

Migration for Active Directory Express

User Guide



© 2025 Quest Software Inc. ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Quest Software Inc.

The information in this document is provided in connection with Quest Software products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest Software products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST SOFTWARE ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST SOFTWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest Software makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest Software does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

Quest Software Inc.

Attn: LEGAL Dept

20 Enterprise Ste 100

Aliso Viejo, CA 92656

Refer to our Web site (<https://www.quest.com>) for regional and international office information.

Patents


Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at <https://www.quest.com/legal>.

Trademarks

Quest, the Quest logo, and Join the Innovation are trademarks and registered trademarks of Quest Software Inc. For a complete list of Quest marks, visit <https://www.quest.com/legal/trademark-information.aspx>. All other trademarks and registered trademarks are property of their respective owners.

Legend

 **CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.**

 **IMPORTANT, NOTE, TIP, MOBILE, or VIDEO:** An information icon indicates supporting information.

Contents

Introduction	7
Planning the Migration Project	8
Phase 1: Create a Workflow with the Workflow Wizard	8
Phase 2: ReACL Devices	8
Phase 3: Cutover Devices	9
Phase 4: Cleanup	9
Requirements	10
Networking	10
Outbound Internet Access	10
Application Ports	10
Domain Controller Ports	10
Accounts	10
Local Active Directory Account	10
Microsoft Entra ID Application Account	11
Microsoft Entra ID PowerShell Accounts	11
Agents	11
Devices	11
Device Agents	12
Operating Systems	12
PowerShell	12
.NET Framework	12
Network Requirement	12
Remote Devices	12
Cached Credentials Action	13
Network	13
How do I set up Offline Domain Join (ODJ)?	13
Web Proxy	13
Proxy Server	13
Proxy Address	13
Security	13
Ports	14
Repositories	14
How do I configure repositories?	14
Setup	15
Workflows	15
What is a Workflow?	15

How do I create and manage Workflows?	15
What should be entered as the Workflow Name?	15
What are the steps to create a Workflow?	16
Environments	18
What is an Environment?	18
Where do I manage Environments?	18
How are Environments added?	18
How do you export a list of Users, Groups, Contacts, and Devices in an environment?	18
How do you unmatch Users, Groups, Contacts, and Devices so they will not be synchronized?	18
How do you view logs for local environments?	18
How do you discover local environments?	19
Agents	19
What is the Directory Sync agent?	19
Where do you install the agent?	19
How do I download and install the agent?	19
Where do I manage agents?	20
How do I manage the agents?	20
How do I uninstall an agent?	20
Profiles	20
Migration Profiles	20
What is a Migration Profile?	20
How are Migration Profiles created?	20
Microsoft Entra Join Profiles	21
What is an Microsoft Entra Join Profile?	21
How are Microsoft Entra Join Profiles created?	21
Credential Profiles	22
What is a Credential Profile?	22
How are Credential Profiles created?	23
Credential Cache Profiles	23
What is a Credential Cache Profile?	23
How are Credential Cache Profiles created?	23
Migration Waves	24
What is a Migration Wave?	24
How do I manage Migration Waves?	24
What can I do with a Migration Wave?	24
How do I create a New Migration Wave?	25
How do I remove a Migration Wave?	25
How do I edit the name of a Migration Wave?	25
How do I filter Devices by Migration Wave?	25
Configurations	26
Downloads	26
How are device agents downloaded?	26
What are the Device Agent Service URL and Auth key used for?	26

How are device agents automatically upgraded?	26
Installing the Active Directory Agent	26
How do you manually install the Active Directory Agent?	27
How do you install the Active Directory Agent using a PowerShell Command?	29
.....	30
How do you install the Active Directory Agent using a GPO (Group Policy Object)?	30
How do you verify the GPO?	32
Repositories	32
What are the Repositories?	32
How do I manage Repositories?	32
What can I do with a Repository?	32
Product Licensing	33
Migrate and Navigate	34
Devices + Servers	34
What is the Devices + Servers page used for?	34
What is the difference between a "Ready Device" and a "Not Ready Device?"	34
What actions can be performed on Devices and Servers?	34
How-To	39
Offline Domain Join (ODJ)	39
1. CREATING ODJ FILES FOR EACH WORKSTATION	39
2. CONFIGURING THE CREDENTIAL CACHE PROFILE	40
3. CACHE CREDENTIALS JOBS	40
4. REACL	41
5. OFFLINE DOMAIN JOIN JOB	41
FAQs	42
Migration for Active Directory Express FAQs	42
What ports does the agent use to connect?	42
I've installed the agent and the device isn't ready, what do I do?	42
How do I adjust the agent polling interval?	42
How many migration actions can I queue at once?	42
Agent last contact time isn't updating every two minutes is something wrong?	43
The machine name was changed during the migration project, will it keep working?	43
How do I remove Devices from Migration for Active Directory Express?	43
Can I migrate to and from GCC/GCCH tenants?	43
Additional Info	44
Migration for Active Directory Express Architecture	44
Standard Configuration	44
Web Proxy Configuration	45
Troubleshooting	46
Cutover Job Result Codes	47

Upload Logs Result Codes	47
SQL Repermission Tool	48
CSV Import File Script	53
About us	69
Technical support resources	69

Introduction

On Demand Migration for Active Directory Express helps you reach your device migration goals by automatically creating workflows for your project using a simple step-by-step setup wizard, with no complex configuration required. This SaaS solution can perform basic user and group provisioning and can migrate devices between Active Directory, Entra ID, and hybrid directory environments.

This is one of three SaaS solutions Quest offers for device migration. Please reference the user guide for the solution you have purchased:

- [On Demand Migration Active Directory User Guide](#): Directory Sync, coexistence, domain rewrite, domain move, AD migration, device migrations
- [On Demand Migration Active Directory Express User Guide](#): Device migrations that do not require coexistence
- [On Demand Migration Entra ID for Devices User Guide](#): Device migrations from on-prem and hybrid AD to Entra ID that do not include an AD migration

Planning the Migration Project

A typical migration project using Migration for Active Directory Express can be broken up into four (4) phases.

- Phase 1: Create a Workflow with the Workflow Wizard
- Phase 2: ReACL Devices
- Phase 3: Cutover Devices
- Phase 4: Cleanup

Note: The Cleanup process typically occurs several months after the completion of the project.

This user guide walks you through the steps required to complete each phase, which can also be used to migrate devices from AD environments to Entra environments. The [Active Directory Entra-Join Quick Start Guide](#) walks you through the process of configuring and performing migrations for AD to Entra migrations.

Best practices for each phase of the migration project are presented below:

Phase 1: Create a Workflow with the Workflow Wizard

- The wizard begins with selecting the environment type for both source and target environments.
- Agents for both source and target local environments are deployed.
- Workflow options are configured and CSV files containing the Users, Groups and Devices to discover from the Source and the Users and Groups to match in the Target are uploaded.
- Migration Profile, Credential Profile, Credential Cache Profile, Microsoft Entra ID Join Profiles (for cloud environments) and Repositories are configured.

Phase 2: ReACL Devices

- Run a ReACL (file level re-permissioning) job on as many Devices as possible early in the process.
- ReACL is a non-destructive process that can be repeated as often as necessary up until Cutover in Phase 5.
- Troubleshoot any Devices with ReACL jobs which did not complete successfully.
- Run a ReACL job again close to the actual Cutover date. This will allow you to complete most of the ReACL process early and provide time to resolve any issues with things such as anti-virus software and Group Policies.

Phase 3: Cutover Devices

- Using some test Devices, Users, and Groups, verify a successful Device Cutover.
- Typically, a final ReACL job should be run the weekend before the scheduled Cutover to ensure any new Users and other changes are processed.
- A workstation reboot is required after the target account is enabled, the source account is disabled, and the Cutover is complete. This is usually completed in the evening when fewer end-users are impacted. Any impacted end-users should be alerted that this reboot is necessary.
- Optionally, use the Autopilot Cleanup option to prepare the AutoPilot-provisioned device for migration. This must be done before the cutover if the source Entra ID Joined device is Autopilot-provisioned and the Entra ID Join Profile has the Autopilot/Intune Cleanup option selected.

Phase 4: Cleanup

- The Cleanup phase typically takes place about two months after all Device Cutovers are complete. During the Cleanup phase, all permissions should be removed from the source domain and then the Active Directory agent should be removed from the Devices.
- Optionally, use the Set Intune Primary User action after the Device Cutover is completed.

Requirements

Networking

Outbound Internet Access

By default, each computer being migrated will require outbound access to the public Internet to securely communicate with the On Demand Migration services.



Important Tip: If your organization requires computers communicate externally using a web proxy see our [web proxy configuration requirements](#).

Application Ports

Each computer being migrated will require the Active Directory device agent and this agent will communicate to the On Demand Migration services, outbound over ports:

- 80
- 443
- 3030

Domain Controller Ports

Active Directory migrations also require a variety of Microsoft defined ports for communication between domain controllers. For a complete list of required ports, click [here](#).



Important Tip: For complete port information, review the [Service overview and network port requirements for Windows](#) documentation from Microsoft Support.

Accounts

Local Active Directory Account

- The agent installer will prompt for a domain account with permission to read and write on-premises Active Directory.

- An agent intended to sync all domains in a forest, must have access rights to all domains and objects used in workflows.

Microsoft Entra ID Application Account

- When creating a new Cloud Environment, an account with the Global Administrator Role is required to grant permissions and establish a connection.
- This account also needs the Global Administrator Role enabled during the first Read workflow so that it can create and configure Microsoft Entra ID PowerShell Accounts required for Directory Sync tasks.

Microsoft Entra ID PowerShell Accounts

- An OAuth token will be used by the application to create two (2) PowerShell accounts which are used to read and update objects in the cloud.
- The accounts will have Exchange Administrator, User Administrator and Teams Administrator roles assigned in order to read and update objects in the cloud.
- The accounts being used do not require any Microsoft 365 licenses.
- The accounts must be excluded from MFA requirements.
- An OAuth token will also be used by the used by the application to create a mail-enabled security group which will contain the two PowerShell accounts.

Agents

Migration for Active Directory Express is a 100% SaaS platform but to commit changes to on-premises directories (if applicable) such as Active Directory, a local agent must be installed and configured.

You will need at least one Directory Sync Agent installed per forest (environment). You may have up to five agents per forest. Adding more agents can offer limited fault-tolerance and can improve synchronization throughput, especially for near real-time password synchronization.



Important Tip: If you are only connecting to Microsoft Entra ID, local agents are not required.

Devices

The following is required for any Active Directory Computer(s) (devices) that will be migrated.

Device Agents

Each Active Directory Computer that will be migrated must have an agent installed on the workstation to orchestrate local jobs that must occur to prepare and execute the workstation's domain move.

Operating Systems

All computers or servers being migrated to the new domain must run one of the following operating systems:

- Windows 10
- Windows 11
- Windows Server 2016
- Windows Server 2019
- Windows Server 2022



Please Note: Entra ID Device Join is only supported for Windows 10 and 11.

PowerShell

- All client operating systems must have at least PowerShell 2.0 installed.

.NET Framework

- All Devices must have .NET Framework 4.7.2 or newer installed. This will appear as ".NET 4.7.2 Extended" in the add/remove programs list.
- If not present, an appropriate version of .NET Framework will be installed during agent installation if an internet connection is available.

Network Requirement

The devices in scope of the migration will need to be able to resolve and communicate to both source and target AD Active Directory Domain in order to perform device migration and rollback. For a complete list of required ports, please refer to the Microsoft Requirement link [Service overview and network port requirements - Windows Server | Microsoft Learn](#)

Remote Devices

To successfully migrate a remote employee's remote device using the Offline Domain Join (ODJ) feature the Cache Credential action must be run to collect the user's target credentials, so later you may cutover the device, while it is disconnected from the network.

The following is required:

Cached Credentials Action

- One-way external trust must be configured from the source domain to the target domain when the Cache Credential activity is processed

For more information about AD Trusts, check out this [MS Press article](#) about configuring trusts.

Network

- Network connectivity to both the source and target environments (Active Directory Domain Controllers) when the Cache Credential activity is processed



Important Tip: Offline domain join files must be created prior to running the Offline Domain Join process. A full explanation of Microsoft's Djoin.exe utility and how to create these files can be found [here](#).

How do I set up Offline Domain Join (ODJ)?

For complete details on how to set up ODJ, click [here](#).

Web Proxy

Some organizations may require all computers communicating externally direct their traffic through a web proxy to centralize communications. Active Directory agents can be configured to use a web proxy for communication to the On Demand Migration cloud services.

Proxy Server

- At least one (1) standard web proxy that supports http/TCP traffic.

Proxy Address

- The associated web proxy URL must be defined during configuration of the device agent.

Security

- If accessing the web proxy requires an additional username and password this will be required during configuration of the device agent.

Ports

All agents configured to use a web proxy will utilize the following outbound TCP ports:

- 80
- 443



Please Note: Agents configured to use a web proxy will not require UDP port 3030. For more information, see the [Web Proxy Configuration](#) under [Architecture](#).



Important Tip: Additional bandwidth overhead may occur when a web proxy is utilized to centralize all traffic.

Repositories

The following four Device Actions, when used, will require a defined storage share accessible from the device being migrated:

1. Upload Logs
2. Device Download
3. Offline Domain Join
4. Microsoft Entra ID Cutover

How do I configure repositories?

For complete details on how to configure repositories, click [here](#).

Workflows

What is a Workflow?

A workflow is a configurable series of steps that provides an easy automation framework to connect and manage Directory object synchronization. Activities such as creating, updating and deleting objects along with property/attribute synchronization and transformation.

How do I create and manage Workflows?

To create a Workflow, simply open the left navigation menu and click **Create a Workflow** in the side navigation menu, see *figure 1*, or click the **New** button under Workflows on the dashboard. The Workflow Wizard will open and will guide you through the creation of the Workflow.

To manage a Workflow, click the **Manage** button under Workflows on the dashboard.

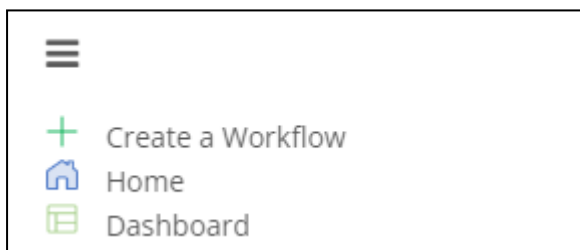


Figure 1: Side Navigation Menu

What should be entered as the Workflow Name?

