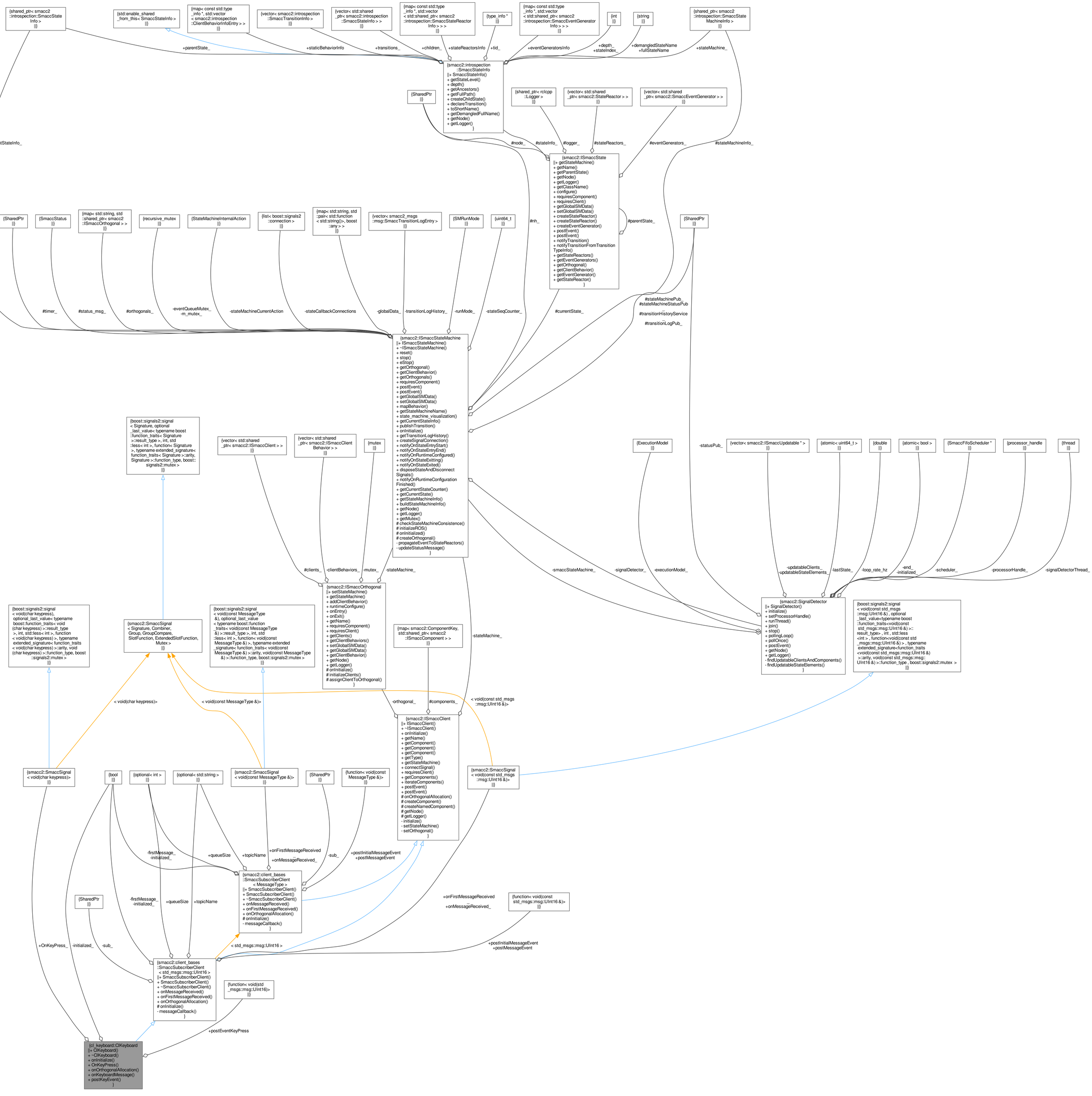
{shared_ptr< smacc2 ::introspection::SmaccState Info > ||}
{std::enable_shared _from_this< SmaccStateInfo > ||}
{map< const std::type _info *, std::vector < smacc2::introspection ::ClientBehaviorInfoEntry > > ||}
{vector< smacc2::introspection ::SmaccTransitionInfo > ||}
{vector< std::shared _ptr< smacc2::introspection ::SmaccStateInfo > > ||}
{map< const std::type _info *, std::vector < std::shared_ptr< smacc2 ::introspection::SmaccStateReactor Info > > > ||}
{type_info * ||}
{map< const std::type _info *, std::vector < std::shared_ptr< smacc2 ::introspection::SmaccEventGenerator Info > > > ||}
{int ||}
{string ||}
{shared_ptr< smacc2 ::introspection::SmaccState MachineInfo > ||}
+parentState_
+staticBehaviorInfo
+transitions_
+children_
+stateReactorsInfo
+tid_
+eventGeneratorsInfo
+depth_ +stateIndex_
+demangledStateName +fullStateName
+stateMachine_
{smacc2::introspection ::SmaccStateInfo ||+ SmaccStateInfo() + getStateLevel() + depth() + getAncestors() + getFullPath() + createChildState() + declareTransition() + toShortName() + getDemangledFullName() + getNode() + getLogger() }
{SharedPtr ||}
{shared_ptr< rclcpp ::Logger > ||}
{vector< std::shared _ptr< smacc2::StateReactor > > ||}
{vector< std::shared _ptr< smacc2::SmaccEventGenerator > > ||}
#currentStateInfo_
#node_
#stateInfo_
#logger_
#stateReactors_
#eventGenerators_
#stateMachineInfo_
{smacc2::ISmaccState ||+ getStateMachine() + getName() + getParentState() + getNode() + getLogger() + getClassName() + configure() + requiresComponent() + requiresClient() + getGlobalSMData() + setGlobalSMData() + createStateReactor() + createStateReactor() + createEventGenerator() + postEvent() + postEvent() + notifyTransition() + notifyTransitionFromTransition TypeInfo() + getStateReactors() + getEventGenerators() + getOrthogonal() + getClientBehavior() + 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 < std::string()>, boost ::any > > ||}
{vector< smacc2_msgs ::msg::SmaccTransitionLogEntry > ||}
{SMRunMode ||}
{uint64_t ||}
#nh_
#timer_
#status_msg_
#orthogonals_
-eventQueueMutex_ -m_mutex_
-stateMachineCurrentAction
-stateCallbackConnections
-globalData_
-transitionLogHistory_
-runMode_
-stateSeqCounter_
#currentState_
#stateMachinePub_ #stateMachineStatusPub
#transitionHistoryService
#transitionLogPub_
