Microsoft Azure

Azure Virtual Desktop
Migration Guide for Remote Desktop Services

Table of contents

Introduction

Why consider a cloud VDI?

As companies adapt to new ways of working and assess how to bring resilience into their businesses, enabling a secure, remote desktop experience that is accessible from anywhere is becoming increasingly important.

A **virtual desktop infrastructure (VDI)** approach is often leveraged to deliver a remote desktop experience to employees and is often delivered through **Remote Desktop Services (RDS)**. However, as an on-premises solution, RDS does not realize the full value of modernization or the benefits of a cloud VDI.

Azure Virtual Desktop is a managed VDI-delivered solution hosted on Microsoft Azure, that gives you the scalability of the cloud. Azure Virtual Desktop not only supports Windows Server but also provides Windows 10 enterprise multi-session, combining the Windows 10 experience with the ability to run multiple concurrent user sessions that was previously only available in Windows Server. You also get an optimized experience for Microsoft 365 Apps including Microsoft Teams and enhanced security for users, company apps, and data. Additionally, with programs such as App Assure, you will be able to modernize and reduce costs associated with three different pillars. *Figure 1* shows these pillars and includes the various areas in which costs can be cut.

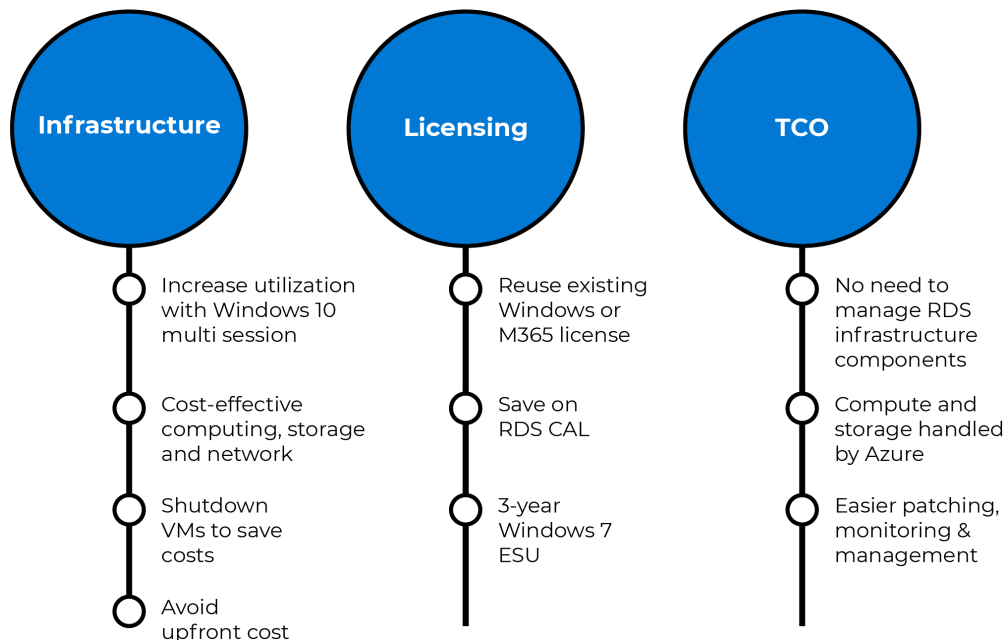| Infrastructure	Licensing	TCO
Increase utilization with Windows 10 multi session	Reuse existing Windows or M365 license	No need to manage RDS infrastructure components
Cost-effective computing, storage and network	Save on RDS CAL	Compute and storage handled by Azure
Shutdown VMs to save costs	3-year Windows 7 ESU	Easier patching, monitoring & management
Avoid upfront cost		

Figure 1: Azure Virtual Desktop cost-saving pillars

In addition to enabling remote work and saving on infrastructure, licensing, and **total cost of ownership** (**TCO**), Azure Virtual Desktop also helps bring other benefits to your business, including simplified IT management, security capabilities that help keep your users, data, organization safe and protection against outages with integrated Azure Site Recovery and Azure Backup technologies.

Why migrate to Azure Virtual Desktop?

If you have RDS and are considering migrating to a Microsoft solution, you can either migrate to an **infrastructure as a service (IaaS)** approach with Azure, or migrate directly to Azure Virtual Desktop.

To understand the differences between RDS on-premises, migrating to Azure, and migrating to Azure Virtual Desktop, take a look at *Table 1.* Although RDS provides you with flexibility, it also comes with a huge list of responsibilities. With an RDS environment, you are essentially responsible for managing and maintaining all required components. Migrating to RDS increases the responsibility provided by Microsoft; however, when migrating to Azure Virtual Desktop, Microsoft also manages the virtualization control plane for you. This allows you to shift your focus to what's really important to you, the perceived end user experience.

Responsibility	RDS on-premises	RDS on Azure	Azure Virtual Desktop
Identity	Customer	Customer	Customer
End user devices (mobile and PCs)	Customer	Customer	Customer
Application security	Customer	Customer	Customer
Session host operating system	Customer	Customer	Customer
Deployment configuration	Customer	Customer	Customer
Network controls	Customer	Customer	Customer
Virtualization Control Plane	Customer	Customer	Microsoft
Physical hosts	Customer	Microsoft	Microsoft
Physical network	Customer	Microsoft	Microsoft
Physical datacenter	Customer	Microsoft	Microsoft

Customer (blue) | **Microsoft** (gray)

Table 1: Responsibilities

This e-book is a guide to assist organizations in moving existing RDS infrastructures to Azure Virtual Desktop. It will discuss how to fully benefit from Azure Virtual Desktop and Azure in general and provide guidance on the seven steps to migrate your RDS workloads to Azure Virtual Desktop.

1 Prerequisites   **2** Azure Migrate Setup   **3** Discover VMs   **4** Review assessment

5 Replicate VMs   **6** Test migration   **7** Migrate to production
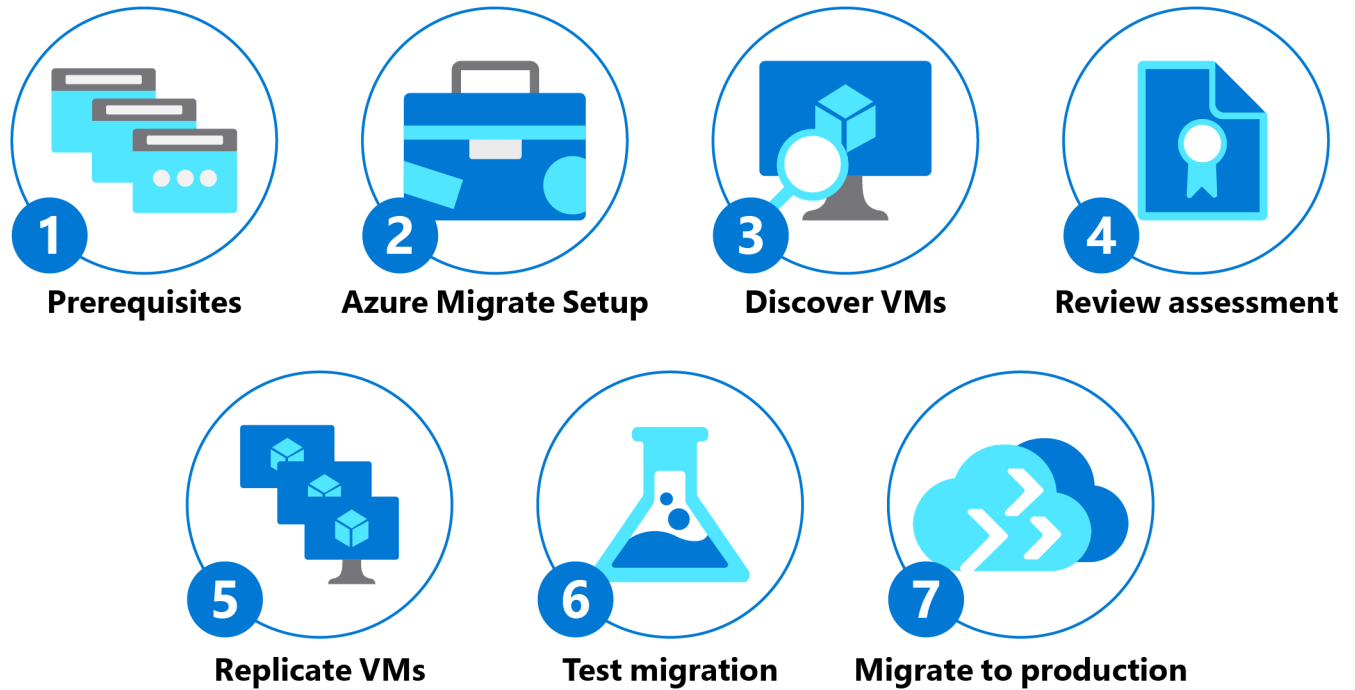
Figure 2: The seven-step migration plan

Figure 2 shows the seven advised steps to allow a smooth migration process from RDS to Azure Virtual Desktop. This e-book will guide you through this seven-step migration process. In the next chapter, we will look in detail at how to implement these seven steps.

