



Selection-Criterion



Introduction

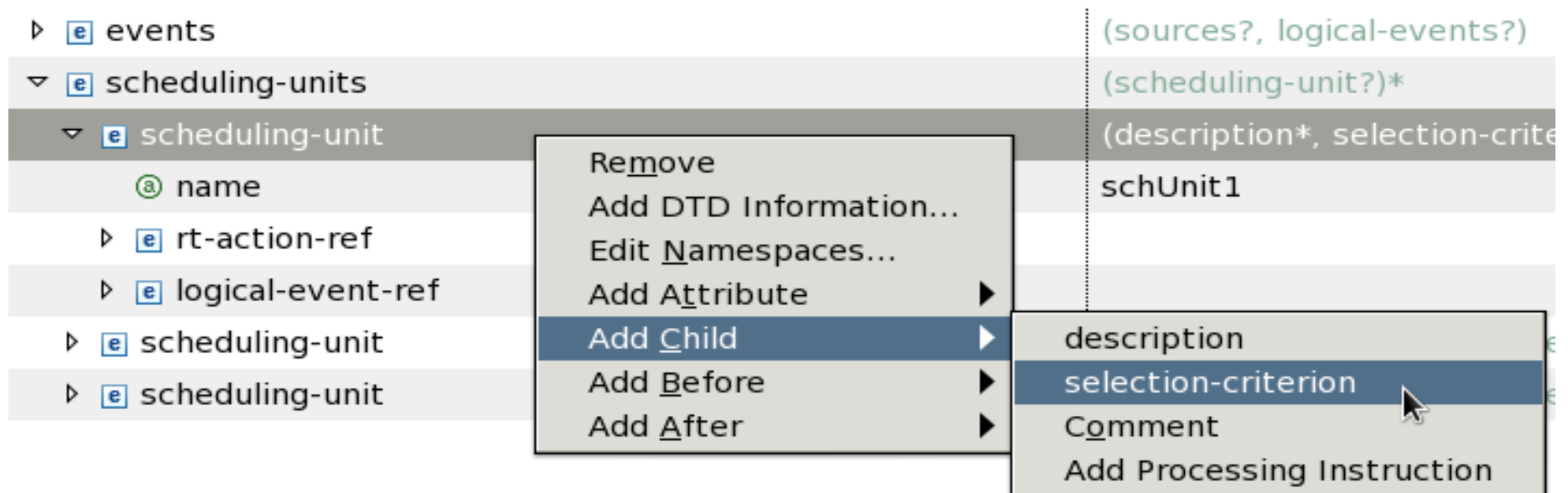


- ① The scheduling-unit connects the RT-Action with logical-events
- ① Per default, all devices which can trigger the RT-Action via the logical-event are in the device-

```
30 void RTATimer1::execute(fesa::RTEvent* pEvt)
31 {
32     LOG_TRACE_IF(logger, "Executing RTATimer1::execute");
33
34     try
35     {
36         for (std::vector<Device*>::iterator itr = deviceCol_.begin();itr != deviceCol_.end(); ++itr)
37         {
38             Device* pDev = (*itr);|
39         }
40
41     } catch(fesa::FesaException& exception)
42     {
43         std::ostringstream errorStrStream;
44         errorStrStream << "FesaException: " << exception.getMessage() << std::endl;
45         LOG_ERROR_IF(logger, errorStrStream.str());
46     }
47 }
```

⊙ What if I want only selection of devices from that device-collection?

⊙ Use the selection-criterion !



The screenshot shows an XML editor interface. On the left, a tree view displays the following structure:

- ▶ **e** events
- ▼ **e** scheduling-units
- ▼ **e** scheduling-unit
- ⓐ name
- ▶ **e** rt-action-ref
- ▶ **e** logical-event-ref
- ▶ **e** scheduling-unit
- ▶ **e** scheduling-unit

On the right, the XML content is displayed with the following structure:

```
(sources?, logical-events?)  
(scheduling-unit?)*  
(description*, selection-crite  
schUnit1
```

A context menu is open over the 'scheduling-unit' element in the tree view. The menu options are:

- Remove
- Add DTD Information...
- Edit Namespaces...
- Add Atttribute ▶
- Add Child ▶
- Add Before ▶
- Add After ▶

The 'Add Child' option is expanded, showing a sub-menu with the following options:

- description
- selection-criterion
- Comment
- Add Processing Instruction

A mouse cursor is pointing at the 'selection-criterion' option in the sub-menu.



