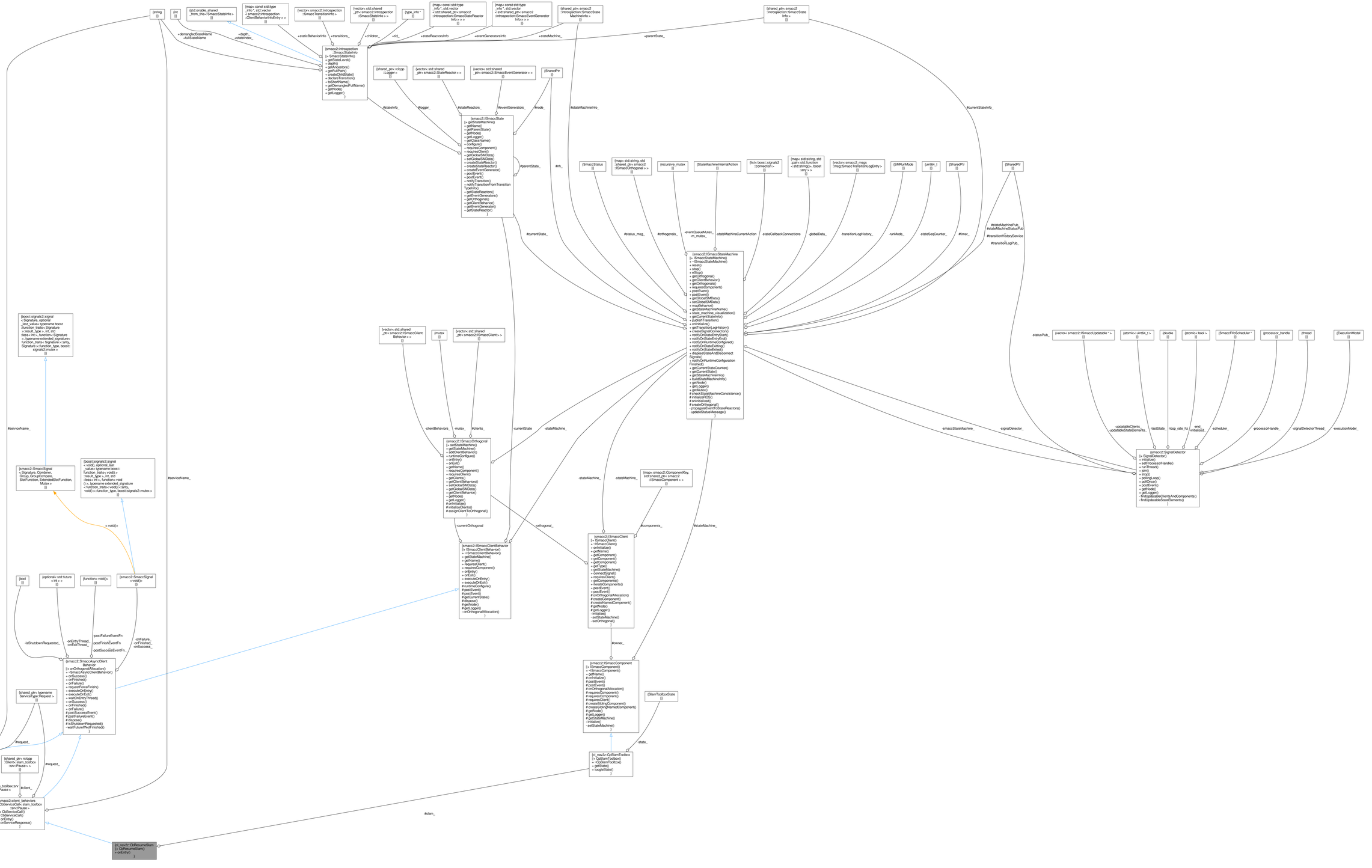
[string]]
[int]]
[std::enable_shared _from_this< SmaccStateInfo >]]
[map< const std::type _info *, std::vector < smacc2::introspection ::ClientBehaviorInfoEntry > >]]
[vector< smacc2::introspection ::SmaccTransitionInfo >]]
[vector< std::shared _ptr< smacc2::introspection ::SmaccStateInfo > >]]
[type_info *]]
[map< const std::type _info *, std::vector < std::shared_ptr< smacc2 ::introspection::SmaccStateReactor Info > > >]]
[map< const std::type _info *, std::vector < std::shared_ptr< smacc2 ::introspection::SmaccEventGenerator Info> > >]]
[shared_ptr< smacc2 ::introspection::SmaccState MachineInfo >]]
[shared_ptr< smacc2 ::introspection::SmaccState Info >]]
+demangledStateName
+fullStateName
+depth_
+stateIndex_
+staticBehaviorInfo
+transitions_
+children_
+id_
+stateReactorsInfo
+eventGeneratorsInfo
+stateMachine_
+parentState_
[smacc2::introspection ::SmaccStateInfo]- SmaccStateInfo()
+ getLevel(level)
+ depth()
+ getAncestors()
+ getFullPath()
+ createChildState()
+ declareTransition()
+ toShortName()
+ getDemangledFullName()
+ getNode()
+ getLogger()
[shared_ptr< rclcpp ::Logger >]]
[vector< std::shared _ptr< smacc2::StateReactor > >]]
[vector< std::shared _ptr< smacc2::SmaccEventGenerator > >]]
[SharedPtr]]
#stateInfo_
#logger_
#stateReactors_
#eventGenerators_
#node_
#stateMachineInfo_
#currentStateInfo_
[smacc2::ISmaccState]+ getStateMachine()
+ getName()
+ getParentState()
+ getNode()
+ getLogger()
+ getClassName()
+ getClass()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ setGlobalSMData()
+ createStateReactor()
+ createStateReactor()
+ createEventGenerator()
+ postEvent()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransition TypeInfo()
+ getStateReactors()
+ getEventGenerators()
+ getOrthogonals()
+ getOrthogonal()
+ getEventGenerator()
+ getStateReactor()
#parentState_
#currentState_
#nh_
[SmaccStatus]]
[map< std::string, std ::shared_ptr< smacc2 ::ISmaccOrthogonal > >]]
[recursive_mutex]]
[StateMachineInternalAction]]
[list< boost::signals2 ::connection >]]
[map< std::string, std ::pair< std::function < std::string()>, boost ::any > >]]
[vector< smacc2_msgs ::msg::SmaccTransitionLogEntry >]]
[SMRUnMode]]
[uint64_t]]
[SharedPtr]]
[SharedPtr]]
#status_msg_
#orthogonals_
#eventQueueMutex_
#m_mutex_
-stateMachineCurrentAction
-stateCallbackConnections
-globalData_
-transitionLogHistory_
-unMode_
-stateSeqCounter_
#timer_
#stateMachinePub_
#stateMachineStatusPub_
#transitionHistoryService
#transitionLogPub_
