

RMI Example V2

RMI Example V2

- The goal of this example is to show the *Dynamic Class Loading* feature of Java
- The Book class in the Server is not in the compilation unit of the client and is downloaded at runtime using the `java.rmi.server.codebase` property

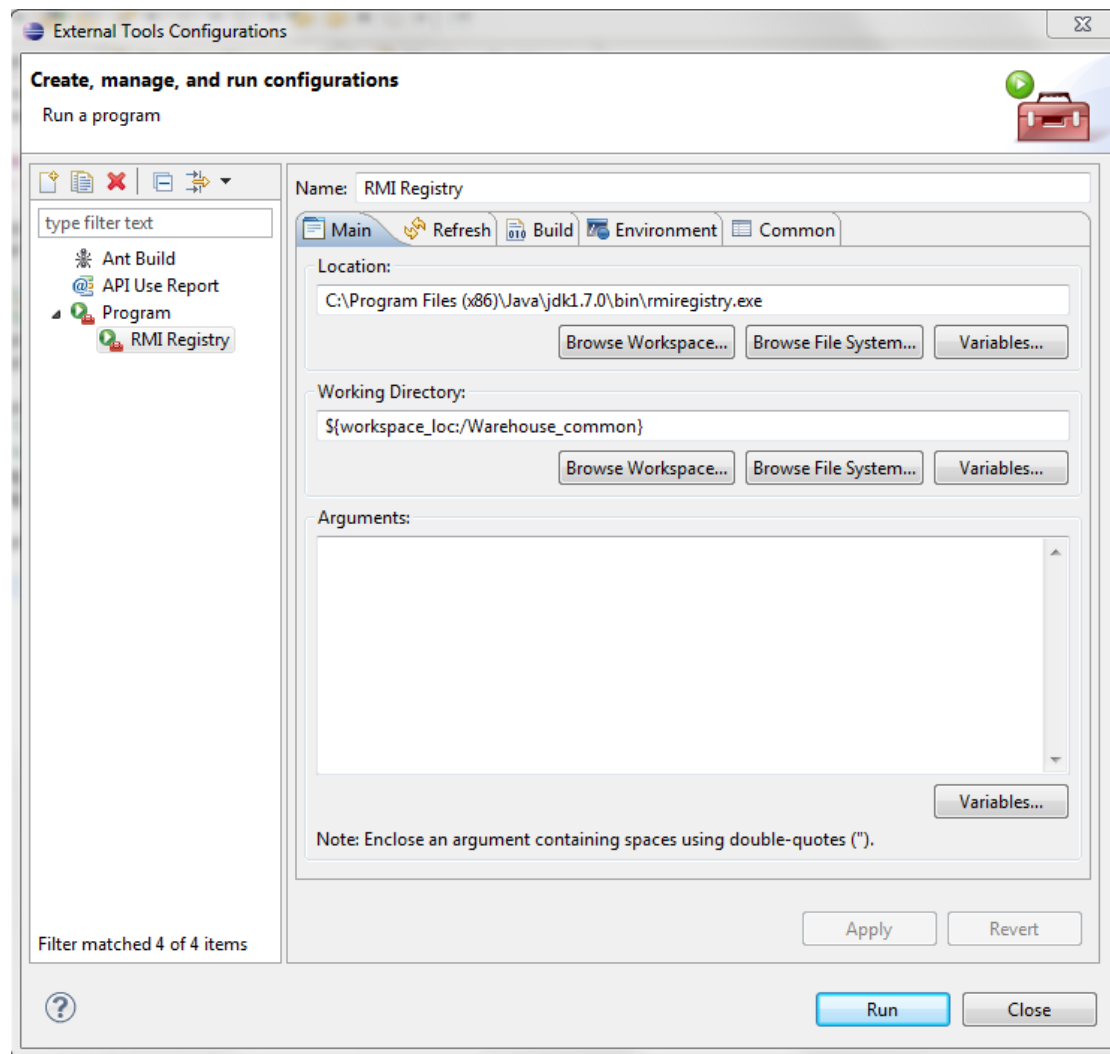
About the project setup

- The example is divided into three projects
 - Warehouse_client
 - Contains the code related to the client
 - Warehouse_common
 - Contains the code shared between server and client. It is imported by both the client project and the server project in order to compile
 - Warehouse_server
 - Contains the code related to the server

Running the example

- To run the example
 1. Setup rmiregistry location
 2. Start rmiregistry
 3. Start NanoHTTPD COMMON
 4. Start NanoHTTPD SERVER
 5. Start Server
 6. Start Client

