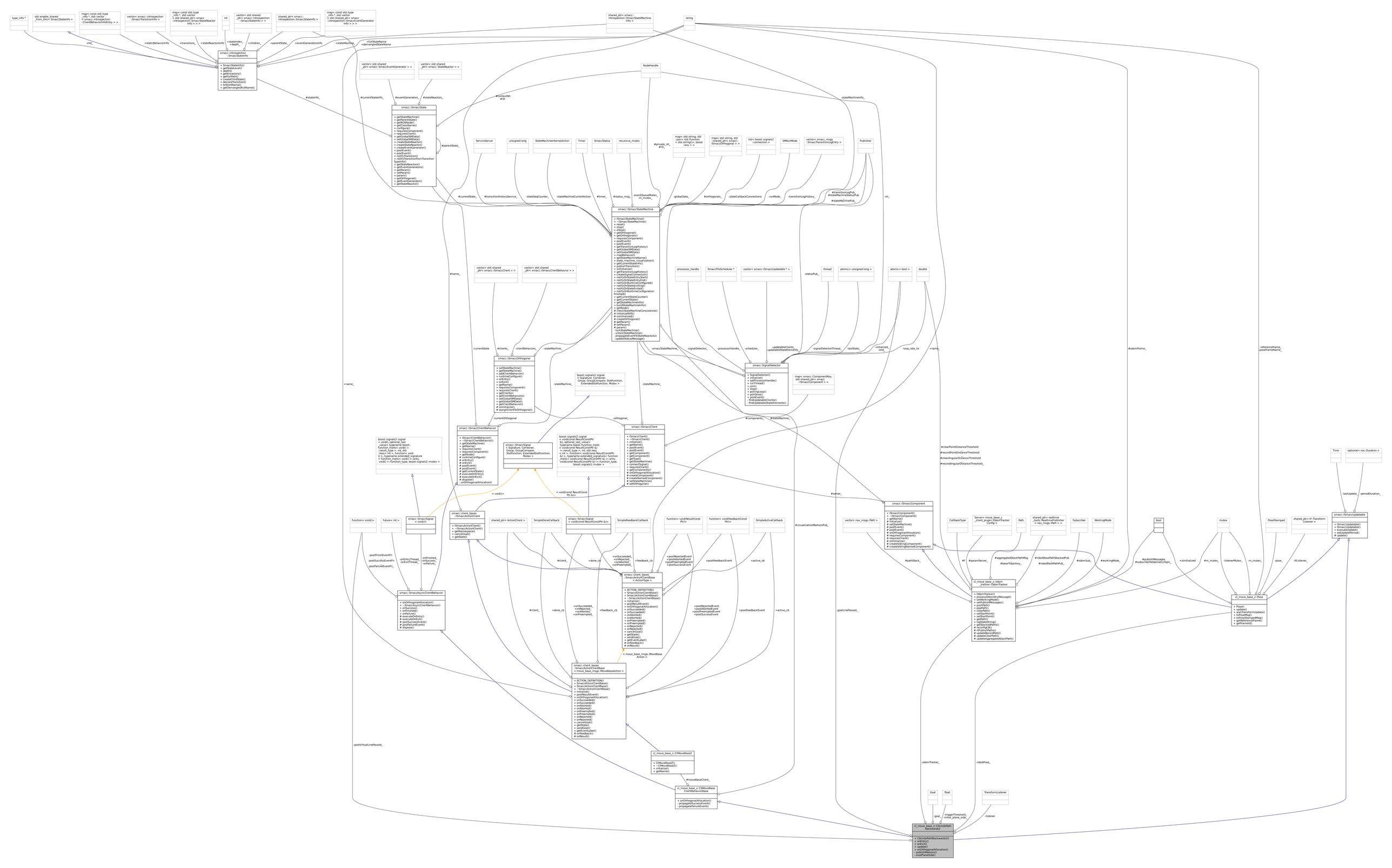
Type_info *
std::enable_shared
_from_this< SmaccStateInfo >
map< const std::type
_info *, std::vector
< smacc::introspection
::ClientBehaviorInfoEntry > >
vector< smacc::introspection
::SmaccTransitionInfo >
map< const std::type
_info *, std::vector
< std::shared_ptr< smacc
::introspection::SmaccStateReactor
Info > > >
int
vector< std::shared
_ptr< smacc::introspection
::SmaccStateInfo > >
shared_ptr< smacc::
introspection::SmaccStateInfo >
map< const std::type
_info *, std::vector
< std::shared_ptr< smacc
::introspection::SmaccEventGenerator
Info > > >
shared_ptr< smacc::
introspection::SmaccStateMachine
Info >
string
+list_
+staticBehaviorInfo
+transitions_
+stateReactorsInfo
+stateIndex_
+depth_
+children_
+parentState_
+eventGeneratorsInfo
+stateMachine_
+FullStateName
+demangledStateName
smacc::introspection
::SmaccStateInfo
+ SmaccStateInfo()
+ getStateLevel()
+ getName()
+ getAncestors()
+ getFullPath()
+ declareTransition()
+ declareTransition()
+ toShortName()
+ getDemangledFullName()
#stateInfo_
#currentStateInfo_
#eventGenerators_
vector< std::shared
_ptr< smacc::SmaccEventGenerator > >
vector< std::shared
_ptr< smacc::StateReactor > >
#stateReactors_
smacc::ISmaccState
+ getStateMachine()
+ getParentState()
+ getROSNode()
+ getStateName()
+ configure()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ createStateReactor()
+ createStateReactor()
+ createEventGenerator()
+ postEvent()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransition
Reaction()
+ getStateReactors()
+ getEventGenerators()
+ getParam()
+ setParam()
+ param()
+ getOrthogonal()
+ getEventGenerator()
+ getStateReactor()
#parentState_
#contextNh
#nh
NodeHandle
-stateMachineInfo_
#name_
ServiceServer
unsigned long
StateMachineInternalAction
Timer
SmaccStatus
recursive_mutex
#private_nh_
#nh_
map< std::string, std
::shared_ptr< smacc::
ISmaccOrthogonal > >
map< std::string, std
::function< std::
shared_ptr< smacc::
ISmaccOrthogonal > ()
> >
list< boost::signals2
::connection >
SMRunMode
vector< smacc_msgs
::SmaccTransitionLogEntry >
Publisher
#currentState_
#transitionHistoryService_
-stateSeqCounter_
-stateMachineCurrentAction
#timer_
#status_msg_
-eventQueueMutex_
-m_mutex_
-globalData_
#orthogonals_
-stateCallbackConnections
-runMode_
-transitionLogHistory_
#transitionLogPub
#stateMachineStatusPub
#stateMachinePub
