NGC Note

- 1）如何申请NGC账号并介绍上面的各种资源；
- **2） 从本机联上NGC，演示pull, install, start镜像，挂载磁盘，数据读写等基本使用操作；**
- 3）进入容器，介绍pycharm 的使用，编译运行一个python小程序或进入keras 环境跑一段小代码

[Github: Nvidia Docker](#)

确保docker/nvidia-docker已经安装成功

验证nvidia-docker安装成功

```
nvidia-docker run hello-world
```

```
Experimental:          false
church@titan-v-0:~$ nvidia-docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```
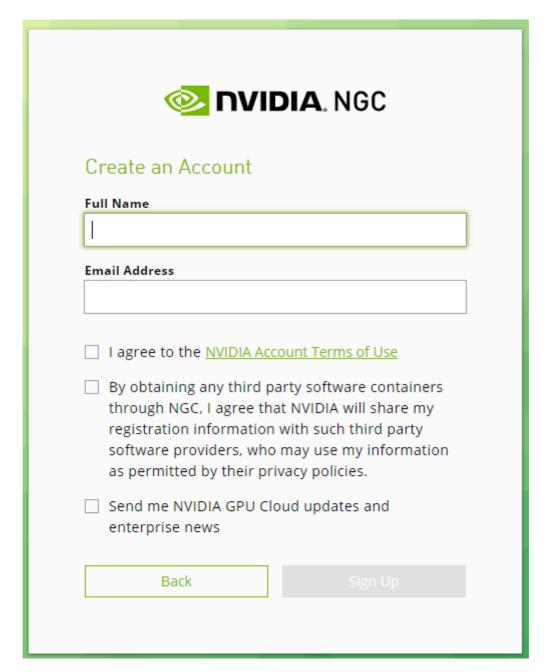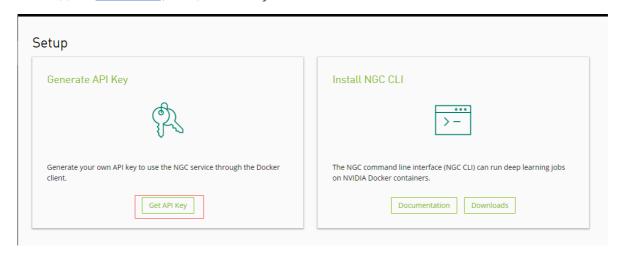

NGC支持的显卡[架构](#)，**Titan V**是Volta架构
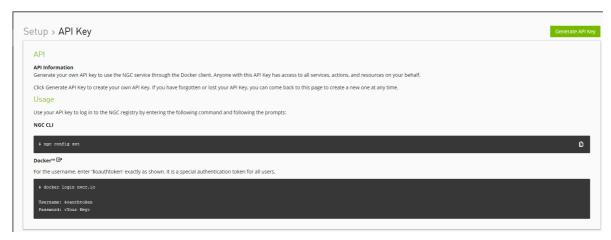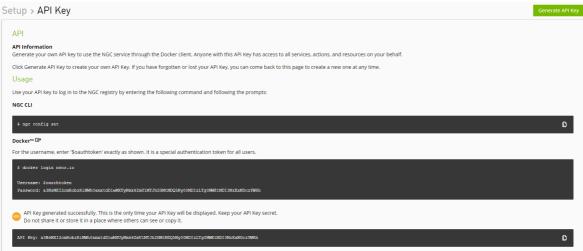
- Pascal
- Volta
- Turing

使用NGC

如果是第一次使用NGC，需要[注册账号](#)
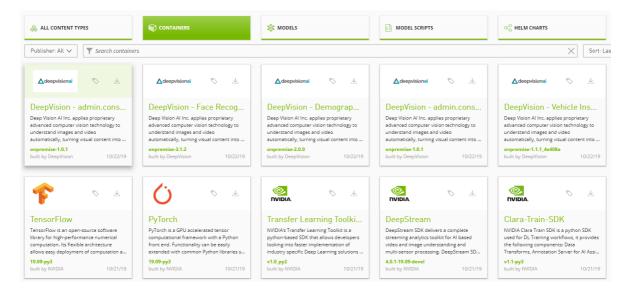
登录之后进入SETUP界面，点击Get API Key

生成 API Key 成功后，回到 GPU 云服务器命令行，运行如下命令进行登录：

