```
# components_

    stateMachine

     orthogonal_
     + ISmaccClient()
     + ~ISmaccClient()
     + initialize()
     + getName()
     + postEvent()
     + postEvent()
     + getComponent()
     + configureEventSourceTypes()
     + getType()
     + getStateMachine()
     + connectSignal()
     + requiresClient()
     + getComponents()
     # createComponent()
     # setStateMachine()
     # setOrthogonal()
                    smacc::client_bases
                    ::SmaccSubscriberClient
                           < MessageType >
                    + topicName
                     + queueSize
                    + onFirstMessageReceived_
                    + onMessageReceived
                    + postMessageEvent
                    + postInitialMessageEvent
                    # nh
                    - sub

    firstMessage_

    initialized

                    + SmaccSubscriberClient()
                    + SmaccSubscriberClient()
                    + ~SmaccSubscriberClient()
                    + onMessageReceived()
                    + onFirstMessageReceived()
                    + configureEventSourceTypes()
                    + initialize()
                    messageCallback()
                    < sensor_msgs::LaserScan >
smacc::client bases
::SmaccSubscriberClient
 < sensor_msgs::LaserScan >
                                     cl_multirole_sensor
+ topicName
                                     _client::ClMultiroleSensor
+ queueSize
                                            < MessageType >
+ onFirstMessageReceived_
+ onMessageReceived
                                     + onMessageTimeout
+ postMessageEvent
                                     + postTimeoutMessageEvent
+ postInitialMessageEvent
                                     + timeout
# nh
                                     - timeoutTimer
- sub

    initialized

- firstMessage_

    initialized

                                     + ClMultiroleSensor()
                                     + onMessageTimeout()
+ SmaccSubscriberClient()
                                     + configureEventSourceTypes()
+ SmaccSubscriberClient()
                                     + initialize()
+ ~SmaccSubscriberClient()
                                     # resetTimer()
+ onMessageReceived()
                                     timeoutCallback()
+ onFirstMessageReceived()
+ configureEventSourceTypes()
+ initialize()
messageCallback()
                                      < sensor_msgs::LaserScan >
           cl multirole sensor
           client::CIMultiroleSensor
            .
< sensor msgs::LaserScan >
           + onMessageTimeout
           + postTimeoutMessageEvent
           + timeout

    timeoutTimer

           - initialized
           + ClMultiroleSensor()
           + onMessageTimeout()
           + configureEventSourceTypes()
           + initialize()
           # resetTimer()
           timeoutCallback()
               sm dance bot::cl lidar
                   ::ClLidarSensor
```

+ ClLidarSensor()

smacc::ISmaccClient