FSLogix Apps for AppDisk Reference Architecture

Building on Citrix AppDisks
to enhance manageability
and reduce IT overhead

Version 1.1
October 2015
RA-1236

TABLE OF CONTENTS

1 Executive Summary 5
2 Introduction 6
2.1 Audience 6
2.2 Purpose 6
3 FSLogix Apps Overview 7
3.1 What is FSLogix Apps? 7
3.2 How Does FSLogix Apps Work? 7
3.3 Terminology 8
4 Citrix AppDisks Overview 10
4.1 What is Citrix AppDisk? 10
4.2 How Does Citrix AppDisk Work? 10
5 Benefits of Using FSLogix Apps with AppDisks 11
5.1 Seamless Applications Revisited 11
5.2 Finer-grained Control 11
5.3 License Compliance and Cost Optimization 12
5.4 Benefits Summary 12
6 Reference Architecture 13
6.1 Reference Architecture Overview 13
6.2 Reference Architecture Components 13
7 Setup and Configuration 15
7.1 Strategy 15
7.2 Preparing Base Images 15
7.3 Preparing AppDisks 15
7.4 Configuring Rules 15
7.5 Considerations for Different Use Cases 20
8 Summary 21
9 Appendix – Additional FSLogix Apps Features 22
9.1 Profile Containers 22
9.2 Java Redirection 22

1 Executive Summary

Optimizing the way applications are delivered and managed has been an ongoing challenge in enterprise IT, and the variety of approaches over the years have been received with varying degrees of success. While every method has pros and cons, some of the most stubborn issues include time-to-deliver, application conflicts, plug-ins and licensing. The too-frequent result is high IT overhead, too many gold images and excess spending on application licenses.

Too many companies are settling on less than optimal solutions because that is all they are presented with and they are unaware of the true state of the art for application management. It is possible to dramatically reduce the number of Windows gold images, in some cases to a single image. It is possible to deliver exactly what each and every end user needs to do their job, and nothing else. It is possible for a user to log into a random system and instantly be presented with their personal desktop, all the right applications, plugins and add-ons, the right printers, and the right fonts. It is possible and it is simple.

Citrix AppDisk, when integrated with FSLogix Apps, provides a unique set of features that can improve end-user productivity, reduce IT overhead and lower the cost of desktop management. With FSLogix patent pending Image Masking technology, AppDisk gains very granular and powerful user-based policy control over every aspect of a user's desktop and applications, and enhances the ability to distribute applications on network attached disk images. FSLogix Apps also enable AppDisks to scale to a much greater degree than any other similar technologies available today.

This document details what is possible when FSLogix is combined with Citrix AppDisk and suggests best practices for installing and configuring the products for maximum flexibility and applicability in various environments.

2 Introduction

2.1 Audience

This reference architecture document is provided by FSLogix and is intended for architecting, configuring and supporting FSLogix Apps as installed in conjunction with Citrix AppDisk. Consumers of this document should be familiar with Citrix AppDisk, Citrix XenApp, Citrix XenDesktop, and FSLogix Apps.

We have organized this document to address key items for each role that focuses on enabling a successful design, implementation, and transition to operation.

2.2 Purpose

This document covers the following subject areas:

- Overview of the FSLogix Apps solution
- Overview of the Citrix AppDisks solution
- The benefits of FSLogix Apps on Citrix AppDisk
- Setting up and configuring FSLogix Apps and Citrix AppDisk
- Configuration considerations for various use cases

3 FSLogix Apps Overview

3.1 What is FSLogix Apps?

FSLogix Apps is a computer software product for Windows Operating Systems that gives an administrator the ability to control the visibility of groups of files, directories, and registry objects. Typically, these groups of objects represent a Windows application, Internet browser plugins, Microsoft Office add-ins, and so on.

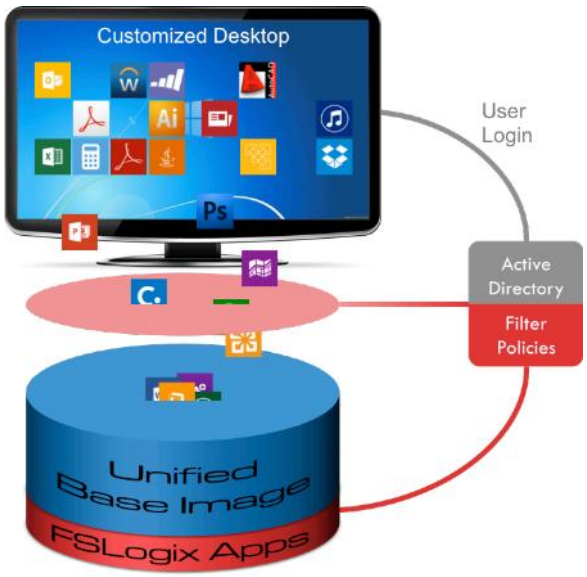
Additionally, the administrator can specify that a particular user or group can see the application when other users or groups cannot. For example, a Citrix server could have MS Office 2010 and MS Office 2013 installed. Using FSLogix Apps, the administrator could tie the visibility of these apps to membership in particular Active Directory groups, for example "Office2010" and "Office2013" respectively. By simply adding and removing users from these groups, these applications are visible or hidden for each user.

