Prerequisites

You will need the following installed first:

- Java 8 runtime - http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html
- Eclipse IDE for Java EE - http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/oxygen/R/eclipse-jee-oxygen-R-win32-x86_64.zip
- In Eclipse install the "**IBM Bluemix tools**" and make sure the "**WebSphere Application Server Liberty tools**" option is also ticked when installing -

  http://marketplace.eclipse.org/content/ibm-eclipse-tools-bluemix

Installing

1. Start and open Eclipse
2. Go to help > install new software
3. The install new software window will open - in the top right is an "add" button - click it
4. After clicking "add" you will see the Add Repository window - name the repository "Coding Challenge" and in the URL put "http://public.dhe.ibm.com/ibmdl/export/pub/software/websphere/wasdev/coderally/downloads/eclipseplugin" (without the quotes)
5. Click OK to return to the add new software main window - on the bottom **un-tick the "Group items by category**" option and Select the **first option in the list** - check the box next to it and click "next" to start the install process - you should now be able to continue to install the software. Note that you will get a warning that the content is unsigned.
6. Once the tools are installed, restart Eclipse then open the Java EE perspective (Window> Perspective> Open Perspective> Other> Java EE)
7. In the Java EE perspective go to the servers view which should be in the bottom section of the Java EE perspective.
8. Right-click on empty space in the Servers view and select new>server
9. In the New Server window find the IBM folder and select "WebSphere Application Server Liberty" and click next
10. You will now see the new server runtime wizard. If you already have an installed copy of WebSphere Liberty, then set the directory location and move to step 12. If you have not installed Liberty before then choose "install from an archive or a repository" and click next.
11. Set a location where you would like to install WebSphere Liberty - it is a small install, but will need to be able to write logs in its install location, so Windows users should **not** install it under "program files". Once you have set an install location, choose to download and install a new environment and select the **WAS Liberty with Web Profile option** - you can then continue the wizard to accept the license agreement and then install the runtime - you do not need any additional add-ons from the add-ons screen.
12. Once the runtime is installed you should see the new server window again - this time click "next" when the "WebSphere Application Server Liberty" option is selected.
13. We will be using the default settings for the Liberty server so you can click the "finish" button immediately and a new Liberty server will be created.

- Install the code Rally WAR to your local server as per the first part of these instructions - https://www.ibm.com/developerworks/community/blogs/code-rally/entry/beta_installer?lang=en
- Once you have done this you should have a working server on your local instance and be able to go to http://localhost:9080/CodeRallyWeb
- Now you can create a car

