

On-Demand Data Center

Quick Start Guide 6.0.0

August 2021



 **Adaptive**
COMPUTING
On-Demand Data Center™

Contents

Legal Notices	4
Chapter 1: On-Demand Data Center Platform Installation	5
1.1 Prerequisites	5
1.1.1 Operating System	6
1.1.2 Supported Browsers	6
1.1.3 Linux Packages	6
1.1.4 Traffic	7
1.1.5 Obtaining Credentials from the Cloud Service Providers	7
1.1.6 Stacks	10
1.1.7 Site Availability	10
1.2 Installation Steps	11
1.3 Verifying the Installation	13
1.4 Adding the Server License Key	13
Chapter 2: On-Demand Data Center CLI Procedures	15
2.1 Stacks	15
2.2 Users	30
2.3 Architecture	30
Chapter 3: Using the On-Demand Data Center GUI	31
3.1 Creating an On-Demand Data Center Account	31
3.2 User Profile	32
3.2.1 Settings - Admins	32
3.2.2 Settings - Users	33
3.3 Credentials Manager	34
3.4 Stack Manager	35
3.4.1 Viewing Stacks	36
3.4.2 Creating a Stack	36
3.4.3 Building a Stack	37
3.5 Cluster Manager	37
3.5.1 Viewing Clusters	37
3.5.2 Creating a Cluster	39
3.5.3 Deploying a Cluster	40
3.5.4 Deleting a Cluster	40
3.6 Job Manager	40
3.6.1 Adding a New Job	41
3.6.2 Adding a Temporary On Demand Job	43

3.7 File Manager	47
3.8 Accounting	47
3.9 Bursting	49
3.9.1 Bursting Configurations	49
3.9.2 Bursting Service	50

Legal Notices

Copyright © 2018, 2021. Adaptive Computing Enterprises, Inc. All rights reserved.

This documentation and related software are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

This documentation and related software may provide access to or information about content, products, and services from third-parties. Adaptive Computing is not responsible for and expressly disclaims all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Adaptive Computing. Adaptive Computing will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Adaptive Computing.

Adaptive Computing, Moab®, Moab HPC Suite, Moab Viewpoint, Moab Grid, NODUS Cloud OS™, On-Demand Data Center™, and other Adaptive Computing products are either registered trademarks or trademarks of Adaptive Computing Enterprises, Inc. The Adaptive Computing logo is a trademark of Adaptive Computing Enterprises, Inc. All other company and product names may be trademarks of their respective companies.

The information contained herein is subject to change without notice and is not warranted to be error free. If you find any errors, please report them to us in writing.

Adaptive Computing Enterprises, Inc.
1100 5th Avenue South, Suite #201
Naples, FL 34102
+1 (239) 330-6093
www.adaptivecomputing.com

Chapter 1: On-Demand Data Center Platform Installation

This chapter provides information about the installer, which contains the entire ODDC platform, along with dependencies to install it.

In this chapter:

1.1	Prerequisites	5
1.1.1	Operating System	6
1.1.2	Supported Browsers	6
1.1.3	Linux Packages	6
1.1.4	Traffic	7
1.1.5	Obtaining Credentials from the Cloud Service Providers	7
1.1.6	Stacks	10
1.1.7	Site Availability	10
1.2	Installation Steps	11
1.3	Verifying the Installation	13
1.4	Adding the Server License Key	13

1.1 Prerequisites

This section contains the following information:

- [Operating System](#)
- [Supported Browsers](#)
- [Linux Packages](#)
- [Traffic](#)
- [Obtaining Credentials from the Cloud Service Providers](#)
- [Stacks](#)
- [Site Availability](#)

1.1.1 Operating System

The ODDC platform can be installed on many different Linux distributions and has been tested on these operating systems:

- CentOS 7
- Oracle 7.7
- Red Hat 7

Note: Refer to the chosen operating system's product specifications for minimum requirements.

1.1.2 Supported Browsers

The ODDC UI supports these web browsers:

- Google Chromium (latest version)
- Google Chrome (latest version)
- Mozilla Firefox (latest version)
- Microsoft Edge (two latest major versions)
- Apple Safari (two latest major versions)

1.1.3 Linux Packages

The following Linux packages must be installed on the ODDC platform server prior to the software installation:

- mongodb-server [version 4.4]
- nginx [version 1.16]

Caution: Using a version other than the ones specified is unsupported and may result in an unstable environment.

(optional) If you want to mount/unmount a remote file system over SSH, you can use the command **nodus cluster:sshfs**.

Environment Modules can be incorporated into clusters to help manage your Linux shell environment. Environment Modules enable your environment to be modified by dynamically adding or removing settings.

1.1.4 Traffic

To enable access to the ODDC platform server, the following ports need to be open with external access:

- 22 (SSH) - user direct access / Moab job routing
- 80 (HTTP) - connection to the ODDC UI
- 443 (HTTPS) - optional, if you secure your HTTP connection
- 12001 (Cluster API) - RESTful API that communicates with the clusters
- 12345 (NODUS-web-API) - the ODDC platform API server that communicates with your UI

Note: The ODDC platform server installation will enable these ports on the server. Ports 80 and 443 are used for the UI, but can be changed to reflect your requirements.

InfiniBand networks are configured when identified in the server instance. InfiniBand is a scalable networking communications standard used in HPC and provides high data throughput with low latency.

1.1.5 Obtaining Credentials from the Cloud Service Providers

These are the supported cloud service providers:

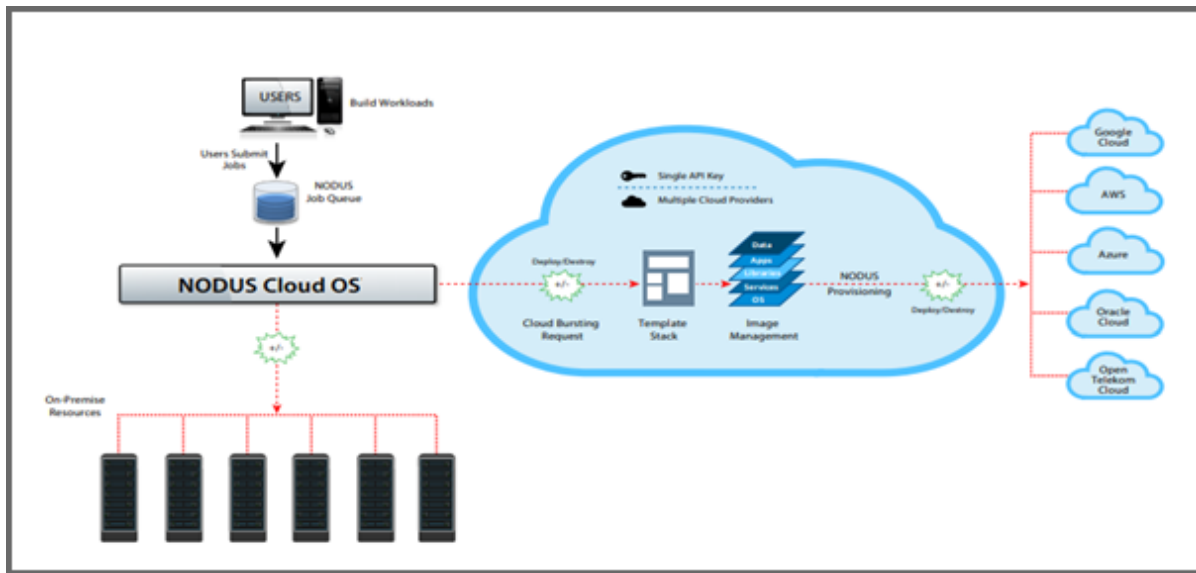
-  Amazon Web Services (AWS)
-  Google Cloud
-  Microsoft Azure
-  Oracle Cloud
-  Open Telekom Cloud (OTC)

Note: For unsupported cloud service providers, please contact us at nodus@adaptivecomputing.com to inquire about adding the provider into the ODDC.

The ODDC requires Programmatic access to deploy nodes and cluster networks on your cloud provider's resources. The steps below will help you obtain the proper cloud credentials.

Refer to the providers' support pages for detailed information. **Note:** If you need a private IP address created within a provider or you cannot get provider credentials / correct admin level credentials, please contact us at nodus@adaptivecomputing.com.

On-Demand Data Center Overview



Continue with the steps for your chosen provider below to obtain credentials:

- [Google Cloud](#)
- [AWS](#)
- [Microsoft Azure](#)
- [Oracle Cloud](#)
- [Open Telekom Cloud](#)

Google Cloud

1. Create a service account.
2. From the **Service account** list, select **New service account**.
3. In the **Service account name** field, enter a name.
4. From the **Role** list, select **Project / Owner**.
5. Click **Create**. A JSON file that contains your key downloads to your computer.

See <https://cloud.google.com/docs/authentication/production> for additional information.

AWS

AWS has two forms of credentials, both require admin privileges to set up:

- Programmatic access (access key, secret key) is a floating set of coupled values.
- EC2 roles only exist within AWS and are attached to AWS instances.

These accounts must have valid permissions associated in order to use the ODDC. The ODDC can deploy clusters using either programmatic access or EC2 roles. However, stack builds can only be done with Programmatic access.

The ODDC requires Programmatic access to access the provider's resources. For example, if the ODDC is outside of the AWS environment, access keys are required. If the ODDC is running inside AWS, the best practice is to use IAM roles instead. An IAM role is a defined set of permissions. It is not associated with a specific user or group and any trusted entity can assume the role to perform a specific business task. The ODDC supports both access methods.

When creating a cluster via **nodus cluster:aws:create**, the flag `--role` can be used to set the cluster to use EC2 role instead of Programmatic access. If the cluster entry already exists, then **nodus cluster:aws:update** can be used with flag `--role` or `--no-role` to switch on EC2 role use or turn it off, respectfully.

1. Sign in to the AWS Management Console and open the **IAM** console.
2. In the navigation pane, select **Users**.
3. Select the name of the user whose access keys you want to create, and then select the **Security credentials** tab.
4. In the **Access keys** section, select **Create access key**.
5. To view the new access key pair, select **Show**. You will not have access to the secret access key again after this dialog box closes.
6. To download the key pair, select **Download .csv file**. Store the keys in a secure location. You will not have access to the secret access key again after this dialog box closes.
7. After you download the `.csv` file, select **Close**. When you create an access key, the key pair is active by default, and you can use the pair right away.

See <https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html> for additional information.

Microsoft Azure

1. Run the script **account-scripts_azure.sh**.

See <https://docs.microsoft.com/en-us/azure/automation/shared-resources/credentials> for additional information.

Oracle Cloud

1. On the Oracle account to be used, create a compartment.
2. Within that compartment, create a VPC (virtual private cloud) network.
3. Within that VPC, create at least one subnet.

See <https://docs.cloud.oracle.com/iaas/Content/home.htm> for additional information.

Open Telekom Cloud

1. On the Open Telekom Cloud (OTC) account to be used, create a VPC.
2. Within that VPC, create a subnet.

See <https://open-telekom-cloud.com/en/products-services/virtual-private-cloud> for additional information.

