. *39) projects points onto the closest point on the given line.*

- class **MotionCommand**

  Base class for motion commands that are generated by draggers.

- class **PlaneProjector**

  PlaneProjector *(*p. *43) projects points onto the given line.*

- class **PointerInfo**
- class **Projector**

  Base class for Projectors.

- class **Rotate3DCommand**

  Command for rotation in 3D.

- class **RotateCylinderDragger**

  Dragger *(*p. *29) for performing 3D rotation on a cylinder.*

- class **RotateSphereDragger**

  Dragger *(*p. *29) for performing 3D rotation on a sphere.*

- class **Scale1DCommand**

  Command for 1D scaling.

- class **Scale1DDragger**

  Dragger *(*p. *29) for performing 1D scaling.*

- class **Scale2DCommand**

  Command for 2D scaling.

- class **Scale2DDragger**

  Dragger *(*p. *29) for performing 2D scaling.*

- class **ScaleAxisDragger**

  Dragger *(*p. *29) for performing scaling on all 3 axes.*

- class **ScaleUniformCommand**

  Command for uniform 3D scaling.

- class **SpherePlaneProjector**

  SpherePlaneProjector *(*p. *70) projects points onto a sphere, failing which it project onto a plane oriented to the viewing direction.*

- class **SphereProjector**

  SphereProjector *(*p. *72) projects points onto the given sphere.*

- class **TabBoxDragger**

  TabBoxDragger *(*p. *74) consists of 6 TabPlaneDraggers to form a box dragger that performs translation and scaling.*

- class **TabBoxTrackballDragger**

  Dragger *(*p. *29) for performing rotation in all axes.*

- class **TabPlaneDragger**

  Tab plane dragger consists of a plane with tabs on it's corners and edges for scaling.

- class **TabPlaneTrackballDragger**

  Dragger *(*p. *29) for performing rotation in all axes.*

- class **TrackballDragger**

  Dragger *(*p. *29) for performing rotation in all axes.*

- class **Translate1DDragger**

  Dragger *(*p. *29) for performing 1D translation.*

- class **Translate2DDragger**

  Dragger *(*p. *29) for performing 2D translation.*

- class **TranslateAxisDragger**

  Dragger *(*p. *29) for performing translation in all three axes.*

- class **TranslateInLineCommand**

  Command for translating in a line.

- class **TranslateInPlaneCommand**

  Command for translating in a plane.

- class **TranslatePlaneDragger**

  Tab plane dragger consists of a plane with tabs on it's corners and edges for scaling.

Typedefs

- typedef osg::MatrixTransform **Selection**

Functions

- OSGMANIPULATOR_EXPORT void **computeNodePathToRoot** (osg::Node &node, osg::NodePath &np)
  Computes the nodepath from the given node all the way upto the root.
- void OSGMANIPULATOR_EXPORT **setDrawableToAlwaysCull** (osg::Drawable &drawable)
  Culls the drawable all the time.
- void OSGMANIPULATOR_EXPORT **setMaterialColor** (const osg::Vec4 &color, osg::Node &node)
  Convenience function for setting the material color on a node.

3.8.1 Detailed Description

The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.

3.8.2 Typedef Documentation

3.8.2.1 typedef osg::MatrixTransform Selection

3.8.3 Function Documentation

3.8.3.1 void computeNodePathToRoot (osg::Node & *node*, osg::NodePath & *np*)

Computes the nodepath from the given node all the way upto the root.

3.8.3.2 void setDrawableToAlwaysCull (osg::Drawable & *drawable*)

Culls the drawable all the time. Used by draggers to have invisible geometry around lines and points so that they can be picked. For example, a dragger could have a line with an invisible cylinder around it to enable picking on that line.

3.8.3.3 void setMaterialColor (const osg::Vec4 & *color*, osg::Node & *node*)

Convenience function for setting the material color on a node.

Chapter 4

Class Documentation

4.1 AntiSquish Class Reference

Class that performs the Anti Squish by making the scaling uniform along all axes.

Public Member Functions

- **AntiSquish** (const **AntiSquish** &pat, const osg::CopyOp &copyop=osg::CopyOp::SHALLOW_COPY)
- **AntiSquish** (const osg::Vec3d &pivot, const osg::Vec3d &position)
- **AntiSquish** (const osg::Vec3d &pivot)
- **AntiSquish** ()
- virtual ∼**AntiSquish** ()
- virtual osg::Object ∗ **clone** (const osg::CopyOp &copyop) const
- virtual osg::Object ∗ **cloneType** () const
- osg::Matrix **computeUnSquishedMatrix** (const osg::Matrix &, bool &flag)
- const osg::Vec3d & **getPivot** ()
- const osg::Vec3d & **getPosition** ()
- virtual bool **isSameKindAs** (const osg::Object ∗obj) const
- void **setPivot** (const osg::Vec3d &pvt)
- void **setPosition** (const osg::Vec3d &pos)

Protected Attributes

- osg::NodeCallback ∗ **_asqCallback**
- osg::Matrix **_cachedLocalToWorld**
- bool **_dirty**
- osg::Vec3d **_pivot**
- osg::Vec3d **_position**
- bool **_usePivot**
- bool **_usePosition**

4.1.1 Detailed Description

Class that performs the Anti Squish by making the scaling uniform along all axes.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 AntiSquish ()

4.1.2.2 AntiSquish (const osg::Vec3d & *pivot*)

4.1.2.3 AntiSquish (const osg::Vec3d & *pivot*, const osg::Vec3d & *position*)

4.1.2.4 AntiSquish (const AntiSquish & *pat*, const osg::CopyOp & *copyop* = `osg::CopyOp::SHALLOW_COPY`)

4.1.2.5 ~AntiSquish () `[virtual]`

4.1.3 Member Function Documentation

4.1.3.1 virtual osg::Object* clone (const osg::CopyOp & *copyop*) const `[inline, virtual]`

4.1.3.2 virtual osg::Object* cloneType () const `[inline, virtual]`

4.1.3.3 osg::Matrix computeUnSquishedMatrix (const osg::Matrix & *LTW*, bool & *flag*)

4.1.3.4 const osg::Vec3d& getPivot () `[inline]`

4.1.3.5 const osg::Vec3d& getPosition () `[inline]`

4.1.3.6 virtual bool isSameKindAs (const osg::Object * *obj*) const `[inline, virtual]`

4.1.3.7 void setPivot (const osg::Vec3d & *pvt*) `[inline]`

4.1.3.8 void setPosition (const osg::Vec3d & *pos*) `[inline]`

4.1.4 Member Data Documentation

4.1.4.1 osg::NodeCallback* _asqCallback `[protected]`

4.1.4.2 osg::Matrix _cachedLocalToWorld `[protected]`

4.1.4.3 bool _dirty `[protected]`

4.1.4.4 osg::Vec3d _pivot `[protected]`

4.1.4.5 osg::Vec3d _position `[protected]`

4.1.4.6 bool _usePivot `[protected]`

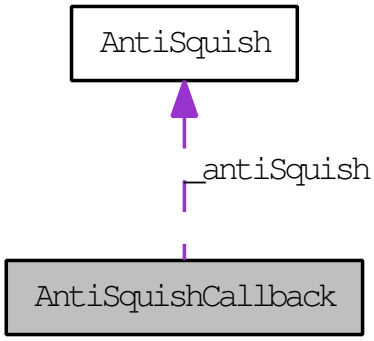
4.1.4.7 bool _usePosition `[protected]`

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

- **AntiSquish**
- **AntiSquish.cpp**

4.2 AntiSquishCallback Class Reference

Collaboration diagram for AntiSquishCallback:



Public Member Functions

- **AntiSquishCallback** (**AntiSquish** ∗asq)
- virtual ∼**AntiSquishCallback** ()
- virtual void **operator()** (osg::Node ∗, osg::NodeVisitor ∗nv)

Protected Attributes

- **AntiSquish** ∗ **_antiSquish**

4.2.1 Constructor & Destructor Documentation

4.2.1.1 AntiSquishCallback (AntiSquish ∗ *asq*) [inline]

4.2.1.2 virtual ∼AntiSquishCallback () [inline, virtual]

4.2.2 Member Function Documentation

4.2.2.1 virtual void operator() (osg::Node ∗, osg::NodeVisitor ∗ *nv*) [inline, virtual]

4.2.3 Member Data Documentation

4.2.3.1 AntiSquish∗ _antiSquish [protected]

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

- **AntiSquish.cpp**

4.3 CommandManager Class Reference

Deprecated.

Public Types

- typedef std::list< osg::ref_ptr< **Selection** > > **Selections**

Public Member Functions

- **CommandManager** ()
- bool **connect** (**Dragger** &dragger, **Constraint** &constraint)
- bool **connect** (**Dragger** &dragger, **Selection** &selection)
- bool **disconnect** (**Dragger** &dragger)
- **Selections getConnectedSelections** (**Dragger** &dragger)

Protected Member Functions

- virtual ~**CommandManager** ()

4.3.1 Detailed Description

Deprecated. **CommandManager** (p. 20) class is now no longer required as **Dragger** (p. 29) now matains all references to Constraints and Selections (now just generic MatrixTransforms). To replace **CommandManager** (p. 20) usage simple replace cmdMgr->connect(*dragger, *selection) with dragger->addTransformUpdating(selection) and cmdMgr->connect(*dragger, *selection) with dragger->addConstaint(constraint).

4.3.2 Member Typedef Documentation

4.3.2.1 typedef std::list< osg::ref_ptr<Selection> > Selections

4.3.3 Constructor & Destructor Documentation

4.3.3.1 CommandManager () [inline]

4.3.3.2 virtual ~CommandManager () [inline, protected, virtual]

4.3.4 Member Function Documentation

4.3.4.1 bool connect (Dragger & *dragger*, Constraint & *constraint*) [inline]

4.3.4.2 bool connect (Dragger & *dragger*, Selection & *selection*) [inline]

4.3.4.3 bool disconnect (Dragger & *dragger*) [inline]

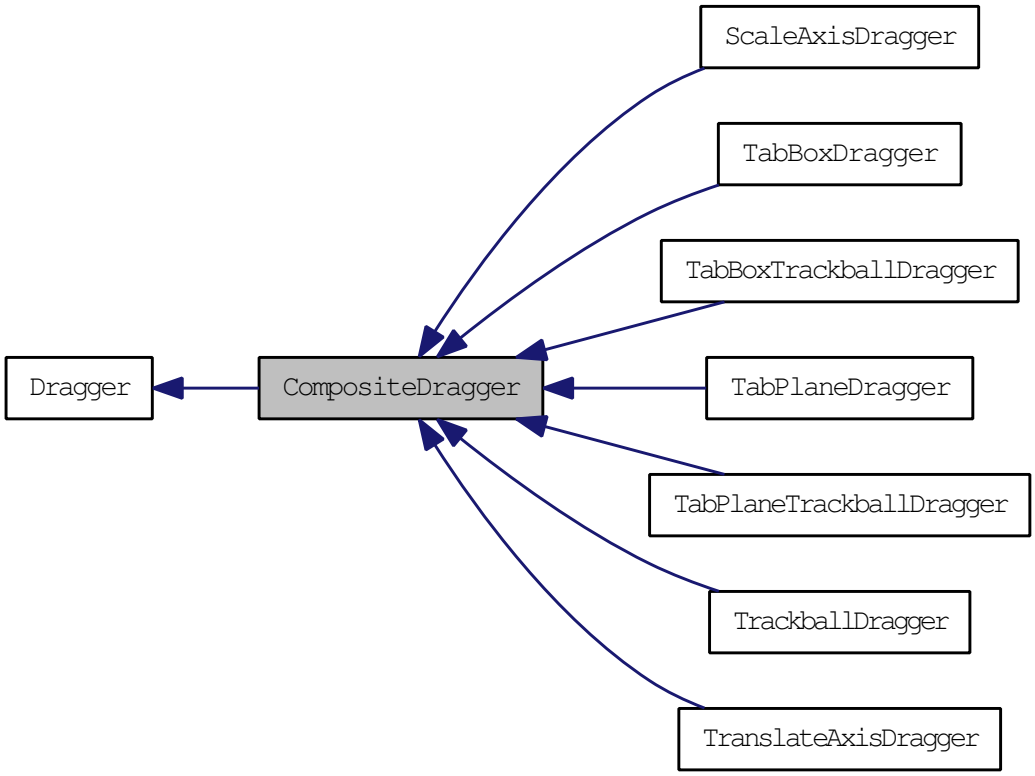
4.3.4.4 Selections getConnectedSelections (Dragger & *dragger*) [inline]

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

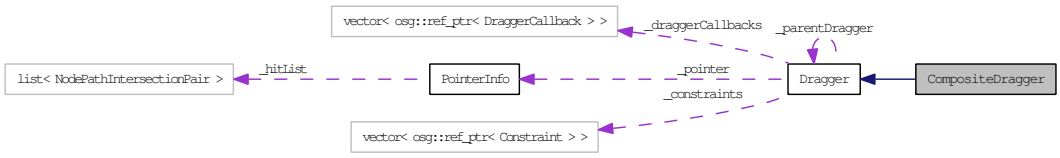
- **CommandManager**

4.4 CompositeDragger Class Reference

CompositeDragger (p. 21) allows to create complex draggers that are composed of a hierarchy of Draggers. Inheritance diagram for CompositeDragger:



Collaboration diagram for CompositeDragger:



Public Member Functions

- virtual bool **addDragger** (**Dragger** ∗dragger)
- bool **containsDragger** (const **Dragger** ∗dragger) const
- DraggerList::iterator **findDragger** (const **Dragger** ∗dragger)
- virtual **CompositeDragger** ∗ **getComposite** ()

  *Returns 0 if this **Dragger** (p. 29) is not a **CompositeDragger** (p. 21).*

- virtual META_Node(osgManipulator, **CompositeDragger**) typedef std const **CompositeDragger** ∗ **getComposite** () const

  *Returns 0 if this **Dragger** (p. 29) is not a **CompositeDragger** (p. 21).*

- const **Dragger** ∗ **getDragger** (unsigned int i) const
- **Dragger** ∗ **getDragger** (unsigned int i)
- unsigned int **getNumDraggers** () const
- virtual bool **handle** (const **PointerInfo** &pi, const osgGA::GUIEventAdapter &ea, osgGA::GUIActionAdapter &aa)
- virtual bool **removeDragger** (**Dragger** ∗dragger)
- virtual void **setParentDragger** (**Dragger** ∗parent)

Protected Member Functions

- **CompositeDragger** (const **CompositeDragger** &rhs, const osg::CopyOp &copyop=osg::CopyOp::SHALLOW_COPY)
- **CompositeDragger** ()
- virtual ∼**CompositeDragger** ()

Protected Attributes

- DraggerList **_draggerList**

4.4.1 Detailed Description

CompositeDragger (p. 21) allows to create complex draggers that are composed of a hierarchy of Draggers.

4.4.2 Constructor & Destructor Documentation

4.4.2.1 CompositeDragger () [inline, protected]

4.4.2.2 CompositeDragger (const CompositeDragger & *rhs*, const osg::CopyOp & *copyop* = osg::CopyOp::SHALLOW_COPY) [protected]

4.4.2.3 virtual ∼CompositeDragger () [inline, protected, virtual]

4.4.3 Member Function Documentation

4.4.3.1 bool addDragger (Dragger ∗ *dragger*) [virtual]

4.4.3.2 bool containsDragger (const Dragger ∗ *dragger*) const

4.4.3.3 CompositeDragger::DraggerList::iterator findDragger (const Dragger ∗ *dragger*)

4.4.3.4 virtual CompositeDragger∗ getComposite () [inline, virtual]

Returns 0 if this **Dragger** (p. 29) is not a **CompositeDragger** (p. 21).

Reimplemented from **Dragger** (p. 31).

4.4.3.5 virtual META_Node (osgManipulator,CompositeDragger) typedef std const CompositeDragger∗ getComposite () const [inline, virtual]

Returns 0 if this **Dragger** (p. 29) is not a **CompositeDragger** (p. 21).

Reimplemented from **Dragger** (p. 31).

4.4.3.6 const Dragger∗ getDragger (unsigned int *i*) const [inline]

4.4.3.7 Dragger∗ getDragger (unsigned int *i*) [inline]

4.4.3.8 unsigned int getNumDraggers () const [inline]

4.4.3.9 bool handle (const PointerInfo & *pi*, const osgGA::GUIEventAdapter & *ea*, osgGA::GUIActionAdapter & *aa*) [virtual]

Reimplemented from **Dragger** (p. 31).

4.4.3.10 bool removeDragger (Dragger ∗ *dragger*) [virtual]

4.4.3.11 void setParentDragger (Dragger ∗ *parent*) [virtual]

4.4.4 Member Data Documentation

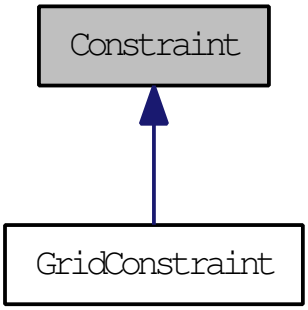
4.4.4.1 DraggerList _draggerList [protected]

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

- **Dragger**
- **Dragger.cpp**

4.5 Constraint Class Reference

Inheritance diagram for Constraint:



Public Member Functions

- virtual bool **constrain** (**ScaleUniformCommand** &command) const
- virtual bool **constrain** (**Scale2DCommand** &command) const
- virtual bool **constrain** (**Scale1DCommand** &command) const
- virtual bool **constrain** (**TranslateInPlaneCommand** &command) const
- virtual bool **constrain** (**TranslateInLineCommand** &command) const
- virtual bool **constrain** (**MotionCommand** &) const

Protected Member Functions

- **Constraint** (osg::Node &refNode)
- virtual ∼**Constraint** ()
- void **computeLocalToWorldAndWorldToLocal** () const
- const osg::Matrix & **getLocalToWorld** () const
- const osg::Node & **getReferenceNode** () const
- osg::Node & **getReferenceNode** ()
- const osg::Matrix & **getWorldToLocal** () const

4.5.1 Constructor & Destructor Documentation

4.5.1.1 Constraint (osg::Node & *refNode*) [inline, protected]

4.5.1.2 virtual ∼Constraint () [inline, protected, virtual]

4.5.2 Member Function Documentation

4.5.2.1 void computeLocalToWorldAndWorldToLocal () const [protected]

4.5.2.2 virtual bool constrain (ScaleUniformCommand & *command*) const [inline, virtual]

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

4.5.2.3 virtual bool constrain (Scale2DCommand & *command*) const [inline, virtual]

Reimplemented in **GridConstraint** (p. 38).

4.5.2.4 virtual bool constrain (Scale1DCommand & *command*) const [inline, virtual]

Reimplemented in **GridConstraint** (p. 38).

4.5.2.5 virtual bool constrain (TranslateInPlaneCommand & *command*) const [inline, virtual]

Reimplemented in **GridConstraint** (p. 38).

4.5.2.6 virtual bool constrain (TranslateInLineCommand & *command*) const [inline, virtual]

Reimplemented in **GridConstraint** (p. 38).

4.5.2.7 virtual bool constrain (MotionCommand &) const [inline, virtual]

4.5.2.8 const osg::Matrix& getLocalToWorld () const [inline, protected]

4.5.2.9 const osg::Node& getReferenceNode () const [inline, protected]

4.5.2.10 osg::Node& getReferenceNode () [inline, protected]

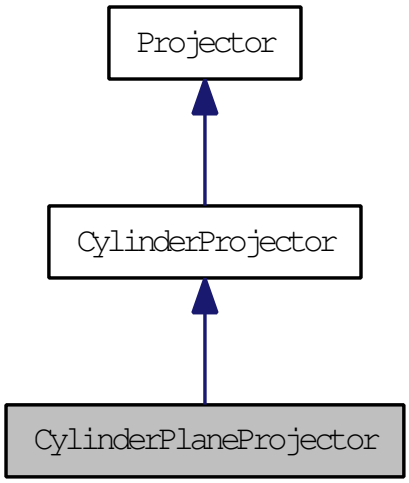
4.5.2.11 const osg::Matrix& getWorldToLocal () const [inline, protected]

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

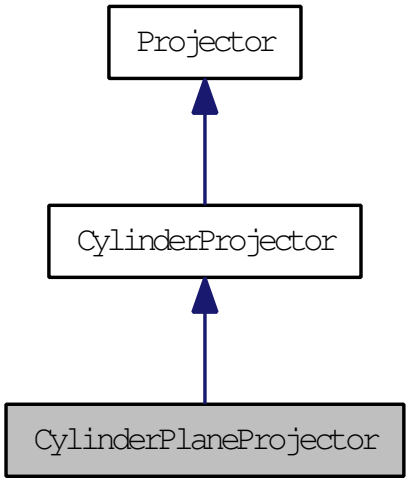
- **Constraint**
- **Constraint.cpp**

4.6 CylinderPlaneProjector Class Reference

CylinderPlaneProjector (p. 25) projects points onto the given cylinder. Inheritance diagram for CylinderPlaneProjector:



Collaboration diagram for CylinderPlaneProjector:



Public Member Functions

- **CylinderPlaneProjector** (osg::Cylinder *cylinder)
- **CylinderPlaneProjector** ()
- osg::Quat **getRotation** (const osg::Vec3d &p1, bool p1OnCyl, const osg::Vec3d &p2, bool p2OnCyl) const
- bool **isProjectionOnCylinder** () const

  Returns true if the previous projection was on the cylinder and false if the projection was on the plane.

- virtual bool **project** (const **PointerInfo** &pi, osg::Vec3d &projectedPoint) const

  Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given plane.

Protected Member Functions

- virtual ~**CylinderPlaneProjector** ()

Protected Attributes

- bool **_onCylinder**
- osg::Plane **_plane**
- osg::Vec3d **_planeLineEnd**
- osg::Vec3d **_planeLineStart**

4.6.1 Detailed Description

CylinderPlaneProjector (p. 25) projects points onto the given cylinder.

4.6.2 Constructor & Destructor Documentation

4.6.2.1 CylinderPlaneProjector ()

4.6.2.2 CylinderPlaneProjector (osg::Cylinder * *cylinder*)

4.6.2.3 ~CylinderPlaneProjector () [protected, virtual]

4.6.3 Member Function Documentation

4.6.3.1 osg::Quat getRotation (const osg::Vec3d & *p1*, bool *p1OnCyl*, const osg::Vec3d & *p2*, bool *p2OnCyl*) const

4.6.3.2 bool isProjectionOnCylinder () const [inline]

Returns true if the previous projection was on the cylinder and false if the projection was on the plane.

4.6.3.3 bool project (const PointerInfo & *pi*, osg::Vec3d & *projectedPoint*) const [virtual]

Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given plane. Returns true on successful projection.

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

4.6.4 Member Data Documentation

4.6.4.1 bool _onCylinder [mutable, protected]

4.6.4.2 osg::Plane _plane [mutable, protected]

4.6.4.3 osg::Vec3d _planeLineEnd [mutable, protected]

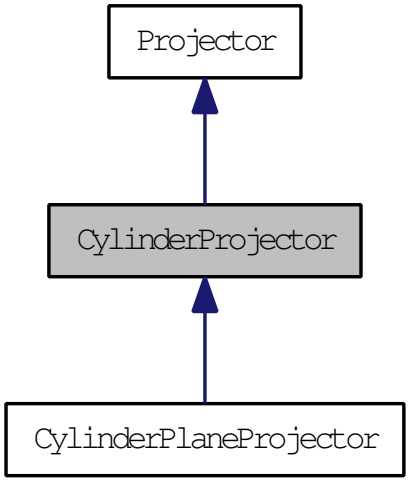
4.6.4.4 osg::Vec3d _planeLineStart [mutable, protected]

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

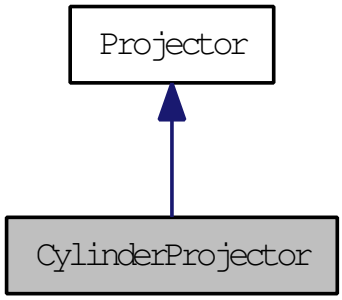
- **Projector**
- **Projector.cpp**

4.7 CylinderProjector Class Reference

CylinderProjector (p. 27) projects points onto the given cylinder. Inheritance diagram for CylinderProjector:



Collaboration diagram for CylinderProjector:



Public Member Functions

- **CylinderProjector** (osg::Cylinder ∗cylinder)
- **CylinderProjector** ()
- const osg::Cylinder ∗ **getCylinder** () const
- bool **isPointInFront** (const **PointerInfo** &pi, const osg::Matrix &localToWorld) const

  Returns true is the point is in front of the cylinder given the eye direction.

- virtual bool **project** (const **PointerInfo** &pi, osg::Vec3d &projectedPoint) const

  Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given plane.

- void **setCylinder** (osg::Cylinder ∗cylinder)
- void **setFront** (bool front)

Protected Member Functions

- virtual ∼**CylinderProjector** ()

Protected Attributes

- osg::ref_ptr< osg::Cylinder > **_cylinder**
- osg::Vec3d **_cylinderAxis**
- bool **_front**

4.7.1 Detailed Description

CylinderProjector (p. 27) projects points onto the given cylinder.

4.7.2 Constructor & Destructor Documentation

4.7.2.1 CylinderProjector ()

4.7.2.2 CylinderProjector (osg::Cylinder * *cylinder*)

4.7.2.3 ~CylinderProjector () [protected, virtual]

4.7.3 Member Function Documentation

4.7.3.1 const osg::Cylinder* getCylinder () const [inline]

4.7.3.2 bool isPointInFront (const PointerInfo & *pi*, const osg::Matrix & *localToWorld*) const

Returns true is the point is in front of the cylinder given the eye direction.

4.7.3.3 bool project (const PointerInfo & *pi*, osg::Vec3d & *projectedPoint*) const [virtual]

Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given plane. Returns true on successful projection.

Implements **Projector** (p. 48).

Reimplemented in **CylinderPlaneProjector** (p. 26).

4.7.3.4 void setCylinder (osg::Cylinder * *cylinder*) [inline]

4.7.3.5 void setFront (bool *front*) [inline]

4.7.4 Member Data Documentation

4.7.4.1 osg::ref_ptr<osg::Cylinder> _cylinder [protected]

4.7.4.2 osg::Vec3d _cylinderAxis [protected]

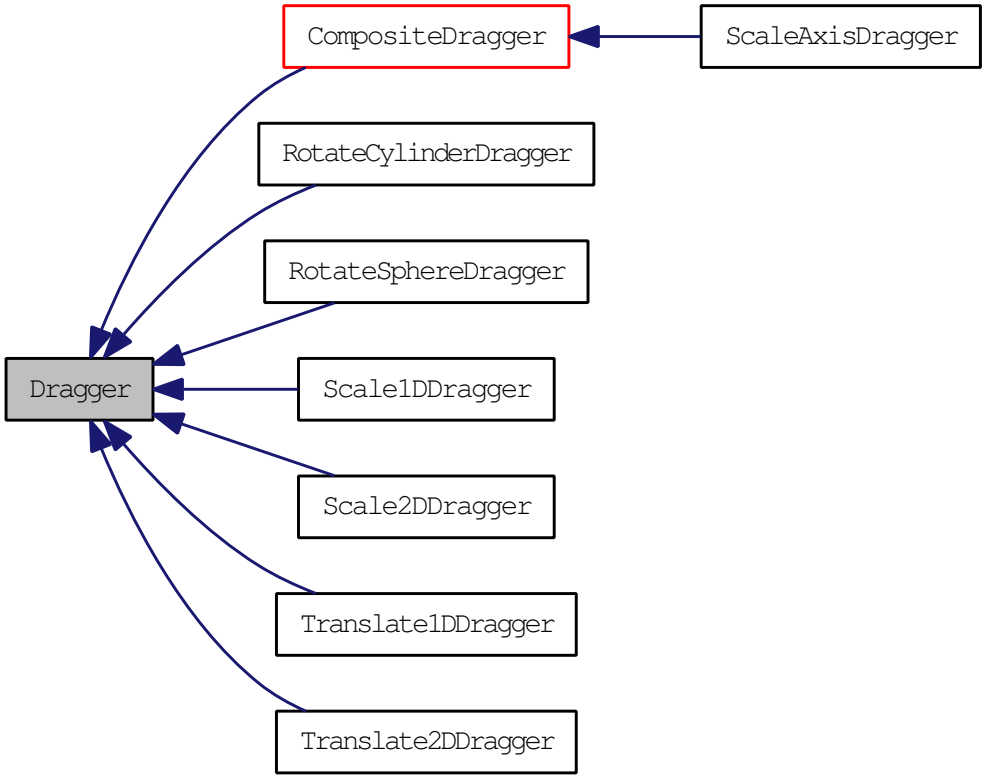
4.7.4.3 bool _front [protected]

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

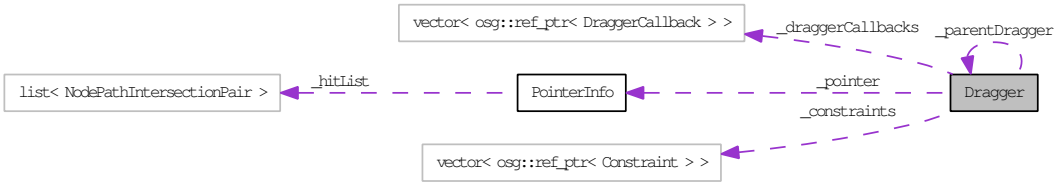
- **Projector**
- **Projector.cpp**

4.8 Dragger Class Reference

Base class for draggers. Inheritance diagram for Dragger:



Collaboration diagram for Dragger:



Public Types

- typedef std::vector< osg::ref_ptr< **Constraint** > > **Constraints**
- typedef std::vector< osg::ref_ptr< **DraggerCallback** > > **DraggerCallbacks**

Public Member Functions

- void **addConstraint** (**Constraint** *constraint)
- void **addDraggerCallback** (**DraggerCallback** *dc)
- void **addTransformUpdating** (MatrixTransform *transform)
- int **getActivationKeyEvent** () const
- unsigned int **getActivationModKeyMask** () const
- virtual **CompositeDragger** * **getComposite** ()

  *Returns 0 if this **Dragger** (p. 29) is not a **CompositeDragger** (p. 21).*

- virtual const **CompositeDragger** * **getComposite** () const

  *Returns 0 if this **Dragger** (p. 29) is not a **CompositeDragger** (p. 21).*

- const **Constraints** & **getConstraints** () const
- **Constraints** & **getConstraints** ()
- const **DraggerCallbacks** & **getDraggerCallbacks** () const
- **DraggerCallbacks** & **getDraggerCallbacks** ()
- bool **getHandleEvents** () const
- const **Dragger** ∗ **getParentDragger** () const
- **Dragger** ∗ **getParentDragger** ()
- virtual bool **handle** (const **PointerInfo** &, const osgGA::GUIEventAdapter &, osgGA::GUIActionAdapter &)
- virtual bool **handle** (const osgGA::GUIEventAdapter &ea, osgGA::GUIActionAdapter &aa)
- **META_Node** (osgManipulator, **Dragger**) virtual void setParentDragger(**Dragger** ∗parent)

  Set/Get parent dragger.

- void **removeConstraint** (**Constraint** ∗constraint)
- void **removeDraggerCallback** (**DraggerCallback** ∗dc)
- void **removeTransformUpdating** (MatrixTransform ∗transform)
- void **setActivationKeyEvent** (int key)
- void **setActivationModKeyMask** (unsigned int mask)
- void **setHandleEvents** (bool flag)
- virtual void **setupDefaultGeometry** ()

  Setup default geometry for dragger.

- virtual void **traverse** (osg::NodeVisitor &nv)

Protected Member Functions

- **Dragger** (const **Dragger** &rhs, const osg::CopyOp &copyop=osg::CopyOp::SHALLOW_COPY)
- **Dragger** ()
- virtual ∼**Dragger** ()
- void **dispatch** (**MotionCommand** &command)
- bool **getDraggerActive** () const
- virtual bool **receive** (const **MotionCommand** &command)
- void **setDraggerActive** (bool active)

Protected Attributes

- int **_activationKeyEvent**
- unsigned int **_activationModKeyMask**
- bool **_activationPermittedByKeyEvent**
- bool **_activationPermittedByModKeyMask**
- **Constraints _constraints**
- bool **_draggerActive**
- **DraggerCallbacks _draggerCallbacks**
- bool **_handleEvents**
- **Dragger** ∗ **_parentDragger**
- **osgManipulator::PointerInfo _pointer**
- osg::ref_ptr< **DraggerCallback** > **_selfUpdater**

4.8.1 Detailed Description

Base class for draggers. Concrete draggers implement the pick event handler and generate motion commands (translate, rotate, ...) and sends these command to all the DraggerCallbacks & Transforms that are connected to the **Dragger** (p. 29) that generates the commands.