[smacc2::ISmaccStateMachine]+ ISmaccStateMachine()
+ ~ISmaccStateMachine()
+ reset()
+ stop()
+ eStop()
+ getOrthogonal()
+ getClientBehavior()
+ getOrthogonals()
+ getOrthogonalComponent()
+ postEvent()
+ postEvent()
+ getGlobalSMData()
+ setGlobalSMData()
+ mapBehavior()
+ getStateMachineName()
+ static_machine_visualization()
+ getCurrentStateInfo()
+ globalInTransition()
+ onInitialize()
+ getTransitionLogHistory()
+ createSignalConnection()
+ notifyOnStateEntryStart()
+ notifyOnStateEntryEnd()
+ notifyOnRuntimeConfigured()
+ notifyOnStateExitStart()
+ notifyOnStateExitEnd()
+ disposeStateAndDisconnect Signals()
+ notifyOnRuntimeConfiguration Finished()
+ getCurrentStateCounter()
+ getStateMachine()
+ getStateMachineInfo()
+ buildStateMachineInfo()
+ getNode()
+ getLogger()
+ getMutex()
# checkStateMachineConsistence()
# initializeROS()
# onInitialize()
# createOrthogonal()
- propagateEventToStateReactors()
- updateStatusMessage()
}
-smaccStateMachine_
-signalDetector_
[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:: signals::mutex >]]
#serviceName_
[smacc2::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 >]]
#serviceName_
< void()>
[bool]]
[optional< std::future < int > >]]
[function< void()>]]
[smacc2::SmaccSignal < void()>]]
-isShutdownRequested_
-onEntryThread_
-onExitThread_
postFailureEventFn
postFinishEventFn
postSuccessEventFn_
-onFailure_
-onFinished_
-onSuccess_
[smacc2::SmaccAsyncClient Behavior]+ onOrthogonalAllocation()
+ ~SmaccAsyncClientBehavior()
+ onSuccess()
+ onFinished()
+ onFailure()
+ waitUntilForceFinish()
+ executeOnEntry()
+ executeOnExit()
+ waitOnEntryThread()
+ onSuccess()
+ onFinished()
+ onFailure()
# postSuccessEvent()
# postFailureEvent()
# dispose()
# isShutdownRequested()
- waitFutureIfNotFinished()
}
[shared_ptr< typename ServiceType::Response >]]
[milliseconds]]
[SharedFuture]]
[shared_ptr< rclcpp ::Client< ServiceType > >]]
[shared_ptr< typename ServiceType::Request >]]
+result_
+pollRate_
+resultFuture_
+client_
#request_
[smacc2::client_behaviors ::CbServiceCall< ServiceType >]+ CbServiceCall()
+ ~CbServiceCall()
+ onEntry()
# onServiceResponse()
}
[shared_ptr< rclcpp ::Client< slam_toolbox ::srv::Pause > >]]
+result_
[SharedFuture]]
+pollRate_
+resultFuture_
#request_
< slam_toolbox::srv ::Pause >
#client_
[smacc2::client_behaviors ::CbServiceCall< slam_toolbox ::srv::Pause >]+ CbServiceCall()
+ ~CbServiceCall()
+ onEntry()
# onServiceResponse()
}
[cl_rev2r::CpResumeSlam]+ CpResumeSlam()
+ onEntry()
}
#slam_
