Pyramid Kubernetes on Azure Guide

Version 1.0

Table of Contents

Overview 3

Instantiating Kubernetes on Azure 3

Create Kubernetes cluster 4

Basics 4

Node pools, Access, Networking, Integrations, Advanced and Tags 5

Connecting to the Cluster 5

Generating the Pyramid YAML 6

Autoscaling the pods: 7

Deploying Pyramid YAML configuration 7

External IP Access for the Pyramid Kubernetes Instance 8

System Initialization 8

Finished 9

Appendix 10

Deploying an MS-SQL database 10

Basics 10

Networking 11

Creating a new Pyramid Repository Database 12

Basics 12

Overview

The following guide is provided to customers to set up and launch a Pyramid Kubernetes cluster on Microsoft Azure Cloud platform.
The guide provides a standard walkthrough but is NOT exhaustive and does not cover every available option.

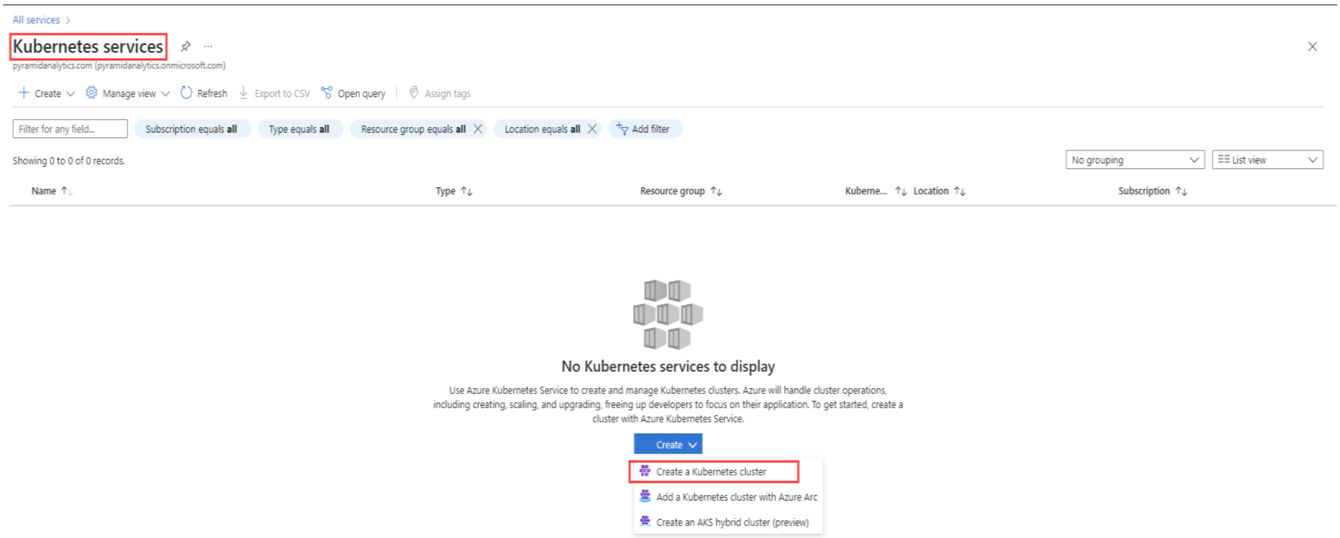
Instantiating Kubernetes on Azure

If you have no prior deployment of Azure Kubernetes engine start here.

Otherwise please start on step 4.

You can either use your existing cluster or choose to deploy one just for Pyramid.

Log into the Azure Admin and search for and then click on “Kubernetes services.”
From the drop down choose “Create a Kubernetes cluster”.



Showing 0 to 0 of 0 records.

No Kubernetes services to display

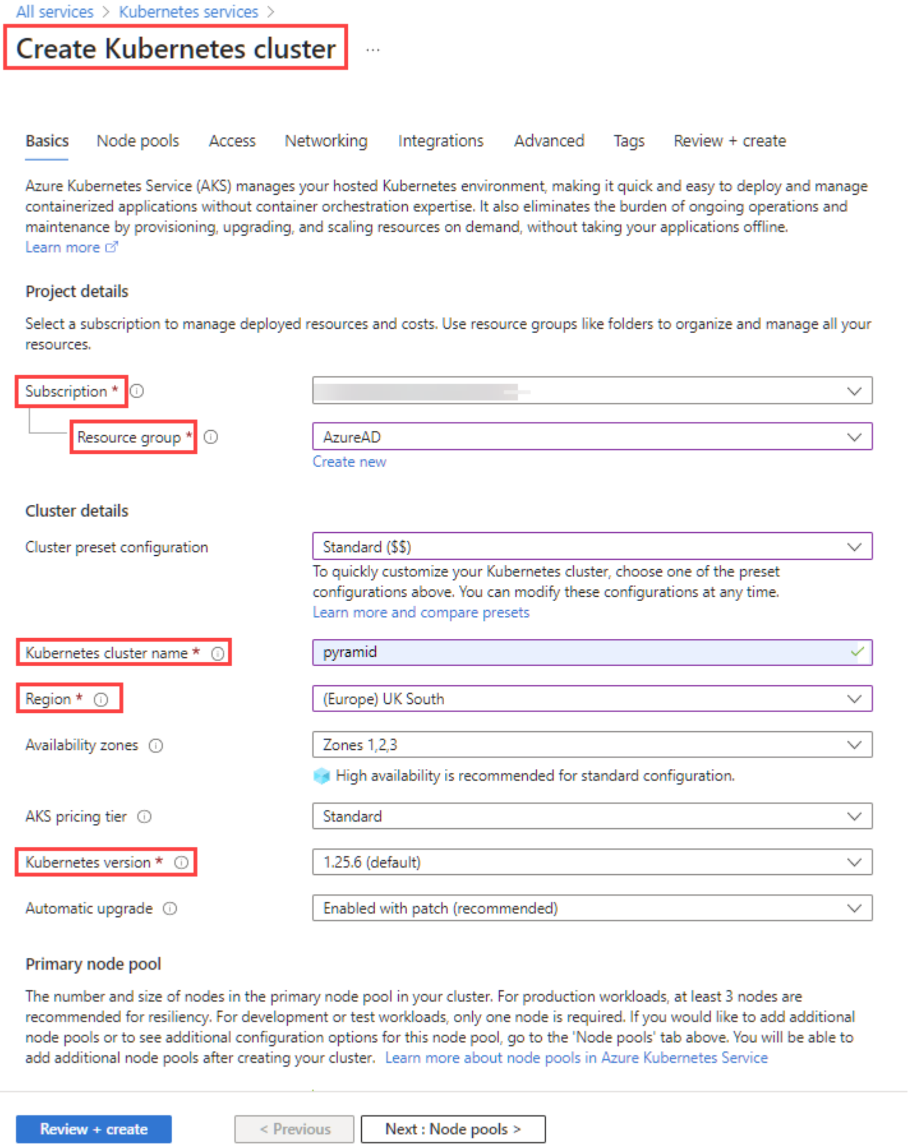
Use Azure Kubernetes Service to create and manage Kubernetes clusters. Azure will handle cluster operations, including creating, scaling, and upgrading, freeing up developers to focus on their application. To get started, create a cluster with Azure Kubernetes Service.