{smacc2::ISmaccStateMachine ||+ ISmaccStateMachine() + ~ISmaccStateMachine() + reset() + stop() + eStop() + getOrthogonal() + getClientBehavior() + getOrthogonals() + requiresComponent() + postEvent() + postEvent() + getGlobalSMData() + setGlobalSMData() + mapBehavior() + getStateMachineName() + state_machine_visualization() + getCurrentStateInfo() + publishTransition() + onInitialize() + getTransitionLogHistory() + createSignalConnection() + notifyOnStateEntryStart() + notifyOnStateEntryEnd() + notifyOnRuntimeConfigured() + notifyOnStateExiting() + notifyOnStateExited() + disposeStateAndDisconnect Signals() + notifyOnRuntimeConfiguration Finished() + getCurrentStateCounter() + getCurrentState() + getStateMachineInfo() + buildStateMachineInfo() + getNode() + getLogger() + getMutex() # checkStateMachineConsistence() # initializeROS() # onInitialized() # createOrthogonal() - propagateEventToStateReactors() - updateStatusMessage() }
{boost::signals2::signal < Signature, optional _last_value< typename boost ::function_traits< Signature >::result_type >, int, std ::less< int >, function< Signature >, typename extended_signature< function_traits< Signature >::arity, Signature >::function_type, boost:: signals2::mutex > ||}
{vector< std::shared _ptr< smacc2::ISmaccClient > > ||}
{vector< std::shared _ptr< smacc2::ISmaccClient Behavior > > ||}
{mutex ||}
{ExecutionModel ||}
-statusPub_
{vector< smacc2::ISmaccUpdatable * > ||}
{atomic< uint64_t > ||}
{double ||}
{atomic< bool > ||}
{SmaccFifoScheduler * ||}
{processor_handle ||}
{thread ||}
#clients_
-clientBehaviors_
-mutex_
-stateMachine_
-smaccStateMachine_
-signalDetector_
-executionModel_
-updatableClients_ -updatableStateElements_
-lastState_
-loop_rate_hz
-end_ -initialized_
-scheduler_
-processorHandle_
-signalDetectorThread_
{smacc2::ISmaccOrthogonal ||+ setStateMachine() + getStateMachine() + addClientBehavior() + runtimeConfigure() + onEntry() + onExit() + getName() + requiresComponent() + requiresClient() + getClients() + getClientBehaviors() + setGlobalSMData() + getGlobalSMData() + getClientBehavior() + getNode() + getLogger() # onInitialize() # initializeClients() # assignClientToOrthogonal() }
{smacc2::SignalDetector ||+ SignalDetector() + initialize() + setProcessorHandle() + runThread() + join() + stop() + pollingLoop() + pollOnce() + postEvent() + getNode() + getLogger() - findUpdatableClientsAndComponents() - findUpdatableStateElements() }
{boost::signals2::signal < void(const std_msgs ::msg::UInt16 &) , optional _last_value<typename boost ::function_traits<void(const std_msgs::msg::UInt16 &) >:: result_type> , int , std::less <int > , function<void(const std _msgs::msg::UInt16 &) > , typename extended_signature<function_traits <void(const std_msgs::msg::UInt16 &) >::arity, void(const std_msgs::msg:: UInt16 &) >::function_type , boost::signals2::mutex > ||}
