



# 703078 PS Parallel Programming SS2022

## Introduction & Administrative Stuff

Philipp Gschwandtner, Tek Chhetri, Simon Lechner


# Organizational Stuff

---


## ▶ Philipp Gschwandtner

- ▶ [philipp.gschwandtner@uibk.ac.at](mailto:philipp.gschwandtner@uibk.ac.at)
- ▶  philgs#1930

## ▶ Tek Chhetri

- ▶ [tek-raj.chhetri@uibk.ac.at](mailto:tek-raj.chhetri@uibk.ac.at)
- ▶  tek#8114

## ▶ Simon Lechner

- ▶ [simon.m.lechner@student.uibk.ac.at](mailto:simon.m.lechner@student.uibk.ac.at)
- ▶  simonlechner#7117

## ▶ Groups 2, 3

- ▶ Tue, 09:15-10:00
- ▶ Tue, 10:15-11:00

## ▶ Group 4

- ▶ Tue, 11:15-12:00

## ▶ Groups 1, 5

- ▶ Tue, 08:15-09:00
- ▶ Tue, 12:15-13:00

## Covid-specific information

---

- ▶ Discord server for weekly proseminar sessions and any discussion
  - ▶ please change your nickname to your full name (can be done per-server)
  - ▶ mute any channels you're not interested in (e.g. of other groups)
  - ▶ make sure you have a working audio setup and that you can share your screen for discussing any measurements, source code, etc.

# More Organizational Stuff

---

## ▶ Prerequisites

- ▶ Interest in parallel programming
- ▶ Programming in C or C++

## ▶ Language

- ▶ Groups 1, 2, 3, 5: German, unless there are non-German speakers?
- ▶ Group 4: English

## ▶ Content

- ▶ General concepts of parallel programming
  - ▶ Concepts apply to many parallel programming models
  - ▶ As an example, we will mainly discuss OpenMP

# Grading: Proseminar

---

- ▶ Weekly assignments, published on OLAT
  - ▶ Link to GitHub
  - ▶ 3 points per week
- ▶ Teamwork is permitted and encouraged
  - ▶ 3 people max. per team
  - ▶ **Every** team member must be able to present and discuss solution
- ▶ Solutions have to be handed in until Mon 17:00!
  - ▶ Solutions of assignments on the LCC2 cluster **must work** on LCC2
  - ▶ Copying solutions (e.g. off the Internet) is acceptable **if cited properly and understood**
  - ▶ Grade is 50 % solutions, 50 % presentations/discussion – both must be  $\geq 50\%$ !

# Literature

---

- ▶ **www.internet.com**
  - ▶ <https://www.openmp.org/resources/> (incl. video tutorials)
  - ▶ stackoverflow
  - ▶ Google
  - ▶ ...
- ▶ **Old school: Printed books**
  - ▶ Let us know and we will look up some references...

# What are we all doing here?

---

- ▶ Discuss key concepts of parallel computing
  - ▶ Hardware **and** software aspects
  - ▶ Multiple non-functional aspects – there's more than just speed
  - ▶ Portability, usability, maintainability, sustainability
- ▶ We still need to actually do some concrete work
  - ▶ (Mostly) OpenMP for implementing and evaluating shared-memory parallelism concepts
  - ▶ We'll also use LCC2 for running experiments



