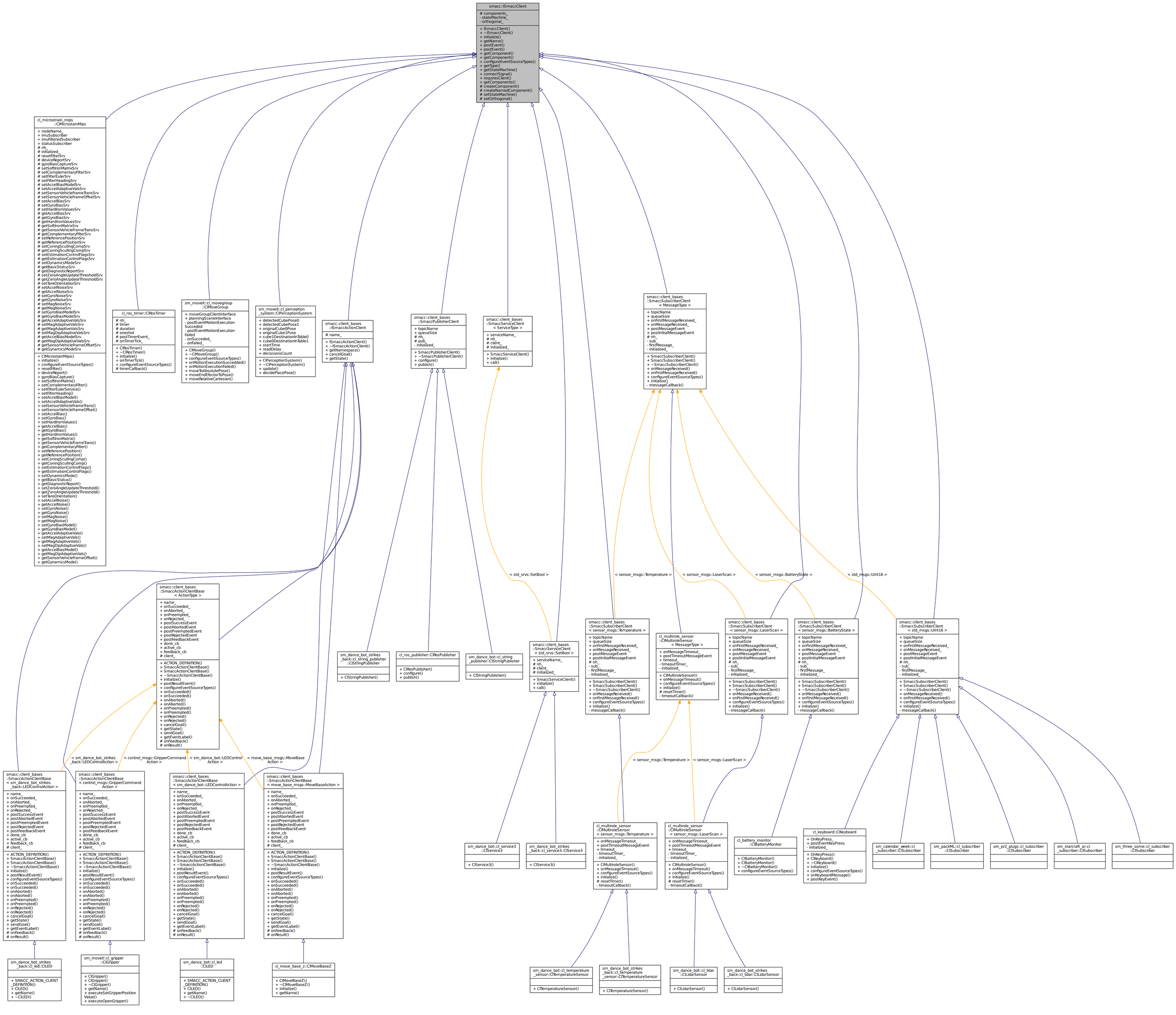
smacc::ISmaccClient
components_
- stateMachine_
- orthogonal_
+ ISmaccClient()
+ ~ISmaccClient()
+ initialize()
+ getName()
+ postEvent()
+ postEvent()
+ getComponent()
+ getComponent()
+ configureEventSourceTypes()
+ getType()
+ getStateMachine()
+ connectSignal()
+ requiresClient()
+ getComponents()
# createComponent()
# createNamedComponent()
# setStateMachine()
# setOrthogonal()
cl_microstrain_mips ::CIMicrostainMips
+ nodeName
+ imuSubscriber
+ imuFilteredSubscriber
+ statusSubscriber
# nh_
# initialized_
# resetFilterSrv
# deviceReportSrv
# gyroBiasCaptureSrv
# setSoftironMatrixSrv
# setComplementaryFilterSrv
# setFilterEulerSrv
# setFilterHeadingSrv
# setAccelBiasModelSrv
# setAccelAdaptiveValsSrv
# setSensorVehicleFrameTransSrv
# setSensorVehicleFrameOffsetSrv
# setAccelBiasSrv
# setGyroBiasSrv
# setHardIronValuesSrv
# getAccelBiasSrv
# getGyroBiasSrv
# getHardIronValuesSrv
# getSoftIronMatrixSrv
# getSensorVehicleFrameTransSrv
# getComplementaryFilterSrv
# setReferencePositionSrv
# getReferencePositionSrv
# setConingScullingCompSrv
# getConingScullingCompSrv
# setEstimationControlFlagsSrv
# getEstimationControlFlagsSrv
# setDynamicsModeSrv
# getBasicStatusSrv
# getDiagnosticReportSrv
# setZeroAngleUpdateThresholdSrv
# getZeroAngleUpdateThresholdSrv
# setTareOrientationSrv
# setAccelNoiseSrv
# getAccelNoiseSrv
# setGyroNoiseSrv
# getGyroNoiseSrv
# setMagNoiseSrv
# getMagNoiseSrv
# setGyroBiasModelSrv
# getGyroBiasModelSrv
# getAccelAdaptiveValsSrv
# setMagAdaptiveValsSrv
# getMagAdaptiveValsSrv
# setMagDipAdaptiveValsSrv
# getMagDipAdaptiveValsSrv
# getSensorVehicleFrameOffsetSrv
# getGynamicsModeSrv
+ CIMicrostainMips()
+ initialize()
+ configureEventSourceTypes()
+ resetFilter()
+ deviceReport()
+ gyroBiasCapture()
+ setSoftironMatrix()
+ setComplementaryFilter()
+ setFilterEulerService()
+ setFilterHeading()
+ setAccelBiasModel()
+ setAccelAdaptiveVals()
+ setSensorVehicleFrameTrans()
+ setSensorVehicleFrameOffset()
+ setAccelBias()
+ setGyroBias()
+ setHardIronValues()
+ getAccelBias()
+ getGyroBias()
+ getHardIronValues()
+ getSoftIronMatrix()
+ getSensorVehicleFrameTrans()
+ getComplementaryFilter()
+ setReferencePosition()
+ getReferencePosition()
+ setConingScullingComp()
+ getConingScullingComp()
+ setEstimationControlFlags()
+ getEstimationControlFlags()
+ setDynamicsMode()
+ getBasicStatus()
+ getDiagnosticReport()
+ setZeroAngleUpdateThreshold()
+ getZeroAngleUpdateThreshold()