FSLogix Apps does not require any change to the way software is deployed or packaged. FSLogix Apps works with previously installed applications, existing Application Management systems (such as MS SCCM, Altiris, and so on), Application Virtualization products, and of course Citrix AppDisks.

3.2 How Does FSLogix Apps Work?

FSLogix Apps is a simple solution compared to the many application virtualization solutions that have come before. Applications are installed normally, instead of a complex process of packaging, layering and/or streaming, so there are never any issues about whether an application is compatible with the solution or will behave the way the vendor designed it to behave.

A small agent is installed on each Windows machine and a set of policies determines what will be hidden or redirected, based upon the user currently logged in. These policies are tied to Active Directory and may be applied to groups or individuals, as necessary. This allows a wide variety of users to operate on the same Windows image, and still have a completely custom experience. This applies to any Windows image.



Most FSLogix customers create very few gold images, each loaded with a large number of applications to satisfy the greatest number of users. Once a user logs in, only the applications authorized for their use appear. All other applications are completely hidden, with no trace, as if they were never installed at all.

This same method works across all application and desktop elements, including plugins, add-ins, printers, fonts and enything else that may be customized from one user to the next.

3.3 Terminology

This section introduces some terms that are used when describing the components and functions of FSLogix Apps:

Agent

The FSLogix Apps agent is a small, lightweight piece of software that is required to be installed on all the systems where the visibility of applications needs to be controlled. In the case of AppDisks, this only needs to be in the base image.

Assignment File

By default, Rule Files apply to all users. When finer control is required an Assignment File is used. An Assignment File contains one or more entries that specify a user or group (local or Active Directory) and whether the Rule File should apply or not. Assignment Files have an extension of fxa and should be deployed with the corresponding Rule File in order to be effective.

Auto-attach VHD

A VHD or VHDX (virtual hard disk) file that contains a set of files and/or folders. A rule created by the admin defines the parent folder for these files and folders. When the parent folder is accessed, the VHD is automatically attached and the contents seamlessly appear as children in the parent folder.

Rule

There are two types of rules: Hiding Rules and Redirection Rules. A Hiding Rule simply specifies a file, directory, registry key, or registry value that should be hidden. A Redirection Rule has a Source, Destination, and a property called "Should Copy". A Redirection Rule's source and destination also represent a file, directory, registry key, or registry value. When a Redirection Rule is active, it will cause all requests targeted at the Source object to be redirected to the Destination object. If the object is a file or registry value and Should Copy is true and if the Destination object does not exist, the Source object will be copied to the destination before the redirection happens. If the object is a directory or registry key and Should Copy is true, if the Destination object does not exist, it will be created before redirection happens. Subfolders, files, subkeys, or values are NOT copied, only the top level object is created.

Rule Editor

The FSLogix Apps Rule Editor is a GUI utility that is used to create and edit Rule Files, and also to manage the user/group assignments for Rule Files.

Rules Folder

The Rules folder is typically located at C:\Program Files\FSLogix\Apps\Rules. To apply Rule Files on a system, the Rule Files (along with corresponding Assignment Files, if any) are simply copied to this folder.

Rule Set / Rule File

A Rule Set is a collection of Rules. A Rule Set is also sometimes referred to as a Rule File because each Rule Set is contained in its own unique file. By convention, Rule Sets typically contain all the Rules necessary to control the visibility of an application, but this is not a requirement. A Rule Set may only contain a single Rule. A Rule File will have a file extension of fxr. For example myapp.fxr.

Profile Container

A VHD or VHDX (virtual hard disk) file that is attached to the system as the user is logging on. This contains all of the files and folders that are usually under C:\Users\<username>. See the Appendix for more on Profile Containers.

4 Citrix AppDisk Overview

Since Citrix is the proper source for complete information about AppDisk, please refer to Citrix sources for complete and current documentation. Here, we will provide a sufficient summary of AppDisk to understand the benefits and to support implementation and configuration of a combined solution.

4.1 What is Citrix AppDisk?

Managing applications while also managing the images they are installed on can be a challenge for many administrators of Virtual Desktop Infrastructure (VDI) environments. The AppDisk feature from Citrix is a solution to this problem.

AppDisks separate applications and groups of applications from the master image's OS, and let you manage these two independently. You can now manage the applications as a distinct entity – an AppDisk.

