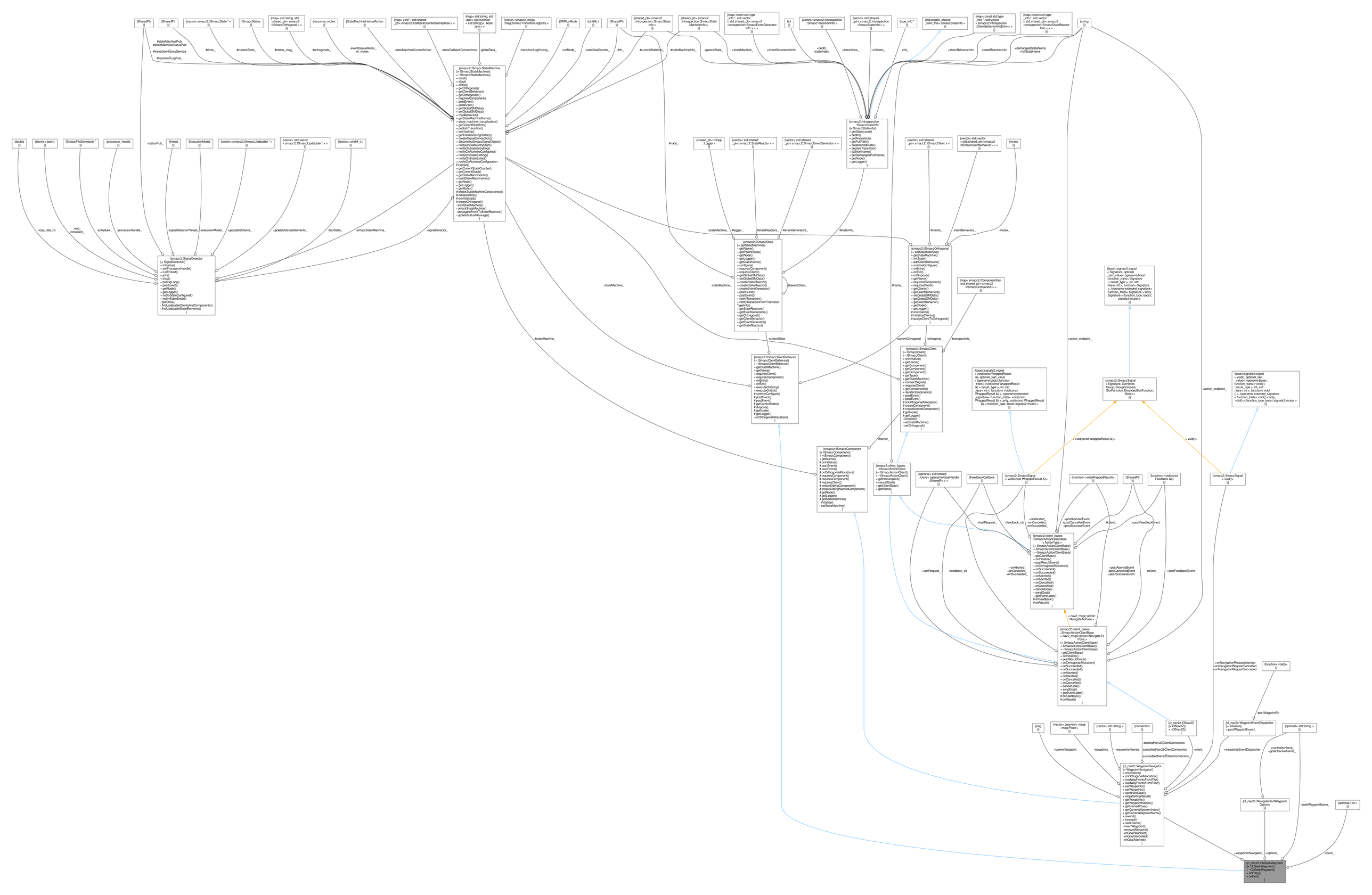

{SharedPtr
||}
{SharedPtr
||}
{vector< smacc2::ISmaccState * >
||}
{SmaccStatus
||}
{map< std::string, std
::shared_ptr< smacc2
::ISmaccOrthogonal > >
||}
{recursive_mutex
||}
{StateMachineInternalAction
||}
{map< void *, std::shared
_ptr< smacc2::CallbackCounterSemaphore > >
||}
{map< std::string, std
::pair< std::function
< std::string()>, boost
::any > >
||}
{vector< smacc2_msgs
::msg::SmaccTransitionLogEntry >
||}
{SMRunMode
||}
{uint64_t
||}
{SharedPtr
||}
{shared_ptr< smacc2
::introspection::SmaccState
Info >
||}
{shared_ptr< smacc2
::introspection::SmaccState
MachineInfo >
||}
{map< const std::type
_info *, std::vector
< std::shared_ptr< smacc2
::introspection::SmaccEventGenerator
Info > > >
||}
{int
||}
{vector< smacc2::introspection
::SmaccTransitionInfo >
||}
{vector< std::shared
_ptr< smacc2::introspection
::SmaccStateInfo > >
||}
{type_info *
||}
{std::enable_shared
_from_this< SmaccStateInfo >
||}
{map< const std::type
_info *, std::vector
< smacc2::introspection
::ClientBehaviorInfoEntry > >
||}
{map< const std::type
_info *, std::vector
< std::shared_ptr< smacc2
::introspection::SmaccStateReactor
Info > > >
||}
{string
||}
#stateMachinePub_
#stateMachineStatusPub
_
#transitionHistoryService
_
#transitionLogPub_
#timer_
#currentState_
#status_msg_
#orthogonals_
-eventQueueMutex_
-m_mutex_
-stateMachineCurrentAction
-stateCallbackConnections
-globalData_
-transitionLogHistory_
-runMode_
-stateSeqCounter_
#nh_
#currentStateInfo_
#stateMachineInfo_
+parentState_
+stateMachine_
+eventGeneratorsInfo
+depth_
+stateIndex_
+transitions_
+children_
+tid_
+staticBehaviorInfo
+stateReactorsInfo
+demangledStateName
+fullStateName
{smacc2::ISmaccStateMachine
||+ ISmaccStateMachine()
+ ~ISmaccStateMachine()
+ reset()
+ stop()
+ eStop()
+ getOrthogonal()
+ getClientBehavior()
+ getOrthogonals()
+ requiresComponent()
+ postEvent()
+ postEvent()
+ getGlobalSMData()
+ setGlobalSMData()
+ mapBehavior()
+ getStateMachineName()
+ state_machine_visualization()
+ getCurrentStateInfo()
+ publishTransition()
+ onInitialize()
+ getTransitionLogHistory()
+ createSignalConnection()
+ disconnectSmaccSignalObject()
+ notifyOnStateEntryStart()
+ notifyOnStateEntryEnd()
+ notifyOnRuntimeConfigured()
+ notifyOnStateExitting()
+ notifyOnStateExited()
+ notifyOnRuntimeConfiguration
Finished()
+ getCurrentStateCounter()
+ getCurrentState()
