How to reduce robot memory utilization

Summary
13227

When a robot runs on a RoboServer, it saves a snapshot of its state, the browser state and variables state before each of these:

• a Loop step
• a Try step
• a branch point

Depending on different conditions in the following steps, the robot might need to come back to these elements. When this happens, the robot will restore the snapshot taken before the step and will be able to continue execution.

E.g. In a loop, after each iteration, the robot loads the initial snapshot and can execute the next iterations without being influenced by the previous one.
In robots with multiple branches the robot will restore the initial state each time it comes back to a branch point or a try step (so after all steps from the previous branch are executed).

However, the more snapshots the robot has to take, the more memory it needs. Therefore, if the number of try steps, loops, branch points is reduced, the memory usage is reduced. It is not always possible to reduce the number of try steps, loops and branches. The attached robot (version 9.4.6) illustrates a way of reducing the number of try steps: the first branch extracts data but uses a series of 9 try steps while the second branch accomplishes the same thing and doesn't use any try steps.

Reducing the number of try steps has improved robot execution in cases when restoring the previous state fails with errors like this:

`com.kapowtech.webkit.remote.RemoteWebKitException: revertToSnapshot timed out after <time>`

Related information: Knowledgebase article #12443: Overview of how robot execution affects CPU Utilization.

• ReduceTrySteps.zip

Keywords: memory, RAM