Graphcore POD64: Getting Started

Bill Arnold
ALCF
arnoldw@anl.gov

June 11, 2024

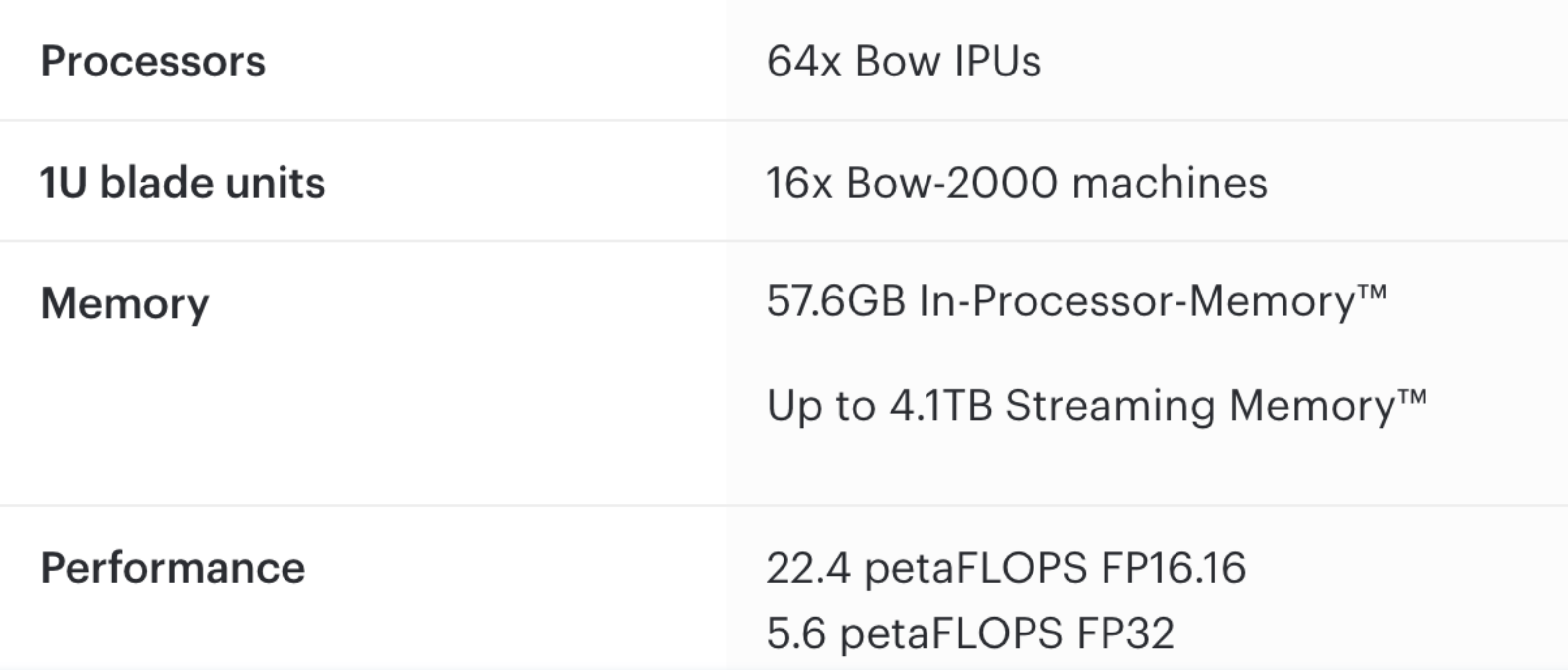
OVERVIEW

- System Overview
- Account access
- Steps to Login
- Environment setup
- Workflow
- Example programs

Graphcore POD-64

https://docs.alcf.anl.gov/ai-testbed/graphcore/system-overview/

POD64

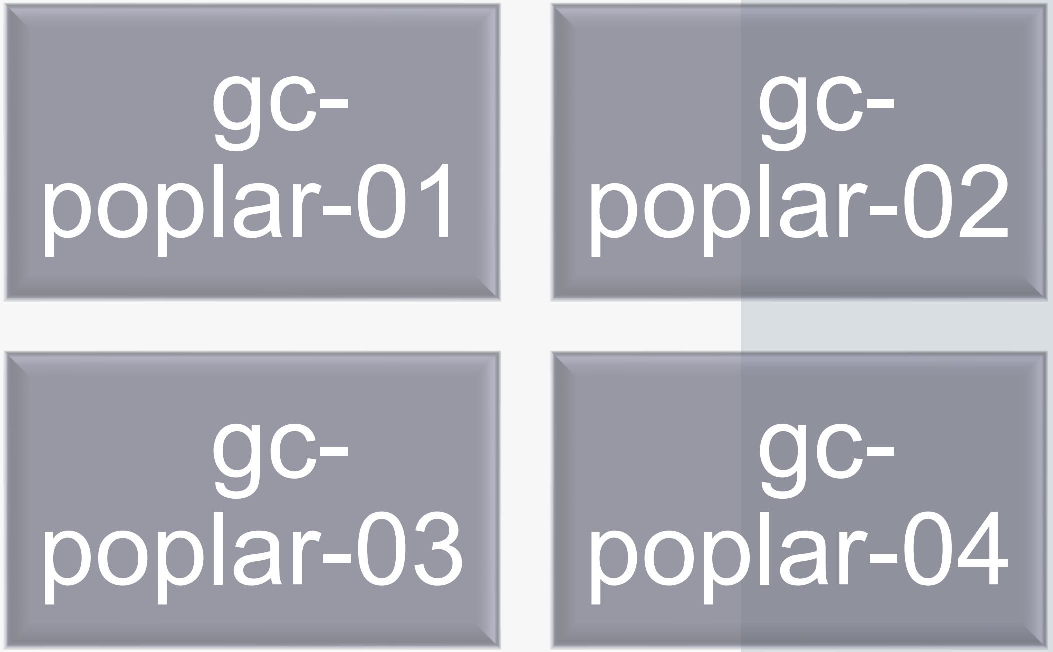


Processors	64x Bow IPUs
1U blade units	16x Bow-2000 machines
Memory	57.6GB In-Processor-Memory™ Up to 4.1TB Streaming Memory™
Performance	22.4 petaFLOPS FP16.16 5.6 petaFLOPS FP32