You can name your workflow anything you'd like but remember that you may be referencing the same environment in multiple workflows. We suggest a name that generally describes the flow of objects. Then use the description field for the distinguishing characteristics. After this step, the wizard will guide you through all the necessary components that will make up your workflow.

What are the steps to create a Workflow?

Migration for Active Directory Express uses a wizard interface to guide you through the steps of creating and configuring the Workflow. To launch the wizard, click **Create a Workflow** in the side navigation menu or click the **New** button under Workflows on the dashboard.

Steps of the Workflow Wizard:

1. Select Environment Types - You are prompted to select the environment type for both source and target environments. Options are Local (a traditional on-premises Active Directory environment.) and Cloud (a Microsoft Entra ID environment.) See the [Environments](#) topic for more information.
2. Configure Source Environment:
 - a. Provide source environment name - Provide a descriptive name for the source environment that makes it easy to identify. For example, the Bluefish Resort on-premises Active Directory environment could be named bluefishresort.com.
 - b. (For Local Environments) Provide a name for the agent - Enter the name of the server used as the Directory Sync Agent. Directory Sync Agent is needed to connect to the local Active Directory Environment. See the [Agents](#) topic for more information.
 - c. (For Local Environments) Deploy a new agent - Download the Agent and use the provided information when installing the agent software on a domain-joined machine. See the [Agents](#) topic for more information.
 - d. (For Local Environments) Deployed Agent status - The current status of the agent deployment.
 - e. (For Cloud Environments) Connect to your Cloud Environment - Add a commercial or GCC tenant. Important: A Service Principal and two PowerShell service accounts will be created in the tenant. These service accounts must be excluded from MFA requirements, as described in the Account Prerequisites
3. Configure Target Environment:
 - a. Provide target environment name - Provide a descriptive name for the target environment that makes it easy to identify. For example, the Bluefish Resort on-premises Active Directory environment could be named bluefishresort.com.
 - b. (For Local Environments) Provide agent name - Enter the name of the server used as the Directory Sync Agent. Directory Sync Agent is needed to connect to the local Active Directory Environment. See the [Agents](#) topic for more information.
 - c. (For Local Environments) Deploy Agent - Download the Agent and use the provided information when installing the agent software on a domain-joined machine. See the [Agents](#) topic for more information.
 - d. (For Local Environments) Deployed Agent status - The current status of the agent deployment.
 - e. (For Cloud Environments) Connect to your Cloud Environment - Add a commercial or GCC tenant. Important: A Service Principal and two PowerShell service accounts will be created in the tenant. These service accounts must be excluded from MFA requirements, as described in the Account Prerequisites

4. Configure Workflow:
 - a. Name the Workflow - You can name your workflow anything you'd like but remember that you may be referencing the same environment in multiple workflows. We suggest a name that generally describes the flow of objects. Then use the description field for the distinguishing characteristics.
 - b. Users File to import for scoping and matching - Upload a CSV containing the Users to discover from the Source and the Users to match in the Target. Note that Import file must include ObjectID's. The file must include at least one user to continue.
 - c. Groups File to import for scoping and matching - Upload a CSV containing the Groups to discover from the Source and the Groups to match in the Target. Note that Import file must include ObjectID's
 - d. Devices File to import for scoping and matching - Upload a CSV containing the Devices to discover from the Source. Note that Import file must include ObjectID's
 - e. Select OU's in which to create objects - This is the Organizational Unit where you plan to store any newly created objects.
 - f. Choose your target domain - Select from the list of domains.
 - g. Password Settings - Enter the default password for new users. Note that the password policy on the source must meet or exceed the policy on the target.
5. (For Local Environments) Configure Migration Profiles - Configure the Device Migration options. See the [Migration Profiles](#) topic for more information.
6. (For Local Environments) Configure Credential Profiles - Credentials Profiles contain the source and target administrator's credentials and domain information used during an Offline Domain Join (ODJ) or remote device cutover process. See the [Credential Profiles](#) topic for more information.
7. (For Local Environments) Configure Credential Cache Profiles - Credentials Cache Profiles contains the target domain controller information required to cache a user's target credentials prior to the Offline Domain Join (ODJ) cutover process. You may skip this step if Offline Domain Join is not in-scope of your migration project. See the [Credential Profiles](#) topic for more information.
8. (For Cloud Environments) Microsoft Entra ID Join Profile - Microsoft Entra ID Join Provisioning Package file contains the target Microsoft Entra ID information used during Microsoft Entra ID Device cutover process.
9. Repositories - Repositories are specified storage locations on your network used for the following specific job types. See the [Repositories](#) topic for more information.
10. Downloads - See the [Downloads](#) topic for more information.
11. Summary - Please verify that all of the information has been correctly entered. Click the **Edit** button next to information that needs to be changed. Click **Run Workflow** to start the workflow or click the **X** button to finish changing the workflow without running the workflow.

Environments

What is an Environment?

If a workflow is a series of action steps, an environment is the receiver of those actions. In the Workflow Wizard, you will choose two environments, a source and a target, that the workflow will take actions against.

Where do I manage Environments?

To manage environments, select an environment from the Environment Summary table on the dashboard and click the **Manage** button.

How are Environments added?

The Workflow Wizard will guide you in adding a Source and Target Environment.

How do you export a list of Users, Groups, Contacts, and Devices in an environment?

On the dashboard, click the **Manage** button under the Environment Summary table. On the Environments page, select an environment and click the **Details** button. Expand an object list and click the **Export** button to download a CSV file of the Users, Groups, Contacts, and Devices.

How do you unmatch Users, Groups, Contacts, and Devices so they will not be synchronized?

On the dashboard, click the **Manage** button under the Environment Summary table. On the Environments page, select an environment and click the **Details** button. Expand an object list. select an object in the table and click the **Unmatch** button. The Match Status for the object will change to "Unmatched" and the object will not be synchronized.

How do you view logs for local environments?

Select a local environment in the Environments table and then click **Discovery Logs** to export a CSV with password or discovery information.

How do you discover local environments?

Select a local environment in the Environments table and then click **Discover** to begin the discovery process for the environment.

Agents

What is the Directory Sync agent?

The Directory Sync agent is the key component that communicates with the Active Directory environment.

Where do you install the agent?

The agent must be installed in every forest that you plan to include as a Directory Sync environment. We suggest that you create a virtual machine exclusively for this purpose. Review the [Requirements](#) for the minimal hardware and software requirements.

How do I download and install the agent?

You will be able to download the latest version of the agent from the Deploy a new agent screen of the Workflow wizard. Copy the URL and the access key that will be needed during the install of the agent. The downloadable executable is the same for all projects, it is the Registration URL and Registration Key that makes the agent unique when it is installed.

To install of the agent enter credentials that have read or read/write access to the domain, depending on the direction of synchronization.

Copy and paste the information from the Directory Sync agent screen.

No further action is needed on the workstation. A look at services confirms that the Directory Sync agent is running.

A list of agents appears on summary screen, including status information as well as the registration URL and access keys should you need them again in the future.



Please Note: If using the agent Auto-Upgrade feature and deployment software that uses MSI ProductCode based detection, the Auto-upgrade feature should be disabled after initial deployment or the detection method should verify via a folder path.

Where do I manage agents?

To manage agents, select the **Manage** button under the Environment Summary table on the dashboard. Then select an environment on the Environments page and click the **Settings** button. Select the **Agents** tab to manage the environment's agents.

How do I manage the agents?

On the Agents page, you can check the current status of your current agents or add new ones. Select an agent for additional options. You have the option to copy the Registration URL or the Registration Key if you need to reinstall the agent for any reason. The History button will give you details on the run history. When the agent is updated, any agent using the old version will offer you the upgrade option so that you can update your current agent installation.

How do I uninstall an agent?

If you need to uninstall an agent from any machine, in order to reinstall on the same machine, you must first delete the registry folder located at HKEY_LOCAL_MACHINE> SOFTWARE> Quest > Agent and then uninstall.

Afterwards, simply create a new agent (with a new access key) under Agents managements from the left navigation menu before re-installing on the same machine.

Profiles

Migration Profiles

What is a Migration Profile?

A Migration Profile is a collection of common settings used to manage the domain join process during Device Cutover which can be defined once and then applied to multiple Devices. Migration Profiles are used for AD and Hybrid Microsoft Entra ID device migrations.

How are Migration Profiles created?

To create a Migration Profile:

1. On the *Migration* section of the *Profiles* page, click the **Add** button. The *Add Your Migration Profile* window appears.
2. Enter values for the following fields, then click **Next**:

- *Profile Name* - A name to identify this profile (for example, "10 Second Reboot Delay").
- *Domain Join Delay* - The delay time a Device agent processing a Cutover job applies before attempting to change the computer's domain to join the new domain.
- *Reboot Delay* - The time for a Device agent processing a Cutover job to delay after joining the new domain before rebooting the computer.
Note: If set to any value other than zero (0), the user will receive a pop-up notification informing them that the workstation will be rebooted when the Cutover is performed. If set to zero, no notification will appear.

3. Enter values for the following fields, then click **Next**:

- *Empty Recycle Bin?* - How to handle the Recycle Bin during Cutover, either **Empty** or **Don't Empty**.

Note: End users may get an error message that their Recycle Bin has been corrupted after migration if the Recycle Bin is not empty. See [Troubleshooting](#) for more information about this issue.

- *Specify Target OU* - The target OU where the Device will be created. If this field is left blank, they will default to the Computers container.
- *Join to Existing Computer Account* - Select **Yes** if you expect to join the Device to the existing target Computer during Cutover. Select **No** if you want the Device to be created in the target by the Cutover job.

4. Click **Save Profile**. The new migration profile is added to the list.

Microsoft Entra Join Profiles

What is an Microsoft Entra Join Profile?

A Microsoft Entra Join Profile is a collection of settings used to manage the Entra join process during Device Cutover which can be defined once and then applied to multiple Devices. Microsoft Entra Join Profiles are used for AD to Entra device migrations.

How are Microsoft Entra Join Profiles created?

To add an Microsoft Entra Join Profile:

1. On the *Microsoft Entra Join* section of the *Profiles* page, Click the **Add** button. The *Add Your Microsoft Entra Join Profile* window appears.
2. Enter a **Profile Name** to identify this Microsoft Entra Join Profile.
3. Enter a value in the following field:

- **Bulk Enrollment Package File Name** - The name of the Microsoft Entra bulk enrollment package in packagename.ppkg format, which has been created by the client administrator using the Windows Configuration Designer and copied to the network share defined in the Microsoft Entra Bulk Enrollment Repository
4. Select an option from the following drop-down list:
 - **Target Environment** – The cloud-only Azure environment associated with the Azure bulk enrollment package used in this Profile
 5. Select a **Device Name Option**:
 - If you choose **Device Name Defined Per Provisioning Package**, the device will be migrated to Entra using the dynamic naming convention configured in the Microsoft Entra bulk enrollment package used in this Profile
 - If you choose **Keep Original Device Name**, the dynamic name assigned by the Microsoft Entra bulk enrollment package will be overwritten and replaced with the original device name when migrating to Entra
 6. Select the **Enroll Into Intune Management** option to enroll the device for Intune management with the first logged on user after cutover as the PrimaryUser.
 7. Select the **Auto-Pilot/Intune Cleanup** option to clear existing Auto-Pilot/Intune provisioning information from the device as part of the cutover.
 8. Select the **Source Directory is Active Directory Joined or Hybrid Microsoft Entra ID Joined** option if you wish to include Active Directory Joined or Hybrid Microsoft Entra Joined devices.
 9. Enter values in the following fields under Source Domain Credentials:
 - **FQDN of Domain** - The domain FQDN of the source in source.domain.com format.
 - **Username** - The username to access the source domain in domain\username or UPN (username@domain.com) format.
 - **Password** - The password credential to access the source domain.
 10. Under *Preflight Check Validation*, select the **Skip Source Local Active Directory Validation** option to not validate the source local Active Directory.
 11. Click **Save Profile**. The Microsoft Entra Join Profile is added to the list.

Credential Profiles

What is a Credential Profile?

A Credential Profile is a set of source and target domain credentials used for Cutover which can be defined once and then applied to multiple Devices.

How are Credential Profiles created?

The specified credentials must be able to join and disjoin a computer from the specified domain as well as disable a computer in the specified domain. A trust between the source and target domain is not required.

To add a Credential Profile:

1. On the Credentials section of the *Profiles* page, Click the **Add** button. The *Add Your Credentials Profile* window appears.
2. Enter a **Credential Name** to identify this Credentials Profile.
3. Enter values in the following fields under Source Domain Credentials:
 - **FQDN of Domain** - The domain FQDN of the source in source.domain.dom format.
 - **Username** - The username to access the source domain in domain\username or UPN (username@domain.dom) format.
 - **Password** - The password credential to access the source domain.
4. Enter values in the following fields under Target Domain Credentials
 - **FQDN of Domain** - The domain FQDN of the target in target.domain.dom format.
 - **Username** - The username to access the target domain in domain\username or UPN (username@domain.dom) format.
 - **Password** - The password credential to access the target domain.
5. Check **Source Device is EntraID Joined** if the source device is disable the entered Source Domain Credentials.
6. Click **Save Profile**. The Credential Profile is added to the list.

Credential Cache Profiles

What is a Credential Cache Profile?

A Credential Cache Profile is a collection of settings related to the target domain controller used for caching a user's target credentials prior to completing an Offline Domain Join cutover which can be defined once and then applied to multiple Devices.

How are Credential Cache Profiles created?

To add a Credential Cache Profile:

1. On the *Credential Cache* section of the *Profiles* page, Click the **Add** button. The *Add Your Credential Cache Profile* window appears.

2. Enter a **Credential Cache Name** to identify this Credential Cache Profile.
3. Enter values in the following fields:
 - **Target Domain Controller IP Address** – The IP address of the target Domain Controller.
 - **Target Domain Controller Ping Interval** - The number of seconds the script will sleep between pings to the defined target domain controller. The default value is 300 seconds.
 - **Timeout Before Job Failure** – The number of minutes to wait after Credential Cache job is downloaded by the agent before marking the job a failure due to timeout. The default value is 180 minutes.
 - **Timeout for User Credential Prompt** – The number of minutes to prompt the user with a dialog box to enter their target domain credentials for caching. The default value is 5 minutes
4. Check the **Hide Cancel/Close/Taskbar Icons** options to stop the icons from appearing.
5. Click **Save Profile**. The Credential Cache Profile is added to the list.