Preparing your migration

In the previous chapter, we introduced the seven-step approach to migrate your RDS workloads to Azure Virtual Desktop. Step 1 is all about gathering the requirements that are needed before you can start migrating your workloads to Azure Virtual Desktop. In step 2, you install and configure Azure Migrate as the tool to migrate your existing workloads. Using Azure, you discover your **virtual machines (VMs)** in step 3. That discovery leads to a review assessment in step 4, which allows you to examine the replication details. When accepting the review assessment, the VMs will be replicated to Azure as part of step 5. Once replicated, you can start testing your workloads in step 6. After a successful test, you will migrate and take Azure Virtual Desktop into production.

This e-book will guide you through this seven-step migration process. These steps are based on Azure Migrate, which can be used to migrate many other on-premises workloads to Azure as well. Azure Migrate offers a lot of benefits, including assessments for readiness, sizing, and cost estimation. It also contains an integrated migration that allows near-zero downtime. You are provided with an integrated experience with end-to-end progress tracking. For more information and guidance on Azure Migrate, visit [this page](#).

In this chapter, we will cover the first four steps as part of the seven-step approach we previously introduced to start migrating your RDS workloads to Azure Virtual Desktop.
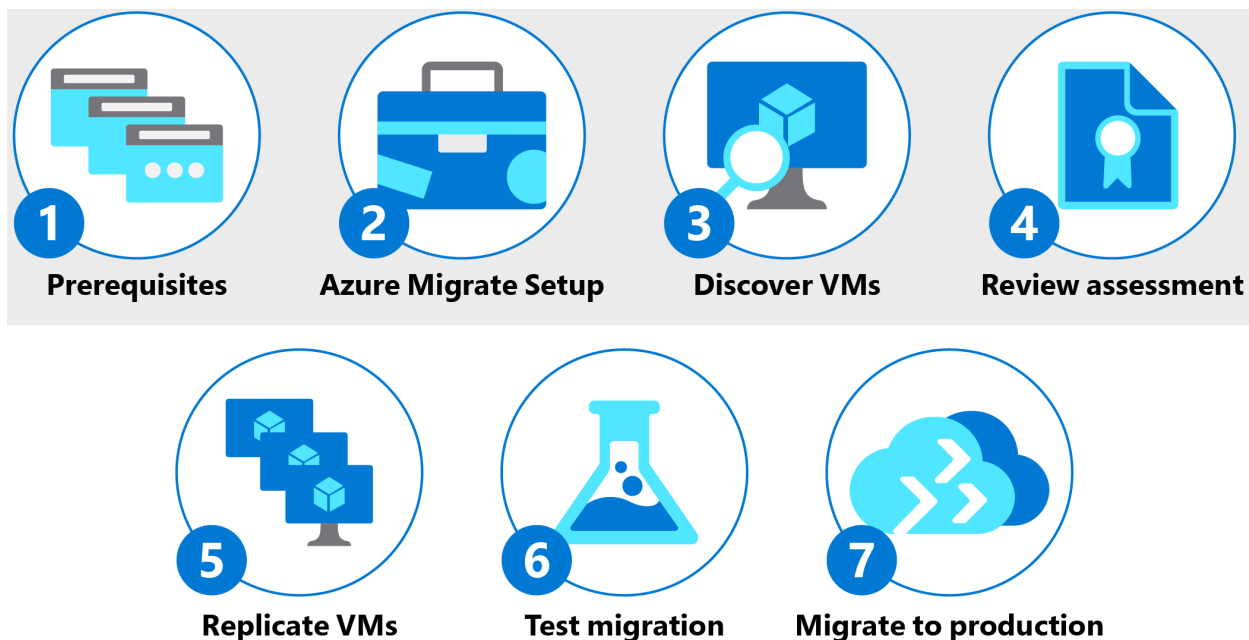

Figure 3: First four steps of the migration process

Step 1: Prerequisites

First, an Azure subscription is required. If your organization already has one, make sure you have the correct level of permissions. During this migration, you need permission to work with storage and networking components, and of course VMs. Make sure that domain services, either Active Directory or Azure Active Directory Domain Services, are synchronized with **Azure Active Directory (Azure AD)**. Ensure the domain service is accessible from the Azure subscription and virtual network to be connected where Azure Virtual Desktop will be deployed. Follow the Azure AD Connect guide for synchronizing Active Directory on-premises with Azure AD.

Step 2: Azure Migrate setup

Now we can start creating the Azure Migrate project. Within the Azure portal, there is a dedicated wizard allowing you to set up Azure Migrate for Azure Virtual Desktop. This can be found in the section called **VDI**, as shown in *Figure 4*:
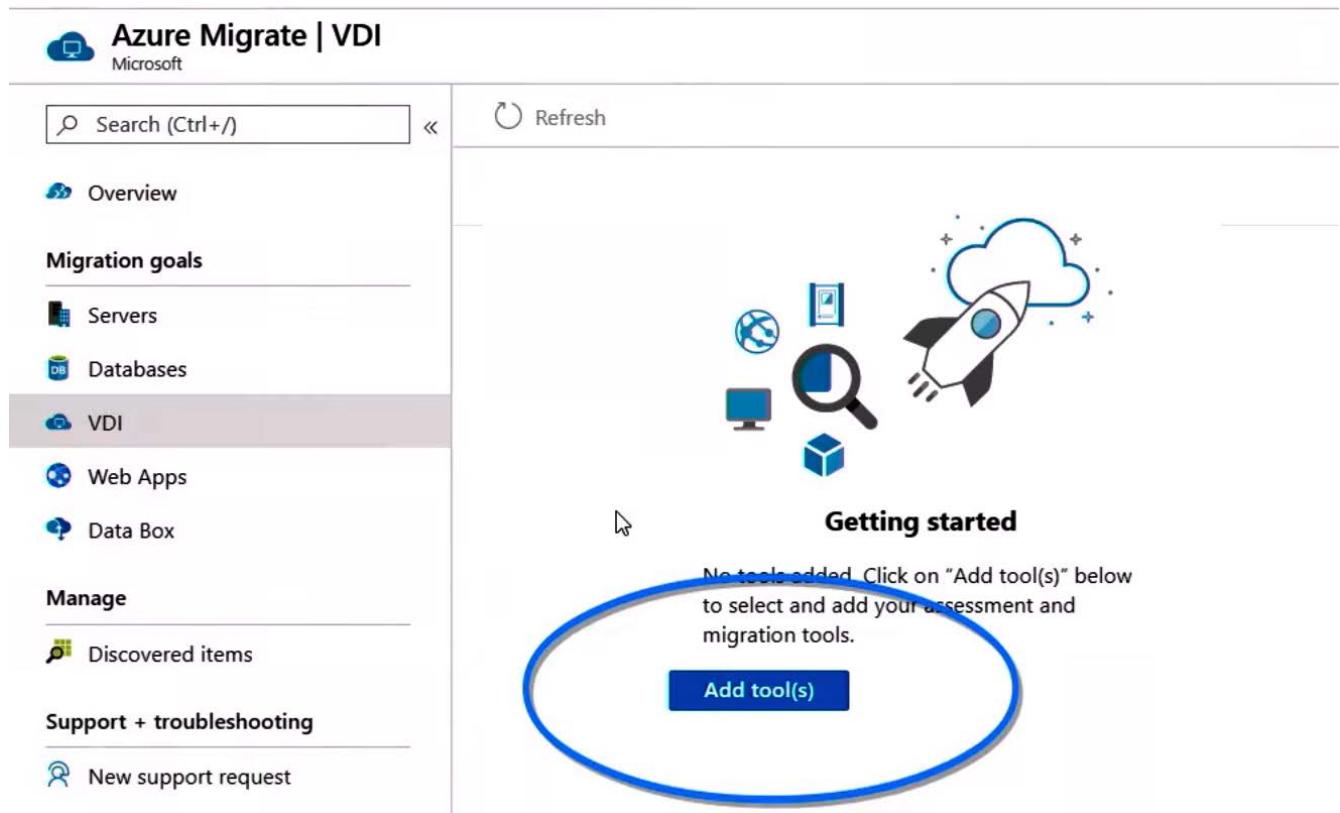

Figure 4: Start the Azure Migrate wizard

In this wizard, you set your subscription, resource group, project name, and geography. You start the assessment of the current RDS environment by selecting **Register**.

During this step, you create a new Azure Migrate project in the destination Azure subscription. This subscription needs to match the prerequisites outlined in step 1. You then select the option to assess and migrate servers, select **VDI**, and add a tool. After configuring basic parameters, such as the subscription, resource group, and location, make sure you select **Azure Migrate: Server Migration** as the migration tool. The setup wizard will also allow you to select optional ecosystem partner tools that provide additional benefits on top of the server migration. As per the example, you can select **Lakeside SysTrack** as your assessment tool on top of Azure Migrate as your migration tool. Lakeside is an ecosystem partner that specializes in assessing RDS and VDI environments. Lakeside SysTrack provides in-depth knowledge about your current workload to help you determine sizing and usage. Besides supporting Microsoft RDS, it also supports VMware and Citrix environments to help you assess those environments for migration to Azure too. After connecting Azure Migrate and optional ecosystem partner tools and accepting any requested permissions, the discovery process starts.