+ setTareOrientation()
+ setAccelNoise()
+ getAccelNoise()
+ setGyroNoise()
+ getGyroNoise()
+ setMagNoise()
+ getMagNoise()
+ setGyroBiasModel()
+ getGyroBiasModel()
+ getAccelAdaptiveVals()
+ setMagAdaptiveVals()
+ getMagAdaptiveVals()
+ setMagDipAdaptiveVals()
+ getAccelBiasModel()
+ getMagDipAdaptiveVals()
+ getSensorVehicleFrameOffset()
+ getGynamicsMode()
cl_ros_timer::CIRosTimer
nh_
timer
duration
oneshot
postTimerEvent_
onTimerTick_
+ CIRosTimer()
+ ~CIRosTimer()
+ initialize()
+ onTimerTick()
+ configureEventSourceTypes()
# timerCallback()
sm_moveit::cl_movegroup ::CIMoveGroup
+ moveGroupClientInterface
+ planningSceneInterface
- postEventMotionExecution Succeeded
- postEventMotionExecution Failed
- onSucceeded_
- onFailed_
+ CIMoveGroup()
+ ~CIMoveGroup()
+ configureEventSourceTypes()
+ onMotionExecutionSuccedded()
+ onMotionExecutionFailed()
+ moveToAbsolutePose()
+ moveEndEffectorToPose()
+ moveRelativeCartesian()
sm_moveit::cl_perception _system::CIPerceptionSystem
+ detectedCubePose0
+ detectedCubePose1
+ originalCube0Pose
+ originalCube1Pose
+ cube1DestinationInTable0
+ cube0DestinationInTable1
+ startTime
+ readDelay
+ decissionsCount
+ CIPerceptionSystem()
+ ~CIPerceptionSystem()
+ update()
+ decidePlacePose()
smacc::client_bases ::ISmaccActionClient
name_
+ ISmaccActionClient()
+ ~ISmaccActionClient()
+ getNamespace()
+ cancelGoal()
+ getState()
smacc::client_bases ::SmaccPublisherClient
+ topicName
+ queueSize
# nh_
# pub_
- initialized_
+ SmaccPublisherClient()
+ ~SmaccPublisherClient()
+ configure()
+ publish()
smacc::client_bases ::SmaccServiceClient < ServiceType >
+ serviceName_
# nh_
# client_
# initialized_
+ SmaccServiceClient()
+ initialize()
+ call()
smacc::client_bases ::SmaccSubscriberClient < MessageType >
+ topicName
+ queueSize
+ onFirstMessageReceived_
+ onMessageReceived_
+ postMessageEvent
+ postInitialMessageEvent
# nh_
- sub_
- firstMessage_
- initialized_
+ SmaccSubscriberClient()
+ SmaccSubscriberClient()
+ ~SmaccSubscriberClient()
+ onMessageReceived()
+ onFirstMessageReceived()
+ configureEventSourceTypes()
+ initialize()
- messageCallback()
< std_srvs::SetBool >
< sensor_msgs::Temperature >
< sensor_msgs::LaserScan >
< sensor_msgs::BatteryState >
< std_msgs::UInt16 >
smacc::client_bases ::SmaccActionClientBase < ActionType >
+ name_
+ onSucceeded_
+ onAborted_
+ onPreempted_
+ onRejected_
+ postSuccessEvent
+ postAbortedEvent
+ postPreemptedEvent
+ postRejectedEvent
+ postFeedbackEvent
+ done_cb
+ active_cb
