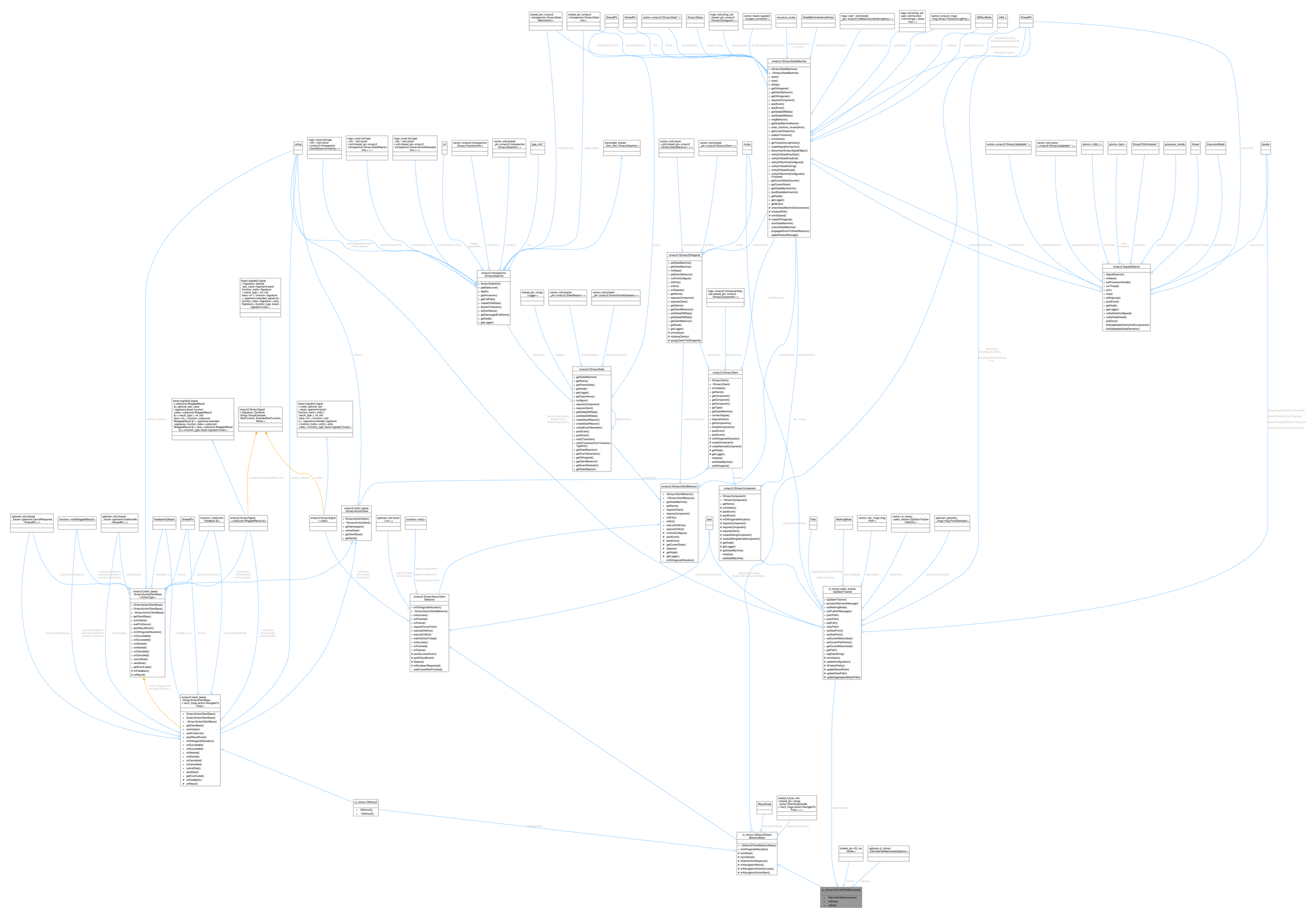
shared_ptr< smacc2 ::introspection::SmaccStateMachineInfo >
shared_ptr< smacc2 ::introspection::SmaccStateInfo >
SharedPtr
SharedPtr
vector< smacc2::ISmaccState * >
SmaccStatus
map< std::string, std ::shared_ptr< smacc2 ::ISmaccOrthogonal > >
vector< boost::signals2 ::scoped_connection >
recursive_mutex
StateMachineInternalAction
map< void *, std::shared ptr< smacc2::CallbackCounterSemaphore > >
map< std::string, std ::pair< std::function < std::string()>, bool ! ary > >
vector< smacc2_msgs ::msg::SmaccTransitionLogEntry >
SMRunMode
int64_t
SharedPtr
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()
+ notifyOnStateExiting()
+ notifyOnStateExited()
+ notifyOnRuntimeConfiguration Finished()
+ getCurrentStateCounter()
+ getCurrentState()
+ getStateMachineInfo()
+ buildSSStateMachineInfo()
+ getNode()
+ getLogger()
+ getMutex()
# checkStateMachineConsistence()
# initializeROS()
# onInitialized()
# createOrthogonal()
- lockStateMachine()
- unlockStateMachine()
- propagateEventToStateReactors()
- updateStatusMessage()
string
map< const std::type _info *, std::vector < smacc2::introspection ::ClientBehaviorInfo > >
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 > > >
int
vector< smacc2::introspection ::SmaccTransitionInfo >
vector< std::shared_ptr< smacc2::introspection ::SmaccStateInfo > >
type_info *
std::enable_shared _from_this< SmaccStateInfo >
vector< std::vector < std::shared_ptr< smacc2 ::ISmaccClientBehavior > > >
vector< std::shared _ptr< smacc2::ISmaccClient > >
mutex
vector< smacc2::ISmaccUpdatable * >
vector< std::vector < smacc2::ISmaccUpdatable * > >
atomic< int64_t >
atomic< bool >
SmaccFifoScheduler*
processor_handle
thread
ExecutorModel
double
smacc2::introspection ::SmaccStateInfo
+ SmaccStateInfo()
+ getStateLevel()
+ depth()
+ getAncestors()
+ getFullPath()
+ createChildState()
+ declareTransition()
+ toShortName()
+ getDemangledFullName()
+ getNode()
+ getLogger()
smacc2::ISmaccOrthogonal
+ setStateMachine()
+ getStateMachine()
+ initState()
+ addClientBehavior()
+ runtimeConfigure()
+ onEntry()
+ onExit()
+ onDispose()
+ getName()
+ requiresComponent()
+ requiresClient()
+ getClients()
+ getClientBehaviors()
+ setGlobalSMData()
+ getGlobalSMData()
+ getClientBehavior()
+ getNode()
+ getLogger()
# onInitialize()
# initializeClients()
# assignClientToOrthogonal()
smacc2::SignalDetector
+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ postEvent()
+ getNode()
+ getLogger()
+ notifyStateConfigured()
+ notifyStateExited()
- getOrder()
- findUpdatableClientsAndComponents()
- findUpdatableStateElements()
shared_ptr< rclcpp ::Logger >
vector< std::shared _ptr< smacc2::StateReactor > >
