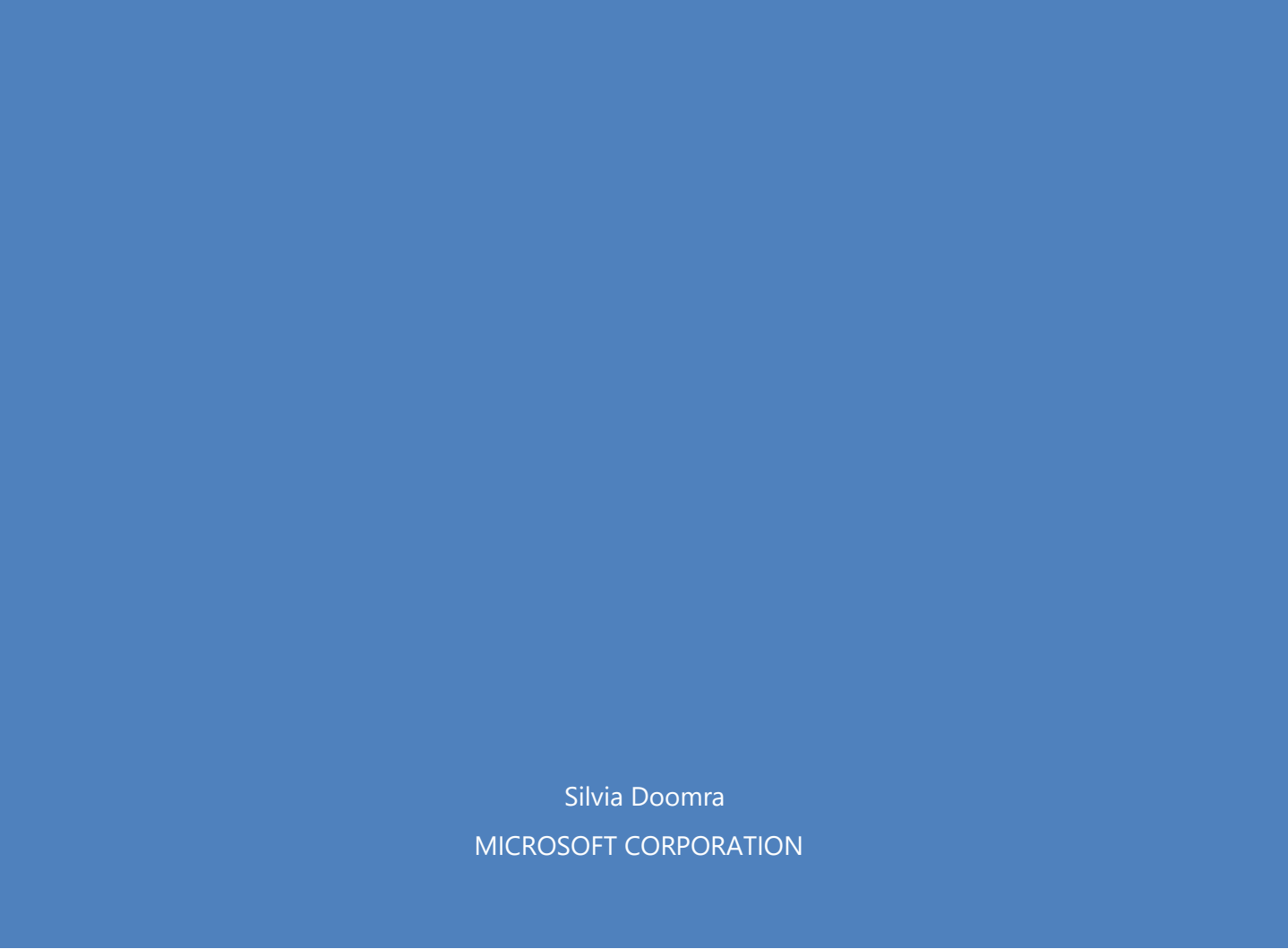




Azure Cross-Region Migration Playbook



Silvia Doomra
MICROSOFT CORPORATION

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

(c) 2019 Microsoft Corporation. All rights reserved. This document is provided "as-is." Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

Table of Contents

Migration process.....	5
Assess	5
Plan.....	6
Migrate	6
Validate	6
Terms	7
Migrate compute resources.....	7
Compute IaaS	7
Cloud Services	10
Service Fabric.....	11
Batch	12
Virtual machine scale sets.....	12
Migrate network resources	13
Virtual networks.....	13
Network security groups	14
ExpressRoute	14
VPN Gateway	15
Application Gateway	15
DNS	15
Network Watcher	16
Traffic Manager.....	17
Load Balancer	18
Migrate storage resources	18
Blobs	18
Managed Disks	19
Import/Export.....	21
StorSimple	22
Migrate web resources	22
Web Apps	22
API Management	23

Migrate database resources	24
SQL Database	24
SQL Server Stretch Database	24
SQL Data Warehouse.....	24
Azure Cache for Redis	24
Migrate analytics resources	26
Event Hubs.....	26
Event Hubs metadata	27
Analysis Services.....	27
Migrate IoT resources to the target Azure region.....	27
Functions.....	28
Notification Hubs.....	28
IoT Hub	28
Migrate integration resources to the target Azure region.....	30
Service Bus.....	30
Logic Apps.....	31
Migrate identity resources	31
Multi-Factor Authentication.....	31
Migrate security resources	32
Key Vault	32
Migrate management tool resources.....	33
Backup.....	33
Scheduler	33
Site Recovery	33
Migrate media resources	34
Media Services	34
Media Player	34

Migration process

This document provides guidance that you may find helpful as you migrate your workloads from one Azure region to another.

While planning the migration of a workload, the focus should be on the **application** as a migration entity. Once the migration planning is complete, the required Azure resources can then be migrated one at a time.