1.1.6 Stacks

The Cloud Security Administrator must be part of the install/implementation team in order to gain proper authorization to spin up instances/images in the Cloud.

To launch instances in the Cloud, a stack must first be created using credentials with appropriate cloud permissions and policies.

The stack consists of the base OS (CentOS 7), services, libraries, applications, and any data that is needed so that once the instance starts up, it is ready to begin processing jobs. The ODDC allows customization options to conform to cloud configurations (proxy, vpc, subnet, etc.).

Two images are generated from the stack build process: a base OS image and a prebuilt/configured image snapshot. You can use a stock image from the marketplace or account to build a private image. Cluster software used is Torque, Ansible, Maui, and Serf.

See the CLI Commands section [Stacks](#) for additional information.

1.1.7 Site Availability

Before building a stack or deploying a cluster, make sure that the site hashicorp.com (the supplier of Terraform) is not blocked by a firewall as this will prevent the Terraform process from completing.

1.2 Installation Steps

1. In a browser, access the server via the IP or hostname (e.g., `http://x.x.x.x/`) to verify that `nginx` is started and the default port is available. If the site is unreachable, then the firewall may be enabled or the `nginx` server is not running.
 - A. Connect to the server via a command line and run **`sudo systemctl start nginx`**.
 - B. Access `http://x.x.x.x/` once again:
 - If you get a `Welcome to CentOS` message, the default port number is available and you can proceed with the installation. Or if the site is still unreachable, then the firewall may be enabled and the ODDC install cannot proceed.
 - If any other data is displayed on the browser, the default port may be in use and your administrator should assign you another port number that will be used in step 6.
2. Log in to the installation user for the operating system where the ODDC is to be installed to get a command line shell.
3. Untar the On-Demand Data Center distribution file with the command **`tar xf Nodus_Cloud_OS-<version>.tgz`**. This will produce a directory with the installation files needed to install the ODDC.
 - A. Look at the files via **`ls`**. You should see a file called `nodus-<version>`.
 - B. Change to the installation directory with the command **`cd nodus-<version>`**.
4. Because a license key is needed to install the ODDC, save the license key in a file on the server. You will also need to provide a username that will be used to access the web server files. For security purposes, never use `root` as the username.
5. Install the ODDC with the command **`./install-nodus.sh <username> <license file>`**.
6. (optional) If there is already a process using port 80 (from step 1), change the `nginx` port number with the command **`sudo <edit> /etc/nginx/nginx.conf`**.

A. Replace these lines:

```
listen 80 default_server;
listen [::] :80 default_server;
```

With these lines:

```
listen <new port> default_server;
listen [::] :<new port> default_server;
```

7. Restart the nginx server with the command **sudo systemctl restart nginx**.
8. The ODDC services use the file `/NODUS/.nodusrc` to modify the functionality of the ODDC. This file contains the following parameters:

`MONGO_HOST=localhost`

This parameter defines the Mongo DB server that is hosting the ODDC database. The default value is `localhost`.

`MONGO_PORT=27017`

This parameter defines the port that the Mongo DB server is listening on. The default value is `27017`.

`MONGO_DATABASE=nodus`

This parameter defines the database in the Mongo DB and saves the information. The default value is `nodus`.

`MONGO_USER=`

This REQUIRED parameter defines the user in the Mongo DB that is authorized to access the ODDC information. This value is configured at installation.

`MONGO_PASSWORD=`

This REQUIRED parameter defines the password for the `MONGO_USER` in the Mongo DB that is authorized to access the ODDC information. This value is configured at installation.

`LICENSE_PATH=/NODUS/.license`

This parameter defines the location of the license key that is needed to use the ODDC. The default value is `/NODUS/.license`.

`BURST_INTERVAL_MS=60000`

This parameter defines the number of milliseconds for the ODDC to check all the clusters that are set to `Bursting`. The default value is `60000` (1 minute).

`PARALLEL_TRANSACTIONS=10`

This parameter defines the maximum parallel transaction for deploying/destroying nodes. The default value is `10`.

`EMAIL_HOST=`

This parameter defines the host name of your email server.

`EMAIL_PORT=`

This parameter defines the sending port for email.

`EMAIL_USER=`

This parameter defines a valid email address on the EMAIL_HOST.

EMAIL_PASS=

This parameter defines the password for the EMAIL_USER.

Note: If a parameter does not exist in the file, the default value is used.

1.3 Verifying the Installation

1. Check the UI configuration files by editing `/NODUS/nodus-web-ui/build/assets/config.json`. This file tells the UI the address and port of the API, and which providers will be used. You can edit the providers to only show which ones you will supply licenses to. The file looks similar to this:

```
{
  "server": "http://localhost:12345",
  "providers": ["aws", "gcp", "azure", "oracle", "otc"],
  "register": true,
  "idleTimer": false
}
```

Note: You can change the idleTimer default value of false, by adding a number (in seconds).

2. Verify the API in a browser by entering **`http://<server_address>:12345/`**. The browser should display `{"reason": "unknown", "message": "Not Found"}`. If not, rerun step 1 and/or step 2.
3. Verify the UI in a browser by entering **`http://<server_address>/`**. You should be prompted to enter a license or to log in. If not, rerun step 1 and/or step 2.

1.4 Adding the Server License Key

Follow one of the procedures below to add a license key:

- [Adding the License Key via the On-Demand Data Center GUI](#)
- [Adding the License Key from a Terminal](#)

Adding the License Key via the On-Demand Data Center GUI

1. Go to `http://<web_server_address>` and enter the license key that was provided. **Note:** `web_server_address` refers to the server where the ODDC platform is installed.

Adding the License Key from a Terminal

1. Log in to the `<user_name>@<web_server_address>` that you installed the ODDC as in the section [Installation Steps](#).
2. Add the license key that was provided to `/NODUS/.license`. **Note:** If the license type is Multi-User, then add the user license to `~/ .nodus/.userKey`.
3. Verify that the license is working by running command **nodus license**.

Chapter 2: On-Demand Data Center CLI Procedures

In this chapter:

2.1 Stacks	15
2.2 Users	30
2.3 Architecture	30

2.1 Stacks

This section contains the following commands:

- `stack:aws:create NAME TYPE`
- `stack:aws:update STACK`
- `stack:azure:create NAME TYPE`
- `stack:azure:update STACK`
- `stack:build STACK`
- `stack:gcp:create NAME TYPE`
- `stack:gcp:update STACK`
- `stack:list`
- `stack:log CLUSTER TYPE`
- `stack:oracle:create NAME TYPE`
- `stack:oracle:update STACK`
- `stack:otc:create NAME TYPE`
- `stack:otc:create STACK`
- `stack:remove STACK`
- `stack:set STACK CREDENTIAL`
- `stack:share STACK USERNAME`

- [stack:show STACK](#)
- [stack:unshare STACK](#)

stack:aws:create NAME TYPE

Description

Create new Amazon Web Services (AWS) stack configuration entry.

Usage

```
$ nodus stack:aws:create NAME TYPE [CREDENTIAL] [OPTIONS]
```

Arguments

NAME	Associated Name
TYPE	(centos-7) Image Type
CREDENTIAL	Set Credential

Options

-D	--dataFile=dataFile	Path (absolute or relative) of user input files for user provisioning process (multiple files supported)
-N	--nodeProvisioner=nodeProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Compute node only
-S	--serverProvisioner=serverProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Head node only
-c	--custom	Use Managed Image
-d	--description=description	Description
-h	--help	Show Help
-i	--instance=instance	(required) [default: t2.micro] Instance Type
-p	--prefix=prefix	(required) [default: nodus] Output Image Prefix
-r	--region=region	(required) Region ID
-s	--source=source	Source Image ID
-u	--username=username	SSH Username
	--private	Provision via Private IP
	--proxy=proxy	HTTP(S) Proxy URL
	--security=security	Security Group ID
	--subnet=subnet	Subnet ID
	--volume=volume	Storage Volume (GB)

Examples

```
$ nodus stack:aws:create myStack centos-7 myCred --region=us-east-2
```



```
$ nodus stack:aws:create myStack centos-7 --prefix=myNodus --region=us-east-2 --instance=t2.large
$ nodus stack:aws:create myStack centos-7 --region=us-east-2 --proxy $http_proxy
$ nodus stack:aws:create myStack centos-7 --region=us-east-2 --custom --source=ami-04152c3a27c49a944 --username=ec2-user --instance=t2.xlarge
$ nodus stack:aws:create myStack centos-7 --region=us-east-2 --instance=t2.xlarge --serverProvisioner ~/myServerScript --nodeProvisioner ~/myNodeScript --dataFile ~/myDataFile2
```

stack:aws:update STACK

Description

Update Google Cloud Platform (GCP) cluster configuration entry.

Usage

```
$ nodus stack:aws:update STACK [OPTIONS]
```

Arguments

STACK	Stack Name or ID
-------	------------------

Options

-D	--dataFile=dataFile	Input files for user provisioning process (multiple files supported)
-N	--nodeProvisioner=nodeProvisioner	User provisioning script to run during stack build on Compute node only
-S	--server-Provisioner=serverProvisioner	User provisioning script to run during stack build on Head node only
-c	--custom	Use Managed Image
-d	--description=description	Description
-h	--help	Show Help
-i	--instance=instance	Instance Type
-n	--name=name	Name Identifier
-p	--prefix=prefix	Output Image Prefix
-r	--region=region	Region ID
-s	--source=source	Source Image ID
-t	--type=centos-7	OS Type
-u	--username=username	SSH Username
	--no-proxy	Remove HTTP(S) Proxy
	--no-security	Remove Security Group
	--no-subnet	Remove Subnet
	--no-volume	Remove Extended Root Storage Volume

Options		
--[no-]private		Provision via Private IP
--proxy=proxy		HTTP(S) Proxy URL
--security=security		Security Group ID
--subnet=subnet		Subnet ID
--volume=volume		Extended Root Storage Volume (GB)
Examples		
\$ nodus stack:aws:update myStack --name newStack		
\$ nodus stack:aws:update myStack --region=us-east-2		
\$ nodus stack:aws:update myStack --prefix=myNodus --region=us-east-2 --instance=t2.large		
\$ nodus stack:aws:update myStack --region us-east-2 --proxy \$http_proxy		
\$ nodus stack:aws:update myStack --region=us-east-2 --custom --source=ami-04152c3a27c49a944 --username=ec2-user --instance=t2.xlarge		

stack:azure:create NAME TYPE

Description

Create new Microsoft Azure stack configuration entry.