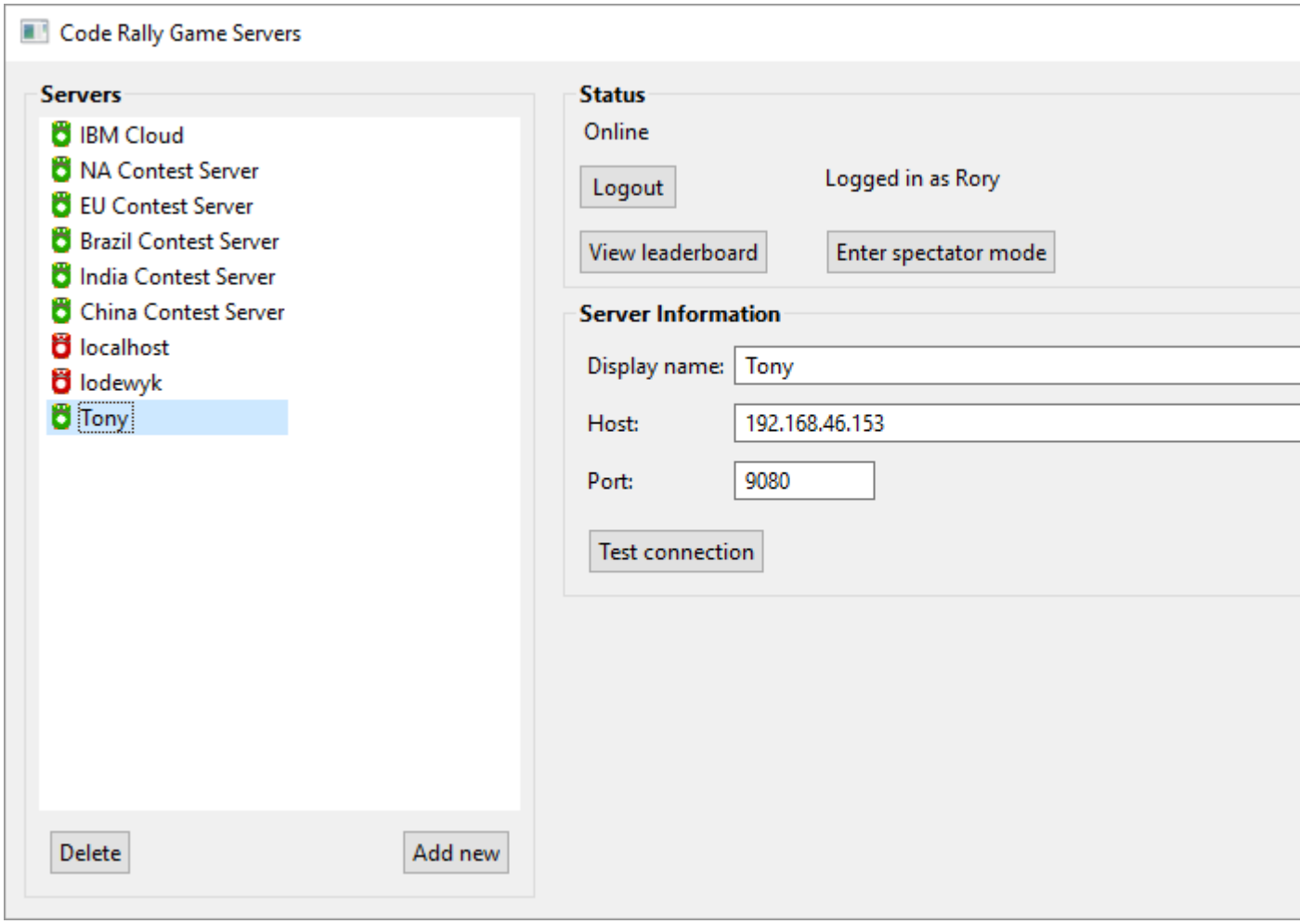
There are 3 options to create and run a car in a race.