The steps in the migration process are as follows:



Assess

It's important to understand your organization's footprint you're looking to migrate by bringing together Azure account owners, subscription admins, tenant admins, and finance and accounting teams. The people who work in these roles can provide a complete picture of Azure usage for a large organization.

In the assessment stage, compile an inventory of resources:

- Each subscription admin and tenant admin should run a series of scripts to list resource groups, the resources in each resource group, and resource group deployment settings in the environment.
- Dependencies across applications in Azure and with external systems should be documented.
- The count of each Azure resource and the amount of data that are associated with each instance you want to migrate should be documented.
- Ensure that application architecture documents are consistent with the Azure resources list.

At the end of this stage, you'll have:

- A complete list of Azure resources that need to be migrated.
- A list of dependencies between resources.
- Information about the complexity of the migration effort.

Plan

In the planning stage, you should complete the following tasks:

- Use the output of the dependency analysis completed in the assessment stage to define related components. Consider migrating related components together in a *migration package*.
- (Optional) Use the migration as an opportunity to apply [Gartner 5-R criteria](#) and to optimize your workload.
- Determine the target environment in the target Azure region.
- Identify the target Azure tenant (create one, if necessary).
- Create subscriptions.
- Choose the target Azure region.
- Execute test migration scenarios that match your architecture in the source Azure region with the architecture in the target region.
- Determine the appropriate timeline and schedule for migration. Create a user acceptance test plan for each migration package.

Migrate

In the migration phase, use the tools, techniques, and recommendations discussed in the following sections to migrate or create new resources in the target region. Then, configure applications.

Validate

In the validation stage, complete the following tasks:

1. Complete user acceptance testing.
2. Ensure that applications are working as expected.
3. Sync the latest data to the target environment, if applicable.
4. Switch to a new application instance in the target region.
5. Verify that the production environment is working as expected.
6. Decommission resources in the source region.

Terms

These terms are used in the following sections:

Source describes where you're migrating resources from:

- **Source tenant name:** The name of the tenant in the source Azure region (everything after @ in the account name).
- **Source tenant ID:** The ID of the tenant in the source Azure region. The tenant ID appears in the Azure portal when you move the mouse over the account name in the upper-right corner.
- **Source subscription ID:** The ID of the resource subscription in the source Azure region. You can have more than one subscription in the same tenant. Always make sure that you're using the correct subscription.
- **Source region:** The region where the resource you want to migrate is located.

Target or **destination** describes where you're migrating resources to:

- **Target tenant name:** The name of the tenant in the target Azure region.
- **Target tenant ID:** The ID of the tenant in the target region.
- **Target subscription ID:** The subscription ID for the resource in the target region.
- **Target region:** The region where you want to migrate your resource to.

NOTE

Verify that the Azure service you're migrating is offered in the target region.

Migrate compute resources

This section provides information that you may find helpful as you migrate your Azure compute resources from one Azure region to another.

Compute IaaS

Azure VMs can be migrated from one Azure region to another using [Azure Site Recovery](#). You first should verify the [source and the target region combination](#) is supported. For instructions on how to migrate a VM from one region to another, see [Move Azure VMs to another region](#). However, if you're looking to migrate the VM outside of the geographic cluster, you can use any of the following methods:

Azure Site Recovery

To use Azure Site Recovery for non-geographic pairs, you'll need to set up a Site Recovery vault in the target environment and to continue like you're moving a physical server to Azure. In the Azure portal, select a replication path labeled **Not virtualized**. When the replication is finished, do a failover.

NOTE

The following steps are the same steps you would take to migrate a physical server that's running on-premises to Azure.

To learn more about this process, review the [Set up disaster recovery to Azure for on-premises physical servers](#) tutorial. The following list shows a short and slightly adapted version of the process:

Install a "configuration and process server" in your source environment. This will act as a replication gateway to transfer the VM disks to another Azure region. Then, replicate the VMs to the Azure Recovery Services vault in your target region. The work is all done by the "configuration and process server". You don't need to touch the individual servers.

1. Sign in to the Azure portal for the source region.
2. Set up a new VM in your source Azure Virtual Network instance to act as the configuration server:
 1. Select DS4v3 or higher (4 to 8 cores, 16-GB memory).
 2. Attach an additional disk that has at least 1 TB of available space (for the VM images).
 3. Use Windows Server 2012 R2 or later.
3. Set up a virtual network in which the migrated VMs will run.
4. Create an Azure Storage account.
5. Set up the Recovery Services vault.
6. Define the **Protection goal (To Azure > Not virtualized/other)**.
7. Download the **Recovery Unified Setup** installation file (**Prepare Infrastructure > Source**). When you open the portal URL from within ConfigurationServer, the file is downloaded to the correct server. From outside the ConfigurationServer, upload the installation file to ConfigurationServer.

8. Download the vault registration key and upload it to the ConfigurationServer as seen in the preceding step, if necessary.
9. Run the Recovery Unified Setup installation on the ConfigurationServer.
10. Set up the target environment. Make sure you're still signed-in to the target portal.
11. Define the replication policy.
12. Start replication.

After replication initially succeeds, test the scenario by doing a test failover. Verify and delete the test. Your final step is to do the real failover.

CAUTION

Syncing back to the source VM doesn't occur. If you want to migrate again, clean up everything and start again at the beginning.

Duplicate by using Resource Manager template export/import

You can export the Azure Resource Manager template that you use to deploy to your local machine. Edit the template to change the location and other parameters or variables. Then, redeploy in the target Azure region.

Export the Resource Manager template in the portal by selecting the resource group. Select **deployments**, and then select the most recent deployment. Select **Template** in the left menu and download the template.

