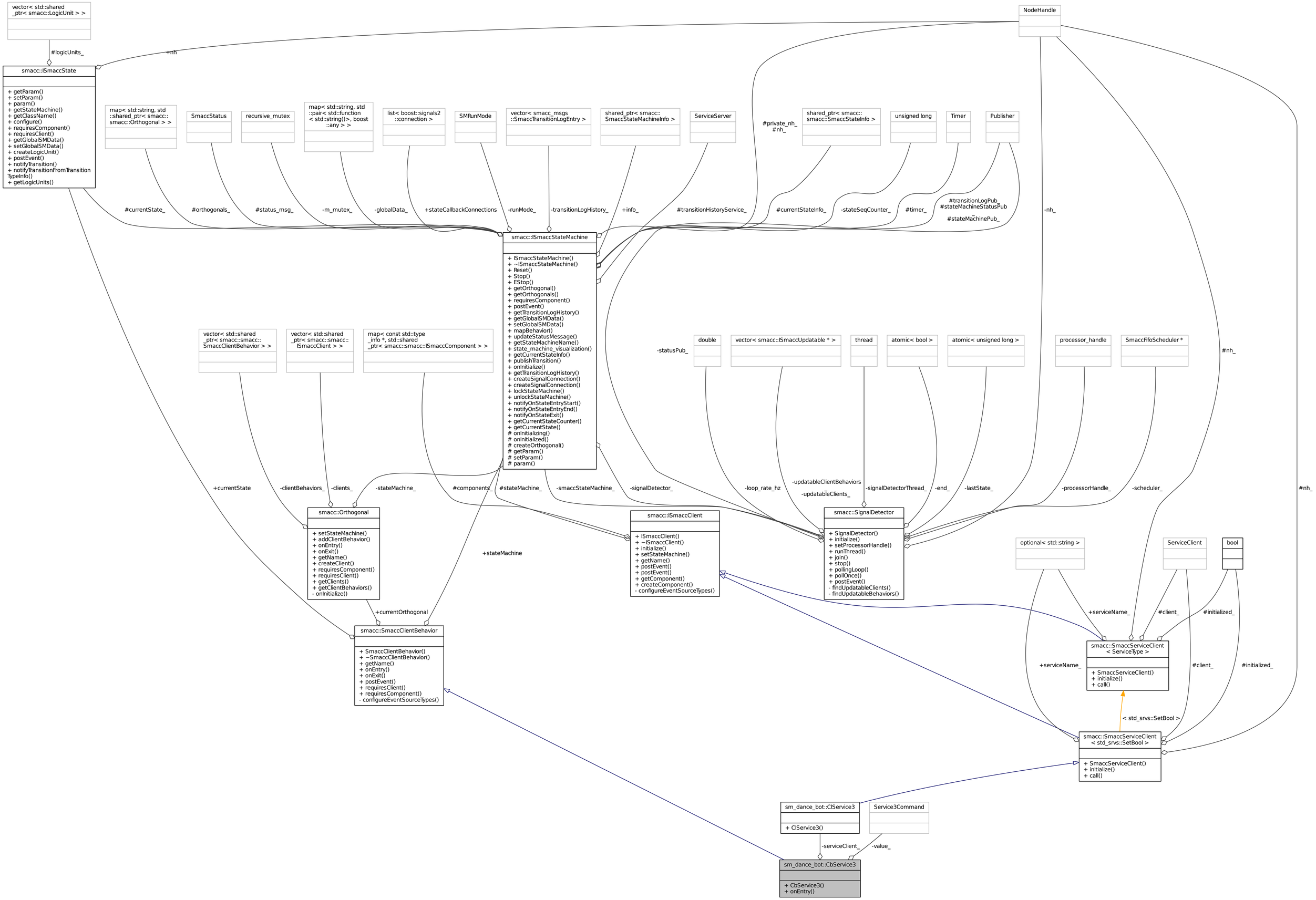
vector< std::shared _ptr< smacc::LogicUnit > >
#logicUnits_
+nh
NodeHandle
smacc::ISmaccState
+ getParam()
+ setParam()
+ param()
+ getStateMachine()
+ getClassName()
+ configure()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ setGlobalSMData()
+ createLogicUnit()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransition TypeInfo()
+ getLogicUnits()
map< std::string, std ::shared_ptr< smacc:: smacc::Orthogonal > >
SmaccStatus
recursive_mutex
map< std::string, std ::pair< std::function < std::string()>, boost ::any > >
list< boost::signals2 ::connection >
SMRunMode
vector< smacc_msgs ::SmaccTransitionLogEntry >
shared_ptr< smacc:: SmaccStateMachineInfo >
ServiceServer
#private_nh_
#nh_
shared_ptr< smacc:: smacc::SmaccStateInfo >
unsigned long
Timer
Publisher
#currentState_
#orthogonals_
#status_msg_
-m_mutex_
-globalData_
+stateCallbackConnections
-runMode_
-transitionLogHistory_
+info_
#transitionHistoryService_
#currentStateInfo_
-stateSeqCounter_
#timer_
#transitionLogPub_
#stateMachineStatusPub_
#stateMachinePub_
-nh_
smacc::ISmaccStateMachine
+ ISmaccStateMachine()
+ ~ISmaccStateMachine()
+ Reset()
+ Stop()
+ EStop()
+ getOrthogonal()
+ getOrthogonals()
+ requiresComponent()
+ 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()
vector< std::shared _ptr< smacc::smacc:: SmaccClientBehavior > >
vector< std::shared _ptr< smacc::smacc:: ISmaccClient > >
map< const std::type _info *, std::shared _ptr< smacc::smacc::ISmaccComponent > >
-statusPub_
double
vector< smacc::ISmaccUpdatable * >
thread