Create Kubernetes cluster

Basics

Settings that are required to be set are:

a. **Subscription and Resource group -** choose your Azure subscription and resource group.
b. **Kubernetes cluster name -** choose a name for your cluster.
c. **Region -** select the region that is best for you.
d. **Kubernetes Version –** This should be left on the default option.
e. **Node size –** depends on expected usage. A suggested start it a machine with 16 cores and 32GB of memory.
f. **Scale method -** Autoscale
g. **Node count range -** depends on expected usage, for production deployments Azure recommended minimum is three nodes.
h. **Review and create -** click on review and create if you do not wish to change any of the other options.



Create Kubernetes cluster ...

Azure Kubernetes Service (AKS) manages your hosted Kubernetes environment, making it quick and easy to deploy and manage containerized applications without container orchestration expertise. It also eliminates the burden of ongoing operations and maintenance by provisioning, upgrading, and scaling resources on demand, without taking your applications offline.
Learn more

Project details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

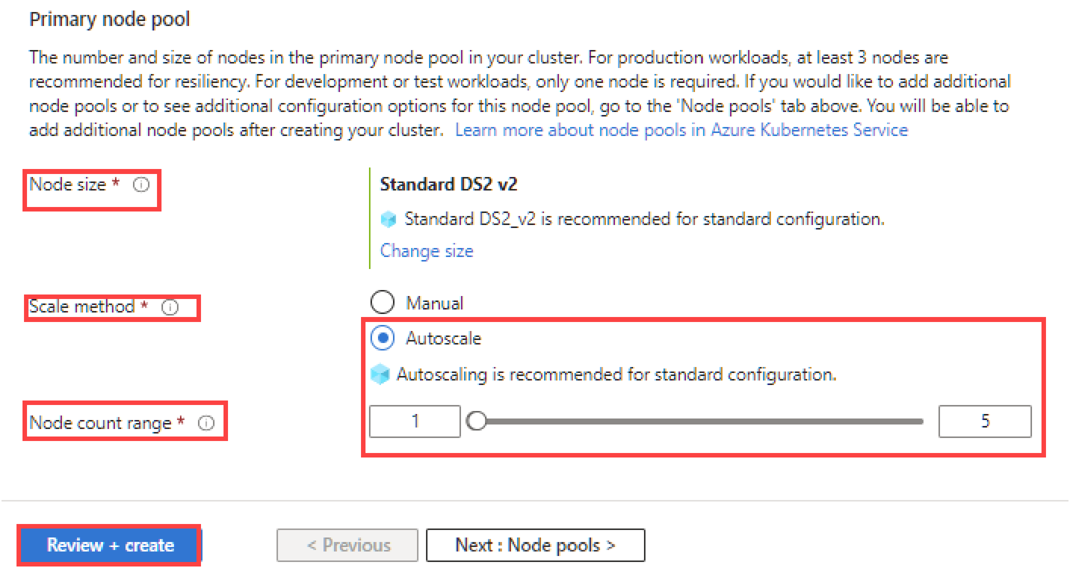
Subscription *	[redacted]
Resource group *	AzureAD Create new

Cluster details

Cluster preset configuration	Standard ($$) To quickly customize your Kubernetes cluster, choose one of the preset configurations above. You can modify these configurations at any time. Learn more and compare presets
Kubernetes cluster name *	pyramid ✓
Region *	(Europe) UK South
Availability zones	Zones 1,2,3 High availability is recommended for standard configuration.
AKS pricing tier	Standard
Kubernetes version *	1.25.6 (default)
Automatic upgrade	Enabled with patch (recommended)

Primary node pool

The number and size of nodes in the primary node pool in your cluster. For production workloads, at least 3 nodes are recommended for resiliency. For development or test workloads, only one node is required. If you would like to add additional node pools or to see additional configuration options for this node pool, go to the 'Node pools' tab above. You will be able to add additional node pools after creating your cluster. Learn more about node pools in Azure Kubernetes Service



Primary node pool

The number and size of nodes in the primary node pool in your cluster. For production workloads, at least 3 nodes are recommended for resiliency. For development or test workloads, only one node is required. If you would like to add additional node pools or to see additional configuration options for this node pool, go to the 'Node pools' tab above. You will be able to add additional node pools after creating your cluster. Learn more about node pools in Azure Kubernetes Service

Node size * : Standard DS2 v2
Standard DS2_v2 is recommended for standard configuration.
Change size

