

Pokretanje i upravljanje poslovima (Supek)

- 1 [Uvod](#)
- 2 [Izvođenje poslova](#)
 - 2.1 [Podnošenje poslova](#)
 - 2.1.1 [Polja poslova](#)
 - 2.2 [Opis posla](#)
 - 2.2.1 [Osnovni PBS parametri](#)
 - 2.2.2 [PBS varijable okoline](#)
- 3 [Dodjeljivanje resursa poslovima](#)
 - 3.1 [Kontrola memorije pomoću cgrupa](#)
 - 3.2 [Dodjeljivanje po traženom chunku](#)
- 4 [Paralelni poslovi](#)
 - 4.1 [OpenMP paralelizacija](#)
 - 4.2 [MPI paralelizacija](#)
 - 4.3 [MPI + OpenMP \(hibridna\) paralelizacija](#)
 - 4.4 [cray-pals](#)
- 5 [Praćenje i upravljanje izvođenja posla](#)
 - 5.1 [Praćenje posla](#)
 - 5.2 [Upravljanje poslovima](#)
 - 5.3 [Odgadjanje izvođenja](#)
 - 5.3.1 [Primjeri](#)

Uvod

Za raspoređivanje i upravljanje poslovima na računalnom klasteru Supek, koristi se **PBS Pro** (*Portable Batch System Professional*) koji obavlja raspoređivanje poslova unutar klastera. Njegova primarna zadaća je raspodjela računalnih zadataka, tj. paketnih poslova, među dostupnim računalnim resursima.

U ovom dokumentu opisano je korištenje PBS Pro 2022.1.1 verzije.



Korištenje GPU-a

GPU-ovi su dostupni samo na GPU pristupnom i radnim poslužiteljima, a njihovo korištenje je ograničeno na *gpu* i *gpu-test* redove poslova. Više o redovima poslova na pročitajte na stranici [Redovi poslova](#).

Izvođenje poslova

Korisničke aplikacije (u nastavku poslovi) koji se pokreću pomoću sustava PBS moraju biti opisani startnom shell skriptom (sh, bash, zsh...). Unutar startne skripte iznad normalnih naredbi, navode se PBS parametri. Te parametre moguće je navesti i prilikom podnošenja posla.

Osnovno pokretanje posla:

```
qsub moj_posao.pbs
```

Pokretanje posla uz parametre:

```
qsub -q cpu -l select=1:ncpus=4:mem=10GB moj_posao.pbs
```

Više informacija za qsub parametre:

```
qsub --help
```

Nakon podnošenja posla moguće je pogledati standardni izlaz i grešku posla koji je u stanju izvođenja naredbama:

```
qcat jobID
qcat -e jobID
qtail jobID
qtail -e jobID
```

Podnošenje poslova

Postoji nekoliko načina na koji se poslovi mogu podnijeti:

- interaktivnim podnošenjem
- korištenjem skripte
- u interaktivnoj sjednici
- poljima poslova

U slučaju interaktivnog podnošenja, direktnim pozivanjem naredbe `qsub` će se u terminalu otvoriti uređivač teksta, putem kojeg se podnose naredbe za izvršavanje:

```
# pokreni qsub
[korisnik@x3000c0s25b0n0:~] $ qsub
Job script will be read from standard input. Submit with CTRL+D.
echo "Hello world"
14571.x3000c0s25b0n0.hsn.hpc.srce.hr

# ispiši sadržaj direktorija
[korisnik@x3000c0s25b0n0:~] $ ls -l
total 5140716
-rw----- 1 korisnik hpc          0 Jun  1 07:44 STDIN.e14571
-rw----- 1 korisnik hpc        12 Jun  1 07:44 STDIN.o14571

# ispiši sadržaj izlazne datoteke
[korisnik@x3000c0s25b0n0:~] $ cat STDIN.o14571
Hello world
```

