Using GPUs with HTCondor 9.0/9.1

John (TJ) Knoeller
Center for High Throughput Computing

GPUs are "custom" resources

- GPUs in HTCondor are a type of custom resource with the tag : **GPUs**
  - MACHINE_RESOURCE_INVENTORY_GPUs
- Startd runs a program to determine the number of GPUs and their properties
  - condor_gpu_discovery
  - Defines number of GPUs and GPU ids

GPU discovery

```
$ condor_gpu_discovery -extra -by-index
DetectedGPUs="CUDA0, CUDA1, CUDA2, CUDA3"
CUDACapability=8.0
CUDAClockMhz=1410.00
CUDAComputeUnits=108
CUDACoresPerCU=64
CUDADeviceName="A100-SXM4-40GB"
CUDADriverVersion=11.20
CUDAECCEnabled=true
CUDAGlobalMemoryMb=40536
CUDAMaxSupportedVersion=11020
CUDA0DevicePciBusId="0000:01:00.0"
CUDA0DeviceUuid="887efb86-35ba-3928-8b22-8f98126311f7"
CUDA1DevicePciBusId="0000:41:00.0"
CUDA1DeviceUuid="ee7237b4-7e82-64c1-4693-db39b705ecfa"
CUDA2DevicePciBusId="0000:81:00.0"
CUDA2DeviceUuid="110a08e4-1c0c-334c-f62b-ce1bf355d691"
CUDA3DevicePciBusId="0000:C1:00.0"
CUDA3DeviceUuid="6c6a9b39-af67-1df1-2827-d30d5ef32421"
```

Startd slot attributes

- condor_gpu_discovery defines slot attributes for the GPUs resource
  - **`GPUs`** : number of items in DetectedGPUs
  - **`AssignedGPUs`** : items assigned to the slot
    - For p-slots usually same as DetectedGPUs
    - For other slots one or more GPU ids from the list
    - All other attributes from discovery become slot attributes in all slots
    - AssignedGPUs from the slot used to set CUDA_VISIBLE_DEVICES for the job

Multiple jobs sharing a GPU (the unsafe way)

- Wrapper script around discovery
  - Change the DetectedGPUs attribute to list a GPU id more than once
  - Maybe change GlobalMemoryMb also?
- New arguments to discovery in 9.0
  - -repeat : duplicate GPU ids
  - -divide : duplicate GPU ids and modify memory
  - Add to GPU_DISCOVERY_EXTRA knob

Using –divide to share GPUs

```
$ condor_gpu_discovery -by-index -extra -divide 2
DetectedGPUs="CUDA0, CUDA0"
CUDACapability=7.0
CUDAClockMhz=1380.00
CUDAComputeUnits=80
CUDACoresPerCU=64
CUDADeviceMemoryMb=16160
CUDADeviceName="Tesla V100-PCIE-16GB"
CUDADevicePciBusId="0000:3B:00.0"
CUDADeviceUuid="c4a646d7-aa14-1dd1-f1b0-57288cda864d"
CUDADriverVersion=11.20
CUDAECCEnabled=true
CUDAGlobalMemoryMb=8080
CUDAMaxSupportedVersion=11020
```

GPU ids

- Historically GPU ids are CUDA<n>
  - CUDA_VISIBLE_DEVICES = <n>
- But device indexes are not stable
- Starting with HTCondor 9.0 we use uuids
  - -short-uuid is the default
  - CUDA_VISIBLE_DEVICES = GPU-<uuid>
- Some GPU property attribute names change
  - More name changes coming in 9.x series

GPU discovery in 9.0

```
$ condor_gpu_discovery -extra
DetectedGPUs="GPU-887efb86, GPU-ee7237b4, GPU-110a08e4, GPU-6c6a9b39"
CUDACapability=8.0
CUDAClockMhz=1410.00
CUDAComputeUnits=108
CUDACoresPerCU=64
CUDADeviceName="A100-SXM4-40GB"
CUDADriverVersion=11.20
CUDAECCEnabled=true
CUDAGlobalMemoryMb=40536
CUDAMaxSupportedVersion=11020
GPU_110a08e4DevicePciBusId="0000:81:00.0"
GPU_110a08e4DeviceUuid="110a08e4-1c0c-334c-f62b-ce1bf355d691"
GPU_6c6a9b39DevicePciBusId="0000:C1:00.0"
GPU_6c6a9b39DeviceUuid="6c6a9b39-af67-1df1-2827-d30d5ef32421"
GPU_887efb86DevicePciBusId="0000:01:00.0"
GPU_887efb86DeviceUuid="887efb86-35ba-3928-8b22-8f98126311f7"
GPU_ee7237b4DevicePciBusId="0000:41:00.0"
GPU_ee7237b4DeviceUuid="ee7237b4-7e82-64c1-4693-db39b705ecfa"
```

Take a GPU offline

- HTCondor lets you take a GPU offline with a reconfig of the Startd
- Offline GPU ids are not assigned to jobs
- Requires stable GPU ids to work properly
  - Works with GPU indexes if the GPU is OK

