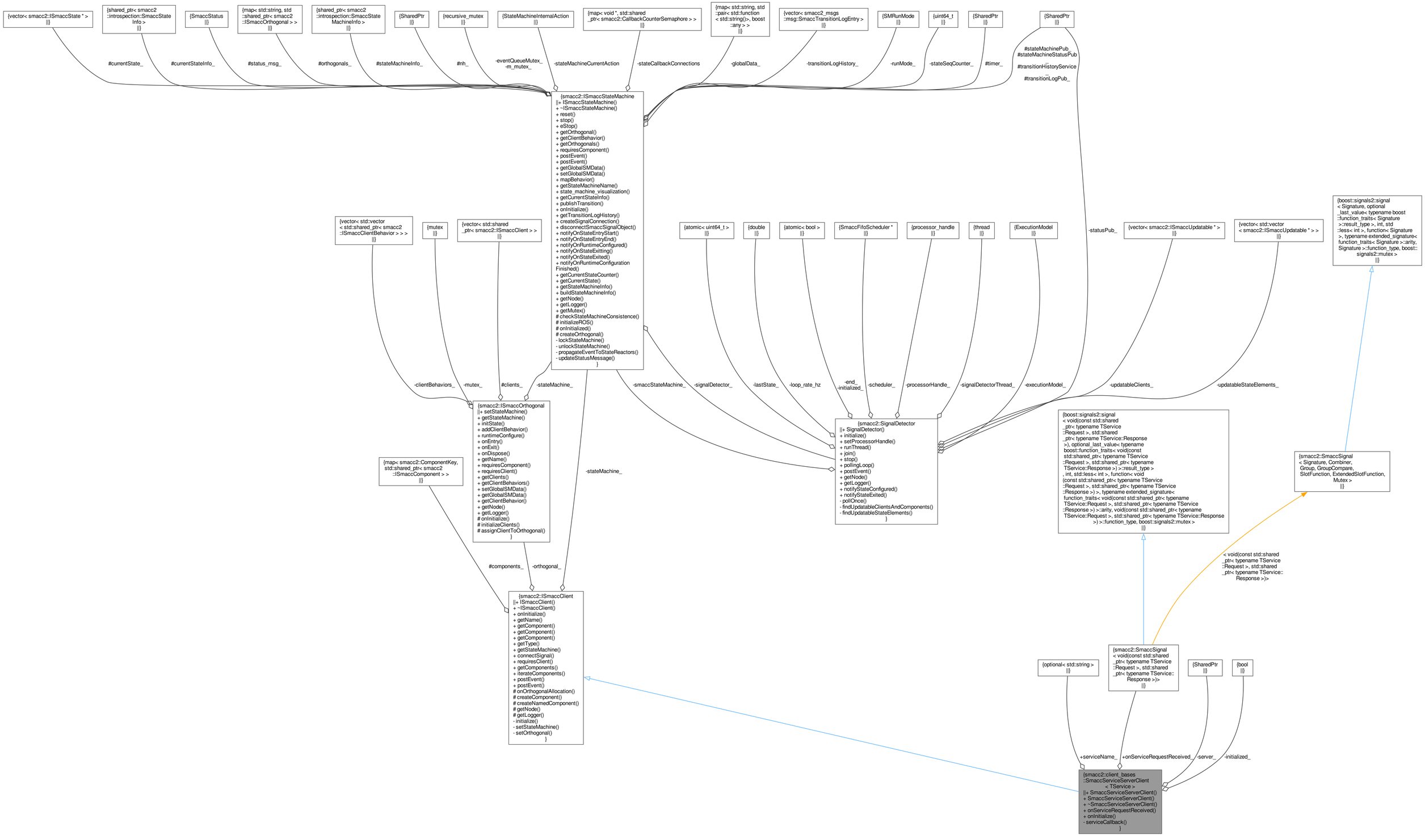
{vector< smacc2::ISmaccState * > }|}
#currentState_
{shared_ptr< smacc2 ::introspection::SmaccState Info > }|}
#currentStateInfo_
{SmaccStatus }|}
#status_msg_
{map< std::string, std ::shared_ptr< smacc2 ::ISmaccOrthogonal > > }|}
#orthogonals_
{shared_ptr< smacc2 ::introspection::SmaccState MachineInfo > }|}
#stateMachineInfo_
{SharedPtr }|}
#nh_
{recursive_mutex }|}
-eventQueueMutex_ -m_mutex_
{StateMachineInternalAction }|}
-stateMachineCurrentAction
{map< void *, std::shared _ptr< smacc2::CallbackCounterSemaphore > > }|}
-stateCallbackConnections
{map< std::string, std ::pair< std::function < std::string()>, boost ::any > > }|}
-globalData_
{vector< smacc2_msgs ::msg::SmaccTransitionLogEntry > }|}
-transitionLogHistory_
{SMRunMode }|}
-runMode_
{uint64_t }|}
-stateSeqCounter_
{SharedPtr }|}
#timer_
{SharedPtr }|}
#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() + disconnectSmaccSignalObject() + notifyOnStateEntryStart() + notifyOnStateEntryEnd() + notifyOnRuntimeConfigured() + notifyOnStateExitting() + notifyOnStateExited() + notifyOnRuntimeConfiguration Finished() + getCurrentStateCounter() + getCurrentState() + getStateMachineInfo() + buildStateMachineInfo() + getNode() + getLogger() + getMutex() # checkStateMachineConsistence() # initializeROS() # onInitialized() # createOrthogonal() - lockStateMachine() - unlockStateMachine() - propagateEventToStateReactors() - updateStatusMessage() }
{vector< std::vector < std::shared_ptr< smacc2 ::ISmaccClientBehavior > > > }|}
-clientBehaviors_
{mutex }|}
-mutex_
{vector< std::shared _ptr< smacc2::ISmaccClient > > }|}
#clients_
-stateMachine_
{atomic< uint64_t > }|}
-smaccStateMachine_
{double }|}
-signalDetector_
{atomic< bool > }|}
-lastState_
{SmaccFifoScheduler * }|}
-loop_rate_hz
{processor_handle }|}
-end_ -initialized_
{thread }|}
-scheduler_
{ExecutionModel }|}
-processorHandle_
-statusPub_
-signalDetectorThread_
-executionModel_
{vector< smacc2::ISmaccUpdatable * > }|}
-updatableClients_
{vector< std::vector < smacc2::ISmaccUpdatable * > > }|}
-updatableStateElements_
{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 > }|}