Migration Waves

What is a Migration Wave?

A Migration Wave in Active Directory is a named logical grouping of Devices. This can be a useful tool for organizing, tracking, and staging your migrations.

How do I manage Migration Waves?

There are two ways to manage Migration Waves in Active Directory. First, you can add Devices to a Wave by applying the 'Set to Migration Wave' Action to Devices from the Ready Devices table. The other way to manager Migration Waves is from the Migration Waves page which is accessible through 'Waves' in the left navigation menu. On that page you can create new Waves, Edit Wave names, Remove empty waves, and view how many Devices are in a Wave.

What can I do with a Migration Wave?

Migrations Waves in Active Directory can be used to filter the Ready Devices table to view Device status and apply Actions.

How do I create a New Migration Wave?

1. Login to Active Directory.
2. Select "Waves" in the left navigation menu.
3. A new page will open listing your current migration waves.
4. Click the Add icon.
5. Name the Migration Wave, remember use a logical name representing the migration event.
6. Click "Save."
7. Now that the Migration Wave is created, Devices can be added to the Wave by applying the 'Set to Migration Wave' Action to Devices from the Ready Devices table.

How do I remove a Migration Wave?

1. Login to Active Directory.
2. Select "Waves" in the left navigation menu.
3. A new page will open listing your current migration waves.
4. Select one or more wave in the table.
5. Click the Remove icon.
6. Click "Yes" to confirm the removal.

How do I edit the name of a Migration Wave?

1. Login to Active Directory.
2. Select "Waves" in the left navigation menu.
3. A new page will open listing your current migration waves.
4. Select a wave in the table.
5. Click the Edit icon.
6. Edit the name of the Migration Wave.
7. Click "Save."

How do I filter Devices by Migration Wave?

1. Login to Active Directory.
2. Select "Devices + Servers" in the left navigation menu.

3. On the Ready Devices tab, click the Filter icon.
4. Select one or more wave under the Waves filter category.

Configurations

Downloads

How are device agents downloaded?

To download a device agent:

1. Select an available agent version from the drop-down menu.
2. Click the **Download** button.
3. Use the browser options to save the agent installer package.

What are the Device Agent Service URL and Auth key used for?

The Device Agent Service URL and Auth Key as defined on the Downloads section of the Configurations page are provided to the Device Agents at install and allow them to connect to the correct customer's On Demand Migration project. They are unique to the agents in a given client and all agents of the same client should use the same values. If installing the agent from the command line without UI the arguments for providing the Service URL and Auth Key are their names in all uppercase i.e. SERVICEURL and AUTHKEY respectively.

How are device agents automatically upgraded?

To automatically upgrade the device agents:

1. Click the **Enable** button at the bottom of the Device Agent section.

Installing the Active Directory Agent

Each Active Directory Computer (device) that will be migrated must have an agent installed on the workstation to orchestrate local jobs that must occur to prepare and execute the workstation's domain move.

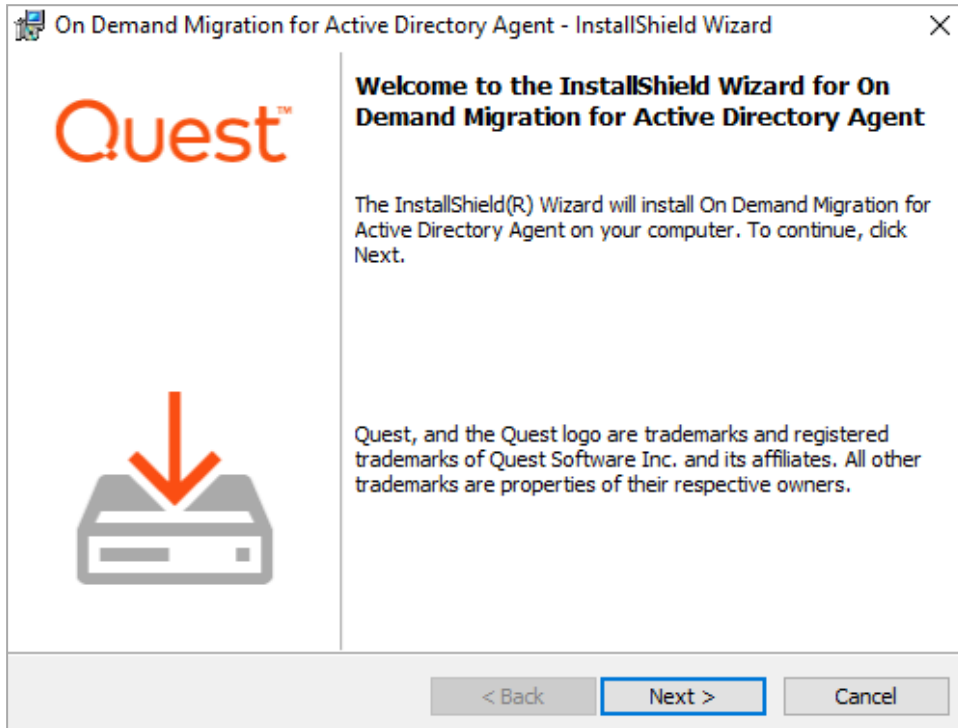
Refer to the [Requirements](#) for to verify all devices meet the requirements for agent installation.

The agent is available as an MSI package from the Downloads section of the Configurations page. You will also need the values of the Service URL and Auth Key found on that page.

You can install the agent by running the MSI manually on the device, with a PowerShell command, or in bulk by using a GPO or other third-party delivery method.

How do you manually install the Active Directory Agent?

1. Download the Active Directory MSI file from the Downloads page.
2. Copy the Active Directory MSI file to each computer.
3. Double-click the file to open the installer.
4. On the Welcome screen, click **Next**.



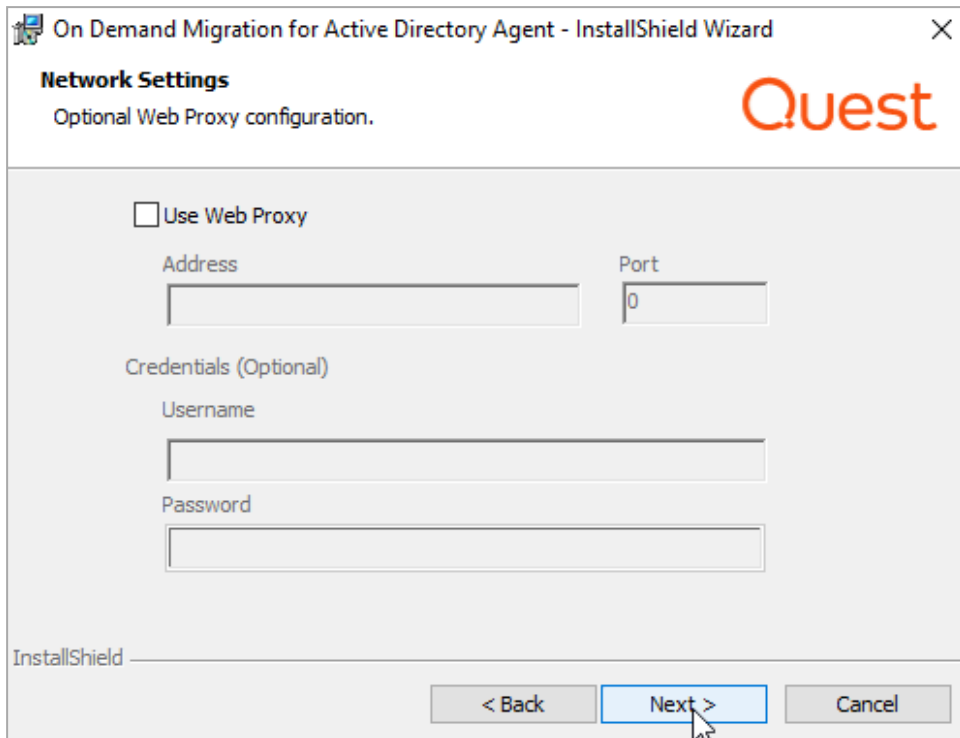
5. On the **License Agreement** screen, accept the agreement and click **Next**.

The screenshot shows a window titled "On Demand Migration for Active Directory Agent - InstallShield Wizard". The main heading is "License Agreement" with the Quest logo in the top right. Below the heading, it says "Please read the following license agreement carefully." The central area contains a scrollable text box titled "Software Transaction Agreement" with the following text: "PLEASE READ THIS AGREEMENT CAREFULLY BEFORE USING THIS PRODUCT. BY DOWNLOADING, INSTALLING OR USING THIS PRODUCT, YOU ACCEPT AND AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT. FOR ORDERS PLACED OUTSIDE THE UNITED STATES OF AMERICA, PLEASE GO TO http://quest.com/legal/sta.aspx TO VIEW THE APPLICABLE VERSION OF THIS AGREEMENT FOR YOUR REGION. IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT OR THE APPLICABLE VERSION OF THIS AGREEMENT FOR YOUR REGION, DO NOT DOWNLOAD, INSTALL OR USE THIS PRODUCT. IF YOU HAVE A SIGNED AGREEMENT WITH PROVIDER THAT IS SPECIFICALLY REFERENCED IN AN ORDER THAT IS EXECUTED BETWEEN YOU AND PROVIDER, THEN THAT SIGNED AGREEMENT WILL SUPERSEDE THIS AGREEMENT." Below the text are two radio buttons: "I accept the terms in the license agreement" (which is selected) and "I do not accept the terms in the license agreement". A "Print" button is to the right of the radio buttons. At the bottom, there are three buttons: "< Back", "Next >" (highlighted with a blue border), and "Cancel". The "InstallShield" logo is in the bottom left corner.

6. On the **Agent Registration** screen, enter the **Service URL** and **Authorization Key**, found on the Downloads page, and then click **Next**.

The screenshot shows a window titled "On Demand Migration for Active Directory Agent - InstallShield Wizard". The main heading is "Agent Registration" with the Quest logo in the top right. Below the heading, it says "Enter URL and Authorization Key". The central area contains two input fields: "Service URL:" with the text "https://us.odmad.quest-on-demand.com/api/ADM" and "Authorization Key:" with a blurred key. At the bottom, there are three buttons: "< Back", "Next >" (highlighted with a blue border), and "Cancel". The "InstallShield" logo is in the bottom left corner.

- On the **Network Settings** screen, if using a Web Proxy, check **Use Web Proxy** and enter the Web Proxy settings. Click **Next**.



- On the **Ready to Install the Program** screen, click **Install**.
- When the install completes, click **Finish**.



Note: Once the agent is installed and the service is running it will connect to the server within four hours. This delay is randomized and uniformly distributed to avoid overloading the server when large numbers of agents come online at the same time.

How do you install the Active Directory Agent using a PowerShell Command?

- Download the Active Directory MSI file from the Downloads page. The Service URL and Auth Key values also found on the Downloads page are required.
- Create and run the PowerShell command with the required SERVICEURL (Service URL) and AUTHKEY (Auth Key) values.

Example:

```
msiexec.exe /I 'C:\workspace\AD.Agent-20.3.1.1401.msi'
SERVICEURL=https://us.odmad.quest-on-demand.com/api/ADM
AUTHKEY
#####
```

3. Walk through the install wizard, filling out the needed information and click Finish when completed. The settings for using a customer web proxy for communications are optional.

As needed the installer can also be invoked in quiet mode with the /QN switch (requires running PowerShell as admin).

Additionally, it is possible to configure the agent to use a Web Proxy using the below command line arguments:

- **WEBPROXYENABLE** – Is a Web Proxy used? Values: Yes=1, No=0
- **WEBPROXYURL** – The Web Proxy Address
- **WEBPROXYPORT** – The Web Proxy Port
- **WEBPROXYUSER** – The optional Web Proxy Credentials Username
- **WEBPROXYPASS** – The optional Web Proxy Credentials Password

How do you install the Active Directory Agent using a GPO (Group Policy Object)?

1. To install the agent using a GPO you must convert the MSI package and the parameters into an MST file. One method to do this is using Microsoft Orca. Install Orca (available in [Windows SDK Components for Windows Installer Developers](#)). Orca will be used to create the necessary MST file.
2. Download the Active Directory Agent MSI file from the Downloads page.
3. Right-click on the MSI file and select **Edit with Orca**.
4. Once you have Orca opened, click on the **Transform** menu and select **New Transform**.

5. Next, navigate to the **Property** table and add the following:
 - Add a Row with property of **SERVICEURL** and the Service URL value found on the Downloads page.
 - Add a Row with property of **AUTHKEY** and the Auth Key value found on the Downloads page.
 - - Optionally, the following properties and values can also be added to configure the agent to use a Web Proxy:
 - **WEBPROXYENABLE** – Is a Web Proxy used? Values: Yes=1, No=0
 - **WEBPROXYURL** – The Web Proxy Address
 - **WEBPROXYPORT** – The Web Proxy Port
 - **WEBPROXYUSER** – The optional Web Proxy Credentials Username
 - **WEBPROXYPASS** – The optional Web Proxy Credentials Password
6. Click the **Transform** menu and select **Generate Transform** to complete the MST file creation. This MST file will be used in a later step.
7. Right-click on the Active Directory Agent MSI, point to **Share with**, and click on **specific people**.
8. Add a security group. The "authenticated users" group already includes all computers and is a good group to use. The group you add must have the shared Read permission and NTFS permission.
9. Click **Share**.
10. Click **Done**.
11. From the **Start** menu, point to **Administrative Tools** and click on **Group Policy Management**.
12. Right-click on the domain or OU you will be migrating and click on **Create a GPO in this domain, and link it here**.
13. In the New GPO dialog box, enter a **Name** for the GPO and click **OK**.
14. Click on the new GPO and click **OK**.
15. Right-click on the GPO and select **Edit**.
16. Open **Computer Configuration > Policies > Software Settings** and right-click on **Software Installation** and then point to **New** and click on **Package**.
17. In the **File Name** field, enter the UNC path to the MSI file and click **Open**.
18. Select the Active Directory Agent MSI file and click **Open**.
19. In the **Deploy Software** window, select the **Advanced** deployment method and click **OK**.
20. Under the **Modifications** tab, add the MST file you created earlier and click **OK**.



Please Note: The computer must be rebooted for the applied group policy to complete the agent installation.

How do you verify the GPO?