Setup rmiregistry location



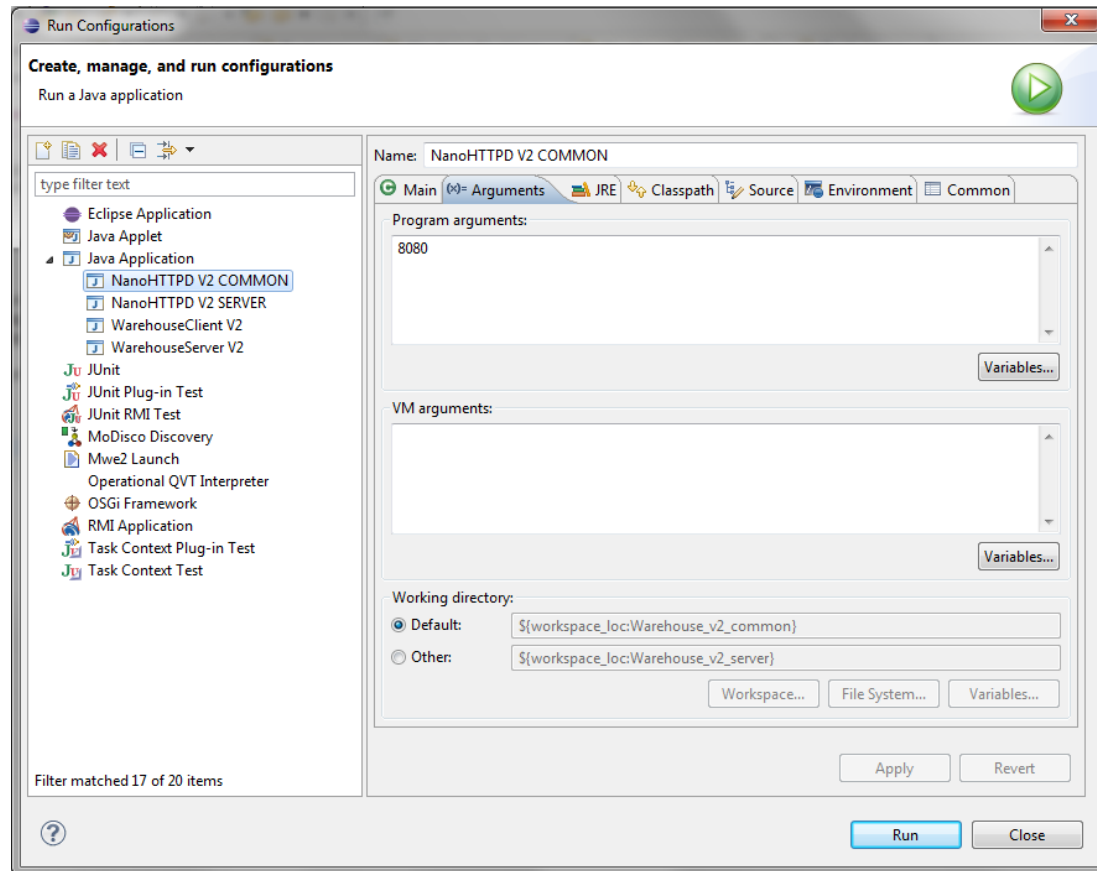
Start rmiregistry

- Click on the Eclipse external tools button
- Run the RMI Registry setup created previously