You can create different AppDisks containing the applications designed for individual user groups, and then assemble the AppDisks on a base image of your choice. By grouping and managing applications this way, you have finer control of your applications, and the number of master images you have to maintain is reduced. This simplifies IT administration and lets you be more responsive to users' needs.

Users are unaware of the separation of applications and OS (or any other aspect of the AppDisks feature). To them, applications appear to be, and behave as if they are, installed on their virtual desktop.

4.2 How Does Citrix AppDisk Work?

First, an IT administrator must create and roll out AppDisk. In brief, this is done by performing the following steps:

1. Install the Virtual Delivery Agent (VDA) on a virtual machine.
2. Create a disk on the machine. This becomes the AppDisk.
3. Install applications on the disk and *seal* them. This allows the AppLibrary service to record all of the installed applications and supporting files.
4. Deliver applications on the AppDisk to users through Delivery Groups.

When virtual machines start up from a Delivery Group, the XenApp and XenDesktop product coordinates with the AppLibrary, Machine Creation Services (MCS), and the Controller to distribute AppDisks.

A new AppDisk driver is used to configure AppDisks for use.

After setup, manage AppDisks through the **Delivery Groups** node in Studio. For compatibility analysis, use the **Configuration** > **AppDNA** node.

5 Benefits of Using FSLogix Apps with AppDisks

5.1 Seamless Applications Revisited

A very effective approach to simplifying desktop and application management was pioneered by Citrix with the concept of the “seamless application”. This breakthrough allowed a simple base image to be used for a wide variety of users, while specialized applications could be installed elsewhere and seamlessly introduced into the user’s environment, as if they were normally installed there.

A simplified base image that contains only the core applications required by the majority of employees is simpler to create, maintain and distribute. Core applications would typically include Microsoft Office, Adobe Reader and perhaps a few other apps common to the particular business. This base image may be located on a user’s individual PC or laptop, or it may be a virtual image in a VDI or RDSH environment.

Citrix AppDisks provides a mechanism to deliver the additional applications required by users to do their jobs. A common implementation would be to set up an AppDisk or a set of AppDisks for each department or specialty function. An AppDisk may be set up for Marketing that includes Adobe Acrobat, Photoshop, Camtasia and other marketing specific applications, while an engineering AppDisk may include AutoCAD and other engineering-specific applications.

5.2 Finer-grained Control

FSLogix adds an additional layer of control to the AppDisk paradigm. An AppDisk may contain a number of applications, and for each AppDisk attached to a base image, all applications on that AppDisk become available to all of those users. Since this is not always desirable, a typical recommendation is to create separate AppDisks for each department or distinct function. In order to more precisely allocate applications to specific users, IT managers may find that they must create more AppDisks, each with fewer applications. Limited licenses may be better allocated and preserved for the proper individuals by creating individual AppDisks for each application or user need. However, the iSCSI drivers used to attach AppDisks allow only a limited number of connections, and may become a severe limitation to scaling in more complex organizations.

FSLogix Image Masking technology allows IT managers to install as many applications as will fit on a single AppDisk. FSLogix looks at the individual AD login to the base image and hides all applications a user is not allowed to use, exposing only those authorized and licensed to the individual. Without FSLogix, every user with the same attached AppDisk will see all installed applications, so they must each have licenses paid for. By exposing applications to only those with productive need of each application, license costs can be reduced without multiplying the number of AppDisks.

Combining FSLogix with AppDisks allows IT managers to create the fewest possible AppDisk variations. Theoretically, a large AppDisk image could contain every application a company uses, creating only a single AppDisk image to manage. This would be attached to the common base image and each user would only see what they were allowed to see. This scenario could dramatically simplify the deployment and management of AppDisks.

5.3 License Compliance and Cost Optimization

As applications are being deployed beyond local desktops, to new technologies like Virtual desktops, and Cloud Based desktops, IT administrators need a solution to provision applications across all their critical infrastructure while ensuring business and licensing compliance. Traditional PC-based products developed for provisioning applications in conventional, internal enterprise, client-server models introduce tremendous overhead to VDI and cloud computing.

