Getting Started

Last updated by | Vitor Tomaz | Nov 16, 2022 at 12:58 PM PST

Contents

- Overview of Instance Pools
- Instance Pools - product characteristics
- Current product limitations
- Getting started with instance pools
- Private preview duration and billing
- Management operations on instance pools and instances c...
- How to create instance pool?
- How to deploy managed instances within instance pool?
- How to update managed instance properties after deploy...
- How to delete instance that belongs to instance pool?
- How to delete instance pool?

Overview of Instance Pools

Instance pools provide a convenient and cost-efficient way to migrate smaller instances to the cloud at scale. Previously, to migrate to the cloud, smaller, less compute-intensive workloads would often have to be consolidated on a larger managed instance. For example, if you plan migrate a fleet of databases running on individual small-size SQL Servers (say 2 vCores), you would need to deploy multiple databases on the same instance (8 vCores, for example).

This typically required careful capacity planning, additional security considerations and some extra data consolidation work at the instance level.

Instances pools bypasses this by pre-provisioning compute according to your total migration requirements (for example 8 vCores), then enabling you to migrate instances up to your pre-provisioned compute level (two 2-vCore and one 4-vCore instances).

Instances within the pool are isolated, eliminating the "noisy neighbor" in cases of multi-tenant SaaS apps, and do not share compute and memory resources with other instances in the pool ensuring predictable performance.

Instance pools represent a new deployment model in which you first deploy pool as compute container and then place multiple instances. Given compute is pre-allocated, all subsequent managed instance operations are fast and predictable (finish in few minutes).

Instance pools are available in the following compute sizes: 8, 16, 24, 32, 40, 64 and 80 vCores. Minimum instance size within the pool is 2 vCores. Only available to **General Purpose** service tier in preview.

In a nutshell, instance pools provide following benefits:

- Ability to create small-size instances
- Fast and predictable instance creation and scaling times (few minutes)
- Minimal IP footprint within subnet (multiple instances deployed on the same VM).

Instance Pools - product characteristics

This section summarizes most important instance pools characteristics that you need to be aware of if you plan instance deployment with the pools.

- Instance pools are available on **Gen5** only.
- Max number of deployed instances is determined by total compute capacity of the pool.

> Compute capacity is determined at pool creation time.
>
> Instances within the pool do not share CPU nor RAM memory, so aggregated number of vCores must be less or equal to number of vCores allocated to the pool.

- All instance level limits apply to instances created within a pool.
- In addition to instance-level limits there are also two limits imposed **at the instance pool level.** During instance provisioning or scaling, or database creation and growth both instance-level and pool-level limits are checked
    - Total storage size per pool: **8TB**.
    - Total number of databases per pool: **100**. This limit can be increased at customer request, for evaluation purposes only.
- License Model (Base Price = AHB or License Included) is set at the pool level. Individual instances comply with pool's license model. If you specify License Model for the instance that is different than in the pool, instance level value will be ignored.
- License Model for the pool can be altered after pool creation.

Current product limitations

Besides product characteristics described above, there are several limitations specific to preview stage of the product:

- **Supported service tiers**: General Purpose only. Business Critical is planned to be added before GA.
- **Azure Portal support**: all operations on instance pools are supported through PowerShell only. Initial instance deployment in a pre-created pool is also supported through PowerShell only.

> Updating and deleting pooled instance is supported through Azure Portal (with few exceptions listed below).

- **Scaling instance pools**: not supported in private preview.
- **Instance movement**: "singleton" managed instances cannot be moved to an existing pool; instances created in a pool cannot be moved to another pool or outside of the parent pool.
- **Typical instance management operations are not supported through Azure Portal.** Use ARM deployment template with **New-AzResourceGroupDeployment** method to change these values:
    - Scaling up/down compute and storage
    - Changing AHB flag ("Save Money")
    - Public endpoint

Getting started with instance pools

Instance pools are now available in private preview. For you to participate, your subscription needs to be Allow_listed in Azure region(s) where you plan to deploy.

Please share subscription Id and Azure region with the product group.

Private preview duration and billing

Private preview has limited duration – up to 3 weeks.

During this period, you will not be charged for using the pools. If you plan to keep pools after this period, you will be charged for the usage at regular SQL MI price for vCores and storage reservation.

Management operations on instance pools and instances created within a pool

In private preview, new management operations (pool creation and instance deployment in a pool) are supported through Azure Resource Manager templates (ARM templates).