4.8.2 Member Typedef Documentation

4.8.2.1 typedef std::vector< osg::ref_ptr<Constraint> > Constraints

4.8.2.2 typedef std::vector< osg::ref_ptr<DraggerCallback> > DraggerCallbacks

4.8.3 Constructor & Destructor Documentation

4.8.3.1 Dragger () [protected]

4.8.3.2 Dragger (const Dragger & *rhs*, const osg::CopyOp & *copyop* = osg::CopyOp::SHALLOW_COPY) [protected]

4.8.3.3 ~Dragger () [protected, virtual]

4.8.4 Member Function Documentation

4.8.4.1 void addConstraint (Constraint * *constraint*)

4.8.4.2 void addDraggerCallback (DraggerCallback * *dc*)

4.8.4.3 void addTransformUpdating (MatrixTransform * *transform*)

4.8.4.4 void dispatch (MotionCommand & *command*) [protected]

4.8.4.5 int getActivationKeyEvent () const [inline]

4.8.4.6 unsigned int getActivationModKeyMask () const [inline]

4.8.4.7 virtual CompositeDragger* getComposite () [inline, virtual]

Returns 0 if this **Dragger** (p. 29) is not a **CompositeDragger** (p. 21).

Reimplemented in **CompositeDragger** (p. 22).

4.8.4.8 virtual const CompositeDragger* getComposite () const [inline, virtual]

Returns 0 if this **Dragger** (p. 29) is not a **CompositeDragger** (p. 21).

Reimplemented in **CompositeDragger** (p. 22).

4.8.4.9 const Constraints& getConstraints () const [inline]

4.8.4.10 Constraints& getConstraints () [inline]

4.8.4.11 bool getDraggerActive () const [inline, protected]

4.8.4.12 const DraggerCallbacks& getDraggerCallbacks () const [inline]

4.8.4.13 DraggerCallbacks& getDraggerCallbacks () [inline]

4.8.4.14 bool getHandleEvents () const [inline]

4.8.4.15 const Dragger* getParentDragger () const [inline]

4.8.4.16 Dragger* getParentDragger () [inline]

4.8.4.17 virtual bool handle (const PointerInfo &, const osgGA::GUIEventAdapter &, osgGA::GUIActionAdapter &) [inline, virtual]

Reimplemented in **CompositeDragger** (p. 22).

4.8.4.18 bool handle (const osgGA::GUIEventAdapter & *ea*, osgGA::GUIActionAdapter & *aa*) [virtual]

4.8.4.19 META_Node (osgManipulator, Dragger) [inline]

Set/Get parent dragger. For simple draggers parent points to itself. For composite draggers parent points to the parent dragger that uses this dragger.

4.8.4.20 bool receive (const MotionCommand & *command*) [protected, virtual]

4.8.4.21 void removeConstraint (Constraint * *constraint*)

4.8.4.22 void removeDraggerCallback (DraggerCallback * *dc*)

4.8.4.23 void removeTransformUpdating (MatrixTransform * *transform*)

4.8.4.24 void setActivationKeyEvent (int *key*) [inline]

4.8.4.25 void setActivationModKeyMask (unsigned int *mask*) [inline]

4.8.4.26 void setDraggerActive (bool *active*) [inline, protected]

4.8.4.27 void setHandleEvents (bool *flag*)

4.8.4.28 virtual void setupDefaultGeometry () [inline, virtual]

Setup default geometry for dragger.

Reimplemented in **RotateCylinderDragger** (p. 52), **RotateSphereDragger** (p. 54), **Scale1DDragger** (p. 59), **Scale2DDragger** (p. 64), **Translate1DDragger** (p. 85), **Translate2DDragger** (p. 87), and **TranslatePlaneDragger** (p. 95).

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

4.8.5 Member Data Documentation

4.8.5.1 int _activationKeyEvent [protected]

4.8.5.2 unsigned int _activationModKeyMask [protected]

4.8.5.3 bool _activationPermittedByKeyEvent [protected]

4.8.5.4 bool _activationPermittedByModKeyMask [protected]

4.8.5.5 Constraints _constraints [protected]

4.8.5.6 bool _draggerActive [protected]

4.8.5.7 DraggerCallbacks _draggerCallbacks [protected]

4.8.5.8 bool _handleEvents [protected]

4.8.5.9 Dragger* _parentDragger [protected]

4.8.5.10 osgManipulator::PointerInfo _pointer [protected]

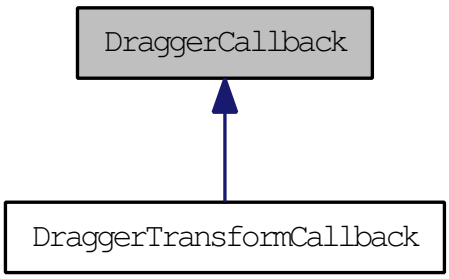
4.8.5.11 osg::ref_ptr<DraggerCallback> _selfUpdater [protected]

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

- **Dragger**
- **Dragger.cpp**

4.9 DraggerCallback Class Reference

Inheritance diagram for DraggerCallback:



Public Member Functions

- **DraggerCallback** (const **DraggerCallback** &, const osg::CopyOp &copyop=osg::CopyOp::SHALLOW_-COPY)
- **DraggerCallback** ()
- **META_Object** (osgManipulator, **DraggerCallback**)
- virtual bool **receive** (const **Rotate3DCommand** &command)
- virtual bool **receive** (const **ScaleUniformCommand** &command)
- virtual bool **receive** (const **Scale2DCommand** &command)
- virtual bool **receive** (const **Scale1DCommand** &command)
- virtual bool **receive** (const **TranslateInPlaneCommand** &command)
- virtual bool **receive** (const **TranslateInLineCommand** &command)
- virtual bool **receive** (const **MotionCommand** &)

  Receive motion commands.

4.9.1 Constructor & Destructor Documentation

4.9.1.1 DraggerCallback () [inline]

4.9.1.2 DraggerCallback (const DraggerCallback &, const osg::CopyOp & *copyop* = osg::CopyOp::SHALLOW_COPY) [inline]

4.9.2 Member Function Documentation

4.9.2.1 META_Object (osgManipulator, DraggerCallback)

4.9.2.2 virtual bool receive (const Rotate3DCommand & *command*) [inline, virtual]

4.9.2.3 virtual bool receive (const ScaleUniformCommand & *command*) [inline, virtual]

4.9.2.4 virtual bool receive (const Scale2DCommand & *command*) [inline, virtual]

4.9.2.5 virtual bool receive (const Scale1DCommand & *command*) [inline, virtual]

4.9.2.6 virtual bool receive (const TranslateInPlaneCommand & *command*) [inline, virtual]

4.9.2.7 virtual bool receive (const TranslateInLineCommand & *command*) [inline, virtual]

4.9.2.8 virtual bool receive (const MotionCommand &) [inline, virtual]

Receive motion commands. Returns true on success.

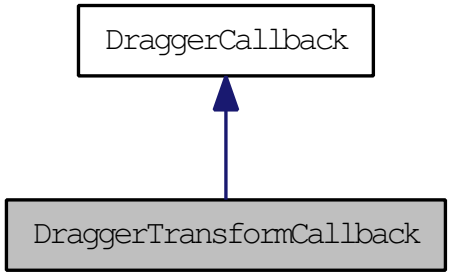
Reimplemented in **DraggerTransformCallback** (p. 34).

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

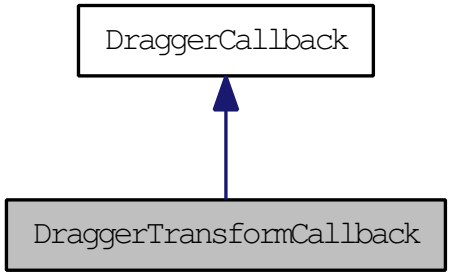
- **Dragger**

4.10 DraggerTransformCallback Class Reference

Inheritance diagram for DraggerTransformCallback:



Collaboration diagram for DraggerTransformCallback:



Public Member Functions

- **DraggerTransformCallback** (osg::MatrixTransform *transform)
- const osg::MatrixTransform * **getTransform** () const
- osg::MatrixTransform * **getTransform** ()
- virtual bool **receive** (const **MotionCommand** &)

  Receive motion commands.

Protected Attributes

- osg::Matrix **_localToWorld**
- osg::Matrix **_startMotionMatrix**
- osg::observer_ptr< osg::MatrixTransform > **_transform**
- osg::Matrix **_worldToLocal**

4.10.1 Constructor & Destructor Documentation

4.10.1.1 DraggerTransformCallback (osg::MatrixTransform * *transform*)

4.10.2 Member Function Documentation

4.10.2.1 const osg::MatrixTransform* getTransform () const [inline]

4.10.2.2 osg::MatrixTransform* getTransform () [inline]

4.10.2.3 bool receive (const MotionCommand &) [virtual]

Receive motion commands. Returns true on success.

Reimplemented from **DraggerCallback** (p. 33).

4.10.3 Member Data Documentation

4.10.3.1 osg::Matrix _localToWorld [protected]

4.10.3.2 osg::Matrix _startMotionMatrix [protected]

4.10.3.3 osg::observer_ptr<osg::MatrixTransform> _transform [protected]

4.10.3.4 osg::Matrix _worldToLocal [protected]

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

- **Dragger**
- **Dragger.cpp**

4.11 ForceCullCallback Class Reference

Public Member Functions

- virtual bool **cull** (osg::NodeVisitor ∗, osg::Drawable ∗, osg::State ∗) const

4.11.1 Member Function Documentation

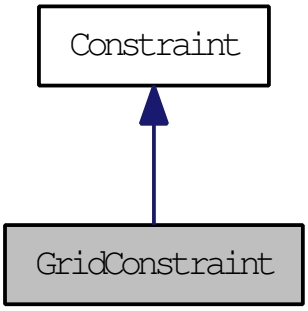
4.11.1.1 virtual bool cull (osg::NodeVisitor ∗, osg::Drawable ∗, osg::State ∗) const [inline, virtual]

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

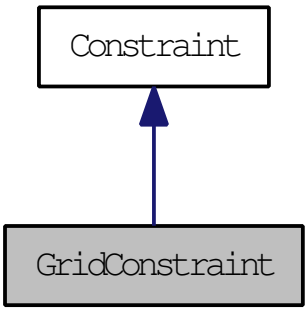
- **Dragger.cpp**

4.12 GridConstraint Class Reference

Constraint (p. 23) to snap motion commands to a sugar cube grid. Inheritance diagram for GridConstraint:



Collaboration diagram for GridConstraint:



Public Member Functions

- **GridConstraint** (osg::Node &refNode, const osg::Vec3d &origin, const osg::Vec3d &spacing)
- virtual bool **constrain** (**ScaleUniformCommand** &command) const
- virtual bool **constrain** (**Scale2DCommand** &command) const
- virtual bool **constrain** (**Scale1DCommand** &command) const
- virtual bool **constrain** (**TranslateInPlaneCommand** &command) const
- virtual bool **constrain** (**TranslateInLineCommand** &command) const
- const osg::Vec3d & **getOrigin** () const
- const osg::Vec3d & **getSpacing** () const
- void **setOrigin** (const osg::Vec3d &origin)
- void **setSpacing** (const osg::Vec3d &spacing)

Protected Member Functions

- virtual ~**GridConstraint** ()

4.12.1 Detailed Description

Constraint (p. 23) to snap motion commands to a sugar cube grid.

4.12.2 Constructor & Destructor Documentation

4.12.2.1 GridConstraint (osg::Node & *refNode*, const osg::Vec3d & *origin*, const osg::Vec3d & *spacing*)

4.12.2.2 virtual ~GridConstraint () `[inline, protected, virtual]`

4.12.3 Member Function Documentation

4.12.3.1 bool constrain (ScaleUniformCommand & *command*) const `[virtual]`

Reimplemented from **Constraint** (p. 23).

4.12.3.2 bool constrain (Scale2DCommand & *command*) const [virtual]

Reimplemented from **Constraint** (p. 23).

4.12.3.3 bool constrain (Scale1DCommand & *command*) const [virtual]

Reimplemented from **Constraint** (p. 23).

4.12.3.4 bool constrain (TranslateInPlaneCommand & *command*) const [virtual]

Reimplemented from **Constraint** (p. 23).

4.12.3.5 bool constrain (TranslateInLineCommand & *command*) const [virtual]

Reimplemented from **Constraint** (p. 23).

4.12.3.6 const osg::Vec3d& getOrigin () const [inline]

4.12.3.7 const osg::Vec3d& getSpacing () const [inline]

4.12.3.8 void setOrigin (const osg::Vec3d & *origin*) [inline]

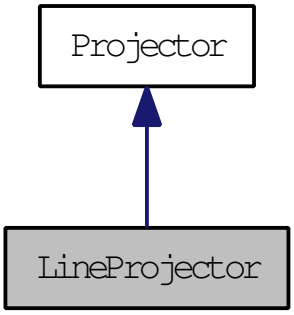
4.12.3.9 void setSpacing (const osg::Vec3d & *spacing*) [inline]

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

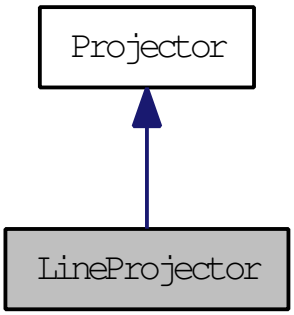
- **Constraint**
- **Constraint.cpp**

4.13 LineProjector Class Reference

LineProjector (p. 39) projects points onto the closest point on the given line. Inheritance diagram for LineProjector:



Collaboration diagram for LineProjector:



Public Member Functions

- **LineProjector** (const osg::LineSegment::vec_type &s, const osg::LineSegment::vec_type &e)
- **LineProjector** ()
- osg::LineSegment::vec_type & **getLineEnd** ()
- const osg::LineSegment::vec_type & **getLineEnd** () const
- osg::LineSegment::vec_type & **getLineStart** ()
- const osg::LineSegment::vec_type & **getLineStart** () const
- virtual bool **project** (const **PointerInfo** &pi, osg::Vec3d &projectedPoint) const

  Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given line.

- void **setLine** (const osg::LineSegment::vec_type &s, const osg::LineSegment::vec_type &e)

Protected Member Functions

- virtual ~**LineProjector** ()

Protected Attributes

- osg::ref_ptr< osg::LineSegment > **_line**

4.13.1 Detailed Description

LineProjector (p. 39) projects points onto the closest point on the given line.

4.13.2 Constructor & Destructor Documentation

4.13.2.1 LineProjector ()

4.13.2.2 LineProjector (const osg::LineSegment::vec_type & *s*, const osg::LineSegment::vec_type & *e*)

4.13.2.3 ~LineProjector () [protected, virtual]

4.13.3 Member Function Documentation

4.13.3.1 osg::LineSegment::vec_type& getLineEnd () [inline]

4.13.3.2 const osg::LineSegment::vec_type& getLineEnd () const [inline]

4.13.3.3 osg::LineSegment::vec_type& getLineStart () [inline]

4.13.3.4 const osg::LineSegment::vec_type& getLineStart () const [inline]

4.13.3.5 bool project (const PointerInfo & *pi*, osg::Vec3d & *projectedPoint*) const [virtual]

Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given line. Returns true on successful projection.

Implements **Projector** (p. 48).

4.13.3.6 void setLine (const osg::LineSegment::vec_type & *s*, const osg::LineSegment::vec_type & *e*) [inline]

4.13.4 Member Data Documentation

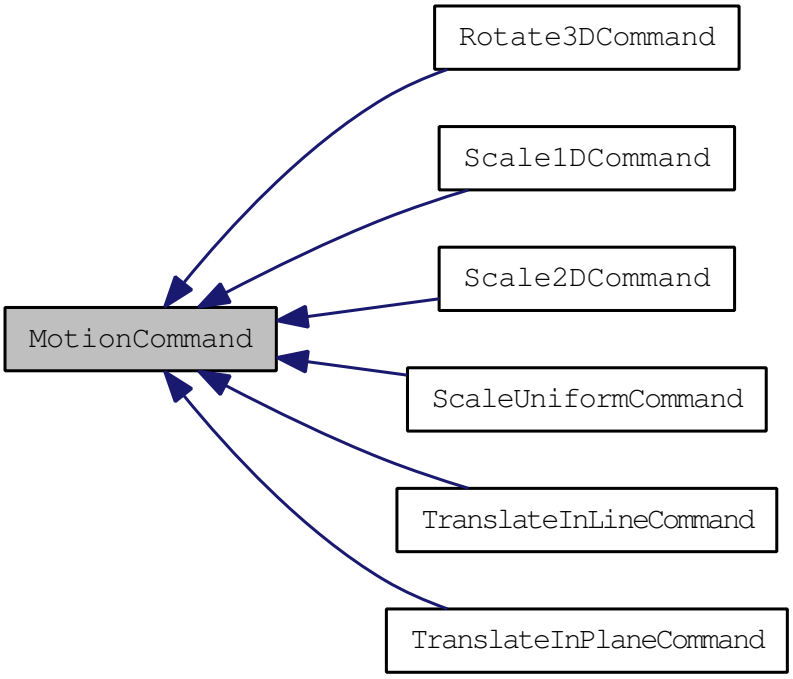
4.13.4.1 osg::ref_ptr<osg::LineSegment> _line [protected]

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

- **Projector**
- **Projector.cpp**

4.14 MotionCommand Class Reference

Base class for motion commands that are generated by draggers. Inheritance diagram for MotionCommand:



Public Types

- enum **Stage** { **NONE**, **START**, **MOVE**, **FINISH** }

  Motion command are based on click-drag-release actions.

Public Member Functions

- **MotionCommand** ()
- virtual **MotionCommand** * **createCommandInverse** ()=0

  *create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.*

- const osg::Matrix & **getLocalToWorld** () const

  Gets the matrix for transforming the command's local coordinate system to the world/object coordinate system.

- virtual osg::Matrix **getMotionMatrix** () const =0

  Gets the matrix for transforming the object being dragged.

- **Stage getStage** () const
- const osg::Matrix & **getWorldToLocal** () const

  Gets the matrix for transforming the command's world/object coordinate system to the command's local coordinate system.

- void **setLocalToWorldAndWorldToLocal** (const osg::Matrix &localToWorld, const osg::Matrix &worldToLocal)

  Sets the matrix for transforming the command's local coordinate system to the world/object coordinate system.

- void **setStage** (const **Stage** s)

Protected Member Functions

- virtual ~**MotionCommand** ()

4.14.1 Detailed Description

Base class for motion commands that are generated by draggers.

4.14.2 Member Enumeration Documentation

4.14.2.1 enum Stage

Motion command are based on click-drag-release actions. So each command needs to indicate which stage of the motion the command represents.

Enumerator:

NONE

START Click or pick start.

MOVE Drag or pick move.

FINISH Release or pick finish.

4.14.3 Constructor & Destructor Documentation

4.14.3.1 MotionCommand ()

4.14.3.2 ~MotionCommand () `[protected, virtual]`

4.14.4 Member Function Documentation

4.14.4.1 virtual MotionCommand* createCommandInverse () `[pure virtual]`

create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.

Implemented in **TranslateInLineCommand** (p. 91), **TranslateInPlaneCommand** (p. 93), **Scale1DCommand** (p. 56), **Scale2DCommand** (p. 61), **ScaleUniformCommand** (p. 69), and **Rotate3DCommand** (p. 49).

4.14.4.2 const osg::Matrix& getLocalToWorld () const `[inline]`

Gets the matrix for transforming the command's local coordinate system to the world/object coordinate system.

4.14.4.3 virtual osg::Matrix getMotionMatrix () const `[pure virtual]`

Gets the matrix for transforming the object being dragged. This matrix is in the command's coordinate systems.

Implemented in **TranslateInLineCommand** (p. 91), **TranslateInPlaneCommand** (p. 93), **Scale1DCommand** (p. 56), **Scale2DCommand** (p. 61), **ScaleUniformCommand** (p. 69), and **Rotate3DCommand** (p. 50).

4.14.4.4 Stage getStage () const `[inline]`

4.14.4.5 const osg::Matrix& getWorldToLocal () const `[inline]`

Gets the matrix for transforming the command's world/object coordinate system to the command's local coordinate system.

4.14.4.6 void setLocalToWorldAndWorldToLocal (const osg::Matrix & *localToWorld*, const osg::Matrix & *worldToLocal*) `[inline]`

Sets the matrix for transforming the command's local coordinate system to the world/object coordinate system.

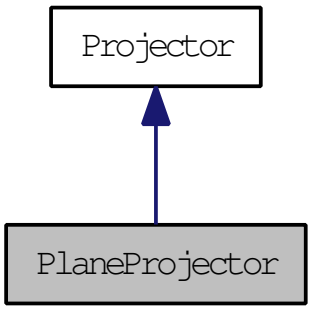
4.14.4.7 void setStage (const Stage *s*) `[inline]`

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

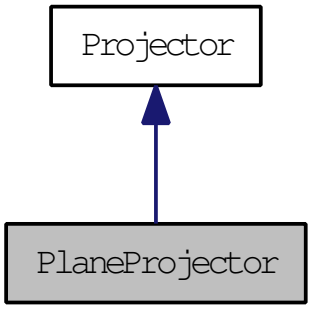
- **Command**
- **Command.cpp**

4.15 PlaneProjector Class Reference

PlaneProjector (p. 43) projects points onto the given line. Inheritance diagram for PlaneProjector:



Collaboration diagram for PlaneProjector:



Public Member Functions

- **PlaneProjector** (const osg::Plane &plane)
- **PlaneProjector** ()
- const osg::Plane & **getPlane** () const
- virtual bool **project** (const **PointerInfo** &pi, osg::Vec3d &projectedPoint) const

  Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given plane.

- void **setPlane** (const osg::Plane &plane)

Protected Member Functions

- virtual ~**PlaneProjector** ()

Protected Attributes

- osg::Plane **_plane**

4.15.1 Detailed Description

PlaneProjector (p. 43) projects points onto the given line.

4.15.2 Constructor & Destructor Documentation

4.15.2.1 PlaneProjector ()

4.15.2.2 PlaneProjector (const osg::Plane & *plane*)

4.15.2.3 ~PlaneProjector () `[protected, virtual]`

4.15.3 Member Function Documentation

4.15.3.1 const osg::Plane& getPlane () const `[inline]`

4.15.3.2 bool project (const PointerInfo & *pi*, osg::Vec3d & *projectedPoint*) const `[virtual]`

Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given plane. Returns true on successful projection.

Implements **Projector** (p. 48).

4.15.3.3 void setPlane (const osg::Plane & *plane*) `[inline]`

4.15.4 Member Data Documentation

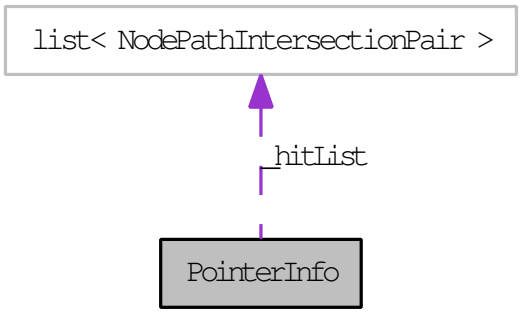
4.15.4.1 osg::Plane _plane `[protected]`

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

- **Projector**
- **Projector.cpp**

4.16 PointerInfo Class Reference

Collaboration diagram for PointerInfo:



Public Types

- typedef std::list< **NodePathIntersectionPair** > **IntersectionList**
- typedef std::pair< osg::NodePath, osg::Vec3d > **NodePathIntersectionPair**

Public Member Functions

- **PointerInfo** (const **PointerInfo** &rhs)
- **PointerInfo** ()
- void **addIntersection** (const osg::NodePath &nodePath, const osg::Vec3d &intersectionPoint)
- bool **completed** () const
- bool **contains** (const osg::Node *node) const
- const osg::Vec3d & **getEyeDir** () const
- osg::Vec3d **getLocalIntersectPoint** () const
- void **getNearFarPoints** (osg::Vec3d &nearPoint, osg::Vec3d &farPoint) const
- void **next** ()
- void **reset** ()
- void **setCamera** (osg::Camera *camera)
- void **setMousePosition** (float pixel_x, float pixel_y)
- void **setNearFarPoints** (osg::Vec3d nearPoint, osg::Vec3d farPoint)

Public Attributes

- IntersectionList::const_iterator **_hitIter**
- **IntersectionList _hitList**

Protected Member Functions

- bool **projectWindowXYIntoObject** (const osg::Vec2d &windowCoord, osg::Vec3d &nearPoint, osg::Vec3d &farPoint) const

Protected Attributes

- osg::Vec3d **_eyeDir**
- osg::Vec3d **_farPoint**
- osg::Matrix **_inverseMVPW**
- osg::Matrix **_MVPW**
- osg::Vec3d **_nearPoint**

4.16.1 Member Typedef Documentation

4.16.1.1 typedef std::list< NodePathIntersectionPair> IntersectionList

4.16.1.2 typedef std::pair<osg::NodePath, osg::Vec3d> NodePathIntersectionPair

4.16.2 Constructor & Destructor Documentation

4.16.2.1 PointerInfo ()

4.16.2.2 PointerInfo (const PointerInfo & *rhs*) [inline]

4.16.3 Member Function Documentation

4.16.3.1 void addIntersection (const osg::NodePath & *nodePath*, const osg::Vec3d & *intersectionPoint*) [inline]

4.16.3.2 bool completed () const [inline]

4.16.3.3 bool contains (const osg::Node * *node*) const

4.16.3.4 const osg::Vec3d& getEyeDir () const [inline]

4.16.3.5 osg::Vec3d getLocalIntersectPoint () const [inline]

4.16.3.6 void getNearFarPoints (osg::Vec3d & *nearPoint*, osg::Vec3d & *farPoint*) const [inline]

4.16.3.7 void next () [inline]

4.16.3.8 bool projectWindowXYIntoObject (const osg::Vec2d & *windowCoord*, osg::Vec3d & *nearPoint*, osg::Vec3d & *farPoint*) const [protected]

4.16.3.9 void reset () [inline]

4.16.3.10 void setCamera (osg::Camera * *camera*) [inline]

4.16.3.11 void setMousePosition (float *pixel_x*, float *pixel_y*) [inline]

4.16.3.12 void setNearFarPoints (osg::Vec3d *nearPoint*, osg::Vec3d *farPoint*) [inline]

4.16.4 Member Data Documentation

4.16.4.1 osg::Vec3d _eyeDir [protected]

4.16.4.2 osg::Vec3d _farPoint [protected]

4.16.4.3 IntersectionList::const_iterator _hitIter

4.16.4.4 IntersectionList _hitList

4.16.4.5 osg::Matrix _inverseMVPW [protected]

4.16.4.6 osg::Matrix _MVPW [protected]

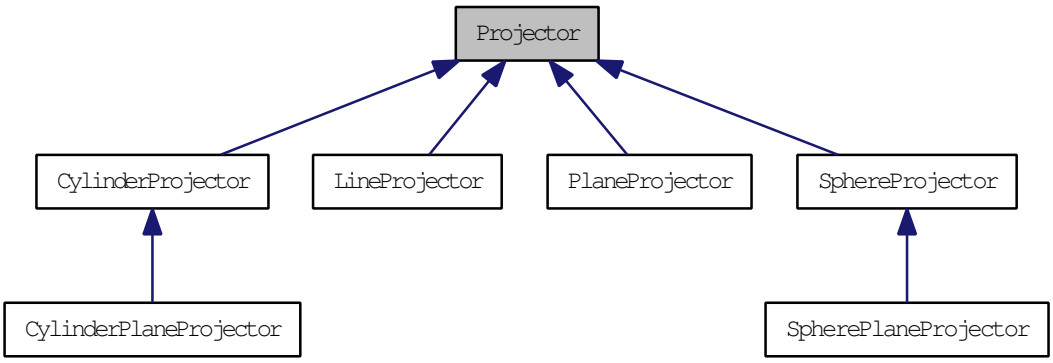
4.16.4.7 osg::Vec3d _nearPoint [protected]

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

- Dragger
- Dragger.cpp

4.17 Projector Class Reference

Base class for Projectors. Inheritance diagram for Projector:



Public Member Functions

- **Projector** ()
- const osg::Matrix & **getLocalToWorld** () const

  Gets the matrix for transforming the projector's local coordinate system to the world/object coordinate system.

- const osg::Matrix & **getWorldToLocal** () const

  Gets the matrix for transforming the world/object coordinate system to the command's local coordinate system.

- virtual bool **project** (const **PointerInfo** &pi, osg::Vec3d &projectedPoint) const =0

  Calculates the object/world coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto some shape or geometry (implemented in derived classes).

- void **setLocalToWorld** (const osg::Matrix &localToWorld)

  Sets the matrix for transforming the projector's local coordinate system to the world/object coordinate system.

Protected Member Functions

- virtual ~**Projector** ()

Protected Attributes

- osg::Matrix **_localToWorld**
- osg::Matrix **_worldToLocal**
- bool **_worldToLocalDirty**

4.17.1 Detailed Description

Base class for Projectors. Projectors maps 2D cursor motions to 3D motions.

4.17.2 Constructor & Destructor Documentation

4.17.2.1 Projector ()

4.17.2.2 ~Projector () `[protected, virtual]`

4.17.3 Member Function Documentation

4.17.3.1 const osg::Matrix& getLocalToWorld () const `[inline]`

Gets the matrix for transforming the projector's local coordinate system to the world/object coordinate system.

4.17.3.2 const osg::Matrix& getWorldToLocal () const [inline]

Gets the matrix for transforming the world/object coordinate system to the command's local coordinate system.

4.17.3.3 virtual bool project (const PointerInfo & *pi*, osg::Vec3d & *projectedPoint*) const [pure virtual]

Calculates the object/world coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto some shape or geometry (implemented in derived classes). SceneView in used for i projecting window coordinates into object coordinates and vice versa. Returns true on successful projection.

Implemented in **LineProjector** (p. 40), **PlaneProjector** (p. 44), **SphereProjector** (p. 73), **SpherePlaneProjector** (p. 71), **CylinderProjector** (p. 28), and **CylinderPlaneProjector** (p. 26).

4.17.3.4 void setLocalToWorld (const osg::Matrix & *localToWorld*) [inline]

Sets the matrix for transforming the projector's local coordinate system to the world/object coordinate system.

4.17.4 Member Data Documentation

4.17.4.1 osg::Matrix _localToWorld [protected]

4.17.4.2 osg::Matrix _worldToLocal [mutable, protected]

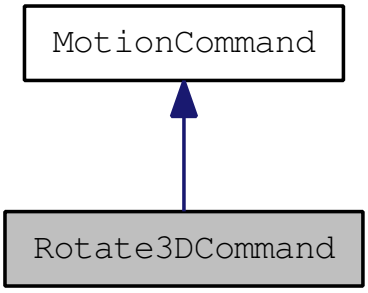
4.17.4.3 bool _worldToLocalDirty [mutable, protected]

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

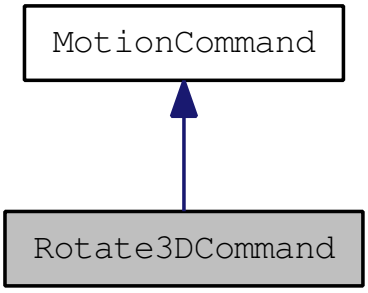
- **Projector**
- **Projector.cpp**

4.18 Rotate3DCommand Class Reference

Command for rotation in 3D. Inheritance diagram for Rotate3DCommand:



Collaboration diagram for Rotate3DCommand:



Public Member Functions

- **Rotate3DCommand** ()
- virtual **MotionCommand** ∗ **createCommandInverse** ()

  *create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.*

- virtual osg::Matrix **getMotionMatrix** () const

  Gets the matrix for transforming the object being dragged.