atomic< bool >
atomic< unsigned long >
processor_handle
SmaccFifoScheduler *
#nh_
+currentState
-clientBehaviors_
-clients_
-stateMachine_
#components_
#stateMachine_
-smaccStateMachine_
-signalDetector_
-loop_rate_hz
-updatableClientBehaviors
-updatableClients_
-signalDetectorThread_
-end_
-lastState_
-processorHandle_
-scheduler_
#nh_
smacc::Orthogonal
+ setStateMachine()
+ addClientBehavior()
+ onEntry()
+ onExit()
+ getName()
+ createClient()
+ requiresComponent()
+ requiresClient()
+ getClients()
+ getClientBehaviors()
- onInitialize()
smacc::ISmaccClient
+ ISmaccClient()
+ ~ISmaccClient()
+ initialize()
+ setStateMachine()
+ getName()
+ postEvent()
+ postEvent()
+ getComponent()
+ createComponent()
- configureEventSourceTypes()
smacc::SignalDetector
+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ pollOnce()
+ postEvent()
- findUpdatableClients()
- findUpdatableBehaviors()
+stateMachine
optional< std::string >
ServiceClient
bool
+currentOrthogonal
smacc::SmaccClientBehavior
+ SmaccClientBehavior()
+ ~SmaccClientBehavior()
+ getName()
+ onEntry()
+ onExit()
+ postEvent()
+ requiresClient()
+ requiresComponent()
- configureEventSourceTypes()
+serviceName_
#client_
#initialized_
smacc::SmaccServiceClient < ServiceType >
+ SmaccServiceClient()
+ initialize()
+ call()
+serviceName_
#client_
#initialized_
< std_srvs::SetBool >
smacc::SmaccServiceClient < std_srvs::SetBool >
+ SmaccServiceClient()
+ initialize()
+ call()
sm_dance_bot::ClService3
+ ClService3()
Service3Command
-serviceClient_
-value_
sm_dance_bot::CbService3
+ CbService3()
+ onEntry()