Pokretanje i upravljanje poslovima

- 1 Uvod
- 2 Izvođenje poslova
  - 2.1 Opisivanje poslova
    - 2.1.1 Osnovni SGE parametri
    - 2.1.2 SGE varijable okoline
  - 2.2 Vrste poslova
    - 2.2.1 Serijski poslovi
    - 2.2.2 Paralelni poslovi
    - 2.2.3 Polja poslova
    - 2.2.4 Interaktivni poslovi
  - 2.3 Napredno opisivanje poslova
    - 2.3.1 Spremanje privremenih rezultata
    - 2.3.2 Resursi
    - 2.3.3 Red poslova
    - 2.3.4 Obavijesti o stanju posla
- 3 Praćenje i upravljanje izvođenjem poslova
  - 3.1 Prikaz stanja poslova
  - 3.2 Prikaz radnih čvorova
  - 3.3 Upravljanje poslovima
  - 3.4 Dohvat informacija o završenim poslovima
- 4 Šalabahteri
  - 4.1 Linux šalabahter
    - 4.1.1 Kretanje po datotečnom sustavu
    - 4.1.2 Upravljanje direktorijima
    - 4.1.3 Kopiranje datoteka i direktorija
    - 4.1.4 Premještanje i preimenovanje datoteka i direktorija
    - 4.1.5 Promjena zaporke
    - 4.1.6 Automatsko nadopunjavanje i pretraživanje povijesti naredbi
    - 4.1.7 Naredbe za dohvaćanje uputa
  - 4.2 SGE šalabahter
    - 4.2.1 Podnošenje poslova
    - 4.2.2 Provjera stanja poslova
    - 4.2.3 Zaustavljanje poslova
    - 4.2.4 Informacije o završenim poslovima

Uvod

Za raspoređivanje i upravljanje poslovima na računalnom klasteru Isabella koristi se **SGE** (engl. ***Son of Grid Engine***), klasterski sustav za upravljanje poslovima (engl. *job management system*, JMS). U ovom dokumentu opisano je korištenje SGE verzije 8.

Izvođenje poslova

Korisničke aplikacije (u nastavku ***poslovi***) koji se pokreću pomoću sustava SGE moraju biti **opisani startnom *shell* skriptom** (npr. sh, bash). Unutar startne skripte, pored normalnih naredbi navode se **SGE parametri**. Iste parametre moguće je navesti i izvan startne skripte, prilikom podnošenja posla.

Pokretanje poslova se ostvaruje naredbom **`qsub`:**

```
qsub <SGE_parametri> <ime_startne_skripte>
```

SGE također sadrži **grafičko sučelje** (engl. *graphical user interface*, GUI) za pristup svim funkcionalnostima sustava. GUI se pokreće naredbom **`qmon`**. Korištenje GUI-ja neće biti opisivano jer unutar njega postoje upute za uporabu (tipke *Help*).

Opisivanje poslova

Za opisivanje poslova koristi se jezik sustava SGE, a **datoteka za opis posla (startna skripta)** je standardna *shell* skripta. U zaglavlju svake skripte navode se **SGE parametri** kojima se detaljno opisuje posao nakon kojih slijede normalne naredbe za izvršavanje korisničke aplikacije.

Struktura startne skripte:

moj_posao.sge

```
#!/bin/bash

#$ -<parametar1> <vrijednost1>
#$ -<parametar2> <vrijednost2>

<naredba1>
<naredba2>
```

Posao opisan ovakvom startnom skriptom podnosi se naredbom:

```
qsub moj_posao.sge
```

Naredba qsub vraća ID posla koji se koristi za kasnije nadziranje stanja posla:

```
Your job <JobID> ("moj_posao") has been submitted
```

Osnovni SGE parametri

```
-N <ime_posla> : ime posla koje e se prikazivati pri dohvatu informacija o poslovima
-P <ime_projekta> : ime projekta kojemu posao pripada
-cwd : definira da je direktorij u kojemu se nalazi startna skripta radni direktorij posla
```