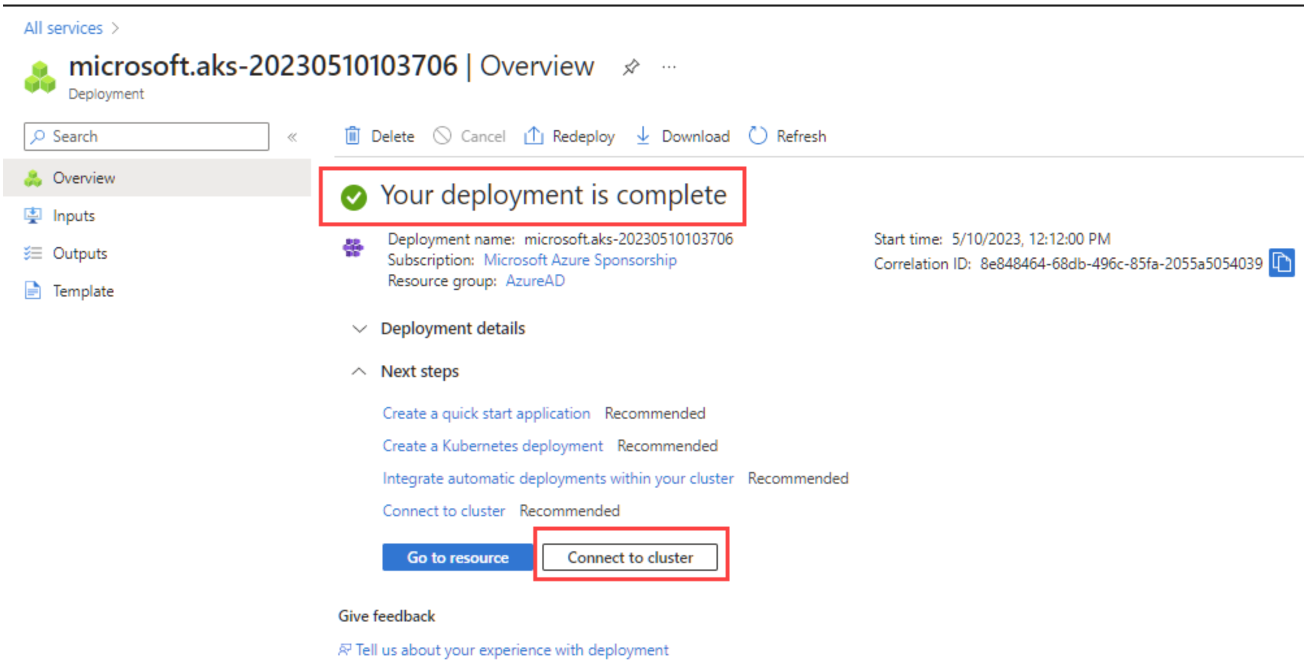
Scale method * : ○ Manual ● Autoscale
Autoscaling is recommended for standard configuration.

Node count range * : 1 – 5

Node pools, Access, Networking, Integrations, Advanced and Tags
All these settings should be customized to confirm with your security and architecture needs. They can all be left as the default option as well.

Connecting to the Cluster

a) Once the cluster has finished being created, click on “connect to cluster” as shown below.



microsoft.aks-20230510103706 | Overview
Deployment

Your deployment is complete

Deployment name: microsoft.aks-20230510103706
Subscription: Microsoft Azure Sponsorship
Resource group: AzureAD

Start time: 5/10/2023, 12:12:00 PM
Correlation ID: 8e848464-68db-496c-85fa-2055a5054039

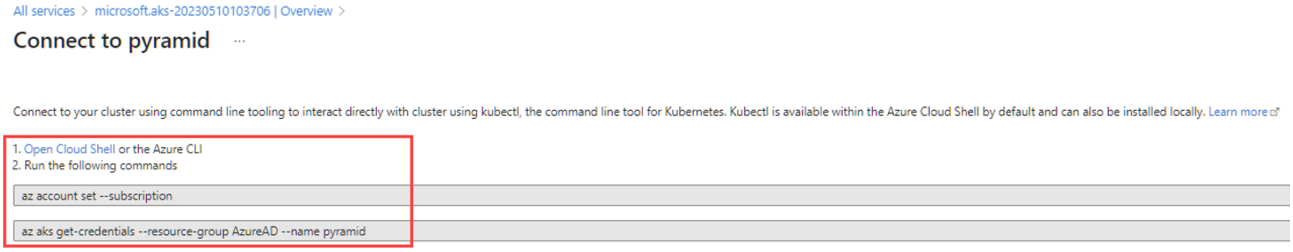
Deployment details

Next steps

Create a quick start application Recommended
Create a Kubernetes deployment Recommended
Integrate automatic deployments within your cluster Recommended
Connect to cluster Recommended

Give feedback
Tell us about your experience with deployment

b) Click on Open in Cloud Shell and **make sure to run the two commands** in step number 2 as they are shown for you.



All services > microsoft.aks-20230510103706 | Overview >

Connect to pyramid ...

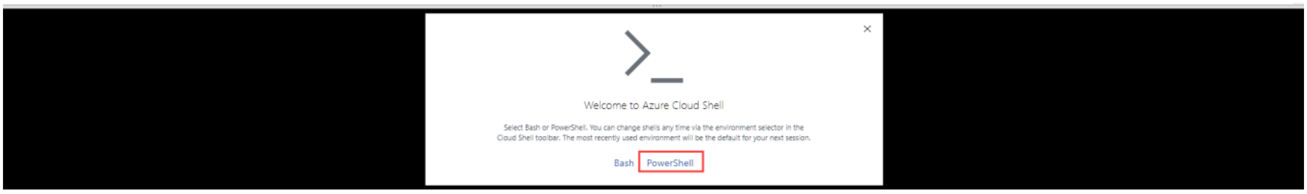
Connect to your cluster using command line tooling to interact directly with cluster using kubectl, the command line tool for Kubernetes. Kubectl is available within the Azure Cloud Shell by default and can also be installed locally. Learn more

1. Open Cloud Shell or the Azure CLI
2. Run the following commands

```
az account set --subscription
```

```
az aks get-credentials --resource-group AzureAD --name pyramid
```

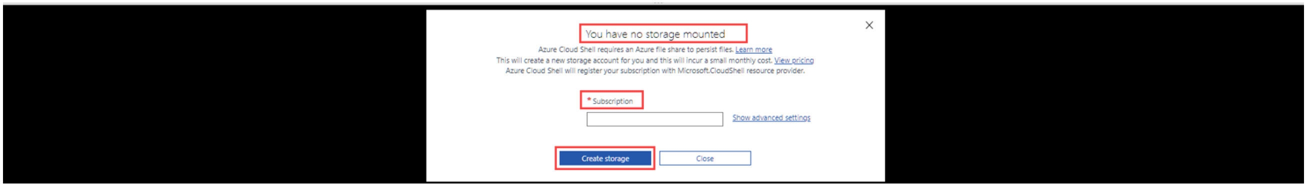
c) Click on either Bash or PowerShell. For this example, we use PowerShell.