Graphcore POD-64

https://docs.alcf.anl.gov/ai-testbed/graphcore/system-overview/

POD64



gc-poplar-01

gc-poplar-02

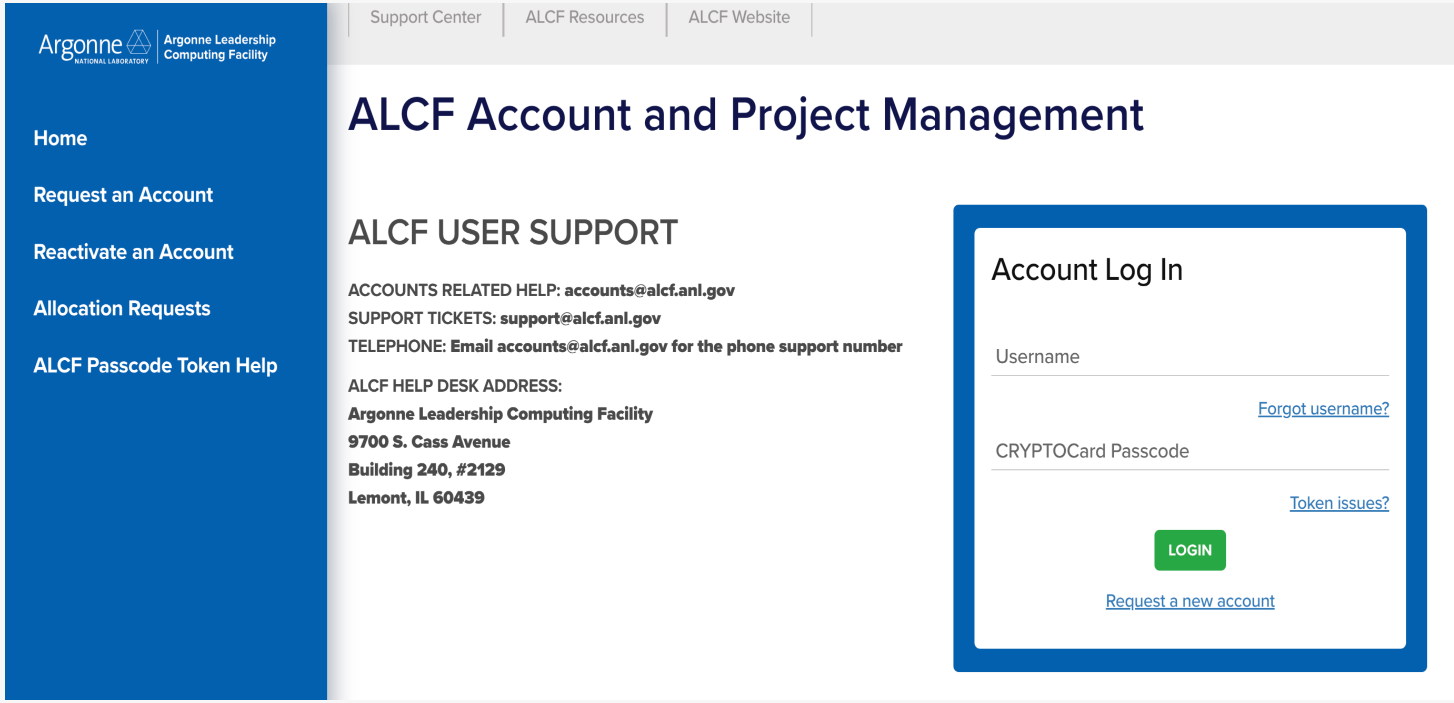
gc-poplar-03

gc-poplar-04

- 4 Nodes
- 64 IPUs

ACCOUNT ACCESS

- **Step 1 : Request an ALCF account**



ALCF Account and Project Management

ALCF USER SUPPORT

ACCOUNTS RELATED HELP: **accounts@alcf.anl.gov**
SUPPORT TICKETS: **support@alcf.anl.gov**
TELEPHONE: **Email accounts@alcf.anl.gov for the phone support number**

ALCF HELP DESK ADDRESS:
Argonne Leadership Computing Facility
9700 S. Cass Avenue
Building 240, #2129
Lemont, IL 60439

Account Log In

Username

Forgot username?

CRYPTOCard Passcode

Token issues?

LOGIN

Request a new account

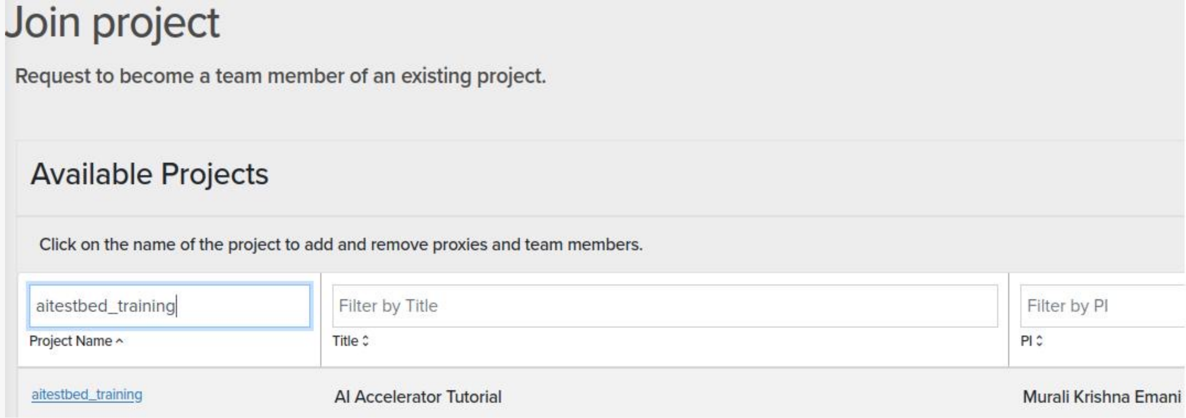
Request an ALCF account on our accounts page.
(https://accounts.alcf.anl.gov/)

Need an active project account.