> Podrazumijevani radni direktorij posla je **`$HOME`**.

```
-o <ime_dat> : ime datoteke u koju se sprema standardni izlaz
-e <ime_dat> : ime datoteke u koju se sprema standardna greška
-j y|n : omoguuje spajanje standardnog izlaza i standardne greške u istu datoteku (podrazumijevana vrijednost
je n)
```

> Ako se standardni izlaz i greška eksplicitno ne navedu, biti će spremljeni u datoteke:
>
> 1. Ukoliko nije definirano ime posla:
>   `<radni_direktorij>/<ime_skripte>.o<id_posla>`
>   `<radni_direktorij>/<ime_skripte>.e<id_posla>`
> 2. inače:
>   `<radni_direktorij>/<ime_posla>.o<id_posla>`
>   `<radni_direktorij>/<ime_posla>.e<id_posla>`
>
> Parametri **`-o`** i **`-e`** mogu pokazivati na direktorij:
>
> ```
> #$ -o outputDir/
> #$ -e outputDir/
> ```
>
> U ovom slučaju, SGE će u direktoriju `outputDir` kreirati datoteke sa standardnim izlazom i standardnom greškom, naziva `<ime_posla>.o<JobID>` i `<ime_posla>.e<JobID>`
>
> **Bitno:** **`outputDir`** je potrebno ručno kreirati prije podnošenja posla.

```
-M <emailAddress>[,<emailAddress>]…        : lista mail adresa na koju se šalju obavijesti o poslu
-m [a][b][e][n]        : definira u kojem se sluaju šalju mail obavijesti:
                                        b - poetak izvoenja posla,
                                        a - greška u izvoenju,
                                        e - završetak izvoenja posla,
                                        n - ne šalji obavijesti (podrazumijevana opcija)

-now y|n : vrijednost y definira da se posao mora izvesti odmah. Kod interaktivnih poslova ovo je
podrazumijevana vrijednost.
                        Ukoliko SGE ne može pronai slobodne resurse, posao se ne sprema u red nego odmah
završava.
-r y|n : treba li se posao u sluaju greške tokom izvoenja ponovno pokrenuti (podrazumijevana vrijednost je n)
-R y|n : vrijednost y definira da e SGE prilikom rasporeivanja rezervirati vorove (bitno za višeprocesorske
poslove)

-l <resurs>=<vrij>[,<resurs>=<vrij>...] : definira resurse koje posao zahtijeva. Za detalje vidi Resursi.

-pe <paralelna_okolina> <raspon> : parametar se koristi za paralelne poslove.
    Prvi parametar definira modul koji pokree traženi oblik paralelnog posla.
    Drugi parametar definira konkretan broj procesora ili raspon u obliku <N>,[<Ni>,...]<S>-<E>,[<Si>-<Ei>,]
koji paralelni posao zahtijeva.
    Za detalje vidi Paralelni poslovi.
-q <ime_reda>[,<ime_reda>...] : red poslova u koje se sprema posao. Ovu opciju je mogue koristiti i za
zahtijevanje odreenog vora, na nain da se zahtijeva
                                                                        lokalni red poslova (npr. a12.q@sl250s-gen8-
08-01.isabella)
-t <poetak>:<kraj>:<korak> : parametar definira da se radi o polju poslova. Za detalje vidi Polja poslova.

-v <var>[=<vrijednost>][,<var>[=<vrijednost>]...] : opcija defnira da SGE prilikom izvoenja posla postavi
varijablu okoline. Ovaj parametar je koristan kada

aplikacija koristi posebne varijable okoline, jer ih SGE podrazumijevano ne postavlja prilikom pokretanja posla.
-V : SGE poslu prenosi sve trenutne varijable okoline.
```

> Napomena: prilikom nabrajanja vrijednosti parametara (npr. `-l` ili `-q`) nije dozvoljeno stavljati razmake.

> Detaljan popis i informacije o parametrima moguće je dobiti naredbom **`man qsub`**.