Step 3: Discover VMs

During this phase, the RD Session Host server of your current environment will be discovered and assessed. During this step, we are going to gather a lot of information about your current infrastructure. If you selected Lakeside SysTrack as your assessment tool in the previous step, this will help you collect even more information about your current RDS workload. Lakeside SysTrack requires an agent that you can easily install using your existing deployment tools. *Figure 5* shows the Lakeside SysTrack visualizer, which makes current usage, consumption, and application inventories easy to digest and helps you determine the sizing of your Azure Virtual Desktop VMs and much more.

Figure 5: Lakeside SysTrack visualizer example

As part of this step, you also gather insights on any application back-end workloads you may or may not want to move to Azure as well. Typically, moving those application back ends to Azure ensures the best performance because, in that scenario, the client side of the application running in Azure Virtual Desktop will be closer to the application back end. Azure Migrate can also assist you with moving these workloads to Azure as well. If you do decide to not move some of these back-end resources, ensure you configure connectivity with your on-premises environment by using either ExpressRoute or a site-to-site VPN. Detailed steps about the discovery can be found [here](#).

Step 4: Review assessment

When an adequate amount of data is captured, you can review the assessment data to determine the best migration path. This assessment data includes the raw assessment data from the desktop and the data broken into different user personas. As you analyze the data, you can determine the most cost-effective use of both pooled Azure Virtual Desktop resources and personal Azure Virtual Desktop resources. The information gathered as part of step 3 is visible in your Azure portal. The following figure shows an example containing information that is collected. This includes information such as the following:

- The number of users in each persona
- Applications in use by users
- Resource consumption by user
- Resource utilization averages by user persona
- VDI server performance data
- Concurrent user reports
- Top software packages in use

Figure 6: Assessment output example

Detailed steps about the assessment can be found here.

Regarding the session host VMs, depending on the results you analyzed as part of the assessment and depending on whether you want to benefit from Windows 10 multi-session or keep on using Windows Server, you have two options:

You can choose to migrate (lift and shift) existing on-premises RD Session Host servers to Azure and transform them into Azure Virtual Desktop session host servers. In that case, you will be using the **Discover** option in the **Azure Migrate: Server Migration** tools. This allows you to convert an appliance in its environment, which is going to manage the replication of the machines, to Azure Virtual Desktop. The replication provider is downloaded, installed, and registered to the Azure Migrate project to start the replication to Azure. As the replication of the hosts into Azure Blob Storage is now started, you can continue to let the replication occur until it's ready to test the VMs, and then migrate them into production. As machines start running in Azure, you make sure to install the Azure Virtual Desktop VM agent on each session host server. As a part of the installation, you enter a registration token for the Azure Virtual Desktop environment to associate the server with the correct environment. As the last step before the final migration, you assign users and groups to the appropriate app groups.

You can also choose to create a new template image based on Windows 10 multi-session to leverage all the benefits that come with the operating system. For the creation of a new template, Windows 10 Enterprise multi-session is available in the Azure image gallery. There are two options for customizing this image:

a. The first option is to provision a VM in Azure and then skip ahead to [Software preparation and installation](#).

b. The second option is to create the image locally by downloading the image, provisioning a Hyper-V VM, and customizing it to suit your needs.

For a full step-by-step guide on how to prepare, create, and deploy custom template images for Azure Virtual Desktop, consult [this guide](#).

Before continuing on to step 5, it is now time to start creating your Azure Virtual Desktop environment. In this step, you will create the Azure Virtual Desktop components and prepare your Azure subscription. Three different Azure objects will be created in your Azure subscription in this step: the host pool, the workspace, and the application group. These are explained as follows:

- **Host pools** are a collection of one or more identical VMs within Azure Virtual Desktop environments. Each host pool can contain an app group, which in turn contains an application(s) or a desktop that users can interact with as they would on a physical desktop.

- The host pool setup process creates a desktop application group by default. For the host pool to work as intended, you will need to assign this app group to users or user groups, and you must register the app group to a **workspace**.

- The default **app group** created for a new Azure Virtual Desktop host pool also publishes the full desktop. In addition, you can create one or more RemoteApp application groups for the host pool.

In chapter 3, we talk about the prerequisites and considerations that are important to become familiar with prior to creating your Azure Virtual Desktop deployment. In chapter 3, we will also elaborate some more on deployment options and guide you through the overall deplyoment process.

Creating the Azure Virtual Desktop environment

Prerequisites

Now that you have followed the migration process as part of the previously outlined steps, we will discuss the creation of the Azure Virtual Desktop environment in greater detail. Before doing so, a couple of things need to be in place. This section discusses these requirements. Make sure you comply with these requirements before you start implementing Azure Virtual Desktop and migrating your RDS workloads.

In terms of operating systems for the session host servers as part of your Azure Virtual Desktop host pools, there is a list of supported operating systems. Depending on the operating system you select, appropriate licenses for users who are connecting to the desktops and applications are also required. Make sure all users who are allowed access to any resource within Azure Virtual Desktop have the required license. *Table 2* shows the required licenses per operating system. You can read more about the required licenses here.

Operating system	Required license
Windows Server 2012 R2, 2016, 2019	RDS Client Access License (CAL) with Software Assurance
Windows 7 Enterprise	Microsoft 365 E3, E5, A3, A5, F3, Business Premium Windows E3, E5, A3, A5
Windows 10 Enterprise multi-session or Windows 10 Enterprise	Microsoft 365 E3, E5, A3, A5, F3, Business Premium Windows E3, E5, A3, A5

Table 2: Operating systems and required licenses

Furthermore, your infrastructure needs the following to be able to support Azure Virtual Desktop:

- An Azure AD instance needs to be in place.
- A Windows Server AD instance that is in sync with Azure AD. You can choose to implement this based on Azure AD Connect (ideal for hybrid organizations) or based on Azure AD Domain Services (ideal for hybrid or cloud organizations). In terms of identity sources and domain membership of the Azure Virtual Desktop session host servers, you can select from the following options:
  - You can use Windows Server AD in sync with Azure AD and the user accounts are sourced from Windows Server AD. The Azure Virtual Desktop session host VM is joined to the Windows Server AD domain.
  - You can use Windows Server AD in sync with Azure AD and the user accounts are sourced from Windows Server AD. The Azure Virtual Desktop session host VM is joined to Azure AD Domain Services.
  - You can use the Azure AD Domain Services domain and the user is sourced from Azure AD. The Azure Virtual Desktop session host VM is joined to the Azure AD Domain Services domain.
  - An Azure subscription is required, which needs to be parented to the same Azure AD tenant that contains the virtual network. The virtual network needs to have access to the Windows Server AD or Azure AD Domain Services instance.

The user connecting to Azure Virtual Desktop must meet the following requirements:

- The user must be sourced from the same AD that is connected to Azure AD. Azure Virtual Desktop does not support B2B or MSA accounts.
- The User Principal Name (UPN) you use to subscribe to Azure Virtual Desktop must exist in the AD domain the VM is joined to.

- The Azure Virtual Desktop session host VMs you create as part of your Azure Virtual Desktop host pool must be [Standard domain-joined](#) or [Hybrid AD-joined](#). VMs cannot be Azure AD-joined only at the moment. Azure AD-joined Azure Virtual Desktop session host VMs are planned to be supported later. And, as outlined before, the Azure Virtual Desktop session host server must be running one of the supported operating systems:
  - Windows 10 Enterprise multi-session, version 1809 or later
  - Windows 10 Enterprise, version 1809 or later
  - Windows 7 Enterprise
  - Windows Server 2019
  - Windows Server 2016
  - Windows Server 2012 R2

Azure Virtual Desktop does not support x86 (32-bit), Windows 10 Enterprise N, or Windows 10 Enterprise KN operating system images. Windows 7 also does not support any virtual hard disk (VHD)- or VHDX-based profile solutions hosted on managed Azure Storage. The available automation and deployment options depend on which operating system and version you choose, as outlined in this article on [supported operating systems](#).

If you are actively blocking outbound traffic from your Azure Virtual Desktop session host servers, you will need to unblock specific URLs so that your Azure Virtual Desktop deployment works properly. The Azure VMs you create for Azure Virtual Desktop must have access to the URLs listed on the [safe URL list](#).

Considerations

Limitations

When deploying Azure Virtual Desktop, there are a couple of technical limitations you need to be aware of. Consider and review these limitations prior to deploying Azure Virtual Desktop in your environment. An up-to-date list of these limitations can be found [here](#).

Optimizing cost

To make full use of Azure Virtual Desktop, we advise you to consider the different ways to realize cost savings. The following list contains six ways to save costs on your Azure Virtual Desktop deployment:

- Make use of Windows 10 multi-session as the operating system of your Azure Virtual Desktop session host servers. By leveraging a multi-session desktop experience for users that have identical compute requirements, you can let more users sign in to a single VM at once. This results in considerable Azure consumption cost savings for the VMs that are running. If you want additional guidance, the Windows 10 Enterprise multi-session FAQ contains more detailed information.