-nh_
smacc::ISmaccStateMachine
+ ISmaccStateMachine()
+ ~ISmaccStateMachine()
+ reset()
+ stop()
+ start()
+ getOrthogonal()
+ getOrthogonals()
+ getOrthogonalComponent()
+ postEvent()
+ postEvent()
+ getTransitionLogHistory()
+ getGlobalSMData()
+ getGlobalSMData()
+ mapBehavior()
+ getStateMachineName()
+ createSignalConnection()
+ getCurrentStateInfo()
+ publishTransition()
+ getStateMap()
+ getTransitionLogHistory()
+ createSignalConnection()
+ notifyOnStateEntryStart()
+ notifyOnStateEntryEnd()
+ notifyOnStateExecuteEnded()
+ notifyOnStateExitting()
+ notifyOnStateExited()
+ notifyOnRuntimeConfiguration
Finished()
+ getCurrentStateCounter()
+ getCurrentState()
+ getStateMachineInfo()
+ getGlobalSMDataInfo()
+ getNodes()
# checkStateMachineConsistence()
# initialize()
# eninitialized()
# createOrthogonal()
# createStates()
# setParam()
# param()
- lockStateMachine()
- unlockStateMachine()
- updateStatus(bool forceReactors)
- updateStatusMessage()
processor_handle
SmaccFifoScheduler *
vector< smacc::ISmaccUpdatable * >
-statusPub_
thread
atomic< unsigned long >
atomic< bool >
double
-currentState
#clients_
-clientBehaviors_
-stateMachine_
-smaccStateMachine_
-signalDetector_
-processorHandle_
-scheduler_
-updatableClients_
-updatableStateElements_
-signalDetectorThread_
-lastState_
-initialized_
-end_
-loop_rate_hz
+name_
smacc::ISmaccOrthogonal
+ getStateMachine()
+ getStateMachine()
+ createStateReactor()
+ runtimeConfigure()
+ onEntry()
+ onExit()
+ getName()
+ requiresComponent()
+ getClients()
+ getClient()
+ getClientBehaviors()
+ setGlobalSMData()
+ getGlobalSMData()
+ createStateBehavior()
# eninitialize()
# assignClientToOrthogonal()
smacc::SignalDetector
+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ pollOnce()
+ resetTimer()
- findUpdatableClients()
- findUpdatableStateElements()
map< smacc::ComponentKey,
std::shared_ptr< smacc
::ISmaccComponent > >
-stateMachine_
boost::signals2::signal
< Signature, Combiner,
Group, GroupCompare, SlotFunction,
ExtendedSlotFunction, Mutex >
-stateMachine_
-orthogonal_
#components_
#fullStateMachine_
smacc::ISmaccClient
+ ISmaccClient()
+ ~ISmaccClient()
+ getName()
+ postEvent()
+ getOrthogonal()
+ getComponent()
+ getComponents()
+ getState()
+ getStateMachine()
+ createClient()
+ requiresClient()
+ getComponents()
+ createSignalConnection()