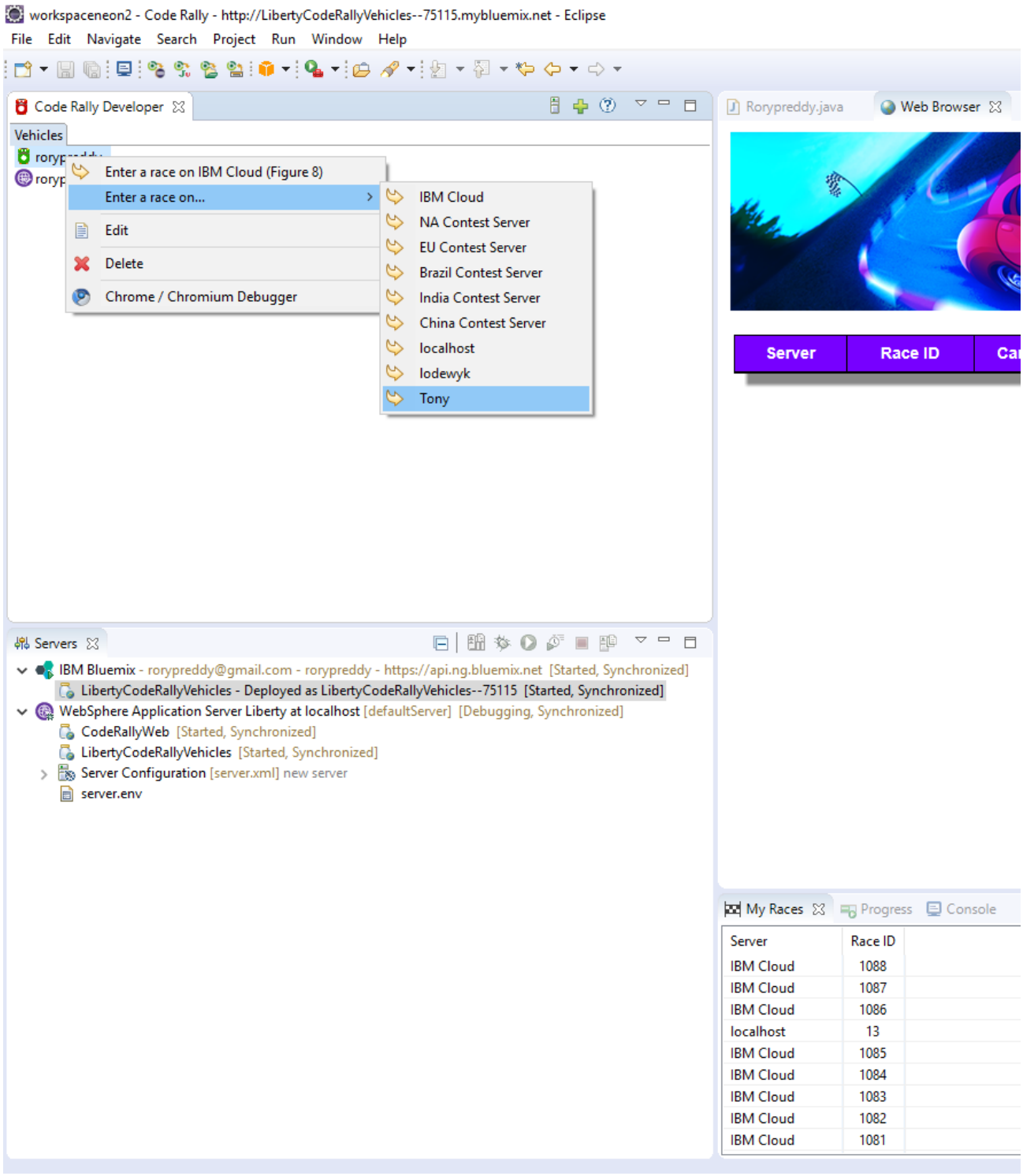
- Option 1 is called "**intermediate**" and is a java file deployed to the race server.
- Option 2 is called "**Agent**" and is a Microservice application that is linked to the race server and send json to handle events.
- Option 3 is where you deploy option 2 to the **bluemix** cloud and basically let it race 24/7