Welcome to Azure Cloud Shell

Select Bash or PowerShell. You can change shells any time via the environment selector in the Cloud Shell toolbar. The most recently used environment will be the default for your next session.

d) If you have no storage mounted, it will prompt you to create one as below. Choose your subscription and then on "create storage".



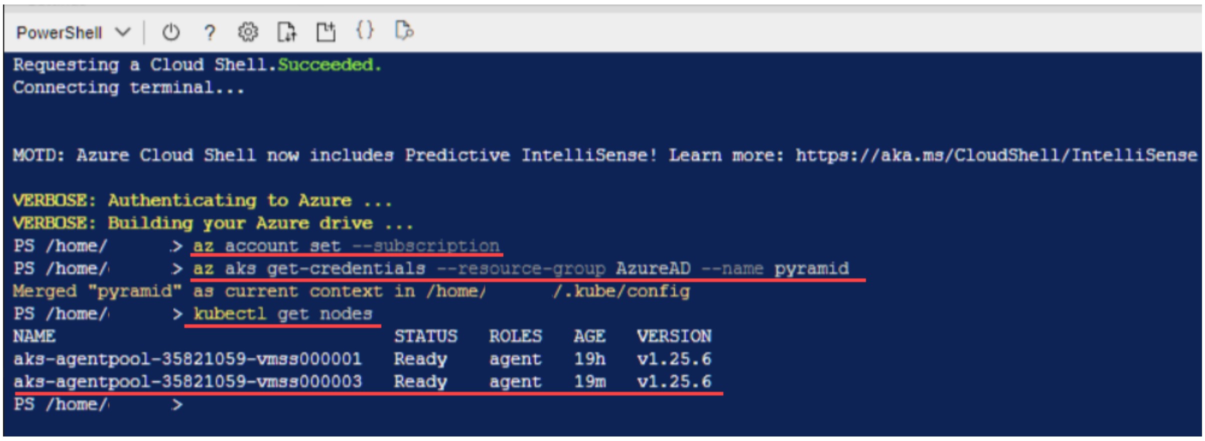
You have no storage mounted

Azure Cloud Shell requires an Azure file share to persist files. Learn more
This will create a new storage account for you and this will incur a small monthly cost. View pricing
Azure Cloud Shell will register your subscription with Microsoft.CloudShell resource provider.

* Subscription

e) After running the set account and getting the credentials command, you can test the connectivity to the cluster, by running the below command, if all is working it will bring back the list of nodes as seen in the screenshot.

```
Kubectl get nodes
```



```
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

MOTD: Azure Cloud Shell now includes Predictive IntelliSense! Learn more: https://aka.ms/CloudShell/IntelliSense

VERBOSE: Authenticating to Azure ...
VERBOSE: Building your Azure drive ...
PS /home/       > az account set --subscription
PS /home/       > az aks get-credentials --resource-group AzureAD --name pyramid
Merged "pyramid" as current context in /home/       /.kube/config
PS /home/       > kubectl get nodes
NAME                                STATUS   ROLES   AGE   VERSION
aks-agentpool-35821059-vmss000001   Ready    agent   19h   v1.25.6
aks-agentpool-35821059-vmss000003   Ready    agent   19m   v1.25.6
PS /home/       >
```

Generating the Pyramid YAML

The setup for Pyramid is *best* driven through a YAML configuration file. If you want, you can also deploy using HELM. For more information on this see this link.
This can be manually created. However, it is simpler to use Pyramid's YAML configurator.

Login to Pyramid's customer portal, go to the Kubernetes setup page: https://customers.pyramidanalytics.com/kubernetes/ and generate a YAML file for your Pyramid config.

Choose one of the storage options in the drop-down list.
More info on the configurator can be found here .

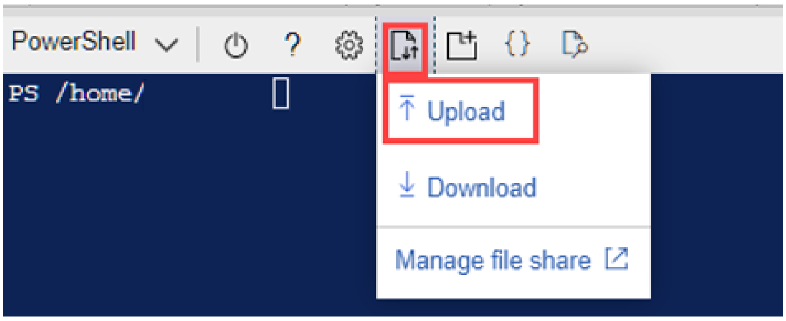
Autoscaling the pods:
Pyramid gives you the option of scaling the pods Horizontally.
You can choose the maximum number of replicas(pods) to spawn by ticking the Elastic Scaling option when creating the Pyramid YAML and entering in the max number of pods that can be spawned.

To enable the auto scaling to work, make sure to Install Keda and the Metric Server, these commands are given to you while creating the Pyramid YAML.