- Leverage Azure Hybrid Benefit. If your organization has Microsoft Software Assurance, you can use Azure Hybrid Benefit for Windows Server to save on the cost of your Azure infrastructure. For more information, visit this link.

- Azure **Reserved Instances (RIs)** can significantly reduce costs by up to about 72 percent compared to pay-as-you-go prices, with one-year or three-year commitment on Windows and Linux VMs. With RIs, you prepay for your VM usage. Optimally, combine RIs with Azure Hybrid Benefit (as outlined previously) to save up to 80 percent on list prices.

- You can reduce your total Azure Virtual Desktop deployment cost by scaling your VMs. This means shutting down and deallocating session host VMs during off-peak usage hours, then turning them back on and reallocating them during peak hours. More detailed information is provided in chapter 6.

- When setting up session host VMs, consider the different load balancing options. Breadth-first load balancing is the default (and most used) mode. Breadth-first load balancing spreads users across session host servers based on the number of sessions per host. Depth-first mode load balancing first fills up a session host server with the maximum number of users before it moves on to the next server. Depending on your use case, depth-first mode load balancing can result in additional cost savings but in some scenarios, it might not be sufficient to deal with user sign-in storms. You can adjust this setting for maximum cost benefits. For more detailed information on both load balancing options, visit this link.

Network guidelines

The general recommendation is to design your Azure networking using a hub-and-spoke topology. Consider the hub as a **demilitarized zone** (**DMZ**) deployed with your virtual network gateways and other security/edge appliances, such as firewalls and so on, while the spoke will act as the back-end zone where your session host servers are deployed to and is peered with the hub. Consult your network team during this phase for an optimal implementation.

The available bandwidth from the local client for the Azure Virtual Desktop backplane components, or more specifically the gateway component within Azure Virtual Desktop, plays a big part in the overall perceived end user experience. *Table 3* shows the minimum recommended bandwidths for a smooth user experience.

Workload type	Recommended bandwidth
Light	1.5 Mbps
Medium	3 Mbps
Heavy	5 Mbps
Power	15 Mbps

Table 3: Minimum recommended bandwidths

Do note that external factors can also increase the bandwidth that is being used. For example, if either the frame rate or display resolution increases, the bandwidth requirement will also increase. The bandwidths in *Table 3* are recommended values and they will vary depending on the applications consumed and the specific Azure Virtual Desktop use case you implement. It is advised to always perform benchmarking and testing to confirm the average values for your Azure Virtual Desktop deployment.

The display resolution that is being used also impacts the available bandwidth. *Table 4* contains a list of recommended bandwidths to allow a smooth user experience based on a frame rate of 30 **frames per second (fps)**.

Typical display resolutions at 30 fps	Recommended bandwidth
About 1024 × 768 px	1.5 Mbps
About 1280 × 720 px	3 Mbps
About 1920 × 1080 px	5 Mbps
About 3840 × 2160 px (4K)	15 Mbps

Table 4: Typical display resolutions and recommended bandwidths

Profile management guidelines

If your current RDS deployment leverages **User Profile Disk (UPD)** or roaming profiles as the profile management solution, be aware that you need to migrate or transition this to FSLogix profile containers. This is because UPD is not supported for Azure Virtual Desktop. Migrating profiles from UPD to FSLogix profile containers can be performed manually or using a method of your choice, but to provide a good migration path between your existing profile solution and profile container, we have created a migration script that is currently available as a private preview.*

Windows 10 multi-session

If you have an RDS deployment running locally today that you want to migrate to Azure Virtual Desktop, you are using a version of Windows Server as the operating system for the RD Session Host servers. This is because Windows 10 multi-session is not supported outside of Azure Virtual Desktop. As part of this migration, you can migrate your existing RD Session Host servers. Windows Server 2012 R2 and any version above is supported in Azure Virtual Desktop. However, for multiple reasons, as outlined in chapter 1, leveraging Windows 10 multi-session provides a lot of additional benefits, including a cost-saving since an RDS CAL is no longer required. To fully optimize Azure Virtual Desktop and Azure, we advise rebuilding your images to Windows 10 multi-user. This article tells you how to prepare a master **virtual hard disk (VHD)** image to upload to Azure, including how to create VMs and install software on them. We suggest following this article to fully leverage the capabilities of Azure Virtual Desktop.

---

* The private preview for UPD-to-FSLogix container conversion for Azure Virtual Desktop is provided without a service-level agreement, and it is not recommended for production workloads. Certain features might not be supported or might have constrained capabilities. FSLogix profile containers may not support all the functionality of the formats that are converted. For more information, see Supplemental Terms of Use for Microsoft Azure Previews.

Naming conventions

With any Azure subscription, it is important to have a solid naming convention. The following list contains considerations regarding naming conventions:

- An effective naming convention assembles resource names by using important resource information as part of a resource's name. When you construct your naming convention, identify the key pieces of information that you want to reflect in a resource name.
- Each workload can consist of many individual resources and services. Incorporating resource type prefixes into your resource names makes it easier to visually identify application or service components.
- When you apply metadata tags to your cloud resources, you can include information about those assets that couldn't be included in the resource name.

If you do not have an existing naming convention for your subscription, please follow the guidance at this link to maintain a consistent naming convention across your resources.

Deployment guidance

After having evaluated the prerequisites and considerations as outlined in the previous two sections, you are ready to deploy the necessary Azure Virtual Desktop objects in your Azure subscription.

Start by deploying a host pool. Host pools are a collection of one or more identical VMs within Azure Virtual Desktop environments. To create the host pool, decide on a unique name in a resource group and provide the Azure region.

Create a host pool

Basics   Virtual Machines   Workspace   Tags   Review + create

Project details

Subscription * ⓘ

> Microsoft Azure Enterprise   ⌄

    Resource group * ⓘ

> Select a resource group   ⌄

Create new

Host pool name *

> [   ]

Location * ⓘ

> East US   ⌄

Metadata will be stored in Azure geography associated with (US) East US

Learn more

Validation environment ⓘ   ● No ○ Yes

Host pool type

If you select pooled (shared), users will still be able to access their personalization and user data, using FSLogix.

Host pool type *

> Select a type   ⌄

Figure 7: The Create a host pool wizard in the Azure portal

The Azure geography associated with the regions you selected is where the metadata for this host pool, and its related objects, will be stored. Make sure you choose the regions inside the geography you want the service metadata to be stored in. Under **Host pool type**, select whether your host pool will be **Personal** or **Pooled**. If you choose **Pooled**, enter a max session limit and load balancing algorithm. Before you finish the creation of the host pool, you can decide on creating new VMs based on a new template image, or decide to not create a new VM yet in order to later migrate them from your on-premises RDS deployment. We touched upon this choice in greater detail in chapter 2. For a detailed technical description of deploying host pools, visit this page.

The host pool setup process creates a desktop **application group** by default. For the host pool to work as intended, you will need to assign this app group to users or user groups, and you must register the app group to a workspace.

Create a host pool

Basics   Virtual Machines   **Workspace**   Tags   Review + create

To save some time, you can register the default desktop application group from this host pool, with a new or pre-existing workspace.

Register desktop app group   ◯ No   ● Yes

Figure 8: Application group creation

If you select *No*, you can register the app group later, but we recommend you get the workspace registration done as soon as you can so that your host pool works properly. Choose whether you want to create a new **workspace** or select from existing workspaces. Only workspaces created in the same location as the host pool will be allowed to be chosen to register the app group to.

Create a host pool

Basics   Virtual Machines   **Workspace**   Tags   Review + create

To save some time, you can register the default desktop application group from this host pool, with a new or pre-existing workspace.

Register desktop app group   ◯ No   ● Yes

To this workspace ⓘ   [⌄]
                            Create new

Create new

Workspace name *

[workspacename   ✓]

We will also create a display name for this workspace, which you can always edit later.

[**OK**]   [Cancel]

Figure 9: Workspace creation

Review the information about your deployment to make sure everything looks correct. When you're done, select **Create**. This starts the deployment process, which creates the following objects:

- A new host pool.
- A desktop app group.
- A workspace, if you chose to create it.
- If you chose to register the desktop app group, the registration will be completed.
- A download link for an Azure resource management template based on your configuration.

Now that you have made your host pool, you can populate it with RemoteApp programs or one desktop per host pool. The default app group created for a new Azure Virtual Desktop host pool also publishes the full desktop. In addition, you can create one or more RemoteApp application groups for the host pool. You can only create 50 application groups for each Azure AD tenant. We added this limit because of service limitations for retrieving feeds for our users. For a detailed technical description of deploying app groups, visit this link.