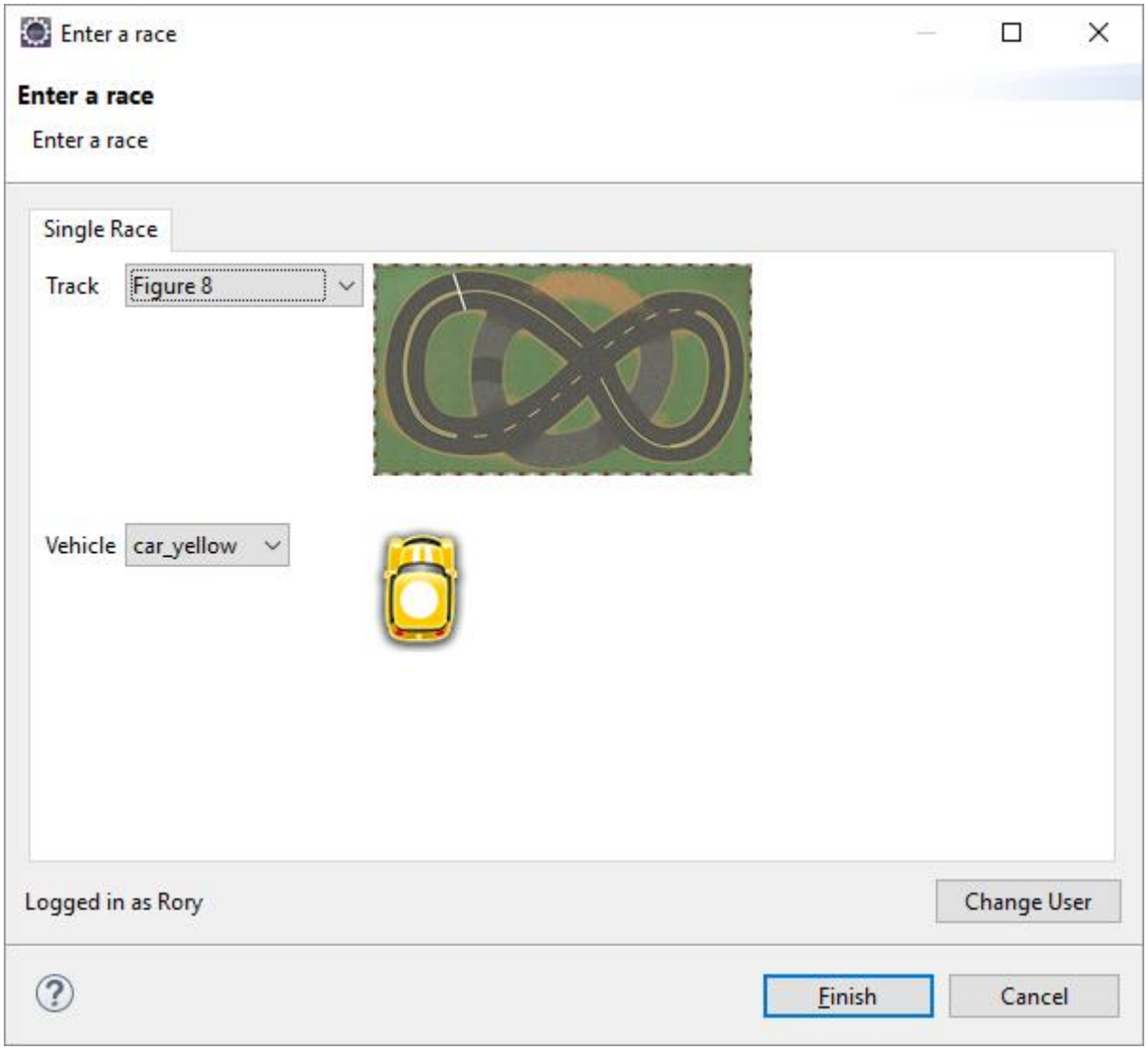
Option 1 – Code a local CAR and deploy it to a central server - https://www.ibm.com/developerworks/community/blogs/code-rally/entry/zero_to_racing_in_60_seconds?lang=en

"**Local**" Server Setup

- Create a new Game server
    - Add the display name, host and port.
    - Add your username Login
    - Test the connection
    - Make sure you can view the leader board and enter spectator mode.
- Enter a Car locally to make sure the server runs locally
- Create the server config on each players pc's and get them to create and deploy a local car
- You can edit the following file to make the server wait longer that 30 seconds to start a game - C:\Users\bbdnetXXXX\user\.coderally\serverConfig.xml
    - Change MAX_PENDING_TIME to the time you want them to wait until the game starts