+ getStateMachineInfo()
+ buildStateMachineInfo()
+ getNode()
+ getLogger()
+ getMutex()
# checkStateMachineConsistence()
# initializeROS()
# onInitialized()
# createOrthogonal()
- lockStateMachine()
- unlockStateMachine()
- propagateEventToStateReactors()
- updateStatusMessage()
}
{smacc2::introspection
::SmaccStateInfo
||+ SmaccStateInfo()
+ getStateLevel()
+ depth()
+ getAncestors()
+ getFullPath()
+ createChildState()
+ declareTransition()
+ toShortName()
+ getDemangledFullName()
+ getNode()
+ getLogger()
}
{double
||}
{atomic< bool >
||}
{SmaccFifoScheduler *
||}
{processor_handle
||}
-statusPub_
{thread
||}
{ExecutionModel
||}
{vector< smacc2::ISmaccUpdatable * >
||}
{vector< std::vector
< smacc2::ISmaccUpdatable * > >
||}
{atomic< uint64_t >
||}
{shared_ptr< rclcpp
::Logger >
||}
{vector< std::shared
_ptr< smacc2::StateReactor > >
||}
{vector< std::shared
_ptr< smacc2::SmaccEventGenerator > >
||}
{vector< std::shared
_ptr< smacc2::ISmaccClient > >
||}
{vector< std::vector
< std::shared_ptr< smacc2
::ISmaccClientBehavior > > >
||}
{mutex
||}
-loop_rate_hz
-end_
-initialized_
-scheduler_
-processorHandle_
-signalDetectorThread_
-executionModel_
-updatableClients_
-updatableStateElements_
-lastState_
-smaccStateMachine_
-signalDetector_
-stateMachine_
#logger_
#stateReactors_
#eventGenerators_
#stateInfo_
#clients_
-clientBehaviors_
-mutex_
#node_
{smacc2::SignalDetector
||+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ postEvent()
+ getNode()
+ getLogger()
+ notifyStateConfigured()
+ notifyStateExited()
- pollOnce()
- findUpdatableClientsAndComponents()
- findUpdatableStateElements()
}
{smacc2::ISmaccState
||+ getStateMachine()
+ getName()
+ getParentState()
+ getNode()
+ getLogger()
+ getClassName()
+ configure()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ setGlobalSMData()
+ createStateReactor()
+ createStateReactor()
+ createEventGenerator()
+ postEvent()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransition
TypeInfo()
+ getStateReactors()
+ getEventGenerators()
+ getOrthogonal()
+ getClientBehavior()
+ getEventGenerator()
+ getStateReactor()
}
#parentState_
-stateMachine_
-stateMachine_
#name_
{smacc2::ISmaccOrthogonal
||+ setStateMachine()
+ getStateMachine()
+ initState()
+ addClientBehavior()
+ runtimeConfigure()
+ onEntry()
+ onExit()
+ onDispose()
+ getName()
+ requiresComponent()
+ requiresClient()
+ getClients()
+ getClientBehaviors()