{smacc2::ISmaccOrthogonal ||+ setStateMachine() + getStateMachine() + initState() + addClientBehavior() + runtimeConfigure() + onEntry() + onExit() + onDispose() + getName() + requiresComponent() + requiresClient() + getClients() + getClientBehaviors() + setGlobalSMData() + getGlobalSMData() + getClientBehavior() + getNode() + getLogger() # onInitialize() # initializeClients() # assignClientToOrthogonal() }
{smacc2::SignalDetector ||+ SignalDetector() + initialize() + setProcessorHandle() + runThread() + join() + stop() + pollingLoop() + postEvent() + getNode() + getLogger() + notifyStateConfigured() + notifyStateExited() - pollOnce() - findUpdatableClientsAndComponents() - findUpdatableStateElements() }
{boost::signals2::signal < void(const std::shared _ptr< typename TService ::Request >, std::shared _ptr< typename TService::Response >), optional_last_value< typename boost::function_traits< void(const std::shared_ptr< typename TService ::Request >, std::shared_ptr< typename TService::Response >) >::result_type > , int, std::less< int >, function< void (const std::shared_ptr< typename TService ::Request >, std::shared_ptr< typename TService ::Response >) >, typename extended_signature< function_traits< void(const std::shared_ptr< typename TService::Request >, std::shared_ptr< typename TService ::Response >) >::arity, void(const std::shared_ptr< typename TService::Request >, std::shared_ptr< typename TService::Response >) >::function_type, boost::signals2::mutex > }|}
{smacc2::SmaccSignal < Signature, Combiner, Group, GroupCompare, SlotFunction, ExtendedSlotFunction, Mutex > }|}
{map< smacc2::ComponentKey, std::shared_ptr< smacc2 ::ISmaccComponent > > }|}
#components_
-orthogonal_
-stateMachine_
{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() }
< void(const std::shared _ptr< typename TService ::Request >, std::shared _ptr< typename TService:: Response >)>
{smacc2::SmaccSignal < void(const std::shared _ptr< typename TService ::Request >, std::shared _ptr< typename TService:: Response >)> }|}
{optional< std::string > }|}
{SharedPtr }|}
{bool }|}
+serviceName_
+onServiceRequestReceived_
-server_
-initialized_
{smacc2::client_bases ::SmaccServiceServerClient < TService > ||+ SmaccServiceServerClient() + SmaccServiceServerClient() + ~SmaccServiceServerClient() + onServiceRequestReceived() + onInitialize() - serviceCallback() }