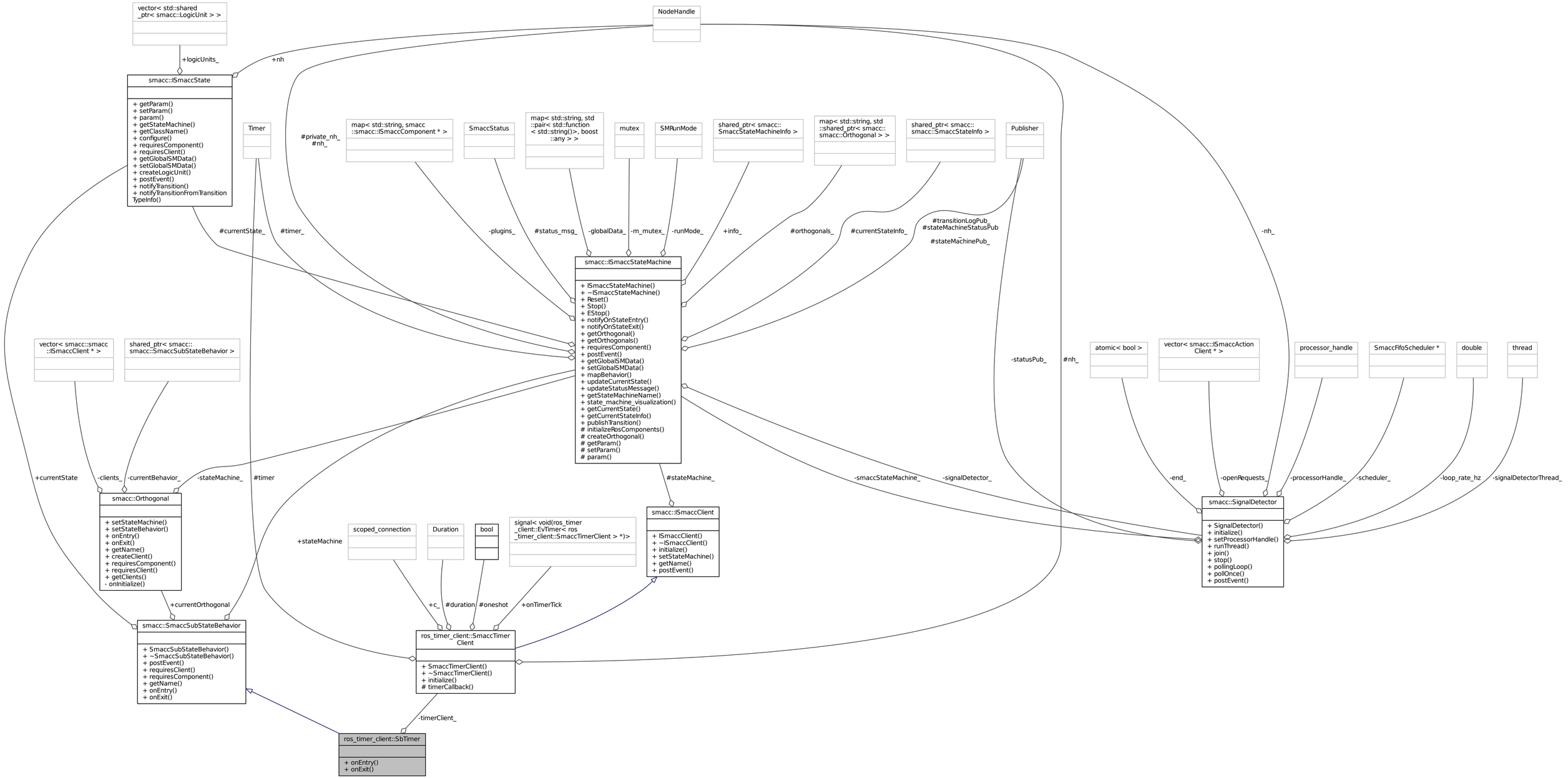
vector< std::shared _ptr< smacc::LogicUnit > >
NodeHandle
+logicUnits_
+nh
smacc::ISmaccState
+ getParam()
+ setParam()
+ param()
+ getStateMachine()
+ getClassName()
+ configure()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ setGlobalSMData()
+ createLogicUnit()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransition TypeInfo()
Timer
#private_nh_
#nh_
map< std::string, smacc ::smacc::ISmaccComponent * >
SmaccStatus
map< std::string, std ::pair< std::function < std::string()>, boost ::any > >
mutex
SMRunMode
shared_ptr< smacc:: SmaccStateMachineInfo >
map< std::string, std ::shared_ptr< smacc:: smacc::Orthogonal > >
shared_ptr< smacc:: smacc::SmaccStateInfo >
Publisher
#currentState_
#timer_
-plugins_
#status_msg_
-globalData_
-m_mutex_
-runMode_
+info_
#orthogonals_
#currentStateInfo_
#transitionLogPub_
#stateMachineStatusPub_
#stateMachinePub_
-nh_
smacc::ISmaccStateMachine
+ ISmaccStateMachine()
+ ~ISmaccStateMachine()
+ Reset()
+ Stop()
+ EStop()
+ notifyOnStateEntry()
+ notifyOnStateExit()
+ getOrthogonal()
+ getOrthogonals()
+ requiresComponent()
+ postEvent()
+ getGlobalSMData()
+ setGlobalSMData()
+ mapBehavior()
+ updateCurrentState()
+ updateStatusMessage()
+ getStateMachineName()
+ state_machine_visualization()
+ getCurrentState()
+ getCurrentStateInfo()
+ publishTransition()
# initializeRosComponents()
# createOrthogonal()
# getParam()
# setParam()
# param()
vector< smacc::smacc ::ISmaccClient * >
shared_ptr< smacc:: smacc::SmaccSubStateBehavior >
-statusPub_
#nh_
atomic< bool >
vector< smacc::ISmaccAction Client * >
processor_handle
SmaccFifoScheduler *
double
thread
+currentState
-clients_
-currentBehavior_
-stateMachine_
#timer
#stateMachine_
-smaccStateMachine_
-signalDetector_
-end_
-openRequests_
-processorHandle_
-scheduler_
-loop_rate_hz
-signalDetectorThread_
smacc::Orthogonal
+ setStateMachine()
+ setStateBehavior()
+ onEntry()
+ onExit()
+ getName()
+ createClient()
+ requiresComponent()
+ requiresClient()
+ getClients()
- onInitialize()
smacc::ISmaccClient
+ ISmaccClient()
+ ~ISmaccClient()
+ initialize()
+ setStateMachine()
+ getName()
+ postEvent()
smacc::SignalDetector
+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ pollOnce()
+ postEvent()
+stateMachine
scoped_connection
Duration
bool
signal< void(ros_timer _client::EvTimer< ros _timer_client::SmaccTimerClient > *)>
+currentOrthogonal
+c_
#duration
#oneshot
+onTimerTick
smacc::SmaccSubStateBehavior
+ SmaccSubStateBehavior()
+ ~SmaccSubStateBehavior()
+ postEvent()
+ requiresClient()
+ requiresComponent()
+ getName()
+ onEntry()
+ onExit()
ros_timer_client::SmaccTimer Client
+ SmaccTimerClient()
+ ~SmaccTimerClient()
+ initialize()
# timerCallback()
-timerClient_
ros_timer_client::SbTimer
+ onEntry()
+ onExit()