

Installing and Configuring VMware User Environment Manager

VMware Dynamic Environment Manager 9.4

VMware User Environment Manager



vmware®

You can find the most up-to-date technical documentation on the VMware website at:

<https://docs.vmware.com/>

If you have comments about this documentation, submit your feedback to

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2018 VMware, Inc. All rights reserved. [Copyright and trademark information.](#)

Contents

About Installing and Configuring VMware User Environment Manager™	6
1 Introduction to User Environment Manager	7
Application Configuration Management	7
User Environment Settings	8
Personalization of Application and Windows Settings	8
Migration of Application Settings	9
Dynamic Configuration of the Desktop	9
2 User Environment Manager Scenario Considerations	10
Using User Environment Manager with Mandatory Profiles	10
Using User Environment Manager with Roaming Profiles	11
Using User Environment Manager with Local Profiles	12
3 Installation and Deployment Prerequisites	13
User Environment Manager Infrastructure and Terminology	13
Overview of the User Environment Manager Deployment	15
Infrastructure Requirements	15
User Environment Manager Configuration Share	16
Profile Archives Share	17
Software Requirements	18
Registry Access Requirements	19
Licensing Requirements	19
4 Installing User Environment Manager	20
Overview of the User Environment Manager Deployment	21
Install User Environment Manager Manually	21
Unattended Installation of User Environment Manager	22
Upgrade User Environment Manager	23
Upgrade FlexEngine on all Windows Desktops and Terminal Servers	23
Upgrade the User Environment Manager Management Console	24
Upgrade the ADMX Templates	25
5 Configuring User Environment Manager	26
Configuring the FlexEngine Group Policy Object	26
Create a User Environment Manager Group Policy Object	27
Configure the User Environment Manager Group Policy Object	28
Open the User Environment Manager Group Policy Object	30

- Configure the Flex Configuration Files Setting 30
- Configure FlexEngine to Run as Group Policy Extension Setting 31
- Configure FlexEngine Logging Setting 31
- Configure Profile Archives Setting 32
- Configure Profile Archive Backups Setting 33
- Configure Application Blocking Logging to the Windows Event Log Setting 33
- Configure Privilege Elevation Logging to the Windows Event Log Setting 33
- Configure Certificate Support for Mandatory Profiles Setting 34
- Configure DirectFlex - Advanced Settings 34
- Configure FlexEngine Logging to the Windows Event Log Setting 35
- Configure Paths Unavailable at Logon Setting 35
- Configure FlexEngine Refresh Settings 36
- Configure Access to VMware UEM Self-Support for End Users 37
- Configure User Environment Manager Logon and Logoff Progress Information 37
- Configure Silo-Specific Flex Configuration Files Settings 37
- Configuring FlexEngine to Run from Logon and Logoff Scripts 38
 - Configure FlexEngine to Run from a Logon Script 38
 - Configure FlexEngine to Run From a Logoff Script 38
- Configuring the User Environment Manager Management Console 39

6 Installing and Configuring User Environment Manager in NoAD Mode 40

- Install User Environment Manager in NoAD Mode 40
- Configuring User Environment Manager with the NoAD Mode 41
 - Configuring FlexEngine Logging Settings 41
 - Configuring Profile Archives Settings 41
 - Configuring Profile Archive Backups Settings 42
 - Configuring Application Blocking Logging to the Windows Event Log Setting 42
 - Configuring Privilege Elevation Logging to the Windows Event Log Setting 42
 - Enable Certificate Support for Mandatory Profiles Setting 43
 - Configure DirectFlex – Advanced Settings 43
 - Configure FlexEngine Logging to the Windows Event Log Setting 43
 - Configure Paths Unavailable at Logon Setting 44
 - Prevent Access to User Environment Manager Self-Support to End Users 45
 - Disable the NoAD Mode for Certain Users 45
 - Remove Registry Settings for the NoAD Mode 45
 - Sample NoAD.xml File 45

7 FlexEngine Command-Line Arguments 47

- FlexEngine Operations and Arguments 47
- Operation Modes 48
- Overriding Group Policy Settings 49

Applying Horizon Smart Policies in Multiple Sessions	49
Additional FlexEngine Operations	49

About Installing and Configuring VMware User Environment Manager™

Installing and Configuring VMware User Environment Manager provides information about installing and configuring VMware User Environment Manager™ on Terminal Services or Windows desktop environments.

Intended Audience

This information is intended for experienced Windows administrators who want to deploy User Environment Manager in their Terminal Services or Windows desktop environments to provide dynamic management of desktop, user, and application settings.

Introduction to User Environment Manager

1

User Environment Manager provides end users with a personalized and dynamic Windows desktop. Through User Environment Manager, you can customize the desktop by providing access to IT resources based on the role, device, and location of the user. In this way, you can create a desktop that adapts to the specific needs of the user.

User Environment Manager manages user and Windows settings and dynamically configures the desktop. For example, it can create drive and printer mappings, file type associations, and shortcuts. User Environment Manager can also manage and provide shortcuts to applications such as ThinApp to users.

This chapter includes the following topics:

- [Application Configuration Management](#)
- [User Environment Settings](#)
- [Personalization of Application and Windows Settings](#)
- [Migration of Application Settings](#)
- [Dynamic Configuration of the Desktop](#)

Application Configuration Management

With User Environment Manager, you can configure the initial settings of an application without having to rely on the defaults of the application. You can define the application settings that the user can personalize and the settings that always remain unchanged each time the user opens the application. In this way, you can combine policy-enforced settings and user personalization.

You can also use User Environment Manager to manage certain user environment settings when an application starts. For example, you can configure drive and printer mappings, apply custom settings for files and folders, and registry, and run custom tasks. You can also define settings and configurations for all users to guarantee compliance and provide a consistent environment.

You can use the User Environment Manager Application Profiler to capture predefined settings for an application. You can run the application on a reference system that Application Profiler monitors and configure the settings that you want.

User Environment Settings

You can use User Environment Manager to centrally manage user settings that users require to perform their daily tasks. The settings are applied when a user logs in or starts a certain application.

For example, a multinational corporation has end users from multiple countries. The company can centrally manage the different display languages, wallpapers, keyboard configurations, and other regional settings.

By using User Environment Manager, you can manage the following settings:

- ADMX-based settings
- Application blocking
- Application shortcuts and file type associations
- Display language
- Drive and printer mappings
- Environment variables
- Files, folders, and registry settings
- Folder redirection
- Hide drives
- Horizon Smart Policies
- Logon and logoff tasks
- Privilege Elevation
- Triggered tasks

Personalization of Application and Windows Settings

User Environment Manager is a profile management capability that enables users to roam their personal desktop and application settings. Personalization separates user-specific desktop and application settings from the Windows operating system. This way, the settings are available across OS versions, devices, and application instances.

Personalization works independent from the Windows user profile management and facilitates the management of virtualization technologies and application delivery mechanisms. User Environment Manager personalization integrates seamlessly with natively installed and virtualized applications, providing a consistent user experience across Windows platforms that are physical or virtual, local, or remote.

User Environment Manager personalization lets users create their own personal settings. For example, developers might want to configure Eclipse according to their own habits and apply that configuration across multiple environments. Quality engineers might want to set the bug tracking Web site as the home page of all browsers.

Migration of Application Settings

User Environment Manager can transfer personal application settings of users from one OS to another, for example from Windows 7 to Windows 10. This is default behavior of User Environment Manager, the only requirement is that the application must store its configuration at the same registry and AppData locations in the user profile.

User Environment Manager also provides an XML-based mechanism for settings migration between application versions. In this way, you can avoid situations where personalized data is lost during application upgrade, because it is stored at different location after the upgrade.

The User Environment Manager download package contains XML migration file samples for migrating between different versions of Microsoft Office.

Dynamic Configuration of the Desktop

By using User Environment Manager, you can provide dynamic adaptation of the content and appearance of the end-user desktop by using conditions. User Environment Manager includes ready-to-use conditions that you can combine for the user, location, and device characteristics.

For example, by using conditions you can provide access to a network printer that is based on the current physical location of the user, or create an application shortcut on the desktop that is based on the user's identity.

You can re-evaluate conditions when users unlock their workstation or reconnect to a remote session. You can manage conditions from the User Environment Manager console and apply them to all configurable items within User Environment Manager.

User Environment Manager Scenario Considerations

2

You can use User Environment Manager to optimize the experience of Windows users with all types of user profiles: mandatory, roaming, and local.

This chapter includes the following topics:

- [Using User Environment Manager with Mandatory Profiles](#)
- [Using User Environment Manager with Roaming Profiles](#)
- [Using User Environment Manager with Local Profiles](#)

Using User Environment Manager with Mandatory Profiles

Mandatory profiles are mostly used in Terminal Services environments, although you can use them with Windows desktops as well. With mandatory profiles, personalization changes of the desktop are effective only during a Windows session. When the user logs out, all changes are deleted. With User Environment Manager, you can eliminate the need of customizing mandatory profiles, manage the settings that are available for personalization, and customize the user environment settings.

The following are the advantages and disadvantages of mandatory profiles:

Advantages

- Short login and logout times.
- Consistent user experience, no matter what the user changes.
- Minimal troubleshooting on user profiles.

Disadvantages

- None of the personalization changes made by users are saved.
- Creating a usable and customized mandatory profile requires a high level of skill.
- Scripting is often necessary to create shortcuts, drive mappings, and so on.

When using User Environment Manager with mandatory profiles, you can address the disadvantages in the following ways:

- Select the settings that users are allowed to personalize within their environment. Settings that you do not manage with User Environment Manager are discarded when the user logs out.

- Configure specific settings for applications or Windows settings by using the Predefined Settings feature of User Environment Manager. By using predefined settings, you do not need to customize a mandatory profile. A mandatory profile that is based on the Default User profile is sufficient.
- Customize the user environment by creating shortcuts, drive mappings, and so on.

