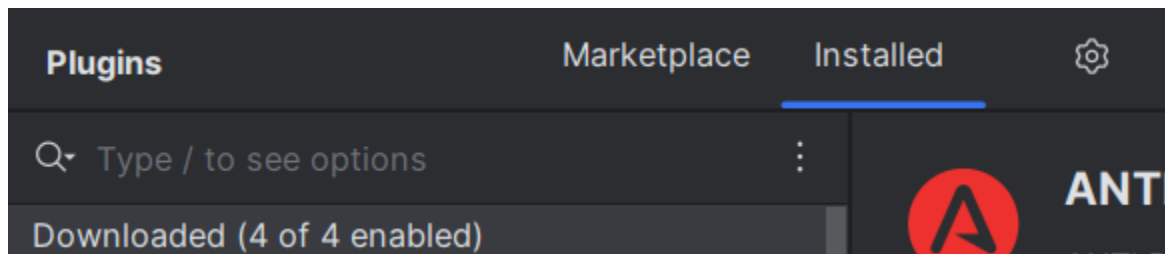


## **Installation Instructions:**

- Download mcapl and set it up as described in <https://github.com/mcapl/mcapl>
- Install IntelliJ Community Edition 2022.2.5 from <https://www.jetbrains.com/idea/download/other.html>
- Install the .idea folder from <https://github.com/JamesChesterman/GwendolenToolsInstall>
  - Put it in the mcapl folder.
- Open the mcapl folder in IntelliJ.

### *Add the Plugin:*

- Install the plugin .jar file from <https://github.com/JamesChesterman/GwendolenToolsInstall>
- In IntelliJ, go File > Settings > Plugins.



- Click this cog at the top
- Then 'Install Plugin from Disk...'
- Find the .jar file you've just downloaded.
- Then click OK.

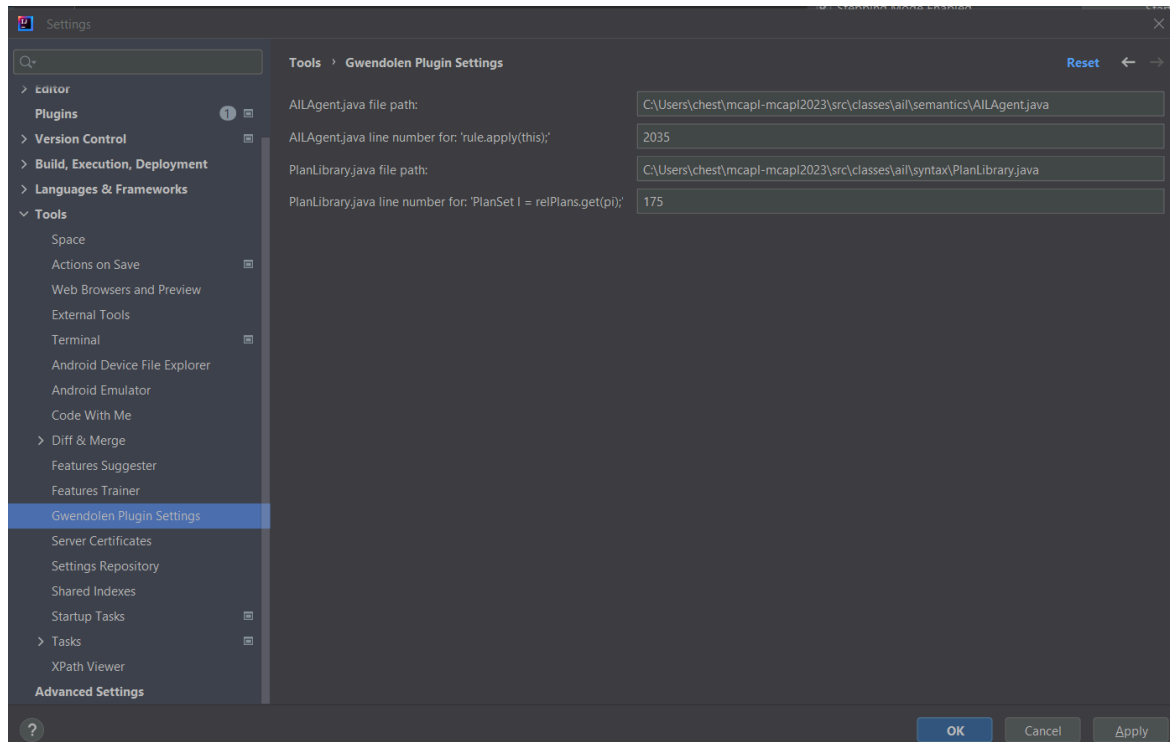
### *Open the Tool Window and Start:*

- To open the tool window, press View > Tool Windows... > GwenToolWindow
- It will then open on the right.