ARM templates can be deployed using

- PowerShell
- Azure Portal

To illustrate ARM template usage, we have prepared few examples of ARM template files

1. **CreateInstancePoolWithNewNSGRouteTableVnetSubnetAndMI.json**: ARM template that deploys a new instance pool with its first managed instance with a new virtual network, subnet, network security group, and route table with settings to enable one step provisioning.
2. **CreateManagedInstanceInInstancePool.json**: ARM template that deploys instance within a pre-created instance pool.

To use PowerShell, make sure you install the latest version of PowerShell Core.

Then follow instructions to install Azure PowerShell module: https://docs.microsoft.com/en-us/powershell/azure/install-az-ps?view=azps-2.4.0# install-the-azure-powershell-module-1

How to create instance pool?

To create instance pool, you can either invoke **New-AzResourceGroupDeployment** in PowerShell

```
New-AzResourceGroupDeployment -ResourceGroupName <yourRgName> -TemplateFile
"CreateInstancePoolWithNewNSGRouteTableVnetSubnetAndMI.json"
```

After that, provide required parameters:

location: <fill>Choose the same location that you've registered your subscription for instance pools as above

instancePoolName: <fill>Please choose a name that is fully lowercase (hyphens are accepted)

managedInstanceName: <fill>Please choose a name that is fully lowercase (hyphens are accepted)

instancePoolVCores: <fill>Please choose a vCore from the range <8, 16, 24, 32, 40, 64, 80>

managedInstanceVCores: <fill>Please choose a vCore from the range <2, 4, 8, 16, 24, 32, 40, 64, 80>

managedInstanceStorageSizeInGb: <fill>Please choose a storage size from the range (Gigabytes) <32[U(1] [JPK2] [BN3] -> 8192>*