Using User Environment Manager with Roaming Profiles

Roaming profiles are mostly used in a managed desktop environment. With roaming profiles, all personalization changes that users make during a Windows session are stored in the central roaming profile when users log out. When a user logs in to a Windows session, the roaming profile is copied again from the central location. With User Environment Manager, you can manage the size of roaming profiles, reduce login times, and achieve greater flexibility in managing application and Windows settings for roaming profiles.

You should avoid using roaming profiles with User Environment Manager for a longer period of time. Typically, User Environment Manager only runs with roaming profiles when you start migrating from the roaming profile to either local or mandatory profiles being managed by User Environment Manager. The User Environment Manager best practice is to use either local or mandatory profiles.

Advantages

- No specific administration necessary besides enabling the roaming profiles.
- Personalized settings roam with the user across different machines that are running the same operating system.

Disadvantages

- Limited control over the settings that the users can change. Everything is saved by default.
- Large roaming profiles might get corrupted and cause the individual roaming profile to reset completely. As a result, users might spend a lot of time getting all personalized settings back.
- Troubleshooting an application defect might cause the individual roaming profile to reset completely because all application and Windows settings are stored in a single container.
- Roaming profiles do not roam across different operating systems. This results in multiple roaming profiles per user in a mixed environment, like desktops and Terminal Services.
- Potential for unnecessary growth of roaming profile, causing long login times.
- Application shortcuts and file type associations get retained in roaming profiles and often cause confusion when users roam to devices where the applications might not be installed.

You can address many of the disadvantages of roaming profiles by using User Environment Manager.

- You can use the Profile Cleanup feature to clean up unimportant or obsolete parts of each user profile at logout.
- Create a mandatory set of settings for business-critical applications by using the Predefined Settings feature. You can also use Predefined Settings to disallow personalized settings for certain applications.
- Decouple and segment personalized application and Windows settings from the roaming profile by using the Import / Export and the Profile Cleanup features.
- Compress all settings that User Environment Manager manages including files and folders to provide shorter login times.
- Save all settings for roaming profiles in a central place that makes the settings available after a total reset.
- Reset certain application or Windows settings without performing a complete reset of the roaming user profile.
- Provide roaming for personalized application and Windows settings across different operating systems for a consistent user experience.
- Provide different application and Windows settings depending on a user's business case by using Condition Sets.

Using User Environment Manager with Local Profiles

With local profiles, personalized changes that users make during a Windows session are stored on the local disk. When a user logs in to the same desktop again, the user environment is the same as in the previous session of the user. When a user logs in to another desktop, none of the settings are the same, as a new local profile is created and stored locally on that desktop.

Advantages

- No specific administration is needed.
- No network storage is required.

Disadvantages

- Personalized settings are not roamed across different machines.
- Each desktop a user logs on to is polluted with a local profile for that specific user.
- If local disk failure or corruption occurs, all user settings are lost.

With User Environment Manager, you can eliminate the disadvantages of local profiles.

- Introduce roaming functionality for application and Windows settings that User Environment Manager manages.
- Create redundancy for application and Windows settings by managing these settings with User Environment Manager when local disk failure or corruption occurs.

Installation and Deployment Prerequisites

3

To install and deploy User Environment Manager, your environment must meet certain infrastructure, system, access, and licensing requirements. You must also get familiar with the User Environment Manager terminology.

This chapter includes the following topics:

- [User Environment Manager Infrastructure and Terminology](#)
- [Overview of the User Environment Manager Deployment](#)
- [Infrastructure Requirements](#)
- [Software Requirements](#)
- [Registry Access Requirements](#)
- [Licensing Requirements](#)

User Environment Manager Infrastructure and Terminology

To install and configure User Environment Manager, you must get familiar with the User Environment Manager components and terminology.

Table 3-1. User Environment Manager Terminology

Component or Term	Description
Management Console	The User Environment Manager Management Console that is the main interface that you can use to manage user profiles.
Flex configuration file	A configuration file where you define all application, Windows, and user environment settings. You create and manage Flex configuration files by using the Management Console.
User Environment Manager configuration share	The UNC path to the share where the Management Console configuration and User Environment Manager configuration files are stored.
FlexEngine	The User Environment Manager client component that you must install on each physical or virtual Windows device where you use User Environment Manager.

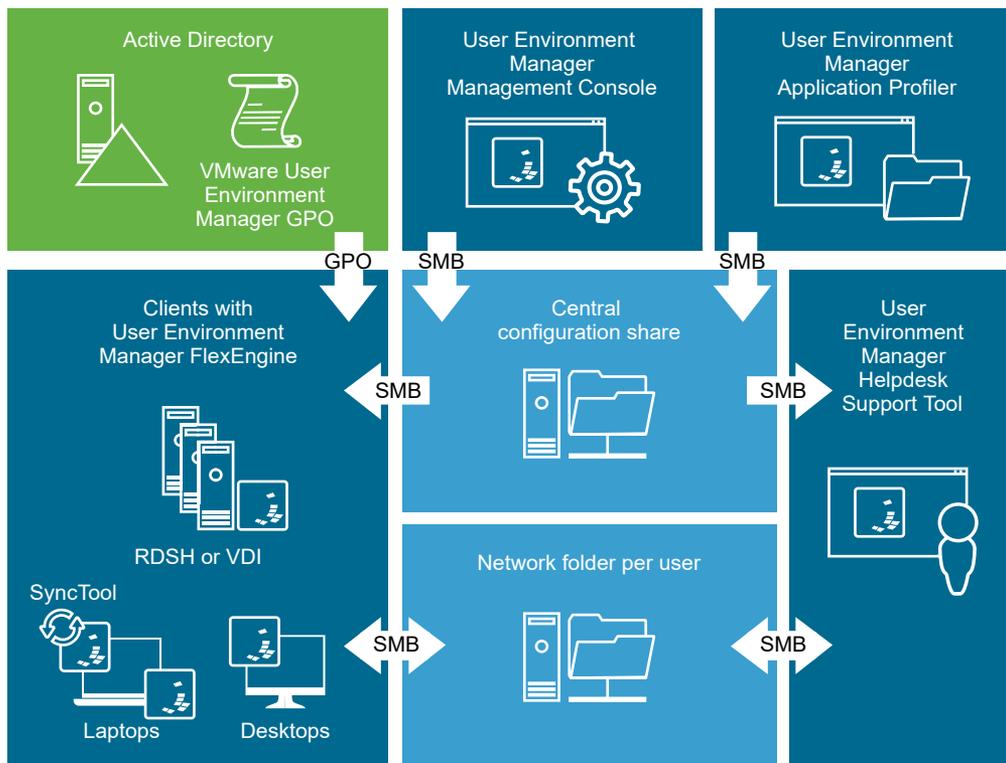
Table 3-1. User Environment Manager Terminology (continued)

Component or Term	Description
Profile archives	Profile archives are ZIP files where FlexEngine stores the personalized settings of users, based on the content of Flex configuration files. For each Flex configuration file that you create, FlexEngine creates a profile archive for each user.
Profile archives path	The path that FlexEngine uses to store the profile archives for individual users.
Profile archives backup path	The path that FlexEngine uses to store backups of the profile archives.
General folder	A folder that is named General, that the Management Console creates in the User Environment Manager configuration share. This folder is the location where Flex configuration files are created, managed, and used from by FlexEngine.

User Environment Manager Infrastructure

Besides the core User Environment Manager components, you can also use tools such as Application Profiler, SyncTool, and Helpdesk Support tool. All components of User Environment Manager that you deploy communicate between each other by using the SMB protocol.

Figure 3-1. User Environment Manager Infrastructure



Overview of the User Environment Manager Deployment

Prepare your environment to meet the User Environment Manager infrastructure requirements, and then install and configure the User Environment Manager components.

- Create a User Environment Manager configuration share on a file server.
- Create a User Environment Manager profile archives share.
- Install FlexEngine on Windows desktops or Terminal Servers.
- Create the User Environment Manager Group Policy by using the administrative template that is provided in the User Environment Manager package.
- Add FlexEngine command to logoff script.
- Install the Management Console on the administrator's machine.
- Perform initial configuration of the Management Console.

After you install and configure the User Environment Manager components, you can start managing personalization and application management settings by creating Flex configuration files. You can use the User Environment Manager Application Profiler to capture application settings in Flex configuration files.

Note The User Environment Manager MSI file has a digital signature, which the Windows Installer infrastructure validates when the installation starts. The installation process includes a certificate revocation check for which the system requires Internet access. If the Internet connectivity is not sufficient, the installation continues, but only after several timeouts. During the process, the installer seems to hang without providing any feedback.

Infrastructure Requirements

To deploy User Environment Manager, your environment must meet certain infrastructure requirements.

Requirement	Description
Active Directory	Active Directory is required for Group Policy configuration of FlexEngine. You configure FlexEngine by creating an Active Directory Group Policy Object (GPO). See Configure the User Environment Manager Group Policy Object . You can configure FlexEngine without Group Policy, but it requires command-line arguments. For more information, see Chapter 7 FlexEngine Command-Line Arguments .
User Environment Manager configuration share	A central share on a file server, which can be a replicated share for multiple sites. In such a case, you can use multiple Active Directory GPOs to configure the path to the share for all client devices, based on the location. This share uses the Server Message Block (SMB) protocol for communication. For more information about the configuration share requirements, see User Environment Manager Configuration Share .
Profile archives share	You must consider a location to store the profile archive ZIP files for the user settings and the profile backups. The profile archives share also uses SMB. For details about the profile archives share, see Profile Archives Share .