- const osg::Quat & **getRotation** () const
- void **setRotation** (const osg::Quat &rotation)

Protected Member Functions

- virtual ∼**Rotate3DCommand** ()

4.18.1 Detailed Description

Command for rotation in 3D.

4.18.2 Constructor & Destructor Documentation

4.18.2.1 Rotate3DCommand ()

4.18.2.2 ∼Rotate3DCommand () [protected, virtual]

4.18.3 Member Function Documentation

4.18.3.1 MotionCommand ∗ createCommandInverse () [virtual]

create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.

Implements **MotionCommand** (p. 42).

4.18.3.2 virtual osg::Matrix getMotionMatrix () const [inline, virtual]

Gets the matrix for transforming the object being dragged. This matrix is in the command's coordinate systems.

Implements **MotionCommand** (p. 42).

4.18.3.3 const osg::Quat& getRotation () const [inline]

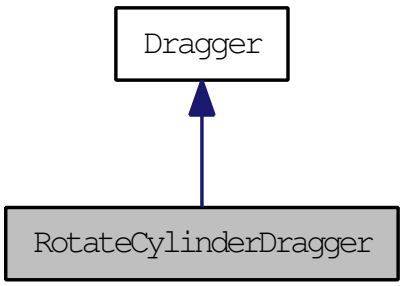
4.18.3.4 void setRotation (const osg::Quat & *rotation*) [inline]

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

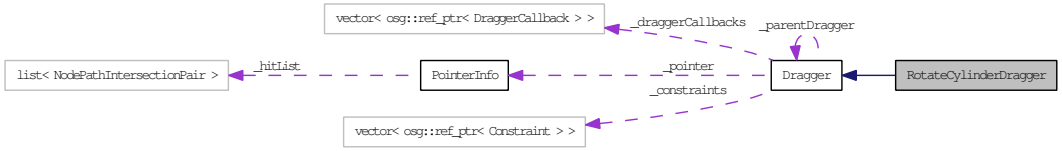
- **Command**
- **Command.cpp**

4.19 RotateCylinderDragger Class Reference

Dragger (p. 29) for performing 3D rotation on a cylinder. Inheritance diagram for RotateCylinderDragger:



Collaboration diagram for RotateCylinderDragger:



Public Member Functions

- **RotateCylinderDragger** ()
- const osg::Vec4 **getColor** () const
- const osg::Vec4 **getPickColor** () const
- **META_OSGMANIPULATOR_Object** (osgManipulator, **RotateCylinderDragger**) virtual bool handle(const **PointerInfo** &pi

  Handle pick events on dragger and generate TranslateInLine commands.

- void **setColor** (const osg::Vec4 &color)

  Set/Get color for dragger.

- void **setPickColor** (const osg::Vec4 &color)

  Set/Get pick color for dragger.

- void **setupDefaultGeometry** ()

  Setup default geometry for dragger.

Public Attributes

- const osgGA::GUIEventAdapter & **ea**
- const osgGA::GUIEventAdapter osgGA::GUIActionAdapter & **us**

Protected Member Functions

- virtual ~**RotateCylinderDragger** ()

Protected Attributes

- osg::Vec4 **_color**
- osg::Vec4 **_pickColor**
- bool **_prevPtOnCylinder**
- osg::Quat **_prevRotation**

- osg::Vec3d **_prevWorldProjPt**
- osg::ref_ptr< **CylinderPlaneProjector** > **_projector**
- osg::Matrix **_startLocalToWorld**
- osg::Matrix **_startWorldToLocal**

4.19.1 Detailed Description

Dragger (p. 29) for performing 3D rotation on a cylinder.

4.19.2 Constructor & Destructor Documentation

4.19.2.1 RotateCylinderDragger ()

4.19.2.2 ~RotateCylinderDragger () [protected, virtual]

4.19.3 Member Function Documentation

4.19.3.1 const osg::Vec4 getColor () const [inline]

4.19.3.2 const osg::Vec4 getPickColor () const [inline]

4.19.3.3 META_OSGMANIPULATOR_Object (osgManipulator, RotateCylinderDragger) const

Handle pick events on dragger and generate TranslateInLine commands.

4.19.3.4 void setColor (const osg::Vec4 & *color*) [inline]

Set/Get color for dragger.

4.19.3.5 void setPickColor (const osg::Vec4 & *color*) [inline]

Set/Get pick color for dragger. Pick color is color of the dragger when picked. It gives a visual feedback to show that the dragger has been picked.

4.19.3.6 void setupDefaultGeometry () [virtual]

Setup default geometry for dragger.

Reimplemented from **Dragger** (p. 32).

4.19.4 Member Data Documentation

4.19.4.1 osg::Vec4 _color [protected]

4.19.4.2 osg::Vec4 _pickColor [protected]

4.19.4.3 bool _prevPtOnCylinder [protected]

4.19.4.4 osg::Quat _prevRotation [protected]

4.19.4.5 osg::Vec3d _prevWorldProjPt [protected]

4.19.4.6 osg::ref_ptr<CylinderPlaneProjector> _projector [protected]

4.19.4.7 osg::Matrix _startLocalToWorld [protected]

4.19.4.8 osg::Matrix _startWorldToLocal [protected]

4.19.4.9 const osgGA::GUIEventAdapter& ea

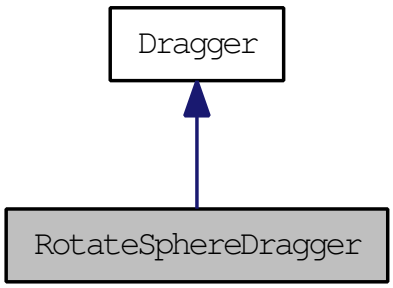
4.19.4.10 const osgGA::GUIEventAdapter osgGA::GUIActionAdapter& us

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

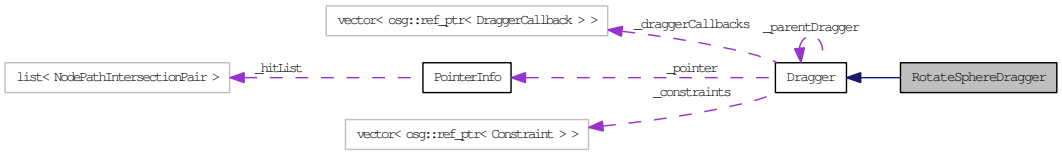
- **RotateCylinderDragger**
- **RotateCylinderDragger.cpp**

4.20 RotateSphereDragger Class Reference

Dragger (p. 29) for performing 3D rotation on a sphere. Inheritance diagram for RotateSphereDragger:



Collaboration diagram for RotateSphereDragger:



Public Member Functions

- **RotateSphereDragger** ()
- const osg::Vec4 **getColor** () const
- const osg::Vec4 **getPickColor** () const
- **META_OSGMANIPULATOR_Object** (osgManipulator, **RotateSphereDragger**) virtual bool handle(const **PointerInfo** &

  Handle pick events on dragger and generate TranslateInLine commands.

- void **setColor** (const osg::Vec4 &color)

  Set/Get color for dragger.

- void **setPickColor** (const osg::Vec4 &color)

  Set/Get pick color for dragger.

- void **setupDefaultGeometry** ()

  Setup default geometry for dragger.

Public Attributes

- const osgGA::GUIEventAdapter & **ea**
- const osgGA::GUIEventAdapter osgGA::GUIActionAdapter & **us**

Protected Member Functions

- virtual ~**RotateSphereDragger** ()

Protected Attributes

- osg::Vec4 **_color**
- osg::Vec4 **_pickColor**
- bool **_prevPtOnSphere**
- osg::Quat **_prevRotation**

- osg::Vec3d **_prevWorldProjPt**
- osg::ref_ptr< **SpherePlaneProjector** > **_projector**
- osg::Matrix **_startLocalToWorld**
- osg::Matrix **_startWorldToLocal**

4.20.1 Detailed Description

Dragger (p. 29) for performing 3D rotation on a sphere.

4.20.2 Constructor & Destructor Documentation

4.20.2.1 RotateSphereDragger ()

4.20.2.2 ~RotateSphereDragger () [protected, virtual]

4.20.3 Member Function Documentation

4.20.3.1 const osg::Vec4 getColor () const [inline]

4.20.3.2 const osg::Vec4 getPickColor () const [inline]

4.20.3.3 META_OSGMANIPULATOR_Object (osgManipulator, RotateSphereDragger) const

Handle pick events on dragger and generate TranslateInLine commands.

4.20.3.4 void setColor (const osg::Vec4 & *color*) [inline]

Set/Get color for dragger.

4.20.3.5 void setPickColor (const osg::Vec4 & *color*) [inline]

Set/Get pick color for dragger. Pick color is color of the dragger when picked. It gives a visual feedback to show that the dragger has been picked.

4.20.3.6 void setupDefaultGeometry () [virtual]

Setup default geometry for dragger.

Reimplemented from **Dragger** (p. 32).

4.20.4 Member Data Documentation

4.20.4.1 osg::Vec4 _color [protected]

4.20.4.2 osg::Vec4 _pickColor [protected]

4.20.4.3 bool _prevPtOnSphere [protected]

4.20.4.4 osg::Quat _prevRotation [protected]

4.20.4.5 osg::Vec3d _prevWorldProjPt [protected]

4.20.4.6 osg::ref_ptr<SpherePlaneProjector> _projector [protected]

4.20.4.7 osg::Matrix _startLocalToWorld [protected]

4.20.4.8 osg::Matrix _startWorldToLocal [protected]

4.20.4.9 const osgGA::GUIEventAdapter& ea

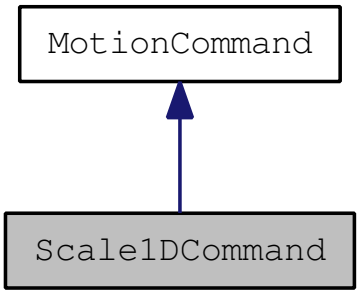
4.20.4.10 const osgGA::GUIEventAdapter osgGA::GUIActionAdapter& us

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

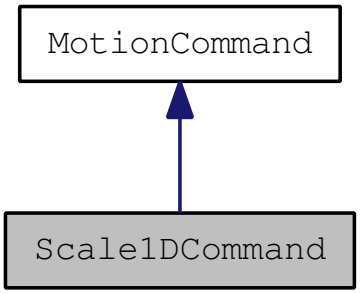
- **RotateSphereDragger**
- **RotateSphereDragger.cpp**

4.21 Scale1DCommand Class Reference

Command for 1D scaling. Inheritance diagram for Scale1DCommand:



Collaboration diagram for Scale1DCommand:



Public Member Functions

- **Scale1DCommand** ()
- virtual **MotionCommand** ∗ **createCommandInverse** ()

  *create a **MotionCommand*** (p. *41) that is the inverse of this command, and if applied will undo this commands changes.*

- double **getMinScale** () const
- virtual osg::Matrix **getMotionMatrix** () const

  Gets the matrix for transforming the object being dragged.

- double **getReferencePoint** () const
- double **getScale** () const
- double **getScaleCenter** () const
- void **setMinScale** (double min)
- void **setReferencePoint** (double rp)

  ReferencePoint is used only for snapping.

- void **setScale** (double s)
- void **setScaleCenter** (double center)

Protected Member Functions

- virtual ∼**Scale1DCommand** ()

4.21.1 Detailed Description

Command for 1D scaling.

4.21.2 Constructor & Destructor Documentation

4.21.2.1 Scale1DCommand ()

4.21.2.2 ~Scale1DCommand () [protected, virtual]

4.21.3 Member Function Documentation

4.21.3.1 MotionCommand * createCommandInverse () [virtual]

create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.

Implements **MotionCommand** (p. 42).

4.21.3.2 double getMinScale () const [inline]

4.21.3.3 virtual osg::Matrix getMotionMatrix () const [inline, virtual]

Gets the matrix for transforming the object being dragged. This matrix is in the command's coordinate systems.

Implements **MotionCommand** (p. 42).

4.21.3.4 double getReferencePoint () const [inline]

4.21.3.5 double getScale () const [inline]

4.21.3.6 double getScaleCenter () const [inline]

4.21.3.7 void setMinScale (double *min*) [inline]

4.21.3.8 void setReferencePoint (double *rp*) [inline]

ReferencePoint is used only for snapping.

4.21.3.9 void setScale (double *s*) [inline]

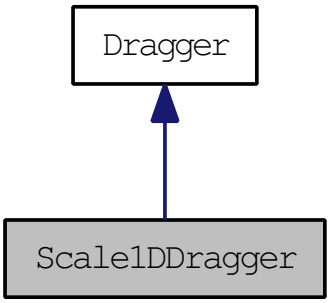
4.21.3.10 void setScaleCenter (double *center*) [inline]

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

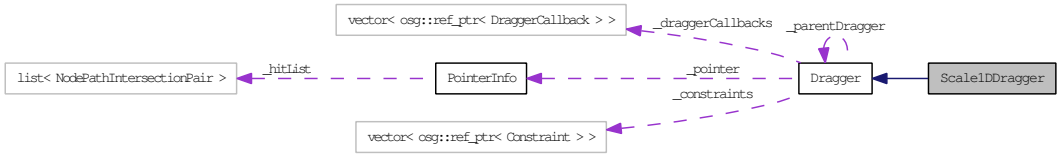
- **Command**
- **Command.cpp**

4.22 Scale1DDragger Class Reference

Dragger (p. 29) for performing 1D scaling. Inheritance diagram for Scale1DDragger:



Collaboration diagram for Scale1DDragger:



Public Types

- enum **ScaleMode** { **SCALE_WITH_ORIGIN_AS_PIVOT** = 0, **SCALE_WITH_OPPOSITE_HANDLE_AS_-PIVOT** }

Public Member Functions

- **Scale1DDragger** (**ScaleMode** scaleMode=SCALE_WITH_ORIGIN_AS_PIVOT)
- const osg::Vec4 **getColor** () const
- osg::Node * **getLeftHandleNode** ()
- double **getLeftHandlePosition** () const
- double **getMinScale** () const
- const osg::Vec4 **getPickColor** () const
- osg::Node * **getRightHandleNode** ()
- double **getRightHandlePosition** ()
- **META_OSGMANIPULATOR_Object** (osgManipulator, **Scale1DDragger**) virtual bool handle(const **PointerInfo** &pi

  Handle pick events on dragger and generate TranslateInLine commands.

- void **setColor** (const osg::Vec4 &color)

  Set/Get color for dragger.

- void **setLeftHandleNode** (osg::Node &node)

  Set/Get left and right handle nodes for dragger.

- void **setLeftHandlePosition** (double pos)

  Set left/right handle position.

- void **setMinScale** (double min)

  Set/Get min scale for dragger.

- void **setPickColor** (const osg::Vec4 &color)

Set/Get pick color for dragger.

- void **setRightHandleNode** (osg::Node &node)
- void **setRightHandlePosition** (double pos)
- void **setupDefaultGeometry** ()

  Setup default geometry for dragger.

Public Attributes

- const osgGA::GUIEventAdapter & **ea**
- const osgGA::GUIEventAdapter osgGA::GUIActionAdapter & **us**

Protected Member Functions

- virtual ∼**Scale1DDragger** ()

Protected Attributes

- osg::Vec4 **_color**
- osg::ref_ptr< osg::Node > **_leftHandleNode**
- double **_minScale**
- osg::Vec4 **_pickColor**
- osg::ref_ptr< **LineProjector** > **_projector**
- osg::ref_ptr< osg::Node > **_rightHandleNode**
- double **_scaleCenter**
- **ScaleMode _scaleMode**
- osg::Vec3d **_startProjectedPoint**

4.22.1 Detailed Description

Dragger (p. 29) for performing 1D scaling.

4.22.2 Member Enumeration Documentation

4.22.2.1 enum ScaleMode

Enumerator:

SCALE_WITH_ORIGIN_AS_PIVOT

SCALE_WITH_OPPOSITE_HANDLE_AS_PIVOT

4.22.3 Constructor & Destructor Documentation

4.22.3.1 Scale1DDragger (ScaleMode *scaleMode* = `SCALE_WITH_ORIGIN_AS_PIVOT`)

4.22.3.2 ∼Scale1DDragger () `[protected, virtual]`

4.22.4 Member Function Documentation

4.22.4.1 const osg::Vec4 getColor () const `[inline]`

4.22.4.2 osg::Node∗ getLeftHandleNode () `[inline]`

4.22.4.3 double getLeftHandlePosition () const `[inline]`

4.22.4.4 double getMinScale () const `[inline]`

4.22.4.5 const osg::Vec4 getPickColor () const `[inline]`

4.22.4.6 osg::Node∗ getRightHandleNode () `[inline]`

4.22.4.7 double getRightHandlePosition () `[inline]`

4.22.4.8 META_OSGMANIPULATOR_Object (osgManipulator, Scale1DDragger) const

Handle pick events on dragger and generate TranslateInLine commands.

4.22.4.9 void setColor (const osg::Vec4 & *color*) [inline]

Set/Get color for dragger.

4.22.4.10 void setLeftHandleNode (osg::Node & *node*) [inline]

Set/Get left and right handle nodes for dragger.

4.22.4.11 void setLeftHandlePosition (double *pos*) [inline]

Set left/right handle position.

4.22.4.12 void setMinScale (double *min*) [inline]

Set/Get min scale for dragger.

4.22.4.13 void setPickColor (const osg::Vec4 & *color*) [inline]

Set/Get pick color for dragger. Pick color is color of the dragger when picked. It gives a visual feedback to show that the dragger has been picked.

4.22.4.14 void setRightHandleNode (osg::Node & *node*) [inline]

4.22.4.15 void setRightHandlePosition (double *pos*) [inline]

4.22.4.16 void setupDefaultGeometry () [virtual]

Setup default geometry for dragger.

Reimplemented from **Dragger** (p. 32).

4.22.5 Member Data Documentation

4.22.5.1 osg::Vec4 _color [protected]

4.22.5.2 osg::ref_ptr< osg::Node > _leftHandleNode [protected]

4.22.5.3 double _minScale [protected]

4.22.5.4 osg::Vec4 _pickColor [protected]

4.22.5.5 osg::ref_ptr< LineProjector > _projector [protected]

4.22.5.6 osg::ref_ptr< osg::Node > _rightHandleNode [protected]

4.22.5.7 double _scaleCenter [protected]

4.22.5.8 ScaleMode _scaleMode [protected]

4.22.5.9 osg::Vec3d _startProjectedPoint [protected]

4.22.5.10 const osgGA::GUIEventAdapter& ea

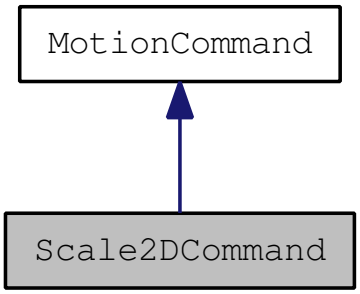
4.22.5.11 const osgGA::GUIEventAdapter osgGA::GUIActionAdapter& us

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

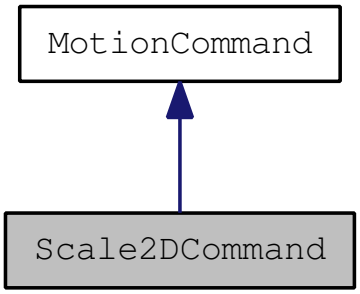
- **Scale1DDragger**
- **Scale1DDragger.cpp**

4.23 Scale2DCommand Class Reference

Command for 2D scaling. Inheritance diagram for Scale2DCommand:



Collaboration diagram for Scale2DCommand:



Public Member Functions

- **Scale2DCommand** ()
- virtual **MotionCommand** ∗ **createCommandInverse** ()

  *create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.*

- const osg::Vec2d & **getMinScale** () const
- virtual osg::Matrix **getMotionMatrix** () const

  Gets the matrix for transforming the object being dragged.

- const osg::Vec2d & **getReferencePoint** () const
- const osg::Vec2d & **getScale** () const
- const osg::Vec2d & **getScaleCenter** () const
- void **setMinScale** (const osg::Vec2d &min)
- void **setReferencePoint** (const osg::Vec2d &rp)

  ReferencePoint is used only for snapping.

- void **setScale** (const osg::Vec2d &s)
- void **setScaleCenter** (const osg::Vec2d &center)

Protected Member Functions

- virtual ∼**Scale2DCommand** ()

4.23.1 Detailed Description

Command for 2D scaling.

4.23.2 Constructor & Destructor Documentation

4.23.2.1 Scale2DCommand ()

4.23.2.2 ∼Scale2DCommand () [protected, virtual]

4.23.3 Member Function Documentation

4.23.3.1 MotionCommand ∗ createCommandInverse () [virtual]

create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.

Implements **MotionCommand** (p. 42).

4.23.3.2 const osg::Vec2d& getMinScale () const [inline]

4.23.3.3 virtual osg::Matrix getMotionMatrix () const [inline, virtual]

Gets the matrix for transforming the object being dragged. This matrix is in the command's coordinate systems.

Implements **MotionCommand** (p. 42).

4.23.3.4 const osg::Vec2d& getReferencePoint () const [inline]

4.23.3.5 const osg::Vec2d& getScale () const [inline]

4.23.3.6 const osg::Vec2d& getScaleCenter () const [inline]

4.23.3.7 void setMinScale (const osg::Vec2d & *min*) [inline]

4.23.3.8 void setReferencePoint (const osg::Vec2d & *rp*) [inline]

ReferencePoint is used only for snapping.

4.23.3.9 void setScale (const osg::Vec2d & *s*) [inline]

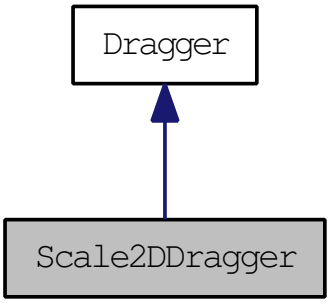
4.23.3.10 void setScaleCenter (const osg::Vec2d & *center*) [inline]

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

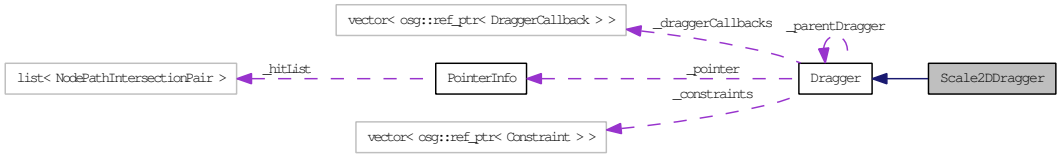
- **Command**
- **Command.cpp**

4.24 Scale2DDragger Class Reference

Dragger (p. 29) for performing 2D scaling. Inheritance diagram for Scale2DDragger:



Collaboration diagram for Scale2DDragger:



Public Types

- enum **ScaleMode** { **SCALE_WITH_ORIGIN_AS_PIVOT** = 0, **SCALE_WITH_OPPOSITE_HANDLE_AS_-PIVOT** }

Public Member Functions

- **Scale2DDragger** (**ScaleMode** scaleMode=SCALE_WITH_ORIGIN_AS_PIVOT)
- osg::Node * **getBottomLeftHandleNode** ()
- const osg::Vec2d & **getBottomLeftHandlePosition** ()
- osg::Node * **getBottomRightHandleNode** ()
- const osg::Vec2d & **getBottomRightHandlePosition** ()
- const osg::Vec4 **getColor** () const
- const osg::Vec2d & **getMinScale** () const
- const osg::Vec4 **getPickColor** () const
- osg::Node * **getTopLeftHandleNode** ()
- const osg::Vec2d & **getTopLeftHandlePosition** ()
- osg::Node * **getTopRightHandleNode** ()
- const osg::Vec2d & **getTopRightHandlePosition** ()
- **META_OSGMANIPULATOR_Object** (osgManipulator, **Scale2DDragger**) virtual bool handle(const **PointerInfo** &pi

  Handle pick events on dragger and generate TranslateInLine commands.

- void **setBottomLeftHandleNode** (osg::Node &node)
- void **setBottomLeftHandlePosition** (const osg::Vec2d &pos)
- void **setBottomRightHandleNode** (osg::Node &node)
- void **setBottomRightHandlePosition** (const osg::Vec2d &pos)
- void **setColor** (const osg::Vec4 &color)

  Set/Get color for dragger.

- void **setMinScale** (const osg::Vec2d &min)

  Set/Get min scale for dragger.

- void **setPickColor** (const osg::Vec4 &color)

  Set/Get pick color for dragger.

- void **setTopLeftHandleNode** (osg::Node &node)

  Set/Get the handle nodes for dragger.

- void **setTopLeftHandlePosition** (const osg::Vec2d &pos)

  Set/Get the handle nodes postion for dragger.

- void **setTopRightHandleNode** (osg::Node &node)
- void **setTopRightHandlePosition** (const osg::Vec2d &pos)
- void **setupDefaultGeometry** ()

  Setup default geometry for dragger.

Public Attributes

- const osgGA::GUIEventAdapter & **ea**
- const osgGA::GUIEventAdapter osgGA::GUIActionAdapter & **us**

Protected Member Functions

- virtual ∼**Scale2DDragger** ()

Protected Attributes

- osg::ref_ptr< osg::Node > **_bottomLeftHandleNode**
- osg::Vec2d **_bottomLeftHandlePosition**
- osg::ref_ptr< osg::Node > **_bottomRightHandleNode**
- osg::Vec2d **_bottomRightHandlePosition**
- osg::Vec4 **_color**
- osg::Vec2d **_minScale**
- osg::Vec4 **_pickColor**
- osg::ref_ptr< **PlaneProjector** > **_projector**
- osg::Vec2d **_referencePoint**
- osg::Vec2d **_scaleCenter**
- **ScaleMode _scaleMode**
- osg::Vec3d **_startProjectedPoint**
- osg::ref_ptr< osg::Node > **_topLeftHandleNode**
- osg::Vec2d **_topLeftHandlePosition**
- osg::ref_ptr< osg::Node > **_topRightHandleNode**
- osg::Vec2d **_topRightHandlePosition**

4.24.1 Detailed Description

Dragger (p. 29) for performing 2D scaling.

4.24.2 Member Enumeration Documentation

4.24.2.1 enum ScaleMode

Enumerator:

SCALE_WITH_ORIGIN_AS_PIVOT

SCALE_WITH_OPPOSITE_HANDLE_AS_PIVOT

4.24.3 Constructor & Destructor Documentation

4.24.3.1 Scale2DDragger (ScaleMode *scaleMode* = SCALE_WITH_ORIGIN_AS_PIVOT)

4.24.3.2 ~Scale2DDragger () [protected, virtual]

4.24.4 Member Function Documentation

4.24.4.1 osg::Node* getBottomLeftHandleNode () [inline]

4.24.4.2 const osg::Vec2d& getBottomLeftHandlePosition () [inline]

4.24.4.3 osg::Node* getBottomRightHandleNode () [inline]

4.24.4.4 const osg::Vec2d& getBottomRightHandlePosition () [inline]

4.24.4.5 const osg::Vec4 getColor () const [inline]

4.24.4.6 const osg::Vec2d& getMinScale () const [inline]

4.24.4.7 const osg::Vec4 getPickColor () const [inline]

4.24.4.8 osg::Node* getTopLeftHandleNode () [inline]

4.24.4.9 const osg::Vec2d& getTopLeftHandlePosition () [inline]

4.24.4.10 osg::Node* getTopRightHandleNode () [inline]

4.24.4.11 const osg::Vec2d& getTopRightHandlePosition () [inline]

4.24.4.12 META_OSGMANIPULATOR_Object (osgManipulator, Scale2DDragger) const

Handle pick events on dragger and generate TranslateInLine commands.

4.24.4.13 void setBottomLeftHandleNode (osg::Node & *node*) [inline]

4.24.4.14 void setBottomLeftHandlePosition (const osg::Vec2d & *pos*) [inline]

4.24.4.15 void setBottomRightHandleNode (osg::Node & *node*) [inline]

4.24.4.16 void setBottomRightHandlePosition (const osg::Vec2d & *pos*) [inline]

4.24.4.17 void setColor (const osg::Vec4 & *color*) [inline]

Set/Get color for dragger.

4.24.4.18 void setMinScale (const osg::Vec2d & *min*) [inline]

Set/Get min scale for dragger.

4.24.4.19 void setPickColor (const osg::Vec4 & *color*) [inline]

Set/Get pick color for dragger. Pick color is color of the dragger when picked. It gives a visual feedback to show that the dragger has been picked.

4.24.4.20 void setTopLeftHandleNode (osg::Node & *node*) [inline]

Set/Get the handle nodes for dragger.

4.24.4.21 void setTopLeftHandlePosition (const osg::Vec2d & *pos*) [inline]

Set/Get the handle nodes postion for dragger.

4.24.4.22 void setTopRightHandleNode (osg::Node & *node*) [inline]

4.24.4.23 void setTopRightHandlePosition (const osg::Vec2d & *pos*) [inline]

4.24.4.24 void setupDefaultGeometry () [virtual]

Setup default geometry for dragger.

Reimplemented from **Dragger** (p. 32).

4.24.5 Member Data Documentation

4.24.5.1 osg::ref_ptr< osg::Node > _bottomLeftHandleNode [protected]

4.24.5.2 osg::Vec2d _bottomLeftHandlePosition [protected]

4.24.5.3 osg::ref_ptr< osg::Node > _bottomRightHandleNode [protected]

4.24.5.4 osg::Vec2d _bottomRightHandlePosition [protected]

4.24.5.5 osg::Vec4 _color [protected]

4.24.5.6 osg::Vec2d _minScale [protected]

4.24.5.7 osg::Vec4 _pickColor [protected]

4.24.5.8 osg::ref_ptr< PlaneProjector > _projector [protected]

4.24.5.9 osg::Vec2d _referencePoint [protected]

4.24.5.10 osg::Vec2d _scaleCenter [protected]

4.24.5.11 ScaleMode _scaleMode [protected]

4.24.5.12 osg::Vec3d _startProjectedPoint [protected]

4.24.5.13 osg::ref_ptr< osg::Node > _topLeftHandleNode [protected]

4.24.5.14 osg::Vec2d _topLeftHandlePosition [protected]

4.24.5.15 osg::ref_ptr< osg::Node > _topRightHandleNode [protected]

4.24.5.16 osg::Vec2d _topRightHandlePosition [protected]

4.24.5.17 const osgGA::GUIEventAdapter& ea

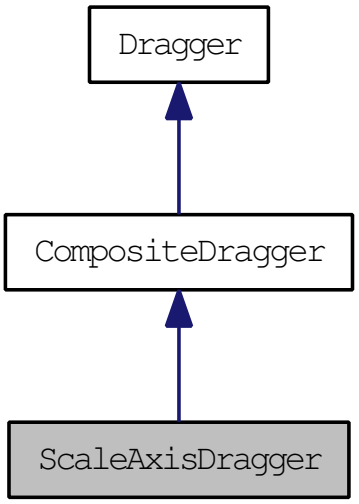
4.24.5.18 const osgGA::GUIEventAdapter osgGA::GUIActionAdapter& us

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

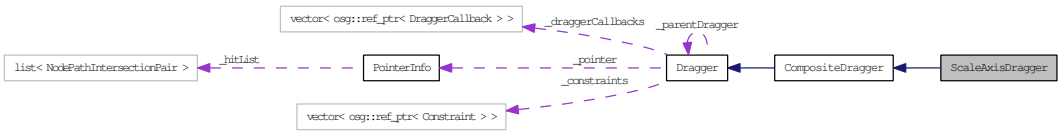
- Scale2DDragger
- Scale2DDragger.cpp

4.25 ScaleAxisDragger Class Reference

Dragger (p. 29) for performing scaling on all 3 axes. Inheritance diagram for ScaleAxisDragger:



Collaboration diagram for ScaleAxisDragger:



Public Member Functions

- **ScaleAxisDragger** ()
- **META_OSGMANIPULATOR_Object** (osgManipulator, **ScaleAxisDragger**) void setupDefaultGeometry()

  Setup default geometry for dragger.

Protected Member Functions

- virtual ∼**ScaleAxisDragger** ()

Protected Attributes

- osg::ref_ptr< **Scale1DDragger** > **_xDragger**
- osg::ref_ptr< **Scale1DDragger** > **_yDragger**
- osg::ref_ptr< **Scale1DDragger** > **_zDragger**

4.25.1 Detailed Description

Dragger (p. 29) for performing scaling on all 3 axes.