Previously, we have completed steps 1 through 4, and in the previous section, we created our Azure Virtual Desktop deployment. We are now ready to start replicating the RD Session Host servers to Azure (step 5) and make them part of the Azure Virtual Desktop deployment, perform a test migration (step 6), and migrate to production (step 7).
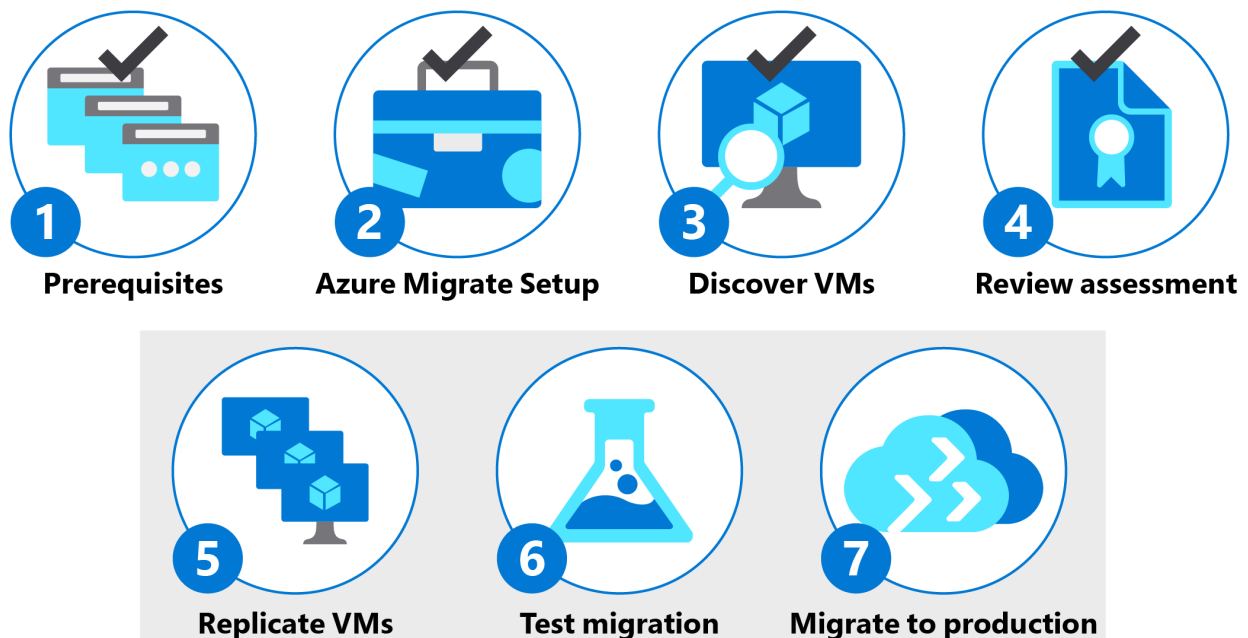

Figure 10: Steps 5, 6, and 7

Step 5: Replicate VMs

In this step, we will replicate the RD Session Host servers from on-premises to Azure to allow us to add the replicated servers to our created Azure Virtual Desktop deployment. You can begin the replication of VMs to Azure. In the Azure Migrate project, click **Replicate** under **Server Migration**. If you've run an assessment for the VMs, you can apply VM sizing and disk type (premium/standard) recommendations from the assessment results. Search for VMs as needed, and check each VM you want to migrate. Enable Azure Hybrid Benefit if you have Windows Server machines that are covered with active Software Assurance or Windows Server subscriptions and you want to apply the benefit to the machines you're migrating. After initial replication finishes, delta replication begins. Incremental changes to on-premises disks are periodically replicated to Azure. You can track the job status in the portal notifications.

Detailed steps about the replication can be found [here](#).

As discussed before, in some scenarios you might not want to migrate your existing VMs but rather build a new VM template; for example, if you want to leverage Windows 10 multi-session, which is only available as part of Azure Virtual Desktop. [This article](#) tells you how to prepare a new master image instead. In this case, the information collected during the assessment phase will be of great help to inform you how to compose your images and configure your application landscape.

Step 6: Test migration

When delta replication begins as part of step 6, you can run a test migration for the VMs, before running a full migration to Azure. We highly recommend that you do this at least once for each RD Session Host server before you migrate it. Running a test migration checks that migration will work as expected, without impacting the on-premises machines, which remain operational and continue replicating. Inside **Migration goals** select **servers**, select **Azure Migrate and Server Migration** and click **Test migrated servers**. Monitor the job in the portal notifications.

Detailed steps about the test migration can be found [here](#).

Step 7: Migrate to production

After you have verified that the test migration works as expected, you can migrate the on-premises RD Session Host servers. In the **Replicating machines** area, right-click the VM and click **Migrate**. By default, Azure Migrate shuts down the on-premises VM and runs an on-demand replication to synchronize any VM changes that occurred since the last replication occurred. This ensures no data loss. A migration job starts for the VM. Track the job in the Azure notifications. After the job finishes, you can view and manage the VM from the **Virtual Machines** page. After the migration is done, you use the **Stop migration** option. This process stops replication for the on-premises machine and removes the machine from the **Replicating servers** count in Azure Migrate.

Detailed steps about the migration to production can be found [here](#).

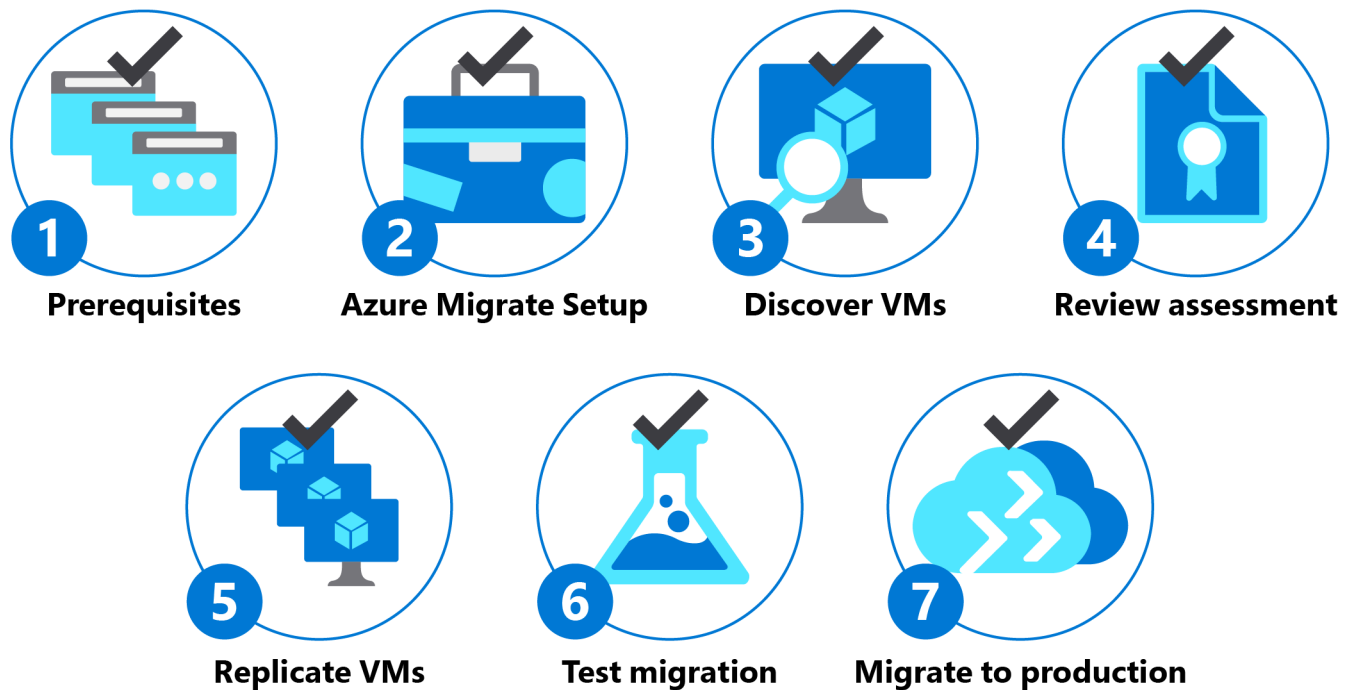|			
1 Prerequisites	**2** Azure Migrate Setup	**3** Discover VMs	**4** Review assessment
5 Replicate VMs	**6** Test migration	**7** Migrate to production	

Figure 11: The completed seven steps

We have now completed the seven-step process of Azure Migrate. In chapter 4, we will continue by explaining how to test and further prepare your Azure Virtual Desktop environment, followed by chapter 5, where we will provide guidance on going live and suggest post-deployment steps.

Testing and preparing the Azure Virtual Desktop deployment

Confirming Azure Virtual Desktop deployment health

The region that you are in can affect the user experience as much as network conditions. The Azure Virtual Desktop client, based on DNS information, will always route the user's connection to the closest Azure region that has Azure Virtual Desktop backplane components. To determine which Azure region works best for your specific location, browse to the Azure Virtual Desktop Experience Estimator. This allows you to estimate the quality of the experience your users will receive when connecting to Azure Virtual Desktop. It measures and estimates the connection **round-trip time (RTT)** from your current location, through the Azure Virtual Desktop service, to each Azure region in which you can deploy VMs as part of your Azure Virtual Desktop host pool.

