Introduction to Slurm

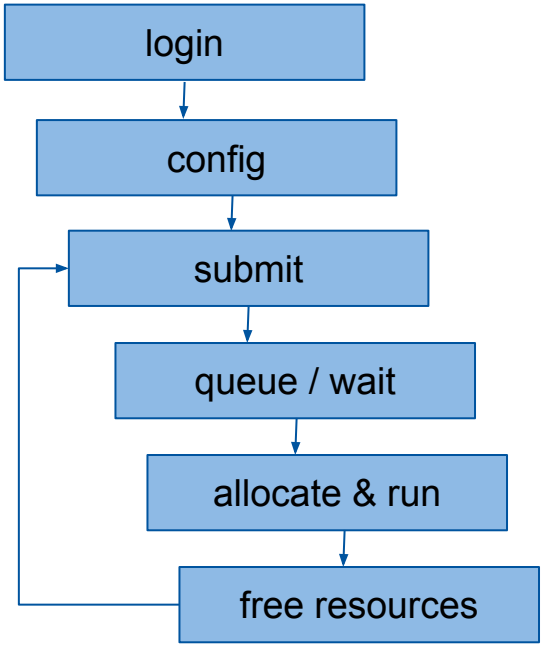
Cluster management and job scheduling system for CLAIX.

RWTH AACHEN UNIVERSITY

Batch system for CLAIX

General Introduction

- What is a batch system and why do I need it?
  - Share resources among many users “fairly”
  - Manages a **queue** of pending **jobs** in the cluster.
  - Considers **priority** to decide job order.
  - **Allocates** time and resources to **“jobs”** from users.
  - Allows to start, execute, and monitor **jobs**.



SLURM: Resource Manager + Scheduler

*18.08.7 -> 21.08.X this year

Batch system for CLAIX

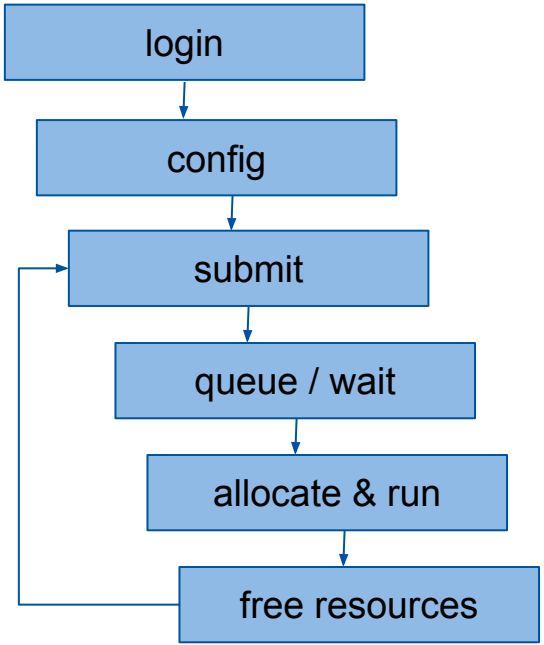
General Introduction

Partition: Group of nodes

Account: Project account

Billing: How many resources used

Batch Job: a chain of commands in a script file.

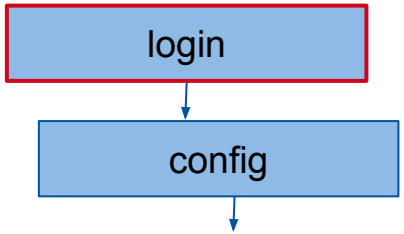


SLURM: Resource Manager + Scheduler

*18.08.7 -> 21.08.X this year

Batch system for CLAIX

General Introduction




```
ssh -l kz743613 login18-4.hpc.itc.rwth-aachen.de
kz743613@login18-4.hpc.itc.rwth-aachen.de's password:
```

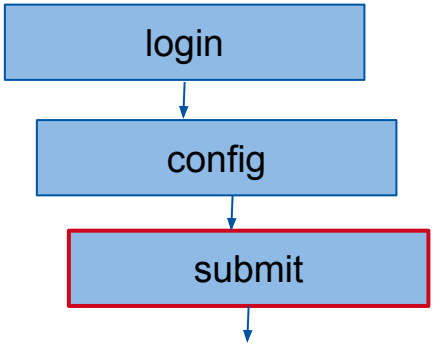
```
You are connected to the node 'login18-4'
kz743613@login18-4:~ $
```