1. Log on to a workstation within the scope of the GPO using administrator credentials.
2. From a command prompt on the workstation, run **gpresult -r**
3. The Computer Settings section will display the applied group policy.



Please Note: A newly applied group policy will not immediately be displayed.

The Computer Settings section displays the applied group policy, but the agent installation is not completed until the computer is rebooted.



Please Note: If using the agent Auto-Upgrade feature and deployment software that uses MSI ProductCode based detection, the Auto-upgrade feature should be disabled after initial deployment or the detection method should verify via a folder path.

Repositories

What are the Repositories?

Repositories are storage locations (network shares) which you configure on your network used for four specific job types: Agent Logs, Custom Downloads, Offline Domain Join, and Microsoft Entra Device Join. These job types access or create files in the defined locations. If you are not using these job types you do not need to configure Repositories.

How do I manage Repositories?

Repositories are managed from the Repositories section of the Configurations page. There you can view and copy the currently defined share path for each job type. You can update and save the configuration. Make sure that the defined network share path is routable from the devices being migrated.

What can I do with a Repository?

Repositories are used to locate files for four job types:

- **Agent Logs** – Agent logs are maintained on individual workstations. To centralize them so that review does not require logging into each machine, the Upload Logs Action can be applied to the Devices of interest. This will create a copy of the Device's logs in the defined network share.
- **Custom Downloads** – Custom Actions with a Download File Task can be used to download a specified file from the defined network share to the workstation. A common use of this job type is distributing a new VPN client to workstations after Cutover.
A Custom Download Repository should not be used with the Offline Domain Join Action type when configuring custom actions.
- **Offline Domain Join** – The Offline Domain Join Action and the associated Cache Credentials Action rely on access to ODJ files which have been generated by a client administrator prior to applying the Actions according to Microsoft instructions as described elsewhere in this guide.
An Offline Domain Join Repository should be used with the Offline Domain Join Action type when configuring custom actions.
- **Microsoft Entra Bulk Enrollment** – The Microsoft Entra ID Cutover Action relies on access to an Azure bulk enrollment package which has been created by a client administrator using the Windows Configuration Designer as described in the [Microsoft Entra Device Join Quick Start Guide](#).

Product Licensing

The product licensing is based on the number of **unique source accounts** per configured **environment workflow** processed by Active Directory Express. The licenses are consumed when the Active Directory Workflow creates or updates the target accounts .

The following object types do not consume any license,


- Distribution Group
- Security Group
- Mail Enabled Security Group
- Teams
- M365 Group mailboxes
- Contacts
- Computer accounts
- Guest

Migrate and Navigate

Devices + Servers

What is the Devices + Servers page used for?

The Servers + Devices screen allows the administrator to register devices and servers, set the ReACL profile, upload migration logs, and manage the device Discovery, ReACL, Cutover, and Cleanup processes.

Select the gear icon  above the "Ready Devices" list to select the columns to display for the current session.

What is the difference between a "Ready Device" and a "Not Ready Device"?

"Ready Devices" are devices that have the necessary agent installed, are communicating, and are ready for actions to be scheduled. "Not Ready Devices" are devices which that have not yet had an agent installed and communicating.

Note: Initial agent registration is uniformly distributed over a four hour interval from agent start time.

What actions can be performed on Devices and Servers?

The following actions can be performed on devices by selecting the devices in the list, selecting an action from the drop-down list, and then clicking the **Apply Action** button.

- **Discovery**

The Discovery process gathers properties (OS versions, network properties, and so on) from the device to allow additional future functionality. The first discovery process begins for a device when the device becomes registered with On Demand Migration which will automatically occur after the Device Agent has been installed, as long as the environment is properly configured.

To select when the process will begin check **Do not start before** and then enter or select a date and time. If using the **Do not start before** option, the Discovery Status will be displayed as Queued in the Devices table and the "Do Not Start Before" column in the Device Jobs table will be populated with the selected date.

- **Set Target Environment**

The Set Target Environment action provides the ability to specify the target environment for the selected devices. The ReACL, Cache Credential, Offline Domain Join, Cutover, Cleanup, Rollback, and ReACL Rollback actions cannot be started for the selected devices if the target environment is not set.

To set to the target environment, select a target environment for the selected devices from the list and click **Save**. The target environment can be cleared by selecting **None** from the list.

- **ReACL**

The ReACL process updates the Device's domain user profiles for use by the matching target user after cutover.

Note: It is recommended to remove or disable anti-virus software immediately prior to the ReACL process and only after a recent clean scan has been completed.

Note: At least one group must be migrated to populate the map.gg file or the ReACL process will fail.

Before ReACL can occur, the target Users and Groups which have permissions set on the Device must be migrated to the target.

To select when the process will begin check **Do not start before** and then enter or select a date and time. If using the **Do not start before** option, the ReACL Status will be displayed as Queued in the Devices table and the "Do Not Start Before" column in the Device Jobs table will be populated with the selected date.

Note: Two checks are performed at the start of the ReACL process. The first check is for invalid Source Profiles, which will be logged as a WARNING and those profiles will be skipped. The second check is for invalid Target Profiles, where a user may have created a profile with the target account before their machine is ReACL'd and cutover. By default, this is logged as a FATAL ERROR and will halt the ReACL process. However, it can be changed to a WARNING with the `-t` switch passed by editing the command in SQL.

The ReACL Agent will automatically create two files on the device being ReACL'd, map.user and map.gg. These files are used to find the source permissions and add the appropriate target permissions during the ReACL process. System groups, such as Domain\Domain Admins and Domain\Domain Users are included in the map.gg file for updating the group permissions during the ReACL process. If the Active Directory environment is non-English, the values in the sAMAccountName column of the BT_SystemGroup table in the SQL database will need to be changed after Directory Sync is installed to have the appropriate non-English values.

If the Mapped Network Drive is being mapped via GPO or using an integrated credential such as the current Windows logon session, ReACL will create a warning entry in the log "...WARNING: The UserName value for drive U was empty and could not be mapped to the target user." This warning does not mean that the mapped drive cannot be accessed after Cutover.

Note: The user profile ReACL process is decoupled from actions against files and folders.

ReACL will update all files and folders entries found on the machine except for the user profile folders, ntuser.dat, and usrclass.dat even if the user profiles option is selected in the ReACL profile.

Remaining ReACL activities against user profiles and registry are processed by a separate ReACL task when a cutover operation is performed.

- **Cache Credential**

The Cache Credentials process assigns a Cache Credentials job to workstation(s). See the [Credential Cache and Offline Domain Join](#) topic for more information.

- **Offline Domain Join**

The Offline Domain Join process is similar to the Cutover process for machines that are directly connected to the network. See the [Credential Cache and Offline Domain Join](#) topic for more information.

Warning: Do not perform the Cutover process on Offline Domain Join workstations. The Offline Domain Join process takes the place of Cutover for workstations connecting via VPN.

- **Cutover**

The Cutover process moves a Device from the source domain to the new target domain.

Check **Ignore ReACL Status** to cutover the device regardless of the ReACL status (otherwise the cutover process will not proceed if there is an error with ReACL process).

Check **Do not start before** and then enter or select a date and time when the process will begin. If using the **Do not start before** option, the Cutover Status will be displayed as Queued in the devices table and the "Do Not Start Before" column in the Device Jobs table will be populated with the selected date. The Cutover process will begin as soon as possible if not using this option.

Note: Devices should not be ReACL'd once they have been cutover to the Target. This is not a best practice and is not supported as this can cause problems with the registry and user profiles.

Note: Certificates are not migrated with Device Cutover.

- **Microsoft Entra ID Cutover**

The Microsoft Entra ID Cutover process moves a Device from the source domain to an Azure target environment.

Check **Ignore ReACL Status** to cutover the device regardless of the ReACL status (otherwise the cutover process will not proceed if there is an error with ReACL process).

Check **Do not start before** and then enter or select a date and time when the process will begin. If using the **Do not start before** option, the Cutover Status will be displayed as Queued in the devices table and the "Do Not Start Before" column in the Device Jobs table will be populated with the selected date. The Cutover process will begin as soon as possible if not using this option.

Note: Devices should not be ReACL'd once they have been cutover to the target. This is not a best practice and is not supported as this can cause problems with the registry and user profiles.

Note: If the Entra ID Join profile has the **Auto-Pilot/Intune Cleanup** option selected, the Autopilot remove status must be completed before the Microsoft Entra ID Cutover action can be used.

- **AutoPilot Cleanup**

The AutoPilot Cleanup action clears existing Auto-Pilot/Intune provisioning information from the device. This must be done before the cutover if the source Entra ID Joined device is Autopilot-provisioned and the Entra ID Join Profile has the Autopilot/Intune Cleanup option selected.

- **Set Intune Primary User**

The Set Intune Primary User action is used after the Device Cutover is completed to set the primary user.

- **Cleanup**

The Cleanup process removes the Source SIDs after the Cutover process completes.

Note: Cleanup should be done when the migration project is completed. Before running the Cleanup process if a trust is in place, the trust can be broken to test if any application permissions are broken.

In the Job Options window, click **Apply** to begin the Cleanup process as soon as possible. To select when the process will begin check **Do not start before** and then enter or select a date and time. If using the **Do not start before** option, the Cleanup Status will be displayed as Queued in the Devices table and the "Do Not Start Before" column in the Device Jobs table will be populated with the selected date.

- **Upload Logs**

Log files from the Active Directory Agent can be uploaded to the Active Directory Web Server using Microsoft BITS. To enable this functionality, the installer enables BITS Server Extensions for IIS and create a virtual directory called **DeviceLogs** where all uploaded files will be stored.

In the Job Options window, click **Apply** to begin the Upload Logs process as soon as possible. To select when the process will begin check **Do not start before** and then enter or select a date and time. If using the **Do not start before** option, the Do Not Start Before column in the Device Jobs table will be populated with the selected date.

The logs will be stored in the configured Agent Logs Repository

The device logs will be zipped, and the file names will be in the following format with a unique file name: SMART-WIN7X86-1_201573111235.zip

- **Rollback**

The Rollback process moves a Device back to the original source domain and restores any modified network settings. The Device must have attempted Cutover for this explicit Rollback process to work.

In the Job Options window, click **Apply** to begin the Rollback process as soon as possible. To select when the process will begin check **Do not start before** and then enter or select a date and time. If using the **Do not start before** option, the "Do Not Start Before" column in the Device Jobs table will be populated with the selected date.

Note: Rollback is not supported for Entra ID Device Cutover.

- **ReACL Rollback**

The ReACL Rollback process rolls back all changes made by the ReACL process. ReACL Rollback can be performed on Devices that have completed the ReACL process.

In the Job Options window, click **Apply** to begin the ReACL Rollback process as soon as possible. To select when the process will begin check **Do not start before** and then enter or select a date and time when the process will begin. If using the **Do not start before** option, the "Do Not Start Before" column in the Device Jobs table will be populated with the selected date.

- **Status Resets**

Use the Status Resets action to reset the statuses of the selected devices.

- **Set Device ReACL Profile**
Use the Set Device ReACL Profile action to assign a Device ReACL profile to the selected devices.
- **Set Migration Wave**
Use the Set Migration Wave action to set a migration wave to the selected devices.
- **View Jobs**
Use the View Jobs action to view the device jobs of the selected devices.
Note: Jobs can be canceled when the Status or Rollback Status is either Queued, Scheduled, Started, or In Progress.
- **View Profiles**
Use the View Profiles action to view the profiles of the selected devices.
- **View Properties**
After the Discovery process has been completed for a Device, you can view the properties of that Device.
Click the **Export All** button to export the content of the window in Excel, text, CSV, or HTML format.

Offline Domain Join (ODJ)

Normally, right after a Device is Cutover to a new domain VPN users can't log in to their workstation because Windows must be able to contact the target domain to authenticate against a domain controller for that very first login. Typically, a remote user not on the VPN would need to log in to their machine first and then establish a VPN connection.

Active Directory answers this problem by building on Microsoft's Offline Domain Join (ODJ) process to allow a workstation to join a domain without contacting a Domain Controller. This solution is achieved by first creating an ODJ file for each workstation and then taking advantage of Windows' ability to cache credentials. If users have logged in to the target domain previously, Windows can log them in again even if they can no longer reach a domain controller by using cached credentials.

Active Directory's ODJ process has users pre-login to the new domain before the computer needs to be Cutover so the target credentials can be cached and used for the first post-Cutover login without the need to contact a domain controller first. Then when the administrator is ready to Cutover workstations using the ODJ Action, Active Directory allows the workstation to join the new domain without having the user connect to the corporate VPN and manually join their workstation to the new domain.

The computers that the ODJ process is being run on must have network connectivity to BOTH the source and target environments at the same time sometime pre-cutover in order to have the Cache Credentials function work properly.

Additionally, computers will be unable to save the cached credentials unless the source environment trusts the target environment. The [AD Offline Domain Join Credential Cache Quick Start Guide](#) provides guidance on configuring an AD trust to enable Offline Domain Join functionality.

1. CREATING ODJ FILES FOR EACH WORKSTATION

The first step in the ODJ process is for an administrator to use Microsoft's DJOIN utility to create a provisioning file. Only the provisioning part of the DJOIN process is needed. Complete information on DJOIN can be found [here](#).

The Provision, Domain, Machine, and Savefile parameters are required at a minimum. There is the option to control where the target machine will be created using the MachineOU parameter as in the sample shown here.

```
DJOIN.EXE /Provision /Domain BTADLAB.com /Machine Sales220 /Savefile "\\server\odj-share\Sales220.txt"  
/MACHINEOU OU=SalesComputers,OU=Sales,DC=BTADLAB,DC=COM
```

The file must be saved in the ODJ folder in the Repositories path that was configured in the UI.

Generating these files can be completed for all in scope workstations early in the migration process.

WARNING: Be sure to name each text file with the exact matching machine name.

2. CONFIGURING THE CREDENTIAL CACHE PROFILE

The next step is to configure the existing Default Credential Cache profile with the IP address of a Target domain controller, or to create a new profile for this setting.

From the Profiles page select Credential Cache Profiles. Click on **Add** to create a new profile, or **Edit** to modify an existing profile. If you choose to use the Default profile, you must edit it to include a Target DC IP address.

- The **Target Domain Controller Ping Interval** setting determines how long the script will sleep before pinging the DC again.
- The **Timeout Before Job Failure** setting determines the Credential Cache app timeout value that will be used for the job once downloaded to the agent managed machine.
- The **Timeout For User Credential** setting determines how long the user is presented with a dialog box to enter their target domain credentials.