4.25.2 Constructor & Destructor Documentation

4.25.2.1 ScaleAxisDragger ()

4.25.2.2 ∼ScaleAxisDragger () [protected, virtual]

4.25.3 Member Function Documentation

4.25.3.1 META_OSGMANIPULATOR_Object (osgManipulator, ScaleAxisDragger)

Setup default geometry for dragger.

4.25.4 Member Data Documentation

4.25.4.1 osg::ref_ptr< Scale1DDragger > _xDragger [protected]

4.25.4.2 osg::ref_ptr< Scale1DDragger > _yDragger [protected]

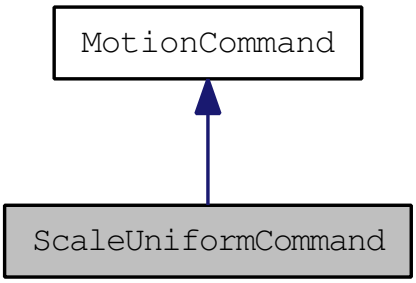
4.25.4.3 osg::ref_ptr< Scale1DDragger > _zDragger [protected]

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

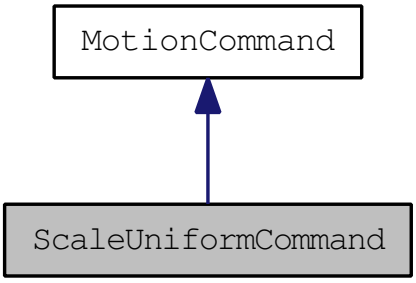
- **ScaleAxisDragger**
- **ScaleAxisDragger.cpp**

4.26 ScaleUniformCommand Class Reference

Command for uniform 3D scaling. Inheritance diagram for ScaleUniformCommand:



Collaboration diagram for ScaleUniformCommand:



Public Member Functions

- **ScaleUniformCommand** ()
- virtual **MotionCommand** ∗ **createCommandInverse** ()

  *create a **MotionCommand** (*p. *41) that is the inverse of this command, and if applied will undo this commands changes.*

- virtual osg::Matrix **getMotionMatrix** () const

  Gets the matrix for transforming the object being dragged.

- double **getScale** () const
- const osg::Vec3d & **getScaleCenter** () const
- void **setScale** (double s)
- void **setScaleCenter** (const osg::Vec3d &center)

Protected Member Functions

- virtual ∼**ScaleUniformCommand** ()

4.26.1 Detailed Description

Command for uniform 3D scaling.

4.26.2 Constructor & Destructor Documentation

4.26.2.1 ScaleUniformCommand ()

4.26.2.2 ∼ScaleUniformCommand () [protected, virtual]

4.26.3 Member Function Documentation

4.26.3.1 MotionCommand ∗ createCommandInverse () [virtual]

create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.

Implements **MotionCommand** (p. 42).

4.26.3.2 virtual osg::Matrix getMotionMatrix () const [inline, virtual]

Gets the matrix for transforming the object being dragged. This matrix is in the command's coordinate systems.

Implements **MotionCommand** (p. 42).

4.26.3.3 double getScale () const [inline]

4.26.3.4 const osg::Vec3d& getScaleCenter () const [inline]

4.26.3.5 void setScale (double *s*) [inline]

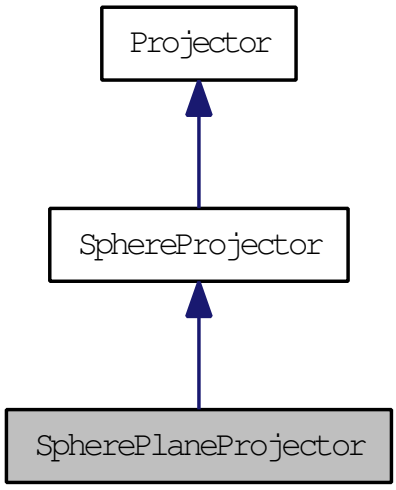
4.26.3.6 void setScaleCenter (const osg::Vec3d & *center*) [inline]

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

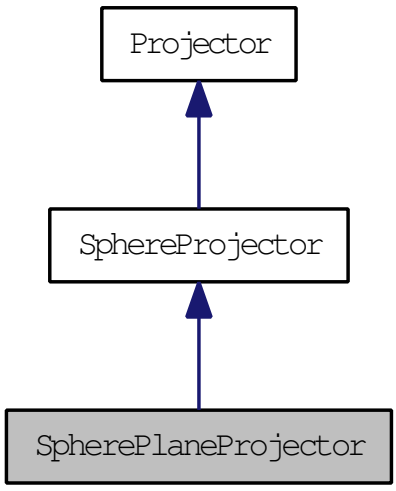
- **Command**
- **Command.cpp**

4.27 SpherePlaneProjector Class Reference

SpherePlaneProjector (p. 70) projects points onto a sphere, failing which it project onto a plane oriented to the viewing direction. Inheritance diagram for SpherePlaneProjector:



Collaboration diagram for SpherePlaneProjector:



Public Member Functions

- **SpherePlaneProjector** (osg::Sphere ∗sphere)
- **SpherePlaneProjector** ()
- osg::Quat **getRotation** (const osg::Vec3d &p1, bool p1OnSphere, const osg::Vec3d &p2, bool p2OnSphere, float radialFactor=0.0f) const
- bool **isProjectionOnSphere** () const

  Returns true if the previous projection was on the sphere and false if the projection was on the plane.

- virtual bool **project** (const **PointerInfo** &pi, osg::Vec3d &projectedPoint) const

  Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given sphere.

Protected Member Functions

- virtual ∼**SpherePlaneProjector** ()

Protected Attributes

- bool **_onSphere**
- osg::Plane **_plane**

4.27.1 Detailed Description

SpherePlaneProjector (p. 70) projects points onto a sphere, failing which it project onto a plane oriented to the viewing direction.

4.27.2 Constructor & Destructor Documentation

4.27.2.1 SpherePlaneProjector ()

4.27.2.2 SpherePlaneProjector (osg::Sphere * *sphere*)

4.27.2.3 ~SpherePlaneProjector () [protected, virtual]

4.27.3 Member Function Documentation

4.27.3.1 osg::Quat getRotation (const osg::Vec3d & *p1*, bool *p1OnSphere*, const osg::Vec3d & *p2*, bool *p2OnSphere*, float *radialFactor* = 0.0f) const

4.27.3.2 bool isProjectionOnSphere () const [inline]

Returns true if the previous projection was on the sphere and false if the projection was on the plane.

4.27.3.3 bool project (const PointerInfo & *pi*, osg::Vec3d & *projectedPoint*) const [virtual]

Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given sphere. Returns true on successful projection.

Reimplemented from **SphereProjector** (p. 73).

4.27.4 Member Data Documentation

4.27.4.1 bool _onSphere [mutable, protected]

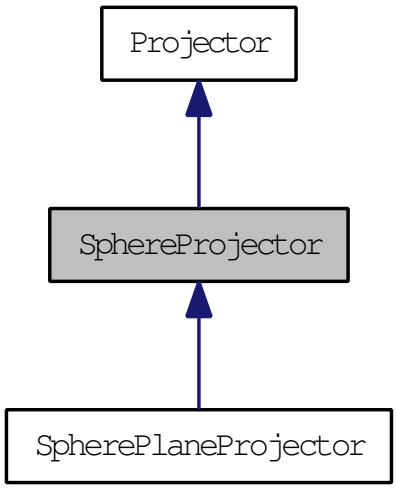
4.27.4.2 osg::Plane _plane [mutable, protected]

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

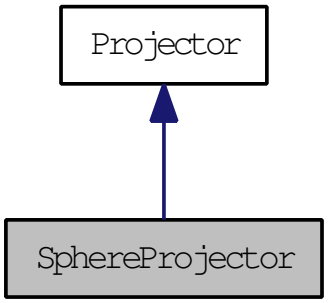
- **Projector**
- **Projector.cpp**

4.28 SphereProjector Class Reference

SphereProjector (p. 72) projects points onto the given sphere. Inheritance diagram for SphereProjector:



Collaboration diagram for SphereProjector:



Public Member Functions

- **SphereProjector** (osg::Sphere ∗sphere)
- **SphereProjector** ()
- const osg::Sphere ∗ **getSphere** () const
- bool **isPointInFront** (const **PointerInfo** &pi, const osg::Matrix &localToWorld) const

  Returns true is the point is in front of the cylinder given the eye direction.

- virtual bool **project** (const **PointerInfo** &pi, osg::Vec3d &projectedPoint) const

  Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given sphere.

- void **setFront** (bool front)
- void **setSphere** (osg::Sphere ∗sphere)

Protected Member Functions

- virtual ∼**SphereProjector** ()

Protected Attributes

- bool **_front**
- osg::ref_ptr< osg::Sphere > **_sphere**

4.28.1 Detailed Description

SphereProjector (p. 72) projects points onto the given sphere.

4.28.2 Constructor & Destructor Documentation

4.28.2.1 SphereProjector ()

4.28.2.2 SphereProjector (osg::Sphere * *sphere*)

4.28.2.3 ~SphereProjector () [protected, virtual]

4.28.3 Member Function Documentation

4.28.3.1 const osg::Sphere* getSphere () const [inline]

4.28.3.2 bool isPointInFront (const PointerInfo & *pi*, const osg::Matrix & *localToWorld*) const

Returns true is the point is in front of the cylinder given the eye direction.

4.28.3.3 bool project (const PointerInfo & *pi*, osg::Vec3d & *projectedPoint*) const [virtual]

Calculates the object coordinates (projectedPoint) of a window coordinate (pointToProject) when projected onto the given sphere. Returns true on successful projection.

Implements **Projector** (p. 48).

Reimplemented in **SpherePlaneProjector** (p. 71).

4.28.3.4 void setFront (bool *front*) [inline]

4.28.3.5 void setSphere (osg::Sphere * *sphere*) [inline]

4.28.4 Member Data Documentation

4.28.4.1 bool _front [protected]

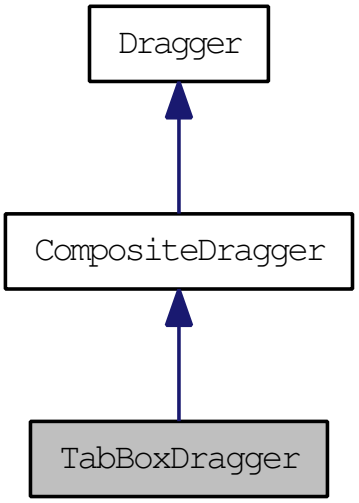
4.28.4.2 osg::ref_ptr<osg::Sphere> _sphere [protected]

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

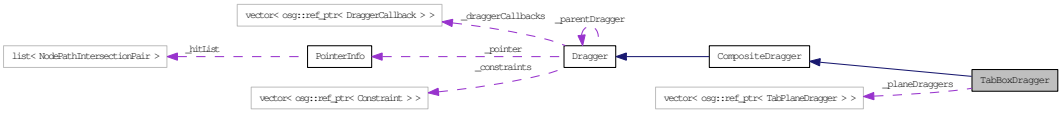
- **Projector**
- **Projector.cpp**

4.29 TabBoxDragger Class Reference

TabBoxDragger (p. 74) consists of 6 TabPlaneDraggers to form a box dragger that performs translation and scaling. Inheritance diagram for TabBoxDragger:



Collaboration diagram for TabBoxDragger:



Public Member Functions

- **TabBoxDragger** ()
- **META_OSGMANIPULATOR_Object** (osgManipulator, **TabBoxDragger**) void setupDefaultGeometry()
  Setup default geometry for dragger.
- void **setPlaneColor** (const osg::Vec4 &color)

Protected Member Functions

- virtual ∼**TabBoxDragger** ()

Protected Attributes

- std::vector< osg::ref_ptr< **TabPlaneDragger** > > **_planeDraggers**

4.29.1 Detailed Description

TabBoxDragger (p. 74) consists of 6 TabPlaneDraggers to form a box dragger that performs translation and scaling.

4.29.2 Constructor & Destructor Documentation

4.29.2.1 TabBoxDragger ()

4.29.2.2 ∼TabBoxDragger () [protected, virtual]

4.29.3 Member Function Documentation

4.29.3.1 META_OSGMANIPULATOR_Object (osgManipulator, TabBoxDragger)

Setup default geometry for dragger.

4.29.3.2 void setPlaneColor (const osg::Vec4 & *color*)

4.29.4 Member Data Documentation

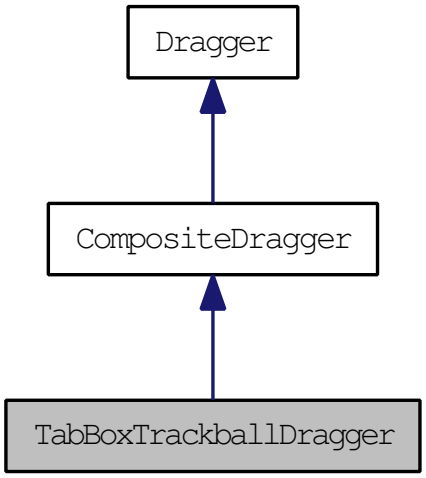
4.29.4.1 std::vector< osg::ref_ptr< TabPlaneDragger > > _planeDraggers [protected]

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

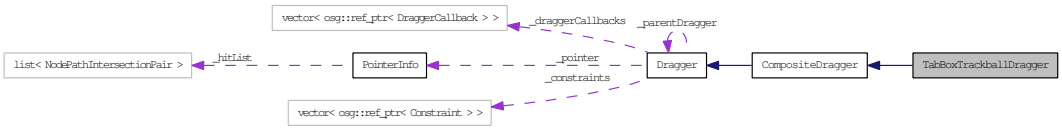
- **TabBoxDragger**
- **TabBoxDragger.cpp**

4.30 TabBoxTrackballDragger Class Reference

Dragger (p. 29) for performing rotation in all axes. Inheritance diagram for TabBoxTrackballDragger:



Collaboration diagram for TabBoxTrackballDragger:



Public Member Functions

- **TabBoxTrackballDragger** ()
- **META_OSGMANIPULATOR_Object** (osgManipulator, **TabBoxTrackballDragger**) void setupDefaultGeometry()

  Setup default geometry for dragger.

Protected Member Functions

- virtual ~**TabBoxTrackballDragger** ()

Protected Attributes

- osg::ref_ptr< **TabBoxDragger** > **_tabBoxDragger**
- osg::ref_ptr< **TrackballDragger** > **_trackballDragger**

4.30.1 Detailed Description

Dragger (p. 29) for performing rotation in all axes.

4.30.2 Constructor & Destructor Documentation

4.30.2.1 TabBoxTrackballDragger ()

4.30.2.2 ~TabBoxTrackballDragger () [protected, virtual]

4.30.3 Member Function Documentation

4.30.3.1 META_OSGMANIPULATOR_Object (osgManipulator, TabBoxTrackballDragger)

Setup default geometry for dragger.

4.30.4 Member Data Documentation

4.30.4.1 osg::ref_ptr<TabBoxDragger> _tabBoxDragger [protected]

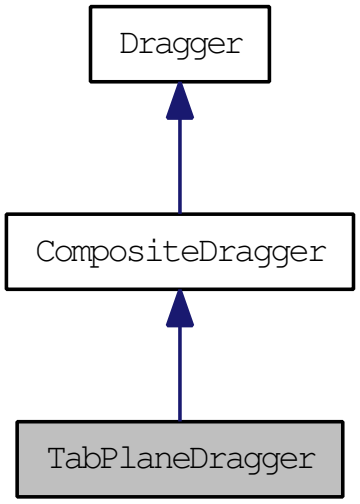
4.30.4.2 osg::ref_ptr<TrackballDragger> _trackballDragger [protected]

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

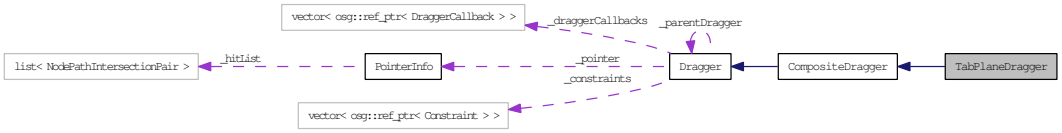
- **TabBoxTrackballDragger**
- **TabBoxTrackballDragger.cpp**

4.31 TabPlaneDragger Class Reference

Tab plane dragger consists of a plane with tabs on it's corners and edges for scaling. Inheritance diagram for TabPlaneDragger:



Collaboration diagram for TabPlaneDragger:



Public Member Functions

- **TabPlaneDragger** ()
- **META_OSGMANIPULATOR_Object** (osgManipulator, **TabPlaneDragger**) virtual bool handle(const **PointerInfo** &pi
- void **setPlaneColor** (const osg::Vec4 &color)
- void **setupDefaultGeometry** (bool twoSidedHandle=true)

  Setup default geometry for dragger.

Public Attributes

- const osgGA::GUIEventAdapter & **ea**
- const osgGA::GUIEventAdapter osgGA::GUIActionAdapter & **us**

Protected Member Functions

- virtual ~**TabPlaneDragger** ()

Protected Attributes

- osg::ref_ptr< **Scale2DDragger** > **_cornerScaleDragger**
- float **_handleScaleFactor**
- osg::ref_ptr< **Scale1DDragger** > **_horzEdgeScaleDragger**
- osg::ref_ptr< **TranslatePlaneDragger** > **_translateDragger**
- osg::ref_ptr< **Scale1DDragger** > **_vertEdgeScaleDragger**

4.31.1 Detailed Description

Tab plane dragger consists of a plane with tabs on it's corners and edges for scaling. And the plane is used as a 2D translate dragger.

4.31.2 Constructor & Destructor Documentation

4.31.2.1 TabPlaneDragger ()

4.31.2.2 ~TabPlaneDragger () [protected, virtual]

4.31.3 Member Function Documentation

4.31.3.1 META_OSGMANIPULATOR_Object (osgManipulator, TabPlaneDragger) const

4.31.3.2 void setPlaneColor (const osg::Vec4 & *color*) [inline]

4.31.3.3 void setupDefaultGeometry (bool *twoSidedHandle* = true)

Setup default geometry for dragger.

4.31.4 Member Data Documentation

4.31.4.1 osg::ref_ptr< Scale2DDragger > _cornerScaleDragger [protected]

4.31.4.2 float _handleScaleFactor [protected]

4.31.4.3 osg::ref_ptr< Scale1DDragger > _horzEdgeScaleDragger [protected]

4.31.4.4 osg::ref_ptr< TranslatePlaneDragger > _translateDragger [protected]

4.31.4.5 osg::ref_ptr< Scale1DDragger > _vertEdgeScaleDragger [protected]

4.31.4.6 const osgGA::GUIEventAdapter& ea

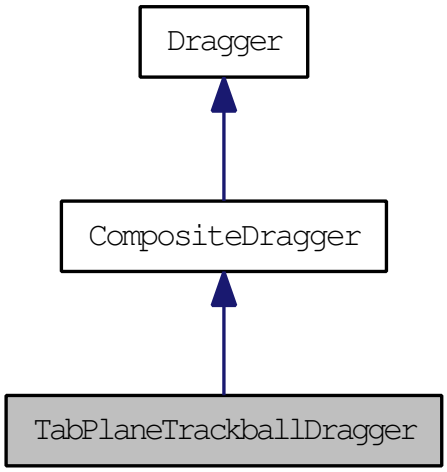
4.31.4.7 const osgGA::GUIEventAdapter osgGA::GUIActionAdapter& us

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

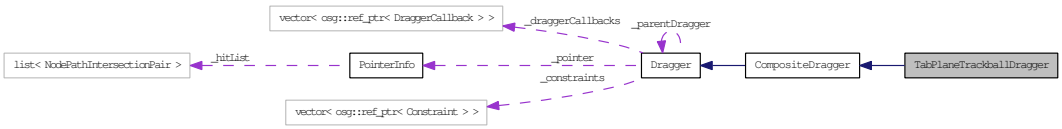
- TabPlaneDragger
- TabPlaneDragger.cpp

4.32 TabPlaneTrackballDragger Class Reference

Dragger (p. 29) for performing rotation in all axes. Inheritance diagram for TabPlaneTrackballDragger:



Collaboration diagram for TabPlaneTrackballDragger:



Public Member Functions

- **TabPlaneTrackballDragger** ()
- **META_OSGMANIPULATOR_Object** (osgManipulator, **TabPlaneTrackballDragger**) void setupDefaultGeometry()

  Setup default geometry for dragger.

- void **setPlaneColor** (const osg::Vec4 &color)

Protected Member Functions

- virtual ∼**TabPlaneTrackballDragger** ()

Protected Attributes

- osg::ref_ptr< **TabPlaneDragger** > **_tabPlaneDragger**
- osg::ref_ptr< **TrackballDragger** > **_trackballDragger**

4.32.1 Detailed Description

Dragger (p. 29) for performing rotation in all axes.

4.32.2 Constructor & Destructor Documentation

4.32.2.1 TabPlaneTrackballDragger ()

4.32.2.2 ∼TabPlaneTrackballDragger () `[protected, virtual]`

4.32.3 Member Function Documentation

4.32.3.1 META_OSGMANIPULATOR_Object (osgManipulator, TabPlaneTrackballDragger)

Setup default geometry for dragger.

4.32.3.2 void setPlaneColor (const osg::Vec4 & *color*) [inline]

4.32.4 Member Data Documentation

4.32.4.1 osg::ref_ptr<TabPlaneDragger> _tabPlaneDragger [protected]

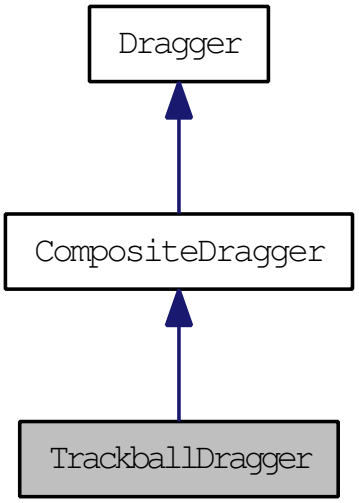
4.32.4.2 osg::ref_ptr<TrackballDragger> _trackballDragger [protected]

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

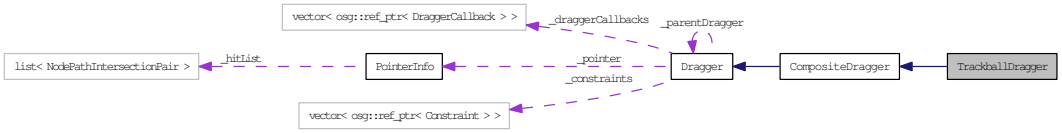
- **TabPlaneTrackballDragger**
- **TabPlaneTrackballDragger.cpp**

4.33 TrackballDragger Class Reference

Dragger (p. 29) for performing rotation in all axes. Inheritance diagram for TrackballDragger:



Collaboration diagram for TrackballDragger:



Public Member Functions

- **TrackballDragger** (bool useAutoTransform=false)
- **META_OSGMANIPULATOR_Object** (osgManipulator, **TrackballDragger**) void setupDefaultGeometry()

  Setup default geometry for dragger.

Protected Member Functions

- virtual ∼**TrackballDragger** ()

Protected Attributes

- osg::ref_ptr< **RotateCylinderDragger** > **_xDragger**
- osg::ref_ptr< **RotateSphereDragger** > **_xyzDragger**
- osg::ref_ptr< **RotateCylinderDragger** > **_yDragger**
- osg::ref_ptr< **RotateCylinderDragger** > **_zDragger**

4.33.1 Detailed Description

Dragger (p. 29) for performing rotation in all axes.

4.33.2 Constructor & Destructor Documentation

4.33.2.1 TrackballDragger (bool *useAutoTransform* = `false`)

4.33.2.2 ~TrackballDragger () `[protected, virtual]`

4.33.3 Member Function Documentation

4.33.3.1 META_OSGMANIPULATOR_Object (osgManipulator, TrackballDragger)

Setup default geometry for dragger.

4.33.4 Member Data Documentation

4.33.4.1 osg::ref_ptr<RotateCylinderDragger> _xDragger `[protected]`

4.33.4.2 osg::ref_ptr<RotateSphereDragger> _xyzDragger `[protected]`

4.33.4.3 osg::ref_ptr<RotateCylinderDragger> _yDragger `[protected]`

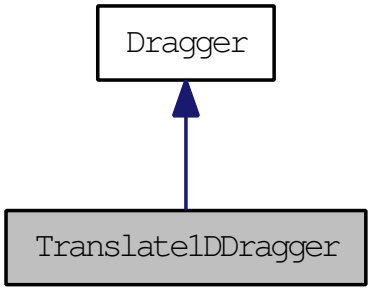
4.33.4.4 osg::ref_ptr<RotateCylinderDragger> _zDragger `[protected]`

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

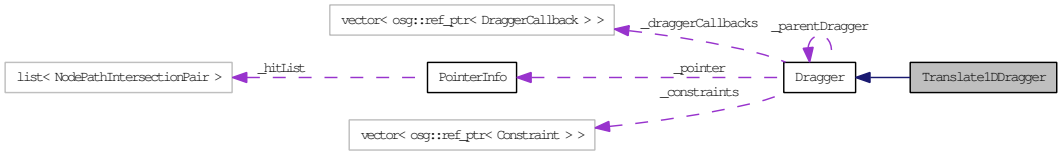
- **TrackballDragger**
- **TrackballDragger.cpp**

4.34 Translate1DDragger Class Reference

Dragger (p. 29) for performing 1D translation. Inheritance diagram for Translate1DDragger:



Collaboration diagram for Translate1DDragger:



Public Member Functions

- **Translate1DDragger** ()
- const osg::Vec4 **getColor** () const
- const osg::Vec4 **getPickColor** () const
- void **setCheckForNodeInNodePath** (bool onOff)
- void **setColor** (const osg::Vec4 &color)

  Set/Get color for dragger.

- void **setPickColor** (const osg::Vec4 &color)

  Set/Get pick color for dragger.

- void **setupDefaultGeometry** ()

  Setup default geometry for dragger.

Protected Member Functions

- virtual ∼**Translate1DDragger** ()

Protected Attributes

- bool **_checkForNodeInNodePath**
- osg::Vec4 **_color**
- osg::Vec4 **_pickColor**
- osg::ref_ptr< **LineProjector** > **_projector**
- osg::Vec3d **_startProjectedPoint**

4.34.1 Detailed Description

Dragger (p. 29) for performing 1D translation.

4.34.2 Constructor & Destructor Documentation

4.34.2.1 Translate1DDragger ()

4.34.2.2 ~Translate1DDragger () [protected, virtual]

4.34.3 Member Function Documentation

4.34.3.1 const osg::Vec4 getColor () const [inline]

4.34.3.2 const osg::Vec4 getPickColor () const [inline]

4.34.3.3 void setCheckForNodeInNodePath (bool *onOff*) [inline]

4.34.3.4 void setColor (const osg::Vec4 & *color*) [inline]

Set/Get color for dragger.

4.34.3.5 void setPickColor (const osg::Vec4 & *color*) [inline]

Set/Get pick color for dragger. Pick color is color of the dragger when picked. It gives a visual feedback to show that the dragger has been picked.

4.34.3.6 void setupDefaultGeometry () [virtual]

Setup default geometry for dragger.

Reimplemented from **Dragger** (p. 32).

4.34.4 Member Data Documentation

4.34.4.1 bool _checkForNodeInNodePath [protected]

4.34.4.2 osg::Vec4 _color [protected]

4.34.4.3 osg::Vec4 _pickColor [protected]

4.34.4.4 osg::ref_ptr< LineProjector > _projector [protected]

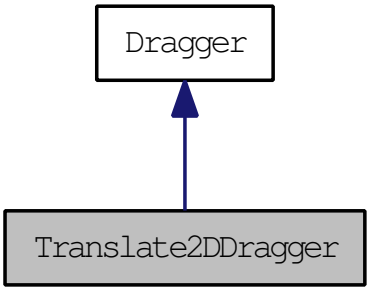
4.34.4.5 osg::Vec3d _startProjectedPoint [protected]

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

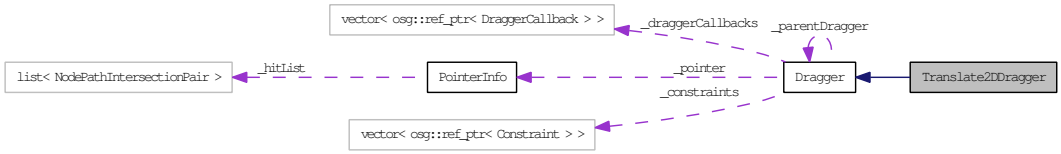
- **Translate1DDragger**
- **Translate1DDragger.cpp**

4.35 Translate2DDragger Class Reference

Dragger (p. 29) for performing 2D translation. Inheritance diagram for Translate2DDragger:



Collaboration diagram for Translate2DDragger:



Public Member Functions

- **Translate2DDragger** (const osg::Plane &plane)
- **Translate2DDragger** ()
- const osg::Vec4 **getColor** () const
- const osg::Vec4 **getPickColor** () const
- **META_OSGMANIPULATOR_Object** (osgManipulator, **Translate2DDragger**) virtual bool handle(const **PointerInfo** &pi

  Handle pick events on dragger and generate TranslateInLine commands.

- void **setColor** (const osg::Vec4 &color)

  Set/Get color for dragger.

- void **setPickColor** (const osg::Vec4 &color)

  Set/Get pick color for dragger.

- void **setupDefaultGeometry** ()

  Setup default geometry for dragger.

Public Attributes

- const osgGA::GUIEventAdapter & **ea**
- const osgGA::GUIEventAdapter osgGA::GUIActionAdapter & **us**

Protected Member Functions

- virtual ∼**Translate2DDragger** ()

Protected Attributes

- osg::Vec4 **_color**
- osg::Vec4 **_pickColor**
- osg::ref_ptr< osg::PolygonOffset > **_polygonOffset**
- osg::ref_ptr< **PlaneProjector** > **_projector**
- osg::Vec3d **_startProjectedPoint**

4.35.1 Detailed Description

Dragger (p. 29) for performing 2D translation.

4.35.2 Constructor & Destructor Documentation

4.35.2.1 Translate2DDragger ()

4.35.2.2 Translate2DDragger (const osg::Plane & *plane*)

4.35.2.3 ∼Translate2DDragger () [protected, virtual]

4.35.3 Member Function Documentation

4.35.3.1 const osg::Vec4 getColor () const [inline]

4.35.3.2 const osg::Vec4 getPickColor () const [inline]

4.35.3.3 META_OSGMANIPULATOR_Object (osgManipulator, Translate2DDragger) const

Handle pick events on dragger and generate TranslateInLine commands.

4.35.3.4 void setColor (const osg::Vec4 & *color*) [inline]

Set/Get color for dragger.

4.35.3.5 void setPickColor (const osg::Vec4 & *color*) [inline]

Set/Get pick color for dragger. Pick color is color of the dragger when picked. It gives a visual feedback to show that the dragger has been picked.

4.35.3.6 void setupDefaultGeometry () [virtual]

Setup default geometry for dragger.

Reimplemented from **Dragger** (p. 32).

4.35.4 Member Data Documentation

4.35.4.1 osg::Vec4 _color [protected]

4.35.4.2 osg::Vec4 _pickColor [protected]

4.35.4.3 osg::ref_ptr<osg::PolygonOffset> _polygonOffset [protected]

4.35.4.4 osg::ref_ptr< PlaneProjector > _projector [protected]

4.35.4.5 osg::Vec3d _startProjectedPoint [protected]

4.35.4.6 const osgGA::GUIEventAdapter& ea

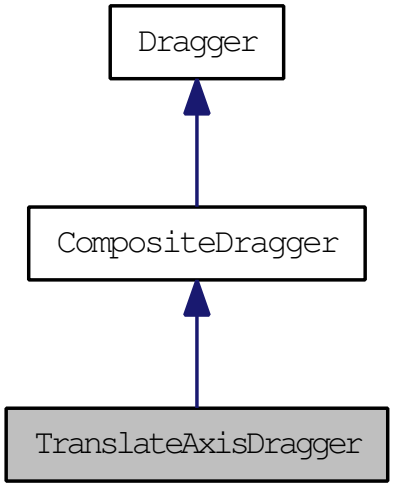
4.35.4.7 const osgGA::GUIEventAdapter osgGA::GUIActionAdapter& us

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

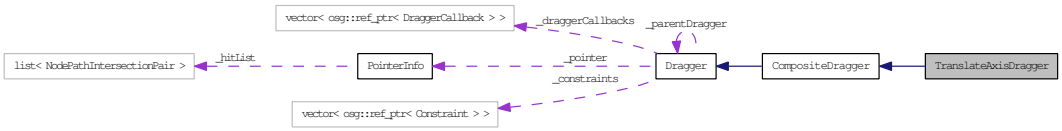
- **Translate2DDragger**
- **Translate2DDragger.cpp**

4.36 TranslateAxisDragger Class Reference

Dragger (p. 29) for performing translation in all three axes. Inheritance diagram for TranslateAxisDragger:



Collaboration diagram for TranslateAxisDragger:



Public Member Functions

- **TranslateAxisDragger** ()
- **META_OSGMANIPULATOR_Object** (osgManipulator, **TranslateAxisDragger**) void setupDefaultGeometry()

  Setup default geometry for dragger.