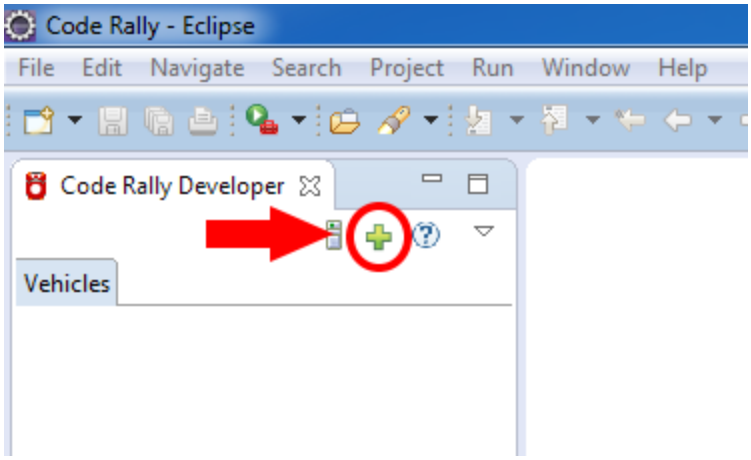

Code Rally Game Servers
Servers
IBM Cloud
NA Contest Server
EU Contest Server
Brazil Contest Server
India Contest Server
China Contest Server
localhost
lodewyk
Tony
Delete
Add new
Status
Online
Logout
Logged in as Rory
View leaderboard
Enter spectator mode
Server Information
Display name:
Tony
Host:
192.168.46.153
Port:
9080
Test connection


workspaceneon2 - Code Rally - http://LibertyCodeRallyVehicles--75115.mybluemix.net - Eclipse
File Edit Navigate Search Project Run Window Help
Code Rally Developer
Vehicles
Enter a race on IBM Cloud (Figure 8)
Enter a race on...
Edit
Delete
Chrome / Chromium Debugger
IBM Cloud
NA Contest Server
EU Contest Server
Brazil Contest Server
India Contest Server
China Contest Server
localhost
lodewyk
Tony
Rorypreddy.java
Web Browser
Server
Race ID
Servers
IBM Bluemix - rorypreddy@gmail.com - rorypreddy - https://api.ng.bluemix.net [Started, Synchronized]
LibertyCodeRallyVehicles - Deployed as LibertyCodeRallyVehicles--75115 [Started, Synchronized]
WebSphere Application Server Liberty at localhost [defaultServer] [Debugging, Synchronized]
CodeRallyWeb [Started, Synchronized]
LibertyCodeRallyVehicles [Started, Synchronized]
Server Configuration [server.xml] new server
server.env
My Races
Progress
Console
Server
Race ID
IBM Cloud 1088
IBM Cloud 1087
IBM Cloud 1086
localhost 13
IBM Cloud 1085
IBM Cloud 1084
IBM Cloud 1083
IBM Cloud 1082
IBM Cloud 1081

