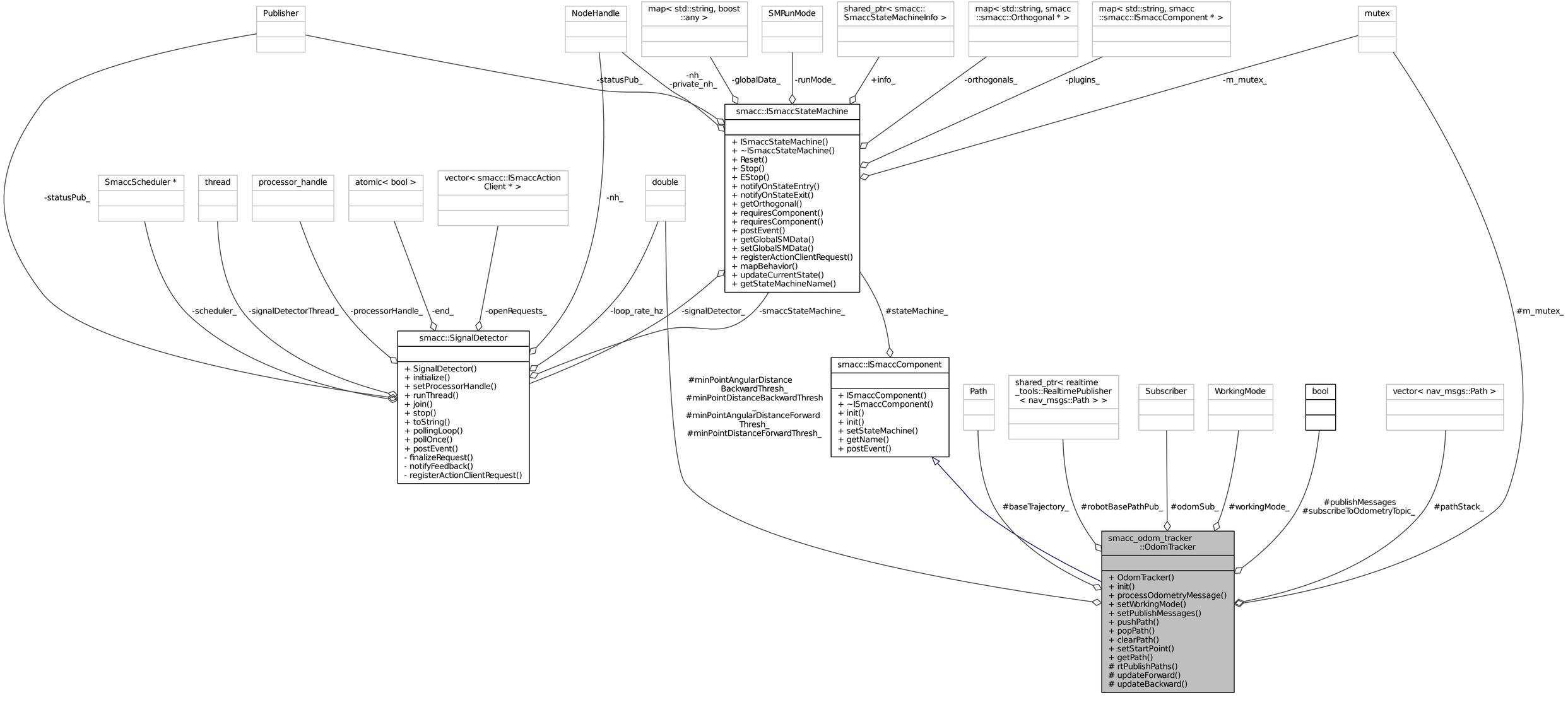
Publisher
NodeHandle
map< std::string, boost ::any >
SMRunMode
shared_ptr< smacc:: SmaccStateMachineInfo >
map< std::string, smacc ::smacc::Orthogonal * >
map< std::string, smacc ::smacc::ISmaccComponent * >
mutex
-statusPub_
-nh_
-private_nh_
-globalData_
-runMode_
+info_
-orthogonals_
-plugins_
-m_mutex_
smacc::ISmaccStateMachine
+ ISmaccStateMachine()
+ ~ISmaccStateMachine()
+ Reset()
+ Stop()
+ EStop()
+ notifyOnStateEntry()
+ notifyOnStateExit()
+ getOrthogonal()
+ requiresComponent()
+ requiresComponent()
+ postEvent()
+ getGlobalSMData()
+ setGlobalSMData()
+ registerActionClientRequest()
+ mapBehavior()
+ updateCurrentState()
+ getStateMachineName()
SmaccScheduler *
thread
processor_handle
atomic< bool >
vector< smacc::ISmaccAction Client * >
double
-statusPub_
-nh_
-scheduler_
-signalDetectorThread_
-processorHandle_
-end_
-openRequests_
-loop_rate_hz
-signalDetector_
-smaccStateMachine_
#stateMachine_
#m_mutex_
smacc::SignalDetector
+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ toString()
+ pollingLoop()
+ pollOnce()
+ postEvent()
- finalizeRequest()
- notifyFeedback()
- registerActionClientRequest()
#minPointAngularDistance BackwardThresh_
#minPointDistanceBackwardThresh_
#minPointAngularDistanceForward Thresh_
#minPointDistanceForwardThresh_
smacc::ISmaccComponent
+ ISmaccComponent()
+ ~ISmaccComponent()
+ init()
+ init()
+ setStateMachine()
+ getName()
+ postEvent()
Path
shared_ptr< realtime _tools::RealtimePublisher < nav_msgs::Path > >
Subscriber
WorkingMode
bool
vector< nav_msgs::Path >
#baseTrajectory_
#robotBasePathPub_
#odomSub_
#workingMode_
#publishMessages
#subscribeToOdometryTopic_
#pathStack_
smacc_odom_tracker ::OdomTracker
+ OdomTracker()
+ init()
+ processOdometryMessage()
+ setWorkingMode()
+ setPublishMessages()
+ pushPath()
+ popPath()
+ clearPath()
+ setStartPoint()
+ getPath()
# rtPublishPaths()
# updateForward()
# updateBackward()