Usage

```
$ nodus stack:azure:create NAME TYPE [CREDENTIAL] [OPTIONS]
```

Arguments

NAME	Associated Name
TYPE	(centos-7) Image Type
CREDENTIAL	Set Credential

Options

-D	--dataFile=dataFile	Path (absolute or relative) of user input files for user provisioning process (multiple files supported)
-N	--nodeProvisioner=nodeProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Compute node only
-S	--serverProvisioner=serverProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Head node only
-c	--custom	Use Managed Image
-d	--description=description	Description
-h	--help	Show Help

-i	--instance=instance	(required) [default: Standard_DS2_v2] Instance Type ID
-p	--prefix=prefix	(required) [default: nodus] Output Image Prefix
-s	--source=source	Source Image ID
-u	--username=username	SSH Username
	--offer=offer	Source Image Offer
	--os-type=os-type	Source Image OS Type
	--publisher=publisher	Source Image Publisher
	--resource=resource	(required) Managed Resource Group Name
	--sku=sku	Source Image SKU
	--source-resource=source-resource	Source Resource Group Name

Examples

```
$ nodus stack:azure:create myStack centos-7 myCred --resource=myResourceGroup --instance=Standard_DS2_v2
```

```
$ nodus stack:azure:create myStack centos-7 myCred --resource=myResourceGroup -S ~/myServerScript -N ~/myNodeScript -D ~/myDataFile2 -D ~/myDataFile2
```

```
$ nodus stack:azure:create myStack centos-7 --prefix=myNodus --resource=myResourceGroup --instance=Standard_DS2_v2
```

stack:azure:update STACK

Description

Update Microsoft Azure stack configuration entry.

Usage

```
$ nodus stack:azure:update STACK [OPTIONS]
```

Arguments

STACK	Stack Name or ID
-------	------------------

Options

-D	--dataFile=dataFile	Input files for user provisioning process (multiple files supported)
-N	--nodeProvisioner=nodeProvisioner	User provisioning script to run during stack build on Compute node only
-S	--serverProvisioner=serverProvisioner	User provisioning script to run during stack build on Head node only
-c	--custom	Use Managed Image
-d	--description=description	Description
-h	--help	Show Help
-i	--instance=instance	(required) [default: Standard_DS2_v2] Instance Type ID

Options		
-n	--name=name	Name Identifier
-p	--prefix=prefix	(required) [default: nodus] Output Image Prefix
-s	--source=source	Source Image ID
-t	--type=centos-7	OS Type
-u	--username=username	SSH Username
	--os-type=os-type	Source Image OS Type
	--publisher=publisher	Source Image Publisher
	--resource=resource	(required) Managed Resource Group Name
	--sku=sku	Source Image SKU
	--source-resource=source-resource	Source Resource Group Name
Examples		
\$ nodus stack:azure:update myStack --name newStack		
\$ nodus stack:azure:update myStack --resource=myResourceGroup --instance=Standard_DS2_v2		
\$ nodus stack:azure:update myStack --prefix=myNodus --resource=myResourceGroup --instance=Standard_DS2_v2		

stack:build STACK

Description

Generates images compatible with NODUS clusters. Generate two images: 'nodus-server'; 'nodus-node'.

Usage

```
$ nodus stack:build STACK [OPTIONS]
```

Argument

STACK	Stack Name or ID
-------	------------------

Options

-S	--slow	Sequential Builds
-h	--help	Show Help
-s	--stack=server node	Sub-Stack

Examples

```
$ nodus stack:build myStack
```

```
$ nodus stack:build myStack --stack server
```

stack:gcp:create NAME TYPE**Description**

Create new Google Cloud Platform (GCP) stack configuration entry.

Usage

```
$ nodus stack:gcp:create NAME TYPE [CREDENTIAL] [OPTIONS]
```

Arguments

NAME	Associated Name
TYPE	(centos-7) Image Type
CREDENTIAL	Set Credential

Options

-D	--dataFile=dataFile	Path (absolute or relative) of user input files for user provisioning process (multiple files supported)
-N	--nodeProvisioner=nodeProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Compute node only
-S	--serverProvisioner=serverProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Head node only
-c	--custom	Managed Image
-d	--disk=disk	Disk Size (GB)
-h	--help	Show Help
-i	--instance=instance	(required) [default: n1-standard-1] Instance Type ID
-p	--prefix=prefix	(required) [default: nodus] Output Image Prefix
-s	--source=source	Source AMI ID
-u	--username=username	SSH Username
-z	--zone=zone	(required) Availability Zone ID
	--[no-]account	Auth via Service Account
	--disk=disk	Extended Root Disk Size (GB)
	--proxy=proxy	HTTP(S) Proxy URL
	--subnet=subnet	Subnet Self Link ID

Examples

```
$ nodus stack:gcp:create myStack centos-7 myCred --zone=us-east1-b --instance=n1-standard-1
```

```
$ nodus stack:gcp:create myStack centos-7 --prefix=myNodus --zone=us-east1-b --instance=n1-standard-1
```

```
$ nodus stack:gcp:create myStack centos-7 myCred -z us-east4-b -p myPrefix --
```

```
serverProvisioner ~/myServerScript --nodeProvisioner ~/myNodeScript --dataFile
~/myDataFile1 --dataFile ~/myDataFile2
```

stack:gcp:update STACK

Description

Update Google Cloud Platform (GCP) stack configuration entry.

Usage

```
$ nodus stack:gcp:update STACK [OPTIONS]
```

Arguments

STACK	Stack Name or ID
-------	------------------

Options

-D	--dataFile=dataFile	Input files for user provisioning process (multiple files supported)
-N	--nodeProvisioner=nodeProvisioner	User provisioning script to run during stack build on Compute node only
-S	--server-Provisioner=serverProvisioner	User provisioning script to run during stack build on Head node only
-c	--custom	Managed Image
-d	--description=description	Description
-h	--help	Show Help
-i	--instance=instance	Instance Type ID
-n	--name=name	Name Identifier
-p	--prefix=prefix	Output Image Prefix
-s	--source=source	Source AMI ID
-t	--type=centos-7	OS Type
-u	--username=username	SSH Username
-w	--walltime=walltime	Job walltime
-z	--zone=zone	Availability Zone ID
	--[no-]account	Auth via Service Account
	--disk=disk	Extended Root Disk Size (GB)
	--no-disk	Remove Disk
	--no-proxy	Remove HTTP(S) Proxy
	--no-subnet	Remove Subnet
	--proxy=proxy	HTTP(S) Proxy URL
	--subnet=subnet	Subnet Self Link ID

Examples

```
$ nodus stack:gcp:update myStack --name newStack
```

Examples

```
$ nodus stack:gcp:update myStack --zone=us-east1-b --instance=n1-standard-1
```

```
$ nodus stack:gcp:update myStack --prefix=myNodus --zone=us-east1-b --instance=n1-standard-1
```

stack:list**Description**

List stack entries from database.

Usage

```
$ nodus stack:list [OPTIONS]
```

Options

-f	--format=text json	[default: text] Output Format
-h	--help	Show Help

Examples

```
$ nodus stack:list
```

```
$ nodus stack:list --format json
```

stack:log CLUSTER TYPE**Description**

Display stack build logs.

Usage

```
$ nodus stack:log CLUSTER TYPE [OPTIONS]
```

Arguments

STACK	Stack Name or ID
TYPE	(master build) [default: master] Log Type

Options

-d	--debug	Debug Logs
-f	--follow	Follow File Output Stream
-h	--help	Show Help
-n	--lines=lines	Last N Lines
	--follow	Follow File Output Stream

Example

```
$ nodus stack:log myCluster
```

stack:oracle:create NAME TYPE**Description**

Create new Oracle Cloud stack configuration entry.

Usage

```
$ nodus stack:oracle:create NAME TYPE [CREDENTIAL] [OPTIONS]
```

Arguments

NAME	Associated Name
TYPE	(centos-7) Image Type
CREDENTIAL	Set Credential

Options

-D	--dataFile=dataFile	Path (absolute or relative) of user input files for user provisioning process (multiple files supported)
-N	--nodeProvisioner=nodeProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Compute node only
-S	--server-Provisioner=serverProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Head node only
-c	--custom	Use Managed Image
-d	--description=description	Description
-h	--help	Show Help
-i	--instance=instance	(required) [default: VM.Standard2.1] Instance Type ID
-n	--subnet=subnet	(required) Subnet OCID
-p	--prefix=prefix	(required) [default: nodus] Output Image Prefix
-r	--region=region	(required) Region ID
-s	--source=source	Source Image ID
-u	--username=username	SSH Username

Example

```
$ nodus stack:oracle:create myStack centos-7 myCred --prefix=myNodus --region=us-ashburn-1 --subnet=ocid1.subnet.oc1.iad.aaaaaaa7v2znysxlclowk7rhgkglvjaxmkkjpiktnahnusgknhw5pcvi6q
$ nodus stack:oracle:create myStack centos-7 myCred --prefix=myNodus --region=us-ashburn-1 --subnet=mySubnet -S ~/myServerScript -N ~/myNodeScript -D ~/myDataFile1 -D ~/myDataFile2
```


stack:oracle:update STACK**Description**

Update Oracle Cloud stack configuration entry.

Usage

```
$ nodus stack:oracle:update STACK [OPTIONS]
```

Arguments

STACK	Stack Name or ID
-------	------------------

Options

-D	--dataFile=dataFile	Input files for user provisioning process (multiple files supported)
-N	--nodeProvisioner=nodeProvisioner	User provisioning script to run during stack build on Compute node only
-S	--server-Provisioner=serverProvisioner	User provisioning script to run during stack build on Head node only
-c	--custom	Use Managed Image
-d	--description=description	Description
-h	--help	Show Help
-i	--instance=instance	Instance Type ID
-n	--name=name	Name Identifier
-n	--subnet=subnet	(required) Subnet OCID
-p	--prefix=prefix	Output Image Prefix
-r	--region=region	Region ID
-s	--source=source	Source Image ID
-t	--type=centos-7	OS Type
-u	--username=username	SSH Username

Examples

```
$ nodus stack:oracle:update myStack --name newStack
```

```
$ nodus stack:oracle:update myStack --prefix=myNodus --region=us-ashburn-1 --
subnet=ocid1.subnet.oc1.iad.aaaaaaa7v2znysxlclowk7rhgkgldvjaxmkkjpiktnahnusgknhw
5pcvi6q
```

stack:otc:create NAME TYPE**Description**

Create new OpenTelekom Cloud stack configuration entry.

Usage

```
$ nodus stack:otc:create NAME TYPE [CREDENTIAL] [OPTIONS]
```

Arguments		
NAME		Associated Name
TYPE		(centos-7) Image Type
CREDENTIAL		Set Credential
Options		
-D	--dataFile=dataFile	Path (absolute or relative) of user input files for user provisioning process (multiple files supported)
-N	--nodeProvisioner=nodeProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Compute node only
-S	--server-Provisioner=serverProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Head node only
-c	--custom	Use Managed Image
-d	--description=description	Description
-h	--help	Show Help
-i	--instance=instance	(required) [default: s2.medium.2] Instance Type ID
-p	--prefix=prefix	(required) [default: nodus] Output Image Prefix
-s	--source=source	Source Image ID
-u	--username=username	SSH Username
	--private	Provision via Private IP
	--proxy=proxy	HTTP(S) Proxy URL
	--security=security	Security Group ID
	--subnet=subnet	(required) Subnet Network ID
	--volume=volume	Extended Root Storage Volume (GB)
Examples		
\$ nodus stack:otc:create myStack centos-7 myCred --subnet 3bc69be8-3ebc-434c-b828-9aeb5a8ec78a --instance s2.medium.2		
\$ nodus stack:otc:create myStack centos-7 --prefix myNodus --subnet 3bc69be8-3ebc-434c-b828-9aeb5a8ec78a --instance s2.medium.2		
\$ nodus stack:otc:create myStack centos-7 --subnet 3bc69be8-3ebc-434c-b828-9aeb5a8ec78a --proxy \$http_proxy		

stack:otc:create STACK

Description

Create new OpenTelekom Cloud stack configuration entry.