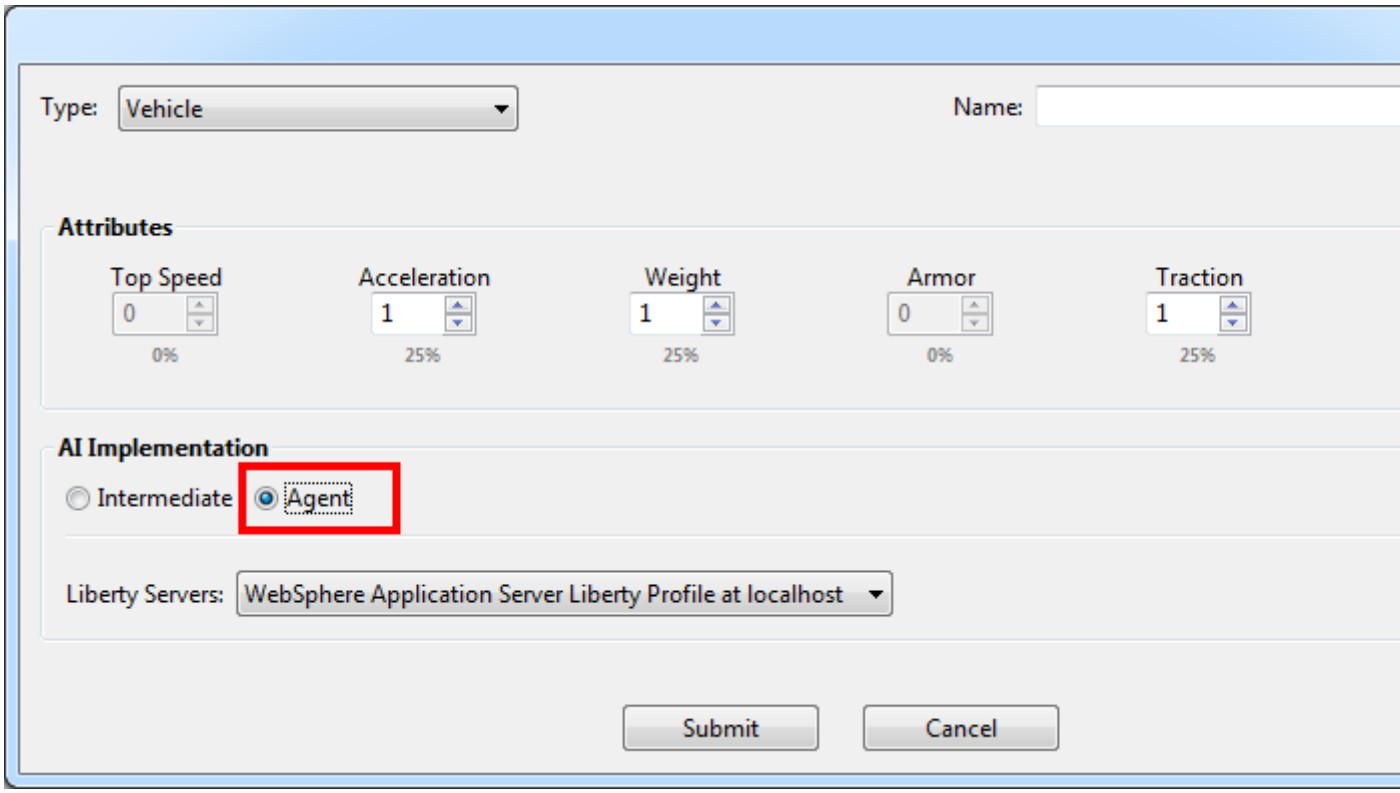


Option 2 – Create an "AI" car by deploying a local Microservice Application that will send JSon to the "Race" server. -
https://www.ibm.com/developerworks/community/blogs/code-rally/entry/Running_your_agent_AI_on_your_laptop?lang=en

