Azure HPC OnDemand Platform: Cloud HPC made easy.

By Xavier Pillons

Published Jul 12 2021 07:01 AM 6,080 Views

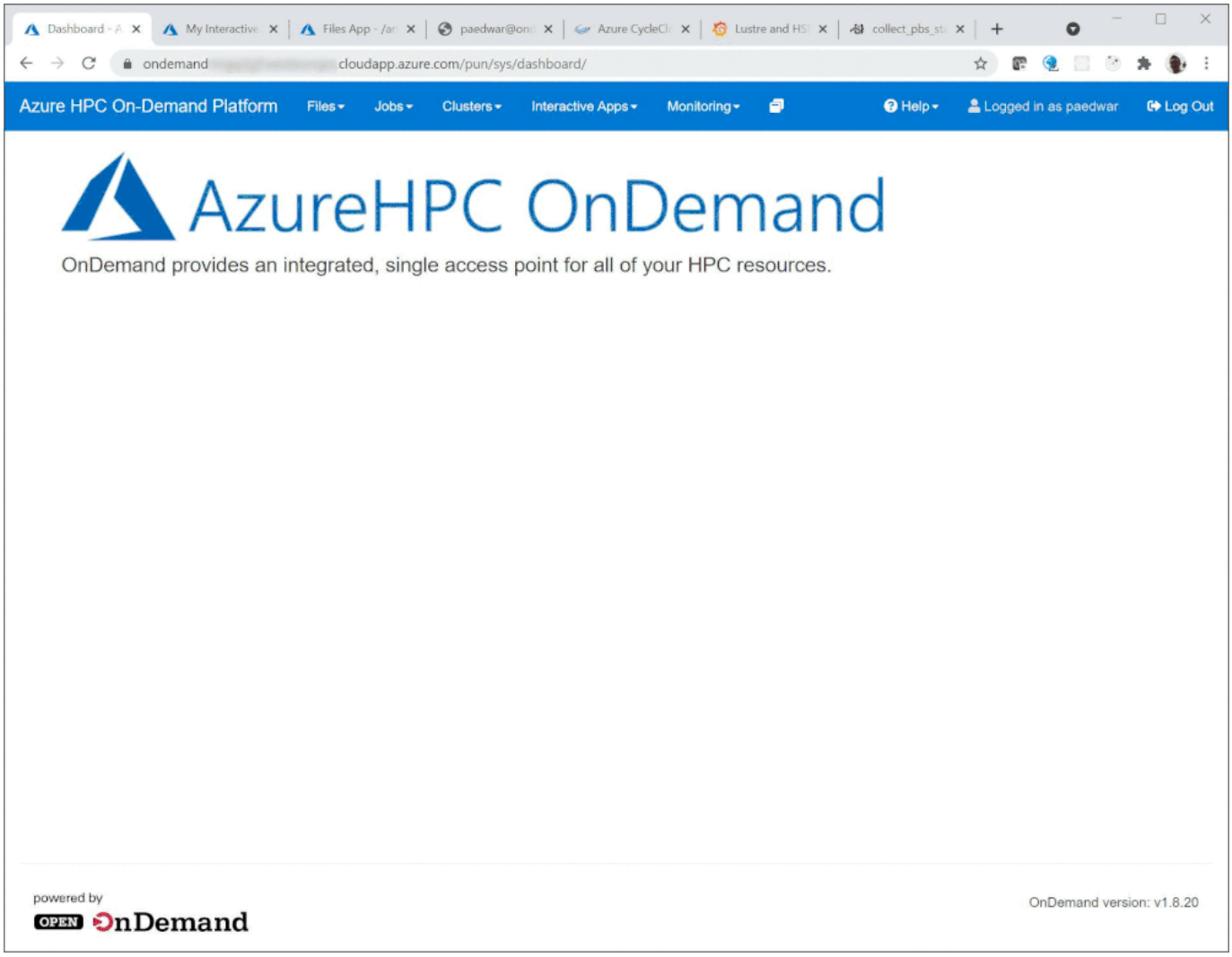
As many customers are looking at running their HPC workloads in the cloud, onboarding effort and cost are key consideration. As an HPC administrator, in such process you try to provide a unified user experience with a minimal disruption, in which the end users and the cluster administrators can retrieve most of their on-premises environment while leveraging the power of running in the cloud.

The ***Specialized Workloads for Industry and Mission*** team that works on some of the most complex HPC customer and partner scenarios has built a solution accelerator **Azure HPC OnDemand Platform** (aka **az-hop**) available in the Azure/az-hop public GitHub repository to help our HPC customers onboard faster. **az-hop** delivers a complete HPC cluster solution ready for users to run applications, which is easy to deploy and manage for HPC administrators. **az-hop** leverages the various Azure building blocks and can be used as-is, or easily customized and extended to meet any uncovered requirements.