This process downloads a .zip file that has several files in it. The PowerShell, Azure CLI, Ruby, or .NET script tabs contain code that help you deploy your template. The file *parameters.json* has all the input from the last deployment. It's likely that you'll need to change some settings in this file. Edit the *template.json* file if you want to redeploy only a subset of the resources.

For more information:

- Refresh your knowledge by completing the [Site Recovery tutorials](#).
- Get information about how to [export Resource Manager templates](#) or read an overview of [Azure Resource Manager](#).
- Learn more about [Move Azure VMs to another region](#).
- Read the [overview of Azure locations](#).
- Learn more about how to [redeploy a template](#).

Cloud Services

Cloud Services can't be migrated from one Azure region to another. You can redeploy Azure Cloud Services resources by providing the .cspkg and .cscfg definitions again.

Azure portal

To redeploy cloud services in the Azure portal:

1. [Create a new cloud service](#) by using your .cspkg and .cscfg definitions.
2. Update the [CNAME or A record](#) to point traffic to the new cloud service.
3. When traffic points to the new cloud service, delete the old cloud service in the source region.

PowerShell

To redeploy cloud services by using PowerShell:

1. [Create a new cloud service](#) by using your .cspkg and .cscfg definitions.

```
New-AzureService -ServiceName <yourServiceName> -Label <MyTestService> -  
Location <westeurope>
```

2. [Create a new deployment](#) by using your .cspkg and .cscfg definitions.

```
New-AzureDeployment -ServiceName <yourServiceName> -Slot <Production> -  
Package <YourCspkgFile.cspkg> -Configuration <YourConfigFile.cscfg>
```

3. Update the [CNAME or A record](#) to point traffic to the new cloud service.
4. When traffic points to the new cloud service, [delete the old cloud service](#) in the source Azure region.

```
Remove-AzureService -ServiceName <yourOldServiceName>
```

REST API

To redeploy cloud services by using the REST API:

1. [Create a new cloud service](#) in the target environment.

```
https://management.core.windows.net/<subscription-  
id>/services/hostedservices
```

2. Create a new deployment by using the [Create Deployment API](#). To find your .cspkg and .cscfg definitions, you can call the [Get Package API](#).

```
https://management.core.windows.net/<subscription-id>/services/hostedservices/<cloudservice-name>/deploymentslots/production
```

3. When traffic points to the new cloud service, [delete the old cloud service](#) in the source Azure region.

```
https://management.core.windows.net/<subscription-id>/services/hostedservices/<old-cloudservice-name>
```

For more information:

- Review the [Cloud Services overview](#).

Service Fabric

To migrate Azure Service Fabric resources from one Azure region to another, you need to create and redeploy the Service Fabric cluster and application resources in the new region. Data must be backed up and restored from old to new cluster for stateful services.

1. Create a new cluster in a resource group in the target Azure region using the [Azure portal](#) or Resource Manager resources using guidelines recommended in the [Production readiness checklist](#).
2. No new calls should show up and no services should talk to each other or do work.
3. If it's a **stateful service**, then there's a requirement to move data from the old cluster to the new cluster. See, [Backup data from the old cluster](#) to learn how to move data from the old cluster to the new one.
4. If it's a **stateless service**, then there's no requirement to move data but you must move the traffic.
5. Update the application configuration to be deployed to a the new region.
6. [Deploy an application to the new region using Resource Manager resources](#)
7. For **stateful services**, you'll need to [restore the data](#) using a backup taken in step #3 above.
8. Update the traffic manager service (**Azure Traffic Manager**) to route active traffic to the target region. Gradually move the application traffic to the new region.
9. Verify the application is deployed to the target Azure region and accepting traffic.
10. Validate there's no traffic flowing to the source Azure region.

11. [Delete the cluster by cleaning resource group associated with the cluster in the source Azure region.](#)

References:

- [Create an Azure windows Service Fabric cluster](#)
- [Create Linux Azure Linux Service fabric cluster](#)
- [Deploy application using PowerShell](#)

For more information:

- Refresh your knowledge by completing the [Service Fabric tutorials](#).
- Learn how to [create a new cluster](#).
- Review the [Service Fabric overview](#).

Batch

You can't automatically migrate your Azure Batch account and data from one region to another. Instead, to migrate you'll need to complete the following steps:

1. [Create a Batch account](#) in the target region. Make sure there are storage accounts in the target region(s) or create them there.
2. Deploy your workloads to the new Batch account and start running the jobs there.

For more information:

- Refresh your knowledge by completing the [Batch tutorials](#).
- Review the [Azure Batch overview](#).

Virtual machine scale sets

To migrate virtual machine scale sets across Azure regions, export the Resource Manager template, adapt it to the new environment, and then redeploy to the target region. Export only the base template and redeploy the template in the new environment. Individual virtual machine scale set instances should all be the same. Before starting the redeployment, ensure dependencies on other resources are understood and migrated to the target region.

IMPORTANT

Change location, Key Vault secrets, certificates, and other GUIDs to be consistent with the new region.

For more information:

- Refresh your knowledge by completing the [virtual machine scale set tutorials](#).
- Learn how to [export Azure Resource Manager templates](#).
- Review the [Azure Resource Manager overview](#).
- Get an overview of [virtual machine scale sets](#).
- Read an [overview of Azure locations](#).
- Learn how to [redeploy a template](#).

Migrate network resources

Most networking services don't support migration across Azure regions. However, you can connect your networks in two cloud environments with [Global VNet Peering](#). Global VNet peering enables you to connect across regions privately using the Microsoft Backbone. Once peered, the virtual networks appear as one for connectivity purposes. The steps you take to set up VNet peering across regions are listed below. Once your virtual networks created, all you need to do is peer them.

NOTE

VNet Peering will only work when connecting the same cloud environment types. If you're connecting different cloud environment sites, such as sovereign and public, then use [VPN Gateway](#).

