Service Integration: How Deploy an Application
Deploy the Application

1. **Configure Datatypes**
   - Determine needed types, describe them
2. **Configure Services**
   - Configure basic service attributes
   - Configure logging
3. **Configure Application**
   - Define the application
   - Define the service mesh
   - Create the templates
4. **Build the Application**
   - Implement the service functionality
   - Install the services
   - Create the templates
   - Build the application
5. **Test the Services**
   - Use generated tests
6. **Deploy the Application**
   - Add application
   - Start up a platform
   - Run the application
<table>
<thead>
<tr>
<th>Table of Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Prerequisites</td>
</tr>
<tr>
<td>• Building an Application</td>
</tr>
</tbody>
</table>
Prerequisites

• Required:
  • Installed the platform and its dependencies or the development container
  • Installed the IDE for IIP-Ecosphere Platform (provided Eclipse Version)
  • How to configure datatypes
  • How to configure services
  • How to configure an application
  • How to build an application
  • How to test an application

• Optional:
  • None
Table of Contents

- Prerequisites
- Starting the platform
Deploy the application on a local platform

- We assume a build platform
- In "tools/Install/gen/broker" start the broker.bat
- In "tools/Install/gen" start in order (always wait till the process says its running)
  - platform.bat
  - ecs.bat
  - serviceMgr.bat
- cli.bat
  - Run "resources list"
  - Back out of resources with ".." Run "services <resourceID>
  - Run "add file://<pathToBuild-bin.jar>" (including the "-bin.jar" itself)
  - Run "listArtifacts"
  - Run "startAll <artifactID>"
Observe the application on a local platform

- On the startup each service will list where its log files will be located
  - They are in “/tmp/<numbers>/”
    - Each directory contains a stdout_0.log and a stderr_0.log
    - Java will only show exceptions in the stderr and outputs in stdout
    - Python will show outputs as well as errors in the stderr
Summary

• What we learned
  • How to start a platform locally
  • How to deploy our application on it
• How to go on
  • You are done
  • Optional: How to create an application with Python services