Help : **accounts@alcf.anl.gov**

ACCOUNT ACCESS

- ## Step 2 : Request to Join project



Join project

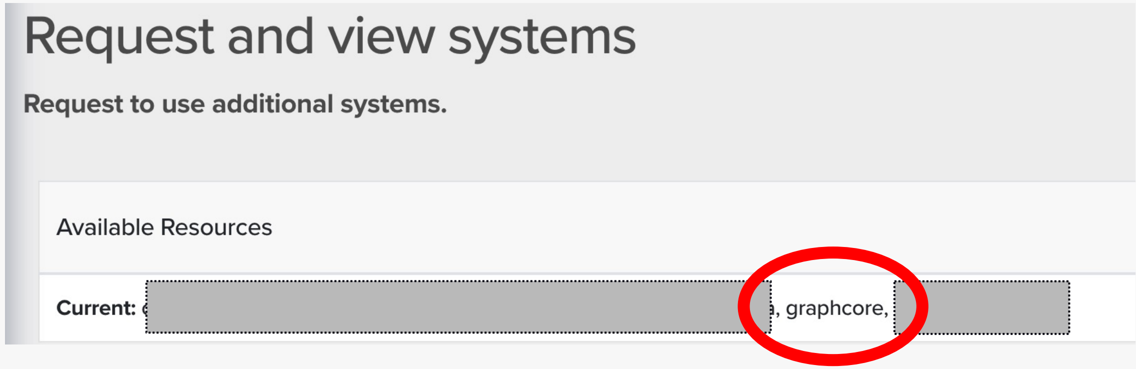
Request to become a team member of an existing project.

Available Projects

Click on the name of the project to add and remove proxies and team members.

aitestbed_training Project Name ^	Filter by Title Title ⇕	Filter by PI PI ⇕
aitestbed_training	AI Accelerator Tutorial	Murali Krishna Emani

Join project under "aitestbed_training".



Request and view systems

Request to use additional systems.

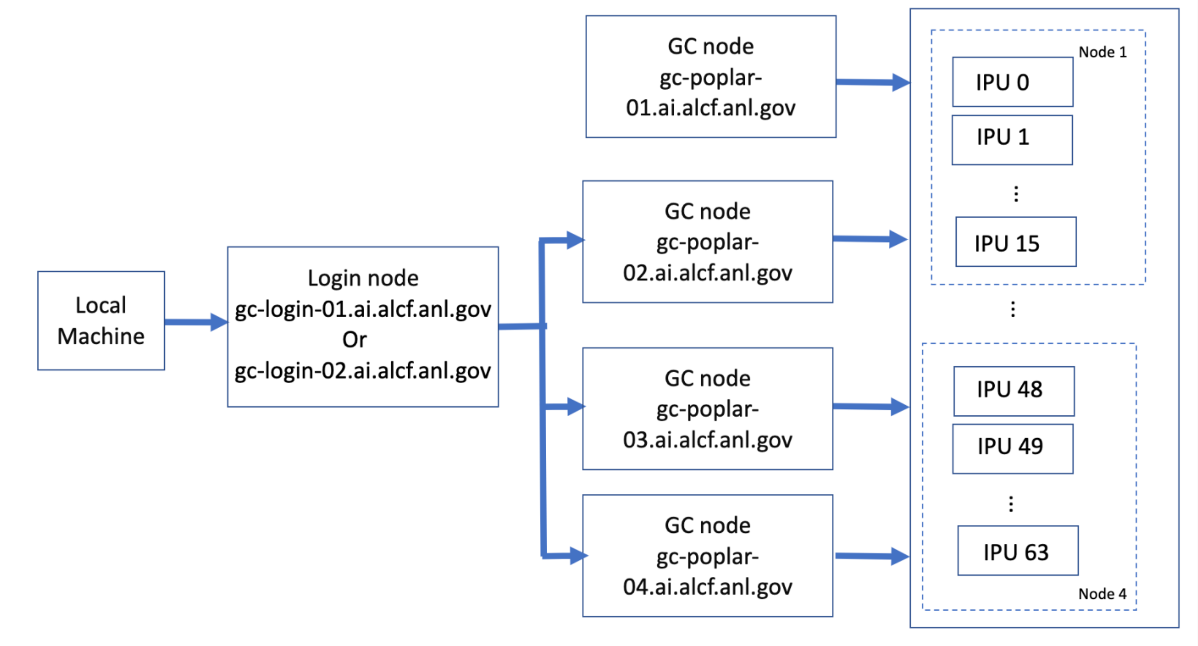
Available Resources

Current: [redacted], graphcore, [redacted]

Check for "graphcore" under Request and view systems

Connection and Login

- **Step 3 : Login to POD64.**



Local Machine

Login node
gc-login-01.ai.alcf.anl.gov
Or
gc-login-02.ai.alcf.anl.gov

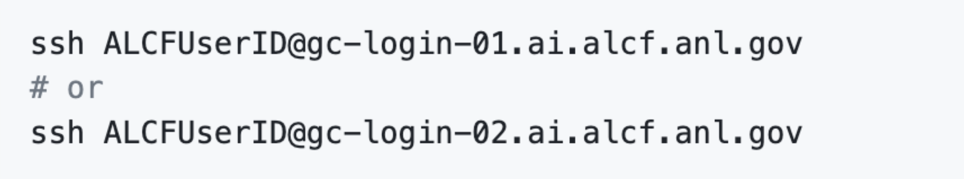
GC node
gc-poplar-01.ai.alcf.anl.gov

GC node
gc-poplar-02.ai.alcf.anl.gov

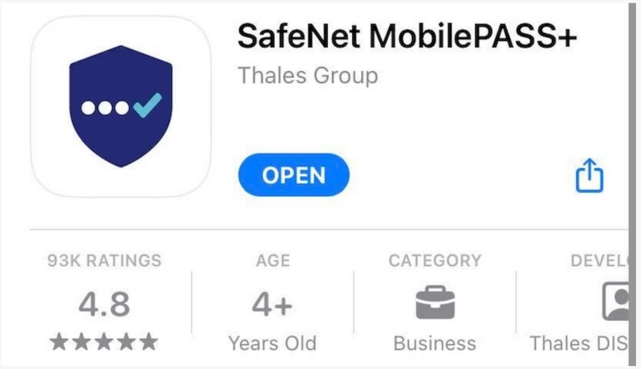
GC node
gc-poplar-03.ai.alcf.anl.gov

GC node
gc-poplar-04.ai.alcf.anl.gov

From local machine to login node :



```
ssh ALCFUserID@gc-login-01.ai.alcf.anl.gov
# or
ssh ALCFUserID@gc-login-02.ai.alcf.anl.gov
```

