HTCondor on Google Cloud
Introducing Cloud HPC Toolkit

25 May 2022
HTCondor Week 2022
Cloud HPC Toolkit Objective

“Make it easy for customers and partners to deploy repeatable turnkey HPC environments following Google Cloud’s HPC best practices”
## Forward, HTCondor

<table>
<thead>
<tr>
<th>Deprecated Solution</th>
<th>Cloud HPC Toolkit</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Based upon Deployment Manager, a Google Cloud specific technology</td>
<td>• Based on standard Open Source infrastructure-as-code and configuration-as-code tools</td>
</tr>
<tr>
<td>• Fine for a single user or small group</td>
<td>• Designed to implement best practices by default while enabling customization</td>
</tr>
<tr>
<td>• Not easily extended to support custom HTCondor configurations</td>
<td>• Can support dedicated and hybrid pools</td>
</tr>
<tr>
<td>• No support for hybrid HTCondor pools</td>
<td>• More sustainable approach</td>
</tr>
<tr>
<td>• Maintenance burden for CHTC</td>
<td></td>
</tr>
</tbody>
</table>
Progress

Current
Scale a homogenous pool based on length of job queue

9.2: Cloud Storage Support
Jobs can transfer data to or from Cloud Storage

9.4: Cloud-Native Machine ClassAds
Machines advertise cloud attributes such as region, preemptibility, and unique identifiers for error resolution

Soon: Cloud-Native Job ClassAds
Jobs using Cloud Storage can easily advertise cost and performance matchmaking attributes

Roadmap: Cloud-native autoscaling
Extend existing autoscaler to support heterogeneous pools by matching "offline" Machine ClassAds
HTCondor 9.2+ supports Cloud Storage

- **Not a shared POSIX filesystem!**
- Accessible by gs:// URLs
  - (Really HTTPS)
- Performance and cost benefits
  - High bandwidth with no bottleneck at access point
- Data resilience/archival guarantees
- **HTCondor 9.0 support exists with custom URLs**
Choosing a location type

Regional
Your data is stored in a specific region with replication across availability zones in that region. *Good for colocating compute and storage for high performance.*

Multi-Region
Your data is distributed redundantly across US, EU, or Asia. Good for serving content to end users and when you want automatic failover.

Dual-region
Your data is replicated across a specific pair of regions. *Good for when you need colocated compute and storage and automatic failover.*

Multi-region typically not recommended for high throughput workloads!
Canonical Machine ClassAd attributes for cloud providers

Enabled with a single metaknob in versions 9.4 and above (script easily backported to 9.0)

Collaborative effort between Google Cloud and CHTC staff, particularly Todd Miller

CloudImage="htcondor-v905-20210825t193910z"
CloudMachineType="c2-standard-4"
CloudZone="us-west1-a"
CloudRegion="us-west1"
CloudInstanceID="1893620332054126642"
CloudProvider="Google"
CloudPlatform="GCE"
CloudInterruptible=True

Todd Miller has implemented functionality for AWS using the same attribute names but EC2 values
Soon: Automated Job ClassAd attributes for cloud data

This "prototype" example adds the location of a dual region bucket to a Job ClassAd. Can easily be extended to automate rank/requirements or use JOB_TRANSFORM at administrator level.

A case for user-executed job transform?
Coming together

An HTCondor pool that autoscales **globally** in the direction of the data
Global network infrastructure

The largest cloud network: 100,000s of miles of fiber optic cable, 8 subsea cables
More edge and peering points than any public cloud
Roadmap: Using Offline Machines to Drive Autoscaling

- One-to-one relationship between offline ClassAds and external agent responsible for scaling external resources
- HTCondor-native and Cloud-native
- Backend can be replaced in default or custom implementations
- Supports admin-focused usage or "bring-your-own-pool" by the user

```
Offline=True
OfflineCpusMatchedThisCycle=A
OfflineDiskMatchedThisCycle=B
OfflineGpusMatchedThisCycle=C
OfflineMemoryMatchedThisCycle=D
OnlineMachineCount=M
OfflineScalerId=projects/x/regions/us-central1/instanceGroups/htc-type1-mig
```
HPC Toolkit Principles