Deploying Pyramid YAML configuration

Upload your YAML file (from previous steps) to your cluster as shown below:

Click on the upload file option and upload the Pyramid YAML file.



```
PS /home/
```

Once you upload the YAML run it as below to pull down the pyramid pods

```
kubectl apply -f pyramid-analytics-config.yaml
```

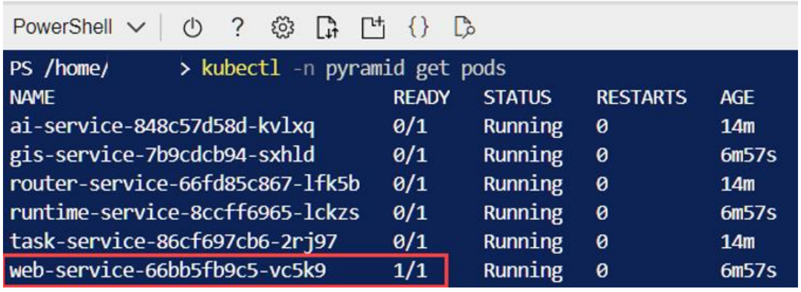
Then run the below command to see the pods generating or look at the Azure control panel under “Workloads” (it will also show the pods as incomplete until after the full deployment has finished)

```
kubectl -n pyramid get pods -w
```

or

```
kubectl -n pyramid get pods
```

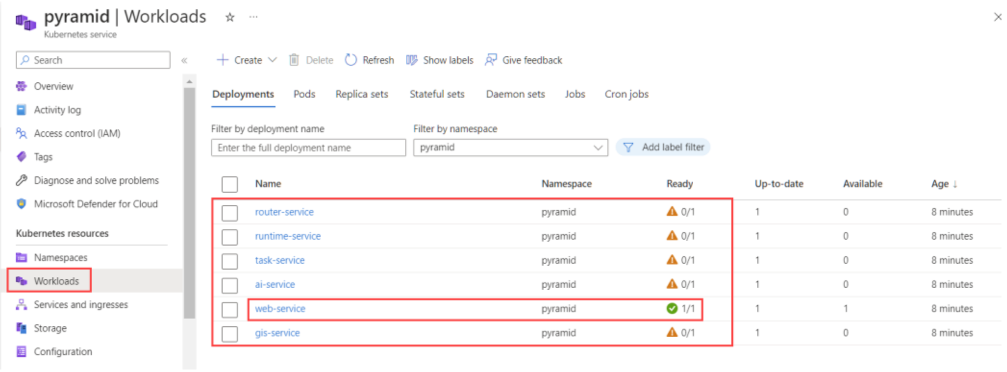
Its normal that only the web-service pod will show 1/1 until the full deployment has finished (until after you have finished the setup in the browser)



```
PS /home/       > kubectl -n pyramid get pods
NAME                                 READY   STATUS    RESTARTS   AGE
ai-service-848c57d58d-kvlxq          0/1     Running   0          14m
gis-service-7b9cdcb94-sxhld          0/1     Running   0          6m57s
router-service-66fd85c867-lfk5b      0/1     Running   0          14m
runtime-service-8ccff6965-lckzs      0/1     Running   0          6m57s
task-service-86cf697cb6-2rj97        0/1     Running   0          14m
web-service-66bb5fb9c5-vc5k9         1/1     Running   0          6m57s
```

Wait until you see that all pods show as “running.” It can take around 10/15 minutes for this to finish.

From the Azure console, it will look as below:

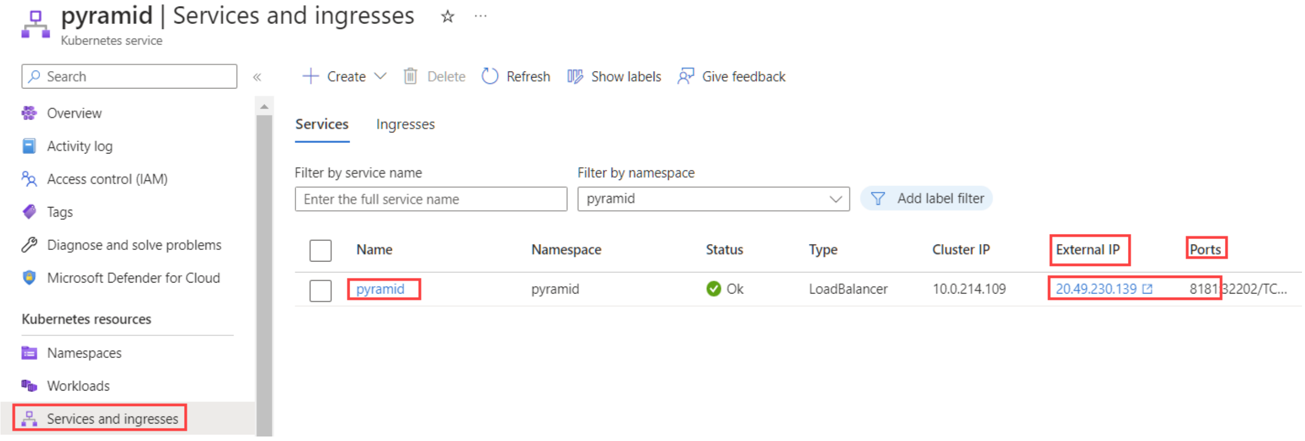


Name	Namespace	Ready	Up-to-date	Available	Age ↓
router-service	pyramid	⚠ 0/1	1	0	8 minutes
runtime-service	pyramid	⚠ 0/1	1	0	8 minutes
task-service	pyramid	⚠ 0/1	1	0	8 minutes
ai-service	pyramid	⚠ 0/1	1	0	8 minutes
web-service	pyramid	✓ 1/1	1	1	8 minutes
gis-service	pyramid	⚠ 0/1	1	0	8 minutes

Once you see that the web-service shows as Ready (Green tick 1/1), continue to the next step.

External IP Access for the Pyramid Kubernetes Instance

To get the external IP to access the Pyramid application on, from the Azure Kubernetes portal click on “Services and ingress”. Look for the external IP address and port which is 8181.



Name	Namespace	Status	Type	Cluster IP	External IP	Ports
pyramid	pyramid	✓ Ok	LoadBalancer	10.0.214.109	20.49.230.139	8181:32202/TC...

Copy the External IP and paste it into a browser adding :8181 on to the end (e.g., http://204.230.139:8181). This will then take you to the below page, where you can fill out all the needed info to finish the Pyramid deployment.