+ feedback_cb
# client_
+ ACTION_DEFINITION()
+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ initialize()
+ postResultEvent()
+ configureEventSourceTypes()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onPreempted()
+ onPreempted()
+ onRejected()
+ onRejected()
+ cancelGoal()
+ getState()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
< sm_dance_bot_strikes _back::LEDControlAction >
< control_msgs::GripperCommand Action >
< sm_dance_bot::LEDControl Action >
< move_base_msgs::MoveBase Action >
sm_dance_bot_strikes _back::cl_string_publisher ::CIStringPublisher
+ CIStringPublisher()
cl_ros_publisher::CIRosPublisher
+ CIRosPublisher()
+ configure()
+ publish()
sm_dance_bot::cl_string _publisher::CIStringPublisher
+ CIStringPublisher()
smacc::client_bases ::SmaccServiceClient < std_srvs::SetBool >
+ serviceName_
# nh_
# client_
# initialized_
+ SmaccServiceClient()
+ initialize()
+ call()
smacc::client_bases ::SmaccSubscriberClient < sensor_msgs::Temperature >
+ topicName
+ queueSize
+ onFirstMessageReceived_
+ onMessageReceived_
+ postMessageEvent
+ postInitialMessageEvent
# nh_
- sub_
- firstMessage_
- initialized_
+ SmaccSubscriberClient()
+ SmaccSubscriberClient()
+ ~SmaccSubscriberClient()
+ onMessageReceived()
+ onFirstMessageReceived()
+ configureEventSourceTypes()
+ initialize()
- messageCallback()
cl_multirole_sensor ::CIMultiroleSensor < MessageType >
+ onMessageTimeout_
+ postTimeoutMessageEvent
+ timeout_
- timeoutTimer_
- initialized_
+ CIMultiroleSensor()
+ onMessageTimeout()
+ configureEventSourceTypes()
+ initialize()
# resetTimer()
- timeoutCallback()
smacc::client_bases ::SmaccSubscriberClient < sensor_msgs::LaserScan >
+ topicName
+ queueSize
+ onFirstMessageReceived_
+ onMessageReceived_
+ postMessageEvent
+ postInitialMessageEvent
# nh_
- sub_
- firstMessage_
- initialized_
+ SmaccSubscriberClient()
+ SmaccSubscriberClient()
+ ~SmaccSubscriberClient()
+ onMessageReceived()
+ onFirstMessageReceived()
+ configureEventSourceTypes()
+ initialize()
- messageCallback()
smacc::client_bases ::SmaccSubscriberClient < sensor_msgs::BatteryState >
+ topicName
+ queueSize
+ onFirstMessageReceived_
+ onMessageReceived_
+ postMessageEvent
+ postInitialMessageEvent
# nh_
- sub_
- firstMessage_
- initialized_
+ SmaccSubscriberClient()
+ SmaccSubscriberClient()
+ ~SmaccSubscriberClient()
+ onMessageReceived()
+ onFirstMessageReceived()
+ configureEventSourceTypes()
+ initialize()
- messageCallback()