Usage		
\$ nodus stack:otc:create STACK [CREDENTIAL] [OPTIONS]		
Arguments		
STACK		Stack Name or ID
CREDENTIAL		Set Credential
Options		
-D	--dataFile=dataFile	Path (absolute or relative) of user input files for user provisioning process (multiple files supported)
-N	--nodeProvisioner=nodeProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Compute node only
-S	--server-Provisioner=serverProvisioner	Path (absolute or relative) of user provisioning script to run during stack build on Head node only
-c	--custom	Use Managed Image
-d	--description=description	Description
-h	--help	Show Help
-i	--instance=instance	Instance Type ID
-n	--name=name	Name Identifier
-p	--prefix=prefix	Output Image Prefix
-s	--source=source	Source Image ID
-t	--type=centos-7	OS Type
-u	--username=username	SSH Username
	--no-proxy	Remove HTTP(S) Proxy
	--no-security	Remove Security Group
	--no-volume	Remove Extended Root Storage Volume
	--[no-]private	Provision via Private IP
	--proxy=proxy	HTTP(S) Proxy URL
	--security=security	Security Group ID
	--subnet=subnet	Subnet Network ID
	--volume=volume	Extended Root Storage Volume (GB)
Examples		
\$ nodus stack:otc:create myStack myCred --name newStack		
\$ nodus stack:otc:create myStack --subnet 3bc69be8-3ebc-434c-b828-9aeb5a8ec78a --instance s2.medium.2		
\$ nodus stack:otc:create myStack --prefix myNodus --subnet 3bc69be8-3ebc-434c-b828-9aeb5a8ec78a --instance s2.medium.2		

```
$ nodus stack:otc:create myStack --subnet 3bc69be8-3ebc-434c-b828-9aeb5a8ec78a --
proxy $http_proxy
```

stack:remove STACK

Description

Remove stack entries from database.

Usage

```
$ nodus stack:remove STACK [OPTIONS]
```

Argument

STACK	Stack Name or ID
-------	------------------

Options

-h	--help	Show Help
	--[no-]clean	Remove Files Created for Cluster

Examples

```
$ nodus stack:remove myStack
$ nodus stack:remove myStack --no-clean
$ nodus stack:remove myStack --force
```

stack:set STACK CREDENTIAL

Description

Associate stack entry with credential entry.

Usage

```
$ nodus stack:set STACK CREDENTIAL [OPTIONS]
```

Arguments

STACK	Stack Name or ID
CREDENTIAL	Credential Name or ID

Option

-h	--help	Show Help
----	--------	-----------

Example

```
$ nodus stack:set myStack myCredential
```

stack:share STACK USERNAME

Description

Share your stack to another user.

Usage

```
$ nodus stack:share STACK USERNAME
```

Argument

STACK	Stack Name or ID
USERNAME	Linux Username

Options

-h	--help	Show Help
----	--------	-----------

Examples

```
$ nodus stack:share myStack otherUser
```

stack:show STACK**Description**

Display details of a particular stack entry from the database.

Usage

```
$ nodus stack:show STACK [OPTIONS]
```

Argument

STACK	Stack Name or ID
-------	------------------

Options

-f	--format=text json xml	[default: text] Output Format
-h	--help	Show Help

Examples

```
$ nodus stack:show myStack
```

```
$ nodus stack:show myStack --format json
```

stack:unshare STACK**Description**

Unshare your stack to another user.

Usage

```
$ nodus stack:unshare STACK [USERNAME] [OPTIONS]
```

Argument

STACK	Stack Name or ID
USERNAME	Linux Username

Options

-h	--help	Show Help
	--all	Remove all shared users

Examples

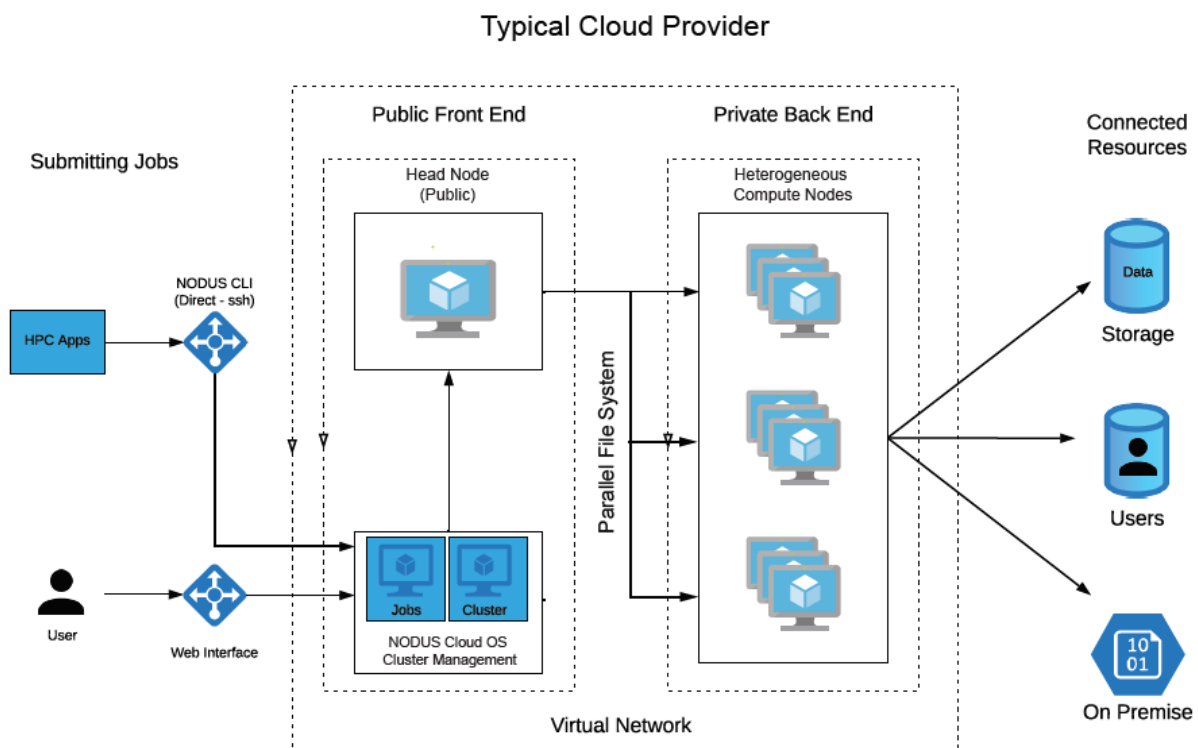
```
$ nodus stack:unshare myStack otherUser
```

```
$ nodus stack:unshare myStack --all
```

2.2 Users

An ODDC user is just a registered Linux system user that enables using the ODDC in various capacities, permissions, and license permitting.

2.3 Architecture



Chapter 3: Using the On-Demand Data Center GUI

This chapter provides information about using the ODDC GUI to run your workloads in the cloud.

In this chapter:

- 3.1 Creating an On-Demand Data Center Account31
- 3.2 User Profile 32
 - 3.2.1 Settings - Admins 32
 - 3.2.2 Settings - Users 33
- 3.3 Credentials Manager 34
- 3.4 Stack Manager 35
 - 3.4.1 Viewing Stacks 36
 - 3.4.2 Creating a Stack 36
 - 3.4.3 Building a Stack 37
- 3.5 Cluster Manager 37
 - 3.5.1 Viewing Clusters 37
 - 3.5.2 Creating a Cluster 39
 - 3.5.3 Deploying a Cluster 40
 - 3.5.4 Deleting a Cluster 40
- 3.6 Job Manager 40
 - 3.6.1 Adding a New Job 41
 - 3.6.2 Adding a Temporary On Demand Job 43
- 3.7 File Manager 47
- 3.8 Accounting 47
- 3.9 Bursting 49
 - 3.9.1 Bursting Configurations 49
 - 3.9.2 Bursting Service 50

3.1 Creating an On-Demand Data Center Account

You can have one of two different types of accounts, depending on how your server is set up: Site Account or Multi-User Account.

1. To create an account, go to `http://<web_server_address>` and click **Create an On-Demand Data Center account**.

Note: `web_server_address` refers to the server where the ODDC platform is installed.

2. Follow the instructions and complete the registration process. Your username and password are required for future logins. The password must be at least 8 characters long, contain at least 1 lowercase, 1 uppercase, 1 numeric, and 1 special character such as `!@#$$%^&*`.

Site Account

Upon successful registration, you will be logged in to `http://<web_server_address>`.

Multi-User Account

This message appears: Account Creation Successful! In order to access the On-Demand Data Center, you will need a USER KEY. If you do not already have one, please contact Adaptive Computing at `license@adaptivecomputing.com`!

3.2 User Profile

From your user profile at the top right of the screen, these options are available:

- **Settings** - Add new users and/or edit profiles, depending on whether you are an admin or a user. Your user profile shows whether you are an admin or a user.
- **User SSH Key** - Download a pem file to access the head node of a cluster via SSH. This provides you full control over the cluster.
- **Logout**.

3.2.1 Settings - Admins

Admins can maintain users' information/roles and create new users. The first user created has an administrator role and all others have a user role.

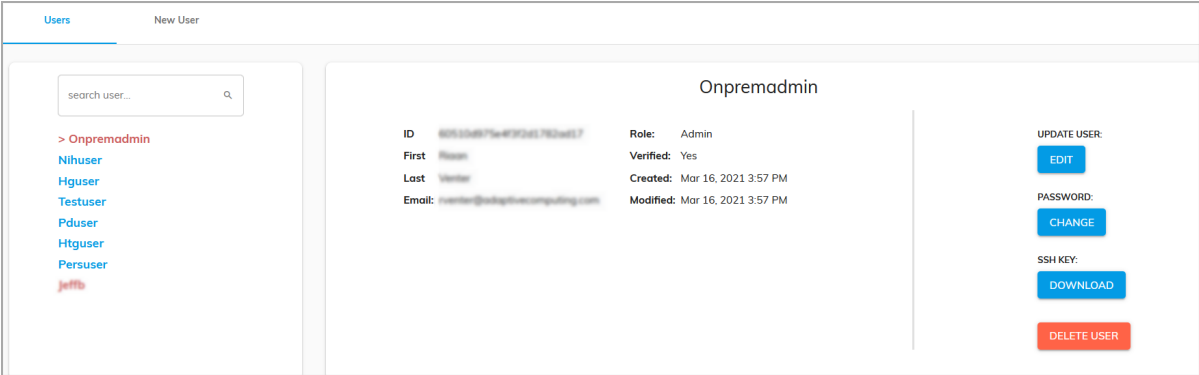
Users have limited ODDC features permissions:

- Cluster Manager - View clusters.
- Stack Manager - No permissions.
- Credentials Manager - No permissions.

