

# C2115

# Praktický úvod do superpočítání

VIII. lekce

Petr Kulhánek, Tomáš Bouchal

[kulhanek@chemi.muni.cz](mailto:kulhanek@chemi.muni.cz)

Národní centrum pro výzkum biomolekul, Přírodovědecká fakulta,  
Masarykova univerzita, Kotlářská 2, CZ-61137 Brno

# **Dávkové systémy (začínáme)**

# Dávkové zpracování

**Dávkové zpracování** je vykonávání série programů (tzv. dávek) na počítači bez účasti uživatele. Dávky jsou připraveny předem, takže mohou být zpracovány předány bez účasti uživatele. Všechna vstupní data jsou předem připravena v souborech (skriptech) nebo zadána pomocí parametrů na příkazovém řádku. Dávkové zpracování je opakem interaktivního zpracování, kdy uživatel až teprve za běhu programu poskytuje požadované vstupy.

## Výhody dávkového zpracování

- sdílení zdrojů počítače mezi mnoha uživateli a programy
- odložení zpracování dávek do doby, kdy je počítač méně vytížen
- odstranění prodlev způsobeným čekáním na vstup od uživatele
- maximalizace využití počítače zlepšuje využití investic (zejména u dražších počítačů)

zdroj: [www.wikipedia.cz](http://www.wikipedia.cz), upraveno

# Nástroje pro dávkové spouštění

➤ **OpenPBS**

<http://www.mcs.anl.gov/research/projects/openpbs/>

➤ **PBSPro**

<http://www.pbsworks.com>

➤ **Oracle Grid Engine**

<http://www.oracle.com/us/products/tools/oracle-grid-engine-075549.html>

➤ **Open Grid Scheduler**

<http://gridscheduler.sourceforge.net/>

➤ **Torque**

<http://www.adaptivecomputing.com/products/open-source/torque/>

# Nástroje pro dávkové spouštění

➤ **OpenPBS**

<http://www.mcs.anl.gov/research/projects/openpbs/>

➤ **PBSPro**

<http://www.pbsworks.com>

➤ **Oracle Grid Engine**

<http://www.oracle.com/us/products/tools/oracle-grid-engine-075549.html>

➤ **Open Grid Scheduler**

<http://gridscheduler.sourceforge.net/>

**open source**

➤ **Torque**

<http://www.adaptivecomputing.com/products/open-source/torque/>

je použit jako dávkový systém v MetaCentrum VO, na našich lokálních klastrech

# Torque

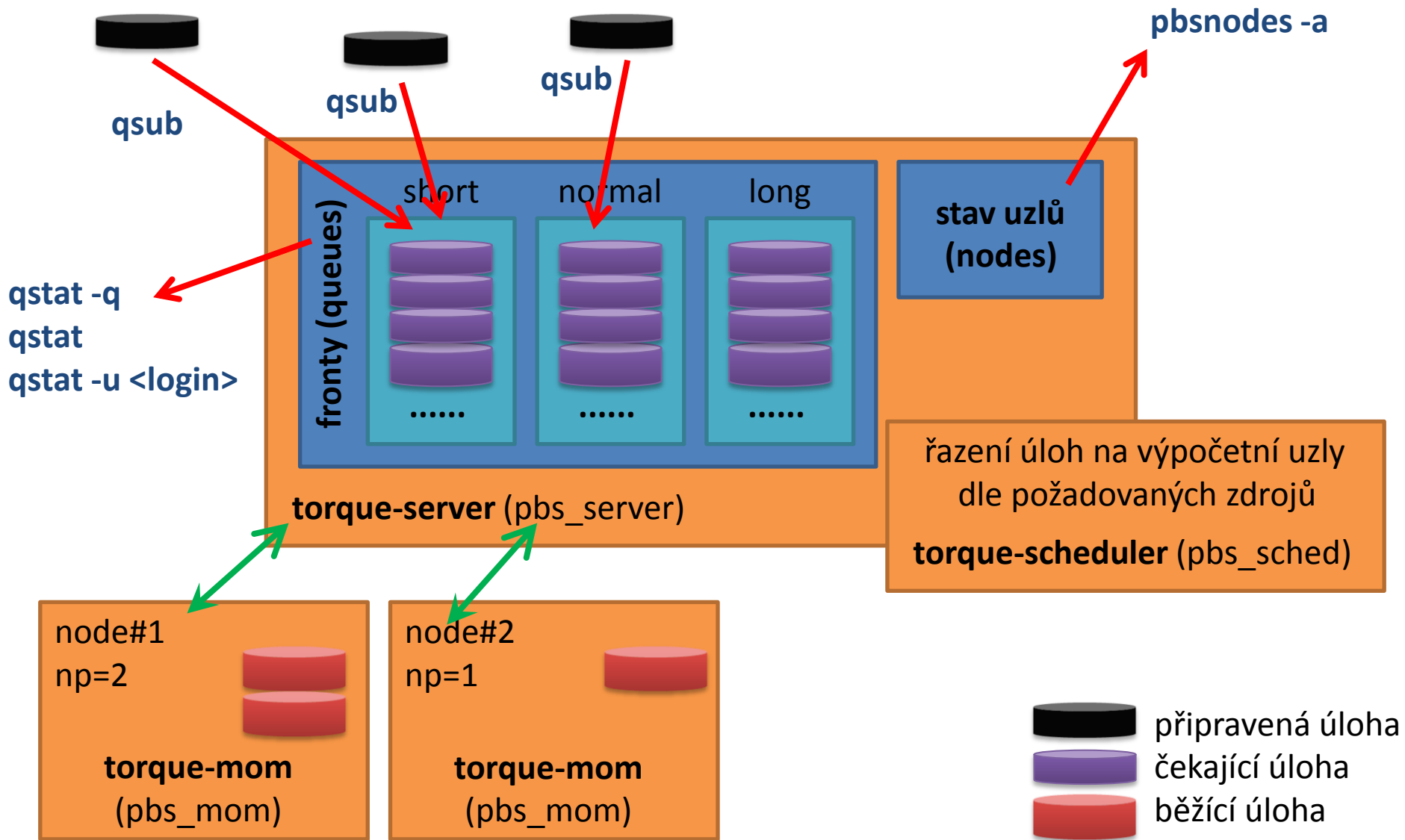
# Nezbytná podmínka

## Přihlašování bez hesla

Mezi výpočetními uzly a serverem (a naopak) je nutné nastavit přihlašování pomocí ssh bez explicitního zadávání hesla.

- **naše lokální klastry (WOLF, ...), IT4I** – pomocí ssh klíčů
- **MetaCentrum** – v době zasílání úlohy příkazem qsub do dávkového systému musíte mít platný kerberovský lístek

# Architektura





# Torque – příkazy, stavy úlohy

<b>qsub</b>	zašle úlohu do dávkového systému
<b>qstat</b>	vypíše informace o dávkovém systému (seznam úloh, seznam front)
<b>pbsnodes</b>	vypíše informace o výpočetních uzlech
<b>qrls</b>	uvolní úlohu ze stavu <b>holded</b> (pokud to okolnosti dovolují)