U slučaju podnošenja skriptom, naredbe za izvršavanje možemo zadati u ulaznoj datoteci koju podnosimo:

```
# ispiši datoteku hello.sh
[korisnik@x3000c0s25b0n0:~] $ cat hello.sh
#!/bin/bash

#PBS -N hello
echo "Hello world"

# podnošenje skripte posla
[korisnik@x3000c0s25b0n0:~] $ qsub hello.sh
14572.x3000c0s25b0n0.hsn.hpc.srce.hr

# ispis sadržaja direktorija
[korisnik@x3000c0s25b0n0:~] $ ls -l
total 5140721
-rw----- 1 korisnik hpc          0 Jun  1 07:44 STDIN.e14571
-rw----- 1 korisnik hpc        12 Jun  1 07:44 STDIN.o14571
-rw----- 1 korisnik hpc          0 Jun  1 08:02 hello.e14572
-rw----- 1 korisnik hpc        12 Jun  1 08:02 hello.o14572
-rw-r--r-- 1 korisnik hpc       46 Jun  1 07:55 hello.sh

# ispis sadržaja datoteke
[korisnik@x3000c0s25b0n0:~] $ cat hello.o14572
Hello world
```

U slučaju interaktivne sjednice, korištenjem opcije `qsub -I` bez ulazne skripte otvorit će se terminal na glavnom radnom čvoru unutar kojeg možemo pokretati komande:

```
# hostname na pristupnom poslužitelju
[korisnik@x3000c0s25b0n0:~] $ hostname
x3000c0s25b0n0

# otvaranje interaktivne sjednice
[korisnik@x3000c0s25b0n0:~] $ qsub -I -N hello-interactive
qsub: waiting for job 14574.x3000c0s25b0n0.hsn.hpc.srce.hr to start
qsub: job 14574.x3000c0s25b0n0.hsn.hpc.srce.hr ready

# hostname na glavnom radnom voru
[korisnik@x8000c0s3b0n0:~] $ hostname
x8000c0s3b0n0
```

Polja poslova

U slučaju polja poslova, korištenjem opcije `qsub -J X-Y[:Z]` možemo podnijeti zadani broj identičnih poslova u rangu X do Y s korakom Z:

```
# podnošenje polja poslova
[korisnik@x3000c0s25b0n0:~] $ qsub -J 1-10:2 hello.sh
14575[ ].x3000c0s25b0n0.hsn.hpc.srce.hr

# ispis sadržaja datoteke
[korisnik@x3000c0s25b0n0:~] $ ls -l
total 5140744
-rw----- 1 korisnik hpc          0 Jun  1 07:44 STDIN.e14571
-rw----- 1 korisnik hpc        12 Jun  1 07:44 STDIN.o14571
-rw----- 1 korisnik hpc          0 Jun  1 08:02 hello.e14572
-rw----- 1 korisnik hpc          0 Jun  1 08:21 hello.e14575.1
-rw----- 1 korisnik hpc          0 Jun  1 08:21 hello.e14575.3
-rw----- 1 korisnik hpc          0 Jun  1 08:21 hello.e14575.5
-rw----- 1 korisnik hpc          0 Jun  1 08:21 hello.e14575.7
-rw----- 1 korisnik hpc          0 Jun  1 08:21 hello.e14575.9
-rw----- 1 korisnik hpc        12 Jun  1 08:02 hello.o14572
-rw----- 1 korisnik hpc        12 Jun  1 08:21 hello.o14575.1
-rw----- 1 korisnik hpc        12 Jun  1 08:21 hello.o14575.3
-rw----- 1 korisnik hpc        12 Jun  1 08:21 hello.o14575.5
-rw----- 1 korisnik hpc        12 Jun  1 08:21 hello.o14575.7
-rw----- 1 korisnik hpc        12 Jun  1 08:21 hello.o14575.9
-rw-r--r-- 1 korisnik hpc        46 Jun  1 07:55 hello.sh
```