- Job Manager - Add and submit jobs.
- File Manager - View, download, and delete jobs.

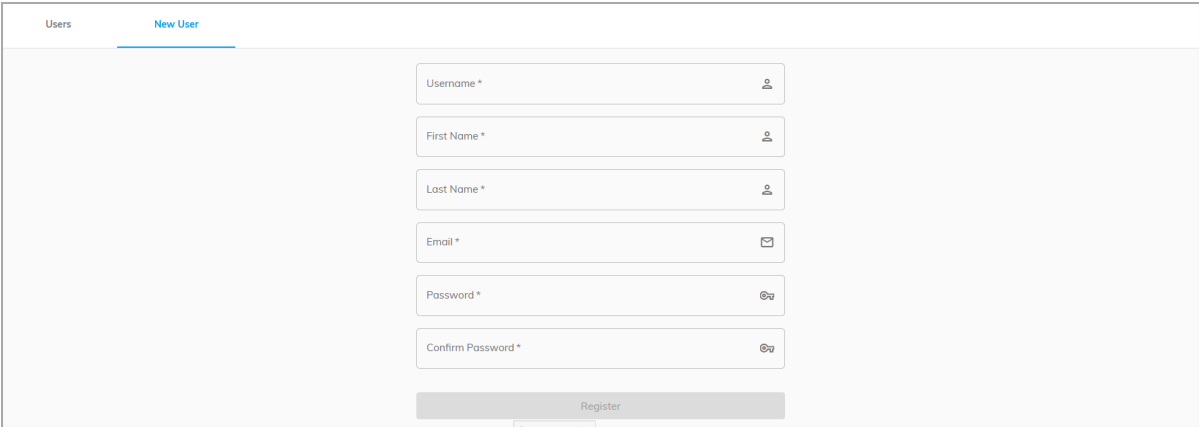
3.2.1.A Editing User Profiles

1. On the **Users** tab, click **Edit** to change a user's name/email or change an admin to a user and vice versa. You can also change a user's password, download the SSH Key, and delete a user.



3.2.1.B Adding a New User

1. To add a new user, on the **New User** tab, complete the fields and then click **Register**.



3.2.2 Settings - Users

Users can edit their name/email and change their password.

user

ID

60c20933209ac74013b48115ed8

First

jeff

Last

boley

Email:

jeffboley@redhat.com

Role:

User

Verified:

Yes

Created:

Jun 10, 2021 8:27 AM

Modified:

Jun 10, 2021 8:27 AM

UPDATE USER:

EDIT

PASSWORD:

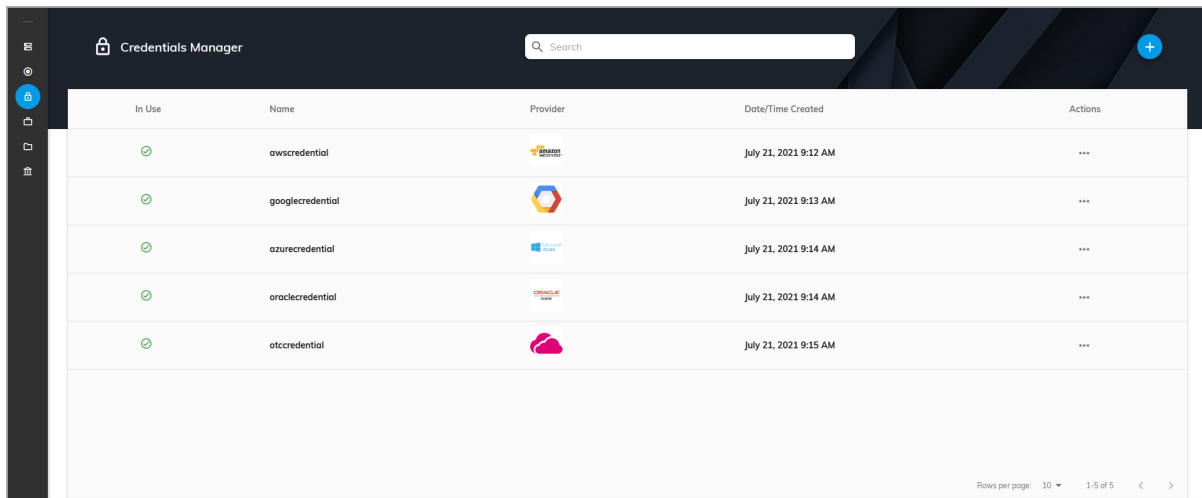
CHANGE

3.3 Credentials Manager

Credentials are required by the cloud service providers in order to access their environment and resources. Adding and saving them now means that you can easily select them with just one click when creating a cluster.

1. To add credentials, log in to your ODDC user account at `http://<web_server_address>`. The **Cluster Manager** screen opens.

2. On the **Applications** menu, click **Credentials Manager**:



The screenshot shows the 'Credentials Manager' interface. It features a dark sidebar on the left with navigation icons. The main area has a header with a search bar and a '+ Add' button. Below is a table with columns: In Use, Name, Provider, Date/Time Created, and Actions. The table contains five rows of credentials for AWS, Google, Azure, Oracle, and OCI. Each row has a green checkmark in the 'In Use' column and a three-dot menu in the 'Actions' column. At the bottom right, there is a pagination control showing 'Rows per page: 10' and '1-5 of 5'.

In Use	Name	Provider	Date/Time Created	Actions
✓	awscredential	AWS	July 21, 2021 9:12 AM	...
✓	googlecredential	Google	July 21, 2021 9:13 AM	...
✓	azurecredential	Azure	July 21, 2021 9:14 AM	...
✓	oraclecredential	Oracle	July 21, 2021 9:14 AM	...
✓	otccredential	OCI	July 21, 2021 9:15 AM	...

The **Credentials Manager** screen shows this information: In Use, Name, Provider, Date/Time Created, and Actions (...). Edit Collaborators, Set, & Delete. **Note:** For Edit Collaborators, the License Type must be Site.

Note: From the **Applications** menu, you can also create clusters, build stacks, submit jobs, and view a job's output file. See the sections [Cluster Manager](#), [Stack Manager](#), [Job Manager](#), and [File Manager](#) for information.

3. Select the appropriate cloud service provider.
4. On the **New Credential** screen, enter the desired name for the credentials.
5. Enter your account credentials: Access Key, Secret Key, Session Token, and Description. Alternatively, click **Upload Credential**.
6. Click **Save** to finish adding the credentials for this provider.

Note: Clicking the name of a credential shows information about it.

Repeat the steps in this section to add credentials for another provider if desired.

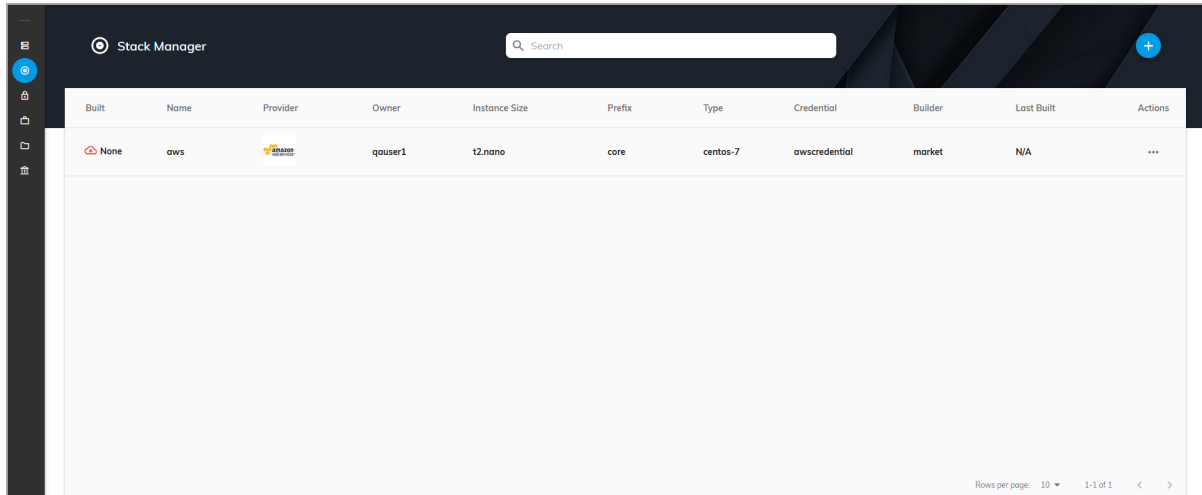
3.4 Stack Manager

This section contains the following information:

- [Viewing Stacks](#)
- [Creating a Stack](#)
- [Building a Stack](#)

3.4.1 Viewing Stacks

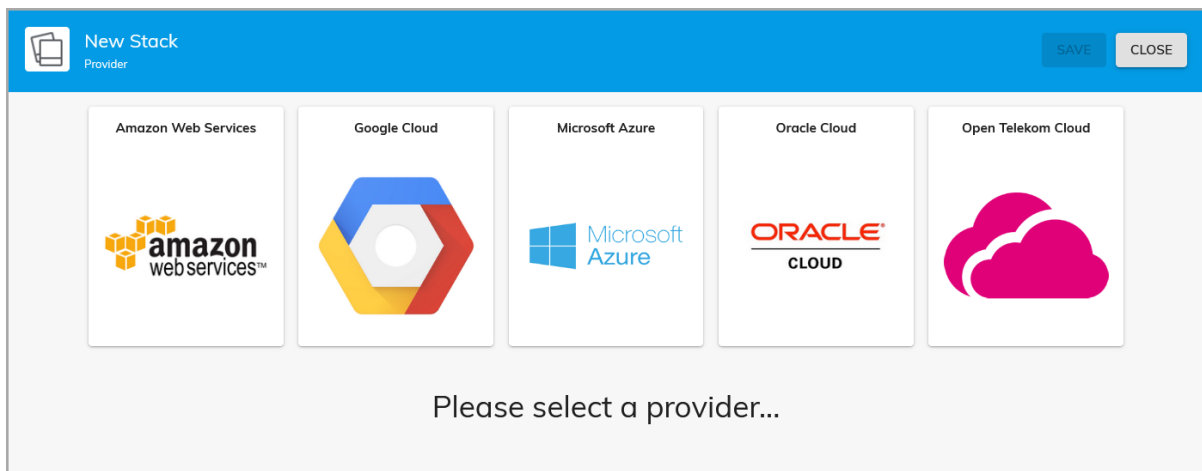
1. On the **Applications** menu, click **Stack Manager**:



On the **Stack Manager** screen, you can create new stacks and see a list of all your current stacks and information such as: Built (Success / Failed / None), Name, Provider, Instance Size, Prefix, Type, Builder, Last Built, and Actions (Edit Collaborators, Edit, Build, Logs, & Delete). **Note:** For Edit Collaborators, the License Type must be Site.

