Welcome to HPCToolKit Training

NERSC & OLCF

March 29, 2021

Introduction

- HPCToolkit is a portable suite of performance tools for measurement and analysis of both CPU and GPU-accelerated applications
  - CPUs: x86_64, ARM, Power
  - GPUs: NVIDIA, Intel, AMD

- Thanks to the HPCToolkit developers from Rice Univ. for providing training to NERSC users (to prepare for Perlmutter) and OLCF users (for use on Summit)
  - Presenters: John Mellor-Crummey, Laksono Adhianto, Keren Zhou
  - Co-aid: Jonathan Anderson, Aaron Cherian, Dejan Grubisic, Mark Krentel, Yumeng Liu, Xiaozhu Meng

Some Logistics

- Users are muted upon joining Zoom (can unmute to speak)
- Please change your name in Zoom session
  - to: first_name last_name
  - Click “Participants”, then “More” next to your name to rename
- Click “CC” to “Show subtitles” or “View Full Transcript”
- Join Slack https://tinyurl.com/hpctoolkit-nersc-slack for Q&A and discussions (instead of Zoom chat)
  - Join “cori-help” or “summit-help” channel
- Slides are uploaded to “presentations” channel
- Examples are available at https://github.com/hpctoolkit/hpctoolkit-tutorial-examples
- Videos will be available after post-processing

Day 1 Agenda, March 29

- Lecture
  - Introduction to HPCToolkit (John Mellor-Crummey)
- Presentations with live demos
  - Using HPCToolkit's Graphical User Interface (Laksono Adhianto)
  - 10-min break
  - Analyzing GPU-accelerated Applications (Keren Zhou)
  - Analyzing CPU Applications (John Mellor-Crummey)
  - 10-min break
- Hands on work by attendees with example codes and/or own applications

Day 2 Agenda, April 2

- Walkthrough of using HPCToolkit with example codes
- 10-min break
- More hands on work and answer questions about experiences with example codes
- Help developers applying HPCToolkit to their own codes

NERSC Cori Usage Info

- Users are added to m3502 (or ntrain) project with GPU access
  - Valid through Apr 11
  - Cori-GPU documentation: https://docs-dev.nersc.gov/cgpu/
- Some GPU and KNL nodes are reserved for the training
  - Mar 29: 11 am - 2 pm, hpc1_gpu, hpc1_knl
  - Apr 2: 10 am - 1 pm, hpc2_gpu, hpc2_knl
- Set two environment variables with above info for Examples in GitHub
- % ssh -Y login_name@cori.nersc.gov
- Recommend to use NX to expedite X-forwarding
  - Necessary for using hpcviewer (GUI) from remote
  - Instructions: https://docs.nersc.gov/connect/nx

Using HPCToolkit on Cori

- Note:
  - Cori CPU: build on login node, run on compute nodes
  - Cori GPU: build and run on compute nodes
- Steps (https://docs.nersc.gov/development/performance-debugging-tools/hpctoolkit)
  - % module purge; module load cgpu (GPU only)
  - % module unload darshan; module load hpctoolkit
  - % build with -g -Ofast for CPU (build with -gopt -fast -gpu=lineinfo for GPU)
  - % srun <...srun flags ..> hpcrun ./mycode.exe (run and measure executable)
  - % hpcstruct … (compute program structure for executable)
  - % hpcstruct --gpucfg … (GPU only, compute program structure for GPU machine code)
  - % hpcprof … (combine measurements with program structure information)
  - % module load hpcviewer
  - % hpcviewer ... (GUI view performance, from NX or ssh -Y)

Cori GPU Example Using HPCSDK Compiler

From a Cori login node via SSH or NX

```
% module purge
% module load cgpu

build and run on a GPU node
% cat my_script
#SBATCH -N 1
#SBATCH -C gpu
#SBATCH -c 10
#SBATCH -G 1
#SBATCH -t 1:00:00
#SBATCH -q shared
#SBATCH --reservation=hpc1_gpu
#SBATCH -A m3502    (or -A ntrain)
% module load hpcsdk/20.11
% module load cuda/11.1.1
< build and run commands ...>

% sbatch my_script
```

```
# sample build commands
nvcc -arch compute_70 -o code_cu code.cu          (CUDA C)
nvfortran -o code_cuf code.cuf                    (CUDA Fortran)
nvc -mp=gpu -gpu=cc70 -fast -o code.exe code.c
                                                  (OpenMP offload, C)
nvfortran -mp=gpu -gpu=cc70 -fast -o code.exe code.F90
                                                  (OpenMP offload, Fortran)
nvc -acc -gpu=cc70 -fast -o code.exe code.c
                                                  (OpenACC, C)
module load openmpi/4.0.3         (and use mpicc, mpicxx,
                  mpif90 to build hybrid MPI/OpenMP codes)

#r sample run commands
export OMP_NUM_THREADS=4
srun -n 1 ./mycode.exe                            (pure OpenMP code)
srun -n xx -c xx --cpu-bind=cores ./code.exe
                                                  (MPI/OpenMP code)
```