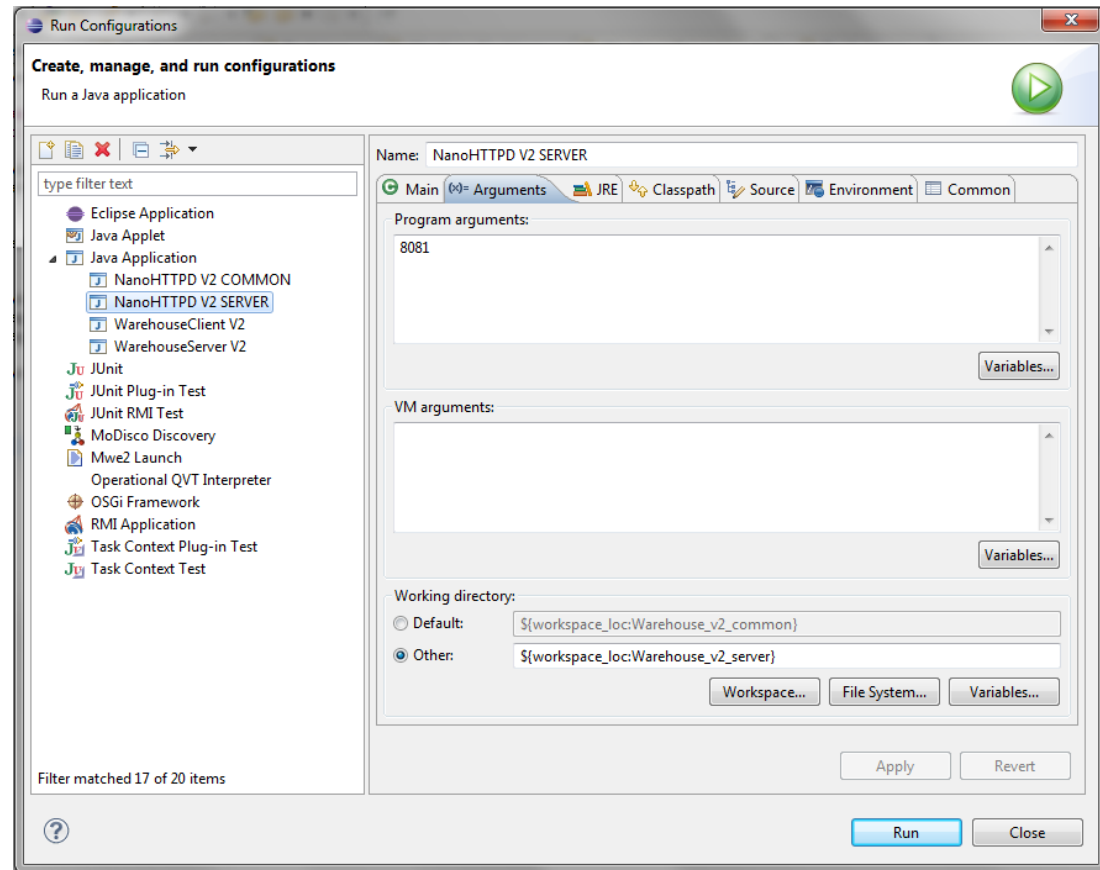
Start NanoHTTPD COMMON

- Go to the run configurations menu and run NanoHTTPD COMMON
- This code repository will be used by the RMI registry during the binding phase



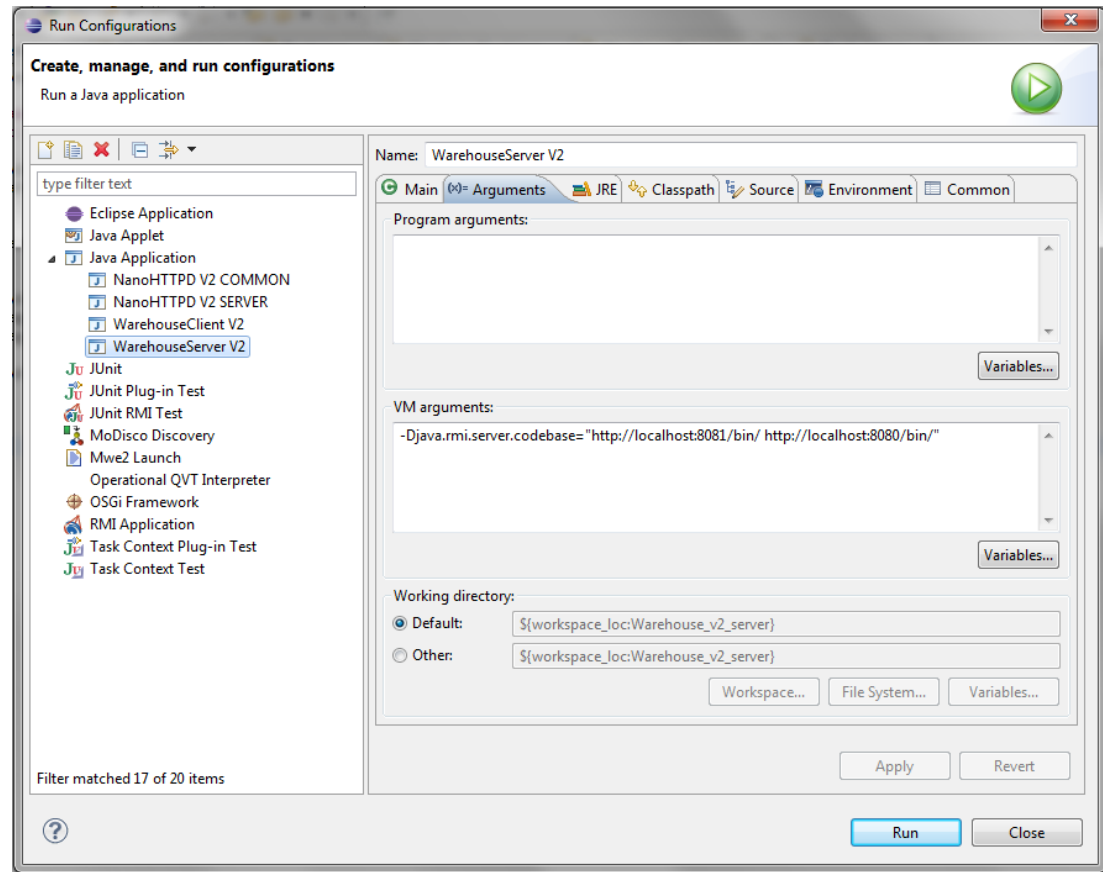
Start NanoHTTPD SERVER

- Go to the run configurations menu and run NanoHTTPD SERVER
- This code repository will be used by the Client to download the Book class



Start Server

- Go to the run configurations and run WarehouseServer V2
- This time we have two codebases where the code can be downloaded



Start Client

- Go to the run configurations and run WarehouseClient V2
- No additional argument is needed

