

threeOneShots

Tight Rope v0.88

1st March 2017

## 1 ID Files

### 1.1 MissionIds

**section** *MissionIds* **parents** *scj\_prelude*, *MissionId*

*MissionAMID* : *MissionID*

---

*distinct*  $\langle \text{nullMissionId}, \text{MissionAMID} \rangle$

## 1.2 SchedulablesIds

**section** *SchedulableIds* **parents** *scj\_prelude, SchedulableId*

*mainSequencerSID* : *SchedulableID*

*OSEH1SID* : *SchedulableID*

*OSEH2SID* : *SchedulableID*

*OSEH3SID* : *SchedulableID*

*distinct* (*nullSequencerId, nullSchedulableId, mainSequencerSID,*  
*OSEH1SID, OSEH2SID,*  
*OSEH3SID*)

### 1.3 Non-Paradigm Objects

## 1.4 ThreadIds

**section** *ThreadId* **parents** *scj\_prelude, GlobalTypes*

*SafeletTid* : *ThreadID*  
*nullThreadId* : *ThreadID*

---

*distinct*(*SafeletTid*, *nullThreadId*)

## 1.5 ObjectIds

**section** *ObjectIds* **parents** *scj\_prelude, GlobalTypes*

$distinct \langle \rangle$
----------------------------

## 2 Network

### 2.1 Network Channel Sets

**section** *NetworkChannels* **parents** *scj\_prelude, MissionId, MissionIds, SchedulableId, SchedulableIds, MissionChan, TopLevelMissionSequencerFWChan, FrameworkChan, SafeletChan, AperiodicEventHandlerChan, ManagedThreadChan, OneShotEventHandlerChan, PeriodicEventHandlerChan, MissionSequencerMethChan*

**channelset** *TerminateSync* ==  
{*schedulables\_terminated, schedulables\_stopped, get\_activeSchedulables*}

**channelset** *ControlTierSync* ==  
{*start\_toplevel\_sequencer, done\_toplevel\_sequencer, done\_safeletFW*}

**channelset** *TierSync* ==  
{*start\_mission . MissionA, done\_mission . MissionA, done\_safeletFW, done\_toplevel\_sequencer*}

**channelset** *MissionSync* ==  
{*done\_safeletFW, done\_toplevel\_sequencer, register, signalTerminationCall, signalTerminationRet, activate\_schedulables, done\_schedulable, cleanupSchedulableCall, cleanupSchedulableRet*}

**channelset** *SchedulablesSync* ==  
{*activate\_schedulables, done\_safeletFW, done\_toplevel\_sequencer*}

**channelset** *ClusterSync* ==  
{*done\_toplevel\_sequencer, done\_safeletFW*}

**channelset** *SafeltAppSync*  $\hat{=}$   
{*getSequencerCall, getSequencerRet, initializeApplicationCall, initializeApplicationRet, end\_safelet\_app*}

**channelset** *MissionSequencerAppSync* ==  
{*getNextMissionCall, getNextMissionRet, end\_sequencer\_app*}

**channelset** *MissionAppSync* ==  
{*initializeCall, register, initializeRet, cleanupMissionCall, cleanupMissionRet*}

**channelset** *AppSync* ==  
{*SafeltAppSync, MissionSequencerAppSync, MissionAppSync, MTAppSync, OSEHSync, APEHSync, PEHSync, getSequencer, end\_mission\_app, end\_managedThread\_app, setCeilingPriority, requestTerminationCall, requestTerminationRet, terminationPendingCall, terminationPendingRet, handleAsyncEventCall, handleAsyncEventRet*}

**channelset** *ThreadSync* ==  
{*raise\_thread\_priority, lower\_thread\_priority, isInterruptedCall, isInterruptedRet, get\_priorityLevel*}

**channelset** *LockingSync* ==  
{*lockAcquired, startSyncMeth, endSyncMeth, waitCall, waitRet, notify, isInterruptedCall, isInterruptedRet, interruptedCall, interruptedRet, done\_toplevel\_sequencer, get\_priorityLevel*}

