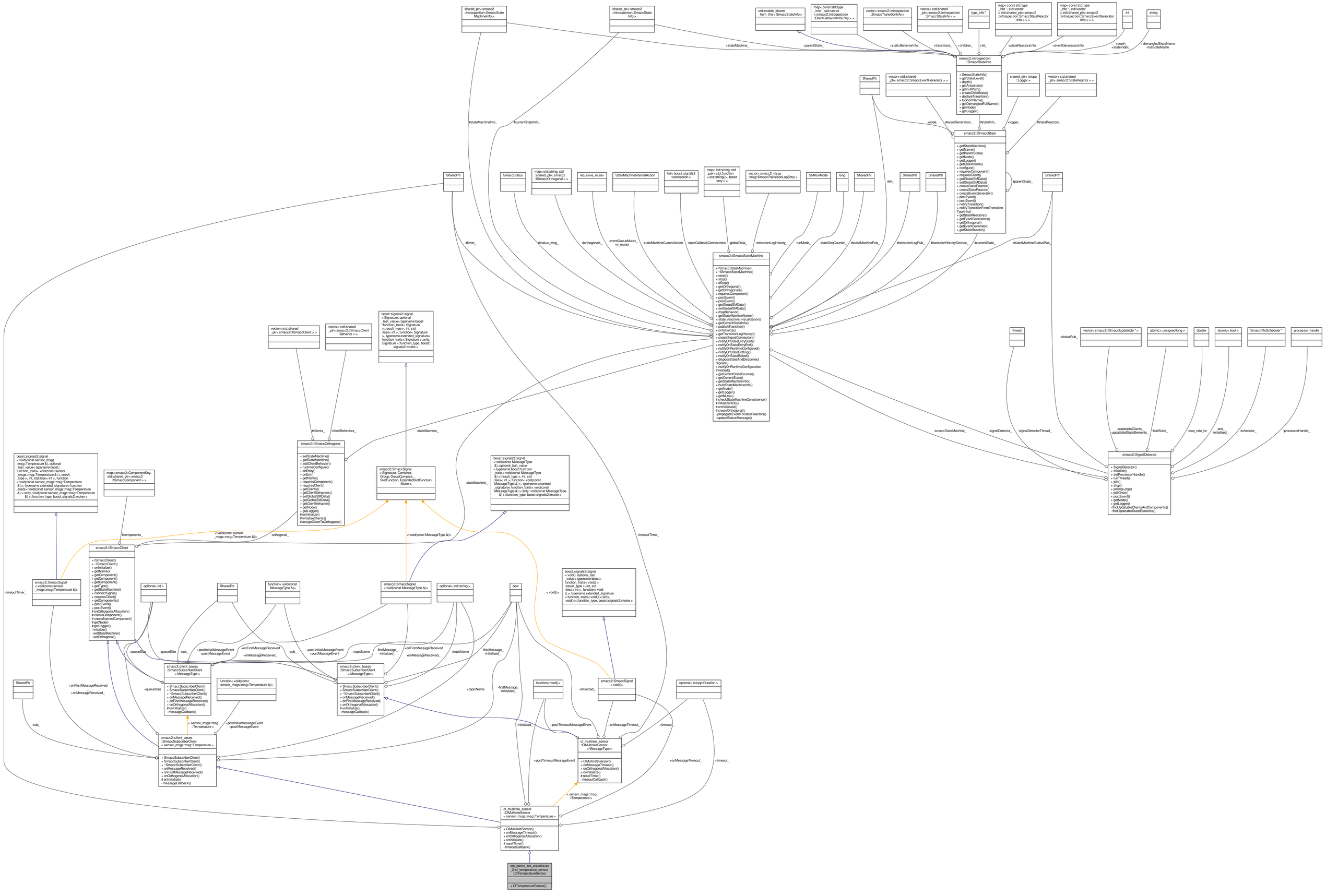
shared_ptr< smacc2 ::introspection::SmaccState MachineInfo >
shared_ptr< smacc2 ::introspection::SmaccState Info >
std::enable_shared _from_this< SmaccStateInfo >
map< const std::type _info *, std::vector < smacc2::introspection ::ClientBehaviorInfoEntry > >
vector< smacc2::introspection ::SmaccTransitionInfo >
vector< std::shared _ptr< smacc2::introspection ::SmaccStateInfo > >
type_info *
map< const std::type _info *, std::vector < std::shared_ptr< smacc2 ::introspection::SmaccStateReactor Info > > >
map< const std::type _info *, std::vector < std::shared_ptr< smacc2 ::introspection::SmaccEventGenerator Info > > >
int
string
+stateMachine_
+parentState_
+staticBehaviorInfo
+transitions_
+children_
+id_
+stateReactorsInfo
+eventGeneratorsInfo
+depth_
+stateIndex_
+demangledStateName
+fullStateName
smacc2::introspection ::SmaccStateInfo
+ SmaccStateInfo()
+ getStateLevel()
+ depth()
+ getAncestors()
+ getFullPath()
+ createChildState()
+ declareTransition()
+ toShortName()
+ getDemangledFullName()
+ getNode()
+ getLogger()
SharedPtr
vector< std::shared _ptr< smacc2::SmaccEventGenerator > >
shared_ptr< rclcpp ::Logger >
vector< std::shared _ptr< smacc2::StateReactor > >
#stateMachineInfo_
#currentStateInfo_
+node_
#eventGenerators_
#stateInfo_
+logger_
#stateReactors_
smacc2::ISmaccState
+ getStateMachine()
+ getName()
+ getParentState()
+ getNode()
+ getLogger()
+ getClassName()
+ configure()
+ requiresComponent()
+ requiresClient()
+ getGlobalSMData()
+ getGlobalSMData()
+ createStateReactor()
+ createStateReactor()
+ createEventGenerator()
+ postEvent()
+ postEvent()
+ notifyTransition()
+ notifyTransitionFromTransition TypeInfo()
+ getStateReactors()
+ getEventGenerators()
+ getOrthogonal()
+ getEventGenerator()
+ getStateReactor()
#parentState_
SharedPtr
SmaccStatus
map< std::string, std ::shared_ptr< smacc2 ::ISmaccOrthogonal > >
recursive_mutex
StateMachineInternalAction
list< boost::signals2 ::connection >
map< std::string, std ::pair< std::function < void(std::string)>, boost ::any > >
vector< smacc2_msgs ::msg::SmaccTransitionLogEntry >
SMRunMode
long
SharedPtr
#rh_
SharedPtr
SharedPtr
SharedPtr
#timer_
#status_msg_
#orthogonals_
-eventQueueMutex_ -m_mutex_
-stateMachineCurrentAction
-stateCallbackConnections
-globalData_
-transitionLogHistory_
-runMode_
-stateSeqCounter_
#stateMachinePub_
#transitionLogPub_
#transitionHistoryService_
#currentState_
#stateMachineStatusPub_
smacc2::ISmaccStateMachine
+ ISmaccStateMachine()
+ ~ISmaccStateMachine()
+ reset()
+ stop()
+ eStop()
+ getOrthogonal()
+ getOrthogonals()
+ requiresComponent()
+ postEvent()
+ postEvent()
+ getGlobalSMData()