System Initialization

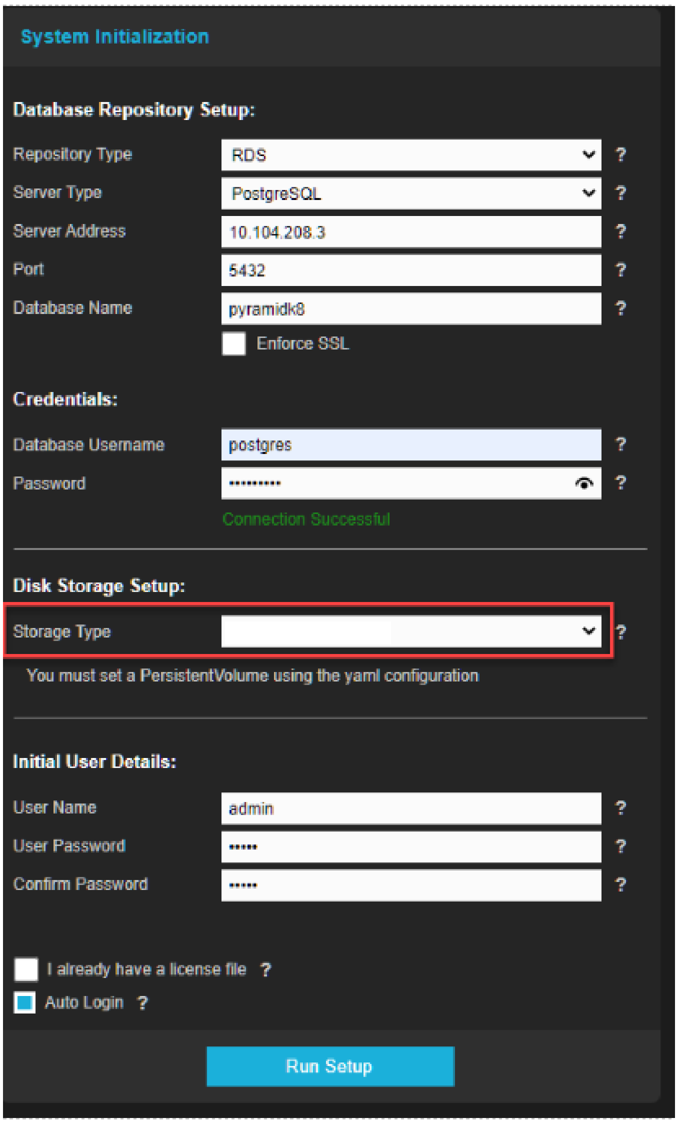
Once the pods have finished being created, and you click on the link as explained above, you will be prompted with the screen below.

This initializes the system, with persistent storage (this is only if you choose this option when creating the YAML. Otherwise choose one of the other options in the list), the Pyramid repository database and creates the first initial user within Pyramid.

For more information on this stage please see this link.

- See the appendix for details on how to setup a database repository on Azure.
- For more information on this stage please see this link.

For the storage type choose the same persistent storage option that was chosen when you generated the Pyramid YAML.



System Initialization

Database Repository Setup:

Repository Type: RDS ?

Server Type: PostgreSQL ?

Server Address: 10.104.208.3 ?

Port: 5432 ?

Database Name: pyramidk8 ?

□ Enforce SSL

Credentials:

Database Username: postgres ?

Password: ••••••••• ?

Connection Successful

Disk Storage Setup:

Storage Type: ?

You must set a PersistentVolume using the yaml configuration

Initial User Details:

User Name: admin ?

User Password: ••••• ?

Confirm Password: ••••• ?

□ I already have a license file ?

☑ Auto Login ?

Finished

Once the initialization setup has finished running (normally around 5-10 mins) it will redirect you to the fully installed Pyramid application.

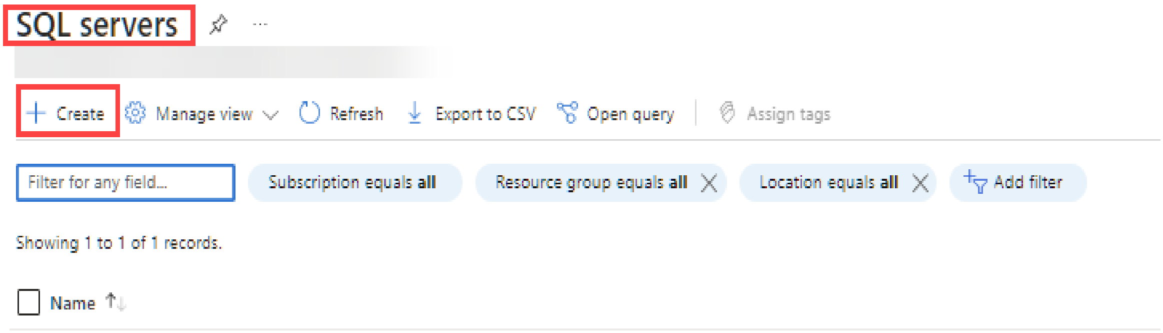
Appendix

Deploying an MS-SQL database

The steps below guide you in the Azure Console for creating an MS-SQL or PostgreSQL instance to host the Pyramid repository.

Notes: it should be a private instance as it does not need to be accessed from outside of your network.
It should be in the same Zone and network as your Kubernetes cluster.

1) Search in the Azure console for "SQL server" and click on "Create"

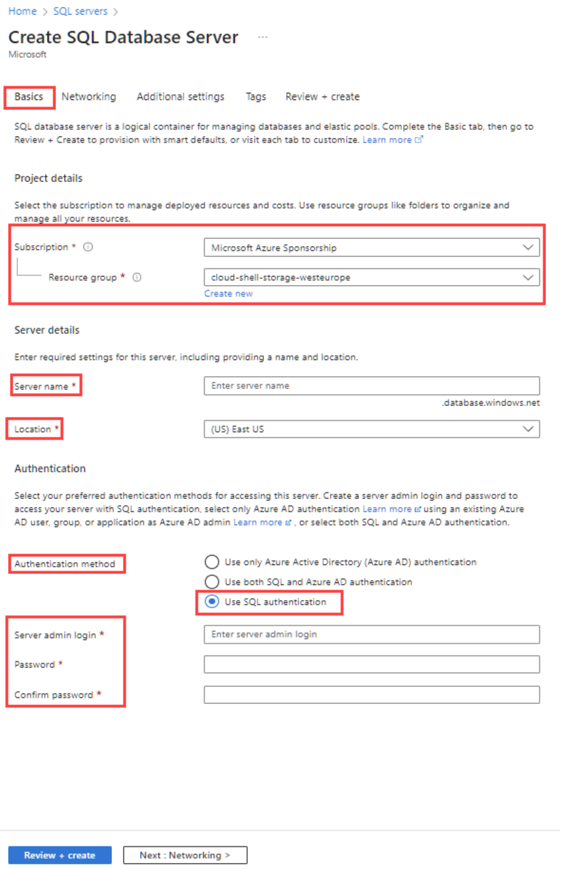


Showing 1 to 1 of 1 records.