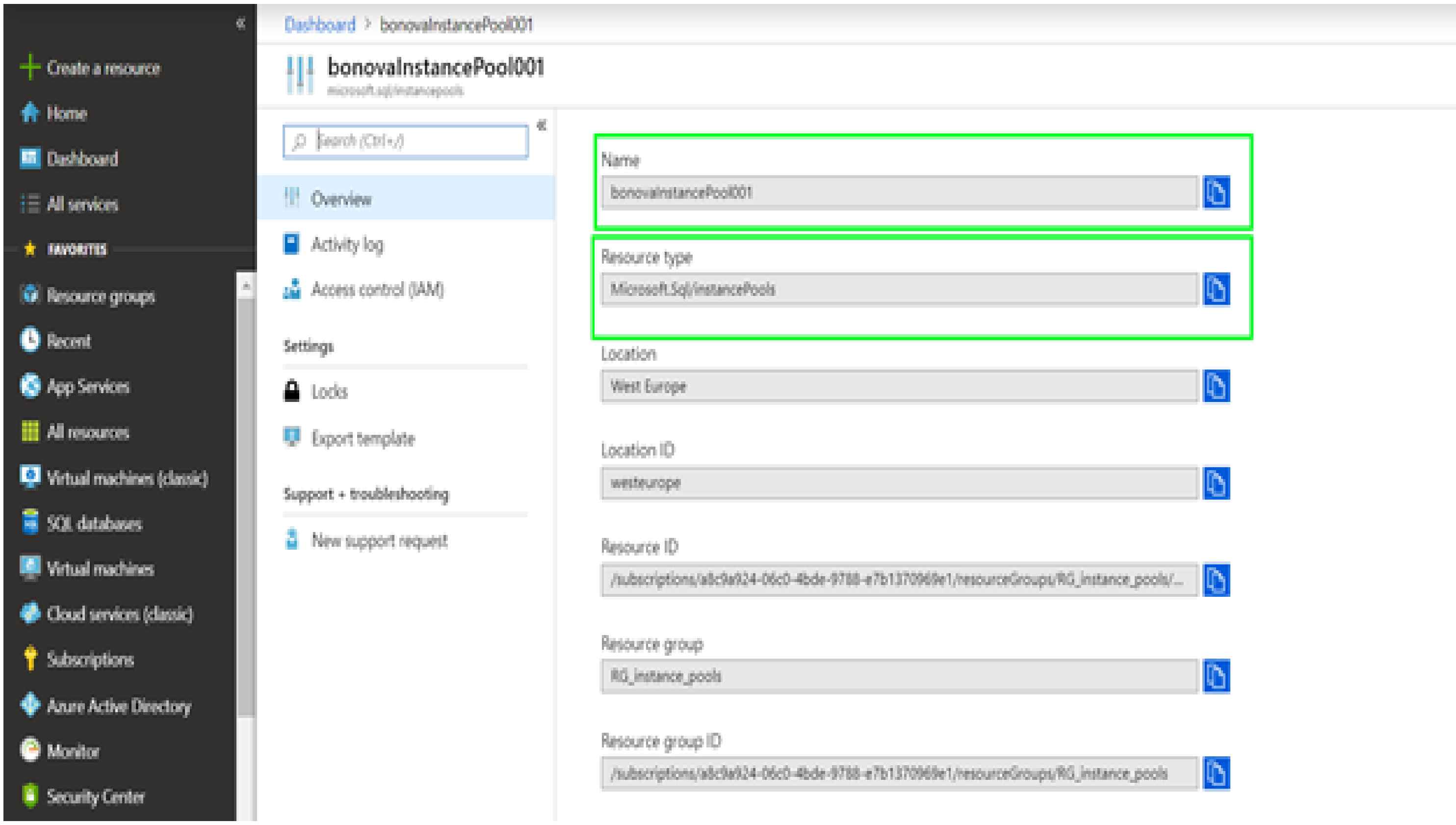
administratorLogin: <fill>

administratorLoginPassword: <fill>

* For allowed max storage sizes for instances between 4 and 80 vCores check out SQL MI Service tier characteristics. Note that 2 vCore instance supports up to ***640 GB.***

Alternatively, you can deploy **CreateInstancePoolWithNewNSGRouteTableVnetSubnetAndMI.json** template from Azure Portal

Once created you can see instance pool as a generic resource on Azure Portal. In private preview specific pool properties such as compute capacity, list of deployed instances, etc. are not available



bonovaInstancePool001
microsoft.sql/instancepools

Name: bonovainstancePool001

Resource type: Microsoft.Sql/instancePools

Location: West Europe

Location ID: westeurope

Resource ID: /subscriptions/a8c9a924-06c0-4bde-9788-e7b1370969e1/resourceGroups/RG_instance_pools/...

Resource group: RG_instance_pools

Resource group ID: /subscriptions/a8c9a924-06c0-4bde-9788-e7b1370969e1/resourceGroups/RG_instance_pools

Alternatively, you can get instance pool properties through PowerShell

```
Get-AzResource -Name <instance pool
name>
```

Method result (example):

```
Name              :
bonovaInstancePool001
ResourceGroupName :
RG_instance_pools
ResourceType      :
Microsoft.Sql/instancePools
Location          :
westeurope
ResourceId        : /subscriptions/Input
SubscriptionId*/resourceGroups/RG_instance_pools/providers/Microsoft.Sql/instancePools/bonovaInstancePool0
01
```

How to deploy managed instances within instance pool?

Again, PowerShell can be used

```
New-AzResourceGroupDeployment -ResourceGroupName < yourRgName > -TemplateFile
"CreateManagedInstanceInInstancePool.json"
```

Make sure you provide require parameters for instance creation

administratorLogin: <fill>

administratorLoginPassword: <fill>

location: <fill> Must be same as for instance pool

managedInstanceName: <fill>

storageSizeInGB: <fill> Reserved storage for new instance

vCores: <fill> instance compute size with the pool. Must be less or equal to the remaining vCore capacity in the pool