SGE varijable okoline

Unutar startne skripte moguće je koristiti SGE varijable. Neke od njih su:

```
$TMPDIR : ime direktorija u koji se mogu spremati privremene datoteke (/scratch)
$JOB_ID : SGE identifikator posla
$SGE_TASK_ID : identifikator zadatka kod polja poslova
$SGE_O_HOST : adresa raunala sa kojeg je pokrenut posao
$SGE_O_PATH : originalna vrijednost varijable okoline PATH prilikom pokretanja posla
$SGE_O_WORKDIR : direktorij iz kojeg je posao pokrenut
$SGE_STDOUT_PATH : datoteka u koju se sprema standardni izlaz
$SGE_STDERR_PATH : datoteka u koju se sprema standardna greška
$HOSTNAME : adresa raunala na kojem se izvršava skripta
$JOB_NAME : ime posla
$PE_HOSTFILE : ime datoteke u kojoj su navedene adrese raunala
$QUEUE : ime reda u kojem se posao izvršava
```

Vrste poslova

Serijski poslovi

Najjednostavniji oblik SGE poslova su serijski poslovi koji zahtijevaju samo jedan procesor za izvođenje. Za njih obično nije potrebno navoditi nikakve posebne parametre, nego se samo navodi ime programa.

Primjeri korištenja:

1. Primjer skripte bez dodatnih parametara

```
#!/bin/bash

date
```

2. Primjer jednostavne skripte s parametrima

```
#!/bin/bash

#$ -N Date_SGE_script
#$ -o Date_SGE.out
#$ -e Date_SGE.err

date
```

3. Primjer izvođenja programa iz trenutnog direktorija

moj_program.sge

```
#!/bin/bash

#$ -N mojprog
#$ -P local
#$ -o mojprog.out
#$ -e mojprog.err
#$ -cwd

mojprog
```

Paralelni poslovi

Za pokretanje paralelnih poslova potrebno je specificirati željenu paralelnu okolinu te broj procesorskih jezgara koji je potreban za izvođenje posla. Sintaksa je:

```
#$ -pe <tip_paralelnog_posla> <N>,[<Ni>,...]<S>-<E>,[<Si>-<Ei>,]
```

Primjeri korištenja:

1. Posao za izvođenje zahtijeva 14 procesorskih jezgara:

```
#$ -pe *mpi 14
```

2. Broj dodijeljenih procesorskih jezgara može biti između 2 i 4:

```
#$ -pe *mpi 2-4
```

3. Broj dodijeljenih procesorskih jezgara može biti 5 ili 10:

```
#$ -pe *mpi 5,10
```

4. Broj dodijeljenih procesorskih jezgara može biti 1 ili između 2 i 4:

```
#$ -pe *mpi 1,2-4
```

Više informacija o paralelnim okolinama dostupnim na Isabelli možete pronaći na stranici Redovi poslova i paralelne okoline.

Obzirom da korisnik ne mora unaprijed znati koliko procesorskih jezgara će mu biti dodijeljeno, SGE postavlja vrijednost varijable **$NSLOTS** na broj dodijeljenih jezgara.

Samo pokretanje paralelnih poslova je specifično jer postoje alati za pokretanje podnaslova (npr. `mpirun`) koji sami obavljaju raspoređivanje podposlova po čvorovima. Kada SGE dodijeli čvorove paralelnom poslu, **popis čvorova** spremi u datoteku **$TMPDIR/machines** koja se unutar skripte za opis posla prenosi kao parametar paralelnim poslovima (npr. `mpirun, mpiexec, pvm`...).

Primjer startne skripte za pokretanje jedne vrste paralelnog posla:

```
#!/bin/bash

#$ -N paralelni-posao
#$ -cwd
#$ -pe *mpi 14

mpirun_rsh -np $NSLOTS -hostfile $TMPDIR/machines ...
```

Polja poslova

SGE omogućava višestruko pokretanje istog posla, tzv. **polja poslova**. Podposlovi unutar polja se nazivaju **zadaci** i svaki zadatak dobiva svoj identifikator.

Prilikom pokretanja poslova korisnik specificira raspon vrijednosti identifikatora zadataka koristeći parametar **`-t`**:

```
#$ -t <pocetak>:<kraj>:<korak>
```

Vrijednost `<poetak>` je identifikator prvog zadatka, `<kraj>` identifikator posljednjeg zadatka, a `<korak>` vrijednost za koju se uvećava svaki sljedeći identifikator između `<poetak>` i `<kraj>`. SGE sprema identifikator svakog zadatka u varijablu **$SGE_TASK_ID**, pomoću koje korisnici mogu dodijeliti različite parametre pojedinom zadatku. Zadaci mogu biti serijski ili paralelni poslovi.