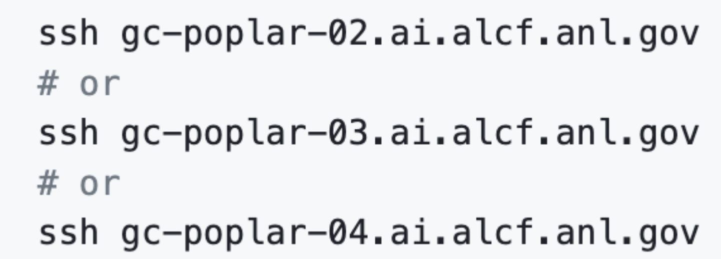


SafeNet MobilePASS+
Thales Group

https://docs.alcf.anl.gov/ai-testbed/graphcore/getting-started/

gc-poplar-01.ai.alcf.anl.gov is not accessible to users. However, its IPU resources are assigned to slurm tasks.

From login node to a POD16 node :

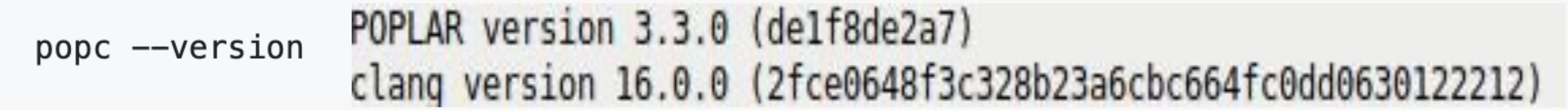


```
ssh gc-poplar-02.ai.alcf.anl.gov
# or
ssh gc-poplar-03.ai.alcf.anl.gov
# or
ssh gc-poplar-04.ai.alcf.anl.gov
```

Environment setup

- Poplar SDK at /software/graphcore/poplar_sdk/. Default poplar enabled at login.

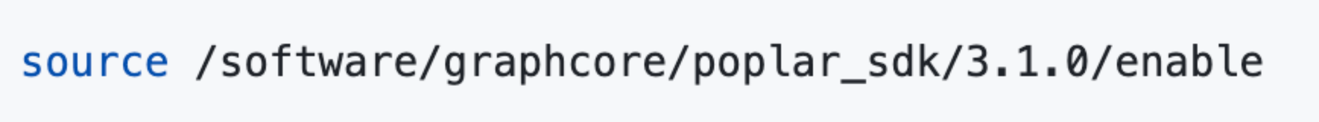
*Latest = 3.3.0



```
popc --version
```

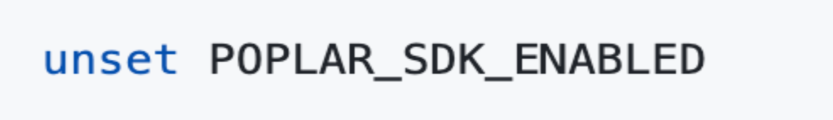
```
POPLAR version 3.3.0 (de1f8de2a7)
clang version 16.0.0 (2fce0648f3c328b23a6cbc664fc0dd0630122212)
```

- To enable a non-3.3.0 SDK



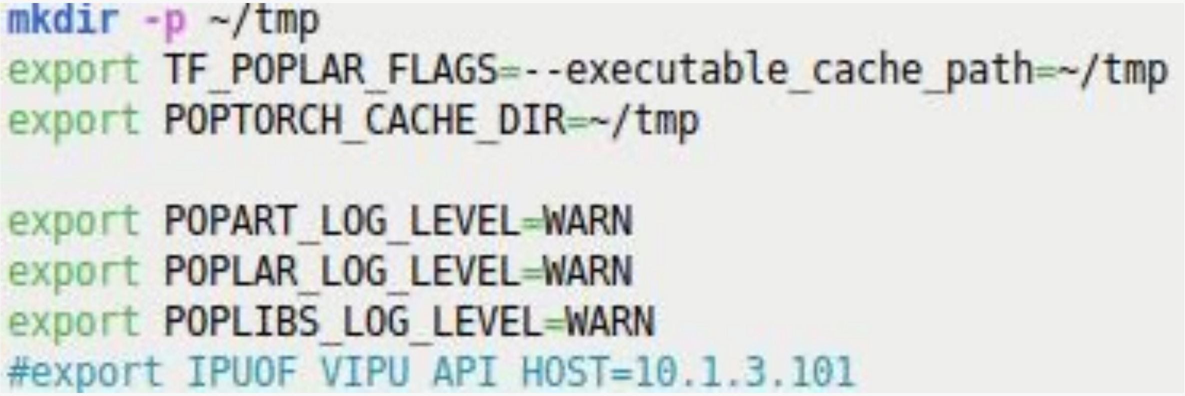
```
source /software/graphcore/poplar_sdk/3.1.0/enable
```

