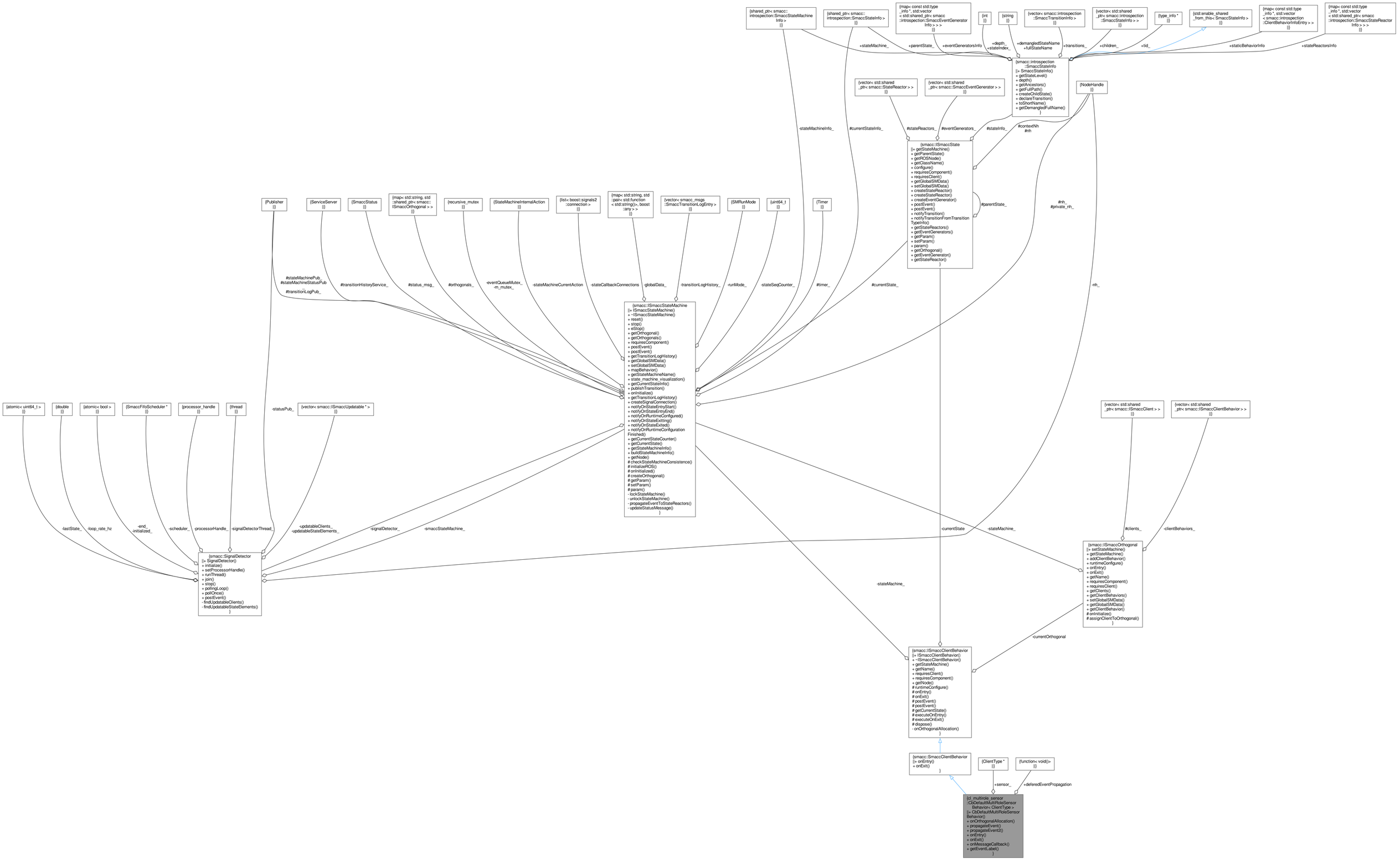
{shared_ptr< smacc::introspection::SmaccStateMachineInfo > }
{shared_ptr< smacc::introspection::SmaccStateInfo > }
{map< const std::type_info *, std::vector< std::shared_ptr< smacc::introspection::SmaccEventGenerator Info > > > }
{int }
{string }
{vector< smacc::introspection::SmaccTransitionInfo > }
{vector< std::shared_ptr< smacc::introspection::SmaccStateInfo > > }
{type_info * }
{std::enable_shared_from_this< SmaccStateInfo > }
{map< const std::type_info *, std::vector< smacc::introspection::ClientBehaviorInfoEntry > > }
{map< const std::type_info *, std::vector< std::shared_ptr< smacc::introspection::SmaccStateReactor Info > > > }
+stateMachine_
+parentState_
+eventGeneratorsInfo
+depth_
+stateIndex_
+demangledStateName
+fullStateName
+transitions_
+children_
+tid_
+staticBehaviorInfo
+staticReactorsInfo
{smacc::introspection::SmaccStateInfo
|+ SmaccStateInfo()
+ getStateLevel()
+ depth()
+ getAncestors()
+ getFullPath()
+ createChildState()
+ declareTransition()
+ toShortName()
+ getDemangledFullName()
}
{vector< std::shared_ptr< smacc::StateReactor > > }
{vector< std::shared_ptr< smacc::SmaccEventGenerator > > }
{NodeHandle }
-stateMachineInfo_
#currentStateInfo_
#stateReactors_
#eventGenerators_
#stateInfo_
#contextNh
#nh
{smacc::ISmaccState
|+ getStateMachine()
+ getParentState()
+ getROSNode()
+ getClassName()
+ configure()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ setGlobalSMData()
+ createStateReactor()
+ createStateReactor()
+ createEventGenerator()
+ postEvent()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransition TypeInfo()
+ getStateReactors()
+ getEventGenerators()
+ getParam()
+ setParam()
+ param()
+ getOrthogonal()
+ getEventGenerator()
+ getStateReactor()
}
#parentState_
#nh_
#private_nh_
{Publisher }
{ServiceServer }
{SmaccStatus }
{map< std::string, std::shared_ptr< smacc::ISmaccOrthogonal > > }
{recursive_mutex }
{StateMachineInternalAction }
{list< boost::signals2::connection > }
{map< std::string, std::pair< std::function< std::string()>, boost::any > > }
{vector< smacc_msgs::SmaccTransitionLogEntry > }
{SMRunMode }
{uint64_t }
{Timer }
#stateMachinePub_
#stateMachineStatusPub
#transitionLogPub_
#transitionHistoryService_
#status_msg_
#orthogonals_
-eventQueueMutex_
-m_mutex_
-stateMachineCurrentAction
-stateCallbackConnections
-globalData_
-transitionLogHistory_
-runMode_
-stateSeqCounter_
#timer_
#currentState_
-nh_
{smacc::ISmaccStateMachine
|+ ISmaccStateMachine()
+ ~ISmaccStateMachine()
+ reset()
+ stop()
+ eStop()
