
Párhuzamosítás shell környezetben

<http://www.gnu.org/software/pexec>

Budapest, 2009.09.02

Párhuzamosítás shell alatt

Jó dolgok:

- sok processzoros és/vagy magas gépek elterjedése
- gyors net, ssh, ...
- shell: minden UNIX rendszeren megtalálható, közismert, ...

Problémák és feltételek:

- programok indításának költsége elhanyagolható legyen
- iterátorok: teljesen független feldolgozás (...)
- közös erőforrások problémája
 - háttértár
 - mások futtatnak-e?
- vezérlés lehetősége

Ciklusok (iterátorok)

```
#!/bin/sh
COLLECTION="alpha beta gamma delta epsilon zeta"
for x in ${COLLECTION} ; do
    echo ${x}
done
```

```
#!/bin/sh
COLLECTION="alpha beta gamma delta epsilon zeta"
SELF="$0" ; x="$1"
if test -n "${x}" ; then
    echo ${x}
else
    for x in ${COLLECTION} ; do $SELF ${x} ; done
fi
```

Alkalmazások

- *Data parallelism*: adatfeldolgozási, képfeldolgozási problémák
- kötegelt rutinfeladatok (pl. képszerkesztés, képátméretezés)
- rendszeradminisztrációs feladatok (több gépen, egyszerre, ugyanazt: a ciklus halmaza és a párhuzamosításra használt gépek halmaza megegyezik)

Iterátorok párhuzamosítva

```
#!/bin/sh
COLLECTION="alpha beta gamma delta epsilon zeta"
for x in ${COLLECTION} ; do
    echo ${x}
done
```



```
#!/bin/sh
COLLECTION="alpha beta gamma delta epsilon zeta"
pexec -r ${COLLECTION} -e x -c -R -- 'echo ${x}'
```

Iterátorok párhuzamosítva

```
#!/bin/sh
COLLECTION="alpha beta gamma delta epsilon zeta eta theta"
SELF="$0"
x="$1"
if test -n "${x}" ; then
    echo ${x}
else for x in ${COLLECTION} ; do $SELF ${x} ; done ; fi
```



```
#!/bin/sh
COLLECTION="alpha beta gamma delta epsilon zeta eta theta"
SELF="$0"
x="$1"
if test -n "${x}" ; then
    echo ${x}
else pexec -r ${COLLECTION} -e x -c -R -- $SELF \ $x ; fi
```

Egyéb problémák

- közös erőforrások:
 - klasszikus párhuzamosítási/vezérlési eszközök támogatása (mutexek, szinkronizálás és atomi parancsindítás)
 - több független felhasználó vezérlés: egy felügyelő szerver egy adott gépen: processzek száma és/vagy terhelés (*loadavg*) szerinti korlátozás
 - hosszú futtatások esetén hasznos lehet a korlátok változtatása
- hálózatok, megosztott háttér:
 - `stdin`, `stdout` és `stderr`: automatikus átirányítás
 - többi file: NFS, SSHFS, mutexek + `ssh`

Fejlesztési tervek

- valamiféle egyszerű API: független programokból is el lehet érni a vezérlési és egyéb funkciókat (mutexek, processzek indítása/leállítása);
- több gépen való futtatás: patkolások (NFS, SSHFS, mutexek + ssh) helyett kultúráltabb és/vagy privilegizálást nem igénylő megoldások
- webes felület (állapotok, terheltség lekérdezése, mikorra fejeződik be a futtatás, stb.)
- ha van/lesz rá igény és/vagy idő...