Cori CPU Example Using Intel Compiler

From a Cori login node via SSH or NX
<do not run module purge>
<do not module load cgpu>

Build on a login node
% module swap craype-haswell craype-mic-knl
% cc -qopenmp -Ofast -o code.exe code.c
% CC -qopenmp -Ofast -o code.exe code.cc
% ftn -qopenmp -Ofast -o code.exe code.F90

Run on a KNL node
% sbatch my_script

```
% cat my_script

#SBATCH -N 1
#SBATCH -C knl
#SBATCH -q regular
#SBATCH -t 1:00:00
#SBATCH --reservation=hpc1_knl
#SBATCH -A m3502    (or -A ntrain)

export OMP_NUM_THREADS=4
./mycode.exe   (pure OpenMP code)
srun -n xx -c xx --cpu-bind=cores ./code.exe
                    (MPI/OpenMP code)
```

OLCF Summit Usage Info

- Summit reservations have been set up for this training. Users will need the follow in their batch scripts to access the reservation:

  On 3/29/21 #BSUB -U hpctoolkit1

  On 4/2/21 #BSUB -U hpctoolkit2

- Summit documentation is here https://docs.olcf.ornl.gov/systems/summit_user_guide.html

Using HPCToolKit on Summit

Steps

- cd /gpfs/alpine/your_project_ID/scratch (need to write profiles to GPFS)
- module load hpctoolkit
- build with -g (if using PGI use -gopt instead of -g)
- run with: jsrun <jsrun flags> *hpcrun* ./mycode.exe
- Use hpcstruct to connect your binary with the hpctoolkit database for your profiles.
- hpcstruct --gpucfg … (GPU only, compute program structure for GPU machine code)
- use *hpcprof* to combine measurements with program structure information

Viewing profiles:

For Summit users, you will need to download your profile with SCP or Globus and view it with a local copy of HPCviewer on your laptop.

Using HPCViewer at OLCF

Laptop Method:

Instructions for downloading and installing HPC viewer are here:

http://hpctoolkit.org/download.html

Instructions for scp/Globus are here:

https://docs.olcf.ornl.gov/data/transferring.html?highlight=globus

Example Workflow

Instructions for downloading and installing HPC viewer are here:

http://hpctoolkit.org/download.html

1. Install Java 11 on your laptop as directed on http://hpctoolkit.org/download.html . (For example, from AdoptOpenJDK)
2. Find instructions for installing Hpcviwer on your OS on http://hpctoolkit.org/download.html

Example MacOS:

```
mkdir hpctoolkit-gui-latest
cd hpctoolkit-gui-latest
curl -O
http://hpctoolkit.org/download/hpcviewer/latest/hpcviewer-macosx.cocoa.x86_64.zip
unzip hpcviewer-macosx.cocoa.x86_64.zip
```

Example Workflow: HPCViewer

Example MacOS:

- The installation gives you Hpcviewer in:

  hpctoolkit-gui-latest/hpcviewer.app/Contents/MacOS/hpcviewer

- Change to that directory and then launch with *./hpcviewer*
- Navigate in the HPCviewer GUI menus to where your downloaded profiles are located
- Click *Open* to launch.

(Alternately from command line: $ *./hpcviewer ../../../../path_to/hpctoolkit-qs-gpu-cuda.d*)

Example Workflow: SCP

To get your profiles to your laptop:

- On Summit tar the hpc-profile directory that is generated by HPCtoolkit.

```
tar -cvf hpctoolkit-qs-gpu-cuda.tar hpctoolkit-qs-gpu-cuda.d
```

On your Laptop

- You may want to make a directory on your laptop to download all the profile databases into so they are easy to find.
- SCP Retrieving a file from OLCF

```
scp $USER@dtn.ccs.ornl.gov:/path/yourfile.tar .
```

- Untar your file *tar -xvf yourfile.tar*

Example Workflow: Globus

To get your profiles to your laptop:

- You may want to make a directory on your laptop to download all the profile databases into so they are easy to find.
- You do not have to tar directories to move them with globus.
- Follow instructions at https://docs.olcf.ornl.gov/data/transferring.html?highlight=globus to:
- 1. login to globus.org with your OLCF user name and PIN+Passcode
- 2. Install a globus connect personal endpoint on your computer and active it.
  - Make sure to name your endpoint memorably: your_name-laptop
- 3. Activate the *OLCF DTN* endpoint (OLCF username and PIN +Passcode)
- Use the two-panel “File Manager” with *OLCF DTN* and your laptop endpoint to do the transfers.