collation: <fill> example: Serbian_Latin_100_CI_AS

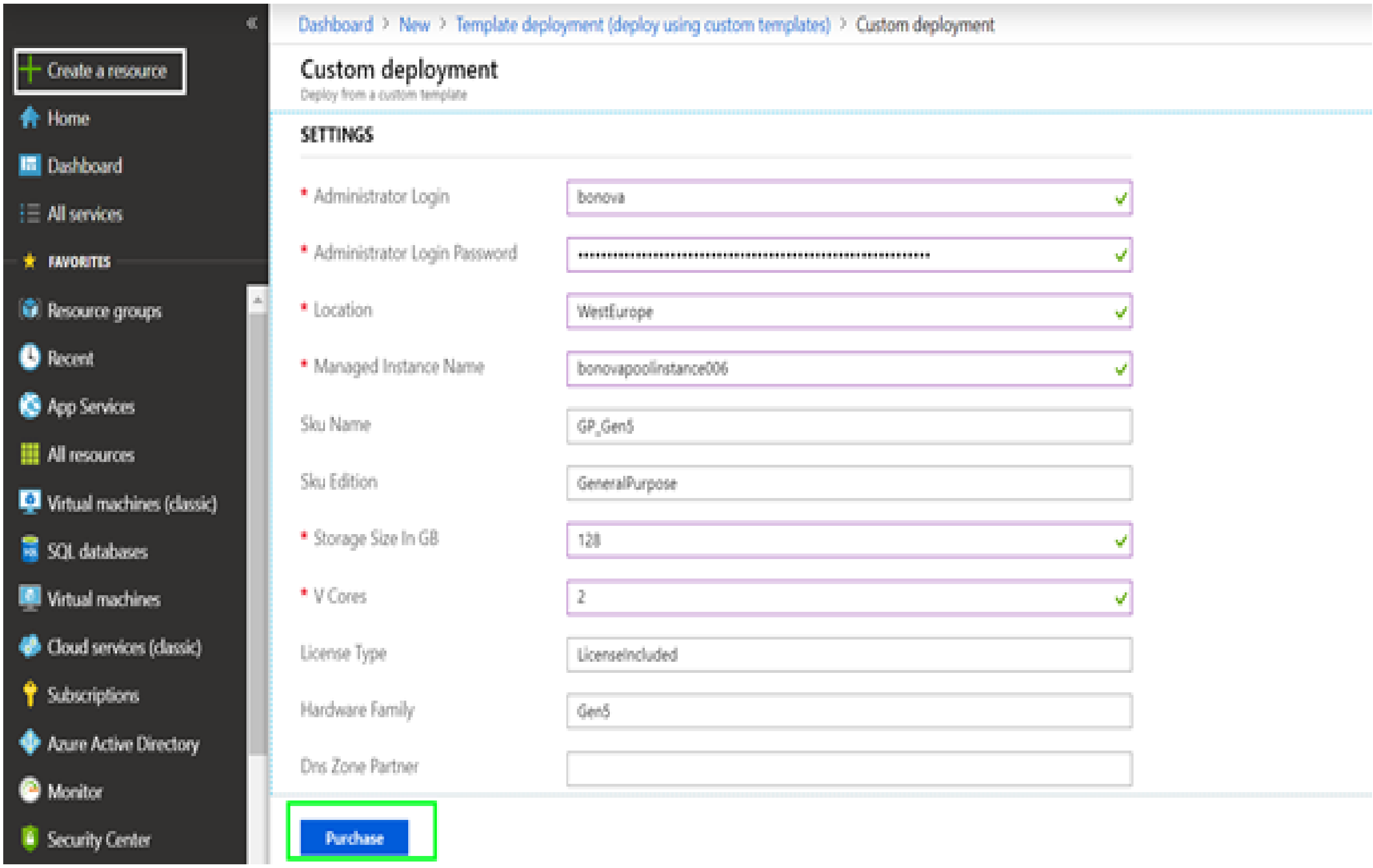
timezoneId: <fill> example: Central European Standard Time

instancePoolName: <fill> **this is the reference to the pool name created in previous step**

publicDataEndpointEnabled: <fill> true / false[U(4] [BN5] [JPK6] [JPK7] *[BN8] [JPK9] [JPK10]

*If you plan to open public endpoint for data traffic, make sure you add inbound rule to the subnet NSG that opens port **3342**

If you deploy using Azure Portal, make sure you populate ARM template form:



Dashboard > New > Template deployment (deploy using custom templates) > Custom deployment

Custom deployment

Deploy from a custom template

SETTINGS

* Administrator Login	bonova
* Administrator Login Password	••
* Location	WestEurope
* Managed Instance Name	bonovapoolinstance006
Sku Name	GP_Gen5
Sku Edition	GeneralPurpose
* Storage Size In GB	128
* V Cores	2
License Type	LicenseIncluded
Hardware Family	Gen5
Dns Zone Partner	

When instance is deployed it becomes visible on Azure Portal (as any other SQL Managed Instance)

In private preview, Azure Portal blade for instances does not provide any reference to the parent instance pool.

How to update managed instance properties after deployment?

Options that can be changed regularly at Pricing tier blade of Azure Portal for singleton instances (created outside of pools) do not work for instances created in pools during preview

Instead, use same method that you used for instance creation (ARM template deployment) and override parameters that you want to change.

How to delete instance that belongs to instance pool?

If you want to remove instance, simply use Power Shell

```
Remove-AzSqlInstance -ResourceGroupName <your RG name>  -Name <your instance name>
```

Alternatively, you can delete instance from Azure Portal.

How to delete instance pool?

Instance pool can be deleted only if it does not contain any instance deployed

In private preview, instance pools can be deleted only from PowerShell

```
Remove-AzResource -ResourceId  <instance pool resource id>
```

To get instance pool resource id for your pool, check pool at Azure Portal or execute **Get-AzResource** -Name <instance pool name> first.

How good have you found this content?

☺ ☹