Based on our experience, from years of customer engagements, we have identified some common principles that are important to our customers and designed **az-hop** with these in mind:

- A pre-packaged HPC Cluster easy to deploy in an existing subscription, which contains all the key building blocks and best practices to run a production HPC environment in Azure,

- A unified and secured access for end users and administrators, so each one can reuse their on-premises tools and scripts,
- A solution to integrate applications under the same unified cloud experience,
- Build on standards, common tools and open blocks so it can be easily extended and customized to accommodate the unique requirements of each customer.



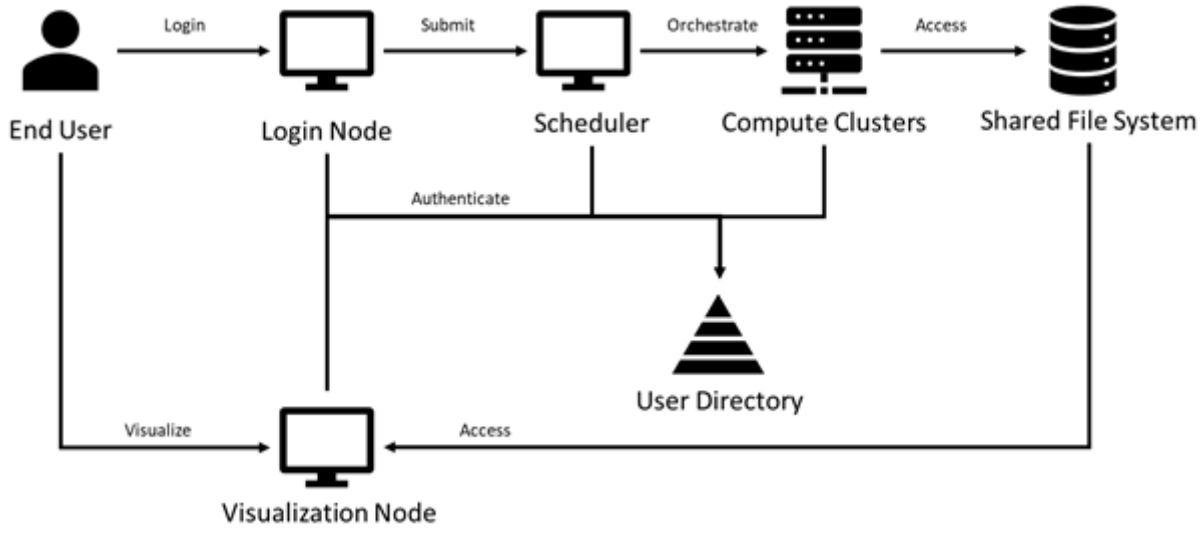
The HPC end-user workflow typically comprises of 3 steps –

Step	Details	Key Features needed
Prepare Model	In this step, the user would get the data to be used by the application.	Fast data transfer and a home directory where they can upload their data, scripts etc.