Polja poslova

Ovaj način poželjan je naspram višetrukog podnošenja (npr. for petljom) jer:

- smanjuje opterećenje reda poslova - svaki posao će se istovremeno natjecati za resurse sa svima ostalim u redu, umjesto jedan za drugim
- lakše upravljanje - modifikacija svih poslova moguća je pozivom glavnog (npr. 14575[]) ili pojedinačnog (npr. 14575[3]) identifikatora posla

Varijable okoliša koje PBS definira pri njihovom izvršavanju su:

- `PBS_ARRAY_INDEX` - redni broj podposla u polju poslova (npr. jedan do devet u primjeru gore)
- `PBS_ARRAY_ID` - identifikator glavnog polja poslova
- `PBS_JOBID` - identifikator podposla u polju poslova

Opis posla

Za opisivanje poslova koristi se jezik sustava PBS, dok datoteka za opis posla je standardna shell skripta. U zaglavlju svake skripte navode se PBS parametri koji detaljno opisuju posao nakon kojih slijede naredbe za izvršavanje željene aplikacije.

Struktura startne skripte:

```
moj_posao.pbs

#!/bin/bash

#PBS -<parametar1> <vrijednost>
#PBS -<parametar2> <vrijednost>

<naredba>
```

Konkretni primjer startne skripte:

```
moj_posao.pbs

#!/bin/bash

#PBS -P testni_primjer
#PBS -q cpu
#PBS -e /home/moj_direktorij
#PBS -l select=2:ncpus=10

module load gcc/12.1.0

gcc --version
```

Osnovni PBS parametri

Opcija	Argument opcije	Značenje opcije
-N	<i>ime</i>	Postavljanje imena posla
-q	<i>destinacija</i>	Specificiranje reda posla i/ili servera
-l	<i>lista_resursa</i>	Traženje resursa potrebnih za izvođenje posla
-M	<i>lista_korisnika</i>	Postavljanje liste primatelja mail-a
-m	<i>opcije_maila</i>	Postavljanje vrste notifikacije mail-a
-o	<i>putanja/do/željenog/direktorija</i>	Postavljanje imena/putanje u koje se sprema standardni izlaz
-e	<i>putanja/do/željenog/direktorija</i>	Postavljanje imena/putanje u koje se sprema standardna greška
-j	<i>oe</i>	Povezivanje standardnog izlaza i greške u istu datoteku
-Wgroup_list	<i>šifra_projekta</i>	Odabir projekta pod kojim će se izvoditi posao

Opcije za slanje obavijesti putem pošte opcijom **-m**:

a	Mail se šalje kada batch sustav prekine posao
b	Mail se šalje kada se posao počne izvršavati
e	Mail se šalje kada posao završi
j	Mail se šalje za podposlove. Mora se kombinirati s jednom ili više podopcija a, b ili e

Primjer maila

```
#!/bin/bash

#PBS -q cpu
#PBS -l select=1:ncpus=2
#PBS -M <ime>@srce.hr,<ime2>@srce.hr
#PBS -m be

echo $PBS_JOBNAME > out
echo $PBS_O_HOST
```

Dobivena su dva maila

Početak posla

```
PBS Job Id: 2686.x3000c0s25b0n0.hsn.hpc.srce.hr
Job Name:   pbs.pbs
Begun execution
```

Završetak posla

```
PBS Job Id: 2686.x3000c0s25b0n0.hsn.hpc.srce.hr
Job Name:   pbs.pbs
Execution terminated
Exit_status=0
resources_used.cpus=0
resources_used.cput=00:00:00
resources_used.mem=0kb
resources_used.ncpus=2
resources_used.vmem=0kb
resources_used.walltime=00:00:01
```