Protected Member Functions

- virtual ~**TranslateAxisDragger** ()

Protected Attributes

- osg::ref_ptr< **Translate1DDragger** > **_xDragger**
- osg::ref_ptr< **Translate1DDragger** > **_yDragger**
- osg::ref_ptr< **Translate1DDragger** > **_zDragger**

4.36.1 Detailed Description

Dragger (p. 29) for performing translation in all three axes.

4.36.2 Constructor & Destructor Documentation

4.36.2.1 TranslateAxisDragger ()

4.36.2.2 ~TranslateAxisDragger () `[protected, virtual]`

4.36.3 Member Function Documentation

4.36.3.1 META_OSGMANIPULATOR_Object (osgManipulator, TranslateAxisDragger)

Setup default geometry for dragger.

4.36.4 Member Data Documentation

4.36.4.1 osg::ref_ptr< Translate1DDragger > _xDragger [protected]

4.36.4.2 osg::ref_ptr< Translate1DDragger > _yDragger [protected]

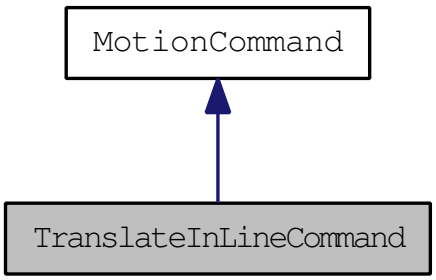
4.36.4.3 osg::ref_ptr< Translate1DDragger > _zDragger [protected]

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

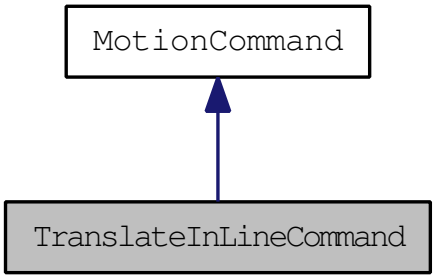
- **TranslateAxisDragger**
- **TranslateAxisDragger.cpp**

4.37 TranslateInLineCommand Class Reference

Command for translating in a line. Inheritance diagram for TranslateInLineCommand:



Collaboration diagram for TranslateInLineCommand:



Public Member Functions

- **TranslateInLineCommand** (const osg::LineSegment::vec_type &s, const osg::LineSegment::vec_type &e)
- **TranslateInLineCommand** ()
- virtual **MotionCommand** ∗ **createCommandInverse** ()

  *create a **MotionCommand*** (p. 41) *that is the inverse of this command, and if applied will undo this commands changes.*

- const osg::LineSegment::vec_type & **getLineEnd** () const
- const osg::LineSegment::vec_type & **getLineStart** () const
- virtual osg::Matrix **getMotionMatrix** () const

  Gets the matrix for transforming the object being dragged.

- const osg::Vec3d & **getTranslation** () const
- void **setLine** (const osg::LineSegment::vec_type &s, const osg::LineSegment::vec_type &e)
- void **setTranslation** (const osg::Vec3 &t)

Protected Member Functions

- virtual ∼**TranslateInLineCommand** ()

4.37.1 Detailed Description

Command for translating in a line.

4.37.2 Constructor & Destructor Documentation

4.37.2.1 TranslateInLineCommand ()

4.37.2.2 TranslateInLineCommand (const osg::LineSegment::vec_type & *s*, const osg::LineSegment::vec_type & *e*)

4.37.2.3 ∼TranslateInLineCommand () [protected, virtual]

4.37.3 Member Function Documentation

4.37.3.1 MotionCommand ∗ createCommandInverse () [virtual]

create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.

Implements **MotionCommand** (p. 42).

4.37.3.2 const osg::LineSegment::vec_type& getLineEnd () const [inline]

4.37.3.3 const osg::LineSegment::vec_type& getLineStart () const [inline]

4.37.3.4 virtual osg::Matrix getMotionMatrix () const [inline, virtual]

Gets the matrix for transforming the object being dragged. This matrix is in the command's coordinate systems.

Implements **MotionCommand** (p. 42).

4.37.3.5 const osg::Vec3d& getTranslation () const [inline]

4.37.3.6 void setLine (const osg::LineSegment::vec_type & *s*, const osg::LineSegment::vec_type & *e*) [inline]

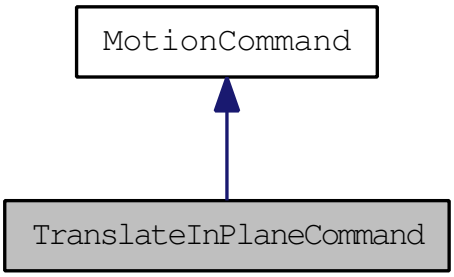
4.37.3.7 void setTranslation (const osg::Vec3 & *t*) [inline]

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

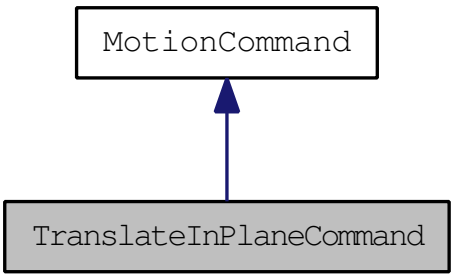
- **Command**
- **Command.cpp**

4.38 TranslateInPlaneCommand Class Reference

Command for translating in a plane. Inheritance diagram for TranslateInPlaneCommand:



Collaboration diagram for TranslateInPlaneCommand:



Public Member Functions

- **TranslateInPlaneCommand** (const osg::Plane &plane)
- **TranslateInPlaneCommand** ()
- virtual **MotionCommand** * **createCommandInverse** ()

  *create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.*

- virtual osg::Matrix **getMotionMatrix** () const

  Gets the matrix for transforming the object being dragged.

- const osg::Plane & **getPlane** () const
- const osg::Vec3d & **getReferencePoint** () const
- const osg::Vec3d & **getTranslation** () const
- void **setPlane** (const osg::Plane &plane)
- void **setReferencePoint** (const osg::Vec3d &rp)

  ReferencePoint is used only for snapping.

- void **setTranslation** (const osg::Vec3d &t)

Protected Member Functions

- virtual ~**TranslateInPlaneCommand** ()

4.38.1 Detailed Description

Command for translating in a plane.

4.38.2 Constructor & Destructor Documentation

4.38.2.1 TranslateInPlaneCommand ()

4.38.2.2 TranslateInPlaneCommand (const osg::Plane & *plane*)

4.38.2.3 ~TranslateInPlaneCommand () [protected, virtual]

4.38.3 Member Function Documentation

4.38.3.1 MotionCommand * createCommandInverse () [virtual]

create a **MotionCommand** (p. 41) that is the inverse of this command, and if applied will undo this commands changes.

Implements **MotionCommand** (p. 42).

4.38.3.2 virtual osg::Matrix getMotionMatrix () const [inline, virtual]

Gets the matrix for transforming the object being dragged. This matrix is in the command's coordinate systems.

Implements **MotionCommand** (p. 42).

4.38.3.3 const osg::Plane& getPlane () const [inline]

4.38.3.4 const osg::Vec3d& getReferencePoint () const [inline]

4.38.3.5 const osg::Vec3d& getTranslation () const [inline]

4.38.3.6 void setPlane (const osg::Plane & *plane*) [inline]

4.38.3.7 void setReferencePoint (const osg::Vec3d & *rp*) [inline]

ReferencePoint is used only for snapping.

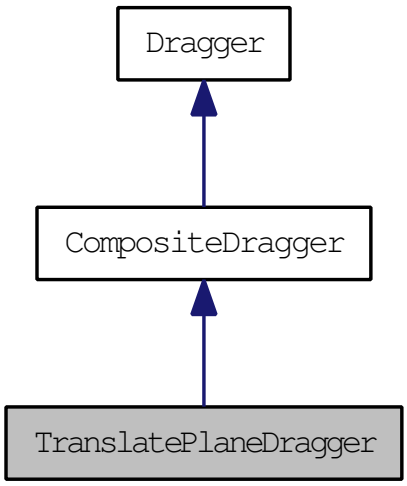
4.38.3.8 void setTranslation (const osg::Vec3d & *t*) [inline]

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

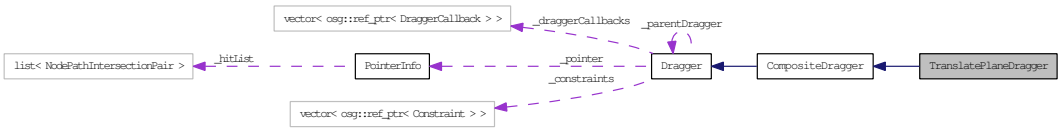
- **Command**
- **Command.cpp**

4.39 TranslatePlaneDragger Class Reference

Tab plane dragger consists of a plane with tabs on it's corners and edges for scaling. Inheritance diagram for TranslatePlaneDragger:



Collaboration diagram for TranslatePlaneDragger:



Public Member Functions

- **TranslatePlaneDragger** ()
- **Translate1DDragger** ∗ **getTranslate1DDragger** ()
- **Translate2DDragger** ∗ **getTranslate2DDragger** ()
- **META_OSGMANIPULATOR_Object** (osgManipulator, **TranslatePlaneDragger**) virtual bool handle(const **PointerInfo** &pi
- void **setColor** (const osg::Vec4 &color)
- void **setupDefaultGeometry** ()

  Setup default geometry for dragger.

Public Attributes

- const osgGA::GUIEventAdapter & **ea**
- const osgGA::GUIEventAdapter osgGA::GUIActionAdapter & **us**

Protected Member Functions

- virtual ∼**TranslatePlaneDragger** ()

Protected Attributes

- osg::ref_ptr< **Translate1DDragger** > **_translate1DDragger**
- osg::ref_ptr< **Translate2DDragger** > **_translate2DDragger**
- bool **_usingTranslate1DDragger**

4.39.1 Detailed Description

Tab plane dragger consists of a plane with tabs on it's corners and edges for scaling. And the plane is used as a 2D translate dragger.

4.39.2 Constructor & Destructor Documentation

4.39.2.1 TranslatePlaneDragger ()

4.39.2.2 ∼TranslatePlaneDragger () [protected, virtual]

4.39.3 Member Function Documentation

4.39.3.1 Translate1DDragger∗ getTranslate1DDragger () [inline]

4.39.3.2 Translate2DDragger∗ getTranslate2DDragger () [inline]

4.39.3.3 META_OSGMANIPULATOR_Object (osgManipulator, TranslatePlaneDragger) const

4.39.3.4 void setColor (const osg::Vec4 & *color*) [inline]

4.39.3.5 void setupDefaultGeometry () [virtual]

Setup default geometry for dragger.

Reimplemented from **Dragger** (p. 32).

4.39.4 Member Data Documentation

4.39.4.1 osg::ref_ptr< Translate1DDragger > _translate1DDragger [protected]

4.39.4.2 osg::ref_ptr< Translate2DDragger > _translate2DDragger [protected]

4.39.4.3 bool _usingTranslate1DDragger [protected]

4.39.4.4 const osgGA::GUIEventAdapter& ea

4.39.4.5 const osgGA::GUIEventAdapter osgGA::GUIActionAdapter& us

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

- **TranslatePlaneDragger**
- **TranslatePlaneDragger.cpp**

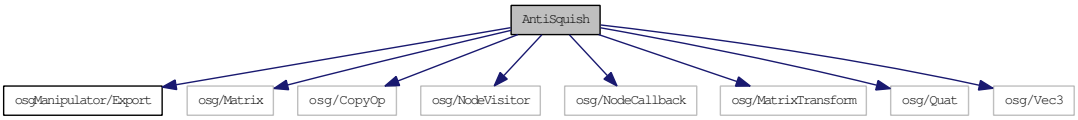
Chapter 5

File Documentation

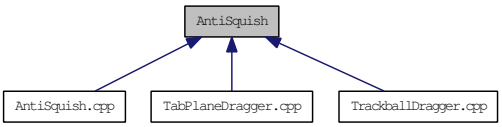
5.1 AntiSquish File Reference

```
#include <osgManipulator/Export>
#include <osg/Matrix>
#include <osg/CopyOp>
#include <osg/NodeVisitor>
#include <osg/NodeCallback>
#include <osg/MatrixTransform>
#include <osg/Quat>
#include <osg/Vec3>
```

Include dependency graph for AntiSquish:



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



Classes

- class **AntiSquish**

  Class that performs the Anti Squish by making the scaling uniform along all axes.

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **_OSG_ANTISQUISH_** 1

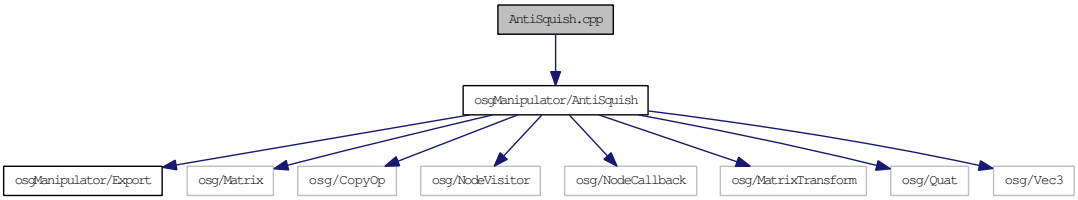
5.1.1 Define Documentation

5.1.1.1 #define _OSG_ANTISQUISH_ 1

5.2 AntiSquish.cpp File Reference

`#include <osgManipulator/AntiSquish>`

Include dependency graph for AntiSquish.cpp:



Classes

- class **AntiSquishCallback**

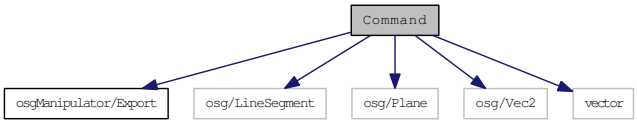
Namespaces

- namespace **anonymous_namespace{AntiSquish.cpp}**

5.3 Command File Reference

```
#include <osgManipulator/Export>
#include <osg/LineSegment>
#include <osg/Plane>
#include <osg/Vec2>
#include <vector>
```

Include dependency graph for Command:



Classes

- class **MotionCommand**
  Base class for motion commands that are generated by draggers.
- class **Rotate3DCommand**
  Command for rotation in 3D.
- class **Scale1DCommand**
  Command for 1D scaling.
- class **Scale2DCommand**
  Command for 2D scaling.
- class **ScaleUniformCommand**
  Command for uniform 3D scaling.
- class **TranslateInLineCommand**
  Command for translating in a line.
- class **TranslateInPlaneCommand**
  Command for translating in a plane.

Namespaces

- namespace **osgManipulator**
  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_COMMAND** 1

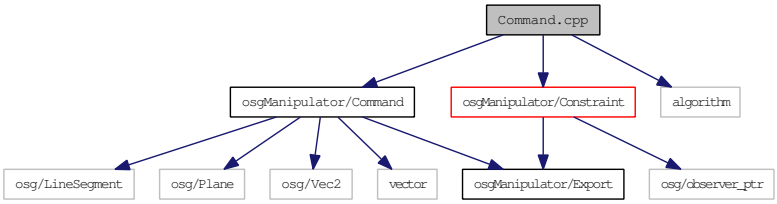
5.3.1 Define Documentation

5.3.1.1 #define OSGMANIPULATOR_COMMAND 1

5.4 Command.cpp File Reference

```
#include <osgManipulator/Command>
#include <osgManipulator/Constraint>
#include <algorithm>
```

Include dependency graph for Command.cpp:



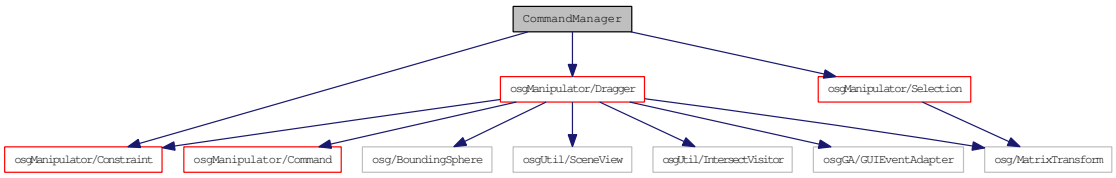
5.5 CommandManager File Reference

`#include <osgManipulator/Dragger>`

`#include <osgManipulator/Selection>`

`#include <osgManipulator/Constraint>`

Include dependency graph for CommandManager:



Classes

- class **CommandManager**

  Deprecated.

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_COMMANDMANAGER** 1

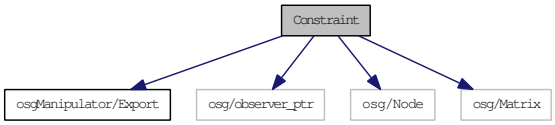
5.5.1 Define Documentation

5.5.1.1 #define OSGMANIPULATOR_COMMANDMANAGER 1

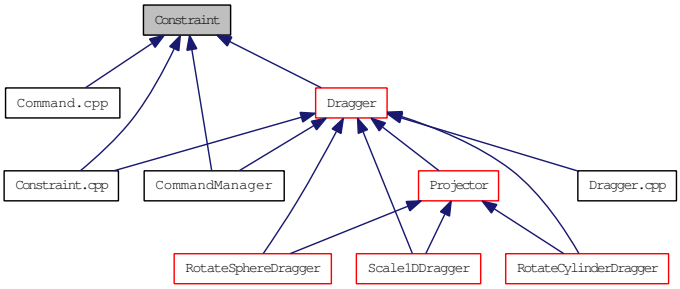
5.6 Constraint File Reference

```
#include <osgManipulator/Export>
#include <osg/observer_ptr>
#include <osg/Node>
#include <osg/Matrix>
```

Include dependency graph for Constraint:



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



Classes

- class **Constraint**
- class **GridConstraint**

  Constraint *(*p. *23) to snap motion commands to a sugar cube grid.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (*p. *14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_CONSTRAINT** 1

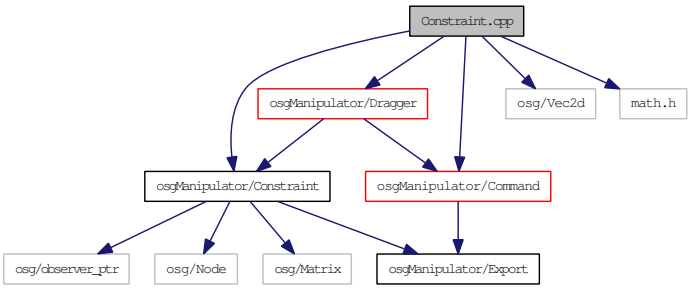
5.6.1 Define Documentation

5.6.1.1 #define OSGMANIPULATOR_CONSTRAINT 1

5.7 Constraint.cpp File Reference

```
#include <osgManipulator/Constraint>
#include <osgManipulator/Command>
#include <osgManipulator/Dragger>
#include <osg/Vec2d>
#include <math.h>
```

Include dependency graph for Constraint.cpp:



Namespaces

- namespace **anonymous_namespace{Constraint.cpp}**

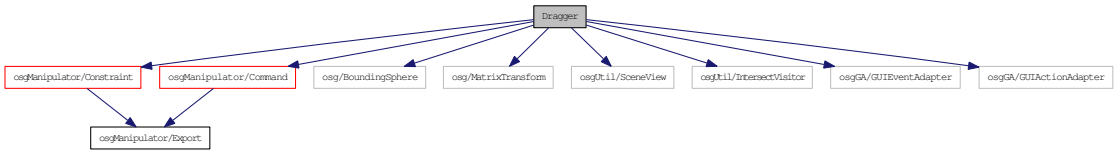
Functions

- double **round_to_nearest_int** (double x)
- osg::Vec3d **snap_point_to_grid** (const osg::Vec3d &point, const osg::Vec3d &origin, const osg::Vec3d &spacing)

5.8 Dragger File Reference

```
#include <osgManipulator/Constraint>
#include <osgManipulator/Command>
#include <osg/BoundingSphere>
#include <osg/MatrixTransform>
#include <osgUtil/SceneView>
#include <osgUtil/IntersectVisitor>
#include <osgGA/GUIEventAdapter>
#include <osgGA/GUIActionAdapter>
```

Include dependency graph for Dragger:



Classes

- class **CompositeDragger**
  CompositeDragger (p. *21) allows to create complex draggers that are composed of a hierarchy of Draggers.*
- class **Dragger**
  Base class for draggers.
- class **DraggerCallback**
- class **DraggerTransformCallback**
- class **PointerInfo**

Namespaces

- namespace **osgManipulator**
  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_DRAGGER** 1

Functions

- OSGMANIPULATOR_EXPORT void **computeNodePathToRoot** (osg::Node &node, osg::NodePath &np)
  Computes the nodepath from the given node all the way upto the root.
- void OSGMANIPULATOR_EXPORT **setDrawableToAlwaysCull** (osg::Drawable &drawable)
  Culls the drawable all the time.
- void OSGMANIPULATOR_EXPORT **setMaterialColor** (const osg::Vec4 &color, osg::Node &node)
  Convenience function for setting the material color on a node.

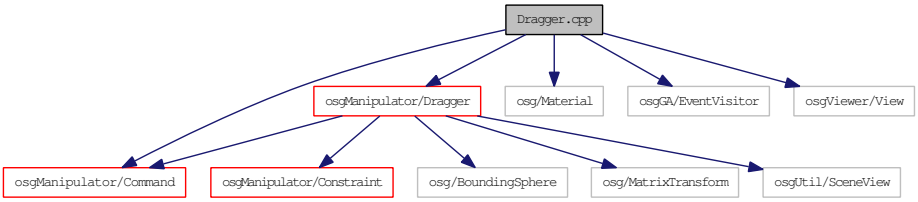
5.8.1 Define Documentation

5.8.1.1 #define OSGMANIPULATOR_DRAGGER 1

5.9 Dragger.cpp File Reference

```
#include <osgManipulator/Dragger>
#include <osgManipulator/Command>
#include <osg/Material>
#include <osgGA/EventVisitor>
#include <osgViewer/View>
```

Include dependency graph for Dragger.cpp:

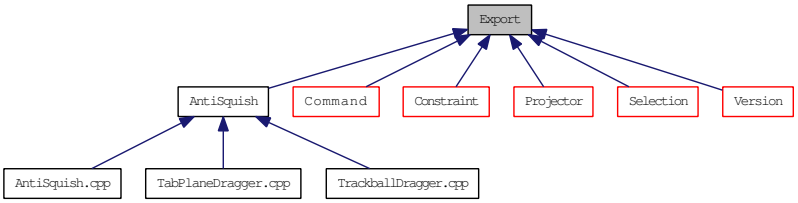


Classes

- class **ForceCullCallback**

5.10 Export File Reference

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



Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **META_OSGMANIPULATOR_Object**(library, name)
- #define **OSGMANIPULATOR_EXPORT**
- #define **OSGMANIPULATOR_EXPORT_** 1

5.10.1 Define Documentation

5.10.1.1 #define META_OSGMANIPULATOR_Object(library, name)

Value:

```
virtual bool isSameKindAs(const osg::Object* obj) const { return dynamic_cast<con
      st name *>(obj)!=NULL; } \
virtual const char* libraryName() const { return #library; }\
virtual const char* className() const { return #name; }
```

5.10.1.2 #define OSGMANIPULATOR_EXPORT

5.10.1.3 #define OSGMANIPULATOR_EXPORT_ 1

5.11 mainpage.h File Reference

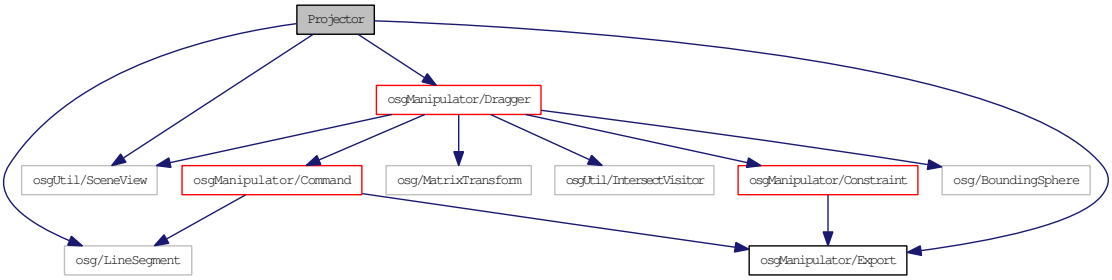
5.11.1 Detailed Description

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.

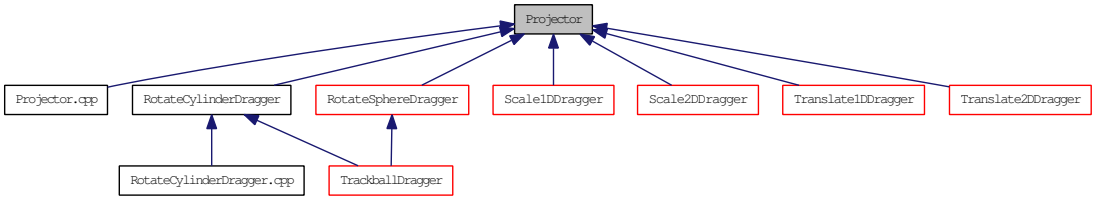
5.12 Projector File Reference

```
#include <osgManipulator/Export>
#include <osg/LineSegment>
#include <osgUtil/SceneView>
#include <osgManipulator/Dragger>
```

Include dependency graph for Projector:



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



Classes

- class **CylinderPlaneProjector**

  CylinderPlaneProjector *(p. 25) projects points onto the given cylinder.*

- class **CylinderProjector**

  CylinderProjector *(p. 27) projects points onto the given cylinder.*

- class **LineProjector**

  LineProjector *(p. 39) projects points onto the closest point on the given line.*

- class **PlaneProjector**

  PlaneProjector *(p. 43) projects points onto the given line.*

- class **Projector**

  Base class for Projectors.

- class **SpherePlaneProjector**

  SpherePlaneProjector *(p. 70) projects points onto a sphere, failing which it project onto a plane oriented to the viewing direction.*

- class **SphereProjector**

  SphereProjector *(p. 72) projects points onto the given sphere.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_PROJECTOR** 1

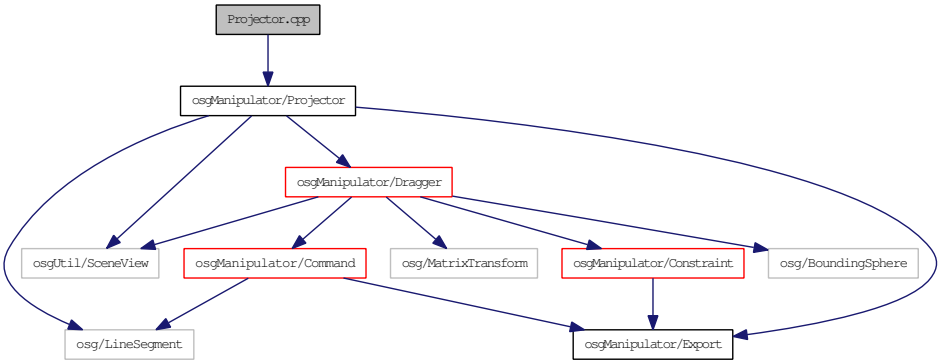
5.12.1 Define Documentation

5.12.1.1 #define OSGMANIPULATOR_PROJECTOR 1

5.13 Projector.cpp File Reference

`#include <osgManipulator/Projector>`

Include dependency graph for Projector.cpp:



Namespaces

- namespace **anonymous_namespace{Projector.cpp}**

Functions

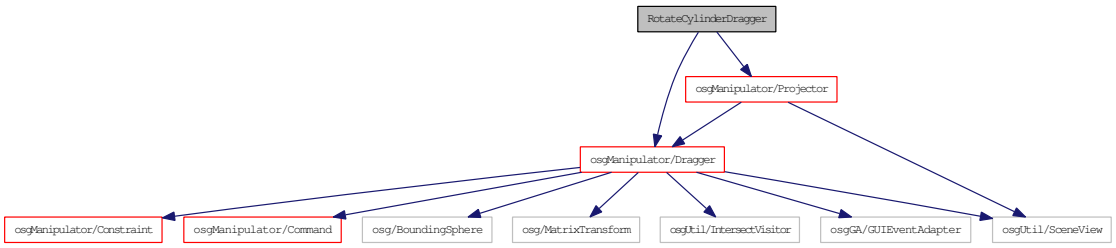
- bool **computeClosestPointOnLine** (const osg::Vec3d &lineStart, const osg::Vec3d &lineEnd, const osg::Vec3d &fromPoint, osg::Vec3d &closestPoint)
- bool **computeClosestPoints** (const osg::LineSegment &l1, const osg::LineSegment &l2, osg::Vec3d &p1, osg::Vec3d &p2)
- osg::Plane **computePlaneParallelToAxisAndOrientedToEye** (const osg::Vec3d &eyeDir, const osg::Matrix &localToWorld, const osg::Vec3d &axisDir, double radius, osg::Vec3d &planeLineStart, osg::Vec3d &planeLineEnd, bool front)
- osg::Plane **computePlaneThruPointAndOrientedToEye** (const osg::Vec3d &eyeDir, const osg::Matrix &localToWorld, const osg::Vec3d &point, bool front)
- bool **getCylinderLineIntersection** (const osg::Cylinder &cylinder, const osg::Vec3d &lineStart, const osg::Vec3d &lineEnd, osg::Vec3d &isectFront, osg::Vec3d &isectBack)
- osg::Vec3d **getLocalEyeDirection** (const osg::Vec3d &eyeDir, const osg::Matrix &localToWorld)
- bool **getPlaneLineIntersection** (const osg::Vec4d &plane, const osg::Vec3d &lineStart, const osg::Vec3d &lineEnd, osg::Vec3d &isect)
- bool **getSphereLineIntersection** (const osg::Sphere &sphere, const osg::Vec3d &lineStart, const osg::Vec3d &lineEnd, osg::Vec3d &frontISect, osg::Vec3d &backISect)
- bool **getUnitCylinderLineIntersection** (const osg::Vec3d &lineStart, const osg::Vec3d &lineEnd, osg::Vec3d &isectFront, osg::Vec3d &isectBack)

5.14 RotateCylinderDragger File Reference

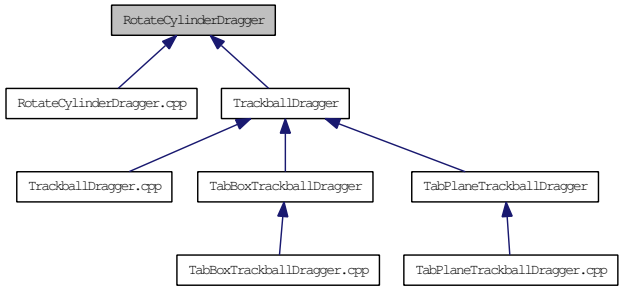
`#include <osgManipulator/Dragger>`

`#include <osgManipulator/Projector>`

Include dependency graph for RotateCylinderDragger:



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



Classes

- class **RotateCylinderDragger**

  Dragger (p. *29*) *for performing 3D rotation on a cylinder.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_ROTATECYLINDERDRAGGER** 1