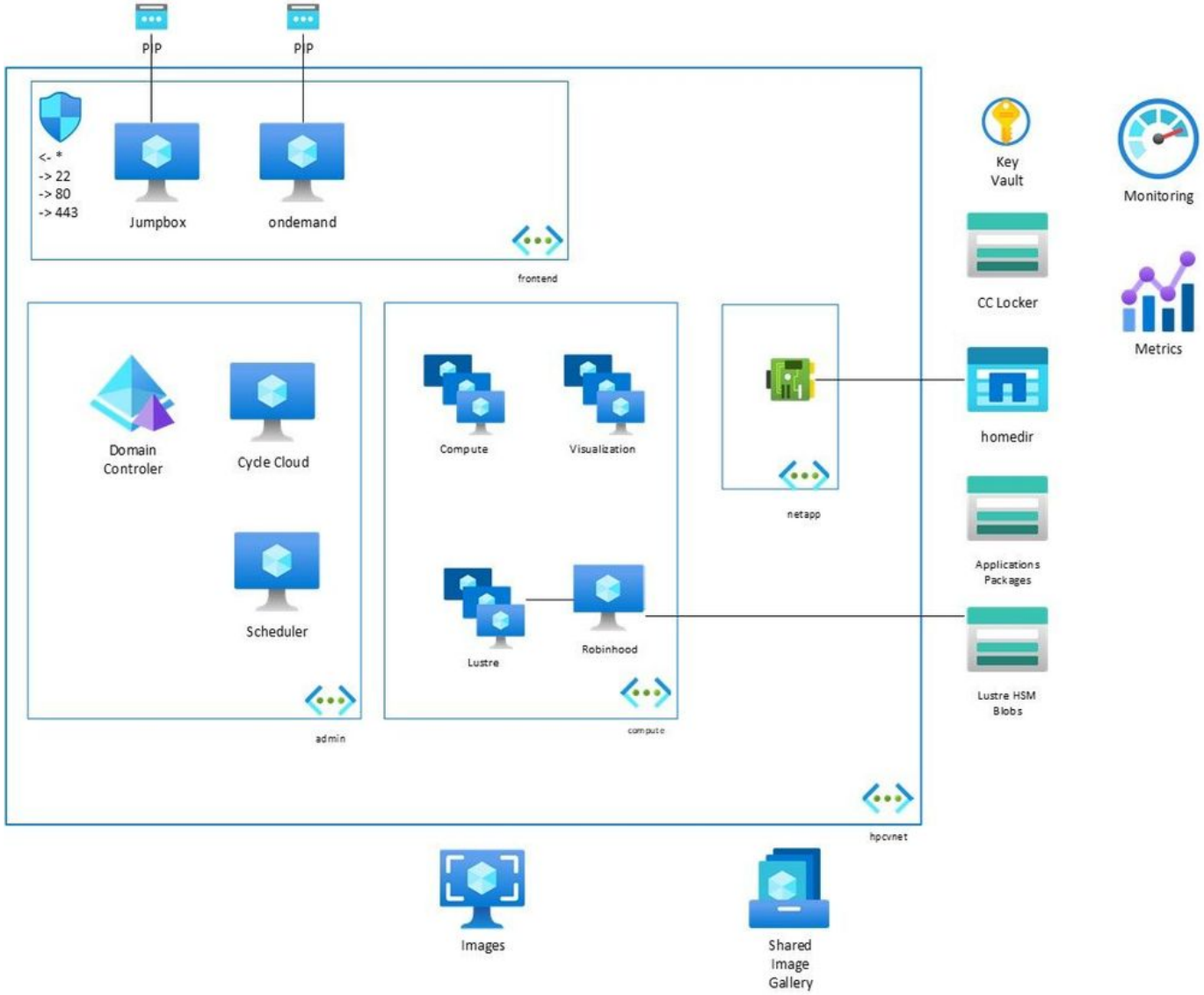
Skip to main content

Run Job	Using their shell session or UI user would submit their job providing details on the slot type and number of nodes they would need for running the job.	Auto-scale compute, scheduler, scratch storage.
Analyze results	Once the job is finished, the user can visualize their results.	Interactive desktop

The below diagram depicts the components needed in a typical on-premise environment to support this workflow.



The default **az-hop** environment supports the above workflow with the following architecture, all accessed from the OnDemand portal for unified access and only with HTTPS for end users and SSH/HTTPS for administrators.



The unified experience is provided by the Open OnDemand web portal from the Ohio Supercomputer Center. Listed below are some of the features that the current **az-hop** environment supports but you can see the releases as we add more features:

- Authentication is managed by Active Directory,
- Job submission in CLI or web UI thru OpenPBS,
- Dynamic resources provisioning and autoscaling is done by Azure Cycle Cloud, pre-configured job queues and integrated health-checks to quickly avoid non-optimal nodes
- A common shared file system for home directory and applications is delivered by Azure Netapp Files,
- A Lustre parallel filesystem using local NVME for high performance that automatically archives to Azure Blob Storage using the Robinhood Policy Engine and Azure Storage data mover,
- Monitoring dashboards are exposed in Grafana,
- Remote Visualization with noVNC and GPU acceleration with VirtualGL.

Skip to main navigation

The whole solution is defined in a single configuration file and deployed with Terraform. Ansible playbooks are used to apply the configuration settings and application packages installation. Packer is used to build the two main custom images for compute nodes and for remote visualization, published into an Azure Shared Image gallery.

The instructions to deploy your **az-hop** environment are available from this page. The **az-hop** GitHub comes with some example tutorials to demonstrate how you can integrate and run your applications in the **az-hop** environment and you can follow them here to give it a test drive or just simply run your own.

5 Likes

You must be a registered user to add a comment. If you've already registered, sign in. Otherwise, register and sign in.

Comment

Co-Authors

- **xpillons**
- **kanchanm**
- **pauledwards**
- **Hugo_Meiland**

Version history

Last update: Jan 19 2022 08:26 AM

Updated by: kanchanm

Labels

HPC 46

Share