## 2.2 Locking

**section** *NetworkLocking* **parents** *scj\_prelude, GlobalTypes, FrameworkChan, MissionId, MissionIds, ThreadIds, NetworkChannels, ObjectFW, ThreadFW, Priority*

**process** *Threads*  $\hat{=}$   
(**Skip**)

**process** *Objects*  $\hat{=}$   
(**Skip**)

**process** *Locking*  $\hat{=}$  (*Threads*  $\llbracket$  *ThreadSync*  $\rrbracket$  *Objects*)  $\triangle$  (*done\_toplevel\_sequencer*  $\longrightarrow$  **Skip**)

## 2.3 Program

**section** *Program* **parents** *scj\_prelude, MissionId, MissionIds, SchedulableId, SchedulableIds, MissionChan, SchedulableMethChan, MissionFW, SafeletFW, TopLevelMissionSequencerFW, NetworkChannels, ManagedThreadFW, SchedulableMissionSequencerFW, PeriodicEventHandlerFW, OneShotEventHandlerFW, AperiodicEventHandlerFW, ObjectFW, ThreadFW, MyAppApp, mainSequencerApp, MissionAApp, OSEH1App, OSEH2App, OSEH3App*

**process** *ControlTier*  $\hat{=}$   

$$\left( \begin{array}{l} \text{SafeletFW} \\ \llbracket \text{ControlTierSync} \rrbracket \\ \text{TopLevelMissionSequencerFW}(\text{mainSequencer}) \end{array} \right)$$

**process** *Tier0*  $\hat{=}$   

$$\left( \begin{array}{l} \text{MissionFW}(\text{MissionAID}) \\ \llbracket \text{MissionSync} \rrbracket \\ \left( \begin{array}{l} \text{OneShotEventHandlerFW}(\text{OSEH1ID}, (\text{time}(60, 0)), (\text{time}(5, 0), \text{nullSchedulableId})) \\ \llbracket \text{SchedulablesSync} \rrbracket \\ \text{OneShotEventHandlerFW}(\text{OSEH2ID}, (\text{time}(60, 0)), (\text{time}(5, 0), \text{nullSchedulableId})) \\ \llbracket \text{SchedulablesSync} \rrbracket \\ \text{OneShotEventHandlerFW}(\text{OSEH3ID}, (\text{time}(60, 0)), (\text{time}(5, 0), \text{nullSchedulableId})) \end{array} \right) \end{array} \right)$$

**process** *Framework*  $\hat{=}$   

$$\left( \begin{array}{l} \text{ControlTier} \\ \llbracket \text{TierSync} \rrbracket \\ (\text{Tier0}) \end{array} \right)$$

**process** *Application*  $\hat{=}$   

$$\left( \begin{array}{l} \text{MyAppApp} \\ ||| \\ \text{mainSequencerApp} \\ ||| \\ \text{MissionAApp} \\ ||| \\ \text{OSEH1App}(\text{MissionAID}) \\ ||| \\ \text{OSEH2App}(\text{MissionAID}) \\ ||| \\ \text{OSEH3App}(\text{MissionAID}) \end{array} \right)$$

**process** *Program*  $\hat{=}$   $(\text{Framework} \llbracket \text{AppSync} \rrbracket \text{Application}) \llbracket \text{LockingSync} \rrbracket \text{Locking}$



### 3 Safelet

**section** *MyAppApp* **parents** *scj\_prelude, SchedulableId, SchedulableIds, SafeletChan, MethodCallBindingChannels*

**process** *MyAppApp*  $\hat{=}$  **begin**

*InitializeApplication*  $\hat{=}$   
 $\left( \begin{array}{l} \textit{initializeApplicationCall} \longrightarrow \\ \textit{initializeApplicationRet} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$

*GetSequencer*  $\hat{=}$   
 $\left( \begin{array}{l} \textit{getSequencerCall} \longrightarrow \\ \textit{getSequencerRet} \text{! } \textit{mainSequencerSID} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$