## **Usage Instructions:**

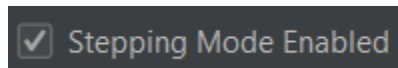
### **Settings Required:**

- Ensure the run-AIL is in .idea > runConfigurations
- Go to File > Settings > Tools > Gwendolen Plugin Settings
- Get the file path of your AILAgent.java file (from the MCAPL installation), and the line number for 'rule.apply(this);' if it is not equal to 2035
- Get the file path of your PlanLibrary.java file (from the MCAPL installation), and the line number for 'PlanSet l = relPlans.get(pi);' if it is not equal to 175
- Make sure to press 'Apply' when you've finished



## **Starting Tools:**

-Ensure the 'Stepping Mode Enabled' checkbox is active

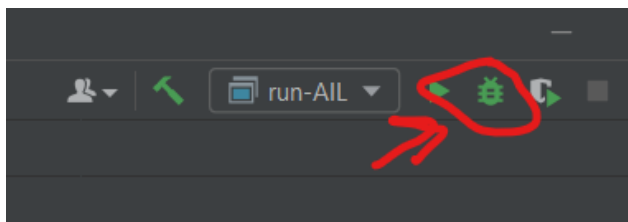


-This makes a breakpoint at the line and file specified in settings  
(rule.apply(this); in AILAgent.java)

-Click on the .ail file you would like to run

-Check you're using the run-AIL run configuration

-Press 'Debug' button in IntelliJ



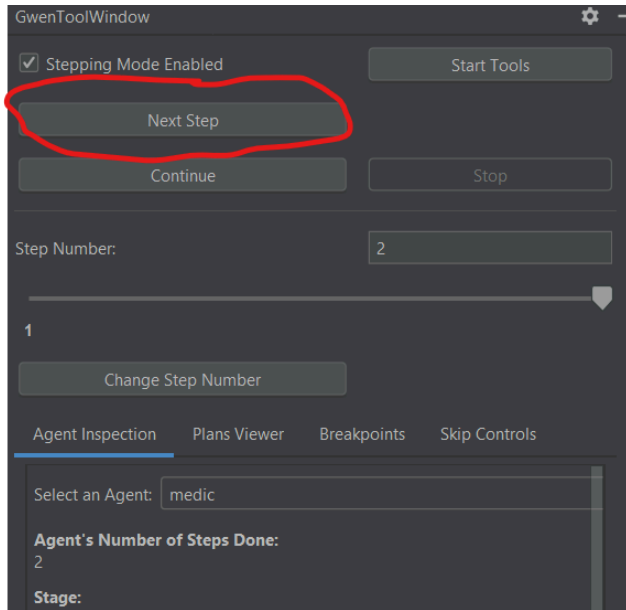
-Press 'Start Tools'

-This starts the tools and links it to the debug session that you've started

-Also displays first step's values in the Agent Inspection tab

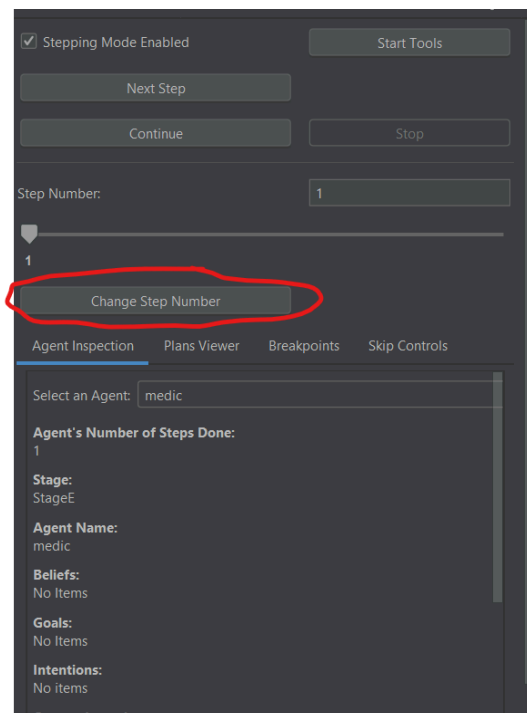
## **Stepping:**

-Press 'Next Step'