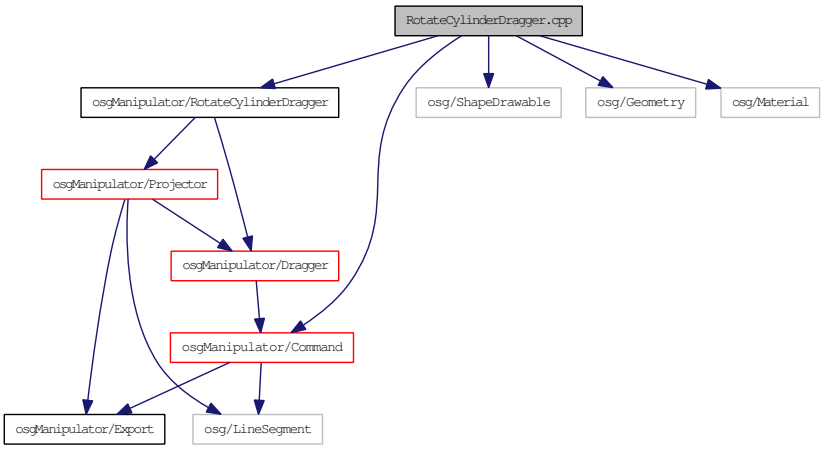
5.14.1 Define Documentation

5.14.1.1 #define OSGMANIPULATOR_ROTATECYLINDERDRAGGER 1

5.15 RotateCylinderDragger.cpp File Reference

```
#include <osgManipulator/RotateCylinderDragger>
#include <osgManipulator/Command>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/Material>
```

Include dependency graph for RotateCylinderDragger.cpp:

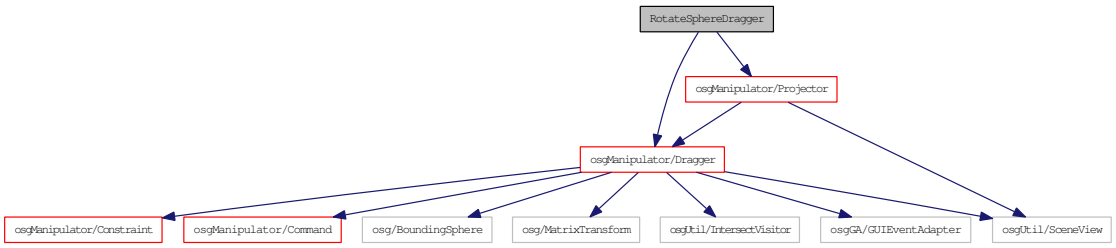


5.16 RotateSphereDragger File Reference

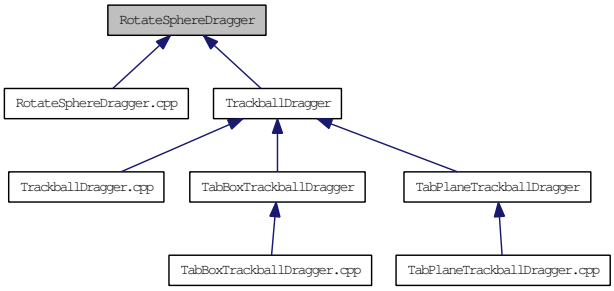
`#include <osgManipulator/Dragger>`

`#include <osgManipulator/Projector>`

Include dependency graph for RotateSphereDragger:



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



Classes

- class **RotateSphereDragger**

  Dragger *(p. 29) for performing 3D rotation on a sphere.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_ROTATESPHEREDRAGGER** 1

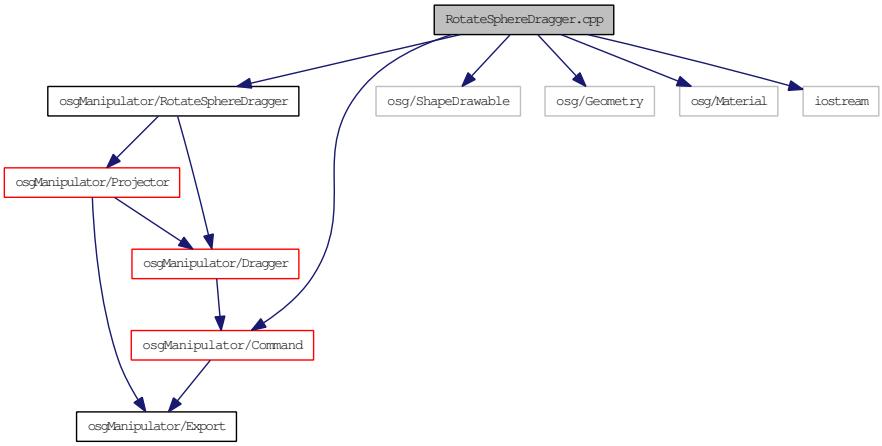
5.16.1 Define Documentation

5.16.1.1 #define OSGMANIPULATOR_ROTATESPHEREDRAGGER 1

5.17 RotateSphereDragger.cpp File Reference

```
#include <osgManipulator/RotateSphereDragger>
#include <osgManipulator/Command>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/Material>
#include <iostream>
```

Include dependency graph for RotateSphereDragger.cpp:

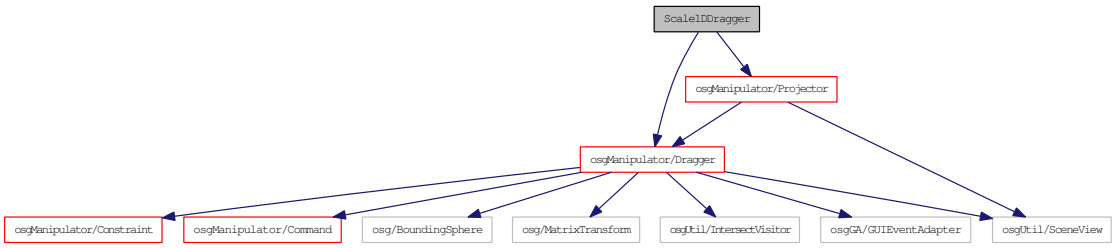


5.18 Scale1DDragger File Reference

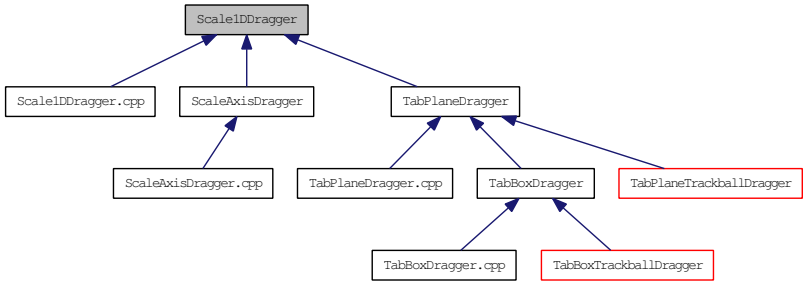
`#include <osgManipulator/Dragger>`

`#include <osgManipulator/Projector>`

Include dependency graph for Scale1DDragger:



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



Classes

- class **Scale1DDragger**

  Dragger *(p. 29) for performing 1D scaling.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_SCALE1DDRAGGER** 1

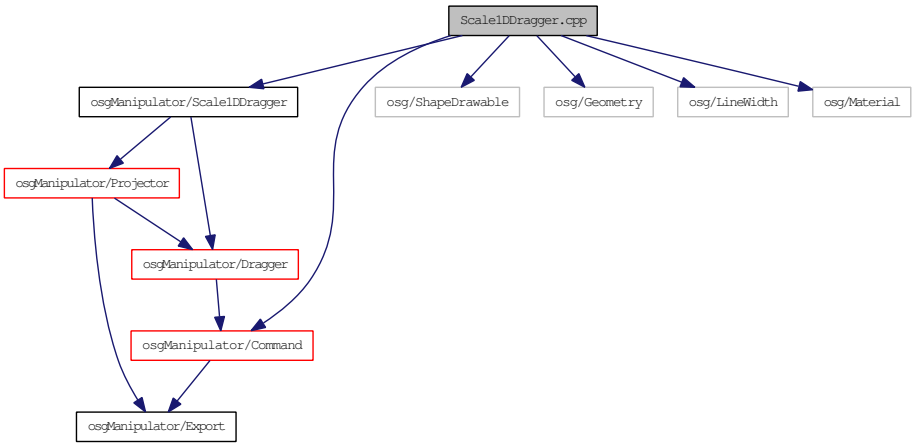
5.18.1 Define Documentation

5.18.1.1 #define OSGMANIPULATOR_SCALE1DDRAGGER 1

5.19 Scale1DDragger.cpp File Reference

```
#include <osgManipulator/Scale1DDragger>
#include <osgManipulator/Command>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/Material>
```

Include dependency graph for Scale1DDragger.cpp:



Namespaces

- namespace **anonymous_namespace{Scale1DDragger.cpp}**

Functions

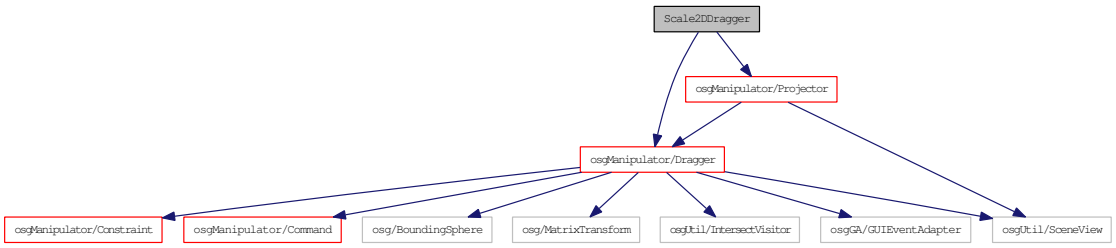
- double **computeScale** (const osg::Vec3d &startProjectedPoint, const osg::Vec3d &projectedPoint, double scaleCenter)

5.20 Scale2DDragger File Reference

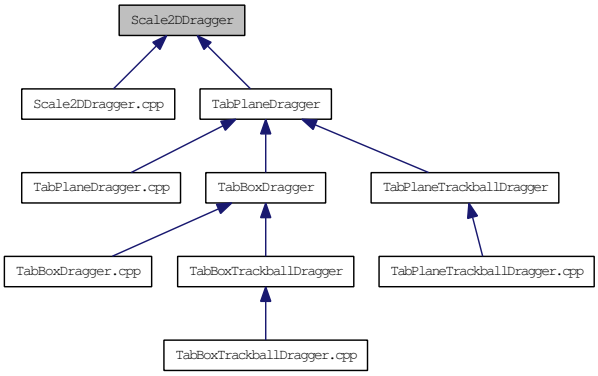
`#include <osgManipulator/Dragger>`

`#include <osgManipulator/Projector>`

Include dependency graph for Scale2DDragger:



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



Classes

- class **Scale2DDragger**

  Dragger (p. *29*) *for performing 2D scaling.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator*** (p. *14*) *library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_SCALE2DDRAGGER** 1

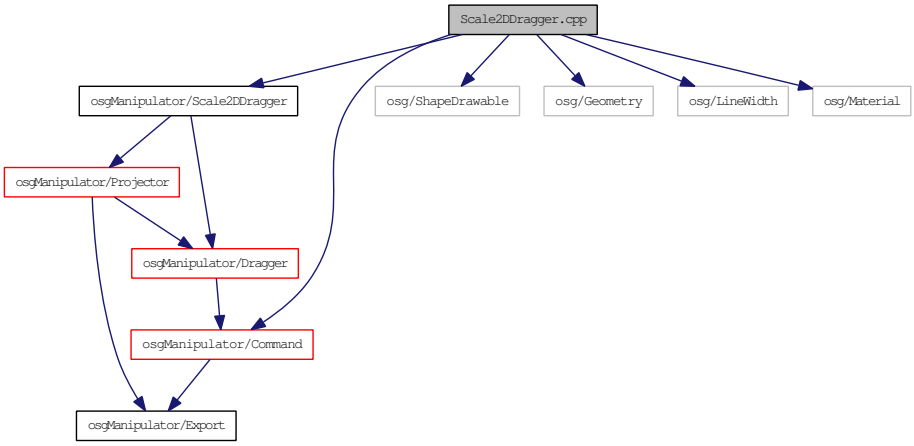
5.20.1 Define Documentation

5.20.1.1 #define OSGMANIPULATOR_SCALE2DDRAGGER 1

5.21 Scale2DDragger.cpp File Reference

```
#include <osgManipulator/Scale2DDragger>
#include <osgManipulator/Command>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/Material>
```

Include dependency graph for Scale2DDragger.cpp:



Namespaces

- namespace **anonymous_namespace{Scale2DDragger.cpp}**

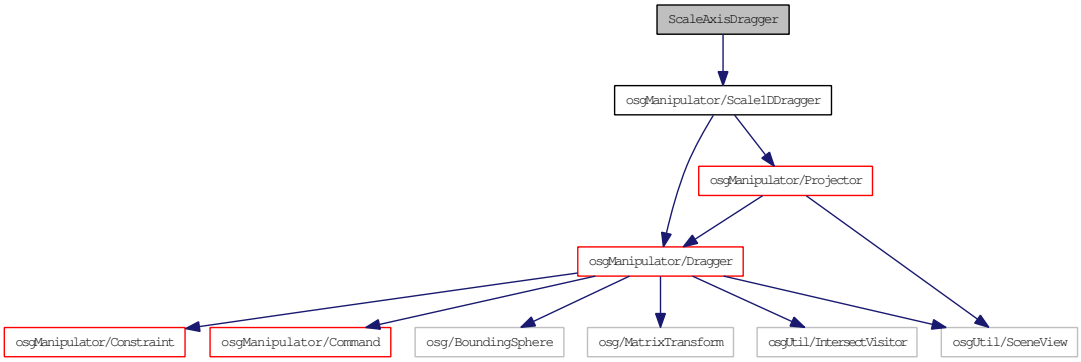
Functions

- osg::Vec2d **computeScale** (const osg::Vec3d &startProjectedPoint, const osg::Vec3d &projectedPoint, const osg::Vec2d &scaleCenter)

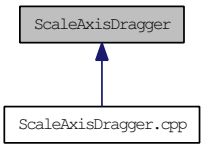
5.22 ScaleAxisDragger File Reference

`#include <osgManipulator/Scale1DDragger>`

Include dependency graph for ScaleAxisDragger:



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



Classes

- class **ScaleAxisDragger**

  Dragger *(p. 29) for performing scaling on all 3 axes.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_SCALEAXISDRAGGER** 1

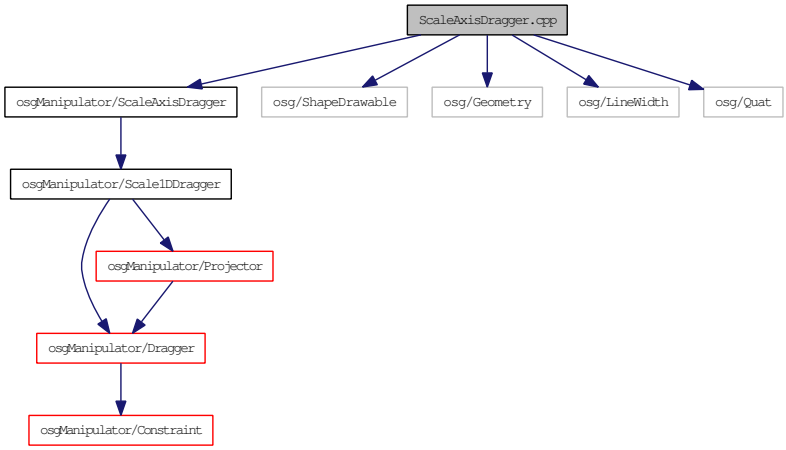
5.22.1 Define Documentation

5.22.1.1 #define OSGMANIPULATOR_SCALEAXISDRAGGER 1

5.23 ScaleAxisDragger.cpp File Reference

```
#include <osgManipulator/ScaleAxisDragger>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/Quat>
```

Include dependency graph for ScaleAxisDragger.cpp:

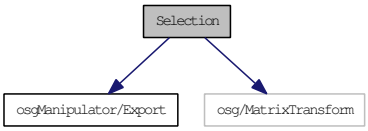


5.24 Selection File Reference

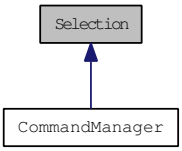
`#include <osgManipulator/Export>`

`#include <osg/MatrixTransform>`

Include dependency graph for Selection:



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



Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_SELECTION** 1

Typedefs

- typedef osg::MatrixTransform **Selection**

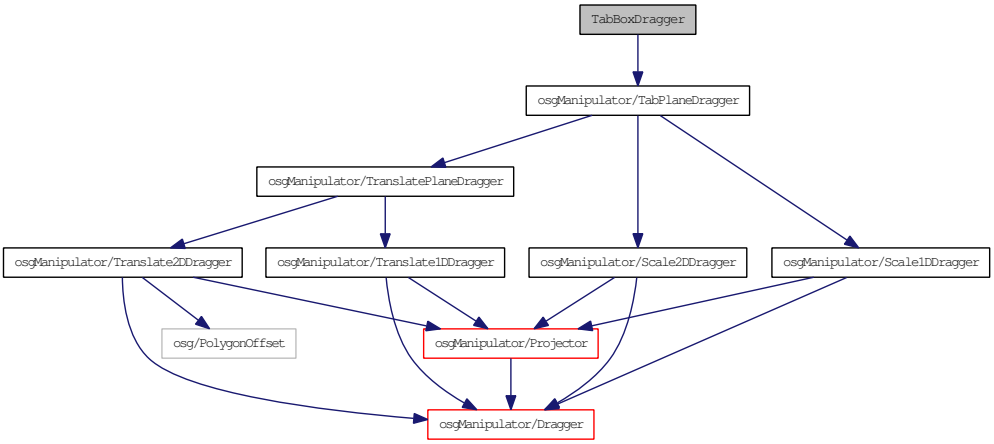
5.24.1 Define Documentation

5.24.1.1 #define OSGMANIPULATOR_SELECTION 1

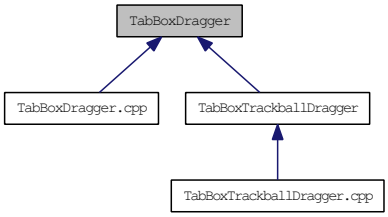
5.25 TabBoxDragger File Reference

`#include <osgManipulator/TabPlaneDragger>`

Include dependency graph for TabBoxDragger:



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



Classes

- class **TabBoxDragger**

  TabBoxDragger *(p. 74) consists of 6 TabPlaneDraggers to form a box dragger that performs translation and scaling.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_TABBOXDRAGGER** 1

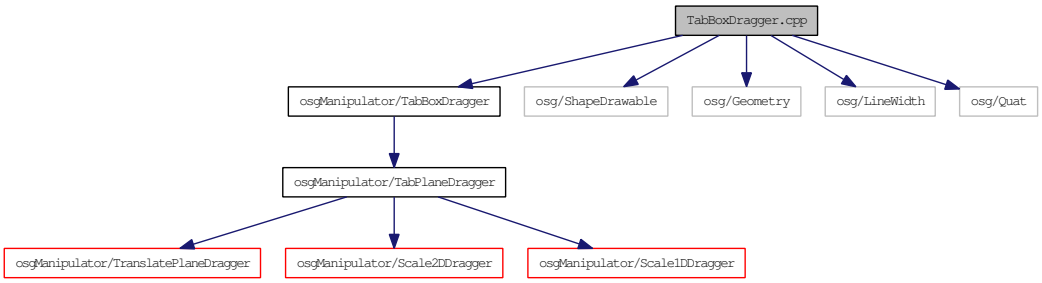
5.25.1 Define Documentation

5.25.1.1 #define OSGMANIPULATOR_TABBOXDRAGGER 1

5.26 TabBoxDragger.cpp File Reference

```
#include <osgManipulator/TabBoxDragger>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/Quat>
```

Include dependency graph for TabBoxDragger.cpp:

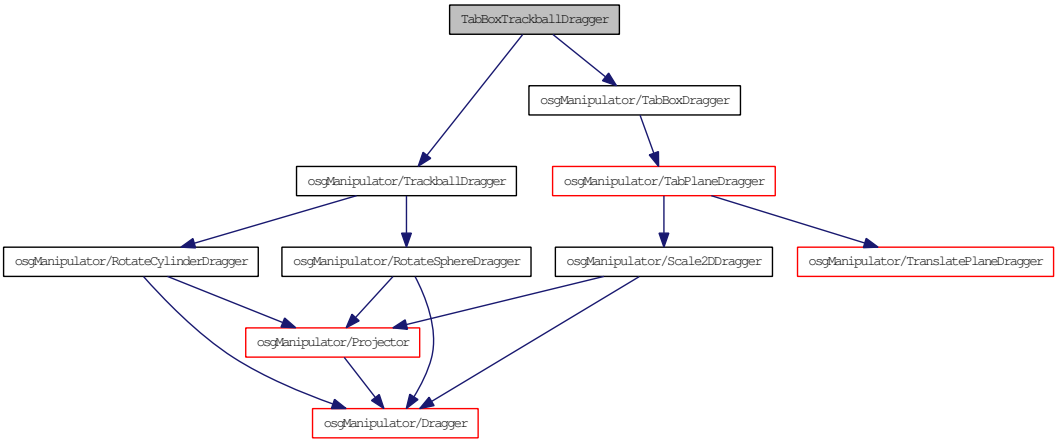


5.27 TabBoxTrackballDragger File Reference

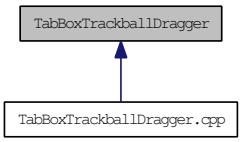
`#include <osgManipulator/TrackballDragger>`

`#include <osgManipulator/TabBoxDragger>`

Include dependency graph for TabBoxTrackballDragger:



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



Classes

- class **TabBoxTrackballDragger**

  Dragger *(*p. *29) for performing rotation in all axes.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (*p. *14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_TABBOXTRACKBALLDRAGGER** 1

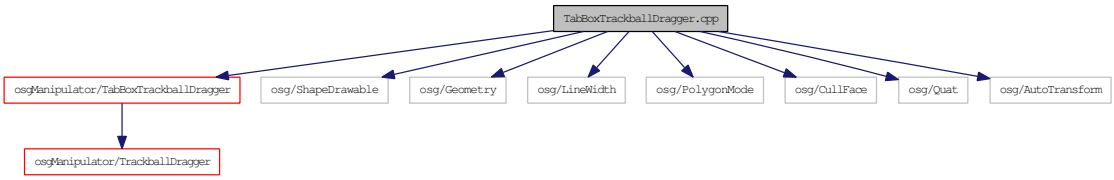
5.27.1 Define Documentation

5.27.1.1 #define OSGMANIPULATOR_TABBOXTRACKBALLDRAGGER 1

5.28 TabBoxTrackballDragger.cpp File Reference

```
#include <osgManipulator/TabBoxTrackballDragger>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/PolygonMode>
#include <osg/CullFace>
#include <osg/Quat>
#include <osg/AutoTransform>
```

Include dependency graph for TabBoxTrackballDragger.cpp:



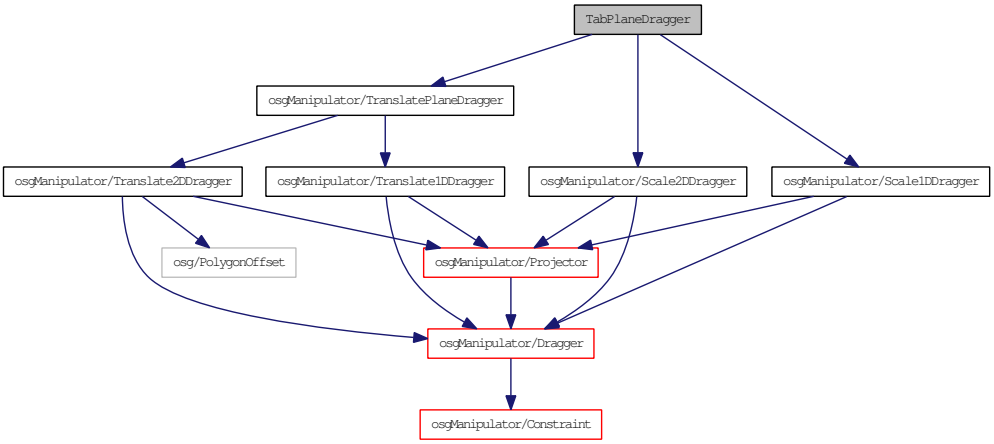
5.29 TabPlaneDragger File Reference

`#include <osgManipulator/TranslatePlaneDragger>`

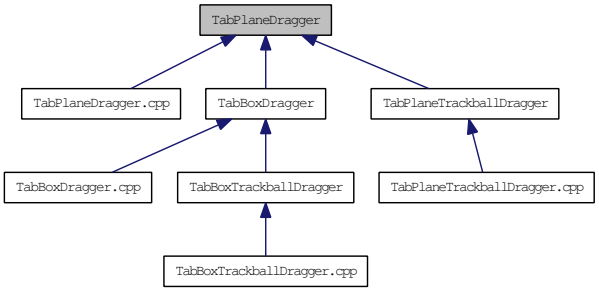
`#include <osgManipulator/Scale2DDragger>`

`#include <osgManipulator/Scale1DDragger>`

Include dependency graph for TabPlaneDragger:



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



Classes

- class **TabPlaneDragger**

  Tab plane dragger consists of a plane with tabs on it's corners and edges for scaling.

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_TABPLANEDRAGGER** 1

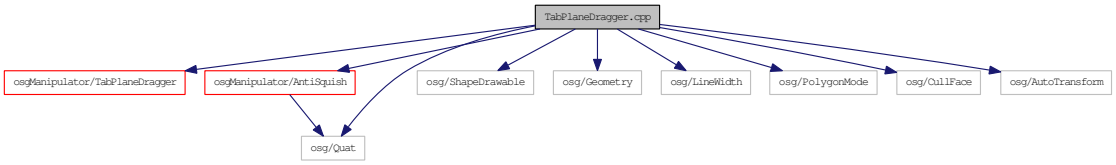
5.29.1 Define Documentation

5.29.1.1 #define OSGMANIPULATOR_TABPLANEDRAGGER 1

5.30 TabPlaneDragger.cpp File Reference

```
#include <osgManipulator/TabPlaneDragger>
#include <osgManipulator/AntiSquish>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/Quat>
#include <osg/PolygonMode>
#include <osg/CullFace>
#include <osg/AutoTransform>
```

Include dependency graph for TabPlaneDragger.cpp:



Namespaces

- namespace **anonymous_namespace{TabPlaneDragger.cpp}**

Functions

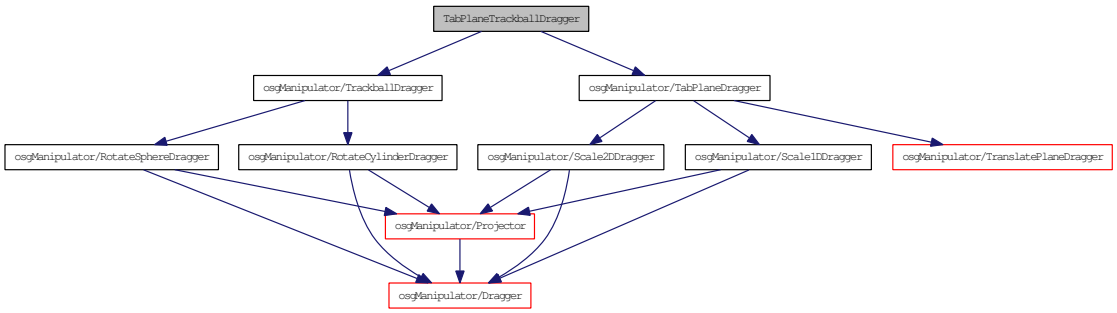
- void **createCornerScaleDraggerGeometry** (**Scale2DDragger** *cornerScaleDragger, osg::Node *handleNode, float handleScaleFactor)
- void **createEdgeScaleDraggerGeometry** (**Scale1DDragger** *horzEdgeScaleDragger, **Scale1DDragger** *vertEdgeScaleDragger, osg::Node *handleNode, float handleScaleFactor)
- osg::Node * **createHandleNode** (**Scale2DDragger** *cornerScaleDragger, float handleScaleFactor, bool twosided)
- osg::Node * **createHandleScene** (const osg::Vec3 &pos, osg::Node *handleNode, float handleScaleFactor)
- void **createTranslateDraggerGeometry** (**Scale2DDragger** *cornerScaleDragger, **TranslatePlaneDragger** *translateDragger)

5.31 TabPlaneTrackballDragger File Reference

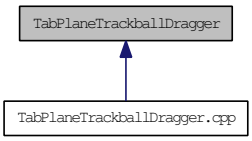
`#include <osgManipulator/TrackballDragger>`

`#include <osgManipulator/TabPlaneDragger>`

Include dependency graph for TabPlaneTrackballDragger:



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



Classes

- class **TabPlaneTrackballDragger**

  Dragger *(p. 29) for performing rotation in all axes.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_TABPLANETRACKBALLDRAGGER** 1

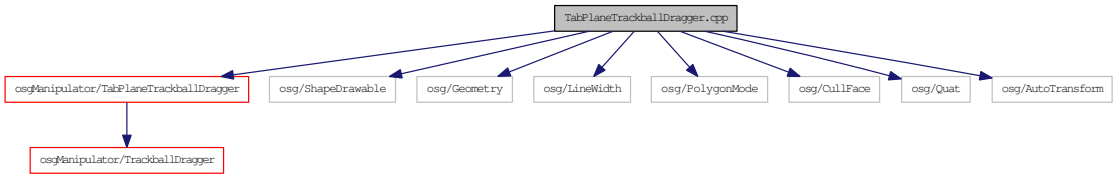
5.31.1 Define Documentation

5.31.1.1 #define OSGMANIPULATOR_TABPLANETRACKBALLDRAGGER 1

5.32 TabPlaneTrackballDragger.cpp File Reference

```
#include <osgManipulator/TabPlaneTrackballDragger>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/PolygonMode>
#include <osg/CullFace>
#include <osg/Quat>
#include <osg/AutoTransform>
```

Include dependency graph for TabPlaneTrackballDragger.cpp:

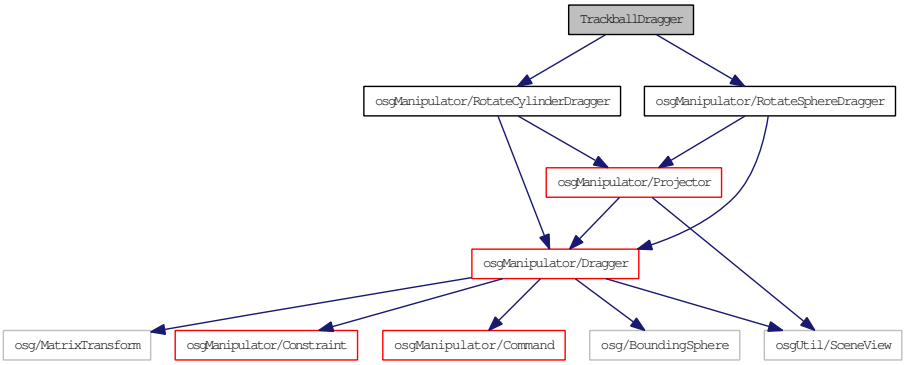


5.33 TrackballDragger File Reference

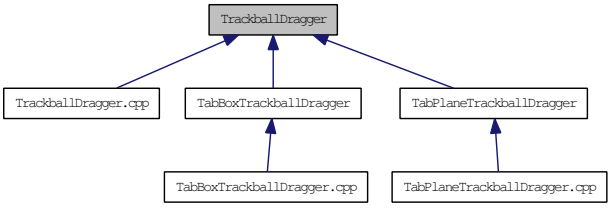
`#include <osgManipulator/RotateCylinderDragger>`

`#include <osgManipulator/RotateSphereDragger>`

Include dependency graph for TrackballDragger:



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



Classes

- class **TrackballDragger**

  Dragger *(p. 29) for performing rotation in all axes.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_TRACKBALLDRAGGER** 1