Simply partitioning groups of applications by department is a start to this solution. However in many organizations, almost every individual may have specific needs. Some need Adobe Acrobat, while others just need the reader. By controlling visibility of every application at the user level, enterprises are able to be very precise about how many licenses are needed and how many are provisioned for use. FSLogix even controls the availability of middleware and licensed add-ins, like those for Microsoft Office. So while everybody may get access to Office, some may additionally get access to one add-in, while certain others may get access to a different add-in. AutoCAD is another good example, with many licensed modules, often used by only one person each

5.4 Benefits Summary

Adding FSLogix to an AppDisk installation addresses a number of specific use cases worth mentioning here that would not be possible with AppDisks alone:

1. Each application on an AppDisk will only be exposed to those users and groups authorized by policy, regardless of how many users have access to that AppDisk.
2. Multiple versions of the same application may be installed on the same AppDisk.
3. Multiple versions of Java may be installed on the same AppDisk and exposed only for specific applications and websites, while the latest Java version is used for everything else.
4. Applications with drivers, including those that activate on startup, may be installed in a common base image and hidden for most users, since AppDisks does not support this class of application.
5. Control granularity to the feature level for Office add-ins, browser plugins, printer visibility, fonts, etc.
6. Simple version control, including instant rollout and rollback for application updates.
7. Optimized license costs while maintaining compliance.

6 Reference Architecture

6.1 Reference Architecture Overview

System requirements for Citrix AppDisk, as of the writing of this document, are more restrictive than those for FSLogix Apps. Since these requirements are currently evolving rapidly, please refer to the latest system requirements documentation for AppDisks.

Functionally, there are two main components to an AppDisks environment:

1. Base Image – one or more desktop images with the most common applications installed.
2. AppDisk image – one or more AppDisks containing additional applications needed by users.

The basic idea for both Citrix AppDisk and FSLogix Apps is that each user will log into a base image that contains the most common applications used across the company. If the applications are chosen wisely, it may be possible to have a single base image for all users. Large, complex enterprises may find this difficult, but you may be confident that the number will be smaller than with previous solutions or no solution at all. Aside from application complexity, if there is a requirement for more than one version of Windows, there must be a separate image for each version of Windows.

Once a user is logged into their base image, the AppDisk configuration will attach one or more AppDisks to that image, allowing additional applications to be seamlessly accessed, and the user will not be able to tell that they are not loaded locally. It may be a common strategy to have a separate AppDisk image fore each department or major function within the company. And each AppDisk image may be duplicated for load balancing. With the addition of FSLogix Apps, the number of different AppDisk images may be reduced.

6.2 Reference Architecture Components

Citrix AppDisks Components:

- XenApp and XenDesktop
- XenServer
- vSphere
- AppLibrary
- Machine Creation Services (MCS)
- Controller – to distribute AppDisks
- Studio > Delivery Groups node – to manage AppDisks
- Studio > AppDNA – for compatibility analysis
- See Citrix documentation for additional requirements for Studio and AppDNA

FSLogix Apps Components:

- FSLogix Apps
  - Driver

    The core of the FSLogix Apps software agent is the driver (frxdrv.sys). This component is a file system mini-filter driver. It is primarily responsible for intercepting requests from other software on the system, to access objects such as files or registry keys, and changing them.

    When an application is being hidden, the changes might involve simply making the object appear to not exist, e.g., an attempt to open a file will fail with a "file not found" error code. Or in a more complex case, it may involve changing the data results of a system call, e.g., an attempt to enumerate all the files in a directory may have one of those files removed from the resulting data.

  - Service

    The second major component in the FSLogix Apps software agent is the service (frxsvc.exe). This component is a Windows service running as system. It is responsible for communicating various data about the system state to the driver, e.g., informing the driver when new users login to the machine. It is also responsible for taking rule files and rule assignment files and "compiling" them to a format that is consumable by the driver. One additional responsibility of the service is that it provides an API for other programs to do things like add rules dynamically, query the rules in place on the system, etc.

  - User Interfaces

    The other components of the software agent are various user interfaces and supporting files. There is a GUI rule editor that assists administrators in creating rule sets, hiding applications installed on a system, making rule assignments to control how rules are applied, testing rules, etc. There is also a command-line interface that can be used for scripting many of the functions that the software can perform. A Windows Event Viewer integration module is also provided to assist administrators in the management and audit of the software.

7 Setup and Configuration

7.1 Strategy

FSLogix may always be added on top of any existing AppDisk installation without causing any issues, and you will get the full benefit of both products. However, there are a number of advantages to coordinating the installations if that is an option. As discussed above, it may be advantageous to install a great number of applications on a single AppDisk. FSLogix Apps allows applications to be installed on a single image that would ordinarily not be allowed due to installation or operational conflicts.

Installers may block installation of a new (or older) version if a current version is already installed. Some applications may conflict because they share or overwrite common resources, causing unpredictable results during runtime. For each of these cases, having FSLogix Apps installed prior to installing the applications will make this possible by allowing you to hide previously installed applications before installing the conflicting ones.

7.2 Preparing Base Images

Build the base image(s) according to your company's best practices for OS and desktop configuration. Ensure that FSLogix is installed in the base image. Do not install FSLogix into an AppDisk. Since FSLogix can manage any application in the base image on a per-user basis, this control will automatically be extended to any attached AppDisks. If an application is not appropriate for use in an AppDisk (such as applications that require boot time drivers, services, COM+ applications), the best practice is to install that application in the base image and allow FSLogix to control its visibility.