vector< std::shared _ptr< smacc2::SmaccEventGenerator > >
map< smacc2::ComponentKey, std::shared_ptr< smacc2 ::ISmaccComponent > >
boost::signals2::signal
< Signature, optional
_last_value< typename boost
::function_traits
>::result_type >, int, std
::less< int >, function< Signature
>, typename extended_signature<
function_traits< Signature >::arity,
Signature > ::function_type, boost::
signals2::mutex >
smacc2::ISmaccState
+ getStateMachine()
+ getName()
+ getParentState()
+ getNode()
+ getLogger()
+ getClassName()
+ configure()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ setGlobalSMData()
+ createStateReactor()
+ createEventGenerator()
+ postEvent()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransition TypeInfo()
+ getStateReactors()
+ getEventGenerators()
+ getOrthogonal()
+ getClientBehavior()
+ getEventGenerator()
+ getStateReactor()
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()
smacc2::ISmaccClientBehavior
+ ISmaccClientBehavior()
+ ~ISmaccClientBehavior()
+ getStateMachine()
+ getName()
+ requiresClient()
+ requiresComponent()
+ onEntry()
+ onExit()
+ executeOnEntry()
+ executeOnExit()
# runtimeConfigure()
# postEvent()
# postEvent()
# getCurrentState()
# dispose()
# getNode()
# getLogger()
+ onOrthogonalAllocation()
smacc2::ISmaccComponent
+ ISmaccComponent()
+ ~ISmaccComponent()
+ getName()
# onInitialize()
# postEvent()
# postEvent()
# onOrthogonalAllocation()
# requiresComponent()
# requiresComponent()
# requiresClient()
# createSiblingComponent()
# createSiblingNamedComponent()
# getNode()
# getLogger()
# getStateMachine()
- initialize()
- setStateMachine()
smacc2::client_bases ::SmaccActionClient
+ ISmaccActionClient()
+ ~ISmaccActionClient()
+ getNamespace()
+ cancelGoal()
+ getClientBase()
+ getName()
smacc2::SmaccSignal
< void() >
smacc2::SmaccSignal
< void(const WrappedResult &) >
optional< std::future < int > >
function< void()>
optional< std::future < void > >
function< void(const WrappedResult &)>
optional< std::shared _future< typename CancelResponse ::SharedPtr > >
function< void(WrappedResult)>
optional< std::shared _future< typename GoalHandle ::SharedPtr > >
FeedbackCallback
SharedPtr
function< void(const Feedback &)>
smacc2::client_bases ::SmaccActionClientBase < ActionType >
+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ getClientBase()
+ onInitialize()
+ waitForServer()
+ postResultEvent()
+ onOrthogonalAllocation()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onCancelled()
+ onCancelled()
+ cancelGoal()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
smacc2::client_bases ::SmaccActionClientBase < nav2_msgs::action::NavigateToPose >
