
Benzina Compilation Documentation

Release 0.0.1

Olexa Bilaniuk, Satya Ortiz-Gagne

Mar 21, 2019

Contents

1	Compile Benzina on Mila's Cluster	3
2	Compile Benzina on Cedar	5
3	Links to the project	7
4	Indices and tables	9

Requirements for compilation:

- [Python](#) 3.5 or later
- [Meson](#) 0.49 or later
- [Ninja](#)
- [Nauka](#) 0.0.11 or later
- [Numpy](#) 1.10 or later

Requirements to use the PyTorch interface:

- [PyTorch](#) 0.4.1 or later

CHAPTER 1

Compile Benzina on Mila's Cluster

To compile Benzina on the Mila's cluster, most of the commands can be executed on one of the login nodes.

Note: It is recommended to setup a virtual environment before compiling and using Benzina.

Clone the Nauka and Benzina projects

```
$ git clone https://github.com:obilaniu/Nauka.git
$ git clone https://github.com:obilaniu/Benzina.git
```

Compile and install Nauka

```
$ cd Nauka
$ python setup.py install
```

Install the dependencies of Benzina

```
$ pip install meson ninja
```

Note: To use the PyTorch interface, you will also need to install PyTorch:

```
$ pip install torch
```

Compile and install Benzina

Request a GPU on the cluster:

```
$ sinter --gres=gpu
```

Then, compile and install Benzina:

```
$ cd Benzina
$ python setup.py install
```


CHAPTER 2

Compile Benzina on Cedar

To compile Benzina on the Cedar cluster, most of the commands must be executed on the login nodes.

Note: It is recommended to setup a virtual environment before compiling and using Benzina.

Load the Python module

```
$ module load python/3.6.3
```

Clone the Nauka and Benzina projects

```
$ git clone https://github.com:obilaniu/Nauka.git
$ git clone https://github.com:obilaniu/Benzina.git
```

Compile and install Nauka

Install the Nauka's dependency then compile and install Nauka:

```
$ pip install --no-index numpy
$ cd Nauka
$ python setup.py install
```

Install the dependencies of Benzina

Ninja requires *SciKit-build* to compile itself. To prevent *SciKit-build* from installing the most recent version of *setup-tools* and use instead the version provided in the Cedar environment, create a constraints files and use it while installing *SciKit-build*:

```
$ echo 'setuptools>=27.2.0,<=28.8.0' > pip_constraints
$ pip install -c pip_constraints scikit-build
```

Then, install Meson and Ninja:

```
$ pip install meson ninja
```

Note: To use the PyTorch interface, you will also need to install PyTorch:

```
$ pip install torch
```

Compile and install Benzina

Request a GPU on the cluster:

```
$ salloc --time=0:10:0 --account=account_id --gres=gpu:1
```

Load the CUDA module:

```
$ module load cuda/10
```

Note: On Cedar, only `cuda/9` and `cuda/10` are compatible since the module `cuda/8` comes with an incompatible version of Video Codec SDK.

Then, compile and install Benzina:

```
$ cd Benzina
$ CUDA_HOME=${CUDA_HOME%:*} python setup.py install
```

Note: `${CUDA_HOME%:*}` will trim the `:` and what follows in the variable

CHAPTER 3

Links to the project

GitHub: <https://github.com/obilaniu/Benzina>

Documentation:

CHAPTER 4

Indices and tables

- `genindex`
- `modindex`
- `search`