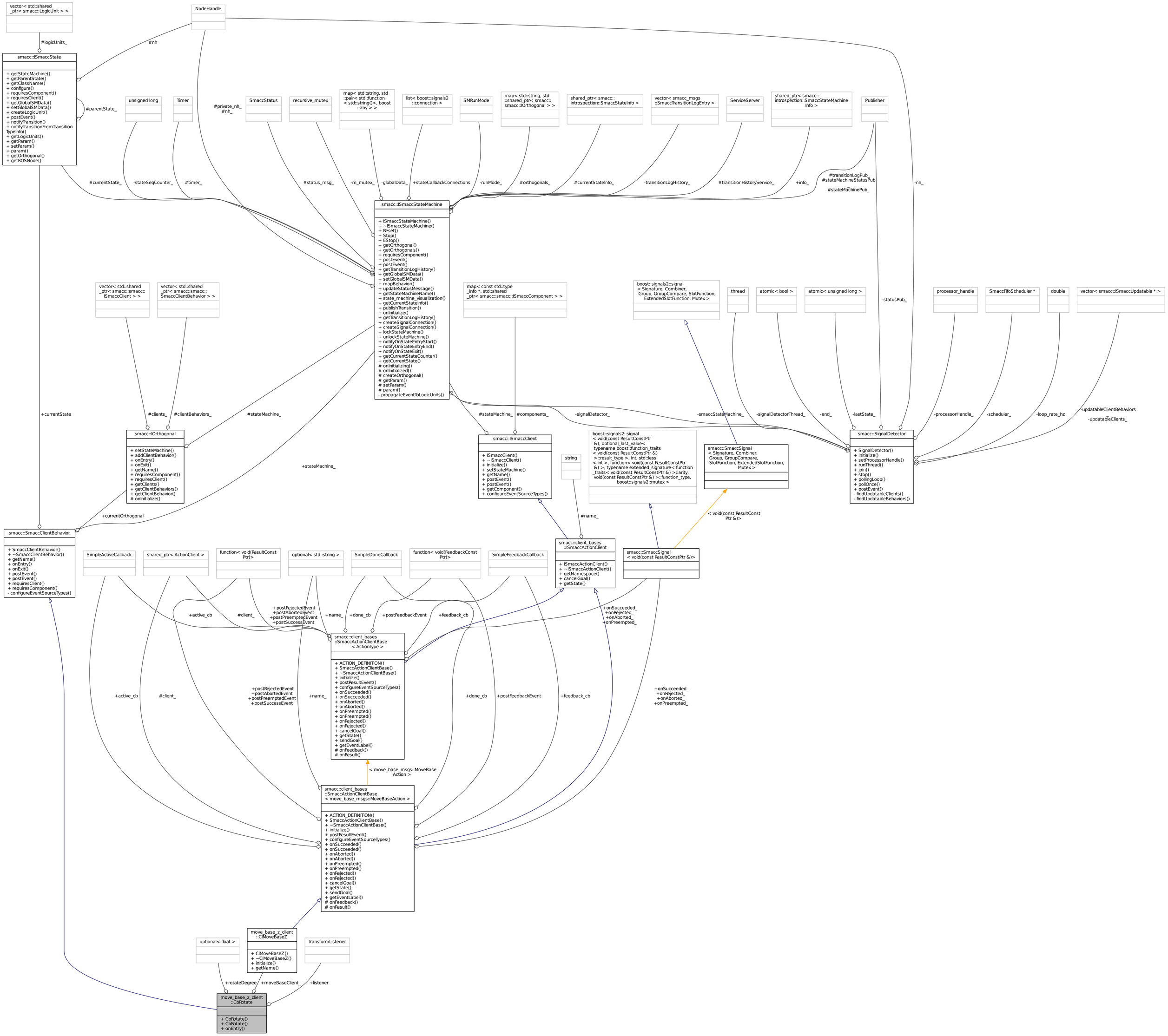
vector< std::shared _ptr< smacc::LogicUnit > >
NodeHandle
#logicUnits_
#nh
smacc::ISmaccState
+ getStateMachine()
+ getParentState()
+ getClassName()
+ configure()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ setGlobalSMData()
+ createLogicUnit()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransition TypeInfo()
+ getLogicUnits()
+ getParam()
+ setParam()
+ param()
+ getOrthogonal()
+ getROSNode()
#parentState_
unsigned long
Timer
#private_nh_
#nh_
SmaccStatus
recursive_mutex
map< std::string, std ::pair< std::function < std::string()>, boost ::any > >
list< boost::signals2 ::connection >
SMRunMode
map< std::string, std ::shared_ptr< smacc:: smacc::IOrthogonal > >
shared_ptr< smacc:: introspection::SmaccStateInfo >
vector< smacc_msgs ::SmaccTransitionLogEntry >
ServiceServer
shared_ptr< smacc:: introspection::SmaccStateMachine Info >
Publisher
#currentState_
-stateSeqCounter_
#timer_
#status_msg_
-m_mutex_
-globalData_
+stateCallbackConnections
-runMode_
#orthogonals_
#currentStateInfo_
-transitionLogHistory_
#transitionHistoryService_
+info_
#transitionLogPub_
#stateMachineStatusPub_
#stateMachinePub_
-nh_
smacc::ISmaccStateMachine
+ ISmaccStateMachine()
+ ~ISmaccStateMachine()
+ Reset()
+ Stop()
+ EStop()
+ getOrthogonal()
+ getOrthogonals()
+ requiresComponent()
+ postEvent()
+ postEvent()
+ getTransitionLogHistory()
+ getGlobalSMData()
+ setGlobalSMData()
+ mapBehavior()
+ updateStatusMessage()
+ getStateMachineName()
+ state_machine_visualization()
+ getCurrentStateInfo()
+ publishTransition()
+ onInitialize()
+ getTransitionLogHistory()
+ createSignalConnection()
+ createSignalConnection()
+ lockStateMachine()
+ unlockStateMachine()
+ notifyOnStateEntryStart()
+ notifyOnStateEntryEnd()
+ notifyOnStateExit()
+ getCurrentStateCounter()
+ getCurrentState()
# onInitializing()
# onInitialized()
# createOrthogonal()
# getParam()
# setParam()
# param()
- propagateEventToLogicUnits()
vector< std::shared _ptr< smacc::smacc:: ISmaccClient > >
vector< std::shared _ptr< smacc::smacc:: SmaccClientBehavior > >
map< const std::type _info *, std::shared _ptr< smacc::smacc::ISmaccComponent > >
boost::signals2::signal < Signature, Combiner, Group, GroupCompare, SlotFunction, ExtendedSlotFunction, Mutex >
thread
atomic< bool >
atomic< unsigned long >
-statusPub_
processor_handle
SmaccFifoScheduler *
double
vector< smacc::ISmaccUpdatable * >
+currentState
#clients_
#clientBehaviors_
#stateMachine_
#stateMachine_
#components_
-signalDetector_
-smaccStateMachine_
-signalDetectorThread_
-end_
-lastState_
-processorHandle_
-scheduler_
-loop_rate_hz
-updatableClientBehaviors
-updatableClients_
smacc::IOrthogonal
+ setStateMachine()
+ addClientBehavior()
+ onEntry()
+ onExit()
+ getName()
+ requiresComponent()
+ requiresClient()