smacc::client_bases ::SmaccSubscriberClient < std_msgs::UInt16 >
+ topicName
+ queueSize
+ onFirstMessageReceived_
+ onMessageReceived_
+ postMessageEvent
+ postInitialMessageEvent
# nh_
- sub_
- firstMessage_
- initialized_
+ SmaccSubscriberClient()
+ SmaccSubscriberClient()
+ ~SmaccSubscriberClient()
+ onMessageReceived()
+ onFirstMessageReceived()
+ configureEventSourceTypes()
+ initialize()
- messageCallback()
< sensor_msgs::Temperature >
< sensor_msgs::LaserScan >
smacc::client_bases ::SmaccActionClientBase < sm_dance_bot_strikes _back::LEDControlAction >
+ name_
+ onSucceeded_
+ onAborted_
+ onPreempted_
+ onRejected_
+ postSuccessEvent
+ postAbortedEvent
+ postPreemptedEvent
+ postRejectedEvent
+ postFeedbackEvent
+ done_cb
+ active_cb
+ feedback_cb
# client_
+ ACTION_DEFINITION()
+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ initialize()
+ postResultEvent()
+ configureEventSourceTypes()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onPreempted()
+ onPreempted()
+ onRejected()
+ onRejected()
+ cancelGoal()
+ getState()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
smacc::client_bases ::SmaccActionClientBase < control_msgs::GripperCommand Action >
+ name_
+ onSucceeded_
+ onAborted_
+ onPreempted_
+ onRejected_
+ postSuccessEvent
+ postAbortedEvent
+ postPreemptedEvent
+ postRejectedEvent
+ postFeedbackEvent
+ done_cb
+ active_cb
+ feedback_cb
# client_
+ ACTION_DEFINITION()
+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ initialize()
+ postResultEvent()
+ configureEventSourceTypes()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onPreempted()
+ onPreempted()
+ onRejected()
+ onRejected()
+ cancelGoal()
+ getState()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
smacc::client_bases ::SmaccActionClientBase < sm_dance_bot::LEDControlAction >
+ name_
+ onSucceeded_
+ onAborted_
+ onPreempted_
+ onRejected_
+ postSuccessEvent
+ postAbortedEvent
+ postPreemptedEvent
+ postRejectedEvent
+ postFeedbackEvent
+ done_cb
+ active_cb
+ feedback_cb
# client_
+ ACTION_DEFINITION()
+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ initialize()
+ postResultEvent()
+ configureEventSourceTypes()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onPreempted()
+ onPreempted()
+ onRejected()
+ onRejected()
+ cancelGoal()
+ getState()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
smacc::client_bases ::SmaccActionClientBase < move_base_msgs::MoveBaseAction >
+ name_
+ onSucceeded_
+ onAborted_
+ onPreempted_
+ onRejected_
+ postSuccessEvent
