

October 29-31, 2024



ALCF Hands-on HPC Workshop

Introduction to SYCL

Thomas Applencour

November 12, 2024

Different Goals, for Different Audiences

SYCL in a nutshell

Hands-on

Different Goals, for Different Audiences

Are you for a Fortran developer?

- Take this presentation of a Hard science fiction film
- Look interesting, but¹ you will never be able to use it.

¹sadly?

Are you for a C++ OpenMP developer?

- Show you a path of pure c++ enlightenment so you can let pragma behind you

Are you for a Kokkos developer?

- Show you a ² lower-level programming model: more control versus maybe a little more verbosity
- At least a alternate Portable Programming Model

²Arguably?

Are you for a developer of a other C++ abstraction layer?

- Please don't.

Are you a Python developer?⁴

- Understand a little bit more what your Python code is doing behind the scene
- Scare you with crazy [=](){}³

³yes, it's a valid C++ syntax

⁴I guess you are not, there is another track for them...

Are you for a developer of cuda?

- Please don't.
- Or at least when you are forced too.⁵

⁵Missing libraries, Missing feature, etc. SYCL give you a way of getting back the native handle when required.
We are not barbarians.

- Choosing a programming model depend of multi-objective optimization problem.
- Just give you a feel of SYCL⁶

⁶And to open issue in <https://github.com/KhronosGroup/SYCL-Docs> if you are unhappy with some of our choice

SYCL in a nutshell

- Pure C++, 2 Implementations (Intel DPCPP, and Adaptivecpp)⁷
- Backed by Khronos⁸, spec development in public
<https://github.com/KhronosGroup/SYCL-Docs>
- CUDA, Hip, Level Zero, OpenCL, FPGA, CPU OpenMP backend⁹
- "Thin C++ layer around native backend"
- BLAS / FFT / Random Number libraries¹⁰

⁷One more implementation than Kokkos and CUDA... Just saying

⁸The Vulkan, SPIRV, OpenCL people

⁹With interopt capabilities

¹⁰via oneMKL

Hands-on

- Let's go together over a set of examples (and improvised exercises)
- Code in the workshop github
`ALCF_Hands_on_HPC_Workshop/programmingModels/SYCL`
- Original (and possibly more up to date) repo:
`https://github.com/argonne-lcf/sycltrain`
- Please read the readme :)
- Don't hesitate to interrupt me / ask question (here or on slack)