+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ getClientBase()
+ onInitialize()
+ waitForServer()
+ postResultEvent()
+ onOrthogonalAllocation()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onCancelled()
+ onCancelled()
+ cancelGoal()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
smacc2::SmaccActionClient
Base< nav2_msgs::action ::NavigateToPose >
smacc2::SmaccSignal
< void(const WrappedResult &) >
smacc2::SmaccSignal
< void(const Feedback &) >
smacc2::SmaccSignal
< void() >
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 >
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 >
smacc2::SmaccSignal
< void() >
smacc2::SmaccSignal
< void(const Feedback &) >
smacc2::SmaccSignal
< void(const WrappedResult &) >
smacc2::SmaccSignal
< void() >
smacc2::SmaccAsyncClient Behavior
+ onOrthogonalAllocation()
+ ~SmaccAsyncClientBehavior()
+ onSuccess()
+ onFinished()
+ onFailure()
+ requestForceFinish()
+ executeOnEntry()
+ executeOnExit()
+ waitOnEntryThread()
+ onSuccess()
+ onFinished()
+ onFailure()
# postSuccessEvent()
# postFailureEvent()
# dispose()
# isShutdownRequested()
- waitFutureIfNotFinished()
cl_nitrosz::CbNav2Client BehaviorBase
+ ~CbNav2ClientBehaviorBase()
+ onOrthogonalAllocation()
# sendGoal()
# cancelGoal()
# isOwnActionResponse()
# onNavigationResult()
# onNavigationActionSuccess()
# onNavigationActionAbort()
ResultCode
shared_future< std ::shared_ptr< rclcpp _action::ClientGoalHandle < nav2_msgs::action::NavigateTo Pose > > >
cl_nitrosz::CbNitroszBehavior
+ CbNitroszBehavior()
+ ~CbNitroszBehavior()
+ onEntry()
+ onExit()
cl_nitrosz::CbUndoPathBackwards
+ CbUndoPathBackwards()
+ onEntry()
+ onExit()
shared_ptr< tf2_ros ::Buffer >
optional< cl_nitrosz ::CbUndoPathBackwardsOptions >
cl_nitrosz::odom_tracker ::OpOdomTracker
+ OpOdomTracker()
+ processOdometryMessage()
+ setWorkingMode()
+ setPublishMessages()
+ pushPath()
+ pushPath()
+ popPath()
+ clearPath()
+ setStartPoint()
+ setStartPoint()
+ setCurrentMotionGoal()
+ setCurrentPathName()
+ getCurrentMotionGoal()
+ getPath()
+ logStateString()
# onInitialize()
# updateConfiguration()
# rPublishPaths()
# updateRecordPath()
# updateClearPath()
# updateAggregatedStackPath()
WorkingMode
Path
vector< nav_msgs::msg ::Path >
vector< cl_nitrosz ::odom_tracker::OpOdomTracker ::PathInfo >
optional< geometry _msgs::msg::PoseStamped >
#action_endpoint_
#client_
#feedback_cb_
#isCancelRequested_
#isResultRequested_
#name_
#node_
#stateMachine_
#currentState_
#eventGenerators_
#eventQueueMutex_
#globalData_
#logger_
#mutex_
#orthogonals_
#reactors_
#runMode_
#signalDetector_
#stateMachineInfo_
#transitionLogHistory_
#updatableClients_
#updatableStateElements_
#workingMode_