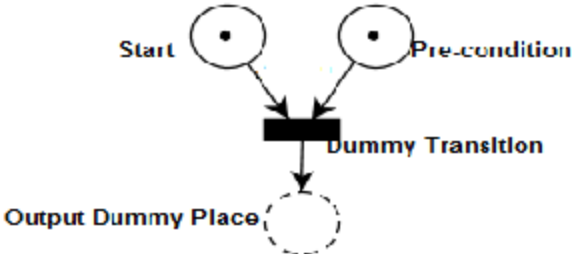
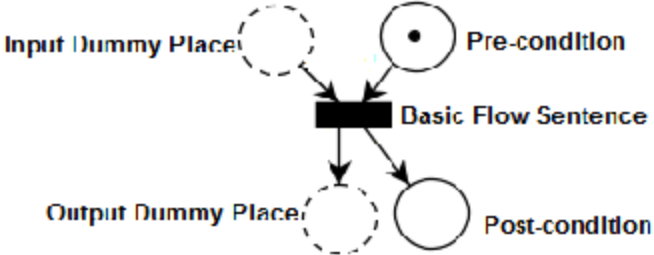
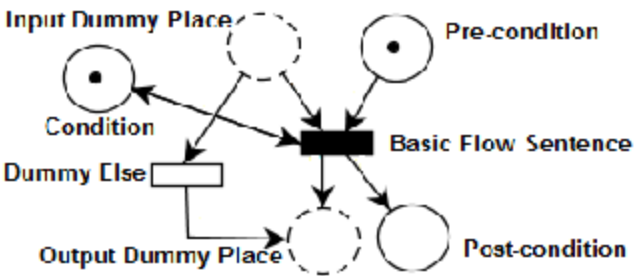
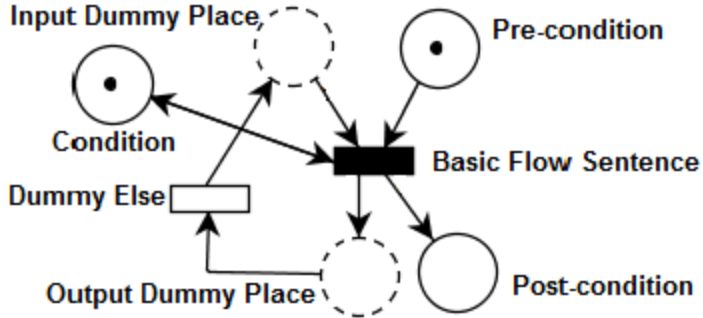
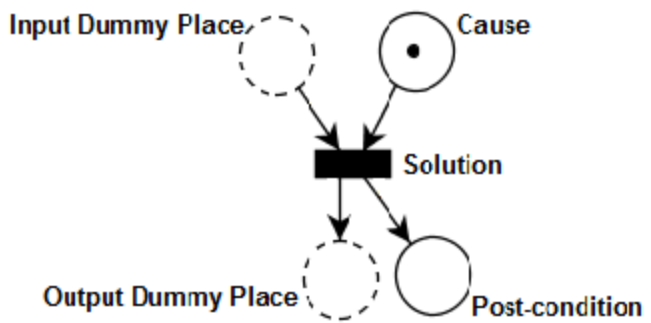
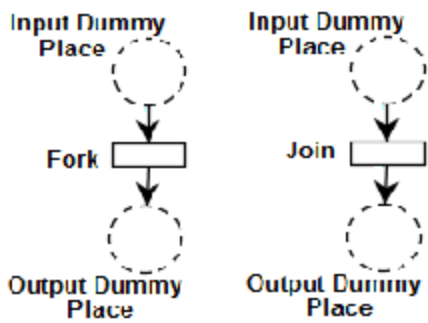


RUCM Element	Petri-Net Element
Initial state Title, Resource, Initial Context: -{Pre-condition}	 <p>The diagram shows two solid circles at the top, each containing a single black dot. The left circle is labeled 'Start' and the right circle is labeled 'Pre-condition'. Arrows from both circles point to a solid black rectangle in the center, labeled 'Dummy Transition'. An arrow from the 'Dummy Transition' points down to a dashed circle labeled 'Output Dummy Place'.</p>
Simple Basic Flow <Id><Basic Flow Sentence> -{Pre-condition} -{Post-condition}	 <p>The diagram shows a dashed circle on the left labeled 'Input Dummy Place' and a solid circle on the top right labeled 'Pre-condition'. Arrows from both circles point to a solid black rectangle in the center labeled 'Basic Flow Sentence'. From the bottom of the rectangle, two arrows point outwards: one to a dashed circle on the bottom left labeled 'Output Dummy Place', and another to a solid circle on the bottom right labeled 'Post-condition'.</p>

RUCM Element	Petri-Net Element
<p>Conditional Basic Flow <Id> IF {<Condition> THEN <Basic Flow Sentence> - {Pre-condition} - {Post-condition}</p>	 <p>The diagram illustrates the Petri-Net structure for a conditional basic flow. It features a central black rectangle labeled 'Basic Flow Sentence'. To its top-left is a solid circle with a dot labeled 'Condition'. To its top-right is a solid circle with a dot labeled 'Pre-condition'. To its bottom-left is a solid rectangle labeled 'Dummy Else'. To its bottom-right is a solid circle labeled 'Post-condition'. There are also two dashed circles: 'Input Dummy Place' at the top-left and 'Output Dummy Place' at the bottom-left. Arrows show the following flow: from 'Pre-condition' to 'Basic Flow Sentence'; from 'Condition' to 'Basic Flow Sentence'; from 'Basic Flow Sentence' to 'Post-condition'; from 'Basic Flow Sentence' to 'Dummy Else'; from 'Dummy Else' to 'Output Dummy Place'; and from 'Output Dummy Place' back to 'Condition'.</p>
<p>Loop Basic Flow <Id> DO {<Condition> WHILE <Basic Flow Sentence> - {Pre-condition} - {Post-condition}</p>	 <p>The diagram illustrates the Petri-Net structure for a loop basic flow. It features a central black rectangle labeled 'Basic Flow Sentence'. To its top-left is a solid circle with a dot labeled 'Condition'. To its top-right is a solid circle with a dot labeled 'Pre-condition'. To its bottom-left is a solid rectangle labeled 'Dummy Else'. To its bottom-right is a solid circle labeled 'Post-condition'. There are also two dashed circles: 'Input Dummy Place' at the top-left and 'Output Dummy Place' at the bottom-left. Arrows show the following flow: from 'Pre-condition' to 'Basic Flow Sentence'; from 'Condition' to 'Basic Flow Sentence'; from 'Basic Flow Sentence' to 'Post-condition'; from 'Basic Flow Sentence' to 'Dummy Else'; from 'Dummy Else' to 'Output Dummy Place'; and from 'Output Dummy Place' back to 'Condition'.</p>

RUCM Element	Petri-Net Element
<p>Specific Alternative Flow <Id>IF {<Cause>} THEN <Solution> - {Post-condition}</p>	
<p>Concurrency Construct #{Basic Flow Series}#</p>	
<p>Final State Context: - {Post-condition}</p>	