[DIALOG / LOGIN NODES]

Batch system for CLAIX

General Introduction

- What is a batch system and why do I need it?
- The login-nodes are NOT „the cluster“.
- So, what is the cluster?
  - **Compute nodes** where computations are done.
  - “See” them from the login-nodes / cluster-nodes.
  - **Submit** programs to the cluster

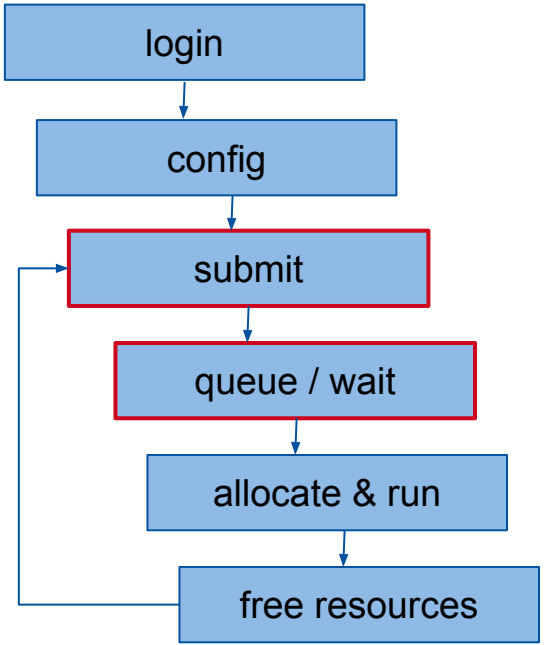


*JupyterHub + SLURM coming this year

Batch system for CLAIX

General Introduction

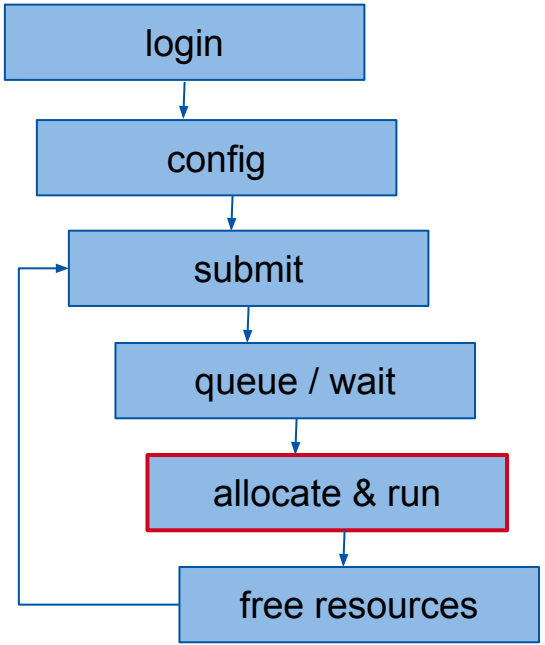
- What is a batch system and why do I need it?
- The login-nodes are NOT „the cluster“.
- So, what is the cluster?
- Use the batch system for calculations!
  - Queue to wait for the resources requested.
  - Submit a batch script with your program workflow.



Batch system for CLAIX

Commands

- **sbatch** script.sh - submit batch script job
- **squeue** -u $USER - list currently queued jobs for my user
- **$MPIEXEC / srun** <cmd> - submit job / step
- **scontrol** - admin tool, read only show info for non-root users



[slurm_docu] [itc.rwth docu]