- Will advance to the next breakpoint at 'rule.apply(this)' in AILAgent.java
- Thus doing a 'step'
- Updates the values in the Agent Inspection tab

- Change the value on the slider OR change the number in the 'Step Number:' textbox
- THEN press 'Change Step Number'



- This changes the values in the Agent Inspection tab to the values recorded in the step inputted

## **Agent Inspection Tab:**

-Values gathered from the debugger window.

-Agent Dropdown:

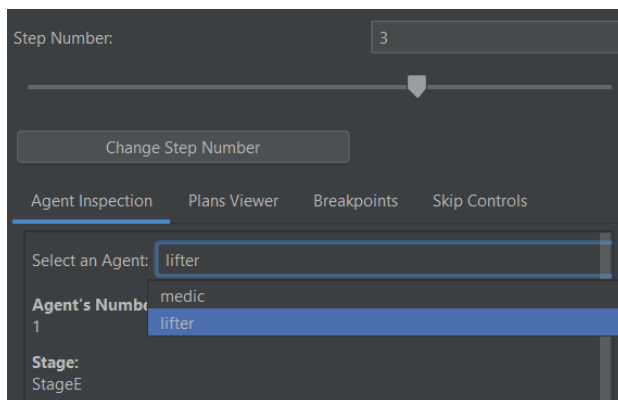
-Clicking the dropdown brings up list of agents in the system

-Clicking on an agent that isn't the 'current agent' (this is the agent that the details are being shown for) will go to the step that agent was last processed at.

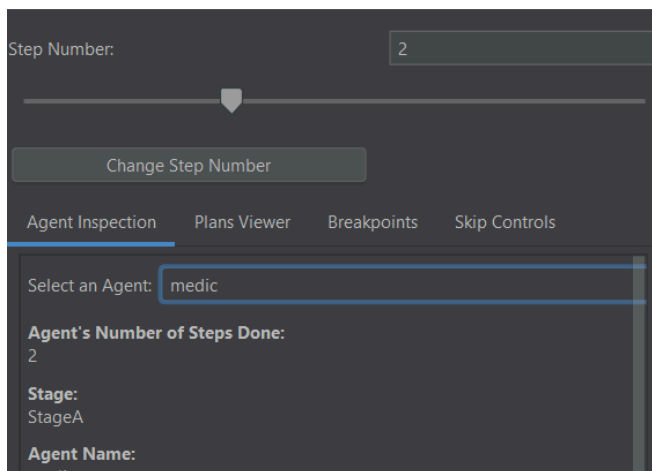
Example:

-At step 2, the current agent is 'medic'

-At step 3, the current agent is 'lifter'



-In step 3, if you click the dropdown and select 'medic' then the Agent Inspection will display step 2 again (the most recent step that processed 'medic')



-Agent's Number of Steps Done

-Records number of steps done for the agent specified in the 'Select an Agent:' dropdown

-Stage:

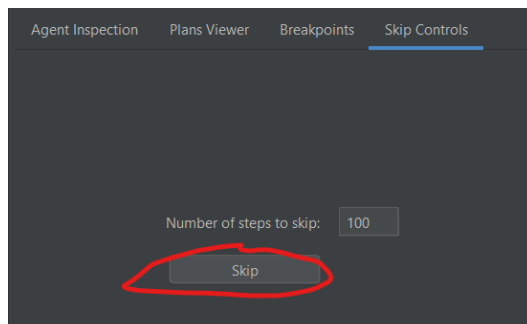
-The stage of the Gwendolen reasoning cycle this step represents

-Agent Name:

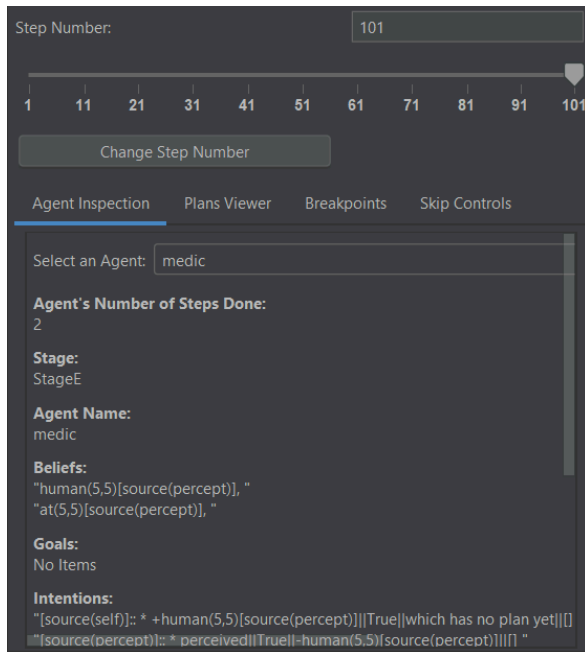
- The name of the agent being processed in this step
- Beliefs:
  - List of beliefs of the current agent
- Goals:
  - List of goals of the current agent
- Intentions:
  - List of intentions of the current agent
- Current Intention:
  - Current Intention of the current agent
- Plans:
  - relPlans at this step
- Inbox:
  - Inbox of messages of the current agent
- Outbox:
  - Outbox of messages of the current agent
- These values change step by step

## **Skip Controls:**

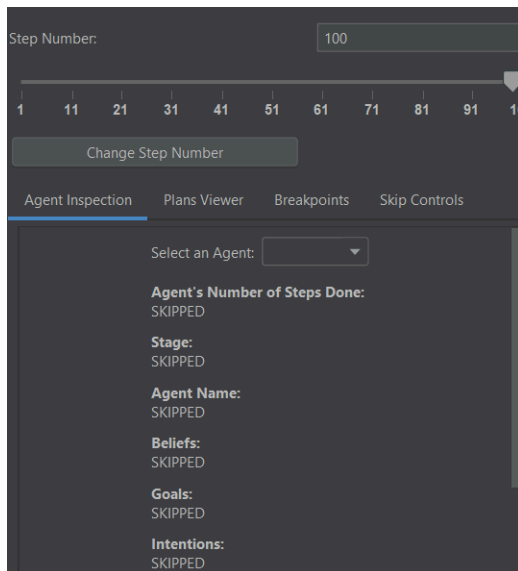
- Press the 'Skip Controls' tab in the window
- Input the number of steps you would like to skip into the text box
- Press the 'skip' button



- This will 'skip' 100 steps
- This will set all values in the Agent Inspection tab to 'SKIPPED' then move to the next step
- Until the last step that it will skip, where it will get the information required for the Agent Inspection tab
- This is what the end result looks like:



-This is what step 100 looks like:

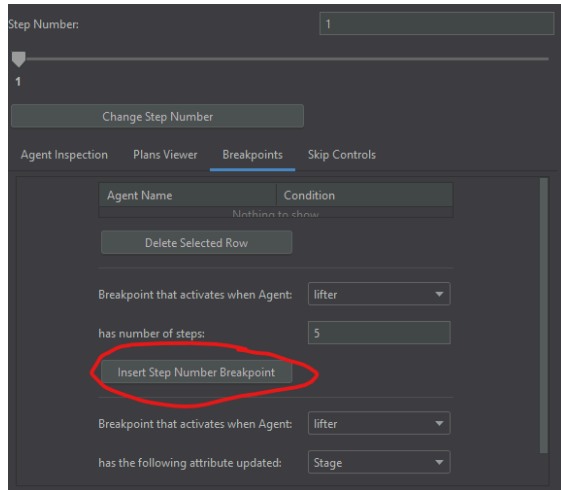


-This will be similar for all steps from step 1

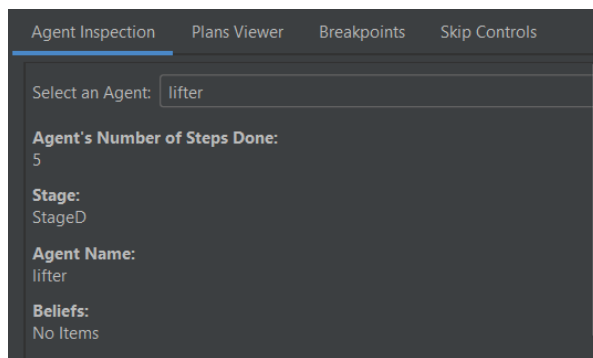
## **Breakpoints:**

*Breakpoint on an Agent's number of steps:*

- Click on the 'Breakpoints' tab
- Click on the dropdown to select which agent you want to set the breakpoint for
- Input the number of steps you want that specific agent to have
- Then press 'Insert Step Number Breakpoint'
- This is where the program will stop