+ getOrthogonal()
+ getOrthogonals()
+ requiresComponent()
+ postEvent()
+ postEvent()
+ getTransitionLogHistory()
+ getGlobalSMData()
+ setGlobalSMData()
+ mapBehavior()
+ getStateMachineName()
+ state_machine_visualization()
+ getCurrentStateInfo()
+ publishTransition()
+ onInitialize()
+ getTransitionLogHistory()
+ createSignalConnection()
+ notifyOnStateEntryStart()
+ notifyOnRuntimeConfigured()
+ notifyOnStateExitting()
+ notifyOnStateExited()
+ notifyOnRuntimeConfiguration Finished()
+ getCurrentStateCounter()
+ getCurrentState()
+ getStateMachineInfo()
+ buildStateMachineInfo()
+ getNode()
# checkStateMachineConsistence()
# initializeROS()
# onInitialized()
# createOrthogonal()
# getParam()
# setParam()
# param()
- lockStateMachine()
- unlockStateMachine()
- propagateEventToStateReactors()
- updateStatusMessage()
}
{atomic< uint64_t > }
{double }
{atomic< bool > }
{SmaccFifoScheduler * }
{processor_handle }
{thread }
-statusPub_
{vector< smacc::ISmaccUpdatable * > }
{vector< std::shared_ptr< smacc::ISmaccClient > > }
{vector< std::shared_ptr< smacc::ISmaccClientBehavior > > }
-lastState_
-loop_rate_hz
-end_
-initialized_
-scheduler_
-processorHandle_
-signalDetectorThread_
-updatableClients_
-updatableStateElements_
-signalDetector_
-smaccStateMachine_
-currentState
-stateMachine_
#clients_
-clientBehaviors_
{smacc::SignalDetector
|+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ pollOnce()
+ postEvent()
- findUpdatableClients()
- findUpdatableStateElements()
}
{smacc::ISmaccOrthogonal
|+ setStateMachine()
+ getStateMachine()
+ addClientBehavior()
+ runtimeConfigure()
+ onEntry()
+ onExit()
+ getName()
+ requiresComponent()
+ requiresClient()
+ getClients()
+ getClientBehaviors()
+ setGlobalSMData()
+ getGlobalSMData()
+ getClientBehavior()
# onInitialize()
# assignClientToOrthogonal()
}
-currentOrthogonal
{smacc::ISmaccClientBehavior
|+ ISmaccClientBehavior()
+ ~ISmaccClientBehavior()
+ getStateMachine()
+ getName()
+ requiresClient()
+ requiresComponent()
+ getNode()
# runtimeConfigure()
# onEntry()
# onExit()
# postEvent()
# postEvent()
# getCurrentState()
# executeOnEntry()
# executeOnExit()
# dispose()
- onOrthogonalAllocation()
}
{smacc::SmaccClientBehavior
|+ onEntry()
+ onExit()
}
{ClientType * }
{function< void()> }
+sensor_
+deferredEventPropagation
{cl_multirole_sensor
::CbDefaultMultiRoleSensor Behavior< ClientType >
|+ CbDefaultMultiRoleSensor Behavior()
+ onOrthogonalAllocation()
+ propagateEvent()
+ propagateEvent2()
+ onEntry()
+ onExit()
+ onMessageCallback()
+ getEventLabel()
}