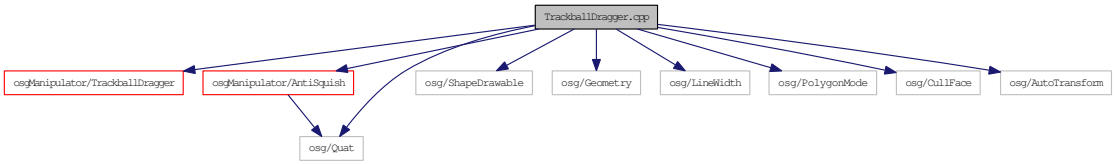
5.33.1 Define Documentation

5.33.1.1 #define OSGMANIPULATOR_TRACKBALLDRAGGER 1

5.34 TrackballDragger.cpp File Reference

```
#include <osgManipulator/TrackballDragger>
#include <osgManipulator/AntiSquish>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/PolygonMode>
#include <osg/CullFace>
#include <osg/Quat>
#include <osg/AutoTransform>
```

Include dependency graph for TrackballDragger.cpp:



Namespaces

- namespace **anonymous_namespace{TrackballDragger.cpp}**

Functions

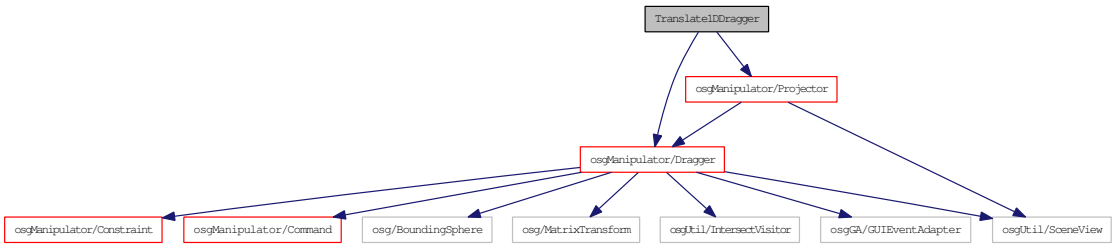
- osg::Geometry * **createCircleGeometry** (float radius, unsigned int numSegments)

5.35 Translate1DDragger File Reference

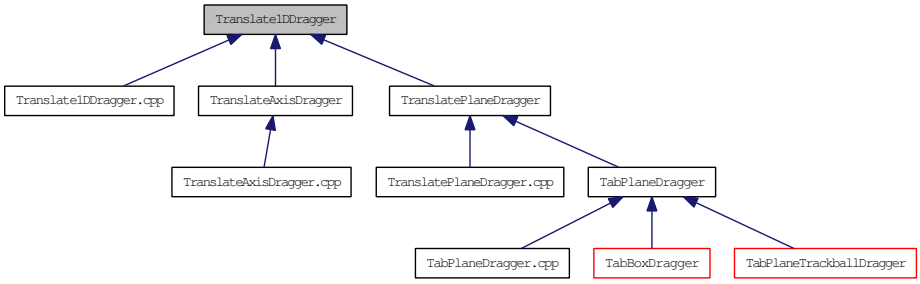
`#include <osgManipulator/Dragger>`

`#include <osgManipulator/Projector>`

Include dependency graph for Translate1DDragger:



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



Classes

- class **Translate1DDragger**

  Dragger *(p. 29) for performing 1D translation.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_TRANSLATE1DDRAGGER** 1

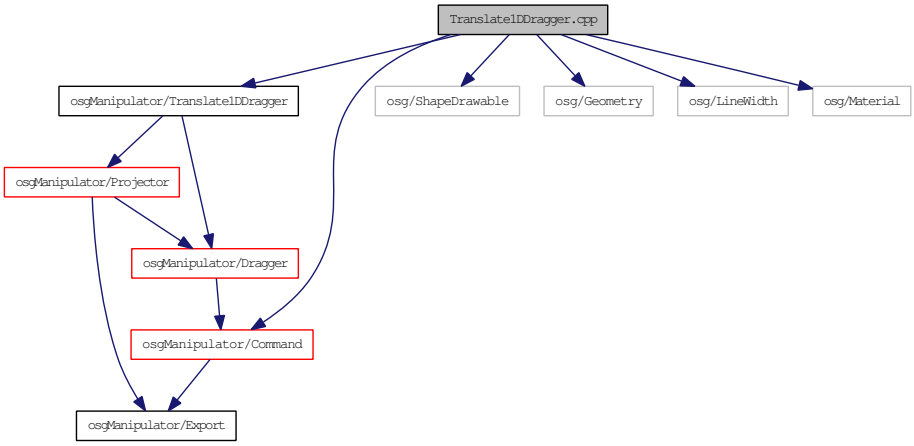
5.35.1 Define Documentation

5.35.1.1 #define OSGMANIPULATOR_TRANSLATE1DDRAGGER 1

5.36 Translate1DDragger.cpp File Reference

```
#include <osgManipulator/Translate1DDragger>
#include <osgManipulator/Command>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/Material>
```

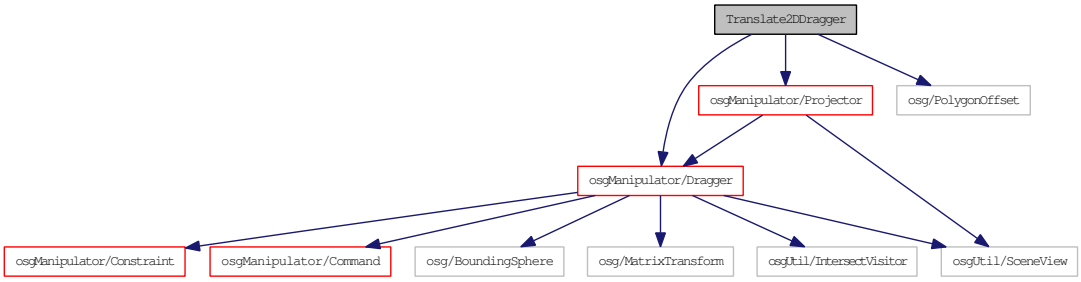
Include dependency graph for Translate1DDragger.cpp:



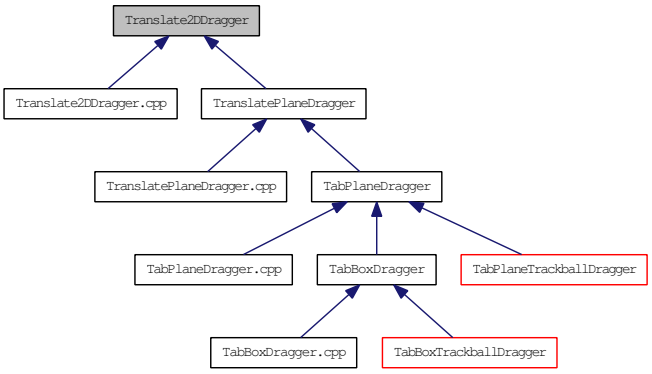
5.37 Translate2DDragger File Reference

```
#include <osgManipulator/Dragger>
#include <osgManipulator/Projector>
#include <osg/PolygonOffset>
```

Include dependency graph for Translate2DDragger:



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



Classes

- class **Translate2DDragger**

  Dragger *(p. 29) for performing 2D translation.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_TRANSLATE2DDRAGGER** 1

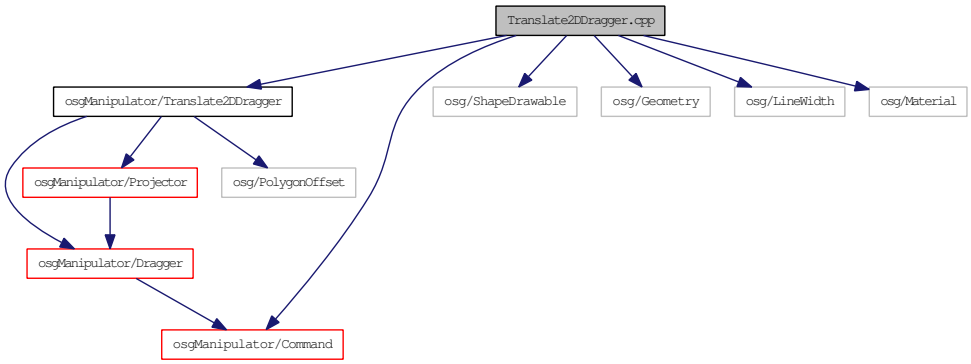
5.37.1 Define Documentation

5.37.1.1 #define OSGMANIPULATOR_TRANSLATE2DDRAGGER 1

5.38 Translate2DDragger.cpp File Reference

```
#include <osgManipulator/Translate2DDragger>
#include <osgManipulator/Command>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/Material>
```

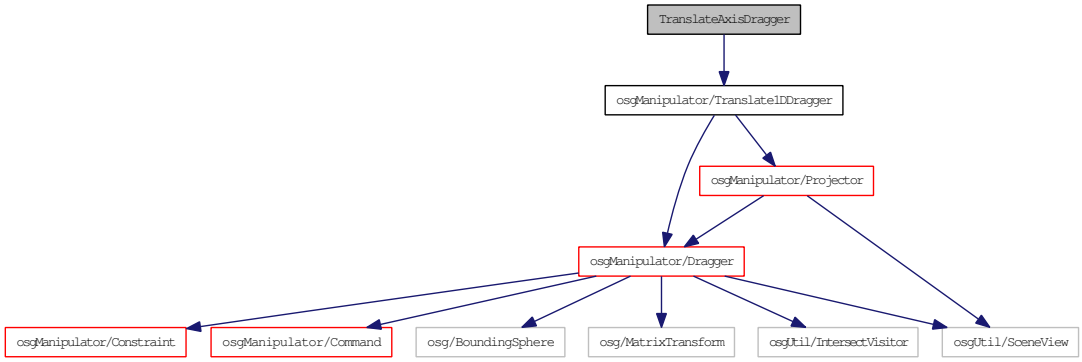
Include dependency graph for Translate2DDragger.cpp:



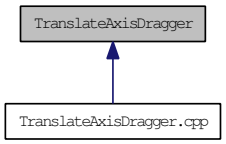
5.39 TranslateAxisDragger File Reference

`#include <osgManipulator/Translate1DDragger>`

Include dependency graph for TranslateAxisDragger:



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



Classes

- class **TranslateAxisDragger**

  Dragger *(p. 29) for performing translation in all three axes.*

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_TRANSLATEAXISDRAGGER** 1

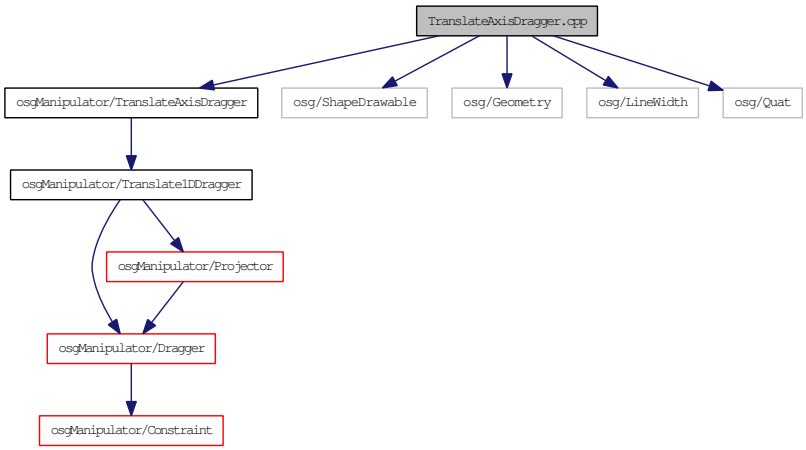
5.39.1 Define Documentation

5.39.1.1 #define OSGMANIPULATOR_TRANSLATEAXISDRAGGER 1

5.40 TranslateAxisDragger.cpp File Reference

```
#include <osgManipulator/TranslateAxisDragger>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/Quat>
```

Include dependency graph for TranslateAxisDragger.cpp:

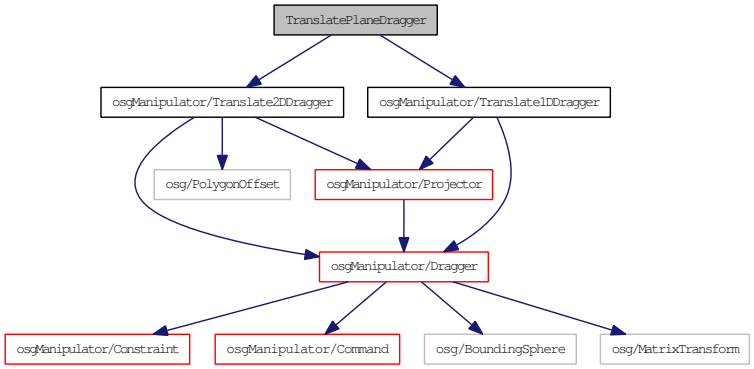


5.41 TranslatePlaneDragger File Reference

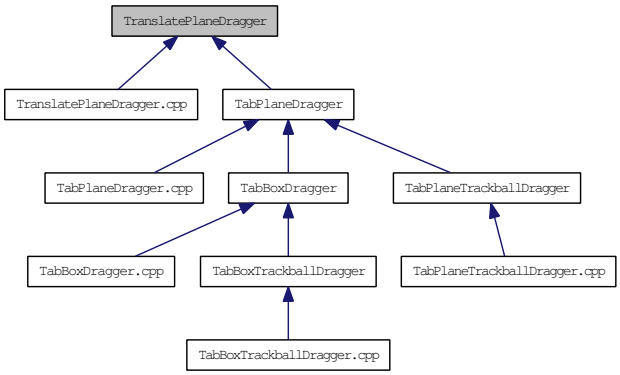
`#include <osgManipulator/Translate2DDragger>`

`#include <osgManipulator/Translate1DDragger>`

Include dependency graph for TranslatePlaneDragger:



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



Classes

- class **TranslatePlaneDragger**

  Tab plane dragger consists of a plane with tabs on it's corners and edges for scaling.

Namespaces

- namespace **osgManipulator**

  *The **osgManipulator** (p. 14) library is a NodeKit that extends the core scene graph to support 3D interactive manipulators.*

Defines

- #define **OSGMANIPULATOR_TRANSLATEPLANEDRAGGER** 1

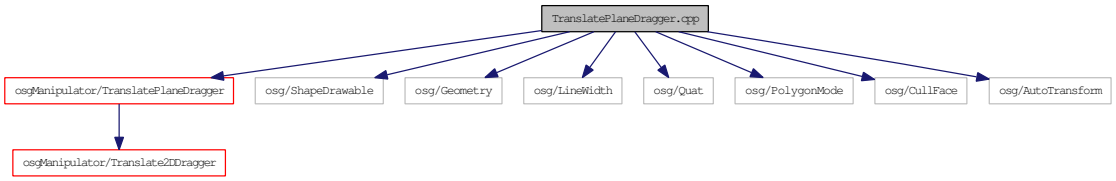
5.41.1 Define Documentation

5.41.1.1 #define OSGMANIPULATOR_TRANSLATEPLANEDRAGGER 1

5.42 TranslatePlaneDragger.cpp File Reference

```
#include <osgManipulator/TranslatePlaneDragger>
#include <osg/ShapeDrawable>
#include <osg/Geometry>
#include <osg/LineWidth>
#include <osg/Quat>
#include <osg/PolygonMode>
#include <osg/CullFace>
#include <osg/AutoTransform>
```

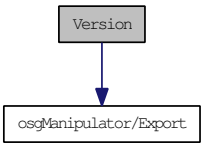
Include dependency graph for TranslatePlaneDragger.cpp:



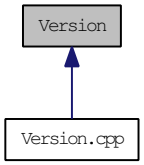
5.43 Version File Reference

`#include <osgManipulator/Export>`

Include dependency graph for Version:



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



Defines

- #define **OSGMANIPULATOR_VERSION** 1

Functions

- OSGMANIPULATOR_EXPORT const char * **osgManipulatorGetLibraryName** ()
  osgManipulatorGetLibraryName() *(*p. *142) returns the library name in human friendly form.*
- OSGMANIPULATOR_EXPORT const char * **osgManipulatorGetVersion** ()
  osgManipulatorGetVersion() (p. *142) returns the library version number.*

5.43.1 Define Documentation

5.43.1.1 #define OSGMANIPULATOR_VERSION 1

5.43.2 Function Documentation

5.43.2.1 OSGMANIPULATOR_EXPORT const char* osgManipulatorGetLibraryName ()

osgManipulatorGetLibraryName() (p. 142) returns the library name in human friendly form.

5.43.2.2 OSGMANIPULATOR_EXPORT const char* osgManipulatorGetVersion ()

osgManipulatorGetVersion() (p. 142) returns the library version number. Numbering convention : OpenSceneGraph-1.0 will return 1.0 from osgManipulatorGetVersion.

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.

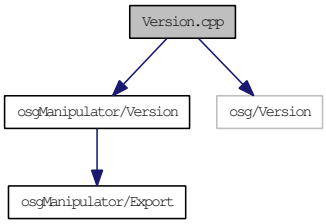
Here is the code to add to your configure.in:

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

5.44 Version.cpp File Reference

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

Include dependency graph for Version.cpp:



Functions

- const char * **osgManipulatorGetLibraryName** ()

  osgManipulatorGetLibraryName() *(*p. *142) returns the library name in human friendly form.*

- const char * **osgManipulatorGetVersion** ()

  osgManipulatorGetVersion() *(*p. *142) returns the library version number.*

5.44.1 Function Documentation

5.44.1.1 const char* osgManipulatorGetLibraryName ()

osgManipulatorGetLibraryName() (p. 142) returns the library name in human friendly form.

5.44.1.2 const char* osgManipulatorGetVersion ()

osgManipulatorGetVersion() (p. 142) returns the library version number. Numbering convention : OpenSceneGraph-1.0 will return 1.0 from osgManipulatorGetVersion.