## Stavy úlohy:

<b>Q</b> (queued)	čeká ve frontě na spuštění na výpočetním uzlu
<b>R</b> (running)	běží na výpočetním uzly
<b>C</b> (completed)	úloha byla dokončena (informace o dokončených úloh se zobrazují jen omezenou dobu – nejčastěji 24 hodin)
<b>H</b> (holded)	úloha byla pozastavena, úlohu je možné uvolnit příkazem <b>qrls</b>
<b>E</b> (error)	došlo k chybě

# Zadáváme úlohy

K zadávání úloh do dávkového systému se používá příkaz **qsub**.

```
$ qsub -q normal uloha.sh  
1.ubuntu
```

jméno fronty, do které  
chceme úlohu zařadit

skript úlohy, např.

```
#!/bin/bash  
echo "Hello world from `hostname`!"
```

příkaz vypíše ID úlohy, pokud  
zařazení proběhne v pořádku

```
$ ls  
uloha.sh uloha.sh.o1 uloha.sh.e1
```

soubory jsou  
dostupné až po  
skončení úlohy

standardní výstup  
úlohy

standardní chybový  
výstup úlohy

# Cvičení 1

1. Jaké fronty dávkového systému jsou dostupné na klastru WOLF. Použijte příkaz `qstat` a volbu dle dokumentace.
2. Jaký je rozdíl mezi volbou `-Q` a `-q` příkazu `qstat`?
3. Jaké úlohy jsou již zařazeny v dávkovém systému klastru WOLF?
4. Skript úlohy z předchozí ukázky vložte do samostatného adresáře a zařadte jej do dávkového systému. Použijte frontu `normal`.
5. Na jakém výpočetním uzlu se úloha spustila?
6. Vytvořte nový skript úlohy, který umístíte do jiného adresáře. Skript vypíše jméno výpočetního uzlu a pozastaví svou činnost na 10 minut. Úlohu zařadte do fronty `normal`.
7. Monitorujte stav dávkového systému příkazy `qstat` a `pbsnodes` a dále pomocí webového rozhraní na stránce <https://wolf.ncbr.muni.cz> -> Vnitřní část -> Vytížení klastru
8. Na kterém výpočetním uzlu se úloha spustila tentokrát?

# Cvičení 2

1. Přihlaste se na čelní uzel MetaCentra `onyx.ncbr.muni.cz`.
2. Jaké fronty dávkového systému jsou dostupné. Použijte příkaz `qstat`.
3. Kolik úloh je aktuálně zařazeno do dávkového systému?
4. Skript úlohy z předchozí ukázky vložte do samostatného adresáře a zařadte jej do dávkového systému. Použijte frontu `default`.
5. Na jakém výpočetním uzlu se úloha spustila? Jak dlouho trvalo, než se úloha spustila?

# Cvičení 3

1. Přihlaste se na čelní uzel MetaCentra `zuphux.cerit-sc.cz`.
2. Jaké fronty dávkového systému jsou dostupné. Použijte příkaz `qstat`. Proč se liší od front, které jste viděli na čelním uzlu `onyx.ncbr.muni.cz`?
3. Kolik úloh je aktuálně zařazeno do dávkového systému?
4. Skript úlohy z předchozí ukázky vložte do samostatného adresáře a zařadte jej do dávkového systému. Použijte frontu `default`. V čem se liší identifikátor úlohy od identifikátoru úlohy zařazené na čelním uzlu `onyx.ncbr.muni.cz`?
5. Na jakém výpočetním uzlu se úloha spustila? Jak dlouho trvalo, než se úloha spustila?