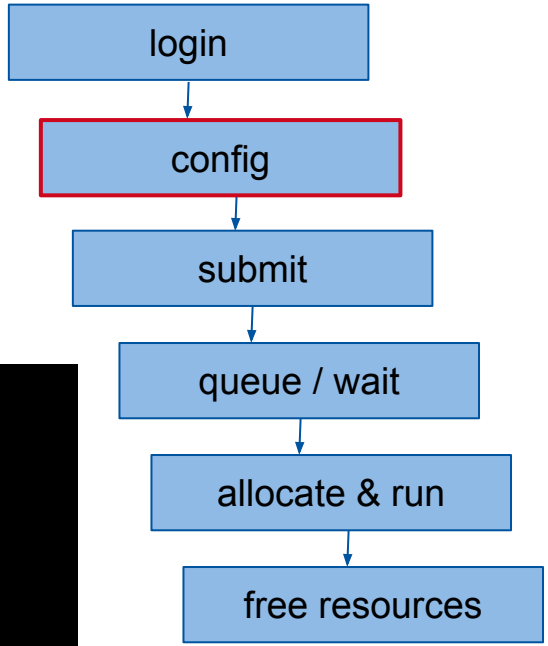
Slurm as the batch system for CLAIX

First batchscript

- Slurm **sbatch** needs a batchscript
  sbatch <batchscript>

- SHEBANG (very first line) of the batchscript should be
  #!/usr/local_rwth/bin/zsh

```
mw44552c@linuxc20:~[409]$
```



Slurm as the batch system for CLAIX

Lessons learned from the output

- **sbatch**: [I] No output file given, set to: output_%j.txt
- **sbatch**: [I] No runtime limit given, set to: 15 minutes

```
mw44552c@linuxc20:~[412]$ sbatch minimalscript.sh
sbatch: [I] No output file given, set to: output_%j.txt
sbatch: [I] No runtime limit given, set to: 15 minutes
Submitted batch job 12757911
```

Slurm as the batch system for CLAIX

Defaults NOT clear from the output:

- the partition was set to „c18m“ (default partition for non-projects)
- 1 core
- 3900 MB memory
- 15 minutes

```
kz743613@login18-1:~/tmp/example $ scontrol show part
PartitionName=c18m
   AllowGroups=ALL AllowAccounts=ALL AllowQos=ALL
   AllocNodes=ALL Default=NO QoS=N/A
   DefaultTime=00:15:00 DisableRootJobs=NO ExclusiveUser=NO
   MaxNodes=UNLIMITED MaxTime=30-00:00:00 MinNodes=0 LLN=NO
   Nodes=ncm[0001-1032],nrm[001-202]
   PriorityJobFactor=1 PriorityTier=100 RootOnly=NO ReqResv
   OverTimeLimit=NONE PreemptMode=OFF
   State=UP TotalCPUs=59232 TotalNodes=1234 SelectTypeParam
   JobDefaults=(null)
   DefMemPerCPU=3900 MaxMemPerNode=UNLIMITED
   TRESBillingWeights=CPU=1.0,Mem=0.25G
```

Slurm as the batch system for CLAIX

Partitions

name	#nodes	cpu arch	#cores / node	#mem / node	#mem / core	accel
c18m	1240	Skylake	48	187.200	3.900	
c18g	54	Skylake	48	187.200	3.900	2 * Volta
c16m	608	Broadwell	24	124.800	5.200	
c16s	8	Broadwell	144	1.020.000	7.050	
c16g	9	Broadwell	24	124.800	5.200	2 * Pascal

[system overview]

Slurm as the batch system for CLAIX

Configure your request (job)

```
#!/usr/local_rwth/bin/zsh

#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --ntasks-per-node=1
#SBATCH --time=00:45:00
#SBATCH --output=prog-out.%j
#SBATCH --error=prog-err.%j
#SBATCH --partition=c18m
#SBATCH -A projectABC


hostname
```

Slurm as the batch system for CLAIX

Configure your request (job)

- Simple 2 cores @ 2 nodes job

```
#!/usr/local_rwth/bin/zsh

#SBATCH --nodes=2
#SBATCH --ntasks=2
#SBATCH --ntasks-per-node=1
#SBATCH --time=00:45:00
#SBATCH --output=prog-out.%j
#SBATCH --error=prog-err.%j
#SBATCH --partition=c18m
#SBATCH -A projectABC


hostname
$MPIEXEC $FLAGS_MPI_BATCH hostname
```

Slurm as the batch system for CLAIX

Configure your request (job)

