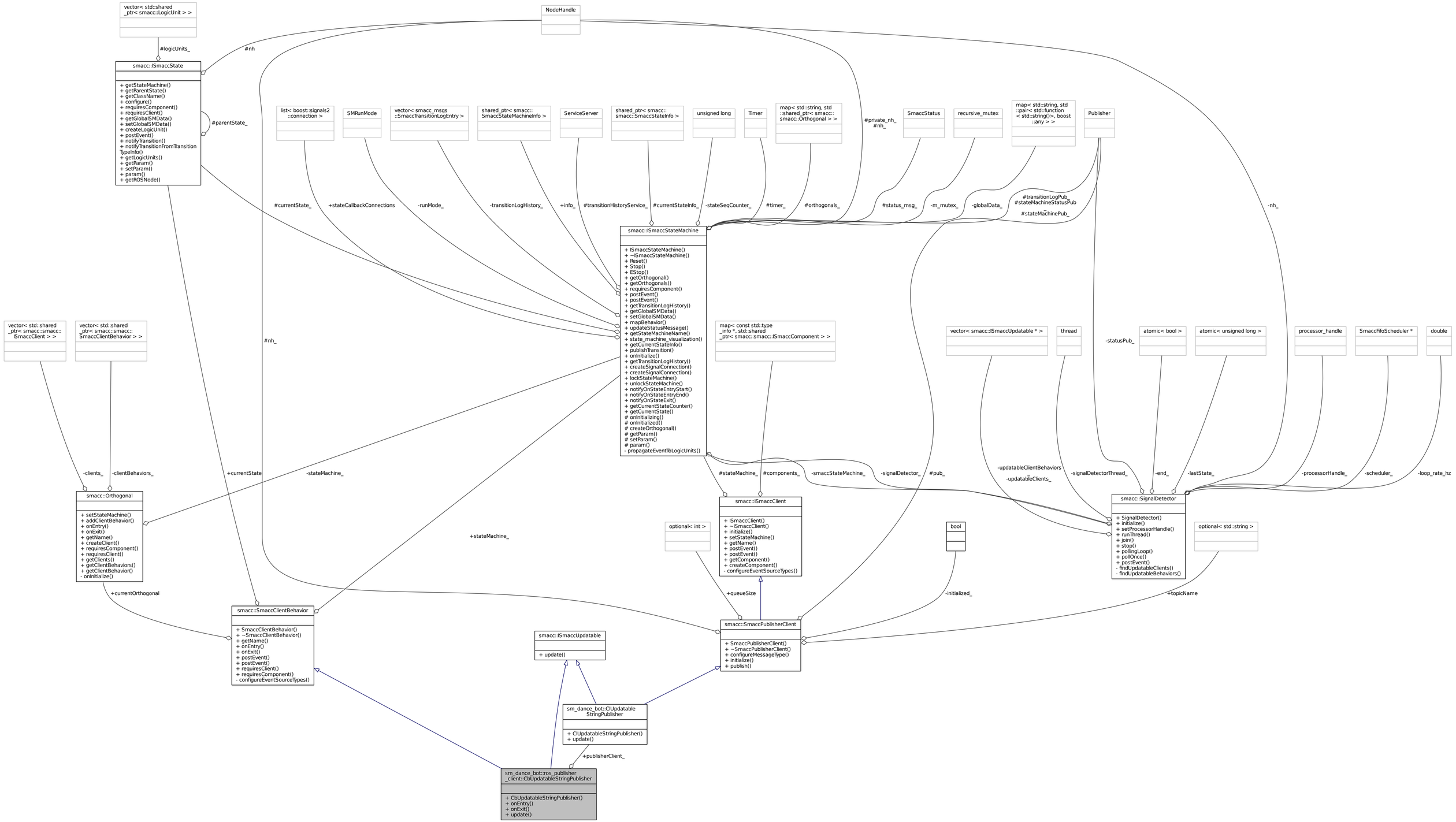
vector< std::shared _ptr< smacc::LogicUnit > >
#logicUnits_
NodeHandle
#nh
smacc::ISmaccState
+ getStateMachine()
+ getParentState()
+ getClassName()
+ configure()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ setGlobalSMData()
+ createLogicUnit()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransition TypeInfo()
+ getLogicUnits()
+ getParam()
+ setParam()
+ param()
+ getROSNode()
#parentState_
list< boost::signals2 ::connection >
SMRunMode
vector< smacc_msgs ::SmaccTransitionLogEntry >
shared_ptr< smacc:: SmaccStateMachineInfo >
ServiceServer
shared_ptr< smacc:: smacc::ISmaccStateInfo >
unsigned long
Timer
map< std::string, std ::shared_ptr< smacc:: smacc::Orthogonal > >
#private_nh_
#nh_
SmaccStatus
recursive_mutex
map< std::string, std ::pair< std::function < std::string()>, boost ::any > >
Publisher
#currentState_
+stateCallbackConnections
-runMode_
-transitionLogHistory_
+info_
#transitionHistoryService_
#currentStateInfo_
-stateSeqCounter_
#timer_
#orthogonals_
#status_msg_
-m_mutex_
-globalData_
#transitionLogPub_
#stateMachinePub
#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 > >
#nh_
map< const std::type _info *, std::shared _ptr< smacc::smacc::ISmaccComponent > >
vector< smacc::ISmaccUpdatable * >
thread
-statusPub_
atomic< bool >
atomic< unsigned long >
processor_handle
SmaccFifoScheduler *
double
-clients_
-clientBehaviors_
+currentState
-stateMachine_
#stateMachine_
#components_
-smaccStateMachine_
-signalDetector_
#pub_
-updatableClientBehaviors
-updatableClients_
-signalDetectorThread_
-end_
-lastState_
-processorHandle_
-scheduler_
-loop_rate_hz
smacc::Orthogonal
+ setStateMachine()
+ addClientBehavior()
+ onEntry()
+ onExit()
+ getName()
+ createClient()
+ requiresComponent()
+ requiresClient()
+ getClients()
+ getClientBehaviors()
+ getClientBehavior()
- onInitialize()
+stateMachine_
smacc::ISmaccClient
+ ISmaccClient()
+ ~ISmaccClient()
+ initialize()
+ setStateMachine()
+ getName()
+ postEvent()
+ postEvent()
+ getComponent()
+ createComponent()
- configureEventSourceTypes()
optional< int >
bool
smacc::SignalDetector
+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ pollOnce()
+ postEvent()
- findUpdatableClients()
- findUpdatableBehaviors()
optional< std::string >
+currentOrthogonal
+queueSize
-initialized_
+topicName
smacc::SmaccClientBehavior
+ SmaccClientBehavior()
+ ~SmaccClientBehavior()
+ getName()
+ onEntry()
+ onExit()
+ postEvent()
+ postEvent()
+ requiresClient()
+ requiresComponent()
- configureEventSourceTypes()
smacc::SmaccPublisherClient
+ SmaccPublisherClient()
+ ~SmaccPublisherClient()
+ configureMessageType()
+ initialize()
+ publish()
smacc::ISmaccUpdatable
+ update()
sm_dance_bot::ClUpdatable StringPublisher
+ ClUpdatableStringPublisher()
+ update()
+publisherClient_
sm_dance_bot::ros_publisher _client::CbUpdatableStringPublisher
+ CbUpdatableStringPublisher()
+ onEntry()
+ onExit()
+ update()