- To unset SDK



```
unset POPLAR_SDK_ENABLED
```

- Environmental variables

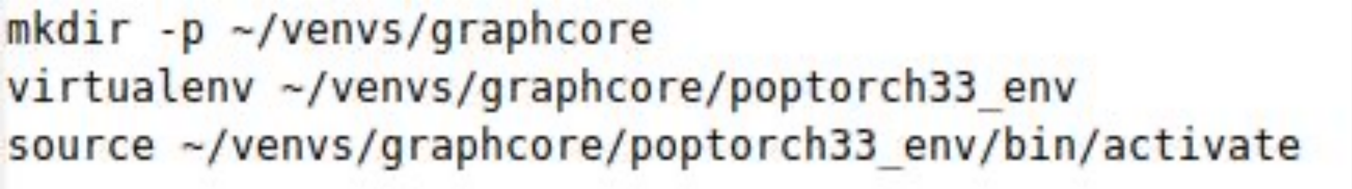


```
mkdir -p ~/tmp
export TF_POPLAR_FLAGS=--executable_cache_path=~/tmp
export POPTORCH_CACHE_DIR=~/tmp

export POPART_LOG_LEVEL=WARN
export POPLAR_LOG_LEVEL=WARN
export POPLIBS_LOG_LEVEL=WARN
#export IPUOF_VIPU_API_HOST=10.1.3.101
```

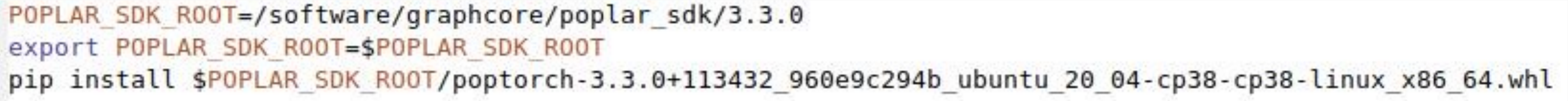
Environment setup

- PyTorch environment
  - Create a virtual environment



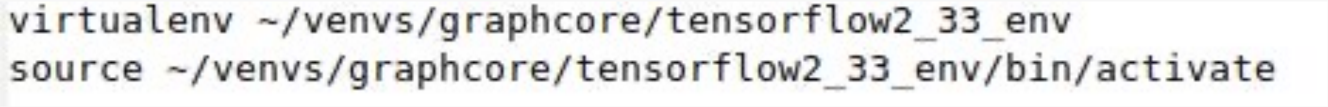
```
mkdir -p ~/venvs/graphcore
virtualenv ~/venvs/graphcore/poptorch33_env
source ~/venvs/graphcore/poptorch33_env/bin/activate
```

  - Install PopTorch



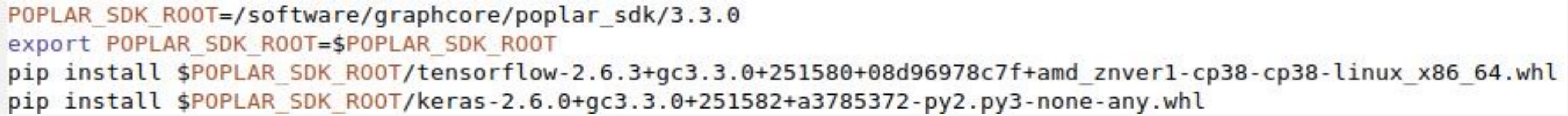
```
POPLAR_SDK_ROOT=/software/graphcore/poplar_sdk/3.3.0
export POPLAR_SDK_ROOT=$POPLAR_SDK_ROOT
pip install $POPLAR_SDK_ROOT/poptorch-3.3.0+113432_960e9c294b_ubuntu_20_04-cp38-cp38-linux_x86_64.whl
```

- TensorFlow environment
  - Create a Tensorflow virtual environment



```
virtualenv ~/venvs/graphcore/tensorflow2_33_env
source ~/venvs/graphcore/tensorflow2_33_env/bin/activate
```

  - Install Tensorflow and Keras

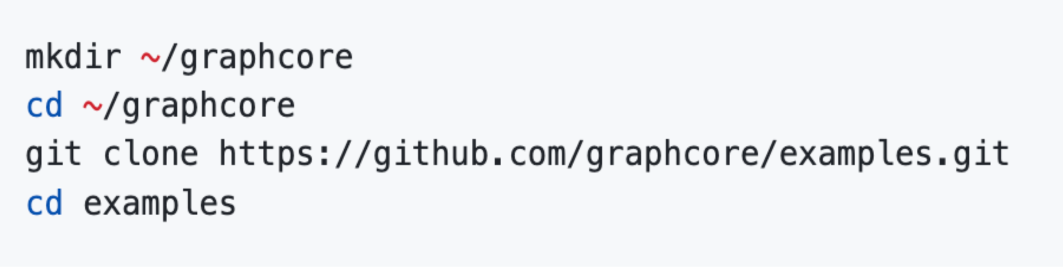


```
POPLAR_SDK_ROOT=/software/graphcore/poplar_sdk/3.3.0
export POPLAR_SDK_ROOT=$POPLAR_SDK_ROOT
pip install $POPLAR_SDK_ROOT/tensorflow-2.6.3+gc3.3.0+251580+08d96978c7f+amd_znver1-cp38-cp38-linux_x86_64.whl
pip install $POPLAR_SDK_ROOT/keras-2.6.0+gc3.3.0+251582+a3785372-py2.py3-none-any.whl
```