Mixed GPU types

```
$ condor_gpu_discovery -extra
DetectedGPUs="GPU-c4a646d7, GPU-6a96bd13"
CUDACoresPerCU=64
CUDADriverVersion=11.20
CUDAMaxSupportedVersion=11020
GPU_6a96bd13Capability=7.5
GPU_6a96bd13ClockMhz=1770.00
GPU_6a96bd13ComputeUnits=72
GPU_6a96bd13DeviceName="TITAN RTX"
GPU_6a96bd13DevicePciBusId="0000:AF:00.0"
GPU_6a96bd13DeviceUuid="6a96bd13-70bc-6494-6d62-1b77a9a7f29f"
GPU_6a96bd13ECCEnabled=false
GPU_6a96bd13GlobalMemoryMb=24220
GPU_c4a646d7Capability=7.0
GPU_c4a646d7ClockMhz=1380.00
GPU_c4a646d7ComputeUnits=80
GPU_c4a646d7DeviceName="Tesla V100-PCIE-16GB"
GPU_c4a646d7DevicePciBusId="0000:3B:00.0"
GPU_c4a646d7DeviceUuid="c4a646d7-aa14-1dd1-f1b0-57288cda864d"
GPU_c4a646d7ECCEnabled=true
GPU_c4a646d7GlobalMemoryMb=16160
```

Multiple jobs sharing a GPU (the safer way)

- Some NVIDIA GPUs have MIG capability
  - Split a GPU device into up to 7 MIGs
  - Each MIG behaves like a smaller GPU device
  - The GPU can no longer be used directly
- condor_gpu_discovery discovers the MIGs
  - (And hides the MIG parent GPU)

Discovering MIGs

```
$ condor_gpu_discovery -extra
DetectedGPUs="MIG-GPU-20e62ffc-155e-49ad-04e1-ffacfc109ce3/1/0, MIG-GPU-20e62ffc-155e-49ad-04e1-ffacfc109ce3/2/0, GPU-124d06a7, GPU-c6cfdc9c, GPU-
      8f9c2d75"
CUDADriverVersion=11.20
CUDAMaxSupportedVersion=11020
GPU_124d06a7Capability=8.0
GPU_124d06a7ClockMhz=1410.00
GPU_124d06a7CoresPerCU=64
GPU_124d06a7DeviceName="A100-SXM4-40GB"
GPU_124d06a7DevicePciBusId="0000:01:00.0"
GPU_124d06a7DeviceUuid="GPU-124d06a7-6642-3962-9afa-c86c31b9a7e6"
GPU_124d06a7ECCEnabled=true
GPU_124d06a7GlobalMemoryMb=40536
GPU_8f9c2d75Capability=8.0
GPU_8f9c2d75ClockMhz=1410.00
GPU_8f9c2d75CoresPerCU=64
GPU_8f9c2d75DeviceName="A100-SXM4-40GB"
GPU_8f9c2d75DevicePciBusId="0000:81:00.0"
GPU_8f9c2d75DeviceUuid="GPU-8f9c2d75-1b7d-5026-b177-283d13ddbb90"
GPU_8f9c2d75ECCEnabled=true
GPU_8f9c2d75GlobalMemoryMb=33192
GPU_c6cfdc9cCapability=8.0
GPU_c6cfdc9cClockMhz=1410.00
GPU_c6cfdc9cCoresPerCU=64
GPU_c6cfdc9cDeviceName="A100-SXM4-40GB"
GPU_c6cfdc9cDevicePciBusId="0000:41:00.0"
GPU_c6cfdc9cDeviceUuid="GPU-c6cfdc9c-63df-c5b3-3605-7bc4b2545434"
GPU_c6cfdc9cECCEnabled=true
GPU_c6cfdc9cGlobalMemoryMb=33192
MIG_GPU_20e62ffc_155e_49ad_04e1_ffacfc109ce3_1_0ComputeUnits=42
MIG_GPU_20e62ffc_155e_49ad_04e1_ffacfc109ce3_1_0DeviceUuid="MIG-GPU-20e62ffc-155e-49ad-04e1-ffacfc109ce3/1/0"
MIG_GPU_20e62ffc_155e_49ad_04e1_ffacfc109ce3_1_0GlobalMemoryMb=20087
MIG_GPU_20e62ffc_155e_49ad_04e1_ffacfc109ce3_2_0ComputeUnits=42
MIG_GPU_20e62ffc_155e_49ad_04e1_ffacfc109ce3_2_0DeviceUuid="MIG-GPU-20e62ffc-155e-49ad-04e1-ffacfc109ce3/2/0"
MIG_GPU_20e62ffc_155e_49ad_04e1_ffacfc109ce3_2_0GlobalMemoryMb=20090
```

MIGs

- Lots of caveats from NVIDIA
  - Big differences between driver 460 and 470
  - Only one MIG can be used per-process
  - CUDA device enumeration does not see them
  - GPU forgets MIGs on reboot
- MIGs "just work" in HTCondor 9.0
  - Must be setup before the Startd runs discovery
    - Restart required to re-run discovery
  - Avoid heterogenous MIG configurations

More change to come

- The NVIDIA driver architecture around MIGs is changing rapidly
- HTCondor will do our best to keep up
- Active work on
  - Chasing MIG changes from NVIDIA
  - More control over which GPUs are assigned to which slot(s)
  - Advertising only properties of assigned GPUs

Any Questions?