Here are the summary of the steps needed to create a peering:

1. Create a Virtual Network in the target region.
2. Create a peering link from the new virtual network that is in the target region to the virtual network in the source region.
3. Create a peering link from the virtual network in the source region to the new virtual network created in the target region.

Virtual networks

Migrating virtual networks across Azure regions isn't supported at this time. We recommend that you create new virtual networks in the target region and migrate resources into those virtual networks.

For more information:

- Refresh your knowledge by completing the [Azure Virtual Network tutorials](#).

- Review the [virtual networks overview](#).
- Learn how to [plan virtual networks](#).
- Learn how to [create a VNet peer](#).
- [How to create VNet peering with different deployment models and subscriptions](#).
- [How to create VNet peering with different subscriptions](#).

Network security groups

Migrating network security groups across Azure regions isn't supported at this time. We recommend that you create new network security groups in the target region and apply the network security groups rules to the new application environment.

Get the current configuration of any network security group from the Azure portal or by running the following PowerShell commands:

```
$nsg=Get-AzureRmNetworkSecurityGroup -ResourceName <nsg-name>
-ResourceGroupName <resourcegroupname>
```

```
Get-AzureRmNetworkSecurityRuleConfig -NetworkSecurityGroup $nsg
```

For more information:

- Refresh your [knowledge about network security groups](#).
- Review the [network security overview](#)
- Learn how to [manage network security groups](#).

ExpressRoute

Migrating an Azure ExpressRoute instance across Azure regions isn't supported at this time. For migration across cloud types, we recommend that you create new ExpressRoute circuits and a new ExpressRoute gateway in the target Azure region.

For more information:

- Refresh your knowledge by completing the [ExpressRoute tutorials](#).
- Learn how to [create a new ExpressRoute gateway](#).
- Learn about [ExpressRoute locations and service providers](#).
- Read about [virtual network gateways for ExpressRoute](#).

VPN Gateway

Migrating an Azure VPN Gateway instance across Azure regions isn't supported at this time. We recommend that you create and configure a new instance of VPN Gateway in the new region.

You can collect information about your current VPN Gateway configuration by using the portal or PowerShell. In PowerShell, use a set of cmdlets that begin with `Get-AzureRmVirtualNetworkGateway`.

Make sure that you update your on-premises configuration. Also, delete any existing rules for the old IP address ranges after you update your Azure network environment.

For more information:

- Refresh your knowledge by completing the [VPN Gateway tutorials](#).
- Learn how to [create a site-to-site connection](#).
- Review the [Get-AzureRmVirtualNetworkGateway](#) PowerShell cmdlet.
- Read the blog post: [Create a site-to-site connection](#).

Application Gateway

Migrating an Azure Application Gateway instance across Azure regions isn't supported at this time. We recommend that you create and configure a new gateway in the new region.

You can collect information about your current gateway configuration by using the portal or PowerShell. In PowerShell, use a set of cmdlets that begin with `Get-AzureRmApplicationGateway`.

For more information:

- Refresh your knowledge by completing the [Application Gateway tutorials](#).
- Learn how to [create an application gateway](#).
- Review the [Application Gateway PowerShell cmdlets](#).

DNS

To migrate your Azure DNS configuration across Azure regions, export the DNS zone file, and then import it under the new subscription. Currently, the only way to export the zone file is by using the Azure CLI.

After you sign in to your source subscription in the source Azure region, configure the Azure CLI to use Azure Resource Manager mode. Export the zone by running this command:

```
az network dns zone export -g <resource group> -n <zone name> -f <zone file name>
```

Example:

```
az network dns zone export -g "myresourcegroup" -n "contoso.com" -f "contoso.com.txt"
```

This command calls the Azure DNS service to export the zone contoso.com in the resource group myresourcegroup. The output is stored as a BIND-compatible zone file in contoso.com.txt in the current folder.

When the export is finished, delete the NS records from the zone file. New NS records are created for the new region and subscription.

Next, sign in to your target environment, create a new resource group (or select an existing one), and then import the zone file:

```
az network dns zone import -g <resource group> -n <zone name> -f <zone file name>
```

When the zone has been imported, you must validate the zone by running the following command:

```
az network dns record-set list -g <resource group> -z <zone name>
```

When validation is finished, contact your domain registrar and redelegate the NS records. To get NS record information, run the following command:

```
az network dns record-set ns list -g <resource group> -z --output json
```

For more information:

- Refresh your knowledge by completing the [Azure DNS tutorials](#).
- Review the [Azure DNS overview](#).
- Learn more about [Azure DNS import and export](#).

Network Watcher

Migrating a Network Watcher instance across Azure regions isn't supported at this time. We recommend that you create and configure a new Network Watcher instance in the target region. Afterward, compare results between the old and new environments.

For more information:

- Refresh your knowledge by completing the [Network Watcher tutorials](#).
- Review the [Network Watcher overview](#).
- Learn more about [network security group flow logs](#).
- Read about [Connection Monitor](#).

Traffic Manager

Azure Traffic Manager can help you complete a smoother migration. While migrating Azure resources across Azure regions, you can add a new target endpoint in the target region, and update the Traffic Manager profile to use the new endpoint.

However, you can't migrate Traffic Manager profiles across Azure cloud types. You can define additional endpoints in the target environment by creating a new Traffic Manager profile in the target region while using the source environment at the same time. When Traffic Manager is running in the new environment, you can still define endpoints that you haven't yet migrated in the source environment. This scenario is known as the [Blue-Green scenario](#).