Opcije za traženje resursa opcijom -l

-l select=3:ncpus=2	Traženje 3 komada(chunk-a) čvora po 2 jezgre (ukupno 6 jezgri)
-l select=1:ncpus=10:mem=20GB	Traženje 1 komada(chunk-a) čvora sa 10 jezgri i 20GB radne memorije
-l ngpus=2	Traženje 2 gpu-a

PBS varijable okoline

Ime	Opis
PBS_JOBID	Identifikator posla koji daje PBS kada se posao preda. Stvoreno nakon izvršenja naredbe qsub .
PBS_JOBNAME	Naziv posla koji je dao korisnik. Zadani naziv je naziv podnešene skripte.
PBS_NODEFILE	Popis radnih čvorova, odnosno procesorskih jezgri na kojima se izvršava posao
PBS_O_WORKDIR	Radni direktorij u kojem je podnesen posao, odnosno u kojem je pozvana naredba qsub .
OMP_NUM_THREADS	OpenMP varijabla koju PBS izvozi u okolinu, a koja je jednaka vrijednosti opcije ncpus iz zaglavlja PBS skripte
NCPUS	Broj zatraženih jezgara. Odgovara vrijednosti iz opcije ncpus iz zaglavlja PBS skripte.
TMPDIR	Putanja do scratch direktorija.



Određivanje radnog direktorija

Dok je u PBS pro određena putanja za output i error datoteke u direktoriju u kojem se izvode, input i output datoteke samog programa se zadano učitavaju/spremaju u \$HOME direktorij. PBS Pro nema opciju određivanja izvođenja posla u trenutnom direktoriju u kojem se nalazimo stoga je potrebno ručno promijeniti direktorij.

Poslije zaglavlja potrebno je napisati:

```
cd $PBS_O_WORKDIR
```

Prebacit će izvođenje posla u direktorij u kojem je pokrenuta skripta.

Dodjeljivanje resursa poslovima

PBS omogućava definiranje potrebnih resursa na nekoliko načina. Glavna jedinica za dodjeljivanje resursa je takozvani "Chunk" ili komad čvora. *Chunk* se definira s opcijom **select**. Broj procesorskih jezgri po *chunk-u* moguće je definirati s **ncpus**, broj mpi procesa s **mpiprocs** i količinu radne memorije s **mem**. Također moguće je definirati walltime (maksimalno vrijeme izvođenja posla) i place (način raspoređivanje chunk-ova po čvorovima).

Ako neki od parametara nisu definirani koristiti će se zadane vrijednost:

Parametar	Zadana (defaultna) vrijednost
select	1
ncpus	1
mpiprocs	1
mem	1800 MB
walltime	24:00:00/48:00:00
place	free

Kontrola memorije pomoću cgrupa

Osim za kontrolu korištenja procesora, cgrupe postavljene su da kontroliraju i potrošnju memorije. To znači da su poslovi koje korisnik pokreće ograničeni na traženu količinu memorije. Ako posao pokuša iskoristiti više memorije nego je to zatraženo u opisu posla, sustav će prekinuti taj posao i u izlaznu error datoteku zapisati:

Poruka za korisnika kojemu cgrupe ubiju posao zbog nedostatka memorije

```
-bash: line 1: PID Killed /var/spool/pbs/mom_priv/jobs/JOB_ID.SC  
Cgroup mem limit exceeded: oom-kill:constraint=CONSTRAINT_MEMCG,nodemask=(null),cpuset=_JOB_ID,mems_allowed=0,  
oom_memcg=/pbs_jobs.service/jobid/JOB_ID,task_memcg=/pbs_jobs.service/jobid/JOB_ID,task=JOB_ID,pid=PID,uid=UID
```

Kod svakog posla ova poruka bit će malo drugačija, jer sadrži podatke kao što su UID (jedinstvena brojčana oznaka korisnika), PID(brojčana oznaka procesa koji je ubijen), JOB_ID (ID posla koji dodjeljuje PBS).