[vector< std::shared _ptr< smacc2::ISmaccClient Behavior > >]]
[mutex]]
[vector< std::shared _ptr< smacc2::ISmaccClient > >]]
-clientBehaviors_
-mutex_
#clients_
[smacc2::ISmaccOrthogonal]+ setStateMachine()
+ addClientBehavior()
+ runtimeConfigure()
+ onEntry()
+ onExit()
+ getName()
+ requiresComponents()
+ requiresClients()
+ getClientBehaviors()
+ getGlobalSMData()
+ setGlobalSMData()
+ getClientBehavior()
+ getNode()
+ getLogger()
# onInitialize()
# initializeClients()
# assignClientToOrthogonal()
}
-currentOrthogonal
-currentState
-stateMachine_
-stateMachine_
-stateMachine_
[map< smacc2::ComponentKey, std::shared_ptr< smacc2 ::ISmaccComponent > >]]
-orthogonal_
#components_
#stateMachine_
[smacc2::ISmaccClientBehavior]+ ISmaccClientBehavior()
+ ~ISmaccClientBehavior()
+ getStateMachine()
+ getClient()
+ requiresClient()
+ requiresComponent()
+ onEntry()
+ onExit()
+ executeOnEntry()
+ executeOnExit()
+ waitOnEntryThread()
# postEvent()
# postEvent()
# getCurrentState()
# dispose()
# getNode()
# getLogger()
- onOrthogonalAllocation()
}
[smacc2::ISmaccClient]+ ISmaccClient()
+ ~ISmaccClient()
+ onInitialize()
+ getName()
+ getComponent()
+ getComponent()
+ getComponent()
+ getComponents()
+ getType()
+ getStateMachine()
+ connectSignal()
+ requiresClients()
+ getComponents()
+ iterateComponents()
+ postEvent()
+ postEvent()
# onOrthogonalAllocation()
# createComponent()
# createSmaccComponent()
# getNode()
# getLogger()
- initialize()
- setStateMachine()
- setOrthogonal()
}
#owner_
[smacc2::ISmaccComponent]+ ISmaccComponent()
+ ~ISmaccComponent()
+ getName()
# onInitialize()
# postEvent()
# postEvent()
# onOrthogonalAllocation()
# requiresComponent()
# requiresComponent()
# requiresClient()
# getOrthogonalComponent()
# createSiblingNamedComponent()
# getNode()
# getLogger()
# getStateMachine()
- initialize()
- setStateMachine()
}
[ISlamToolboxState]]
-state_
[cl_rev2r::CpSlamToolbox]+ CpSlamToolbox()
+ ~CpSlamToolbox()
+ getState()
+ toggleState()
}
-statusPub_
[vector< smacc2::ISmaccUpdatable* >]]
[atomic< uint64_t >]]
[double]]
[atomic< bool >]]
[SmaccFifoScheduler*]]
[processor_handle]]
[thread]]
[ExecutionModel]]
-updatableClients_
-updatableStateElements_
-lastState_
-loop_rate_hz_
-end_initialized_
-scheduler_
-processorHandle_
-signalDetectorThread_
-executionModel_
[smacc2::SignalDetector]+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ pollOnce()
+ getNode()
+ getNode()
+ getLogger()
- findUpdatableClientsAndComponents()
- findUpdatableStateElements()
}