Basics

Settings that are required are:

a. **Subscription and Resource group -** choose your Azure subscription and resource group.
b. **Server name -** give the server a name.
c. **Location –** should be in the same location as your Kubernetes cluster.
d. **Authentication method** must include SQL authentication because this is the only way Pyramid can connect to it as a repository.
e. **Server admin login / password/ confirm password –** give the server an admin login and password.



Home > SQL servers >

Create SQL Database Server ···

Microsoft

Basics | Networking | Additional settings | Tags | Review + create

SQL database server is a logical container for managing databases and elastic pools. Complete the Basic tab, then go to Review + Create to provision with smart defaults, or visit each tab to customize. Learn more

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Microsoft Azure Sponsorship
Resource group *	cloud-shell-storage-westeurope Create new

Server details

Enter required settings for this server, including providing a name and location.

Server name *	Enter server name .database.windows.net
Location *	(US) East US

Authentication

Select your preferred authentication methods for accessing this server. Create a server admin login and password to access your server with SQL authentication, select only Azure AD authentication Learn more using an existing Azure AD user, group, or application as Azure AD admin Learn more , or select both SQL and Azure AD authentication.

Authentication method

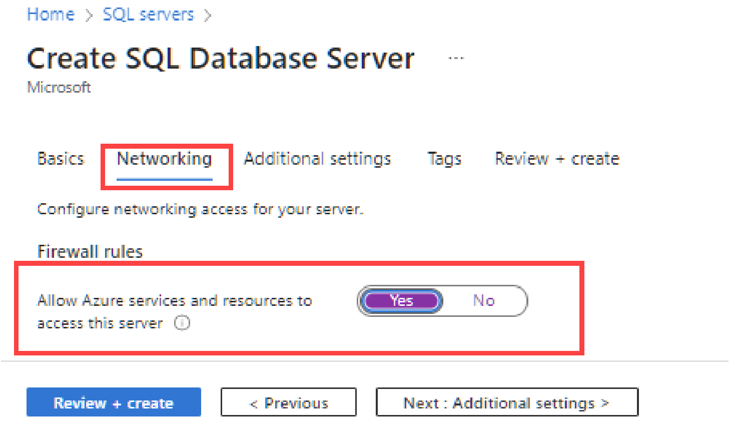
○ Use only Azure Active Directory (Azure AD) authentication
○ Use both SQL and Azure AD authentication
● Use SQL authentication

Server admin login *	Enter server admin login
Password *	
Confirm password *	

Networking

Settings that are required are:

a. **Allow Azure services and resources to access this server –** this should be turned to “yes” so that the Kubernetes pods can access the SQL instances.



Home > SQL servers >

Create SQL Database Server ···

Microsoft

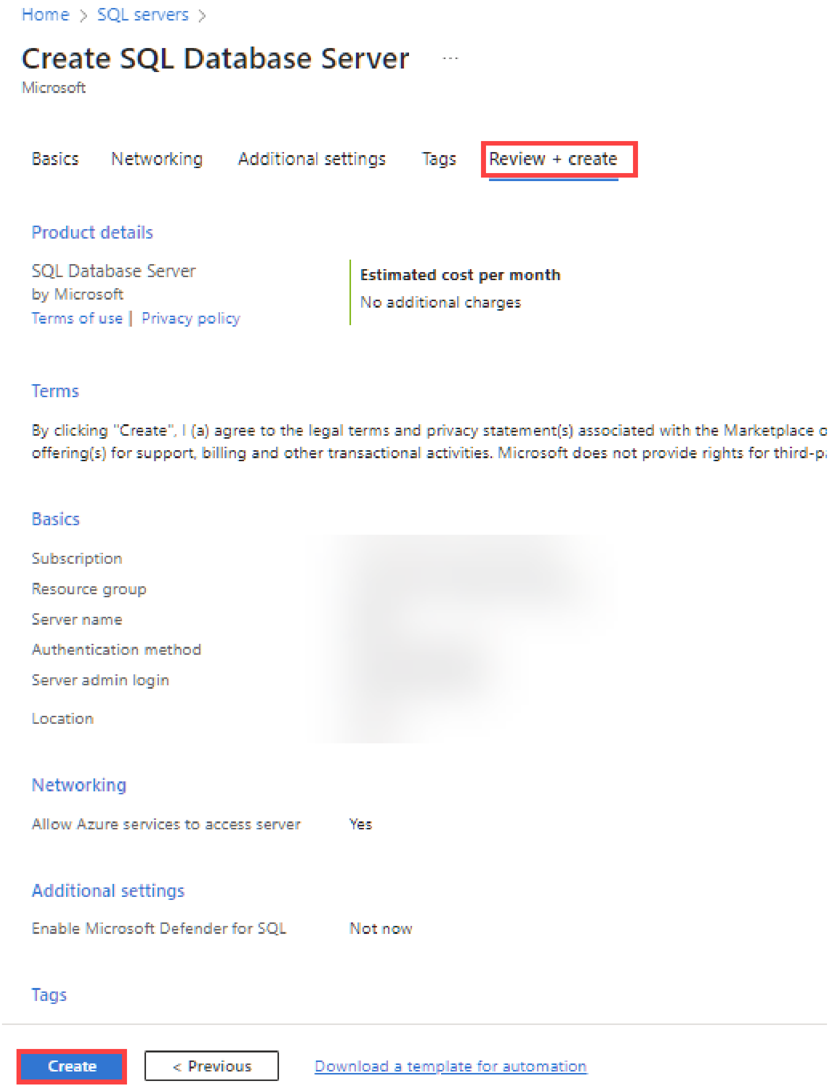
Basics | Networking | Additional settings | Tags | Review + create

Configure networking access for your server.

