Übung: Batch Systeme

Wo? -> Login auf Maxwell

- Haben Sie ihre Zugangsdaten noch?

- ssh school**NN**@max-display.desy.de
- Oder wenn Sie keinen ssh-Clienten haben:
  - https://maxdisplay.desy.de:3443/
  - Account schoolNN und das dazugehörige Passwort

- Slurm-Befehle: Fangen mit **s** an

Zusaetzliche Software: module

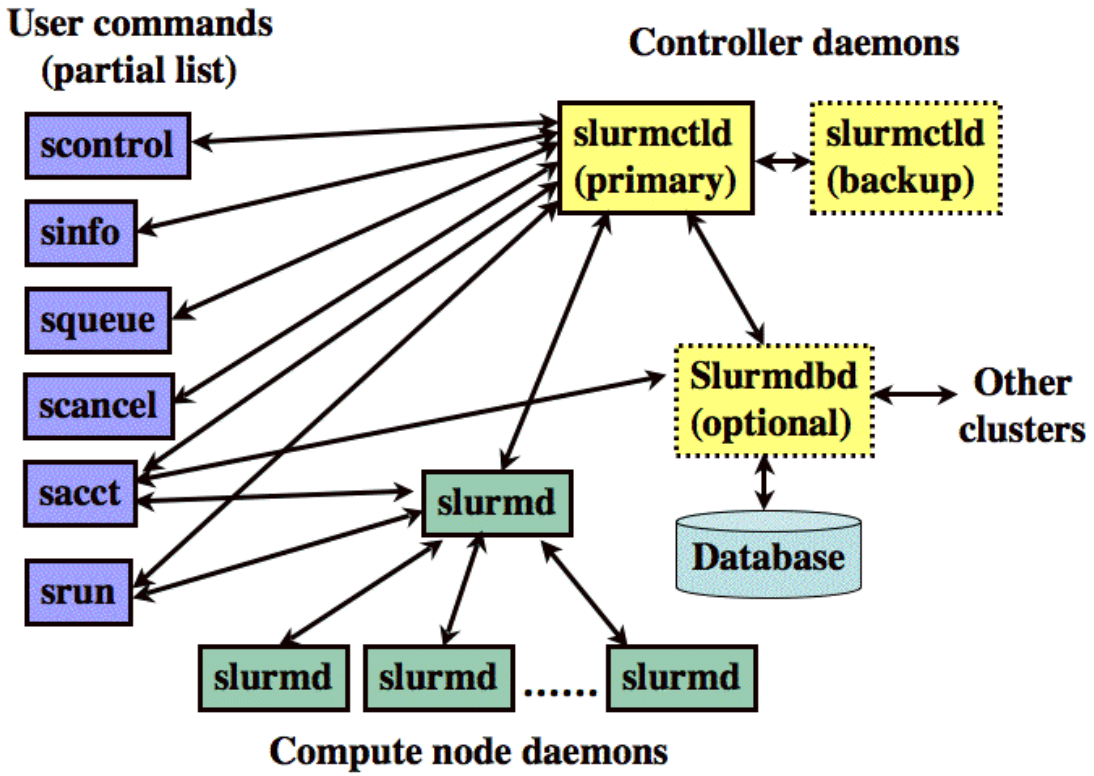
- Auflisten:
- `$ module avail`

- Im Batch Job ist die Shell-Funktion „module“ eventuell nicht aktiv. Deshalb im Batch Job vor dem ersten Aufruf von “module“:
- `source /etc/profile.d/module.sh`

Basic Information

What is slurm

- In simple word, SLURM is a workload manager, or a batch scheduler
- SLURM stands for Simple Linux UTlity for Resource Management



sinfo

sinfo(1) Slurm Commands sinfo(1)

NAME

sinfo - view information about Slurm nodes and partitions.

SYNOPSIS

sinfo [OPTIONS...]

DESCRIPTION

sinfo is used to view partition and node information for a system running Slurm.

...

Aufgaben:

- Wie viele Knoten hat das Maxwell Cluster?
- Was koennte eine “Partition“ sein?

Ein Batch-Job: Was ist das?

- Am Ende soll ein Executable ausgefuehrt werden, ggf. mit Parametern
- Wie soll man das paketieren fuers Batch-System?

Ein Programm über das Batch System ausführen

- sbatch

- zB: sbatch --wrap hostname

- Was läuft?
- squeue

Interaktives Arbeiten

- salloc -N 1 -J test
- Danach ssh zum Knoten

- Praktikabel?

Ein Batch-Job: Was ist das?

- Am Ende soll ein Executable ausgefuehrt werden, ggf. mit Parametern
- Wie soll man das paketieren fuers Batch-System?

Aufgabe 1:

- Erstellen Sie einen Job, der folgende Informationen sammelt:
  - Username und Gruppenmitgliedschaften
  - Hostname des Servers
  - User-Environment ausgibt
  - 60 Sekunden schläft
  - Einen return-code 0 ausgibt
- Der Job soll auf einem Knoten laufen
- Welche Laufzeit soll eingestellt werden?
- Submittieren Sie den Job (sbatch) und beobachten Sie die queue (squeue)

Lösung 1 (Beispiel):

```
#!/bin/bash
#SBATCH --time      0-00:02:00
#SBATCH --nodes     1
#SBATCH --partition all
#SBATCH --job-name  slurm-01
id
hostname
env
sleep 60
exit 0
```

```
sbatch skript.sh
squeue –u schoolNN
```

Wo steht das Ergebnis?