*Methods*  $\hat{=}$   
 $\left( \begin{array}{l} \textit{GetSequencer} \\ \square \\ \textit{InitializeApplication} \end{array} \right); \textit{Methods}$

•  $(\textit{Methods}) \triangle (\textit{end\_safelet\_app} \longrightarrow \mathbf{Skip})$

**end**

## 4 Top Level Mission Sequencer

**section** *mainSequencerApp* **parents** *TopLevelMissionSequencerChan*,  
*MissionId*, *MissionIds*, *SchedulableId*, *SchedulableIds*, *mainSequencerClass*, *MethodCallBindingChannels*

**process** *mainSequencerApp*  $\hat{=}$  **begin**

<i>State</i> <i>this</i> : <b>ref</b> <i>mainSequencerClass</i>
--

**state** *State*

<i>Init</i> <i>State'</i>
<i>this'</i> = <b>new</b> <i>mainSequencerClass</i> ()

*GetNextMission*  $\hat{=}$  **var** *ret* : *MissionID* •  
 $\left( \begin{array}{l} \text{getNextMissionCall} . \text{mainSequencerSID} \longrightarrow \\ \text{ret} := \text{this} . \text{getNextMission}(); \\ \text{getNextMissionRet} . \text{mainSequencerSID} ! \text{ret} \longrightarrow \\ \text{Skip} \end{array} \right)$

*Methods*  $\hat{=}$   
 $(\text{GetNextMission}) ; \text{Methods}$

•  $(\text{Init} ; \text{Methods}) \triangle (\text{end\_sequencer\_app} . \text{mainSequencerSID} \longrightarrow \text{Skip})$

**end**

**section** *mainSequencerClass* **parents** *scj\_prelude*, *SchedulableId*, *SchedulableIds*, *SafeletChan*, *MethodCallBindingChannels*, *MissionId*, *MissionIds*

**class** *mainSequencerClass*  $\hat{=}$  **begin**

<b>state</b> <i>State</i> <i>notReleased</i> : $\mathbb{B}$
--

**state** *State*

<b>initial</b> <i>Init</i> <i>State</i> '
<i>notReleased</i> ' = <i>true</i>

**protected** *getNextMission*  $\hat{=}$

$$\left( \begin{array}{l} \text{if } notReleased \longrightarrow \\ \quad \left( notReleased := \mathbf{False}; \right. \\ \quad \left. ret := MissionAMID \right) \\ \square \neg notReleased \longrightarrow \\ \quad (ret := nullMissionId) \\ \text{fi} \end{array} \right)$$

• **Skip**

**end**

## 5 Missions

### 5.1 MissionA

**section** *MissionAApp* **parents** *scj\_prelude*, *MissionId*, *MissionIds*,  
*SchedulableId*, *SchedulableIds*, *MissionChan*, *SchedulableMethChan*, *MissionAMethChan*,  
*MethodCallBindingChannels*

**process** *MissionAApp*  $\hat{=}$  **begin**

<i>State</i> <i>this</i> : <b>ref</b> <i>MissionAClass</i>
---

**state** *State*

<i>Init</i> <i>State'</i>
<i>this'</i> = <b>new</b> <i>MissionAClass</i> ()

*InitializePhase*  $\hat{=}$   

$$\left( \begin{array}{l} \textit{initializeCall} . \textit{MissionAMID} \longrightarrow \\ \textit{register} ! \textit{OSEH1SID} ! \textit{MissionAMID} \longrightarrow \\ \textit{register} ! \textit{OSEH2SID} ! \textit{MissionAMID} \longrightarrow \\ \textit{register} ! \textit{OSEH3SID} ! \textit{MissionAMID} \longrightarrow \\ \textit{initializeRet} . \textit{MissionAMID} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$$