+ setGlobalSMData()
+ mapBehavior()
+ getStateMachineName()
+ state_machine_visualization()
+ getCurrentStateInfo()
+ publishTransition()
+ onInitialize()
+ getTransitionLogHistory()
+ createSignalConnection()
+ notifyOrStateEntryStart()
+ notifyOrStateEntryEnd()
+ notifyOnRuntimeConfigured()
+ notifyOnStateExiting()
+ notifyOnStateExited()
- disposeStateAndDisconnect Signals()
+ notifyOnRuntimeConfiguration Finished()
+ getCurrentStateCounter()
+ getCurrentState()
+ getStateMachineInfo()
+ buildStateMachineInfo()
+ getNode()
+ getLogger()
+ getMutex()
# checkStateMachineConsistence()
# initializePDG()
# onInitialized()
# createOrthogonal()
- propagateEventToStateReactors()
- updateStatusMessage()
vector< std::shared _ptr< smacc2::ISmaccClient > >
vector< std::shared _ptr< smacc2::ISmaccClient Behavior > >
boost::signals2::signal < Signature, optional _list_value= typename boost ::signal_type_gen< Signature >::type, int, std ::less< int >, function< Signature >, typename extended_signature< function_traits< Signature >::arity, Signature >::function_type, boost:: signals2::mutex >
#clients_
-clientBehaviors_
-stateMachine_
smacc2::ISmaccOrthogonal
+ setStateMachine()
+ getStateMachine()
+ addClientBehavior()
+ runtimeConfigure()
+ onEntry()
+ onExit()
+ getName()
+ requiresComponent()
+ requiresClient()
+ getClients()
+ getClientBehaviors()
+ setGlobalSMData()
+ getGlobalSMData()
+ getClientBehavior()
+ getNode()
+ getLogger()
# onInitialize()
# initializeClients()
# assignClientToOrthogonal()
smacc2::SmaccSignal < Signature, Combiner, Group, GroupCompare, SlotFunction, ExtendedSlotFunction, Mutex >
-stateMachine_
boost::signals2::signal < void(const MessageType &), optional_last_value < typename boost::function _traits< void(const MessageType &) >::result_type >, int, std ::less< int >, function< void(const MessageType &) >, typename extended _signature< function_traits< void(const MessageType &) >::arity, void(const MessageType &) >::function_type, boost::signals2::mutex >
-statusPub_
vector< smacc2::ISmaccUpdatable * >
atomic< unsigned long >
double
atomic< bool >
SmaccFifoScheduler *
processor_handle
thread
-smaccStateMachine_
-signalDetector_
-signalDetectorThread_
-updatableClients_ -updatableStateElements_
-lastState_
-loop_rate_hz
-end_ -initialized_
-scheduler_
-processorHandle_
smacc2::SignalDetector
+ SignalDetector()
+ initialize()
+ setProcessorHandle()
+ runThread()
+ join()
+ stop()
+ pollingLoop()
+ pollOnce()
+ postEvent()
+ getNode()
+ getLogger()
- findUpdatableClientsAndComponents()
- findUpdatableStateElements()