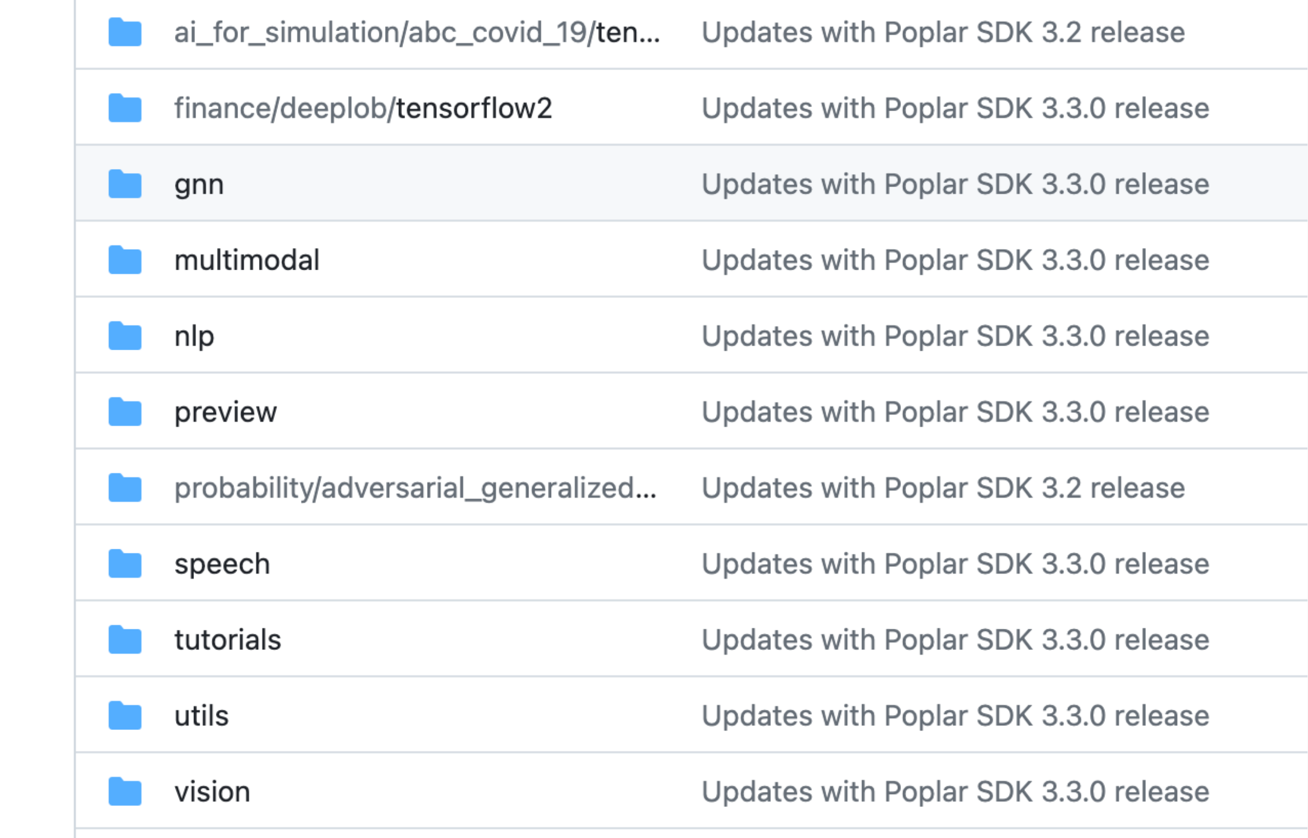
- See https://docs.alcf.anl.gov/ai-testbed/graphcore/virtual-environments/

Workflow

- **Compile + Run**
  - Command compiles and runs the training of the model.
  - Recompiles only if model parameters change or if the cache dir path is not set.
  - Graphcore repository



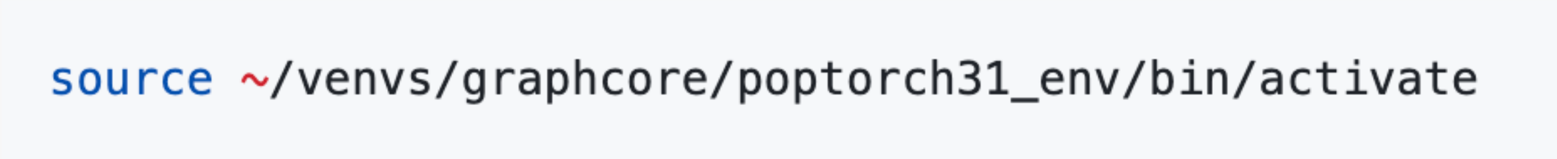
```
mkdir ~/graphcore
cd ~/graphcore
git clone https://github.com/graphcore/examples.git
cd examples
```



ai_for_simulation/abc_covid_19/ten...	Updates with Poplar SDK 3.2 release
finance/deeplob/tensorflow2	Updates with Poplar SDK 3.3.0 release
gnn	Updates with Poplar SDK 3.3.0 release
multimodal	Updates with Poplar SDK 3.3.0 release
nlp	Updates with Poplar SDK 3.3.0 release
preview	Updates with Poplar SDK 3.3.0 release
probability/adversarial_generalized...	Updates with Poplar SDK 3.2 release
speech	Updates with Poplar SDK 3.3.0 release
tutorials	Updates with Poplar SDK 3.3.0 release
utils	Updates with Poplar SDK 3.3.0 release
vision	Updates with Poplar SDK 3.3.0 release

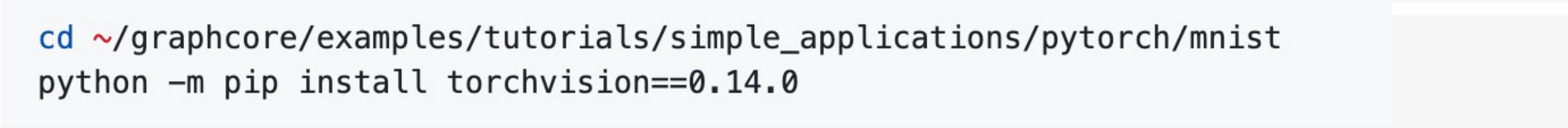
Example program - mnist

- Activate the environment :



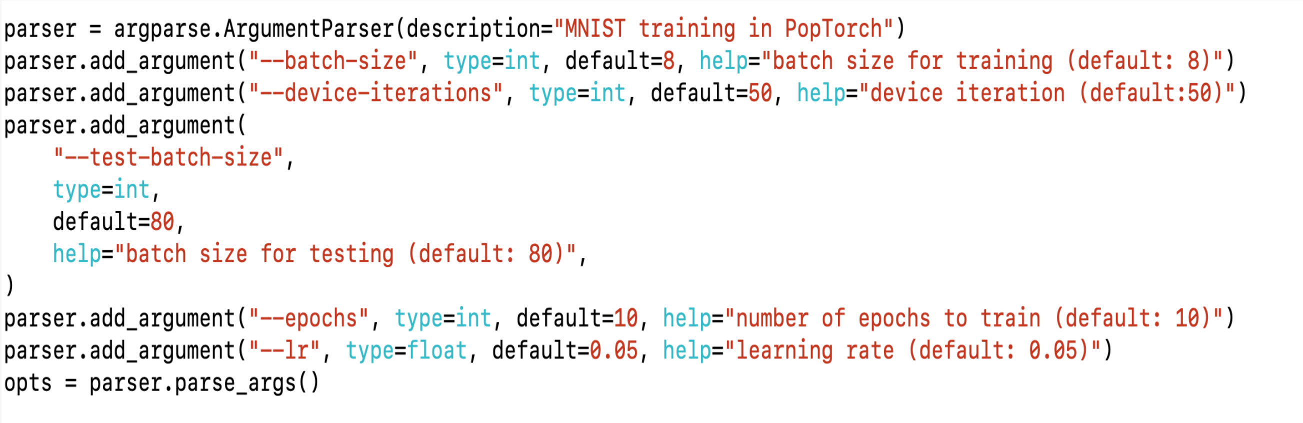
```
source ~/venvs/graphcore/poptorch31_env/bin/activate
```