User Environment Manager Configuration Share

The User Environment Manager configuration share is a central share on a file server. It contains all the configuration files for personalization and application configuration management of User Environment Manager. FlexEngine reads configuration data from the User Environment Manager configuration share when a user logs in or logs out of the environment, or when the user opens or closes applications that are configured with DirectFlex.

Folder Structure

The User Environment Manager configuration share has a predefined structure. The first time when you start the Management Console, the `General` folder is automatically created in the configuration share. The `General` folder contains the Flex configuration files that you use to define settings for personalization and application configuration management.

The `General` folder also contains the mandatory `FlexRepository` folder. The Management Console creates the `FlexRepository` folder the first time you configure a user environment setting, such as a printer mapping. The `FlexRepository` folder contains all the configuration files for the user environment settings and condition sets.

Requirements

Requirement	Description
Networking	To optimize login times, the computer where the user logs in should have a 1-Gbps connection to the configuration share.
Storage	Storage requirements might vary based on the specific deployment. A general guideline is to have at least 200 kB per application with a starting minimum size of 1 GB.
NTFS security permission	<ul style="list-style-type: none"> ■ Administrators must have Full Control permissions. ■ End users must have Read and Execute permissions. <p>Caution For security reasons, non-administrators must not have write permissions on the User Environment Manager configuration share.</p>

Profile Archives Share

The profile archives share stores the personal settings for users as FlexEngine creates a subfolder for each user. The share contains User Environment Manager profile archives, which are ZIP files. FlexEngine reads personal user settings from the profile archives share when a user logs in to the environment or launches a DirectFlex-enabled application. FlexEngine writes the modified settings when the user logs out, or closes a DirectFlex-enabled application.

In a typical deployment, profile archive backups and log files are stored on the same share, but you can configure different locations in the FlexEngine GPO.

You should use a share that is dedicated to the profile archives. A dedicated share improves performance, simplifies configuring the User Environment Manager SyncTool, and makes it easier to configure permissions for the Helpdesk Support Tool.

Note Do not use the Home drive share. Using this share can cause synchronization conflicts between Offline Files and the User Environment Manager SyncTool, and lets users delete their profile archives.

Folder Structure

The profile archives share has a one-on-one relation to the naming and folder structure of the User Environment Manager configuration share and the Management Console.

Requirements

Requirement	Description
Networking requirements	For best performance and to optimize login times, the computer where the end user logs in should have a 1-Gbps connection to the profile archives share. If an end user has limited bandwidth or has a laptop that is often offline, use the SyncTool. This tool improves connectivity to the profile archives share under these conditions.
Storage	Storage requirements might vary based on the specific deployment. A general guideline is to have at least 100 MB per user.
NTFS security permissions	<p>Setting the following NTFS security permissions on the profile archives share automatically creates a folder for each user on first login and limits the user to their own folder.</p> <ul style="list-style-type: none"> ■ For User Environment Manager administrators and help desk: Full control, applied to this folder, sub folders, and files. ■ For End users: Create folders and append data, applied to this folder only. ■ For Creator owner: Full control, applied to sub folders and files only. <p>The minimum share permissions for all users should be Change and Read.</p>

Software Requirements

The system on which you plan to install User Environment Manager must meet certain software requirements.

Supported Windows Versions

- Windows 7 Professional, Enterprise, and Ultimate x86 and x64 SP1
- Windows Server 2008 R2 Standard and Enterprise x64 SP1
- Windows Server 2012 Standard and Datacenter x64
- Windows 8.1 Professional and Enterprise x86 and x64 with Update
- Windows Server 2012 R2 Standard and Datacenter x64 with Update
- Windows 10 Version 1803 (April 2018 Update) Professional and Enterprise x86 and x64
- Windows Server 2016 Standard and Datacenter x64

Supported Application Virtualization Products and Versions

- App-V 4.6 Service Pack 3
- App-V 5.0 Service Pack 3

- App-V 5.1
- ThinApp 5.2
- SWV 7.5
- SWV 7.6

Registry Access Requirements

Access to `Regedit.exe` or `Reg.exe` must not be disabled through Group Policy. FlexEngine uses `Regedit.exe` to add user-specific settings to the registry. Depending on the User Account Control (UAC) settings on Windows 7 or later, FlexEngine might use `Reg.exe`.

User Environment Manager might not work properly on some Windows versions if access to `Regedit.exe` is disabled through Group Policy, unless the option `Disable regedit from running silently?` is set to No. However, this setting is insufficient for `Reg.exe`. This means that if `Regedit.exe` cannot run due to UAC, this policy must remain unset.

If users are not allowed to run `Regedit.exe` silently, an error message might appear when they log in. An error message is also written to the FlexEngine log file.

Licensing Requirements

FlexEngine requires a valid license file. To switch from an evaluation license file to a production license file, reinstallation of any User Environment Manager component is not required. You must only replace the old license file with the new license file, retaining both license filename and location in the file system.

Note A license file is not required on Horizon 7 for installation. A license file is required only when you install User Environment Manager as a standalone installation.

Installing User Environment Manager

4

You must install User Environment Manager on your environment to manage personalization and application configuration settings. The User Environment Manager MSI consists of several installation features that you can install on your environment.

Table 4-1. User Environment Manager Installation Features

Installation Feature	Description
VMware UEM FlexEngine	<p>Client component that you must install on each desktop or Terminal Server that you want to manage by using User Environment Manager.</p> <p>If you are deploying FlexEngine to physical machines, you can use any software deployment tool to perform batch deployment or use Active Directory Group Policy software deployment.</p> <p>If you deploy FlexEngine in a VDI or RDSH environment, such as VMware Horizon, you can manually install FlexEngine in the template or parent virtual machines and then deploy pools and farms of VMware Horizon® View™ desktops and RDSH servers based on these templates.</p> <p>Important When you are deploying FlexEngine in virtual machines used in VMware Horizon® Cloud Service™, some differences apply, such as installation paths and other specifics related to use in the Horizon Cloud environment. For details about deploying FlexEngine when using a Horizon Cloud environment, see the administration documentation that is appropriate for your Horizon Cloud deployment mode. Horizon Cloud documentation is available from the Horizon Cloud documentation landing page.</p>
VMware UEM Management Console	Administration console that you can install on any desktop or Terminal Server where you want to manage User Environment Manager.
Application Migration	Optional. You can install Application Migration on desktops or Terminal Servers if you want to migrate application settings across application versions. This feature depends on FlexEngine and cannot work standalone.
Self-Support	Optional. You can install the Self-Support tool on desktops or Terminal Servers where you want users to support their application settings by themselves, without administrator intervention. This feature depends on FlexEngine and cannot work standalone.

This chapter includes the following topics:

- [Overview of the User Environment Manager Deployment](#)
- [Install User Environment Manager Manually](#)
- [Unattended Installation of User Environment Manager](#)
- [Upgrade User Environment Manager](#)

Overview of the User Environment Manager Deployment

Prepare your environment to meet the User Environment Manager infrastructure requirements, and then install and configure the User Environment Manager components.

- Create a User Environment Manager configuration share on a file server.
- Create a User Environment Manager profile archives share.
- Install FlexEngine on Windows desktops or Terminal Servers.
- Create the User Environment Manager Group Policy by using the administrative template that is provided in the User Environment Manager package.
- Add FlexEngine command to logoff script.
- Install the Management Console on the administrator's machine.
- Perform initial configuration of the Management Console.

After you install and configure the User Environment Manager components, you can start managing personalization and application management settings by creating Flex configuration files. You can use the User Environment Manager Application Profiler to capture application settings in Flex configuration files.

Note The User Environment Manager MSI file has a digital signature, which the Windows Installer infrastructure validates when the installation starts. The installation process includes a certificate revocation check for which the system requires Internet access. If the Internet connectivity is not sufficient, the installation continues, but only after several timeouts. During the process, the installer seems to hang without providing any feedback.

Install User Environment Manager Manually

You install User Environment Manager by running VMware User Environment Manager 9.4 x86.msi or VMware User Environment Manager 9.4 x64.msi based on your operating system. This launches a setup wizard which guides you through the installation.

Prerequisites

- Verify that you have a valid license file. If you install on VMware Horizon[®] 7, you do not need a separate license file.
- Verify that you have administrative privileges on the account where you will run the MSI file.
- Download and extract the MSI file package for your operating system.

Procedure

- 1 Run the MSI that corresponds to your OS architecture, and click **Next**.
- 2 Read and accept the **End User License Agreement** and click **Next**.
- 3 Select the destination folder where you want to install the application and click **Next**.

VMware recommends you install User Environment Manager in the default folder.

4 Select an installation option for User Environment Manager.

Option	Description
Typical	Install the User Environment Manager FlexEngine, Application Migration, and Self-Support tool.
Custom	Manually select components to install.
Complete	Install User Environment Manager FlexEngine, Application Migration, Self-Support tool, and Management Console.

5 Select the license file and click **Next**.

6 Click **Install**, and after the installation is complete, click **Finish**.

What to do next

- Create the User Environment Manager Group Policy by using the administrative template that is provided in the User Environment Manager package. See [Configuring the FlexEngine Group Policy Object](#).
- Add FlexEngine command to logoff script. See [Configure FlexEngine to Run From a Logoff Script](#).
- Configure the Management Console. See [Configuring the User Environment Manager Management Console](#).

Unattended Installation of User Environment Manager

The User Environment Manager MSI supports unattended installations by using MSI properties to specify installation parameters. To perform an unattended installation, run the `msiexec` utility from the command line with the following properties.