Azure Region*	Round Trip Time (ms)
Germany West Central	31
Switzerland North	37
France Central	39
West Europe	40
Germany North	40
Switzerland West	40
UK South	45
France South	47

Figure 12: Example output of the Azure Virtual Desktop Experience Estimator

The Azure region with the lowest connection RTT from your current location is highlighted. Note that the displayed times are estimates intended to help assess the perceived end user experience and quality for your Azure Virtual Desktop deployment. The actual experience will vary depending on other conditions, for example, other specifications of the network, the end user device, and the configuration of the deployed VMs.

Azure Virtual Desktop host pool health

The Azure portal provides an easy way to determine the health of your Azure Virtual Desktop host pools. When inside the Azure portal, browse to Azure Virtual Desktop, click on **Host pools**, and open the corresponding host pool. The status of all individual Azure Virtual Desktop session host servers should be *Available* if switched on.

Figure 13: The status of the Azure Virtual Desktop session host server is Available

Guidance on testing Azure Virtual Desktop deployments

Once you have confirmed the health of your Azure Virtual Desktop environment, you can start testing the deployment.

Now that you have made your host pool, you can populate it with RemoteApp programs or one desktop per host pool. The default app group created for a new Azure Virtual Desktop host pool also publishes the full desktop. In addition, you can create one or more RemoteApp application groups for the host pool. Use the Azure portal to customize your Azure Virtual Desktop deployment by publishing applications, assigning users, and configuring host pool options.

You can access Azure Virtual Desktop resources on devices with Windows 7, Windows 10, and Windows 10 IoT Enterprise using the Windows Desktop client. Besides the Windows platform, you can also leverage Android, iOS, macOS, and web clients. The following list contains links to the various clients and guides on installing, configuring, and using the client:

- Connect with the Windows Desktop client
- Connect with the web client
- Connect with the Android client
- Connect with the macOS client
- Connect with the iOS client

Figure 14 shows an example of the Azure Virtual Desktop Windows client, which contains published desktops as well as RemoteApps.
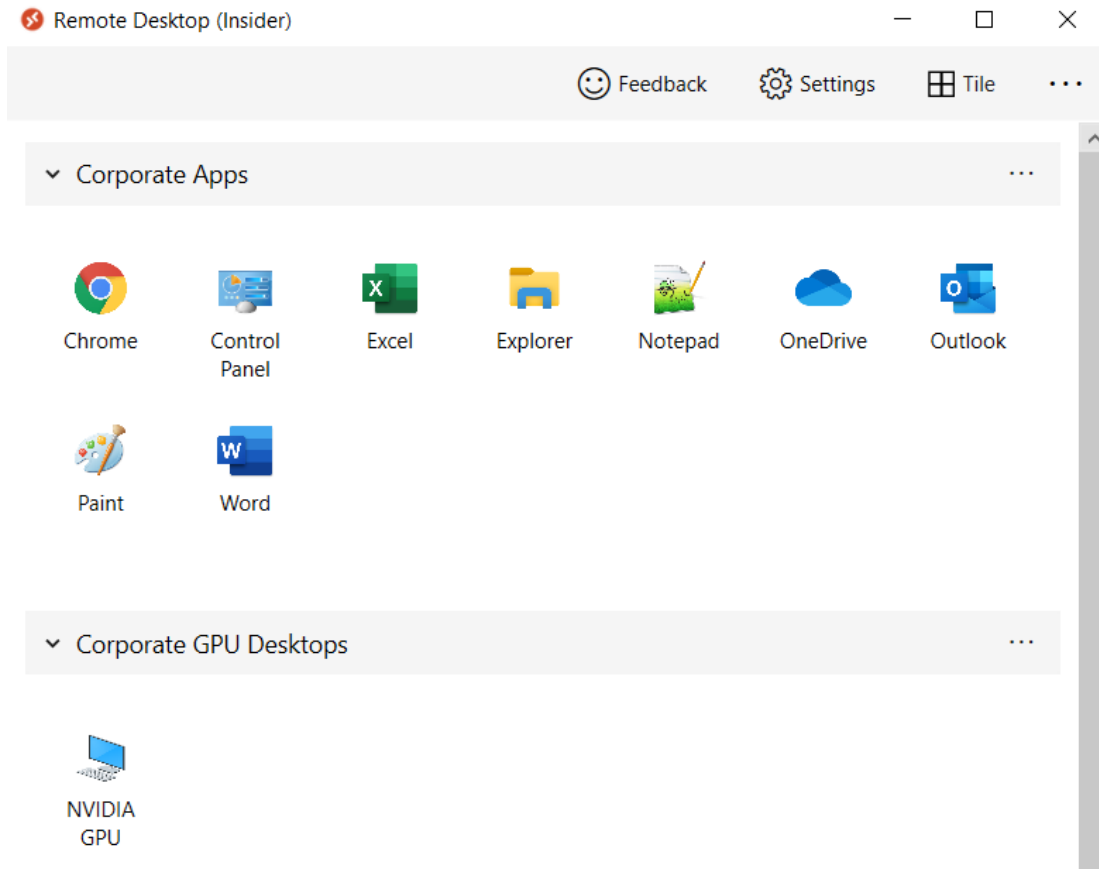
Figure 14: Azure Virtual Desktop client example

If you experience connection issues at this point, consult this page, which contains common error scenarios, management errors, and external connection error codes.

Final preparations for going live

User adaption is a key part of every IT project that involves changes for end users. Although users will be able to access their published applications and desktops as before, you need to thoroughly prepare user adoption for Azure Virtual Desktop. Users require training, manuals, and FAQ-style resources to help them start using the Azure Virtual Desktop service to its full potential. Accessing Azure Virtual Desktop applications and desktops is a little different compared to RDS as users need to go through a one-time initial signup on their device. Make sure users have the correct instructions and incorporate links to the various Azure Virtual Desktop clients as outlined in the previous chapter.

Going live and post-deployment steps

Confirming Azure Virtual Desktop health and usage

Once you have migrated to Azure Virtual Desktop, investigate the usage and the Azure Virtual Desktop health of your environment. During the preparation phase, we have determined which VM size to use for the session host VMs and investigated the usage of the environment by collecting telemetry data about the resources that were consumed; for example, by implementing Log Analytics as outlined in the *Monitoring* section later. This allows you to get insights into how many users are leveraging the Azure Virtual Desktop published applications and desktops and provides you with telemetry data on resource consumption. Use this information to rescale your session host servers as needed.

As explained in a previous chapter, the status of all individual Azure Virtual Desktop session host servers should be *Available*. Investigate the status of all your session host servers to make sure they are all properly connected and working. Do note that if you implemented autoscaling, as explained in chapter 6, currently scaled-down VMs do show up as *Unavailable* but do not require action.

Collect information about how many users are actively using Azure Virtual Desktop and gather information about the diagnostics to get insights into which sessions resulted in connection errors. Investigate these errors and follow up with suggested fixes.

Azure Advisor provides you with information about your Azure Virtual Desktop environment and guides you to best practices you might have missed during your deployment. Closely investigate the recommendations that Azure Advisor contains and implement the suggested best practices shown there.

Figure 15 shows an example of Azure Advisor providing guidelines for Azure Virtual Desktop.
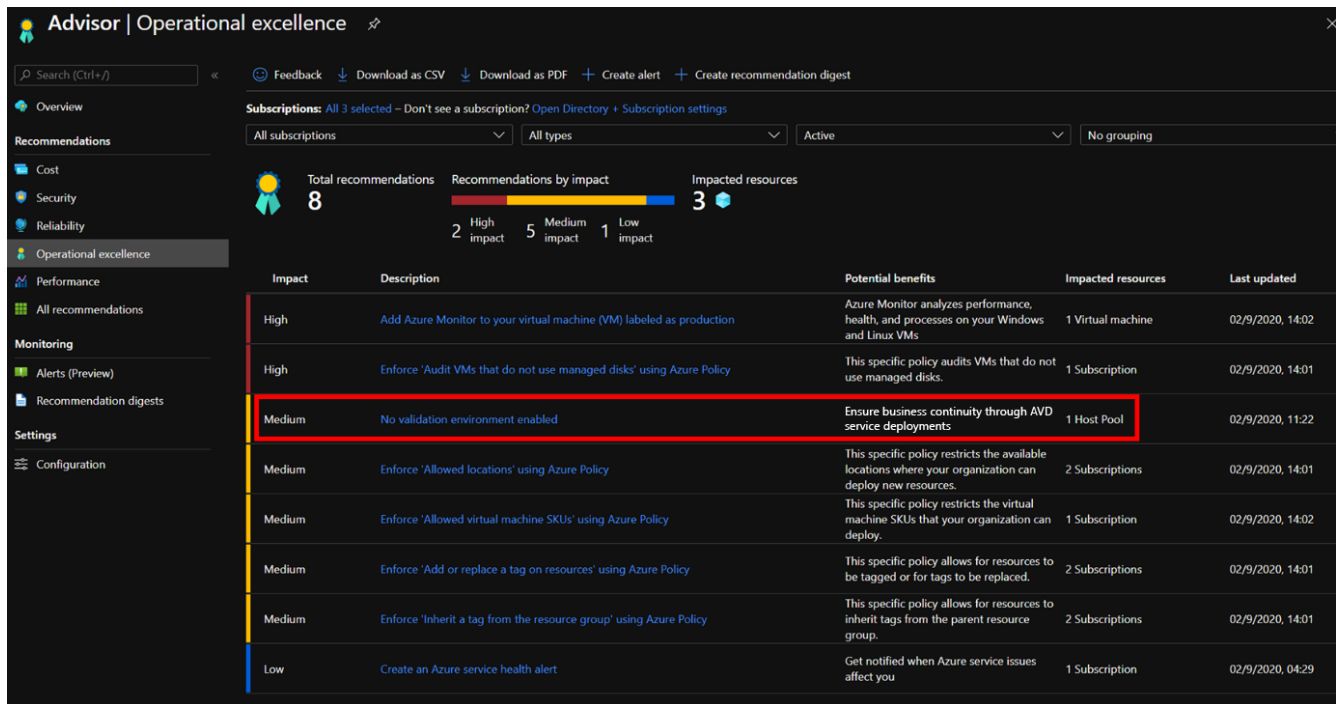

