WARSAW TEAM

Who we are?

The **Warsaw Team** was formed in December 2016 with the mentorship of the Interdisciplinary Centre for Mathematical and Computational Modelling (ICM) at the **University of Warsaw**. Our team has been constantly evolving over the past four years and we are proud that we have represented the Polish students community in all major Student Cluster Competitions around the world including ASC, ISC and SC.

Our team consists of six members - two first-year and four second-year undergraduate students. Our team members come from all backgrounds. We demonstrate diversity in terms of our place of origin, gender and socioeconomic background. Our interests extend from reverse engineering through deep learning to neuroscience.

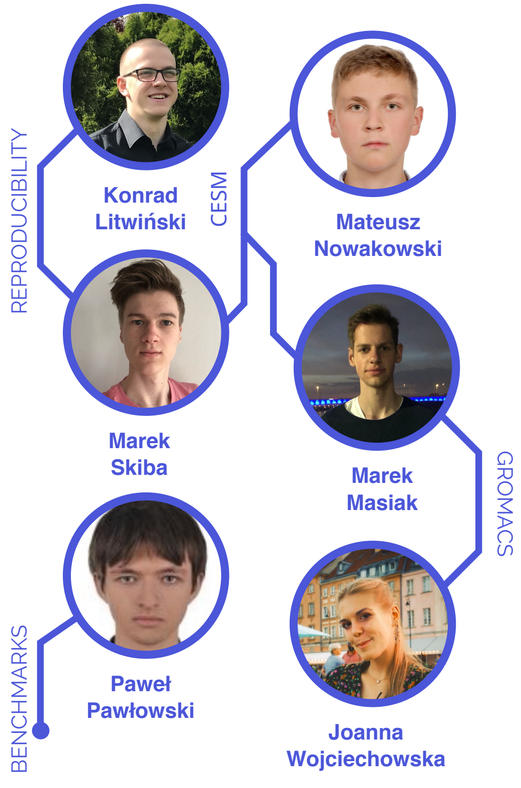
We think **we will win because:**

- We are prepared and extremely motivated
- For most of our team, this is their first competition, which means we are out for blood
- We find high performance computing to be an exciting problem and opportunity to test our knowledge
- We enjoy learning and exploring new things!

Preparation

To best prepare for the upcoming competition, **the team has**

- worked in small groups on specific applications
- tried out many different compiler/MPI configurations to find the optimal setup
- coordinated work in full-team meetings
- collaborated with Microsoft to prepare for the transition from classic HPC cluster infrastructure to the Cloud
- participated in all SC seminars



Strategy

Our strategy for the competition:

- Several CycleCloud clusters to ensure that unrelated tasks don't affect one another
- Custom CycleCloud templates to ensure that the cloud configuration can be adjusted as needed
- Cluster and software configurations we have tested during the training period
- At least one person available at every point during competition who can fix cluster and software configuration issues
- Active monitoring of cloud credit usage and setting of reasonable alerts in the CycleCloud user interface

Since many aspects of the Azure SKU availability are still unknown this section only represents our general plans.

We hope to use the ND40rs v2 VMs for MPI accelerated tasks, where using a GPU gives large benefits if they are available and fit within the budget since only these support Infiniband.

We also plan on using the **NC24r**, **NC24rs v2** or **NC24rs v3** VM series depending on availability. Other GPU instances have significantly slower network connections.

For CPU workloads we plan on using the **HB120rs_v2** VMs, falling to **HC44rs** and **HB60rs** in case of unavailability. We might decide to go with **HB44rs** if Intel CPUs have a significant edge in one of the benchmarks.

For the IO500 benchmark we are considering using **L80s_v2** or CPU workload VMs for the storage cluster (and CPU VMs for clients).

Since the workload is primarily GPU based, we hope to base our architecture on the **ND40rs v2** VMs if the cost and quota allow it. We will use the **NC24r+** series for GPU-first workloads if we need to sacrifice some performance for cost.

SKU	Core	RAM	GPU	Price
ND40rs v2	40	472GiB	8x V100	$26,44 /h
NC24r	24	224GiB	4x K80	$4,75 /h
NC24rs v2	24	448GiB	4x P100	$10,93 /h
NC24rs v3	24	448GiB	4x V100	$14,81 /h
HC44rs	44	352GiB	N/A	$3,48 /h
HB60rs	60	240GiB	N/A	$2,50 /h
HB120rs v2	120	480GiB	N/A	$3,96 /h
L80s_v2	80	640GiB	N/A	$7,49 /h

Software

Software we are using:

- **Cent OS 7.7 & 8.1** as the OS
- **HPCX 2.7.0** as the MPI implementation
- **OpenBLAS** and **Intel MKL**
- **Slurm** for job scheduling
- **Grafana** for monitoring

For HPL & HPCG we used a similar strategy:

- We tested different versions of compilers and OpenMPI libraries for the CPU variant
- We compared the performance difference between physical bare-metal configuration and cloud virtualisation
- We tested the GPU variants of the libraries

For GROMACS:

- We have researched the changes that were brought to GROMACS with the 2020 version and learned how to fully utilize the optimizations
- We have tested with a number of possible configurations to determine the optimal balance between MPI processes and OpenMP, what benefits we can obtain from utilizing GPU accelerators
- We have experimented with numerous additional configurations to get the best performance results

For CESM:

- We have compiled dependencies and CESM
- We have experimented with a number of compsets and resolutions to learn how they influence the overall runtime
- We have performed runs on multiple nodes on our university's cluster

For Reproducibility:

- We have run both the GPU and the CPU version of the code on our university's cluster on all datasets
- We have studied the paper and analysed described optimizations