Machine Learning System Nvidia DGX-1 and OpenStack GPU VMs

Intro

The Machine Learning System DGX-1 is a “Supercomputer in a box” with a peak performance of 170 TFlop/s (FP16). It contains eight high end GPGPUs from NVIDIA (Tesla P100) with each 16 GB RAM and 28.672 CUDA-compute units which are connected to each other by an NVLink Interconnect and a host x86 compatible system with 40 cores (Intel Xeon). Users can reserve the whole DGX-1 exclusively and run complex machine-learning tasks, which are available via Docker images.

A set of preinstalled images covers deep learning toolkits such as TensorFlow, Theano, CNTK, Torch, DIGITS, Caffe and others.

The system is running Ubuntu 18.04 LTS in the version supported by NVidia.

Below is a schematic drawing about the internals of the system. The 8 GPUs are connected via NVlink High Speed Interconnect and the x86 cores are connected via PCIe Switches to the GPUs. For a detailed documentation about the hardware please see the Nvidia website directly.

Also available are 4 single node systems with one NVIDIA GPU P100 for development purposes. On these systems only a general purpose image is available which provides a Ubuntu 18.04 LTS VM with Cuda 10 and Nvidia docker installed.

Access and Login

**LRZ Linux Cluster Users** (Here is how you can get an LRZ Linux Cluster account: [https://www.lrz.de/services/compute/application/](https://www.lrz.de/services/compute/application/). Please note: your tum /imu or lrz account is not an LRZ Linux Cluster account!!) can get access to the system by submitting an Incident ticket (subject linux cluster) to the LRZ service desk (servicedesk@lrz.de). Please briefly describe your research in your application and notice that DGX-1 is in high demand, and it is a production machine, meaning it should not be used as a debugging tool. Any software you want to run on DGX-1 should have been already tested with GPU elsewhere. Currently all of LRZ GPU systems are in testing phase, there is no backup for data. You have to backup your data by yourself.

The system can be reserved via a online calendar system ([https://datalab.srv.lrz.de](https://datalab.srv.lrz.de)) which is at the moment only available in the MWN. If you want to use the reservation system and the login to the compute system from outside the MWN you first have to connect to the LRZ VPN (see VPN documentation).

In the online calendar reservation the user can see the available timeslots and book the complete system for maximal 6 hours per slot.
Please remember that on DGX-1 only /home/ is kept between sessions. On VMs **everything** is lost. We are working on that but in the meantime always copy your data. See https://doku.lrz.de/display/PUBLIC/DSS+Understanding+DSS+on+DGX-1.

The user has then to upload a ssh key into the online calendar which will be used for authentication on the system.

When the date of the reservation approaches, the user will obtain an email with further instructions how to connect to the system via ssh and via a http link. Please be aware that the users computer has to be in the Munich Science Network or LRZ VPN in order to connect to the system.

**NVIDIA GPU Optimized Deep Learning Frameworks**

The NVIDIA Deep Learning SDK accelerates widely-used deep learning frameworks.

This release provides containerized versions of those frameworks optimized for the NVIDIA DGX-1, pre-built, tested, and ready to run, including all necessary dependencies. Together with Nvidia deep learning institute, LRZ regularly offers Deep Learning Workshops on site, please check LRZ workshop/course page.

**Using nvidia-docker on single GPU virtual servers**

As single GPU virtual servers are deleted completely after each session a quick way to install your software is helpful. The fastest way to run a machine learning application on the single GPU system is to use nvidia-docker:

**Example Tensorflow:**

Log in to virtual server using

```
ssh -L 8888:localhost:8888 ubuntu@your ip address from setup email>
```
Pull and start the container:

```
nvidia-docker run -it -p 8888:8888 gcr.io/tensorflow/tensorflow:latest-gpu
```


You can use the option "-v /ssdtemp:/ssdtemp" to map the 800 GB of storage space mounten on /ssdtemp in the server to the container.

A list of Tensorflow containers is available here: [https://hub.docker.com/r/tensorflow/tensorflow/tags/](https://hub.docker.com/r/tensorflow/tensorflow/tags/)

You can of course package your own applications using nvidia-docker, see [https://devblogs.nvidia.com/parallelforall/nvidia-docker-gpu-server-application-deployment-made-easy/](https://devblogs.nvidia.com/parallelforall/nvidia-docker-gpu-server-application-deployment-made-easy/)