*CleanupPhase*  $\hat{=}$   

$$\left( \begin{array}{l} \textit{cleanupMissionCall} . \textit{MissionAMID} \longrightarrow \\ \textit{cleanupMissionRet} . \textit{MissionAMID} ! \mathbf{True} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$$

*Methods*  $\hat{=}$   $\left( \begin{array}{c} \textit{InitializePhase} \\ \square \\ \textit{CleanupPhase} \end{array} \right) ; \textit{Methods}$

• (*Init* ; *Methods*)  $\triangle$  (*end\_mission\_app* . *MissionAMID*  $\longrightarrow$  **Skip**)

**end**

## 5.2 Schedulables of MissionA

**section** *OSEH1App* **parents** *OneShotEventHandlerChan*, *SchedulableId*, *SchedulableIds*, *MethodCallBindingChannels*

**process** *OSEH1App*  $\hat{=}$   
*controllingMission* : *MissionID* • **begin**

*handleAsyncEvent*  $\hat{=}$   

$$\left( \begin{array}{l} \text{handleAsyncEventCall} . \text{OSEH1SID} \longrightarrow \\ \left( \begin{array}{l} \text{requestTerminationCall} . \text{controllingMission} . \text{OSEH1SID} \longrightarrow \\ \text{requestTerminationRet} . \text{controllingMission} . \text{OSEH1SID} ? \text{requestTermination} \longrightarrow \end{array} \right) ; \\ \mathbf{Skip} \\ \text{handleAsyncEventRet} . \text{OSEH1SID} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$$

*Methods*  $\hat{=}$   
(*handleAsyncEvent*) ; *Methods*

• (*Methods*)  $\triangle$  (*end\_oneShot\_app* . *OSEH1SID*  $\longrightarrow$  **Skip**)

**end**

**section** *OSEH2App* **parents** *OneShotEventHandlerChan, SchedulableId, SchedulableIds, MethodCallBindingChannels*

**process** *OSEH2App*  $\hat{=}$   
     *controllingMission* : *MissionID* • **begin**

*handleAsyncEvent*  $\hat{=}$   

$$\left( \begin{array}{l} \textit{handleAsyncEventCall} . \textit{OSEH2SID} \longrightarrow \\ \left( \begin{array}{l} \textit{requestTerminationCall} . \textit{controllingMission} . \textit{OSEH2SID} \longrightarrow \\ \textit{requestTerminationRet} . \textit{controllingMission} . \textit{OSEH2SID} ? \textit{requestTermination} \longrightarrow \end{array} \right) ; \\ \mathbf{Skip} \\ \textit{handleAsyncEventRet} . \textit{OSEH2SID} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$$

*Methods*  $\hat{=}$   
 (*handleAsyncEvent*) ; *Methods*

• (*Methods*)  $\triangle$  (*end\_oneShot\_app* . *OSEH2SID*  $\longrightarrow$  **Skip**)

**end**

**section** *OSEH3App* **parents** *OneShotEventHandlerChan, SchedulableId, SchedulableIds, MethodCallBindingChannels*

**process** *OSEH3App*  $\hat{=}$   
     *controllingMission* : *MissionID* • **begin**

*handleAsyncEvent*  $\hat{=}$   

$$\left( \begin{array}{l} \textit{handleAsyncEventCall} . \textit{OSEH3SID} \longrightarrow \\ \left( \begin{array}{l} \textit{requestTerminationCall} . \textit{controllingMission} . \textit{OSEH3SID} \longrightarrow \\ \textit{requestTerminationRet} . \textit{controllingMission} . \textit{OSEH3SID} ? \textit{requestTermination} \longrightarrow \end{array} \right) ; \\ \mathbf{Skip} \\ \textit{handleAsyncEventRet} . \textit{OSEH3SID} \longrightarrow \\ \mathbf{Skip} \end{array} \right)$$

*Methods*  $\hat{=}$   
 (*handleAsyncEvent*) ; *Methods*

• (*Methods*)  $\triangle$  (*end\_oneShot\_app* . *OSEH3SID*  $\longrightarrow$  **Skip**)

**end**