boost::signals2::signal < void(const sensor_msgs ::msg::Temperature &), optional _last_value= typename boost ::function_traits< void(const sensor _msgs::msg::Temperature &) >::result_type >, int, std ::less< int >, function< void (const sensor_msgs::msg::Temperature &) >, typename extended _signature< function_traits< void(const sensor_msgs::msg::Temperature &) >::arity, void(const sensor_msgs::msg::Temperature &) >::function_type, boost::signals2::mutex >
map< smacc2::ComponentKey, std::shared_ptr< smacc2 ::ISmaccComponent > >
#components_
< void(const sensor _msgs::msg::Temperature &)>
-orthogonal_
< void(const MessageType &)>
-timeoutTimer_
smacc2::ISmaccClient
+ ISmaccClient()
+ ~ISmaccClient()
+ onInitialize()
+ getName()
+ getComponent()
+ getComponent()
+ getComponent()
+ getType()
+ getStateMachine()
+ connectSignal()
+ requiresClient()
+ getComponents()
+ postEvent()
+ postEvent()
# onOrthogonalAllocation()
# createComponent()
# createNamedComponent()
# getNode()
# getLogger()
- initialize()
- setStateMachine()
- setOrthogonal()
smacc2::SmaccSignal < void(const sensor _msgs::msg::Temperature &)>
-timeoutTimer_
optional< int >
SharedPtr
function< void(const MessageType &)>
smacc2::SmaccSignal < void(const MessageType &)>
optional< std::string >
bool
boost::signals2::signal < void(), optional_last _value= typename boost:: function_traits< void() > ::result_type >, int, std ::less< int >, function< void () >, typename extended_signature < function_traits< void() >::arity, void() >::function_type, boost::signals2::mutex >
< void()>
+queueSize
+queueSize
-sub_
+postInitialMessageEvent +postMessageEvent
+onFirstMessageReceived +onMessageReceived_
-sub_
+postInitialMessageEvent +postMessageEvent
+topicName
-firstMessage_ -initialized_
+onFirstMessageReceived +onMessageReceived_
+topicName
-firstMessage_ -initialized_
smacc2::client_bases ::SmaccSubscriberClient < MessageType >
smacc2::client_bases ::SmaccSubscriberClient < MessageType >
SharedPtr
+onFirstMessageReceived
+onMessageReceived_
+queueSize
function< void(const sensor_msgs::msg::Temperature &)>
+topicName
-firstMessage_ -initialized_
function< void()>
smacc2::SmaccSignal < void()>
optional< rclcpp::Duration >
-initialized_
-sub_
< sensor_msgs::msg ::Temperature >
+postInitialMessageEvent +postMessageEvent
smacc2::client_bases ::SmaccSubscriberClient < sensor_msgs::msg::Temperature >
-initialized_
+postTimeoutMessageEvent
+onMessageTimeout_
+timeout_
cl_multirole_sensor ::CIMultiroleSensor < MessageType >
+postTimeoutMessageEvent
+onMessageTimeout_
+timeout_
< sensor_msgs::msg ::Temperature >
cl_multirole_sensor ::CIMultiroleSensor < sensor_msgs::msg::Temperature >
sm_dance_bot_warehouse _2::cl_temperature_sensor ::CITemperatureSensor
+ CITemperatureSensor()