Primjeri korištenja

1. Primjer skripte za pokretanje polja poslova od 10 serijskih poslova:

```
#!/bin/bash

#$ -N job_array_serial
#$ -cwd
#$ -o output/
#$ -j y
#$ -t 1-10

./starSeeker starCluster.$SGE_TASK_ID
```

2. Primjer skripte za pokretanje polja poslova od 10 paralelnih poslova:

```
#!/bin/bash

#$ -N job_array_parallel
#$ -cwd
#$ -o output/
#$ -j y
#$ -t 1-10

mpiexec -machinefile $TMPDIR/machines ./starseeker starCluster.$SGE_TASK_ID
```

Interaktivni poslovi

SGE omogućava pokretanje interaktivnih poslova. Za interaktivno izvođenje poslova koristi se naredba **`qrsh`**.

Preporuča se koristiti ovaj oblik poslova u slučaju kada je potrebno **prevesti ili debuggirati aplikacije na čvorovima**. Za razliku od korištenja ssh, na ovaj način se daje do znanja SGE-u da su čvorovi zauzeti i da ne pokreće druge poslove na njima. Prilikom interaktivnog izvođenja naredbe potrebno je navesti puni put do naredbe.

Ukoliko SGE trenutno nema slobodnih resursa i posao se želi ostaviti da čeka u redu, potrebno je navesti parametar "`-now n`". U suprotnom će SGE odmah završiti izvođenje posla s porukom:

```
Your "qrsh" request could not be scheduled, try again later.
```

Primjeri korištenja

1. Izravan pristup do komandne linije testnog čvora:

   ```
   qrsh
   ```

2. Interaktivno izvođenje naredbe

   ```
   qrsh /home/user/moja_skripta
   ```

3. Interaktivno izvođenje aplikacije s grafičkim sučeljem

   ```
   qrsh -DISPLAY=10.1.1.1:0.0 <moja_skripta>
   ```

Napredno opisivanje poslova

Spremanje privremenih rezultata

Za spremanje privremenih rezultata koji se generiraju tijekom izvođenja posla ne preporuča se koristiti $HOME direktorij. Time se smanjuje učinkovitost aplikacije te opterećuju *frontend* i mreža klastera.

SGE za svaki pojedini posao kreira direktorij na disku na radnim čvorovima (**/scratch**), oblika `/scratch/<jobID>.<taskID>.<queue>`. Adresu ovog direktorija SGE sprema u varijablu **$TMPDIR**.

> Zbog veće brzine, korištenje privremenog direktorija na /scratch disku preporuča se i za poslove koji često zahtijevaju nasumičan pristup podacima na disku, kao što su TensorFlow i PyTorch poslovi.

> Ukoliko postoje naznake da će kreirane privremene datoteke **prijeći 500 GB**, spremanje privremenih podataka treba se vršiti na **disk `/shared`** (opisano niže).

> Privremeni direktorij na /scratch disku se **briše automatski** po završetku izvršavanja posla.

Primjeri korištenja:

1. Primjer jednostavnog korištenja $TMPDIR varijable:

```
#!/bin/bash

#$ -N scratch_1
#$ -cwd
#$ -o output/scratch.out
#$ -j y
#$ -l scratch=50

cd $TMPDIR
pwd > test
cp test $SGE_O_WORKDIR
```

2. Primjer kopiranja podataka na scratch disk:

```
#!/bin/bash

#$ -N scratch_2
#$ -cwd
#$ -o output/scratch.out
#$ -j y
#$ -l scratch=50

mkdir -p $TMPDIR/data
cp -r $HOME/data/* $TMPDIR/data

python3.5 main.py $TMPDIR/data
```

Ukoliko privremeni podaci premašuju 500 GB potrebno je koristiti **/shared**. Za razliku od scratch-a, direktorij na shared-u je potrebno ručno kreirati i nema automatskog uklanjanja direktorija.