Property	Description
INSTALLDIR	The absolute path to the installation directory. The default value is %ProgramFiles%\Immidio\Flex Profiles
ADDLOCAL	<p>The features that you want to install. The default values are FlexEngine, FlexMigrate, and FlexProfilesSelfSupport.</p> <p>The following values are supported:</p> <ul style="list-style-type: none"> ■ ALL. ■ FlexEngine. Installs FlexEngine. ■ FlexMigrate. Installs Application Migration and FlexEngine. ■ FlexProfilesSelfSupport. Installs the Self-Support tool and FlexEngine. ■ FlexManagementConsole. Installs the Management Console. <p>To install multiple features, separate the values with commas, without any spaces. For example, to select FlexMigrate and FlexProfilesSelfSupport:</p> <pre>ADDLOCAL="FlexMigrate, FlexProfilesSelfSupport"</pre> <p>Note The property values are case-sensitive.</p>
LICENSEFILE	<p>The path to the location of the User Environment Manager license file. The installer copies that license to the installation folder.</p> <p>Note If LICENSEFILE only contains the name of the license file, the installer looks for that file in the folder where the MSI resides.</p>

The following is an example for a custom unattended installation command:

```
msiexec.exe /i "VMware User Environment Manager 9.4 x64.msi" /qn INSTALLDIR="D:\Apps\VMware UEM"
ADDLOCAL="FlexProfilesSelfSupport" LICENSEFILE="\\filesrv1\share\VMware UEM.lic" /l* InstallUEM.log
```

The following is an example of a typical unattended installation that installs FlexEngine, Application Migration, and Self-Support in the default installation directory:

```
msiexec.exe /i "VMware User Environment Manager 9.4 x64.msi" /qn LICENSEFILE="\\filesrv1\share\VMware
UEM.lic" /l* InstallUEM.log
```

Upgrade User Environment Manager

You can upgrade to the latest version of User Environment Manager from Immidio Flex+ 8.x or from earlier versions of User Environment Manager.

To upgrade User Environment Manager, you must upgrade FlexEngine, the Management Console, and the ADMX templates in the given order.

Upgrade FlexEngine on all Windows Desktops and Terminal Servers

To upgrade User Environment Manager, you must first upgrade FlexEngine on all Windows desktops and Terminal Servers.

Prerequisites

Verify that you have an appropriate license file. If you install on VMware Horizon 7, you do not need a separate license file.

Procedure

- 1 As an administrator, run VMware User Environment Manager 9.4 x86.msi or VMware User Environment Manager 9.4 x64.msi.
- 2 Read and accept the **End User License Agreement** and click **Next**.
- 3 Select the destination folder where you want to install the application and click **Next**.
Install User Environment Manager in the default folder.
- 4 Select only FlexEngine and its subfeatures to upgrade and click **Next**.
- 5 Choose the license file and click **Next**.
- 6 Click **Upgrade**.

What to do next

[Upgrade the User Environment Manager Management Console.](#)

Upgrade the User Environment Manager Management Console

After you upgrade FlexEngine, you must upgrade the Management Console on the Windows desktop or Terminal Server it is installed on.

Prerequisites

Upgrade FlexEngine.

Procedure

- 1 As an administrator, run VMware User Environment Manager 9.4 x86.msi or VMware User Environment Manager 9.4 x64.msi.
- 2 Read and accept the **End User License Agreement** and click **Next**.
- 3 Select the destination folder where User Environment Manager is installed and click **Next**.
- 4 Select only the Management Console feature to upgrade and click **Next**.
- 5 Choose the license file and click **Next**.
- 6 Click **Upgrade**.
- 7 Select all Flex configuration files that contain Application Templates or Windows Common Settings to have them automatically updated to the new definitions. If an updated template is available, you are prompted for your approval of the update.

What to do next

[Upgrade the ADMX Templates.](#)

Upgrade the ADMX Templates

Install the ADMX templates from the VMware User Environment Manager download package to complete the upgrade process.

Prerequisites

- Upgrade FlexEngine.
- Upgrade the Management Console.

Procedure

- 1 Remove the previous User Environment Manager ADMX templates from the Central Store on the Windows domain controller.
- 2 Open the User Environment Manager download package.
- 3 Copy the new ADMX templates from the download package to the Central Store on the Windows domain controller.

You successfully upgraded User Environment Manager.

Configuring User Environment Manager

5

After you install User Environment Manager on Windows desktops or Terminal Services, you must configure FlexEngine and the Management Console.

To have User Environment Manager running correctly, you must configure FlexEngine.

- Create and configure an Active Directory GPO for User Environment Manager. You must configure Group Policies to enable FlexEngine to run when the users log on to their Windows machines, and set up the locations of the configuration and profile archives shares. The rest of the User Environment Manager Group Policies are optional.
- Configure a logoff script to enable FlexEngine to run at Windows logoff process.

If you do not use a GPO to configure FlexEngine, you can configure it by using command-line arguments. See [Chapter 7 FlexEngine Command-Line Arguments](#). Or, you can use the NoAD mode. See [Chapter 6 Installing and Configuring User Environment Manager in NoAD Mode](#).

This chapter includes the following topics:

- [Configuring the FlexEngine Group Policy Object](#)
- [Create a User Environment Manager Group Policy Object](#)
- [Configure the User Environment Manager Group Policy Object](#)
- [Configuring FlexEngine to Run from Logon and Logoff Scripts](#)
- [Configuring the User Environment Manager Management Console](#)

Configuring the FlexEngine Group Policy Object

You configure FlexEngine by creating an Active Directory Group Policy Object (GPO). You use the User Environment Manager administrative templates that are provided in the download package.

If you want to provide different FlexEngine configurations, you can use multiple GPOs. For example, you can manage multiple User Environment Manager environments such as test and production.

GPO Mandatory Settings

After you deploy FlexEngine to the client devices, you must configure FlexEngine to run during the Windows logon and logoff processes.

FlexEngine runs during Windows logon to get all the settings for the client device and apply them as soon as the user logs in.

You can configure FlexEngine in the following ways to run during Windows logon process:

- By setting Group Policy Run FlexEngine as Group Policy Extension. See [Configure FlexEngine to Run as Group Policy Extension Setting](#).
- By configuring a logon script in Group Policy. Use this method if you prefer to write a logon script. See [Configure FlexEngine to Run from a Logon Script](#).

FlexEngine runs again during Windows logoff to save all the settings for the client device to the profile archives share. To run FlexEngine during Windows logoff, configure a logoff script in Group Policy. See [Configure FlexEngine to Run From a Logoff Script](#).

You must also configure the path to the configuration share and the profile archives share in the GPO. See [Configure the Flex Configuration Files Setting](#) and [Configure Profile Archives Setting](#).

GPO Optional Settings

The two most common optional GPO settings for User Environment Manager are as follows:

- Use the Profile Archives Backups setting to configure the location and number of backups. Users can restore their settings from a backup using the Self-Support Tool or help desk personnel can do this by using the Helpdesk Support tool.
- Use the FlexEngine Logging setting to configure the location and filename of the log file, the level of log detail, and the maximum file size. The log file helps with troubleshooting.

For more information on all FlexEngine GPO settings, see [Configure the User Environment Manager Group Policy Object](#) for details.

Create a User Environment Manager Group Policy Object

To configure FlexEngine, you must create a GPO for User Environment Manager by using the administrative templates that are provided in the download package.

You can create a new GPO for User Environment Manager or use an existing one that is applied to the users for which you want to configure FlexEngine.

Procedure

- 1 Copy the User Environment Manager ADMX files and their corresponding ADML files from the download package to the correct PolicyDefinitions folder on your Windows Domain Controller.

The User Environment Manager ADMX files are located in the Administrative Templates (ADMX) folder in the download package, and their corresponding ADML files are located in the Administrative Templates (ADMX)\en-US folder. For more information about the location of the PolicyDefinitions folder, go to the [Microsoft Web site](#).

- 2 Open the Group Policy Management Console.
- 3 Create a new GPO.

What to do next

- Configure the appropriate User Environment Manager Group Policy settings. See [Configure the User Environment Manager Group Policy Object](#).
- Configure FlexEngine to run during logout by configuring a logoff script. See [Configure FlexEngine to Run From a Logoff Script](#).

Configure the User Environment Manager Group Policy Object

After you create a User Environment Manager GPO, you must configure its settings. These settings are required to configure the location of the User Environment Manager configuration and profile archives shares, and configure FlexEngine to start automatically during login.

Prerequisites

You must at least configure the following settings:

- Flex configuration files.
- Profile archives.
- Run FlexEngine as Group Policy Extension.

If you do not configure Run FlexEngine as Group Policy Extension, you must configure FlexEngine to run using a logon script. See [Configure FlexEngine to Run from a Logon Script](#).

The rest of the User Environment Manager GPO settings are optional and enabling them depends on your infrastructure and requirements.

Note Many of the FlexEngine settings that you configure through a GPO can be overridden by command-line arguments. Command-line arguments have higher priority than GPO settings. See [Chapter 7 FlexEngine Command-Line Arguments](#).

Procedure

1 [Open the User Environment Manager Group Policy Object](#)

To configure User Environment Manager, you must edit the settings of the User Environment Manager GPO that you created. Open the settings of the GPO from the Group Policy Management Editor. The User Environment Manager GPO configures FlexEngine with the correct User Environment Manager share locations and is therefore a required step.

2 [Configure the Flex Configuration Files Setting](#)

You configure the location of the central share that stores the Flex configuration files in the **Flex config files** setting. Flex configuration files contain User Environment Manager data that FlexEngine uses to read and store user settings. FlexEngine runs with the user's credentials, and processes each Flex configuration file for which the user has NTFS read access.

3 [Configure FlexEngine to Run as Group Policy Extension Setting](#)

You can enable FlexEngine to run automatically during logon by running as a Group Policy client-side extension.

4 [Configure FlexEngine Logging Setting](#)

You can configure the location and file name of the FlexEngine log file, the level of logging detail, and the maximum size of the log file.