+ getClients()
+ getClientBehaviors()
+ getClientBehavior()
# onInitialize()
smacc::ISmaccClient
+ ISmaccClient()
+ ~ISmaccClient()
+ initialize()
+ setStateMachine()
+ getName()
+ postEvent()
+ postEvent()
+ getComponent()
+ configureEventSourceTypes()
string
boost::signals2::signal < void(const ResultConstPtr &), optional_last_value< typename boost::function_traits < void(const ResultConstPtr &) >::result_type >, int, std::less < int >, function< void(const ResultConstPtr &) >, typename extended_signature< function _traits< void(const ResultConstPtr &) >::arity, void(const ResultConstPtr &) >::function_type, boost::signals2::mutex >
smacc::SmaccSignal < Signature, Combiner, Group, GroupCompare, SlotFunction, ExtendedSlotFunction, Mutex >
smacc::SignalDetector
+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ pollOnce()
+ postEvent()
- findUpdatableClients()
- findUpdatableBehaviors()
+stateMachine_
+currentOrthogonal
#name_
< void(const ResultConst Ptr &)>
smacc::SmaccClientBehavior
+ SmaccClientBehavior()
+ ~SmaccClientBehavior()
+ getName()
+ onEntry()
+ onExit()
+ postEvent()
+ postEvent()
+ requiresClient()
+ requiresComponent()
- configureEventSourceTypes()
smacc::client_bases ::ISmaccActionClient
+ ISmaccActionClient()
+ ~ISmaccActionClient()
+ getNamespace()
+ cancelGoal()
+ getState()
smacc::SmaccSignal < void(const ResultConstPtr &)>
SimpleActiveCallback
shared_ptr< ActionClient >
function< void(ResultConst Ptr)>
optional< std::string >
SimpleDoneCallback
function< void(FeedbackConst Ptr)>
SimpleFeedbackCallback
+active_cb
#client_
+postRejectedEvent
+postAbortedEvent
+postPreemptedEvent
+postSuccessEvent
+name_
+done_cb
+postFeedbackEvent
+feedback_cb
+onSucceeded_
+onRejected_
+onAborted_
+onPreempted_
smacc::client_bases ::SmaccActionClientBase < ActionType >
+ ACTION_DEFINITION()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ initialize()
+ postResultEvent()
+ configureEventSourceTypes()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onPreempted()
+ onPreempted()
+ onRejected()
+ onRejected()
+ cancelGoal()
+ getState()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
< move_base_msgs::MoveBase Action >
smacc::client_bases ::SmaccActionClientBase < move_base_msgs::MoveBaseAction >
+ ACTION_DEFINITION()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ initialize()
+ postResultEvent()
+ configureEventSourceTypes()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onPreempted()
+ onPreempted()
+ onRejected()
+ onRejected()
+ cancelGoal()
+ getState()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
optional< float >
move_base_z_client ::ClMoveBaseZ
+ ClMoveBaseZ()
+ ~ClMoveBaseZ()
+ initialize()
+ getName()
TransformListener
+rotateDegree
+moveBaseClient_
+listener
move_base_z_client ::CbRotate
+ CbRotate()
+ CbRotate()
+ onEntry()