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) Manipulator library
#
AC_CHECK_LIB(osg, osgManipulatorGetVersion, ,
   [AC_MSG_ERROR(OpenSceneGraph Manipulator library not found. See http://www.openscenegraph.org)],)
```

Index

- Symbols -

~AntiSquish
osgManipulator::AntiSquish, 18
~AntiSquishCallback
anonymous_namespace{AntiSquish.cpp}::AntiSquishCallback, 19
~CommandManager
osgManipulator::CommandManager, 20
~CompositeDragger
osgManipulator::CompositeDragger, 22
~Constraint
osgManipulator::Constraint, 23
~CylinderPlaneProjector
osgManipulator::CylinderPlaneProjector, 26
~CylinderProjector
osgManipulator::CylinderProjector, 28
~Dragger
osgManipulator::Dragger, 31
~GridConstraint
osgManipulator::GridConstraint, 37
~LineProjector
osgManipulator::LineProjector, 40
~MotionCommand
osgManipulator::MotionCommand, 42
~PlaneProjector
osgManipulator::PlaneProjector, 44
~Projector
osgManipulator::Projector, 47
~Rotate3DCommand
osgManipulator::Rotate3DCommand, 49
~RotateCylinderDragger
osgManipulator::RotateCylinderDragger, 52
~RotateSphereDragger
osgManipulator::RotateSphereDragger, 54
~Scale1DCommand
osgManipulator::Scale1DCommand, 56
~Scale1DDragger
osgManipulator::Scale1DDragger, 58
~Scale2DCommand
osgManipulator::Scale2DCommand, 61
~Scale2DDragger
osgManipulator::Scale2DDragger, 64
~ScaleAxisDragger
osgManipulator::ScaleAxisDragger, 66
~ScaleUniformCommand
osgManipulator::ScaleUniformCommand, 69
~SpherePlaneProjector
osgManipulator::SpherePlaneProjector, 71
~SphereProjector
osgManipulator::SphereProjector, 73
~TabBoxDragger
osgManipulator::TabBoxDragger, 74
~TabBoxTrackballDragger
osgManipulator::TabBoxTrackballDragger, 76
~TabPlaneDragger
osgManipulator::TabPlaneDragger, 79
~TabPlaneTrackballDragger
osgManipulator::TabPlaneTrackballDragger, 80
~TrackballDragger
osgManipulator::TrackballDragger, 83
~Translate1DDragger
osgManipulator::Translate1DDragger, 85
~Translate2DDragger
osgManipulator::Translate2DDragger, 87
~TranslateAxisDragger
osgManipulator::TranslateAxisDragger, 88
~TranslateInLineCommand
osgManipulator::TranslateInLineCommand, 91
~TranslateInPlaneCommand
osgManipulator::TranslateInPlaneCommand, 93
~TranslatePlaneDragger
osgManipulator::TranslatePlaneDragger, 95
_MVPW
osgManipulator::PointerInfo, 46
_OSG_ANTISQUISH_
AntiSquish, 98
_activationKeyEvent
osgManipulator::Dragger, 32
_activationModKeyMask
osgManipulator::Dragger, 32
_activationPermittedByKeyEvent
osgManipulator::Dragger, 32
_activationPermittedByModKeyMask
osgManipulator::Dragger, 32
_antiSquish
anonymous_namespace{AntiSquish.cpp}::AntiSquishCallback, 19
_asqCallback
osgManipulator::AntiSquish, 18
_bottomLeftHandleNode
osgManipulator::Scale2DDragger, 65
_bottomLeftHandlePosition
osgManipulator::Scale2DDragger, 65
_bottomRightHandleNode
osgManipulator::Scale2DDragger, 65
_bottomRightHandlePosition
osgManipulator::Scale2DDragger, 65
_cachedLocalToWorld
osgManipulator::AntiSquish, 18
_checkForNodeInNodePath
osgManipulator::Translate1DDragger, 85
_color
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 65
osgManipulator::Translate1DDragger, 85
osgManipulator::Translate2DDragger, 87
_constraints
osgManipulator::Dragger, 32
_cornerScaleDragger
osgManipulator::TabPlaneDragger, 79
_cylinder
osgManipulator::CylinderProjector, 28
_cylinderAxis
osgManipulator::CylinderProjector, 28
_dirty
osgManipulator::AntiSquish, 18
_draggerActive
osgManipulator::Dragger, 32

_draggerCallbacks
osgManipulator::Dragger, 32
_draggerList
osgManipulator::CompositeDragger, 22
_eyeDir
osgManipulator::PointerInfo, 46
_farPoint
osgManipulator::PointerInfo, 46
_front
osgManipulator::CylinderProjector, 28
osgManipulator::SphereProjector, 73
_handleEvents
osgManipulator::Dragger, 32
_handleScaleFactor
osgManipulator::TabPlaneDragger, 79
_hitIter
osgManipulator::PointerInfo, 46
_hitList
osgManipulator::PointerInfo, 46
_horzEdgeScaleDragger
osgManipulator::TabPlaneDragger, 79
_inverseMVPW
osgManipulator::PointerInfo, 46
_leftHandleNode
osgManipulator::Scale1DDragger, 59
_line
osgManipulator::LineProjector, 40
_localToWorld
osgManipulator::DraggerTransformCallback, 35
osgManipulator::Projector, 48
_minScale
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 65
_nearPoint
osgManipulator::PointerInfo, 46
_onCylinder
osgManipulator::CylinderPlaneProjector, 26
_onSphere
osgManipulator::SpherePlaneProjector, 71
_parentDragger
osgManipulator::Dragger, 32
_pickColor
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 65
osgManipulator::Translate1DDragger, 85
osgManipulator::Translate2DDragger, 87
_pivot
osgManipulator::AntiSquish, 18
_plane
osgManipulator::CylinderPlaneProjector, 26
osgManipulator::PlaneProjector, 44
osgManipulator::SpherePlaneProjector, 71
_planeDraggers
osgManipulator::TabBoxDragger, 75
_planeLineEnd
osgManipulator::CylinderPlaneProjector, 26
_planeLineStart
osgManipulator::CylinderPlaneProjector, 26
_pointer
osgManipulator::Dragger, 32
_polygonOffset
osgManipulator::Translate2DDragger, 87
_position
osgManipulator::AntiSquish, 18
_prevPtOnCylinder
osgManipulator::RotateCylinderDragger, 52
_prevPtOnSphere
osgManipulator::RotateSphereDragger, 54
_prevRotation
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
_prevWorldProjPt
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
_projector
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 65
osgManipulator::Translate1DDragger, 85
osgManipulator::Translate2DDragger, 87
_referencePoint
osgManipulator::Scale2DDragger, 65
_rightHandleNode
osgManipulator::Scale1DDragger, 59
_scaleCenter
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 65
_scaleMode
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 65
_selfUpdater
osgManipulator::Dragger, 32
_sphere
osgManipulator::SphereProjector, 73
_startLocalToWorld
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
_startMotionMatrix
osgManipulator::DraggerTransformCallback, 35
_startProjectedPoint
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 65
osgManipulator::Translate1DDragger, 85
osgManipulator::Translate2DDragger, 87
_startWorldToLocal
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
_tabBoxDragger
osgManipulator::TabBoxTrackballDragger, 77
_tabPlaneDragger
osgManipulator::TabPlaneTrackballDragger, 81
_topLeftHandleNode
osgManipulator::Scale2DDragger, 65
_topLeftHandlePosition
osgManipulator::Scale2DDragger, 65
_topRightHandleNode
osgManipulator::Scale2DDragger, 65
_topRightHandlePosition
osgManipulator::Scale2DDragger, 65
_trackballDragger
osgManipulator::TabBoxTrackballDragger, 77
osgManipulator::TabPlaneTrackballDragger, 81
_transform
osgManipulator::DraggerTransformCallback, 35
_translate1DDragger
osgManipulator::TranslatePlaneDragger, 95
_translate2DDragger
osgManipulator::TranslatePlaneDragger, 95
_translateDragger

osgManipulator::TabPlaneDragger, 79
_usePivot
osgManipulator::AntiSquish, 18
_usePosition
osgManipulator::AntiSquish, 18
_usingTranslate1DDragger
osgManipulator::TranslatePlaneDragger, 95
_vertEdgeScaleDragger
osgManipulator::TabPlaneDragger, 79
_worldToLocal
osgManipulator::DraggerTransformCallback, 35
osgManipulator::Projector, 48
_worldToLocalDirty
osgManipulator::Projector, 48
_xDragger
osgManipulator::ScaleAxisDragger, 67
osgManipulator::TrackballDragger, 83
osgManipulator::TranslateAxisDragger, 89
_xyzDragger
osgManipulator::TrackballDragger, 83
_yDragger
osgManipulator::ScaleAxisDragger, 67
osgManipulator::TrackballDragger, 83
osgManipulator::TranslateAxisDragger, 89
_zDragger
osgManipulator::ScaleAxisDragger, 67
osgManipulator::TrackballDragger, 83
osgManipulator::TranslateAxisDragger, 89

- A -

addConstraint
osgManipulator::Dragger, 31
addDragger
osgManipulator::CompositeDragger, 22
addDraggerCallback
osgManipulator::Dragger, 31
addIntersection
osgManipulator::PointerInfo, 46
addTransformUpdating
osgManipulator::Dragger, 31
anonymous_namespace{AntiSquish.cpp}, 7
anonymous_namespace{AntiSquish.cpp}::AntiSquishCallback, 19
∼AntiSquishCallback, 19
_antiSquish, 19
AntiSquishCallback, 19
operator(), 19
anonymous_namespace{Constraint.cpp}, 8
round_to_nearest_int, 8
snap_point_to_grid, 8
anonymous_namespace{Projector.cpp}, 9
computeClosestPointOnLine, 9
computeClosestPoints, 9
computePlaneParallelToAxisAndOrientedToEye, 9
computePlaneThruPointAndOrientedToEye, 9
getCylinderLineIntersection, 9
getLocalEyeDirection, 9
getPlaneLineIntersection, 9
getSphereLineIntersection, 9
getUnitCylinderLineIntersection, 9
anonymous_namespace{Scale1DDragger.cpp}, 10
computeScale, 10
anonymous_namespace{Scale2DDragger.cpp}, 11
computeScale, 11
anonymous_namespace{TabPlaneDragger.cpp}, 12
createCornerScaleDraggerGeometry, 12
createEdgeScaleDraggerGeometry, 12
createHandleNode, 12
createHandleScene, 12
createTranslateDraggerGeometry, 12
anonymous_namespace{TrackballDragger.cpp}, 13
createCircleGeometry, 13
AntiSquish, 97
_OSG_ANTISQUISH_, 98
osgManipulator::AntiSquish, 18
AntiSquish.cpp, 99
AntiSquishCallback
anonymous_namespace{AntiSquish.cpp}::AntiSquishCallback, 19

- C -

clone
osgManipulator::AntiSquish, 18
cloneType
osgManipulator::AntiSquish, 18
Command, 100
OSGMANIPULATOR_COMMAND, 100
Command.cpp, 101
CommandManager, 102
osgManipulator::CommandManager, 20
OSGMANIPULATOR_COMMANDMANAGER, 102
completed
osgManipulator::PointerInfo, 46
CompositeDragger
osgManipulator::CompositeDragger, 22
computeClosestPointOnLine
anonymous_namespace{Projector.cpp}, 9
computeClosestPoints
anonymous_namespace{Projector.cpp}, 9
computeLocalToWorldAndWorldToLocal
osgManipulator::Constraint, 23
computeNodePathToRoot
osgManipulator, 16
computePlaneParallelToAxisAndOrientedToEye
anonymous_namespace{Projector.cpp}, 9
computePlaneThruPointAndOrientedToEye
anonymous_namespace{Projector.cpp}, 9
computeScale
anonymous_namespace{Scale1DDragger.cpp}, 10
anonymous_namespace{Scale2DDragger.cpp}, 11
computeUnSquishedMatrix
osgManipulator::AntiSquish, 18
connect
osgManipulator::CommandManager, 20
constrain
osgManipulator::Constraint, 23
osgManipulator::GridConstraint, 37, 38
Constraint, 103
osgManipulator::Constraint, 23
OSGMANIPULATOR_CONSTRAINT, 103
Constraint.cpp, 104
Constraints
osgManipulator::Dragger, 31
contains
osgManipulator::PointerInfo, 46
containsDragger
osgManipulator::CompositeDragger, 22
createCircleGeometry
anonymous_namespace{TrackballDragger.cpp}, 13
createCommandInverse
osgManipulator::MotionCommand, 42
osgManipulator::Rotate3DCommand, 49

osgManipulator::Scale1DCommand, 56
osgManipulator::Scale2DCommand, 61
osgManipulator::ScaleUniformCommand, 69
osgManipulator::TranslateInLineCommand, 91
osgManipulator::TranslateInPlaneCommand, 93
createCornerScaleDraggerGeometry
anonymous_namespace{TabPlaneDragger.cpp}, 12
createEdgeScaleDraggerGeometry
anonymous_namespace{TabPlaneDragger.cpp}, 12
createHandleNode
anonymous_namespace{TabPlaneDragger.cpp}, 12
createHandleScene
anonymous_namespace{TabPlaneDragger.cpp}, 12
createTranslateDraggerGeometry
anonymous_namespace{TabPlaneDragger.cpp}, 12
cull
ForceCullCallback, 36
CylinderPlaneProjector
osgManipulator::CylinderPlaneProjector, 26
CylinderProjector
osgManipulator::CylinderProjector, 28

- D -

disconnect
osgManipulator::CommandManager, 20
dispatch
osgManipulator::Dragger, 31
Dragger, 105
osgManipulator::Dragger, 31
OSGMANIPULATOR_DRAGGER, 105
Dragger.cpp, 106
DraggerCallback
osgManipulator::DraggerCallback, 33
DraggerCallbacks
osgManipulator::Dragger, 31
DraggerTransformCallback
osgManipulator::DraggerTransformCallback, 34

- E -

ea
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 65
osgManipulator::TabPlaneDragger, 79
osgManipulator::Translate2DDragger, 87
osgManipulator::TranslatePlaneDragger, 95
Export, 107
META_OSGMANIPULATOR_Object, 107
OSGMANIPULATOR_EXPORT, 107
OSGMANIPULATOR_EXPORT_, 107

- F -

findDragger
osgManipulator::CompositeDragger, 22
FINISH
osgManipulator::MotionCommand, 42
ForceCullCallback, 36
cull, 36

- G -

getActivationKeyEvent
osgManipulator::Dragger, 31
getActivationModKeyMask
osgManipulator::Dragger, 31
getBottomLeftHandleNode
osgManipulator::Scale2DDragger, 64
getBottomLeftHandlePosition
osgManipulator::Scale2DDragger, 64
getBottomRightHandleNode
osgManipulator::Scale2DDragger, 64
getBottomRightHandlePosition
osgManipulator::Scale2DDragger, 64
getColor
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 58
osgManipulator::Scale2DDragger, 64
osgManipulator::Translate1DDragger, 85
osgManipulator::Translate2DDragger, 87
getComposite
osgManipulator::CompositeDragger, 22
osgManipulator::Dragger, 31
getConnectedSelections
osgManipulator::CommandManager, 20
getConstraints
osgManipulator::Dragger, 31
getCylinder
osgManipulator::CylinderProjector, 28
getCylinderLineIntersection
anonymous_namespace{Projector.cpp}, 9
getDragger
osgManipulator::CompositeDragger, 22
getDraggerActive
osgManipulator::Dragger, 31
getDraggerCallbacks
osgManipulator::Dragger, 31
getEyeDir
osgManipulator::PointerInfo, 46
getHandleEvents
osgManipulator::Dragger, 31
getLeftHandleNode
osgManipulator::Scale1DDragger, 58
getLeftHandlePosition
osgManipulator::Scale1DDragger, 58
getLineEnd
osgManipulator::LineProjector, 40
osgManipulator::TranslateInLineCommand, 91
getLineStart
osgManipulator::LineProjector, 40
osgManipulator::TranslateInLineCommand, 91
getLocalEyeDirection
anonymous_namespace{Projector.cpp}, 9
getLocalIntersectPoint
osgManipulator::PointerInfo, 46
getLocalToWorld
osgManipulator::Constraint, 24
osgManipulator::MotionCommand, 42
osgManipulator::Projector, 47
getMinScale
osgManipulator::Scale1DCommand, 56
osgManipulator::Scale1DDragger, 58
osgManipulator::Scale2DCommand, 61
osgManipulator::Scale2DDragger, 64
getMotionMatrix
osgManipulator::MotionCommand, 42
osgManipulator::Rotate3DCommand, 50
osgManipulator::Scale1DCommand, 56
osgManipulator::Scale2DCommand, 61
osgManipulator::ScaleUniformCommand, 69
osgManipulator::TranslateInLineCommand, 91

osgManipulator::TranslateInPlaneCommand, 93
getNearFarPoints
osgManipulator::PointerInfo, 46
getNumDraggers
osgManipulator::CompositeDragger, 22
getOrigin
osgManipulator::GridConstraint, 38
getParentDragger
osgManipulator::Dragger, 31
getPickColor
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 58
osgManipulator::Scale2DDragger, 64
osgManipulator::Translate1DDragger, 85
osgManipulator::Translate2DDragger, 87
getPivot
osgManipulator::AntiSquish, 18
getPlane
osgManipulator::PlaneProjector, 44
osgManipulator::TranslateInPlaneCommand, 93
getPlaneLineIntersection
anonymous_namespace{Projector.cpp}, 9
getPosition
osgManipulator::AntiSquish, 18
getReferenceNode
osgManipulator::Constraint, 24
getReferencePoint
osgManipulator::Scale1DCommand, 56
osgManipulator::Scale2DCommand, 61
osgManipulator::TranslateInPlaneCommand, 93
getRightHandleNode
osgManipulator::Scale1DDragger, 58
getRightHandlePosition
osgManipulator::Scale1DDragger, 58
getRotation
osgManipulator::CylinderPlaneProjector, 26
osgManipulator::Rotate3DCommand, 50
osgManipulator::SpherePlaneProjector, 71
getScale
osgManipulator::Scale1DCommand, 56
osgManipulator::Scale2DCommand, 61
osgManipulator::ScaleUniformCommand, 69
getScaleCenter
osgManipulator::Scale1DCommand, 56
osgManipulator::Scale2DCommand, 61
osgManipulator::ScaleUniformCommand, 69
getSpacing
osgManipulator::GridConstraint, 38
getSphere
osgManipulator::SphereProjector, 73
getSphereLineIntersection
anonymous_namespace{Projector.cpp}, 9
getStage
osgManipulator::MotionCommand, 42
getTopLeftHandleNode
osgManipulator::Scale2DDragger, 64
getTopLeftHandlePosition
osgManipulator::Scale2DDragger, 64
getTopRightHandleNode
osgManipulator::Scale2DDragger, 64
getTopRightHandlePosition
osgManipulator::Scale2DDragger, 64
getTransform
osgManipulator::DraggerTransformCallback, 34
getTranslate1DDragger
osgManipulator::TranslatePlaneDragger, 95
getTranslate2DDragger
osgManipulator::TranslatePlaneDragger, 95
getTranslation
osgManipulator::TranslateInLineCommand, 91
osgManipulator::TranslateInPlaneCommand, 93
getUnitCylinderLineIntersection
anonymous_namespace{Projector.cpp}, 9
getWorldToLocal
osgManipulator::Constraint, 24
osgManipulator::MotionCommand, 42
osgManipulator::Projector, 47
GridConstraint
osgManipulator::GridConstraint, 37

- H -

handle
osgManipulator::CompositeDragger, 22
osgManipulator::Dragger, 31

- I -

include/ Directory Reference, 3
include/osgManipulator/ Directory Reference, 5
IntersectionList
osgManipulator::PointerInfo, 46
isPointInFront
osgManipulator::CylinderProjector, 28
osgManipulator::SphereProjector, 73
isProjectionOnCylinder
osgManipulator::CylinderPlaneProjector, 26
isProjectionOnSphere
osgManipulator::SpherePlaneProjector, 71
isSameKindAs
osgManipulator::AntiSquish, 18

- L -

LineProjector
osgManipulator::LineProjector, 40

- M -

mainpage.h, 108
META_Node
osgManipulator::Dragger, 31
META_Object
osgManipulator::DraggerCallback, 33
META_OSGMANIPULATOR_Object
Export, 107
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 58
osgManipulator::Scale2DDragger, 64
osgManipulator::ScaleAxisDragger, 66
osgManipulator::TabBoxDragger, 74
osgManipulator::TabBoxTrackballDragger, 76
osgManipulator::TabPlaneDragger, 79
osgManipulator::TabPlaneTrackballDragger, 80
osgManipulator::TrackballDragger, 83
osgManipulator::Translate2DDragger, 87
osgManipulator::TranslateAxisDragger, 88
osgManipulator::TranslatePlaneDragger, 95
MotionCommand
osgManipulator::MotionCommand, 42
MOVE

osgManipulator::MotionCommand, 42

- N -

next
osgManipulator::PointerInfo, 46
NodePathIntersectionPair
osgManipulator::PointerInfo, 46
NONE
osgManipulator::MotionCommand, 42

- O -

operator()
anonymous_namespace{AntiSquish.cpp}::AntiSquishCallback, 19
osgManipulator, 14
computeNodePathToRoot, 16
Selection, 16
setDrawableToAlwaysCull, 16
setMaterialColor, 16
osgManipulator::AntiSquish, 17
∼AntiSquish, 18
_asqCallback, 18
_cachedLocalToWorld, 18
_dirty, 18
_pivot, 18
_position, 18
_usePivot, 18
_usePosition, 18
AntiSquish, 18
clone, 18
cloneType, 18
computeUnSquishedMatrix, 18
getPivot, 18
getPosition, 18
isSameKindAs, 18
setPivot, 18
setPosition, 18
osgManipulator::CommandManager, 20
∼CommandManager, 20
CommandManager, 20
connect, 20
disconnect, 20
getConnectedSelections, 20
Selections, 20
osgManipulator::CompositeDragger, 21
∼CompositeDragger, 22
_draggerList, 22
addDragger, 22
CompositeDragger, 22
containsDragger, 22
findDragger, 22
getComposite, 22
getDragger, 22
getNumDraggers, 22
handle, 22
removeDragger, 22
setParentDragger, 22
osgManipulator::Constraint, 23
∼Constraint, 23
computeLocalToWorldAndWorldToLocal, 23
constrain, 23
Constraint, 23
getLocalToWorld, 24
getReferenceNode, 24
getWorldToLocal, 24
osgManipulator::CylinderPlaneProjector, 25
∼CylinderPlaneProjector, 26
_onCylinder, 26
_plane, 26
_planeLineEnd, 26
_planeLineStart, 26
CylinderPlaneProjector, 26
getRotation, 26
isProjectionOnCylinder, 26
project, 26
osgManipulator::CylinderProjector, 27
∼CylinderProjector, 28
_cylinder, 28
_cylinderAxis, 28
_front, 28
CylinderProjector, 28
getCylinder, 28
isPointInFront, 28
project, 28
setCylinder, 28
setFront, 28
osgManipulator::Dragger, 29
∼Dragger, 31
_activationKeyEvent, 32
_activationModKeyMask, 32
_activationPermittedByKeyEvent, 32
_activationPermittedByModKeyMask, 32
_constraints, 32
_draggerActive, 32
_draggerCallbacks, 32
_handleEvents, 32
_parentDragger, 32
_pointer, 32
_selfUpdater, 32
addConstraint, 31
addDraggerCallback, 31
addTransformUpdating, 31
Constraints, 31
dispatch, 31
Dragger, 31
DraggerCallbacks, 31
getActivationKeyEvent, 31
getActivationModKeyMask, 31
getComposite, 31
getConstraints, 31
getDraggerActive, 31
getDraggerCallbacks, 31
getHandleEvents, 31
getParentDragger, 31
handle, 31
META_Node, 31
receive, 31
removeConstraint, 32
removeDraggerCallback, 32
removeTransformUpdating, 32
setActivationKeyEvent, 32
setActivationModKeyMask, 32
setDraggerActive, 32
setHandleEvents, 32
setupDefaultGeometry, 32
traverse, 32
osgManipulator::DraggerCallback, 33
DraggerCallback, 33
META_Object, 33
receive, 33
osgManipulator::DraggerTransformCallback, 34

_localToWorld, 35
_startMotionMatrix, 35
_transform, 35
_worldToLocal, 35
DraggerTransformCallback, 34
getTransform, 34
receive, 34
osgManipulator::GridConstraint, 37
~GridConstraint, 37
constrain, 37, 38
getOrigin, 38
getSpacing, 38
GridConstraint, 37
setOrigin, 38
setSpacing, 38
osgManipulator::LineProjector, 39
~LineProjector, 40
_line, 40
getLineEnd, 40
getLineStart, 40
LineProjector, 40
project, 40
setLine, 40
osgManipulator::MotionCommand, 41
~MotionCommand, 42
createCommandInverse, 42
FINISH, 42
getLocalToWorld, 42
getMotionMatrix, 42
getStage, 42
getWorldToLocal, 42
MotionCommand, 42
MOVE, 42
NONE, 42
setLocalToWorldAndWorldToLocal, 42
setStage, 42
Stage, 42
START, 42
osgManipulator::PlaneProjector, 43
~PlaneProjector, 44
_plane, 44
getPlane, 44
PlaneProjector, 44
project, 44
setPlane, 44
osgManipulator::PointerInfo, 45
_MVPW, 46
_eyeDir, 46
_farPoint, 46
_hitIter, 46
_hitList, 46
_inverseMVPW, 46
_nearPoint, 46
addIntersection, 46
completed, 46
contains, 46
getEyeDir, 46
getLocalIntersectPoint, 46
getNearFarPoints, 46
IntersectionList, 46
next, 46
NodePathIntersectionPair, 46
PointerInfo, 46
projectWindowXYIntoObject, 46
reset, 46
setCamera, 46
setMousePosition, 46
setNearFarPoints, 46
osgManipulator::Projector, 47
~Projector, 47
_localToWorld, 48
_worldToLocal, 48
_worldToLocalDirty, 48
getLocalToWorld, 47
getWorldToLocal, 47
project, 48
Projector, 47
setLocalToWorld, 48
osgManipulator::Rotate3DCommand, 49
~Rotate3DCommand, 49
createCommandInverse, 49
getMotionMatrix, 50
getRotation, 50
Rotate3DCommand, 49
setRotation, 50
osgManipulator::RotateCylinderDragger, 51
~RotateCylinderDragger, 52
_color, 52
_pickColor, 52
_prevPtOnCylinder, 52
_prevRotation, 52
_prevWorldProjPt, 52
_projector, 52
_startLocalToWorld, 52
_startWorldToLocal, 52
ea, 52
getColor, 52
getPickColor, 52
META_OSGMANIPULATOR_Object, 52
RotateCylinderDragger, 52
setColor, 52
setPickColor, 52
setupDefaultGeometry, 52
us, 52
osgManipulator::RotateSphereDragger, 53
~RotateSphereDragger, 54
_color, 54
_pickColor, 54
_prevPtOnSphere, 54
_prevRotation, 54
_prevWorldProjPt, 54
_projector, 54
_startLocalToWorld, 54
_startWorldToLocal, 54
ea, 54
getColor, 54
getPickColor, 54
META_OSGMANIPULATOR_Object, 54
RotateSphereDragger, 54
setColor, 54
setPickColor, 54
setupDefaultGeometry, 54
us, 54
osgManipulator::Scale1DCommand, 55
~Scale1DCommand, 56
createCommandInverse, 56
getMinScale, 56
getMotionMatrix, 56
getReferencePoint, 56
getScale, 56
getScaleCenter, 56
Scale1DCommand, 56

setMinScale, 56
setReferencePoint, 56
setScale, 56
setScaleCenter, 56
osgManipulator::Scale1DDragger, 57
∼Scale1DDragger, 58
_color, 59
_leftHandleNode, 59
_minScale, 59
_pickColor, 59
_projector, 59
_rightHandleNode, 59
_scaleCenter, 59
_scaleMode, 59
_startProjectedPoint, 59
ea, 59
getColor, 58
getLeftHandleNode, 58
getLeftHandlePosition, 58
getMinScale, 58
getPickColor, 58
getRightHandleNode, 58
getRightHandlePosition, 58
META_OSGMANIPULATOR_Object, 58
Scale1DDragger, 58
SCALE_WITH_OPPOSITE_HANDLE_AS_PIVOT, 58
SCALE_WITH_ORIGIN_AS_PIVOT, 58
ScaleMode, 58
setColor, 58
setLeftHandleNode, 59
setLeftHandlePosition, 59
setMinScale, 59
setPickColor, 59
setRightHandleNode, 59
setRightHandlePosition, 59
setupDefaultGeometry, 59
us, 59
osgManipulator::Scale2DCommand, 60
∼Scale2DCommand, 61
createCommandInverse, 61
getMinScale, 61
getMotionMatrix, 61
getReferencePoint, 61
getScale, 61
getScaleCenter, 61
Scale2DCommand, 61
setMinScale, 61
setReferencePoint, 61
setScale, 61
setScaleCenter, 61
osgManipulator::Scale2DDragger, 62
∼Scale2DDragger, 64
_bottomLeftHandleNode, 65
_bottomLeftHandlePosition, 65
_bottomRightHandleNode, 65
_bottomRightHandlePosition, 65
_color, 65
_minScale, 65
_pickColor, 65
_projector, 65
_referencePoint, 65
_scaleCenter, 65
_scaleMode, 65
_startProjectedPoint, 65
_topLeftHandleNode, 65
_topLeftHandlePosition, 65
_topRightHandleNode, 65
_topRightHandlePosition, 65
ea, 65
getBottomLeftHandleNode, 64
getBottomLeftHandlePosition, 64
getBottomRightHandleNode, 64
getBottomRightHandlePosition, 64
getColor, 64
getMinScale, 64
getPickColor, 64
getTopLeftHandleNode, 64
getTopLeftHandlePosition, 64
getTopRightHandleNode, 64
getTopRightHandlePosition, 64
META_OSGMANIPULATOR_Object, 64
Scale2DDragger, 64
SCALE_WITH_OPPOSITE_HANDLE_AS_PIVOT, 63
SCALE_WITH_ORIGIN_AS_PIVOT, 63
ScaleMode, 63
setBottomLeftHandleNode, 64
setBottomLeftHandlePosition, 64
setBottomRightHandleNode, 64
setBottomRightHandlePosition, 64
setColor, 64
setMinScale, 64
setPickColor, 64
setTopLeftHandleNode, 64
setTopLeftHandlePosition, 64
setTopRightHandleNode, 64
setTopRightHandlePosition, 64
setupDefaultGeometry, 64
us, 65
osgManipulator::ScaleAxisDragger, 66
∼ScaleAxisDragger, 66
_xDragger, 67
_yDragger, 67
_zDragger, 67
META_OSGMANIPULATOR_Object, 66
ScaleAxisDragger, 66
osgManipulator::ScaleUniformCommand, 68
∼ScaleUniformCommand, 69
createCommandInverse, 69
getMotionMatrix, 69
getScale, 69
getScaleCenter, 69
ScaleUniformCommand, 69
setScale, 69
setScaleCenter, 69
osgManipulator::SpherePlaneProjector, 70
∼SpherePlaneProjector, 71
_onSphere, 71
_plane, 71
getRotation, 71
isProjectionOnSphere, 71
project, 71
SpherePlaneProjector, 71
osgManipulator::SphereProjector, 72
∼SphereProjector, 73
_front, 73
_sphere, 73
getSphere, 73
isPointInFront, 73
project, 73
setFront, 73
setSphere, 73
SphereProjector, 73

osgManipulator::TabBoxDragger, 74
~TabBoxDragger, 74
_planeDraggers, 75
META_OSGMANIPULATOR_Object, 74
setPlaneColor, 74
TabBoxDragger, 74
osgManipulator::TabBoxTrackballDragger, 76
~TabBoxTrackballDragger, 76
_tabBoxDragger, 77
_trackballDragger, 77
META_OSGMANIPULATOR_Object, 76
TabBoxTrackballDragger, 76
osgManipulator::TabPlaneDragger, 78
~TabPlaneDragger, 79
_cornerScaleDragger, 79
_handleScaleFactor, 79
_horzEdgeScaleDragger, 79
_translateDragger, 79
_vertEdgeScaleDragger, 79
ea, 79
META_OSGMANIPULATOR_Object, 79
setPlaneColor, 79
setupDefaultGeometry, 79
TabPlaneDragger, 79
us, 79
osgManipulator::TabPlaneTrackballDragger, 80
~TabPlaneTrackballDragger, 80
_tabPlaneDragger, 81
_trackballDragger, 81
META_OSGMANIPULATOR_Object, 80
setPlaneColor, 80
TabPlaneTrackballDragger, 80
osgManipulator::TrackballDragger, 82
~TrackballDragger, 83
_xDragger, 83
_xyzDragger, 83
_yDragger, 83
_zDragger, 83
META_OSGMANIPULATOR_Object, 83
TrackballDragger, 83
osgManipulator::Translate1DDragger, 84
~Translate1DDragger, 85
_checkForNodeInNodePath, 85
_color, 85
_pickColor, 85
_projector, 85
_startProjectedPoint, 85
getColor, 85
getPickColor, 85
setCheckForNodeInNodePath, 85
setColor, 85
setPickColor, 85
setupDefaultGeometry, 85
Translate1DDragger, 85
osgManipulator::Translate2DDragger, 86
~Translate2DDragger, 87
_color, 87
_pickColor, 87
_polygonOffset, 87
_projector, 87
_startProjectedPoint, 87
ea, 87
getColor, 87
getPickColor, 87
META_OSGMANIPULATOR_Object, 87
setColor, 87
setPickColor, 87
setupDefaultGeometry, 87
Translate2DDragger, 87
us, 87
osgManipulator::TranslateAxisDragger, 88
~TranslateAxisDragger, 88
_xDragger, 89
_yDragger, 89
_zDragger, 89
META_OSGMANIPULATOR_Object, 88
TranslateAxisDragger, 88
osgManipulator::TranslateInLineCommand, 90
~TranslateInLineCommand, 91
createCommandInverse, 91
getLineEnd, 91
getLineStart, 91
getMotionMatrix, 91
getTranslation, 91
setLine, 91
setTranslation, 91
TranslateInLineCommand, 91
osgManipulator::TranslateInPlaneCommand, 92
~TranslateInPlaneCommand, 93
createCommandInverse, 93
getMotionMatrix, 93
getPlane, 93
getReferencePoint, 93
getTranslation, 93
setPlane, 93
setReferencePoint, 93
setTranslation, 93
TranslateInPlaneCommand, 93
osgManipulator::TranslatePlaneDragger, 94
~TranslatePlaneDragger, 95
_translate1DDragger, 95
_translate2DDragger, 95
_usingTranslate1DDragger, 95
ea, 95
getTranslate1DDragger, 95
getTranslate2DDragger, 95
META_OSGMANIPULATOR_Object, 95
setColor, 95
setupDefaultGeometry, 95
TranslatePlaneDragger, 95
us, 95
OSGMANIPULATOR_COMMAND
Command, 100
OSGMANIPULATOR_COMMANDMANAGER
CommandManager, 102
OSGMANIPULATOR_CONSTRAINT
Constraint, 103
OSGMANIPULATOR_DRAGGER
Dragger, 105
OSGMANIPULATOR_EXPORT
Export, 107
OSGMANIPULATOR_EXPORT_
Export, 107
OSGMANIPULATOR_PROJECTOR
Projector, 110
OSGMANIPULATOR_ROTATECYLINDERDRAGGER
RotateCylinderDragger, 112
OSGMANIPULATOR_ROTATESPHEREDRAGGER
RotateSphereDragger, 114
OSGMANIPULATOR_SCALE1DDRAGGER
Scale1DDragger, 116
OSGMANIPULATOR_SCALE2DDRAGGER

Scale2DDragger, 118
OSGMANIPULATOR_SCALEAXISDRAGGER
ScaleAxisDragger, 120
OSGMANIPULATOR_SELECTION
Selection, 122
OSGMANIPULATOR_TABBOXDRAGGER
TabBoxDragger, 123
OSGMANIPULATOR_TABBOXTRACKBALLDRAGGER
TabBoxTrackballDragger, 125
OSGMANIPULATOR_TABPLANEDRAGGER
TabPlaneDragger, 127
OSGMANIPULATOR_TABPLANETRACKBALLDRAGGER
TabPlaneTrackballDragger, 129
OSGMANIPULATOR_TRACKBALLDRAGGER
TrackballDragger, 131
OSGMANIPULATOR_TRANSLATE1DDRAGGER
Translate1DDragger, 133
OSGMANIPULATOR_TRANSLATE2DDRAGGER
Translate2DDragger, 135
OSGMANIPULATOR_TRANSLATEAXISDRAGGER
TranslateAxisDragger, 137
OSGMANIPULATOR_TRANSLATEPLANEDRAGGER
TranslatePlaneDragger, 139
OSGMANIPULATOR_VERSION
Version, 141
osgManipulatorGetLibraryName
Version, 141
Version.cpp, 142
osgManipulatorGetVersion
Version, 141
Version.cpp, 142

- P -

PlaneProjector
osgManipulator::PlaneProjector, 44
PointerInfo
osgManipulator::PointerInfo, 46
project
osgManipulator::CylinderPlaneProjector, 26
osgManipulator::CylinderProjector, 28
osgManipulator::LineProjector, 40
osgManipulator::PlaneProjector, 44
osgManipulator::Projector, 48
osgManipulator::SpherePlaneProjector, 71
osgManipulator::SphereProjector, 73
Projector, 109
osgManipulator::Projector, 47
OSGMANIPULATOR_PROJECTOR, 110
Projector.cpp, 111
projectWindowXYIntoObject
osgManipulator::PointerInfo, 46

- R -

receive
osgManipulator::Dragger, 31
osgManipulator::DraggerCallback, 33
osgManipulator::DraggerTransformCallback, 34
removeConstraint
osgManipulator::Dragger, 32
removeDragger
osgManipulator::CompositeDragger, 22
removeDraggerCallback
osgManipulator::Dragger, 32
removeTransformUpdating
osgManipulator::Dragger, 32
reset
osgManipulator::PointerInfo, 46
Rotate3DCommand
osgManipulator::Rotate3DCommand, 49
RotateCylinderDragger, 112
osgManipulator::RotateCylinderDragger, 52
OSGMANIPULATOR_ROTATECYLINDERDRAGGER, 112
RotateCylinderDragger.cpp, 113
RotateSphereDragger, 114
osgManipulator::RotateSphereDragger, 54
OSGMANIPULATOR_ROTATESPHEREDRAGGER, 114
RotateSphereDragger.cpp, 115
round_to_nearest_int
anonymous_namespace{Constraint.cpp}, 8

- S -

Scale1DCommand
osgManipulator::Scale1DCommand, 56
Scale1DDragger, 116
osgManipulator::Scale1DDragger, 58
OSGMANIPULATOR_SCALE1DDRAGGER, 116
Scale1DDragger.cpp, 117
Scale2DCommand
osgManipulator::Scale2DCommand, 61
Scale2DDragger, 118
osgManipulator::Scale2DDragger, 64
OSGMANIPULATOR_SCALE2DDRAGGER, 118
Scale2DDragger.cpp, 119
SCALE_WITH_OPPOSITE_HANDLE_AS_PIVOT
osgManipulator::Scale1DDragger, 58
osgManipulator::Scale2DDragger, 63
SCALE_WITH_ORIGIN_AS_PIVOT
osgManipulator::Scale1DDragger, 58
osgManipulator::Scale2DDragger, 63
ScaleAxisDragger, 120
osgManipulator::ScaleAxisDragger, 66
OSGMANIPULATOR_SCALEAXISDRAGGER, 120
ScaleAxisDragger.cpp, 121
ScaleMode
osgManipulator::Scale1DDragger, 58
osgManipulator::Scale2DDragger, 63
ScaleUniformCommand
osgManipulator::ScaleUniformCommand, 69
Selection, 122
osgManipulator, 16
OSGMANIPULATOR_SELECTION, 122
Selections
osgManipulator::CommandManager, 20
setActivationKeyEvent
osgManipulator::Dragger, 32
setActivationModKeyMask
osgManipulator::Dragger, 32
setBottomLeftHandleNode
osgManipulator::Scale2DDragger, 64
setBottomLeftHandlePosition
osgManipulator::Scale2DDragger, 64
setBottomRightHandleNode
osgManipulator::Scale2DDragger, 64
setBottomRightHandlePosition
osgManipulator::Scale2DDragger, 64
setCamera
osgManipulator::PointerInfo, 46
setCheckForNodeInNodePath
osgManipulator::Translate1DDragger, 85

setColor
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 58
osgManipulator::Scale2DDragger, 64
osgManipulator::Translate1DDragger, 85
osgManipulator::Translate2DDragger, 87
osgManipulator::TranslatePlaneDragger, 95
setCylinder
osgManipulator::CylinderProjector, 28
setDraggerActive
osgManipulator::Dragger, 32
setDrawableToAlwaysCull
osgManipulator, 16
setFront
osgManipulator::CylinderProjector, 28
osgManipulator::SphereProjector, 73
setHandleEvents
osgManipulator::Dragger, 32
setLeftHandleNode
osgManipulator::Scale1DDragger, 59
setLeftHandlePosition
osgManipulator::Scale1DDragger, 59
setLine
osgManipulator::LineProjector, 40
osgManipulator::TranslateInLineCommand, 91
setLocalToWorld
osgManipulator::Projector, 48
setLocalToWorldAndWorldToLocal
osgManipulator::MotionCommand, 42
setMaterialColor
osgManipulator, 16
setMinScale
osgManipulator::Scale1DCommand, 56
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DCommand, 61
osgManipulator::Scale2DDragger, 64
setMousePosition
osgManipulator::PointerInfo, 46
setNearFarPoints
osgManipulator::PointerInfo, 46
setOrigin
osgManipulator::GridConstraint, 38
setParentDragger
osgManipulator::CompositeDragger, 22
setPickColor
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 64
osgManipulator::Translate1DDragger, 85
osgManipulator::Translate2DDragger, 87
setPivot
osgManipulator::AntiSquish, 18
setPlane
osgManipulator::PlaneProjector, 44
osgManipulator::TranslateInPlaneCommand, 93
setPlaneColor
osgManipulator::TabBoxDragger, 74
osgManipulator::TabPlaneDragger, 79
osgManipulator::TabPlaneTrackballDragger, 80
setPosition
osgManipulator::AntiSquish, 18
setReferencePoint
osgManipulator::Scale1DCommand, 56
osgManipulator::Scale2DCommand, 61
osgManipulator::TranslateInPlaneCommand, 93
setRightHandleNode
osgManipulator::Scale1DDragger, 59
setRightHandlePosition
osgManipulator::Scale1DDragger, 59
setRotation
osgManipulator::Rotate3DCommand, 50
setScale
osgManipulator::Scale1DCommand, 56
osgManipulator::Scale2DCommand, 61
osgManipulator::ScaleUniformCommand, 69
setScaleCenter
osgManipulator::Scale1DCommand, 56
osgManipulator::Scale2DCommand, 61
osgManipulator::ScaleUniformCommand, 69
setSpacing
osgManipulator::GridConstraint, 38
setSphere
osgManipulator::SphereProjector, 73
setStage
osgManipulator::MotionCommand, 42
setTopLeftHandleNode
osgManipulator::Scale2DDragger, 64
setTopLeftHandlePosition
osgManipulator::Scale2DDragger, 64
setTopRightHandleNode
osgManipulator::Scale2DDragger, 64
setTopRightHandlePosition
osgManipulator::Scale2DDragger, 64
setTranslation
osgManipulator::TranslateInLineCommand, 91
osgManipulator::TranslateInPlaneCommand, 93
setupDefaultGeometry
osgManipulator::Dragger, 32
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 64
osgManipulator::TabPlaneDragger, 79
osgManipulator::Translate1DDragger, 85
osgManipulator::Translate2DDragger, 87
osgManipulator::TranslatePlaneDragger, 95
snap_point_to_grid
anonymous_namespace{Constraint.cpp}, 8
SpherePlaneProjector
osgManipulator::SpherePlaneProjector, 71
SphereProjector
osgManipulator::SphereProjector, 73
src/ Directory Reference, 6
src/osgManipulator/ Directory Reference, 4
Stage
osgManipulator::MotionCommand, 42
START
osgManipulator::MotionCommand, 42

- T -

TabBoxDragger, 123
osgManipulator::TabBoxDragger, 74
OSGMANIPULATOR_TABBOXDRAGGER, 123
TabBoxDragger.cpp, 124
TabBoxTrackballDragger, 125
osgManipulator::TabBoxTrackballDragger, 76
OSGMANIPULATOR_TABBOXTRACKBALLDRAGGER, 125
TabBoxTrackballDragger.cpp, 126
TabPlaneDragger, 127

osgManipulator::TabPlaneDragger, 79
OSGMANIPULATOR_TABPLANEDRAGGER, 127
TabPlaneDragger.cpp, 128
TabPlaneTrackballDragger, 129
osgManipulator::TabPlaneTrackballDragger, 80
OSGMANIPULATOR_TABPLANETRACKBALLDRAGGER, 129
TabPlaneTrackballDragger.cpp, 130
TrackballDragger, 131
osgManipulator::TrackballDragger, 83
OSGMANIPULATOR_TRACKBALLDRAGGER, 131
TrackballDragger.cpp, 132
Translate1DDragger, 133
osgManipulator::Translate1DDragger, 85
OSGMANIPULATOR_TRANSLATE1DDRAGGER, 133
Translate1DDragger.cpp, 134
Translate2DDragger, 135
osgManipulator::Translate2DDragger, 87
OSGMANIPULATOR_TRANSLATE2DDRAGGER, 135
Translate2DDragger.cpp, 136
TranslateAxisDragger, 137
osgManipulator::TranslateAxisDragger, 88
OSGMANIPULATOR_TRANSLATEAXISDRAGGER, 137
TranslateAxisDragger.cpp, 138
TranslateInLineCommand
osgManipulator::TranslateInLineCommand, 91
TranslateInPlaneCommand
osgManipulator::TranslateInPlaneCommand, 93
TranslatePlaneDragger, 139
osgManipulator::TranslatePlaneDragger, 95
OSGMANIPULATOR_TRANSLATEPLANEDRAGGER, 139
TranslatePlaneDragger.cpp, 140
traverse
osgManipulator::Dragger, 32

- U -

us
osgManipulator::RotateCylinderDragger, 52
osgManipulator::RotateSphereDragger, 54
osgManipulator::Scale1DDragger, 59
osgManipulator::Scale2DDragger, 65
osgManipulator::TabPlaneDragger, 79
osgManipulator::Translate2DDragger, 87
osgManipulator::TranslatePlaneDragger, 95

- V -

Version, 141
OSGMANIPULATOR_VERSION, 141
osgManipulatorGetLibraryName, 141
osgManipulatorGetVersion, 141
Version.cpp, 142
osgManipulatorGetLibraryName, 142
osgManipulatorGetVersion, 142