7.3 Preparing AppDisk

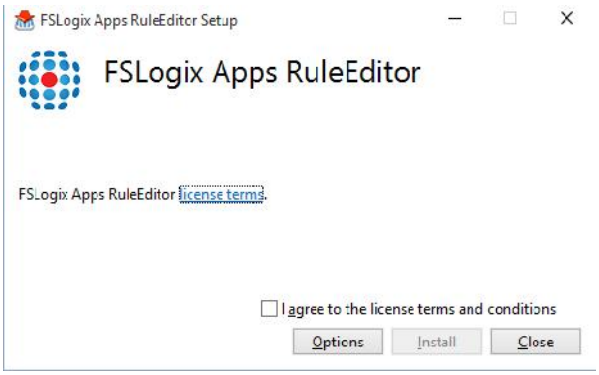
Prepare AppDisks according to Citrix provided best practices, using AppDNA as your guide. There are no fundamental changes needed in order to support FSLogix Apps.

7.4 Configuring Rules

Rule Editor Installation

The rule editor is installed on the admin computer where you want to create rules. It is not installed on client computers.

The Rule Editor is installed by running the Setup-RuleEditor.exe program. There is a 32-bit version and a 64-bit version. Choose the correct version for the platform where you are installing.

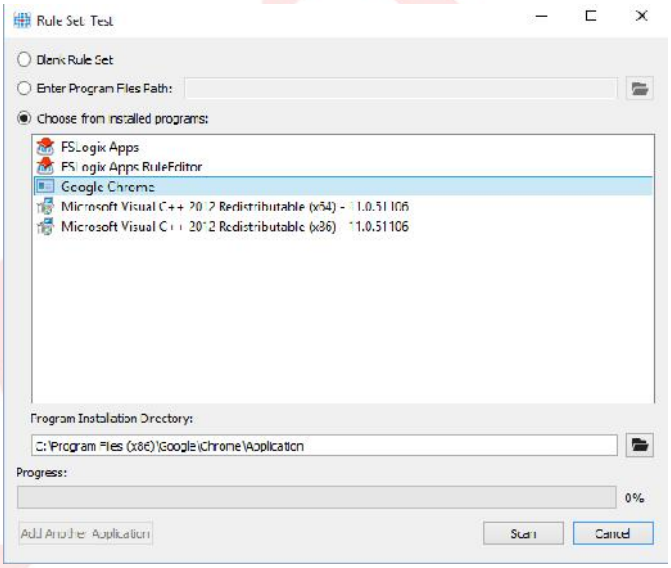


By default, the Agent is installed at C:\Program Files\FSLogix\Apps. To change the location where the Agent files are installed, click **Options**.

The FSLogix Apps Rule Editor installer can be run in unattended-mode or silent-mode for automated installations.

Create a New Rule Set

A Rule set determines which installed program is affected by a rule. To create a new Rule Set, click **File** > **New**. Provide a file name and a directory where the rule file should be created. Select the application that you want to manage using the rule. You can select from the following application types:



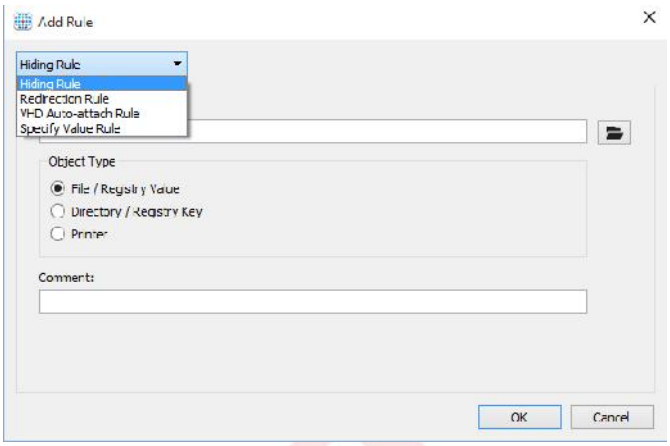
- Registered Application: If the selected program has registered an installation directory, it is listed automatically in the Program Installation Directory field. Click the application name and then click **Scan**. When the scan is completed, click on the **OK**.
- Unregistered Application: If an installation directory is not specified by the application, browse to the installation folder by clicking the folder icon. Program installation directories are typically under C:\Program Files or C:\Program Files (x86), though they can be at any location on the file system. After you browse to the installation directory, click **Scan**. When the scan is completed, click on the **OK**.
- Blank Rule File: Create a rule that is not tied to a specific application.

