Service Integration: How to use the Development Container
Table of Contents

• The Development Container
• Using the UI
• Eclipse
• Container Persistence
• Using the Terminal
The Development Container

• Docker based development environment for IIP-Ecosphere
  • Enables the development on every platform that can run Docker
  • There is no need for outside installation
  • The container is portable
  • Container runs Ubuntu 20.04
Getting the Container

• On first start:
  • Make sure docker is running (Windows, depending on installation)
  • Run "docker run -p 6080:80 -v /dev/shm:/dev/shm iipecosphere/dev-container:0.2" to get the image from docker hub and start it
  • Start a browser and type "localhost:6080" in the upper search bar
• The development container
• **Using the UI**
• Eclipse
• Container Persistence
• Using the Terminal
Using the Container UI (1)

- Start a browser and type "localhost:6080" in the upper search bar
- To make the image fill the screen: set "Skalierungsmodus" to "Lokales skalieren"
Using the Container UI (2)

- Changing the resolution:
  - The container UI will take the screen resolution available when loading the UI in the browser for the first time after the container start
  - It cannot be changed otherwise
Table of Contents

• The development container
• Using the UI
• **Eclipse**
• Container Persistence
• Using the Terminal
• A specific Eclipse is installed
  • Start Eclipse

• It will NOT ask for a workspace on start
• Initially the workspace is empty
• Import the project from “/root/Desktop/eclipse-workspace”
Eclipse: Import Project (1)

- Import the project from "/root/Desktop/eclipse-workspace"
• Import the project from "/root/Desktop/eclipse-workspace"
# Table of Contents

- The development container
- Using the UI
- Eclipse
- **Container Persistence**
- Using the Terminal
Container Persistence

- To keep the changes done the container needs to be stopped
  - Use "docker ps" to list all containers
  - Use "docker stop <containerID>" to stop the container

- To re-run the container
  - Use "docker ps -a" to list all containers including the stopped once
  - Use "docker start <containerID>"
  - Load "localhost:6080" in a browser to see the container UI
Table of Contents

• The development container
• Getting the Container
• Using the UI
• Eclipse
• Container Persistence
• Using the Terminal
• Even though Eclipse has a Maven integration it tends to work best to directly use the terminal
  • Does not work!
Using the Terminal (2)

- You need to navigate to the files via explorer
  ![Image of file explorer]

- When in the correct directory
  ![Image of terminal with directory structure]
If Eclipse does not reflect the change directly after successfully executing a Maven command in the terminal

Select the project in question

If there are errors present:
try Maven update

If a directory does not show up: try refresh
To get files into the container utilise the command “docker cp”

Syntax: “docker cp <src_path> <dest_path>”

One of the two paths should be inside the container and is given as “<containerID>:path”

When docker is installed using Windows Subsystem Linus you might need to add “wsl” in front of the command

From container to Windows: start “<dest_path>” with “/mnt/” as this is how WSL sees the filesystem. E.g. “/mnt/C/work/…”

Example:

wsl docker cp C:/work/Text.txt 4d39848b0dff:/root/Desktop
Summary

- What we learned
  - How to start and utilise the development container
- How to go on
  - How to edit datatypes