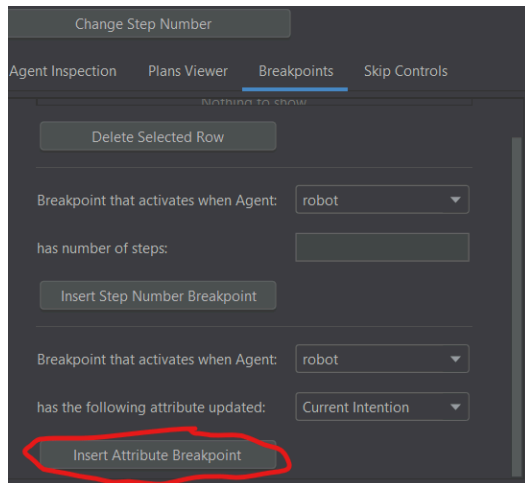
- If you then press continue, the program will gather information for the Agent Inspection tab, move to the next step and repeat
- Until the Agent 'lifter' has number of steps '5'. Won't necessarily be on step number 5.



- You can also press the 'Stop' button to get the program to stop continuing

#### *Breakpoint on an Agent's Attribute:*

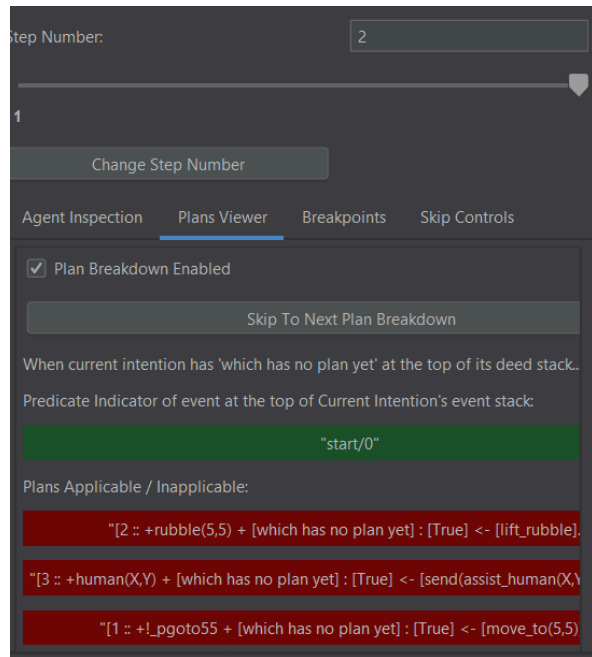
- Click on the 'Breakpoints' tab
- Click on the dropdown to select which agent you want to set the breakpoint for (BELOW THE NUMBER OF STEPS OPTION)
- Click on the dropdown to select which attribute you want to choose the breakpoint for
- Press 'Insert Attribute Breakpoint'
- The program will stop if
  - For the agent you have specified
  - Their value of the attribute you have specified at the current step
  - Is different from its value at the last step the agent was processed at
- So it will stop whenever this attribute is updated for that agent.



## **Plans Viewer:**

- Click on the 'Plans Viewer' tab
- Make sure the 'Plan Breakdown Enabled' tickbox is ticked
  - This sets a breakpoint in a good place for plan inspection
- Then press 'Skip To Next Plan Breakdown'
  - This skips over any breakpoints encountered (for example the one in AILAgent set for stepping mode) until it reaches the breakpoint in PlanLibrary.java
- Will then get the predicate indicator of the event at the top of the Current Intention's event stack when the program reaches a certain point in PlanLibrary.java
- And will get all relPlans
- If the plan in relPlans has a predicate indicator matching the predicate indicator specified above, then the plan is applicable and it's highlighted in green
- Otherwise, it's highlighted in red





## **Syntax Highlighting and Settings for it:**

- Go to File > Settings > Editor > Gwendolen
- Can customise the colours of the 10 groups of tokens
- Will then apply to the .gwen files