3. CACHE CREDENTIALS JOBS

Now that a profile has been configured with a target DC IP address, we can assign a Cache Credentials job to each in scope workstation.

In the Devices list, select one or more Computers. Select **Cache Credentials** from the Actions dropdown menu and click the **Apply Action** button. The Credential Cache Options box appears.

Select a Credential Cache Profile.

A date and time for the Cache Credentials job can be chosen to run the job at a later time. This date/time combination represents the earliest time that this job could run.

If a date/time is not chosen, this job will run on the workstation the next time the agent checks for jobs.

The Devices list will reflect a status of "Queued". When the job is collected by the agent the status will change to "In Progress", and then finally it will transition to a status of "Completed" or "Failed".

On the workstation side, when the Cache Credentials Job is received, the user will be prompted to enter their target credentials. Below is an example of what the user will see when the Cache Credential job runs:

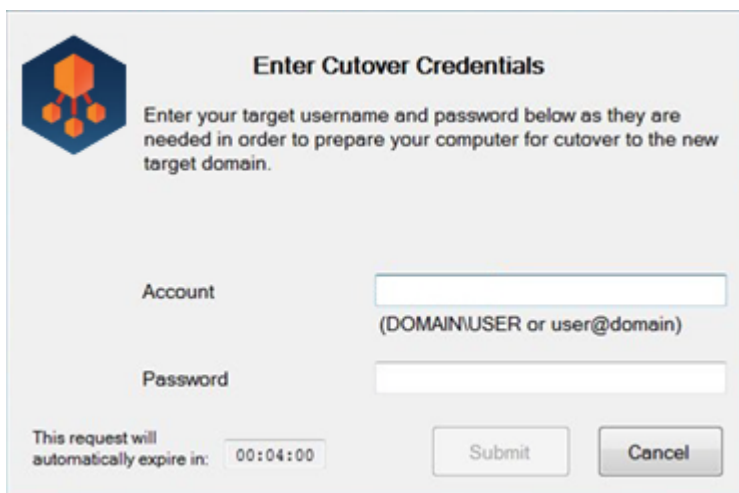


Figure 1: Enter Cutover Credentials

4. REACL

Following Cache Credentials the next recommended step is the ReACL process. The ReACL process can be run repeatedly as needed before ODJ, but it is suggested to be run at least once right after the Cache Credentials process is run.

In the Devices list, select one or more Computers. Select ReACL from the Actions dropdown menu and click the Apply Action button.

The Job Options box appears. A specific date/time combination can be chosen for when to run the job, or just click Apply Action to have this job received by the workstations during their next check for jobs.

5. OFFLINE DOMAIN JOIN JOB

The final step is the actual Offline Domain Join job itself. This is similar to the Cutover process used for machines that are directly connected to the network.

WARNING: Do not perform the Cutover process on Offline Domain Join workstations. The Offline Domain Join process takes the place of Cutover for workstations connecting via VPN.

In the Devices list, select one or more Computers. Select **Offline Domain Join** from the Actions dropdown menu and click the **Apply Action** button.

The Job Options box appears. A specific date/time combination can be chosen for when to run the job, or just click **Apply Action** to have this job received by the workstations during their next check for jobs.

WARNING: The Offline Domain Join (Job Scheduling Options dialog box) start date and time must be set AFTER the Cache Credentials job (Cache Credential Options dialog box) start date and time.

The Offline Domain Join process does not support rollback.

Migration for Active Directory Express FAQs

What ports does the agent use to connect?

The Active Directory Agent uses three outbound ports

- TCP 443/80
- UDP 3030

I've installed the agent and the device isn't ready, what do I do?

On startup the Active Directory agent will phone home sometime in the first four hours. This communication offset time is an agent specific random and evenly distributed offset so the initial communications of a large number of devices set up at once will be spaced out and not overload the servers. If your device has the agent installed but you haven't waited up to four hours yet, wait up to four hours and then re-check the Ready Devices list to see if your device shows up before proceeding to other network troubleshooting operations.

How do I adjust the agent polling interval?

In Migration for Active Directory Express, the agent polling interval is not end user adjustable. The polling interval is every two minutes over UDP and every four hours over TCP. Should this amount of network traffic be an issue contact Support to investigate reducing the polling interval. Note that reducing the polling interval will cause queued migration actions to take longer to be picked up by the agents.

How many migration actions can I queue at once?

In Migration for Active Directory Express you can queue as many migration actions at once as you would like. There is a hard limit of 600 agents per each two minutes who will be notified by the job availability cache that they have a job waiting for them. Therefore the minimum amount of time which could be required for those queued migration actions to be picked up will be the number of queued migration actions divided by 600 and multiplied by two minutes.

Agent last contact time isn't updating every two minutes is something wrong?

In Migration for Active Directory Express, the Agent last contact time is updated based on the TCP connection with the back end. This will occur if a job is retrieved or every 4 hours as a fall back. It is not expected that the agent last contact time will update every two minutes even if the UDP requests to the job availability cache are functioning correctly.

The machine name was changed during the migration project, will it keep working?

Changing the machine name during a migration project may have unexpected implications. We recommend that you avoid doing so if possible. If not, you will need to delete the device from Migration for Active Directory Express and reinstall the agent. Use the Status Resets action to reset the registration status of the device, then set the Workflow to reconcile on next run and run it. This will delete the old device from the database. Finally, reinstall the Active Directory Agent on that device and run the Workflow again to read it in.

How do I remove Devices from Migration for Active Directory Express?

To delete registered Devices from Migration for Active Directory Express, first use the Status Resets action to clear their registration status, then run the related Workflow with the setting 'Reconcile on next run' enabled. Any unregistered device records in scope of the Workflow will be removed from the database during a reconcile.

Can I migrate to and from GCC/GCCH tenants?

Active Directory supports GCC and GCCH as target environments, in addition to Commercial tenants. Only Commercial and GCC are supported as source environments. To enable GCC/GCCH Device Migration, please contact Quest Support for additional configuration described in the following article: [Update GCC/GCCH tenant object's RID value for ReACL](#). This configuration is required for successfully switching the user profile during cutover.

Migration for Active Directory Express Architecture

The first step towards success on a project using Migration for Active Directory Express is to understand the product architecture and how this architecture will operate in your environment.

Migration for Active Directory Express consists of the following components:

- A directory synchronization engine
- A REST based web service
- A management interface
- A lightweight agent for workstations and member servers

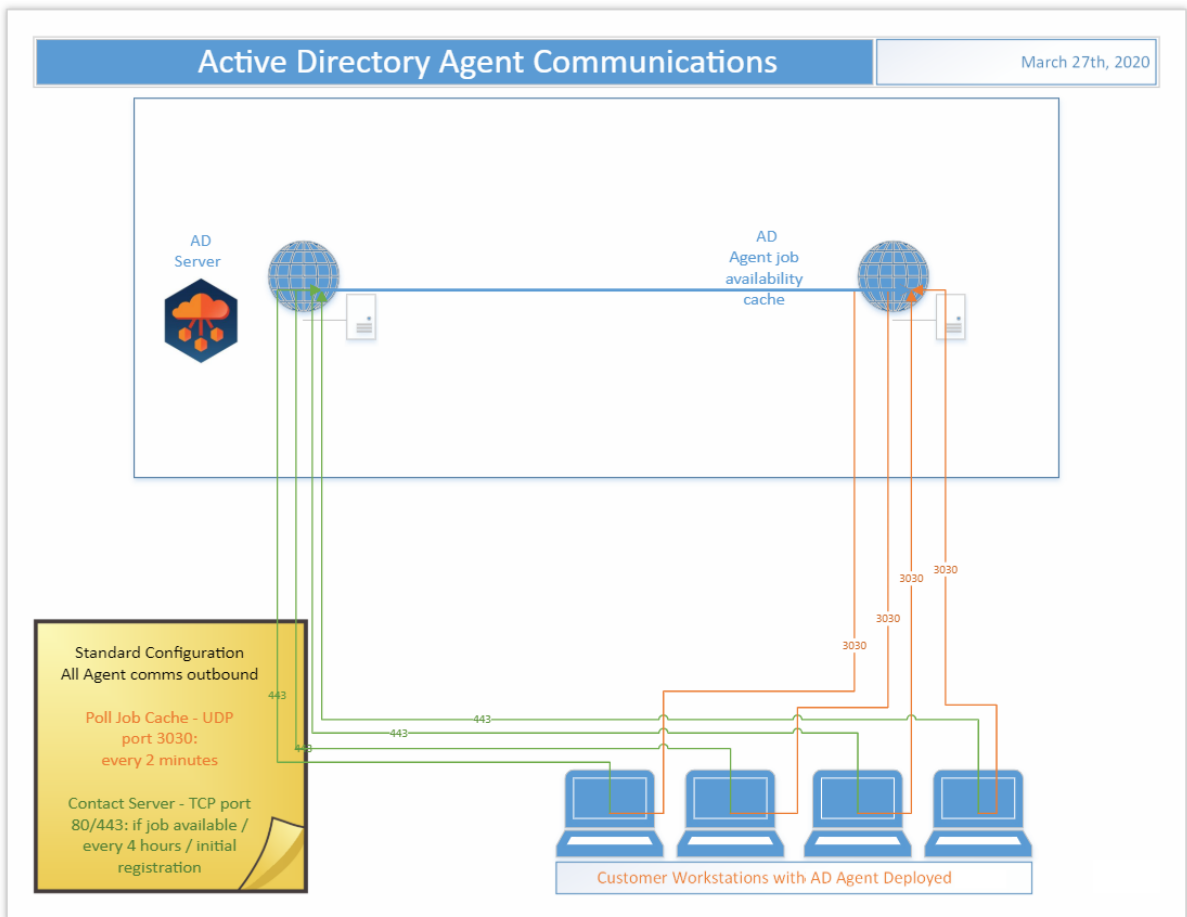
The directory synchronization engine, the web service, and the management interface will all access the same SQL database. In most scenarios, these components will be installed on the same system. In larger or more complex network environments, the components can be distributed across multiple systems.

User workstations, member servers and computers are collectively referred to as Devices in Migration for Active Directory Express. Computers communicate with the Migration for Active Directory Express web service using the Active Directory Agent. The Active Directory Agent is a lightweight application that installs as a service on Windows computers.

To ensure that no firewall exceptions are required, the web service does not “call” the Devices to be migrated. Instead, the Active Directory Agents contact the web service at defined polling intervals, using standard HTTPS or HTTP requests to collect jobs. Jobs include key tasks such as system discovery, updating the operating system, file system, and user profile permissions, and migrating the computer to the new domain.

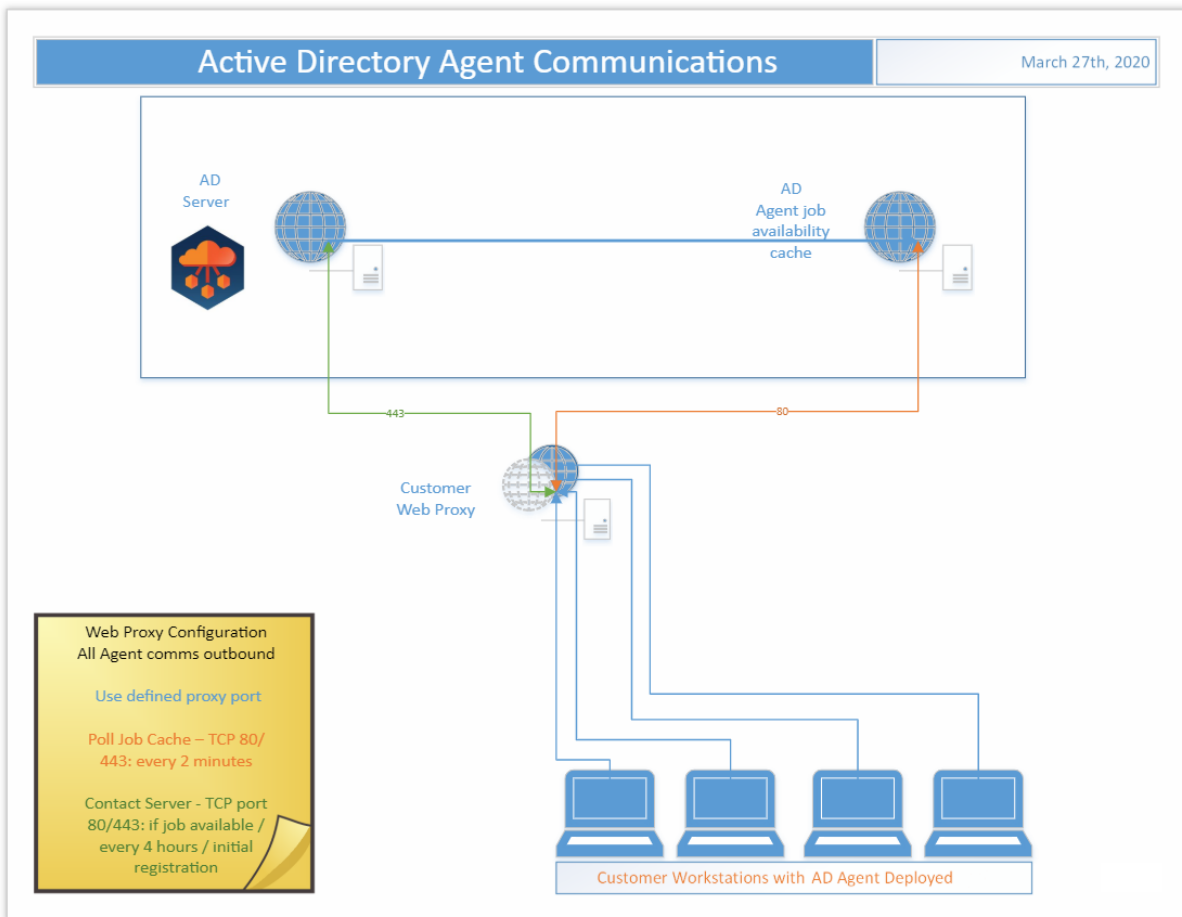
Standard Configuration

In the Standard Configuration for Migration for Active Directory Express, the Agent is deployed to each Device to be migrated. Those Agents communicate outbound to the Active Directory webserver in Azure over ports 80/443 every 4 hours, when a job is available, or when initially registering. They also communicate outbound to the Active Directory Agent job availability cache in Azure over UDP on port 3030 every 2 minutes.



