

## **Wikiprint Book**

**Title: Enviamatlab**

**Subject: TracMeteo - Enviamatlab**

**Version: 23**

**Date: 10/23/2021 07:43:57 PM**

## Table of Contents

Configuración de correo para recepción de mensajes (job terminado, etc.)	4
Observaciones	4

El acceso al cluster se realiza por `ssh` a [mar.macc.unican.es](http://mar.macc.unican.es) y los trabajos en Matlab hay que lanzarlos utilizando `enviamatlab.sh`, que se encarga de mandar el trabajo a la cola estadística (que es la cola por defecto para los trabajos de Matlab; esta cola puede ejecutar hasta 24 trabajos a la vez: nodos WN10, WN11 y WN12, con 8 procesadores cada uno). En la versión 2009 (que es la que usa el cluster actualmente) Matlab puede consumir más CPU de la asignada si no se usa correctamente. Por tanto, sólo debe utilizarse `enviamatlab.sh` para ejecutar trabajos matlab en el cluster, salvo que se haga en sesiones interactivas con la ayuda de Antonio o Sixto (en este caso hay que habilitar una opción especial de la sesión 'help `maxNumCompThreads`', que no es trivial utilizar).

El script `enviamatlab.sh` se ha añadido en un directorio del cluster (`/software/MachineIndependent/bin/`) accesible por todos los usuarios. Por tanto, si os logueais en `mar` y poneis `'which enviamatlab.sh'` (es decir, busca la ruta donde está `'enviamatlab.sh'`) todos deberíais ver (sino es así, [hablar con Antonio o Sixto](#)):

```
[gutierjm@mar ~]$ which enviamatlab.sh
/software/meteo/bin/enviamatlab.sh
```

Para ejecutar cualquier programa de Matlab hay que llamar a este script desde el directorio donde tengáis el `script.m` que se quiera ejecutar (en el ejemplo de abajo, en el home "-"):

```
[gutierjm@mar ~]$ enviamatlab.sh script.m
```

De este modo, se generarán en la carpeta dos ficheros:

- `matlabqsub.20091123T105716`: este fichero lo genera el `enviamatlab` y será el que arranque matlab y ejecute el `script.m`. El final del nombre es la fecha de ejecución.
- `script.20091123T105716.log`: en este fichero se recoge la salida por pantalla de Matlab.

Al final de la ejecución del `script.m` se genera un nuevo fichero `'script.o114037'`, donde el número indica el identificador del job ejecutado.

Una vez ejecutamos la función nos aparecerá por pantalla una serie de mensajes con la información del trabajo enviado.

```
[gutierjm@mar ~]$ enviamatlab.sh script.m
+ mfile=script
+ mlscrip=matlabqsub.20111005T182930
+ mfilepath=/oceano/gmeteo/users/gutierjm/
Scientific Linux CERN SLC release 4.6 (Beryllium)
482820.ce01.macc.unican.es
```

En la última línea se muestra el identificador del trabajo (**JobId**, en el ejemplo **482820**) seguido del nombre de la máquina `ce01.macc.unican.es`; esta máquina es el submitting machine, pero no es la que ejecuta el job; para monitorizar los jobs ver la sección de monitorización; por ejemplo, los trabajos que están corriendo en la cola 'estadística' se pueden ver con:

```
gutierjm@mar dynamicReservoir_Basic]$ qstat -nl | grep estad
482820.ce01.macc.uni gutierjm estadist simulation 29340 1 -- 750mb -- R -- wn011/0
```

Por defecto, `enviamatlab.m` envía los trabajos a **un sólo procesador (ppn) con un límite de 750mb de memoria (mem)**. Si el job consume más memoria, la cola lo interrumpe. Si se necesita correr trabajos que consuman más memoria hay una opción para reservar el doble de memoria:

```
[gutierjm@mar ~]$ enviamatlab.sh test.m -l mem=1500mb -l nodes=1:ppn=2
```

o cuatro veces más (hasta 3GB de memoria). Si necesitáis ejecutar jobs que requieran más memoria, [hablar con Antonio o Sixto](#).

```
[gutierjm@mar ~]$ enviamatlab.sh test.m -l mem=3000mb -l nodes=1:ppn=4
```

Aunque la configuración inicial del script `enviamatlab.sh` remite los jobs a la cola estadística, al igual que para el caso de la memoria, existe un flag o modificador que nos permite seleccionar otras colas que puedan disponer de recursos disponibles en algún momento.

```
[gutierjm@mar ~]$ enviamatlab.sh test.m -q amd
```

En el ejemplo, se modifica la cola enviando el job a la cola `amd`, en lugar de a estadística. Aunque existen numerosas colas, en la medida de lo posible los trabajos asociados a la cola estadística deben derivarse a esta cola. En los casos en que ambos recursos, los de las colas estadística y `amd`, no sea suficientes, ponerse en contacto con Jesús (Chus) para ver qué cola podría usarse de las habitualmente utilizadas por los ASNA para realizar las

simulaciones y lanzar el operativo del WRF.

## Configuración de correo para recepción de mensajes (job terminado, etc.)

**NOTA:** La primera (y **sólo la primera**) vez que queramos enviar un trabajo de Matlab al cluster debemos editar el fichero `.bashrc` de nuestro home de oceano, incluyendo al final del mismo las líneas:

```
export EMAIL=tuemail[at]unican.es
source /software/etc/bashrc
```

El fichero final debería ser tal que así (se ha cogido como ejemplo el `.bashrc` del home de Rodri):

```
export EMAIL="rmanzanas@ifca.unican.es"
source /software/etc/bashrc
```

Para ello, basta hacer:

```
[gutierjm@mar cluster]$ vi .bashrc
```

editar el fichero justo al final "g", incluir "i" las dos líneas anteriores. Luego presionar "escape" y escribir ":" (comando para salir guardando) y presionar "enter".

El que no sepa, que me pregunte a mi o a Sixto, pero no tengáis copias del script 'enviamatlab' en local porque estaréis generando errores para el resto de usuarios.

## Observaciones

En este apartado podéis escribir las observaciones, dificultades o aclaraciones que encontréis o creáis necesarias a la hora de usar el cluster.

- Es importante tener en cuenta que el Matlab en el que se arranca el proceso no tiene porque tener inicializada MeteoLab y que, por lo tanto, debemos añadir al script una orden para que la inicialice. Bastaría con añadir al inicio del script la ruta de acceso al MeteoLab que utilicemos (es recomendable tenerla siempre actualizada en nuestro home), ejecutar el programa `init` (que arranca la toolbox) y volver al directorio en el que queremos trabajar. De este modo podremos trabajar con todas las funciones de MeteoLab.
- A la hora de marcar las rutas en Linux debemos usar los elementos propios de este sistema. En particular debemos usar la '/' en lugar de la '\' usada habitualmente en Windows. Es importante crear los scripts de modo que puedan ser entendidos en Linux. Para consultar dudas podéis [preguntar a Chus, Antonio, Sixto, etc.](#)
- Siempre que tengamos algún problema con la sintaxis de un comando o no sepamos exactamente qué es lo que hace podemos acudir a su manual. Por ejemplo, en el caso del comando `qstat` (monitorización de trabajos) bastaría con teclear `man qstat`.
- Si teneis algún problema, y necesitais la imperiosa necesidad de enviar un correo, es mejor que useis el sistema de tickets que tenemos, en lugar del correo, ya que así las conversaciones se guardan y pueden servir para el futuro. [Pincha aquí para mandar un ticket, o guarda este enlace en tus 'Favoritos' para futuras ocasiones](#)