The scenario involves the following steps:

1. Create a new Traffic Manager profile in the target Azure region.
2. Migrate and configure endpoints. For each endpoint in the source Azure region:
 1. Migrate the endpoint to the target Azure region.
 2. Add the endpoint in the new Traffic Manager profile.
3. Change your DNS CNAME record to the new Traffic Manager profile.
4. Wait until the queries to the old ATM profiles totally stop by monitoring the [Queries by endpoint returned](#) metric. Some LDNS might have cached the old profile names – it's a good idea to wait for some time to make sure all queries are now routed to the new ATM profile before disabling the old profile.
5. Disable the old Traffic Manager profile
6. Once you're sure the old ATM profile can be safely deleted, delete the old ATM profile.

For more information:

- Refresh your knowledge by completing the [Traffic Manager tutorials](#).
- Review the [Traffic Manager overview](#).
- Learn how to [create a Traffic Manager profile](#).

Load Balancer

Migrating a Load Balancer instance across Azure regions isn't supported at this time. We recommend that you create and configure a new load balancer in the target Azure region. If you're currently using the [Azure Load Balancer - Basic](#), it is recommended to upgrade to the [Azure Load Balancer – Standard](#).

Learn more about [Why use Standard Load Balancer](#), including [limits](#) and [pricing](#).

NOTE

As we continue to add new capabilities and features for the Load Balancer, we anticipate they'll only be available on the Standard SKU.

For more information:

- Refresh your knowledge by completing the [Load Balancer tutorials](#).
- Review the [Load Balancer overview](#).
- Learn how to [create a new load balancer](#).

Migrate storage resources

This section contains information that you may find helpful as you migrate Azure storage resources across Azure regions.

Blobs

AzCopy is a free tool you can use to copy blobs, files, and tables. Use AzCopy for your migration to copy blobs across Azure regions.

If you don't use managed disks for your source VM, use AzCopy to copy the .vhd files to the target environment. If you use managed disks, see [Managed Disks](#).

The following example shows how AzCopy works. For a complete reference, see the [AzCopy documentation](#).

AzCopy uses the terms Source and Dest expressed as URIs.

You get the three parts of the URI (storageaccountname, containername, blobname) from the portal, by using PowerShell, or by using the Azure CLI. The name of the blob can be part of the URI or it can be given as a pattern, like vm121314.vhd.

You also need to authenticate with the Azure Active Directory or SAS tokens to access the Azure Storage account. For instructions on how to authenticate, see [Authentication options](#).

Example:

URI part	example value
Source storageAccount	migratetest
Source container	vhds
Source blob	vm-121314.vhd
Target storageAccount	migratetarget
Target container	targetcontainer

This command copies a virtual hard disk across Azure regions:

```
azcopy cp https://migratetest.blob.core.windows.net/vhds/vm-121314.vhd?<sastokenhere> https://migratetarget.blob.core.windows.net/targetcontainer?<sastokenhere>
```

To get a consistent copy of the VHD, shut down the VM before you copy the VHD. Plan some downtime for the copy activity. When the VHD is copied, [rebuild your VM in the target environment](#).

For more information:

- Review the [AzCopy documentation](#).
- Learn how to [create a VM from restored disks](#).

Managed Disks

Azure Managed Disks simplifies disk management for Azure infrastructure as a service (IaaS) VMs by managing the storage accounts that are associated with the VM disk.

Because you don't have direct access to the .vhd file, you can't directly use tools like AzCopy to copy your files. The workaround is to first export the managed disk by getting a temporary shared access signature URI, and then download it or copy it by using this information. The following sections show an example of how to get the shared access signature URI and what to do with it:

Step 1: Get the shared access signature URI

1. In the portal, search for your managed disk. It's in the same resource group as your VM and has a resource type of **Disk**.
2. On the **Overview** page, select the **Export** button in the top menu. You must shut down and deallocate your VM first, or unattach the VM to complete the export.
3. Define a time for the URI to expire. The default time is 3,600 seconds.
4. Generate a URL.
5. Copy the URL. The url will only be showed one time after creation.

Step 2: AzCopy

For examples of how to use AzCopy, see [blobs](#). Use AzCopy or a similar tool to copy the disk directly from your source environment to the target environment. In AzCopy, you must split the URI into the base URI and the shared access signature part. The shared access signature part of the URI begins with the character "?". The portal provides this URI for the shared access signature URI:

```
https://md-kp4qvrzhj4j5.blob.core.windows.net/r0pmw4z3vk1g/abcd?sv=2017-04-17&sr=b&si=22970153-4c56-47c0-8cbb-156a24b6e4b5&sig=5Hfu0qMw9rkZf6mCjuCE4VMV6W3IR8FXQSY1viji9bg%3D>
```

The following commands show the source parameters for AzCopy:

```
/source:"https://md-kp4qvrzhj4j5.blob.core.windows.net/r0pmw4z3vk1g/abcd"
/sourceSAS:"
?sv=2017-04-17&sr=b&si=22970153-4c56-47c0-8cbb-
156a24b6e4b5&sig=5Hfu0qMw9rkZf6mCjuCE4VMV6W3IR8FXQSY1viji9bg%3D"
```

Here's the complete command:

```
azcopy -v /source:"https://md-kp4qvrzhj4j5.blob.core.windows.net/r0pmw4z3vk1g/abcd" /sourceSAS:"?sv=2017-04-17&sr=b&si=22970153-4c56-47c0-8cbb-156a24b6e4b5&sig=5Hfu0qMw9rkZf6mCjuCE4VMV6W3IR8FXQSY1viji9bg%3D"
/dest:"https://migratetarget.blob.core.windows.net/targetcontainer/newdisk.vhd" /DestKey:"o//ucD\... Kdpw=="
```

Step 3: Create a new managed disk in the target environment

There are several options for creating a new managed disk. Here's how to do it in the Azure portal:

1. In the portal, select **New > Managed Disk > Create**.

2. Enter a name for the new disk.
3. Select a resource group.
4. Under **Source type**, select **Storage blob**. Then, either copy the destination URI from the AzCopy command or browse to select the destination URI.
5. If you copied an OS disk, select the **OS** type. For other disk types, select **Create**.

