

# My Latex Example L<sup>A</sup>T<sub>E</sub>Xdocument

- If you start doing scaling/debugging tests, start with the very simplest thing first and the scale up on HPC system.
- Simplest case is single node; 1 CPU, 1GPU. Run this and check the job report in LLView. You can start writing the report on this and get a few pages. You have to understand that case.
- From there on, on single node: work up in separate directions at a time. You could do 1 CPU, 2 GPUs, then 1CPU, 3GPUs. Then you can try 2 CPUs and 1GPU and cover this 'area' in an exponential manner
- once you cover up to 128CPUS and 4GPUs then you can look at the multiple nodes case.
- On 2 nodes, the use 1CPU and 1GPU per node. A rank is '-n 2' in an srun command.
- Then you need to check 'correctness' and 'scaling' and 'load balancing'
- From 26. Juli:
- Q1: Why can I have 4 srns running at the same time on 1 GPU?
- Q2: How do I make it possible or not, that more than 1 executable can utilize 1 GPU at a time?
- Q3: Get in touch with Dr. Diaz
- TODO1: (MUST) Look into the executables (e.g. the optimizer.py and optimizee.py)
- TODO2: (FOR MYSELF) I cannot finish my internship without knowing what MPI and OpenMP, vectorization, threading... These are all concepts in paralelization.
- TODO3:Check the scaling on 1 node. If 4 GPUs makes things run 4 times faster. And if 2 GPUs make things run twice as fast.
- TODO4: At the top level, try to complete the internship. Write down the actual goals and dont forget about them. Keep it simple, but do understand the simple things.

- Write down the contact people: Alper, Sandra Diaz, Meinke
- Try to look at the slides from the HPC course which is 10-12 lectures which runs twice a year. Spend every day 15-30 mins in one of those presentations. This is a crash course on HPC. Those reading could be a good start from that documents. Ask Alex Strube.