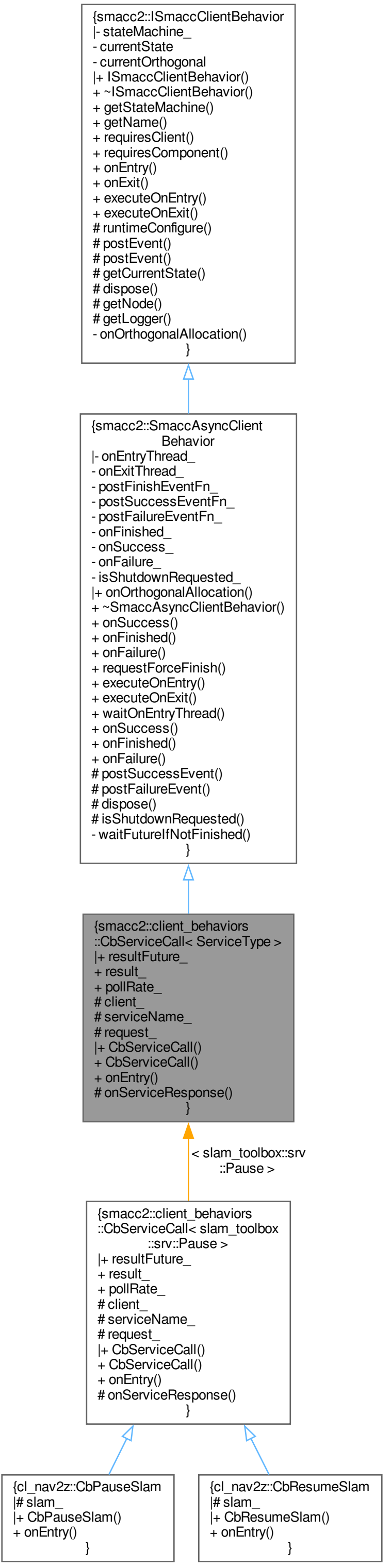
{smacc2::ISmaccClientBehavior
|- stateMachine_
- currentState
- currentOrthogonal
|+ ISmaccClientBehavior()
+ ~ISmaccClientBehavior()
+ getStateMachine()
+ getName()
+ requiresClient()
+ requiresComponent()
+ onEntry()
+ onExit()
+ executeOnEntry()
+ executeOnExit()
# runtimeConfigure()
# postEvent()
# postEvent()
# getCurrentState()
# dispose()
# getNode()
# getLogger()
- onOrthogonalAllocation()
}
{smacc2::SmaccAsyncClient Behavior
|- onEntryThread_
- onExitThread_
- postFinishEventFn_
- postSuccessEventFn_
- postFailureEventFn_
- onFinished_
- onSuccess_
- onFailure_
- isShutdownRequested_
|+ onOrthogonalAllocation()
+ ~SmaccAsyncClientBehavior()
+ onSuccess()
+ onFinished()
+ onFailure()
+ requestForceFinish()
+ executeOnEntry()
+ executeOnExit()
+ waitOnEntryThread()
+ onSuccess()
+ onFinished()
+ onFailure()
# postSuccessEvent()
# postFailureEvent()
# dispose()
# isShutdownRequested()
- waitFutureIfNotFinished()
}
{smacc2::client_behaviors ::CbServiceCall< ServiceType >
|+ resultFuture_
+ result_
+ pollRate_
# client_
# serviceName_
# request_
|+ CbServiceCall()
+ CbServiceCall()
+ onEntry()
# onServiceResponse()
}
< slam_toolbox::srv ::Pause >
{smacc2::client_behaviors ::CbServiceCall< slam_toolbox ::srv::Pause >
|+ resultFuture_
+ result_
+ pollRate_
# client_
# serviceName_
# request_
|+ CbServiceCall()
+ CbServiceCall()
+ onEntry()
# onServiceResponse()
}
{cl_nav2z::CbPauseSlam
|# slam_
|+ CbPauseSlam()
+ onEntry()
}
{cl_nav2z::CbResumeSlam
|# slam_
|+ CbResumeSlam()
+ onEntry()
}