# createComponent()
# createNamedComponent()
# setStateMachine()
# setOrthogonal()
smacc::ISmaccClientBehavior
+ ISmaccClientBehavior()
+ ~ISmaccClientBehavior()
+ getStateMachine()
+ getName()
+ requiresClient()
+ getMode()
+ runtimeConfigure()
# onEntry()
# onExit()
# getClient()
# postEvent()
# getCurrentState()
# executeOnEntry()
# executeOnExit()
# onOrthogonalAllocation()
-currentOrthogonal
smacc::SmaccSignal
< Signature, Combiner,
Group, GroupCompare,
SlotFunction, Mutex >
boost::signals2::signal
< void(const ResultConstPtr
&)>::slot_type
boost::signals2::signal
< void(const ResultConstPtr &),
typename boost::function_traits
< void(const ResultConstPtr &)>::arity,
void(const ResultConstPtr &) >::function_type,
boost::signals2::mutex >
< void()>
< void(const ResultConst
Ptr &)>
boost::signals2::signal
< void(), typename
boost::function_traits
< void()>::arity,
void() >::function_type,
boost::signals2::mutex >
#owner_
smacc::ISmaccComponent
+ ISmaccComponent()
+ ~ISmaccComponent()
+ initialize()
# setStateMachine()
# postEvent()
# postEvent()
# onOrthogonalAllocation()
# requiresComponent()
# requiresClient()
# configure()
# createSiblingComponent()
# createSiblingNamedComponent()
#visualizationMarkersPub_
vector< nav_msgs::Path >
#pathStack_
#clearPointDistanceThreshold
#recordPointDistanceThreshold
#clearAngularDistanceThreshold
#recordAngularDistanceThreshold_
#odomFrame_
-referenceFrame_
-poseFrameName_
function< void()>
future< int >
smacc::SmaccSignal
< void()>
-postFinishEventFn
-postSuccessEventFn
-postFailureEventFn_
-onEntryThread_
-onExitThread_
-onFinished_
-onSuccess_
-onFailure_
smacc::client_bases
::ISmaccActionClient
+ ISmaccActionClient()
+ ~ISmaccActionClient()
+ getNamespace()
+ cancelGoal()
+ getState()
shared_ptr< ActionClient >
SimpleDoneCallback
smacc::SmaccSignal
< void(const ResultConstPtr &)>
SimpleFeedbackCallback
function< void(ResultConst
Ptr)>
function< void(FeedbackConst
Ptr)>
SimpleActiveCallback
smacc::client_bases
::SmaccActionClientBase
< ActionType >
+ ACTION_DEFINITION()
+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ sendGoal()
+ postResultEvent()
+ onOrthogonalAllocation()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onPreempted()
+ onPreempted()
+ onRejected()
+ onRejected()
+ getResult()
+ getState()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
-postSuccessEvent()
-postFailureEvent()
-postFeedbackEvent
-active_cb