Add a Rule

When a Rule Set is open and selected, a new Rule can be added. Open a Rule Set and click **Edit** > **New Rule**. Select the type of Rule to create:

- Hiding Rule: Specifies a file, directory, font, registry key, or registry value to hide. Fonts are hidden by hiding the associated font files. Be sure to hide each file that is part of the font family (bold, italic, and so on). For example, the Corbel font consists of four files: corbel.ttf, corbelb.ttf, corbeli.ttf, and corbelz.ttf. Also note that Windows Explorer does not allow you to browse the font directory, so you must enter the path to the font file manually. Make sure to enter the full path of the file, for example: C:\Windows\Fonts\corbel.ttf.

- Redirect Rule: Causes all requests targeted at the Source to be redirected to the Destination.
- Specify Value Rule: Specify the data to be returned for a specific registry value.



Fill in the appropriate fields by entering the text directly or by browsing using the folder icons. If the object is a directory or a registry key, check **Directory/Key** for Object Type. Click **OK**.

To edit an existing rule, click the rule and then click **Edit** > **Edit Rule**.

To remove a rule, click **Edit** > **Delete Rule**.

For Redirection Rules, files and directories can be redirected to resources located on a network. The user must have appropriate rights to the network resource. To redirect to a network location, enter the path (in UNC format) into the Destination field.

Java Rule Editor

The Java Rule Editor is installed on the admin computer where you want to create rules. It is not installed on client computers. Run Setup-JavaRuleEditor.exe. There is a 32-bit version and a 64-bit version. Choose the correct version for the platform where you are installing.

You can create rules to configure an application or website to use a specific version of Java. Java Rules are added using the Java Rule Editor and then deployed using the same process as other rules.

1. In the Java Rule Editor, click **Edit > Add**.
2. Select the type of Rule to create:
   - Application rule: Select Application as the type and provide the path to the executable.
   - URL rule: Select URL as the type and provide the web page URL. There are three locations in the URL that can accept a wildcard: protocol, subdomain, and path.
     - protocol: `*://test.fslogix.com/folder/`- matches http or https.
     - subdomain: `https://*.fslogix.com/folder/` - matches test.fxlogix.com, www.fxlogix.com, or simply fslogix.com.
     - path: `https://test.fslogix.com/folder/*` - Any URL under the specified path matches. You can also specify a wildcard to match an entire domain, `https://test.fslogix.com/*`

     For example, `*://*.fslogix.com/*` matches http or https, any subdomain, and any path on the domain.

     Note: URL Rules require that the Java application runs in Internet Explorer 7 or later on the client computer.

3. Select the version of Java to use. The Java Version list is pre-populated with the Java versions on the admin computer where the Java Rule Editor is installed. You can also type in the required version using the following format:

   `1.MajorVersion#.MinorVersion#_update#`

   For example:

   `1.6.0_45`

   Note: The selected Java version must be installed on the client computer or the rule will not work. Multiple versions of Java can be installed side-by-side. Major Java versions will not conflict with each other, as each installs to a unique directory. However, Java versions that are the same major version but different minor versions install into the same directory by default. You can simply change the path during the installation to avoid this problem.

4. Save the Java Project file.
5. Click **File > Generate** to generate the Java Rule Files.

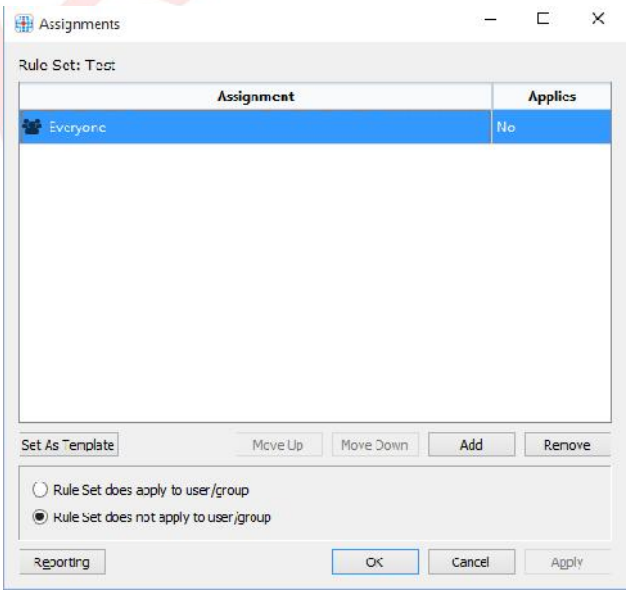
   For each Java version specified, there will be two files generated, one Rule Set file (.fxr) and one Assignment file (.fxa). Additionally, if any URL rules were specified, an XML file is also created.

6. Deploy the generated files. (See Deploying Rule Sets and Assignment Files below).

Managing User and Group Assignments

When no assignments are specified, a Rule Set applies to all users when it is active.