Dodjeljivanje po traženom chunku

Primjeri:

Korisnik traži dva chunka od kojih se svaki sastoji od 10 procesorskih jezgara i 10GB RAM-a, s time da korisnik nije specificirao na koliko čvorova već će sustav optimizirati dodjelu. U ovom slučaju korisnik će na korištenje dobiti 20 procesorskih jezgara i 20 GB radne memorije.

Primjer traženja resursa

```
#PBS -l select=2:ncpus=10:mem=10GB
```

Korisnik traži 10 chunkova od kojih se svaki sastoji od jedne procesorske jezgre i 1 GB RAM-a, ali s uvjetom na jednom čvoru, pa će korisnik dobiti ukupno 10 procesorskih jezgara i 10 GB RAM-a.

Primjer traženja resursa

```
#PBS -l select=10:ncpus=1:mem=1GB:place=pack
```

U gornjim primjerima poslovi su definirani kroz količinu chunkova, jezgara i memorije, ali sustav dozvoljava da se poslovima dodjeljuju zadani resursi ako oni nisu zatraženi (*default* resursi):

Primjer traženja resursa

```
#PBS -l ncpus=4
#PBS -l mem=14GB
```

U ovom slučaju korisnik dobiva 4 procesorske jezgre i ukupno 14GB memorije na **jednom chunku**. Kad se poslovi opisuju bez opcije `select`, nije moguće "ulančavanje resursa" (odvajanje traženih resursa dvotočkom, potrebno je za svaki resurs staviti u novi red `-l` opciju)



Memorija

Ako definirate poslove koristeći **ncpus** bez opcije **select**, poželjno je definirati i količinu memorije, jer će u suprotnom dostupna radna memorija iznositi **1800 MB**.

Paralelni poslovi

OpenMP paralelizacija

Ukoliko Vaša aplikacija koristi paralelizaciju isključivo na razini OpenMP dretvi (engl. *threads*) i ne može se širiti van jednog radnog čvora (odnosno radi s dijeljenom memorijom), posao možete pozvati na način kako je prikazano u primjeru xTB aplikacije niže.



OpenMP aplikacije zahtijevaju definiranje varijable **OMP_NUM_THREADS**.

PBS sustav vodi računa o tome umjesto Vas, te joj pridružuje vrijednost varijable **ncpus**, definirane u zaglavlju PBS skripte.

```
#!/bin/bash

#PBS -q cpu
#PBS -l ncpus=8

cd ${PBS_O_WORKDIR}

xtb C2H4BrCl.xyz --chrg 0 --uhf 0 --opt vtight
```

MPI paralelizacija

Ukoliko Vaša aplikacija koristi paralelizaciju isključivo na razini MPI procesa i može se širiti van jednog radnog čvora (odnosno radi s raspodijeljenom memorijom), posao možete pozvati na način kako je prikazano u primjeru Quantum ESPRESSO aplikacije niže.



Vrijednost varijable **select** iz zaglavlja PBS skripte odgovara broju MPI procesa.

```
#!/bin/bash

#PBS -q cpu
#PBS -l select=16

cd ${PBS_O_WORKDIR}

mpiexec pw.x -i calcite.in
```

MPI + OpenMP (hibridna) paralelizacija

Ukoliko se Vaša aplikacije može paralelizirati hibridno, odnosno dijeliti svoje MPI procese u OpenMP *threadove*, možete posao pozvati na način kako je prikazano u primjeru GROMACS aplikacije niže:



OpenMP aplikacije zahtijevaju definiranje varijable **OMP_NUM_THREADS**. PBS sustav joj automatski pridružuje vrijednost varijable **ncpus**, definirane u zaglavlju PBS skripte.