- Internal workings, srun

```
#!/usr/local_rwth/bin/zsh

#SBATCH --nodes=2
#SBATCH --ntasks=2
#SBATCH --ntasks-per-node=1
#SBATCH --time=00:45:00
#SBATCH --output=prog-out.%j
#SBATCH --error=prog-err.%j
#SBATCH --partition=c18m
#SBATCH -A projectABC


srun -n 2 hostname
```

Slurm as the batch system for CLAIX

Configure your request (job)

- Simple 2 cores @ 2 nodes job

```
#!/usr/local_rwth/bin/zsh

#SBATCH --nodes=10
#SBATCH --ntasks=480
#SBATCH --ntasks-per-node=48
#SBATCH --time=00:45:00
#SBATCH --output=prog-out.%j
#SBATCH --error=prog-err.%j
#SBATCH --partition=c18m
#SBATCH -A projectABC


$MPIEXEC $FLAGS_MPI_BATCH hostname
```

Slurm as the batch system for CLAIX

Configure your request (job)

- Custom distribution openmp:

```
#!/usr/local_rwth/bin/zsh

#SBATCH --nodes=2
#SBATCH --ntasks=96
#SBATCH --ntasks-per-node=48
#SBATCH --time=00:45:00
#SBATCH --output=prog-out.%j
#SBATCH --error=prog-err.%j
#SBATCH --partition=c18m
#SBATCH -A projectABC

OMP_NUM_THREADS=48

srun -n 2 -N 2 openmp_program   #assume this uses 48 threads internally
```

Slurm as the batch system for CLAIX

Submit your job: sbatch <script>

```
kz743613@login18-1:~/tmp/example $ sbatch batchscript.sh
Submitted batch job 25824062
kz743613@login18-1:~/tmp/example $ squeue -u $USER
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
          25824062      c18m job_test kz743613 PD       0:00      2 (None)
kz743613@login18-1:~/tmp/example $ scancel 25824062
kz743613@login18-1:~/tmp/example $ squeue -u $USER
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
          25824062      c18m job_test kz743613 CG       0:02      2 ncm[1014-1015]
kz743613@login18-1:~/tmp/example $ squeue -u $USER
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
          25824062      c18m job_test kz743613 CG       0:02      2 ncm[1014-1015]
kz743613@login18-1:~/tmp/example $ squeue -u $USER
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
kz743613@login18-1:~/tmp/example $
```

Slurm as the batch system for CLAIX

Job control

- Show job details
  - scontrol show job <jobid>
- Show queue of jobs
  - squeue –u <username>
- Cancel job
  - scancel <jobid>

```
kz743613@linuxc27:~ $ scontrol show job 25813729
JobId=25813729 JobName=zsh
   UserId=kz743613(43398) GroupId=kz743613(43398) MCS_label=N/A
   Priority=277638 Nice=0 Account=default QOS=normal
   JobState=CONFIGURING Reason=None Dependency=(null)
   Requeue=1 Restarts=0 BatchFlag=0 Reboot=0 ExitCode=0:0
   RunTime=00:00:01 TimeLimit=00:15:00 TimeMin=N/A
   SubmitTime=2022-03-03T11:44:55 EligibleTime=2022-03-03T11:44:55
   AccrueTime=2022-03-03T11:44:55
   StartTime=2022-03-03T11:45:07 EndTime=2022-03-03T12:00:07 Deadlin
   PreemptTime=None SuspendTime=None SecsPreSuspend=0
   LastSchedEval=2022-03-03T11:45:07
   Partition=c18m AllocNode:Sid=linuxc27:28031
   ReqNodeList=(null) ExcNodeList=(null)
   NodeList=ncm1028
   BatchHost=ncm1028
   NumNodes=1 NumCPUs=1 NumTasks=1 CPUs/Task=1 ReqB:S:C:T=0:0:*:*
   TRES=cpu=1,mem=3900M,node=1,billing=1
   Socks/Node=* NtasksPerN:B:S:C=0:0:*:* CoreSpec=*
   MinCPUsNode=1 MinMemoryCPU=3900M MinTmpDiskNode=0
   Features=hostok DelayBoot=00:00:00
   OverSubscribe=OK Contiguous=0 Licenses=(null) Network=(null)
```