Step 4: Create the VM

As noted earlier, there are multiple ways to create a VM by using this new managed disk. Here are two options:

- In the portal, select the disk, and then select **Create VM**. Define the other parameters of your VM as usual.
- For PowerShell, see [Create a VM from restored disks](#).

For more information:

- Learn how to export to disk [via API](#) by getting a shared access signature URI.
- Learn how to create a managed disk [via API](#) from an unmanaged blob.

Import/Export

You can't directly migrate Azure Import/Export job resources across Azure regions. The Azure Import/Export service doesn't support resource export or resource import.

An Azure Import/Export job resource created in a region needs to finish in the region so that the data is ingested to or exported from the storage account in that region.

However, you can create a new Azure Import/Export job resource in the new region, with a storage account in the new region.

Creating a new job resource can also be done by exporting Azure Import/Export resources [as a Resource Manager template](#), and then adapting the exported template for the target Azure region to re-create the resources.

NOTE

Exporting an Azure Import/Export template doesn't copy data (for example, blobs created in storage account). Exporting a template only re-creates Azure Import/Export metadata.

Consider changing the delivery Package, shipping Information, storage account Id, and other job properties as appropriate for the new region.

Azure Import/Export metadata

The following metadata elements are re-created when you export an Azure Import/Export template:

- Job Resource

For more information:

- Review the [Azure Import/Export Service overview](#).
- Azure Import/Export [Frequently Asked Questions](#)
- Become familiar with how to [export Azure Resource Manager templates](#) or read the overview of [Azure Resource Manager](#).

StorSimple

Migrating StorSimple service from one Azure region to another isn't supported at this time. We recommend you follow the manual process described [here](#) or contact customer support.

Migrate web resources

This section contains information that you may helpful as you migrate Azure web resources across Azure regions.

Web Apps

Migrating apps that you created by using the Web Apps feature of Azure App Service across Azure regions isn't supported at this time. We recommend that you export a web app as a Resource Manager template, back up your web app's file content (or ensure the application's source code is available externally in a source code control repository) and ensure you've stored offline any custom SSL certificates used with your web apps. Then recreate your web app after you change the location property in your Resource Manager template to the new region. Once the web app has been recreated in the new region, republish your web app's file content and upload and rebind any custom SSL certificates used by the web app.

IMPORTANT

Change location, Azure Key Vault secrets, certificates, and other GUIDs to be consistent with the new region.

For more information:

- Refresh your knowledge by completing the [App Service tutorials](#).

- Get information about how to [export Azure Resource Manager templates](#).
- Review the [Azure Resource Manager overview](#).
- Review the [App Service overview](#).
- Get an [overview of Azure locations](#).
- Learn how to [redeploy a template](#).

API Management

To migrate API Management endpoints from one Azure region to another, you can use the [backup and restore](#) feature. You should choose the same API Management SKU in the source and the target region.

NOTE

Backup and restore won't work while migrating between different cloud types. For that, you'll need to export the resource [as a template](#). Then, adapt the exported template for the target Azure region and re-create the resources.

Option 1: If you're okay with a different API Management instance name, then follow these instructions:

1. Create a new API Management instance with the same SKU as the source API Management instance in the target region with a new name.
2. Backup existing API Management instance to a storage account.
3. Restore the backup created in Step 2 to API Management instance created in Step 1 in the target region.
4. If you have a custom domain pointing to the source region API Management instance, update the custom domain CNAME to point to the new API Management instance.

Option 2: If you would like to preserve the API Management instance name, then follow the below instructions

NOTE

This is a riskier option and will result in downtime of the service.

1. Back up the API Management instance in the source region to a storage account
2. Delete the API Management instance in the source region

3. Create a new API Management instance in the target region with the same name as the one in the source region.
4. Restore the backup created in Step 1 to the new API Management instance in the target region.

Migrate database resources

This section contains information that you may find helpful as you migrate Azure database resources across Azure regions.

SQL Database

To migrate Azure SQL Database workloads, use Geo-Replication. For detailed instructions, see the blog post [Migrating Azure services to new regions](#).

NOTE

The connection string changes after the export operation because the DNS name of the server changes during export.

For more information:

- Learn how to [export a database to a BACPAC file](#).
- Learn how to [import a BACPAC file to a database](#).
- Review the [Azure SQL Database documentation](#).

SQL Server Stretch Database

Migrating SQL Server Stretch Database across Azure regions isn't supported at this time.

SQL Data Warehouse

Migrating SQL Data Warehouse across Azure regions isn't supported at this time.

Azure Cache for Redis

There are a few options if you want to migrate an Azure Cache for Redis instance across Azure regions. The option you choose depends on your requirements.

Option 1: Accept data loss, create a new instance

This approach makes the most sense when both of the following conditions are true:

- You're using Azure Cache for Redis as a transient data cache.

- Your application will repopulate the cache data automatically in the new region.

To migrate with data loss and create a new instance:

1. Create a new Azure Cache for Redis instance in the new target region.
2. Update your application to use the new instance in the new region.
3. Delete the old Azure Cache for Redis instance in the source region.

Option 2: Copy data from the source instance to the target instance

A member of the Azure Cache for Redis team wrote an open-source tool that copies data from one Azure Cache for Redis instance to another without requiring import or export functionality. See step 4 in the following steps for information about the tool.

To copy data from the source instance to the target instance:

1. Create a VM in the source region. If your dataset in Azure Cache for Redis is large, make sure that you select a relatively powerful VM size to minimize copying time.
2. Create a new Azure Cache for Redis instance in the target region.
3. Flush data from the **target** instance. (Make sure *not* to flush from the **source** instance. Flushing is required because the copy tool *doesn't overwrite* existing keys in the target location.)
4. Use the following tool to automatically copy data from the source Azure Cache for Redis instance to the target Azure Cache for Redis instance: [Tool source](#) and [tool download](#).

