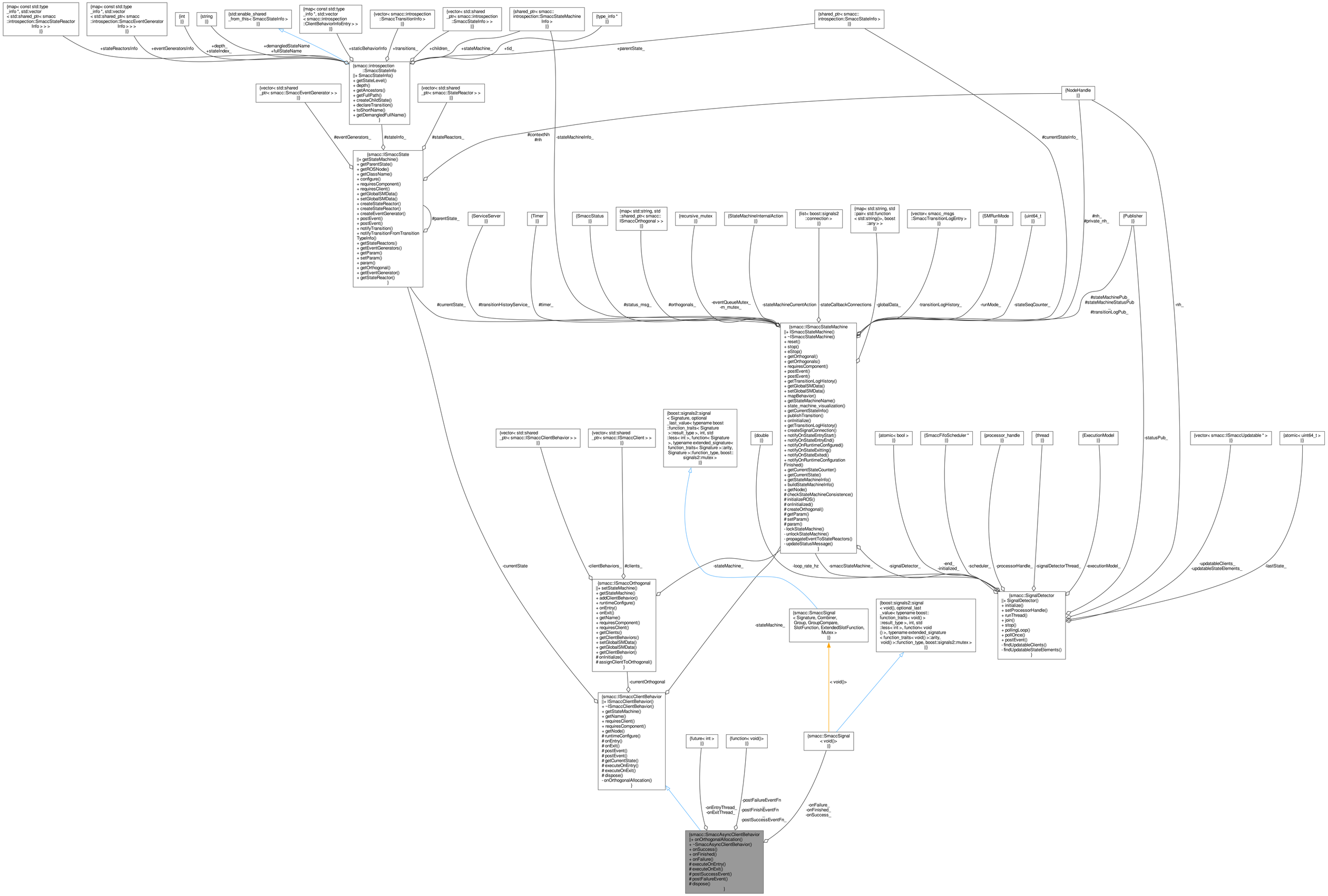
{map< const std::type_info *, std::vector< std::shared_ptr< smacc::introspection::SmaccStateReactorInfo > > >}
{map< const std::type_info *, std::vector< std::shared_ptr< smacc::introspection::SmaccEventGeneratorInfo > > >}
{int}
{string}
{std::enable_shared_from_this< SmaccStateInfo >}
{map< const std::type_info *, std::vector< smacc::introspection::ClientBehaviorInfoEntry > >}
{vector< smacc::introspection::SmaccTransitionInfo >}
{vector< std::shared_ptr< smacc::introspection::SmaccStateInfo > >}
{shared_ptr< smacc::introspection::SmaccStateMachineInfo >}
{type_info *}
{shared_ptr< smacc::introspection::SmaccStateInfo >}
+stateReactorsInfo
+eventGeneratorsInfo
+depth_
+stateIndex_
+demangledStateName
+fullStateName
+staticBehaviorInfo
+transitions_
+children_
+stateMachine_
+tid_
+parentState_
{smacc::introspection::SmaccStateInfo
|+ SmaccStateInfo()
+ getStateLevel()
+ depth()
+ getAncestors()
+ getFullPath()
+ createChildState()
+ declareTransition()
+ toShortName()
+ getDemangledFullName()
}
{vector< std::shared_ptr< smacc::SmaccEventGenerator > >}
{vector< std::shared_ptr< smacc::StateReactor > >}
{NodeHandle}
#eventGenerators_
#stateInfo_
#stateReactors_
#contextNh
#nh
-stateMachineInfo_
#currentStateInfo_
{smacc::ISmaccState
|+ getStateMachine()
+ getParentState()
+ getROSNode()
+ getClassName()
+ configure()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ setGlobalSMData()
+ createStateReactor()
+ createStateReactor()
+ createEventGenerator()
+ postEvent()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransitionTypeInfo()
+ getStateReactors()
+ getEventGenerators()
+ getParam()
+ setParam()
+ param()
+ getOrthogonal()
+ getEventGenerator()
+ getStateReactor()
}
#parentState_
{ServiceServer}
{Timer}
{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}
#nh_
#private_nh_
{Publisher}
#currentState_
#transitionHistoryService_
#timer_
#status_msg_
#orthogonals_
-eventQueueMutex_
-m_mutex_
-stateMachineCurrentAction
-stateCallbackConnections
-globalData_
-transitionLogHistory_
-runMode_
-stateSeqCounter_
#stateMachinePub_
#stateMachineStatusPub
#transitionLogPub_
-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()
+ notifyOnStateEntryEnd()
+ notifyOnRuntimeConfigured()
+ notifyOnStateExitting()
+ notifyOnStateExited()
+ notifyOnRuntimeConfigurationFinished()
+ getCurrentStateCounter()
+ getCurrentState()
+ getStateMachineInfo()
+ buildStateMachineInfo()
+ getNode()
# checkStateMachineConsistence()
# initializeROS()
# onInitialized()
# createOrthogonal()
# getParam()
# setParam()
# param()
- lockStateMachine()
