



Techila with Python

Contents

- **Short introduction of the Python peach syntax**
 - Minimalistic syntax and optional parameters
- **Preparation**
 - Creating the GridManagement.conf file (not always required)
 - Installing the numpy package (required in the 'asian' example)
- **Creating a test Project**
 - Running the 'asian' example in the Techila Grid Management Kit

Introduction of the Python peach syntax

▪ Simple interface for creating Projects

- In the current version, commands parsed with a dynamic-link library (.dll) for Windows or dynamically linked shared object libraries (.so) for Linux
- Plans for a new implementation
 - Will make more features available

▪ Typically the Python peach syntax will define:

- The function that will be executed on Workers
- The Python-script containing the executable functions
- Input arguments for the executable function
- Files that will be transferred to the participating Workers
- Number of Jobs in the Project
- Location of the gmk directory

Introduction of the Python peach syntax

- Example of a Python peach syntax:

```
result = peach("test_dist",           # Name of the executable function
               [a,b,"<param>"],       # Input arguments
               ["test_dist.py",       # File given to 'execfile' on Workers
                "datafile.txt"],      # Transfer 'datafile.txt' to Workers
               range(0,10),           # Number of Jobs set to 10
               gmRoot="../../../..")   # Path of the 'gmk' directory
```

Introduction of the Python peach syntax

- Streaming and Callback functions can be enabled with additional parameters:

```
result = peach("test_dist",           # Name of the executable function
               [a,b,"<param>"],       # Input arguments
               ["test_dist.py",       # File given to 'execfile' on Workers
                "datafile.txt"],      # Transfer 'datafile.txt' to Workers
               range(0,10),           # Number of Jobs set to 10
               gmkRoot="../../../",    # Path of the 'gmk' directory
               streamResults=True,    # Enable streaming
               callbackMethod=callbackF) # Name of the callback function
```

Preparation

- **Creating the GridManagement.conf file**

- This step is typically required if you receive the following error message when running the example: "RuntimeError: ('Grid error', -22, 'Unable to find Java runtime library')"

- **Steps:**

1. Navigate to the <full path>\gmk\grid folder
2. Rename the '**GridManagement.conf.example**' to '**GridManagement.conf**'
3. Open the '**GridManagement.conf**' file with a text editor
4. Enter path of the jvm.dll file as the value of the JVMPATH. An example is shown below:

JVMPATH=C:\Program Files\Java\jdk1.6.0_24\jre\bin\server\jvm.dll

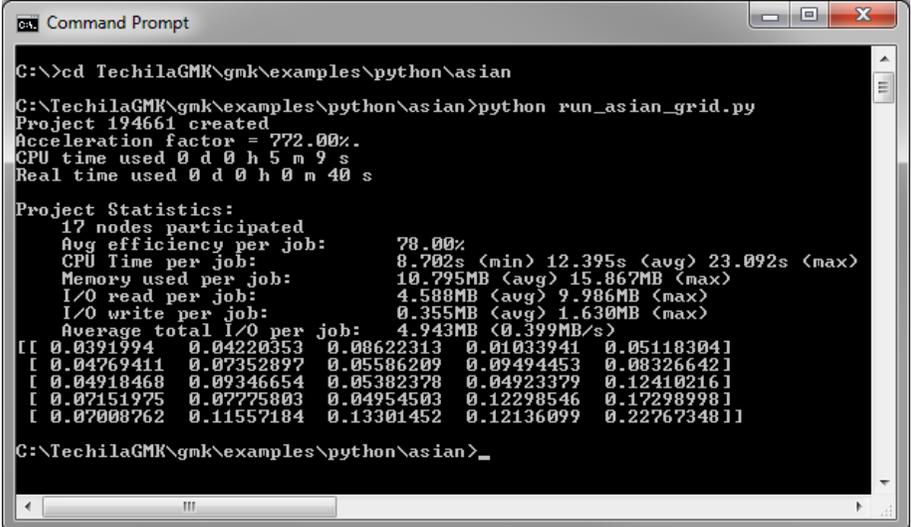
5. Save the file

Preparation

- **Installing the 'numpy' package**
 - This step is only required if you do not have 'numpy' installed.
 - Download links available at: <http://www.scipy.org/Download>
 - Select and install the package compatible with your system
 - When applicable in Linux, use apt-get install python-numpy

Creating a test Project

1. Launch a Command Prompt / Terminal
2. Change your current working directory to:
 <full path>\gmk\examples\python\asian
3. Create the Project with command:
`python run_asian_grid.py`
4. When prompted, enter your password
5. The computational Project will be created
 - Progress will be visible in the status bar



```

C:\>cd TechilaGMK\gmk\examples\python\asian
C:\TechilaGMK\gmk\examples\python\asian>python run_asian_grid.py
Project 194661 created
Acceleration factor = 772.00%.
CPU time used 0 d 0 h 5 m 9 s
Real time used 0 d 0 h 0 m 40 s

Project Statistics:
 17 nodes participated
Avg efficiency per job:      78.00%
CPU Time per job:          8.702s (min) 12.395s (avg) 23.092s (max)
Memory used per job:       10.795MB (avg) 15.867MB (max)
I/O read per job:          4.588MB (avg) 9.986MB (max)
I/O write per job:         0.355MB (avg) 1.630MB (max)
Average total I/O per job: 4.943MB (0.399MB/s)
[[ 0.0391994  0.04220353 0.08622313 0.01033941 0.051183041
  0.04769411 0.07352897 0.05586209 0.09494453 0.083266421
  0.04918468 0.09346654 0.05382378 0.04923379 0.124102161
  0.07151975 0.07775803 0.04954503 0.12298546 0.172989981
  0.07008762 0.11557184 0.13301452 0.12136099 0.227673481]]
C:\TechilaGMK\gmk\examples\python\asian>_
  
```

WWW.TECHILA.FI