Pan version 2.2

www.kettle.be

Table of contents

1 PAN

1.1 What is Pan?

Pan is a program that can execute transformations designed by Spoon in XML or in a database repository. Usually transformations are scheduled in batch mode to be run automatically at regular intervals.

1.2 Installation

The first step is the installation of Sun Microsystems Java Runtime Environment version 1.4 or higher. You can download a JRE for free at http://www.javasoft.com/.

After this, you can simply unzip the zip-file: Kettle-2.0.zip in a directory of your choice.
In the Kettle directory where you unzipped the file, you will find a number of files.
Under Unix-like environments (Solaris, Linux, MacOS, …) you will need to make the shell scripts executable. Execute these commands to make all shell scripts in the Kettle directory executable:

```
cd Kettle
chmod +x *.sh
```

1.3 Launching Pan

To launch Pan on the different platforms these are the scripts that are provided:

- ✓ pan.bat: run Pan on the Windows platform.
- ✓ pan.sh: run Pan on Unix platforms and Mac OSX

Other platforms can be supported too, just write to info@kettle.be if you have a need for this.

1.4 Command line options

These are the command line options that you can use.
Fields in *italic* represent the values that the options use.

```
-file=filename
```

This option runs the transformation defined in the XML file. (.ktr : Kettle Transformation)

```
-log=Logging Filename
```

Specifies the log file. The default is the standard output.

```
-level=Logging Level
```

The level option sets the log level for the transformation that's being run.
These are the possible values:

- ✓ Error: Only show errors
- ✓ Nothing: Don't show any output
- ✓ Minimal: Only use minimal logging
- ✓ Basic: This is the default basic logging level
- ✓ Detailed: Give detailed logging output
- ✓ Debug: For debugging purposes, very detailed output.
- ✓ Rowlevel: Logging at a row level, this can generate a lot of data.

```
-rep=Repository name
```

Connect to the repository with name "*Repository name*".
You also need to specify the options –user, –pass and –trans.

```
-user=Username
```

This is the username with which you want to connect to the repository.

```
-pass=Password
```

The password to use to connect to the repository

```
-trans=Transformation Name
```

Use this option to select the transformation to run from the repository

NOTE: *It's important that if spaces are present in the option values, you use quotes or double quotes to keep them together. Take a look at the examples below for more info.*

```
-listdir=Y
```

Print a listing of all the sub-directories in the repository directory specified with the option "-dir".

```
-dir=directory
```

Specify the directory in the repository to use.

```
-listtrans=Y
```

Show a list of all the transformations in the repository directory specified with the option "-dir".

```
-listrep=Y
```

Print a listing of all the defined repositories.

1.5 Path

Please make sure that you are positioned in the Kettle directory before running the samples below. If you put these scripts into a batch file or shell script, simply do a change directory to the installation directory:

If Kettle was installed on windows on the D:\ drive

```
D:
cd \Kettle
```

If Kettle was installed in the /product directory on a Unix system:

```
cd /product/Kettle/
```

1.6 Run a transformation from file

This example runs a transformation from file on a windows platform:

```
pan.bat -file="D:\Transformations\Customer Dimension.ktr" -level=Basic
```

This example runs a transformation from file on a Linux box:

```
pan.sh –file=“/PRD/Customer Dimension.ktr” –level=Minimal
```

1.7 Run a transformation from Repository

This example runs a transformation from the repository on a windows platform:
(Enter on a single line without returns…)

```
pan.bat –rep=“Production Repository”
        -trans=”update Customer Dimension”
        -dir=”\Dimensions\”
        -user=”matt”
        -pass=”somepassword123”
        –level=Basic
```

1.8 Redirecting output

If you don’t want the output of the file to appear on the screen but rather be put into a log file, you can use redirection.

This example adds the Pan output to an ever-growing log file:

```
pan.sh –file=“/PRD/trans.ktr” –level=Minimal >> /LOG/trans.log
```

This example writes the Pan output to a file that gets overwritten every time:

```
pan.bat –file=“C:\PRD\trans.ktr” –level=Basic > C:\LOG\trans.log
```

2 SCHEDULING

2.1 Schedule a transformation on windows

The best way to go at it is to test the command first at the dos prompt.
Then you can use the windows scheduler to launch this command.
Windows versions since Windows 2000 have a GUI for doing this accessible through the control panel. However it's also possible to use the command line to do this:

```
at 23:30 /every:Monday,Wednesday,Friday "D:\update_dimensions.bat"
```

To see a list of the scheduled commands simply type:

```
at
```

2.2 Schedule a transformation on Unix

First create a shell script that runs all the transformations you need. Then you can schedule this script to run.
On Unix like systems the easiest way to schedule a command is by using the "cron table".
You can do this by entering the following command:

```
crontab -e
```

Then you can enter the time at which the command needs to be run as well as the command on a single line in the text file that is presented.
The first options are:

- ✓ Minute: The minute of the hour, 0-59
- ✓ Hour: The hour of the day, 0-23
- ✓ Month day: The day of the month, 1-31
- ✓ Month: The month of the year, 1-12
- ✓ Weekday: The day of the week, 0-6, 0=Sunday

You can specify more then 1 number for each of these values by separating 2 number with a hyphen (-). This means an inclusive number range. If you separate the number by commas (,), it means distinct values. If you use * instead of a number, it means: every possible hour, minute, day, month or weekday.

So, if you want to update the dimensions every hour, at 15 and 45 minutes past the hour during the weekdays, you might enter these lines in a crontab:

```
#
# Launches the update of the dimensions in the warehouse
#
15,45 * * * 1-5 /PROD/update_dimensions.sh
#
```