Primjer korištenja:

```
#!/bin/bash

#$ -N shared
#$ -cwd
#$ -o output/shared.out
#$ -j y

mkdir -p /shared/$USER/$TMPDIR
cd /shared/$USER/$TMPDIR

pwd > test
cp test $SGE_O_WORKDIR
```

Resursi

Prilikom pokretanja poslova korisnik može detaljnije opisati koji uvjeti trebaju biti ispunjeni za posao. Primjerice, moguće je zahtijevati samo određenu arhitekturu radnog čvora, količinu memorije ili vrijeme izvršavanja. Navođenje potrebnih resursa omogućava bolje raspoređivanje poslova, a poslovima daje veći prioritet (više na stranici Prioriteti na Isabelli).

Potrebni resursi se navode pomoću parametra `-l`:

```
#$ -l <resurs>=<vrijednost>
```

Resursi koji **definiraju potrebe posla**:

```
arch : arhitektura vora (npr. lx26-x86, lx25-amd64)
hostname : adresa vora (npr. c4140-01.isabella)
```

Resursi koji **postavljaju stvarna ograničenja na poslove:**

```
vmem : koliina virtualne memorije (format: <num>K|M|G)
rss : koliina stvarne memorije
stack : veliina stoga
data : ukupna koliina memorije (bez stoga)
fsize : ukupna veliina datoteka
cput : procesorsko vrijeme (format: [<hours>:<min>:]<sec>)
rt : stvarno vrijeme
scratch : prostor na scratch disku izrazen u GB
```

⚠ Vrijednosti ovih resursa treba pažljivo definirati (npr. uzimati 50% veće vrijednosti od očekivanog). U slučaju prekoračenja posao će biti zaustavljen sa signalom "segmentation fault".

⚠ Vrijednosti se ne mogu mijenjati za aktivne poslove.

ⓘ **Vrijednosti resursa** definirane u startnoj skripti posla **postavljaju se po procesu**. Na primjer, ukoliko korisnik na jednom čvoru zahtijeva 3 procesorske jezgre, vrijednosti svih zatraženih resursa biti će pomnožene s 3.

Primjeri korištenja:

1. Primjer posla koji zahtijeva 20 CPU jezgara i 10 GB radne memorije **po procesu** (poslu će ukupno biti dodijeljeno 200 GB radne memorije):

   ```
   #$ -pe *mpi 20
   #$ -l memory=10
   ```

2. Posao zahtijeva 100 GB prostora na scratchu:

   ```
   #$ -pe *mpisingle 4
   #$ -l scratch=25
   ```

Red poslova

SGE podržava više redova poslova, za različite vrste i maksimalna trajanja poslova, za različite vrste poslova (interaktivni, vSMP poslovi, ...) itd.

Željeni red poslova može se definirati parametrom **`-q`**:

```
#$ -q <queueName>
```

Popis dostupnih redova poslova na Isabelli može se pronaći na Redovi poslova i paralelne okoline.

Obavijesti o stanju posla

SGE podržava slanje e-mail obavijesti o promjeni stanja posla.

Parametar **`-M`** određuje na koju email adresu će stizati obavijesti o stanju poslova:

```
#$ -M <email_adresa>
```

Parametar **`-m`** određuje u kojim slučajevima će se slati obavijest:

```
#$ -m [b][a][e][n]
```

- b - početak izvođenja
- a - greška u izvođenju
- e - završetak izvođenja
- n - ne slati obavijesti (pretpostavljena opcija)

Primjer korištenja:

```
#$ -m ae
#$ -M moj@mail.com
```

Obavijest će biti poslana na adresu moj@mail.com prilikom **prekida** ili **završetka** posla.