NOTE

This process can take a long time depending on the size of your dataset.

Option 3: Export from the source instance, import to the destination instance

This approach takes advantage of features that are available only in the Premium tier.

To export from the source instance and import to the destination instance:

1. Create a new Premium tier Azure Cache for Redis instance in the target region. Use the same size as the source Azure Cache for Redis instance.
2. [Export data from the source cache](#) or use the [Export-AzureRmRedisCache PowerShell cmdlet](#).

NOTE

The export Azure Storage account must be in the same region as the cache instance.

3. Copy the exported blobs to a storage account in the destination region by using a tool like AzCopy.
4. [Import data to the destination cache](#) or use the [Import-AzureRmRedisCache PowerShell cmdlet](#).
5. Reconfigure your application to use the target Azure Cache for Redis instance.

Option 4: Write data to two Azure Cache for Redis instances, read from one instance

For this approach, you must modify your application. The application needs to write data to more than one cache instance while reading from one of the cache instances. This approach makes sense if the data stored in Azure Cache for Redis meets the following criteria:

- The data is refreshed on a regular basis.
- All data is written to the target Azure Cache for Redis instance.
- You have enough time for all data to be refreshed.

For more information:

- Review the [overview of Azure Cache for Redis](#).

Migrate analytics resources

This section provides information that you may find helpful as you migrate Azure analytics resources across Azure regions.

Event Hubs

You can't directly migrate Azure Event Hubs resources across Azure regions. The Event Hubs service doesn't have data export or import capabilities. You can export Event Hubs resources [as a Resource Manager template](#), adapt the exported template for the target Azure region and re-create the resources.

NOTE

Exporting an Event Hubs template doesn't copy data (for example, messages). Exporting a template only re-creates Event Hubs metadata.

IMPORTANT

Change location, Azure Key Vault secrets, certificates, and other GUIDs to be consistent with the new region.

Event Hubs metadata

The following metadata elements are re-created when you export an Event Hubs template:

- Namespaces
- Event hubs
- Consumer groups
- Authorization rules

For more information:

- Review the [Event Hubs overview](#).
- Refresh your knowledge by completing the [Event Hubs tutorials](#).
- Check the migration steps for [Azure Service Bus](#).
- Become familiar with how to [export Azure Resource Manager templates](#) or read the overview of [Azure Resource Manager](#).

Analysis Services

To migrate your Azure Analysis Services models across Azure regions, use the [backup and restore operations](#).

If you want to migrate only the model metadata and not the data, an alternative is to [redeploy the model from SQL Server Data Tools](#).

For more information:

- Learn about [Analysis Services backup and restore](#).
- Review the [Analysis Services overview](#).

Migrate IoT resources to the target Azure region

This section contains information that you may find helpful as you migrate Azure IoT resources across Azure regions.

Functions

Migrating Azure Functions resources from across Azure regions isn't supported at this time. We recommend that you export a Resource Manager template, change the location, and then redeploy to the target region.

IMPORTANT

Change location, Azure Key Vault secrets, certificates, and other GUIDs to be consistent with the new region.

For more information:

- Refresh your knowledge by completing the [Functions tutorials](#).
- Learn how to [export Resource Manager templates](#) or read the overview of [Azure Resource Manager](#).
- Review the [Azure Functions overview](#).
- Read an [overview of Azure locations](#).
- Learn how to [redeploy a template](#).

Notification Hubs

To migrate settings from one instance of Azure Notification Hubs to another instance, export and then import all registration tokens and tags:

1. [Export the existing notification hub registrations](#) to an Azure Blob storage container.
2. Create a new notification hub in the target environment.
3. [Import your registration tokens](#) from Blob storage to your new notification hub.

For more information:

- Refresh your knowledge by completing the [Notification Hubs tutorials](#).
- Review the [Notification Hubs overview](#).

IoT Hub

To migrate IoT Hub, re-create the IoT Hub and use the export/import device identities function:

Note

This migration might cause downtime and data loss in your Azure IoT application. All telemetry messages, C2D commands, and job-related information (schedules and history) aren't migrated. You must reconfigure your devices and back-end applications to start using the new connection strings.

Step 1: Re-create the IoT hub

IoT Hub doesn't support cloning natively. However, you can use the Azure Resource Manager feature to [export a resource group as a template](#) to export your IoT Hub metadata. Configured routes and other IoT hub settings are included in the exported metadata. Then, redeploy the template in global Azure. You might find it easier to re-create the IoT hub in the Azure portal by looking at the details in the exported JSON.

Step 2: Migrate device identities

To migrate device identities:

1. In the source tenant, use the [ExportDevices](#) Resource Manager API to export all device identities, device twins, and module twins (including the keys) to a storage container. You can use a storage container in either the source or the target Azure region. Make sure that the generated shared access signature URI has sufficient permissions.
2. Run the [ImportDevices](#) Resource Manager API to import all device identities from the storage container to the cloned IoT hub in the target Azure region.
3. Reconfigure your devices and back-end services to start using the new connection strings from the new IoT hub created in Step 1.

Note

If you are migrating the resources across cloud types, the root certificate authority may be different in the source and the target region. Account for this when you reconfigure your devices and back-end applications that interact with the IoT Hub instance.

For more information:

- Learn how to [export IoT Hub bulk identities](#).
- Learn how to [import IoT Hub bulk identities](#).
- Review the [Azure IoT Hub overview](#).

Migrate integration resources to the target Azure region

This section contains information that you may find helpful as you migrate your Azure integration resources across Azure regions.