Vrijednost varijable **select** iz zaglavlja PBS skripte odgovara broju MPI procesa.

```
#!/bin/bash

#PBS -q cpu
#PBS -l select=8:ncpus=4

cd ${PBS_O_WORKDIR}

mpiexec -d ${OMP_NUM_THREADS} --cpu-bind depth gmx mdrun -v -deffnm md
```

cray-pals

Za izvođenje aplikacija koje koriste paralelizaciju MPI (ili hibridno MPI+OMP) potrebno je podignuti modul **cray-pals** prije pozivanja naredbe **mpiexec**, čime se osigurava ispravna integracija aplikacije sa sustavom za podnošenje poslova PBS Pro i Crayevom inačicom aplikacije MPI **mpiexec** baziranoj na [implementaciji MPICH](#).

Primjer pozivanja ovog modula i izvršavanje paralelne aplikacije na dva procesora:

```
#!/bin/bash
#PBS -l ncpus=2
module load cray-pals
mpiexec -np 2 moja_aplikacija_MPI
```

Varijable okoliša koje će naredba **mpiexec** podesiti na svakom od rangova MPI biti će:

Varijabla okoliša	Opis
PALS_APIID	Jedinstveni identifikator aplikacije koju ste izvršili
PALS_DEPTH	Broj procesorskih jezgri po rangu
PALS_LOCAL_RANKID	Lokalni rang procesa MPI (ako se posao vrši na više čvorova)
PALS_RANKID	Totalni rang procesa MPI
PALS_NODEID	Redni broj lokalnog čvora (ako se posao vrši na više njih)
PALS_SPOOL_DIR	Privremeni direktorij



Znanstvene aplikacije na Supeku i cray-pals

Znanstvene aplikacije koje su dostupne na Supeku putem [modulefiles alata](#) u sebi već pozivaju ovaj modul, stoga ga nije potrebno opet podizati.

Praćenje i upravljanje izvođenja posla

Praćenje posla

Za prikaz stanja poslova koristi se PBS-ova naredba qstat. Osnovna sintaksa naredbe je:

```
qstat <opcije> <ID_posla>
```

Izvršavanjem naredbe qstat bez dodatnih opcija dobiva se ispis svih trenuthi poslova svih korisnika:

Job id	Name	User	Time Use	S	Queue
2663.x3000c0s25b*	mpi+omp_s	kmrkalj	00:36:09	R	cpu

Neke od korištenijih opcija su:

-E	Grupira poslove prema poslužitelju i prikazuje poslove poredane prema uzlaznom ID-u. Kada se qstat prikaže s popisom poslova, poslovi su grupirani po poslužitelju i svaka grupa je prikazana uzlaznim ID-om. Ova opcija također poboljšava performanse qstata.
-t	Prikazuje informacije o statusu za poslove, nizove poslova i podposlove.
-p	Prikaz stupca za Iskorišteno vrijeme zamjenjuje se postotkom obavljenog posla. Za posao niza ovo je postotak završenih podposlova. Za normalan posao, to je postotak iskorištenog dodijeljenog CPU vremena.
-x	Prikazuje informacije o statusu za dovršene i premještene poslove uz poslove u čekanju i pokrenute poslove.
-Q	Prikazuje status redova u standardnom formatu.
-q	Prikazuje status redova u alternativnom formatu.
-f	Prikazuje status posla u alternativnom formatu

Primjeri korištenja:

Detaljan prikaz posla:

```
qstat -fxw 2648
```

Tracejob naredba vadi i prikazuje log poruke za PBS posao po kronološkom redu.

```
tracejob <ID_posla>
```

Primjer:

```
$ tracejob 2670
```

```
Job: 2670.x3000c0s25b0n0.hsn.hpc.srce.hr
```

```
03/30/2023 11:23:24 L    Considering job to run
03/30/2023 11:23:24 S    Job Queued at request of mhrzenja@x3000c0s25b0n0.hsn.hpc.srce.hr, owner =
                        mhrzenja@x3000c0s25b0n0.hsn.hpc.srce.hr, job name = mapping, queue = cpu
03/30/2023 11:23:24 S    Job Run at request of Scheduler@x3000c0s25b0n0.hsn.hpc.srce.hr on exec_vnode
                        (x8000c0s0b0n0:ncpus=40:mem=104857600kb)
03/30/2023 11:23:24 L    Job run
03/30/2023 11:23:24 S    enqueueing into cpu, state Q hop 1
03/30/2023 11:23:56 S    Holds u set at request of mhrzenja@x3000c0s25b0n0.hsn.hpc.srce.hr
03/30/2023 11:24:22 S    Holds u released at request of mhrzenja@x3000c0s25b0n0.hsn.hpc.srce.hr
```