5 [Configure Profile Archives Setting](#)

Configure the location of the profile archives share from where FlexEngine reads and stores user profile archives and other settings that are related to the profile archives.

6 [Configure Profile Archive Backups Setting](#)

Use the **Profile Archive Backups** setting to configure the location where FlexEngine stores the backups of profile archives.

7 [Configure Application Blocking Logging to the Windows Event Log Setting](#)

You can enable **Application blocking logging to the Windows event log** to have the details on blocked application launches logged to the Windows event log.

8 [Configure Privilege Elevation Logging to the Windows Event Log Setting](#)

You can configure User Environment Manager to log the details of elevated application launches and, if desired, de-elevated child processes.

9 [Configure Certificate Support for Mandatory Profiles Setting](#)

You can enable the use of personal certificates in a mandatory profile. In addition to enabling this support, you also must create a Flex configuration file with the Personal Certificates Windows Common Setting.

10 [Configure DirectFlex - Advanced Settings](#)

You can configure advanced DirectFlex settings for more fine-grained control over DirectFlex export settings and visual feedback.

11 [Configure FlexEngine Logging to the Windows Event Log Setting](#)

Use FlexEngine logging to the Windows Event Log to configure the events that FlexEngine logs to the Windows event log. When this setting is enabled, FlexEngine logs informational messages to the event log indicating the start and finish of path-based import and export actions.

12 [Configure Paths Unavailable at Logon Setting](#)

You can configure **Paths Unavailable at Logon** to determine the behavior if the Flex configuration files path or profile archives path is unavailable at login.

13 [Configure FlexEngine Refresh Settings](#)

Use **FlexEngine Refresh Settings** when you run FlexEngine from a logon script. When you run FlexEngine as Group Policy extension, this setting is ignored.

14 [Configure Access to VMware UEM Self-Support for End Users](#)

You can control whether users have access to VMware User Environment Manager Self-Support.

15 [Configure User Environment Manager Logon and Logoff Progress Information](#)

You can configure **Show VMware UEM logon and logoff progress information** to show a splash screen with progress bar when FlexEngine runs.

16 [Configure Silo-Specific Flex Configuration Files Settings](#)

Use the **Silo-Specific Flex Config Files** setting to specify an additional, silo-specific path with Flex configuration files that is processed in addition to the ones configured through the **Flex config files** setting.

Open the User Environment Manager Group Policy Object

To configure User Environment Manager, you must edit the settings of the User Environment Manager GPO that you created. Open the settings of the GPO from the Group Policy Management Editor. The User Environment Manager GPO configures FlexEngine with the correct User Environment Manager share locations and is therefore a required step.

Prerequisites

Create a GPO for FlexEngine. See [Create a User Environment Manager Group Policy Object](#).

Procedure

- 1 Open the Group Policy Management console.
- 2 Right-click the User Environment Manager GPO that you created and select **Edit**.
The Group Policy Management Editor opens.
- 3 To access the User Environment Manager administrative templates, navigate to **User configuration > Policies > Administrative Templates > VMware UEM > FlexEngine**.

What to do next

Configure the following FlexEngine mandatory settings:

- Flex configuration files.
- Profile archives.
- Run FlexEngine as Group Policy Extension. Either enable this setting or configure FlexEngine to run from a logon script. See [Configure FlexEngine to Run from a Logon Script](#).
- Configure FlexEngine to run from a logoff script. See [Configure FlexEngine to Run From a Logoff Script](#).

Configure the Flex Configuration Files Setting

You configure the location of the central share that stores the Flex configuration files in the **Flex config files** setting. Flex configuration files contain User Environment Manager data that FlexEngine uses to read and store user settings. FlexEngine runs with the user's credentials, and processes each Flex configuration file for which the user has NTFS read access.

Procedure

- 1 In the Group Policy Management Editor, double-click the **Flex config files** setting.
- 2 Select **Enabled**.
- 3 In the **Central location of Flex config files** text box, enter the location of the central configuration share.

Use a UNC path for this setting. Typically, this path points to the General folder created by the Management Console in the User Environment Manager configuration share.

For example: \\Filesrv\UemConfig\$\General

- 4 Select **Process folder recursively** to also enable processing Flex configuration files that are located in subfolders of the specified path.

Configure FlexEngine to Run as Group Policy Extension Setting

You can enable FlexEngine to run automatically during logon by running as a Group Policy client-side extension.

By running FlexEngine as Group Policy Extension, settings that User Environment Manager manages are applied earlier during the logon phase rather than when running FlexEngine from a logon script. This way, the range of settings that can be managed by User Environment Manager is extended, such as the Windows Multilanguage User Interface or slideshow backgrounds.

Prerequisites

- Enable the **Always wait for the network at computer startup and logon** Computer Group Policy setting to ensure that the FlexEngine Group Policy client-side extension runs during each logon. Apply this setting to an OU in Active Directory where all the Windows clients are located.
- Configure a logon script through Group Policy to run FlexEngine with the `-OfflineImport` argument. This is needed to ensure that FlexEngine still runs at logon when a computer is offline and a user logs on with cached credentials, because Group Policy client-side extensions do not run in such a scenario. See [Additional FlexEngine Operations](#).
- Configure FlexEngine logout commands through a Group Policy logoff script as described in [Configure FlexEngine to Run From a Logoff Script](#). FlexEngine Group Policy client-side extension runs only during logon.

Procedure

- 1 In the Group Policy Management Editor, double-click the **Run FlexEngine as Group Policy Extension** setting.
- 2 Select **Enabled**.

Configure FlexEngine Logging Setting

You can configure the location and file name of the FlexEngine log file, the level of logging detail, and the maximum size of the log file.

Procedure

- 1 In the Group Policy Management Editor, double-click the **FlexEngine Logging** setting.
- 2 Select **Enabled**.
- 3 Enter the settings for FlexEngine logging.

Option	Description
Path and name of log file	Enter a location that is unique for each user, for example: \\Filesrv\Users\$\%username%\Logs\FlexEngine.Log If you enter a subdirectory that does not exist, FlexEngine automatically creates it when a user logs in.
Log level	Set the amount of detail that is logged. Do not use Debug or Info in production environments, because the amount of logging information might slow down the logon and logoff process.
Maximum log file size in kB	Set the maximum size of the log file. If you set a maximum log file size, the log file is created again after that size is reached. If you set the maximum size to 0, the log file grows indefinitely.
Log total size of profile archive and profile archive backups folders	FlexEngine logs the number of profile archives and profile archives backups, and their file sizes at the end of a path-based export.

Note You should avoid using Debug log level in production, but it is extremely helpful when troubleshooting issues.

Configure Profile Archives Setting

Configure the location of the profile archives share from where FlexEngine reads and stores user profile archives and other settings that are related to the profile archives.

Procedure

- 1 In the Group Policy Management Editor, double-click the **Profile Archives** setting.
- 2 Select **Enabled**.
- 3 Configure the settings for storing the profile archives.

Option	Description
Location for storing user profile archives	Enter the location of the profile archives share. Use a location that is unique for each user, for example: \\Filesrv\UemUsers\$\%username%\Archives
Hide profile archives folder	Mark the specified profile archives folder as hidden after FlexEngine performs a path-based export.
Compress profile archives	Enable ZIP compression for the user profile archives.
Retain file modification dates	Restore last modified dates when FlexEngine imports profile archives. This setting is required if you want to use the Do not export files older than ... days function.

Configure Profile Archive Backups Setting

Use the **Profile Archive Backups** setting to configure the location where FlexEngine stores the backups of profile archives.

Procedure

- 1 In the Group Policy Management Editor, double-click the **Profile Archive Backups** setting.
- 2 Select **Enabled**.
- 3 Enter the settings for storing profile archives backups.

Option	Description
Location for storing user profile archive backups	Provide a unique location to store the profile archives backups for every user. For example: <code>\\Filesrv\UemUsers\$\%username%\Backups</code> If you enter a subdirectory that does not exist, FlexEngine automatically creates a subdirectory when a user creates a backup for the first time.
Hide backup folder	Mark the specified profile archives backup folder as hidden after a path-based export.
Number of backups per profile archive	Specify the number of backups you want to create for each profile archive for each user. Note You can override this setting in the Flex configuration files by setting a different value on the Backups tab in the Management Console.
Create single backup per day	Treat the number of backups as the number of days for which to keep backups. This prevents DirectFlex from overwriting backups from the previous day or older days.

Configure Application Blocking Logging to the Windows Event Log Setting

You can enable **Application blocking logging to the Windows event log** to have the details on blocked application launches logged to the Windows event log.

Procedure

- 1 In the Group Policy Management Editor, double-click **Application blocking logging to the Windows event log**.
- 2 Select **Enabled**.

Configure Privilege Elevation Logging to the Windows Event Log Setting

You can configure User Environment Manager to log the details of elevated application launches and, if desired, de-elevated child processes.

The default behavior of this setting is to log the details of the following events to the Windows event log.

- An application privilege is elevated.
- An elevated application launches a de-elevated child process.

Procedure

- 1 In the Group Policy Management Editor, double-click **Privilege elevation logging to the Windows event log**.
- 2 Select **Enabled**.
- 3 (Optional) If you do not want User Environment Manager to log de-elevated child processes, deselect **Log de-elevated application launches**.
- 4 Click **OK**.

Configure Certificate Support for Mandatory Profiles Setting

You can enable the use of personal certificates in a mandatory profile. In addition to enabling this support, you also must create a Flex configuration file with the Personal Certificates Windows Common Setting.

Note Do not enable this setting when you are using roaming or local profiles.

Procedure

- 1 In the Group Policy Management Editor, double-click **Certificate Support for Mandatory Profiles**.
- 2 Select **Enabled**.