To manage the assignments for a Rule Set, select the Rule Set on the left and then select **File** >**Manage Assignments**. The following dialog appears:

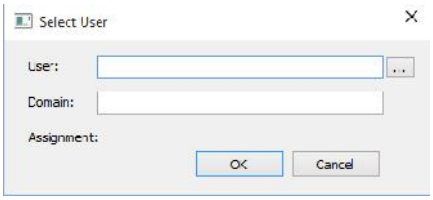


The order in which assignments are listed is important. See Assignment Order below.

Add Users or Groups

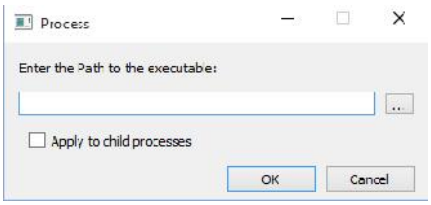
You can specify users and groups to which a rule does or does not apply.

Click **Add** > **User** or **Add** > **Group**. The following dialog appears:



Here you can enter the user (or group) name.

Add Process

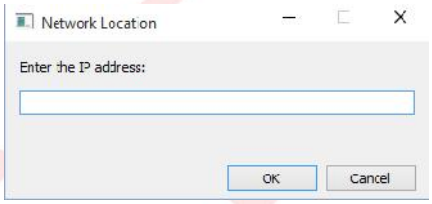


You can specify a process to which a Rule Set does or does not apply.

Click **Add** > **Process**. The following dialog appears:

Enter the full path to the executable. Select whether or not this assignment applies to child processes.

Add Network Location

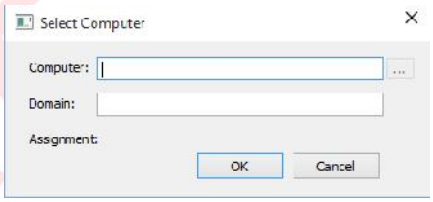


You can specify a network address.

Click **Add** > **Network Location**. The following dialog appears:

Enter the IPv4 or IPv6 address. Partial strings are allowed. For example, if you enter "192.168", an address of "192.168.0.1" will be considered to match.

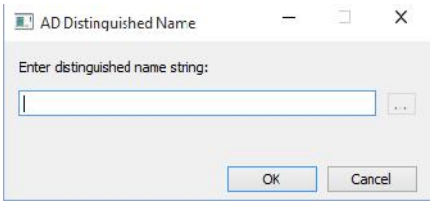
Add Computer



You can specify a computer object. Click **Add** > **Computer**. The following dialog appears:

Enter the Full Distinguished Name of the computer object, or the computer name (wildcards accepted).

Add Active Directory Container



You can specify an Active Directory Container and the assignment will be effective for all of the objects in that container.

Click **Add** > **Directory Container**. The following dialog appears:

Enter the Full Distinguished Name of the container.

Set As Template

After you have configured the assignment list, you can select **Set As Template** and all new assignment lists will default to the current state of the assignment list.

Assignment Order

When adding user and group assignments, pay close attention to the order in which assignments are listed.

Assignments are processed from top to bottom. For example, if two assignments were listed, the first specifying "Everyone" with "Rule Set does apply" and the second one specifying "User1" with "Rule Set does not apply", then the Rule Set would apply to Everyone except User1. If these two specifications were simply reversed with "User1" with "Rule Set does not apply" being first, then the Rule Set would apply to Everyone.

It is important to note that a user's group membership list is calculated at log on and will remain the same while that user is logged on. This means that changing what is visible to a user by manipulating their group membership requires the user to log off and then back on again in order for the changes to be seen. This is not the case for changes in the Rule Files. Those changes are immediately active upon distribution of the Rule File.

Deploying Rule Sets and Assignment Files

Deploying rule files and their associated assignment files is as simple as copying those files into the Rules folder on the managed computer.

By default the rules folder is located at C:\Program Files\FSLogix\Apps\Rules.

Any method can be used to copy the files into this folder: manual copy, file distribution systems, ftp, and so on.

After the files are copied into the Rules folder, the service detects a file change and then "compiles" the Rule File into a format that is easily consumable by the driver. The service then puts the compiled version into the CompiledRules folder where it is used by the driver.

When existing rule files are updated or deleted, the service updates or deletes the corresponding file in CompiledRules, and then notifies the driver.

Deploying Java Rules

To deploy Java Rule files, copy the Rule Set files (.fxr), Assignment files (.fxa) and XML files that were generated by the Java Rule Editor to the Rules folder.

7.5 Considerations for Different Use Cases

When using AppDisks without FSLogix, considerable time and effort must go into the planning of where applications reside and how they are distributed across multiple AppDisks.

For example, if AppDNA indicates that all of a department's applications can reside on an AppDisk, further consideration must be given to whether all users in the department have license to use the application. With FSLogix those considerations go away because any application may be hidden from those not authorized to use it.

