

MANUAL EJECUCIÓN ETL

Datos del servidor: dubra.itg.es **usuario:** www-data **contraseña:** 27ws9mk

Para ejecutar el ETL de forma automática en el servidor se ha creado un script (**start.sh**) situado en el directorio **/var/www/data-integration/** que se explica a continuación:

```
#!/bin/sh
PATH="/usr/local/java/jdk/bin:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games"
export PATH
JAVA_HOME="/usr/local/java/jdk"
export JAVA_HOME
/var/www/data-integration/kitchen.sh -file=../ETL/job.kjb -logfile=../ETL/log1.log
```

shebang - en la jerga de Unix, el nombre que recibe el par de caracteres **#!** que se encuentran al inicio de los programas ejecutables interpretados (WIKIPEDIA).

Seguidamente se especifica el interprete que ejecutará las ordenes (/bin/sh)

/var/www/data-integration/kitchen.sh ruta absoluta al comando de pentaho para ejecutar el job (ETL).

opciones:

-file= : ruta relativa del ETL a ejecutar

-logfile= : ruta relativa del fichero de registro que usará el proceso.

***PARA MAS INFORMACIÓN:**

<http://wiki.pentaho.com/display/EAI/Kitchen+User+Documentation>

En esto dos pasos se incluye el contenido de la variable **\$PATH** (variable de entorno Unix que indica donde están los ficheros ejecutables).

El comando **export** hace la variable accesible para todos los subprocesos.

Estos dos comandos son necesarios ya las variables pueden no estar definidas.

****** Los scripts interpretados ejecutan los comandos línea a línea, utilizando el interprete que se indica en el shebang.

De cara a ejecutar el mismo de forma automática, se añade este al crontab (con el comando **crontab -e**)

******Según la wikipedia : **cron** es un administrador regular de [procesos](#) en segundo plano ([demonio](#)) que ejecuta procesos o [guiones](#) a intervalos regulares (por ejemplo, cada minuto, día, semana o mes). Los procesos que deben ejecutarse y la hora en la que deben hacerlo se especifican en el fichero crontab.

```
crontab: installing new crontab
www-data@dubra:~/data-integration$ crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tar.gz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow command
00 03 * * * /var/www/data-integration/start.sh
```

Comando para listar las entradas del crontab **crontab -l**

Fila con la indicación de los campos necesarios para ejecutar la tarea.
-m: minuto de ejecución.
-h: hora de ejecución.
-dom: (day of month) día del mes.
-mon: mes.
-dow: (day of week) día de la semana.
-command: comando a ejecutar.

hora 03
minuto 00
el valor asterisco (*) indica todos los días de todos los meses
Ruta absoluta al script a ejecutar.