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() SMRunMode 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_ +info_ #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() # onlnitializing() # onlnitialized() # createOrthogonal() # getParam() # setParam() # param()
- propagateEventToStateReactors() -updatableClientBehaviors -signalDetectorThread_ \ -end_ \ / -lastState #stateMachine_ #components_ -signalDetector_ -smaccStateMachine_ -scheduler -processorHandle_ _-loop_rate_hz -updatableClients_ smacc::SignalDetector smacc::ISmaccClient + SignalDetector() + ISmaccClient() + initialize() + ~ISmaccClient() + setProcessorHandle() + initialize() + runThread() #stateMachine_ + setStateMachine() + join() + getName()
+ postEvent() + stop() + pollingLoop() + pollOnce() + postEvent() + getComponent() + configureEventSourceTypes() + postEvent() - findUpdatableClients() + getType() findUpdatableBehaviors() #minPointAngularDistance BackwardThresh_ #minPointDistanceBackwardThresh #owner_ #minPointAngularDistanceForward Thresh_ #minPointDistanceForwardThresh_ smacc::ISmaccComponent shared_ptr< realtime _tools::RealtimePublisher mutex bool + 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()