Configure DirectFlex - Advanced Settings

You can configure advanced DirectFlex settings for more fine-grained control over DirectFlex export settings and visual feedback.

Procedure

- 1 In the Group Policy Management Editor, double-click **DirectFlex - Advanced Settings**.
- 2 Select **Enabled**.
- 3 Configure **DirectFlex - Advanced Settings** settings.

Option	Description
Only export at logoff	By default, DirectFlex exports profile information when an application is closed. When you enable this setting, the export action runs when the user logs out. Note This setting can be overridden through the DirectFlex settings in the Management Console.
Show DirectFlex notifications	Enable this option to display a message in the notification area when DirectFlex performs an import or export.

Option	Description
Notification delay in seconds	If the DirectFlex import or export takes less time than the configured delay, no message appears. Configure this setting to only display messages when the access to the profile archives path is slow. If the delay is set to 0, messages are shown immediately.
Hide DirectFlex exit notification	Enable this option to only show a message when DirectFlex is performing an import.

Configure FlexEngine Logging to the Windows Event Log Setting

Use FlexEngine logging to the Windows Event Log to configure the events that FlexEngine logs to the Windows event log. When this setting is enabled, FlexEngine logs informational messages to the event log indicating the start and finish of path-based import and export actions.

Procedure

- 1 In the Group Policy Management Editor, double-click the **FlexEngine logging to the Windows Event Log** setting.
- 2 Select **Enabled**.
- 3 (Optional) Configure settings for additional settings.

Option	Description
Asynchronous UEM actions DirectFlex refresh UEM refresh	Enable these options to instruct FlexEngine to log start and finish events for these features.
Warn if size of single profile archive exceeds this size in kB	If you set a size other than 0, FlexEngine logs an event whenever a profile archive that is exported is larger than the specified size in kB. This applies both to DirectFlex exports and path-based exports.
Warn if size of profile archive folder exceeds this size in kB	If you set a size other than 0, FlexEngine logs an event whenever the total size of profile archives in the profile archives folder is larger than the specified size in kB. This size check only takes place after a path-based export.
Include profile archive backup folder when determining folder size	When set, the size of profile archives backups in the backup folder is taken into account when computing the size of the profile archives folder.

Configure Paths Unavailable at Logon Setting

You can configure **Paths Unavailable at Logon** to determine the behavior if the Flex configuration files path or profile archives path is unavailable at login.

Procedure

- 1 In the Group Policy Management Editor, double-click the **Paths Unavailable at Logon** setting.
- 2 Select **Enabled**.

3 Configure the **Paths Unavailable at Logon** options.

Option	Description
If Flex config files path is not available	<p>Select one of the following options when the Flex configuration files path is not available at login.</p> <ul style="list-style-type: none"> ■ Skip import. Allows the user to log in. The user profile archives are not imported and the user environment settings are not applied. ■ Logoff. Automatically logs off the user.
If profile archive path is not available	<p>Select one of the following options when the profile archives path is not available at login.</p> <ul style="list-style-type: none"> ■ Skip import. Allows the user to log in. The user profile archives are not imported and the user environment settings are not applied. ■ Apply user environment settings. Applies the user environment settings. The user profile archives are not imported. ■ Logoff. Automatically logs off the user.
Optional message to display	<p>Use this setting to display a message in case the path is missing. You can configure this setting separately for the Flex configuration files path, if Skip import is selected, and for the profile archives path, if Skip import or Apply user environment settings is selected.</p>
Timeout after which to dismiss message	<p>Use this setting to configure how long the message remains displayed. The user can dismiss the message manually at any time.</p>

Note If the Flex configuration files path is not available, the user is immediately logged out by default. If the profile archives path is not available, only the user environment settings are applied by default and no user profile archives are imported.

Configure FlexEngine Refresh Settings

Use **FlexEngine Refresh Settings** when you run FlexEngine from a logon script. When you run FlexEngine as Group Policy extension, this setting is ignored.

Procedure

- 1 In the Group Policy Management Editor, double-click **FlexEngine Refresh Settings**.
- 2 Select **Enabled**.
- 3 Configure the options for **FlexEngine Refresh Settings**.

Option	Description
Refresh Windows appearance	<p>Refresh Windows appearance-related settings at logon, such as the desktop wallpaper. In addition, you must create a Flex configuration file from the Management Console that contains the appropriate information to be processed by FlexEngine.</p>
Refresh mouse settings	<p>Refresh mouse-related settings at logon. In addition, you must create a Flex configuration file that contains the Mouse Windows Common Setting.</p>
Refresh keyboard settings	<p>Refresh keyboard-related settings at logon. In addition, you must create a Flex configuration file that contains the Keyboard Windows Common Setting.</p>

Configure Access to VMware UEM Self-Support for End Users

You can control whether users have access to VMware User Environment Manager Self-Support.

Procedure

- 1 In the Group Policy Management Editor, double-click **Prevent access to VMware UEM Self-Support**.
- 2 Select **Enabled**.

Configure User Environment Manager Logon and Logoff Progress Information

You can configure **Show VMware UEM logon and logoff progress information** to show a splash screen with progress bar when FlexEngine runs.

Procedure

- 1 In the Group Policy Management Editor, double-click **Show VMware UEM logon and logoff progress information**.
- 2 Select **Enabled**.

Configure Silo-Specific Flex Configuration Files Settings

Use the **Silo-Specific Flex Config Files** setting to specify an additional, silo-specific path with Flex configuration files that is processed in addition to the ones configured through the **Flex config files** setting.

Procedure

- 1 In the Group Policy Management Editor, double-click **Silo-Specific Flex Config Files**.
- 2 Select **Enabled**.
- 3 Configure the silo-specific settings.

Option	Description
Central location of silo-specific Flex config files	Use a UNC path for this setting. Typically this points to a silo subfolder of the Silos folder created by the Management Console in the UEM configuration share, for example: <code>\\Filesrv\UemConfig\$\Silos\ts-1</code>
Silo-specific suffix	To distinguish profile archives corresponding to Flex configuration files in the configured silo-specific folder, this suffix is used as a subfolder in the configured profile archives and backup paths. If you do not set a suffix, the last component of the silo-specific Flex configuration files path is used, ts-1 in the example above.

Configuring FlexEngine to Run from Logon and Logoff Scripts

For the User Environment Manager to run correctly, FlexEngine must run during the logon and logoff process. To run FlexEngine at logoff, you must run the FlexEngine logoff command from a logoff script. To run FlexEngine at logon, you can either configure Flex Engine as Group Policy client-side extension, or run the FlexEngine logon command from a logon script.

For information about how to run FlexEngine as Group Policy client-side extension, see [Configure FlexEngine to Run as Group Policy Extension Setting](#).

Configure FlexEngine to Run from a Logon Script

If you do not configure FlexEngine to run as a Group Policy client-side extension, you must configure FlexEngine to run from a logon script.

If you configure FlexEngine to run as a Group Policy client-side extension, you do not need to configure FlexEngine to run from a logon script. See [Configure FlexEngine to Run as Group Policy Extension Setting](#).

However, in case a computer is offline and a user logs in with cached credentials, Group Policy client-side extensions do not run. If FlexEngine is configured to run as a Group Policy extension, no import takes place. Therefore, configure a logon script with the `-OfflineImport` argument to enable importing of user settings at login even when the user's computer is offline.

You can add the FlexEngine logon command to an existing logon script or call it directly as a logon script. For this purpose, use `User Configuration\Windows Settings\Scripts`.

Configure the following command that runs during logon:

```
C:\Program Files\Immidio\Flex Profiles\FlexEngine.exe -r
```

This command reads the settings that are configured through the User Environment Manager Group Policy Object and performs the path-based import accordingly.

User Environment Manager manages profile information that often must be imported before the Windows shell is initialized. To make sure that the shell initialization waits until the logon script has completed, enable the **Run logon scripts synchronously** Windows Group Policy setting. This setting is located at `User Configuration\Policies\Administrative Templates\System\Scripts`.

Note On Windows 7 and Windows Server 2008 R2, the Microsoft Windows GPO ignores the **Run logon scripts synchronously** policy setting when using mandatory profiles. When using local or roaming profiles, the policy setting is ignored the first time a user logs in. Microsoft hotfix 2550944 addresses this issue.

Configure FlexEngine to Run From a Logoff Script

You must always configure the FlexEngine logoff command to run from a logoff script. You can add the FlexEngine logoff command to an existing logoff script or call it directly as if it were a logoff script. Use `User Configuration\Windows Settings\Scripts` for this purpose.

Configure the following logoff command that will run during the logoff process:

```
C:\Program Files\Immidio\Flex Profiles\FlexEngine.exe -s
```

This command reads the settings configured through the User Environment Manager Group Policy Object and performs the path-based export accordingly.

Configuring the User Environment Manager Management Console

After you install and configure User Environment Manager, you can start the User Environment Manager Management Console. The Management Console requires initial configuration and you can also configure which features you want to manage through the console.

Initial Configuration

When you start the Management Console for the first time, you must specify the location of the configuration share. With the configuration share set, you can start using the Management Console.

Management Console searches for the `Immidio Flex Profiles Configuration.xml` configuration file at the specified location. If the file exists, this central file is read and used as configuration for the Management Console. Otherwise, a new configuration file, with default values, and a `General` folder is created.

Note Management Console configuration is stored in the central configuration share. Any changes that you make to the configuration affect all Management Console installations that are configured to use this share.

Further Configuration

You can control which features you want to manage through the Management Console by configuring the Management Console settings. You can display the settings by clicking **Configure** in the Management Console.