Slurm as the batch system for CLAIX

Further parameters

project (account)	-A <account> or --account=<account>
shared memory job	--ntasks=1 --cpus-per-task=<numthreads>
distributed memory job	--ntasks=<numtasks>
hybrid job, „r“ MPI ranks, „p“ tasks per node, „t“ threads per task	--ntasks=<r> --tasks-per-node=<p> --cpus-per-task=<t>
Gpu (max 2 per node atm.)	pascal: --gres=gpu:pascal:<numgpus per node> volta: --gres=gpu:volta:<numgpus per node>

[slurm docu] [itc.rwth docu]

Pending Reasons

- None
  - The job has not been in a schedule run of Slurm up to now
- Priority
  - At least one other job has a higher priority and will run first on the same resources
- Resources
  - The job is waiting for resources to become free
- AssocMaxWallDurationPerJobLimit
  - The job requested a longer runtime than it is allowed
- AssocMaxCpuPerJobLimit
  - The job requested more cpus than it is allowed
- JobArrayTaskLimit
  - The job array has more running tasks than allowed
- Dependency
  - The job is waiting for another specific job to end

Projects (--account) and Quota

Slurm as the batch system for CLAIX

Accounts

- Accounts have a **default** partition and **„allowed“** partitions
  - The account **„default“** has **„c18m“** as default partition and is allowed to use **„c16g“** and **„c18g“**
- Submission without a project results in submission to the **„default“** account
- In which projects am I involved?
  - Use „**r_wlm_usage -a <projectname/account>**“
  - Shows
    - Allowed partitions
    - Max usable cores per job
    - Max runtime limit per job
    - Consumed corehours of the last 4 weeks
    - Consumed corehours up to now and total granted corehours
- **r_wlm_usage -q** shows user quota information

Quota

- Projects have a contingent of corehours they are **granted** to use **per month**
  - This is NOT a bank account, you cannot save corehours or time to use it later

- But: you have a „**3-month-window**“ to use
  - This month you could use unused quota from the previous month and „borrow“ quota from the next month.

- If you use more than three times of the grant, you will be scheduled to the so called „low“ priority queue. Your jobs will only start, if they do not hinder „normal“ jobs from starting.

Slurm as the batch system for CLAIX

```
kz743613@login18-1:~/tmp/example $ r_wlm_usage -q
User:                                        kz743613
Status of user:                     RWTH Mitarbeiter
Quota monthly (core-h):                         2000
Remaining  core-h of prev. month:               1331
Consumed   core-h current month:                   5
Consumable core-h (%):                           166
Consumable core-h:                              5326
-------------------------------------------------
Consumed   core-h last 4 weeks:                  674
Consumed   core-h last year:                     826
```

```
You are involved in the following SLURM projects:

Account:                                     supp0001
Type:                                            supp
Start of Accounting Period:                08.05.2021
End of Accounting Period:                  07.05.2022
State of project:                              active
-------------------------------------------------
Quota monthly (core-h):                        100341
Remaining  core-h of prev. month:               58738
Consumed   core-h current month:                 6850
Consumed   core-h last 4 weeks:                 14028
Consumable core-h (%):                            152
Consumable core-h:                             252571
-------------------------------------------------
Total quota (core-h):                      1.200 Mio
Total consumed core-h so far:              0.415 Mio
-------------------------------------------------
Default partition:                               c18m
Allowed partitions:          c16g,c18m,c18g,c16m,c16s
Max. allowed wallclocktime:              120.0 hours
Max. allowed cores per job:                    57600
```

Slurm as the batch system for CLAIX

```
kz743613@login18-1:~/tmp/example $ r_wlm_usage -a supp0001

batchusage from Fri 01.10.2021 00:00:00 to Thu 03.03.2022 23:59:59

 of user kz743613 in corehours:
+====================+============+============+============+============+============+============+============+============+
#          Partition +    Account +   Mar 2022 +   Feb 2022 +   Jan 2022 +   Dec 2021 +   Nov 2021 +   Oct 2021 +      total #
+====================+============+============+============+============+============+============+============+============+
#     c16g - normal |    default |            |            |      12.07 |            |            |            |      12.07  #
+--------------------+------------+------------+------------+------------+------------+------------+------------+------------+
#     c18g - normal |    default |       4.77 |       0.29 |       7.80 |            |            |            |      12.85  #
+--------------------+------------+------------+------------+------------+------------+------------+------------+------------+
#     c18m - normal |    default |       0.53 |     668.70 |     132.21 |            |            |            |     801.44  #
#                   |   supp0001 |            |       0.25 |      47.25 |            |            |            |      47.51  #
+====================+============+============+============+============+============+============+============+============+
#             total |            |       5.30 |     669.24 |     199.33 |            |            |            |     873.87  #
+====================+============+============+============+============+============+============+============+============+
```