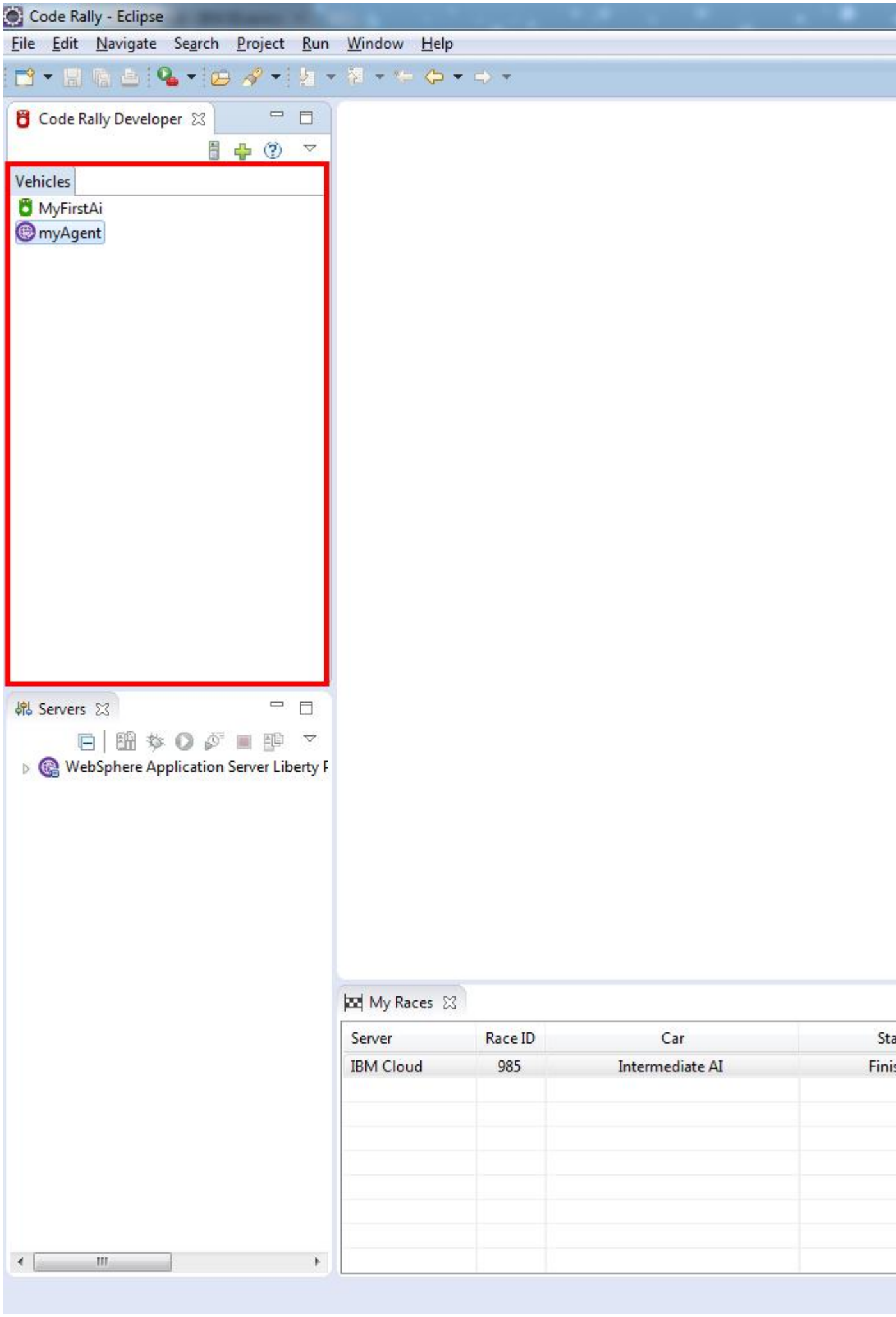
- You need to have a liberty server already setup as per the previous section.
- Once that is done you can go to the new AI window by clicking the green "+" above your vehicles list:



