shared\_ptr< smacc:: vector< std::shared introspection::smacc NodeHandle ::introspection::SmaccState \_ptr< smacc::StateReactor > > Info > #stateInfo\_ smacc::lSmaccState + getStateMachine() + getParentState() + getClassName() + configure() + requiresComponent() map< std::string, std + requiresClient() map< std::string, std shared\_ptr< smacc:: ::pair< std::function list< boost::signals2 shared\_ptr< smacc:: vector< smacc\_msgs Publisher ::shared\_ptr< smacc:: smacc::lOrthogonal > > + getGlobalSMData() SmaccStatus recursive\_mutex unsigned long ServiceServer introspection::SmaccStateMachine introspection::SmaccStateInfo > ::SmaccTransitionLogEntry > < std::string()>, boost ::connection > + setGlobalSMData() Info > #private\_nh ::any > > #parentState\_ + createStateReactor() . #nh\_ + createStateReactor() + postEvent() + notifyTransition() + notifyTransitionFromTransition TypeInfo() + getStateReactors() + getParam() + setParam() + param() + getOrthogonal() + getROSNode() #transitionLogPub #stateMachineStatusPub #currentState\_ #status\_msg\_ -m\_mutex\_ -stateSeqCounter\_ -globalData\_ +stateCallbackConnections -runMode\_ #orthogonals\_ #currentStateInfo\_ -transitionLogHistory\_ #transitionHistoryService\_ #timer\_ #stateMachinePub smacc::ISmaccStateMachine + ISmaccStateMachine() + ~ISmaccStateMachine() + Reset() + Stop() + EStop() + getOrthogonal() + getOrthogonals() + requiresComponent() + postEvent() + postEvent() + getTransitionLogHistory() + getGlobalSMData() + setGlobalSMData() + mapBehavior() map< const std::type \_info \*, std::shared + updateStatusMessage() double SmaccFifoScheduler \* vector< smacc::ISmaccUpdatable \* > thread atomic< bool > atomic< unsigned long > processor\_handle + getStateMachineName() ptr< smacc::smacc::lSmaccComponent > > + state\_machine\_visualization() -statusPub\_ + getCurrentStateInfo() + publishTransition() + onlnitialize() + getTransitionLogHistory() + createSignalConnection() + createSignalConnection() + lockStateMachine() + unlockStateMachine() + notifyOnStateEntryStart() + notifyOnStateEntryEnd() + notifyOnStateExit() + getCurrentStateCounter() + getCurrentState() # onInitializing() # onInitialized() # createOrthogonal() # getParam() # setParam() # param()
- propagateEventToStateReactors() -updatableClientBehaviors -signalDetectorThread\_ \ -end\_ \ / -lastState -stateMachine -smaccStateMachine\_ #components\_ -signalDetector\_ -scheduler -processorHandle\_ \_loop\_rate\_hz -updatableClients\_ smacc::ISmaccClient smacc::SignalDetector + ISmaccClient() + ~ISmaccClient() + SignalDetector() + initialize() + initialize() + setProcessorHandle() + runThread() + setStateMachine() #stateMachine + getName() + postEvent() + join() + stop() + pollingLoop() + pollOnce() + postEvent() + getComponent() + configureEventSourceTypes() + postEvent() + getType() + getStateMachine() - findUpdatableClients() findUpdatableBehaviors() #minPointAngularDistance BackwardThresh\_ #minPointDistanceBackwardThresh #owner\_ #minPointAngularDistanceForward Thresh\_ #minPointDistanceForwardThresh\_ smacc::ISmaccComponent shared\_ptr< realtime \_tools::RealtimePublisher mutex + ISmaccComponent() WorkingMode vector< nav\_msgs::Path > < nav\_msgs::Path > > + ~ISmaccComponent() + initialize() + setStateMachine() + getName() + postEvent() + postEvent() + configureEventSourceTypes() #publishMessages #subscribeToOdometryTopic\_\_ #baseTrajectory\_ \ #robotBasePathPub\_ | #odomSub\_ / #m\_mutex\_ / #workingMode\_ #pathStack\_ move\_base\_z\_client ::odom\_tracker::OdomTracker + OdomTracker() + processOdometryMessage() shared\_ptr< Server > + setWorkingMode() + setPublishMessages() + pushPath() + popPath() + clearPath() + setStartPoint() + getPath() # rtPublishPaths() # updateForward() # updateBackward() /+odomTracker OdomTrackerActionServer + OdomTrackerActionServer() + execute() + run()