- Personalization features. Each check box from the Personalization features corresponds to a tab in the Management Console, except the **Silo support** check box. Silo support lets you use Management Console to manage both General configuration files and silo-specific ones. When enabled, a `Silos` folder is created in the User Environment Manager configuration share. Within the `Silos` folder, you can create subfolders for each silo, and within these subfolders you can create and manage silo-specific Flex configuration files.
- App-V and SWV configuration. The App-V and SWV tabs contain settings for application virtualization support. For Symantec Workspace Virtualization 7.5 there is only a single setting, which enables or disables support. For more information, see the *VMware User Environment Manager Administration Guide*.

Installing and Configuring User Environment Manager in NoAD Mode

6

The NoAD mode is an alternative to configuring User Environment Manager with Active Directory Group Policy. You do not need to create a GPO, logon and logoff scripts, or configure Windows Group Policy settings.

In NoAD mode, all User Environment Manager GPO settings are ignored. If settings from a previous GPO-based deployment are encountered, no actions are performed and a message is logged to the FlexEngine log file.

This chapter includes the following topics:

- [Install User Environment Manager in NoAD Mode](#)
- [Configuring User Environment Manager with the NoAD Mode](#)

Install User Environment Manager in NoAD Mode

You install User Environment Manager in NoAD mode by performing unattended installation through a command-line interface.

Procedure

- ◆ To install User Environment Manager in NoAD mode, specify the path to the Flex configuration files through the NOADCONFIGFILEPATH MSI property at the time of installation.

```
msiexec.exe /i "VMware User Environment Manager 9.4 x64.msi" /qn LICENSEFILE="\\filesrv1\share\VMware UEM.lic" /l* InstallUEM.log NOADCONFIGFILEPATH=\\Filesrv\UemConfig$\General
```

This command inserts the basic NoAD configuration in the HKLM registry hive and enables the NoAD mode.

Note To disable the NoAD mode, uninstall User Environment Manager, and reinstall without the NOADCONFIGFILEPATH MSI property.

Configuring User Environment Manager with the NoAD Mode

You can provide the settings for configuring User Environment Manager with the NoAD mode through an XML file on the central configuration share. When a user logs in, FlexEngine reads the settings from the XML file and applies them to the registry.

The XML file is called `NoAD.xml` and must reside in the `...\General\FlexRepository\NoAD` subfolder.

The `NoAD.xml` is a UTF-8-encoded XML file with an explicit UTF-8 BOM. You can create this file by using Notepad and selecting the UTF-8 encoding when saving. The file must have the following basic structure:

```
<?xml version="1.0" encoding="utf-8"?>
<userEnvironmentSettings>
    <setting type="noAD" additionalSetting="..." otherSetting="..." ... />
</userEnvironmentSettings>
```

Configuring FlexEngine Logging Settings

You can configure the location and filename of the FlexEngine log file, the level of logging details, and the maximum size of the log file.

Setting	Value
LogFileNames	Set a location that is unique for each user, for example: <code>\\Filesrv\Users\$\%username%\Logs\FlexEngine.log</code> If you enter a subdirectory that does not exist, FlexEngine creates it when a user logs in.
LogLevel	Configure the amount of details to log by setting this property to one of the following values: <ul style="list-style-type: none"> ■ 0 (DEBUG) ■ 1 (INFO) ■ 2 (WARN) ■ 3 (ERROR) ■ 4 (FATAL) <p>Note Do not use 0 (DEBUG) or 1 (INFO) in production environments, because the amount of logging information might slow down the logon and logoff process.</p>
MaximumLogFileSize	Set the maximum size of the log file in kB. If you set a maximum log file size, the log file is created again after that size limit is reached. If you set the maximum size to 0, the log file expands indefinitely.
LogProfileArchiveFolderSize	To enable this setting, set the value to 1. If enabled, FlexEngine logs the number of profile archives and profile archives backups, and their sizes, at the end of a path-based export.

Configuring Profile Archives Settings

Configure the location of the profile archives share from where the FlexEngine reads and stores user profile archives and other settings related to profile archives.

Setting	Value
ProfileArchivePath	Set the location of the profile archives share. Use a location that is unique for each user. For example: \\Filesrv\UemUsers\$\%username%\Archives
HideProfileArchiveFolder	To enable this setting, set the value to 1. This setting marks the specified profile archives folder hidden after FlexEngine performs a path-based export.
RestoreLastModified	To enable this setting, set the value to 1. If enabled, FlexEngine restores last modified dates when it imports profile archives. This setting is required if you want to use the Do not export files older than ... days option.

Configuring Profile Archive Backups Settings

Use this setting to configure the location where FlexEngine stores the backups of profile archives.

Setting	Value
BackupPath	Set a unique location to store the profile archives backups for every user. For example: \\Filesrv\UemUsers\$\%username%\Backups If you enter a subdirectory that does not exist, FlexEngine creates a subdirectory when a user creates a backup for the first time.
HideBackupFolder	To enable this setting, set the value to 1. Marks the specified profile archives backup folder as hidden after a path-based export.
BackupCount	Set the number of backups that you want to create for each profile archive for each user. Note You can override this setting in the Flex configuration files by setting a different value on the Backups tab in the Management Console.
BackupDaily	To enable this setting, set the value to 1. Treats the number of backups as the number of days for which to retain backups. This prevents DirectFlex from overwriting backups from the previous day or recent days.

Configuring Application Blocking Logging to the Windows Event Log Setting

As an option, you can enable this setting to log the details of the blocked application launches to the Windows event log.

Setting	Value
AppBlockingEventLog	To enable this setting, set the value to 1.

Configuring Privilege Elevation Logging to the Windows Event Log Setting

You can configure User Environment Manager to log the details of elevated application launches and, if desired, de-elevated child processes.

The default behavior of the `PrivilegeElevationEventLog` setting is to log the details of the following events to the Windows event log.

- An application privilege is elevated.
- An elevated application launches a de-elevated child process.

Setting	Value
<code>PrivilegeElevationEventLog</code>	To enable this setting, set the value to 1.
<code>DeElevationEventLog</code>	If the value of the <code>PrivilegeElevationEventLog</code> setting is 1 but you do not want User Environment Manager to log de-elevated child processes, too, set this value to 0.

Enable Certificate Support for Mandatory Profiles Setting

As an option, you can enable the use of personal certificates in a mandatory profile.

Setting	Value
<code>CertificateSupport</code>	To enable this setting, set the value to 1.

Note Do not enable this setting when you are using roaming or local profiles.

Configure DirectFlex - Advanced Settings

As an option, you can configure advanced DirectFlex settings for fine-grained control over DirectFlex export settings and visual feedback.

Setting	Value
<code>OnlyExportAtLogoff</code>	To enable this setting, set the value to 1. By default, DirectFlex exports profile information when an application is closed. When you enable this setting, the export action runs when the user logs out. Note This setting can be overridden through the DirectFlex settings in the Management Console.
<code>DirectFlexNotification</code>	To enable this setting, set the value to 1. When enabled, a message is displayed in the notification area when DirectFlex performs an import or export.
<code>DirectFlexNotificationDelay</code>	Set the number of seconds to delay. If the DirectFlex import or export takes less time than the configured delay, no message appears. Configure this setting to display messages only when the access to the profile archives path is slow. If the delay is set to 0, messages are shown immediately.
<code>DirectFlexHideExitNotification</code>	To enable this setting, set the value to 1. Enable this option to show a message only when DirectFlex is performing an import.

Configure FlexEngine Logging to the Windows Event Log Setting

As an option, you can use FlexEngine logging to the Windows event log to configure the events FlexEngine logs to the Windows event log. When this setting is enabled, FlexEngine logs informational messages to the event log indicating the beginning and end of the path-based import and export actions.

Setting	Value
EventLog	To enable this setting, set the value to 1. If enabled, FlexEngine logs events for path-based import and export actions.
EventLogAsync	To enable this setting, set the value to 1. If enabled, FlexEngine also logs events for asynchronous User Environment Manager actions.
EventLogDirectFlexRefresh	To enable this setting, set the value to 1. If enabled, FlexEngine also logs events for DirectFlex refresh actions.
EventLogUEMRefresh	To enable this setting, set the value to 1. If enabled, FlexEngine also logs events for user environment refresh actions.
EventLogMaxFileSize	Set to profile archive file size in kB. If you set a size other than 0, FlexEngine logs an event when a profile archive that is exported is larger than the specified size in kB. This applies both to DirectFlex exports and path-based exports.
EventLogMaxFolderSize	Set to profile archive folder size in kB. If you set a size other than 0, FlexEngine logs an event whenever the total size of profile archives in the profile archives folder is larger than the specified size in kB. This size check only takes place after a path-based export.
EventLogIncludeBackupFolder	To enable this setting, set the value to 1. When set, the size of profile archives backups in the backup folder is taken into account when computing the size of the profile archives folder.

Note EventLog must be set to 1 for any of the other settings to take effect.

Configure Paths Unavailable at Logon Setting

As an option, you can configure this setting to determine the behavior if the Flex configuration path or profile archives path is unavailable at login.

Setting	Value
ConfigPathMissingAction	Set to one of the following values when the Flex configuration files path is not available at login. <ul style="list-style-type: none"> ■ 0 (Skip import). Allows the user to log in. The user profile archives are not imported and the user environment settings are not applied. ■ 1000 (Logoff). Automatically logs out the user.
ConfigPathMissingMessage	Set a message to display. Use this setting to display a message in case the Flex configuration path is missing and ConfigPathMissingAction is set to 0 (Skip import).
ConfigPathMissingMessageTimeout	Set a timeout in seconds. Use this setting to configure how long the message remains displayed. The user can close the message manually at any time.
ArchivePathMissingAction	Set to one of the following values when the profile archives path is not available at logon: <ul style="list-style-type: none"> ■ 0 (Skip import). Allows the user to log in. The user profile archives are not imported and the user environment settings are not applied. ■ 1 (Apply user environment settings). Applies the user environment settings. The user profile archives are not imported. ■ or 1000 (Logoff). Automatically logs off the user.