## Hints (not only) for this Course

---

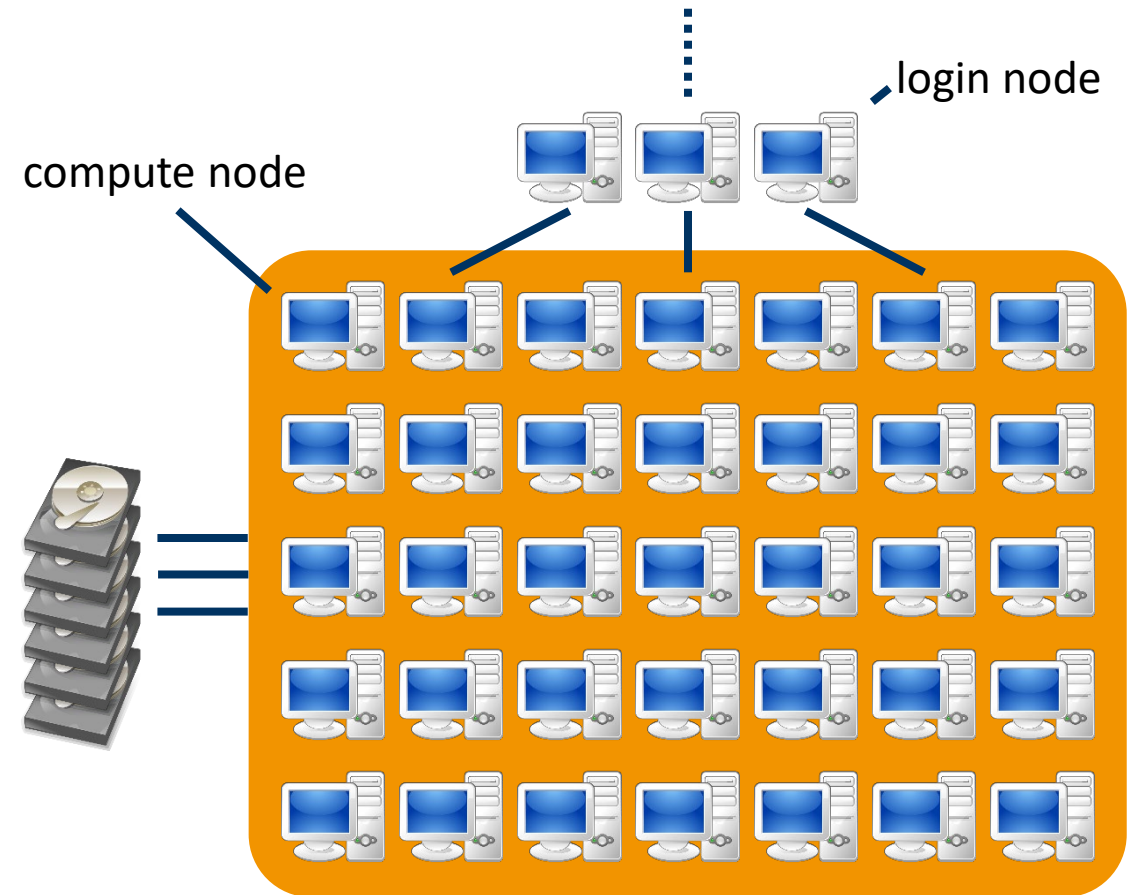
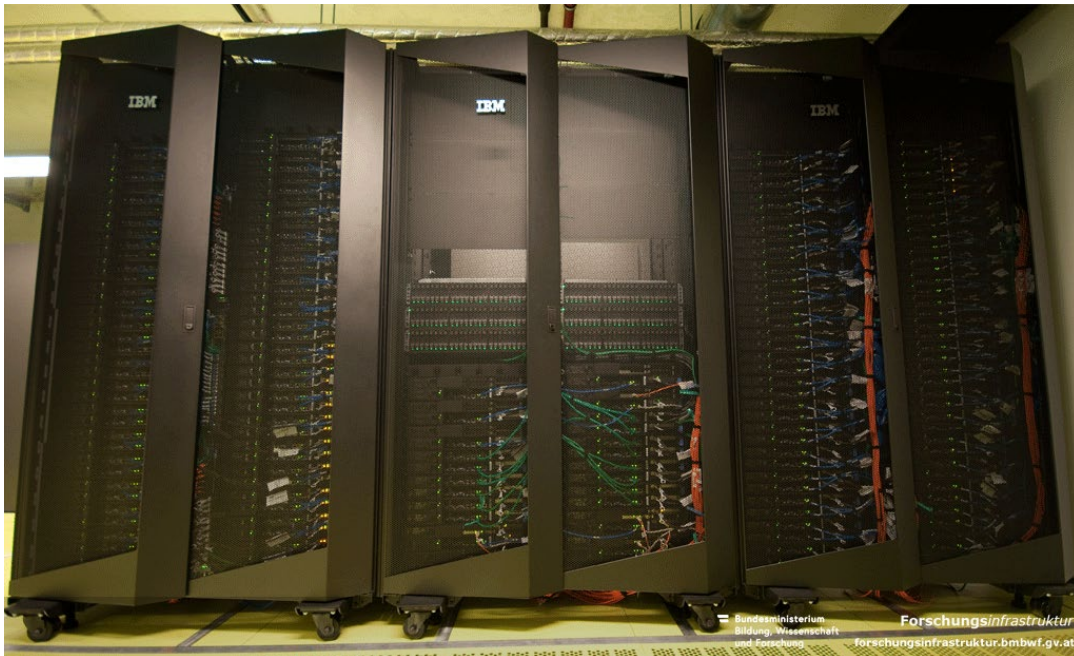
- ▶ choose a suitable source code editor / IDE and choose it wisely!
- ▶ get acquainted with your toolchain
  - ▶ debuggers, version control (git), etc.
- ▶ use common sense and sanity checks!