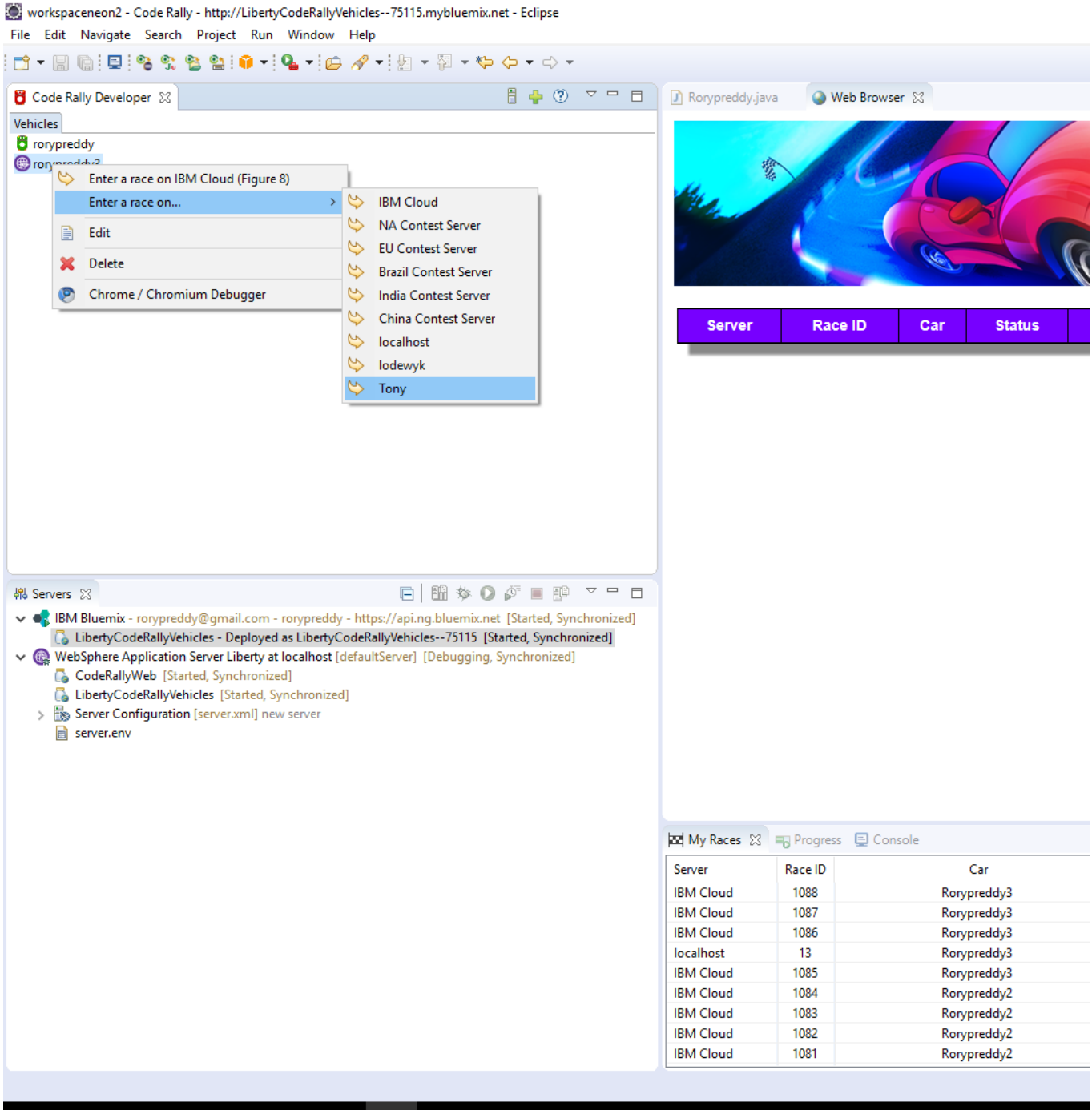
- In the new AI window give your AI a name (this is a Java class name so class naming rules apply), change the radio button to "agent", make sure the Liberty profile server is in the dropdown and click "submit":



- Your AI will now appear in the "vehicles" list on the left side of Eclipse:


Code Rally - Eclipse
File Edit Navigate Search Project Run Window Help
Code Rally Developer
Vehicles
MyFirstAi
myAgent
Servers
WebSphere Application Server Liberty F
My Races
Server
Race ID
Car
IBM Cloud
985
Intermediate AI

- Double-left click on your agent in the list and it will open the code for your agent AI.
- Right click on the AI Agent and select the server to **link** it to



- You can find out more about what you can do with the code in our [creating your AI article].
- One of big changes when running races with an Agent AI is that races with agents in them run in real time (so you can watch a race live when entering) and you can change your code as the race is running - as long as it compiles hit save and the change will impact all races the AI is currently racing in
    - (note: the live race window can be a few seconds behind what is happening).

- This is possible thanks to WAS Liberty being able to dynamically update web applications on the fly without restarting the runtime.
- The other advantage is that you can use System.out and System.err logging to print out information to your local Eclipse client. Don't use debug mode though as the game will not wait for your AI to respond if it's paused waiting for you in debug mode.

Option 3 – Deploy your Car to its own cloud server

https://www.ibm.com/developerworks/community/blogs/code-rally/entry/never_not_racing_creating_a_code_rally_cloud_racer_on_bluemix?lang=en