

under heavy construction

CSLab CheatSheet

## Εκτέλεση εφαρμογών και χρονοδρομολόγηση στα clusters του CSLab

Οι εφαρμογές που εκτελούνται στα clusters του CSLab χρονοδρομολογούνται με χρήση του TORQUE/PBS. Ο TORQUE χειρίζεται τους μηχανισμούς χαμηλού επιπέδου που αναφέρονται στην αίτηση για εκτέλεση (submission), στην παρακολούθηση του φόρτου των κόμβων και στην εκτέλεση και καταστροφή των εφαρμογών. Ο MAUI scheduler καθορίζει **πού** και **πότε** θα εκτελεστούν οι εφαρμογές βάσει των πληροφοριών που λαμβάνει από τον TORQUE και της πολιτικής που ορίζεται στο configuration. Με τις βασικές εντολές του MAUI μπορεί κανείς να παρακολουθεί την κατάσταση των εφαρμογών που εκτελούνται, ή να τερματίζει την εκτέλεσή τους. Μερικές φορές, όταν δεν είναι εφικτή η επικοινωνία του MAUI με τον TORQUE, μπορεί να χρειάζεται να χρησιμοποιούνται εντολές του TORQUE. Στις σελίδες του TORQUE και του MAUI υπάρχει αναλυτική τεκμηρίωση για τις διαθέσιμες εντολές.

Η αίτηση εκτέλεσης μιας εφαρμογής γίνεται με την εντολή του TORQUE `qsub`. Η εφαρμογή πρέπει να βρίσκεται μέσα σε αρχείο δέσμης ενεργειών (script) -- ο TORQUE δε δέχεται εντολές ως απευθείας ορίσματα.

### Εκτέλεση σειριακών εφαρμογών

Η αίτηση για εκτέλεση μιας σειριακής εφαρμογής γίνεται με την εντολή:

```
qsub αρχείο_εφαρμογής
```

Αυτή εντολή καταθέτει τις εντολές που βρίσκονται μέσα στο `αρχείο_εφαρμογής` σε έναν κόμβο. Το αρχείο `αρχείο_εφαρμογής` μπορεί να είναι απλά:

```
#!/bin/bash

efarmogi < arxeio_eisodou > arxeio_eksodou
```

Η παραπάνω εντολή θεωρεί πως η το εκτελέσιμο `efarmogi` βρίσκεται στο φάκελο `$HOME` του χρήστη που την εκτελεί. Αν δεν ισχύει η παραπάνω υπόθεση, θα πρέπει πριν την εκτέλεση να προσθέσουμε `cd path/to/efarmogi`. Πχ:

```
#!/bin/bash

cd $HOME/src/efarmogi/
efarmogi < arxeio_eisodou > arxeio_eksodou
```

Ο κάθε κόμβος, που περιέχει συγκεκριμένο αριθμό επεξεργαστών, μπορεί να εκτελέσει ισάριθμες σειριακές εφαρμογές, δεδομένης της διαθεσιμότητας μνήμης. Για να αποφευχθούν σφάλματα λόγω έλλειψης μνήμης, είναι επιθυμητό να καθορίζουμε εκ των προτέρων την ανάγκη της εφαρμογής για μνήμη. Π.χ. η παρακάτω εντολή:

```
qsub -l vmem=3gb αρχείο_εφαρμογής
```

ζητά 3GB μνήμη για τις εντολές του `αρχείο_εφαρμογής` και ο MAUI θα επιτρέψει την εκτέλεση 2 εφαρμογών σε έναν κόμβο που έχει 8GB μνήμης.

### Εκτέλεση παράλληλων εφαρμογών

Για την εκτέλεση εφαρμογών σε πολλαπλούς κόμβους, χρειάζεται η εκτέλεση της εντολής `qsub` με τις ακόλουθες παραμέτρους:

```
qsub -l nodes=XX:ppn=YY αρχείο_εφαρμογής
```

Με το `nodes=XX` αιτούμαστε XX κόμβους ενώ με το `ppn=YY` αιτούμαστε ο κάθε κόμβος να έχει YY πυρήνες.