3.4.2 Creating a Stack

1. To create a new stack, on the **Stack Manager** screen, click the add icon (+). The **New Stack** screen shows the cloud service providers that you have credentials for:



2. Select the cloud service provider that you want to create a stack in.

3. Enter the desired information for the new stack: Name, Type, Prefix, Credential, Head Node Size, Subnet Network, and Description.

Advanced

Clicking the **Advanced** button shows these additional fields: Private IP, Proxy, Source Image, SSH Username, Security, and Volume. **Note:** The available fields vary based on the provider.

4. Click **Save**. The state shows `None` until it is built.
5. Repeat the steps in this section to create additional clusters as desired.

3.4.3 Building a Stack

1. To build a stack, under **Actions**, click **Build**. The state changes to `Success`. If it fails, the state changes to `Failed`.

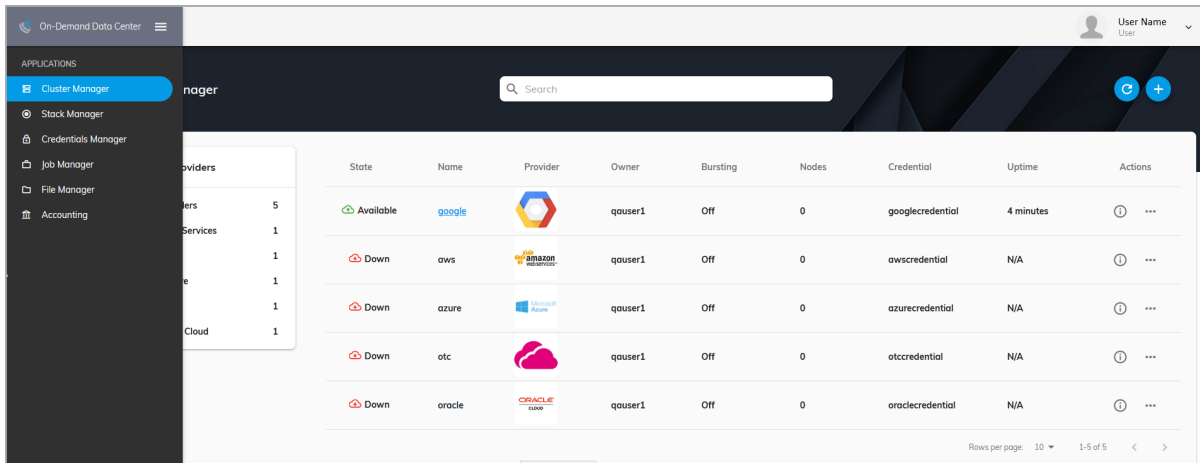
3.5 Cluster Manager

This section contains the following information:

- [Viewing Clusters](#)
- [Creating a Cluster](#)
- [Deploying a Cluster](#)
- [Deleting a Cluster](#)

3.5.1 Viewing Clusters

1. On the **Applications** menu, click **Cluster Manager**:



On the **Cluster Manager** screen, you can configure new clusters and see a list of all your current clusters and information such as their State (Available / Down / Failed), Name, Provider, Bursting, Nodes, Credential, Uptime, and Actions (Edit / Edit Collaborators, Deploy, Destroy, Configuration, Logs, Delete, and Cluster SSH Key). The Actions vary based on the State and License Type (see the notes below for additional information).

Notes:

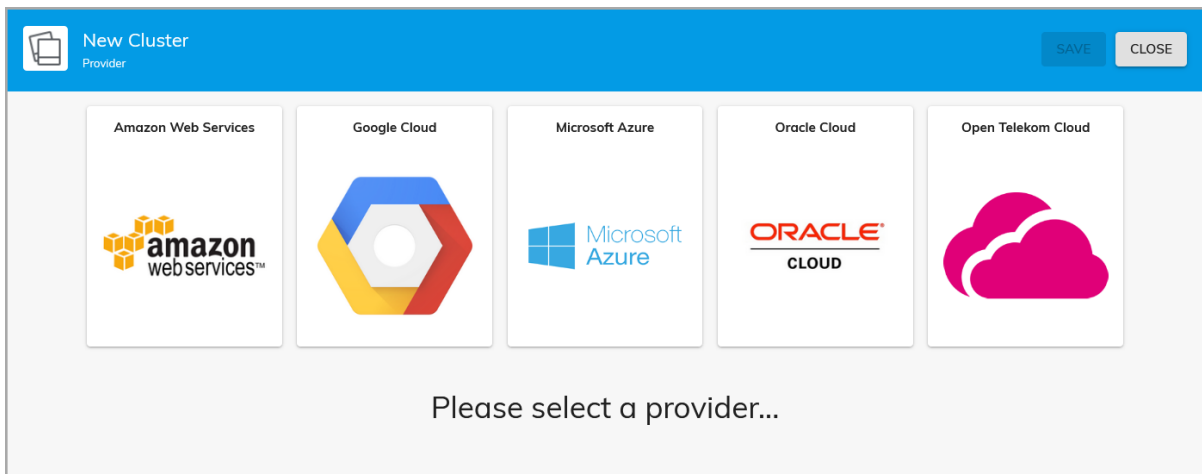
- Clicking **User SSH Key** next to your user name will download a .pem file that can be used to access the head node of a cluster via SSH. This provides you full control over the cluster.
- Clicking the refresh icon (🔄) at the top right of the cluster list, refreshes the list.
- Clicking the add icon (+) at the top right opens the **New Cluster** screen.
- Clicking the name of a cluster shows cluster details (see the section [Adding a New Job](#) for information).
- Clicking the information icon (i) at the end of the row shows additional details about the cluster, including: Cluster ID, SSH Username, Manager, Created, Modified, Expected Deploy Time, Actual Deploy Time, Uptime, Head Node Public IP, Prefix, Region, Head Node Size, and Compute Node Sizes. The buttons **Deploy/Destroy** and **Logs** also show.
- Clicking the horizontal ellipsis (...) at the end of the cluster row shows actions that you can take regarding the cluster:
 - **Edit** - Modifies the configuration of the cluster. **Note:** The cluster must be in a Down state to edit.
 - **Configuration** - Open the Configuration screen to view/modify Cluster Information, Cluster Compute Nodes, Bursting Service, and Disks Information.

- **Deploy** - Deploys the cluster into an available state.
 - **Destroy** - Destroys the head node and the linked compute nodes so that they are no longer in a state to do work. This also means that you will not be billed further for these cloud resources. After a cluster is destroyed, the state shows as `Down`.
 - **Logs** - Shows information about the provisioning. A screen shows information in real-time about the cluster's state. Additionally, these logs contain information for deploy and destroy.
 - **Delete** - Permanently deletes the cluster. **Note:** The cluster must be down before it can be deleted.
 - **Cluster SSH Key** - Downloads the cluster SSH key.
 - **Edit Collaborators** - Shares the cluster. **Note:** The License Type must be Site.
-

3.5.2 Creating a Cluster

If you have already added your cloud credentials for the chosen provider, follow the steps below to create a new cluster, which can then be deployed at any time. If you have not yet added your credentials, you must first follow the steps in the section [Credentials Manager](#).

1. To create a new cluster, on the **Cluster Manager** screen, click the add icon (+). The **New Cluster** screen shows the Cloud service providers that you have credentials for:



2. Select the Cloud service provider that you want to deploy a cluster in.
3. Enter the desired information for the new cluster, such as: Name, OS Type, Prefix, Credential, Head Node Size, Manager, Region, Availability Zone, Bursting Configuration (Off, Min, Max, All), Compute Nodes, Size, Count, and Description.

Advanced

Clicking the **Advanced** button shows these additional fields: Subnet, VPC, Security

Group, Elastic IP Address, Private IP, Server Volume (GB), Node Volume (GB), Idle Purge Time, User Script, Share (NFS/BeeGFS), OpenVPN, and Packages.

Notes:

- The available fields vary based on the provider.
 - Multiple Compute Node Instances of different sizes can be added.
 - The minimum Idle Purge Time cannot be less than 180 seconds (3 minutes). When selecting Bursting Configuration: All, we recommend setting the Idle Purge Time to at least 1,800 seconds (30 minutes).
 - The available Regions, Availability Zones, Server Instance Sizes, Compute Node Instance Sizes, and quantity of instances may vary based on your provider account status.
-

4. Click **Save**. The state shows `Down` until it is deployed.
5. Repeat the steps in this section to create additional clusters as desired.

3.5.3 Deploying a Cluster

1. To deploy a cluster, click the ellipsis and then click **Deploy**. The state changes to `Available`.

3.5.4 Deleting a Cluster

To delete a cluster, the state must first be `Down`.

1. Click the ellipsis for the desired cluster and then click **Destroy**.
2. After the state changes to `Down`, click **Delete**.

3.6 Job Manager

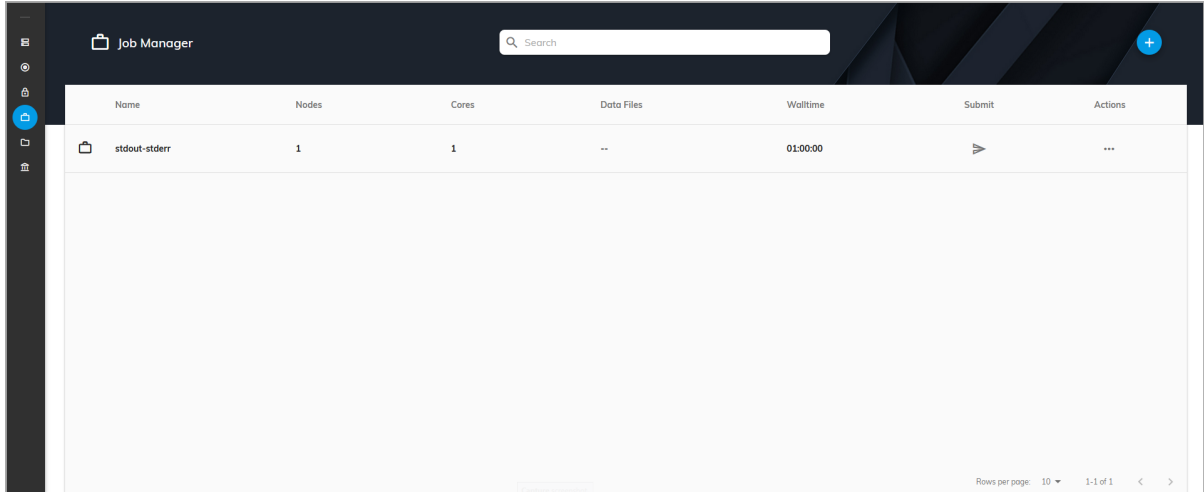
You can add jobs and submit them to either an existing cluster or a temporary on demand cluster by using Job Manager.