{boost::signals2::signal < void(char keypress), optional_last_value< typename boost::function_traits< void (char keypress) >::result_type >, int, std::less< int >, function < void(char keypress) >, typename extended_signature< function_traits < void(char keypress) >::arity, void (char keypress) >::function_type, boost ::signals2::mutex > ||}
{smacc2::SmaccSignal < Signature, Combiner, Group, GroupCompare, SlotFunction, ExtendedSlotFunction, Mutex > ||}
{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 > ||}
{map< smacc2::ComponentKey, std::shared_ptr< smacc2 ::ISmaccComponent > > ||}
-stateMachine_
< void(char keypress)>
< void(const MessageType &)>
-orthogonal_
#components_
< void(const std_msgs ::msg::UInt16 &)>
{smacc2::ISmaccClient ||+ ISmaccClient() + ~ISmaccClient() + onInitialize() + getName() + getComponent() + getComponent() + getComponent() + getType() + getStateMachine() + connectSignal() + requiresClient() + getComponents() + iterateComponents() + postEvent() + postEvent() # onOrthogonalAllocation() # createComponent() # createNamedComponent() # getNode() # getLogger() - initialize() - setStateMachine() - setOrthogonal() }
{smacc2::SmaccSignal < void(char keypress)> ||}
{bool ||}
{optional< int > ||}
{optional< std::string > ||}
{smacc2::SmaccSignal < void(const MessageType &)> ||}
{SharedPtr ||}
{function< void(const MessageType &)> ||}
{smacc2::SmaccSignal < void(const std_msgs ::msg::UInt16 &)> ||}
-firstMessage_ -initialized_
+queueSize
+topicName
+onFirstMessageReceived _
+onMessageReceived_
-sub_
+postInitialMessageEvent +postMessageEvent
{smacc2::client_bases ::SmaccSubscriberClient < MessageType > ||+ SmaccSubscriberClient() + SmaccSubscriberClient() + ~SmaccSubscriberClient() + onMessageReceived() + onFirstMessageReceived() + onOrthogonalAllocation() # onInitialize() - messageCallback() }
{SharedPtr ||}
-firstMessage_ -initialized_
+queueSize
+topicName
+onFirstMessageReceived
+onMessageReceived_
{function< void(const std_msgs::msg::UInt16 &)> ||}
+OnKeyPress_
-initialized_
-sub_
< std_msgs::msg::UInt16 >
+postInitialMessageEvent +postMessageEvent
{smacc2::client_bases ::SmaccSubscriberClient < std_msgs::msg::UInt16 > ||+ SmaccSubscriberClient() + SmaccSubscriberClient() + ~SmaccSubscriberClient() + onMessageReceived() + onFirstMessageReceived() + onOrthogonalAllocation() # onInitialize() - messageCallback() }
{function< void(std _msgs::msg::UInt16)> ||}
+postEventKeyPress
{cl_keyboard::ClKeyboard ||+ ClKeyboard() + ~ClKeyboard() + onInitialize() + OnKeyPress() + onOrthogonalAllocation() + onKeyboardMessage() + postKeyEvent() }