Service Bus

Azure Service Bus services don't have data export or import capabilities. To migrate Service Bus resources across Azure regions, you can export the resources [as an Azure Resource Manager template](#). Then, adapt the exported template for the target Azure region and re-create the resources.

NOTE

Exporting a Resource Manager template doesn't copy the data (for example, messages). Exporting a template only re-creates the metadata.

IMPORTANT

Change location, Azure Key Vault secrets, certificates, and other GUIDs to be consistent with the new region.

Service Bus metadata

The following Service Bus metadata elements are re-created when you export a Resource Manager template:

- Namespaces
- Queues
- Topics
- Subscriptions
- Rules
- Authorization rules

Keys

The preceding steps to export and re-create don't copy the shared access signature keys that are associated with authorization rules. If you need to preserve the shared access signature keys, use the `New-AzureRmServiceBuskey` cmdlet with the optional parameter `-Keyvalue` to accept the key as a string. The updated cmdlet is available in the [PowerShell Gallery release 6.4.0 \(July 2018\)](#) or on [GitHub](#).

Usage example

```
New-AzureRmServiceBusKey -ResourceGroupName <resourcegroupname> -Namespace  
<namespace> -Name <name of Authorization rule> -RegenerateKey  
<PrimaryKey/SecondaryKey> -KeyValue <string-keyvalue>
```

```
New-AzureRmServiceBusKey -ResourceGroupName <resourcegroupname> -Namespace  
<namespace> -Queue <queuenam> -Name <name of Authorization rule> -  
RegenerateKey <PrimaryKey/SecondaryKey> -KeyValue <string-keyvalue>
```

```
New-AzureRmServiceBusKey -ResourceGroupName <resourcegroupname> -Namespace  
<namespace> -Topic <topicname> -Name <name of Authorization rule> -  
RegenerateKey <PrimaryKey/SecondaryKey> -KeyValue <string-keyvalue>
```

NOTE

You must update your applications to use a new connection string even if you preserve the keys.

For more information:

- Refresh your knowledge by completing the [Service Bus tutorials](#).
- Become familiar with how to [export Resource Manager templates](#) or read the overview of [Azure Resource Manager](#).
- Review the [Service Bus overview](#).

Logic Apps

Azure Scheduler will [retire on September 30, 2019](#). Use Logic Apps to create scheduling jobs in the target Azure region. For detailed steps on migrating from Scheduler to Logic Apps, see [Migrate Azure Scheduler jobs to Azure Logic Apps](#).

For more information:

- Become familiar with features in Azure Logic Apps by completing the [Logic Apps tutorials](#).
- Review the [Azure Logic Apps overview](#).

Migrate identity resources

This section contains information that you may find helpful as you migrate Azure identity resources across Azure regions.

Multi-Factor Authentication

You'll need to re-create users and redefine your Azure Multi-Factor Authentication instance in your new environment.

To get a list of user accounts for which multi-factor authentication is enabled or enforced:

1. Sign in to the Azure portal.
2. Select **Users > All Users > Multi-Factor Authentication**.
3. When you're redirected to the multi-factor authentication service page, set the appropriate filters to get a list of users.

For more information:

- Learn more about [Azure Multi-Factor Authentication](#).

Migrate security resources

This article has information that you may find helpful as you migrate Azure security resources across Azure regions.

Key Vault

Some features of Azure Key Vault can't be migrated across Azure regions.

Encryption keys

You can't migrate encryption keys. Create new keys in the target region, and then use the keys to protect the target resource (for example, Azure Storage or Azure SQL Database). Securely migrate the data from the old region to the new region.

Application secrets

Application secrets are certificates, storage account keys, and other application-related secrets. During a migration, first create a new key vault in the target Azure region. Then, complete one of the following actions:

- Create new application secrets.
- Read the current secrets in the source Azure region, and then enter the value in the new vault.

```
Get-AzureKeyVaultSecret -vaultname mysecrets -name Deploydefaultpw
```

For more information:

- Refresh your knowledge by completing the [Key Vault tutorials](#).
- Review the [Key Vault overview](#).

- Review the [Key Vault PowerShell cmdlets](#).

Migrate management tool resources

This article has information that you may find helpful as you migrate Azure management tools across Azure regions.

Backup

You can't migrate Azure Backup jobs and snapshots across Azure regions.

For more information:

- Refresh your knowledge by completing the [Backup tutorials](#).
- Review the [Azure Backup overview](#).

Scheduler

Azure Scheduler will [retire on September 30, 2019](#). Use Azure Logic Apps to create scheduling jobs. For detailed steps on migrating from Scheduler to Logic Apps, see [Migrate Azure Scheduler jobs to Azure Logic Apps](#).

For more information:

- Become familiar with features in Azure Logic Apps by completing the [Logic Apps tutorials](#).
- Review the [Logic Apps overview](#).

Site Recovery

You can't move an existing Azure Site Recovery setup across Azure regions. [Disable](#) the existing configuration and set up a new Site Recovery solution in the target Azure region.

Refresh your knowledge by completing these step-by-step tutorials:

- [Azure-to-Azure disaster recovery](#)
- [VMware-to-Azure disaster recovery](#)
- [Hyper-V-to-Azure disaster recovery](#)

Migrate media resources

This section contains information that you may find helpful as you migrate Azure media resources across Azure regions.

Media Services

In Azure Media Services, you configure your own storage account and all media assets. First, create a new Media Services account in the target Azure regions. Then, reload the corresponding media artifacts and complete encoding and streaming under the new Media Services account.

For more information:

- Refresh your knowledge by completing the [Media Services tutorials](#).
- Review the [Media Services overview](#).
- Learn how to [create a Media Services account](#).

Media Player

You can select multiple endpoints in Azure Media Player. You can stream your content from the source Azure endpoints to the target Azure endpoints.

For more information, see [Azure Media Player](#).