alpaka Parallel Programming – Online Tutorial

Lecture 00 – Getting Started with alpaka

Lesson 05: Testing Your alpaka Installation

CASUS
CENTER FOR ADVANCED SYSTEMS UNDERSTANDING

www.casus.science

HZDR HELMHOLTZ ZENTRUM DRESDEN ROSSENDORF
TECHNISCHE UNIVERSITÄT DRESDEN
CBG Max Planck Institute of Molecular Cell Biology and Genetics
UFZ HELMHOLTZ Centre for Environmental Research
Uniwersytet Wrocławski

SPONSORED BY THE
Federal Ministry of Education and Research
STAATSMINISTERIUM FÜR WISSENSCHAFT KULTUR UND TOURISMUS
Freistaat SACHSEN

Lesson 05: Testing Your alpaka Installation

Running the alpaka test suite

- Reconfigure your `build` directory (created in the previous lesson):
  ```
  cmake -DBUILD_TESTING=ON -DCMAKE_BUILD_TYPE=Release ..
  ```
- Build the test suite (note the dot). Adjust the number of parallel compilation jobs to your number of CPU cores and available RAM (requires ~2.5 GiB of **free** memory per job):
  ```
  cmake --build . --parallel 2 --config Release
  ```
- Run the test suite:
  ```
  ctest -C Release
  ```
- Any errors? Please report them: **https://github.com/alpaka-group/alpaka/issues**

Lesson 05: Testing Your alpaka Installation

Accessing alpaka examples

- There are many examples in the `example` directory! You can build them all at once from your `build` directory:

```
cmake -Dalpaka_BUILD_EXAMPLES=ON -DCMAKE_BUILD_TYPE=Release ..
cmake --build . --config Release
```

- Additional examples are available at **https://github.com/alpaka-group/**

Lesson 05: Testing Your alpaka Installation

Advertising your alpaka installation

- Switch to the `vectorAdd` example (from the top-level source directory):
```
cd example/vectorAdd
mkdir build
cd build
```
- If you installed alpaka to a standard location (= default `CMAKE_INSTALL_PREFIX`):
```
cmake -DCMAKE_BUILD_TYPE=Release ..
```
- If you installed alpaka to a non-standard location:
```
cmake -DCMAKE_BUILD_TYPE=Release -Dalpaka_ROOT=/path/to/installed/alpaka ..
```
- Alternative: Set the `CMAKE_PREFIX_PATH` environment variable (note the missing `alpaka`):
  - Linux / macOS: `export CMAKE_PREFIX_PATH=/path/to/installed`
  - Windows: `set CMAKE_PREFIX_PATH=C:\path\to\installed`

Lesson 05: Testing Your alpaka Installation

Building the example

- Build the example (note the dot):

```
cmake --build . --config Release
```

Lesson 05: Testing Your alpaka Installation

Executing the example

- If everything worked correctly you should see the executable somewhere in your `build` tree:
  ```
  ls
  ```
  - Linux / macOS: `CMakeCache.txt CMakeFiles cmake_install.cmake Makefile` **`vectorAdd`**
  - Windows: The executable should be in the `Release` subdirectory of `build`. `cd` to the `Release` directory.
- Execute the example:
  ```
  ./vectorAdd
  ```
  Expected output: `Execution results correct!`
- Unexpected results? Please report them: **https://github.com/alpaka-group/alpaka/issues**

CASUS

CENTER FOR ADVANCED
SYSTEMS UNDERSTANDING

www.casus.science

HZDR HELMHOLTZ ZENTRUM DRESDEN ROSSENDORF

TECHNISCHE UNIVERSITÄT DRESDEN

CBG Max Planck Institute of Molecular Cell Biology and Genetics

UFZ HELMHOLTZ Centre for Environmental Research

Uniwersytet Wrocławski

SPONSORED BY THE

Federal Ministry of Education and Research

STAATSMINISTERIUM FÜR WISSENSCHAFT KULTUR UND TOURISMUS

Freistaat SACHSEN