- Path :

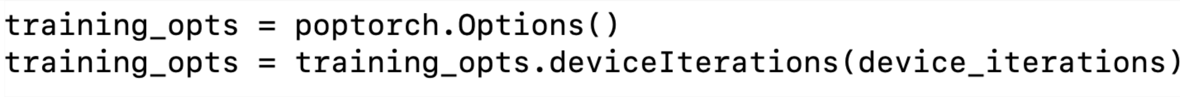


```
cd ~/graphcore/examples/tutorials/simple_applications/pytorch/mnist
python -m pip install torchvision==0.14.0
```

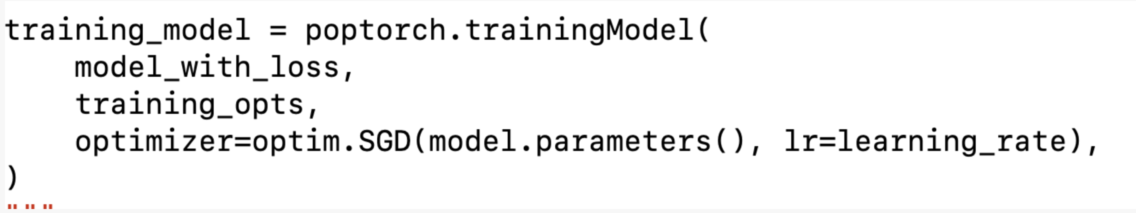
Pytorch -> PopTorch



```
parser = argparse.ArgumentParser(description="MNIST training in PopTorch")
parser.add_argument("--batch-size", type=int, default=8, help="batch size for training (default: 8)")
parser.add_argument("--device-iterations", type=int, default=50, help="device iteration (default:50)")
parser.add_argument(
    "--test-batch-size",
    type=int,
    default=80,
    help="batch size for testing (default: 80)",
)
parser.add_argument("--epochs", type=int, default=10, help="number of epochs to train (default: 10)")
parser.add_argument("--lr", type=float, default=0.05, help="learning rate (default: 0.05)")
opts = parser.parse_args()
```

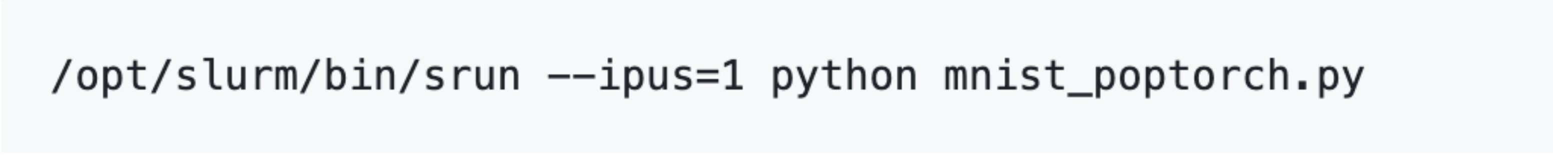


```
training_opts = poptorch.Options()
training_opts = training_opts.deviceIterations(device_iterations)
```

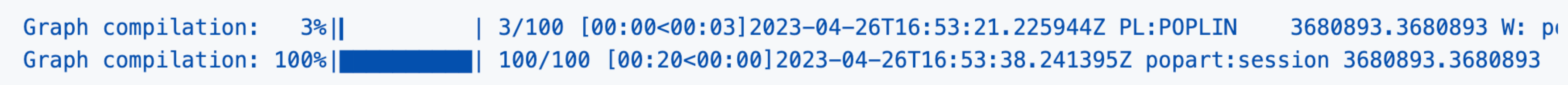


```
training_model = poptorch.trainingModel(
    model_with_loss,
    training_opts,
    optimizer=optim.SGD(model.parameters(), lr=learning_rate),
)
"""
```

- Compile and Run



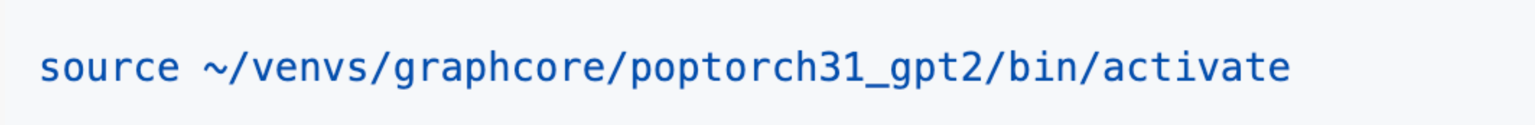
```
/opt/slurm/bin/srun --ipus=1 python mnist_poptorch.py
```



```
Graph compilation:   3%|▌          | 3/100 [00:00<00:03]2023-04-26T16:53:21.225944Z PL:POPLIN    3680893.3680893 W: p(
Graph compilation: 100%|██████████| 100/100 [00:20<00:00]2023-04-26T16:53:38.241395Z popart:session 3680893.3680893
```