Praćenje i upravljanje izvođenjem poslova

Prikaz stanja poslova

Za prikaz stanja poslova koristi se SGE-ova naredba **qstat.** Sintaksa naredbe je:

```
$ qstat <opcije>
```

Izvršavanjem naredbe **`qstat` bez dodatnih opcija** dobiva se ispis svih trenutnih korisnikovih poslova:

```
[tsmolcic@teran ~]$ qstat
job-ID  prior   name       user         state submit/start at     queue                          slots ja-task-
ID
-----------------------------------------------------------------------------------------------------------------
-
 131982 0.00001 md5-10k.0  tsmolcic     r     05/28/2019 06:24:06 gpu.0.q@c4140-03.isabella          1
 131988 0.00001 md5-10k.2  tsmolcic     r     05/28/2019 09:29:26 gpu.2.q@c4140-01.isabella          1
 131990 0.00001 md1-5k.3   tsmolcic     r     05/28/2019 09:29:41 gpu.3.q@c4140-01.isabella          1
 131991 0.00001 md10-15.4  tsmolcic     r     05/28/2019 10:18:08 gpu.1.q@c4140-01.isabella          1
 131994 0.04119 md1kp.5    tsmolcic     qw    05/27/2019 13:14:14                                   20
 131996 0.00005 md1-5k.6   tsmolcic     qw    05/27/2019 13:16:30                                    1
```

Neke od važnijih opcija su:

```
-s [r|p|s|h] : filtrira poslove prema stanju: r - aktivni, s - zaustavljeni, h - zaustavljeni u redu, p - u
redu ekanja
-j [job_ID] : detaljan prikaz stanja posla (ukoliko se ne navede job_ID, prikazuju se podaci o svim aktivnim
poslovima)
-f : prikazuju se detaljne informacije o optereenosti vorova i rasporedu poslova na vorove
-F : detaljan prikaz podataka o vorovima
-u <ime_korisnika> : prikazuju se samo poslovi od korisnika <ime_korisnik> (* - za sve korisnike)
-q [ime_reda] : naredba dohvaa iskljuivo informacije o redovima poslova
-g c : dohvat zbirnih informacija o redovima poslova: ukupno optereenje reda, ukupan broj vorova u redu, broj
zauzetih
                i slobodnih vorova te broj vorova u posebnim stanjima
-g t : u prikazu poslova su oznaeni slave i master zadaci
-l <resurs>=<vrijednost> : filtriranje poslova prema resursima
-ext : dodatne informacije o poslovima
-pri : detaljni podaci o prioritetima poslova
```

Opcije naredbe qstat je moguće kombinirati što je naročito bitno kada je potrebno filtrirati kroz nekoliko stotina poslova.

Primjeri korištenja:

1. Prikaz svih poslova u redu čekanja

```
$ qstat -u "*" -s p
```

2. Prikaz poslova svih korisnika koji se trenutno izvršavaju i zahtijevali su paralelnu okolinu `a16-mpi`

```
$ qstat -u "*" -s r -pe a16-mpi
```

3. Prikaz svih poslova i čvorova korisnika pero

```
$ qstat –s r –f –u pero
```

4. Prikaz svi poslova koji se trenutno izvršavaju na čvorovima s grafičkim procesorima:

```
$ qstat –u "*" -q gpu.*.q
```

Prikaz radnih čvorova

Za ispis broja procesora, jezgri i količine radne memorije po čvoru koristi se naredba

```
$ qhost
```

Naredom se mogu ispisati vrijednosti raspoloživih resursa po radnim čvorovima:

```
$ qhost -F vendor,scratch,memory
```

Upravljanje poslovima

Poslom se može upravljati i nakon pokretanja.

Dok je posao u redu čekanja, moguće je privremeno zaustaviti njegovo izvršavanje naredbom

```
$ qhold <ID_posla>
```

Tako zaustavljen posao može se vratiti u red čekanja naredbom

```
$ qrls <ID_posla>
```

Veću kontrolu nad poslom omogućava naredba **qmod**. Naredba omogućava privremeno zaustavljanje aktivnih poslova i to na način da poslu pošalje signal `SIGSTOP`. Posao će preći u neaktivno stanje (T), ali neće osloboditi resurse (memoriju, opisnike datoteka). Naredba omogućava i spremanje stanja procesa na disk (*checkpointing*) za poslove koji imaju tu mogućnost. Nadalje pomoću `qmod` naredbe korisnik može zaustaviti aktivni proces i vratiti ga natrag u red poslova. Parametri naredbe su:

```
-c : naredba izvodi spremanje stanja proces
-f : parametar definira da se naredba izvede bez obzira da li je to mogue ili ne
       (korisno kod vraanja u red poslova koji su oznaeni s parametrom -r n).
-r: naredba zaustavlja posao i vraa ga natrag u red.
-s : naredba zaustavlja izvoenje aktivnog procesa.
-us : naredba nastavlja izvoenje zaustavljenog aktivnog posla.
```

