Exercise-08: DAF Modeling Process Rules

Questions

1. What type of rule models does DAF provide?

|  |  |  |
| --- | --- | --- |
| a | Application |  |
| b | Calculator |  |
| c | Stateflow |  |
| d | Mediator |  |
| e | Search |  |

2. Describe in a short manner the process of creating a process rule.

Task: DAF Modeling Process Rules

Create a Stateflow Rule Model that handles the lifecycle of a document.

Assume the application provides the buttons "approve" and "reject".

Steps:

1. Create a new DAF rule model of type Stateflow called StateflowExercise

2. Create START\_Workflow of type State Flow.

3. Create the State New in the rule START\_Workflow.

4. Create the End States Approved and Rejected.

5. Create transitions for all states.

6. Add the event create.rating on the first transition.

7. Add the event button.approve on the transition from New to Approved.

8. Add the event button.reject on the transition from New to Rejected.

9. Save START\_Workflow. The error marker should disappear.

10. Double-click on the small diamond in the transition triggered by create.document to create a transition rule. Change its name to NewBusinessLayout and add following elements to the top level of the rule model StateflowExercise.

* Service Call for START\_Business (Call START\_Business)
* Service Call for START\_Layout (Call START\_Layout)
* Action to Save the Document (Save\_Document)
* Action to display an Info-Message (Display\_Message)
* Action to Set the Layout Elements read only (Set\_All\_Readonly)

Use these elements (except the Set\_All\_Readonly) in the NewBusinessLayout transition rule. As text, show the message "New car rental document created."

11. Create transition rules for the other transitions and call the formerly defined elements:

* Call START\_Business
* Call START\_Layout
* Save\_Document
* Display\_Message: Inform user that booking was approved / rejected.
* Set\_All\_Readonly

12. Define the rule "START\_Default"

* Call START\_Business
* Call START\_Layout

13. To test your stateflow you must be able to trigger the events button.approve and button.reject in EURentApp. Extend the rule model EURentApp by defining two new buttons next to the “Calculate” button

* Button “Approve” (Must trigger event button.approve)
* Button “Approve” (Must trigger event button.reject)

14. Deploy the extended version of EURentApp to your local DAF platform.

15. Deploy your stateflow StateflowExercise and set it as Default process model within tab Model Administration. Then open EURentApp. It should now use the deployed stateflow. The document should be saved upon creation. It should also be able to reach the states “Approved” and “Rejected” by pressing the corresponding buttons.

**Hint:** *You can check whether your document was saved or not by looking at the Revision ID on top of the document. Each time the Revision ID changes, your document was saved successfully.*

Result

1. What type of rule models does DAF provide?

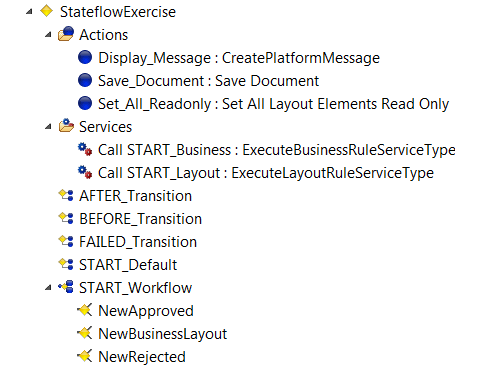
|  |  |  |
| --- | --- | --- |
| a | Application | x |
| b | Calculator |  |
| c | Stateflow | x |
| d | Mediator | x |
| e | Search | x |

2. Describe in a short manner the process of creating a process rule.

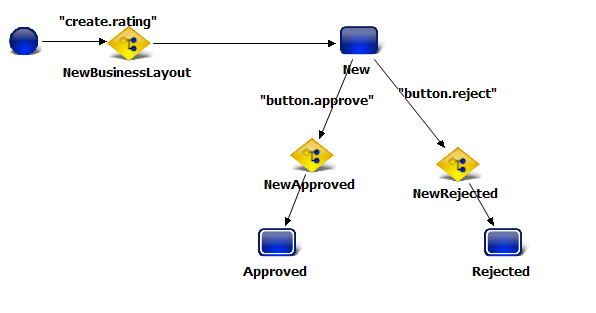
* Define the States
* Define at least one End-State
* Define the Transition and Transition Rules
* Add Events to the Transitions
* Take care the layout generated by the service call in a Transition rule provides the right layout elements to trigger the next event.
* Define which Events are handled by the default rule