This section contains the following information:

- [Adding a New Job](#)
- [Adding a Temporary On Demand Job](#)

3.6.1 Adding a New Job

1. To add a new job, on the **Applications** menu, click **Job Manager**. The **Job Manager** screen opens:

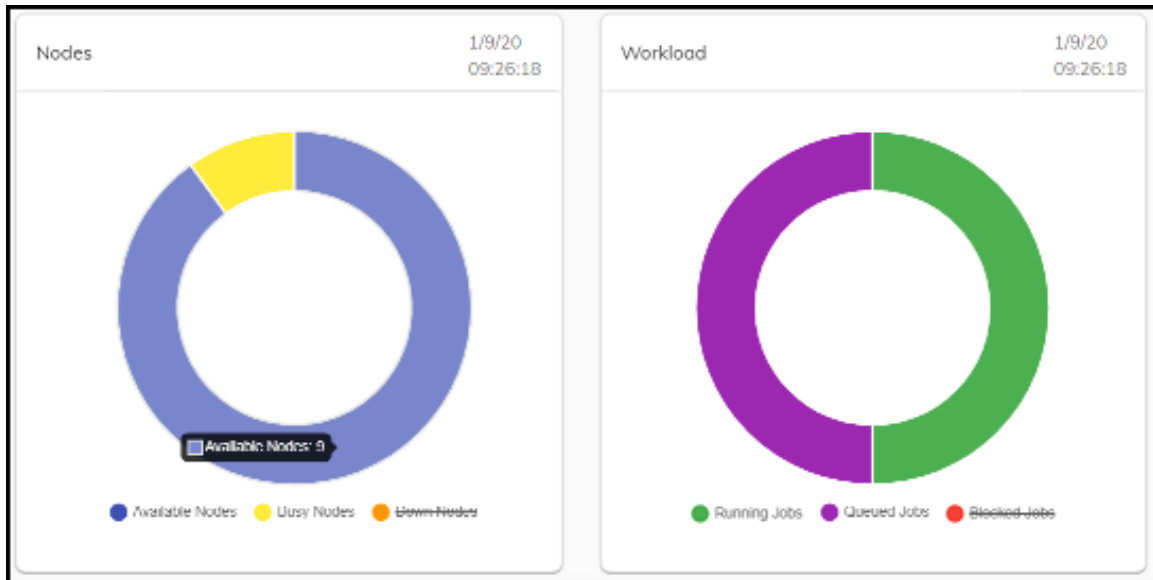


The screenshot shows the Job Manager interface. It features a dark sidebar on the left with navigation icons. The main area has a header with 'Job Manager' and a search bar. Below the header is a table with the following columns: Name, Nodes, Cores, Data Files, Walltime, Submit, and Actions. A single row is visible with the job name 'stdout-stderr', 1 node, 1 core, and a walltime of 01:00:00. The Actions column contains a submit icon and a menu icon. At the bottom right, there is a pagination control showing 'Rows per page: 10' and '1-1 of 1'.

Name	Nodes	Cores	Data Files	Walltime	Submit	Actions
stdout-stderr	1	1	--	01:00:00	▶	⋮

This screen shows this information: Name, Nodes, Cores, Data Files, Walltime, Submit, and Actions (Edit, Copy, & Delete).

2. Click the add icon (+) at the top right to open the **New Job** screen:
 - A. On the **Job Info** panel, enter the desired information for the new job: Name, Walltime, Number of Nodes, Cores per Node, On Demand, and Description. This can be edited after the job is saved by clicking the name of the job.
 - B. On the **Script** panel, either edit the script for this job or click **Upload Script**. This can be edited after the job is saved by clicking the name of the job.
 - C. (optional) On the **Data Files** panel, click **Upload Data** to upload any data files that the job may require. This can be edited after the job is saved by clicking the name of the job.
 - D. Click **Save**.
3. Click the submit (▶) icon, then select the cluster to run the job on. If there are no created clusters or ones that meet the hardware requirements of the job, then certain clusters may not appear.
4. Click **Submit**. The job is queued and runs on the selected cluster. The cluster details screen shows the submitted job information:
 - **Cluster Info** - Jobs Submitted (Total, Today, This Month, This Year), Queue (Running, Queued), Nodes (Available, Busy, Down, Offline), Activities Log (All, Job), and Nodes/Workload graph (see below).



This graph shows the status of Available Nodes, Busy Nodes, & Down Nodes, and the status of the Running Jobs, Queued Jobs, & Blocked Jobs. Click to deselect nodes and jobs to narrow the display results. Hover over the graph to see the number of nodes or jobs.

- **Queue** - ID, Name, State (Queued, Running, Completed), Submitted, Cores, Nodes, and Actions (⊗). **Note:** After a job completes, it only shows here temporarily, but can be viewed in [File Manager](#).
- **Nodes** - ☐ (Deploy, Destroy), Name, State (All Nodes, Busy, Available, Deploying, Offline, Destroying, and Down), Cores, Load Avg, and Actions (Details, Deploy, Destroy). These are the Actions / Details that display: State, Power State, Processors, Type, Service Port, Manager Port, Total Sockets, Total Nodes, Total Cores, Total Threads, Dedicated Sockets, Dedicated Nodes, Dedicated Cores, Dedicated Threads, Operating System, User, Sessions, N Sessions, Idle Time, Total Memory, Available Memory, Physical Memory, CPUs, Load Average, Net Load, State, CPU Clock, MAC Address, Torque Version, REC Time, and Jobs.
Notes: Nodes are available by default, but can be taken down by selecting **Destroy**. A node that is down has been deprovisioned in the Cloud service provider and does not count as a billable resource. A node that is offline is still provisioned in the Cloud service provider and isn't being used to run jobs at the moment, but counts as a billable resource.
- **Configuration - Cluster Information** (Cluster ID, Head Node Size, Image Name, Cluster IP, SSH Username, Created, Download Cluster SSH Key, and Install OpenVPN), **Cluster Compute Nodes** (Size, Count, and Resize Cluster), and **Bursting Service** (Min/Max, View Logs, Burst Once), and **Disks Information** (Manager Disks). The Manager Disks feature enables ODDC Administrators to add extra drives to the

head node. When the cluster is deployed, another drive is attached to the head node that can be shared. When the cluster is destroyed, the drive is also destroyed.

Notes: Resize requires bursting service to be disabled. See the sections [Bursting Configurations](#) and [Bursting Service](#) for information about bursting.

Note: Jobs can also quickly be run from any screen at any time by clicking the **Quick Launch / Briefcase** icon (📁) near the top left of the screen.

5. (optional) To add extra drives to the head node, click **Configuration / Manager Disks**, and then click the add icon (+). Select the desired information and then click **Close/Do**.

3.6.2 Adding a Temporary On Demand Job


Follow one of the options below to add a temporary on demand job:

- [Using the Add Icon](#)
- [Using the Quick Launch Icon](#)
- [Using the Submit Icon](#)

Using the Add Icon

1. On the **Applications** menu, click **Job Manager**.
2. Click the add icon (+) at the top right to open the **New Job** screen.
3. On the **Job Info** panel, enter the desired information for the new job: Name, Walltime, Number of Nodes, Number of Cores per Node, and Description. This can be edited after the job is saved by clicking the name of the job.
4. On the **Script** panel, either edit the script for this job or click **Upload Script**. This can be edited after the job is saved by clicking the name of the job.
5. On the **Data Files** panel, click **Upload Data** to upload any data files that the job may require. This can be edited after the job is saved by clicking the name of the job.
6. To run an on demand job, select **On Demand**, which will add On Demand Configuration options:

Chapter 3: Using the On-Demand Data Center GUI

 **mpi-test**
Job Details

SAVE

CLOSE

Job Info

Script

Data Files

Name *

mpi-test

Walltime

01:00:00

of Nodes *

1

Cores per Node *

2

On Demand

☒

On Demand Configuration

Provider *

▼

Type *

Prefix *

Credential *

▼

Head Node Size *

▼

Manager *

▼

Region *

▼

Description

7. Complete the required fields.

These are the **On Demand Types**:

- **Destroy Cluster** - The full cluster is destroyed including the head node.
 - **Destroy Compute Nodes** - The head node stays active and the compute nodes are destroyed.
 - **Offline Compute Nodes** - The head node stays active and the compute nodes go offline.
 - **Persist Cluster** - The head and compute nodes stay active.
-

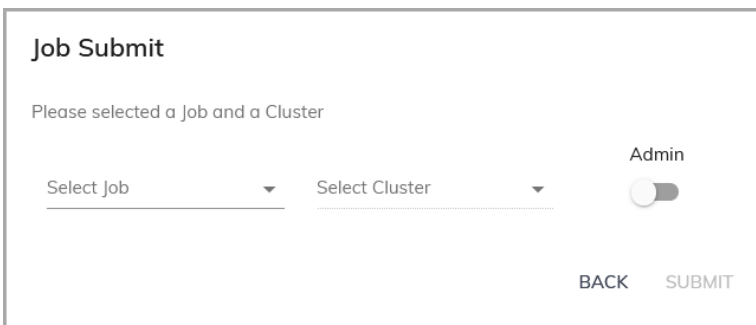
8. Click **Save** to finish adding the job.

9. On the **Job Manager** screen, click the arrow on your newly created job to submit your job. The ODDC will create a system-defined cluster name. **Note:** If you want to define your own cluster name, use the option just below before submitting your job, selecting your job defined here.

The defined cluster and node will be deployed, the job will run, and then your defined cluster will be destroyed based on your on demand type chosen.

Using the Quick Launch Icon

1. On the **Applications** menu, click **Job Manager**.
2. On the **Job Manager** screen, click the **Quick Launch / Briefcase** icon (📁) near the top left of the screen. The **Job Submit** dialog box opens:



Job Submit

Please selected a Job and a Cluster

Select Job ▼ Select Cluster ▼ Admin ☐

BACK SUBMIT

3. Select your job from the drop-down list and then select **On Demand** from the **Select Cluster** drop-down.
4. If you want to install software from your script before running the job or if the job requires admin rights, click the **Admin** button to be assigned admin rights.

5. Click **Submit**
6. Enter the cluster configuration options:

New Cluster

ADVANCED SAVE CLOSE

Name * OS Type * centos-7 Prefix * core

Credential * Head Node Size * s1.large - vCPU: 2 Manager * Torque

Region * Availability Zone *

Bursting Configuration: Persistent: ☐ Bursting: ☐ Min: ☐ Max: ☐

Compute Nodes: ⓘ +

Size * s1.large - vCPU: 2 Count * 0

Description

7. Click **Save** to finish adding the job and then click **Submit**.

The defined cluster and node will be deployed, the job will run, and then your defined cluster will be destroyed based on your on demand type chosen.

Using the Submit Icon

1. On the **Applications** menu, click **Job Manager**.
2. Click the submit (▶) icon:

Name	Nodes	Cores	Data Files	Walltime	Submit	Actions
stdout-stderr	1	1	--	01:00:00	▶	...