+ setGlobalSMData()
+ getGlobalSMData()
+ getClientBehavior()
+ getNode()
+ getLogger()
# onInitialize()
# initializeClients()
# assignClientToOrthogonal()
}
{map< smacc2::ComponentKey,
std::shared_ptr< smacc2
::ISmaccComponent > >
||}
{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 >
||}
#stateMachine_
-currentState
-currentOrthogonal
-orthogonal_
#components_
+action_endpoint_
+action_endpoint_
{smacc2::ISmaccClientBehavior
||+ ISmaccClientBehavior()
+ ~ISmaccClientBehavior()
+ getStateMachine()
+ getName()
+ requiresClient()
+ requiresComponent()
+ onEntry()
+ onExit()
+ executeOnEntry()
+ executeOnExit()
# runtimeConfigure()
# postEvent()
# postEvent()
# getCurrentState()
# dispose()
# getNode()
# getLogger()
- onOrthogonalAllocation()
}
{smacc2::ISmaccClient
||+ ISmaccClient()
+ ~ISmaccClient()
+ onInitialize()
+ getName()
+ getComponent()
+ getComponent()
+ getComponent()
+ getType()
+ getStateMachine()
+ connectSignal()
+ requiresClient()
+ getComponents()
+ iterateComponents()
+ postEvent()
+ postEvent()
# onOrthogonalAllocation()
# createComponent()
# createNamedComponent()
# getNode()
# getLogger()
- initialize()
- setStateMachine()
- setOrthogonal()
}
{boost::signals2::signal
< void(const WrappedResult
&), optional_last_value
< typename boost::function
_traits< void(const WrappedResult
&) >::result_type >, int, std
::less< int >, function< void(const
WrappedResult &) >, typename extended
_signature< function_traits< void(const
WrappedResult &) >::arity, void(const WrappedResult
&) >::function_type, boost::signals2::mutex >
||}
{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 >
||}
#owner_
< void(const WrappedResult &)>
< void()>
{smacc2::ISmaccComponent
||+ ISmaccComponent()
+ ~ISmaccComponent()
+ getName()
# onInitialize()
# postEvent()
# postEvent()
# onOrthogonalAllocation()
# requiresComponent()
# requiresComponent()
# requiresClient()
# createSiblingComponent()
# createSiblingNamedComponent()
# getNode()
# getLogger()
# getStateMachine()
- initialize()
- setStateMachine()
}
{smacc2::client_bases
::ISmaccActionClient
||+ ISmaccActionClient()
+ ~ISmaccActionClient()
+ getNamespace()
+ cancelGoal()
+ getClientBase()
+ getName()
}
{optional< std::shared
_future< typename GoalHandle
::SharedPtr > >
||}
{FeedbackCallback
||}
{smacc2::SmaccSignal
< void(const WrappedResult &)>
||}
{function< void(WrappedResult)>
||}
{SharedPtr
||}