Slurm as the batch system for CLAIX

Fair use of compute hardware

- corehours vs. billingvalue

- Example node:
  - 10 cores, 100 GB memory, 2 GPU cards
    -> 10 GB is worth 1 core, 1 GPU is worth 5 cores

- Job uses 1 core, 1 GB memory and 0 GPUs -> billing is 1

Slurm as the batch system for CLAIX

Fair use of compute hardware

- corehours vs. billingvalue

- Example node:
  - 10 cores, 100 GB memory, 2 GPU cards
    -> 10 GB is worth 1 core, 1 GPU is worth 5 cores

- Job uses 10 cores, 1 GB memory and 0 GPUs -> billing is 10

Slurm as the batch system for CLAIX

Fair use of compute hardware

- corehours vs. billingvalue

- Example node:
  - 10 cores, 100 GB memory, 2 GPU cards
    -> 10 GB is worth 1 core, 1 GPU is worth 5 cores

- Job uses 1 core, 1 GB memory and 1 GPUs -> billing is 5

Slurm as the batch system for CLAIX

Fair use of compute hardware

- corehours vs. billingvalue

- Example node:
  - 10 cores, 100 GB memory, 2 GPU cards
    -> 10 GB is worth 1 core, 1 GPU is worth 5 cores

- Job uses 1 core, 90 GB memory and 0 GPUs -> billing is 9

Slurm as the batch system for CLAIX

Fair use of compute hardware

- corehours vs. billingvalue

- Example node:
  - 10 cores, 100 GB memory, 2 GPU cards
    -> 10 GB is worth 1 core, 1 GPU is worth 5 cores

- Job uses 7 cores, 80 GB memory and 1 GPUs -> billing is 8

Thank you for your attention.
Any questions?

Single Core + Single Node

```
#!/usr/local_rwth/bin/zsh

#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --ntasks-per-node=1
#SBATCH --time=00:45:00
#SBATCH --output=prog-out.%j
#SBATCH --error=prog-err.%j
#SBATCH --partition=c18m
#SBATCH -A projectABC


sleep 5

hostname
```

2 Cores + 2 Nodes

```
#!/usr/local_rwth/bin/zsh

#SBATCH --nodes=2
#SBATCH --ntasks=2
#SBATCH --ntasks-per-node=1
#SBATCH --time=00:45:00
#SBATCH --output=prog-out.%j
#SBATCH --error=prog-err.%j
#SBATCH --partition=c18m
#SBATCH -A projectABC


sleep 5

srun -n 2 hostname
```

Many Cores + many Nodes

```
#!/usr/local_rwth/bin/zsh

#SBATCH --nodes=4
#SBATCH --ntasks=192
#SBATCH --ntasks-per-node=48
#SBATCH --time=00:45:00
#SBATCH --output=prog-out.%j
#SBATCH --error=prog-err.%j
#SBATCH --partition=c18m
#SBATCH -A projectABC


sleep 5

hostname
```

???

Slurm as the batch system for CLAIX

X11-Forwarding

- Sad to say, but X11 forwarding is not working for us for now, maybe with a later Slurm version
- Yet, we have „some kind“ of X11 forwarding for you
  - Write your jobscript, but don‘t execute your application, insert a sleep according to the --time parameter
  - Use „guishell batchscript“, a new xterm will be started for you on the starting computenode as soon as the job begins running
    - Please remark, that this is a pure ssh-session, that implies that no SLURM variables are set
    - Your ssh session will still be restricted to the requested resources though
  - Still useful for example for debugging with totalview