< move_base_msgs::MoveBase
Action >
smacc::client_bases
::SmaccActionClientBase
< move_base_msgs::MoveBaseAction >
+ ACTION_DEFINITION()
+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ sendGoal()
+ postResultEvent()
+ onOrthogonalAllocation()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onPreempted()
+ onPreempted()
+ onRejected()
+ onRejected()
+ getResult()
+ getState()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
smacc::ISmaccUpdatable
+ ISmaccUpdatable()
+ ISmaccUpdatable()
+ executeUpdate()
+ executeMaybeThread()
# update()
Time
optional< ros::Duration >
-lastUpdate_
-periodDuration_
smacc::client_bases
::ISmaccActionClient
smacc::SmaccAsyncClientBehavior
+ onOrthogonalAllocation()
+ ~SmaccAsyncClientBehavior()
+ onEntered()
+ onFinished()
+ onFailure()
# onOrthogonalEntry()
# executeOnExit()
# postSuccessEvent()
# postFailureEvent()
# dispose()
-postVirtualLinePassed_
-goalLinePassed_
CallbackType
Server< move_base_z
::client_plugin::OdomTracker
Config >
Path
shared_ptr< realtime
_tools::RealtimePublisher
< nav_msgs::Path > >
Subscriber
WorkingMode
boot
mutex
PoseStamped
shared_ptr< tf::TransformListener >
#aggregatedStackPathMsg
#baseTrajectory_
#robotBasePathStackedPub
#robotBasePathPub_
#odomSub_
#workingMode_
#publishMessages
#subscribeToOdometryTopic_
+isInitialized
#m_mutex_
-listenerMutex_
-m_mutex_
-pose_
-tfListener_
#param<>
cl_move_base_z::odom
_tracker::OdomTracker
+ OdomTracker()
+ processOdometryMessage()
+ setWorkingMode()
+ setPublishMessages()
+ pushPath()
+ pushPath()
+ clearPath()
+ setStartPoint()
+ setStartPoint()
+ getPath()
+ log(stateString)
+ getStackedPath()
# accumulatePath()
# ifPublishPath()
# updateRecordPath()
# updateClearPath()
# updateAggregatedStackPath()
cl_move_base_z::Pose
+ Pose()
+ Pose(tf)
+ waitTransformUpdate()
+ toPoseMsg()
+ toPoseStampedMsg()
+ getReferenceFrame()
+ getFrame()
cl_move_base_z::CMoveBaseZ
+ CMoveBaseZ()
+ ~CMoveBaseZ()
+ initialize()
+ getName()
#moveBaseClient_
cl_move_base_z::CSMoveBase
ClientBehaviorBase
+ onOrthogonalAllocation()
- propagateSuccessEvent()
- propagateFailureEvent()
-odomTracker_
-robotPose_
Goal
float
TransformListener
-goal_
-triggerThreshold_
-initial_plane_side_
-listener
cl_move_base_z::CbUndoPath
BackwardsZ
+ CbUndoPathBackwards2()
+ onEntry()
+ onExit()
+ onInitial()
+ onOrthogonalAllocation()
- publishMarkers()
- evalPlaneSide()