```
docker login nvcr.io
Username: $oauthtoken
Password: <Your Key>
```
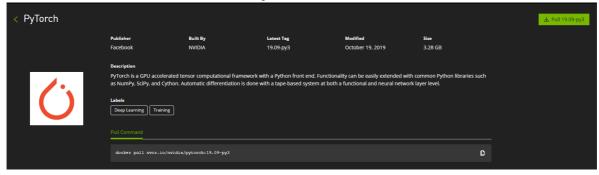
其中用户名固定为 **$oauthtoken**，密码即前面生成的 **API Key**。

登录成功

下面获取官方Docker Image, 回到容器镜像页面：NGC 提供的深度学习环境镜像

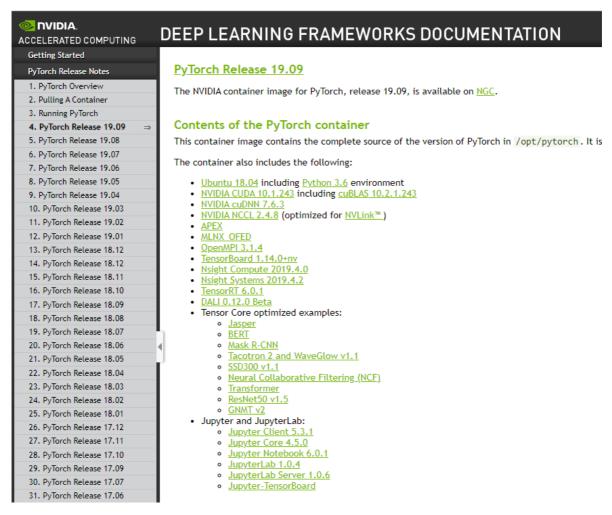选择需要拉取的镜像，进入对应的页面，以PyTorch为例

运行如下命令 pull 该 image（整个image pull下来需要很长的时间）：

```
docker pull nvcr.io/nvidia/pytorch:19.09-py3
```

点击这里可以看到对应镜像版本中包含的全部软件

PyTorch Release 19.09

The NVIDIA container image for PyTorch, release 19.09, is available on NGC.

Contents of the PyTorch container

This container image contains the complete source of the version of PyTorch in /opt/pytorch. It is

The container also includes the following:

- Ubuntu 18.04 including Python 3.6 environment
- NVIDIA CUDA 10.1.243 including cuBLAS 10.2.1.243
- NVIDIA cuDNN 7.6.3
- NVIDIA NCCL 2.4.8 (optimized for NVLink™)
- APEX
- MLNX_OFED
- OpenMPI 3.1.4
- TensorBoard 1.14.0+nv
- Nsight Compute 2019.4.0
- Nsight Systems 2019.4.2
- TensorRT 6.0.1
- DALI 0.12.0 Beta
- Tensor Core optimized examples:
    - Jasper
    - BERT
    - Mask R-CNN
    - Tacotron 2 and WaveGlow v1.1
    - SSD300 v1.1
    - Neural Collaborative Filtering (NCF)
    - Transformer
    - ResNet50 v1.5
    - GNMT v2
- Jupyter and JupyterLab:
    - Jupyter Client 5.3.1
    - Jupyter Core 4.5.0
    - Jupyter Notebook 6.0.1
    - JupyterLab 1.0.4
    - JupyterLab Server 1.0.6
    - Jupyter-TensorBoard

拉取镜像完成之后，运行如下命令查看docker中的镜文件

```
docker images
```

启动 NGC 容器

运行一下命令启动NGC容器

```
nvidia-docker run -ti -v /mnt:/mnt 9d6 /bin/bash
```

其中 9d6 为上一步 pull 下来 image 的 ID 前 3 个字母，缩写是为了输入方便

也可以使用镜像名：版本号（nvcr.io/nvidia/pytorch :19.09-py3）。

-v /mnt: /mnt 表示将本地 /mnt 目录映射到 Docker 容器 /mnt 位置，方便 Docker 与主机之间互拷文件。

启动界面如下

检查容器中 GPU 驱动是否正常：

```
nvidia-smi
```