Web Proxy Configuration

In the Web Proxy Configuration for Migration for Active Directory Express, the Agent is deployed to each Device to be migrated and use of a web proxy is enabled. Those Agents communicate outbound through the defined proxy port to the Active Directory webserver in Azure over port 443 every 4 hours, when a job is available, or when initially registering. They also communicate outbound through the defined proxy port to the Active Directory Agent job availability cache in Azure on port 80 every 2 minutes.



Troubleshooting

- Problem:** A workstation that has been successfully cutover no longer responds to any additional jobs, such as Cleanup.

Solution: If a workstation that has been successfully cutover now fails to respond to any additional jobs, such as Cleanup, check the Application event log. If you see a *"The remote name could not be resolved"* error, this most likely means that the SRV record for the Migration for Active Directory Express Server can no longer be resolved due to a DNS lookup failure.

If you cannot "Ping" the Migration for Active Directory Express server from any other machines in the target domain, then you will need to remedy this on a more global scale, such as creating a conditional forwarder on the target machines' current DNS server pointing to the appropriate location.

If you are able to "Ping" the Migration for Active Directory Express server, then check the Network Profile that was used during the Cutover to verify that the DNS settings were correct in that profile.

Cutover Job Result Codes

Result Code	Error	Rollback Possible
1	Unidentified Error - PowerShell Command Error	No
2	Source Domain could not be contacted	No
4	Bad Source Credentials	No
8	Target Domain could not be contacted	No
16	Bad Target Credentials	No
32	Target DNS Server could not be contacted or could not resolve the target DNS domain	No
64	Change Obtain DNS by DHCP	
128	Set DNS Server IPs	
256	Set WINS Servers	
512	Register NIC with DNS	
1024	Clear DNS Suffix Search List / Set to use NIC	
2048	Set Alternate DNS Suffix List	
4096	Enable Dynamic DNS Registration	
8192	Set NIC Specific DNS Suffix	
16384	Domain Disjoin Failed	
32768	Domain Join Failed	
65536	Source domain name does not match the system's domain	No
131072	Computer Reboot failed	
262144	Target Domain Name could not be resolved via existing DNS, and new DNS Servers were not provided	No

Note: An odd numbered result code represents an error running the Cutover PowerShell script. The most common cause of an odd numbered result code during Cutover is that the computer either has no network card with a default gateway or more than one network card with a default gateway.

Note: Result codes are additive. There are likely multiple errors if the result code is not represented in the table.

Upload Logs Result Codes

This table includes result codes for BT-UploadLogs PowerShell jobs.

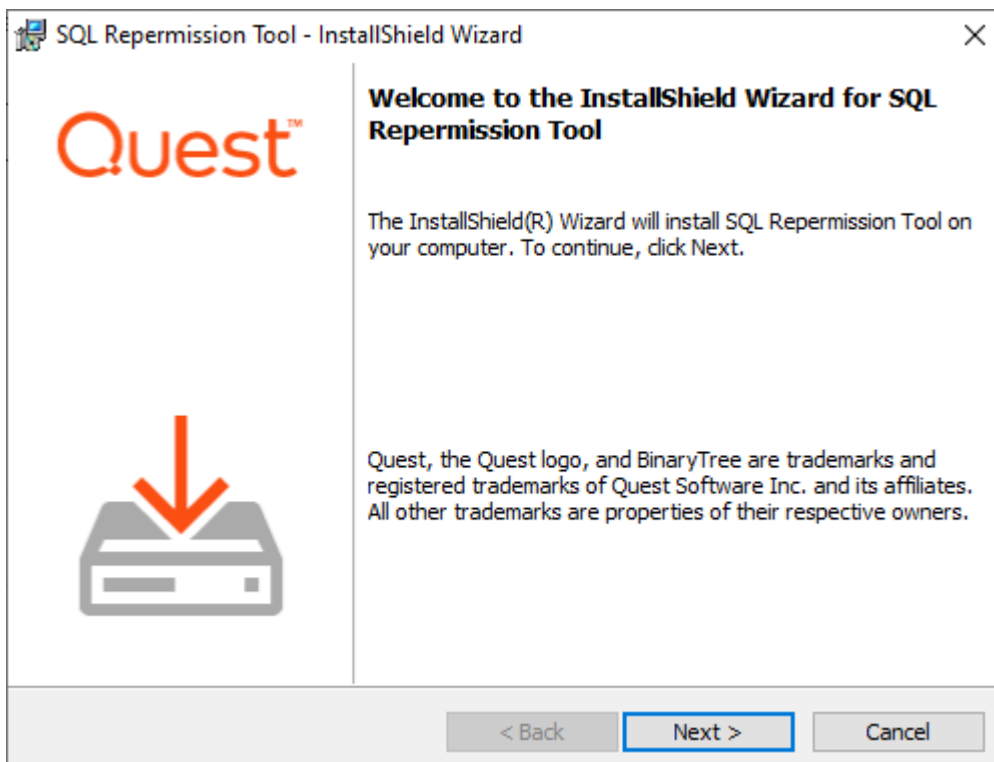
Result Code	Error	Rollback Possible
32	(zip folder) could not be created.	No
64	Failed to Zip log files on computer.	No
128	Upload failed to contact the server. Please verify the URL (url) is correct and BITS is enabled.	No

SQL Repermission Tool

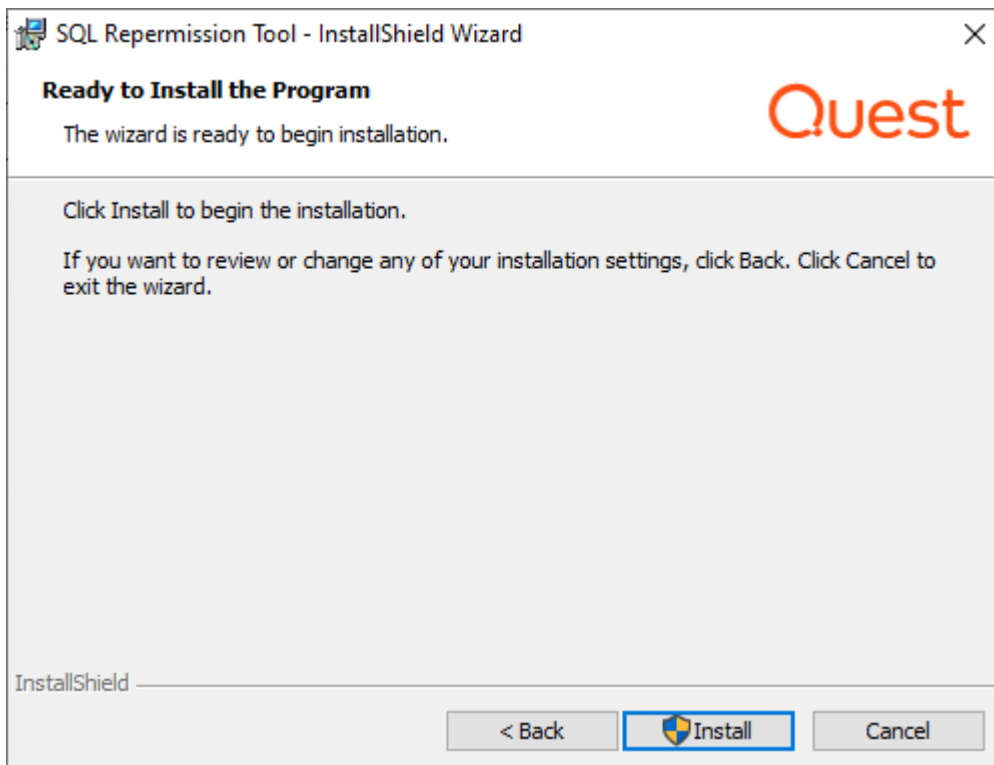
Use the SQL Repermission Tool to update a SQL Server's Source AD Windows Authentication Group Login permissions and their associated database users for the Target domain to which the user and group objects have been migrated.

Prior to re-permissioning SQL servers, accounts must be migrated and Mapping Files created.

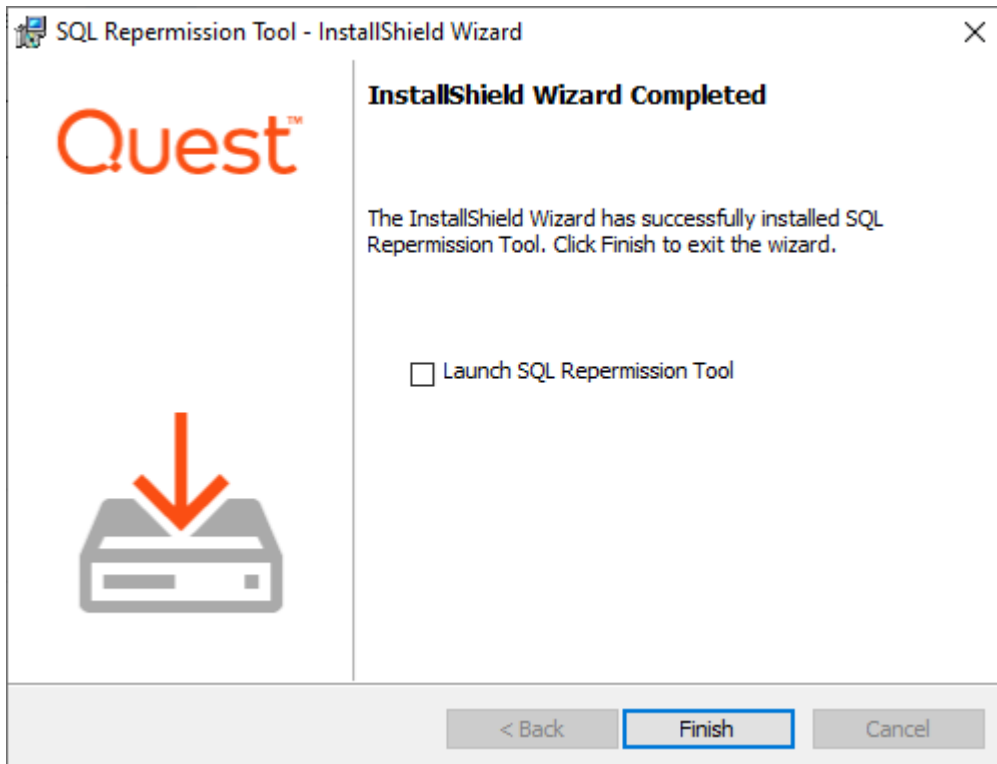
1. Download the **SQL Repermission.msi** file from the Downloads screen.
2. Run the installer on a machine that will have access to the SQL Server instance and databases to be re-permissioned.
3. The Welcome screen appears. Click **Next** to continue.



4. The "Ready to Install the Program" screen appears. Click **Install** to begin the installation.



5. When the installation completes, the "InstallShield Wizard Completed" message appears. Click **Finish** to close the wizard.



6. Copy the **map.usr** and **map.gg** files created to the SQL Repermission installation folder, default location is "C:\Program Files\Quest\SQL Repermission".

7. Launch the SQL Repermission tool from the Apps screen or the Start menu.

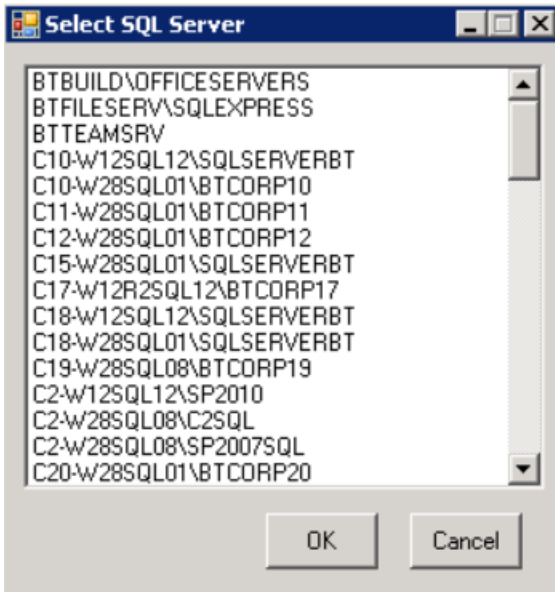
The screenshot shows the 'SQL Server RePermission' application window. It is divided into four main sections:

- 1. SQL Logon Method:** Contains two radio buttons. The first is 'Windows Authentication (use my currently logged on account)', which is selected. The second is 'SQL Server Authentication', which is unselected. Below these are two text input fields labeled 'Username' and 'Password'.
- 2. Select SQL Server:** Contains a text input field labeled 'Server\InstanceName' with the value 'localhost'. To the right of this field are two buttons: 'Browse...' and 'Connect'.
- 3. Select database to copy permissions from:** Contains a dropdown menu that is currently empty.
- 4. Generate SQL Script:** Contains a 'Generate SQL' button and a checked checkbox labeled 'Open script in Notepad'.

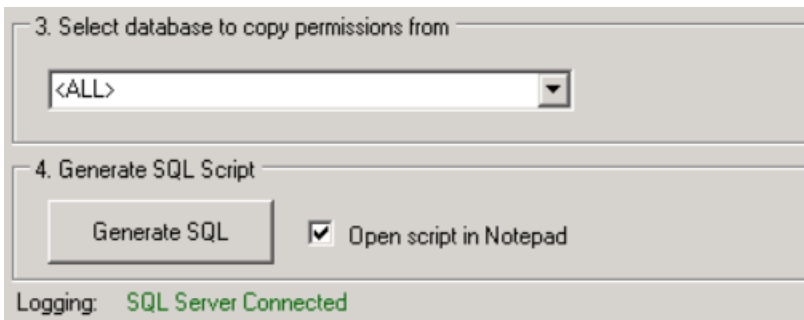
At the bottom of the window, there is a section labeled 'Logging:' which is currently empty.

8. Select the **SQL Logon Method**. If SQL Server Authentication is selected, enter the Username and Password. If Windows Authentication will be used, you must be logged onto the machine with an account that has access to the SQL Server instance and databases to be re-permissioned.
9. Click on **Browse** to select the SQL Server instance to be re-permissioned.

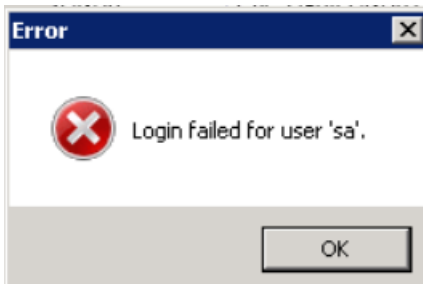
10. Select the SQL server instance from the list and click on **OK**.