# Clusters and Supercomputers

- Looks like:



# Get User Credentials, Log in and Change Your Password!

---

- ▶ `ssh cbxxxxxx@lcc2.uibk.ac.at`
- ▶ Change password with `passwd`
- ▶ You are responsible for your account!
  - ▶ don't use these credentials for anything other than this course
  - ▶ coin mining isn't worth it anyways...

# Submission Systems

---

- ▶ Responsible for resource management and job orchestration
  - ▶ used to submit or cancel “jobs”, query their status, get information about cluster, ...
- ▶ Very popular: SLURM
  - ▶ modern, complex but very capable
  - ▶ de-facto standard on most systems these days



- ~~▶ On LCC2: Sun Grid Engine (SGE)
  - ~~▶ older and deprecated ☹️~~
  - ~~▶ switch to SLURM currently in progress~~~~

SLURM!!11!!

# Jobs: Submission, Deletion, Status

---

- ▶ `sbatch name_of_script`
  - ▶ allocates resources
  - ▶ sets up environment
  - ▶ executes application
  - ▶ frees allocation
- ▶ `scancel job_id_list`
  - ▶ terminates application
  - ▶ frees up resources
- ▶ `squ (squeue -u $USER)`
  - ▶ queries for job status
  - ▶ squeue for all users

```
[cb761011@login.lcc2 ~]$ sbatch job.sh
Submitted batch job 184
[cb761011@login.lcc2 ~]$ scancel 184
[cb761011@login.lcc2 ~]$ sbatch job.sh
Submitted batch job 185
[cb761011@login.lcc2 ~]$ squ
```

JOBID	PRIORITY	PARTITION	NAME	USER	STATE	NODES	CPUS	TIME_LIMIT	NODELIST(REASON)
185	504	lva	test	cb761011	RUNNING	1	8	30:00	n002



Questions?

# Image Sources

---

- ▶ LCC2: <https://www.uibk.ac.at/zid/systeme/hpc-systeme/lcc/hardware/>
- ▶ Sandbox: <http://www.googblogs.com/open-sourcing-sandboxed-api/>
- ▶ Cluster Photo: [https://forschungsinfrastruktur.bmbwf.gv.at/de/fi/hpc-compute-cluster-leo3-leo3e\\_513](https://forschungsinfrastruktur.bmbwf.gv.at/de/fi/hpc-compute-cluster-leo3-leo3e_513)
- ▶ SLURM: [https://justjimsthoughts.blogspot.com/2017/07/trivia\\_24.html](https://justjimsthoughts.blogspot.com/2017/07/trivia_24.html)