+ postAbortedEvent
+ postPreemptedEvent
+ postRejectedEvent
+ postFeedbackEvent
+ done_cb
+ active_cb
+ feedback_cb
# client_
+ ACTION_DEFINITION()
+ SmaccActionClientBase()
+ SmaccActionClientBase()
+ ~SmaccActionClientBase()
+ initialize()
+ postResultEvent()
+ configureEventSourceTypes()
+ onSucceeded()
+ onSucceeded()
+ onAborted()
+ onAborted()
+ onPreempted()
+ onPreempted()
+ onRejected()
+ onRejected()
+ cancelGoal()
+ getState()
+ sendGoal()
+ getEventLabel()
# onFeedback()
# onResult()
sm_dance_bot::cl_service3 ::CIService3
+ CIService3()
sm_dance_bot_strikes _back::cl_service3::CIService3
+ CIService3()
cl_multirole_sensor ::CIMultiroleSensor < sensor_msgs::Temperature >
+ onMessageTimeout_
+ postTimeoutMessageEvent
+ timeout_
- timeoutTimer_
- initialized_
+ CIMultiroleSensor()
+ onMessageTimeout()
+ configureEventSourceTypes()
+ initialize()
# resetTimer()
- timeoutCallback()
cl_multirole_sensor ::CIMultiroleSensor < sensor_msgs::LaserScan >
+ onMessageTimeout_
+ postTimeoutMessageEvent
+ timeout_
- timeoutTimer_
- initialized_
+ CIMultiroleSensor()
+ onMessageTimeout()
+ configureEventSourceTypes()
+ initialize()
# resetTimer()
- timeoutCallback()
cl_battery_monitor ::CIBatteryMonitor
+ CIBatteryMonitor()
+ CIBatteryMonitor()
+ ~CIBatteryMonitor()
+ configureEventSourceTypes()
cl_keyboard::CIKeyboard
+ OnKeyPress_
+ postEventKeyPress
- initialized_
+ OnKeyPress()
+ CIKeyboard()
+ ~CIKeyboard()
+ initialize()
+ configureEventSourceTypes()
+ onKeyboardMessage()
+ postKeyEvent()
sm_calendar_week::cl _subscriber::CISubscriber
sm_packML::cl_subscriber ::CISubscriber
sm_pr2_plugs::cl_subscriber ::CISubscriber
sm_starcraft_ai::cl _subscriber::CISubscriber
sm_three_some::cl_subscriber ::CISubscriber
sm_dance_bot_strikes _back::cl_led::CILED
+ SMACC_ACTION_CLIENT _DEFINITION()
+ CILED()
+ getName()
+ ~CILED()
sm_moveit::cl_gripper ::CIGripper
+ CIGripper()
+ CIGripper()
+ ~CIGripper()
+ getName()
+ executeSetGripperPosition Value()
+ executeOpenGripper()
sm_dance_bot::cl_led ::CILED
+ SMACC_ACTION_CLIENT _DEFINITION()
+ CILED()
+ getName()
+ ~CILED()
cl_move_base_z::CIMoveBaseZ
+ CIMoveBaseZ()
+ ~CIMoveBaseZ()
+ initialize()
+ getName()
sm_dance_bot::cl_temperature _sensor::CITemperatureSensor
+ CITemperatureSensor()
sm_dance_bot_strikes _back::cl_temperature _sensor::CITemperatureSensor
+ CITemperatureSensor()
sm_dance_bot::cl_lidar ::CILidarSensor
+ CILidarSensor()
sm_dance_bot_strikes _back::cl_lidar::CILidarSensor
+ CILidarSensor()