Aufgabe 2:

- Verändern Sie das vorherige Skript, indem Sie
  - Ihre matmult Applikation aufrufen
  - 10x hintereinander in einer Schleife

Lösung 2 (Beispiel):

```
#!/bin/bash
#SBATCH --time      0-00:10:00
#SBATCH --nodes     1
#SBATCH --partition all
#SBATCH --job-name  slurm-01
for i in 0 1 2 3 4 5 6 7 8 9; do
  ./a.out
done

sbatch skript.sh
squeue –u schoolNN
```

Wo steht das Ergebnis?

Features

- Slurm versteht die Angabe von „constraints“ , die bestimmte „Features“ der Worker Nodes erfordern
- ZB Vorhandensein einer GPU, Minimum RAM, Bestimmter CPU Typ,...

```
$ sinfo -N -l
Wed Jun 12 16:30:49 2019
NODELIST        NODES PARTITION       STATE CPUS    S:C:T MEMORY TMP_DISK WEIGHT AVAIL_FE REASON
max-cfel003         1       all   allocated   64   2:16:2 512000        0     10 INTEL,V3 none
max-cfel004         1       all   allocated   64   2:16:2 512000        0     10 INTEL,V3 none
...
$ sinfo -N  -o " %30f"
 AVAIL_FEATURES
 INTEL,V3,E5-2698,512G
...
```

Standardausgabe, unleserlich

Nur die Features (komma-getrennt)

Available features (ohne GPU)

- Anzahl AVAIL_FEATURES // Kommentar
- 2 AMD,7351,256G // AMD EPYC CPU
- 9 INTEL,Gold-6126,1536G // Intel Skylake 12 cores @ 2.6 GHz
- 88 INTEL,Gold-6140,768G // Intel Skylake 18 cores @ 2.3 GHz
- 12 INTEL,Silver-4114,384G // Intel Skylake 10 cores @ 2.2. GHz
- 28 INTEL,V3,E5-2640,256G // Intel Haswell 8 cores @ 2.6 GHz
- 6 INTEL,V3,E5-2698,512G // Intel Haswell 16 cores @ 2.3 GHz
- 165 INTEL,V4,E5-2640,256G // Intel Broadwell 10 cores @ 2.4 GHz
- 16 INTEL,V4,E5-2640,512G // Intel Broadwell 10 cores @ 2.4 GHz
- 80 INTEL,V4,E5-2698,512G // Intel Broadwell 20 cores @ 2.2 GHz

Aufgabe 3

- Variieren Sie das vorherige Skript, und setzen Sie ein contraint auf einen bestimmten CPU Typ

Lösung 3 (Beispiel):

```
#!/bin/bash
#SBATCH --time      0-00:10:00
#SBATCH --nodes     1
#SBATCH --partition all
#SBATCH --job-name  slurm-01
#SBATCH --constraint="V3,E5-2640"

for i in 0 1 2 3 4 5 6 7 8 9; do
  ./a.out
done

sbatch skript.sh
squeue –u schoolNN
```

Wo steht das Ergebnis?

Interpretation der Resultate

- Annahme: Ihr Skript ist single-threaded (kein OpenMP, kein MPI)
- Submittieren Sie auf zwei CPUs gleicher Architektur (zB Haswell)
  - V3,E5-2640 und V3,E5-2698
  - Intel Haswell 8 cores @ 2.6 GHz und Intel Haswell 16 cores @ 2.3 GHz
- Wir erwarten ~13% (2.6/2.3) mehr Leistung bei der ersten Architektur
- Wieso sind die Ergebnisse ungefähr gleich?

Erklärung der Resultate

- Einige Erklaerungsansaetze, zB ueber „TurboFrequenz“
- Bei der V3,E5-2640 CPU mit Grundfrequenz 2.6 GHz ist die Max. Turbo-Taktfrequenz 3.4 GHz
- Bei der V3,E5-2698 CPU mit Grundfrequenz 2.3 GHz ist die Max. Turbo-Taktfrequenz 3.6 GHz
- Mit welcher Frequenz die Applikation wirklich laeuft entscheidet der Kernel und die CPU

- https://ark.intel.com/content/www/de/de/ark/products/83359/intel-xeon-processor-e5-2640-v3-20m-cache-2-60-ghz.html
- https://ark.intel.com/content/www/de/de/ark/products/81060/intel-xeon-processor-e5-2698-v3-40m-cache-2-30-ghz.html

Optimierung des Benchmarks:

- Single-Threaded Applikationen sind nur bedingt sinnvoll
- Mittels OpenMP müssen alle Cores ausgelastet werden (denken Sie daran: Es gibt zwei CPUs in den Systemen, also doppelte Anzahl Cores)
- Noch besser:
  - Skalierung vergleichen, zB: 1,2,4,8,...,Ncores,...2xNcores,3xNcores,4xNcores
- Wo tritt eine Sättigung auf, oder eventuell sogar wieder Abfall?