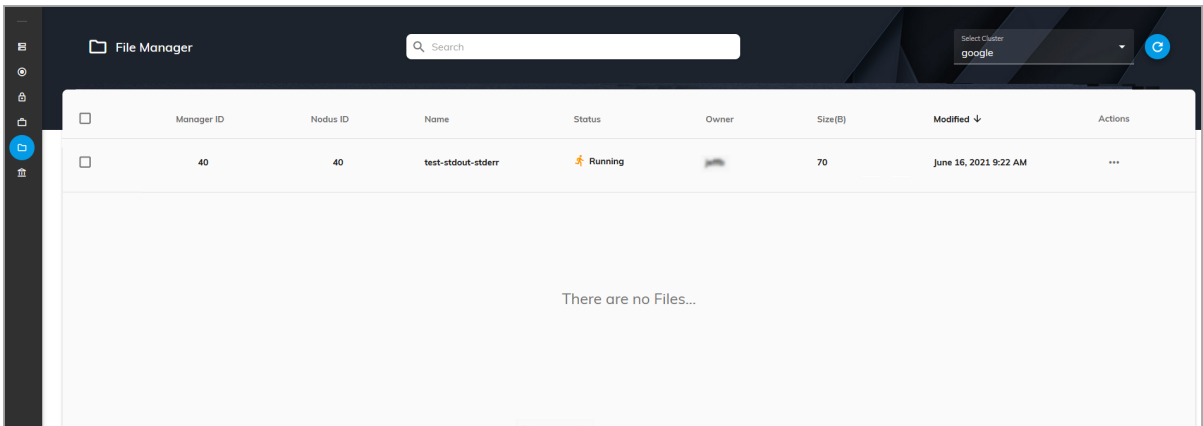
3. Enter the cluster configuration options.
4. Click **Save** to finish adding the job and then click **Submit**.

The defined cluster and node will be deployed, the job will run, and then your defined cluster will be destroyed based on your on demand type chosen.

3.7 File Manager

You can view, download, and delete a job's standard output file or error file using **File Manager**.

1. On the **Applications** menu, click **File Manager**, and then from the **Select Cluster** drop-down list at the top right, select a cluster:



The **File Manager** screen shows this information: Manager ID, Nodus ID, Name, Status, Owner, Size(B), Modified, and Actions.

2. Click the ellipsis at the end of a row and select the desired option: View STD-Out, Download STD-Out, View STD-Err, Download STD-Err, or Delete.

3.8 Accounting

Accounting shows an accounting of all the jobs run by each user over a period of time specified, and also the clusters. You must be an Admin to access the Accounting feature.

1. On the **Applications** menu, click **Accounting**. The **Accounts** screen opens:

User	Jobs	Time	Details	Download
qauser1	2	0.033h		

The Jobs panel shows this information: Users, Job, Time, Details, and Download.

A. To see a user's jobs that ran, click **Details** on the desired user row:

Job	Start Time	Provider	Cluster	Node	Cores	Run Time
mpi-1x8	30 Jun 2021 9:29	otc	otc-moab	nodus-s2-8xlarge-1-4797-0	8	1.067h
random	30 Jun 2021 9:55	otc	otc-moab	nodus-s2-8xlarge-1-4797-0	1	0.017h
Moab.4309	30 Jun 2021 11:57	otc	otc-moab	nodus-s2-8xlarge-1-4797-2	16	0.8h
Moab.4310	30 Jun 2021 11:57	otc	otc-moab	nodus-s2-8xlarge-1-4797-3	24	1.2h
Moab.4311	30 Jun 2021 11:57	otc	otc-moab	nodus-s2-8xlarge-1-4797-1	32	1.067h
Moab.4308	30 Jun 2021 11:57	otc	otc-moab	nodus-s2-8xlarge-1-4797-2	8	1.067h
Moab.4312	30 Jun 2021 11:57	otc	otc-moab	nodus-s2-8xlarge-1-4797-2 nodus-s2-8xlarge-1-4797-3	8 8	2.4h
Moab.4320	30 Jun 2021 12:01	otc	otc-moab	nodus-s2-8xlarge-1-4797-0	8	1.067h
Moab.4321	30 Jun 2021 12:01	otc	otc-moab	nodus-s2-8xlarge-1-4797-0	16	1.067h
Moab.4313	30 Jun 2021 11:59	otc	otc-moab	nodus-s2-8xlarge-1-4797-3 nodus-s2-8xlarge-1-4797-1	16 16	1.067h

This screen shows this information: Job, Start Time, Provider, Cluster, Node, Cores, and Run Time.

2. To see cluster information, click the **Clusters** tab:

Name	User	Deploy	Destroy	Uptime	Details	Download
otc-moab	nodus	3 Aug 2021 17:11	19 Aug 2021 11:22 ...	227.40h		
aws-moab	nodus	10 Aug 2021 13:17	19 Aug 2021 11:22 ...	216.03h		
aws	qauser1	12 Aug 2021 8:44	19 Aug 2021 11:22 ...	11.37h		
oracle	qauser1	19 Aug 2021 11:08	19 Aug 2021 11:22 ...	0.52h		

This screen shows this information: Name, User, Deploy, Destroy, Uptime, Details, and Download.

Bursting

This section contains the following information:

- [Bursting Configurations](#)
- [Bursting Service](#)

Bursting Configurations

The ODDC burst function detects what jobs are in the queue and automatically spins up, takes offline, or shuts down nodes depending on the total requirements for the queue. If there are not enough online nodes to run all jobs, bursting will bring on as many nodes as needed. If there are more nodes than needed, the excess nodes will be taken offline. If the job queue is empty, all nodes will be shut down after a specified period of time.

Min Burst spins up the minimum number of compute nodes required to complete all jobs in the queue, which is ideal for budgeting and controlling cloud costs.

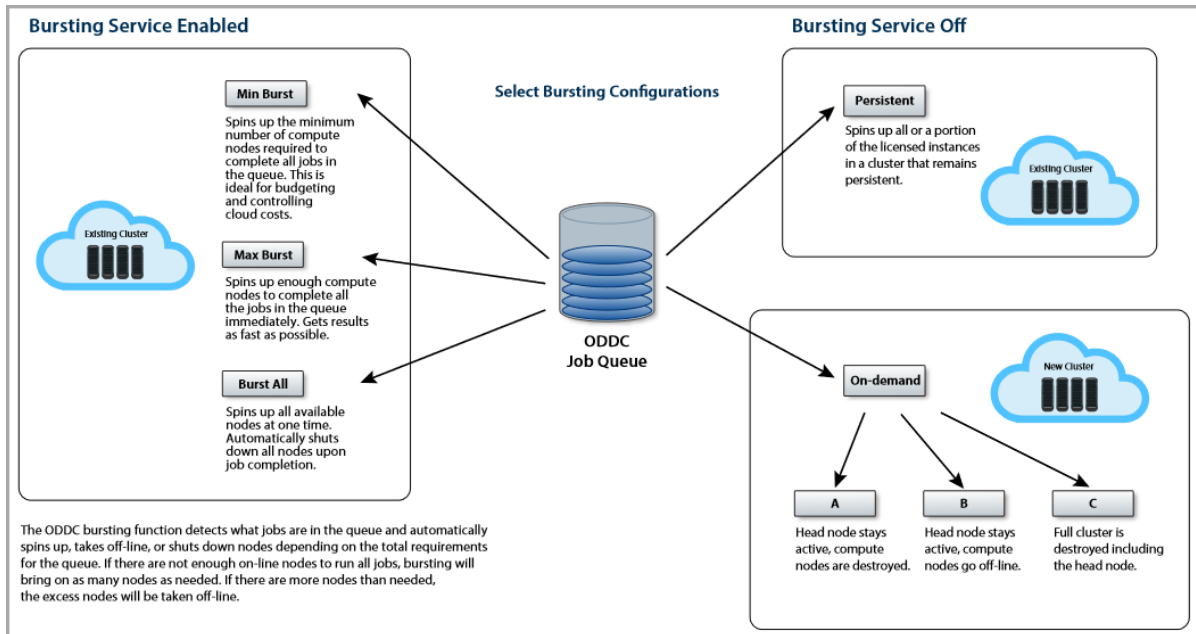
Max Burst spins up enough compute nodes to complete all the jobs in the queue immediately; this gets results as fast as possible. Max Burst is limited by the size of the cluster and will not create new nodes.

Burst All spins up all available nodes at one time and automatically shuts down all nodes upon job completion.

Persistent bursting spins up all or a portion of the licensed instances in a cluster that remain persistent for a period of time and brings nodes online or shuts them down as needed.

On demand bursting spins up the number of nodes required to run one job now; this is an isolated cluster, not for sharing with other jobs. The on demand types are: Destroy Compute Nodes (the head node stays active and the compute nodes are destroyed), Offline Compute Nodes (the head node stays active and the compute nodes go offline), and Destroy Cluster (the full cluster is destroyed including the head node).

See the diagram below for details.



3.9.2 Bursting Service

To manage cluster cloud bursting configurations, on the **Cluster Manager** screen, select a cluster and click the **Configuration** tab. The **Cluster Information** screen displays:

The screenshot shows the "Cluster Information" screen with the following details:

- Cluster ID:** 60f9a4e0c0350861f84f5662
- Head Node Size:** VM.Standard2.2 - vCPU: 2, Mem (GB): 30
- Image Name:** core-server
- Cluster IP:** 158.101.117.40
- SSH Username:** opc
- Created:** July 22, 2021 1:03 PM
- Buttons:** DOWNLOAD CLUSTER SSH KEY, INSTALL OPENVPN

The **Bursting Service** section shows:

- Bursting Service:** Off (toggle switch)
- Mode Service:** Min (selected), Max, All
- Buttons:** VIEW LOGS, BURST ONCE

The **Cluster Compute Nodes** section shows:

- Size:** VM.Standard2.2 - vCPU: 2, Mem (GB): 30
- Count:** 0
- Button:** RESIZE CLUSTER

The **Disks Information** section shows:

- Disks Information:** None
- Button:** MANAGER DISKS

Bursting Service (Off/On) - Disable or enable bursting functionality for a single cluster:

- (disabled) - The cluster remains in its current state.
- (enabled) - Jobs are run and compute nodes are provisioned to handle the jobs:
 - **Min** - Spins up the minimum number of nodes required to complete all jobs in the queue. This is ideal for budgeting and controlling cloud costs.
 - **Max** - Spins up enough nodes (as set from the jobs' node count) to complete all the jobs in the queue immediately. This gets results as fast as possible.

- All - Spins up all available nodes at one time and automatically shuts down all nodes upon job completion.

Burst Once - This spins up or tears down nodes as required to complete all the jobs in the queue. Select a one time bursting size - Min, Max, or All.

Note: For Google, when bursting is selected, you can select the following actions in conjunction with bursting:

- Terminate (default) - Clears the node instances from the cloud service provider.
- Stop - Stops the node instances on the cloud service provider, but they remain configured so they can quickly be restarted.