TECHILA

Techila with Java

Contents

- **Short introduction of the Java Peach syntax**
  - Minimalistic syntax and optional parameters
- **Preparation**
  - Updating the CLASSPATH environment variable
- **Testing**
  - Running an example application included in the Techila Grid Managent Kit

Introduction of the Java peach syntax

- **Simple interface for creating Projects**
  - Accessed by adding the following imports:
    - import fi.techila.grid.management.oma.client.GridManagerFactory;
    - import fi.techila.grid.management.oma.client.GridManager;
    - import fi.techila.grid.management.oma.client.Peach;
    - import fi.techila.grid.management.oma.client.Support;
  - Or to reduce typing:
    - import fi.techila.grid.management.oma.client.*;
  - Information can be found in the Javadocs
    - Located in the 'doc/javadoc' folder in the Techila Grid Management Kit

Introduction of the Java peach syntax

- **Typically the Java peach syntax will define:**
  - The Java class file that will be executed on Workers
    - Class files can also be executed from a JAR file
  - Input arguments for the Java class file
    - Defined with peach.putExeExtras()
  - Files that will be transferred to the participating Workers
    - Defined with peach.addParamFile()
  - Number of Jobs in the Project
    - Defined with peach.setJobs();
  - Location of the grid_settings.ini file
    - Defined with grid.initFile()
- **Additional features can be implemented with optional parameters**
  - For example, enable streaming with command setStream(true)

Introduction of the Java peach syntax

- **Java Peach syntax for creating a Project**
  - Entire source code located at <full path>\gmk\examples\java\peach

```
statuscode = grid.initFile("../../../grid_settings.ini"); // Location of the configuration file
checkStatus(statuscode);                                  // Check that everything is OK
Peach peach = grid.newPeach(NAME);                        // Create peach object.'NAME' defined
peach.setMessages(true);                                  // Enable messages
peach.addExeFile("file1", "MCPIClient.class");            // Bundle the "Example.class" file
peach.setExecutable("../../jre/bin/java;osname=Linux,"    // Define where Java is located
                    + "..\\\\..\\\\jre\\\\bin\\\\java.exe"
                    + ";osname=Windows");
peach.putExeExtras("Parameters", "MCPIClient %P(jobidx) %P(loops) %O(output)"); // Java params
peach.putProjectParam("loops", "10000000");                   // Value for the loops parameter
peach.putExeExtras("OutputFiles", "output;file=custom_file"); // Name of the output file
peach.addParamFile("datafile.txt");                           // Transfer datafile.txt
peach.setJobs(10);                                            // Number of Jobs
peach.execute();                                              // Starts the Project
```

- **Command executed on Workers (for Job #1)**

```
java MCPIClient 1 10000000 custom_file
```

Introduction of the Java peach syntax

- **Post processing of result files:**
  - All result files are read and the two integer values in the files are summed

```
String file;
while ((file = peach.nextFile()) != null) { // Returns the path and name of the result file
FileInputStream fis = new FileInputStream(file);
DataInputStream dis = new DataInputStream(fis);
int in = dis.readInt();                     // Read the first integer (number of points inside)
int loops = dis.readInt();                  // Read the second integer (total number of points)
totalin += in;                              // Update total counters based on Job results
totalloops += loops;
dis.close();
fis.close();
}
peach.done(); // Must be called after the Project is complete. Clean up and release resources.
```

Preparation

- **Modifying the CLASSPATH environment variable**
  - The CLASSPATH variable must contain the following paths
    - The current working directory i.e. "."
    - The gridmgmt.jar file located at <full path>\gmk\grid\gridmgmt.jar
- **Windows**
  - set CLASSPATH=%CLASSPATH%;<full path>\gmk\grid\gridmgmt.jar;.
- **Linux**
  - export CLASSPATH=$CLASSPATH:<full path>/gmk/grid/gridmgmt.jar:.
- **Will need to be set again if a new terminal is launched**
  - Add to the environment variables or shell startup scripts to make changes automatic

Creating a test Project

1. **Launch a Command Prompt / Terminal**

2. **Change current working directory to:**

   <full path>\gmk\examples\java\peach

3. **Compile MCPIClient.java with command:**

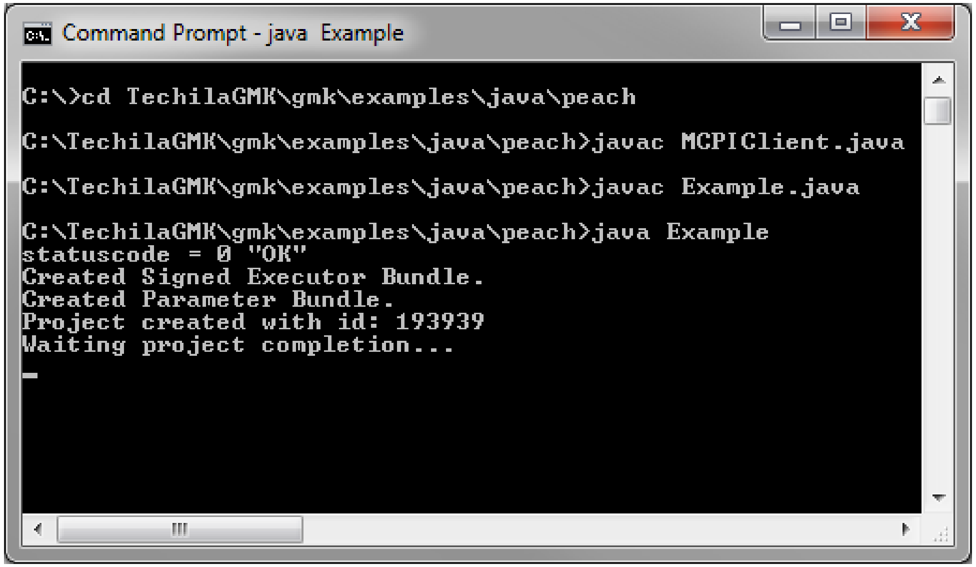
   ```
   javac MCPIClient.java
   ```

4. **Compile Example.java with command:**

   ```
   javac Example.java
   ```

5. **Create the Project with command:**

   ```
   java Example
   ```



```
C:\>cd TechilaGMK\gmk\examples\java\peach

C:\TechilaGMK\gmk\examples\java\peach>javac MCPIClient.java

C:\TechilaGMK\gmk\examples\java\peach>javac Example.java

C:\TechilaGMK\gmk\examples\java\peach>java Example
statuscode = 0 "OK"
Created Signed Executor Bundle.
Created Parameter Bundle.
Project created with id: 193939
Waiting project completion...
_
```

WWW.TECHILA.FI