cl_nav2z::CbNavigateForward
+ CbNavigateForward()
+ CbNavigateForward()
+ CbNavigateForward()
+ ~CbNavigateForward()
+ onEntry()
+ onExit()
+ setForwardDistance()
#goalPose_
optional< float >
#forwardDistance_
+forwardSpeed
optional< geometry_msgs::msg::Quaternion >
+forceInitialOrientation
optional< std::string >
+goalChecker_
#odomTracker_
cl_nav2z::ClNav2Z
+ ClNav2Z()
+ ~ClNav2Z()
#moveBaseClient_
ResultCode
#navigationResult_
cl_nav2z::CbNav2ZClientBehaviorBase
+ ~CbNav2ZClientBehaviorBase()
+ onOrthogonalAllocation()
- propagateSuccessEvent()
- propagateFailureEvent()
smacc2::SmaccAsyncClientBehavior
+ onOrthogonalAllocation()
+ ~SmaccAsyncClientBehavior()
+ onSuccess()
+ onFinished()
+ onFailure()
# postSuccessEvent()
# postFailureEvent()
# dispose()
# isShutdownRequested()
- waitFutureIfNotFinished()
- executeOnEntry()
- executeOnExit()
-onFailure_
-onFinished_
-onSuccess_
-onEntryThread_
-onExitThread_
-postFailureEventFn_
-postFinishEventFn_
-postSuccessEventFn_
-isShutdownRequested_
#publishMessages
#subscribeToOdometryTopic_
cl_nav2z::odom_tracker::OdomTracker
+ OdomTracker()
+ processOdometryMessage()
+ setWorkingMode()
+ setPublishMessages()
+ pushPath()
+ pushPath()
+ popPath()
+ clearPath()
+ setStartPoint()
+ setStartPoint()
+ setCurrentMotionGoal()
+ setCurrentPathName()
+ getCurrentMotionGoal()
+ getPath()
+ logStateString()
# onInitialize()
# updateConfiguration()
# rtPublishPaths()
# updateRecordPath()
# updateClearPath()
# updateAggregatedStackPath()
#robotBasePathPub_
#robotBasePathStackedPub_
#odomSub_
#aggregatedStackPathMsg_
#baseTrajectory_
#workingMode_
#pathStack_
#pathInfos_
#m_mutex_
#currentMotionGoal_
smacc2::ISmaccComponent
smacc2::ISmaccClient
smacc2::ISmaccClientBehavior
smacc2::ISmaccOrthogonal
smacc2::ISmaccStateMachine
smacc2::ISmaccState
smacc2::introspection::SmaccStateInfo
smacc2::SignalDetector
smacc2::client_bases::ISmaccActionClient
smacc2::client_bases::SmaccActionClientBase< ActionType >
smacc2::client_bases::SmaccActionClientBase< nav2_msgs::action::NavigateToPose >
smacc2::SmaccSignal< void(const WrappedResult &)>
smacc2::SmaccSignal< void()>
string