Open
All source code on GitHub, subject to review, available for Pull Requests and a community enabled by GitHub Discussions and Issues

Configurable
HPC Toolkit runners customize VMs using a combination of shell scripts and Ansible playbooks

Scalable
Easily building custom images using Packer and Toolkit Runners is a core feature, enabling rapid provisioning of new VMs with custom application

Reliable
Infrastructure-as-code is code. Each blueprint is integration tested regularly to ensure that it behaves as designed

Starting point for all Google Cloud support for HPC and schedulers!
HPC Toolkit Architecture

**HPC Environment Configuration**
- **HPC Blueprint**
  - Format: YAML
  - Source: HPC Toolkit User
  - Customized HPC high-level HPC environment definition
- **HPC Blueprint Examples**
  - Format: YAML
  - Source: Google
  - Example HPC environment configurations to start from

**Creating an HPC Deployment**
- **ghpc Engine**
  - Format: Binary
  - Source: Google
- **HPC Modules**
  - Format: Terraform, Packer, Scripts, etc
  - Source: Partners, PSO, Advanced Users
  - Modular HPC cluster building blocks, imported from GitHub or local disk.

**Output**
- **HPC Deployment Folder**
  - Format: Terraform, Packer, Scripts, etc
  - Source: ghpc, Advanced Users
  - Repeatable, well-defined HPC deployments encapsulated in scripts, ready to be deployed directly by admins, shared to end-users, or saved

**GCP Environment**
- **Deployed HPC environment on Google Cloud**
- **Monitoring and Analytics (Optional)**
  - HPC Environment Configuration
  - Creating an HPC Deployment
  - Output
  - GCP Environment
  - HPC Blueprint
  - HPC Blueprint Examples
  - ghpc Engine
  - HPC Modules
  - HPC Deployment Folder
  - Deployed HPC environment on Google Cloud
  - Monitoring and Analytics (Optional)
Terraform DAG model of infrastructure

Declarative syntax automatically synchronizes state of cloud resources with local modules. “Puppet” or “Ansible” but for virtual hardware!
blueprint_name: mycluster
vars:
  project_id: ## Set GCP Project ID Here ##
  deployment_name: hpc-slurm-small
  region: europe-west4
  zone: europe-west4-a
deployment_groups:
  - group: primary
    modules:
      - source: modules/network/vpc
        kind: terraform
        id: network1
      - source: resources/file-system/filestore
        kind: terraform
        id: homefs
        use: [network1]
        settings:
          local_mount: /home
      - source: community/modules/compute/SchedMD-slurm-on-gcp-partition
        kind: terraform
        id: compute_partition
        use: [network1, homefs]
        settings:
          partition_name: compute
          max_node_count: 1024
      - source: community/modules/scheduler/SchedMD-slurm-on-gcp-controller
        kind: terraform
        id: slurm_controller
        use: [network1, homefs, compute_partition]
        settings:
          login_node_count: 1
      - source: community/modules/scheduler/SchedMD-slurm-on-gcp-login-node
        kind: terraform
        id: slurm_login
        use: [network1, homefs, slurm_controller]

ghpc create mycluster.yaml
terraform -chdir=hpc-slurm-small/primary init
terraform -chdir=hpc-slurm-small/primary apply
HTCondor in Cloud HPC Toolkit

- Public announcement and documentation: May 30
  - https://cloud.google.com/blog/topics/hpc
- Code: https://github.com/GoogleCloudPlatform/hpc-toolkit
- HTCondor branch will be merged shortly after release
  - Blueprint for automatic provisioning
  - Support for autoscaling homogenous pool
- Features at release
  - IDTOKENS security
  - 9.X series cloud-native features
NSF 22-087 CISE awardees may apply for cloud funds with rapid onboarding into Google Cloud

All Cloud-Bank eligible solicitations

Data egress waiver for Internet2 / GÉANT members
Thank you.

https://cloud.google.com/hpc