Figure 15: Azure Advisor for Azure Virtual Desktop example

Considerations and post-deployment steps

After you have taken your Azure Virtual Desktop deployment into production, it is good practice to consider and plan post-deployments steps. In a previous step, you confirmed the health and usage of your Azure Virtual Desktop deployment, but it is advised to monitor that health on an on-going basis. Consider implementing monitoring based on Azure Monitor and Log Analytics, as described in the *Monitoring* section later. Security is an important part of Azure Virtual Desktop. You might have used **multi-factor authentication (MFA)** in your RDS deployment. Now is a good time to consider adding Conditional Access to your deployment. Since Azure Virtual Desktop is based on Azure AD, this is a relatively easy task. We outline this task in the *Conditional Access* section later. As you start leveraging Azure Virtual Desktop, consider other scenarios that can benefit from this deployment. For example, you might allow your administrators remote access to your environment to perform maintenance tasks. You might currently be using a VPN solution for that. Azure Virtual Desktop can also be leveraged by your administrators by providing secure access to a central management (jump host) server. It is important to plan continuous autoscaling and updates to the session host VMs as part of your host pool. Automate these updates as outlined in the upcoming *Autoscaling* section.

Cleaning up an RDS deployment

Once you have successfully migrated your RDS deployment to Azure Virtual Desktop, it is advised to also clean up your RDS deployment. It's important to investigate, plan, and execute this cleanup thoroughly to make sure no components or configurations are left behind. There are a couple of areas where cleanup of the RDS environment is advised:

- It's obvious that the VMs of your RDS deployment can be removed. The VMs that are running your RDS infrastructure roles, such as RD Connection Broker, RD Web Access, and RD Gateway, are not needed anymore. The RD Session Host VMs have been migrated to Azure as part of the migration to Azure Virtual Desktop and can also be removed. It is a good practice to export your RDS configuration settings, such as, for example, the published apps, redirection options, and so on, to make sure you have that information available in case you want to compare Azure Virtual Desktop settings at a later stage. It might also be a good idea to snapshot/back up one of your RD Session Host servers in case you experience unexpected behavior with applications or settings inside the session host servers as part of Azure Virtual Desktop at a later stage and you want to compare settings with the previous RDS deployment.

- Your RDS deployment will have also used various DNS records and most likely these were created in public and private DNS services. DNS type A records were used to access the RD Connection Broker, RD Gateway, and RD Web Access components. Most likely, you also used DNS TXT records to allow the **RemoteApp and Desktop Connections (RADC)** Control Panel applet to automatically configure the web feed URL based on a user's email address. These DNS records can now be safely removed as they are not needed anymore for Azure Virtual Desktop.

- The infrastructure components of the RDS deployment, including the actual RD Session Host servers, are all members of your internal AD Domain Services. Since we have removed the infrastructure VMs, the corresponding AD computer objects, including their DNS entries, can now also be removed. Whether or not you can also remove the RD Session Host server computer object depends on how you migrated those workloads. If you migrated the VMs themselves, you are reusing those computer objects and you should not remove them. If you migrated based on a new set of VMs in Azure, for example, as part of your move from Windows Server to Windows 10 multi-session, you are most likely also using new names and computer objects, which means you can remove the old objects.

Guidance on additional capabilities

Autoscaling

As briefly mentioned in chapter 3, by autoscaling your session host VMs, you can reduce your total Azure Virtual Desktop deployment cost. Autoscaling for Azure Virtual Desktop makes sure you are shutting down and deallocating session host VMs during off-peak usage hours and turning them back on again and reallocating them during peak hours. The scaling tool provides you with a low-cost automation option for a scenario where you want to optimize your session host VM costs.

The autoscaling tool that Microsoft provides will help you automate and schedule session host VMs to start and stop based on peak and off-peak business hours that you define as parameters. The following two actions will take place after you have configured autoscaling:

- The autoscaling tool will scale out session host VMs based on the number of sessions per vCPU defined as one of the parameters.
- The autoscaling tool will scale in VMs during off-peak hours, leaving the minimum number of session host VMs running defined as one of the parameters.

The following sections provide more details on peak and off-peak hours.

During peak usage time, the job checks the current number of sessions and the VM capacity of the currently running session host for each host pool. It uses this information to calculate whether the running session host VMs can support existing sessions based on the `SessionThresholdPerCPU` parameter.

During the off-peak usage time, the job determines how many session host VMs should be shut down based on the `MinimumNumberOfRDSH` parameter. The job will then notify any currently signed-in users to save their work, wait the configured amount of time, and then force the users to sign out. Once all user sessions on the session host VM have been signed out, the job will shut down the VM. After the VM shuts down, the job will reset its session host drain mode.

At any time, the job also takes the host pool's `MaxSessionLimit` value into account to determine whether the current number of sessions is more than 90 percent of the maximum capacity. The job runs periodically based on a set recurrence interval. You can customize this interval based on the size of your Azure Virtual Desktop environment. Do note that starting and shutting down VMs can take some time, so remember to account for the delay. The recommended recurrence interval is every 15 minutes.

Currently, the tool also has the following limitations you need to be aware of:

- The autoscaling tool applies only to pooled multi-session session host VMs. Personal assigned Azure Virtual Desktop hosts cannot be autoscaled using this tool.

- The autoscaling tool manages VMs in any region; however, it can only be used in the same subscription as your Azure Automation account and Azure Logic App (the Azure object that the autoscaling tool consists of).

- If starting or stopping the VMs in your host pool takes longer than three hours, the job will fail.

To get started with autoscaling, follow [this guide](#).

Conditional Access

As briefly mentioned in chapter 1, Azure Virtual Desktop is based on Azure AD. This means Azure Virtual Desktop can instantly leverage all Azure AD security features. In most production environments, we advise configuring Conditional Access for Azure Virtual Desktop. This allows you to define additional security requirements that a user's session needs to meet before being able to get access to the published desktops and applications.

A common Conditional Access example is Azure MFA. After configuring Azure MFA, when a user signs in, the client asks for your username and password, followed by an Azure MFA prompt. When you select **Remember me**, your users can sign in after restarting the client without needing to reenter their credentials. These credentials are stored on the local credential manager. The latter applies to using the Windows Azure Virtual Desktop client; other Azure Virtual Desktop clients might show a different experience depending on the platform and the version of the client. While remembering credentials is convenient, it can also make deployments for Enterprise scenarios or personal devices less secure. To protect your users, you'll need to make sure the client always asks for Azure MFA credentials. More information on setting up and configuring Azure MFA for Azure Virtual Desktop is provided [here](#).

Monitoring

Like many other Azure services, Azure Virtual Desktop can also use Azure Monitor for monitoring and alerts. This enables Azure Virtual Desktop administrators to identify issues through a consolidated interface and dashboard. Azure Monitor and Log Analytics can gather activity logs for both end user and administrative actions. Each captured activity log falls under one of the following categories:

- Management activities
- Feed
- Connections
- Host registration
- Errors
- Checkpoints

Connections that don't reach Azure Virtual Desktop won't show up in the diagnostics results because the diagnostics role service itself is part of Azure Virtual Desktop. Azure Virtual Desktop connection issues can happen when the user is experiencing network connectivity issues.

Azure Monitor lets you analyze the consolidated information of your Azure Virtual Desktop deployment as well as telemetry data about the session host VMs. By defining performance counters, you can create a custom dashboard. Azure Monitor also allows administrators to create custom dashboards and share them with other Azure administrators. These dashboards are easily and conveniently accessed from within the Azure portal.

The following screenshot shows an example output of Azure Monitor and Log Analytics for Azure Virtual Desktop.

Figure 16: Azure Monitor for Azure Virtual Desktop example

To get started with Azure Monitor for Azure Virtual Desktop, use this guide.

Automation

Azure Virtual Desktop is fully based on **Azure Resource Manager (ARM)**. This also means you can leverage various ways to automate the deployment and maintenance of Azure Virtual Desktop. For automation using ARM, Microsoft provides and maintains a location on GitHub where you can retrieve automation scripts for various tasks.

ARM allows you to define the infrastructure that needs to be deployed in code. The infrastructure code becomes part of your project. Just like application code, you store the infrastructure code in a source repository and version it. Anyone on your team can run the code and deploy similar environments.