- unlockStateMachine()
- propagateEventToStateReactors()
- updateStatusMessage()
}
{vector< std::shared_ptr< smacc::ISmaccClientBehavior > >}
{vector< std::shared_ptr< smacc::ISmaccClient > >}
{boost::signals2::signal< Signature, optional_last_value< typename boost::function_traits< Signature >::result_type >, int, std::less< int >, function< Signature >, typename extended_signature< function_traits< Signature >::arity, Signature >::function_type, boost::signals2::mutex >}
{double}
{atomic< bool >}
{SmaccFifoScheduler *}
{processor_handle}
{thread}
{ExecutionModel}
-statusPub_
{vector< smacc::ISmaccUpdatable * >}
{atomic< uint64_t >}
-currentState
-clientBehaviors_
#clients_
-stateMachine_
-loop_rate_hz
-smaccStateMachine_
-signalDetector_
-end_
-initialized_
-scheduler_
-processorHandle_
-signalDetectorThread_
-executionModel_
-updatableClients_
-updatableStateElements_
-lastState_
{smacc::ISmaccOrthogonal
|+ setStateMachine()
+ getStateMachine()
+ addClientBehavior()
+ runtimeConfigure()
+ onEntry()
+ onExit()
+ getName()
+ requiresComponent()
+ requiresClient()
+ getClients()
+ getClientBehaviors()
+ setGlobalSMData()
+ getGlobalSMData()
+ getClientBehavior()
# onInitialize()
# assignClientToOrthogonal()
}
{smacc::SignalDetector
|+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ pollOnce()
+ postEvent()
- findUpdatableClients()
- findUpdatableStateElements()
}
-stateMachine_
{smacc::SmaccSignal
< Signature, Combiner, Group, GroupCompare, SlotFunction, ExtendedSlotFunction, Mutex >
|||}
{boost::signals2::signal
< void(), optional_last_value< typename boost::function_traits< void() >::result_type >, int, std::less< int >, function< void () >, typename extended_signature
< function_traits< void() >::arity, void() >::function_type, boost::signals2::mutex >
|||}
-currentOrthogonal
< void()>
{smacc::ISmaccClientBehavior
|+ ISmaccClientBehavior()
+ ~ISmaccClientBehavior()
+ getStateMachine()
+ getName()
+ requiresClient()
+ requiresComponent()
+ getNode()
# runtimeConfigure()
# onEntry()
# onExit()
# postEvent()
# postEvent()
# getCurrentState()
# executeOnEntry()
# executeOnExit()
# dispose()
- onOrthogonalAllocation()
}
{future< int >}
{function< void()>}
{smacc::SmaccSignal
< void()>}
-onEntryThread_
-onExitThread_
-postFailureEventFn
-postFinishEventFn
-postSuccessEventFn_
-onFailure_
-onFinished_
-onSuccess_
{smacc::SmaccAsyncClientBehavior
|+ onOrthogonalAllocation()
+ ~SmaccAsyncClientBehavior()
+ onSuccess()
+ onFinished()
+ onFailure()
# executeOnEntry()
# executeOnExit()
# postSuccessEvent()
# postFailureEvent()
# dispose()
}