11. Click on the **Connect** button. If connection is successful, the Logging message should read “SQL Server Connected” and there should be databases available to pick from in the drop-down list.



Note: If the login information is incorrect, the following error will be received. The credentials to connect to the SQL server instance should be corrected, and then the Connect should be retried.



12. Once successfully connected to the SQL Server instance, in the drop down select either **<ALL>** or a specific database that needs to be re-permissioned.
13. Click on the **Generate SQL** button. If the **Open script in Notepad** option is left checked, the results will be displayed when completed.

14. The Logging information will display the name of the SQL file created in the directory that the application is located.
Logging: C:\Program Files\Quest\SQL Repermission\SQLPermissions_20150820112148.sql
15. If the option to open the file in Notepad was not checked, navigate to where the file was saved and right-click and choose **Open with Notepad**.
16. The results of the process can be reviewed prior to actually executing it on the SQL Server instance.

```

-----
-- DATABASE
-----
USE [Testing SQL Permissions]
-- [CREATE DATABASE USER]
CREATE USER [testadmin] WITHOUT LOGIN WITH DEFAULT_SCHEMA=[dbo]
-- [Add Schemas Owned]
-- [Add Database Roles]
-- [Securables]
-- [CREATE DATABASE USER]
CREATE USER [QA-DOM1\TestUser1] FOR LOGIN [QA-DOM1\TestUser1] WITH
DEFAULT_SCHEMA=[dbo]
-- [Add schemas Owned]
-- [Add Database Roles]
-- [Securables]

```

17. When the SQL file has been reviewed, either open the file in SQL Server Management Studio or create a new Query in SQL Server Management Studio and copy/paste the contents of the file into the Query.
18. Execute the Query and the new target credentials will be created.

CSV Import File Script

Import File Script Guide

```

#####Guide on how to search objects to create mapping files for ADExpress#####
-----

##### A. Helper script description

script has functions which helps user to convert the AD object search results into .csv
mapping files

1. ADConvertToCsv

```

Import File Script Guide

- helps organize the search result into .csv list files or .csv mapping files
- called by adding to AD object search pipeline
<AD object search command> | ADConvertToCsv
(AD object search commands are explained below)
- will create .csv file to the current script directory from the search result
- if "-Path" is specified, will create file in the path (... | ADConvertToCsv -Path "C:\Users\userx\objects.csv")
- if "-Append" switch is selected, will append existing file (must be called with -Path)
- supports AD, azureAD graph and microsoft graph search command results
- there are two main modes:
 - using "-Source" switch: creates .csv mapping file with all objects returned by the AD object search command as Source objects. User must manually add the Target objects to complete mapping file.
 - default (without "-Source" switch): creates .csv object file with all objects returned by the AD object search command. This object file cannot be used as mapping file. In some cases it contains more information about objects than mapping file.

Main way to use it :

1. Create .csv mapping template file with filled source objects:
<AD object search command> | ADConvertToCsv -Source
2. Create .csv object file with target objects:
<AD object search command> | ADConvertToCsv
3. Manually add the target objects to the source .csv mapping template file
4. Upload mapping file

2. ADMapLocalUsers, ADMapLocalGroups

- helps with mapping of the objects by matching objects with the same SamAccountName
- only works for Local->Local mapping
- functions will create .csv mapping files for Users and Groups respectively in the same directory
- requires two .csv object files (created using ADConvertToCsv - default mode, without "-Source")
- objects that dont have matching SamAccountName will be put to the bottom of the mapping file and will need to be adjusted before uploading mapping file

Main way to use it:

1. Create .csv object file with source objects:
<AD object search command> | ADConvertToCsv
2. Create .csv object file with target objects:
<AD object search command> | ADConvertToCsv
3. Map objects
ADMapLocalUsers <source_object_file_path> <target_object_file_path>
or

Import File Script Guide

```
ADMapLocalGroups <source_object_file_path> <target_object_file_path>
```

B. ADDITIONAL GUIDES

#####1. prepare powershell window

for getting AD local objects

- available by default
- any version (<https://learn.microsoft.com/en-us/powershell/module/activedirectory/?view=windowsserver2022-ps&source=recommendations>)

for getting cloud AD objects via microsoft graph

- <https://learn.microsoft.com/en-us/powershell/microsoftgraph/installation?view=graph-powershell-1.0>
- powershell version 5.1 and higher (powershell 7 recommended)
- powershell 7 installation guide <https://learn.microsoft.com/en-us/powershell/scripting/install/installing-powershell-on-windows?view=powershell-7.4>
- docs state it should work with 5.1 but 7 is recommended. i havent made it work with 5.1

(DEPRECATED)for getting AD cloud objects via azureAd graph

- <https://learn.microsoft.com/en-us/powershell/azure/active-directory/install-adv2?view=azureadps-2.0>
- powershell version 5.1 and lower

#####2. Import script to the current powershell session

- run powershell
 - move to directory where helper script is located (find directory/folder in UI -> shift-rightmouseclick -> "copy as path")
- ```
cd <directory_path>
```
- load script
  - . .\BT-ADObjectMapper.ps1

#### #####3. AD Object search commands

##### #####3.1 LOCAL

##### ##3.1.0 Filtering Local objects

-Filter <query>

specifies query string that retrieves AD objects

-SearchBase <path>

specifies Active Directory path to search under

#examples:

```
Get-ADUser -Filter * -SearchBase "OU=Finance,OU=UserAccounts,DC=FABRIKAM,DC=COM"
```

-gets all users (specified by -Filter \*) in the container

```
"OU=Finance,OU=UserAccounts,DC=FABRIKAM,DC=COM"
```

```
Get-ADUser -Filter 'SamAccountName -like "*test"' -SearchBase
```

```
"OU=Finance,OU=UserAccounts,DC=FABRIKAM,DC=COM"
```

## Import File Script Guide

```
-gets all users, which have SamAccountName ending with "test" in the container
"OU=Finance,OU=UserAccounts,DC=FABRIKAM,DC=COM"
```

\*more in doc links

### ##3.1.1 USER

function:

```
Get-AdUser (https://learn.microsoft.com/en-us/powershell/module/activedirectory/get-aduser?view=windowsserver2022-ps)
```

example:

```
Get-ADUser -Filter * -SearchBase "OU=Finance,OU=UserAccounts,DC=FABRIKAM,DC=COM" |
ADConvertToCsv
```

### ##3.1.2 GROUP

function:

```
Get-ADGroup (https://learn.microsoft.com/en-us/powershell/module/activedirectory/get-adgroup?view=windowsserver2022-ps)
```

example:

```
Get-ADGroup -Filter * -SearchBase "OU=Finance,OU=UserAccounts,DC=FABRIKAM,DC=COM" |
ADConvertToCsv
```

### ##3.1.3 DEVICE

function:

```
Get-ADComputer (https://learn.microsoft.com/en-us/powershell/module/activedirectory/get-adcomputer?view=windowsserver2022-ps)
```

example:

```
Get-ADComputer -Filter * -SearchBase "OU=Finance,OU=UserAccounts,DC=FABRIKAM,DC=COM" |
ADConvertToCsv
```

## ####3.2 CLOUD using azureAD (deprecated)

<https://techcommunity.microsoft.com/t5/microsoft-entra-blog/important-azure-ad-graph-retirement-and-powershell-module/ba-p/3848270>)

### ##3.1.0 Filtering Cloud objects using azureAD

before calling any function we need to call Connect-AzureAD to connect to the azure Azure Active Directory cmdlet (<https://learn.microsoft.com/en-us/powershell/module/azuread/connect-azuread?view=azureadps-2.0>)

-All

returns all users

-Filter <filter>

allows for OData v3.0 filtering (<https://www.odata.org/documentation/odata-version-3-0/odata-version-3-0-core-protocol/#queryingcollections>)

-SearchString <string>

gets all objects that match the search string against the first characters in DisplayName or UserPrincipalName

#examples:

```
Get-AzureADUser -Filter "userPrincipalName eq 'jondoe@contoso.com'"
```

```
-gets all users that have userPrincipalName jondoe@contoso.com
```

```
Get-AzureADGroup -SearchString "Special"
```

```
-gets all groups that have display names starting with "Special"
```

```
Get-AzureADDevice -Filter "startswith(DeviceOSType,'Windows')"
```



## Import File Script Guide

-gets all devices that have DeviceOSType starting with "Windows"

### ##3.2.1 USER

function:

Get-AzureADUser (<https://learn.microsoft.com/en-us/powershell/module/azuread/get-azureaduser?view=azureadps-2.0>)

example:

Connect-AzureAD (interactive login)

Get-AzureADUser -All:\$true | ADConvertToCsv

### ##3.2.2 GROUP

function:

Get-AzureADGroup (<https://learn.microsoft.com/en-us/powershell/module/azuread/get-azureadgroup?view=azureadps-2.0>)

example:

Connect-AzureAD (interactive login)

Get-AzureADGroup -All:\$true | ADConvertToCsv

### ##3.2.3 DEVICE

function:

Get-AzureADDevice (<https://learn.microsoft.com/en-us/powershell/module/azuread/get-azureaddevice?view=azureadps-2.0>)

example:

Connect-AzureAD (interactive login)

Get-AzureADDevice -All:\$true | ADConvertToCsv

## ####3.3 CLOUD using Microsoft Graph (recommended)

### ##3.1.0 Filtering Cloud objects using Microsoft Graph

before calling any function we need to call Connect-MgGraph to connect to the azure entra id (<https://learn.microsoft.com/en-us/powershell/module/microsoft.graph.authentication/connect-mggraph?view=graph-powershell-1.0>)

-All

returns all users

-Filter <filter>

allows for OData v3.0 filtering (<https://www.odata.org/documentation/odata-version-3-0/odata-version-3-0-core-protocol/#queryingcollections>)

-Search <phrase>

search items by search phrases

#examples:

Get-MgUser -Filter "startsWith(DisplayName, 'a')"

-search users which displayName starts with "a" (case-insensitive)

Get-MgUser -Search "'Mail:Peter'"

-search users which email adress contains "Peter"

\*more in doc links

## Import File Script Guide

### ##3.3.1 USER

function:

```
Get-MgUser (https://learn.microsoft.com/en-us/powershell/module/microsoft.graph.users/get-mguser?view=graph-powershell-1.0&preserve-view=true)
```

example:

```
Connect-MgGraph -Scopes 'User.Read.All' (to get permissions and choose entra id-interactive login)
```

```
Get-MgUser -All | ADConvertToCsv
```

### ##3.3.2 GROUP

function:

```
Get-MgGroup (https://learn.microsoft.com/en-us/powershell/module/microsoft.graph.groups/get-mggroup?view=graph-powershell-1.0)
```

example:

```
Connect-MgGraph -Scopes 'Group.Read.All' (to get permissions and choose entra id-interactive login)
```

```
Get-MgGroup -All | ADConvertToCsv
```

### ##3.3.3 DEVICE

function:

```
Get-MgDevice (https://learn.microsoft.com/en-us/powershell/module/microsoft.graph.identity.directorymanagement/get-mgdevice?view=graph-powershell-1.0)
```

example:

```
Connect-MgGraph -Scopes 'Device.Read.All' (to get permissions and choose entra id-interactive login)
```

```
Get-MgDevice -All | ADConvertToCsv
```

##to switch entra id

-disconnect

```
Disconnect-MgGraph
```

-connect to a different one

```
Connect-MgGraph -Scopes <required_scopes>
```

## Import File Script

```
function ADConvertToCsv() #Used to convert AD(onprem/cloud) objects (users,groups,devices/computers) to a .csv list for AD Express mapping file.
```

```
{
```

```
 Param(
```

```
 [Parameter(
```

```
 Mandatory=$true,
```

```
 ValueFromPipeline=$true)]
```

```
 $InputObject, #Can take input from pipeline
```

```
 [Parameter(Mandatory=$false)]
```

```
 [string]$Path, #Path to output .csv file, if not provided, output will be
```

created in current ps directory

```
[Parameter(Mandatory=$false)]
[switch]$Append, #Appends existing .csv file (-Path must be specified)

[Parameter(Mandatory=$false)]
[switch]$Source #To create template mapping file with input as source objects
)
Begin
{
 #append or rewrite
 if ($Append)
 {
 if (!$Path)
 {
 Throw "[ERROR]: Path must be specified (-Path) in order to append
existing file."
 }
 elseif (!(Test-Path $Path))
 {
 Throw "[ERROR]: File does not exist. It needs to exist in order to be
appended."
 }

 $saveOperationType = @{ Append = $true }
 }
 else {
 $saveOperationType = @{ Force = $true }
 }

 #specify output path
 $outputPath = ""
 if ($Path)
 {
 $outputPath = $Path
 }
 else
 {
 if ($Source)
 {
 $outputPath = "$PSScriptRoot\object_mapping_file_" +
[DateTime]::Now.ToString('yyyy_MM_dd_HH_mm_ss') + ".csv"
 }
 else
 {
 $outputPath = "$PSScriptRoot\object_list_" + [DateTime]::Now.ToString
('yyyy_MM_dd_HH_mm_ss') + ".csv"
 }
 }

 $inputList = @()
 $inputType = ""
}
}
```

```

Process
{
 if ($InputObject)
 {
 #make sure the type is allowed and all objects are of same type
 if ($inputType -eq "")
 {
 $inputType = $InputObject.GetType().FullName

 if ($inputType -notin @(
 "Microsoft.ActiveDirectory.Management.ADUser",
 "Microsoft.ActiveDirectory.Management.ADGroup",
 "Microsoft.ActiveDirectory.Management.ADComput
er",
 "Microsoft.Graph.PowerShell.Models.MicrosoftGr
aphUser",
 "Microsoft.Graph.PowerShell.Models.MicrosoftGr
aphGroup",
 "Microsoft.Graph.PowerShell.Models.MicrosoftGr
aphDevice",
 "Microsoft.Open.AzureAD.Model.User",
 "Microsoft.Open.AzureAD.Model.Group",
 "Microsoft.Open.AzureAD.Model.Device"))
 {
 Throw "[ERROR]: Type $inputType is not one of the accepted types."
 }
 }
 elseif ($InputObject.GetType().FullName -ne $inputType)
 {
 Throw "[ERROR]: Type $($InputObject.GetType().FullName) is not the same
type as previous $inputType."
 }

 $inputList += $InputObject
 }
}
End
{
 if ($inputList.Count -eq 0)
 {
 Write-Host "[INFO]: There are no records. Csv file will not be created."
 return
 }

 #transform input into .csv file
 switch ($inputType)
 {
 "Microsoft.ActiveDirectory.Management.ADUser"
 {
 if ($Source)
 {
 $inputList |
 Select-Object -Property @{Name="SourceObjectId"; Expression=
{$_ObjectGUID}}, `