Posao se u potpunosti zaustavlja ili miče iz reda čekanja naredbom:

```
$ qdel <ID_posla>
```

Moguće je zaustaviti sve poslove korisnika:

```
$ qdel -u <ime_korisnika>
```

Za zaglavljene poslove treba koristiti prisilno zaustavljanje:

```
$ qdel -f <ID_posla>
```

Dohvat informacija o završenim poslovima

Za dohvat informacija o završenim poslovima koristi se naredba qacct. Sintaksa je:

```
$ qacct <opcije>
```

Najčešći primjer korištenja ove naredbe je

```
$ qacct -j <ID_posla>
```

kojom se ispisuju svi podaci o završenom poslu.

Ostale korisne opcije naredbe qacct su:

```
-j <ID_posla> : detaljan opis pojedinih poslova
-h <adresa_vora> : statistike za pojedine vorove
-q <ime_reda> : statistike za pojedine redove
-o <ime_korisnika> : statistike za pojedine korisnike
-pe <ime_paralelne_okoline> : statistike za pojedine paralelne okoline
-slots <broj_procesora> : statistike za poslove za odreeni broj procesora
-P <projekt> : prikaz sažetka potrošnje definiranog projekta
```

Primjeri korištenja:

1. Detaljne informacije o svim poslovima izvršenim na klasteru (oprez: velika količina podataka):

   ```
   $ qacct -j
   ```

2. Prikaz informacija o svim poslovima definiranog korisnika:

   ```
   $ qacct -j -o <korisnik>
   ```

   Prikaz sažetka potrošnje računalnih resursa definiranog korisnika (ako se ne definira `<korisnik>`, prikazuju se podaci za sve korisnike):

   ```
   $ qacct -o <korisnik>
   ```

3. Prikaz informacija o svim poslovima za definirani projekt:

   ```
   $ qacct -j -P <projekt>
   ```

   Prikaz sažetka potrošnje definiranog projekta (ako se ne definira `<projekt>`, prikazuju se podaci za sve projekte):

   ```
   $ qacct -P <projekt>
   ```

Šalabahteri

Linux šalabahter

Kretanje po datotečnom sustavu

Naredba	Opis naredbe
`pwd`	Prikazuje trenutačnu lokaciju korisnika. Lokacija se prikazuje u obliku apsolutne putanje do trenutačnog direktorija.
`cd`	Promjena trenutačnog direktorija (cd - *change directory*).
`cd -`	Povratak u prethodni direktorij.

Upravljanje direktorijima

Naredba	Opis naredbe
`mkdir dir1`	Kreira direktorij imena `dir1`.
`mkdir -p /tmp/novi/dir1`	Opcija `-p` automatski kreira potrebne poddirektorije.
`rm -rf dir1/*`	Briše sve datoteke i poddirektorije unutar direktorija `dir1`, tj. ostavlja direktorij `dir1` praznim.
`rm -rf dir1/`	Briše sve datoteke i poddirektorije uključujući i `dir1`.

Kopiranje datoteka i direktorija

Naredba	Opis naredbe
`cp dat1 dat2`	Kopira datoteku `dat1` u `dat2` (`dat1` je nepromijenjena).
`cp dat1 dir/`	Kopira datoteku `dat1` u direktorij `dir`.
`cp -r dir1/* dir2/`	Kopira sve datoteke iz direktorija `dir1` u direktorij `dir2` bez samog direktorija `dir1`.
`cp -r dir1/ dir2/`	Kopira sve datoteke i poddirektorije u direktoriju dir u direktorij `dir2`, uključujući direktorij `dir1`.

Premještanje i preimenovanje datoteka i direktorija

Naredba	Opis naredbe
`mv dat1 dat2`	Preimenuje datoteku `dat1` u `dat2`
`mv dat1 dir1`	Ako je `dir1` ime direktorija, premješta datoteku `dat1` u direktorij `dir1`

Promjena zaporke

Naredba	Opis naredbe
`passwd`	Promjena zaporke trenutnog korisnika. Naredba prvo traži upis stare lozinke, a zatim traži da se upiše nova lozinka (dva puta). Napomena: prilikom upisivanja lozinke, iz sigurnosnih razloga, u terminalu se ne ispisuje tekst.