Firewall rules

Allow Azure services and resources to access this server: [✓ Yes] No

Lastly click on Review + create to create your SQL server as shown below:



Product details

SQL Database Server
by Microsoft
Terms of use | Privacy policy

Estimated cost per month
No additional charges

Terms

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace o offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-p

Basics

Subscription
Resource group
Server name
Authentication method
Server admin login
Location

[redacted]

Networking

Allow Azure services to access server: Yes

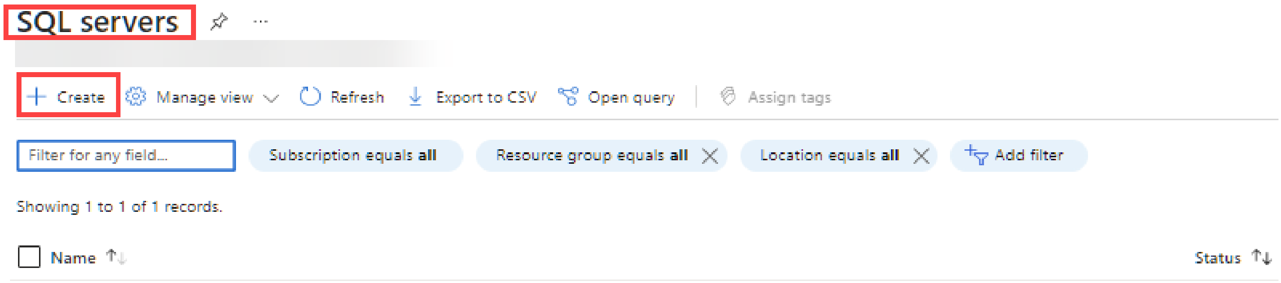
Additional settings

Enable Microsoft Defender for SQL: Not now

Tags

Creating a new Pyramid Repository Database

Once the setup has completed, click on your new SQL instance and create a new blank database. (search for SQL servers and click on the newly created instance).
Click on “Create” as shown in the image below:



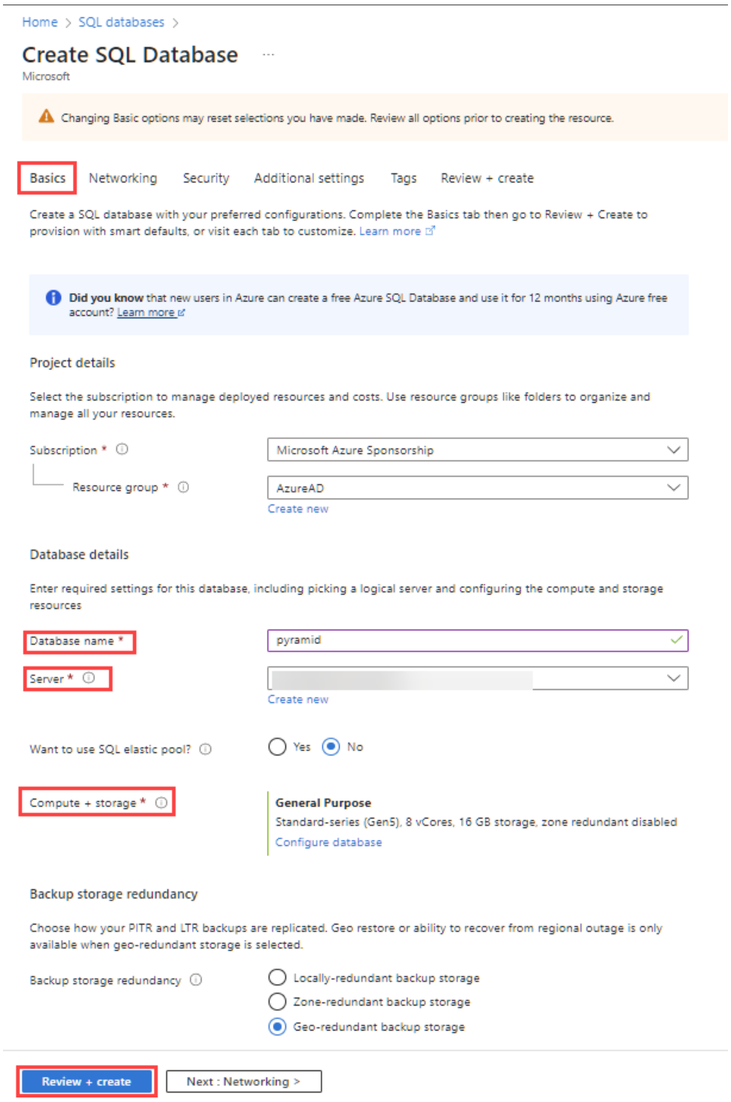
Showing 1 to 1 of 1 records.

Basics

Settings that are required are:

a. **Database name –** give a name for your pyramid repository database.
b. **Server -** select the SQL server that you created or already have.
c. **Compute + storage -** Ensure that the database is not underpowered. It should not be less than 4 CPU’s (8 is the recommended minimum) and 12-16Gb of Memory.

d. **Review + created –** no other settings need to be changed, click on this to create the database.



Home > SQL databases >

Create SQL Database ···

Microsoft

⚠ Changing Basic options may reset selections you have made. Review all options prior to creating the resource.

Create a SQL database with your preferred configurations. Complete the Basics tab then go to Review + Create to provision with smart defaults, or visit each tab to customize. Learn more

Did you know that new users in Azure can create a free Azure SQL Database and use it for 12 months using Azure free account? Learn more

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ: Microsoft Azure Sponsorship

Resource group * ⓘ: AzureAD

Create new

Database details

Enter required settings for this database, including picking a logical server and configuring the compute and storage resources

Database name *: pyramid

Server * ⓘ: [redacted]

Create new

Want to use SQL elastic pool? ⓘ ○ Yes ● No

Compute + storage * ⓘ: **General Purpose**
Standard-series (Gen5), 8 vCores, 16 GB storage, zone redundant disabled
Configure database

Backup storage redundancy

Choose how your PITR and LTR backups are replicated. Geo restore or ability to recover from regional outage is only available when geo-redundant storage is selected.

Backup storage redundancy ⓘ
○ Locally-redundant backup storage
○ Zone-redundant backup storage
● Geo-redundant backup storage