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>
```

Vraćanje natrag na red čekanja:

```
qr ls <ID_posla>
```

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

```
qdel <ID_posla>
```

Za zaglavljene poslove treba koristiti prisilno zaustavljanje:

```
qdel -W force -x <ID_posla>
```

Odgadanje izvođenja

PBS pruža mogućnost izvođenja poslova u ovisnosti o drugima, što je korisno u slučajevima poput:

- izvršavanje poslova ovisi o izlazu ili stanju prethodno izvršenog
- aplikacija zahtijeva sekvencijalno izvođenje raznih komponenata
- ispis podataka jednog posla može ugroziti izvođenje drugog

Direktiva koja omogućuje ovu funkcionalnost pri trenutnom podnošenju posla je:

```
qsub -W depend=<tip>:<ID_posla>[:<ID_posla>] ...
```

Gdje <tip> može biti:

- **after*** - pokretanje trenutnog s obzirom na ostale
 - **after** - izvršavanje trenutnog nakon početka izvršavanja navedenih
 - **afterok** - izvršavanje trenutnog nakon uspješnog završetka navedenih
 - **afternotok** - izvršavanje trenutnog nakon greške u završetku navedenih
 - **afterany** - izvršavanje trenutnog nakon završetka navedenih
- **before*** - pokretanje ostalih s obzirom na trenutni
 - **before** - pokretanje navedenih nakon početka trenutnog
 - **beforeok** - pokretanje navedenih nakon uspješnog završetka trenutnog
 - **beforenotok** - pokretanje navedenih nakon greške u izvršavanju trenutnog
 - **beforeany** - pokretanje navedenih nakon završetka trenutnog

- **on:<broj>** - izvršavanje posla koji će ovisiti o naknadno navedenom broju **before*** tipa poslova



Posao s direktivom `-W depend=...` neće biti podnesen ako navedeni ID-ovi poslova ne postoje (iliti, ako nisu u redu čekanja)

Primjeri

Ako želimo da **posao1** započne nakon uspješnog završetka **posao0** :

```
[korisnik@x3000c0s25b0n0] $ qsub posao0
1000.x3000c0s25b0n0.hsn.hpc.srce.hr

[korisnik@x3000c0s25b0n0] $ qsub -W depend=afterok:1000 posao1
1001.x3000c0s25b0n0.hsn.hpc.srce.hr

[korisnik@x3000c0s25b0n0] $ qstat 1000 1001
Job id          Name          User          Time Use S Queue
-----
1000.x3000c0s25b0n0  posao0      korisnik      00:00:00 R cpu
1001.x3000c0s25b0n0  posao1      korisnik      0 H cpu
```

Ako želimo da **posao0** započne tek nakon uspješnog završetka **posao1** :

```
[korisnik@x3000c0s25b0n0] $ qsub -W depend=on:1 posao0
1002.x3000c0s25b0n0.hsn.hpc.srce.hr

[korisnik@x3000c0s25b0n0] $ qsub -W depend=beforeok:1002 posao1
1003.x3000c0s25b0n0.hsn.hpc.srce.hr

[korisnik@x3000c0s25b0n0] $ qstat 1002 1003
Job id          Name          User          Time Use S Queue
-----
1002.x3000c0s25b0n0  posao0      korisnik      0 H cpu
1003.x3000c0s25b0n0  posao1      korisnik      00:00:00 R cpu
```