Automatsko nadopunjavanje i pretraživanje povijesti naredbi

Naredba	Opis naredbe
[Tab]	Brzo nadopunjavanje naredbi. Kad korisnik počne pisati naredbu, npr. passwd, može napisati prvih nekoliko slova (npr. pass) i pritisnuti tipku **[Tab]**. Ljuska će tada automatski završiti naredbu ili ispisati sve naredbe koje započinju nizom pass. Na isti način se mogu nadopunjavati imena datoteka na disku.

[Ctrl] + [r]	Pretraživanje povijesti naredbi. U terminalu, držite pritisnutu tipku **[Ctrl]** i pritisnete tipku **[r]**. Počnete pisati slova iz naredbe, i pojavljuju vam se prijašnje naredbe koje sadržavaju slova koja tipkate. Ako želite rotirati kroz sve naredbe koje sadrže utipkana slova, ponovno stisnete **[Ctrl]** + **[r]**.

Naredbe za dohvaćanje uputa

Naredba	Opis naredbe
`man`	Upute za korištenje naredbi sustava. Ako želite vidjeti kako se koristi, npr. naredba `ls`, za otvaranje man-stranice napisat ćete `man ls`.
`<naredba> [-h/--help]`	Dobar dio aplikacija ima mogućnost dohvaćanja uputa korištenjem parametara `-h` ili `--help` (ne vrijedi za sve aplikacije!).

SGE šalabahter

Podnošenje poslova

Naredba	Opis naredbe
`qsub`	Podnosi posao i vraća ID posla.

Provjera stanja poslova

Naredba	Opis naredbe
`qstat`	Prikazuje stanje poslova na klasteru za trenutnog korisnika.
`qstat -f`	Prikazuje stanje poslova i čvorova.
`qstat -s rpsh`	Prikazuje: `p` - poslove koji čekaju u redu; `r` - aktivne poslove; `s` -privremeno zaustavljene aktivne poslove; `h` - privremeno zaustavljene poslove u redu.
`qstat -g c`	Prikazuje sažetak stanja pojedinih redova poslova.
`qstat -j <job_id>`	Prikazuje detaljan prikaz informacija o jednom poslu.
`qstat -u <user>`	Prikaz poslova određenog korisnika ("*"- za sve)
`qstat -pe <name>`	Prikaz poslova koji koriste definiranu paralelnu okolinu.
`qstat -q <queue>`	Prikaz poslova u definiranom redu poslova.

Zaustavljanje poslova

Naredba	Opis naredbe
`qdel <job_id>`	Posao se u potpunosti zaustavlja ili miče iz reda čekanja.
`qdel -u <user>`	Zaustavljaju se svi poslovi zadanog korisnika.
`qdel -f <job_id>`	Prisilno zaustavljanje za zaglavljene poslove.

Informacije o završenim poslovima

Naredba	Opis naredbe
`qacct -j <job_id>`	Detaljne informacije o poslu s ID-jem `<job_id>`.

`qacct -j`	Detaljne informacije o svim poslovima (velika količina podataka).
`qacct -j -o <user>`	Prikaz informacija o svim poslovima definiranog korisnika.
`qacct -o <user>`	Prikaz sažetka potrošnje definiranog korisnika; ako se ne definira `<user>`, prikazuju se podaci za sve korisnike.
`qacct -slots [<count>]`	Prikaz sažetka svih poslova koji su koristili zadani broj procesora; ako se ne definira `<count>`, prikazuju se podaci za sve vrijednosti.
`qacct -j -P <project>`	Prikaz informacija o svim poslovima za definirani projekt.
`qacct -P <project>`	Prikaz sažetka potrošnje definiranog projekta; ako se ne definira `<project>`, prikazuju se podaci za sve projekte.
`qacct -j -q <queue>`	Prikaz informacija o svim poslovima za definirani red poslova.
`qacct -q <queue>`	Prikaz sažetka potrošnje definiranog reda poslova.