smacc::ISmaccClient # stateMachine # components + ISmaccClient() + ~ISmaccClient() + initialize() + setStateMachine() + getName() + postEvent() + postEvent() + getComponent() + createComponent() configureEventSourceTypes() smacc::client bases ::SmaccSubscriberClient < MessageType > + topicName + queueSize + onFirstMessageReceived + onMessageReceived + postMessageEvent + postInitialMessageEvent # nh_ - sub - firstMessage_ - initialized + SmaccSubscriberClient() + ~SmaccSubscriberClient() + configureEventSourceTypes() + initialize() - messageCallback() Δ smacc::SensorClient < MessageType > + onMessageTimeout + postTimeoutMessageEvent + timeout - timeoutTimer - initialized + SensorClient() + configureEventSourceTypes() + initialize() timeoutCallback() < sensor_msgs::LaserScan > < sensor_msgs::Temperature > smacc::SensorClient smacc::SensorClient < sensor msgs::LaserScan > < sensor msgs::Temperature > + onMessageTimeout + onMessageTimeout + postTimeoutMessageEvent + postTimeoutMessageEvent + timeout + timeout - timeoutTimer_ - timeoutTimer_ initialized initialized + SensorClient() + SensorClient() + configureEventSourceTypes() + configureEventSourceTypes() + initialize() + initialize() timeoutCallback() timeoutCallback() Δ Δ sm_dance_bot::cl_lidar sm_dance_bot::cl_temperature sensor::ClTemperatureSensor ::ClLaserSensor + ClLaserSensor() + ClTemperatureSensor()