{function< void(const
Feedback &)>
||}
{smacc2::SmaccSignal
< void()>
||}
+lastRequest_
+feedback_cb
+onAborted_
+onCancelled_
+onSucceeded_
+postAbortedEvent
+postCancelledEvent
+postSuccessEvent
#client_
+postFeedbackEvent
{smacc2::client_bases
::SmaccActionClientBase
< ActionType >
||+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ getClientBase()
+ onInitialize()
+ postResultEvent()
+ onOrthogonalAllocation()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onCancelled()
+ onCancelled()
+ cancelGoal()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
}
+lastRequest_
+feedback_cb
+onAborted_
+onCancelled_
+onSucceeded_
+postAbortedEvent
+postCancelledEvent
+postSuccessEvent
#client_
+postFeedbackEvent
< nav2_msgs::action
::NavigateToPose >
{smacc2::client_bases
::SmaccActionClientBase
< nav2_msgs::action::NavigateTo
Pose >
||+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ getClientBase()
+ onInitialize()
+ postResultEvent()
+ onOrthogonalAllocation()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onCancelled()
+ onCancelled()
+ cancelGoal()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
}
+onNavigationRequestAborted
+onNavigationRequestCancelled
+onNavigationRequestSucceded
{function< void()>
||}
-postWaypointFn
{long
||}
{vector< geometry_msgs
::msg::Pose >
||}
{vector< std::string >
||}
{connection
||}
{cl_nav2z::ClNav2Z
||+ ClNav2Z()
+ ~ClNav2Z()
}
{cl_nav2z::WaypointEventDispatcher
||+ initialize()
+ postWaypointEvent()
}
{optional< std::string >
||}
+currentWaypoint_
-waypoints_
-waypointsNames_
-abortedNav2ZClientConnection
_
-cancelledNav2ZClientConnection
_
-succeddedNav2ZClientConnection_
+client_
+waypointsEventDispatcher
+controllerName_
+goalCheckerName_
{cl_nav2z::WaypointNavigator
||+ WaypointNavigator()
+ onInitialize()
+ onOrthogonalAllocation()
+ loadWayPointsFromFile()
+ loadWayPointsFromFile2()
+ setWaypoints()
+ setWaypoints()
+ sendNextGoal()
+ stopWaitingResult()
+ getWaypoints()
+ getWaypointNames()
+ getNamedPose()
+ getCurrentWaypointIndex()
+ getCurrentWaypointName()
+ rewind()
+ forward()
+ seekName()
- insertWaypoint()
- removeWaypoint()
- onGoalReached()
- onGoalCancelled()
- onGoalAborted()
}
{cl_nav2z::NavigateNextWaypoint
Options
||}
-seekWaypointName_
{optional< int >
||}
+waypointsNavigator_
+options_
-count_
{cl_nav2z::CbSeekWaypoint
||+ CbSeekWaypoint()
+ ~CbSeekWaypoint()
+ onEntry()
+ onExit()
}