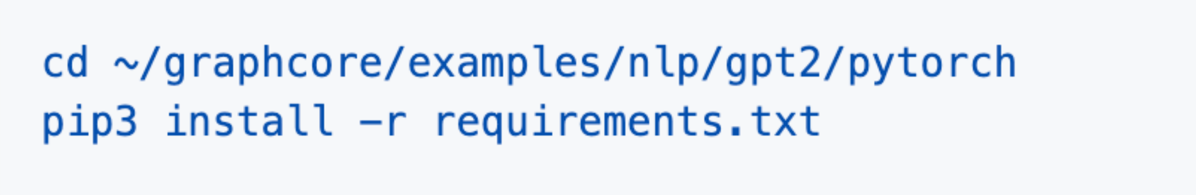
Example program – POD16 GPT2

- Path : https://github.com/graphcore/examples/tree/master/nlp/gpt2/pytorch
- Activate environment :



```
source ~/venvs/graphcore/poptorch31_gpt2/bin/activate
```

- Install packages :



```
cd ~/graphcore/examples/nlp/gpt2/pytorch
pip3 install -r requirements.txt
```

- Run command :

```
/opt/slurm/bin/srun --ipus=16 python train_gpt2.py --model gpt2 --ipus-per-replica 4 --replication-factor 4 --gradient-accumulation 2048 --device-iterations 8 --batch-size 1 --layers-per-ipu 0 4 4 4 --matmul-proportion 0.15 0.15 0.15 0.15 --max-len 1024 --optimizer AdamW --learning-rate 0.00015 --lr-schedule cosine --lr-warmup 0.01 --remap-logit True --enable-sequence-serialized True --embedding-serialization-factor 4 --recompute-checkpoint-every-layer True --enable-half-partials True --replicated-tensor-sharding True --dataset 'generated' --epochs 10
```

Global batch size = (micro)batch-size * gradient-accumulation * replication-factor

Utility commands

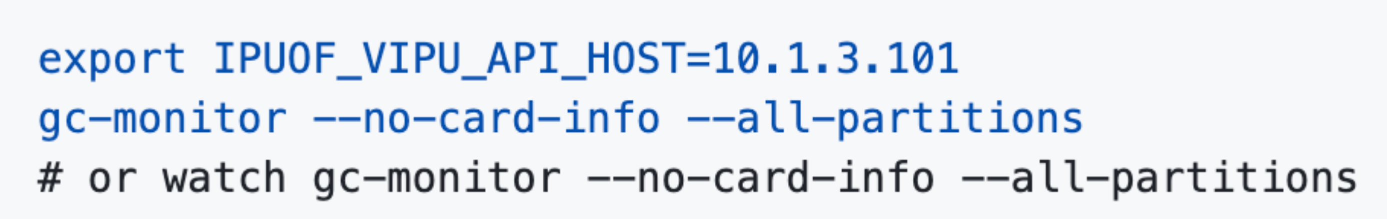
srun - run scripts in parallel.

squeue – jobs in the slurm queue.

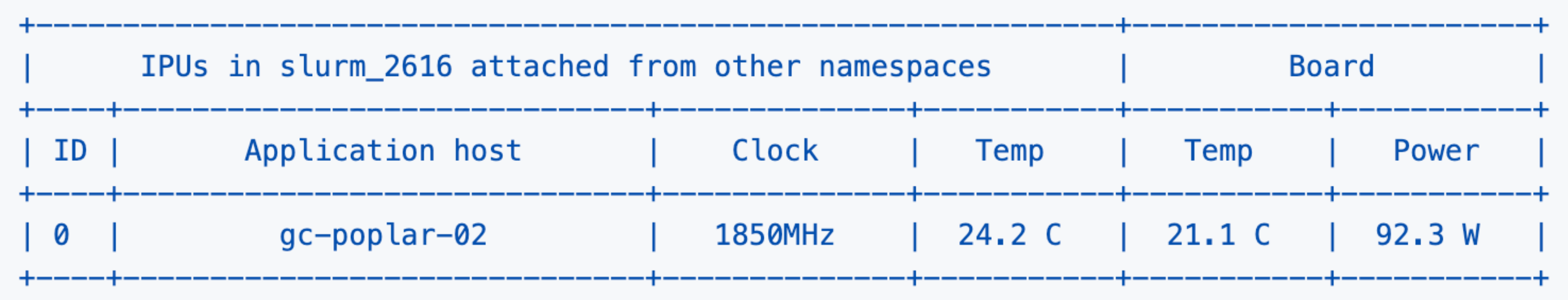
scancel - cancel an existing job.

sinfo - view partition and node information

gc-monitor -



```
export IPUOF_VIPU_API_HOST=10.1.3.101
gc-monitor --no-card-info --all-partitions
# or watch gc-monitor --no-card-info --all-partitions
```



```
+-----------------------------------------------------------+-----------------------+
|   IPUs in slurm_2616 attached from other namespaces   |         Board         |
+----+----------------------------+-------------+-----------+-----------+-----------+
| ID |      Application host      |    Clock    |    Temp   |    Temp   |   Power   |
+----+----------------------------+-------------+-----------+-----------+-----------+
| 0  |        gc-poplar-02        |   1850MHz   |  24.2 C   |  21.1 C   |  92.3 W   |
+----+----------------------------+-------------+-----------+-----------+-----------+
```

Important directory paths and links

- Poplar SDK - https://docs.graphcore.ai/projects/sdk-overview/en/latest/overview.html?highlight=poptorch#introduction

- PyTorch for the IPU: User Guide - https://docs.graphcore.ai/projects/poptorch-user-guide/en/latest/index.html
- Targetting the IPU from Tensorflow 2 - https://docs.graphcore.ai/projects/tensorflow-user-guide/en/latest/index.html
- IPU programming guide - https://docs.graphcore.ai/projects/ipu-programmers-guide/en/latest/
- Examples Github Repo - https://github.com/graphcore/examples

- /mnt/localdata , /software/graphcore/projects/datasets/ , /software/graphcore/poplar_sdk

-

Allocation programs

Director’s Discretionary (DD) awards support various project objectives from scaling code to preparing for future computing competition to production scientific computing in support of strategic partnerships.

Graphcore system is available for Director’s Discretionary (DD) allocations.
Allocation Request Form

Allocations also available under INCITE, ALCC, and ALCF Data Science programs

THANK YOU

www.anl.gov

Argonne NATIONAL LABORATORY