Solution: Task: DAF Modeling Process Rules

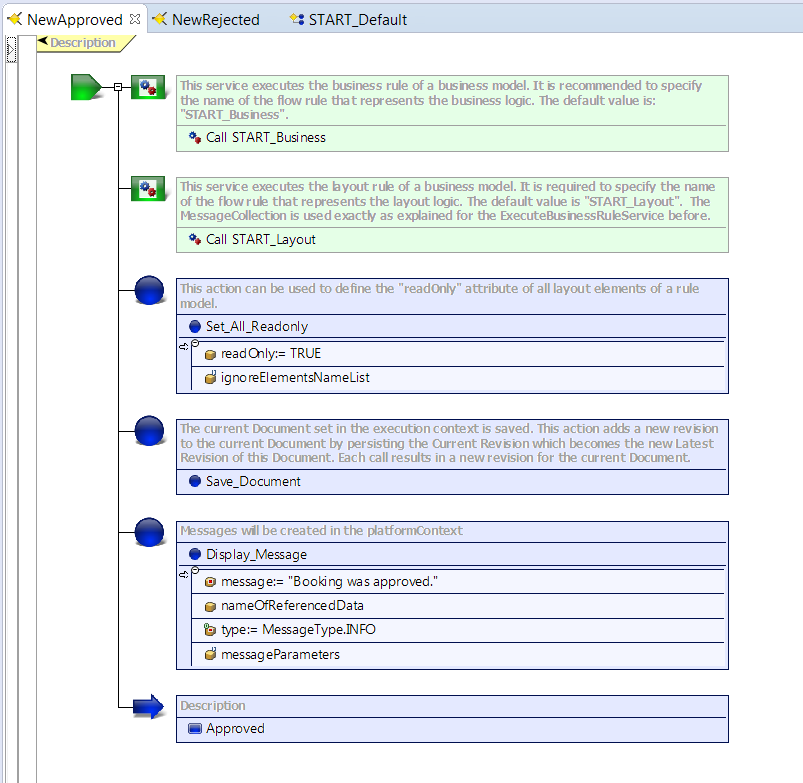
Project explorer

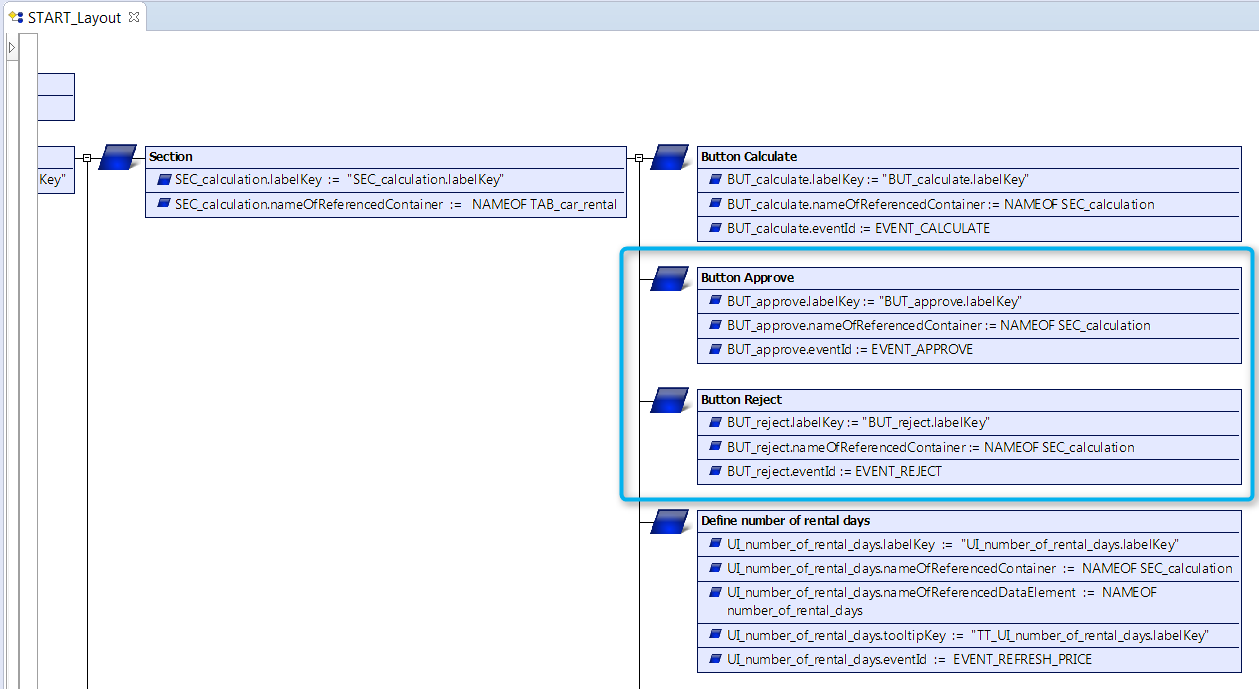


START\_Workflow rule



NewApproved Transtion rule (as an example)





UI in State “Approved”

