Accelerated computing in EGI Federated Cloud

Viet Tran
Institute of Informatics SAS, Slovakia

eosc-hub.eu
@EOSC_eu
• Applications need GPUs
  - Computational intensive apps, especially with vectors and matrices
  - Image processing, machine learning, deep learning

• Examples of commercial clouds providing GPUs
  - Amazon EC2, Google Cloud, NVIDIA, …
Cloud providers with GPUs in EGI Federated cloud

- Institute of Informatics SAS, Slovakia (IISAS)
  - IISAS-GPUCloud
  - IISAS-Nebula
- Instituto de Física de Cantabria CSIC, Spain (IFCA-CSIC)
  - IFCA-LCG2
- Laboratório de Instrumentação e Física Experimental de Partículas, Portugal (LIP-INCD)
  - NCG-INGRID-PT
- CESNET, Czech republic
  - CESNET-MCC
- See [https://docs.egi.eu/users/cloud-compute/gpgpu/](https://docs.egi.eu/users/cloud-compute/gpgpu/) for complete site information
How to use GPU in EGI Federated Cloud

• Prerequisite:
  - Have an account in EGI CheckIn
    ▪ Log in or register: https://aai.egi.eu/
  - Be member of a VO with GPU supported
    ▪ If not having membership, join acc-comp.egi.eu for testing or order access via EGI Marketplace (see next slide)
  - Have an application using GPUs
    ▪ Docker container format preferred

• Basic usage:
  - Create a VM with GPU flavor and image (e.g. via Horizon dashboard)
  - Login and execute apps on the VM (e.g. “docker run --gpus=all ....”)
  - Delete VM when done and release resources for others
• Intended for **short-term** use of GPU for **demo, testing and development** purposes
• Supported on IISAS-GPUCloud.
• VO Id Card:  
• Link to join:  
  [https://aai.egi.eu/registry/co_petitions/start/coef:35](https://aai.egi.eu/registry/co_petitions/start/coef:35)
• This VO cannot be used for long-term execution. For **long-term production execution**, use EGI Marketplace to order compute instances or negotiate with sites directly  
• For convenience, a VM image with GPU driver and Docker preinstalled is created and frequently updated: NVIDIA Docker (CentOS 7)
• See https://appdb.egi.eu/store/vappliance/nvidia.docker.centos.7

• Other (vanilla) images can be used for VM with GPUs, but users need to install the GPU driver on VM after creating
  - Different Linux distros have different ways, check official docs
    ▪ e.g. on Ubuntu: “sudo ubuntu-drivers autoinstall”
  - Reboot VM after installing the GPU driver
Log in to site via Openstack Horizon dashboard
- Use the dashboard of a specific site
  - Check site info at [https://docs.egi.eu/users/cloud-compute/gpgpu/](https://docs.egi.eu/users/cloud-compute/gpgpu/)
  - Or FedCloud common dashboard at: [https://dashboard.fedcloud.eosc-synergy.eu](https://dashboard.fedcloud.eosc-synergy.eu)

Select the VO with GPU (if more VOs supported)
Click on “Launch instance”
- Select flavor with GPU (see site info in previous slides)
- (Optional) Select image with GPU driver pre-installed
- And launch
Delete the VM after successful test
Use GPU in EGI FedCloud via Infrastructure manager

- Infrastructure Manager (IM) was developed by the GRyCAP at Universitat Politècnica de València, Spain
- Can create more complex system, e.g. Kubernetes cluster with GPU
- Access via https://appsgrycap.i3m.upv.es:31443/im-dashboard/

- For creating a Kubernetes cluster with GPUs via IM, just explicitly define flavor name with GPU in IM dashboard, and select `nvidia_support` to “True”
Docker container is strongly recommended way for application delivery
- simple installation, avoiding software conflict, many ready-to-use containers available

Prerequisite: correct GPU driver and plugin is installed (e.g. using the mentioned NVIDIA Docker image)

For executing container with GPU in Docker version \( \geq 19.03 \)
- `docker run --gpus all` option container command

For older versions of Docker
- `docker run --runtime=nvidia` option container command
- `nvidia-docker run` option container command
• Prerequisite:
  - Have an account in EGI CheckIn and membership of a VO with GPU support
  - Have some demo code using tensorflow (let’s say “demo.py”)

• Note: nothing related to GPUs needed in the demo code, tensorflow takes care all about GPUs

• Good sources of demo code:
  https://www.tensorflow.org/tutorials/quickstart/beginner
Login to IISAS-GPUCloud via [https://nova3.ui.savba.sk/horizon](https://nova3.ui.savba.sk/horizon) using EGI CheckIn account

Create a VM with `gpu1cpu6` flavor and **NVIDIA Docker (CentOS 7)** image

Assign a public IP address to the VM and login to the VM via ssh

Copy your code (“`demo.py`”) and data (if any) to working directory

Execute your code:

```
"sudo docker run --gpus all -it --rm -v $PWD:/tmp -w /tmp tensorflow/tensorflow:latest-gpu python ./demo.py"
```

You can open second terminal, execute “`nvidia-smi -l`” and check GPU utilization during training
On VM, execute:

```
sudo docker run --gpus all -p 8888:8888 -it --rm tensorflow/tensorflow:latest-gpu-jupyter
```

On your client (laptop/desktop), make SSH tunnel to VM:

```
ssh -f centos@$VM_PUBLIC_IP -L 8888:localhost:8888 -N
```

- See [How to access Jupyter via SSH](#) for more details

In your browser, open [http://localhost:8888/](http://localhost:8888/)

Login to Jupyter using token (from the output of the docker run command)

Try to execute an included demo notebook, (e.g. “classification.ipynb”)

Check the output from “nvidia-smi -l” on the second terminal to see GPU utilization
How fast is GPU in comparison with CPU?

- Short answer: very fast
- Long answer: depending on many factors: algorithm, problem size, type of CPU and GPU, …
  - Even with standard gaming GPUs (NVIDIA GTX, RTX), some can reach 6-10x speedup
  - Specialized GPUs (V100, A100) are still faster but very expensive
  - Extensive testing with different GPUs
    - [https://timdettmers.com/2020/09/07/which-gpu-for-deep-learning/](https://timdettmers.com/2020/09/07/which-gpu-for-deep-learning/)
  - GPU memory capacity may be the limit factor
How to change my code to use GPUs

• The first attempt should be:
  - Choose libraries/frameworks supporting GPUs
  - Just change some parameters/options of the framework to use GPUs
  - No changes in the code itself

• As shown in the live demo, there was no GPU-specific code in the applications, just some GPU specific parameters in the command lines
• Using GPUs in EGI Federated Cloud is easy for users
  - for basic usages, very little or no knowledge about GPU-specific development/deployment required
    ▪ Practically no differences between codes with and without GPUs if using high-level frameworks like tensorflow

  - just to select correct options (site, flavor, image, command options/parameters) according to documentations
Getting started with EGI Federated Cloud: https://docs.egi.eu/users/cloud-compute/

Using GPUs in EGI Federated Cloud: https://docs.egi.eu/users/cloud-compute/gpgpu/

Tensorflow tutorial: https://www.tensorflow.org/tutorials/quickstart/beginner

Using Tensorflow via Docker: https://www.tensorflow.org/install/docker

EGI Marketplace for ordering CPU/GPU instances: https://marketplace.egi.eu/12-compute
Thank you for your attention!

Questions?

Contact: viet.tran@savba.sk