To implement infrastructure as code for your Azure solutions, you use ARM templates. The template is a **JavaScript Object Notation (JSON)** file that defines the infrastructure and configuration for your project. The template uses declarative syntax, which lets you state what you intend to deploy without having to write the sequence of programming commands to create it. In the template, you specify the resources to deploy and the properties for those resources. More information about ARM templates can be found [here](#).

There are ARM templates available to create and update Azure Virtual Desktop host pools, to configure autoscaling, and much more. The main GitHub repository can be found [here](#).

Azure Advisor

Azure Advisor can help users resolve common issues in Azure Virtual Desktop. These recommendations can reduce the need to submit help requests, saving you time and costs. Azure Advisor analyzes your configurations and telemetry to offer personalized recommendations to solve common problems. With these recommendations, you can optimize your Azure resources for reliability, security, operational excellence, performance, and cost.

Make sure to check your recommendations frequently, at least more than once a week. Always try to solve the issues with the highest priority level in Azure Advisor. If a recommendation seems less important, you can dismiss it or postpone it. When you notice an issue in Azure Virtual Desktop, always check Azure Advisor first. Azure Advisor will give you directions on how to solve the problem, or at least point you toward a resource that can help. For more information about how to access and configure Azure Advisor for Azure Virtual Desktop, visit [this link](#).

Microsoft Teams

Microsoft Teams on Azure Virtual Desktop supports chat and collaboration by default. With media optimizations for Azure Virtual Desktop, it also supports the calling and meeting functionality. With media optimization installed and configured for Microsoft Teams, the Windows Desktop client handles audio and video locally for Teams calls and meetings. Currently, the media optimizations are limited to the Azure Virtual Desktop Windows client only. You can, however, still use Microsoft Teams on Azure Virtual Desktop with other clients without optimized calling and meetings. Teams chat and collaboration features are supported on all platforms without limitations. For more information on how to enable and configure the media optimizations, follow [this link](#).

There are a couple of current limitations to using the Teams desktop client in Azure Virtual Desktop environments. Visit [this link](#) to familiarize yourself with the current limitations.

MSIX app attach

To improve the packaging experience for all Windows applications, MSIX has been introduced as a new packaging format that offers many great features. MSIX app attach is a way to deliver MSIX applications to both physical and virtual machines. In a Azure Virtual Desktop deployment, MSIX app attach creates full separation between user data, the operating system, and applications by using MSIX inside a virtual container. It removes the need for repackaging because it can leverage existing MSIX packages. The use of the MSIX app attach application can be made available to the Azure Virtual Desktop session host server and to the user in a fast and efficient way. For both the operating system as well as the user, an MSIX app attach application is treated as any other MSIX application.

There are a few things that need to be in place before you can get started with MSIX app attach:

- You need to have at least Windows 10 version 2004 as the operating system of the Azure Virtual Desktop session host server.
- A functioning Azure Virtual Desktop deployment needs to be in place.
- Preferably, you will leverage MSIX app attach with MSIX native applications provided by the application vendor. Consult the application vendor to receive more information. If an MSIX native application is not provided, you can also use the MSIX packaging tool to transform existing applications (MSI, EXE, and so on) into MSIX packages.
- A network share in your Azure Virtual Desktop deployment, where the MSIX package will be stored. This network share can be based on an Azure IaaS file server and Azure Files.

More information on setting up MSIX app attach for Azure Virtual Desktop can be found [here](#).

Conclusion

Summary

In chapter 1, we introduced the benefits of leveraging cloud VDI for your remote work strategy and reasons for migrating to Azure Virtual Desktop. We touched on advantages from a cost perspective and provided an introduction to the Azure Virtual Desktop service. Furthermore, we introduced the seven-step plan to migrate RDS to Azure Virtual Desktop by leveraging Azure Migrate. In chapter 2, we covered the first four steps from the seven-step plan in greater detail and provided guidance on performing these steps. Chapter 3 covered the creation of a Azure Virtual Desktop environment and we covered the remaining three steps of the plan. Chapter 4 talked about testing your Azure Virtual Desktop environment and preparing to go live. In chapter 5, we covered going live and provided guidance on post-deployment steps. In chapter 6, we provided guidance on additional capabilities to consider as part of your Azure Virtual Desktop deployment. Read further for additional resources, guidance on additional capabilities, a glossary, and more information about the author of this e-book.

Resources

We hope you enjoyed the tour and that you feel more prepared to migrate to Azure Virtual Desktop now! There are a lot of other resources and support to help—here are a few key references:

1. Read more Azure Virtual Desktop documentation to get the latest technical guidance.

2. Take a tutorial on getting started with Azure Virtual Desktop.

3. Watch the Microsoft Mechanics video to get a quick overview of how to migrate.

4. Sign up for a free Azure account to try deploying your virtualized Windows desktops and apps.

5. Get hands-on deployment guidance if you already have an Azure subscription.

6 Join the Azure Migration and Modernization Program to get guidance and expert help.

7. Contact sales to discuss pricing, technical requirements, and short- and long-term solutions for enabling secure remote work.

If needed, use the following resources for more in-depth information on specific topics mentioned in this e-book:

- Azure Reserved VM Instances (RIs)
- Azure Virtual Desktop partner integrations
- What is Azure Virtual Desktop?
- Windows 10 Computer Specifications and Systems Requirements
- SLA for Virtual Machines
- Remote Desktop Services – GPU acceleration
- GPU optimized virtual machine sizes
- Virtual Machine series
- Azure Virtual Desktop partner integrations
- About Azure Migrate
- Prepare and customize a master VHD image
- FSLogix Migration Preview Module
- Azure Virtual Desktop pricing
- Supported virtual machine OS images
- Safe URL list
- Windows 10 Enterprise multi-session FAQ
- Host pool load-balancing methods
- Recommended naming and tagging conventions
- Create a host pool with the Azure portal
- Manage app groups with the Azure portal
- Azure Virtual Desktop Experience Estimator
- Scale session hosts using Azure Automation
- Enable Azure Multi-Factor Authentication for Azure Virtual Desktop
- Use Log Analytics for the diagnostics feature
- What are ARM templates?
- RDS/Azure Virtual Desktop ARM templates
- Use Azure Advisor with Azure Virtual Desktop
- Use Microsoft Teams on Azure Virtual Desktop
- Set up MSIX app attach

Glossary

The following table contains a glossary of the terminology used throughout this book.

Term	Description
Active Directory Domain Services	A directory is a hierarchical structure that stores information about objects on the network. A directory service, such as Active Directory Domain Services (AD DS), provides the methods for storing directory data and making this data available to network users and administrators.
App Group	An app group is a logical grouping of applications installed on session hosts in the host pool. An app group can be of type Desktop of Remote App.
Azure Active Directory (Azure AD)	Azure Active Directory (Azure AD) is Microsoft's cloud-based identity and access management service, which helps your employees sign in and access resources
End users	After you've assigned users to their app groups, they can connect to a Azure Virtual Desktop deployment with any of the Azure Virtual Desktop clients.
FSLogix	FSLogix is designed to roam profiles in remote computing environments, such as Azure Virtual Desktop. It stores a complete user profile in a single container.
Host pool	A host pool is a collection of Azure virtual machines that register to Azure Virtual Desktop as session hosts when you run the Azure Virtual Desktop agent.
MFA	Multi-factor authentication is a process where a user is prompted during the sign-in process for an additional form of identification, such as to enter a code on their cellphone or to provide a fingerprint scan.
MSIX app attach	MSIX app attach is a way to deliver MSIX applications to both physical and virtual machines.
RDS	Remote Desktop Services, is the IaaS platform building virtualization solutions.
Windows 10 Enterprise multi-session	Windows 10 Enterprise multi-session, formerly known as Windows 10 Enterprise for Virtual Desktops (EVD), is a new Remote Desktop Session Host that allows multiple concurrent interactive sessions.
Workspace	A workspace is a logical grouping of application groups in Azure Virtual Desktop.
Azure Virtual Desktop	A desktop and app virtualization service that runs on Microsoft Azure.

Table 5: Glossary

About the author

Freek Berson is a Cloud Solutions Architect with a specialization in application and desktop delivery based on remoting technology. He has a long track record in the RDS space and has been awarded Microsoft **Most Valuable Professional (MVP)** since 2011.

Freek actively engages in the community. He speaks at various conferences around the world, including Microsoft Ignite, Microsoft Ignite | The Tour, Microsoft TechSummit, Microsoft TechDays, Azure Saturday, BriForum, E2EVC, ExpertsLive, and many more (online) events. He is also a published book author.

He works at Wortell, a cloud integrator company based in the Netherlands, where he focuses on end user computing, mostly on the Microsoft platform with a strong focus on Azure. He is also a managing partner at RDS Gurus.

He maintains his personal blog at themicrosoftplatform.net, where he writes articles and blog posts related to Azure Virtual Desktop, RDS, the Azure cloud, and other Microsoft technologies.

You can follow him on Twitter at @fberson and check his contributions through [his GitHub account](#).