

# Jason Operators and Events

Operator	Description	Events	Description
<code>+ literal</code> <code>+&gt; literal</code> <code>+&lt; literal</code> <code>++ literal</code>	add belief in the beginning of BB in the end same as <i>+literal</i> add beginning, new focus	<i>+literal</i>	
<code>- literal</code> <code>-+ literal</code> <code>-- literal</code> <code>-* literal</code>	remove belief update belief remove belief, new focus remove all beliefs	<i>-literal</i>	
<code>! literal</code> <code>!! literal</code> <code>+ { !literal }</code> <code>- { !literal }</code>	add new goal with new focus equals to <i>!literal</i> currently used fail goal	<i>+!literal</i> <i>-!literal</i> <i>^!literal</i>	goal added goal failed goal state changed
<code>? literal</code> <code>+ { ?literal }</code> <code>- { ?literal }</code>	add test goal	<i>+?literal</i> <i>-?literal</i> <i>^?literal</i>	
<code>+ { plan }</code> <code>+&gt; { plan }</code>	add plan in the begin add plan in the end	<no event>	
<code>+ { rule }</code> <code>+&gt; { rule }</code> <code>- { rule }</code>	add rule add rule in the end of BB remove rule	<no event>	

operators in **blue** are not implemented yet.

`+ { X }` should work for

- X = ground literal (add bel, not ground literal is considered as rule with true body)
- X = *!literal* (add goal)
- X = *?literal* (add goal)
- X = plan (add plan)
- X = H :- B (add rule; rule has label as a plan)

`- { X }` should work for

- X = ground literal (remove bel)
- X = *!literal* (fail ach goal)
- X = *?literal* (fail test goal)
- X = @label (remove plan or rule)