Setting	Value
ArchivePathMissingMessage	Set a message to display. Use this setting to display a message in case the profile archives path is missing and ArchivePathMissingAction is set to 1 (Apply user environment settings).
ArchivePathMissingMessageTimeout	Set a timeout in seconds. Use this setting to configure how long the message remains displayed. The user can close the message manually at any time.

Note If the Flex configuration files path is not available, the user is immediately logged off by default. If the profile archives path is not available, only the user environment settings are applied by default and no user profile archives are imported.

Prevent Access to User Environment Manager Self-Support to End Users

As an option, you can control user access to VMware User Environment Manager Self-Support.

Setting	Value
SelfSupportDisallowed	To enable this setting, set the value to 1.

Disable the NoAD Mode for Certain Users

As an option, you can use the following additional settings to control the NoAD mode.

Setting	Value
DisableForAdmin	Set this value to 1 to disable all User Environment Manager functionality for local administrators.
DisableForGroupMembers	Set this value to a comma-separated list of groups in Domain\GroupName format. All User Environment Manager functionality is disabled for members of those groups.

Remove Registry Settings for the NoAD Mode

Settings in the NoAD.xml file overwrite existing registry settings. As an option, you can remove a registry setting by specifying an attribute value of -/- in the XML attribute.

Sample NoAD.xml File

The following sample NoAD.xml file configures the profile archive path, a log file at the INFO level, allows users to log in if the Flex configuration files path or the profile archive path is not accessible, and enables certain Windows event logging options.

```
<?xml version="1.0" encoding="utf-8"?>
<userEnvironmentSettings>
  <setting type="noAD"
    ProfileArchivePath="\\Filesrv\UemUsers$\%username%\Archives"
    LogFileName="\\Filesrv\UemUsers$\%username%\Logs\FlexEngine.log"
    LogLevel="1"
    ConfigPathMissingAction="0"
```

```
ArchivePathMissingAction="1"  
AppBlockingEventLog="1"  
EventLog="1"  
EventLogAsync="1"  
EventLogDirectFlexRefresh="1"  
EventLogUEMRefresh="1"  
  />  
</userEnvironmentSettings>
```

Note This sample configuration file is available in the NoAD Mode folder in the download package.

FlexEngine Command-Line Arguments

7

When FlexEngine starts, it first reads its Group Policy configuration and then FlexEngine reads the command-line arguments specified.

Note Command-line arguments override the settings provided through Group Policy.

This chapter includes the following topics:

- [FlexEngine Operations and Arguments](#)
- [Operation Modes](#)
- [Overriding Group Policy Settings](#)
- [Applying Horizon Smart Policies in Multiple Sessions](#)
- [Additional FlexEngine Operations](#)

FlexEngine Operations and Arguments

FlexEngine performs operations at logon and logoff events. You can pass different arguments to FlexEngine to configure User Environment Manager.

FlexEngine Operations

FlexEngine can perform two main operations:

- `-s` stores profile information.
- `-r` reads profile information.

Flex Engine Arguments

You can further configure the operations by using additional arguments, some of which are optional (●), some of which mandatory (x):

Argument	-s	-r	Description
<code>-i</code>	x	x ¹	Argument value <code>config</code> specifies the Flex configuration files to use.
<code>-S</code>	● ²	● ²	First argument value <code>silopath</code> specifies silo-specific configuration files to use. Second argument value <code>suffix</code> specifies suffix to use in profile archives and backup folders, if <code>suffix</code> is an empty string, the last folder of <code>silopath</code> is used as suffix.

Argument	-s	-r	Description
-R	●	●	Indicates that the <code>config</code> directory (and the <code>silos</code> path directory, if configured) should be processed recursively; that is, also processes Flex configuration files in subfolders. Ignored for file-based operations.
-b	●		Argument value <code>backup_path</code> specifies the location for profile archives backups. If not specified, no backups are created.
-B	●		Argument value <code>backup_count</code> indicates how many backups to create.
-Bd	●		Indicates that a single backup should be created per day.
-C	●		Enables compression for profile archives. FlexEngine can read both compressed and uncompressed archives regardless of this setting. This switch only controls the creation of archives.
-c	● ²	● ²	Enables certificate support for mandatory profiles.
-F	● ²	● ²	Indicates that export of the settings should be performed always, regardless of whether the previous import was successful.
-M	● ²	● ²	Argument value <code>log_size</code> specifies the maximum size of the log file in kB. If the log file is larger than that size at the start of an import or export action, the log file is cleared.
-L	●		Restores the last modified dates for imported files.
-H	● ²		Marks the profile archives folder and the backup folder, if configured, as hidden after a path-based export.
-v	●	●	Verbose mode: displays a splash screen with a progress bar.
-f	●	●	Argument value <code>log_file</code> specifies the name of the log file.
-l	●	●	Argument value <code>log_level</code> specifies what information should be logged. <code>log_level</code> can be DEBUG, INFO, WARN (the default), ERROR, or FATAL.
-rw		● ²	Refreshes Windows appearance.
-rk		● ²	Refreshes keyboard settings.
-rm		● ²	Refreshes mouse settings.
-ra		● ²	Combination of the three options above.

Legend:

- x** Mandatory.
- Optional.
- ¹ Mandatory for path-based, not applicable for file-based.
- ² Optional for path-based, not applicable for file-based.

Operation Modes

The `-s` and `-r` operations can operate on a single profile archives, which is called file-based mode. They can also operate on a directory of profile archives, which is called path-based mode. This operation mode

is determined from the path that is specified as the next argument, `-s \\...\ie.zip` as compared to `-s \\...\...\Archives` for instance. If no path is specified, the profile archives policy setting is used.

For a particular operation mode, some of the other arguments must follow the same pattern:

- For file-based operation mode:
 - The `-i` argument must be a Flex configuration file.
 - The `-r/-s` argument must be a profile archive file.
 - The `-b` argument must also be a file.
- For path-based operation mode:
 - The `-i`, `-r/-s`, and `-b` arguments must all refer to directories.

Overriding Group Policy Settings

Command-line arguments for FlexEngine take precedence over settings configured through Group Policy. For instance, if you set the log level to ERROR through policy, but specify `-l DEBUG` on the command-line, the latter is used.

You can also reset an argument that is configured through policy, in effect making it unconfigured, by appending a hyphen (-) to the command-line argument. For instance, if you have configured a backup path through policy, but want to run an export without creating backups, you can specify `-b-`.

You can reset the following arguments: `-S`, `-R`, `-b`, `-B`, `-Bd`, `-C`, `-c`, `-F`, `-M`, `-L`, `-H`, `-v`, `-f`, `-l`, `-rw`, `-rk`, `-rm`, and `-ra`.

Applying Horizon Smart Policies in Multiple Sessions

When using a Horizon Remote Desktop with Horizon Remote Applications, the same user can have multiple sessions on the same computer.

To correctly apply Horizon Smart Policies in such a situation, configure logon and logoff scripts with an extra command-line argument.

- For the logon script, specify the following arguments: `-HorizonMultiSession -r`
- For the logoff script, specify the following arguments: `-HorizonMultiSession -s`

Note `-HorizonMultiSession` must be the first argument.

Additional FlexEngine Operations

In addition to `-r` and `-s`, FlexEngine supports three other modes of operation.

-OfflineImport

When a computer is offline and a user logs in with cached credentials, Group Policy client-side extensions do not run. If FlexEngine is configured to run as a Group Policy extension, no import takes place in such a situation.

As a fallback approach (where a user logs on when the computer is offline), configure a logon script but instead of the `-r` script parameter, specify `-OfflineImport`. For information about configuring a logon script see [Configure FlexEngine to Run from a Logon Script](#).

If FlexEngine runs with this argument, it checks whether an import already took place. If so, it exits quietly. If not, a path-based import is performed by using the configuration from the User Environment Manager Group Policy.

-DirectFlexRefresh

DirectFlex configuration is processed during logon. If you add Flex configuration files with DirectFlex enabled, or modify DirectFlex-related settings of existing files while a user is logged on, these changes are not automatically picked up during the session.

This might not be a problem, but you can force an update by running the following command in the user's session:

```
"C:\Program Files\Immidio\Flex Profiles\FlexEngine.exe" -DirectFlexRefresh
```

-UemRefresh

User environment settings are applied at logon. For certain types of settings, you can perform a refresh while the user is logged in.

Table 7-1. Refresh Operations

Operation	Description
<code>-UemRefresh</code>	Refresh the User Environment Manager file type associations, shortcuts, and printer mappings.
<code>-UemRefreshFtas</code>	Refresh the User Environment Manager file type associations.
<code>-UemRefreshShortcuts</code>	Refresh the User Environment Manager shortcuts.
<code>-UemRefreshPrinters</code>	Refresh the User Environment Manager printer mappings.
<code>-UemRefreshADMX</code>	Refresh the User Environment Manager ADMX-based settings.
<code>-UemRefreshDrives</code>	Refresh the User Environment Manager drive mappings.
<code>-UemRefreshEnvVars</code>	Refresh the User Environment Manager environment variables.
<code>-UemRefreshApplicationBlocking</code>	Refresh the User Environment Manager application blocking settings.
<code>-UemRefreshHorizonPolicy</code>	Refresh the User Environment Manager Horizon Smart Policies.
	Note You can refresh Horizon Smart Policies at any time. However, the Horizon remote desktop experience components determine if the changes actually take effect at that time.

Table 7-1. Refresh Operations (continued)

Operation	Description
-UemRefreshTriggeredTasks	Refresh the User Environment Manager triggered tasks.
-UemRefreshPrivilegeElevation	Refresh the User Environment Manager privilege elevation settings.