```

```

 @{Name="SourceUserPrincipalName (Optional)"; Expression=
 {$_UserPrincipalName}}, `
 @{Name="SourceSamAccountName (Optional)"; Expression=
 {$_SamAccountName}}, `
 @{Name="TargetObjectId"; Expression="{""}"}, `
 @{Name="TargetUserPrincipalName (Optional)"; Expression=
 {""}}, `
 @{Name="TargetSamAccountName (Optional)"; Expression="{""}"},
 `
 @{Name="Comments (Optional)"; Expression="{""}" } |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
 else
 {
 $inputList |
 Select-Object -Property @{Name="ObjectId"; Expression=
 {$_ObjectGUID}}, `
 @{Name="UserPrincipalName"; Expression=
 {$_UserPrincipalName}}, `
 @{Name="SamAccountName"; Expression={$_SamAccountName}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
}

"Microsoft.ActiveDirectory.Management.ADGroup"
{
 if ($Source)
 {
 $inputList |
 Select-Object -Property @{Name="SourceObjectId"; Expression=
 {$_ObjectGUID}}, `
 @{Name="SourceSamAccountName (Optional)"; Expression=
 {$_SamAccountName}}, `
 @{Name="TargetObjectId"; Expression="{""}"}, `
 @{Name="TargetSamAccountName (Optional)"; Expression="{""}"},
 `
 @{Name="Comments (Optional)"; Expression="{""}" } |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
 else
 {
 $inputList |
 Select-Object -Property @{Name="ObjectId"; Expression=
 {$_ObjectGUID}}, `
 @{Name="SamAccountName"; Expression={$_SamAccountName}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
}

"Microsoft.ActiveDirectory.Management.ADComputer" {
 if ($Source)

```

```

 {
 $inputList |
 Select-Object -Property @{Name="SourceObjectId"; Expression=
{$_ .ObjectGUID}}, `
 @{Name="SourceSamAccountName (Optional)"; Expression=
{$_ .SamAccountName}}, `
 @{Name="Comments (Optional)"; Expression={" "}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
 else
 {
 $inputList |
 Select-Object -Property @{Name="ObjectId"; Expression=
{$_ .ObjectGUID}}, `
 @{Name="SamAccountName"; Expression={$_ .SamAccountName}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
}
"Microsoft.Graph.PowerShell.Models.MicrosoftGraphUser"{
 if ($Source)
 {
 $inputList |
 Select-Object -Property @{Name="SourceObjectId"; Expression=
{$_ .Id}}, `
 @{Name="SourceUserPrincipalName (Optional)"; Expression=
{$_ .UserPrincipalName}}, `
 @{Name="SourceSamAccountName (Optional)"; Expression=
{$_ .OnPremisesSamAccountName}}, `
 @{Name="TargetObjectId"; Expression={" "}}, `
 @{Name="TargetUserPrincipalName (Optional)"; Expression=
{" "}}, `
 @{Name="TargetSamAccountName (Optional)"; Expression={" "}},
 @{Name="Comments (Optional)"; Expression={" "}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
 else
 {
 $inputList |
 Select-Object -Property @{Name="ObjectId"; Expression={$_ .Id}},
 @{Name="UserPrincipalName"; Expression=
{$_ .UserPrincipalName}}, `
 @{Name="SamAccountName"; Expression=
{$_ .OnPremisesSamAccountName}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
}
"Microsoft.Graph.PowerShell.Models.MicrosoftGraphGroup"{
 if ($Source)
 {

```

```

 $inputList |
 Select-Object -Property @{Name="SourceObjectId"; Expression=
{$_ .Id}}, `
 @{Name="SourceSamAccountName (Optional)"; Expression=
{$_ .OnPremisesSamAccountName}}, `
 @{Name="TargetObjectId"; Expression={" "}}, `
 @{Name="TargetSamAccountName (Optional)"; Expression={" "}},
 `
 @{Name="Comments (Optional)"; Expression={" "}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
 else
 {
 $inputList |
 Select-Object -Property @{Name="ObjectId"; Expression={$_ .Id}},
 `
 @{Name="SamAccountName"; Expression=
{$_ .OnPremisesSamAccountName}}, `
 @{Name="DisplayName"; Expression={$_ .DisplayName}}, `
 @{Name="EmailAddress"; Expression={$_ .Mail}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
}
"Microsoft.Graph.PowerShell.Models.MicrosoftGraphDevice"{
 if ($Source)
 {
 $inputList |
 Select-Object -Property @{Name="SourceObjectId"; Expression=
{$_ .Id}}, `
 @{Name="SourceSamAccountName (Optional)"; Expression=
{$_ .OnPremisesSamAccountName}}, `
 @{Name="Comments (Optional)"; Expression={" "}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
 else
 {
 $inputList |
 Select-Object -Property @{Name="ObjectId"; Expression={$_ .Id}},
 `
 @{Name="SamAccountName"; Expression=
{$_ .OnPremisesSamAccountName}}, `
 @{Name="DisplayName"; Expression={$_ .DisplayName}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
}
"Microsoft.Open.AzureAD.Model.User" {
 if ($Source)
 {
 $inputList |
 Select-Object -Property @{Name="SourceObjectId"; Expression=

```

```

 {$_.ObjectId}}, `
 {$_.UserPrincipalName}}, `
 ,
 ,
 @Name="SourceUserPrincipalName (Optional)"; Expression=
 @Name="SourceSamAccountName (Optional)"; Expression="{""}",
 ,
 @Name="TargetObjectId"; Expression="{""}", `
 @Name="TargetUserPrincipalName (Optional)"; Expression=
 {""}, `
 ,
 @Name="TargetSamAccountName (Optional)"; Expression="{""}",
 ,
 @Name="Comments (Optional)"; Expression="{""}" |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
 else
 {
 $inputList |
 Select-Object -Property @Name="ObjectId"; Expression=
 {$_.ObjectId}}, `
 @Name="UserPrincipalName"; Expression=
 {$_.UserPrincipalName}}, `
 @Name="SamAccountName"; Expression="{""}" |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
}
"Microsoft.Open.AzureAD.Model.Group" {
 if ($Source)
 {
 $inputList |
 Select-Object -Property @Name="SourceObjectId"; Expression=
 {$_.ObjectId}}, `
 @Name="SourceSamAccountName (Optional)"; Expression="{""}",
 ,
 @Name="TargetObjectId"; Expression="{""}", `
 @Name="TargetSamAccountName (Optional)"; Expression="{""}",
 ,
 @Name="Comments (Optional)"; Expression="{""}" |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
 else
 {
 $inputList |
 Select-Object -Property @Name="ObjectId"; Expression=
 {$_.ObjectId}}, `
 @Name="SamAccountName"; Expression="{""}", `
 @Name="DisplayName"; Expression={$_.DisplayName}}, `
 @Name="EmailAddress"; Expression={$_.Mail}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
}
"Microsoft.Open.AzureAD.Model.Device" {

```



```

 if ($Source)
 {
 $inputList |
 Select-Object -Property @{Name="SourceObjectId"; Expression=
{$_ .ObjectId}}, `
 @{Name="SourceSamAccountName (Optional)"; Expression={""}},
 @{Name="Comments (Optional)"; Expression={""}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
 else
 {
 $inputList |
 Select-Object -Property @{Name="ObjectId"; Expression=
{$_ .ObjectId}}, `
 @{Name="SamAccountName"; Expression={""}}, `
 @{Name="DisplayName"; Expression={$_.DisplayName}} |
 Export-Csv -Path $outputPath -Delimiter "," -
NoTypeInfoInformation @saveOperationType
 }
 }
 default {
 Throw "[ERROR]: Type $inputType is not one of the accepted types."
 }
}

if ($Append)
{
 Write-Host "[INFO]: File '$($outputPath)' appended"
}
else
{
 Write-Host "[INFO]: File '$($outputPath)' created"
}
}
}

function ADMapLocalUsers($sourceUsersPath, $targetUsersPath) #Map two user files based
on SamAccountName
{
 $sourceUsers = Import-Csv -Path $sourceUsersPath -Delimiter ","
 $targetUsers = Import-Csv -Path $targetUsersPath -Delimiter ","
 $mappedUsers = @()
 $nonMappedUsers = @()

 #for each source user, try to find target user via SamAccountName
 $sourceUsers | % {
 $sourceObject = $_
 $targetMatch = $null
 if ($sourceObject.SamAccountName -ne "")
 {
 $targetMatch = $targetUsers | Where-Object {$_.SamAccountName -eq

```

```

$sourceObject.SamAccountName}
}

if ($targetMatch) #if found, create match record
{
 if ($targetMatch.Count -gt 1)
 {
 throw "[ERROR]: Target User SamAccountName is not unique
'${$sourceObject.SamAccountName}'"
 }

 $mappedUsers += [PSCustomObject]@{
 SourceObjectId = $sourceObject.ObjectId
 SourceUserPrincipalName = $sourceObject.UserPrincipalName
 SourceSamAccountName = $sourceObject.SamAccountName
 TargetObjectId = $targetMatch.ObjectId
 TargetUserPrincipalName = $targetMatch.UserPrincipalName
 TargetSamAccountName = $targetMatch.SamAccountName
 Comments = ""
 }
}
else #if havent found, create record without
{
 $nonMappedUsers += [PSCustomObject]@{
 SourceObjectId = $sourceObject.ObjectId
 SourceUserPrincipalName = $sourceObject.UserPrincipalName
 SourceSamAccountName = $sourceObject.SamAccountName
 TargetObjectId = ""
 TargetUserPrincipalName = ""
 TargetSamAccountName = ""
 Comments = ""
 }
}
}

#save target users which dont have source user match
$sourceUsersWithSam = $sourceUsers | Where-Object {$_.SamAccountName -ne ""}
$targetUsersWithoutSourceSam = $targetUsers | Where-Object { $_.SamAccountName -
notin $sourceUsersWithSam.SamAccountName}
$targetUsersWithoutSourceSam | % {
 $nonMappedUsers += [PSCustomObject]@{
 SourceObjectId = ""
 SourceUserPrincipalName = ""
 SourceSamAccountName = ""
 TargetObjectId = $_.ObjectId
 TargetUserPrincipalName = $_.UserPrincipalName
 TargetSamAccountName = $_.SamAccountName
 Comments = ""
 }
}

#save
$mappedUsers += $nonMappedUsers
$outputPath = "$PSScriptRoot\user_map_" + [DateTime]::UtcNow.ToString('yyyy_MM_dd_

```

```

HH_mm_ss') + ".csv"
 $mappedUsers | Export-Csv -Path $outputPath -Delimiter "," -NoTypeInformation
 Write-Host "[INFO]: File '$($outputPath)' created"
}
function ADMapLocalGroups($sourceGroupsPath, $targetGroupsPath) #Map two group files
based on SamAccountName
{
 $sourceGroups = Import-Csv -Path $sourceGroupsPath -Delimiter ","
 $targetGroups = Import-Csv -Path $targetGroupsPath -Delimiter ","
 $mappedGroups = @()
 $nonMappedGroups = @()

 #for each source group, try to find target group via SamAccountName
 $sourceGroups | % {
 $sourceObject = $_
 $targetMatch = $null
 if ($sourceObject.SamAccountName -ne "")
 {
 $targetMatch = $targetGroups | Where-Object { $_.SamAccountName -eq
$sourceObject.SamAccountName}
 }

 if ($targetMatch) #if found, create match record
 {
 if ($targetMatch.Count -gt 1)
 {
 throw "Target Group SamAccountName is not unique
'$($sourceObject.SamAccountName)'"
 }

 $mappedGroups += [PSCustomObject]@{
 SourceObjectId = $sourceObject.ObjectId
 SourceSamAccountName = $sourceObject.SamAccountName
 TargetObjectId = $targetMatch.ObjectId
 TargetSamAccountName = $targetMatch.SamAccountName
 Comments = ""
 }
 }
 else #if havent found, create record without
 {
 $nonMappedGroups += [PSCustomObject]@{
 SourceObjectId = $sourceObject.ObjectId
 SourceSamAccountName = $sourceObject.SamAccountName
 TargetObjectId = ""
 TargetSamAccountName = ""
 Comments = ""
 }
 }
 }

 #save target groups which dont have source group match
 $sourceGroupsWithSam = $sourceGroups | Where-Object { $_.SamAccountName -ne ""}
 $targetGroupsWithoutSourceSam = $targetGroups | Where-Object { $_.SamAccountName -
notin $sourceGroupsWithSam.SamAccountName}

```

```

$targetGroupsWithoutSourceSam | % {
 $nonMappedGroups += [PSCustomObject]@{
 SourceObjectId = ""
 SourceSamAccountName = ""
 TargetObjectId = $_.ObjectId
 TargetSamAccountName = $_.SamAccountName
 Comments = ""
 }
}

#save
$mappedGroups += $nonMappedGroups
$outputPath = "$PSScriptRoot\group_map_" + [DateTime]::UtcNow.ToString('yyyy_MM_dd_
HH_mm_ss') + ".csv"
$mappedGroups | Export-Csv -Path $outputPath -Delimiter "," -NoTypeInformation
Write-Host "[INFO]: File '$($outputPath)' created"
}

```

# About us

---

Quest creates software solutions that make the benefits of new technology real in an increasingly complex IT landscape. From database and systems management, to Active Directory and Office 365 management, and cyber security resilience, Quest helps customers solve their next IT challenge now. Around the globe, more than 130,000 companies and 95% of the Fortune 500 count on Quest to deliver proactive management and monitoring for the next enterprise initiative, find the next solution for complex Microsoft challenges and stay ahead of the next threat. Quest Software. Where next meets now. For more information, visit [www.quest.com](http://www.quest.com).

## Technical support resources

Technical support is available to Quest customers with a valid maintenance contract and customers who have trial versions. You can access the Quest Support Portal at <https://support.quest.com>.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to-videos
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product