### Καθορισμός των πόρων της εφαρμογής

Με την αίτηση για εκτέλεση της εφαρμογής, μπορούμε να καθορίσουμε το συνολικό χρόνο εκτέλεσης, τον αριθμό των κόμβων, τον όγκο της μνήμης που θα πρέπει να έχει κάθε κόμβος, ή οποιαδήποτε παράμετρο αναφέρεται στο `man page` του `pbs_resources`. Αν δεν τα καθορίσουμε, τα προεπιλεγμένα είναι 1 κόμβος για 5 ώρες. Οι μη προεπιλεγμένες παράμετροι καθορίζονται χρησιμοποιώντας την επιλογή `-l` της εντολής `qsub` ή με το `#PBS directive` στο `αρχείο_εφαρμογής`. Για παράδειγμα:

Η εντολή

```
qsub -l walltime=10:00:00,vmem=8gb αρχείο_εφαρμογής
```

θα αιτηθεί έναν επεξεργαστή σε έναν κόμβο με 8GB ελεύθερης μνήμης για εκτέλεση του αρχείο\_εφαρμογής για 10 ώρες.

Η εντολή αυτή ισοδυναμεί με την

```
qsub αρχείο_εφαρμογής
```

αν το αρχείο\_εφαρμογής περιείχε το κατάλληλο PBS directive όπως παρακάτω:

```
#!/bin/bash

#PBS -l walltime=10:00:00,vmem=8gb

cd $HOME/src/efarmogi/
efarmogi < arxeio_eisodou > arxeio_eksodou
```

**ΠΡΟΣΟΧΗ** τα directives που δίνουμε στη γραμμή εντολών (με την qsub -l υπερισχύουν αυτών στο αρχείο\_εφαρμογής

## Παρακολούθηση της εκτέλεσης

Η παρακολούθηση των εργασιών που βρίσκονται στην ουρά γίνεται είτε με τον TORQUE είτε με τον MAUI. Η κατάσταση μιας συγκεκριμένης εργασίας εμφανίζεται είτε με την εντολή qstat είτε με την checkjob και το jobid (που το επιστρέφει η qsub). Πχ:

```
qstat 4278
```

Για την εκτίμηση του χρόνου εκκίνησης της εκτέλεσης μιας εργασίας που έχουμε καταθέσει για εκτέλεση χρησιμοποιούμε την εντολή showstart με παράμετρο το jobid.

## Διαγραφή εργασιών

Για να διαγράψουμε εργασίες από την ουρά του χρονοδρομολογητή χρησιμοποιούμε την qdel ή την canceljob.

## Διαδραστικές εργασίες

Για να εκτελέσουμε "με το χέρι" εντολές σε έναν κόμβο εκτελούμε την εντολή qsub με την παράμετρο -I. Πχ:

```
ananos@scirouter:~$ qsub -I
qsub: waiting for job 5001.localhost to start
qsub: job 5001.localhost ready

ananos@xenon3:~$ uname -a
Linux xenon3 2.6.38 #4 SMP Tue Mar 15 13:37:18 EET 2011 x86_64 GNU/Linux
ananos@xenon3:~$ logout

qsub: job 5001.localhost completed
ananos@scirouter:~$
```

## Παράδειγμα αίτησης για εκτέλεση εργασίας

login στο entry point ( scirouter )

για να δούμε ποια nodes είναι online για να εκτελέσουν εφαρμογές από την ουρά:

```
pbsnodelist
```

```
cat << EOF > some_script.sh
#!/bin/bash -l

#πόσα nodes να κρατήσει με πόσα cores / node
#PBS -l nodes=1:ppn=1

#πού να βάλει το output;
#PBS -N test_exception
```

```
#ας δημιουργήσει ένα αρχείο για error/output  
#PBS -j oe  
  
#η εφαρμογή μας  
uname -a  
uptime
```

```
qsub some_script.sh
```

στο `test_exception.TorquePID` θα έχουμε την έξοδο της εφαρμογής μας.