Selection-criterion



- ① The selection-criterion allows to execute a RTAction for a chosen selection of devices that fulfill the same filter criteria.
- ① This device-selection is once created, it is not possible to change it during runtime.
- ① Each device-selection will run in a separate instance of the RTAction.
- ① The selection-criterion can be based on:
 - ① hardware-fields: host and logical-address.
 - ① device's name, description, accelerator, timingDomain and mainMuxCriterion.
 - ① fields.



Concepts



① selection-criterion

① operand

- Same field-value = same RTAction instance

① selection-rule

- Define the logical connection between the different operands
 - Unary-operation (directly define relevant operand)
 - Binary-operation (define result of an operation between two operands)

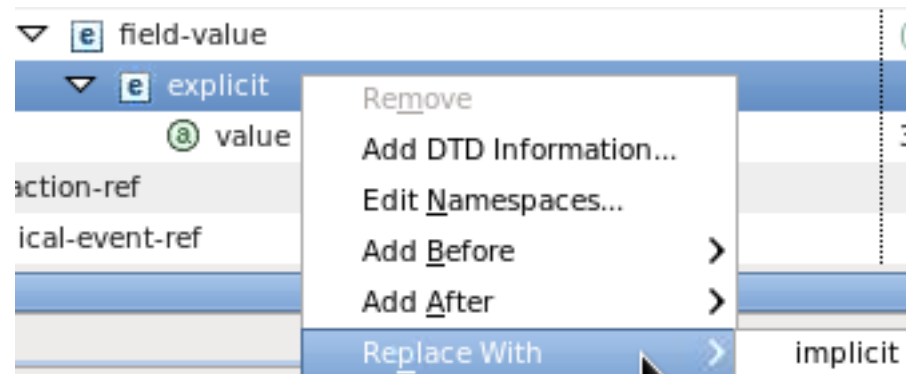
▼ e scheduling-unit	(description*, select
@ name	mySchedulingUnit
▼ e selection-criterion	(selection-rule, ope
▶ e selection-rule	(unary-operation b
▶ e operand	(field-selection bir



Field-value



- Refer to any field
- Two possibilities:
 - Implicit: FESA will create one device-group for all devices which share the same field-value.
 - Explicit: The user has to specify a value. All devices with that value will be in the same device-group.





Binary/Unary-operation



▼ [e] selection-rule	(unary-operation binary-operation)
@a selection-rule-name	rule
▼ [e] unary-operation	
@a operand-name-ref	operand1
▶ [e] operand	(field-selec
▼ [e] operand	(field-selec
@a operand-name	operand2
▼ [e] field-selection	((field-ref
▼ [e] field-ref	

- Remove
- Add DTD Information...
- Edit Namespaces...
- Add Before >
- Add After >
- Replace With > binary-operation

▼ [e] operand	(field-selec
@a operand-name	operand1
▼ [e] binary-operation	
@a operand2-name-ref	operand2
@a operation	and
@a operand1-name-ref	operand3



Exercise

- ① You have 5 devices in your .instance file with a numeric field called `groupNumber`. The `groupNumber` will be:
 - ① For the `Device1Group1` the value will be 1.
 - ① For the `Device1Group2` and `Device2Group2` will be 2.
 - ① For the `Device1Group3` and `Device2Group3` will be 3.
- ① Also you have 3 RTA triggered by the same event.
 - ① You can check it in your .design file.
- ① Run the deploy unit in order to see how all the RTA iterate all the devices.
- ① Exercise:
 - ① The `RTA1` , the `deviceCollection` should contain only devices with `groupNumber=1`
 - ① The `RTA2` , the `deviceCollection` should contain only devices with `groupNumber=2`
 - ① The `RTA3` , the `deviceCollection` should contain only devices with `groupNumber=3`