FSLogix enables AppDisk to contain multiple versions of the same application: e.g., Office 2013 and Office 2016 in the same AppDisk. Or Adobe Acrobat and Adobe Reader in the same AppDisk. In this scenario, policy can dictate that certain users will get Office 2013 and Adobe Acrobat while other users get 2016 and Reader and so on.

Base your AppDisks on pool usage: for example, all applications that have vGPU requirements or all applications that require two CPUs and so on. This is a good way to optimize resources, ensuring that high-powered machines will be available and utilized for high-power applications.

A note on Scalability:
iSCSI has 64 slots per hypervisor. Most Hypervisors require at least 4 for their own use. That leaves 60 for the VMs. Assuming 4 VMs per hypervisor, that leaves 15 AppDisks per VM. FSLogix alleviates this problem by enabling each AppDisk to contain multiple applications.

8 Summary

Citrix AppDisk plus FSLogix offers enterprises a unique and powerful set of features not found anywhere else today. Building on the original Citrix concept of Seamless Applications, AppDisk and FSLogix Apps provides ultimate flexibility to create and arrange base images and attached AppDisks for optimal performance and manageability.

IT managers are always looking for ways to make their job easier while delivering a better experience to their users, and application management consumes enormous amounts of time and effort. Application compatibility testing, silos, service requests to install or roll back applications, and unmanageable license costs are among the most stubborn issues over the last twenty years. And it has only gotten more difficult as users become more mobile.

The union of AppDisk and FSLogix Apps is a simple and powerful solution that delivers a resounding answer to all of these issues.

9 Appendix – Additional FSLogix Apps Features

9.1 Profile Containers

The Profile Containers feature was introduced with version 2.0 of FSLogix Aps in early 2015, along with other powerful features like Application Containers and advanced Java version control. 2.0 features are not required when working with AppDisks, yet customers may find significant value if they choose to use them. All features are included in the product with no additional cost or licensing.

Windows Profiles are a required part of every Windows desktop, whether physical or virtual. They include everything from a user's vacation photos and desktop personalization, to critical application configuration information. Without real-time access to profile data, users can have a terrible and sometimes unusable experience with their corporate workspace. Depending on the user type, profiles can range in size from 50MB to 4GB and larger.

As anyone who has ever administrated Windows desktops knows, user profiles have been a major stumbling block to implementing virtual desktops whether Citrix XenApp or VDI. Instead of placing all of the user's files on a network share like in the redirected files approach, FSLogix encapsulates the entire profile (including the registry) in an in-guest container. This has several benefits:

Network Traffic Generated at Login

RDSH Server / Profile Server	FSLogix Profiles	Roaming w/ Redirection	Roaming Profiles
Client -> Server Packets	**32**	3,165	412,361
Server -> Client Packets	**80**	9,569	2,532,731
Client -> Server Bytes	**4KB**	463KB	34,475KB
Server -> Client Bytes	**118KB**	11,808KB	3,769,980KB

1. Eliminates the chance of a file server bottleneck by avoiding full file copies and opens.
2. Eliminates the need for "Just-in-time" file copying by only streaming the required blocks.
3. Large files such as Outlook .OST files no longer cause issues because all file operations are handled at a lower level, bypassing gating operations.

Slow logon and application launch times are one of the top complaints in virtual desktops. Legacy products and folder redirection add unnecessary network traffic and server utilization. Additional management and configuration systems add complexity, but don't address the root cause.

FSLogix advanced filtering approach removes the maximum amount of resource utilization from processing user profile data and eliminates the need for legacy profile products and folder redirection. User profile performance is near-local, yet administrators receive all of the benefits of centralized profiles with little or no ongoing administration.

9.2 Java Redirection

A common requirement in Enterprise environments is the need to support multiple Java versions. This generally occurs when IT needs to upgrade Java for security and interoperability reasons, but has applications or websites that require an older version of Java to function properly. These applications and websites may not be easy or cost effective to update, or may be managed by third parties outside the control of the IT department.

There are three general approaches IT uses today to provide enterprise support for multiple Java versions, all with various challenges.

1. VDI or RDSH environments may be used to group applications requiring a common Java version. This can cause compatibility problems across applications that may still need to interact.
2. Application Virtualization allows different Java versions to be packaged as part of a particular application. Packaging multiple applications can be incredibly complex, with a reported success rate as low as 60%. This also creates a large number of custom packages for IT to create and support.
3. Using the oldest compatible version of Java for everything creates compatibility issues for applications needing newer versions and exposes the entire environment to security risks.

Java redirection is a core feature of FSLogix Apps that allows all applications and websites to run with the version of Java they require to properly operate – all within the same client system. After administrative setup is complete, end users automatically use the appropriate Java version without any intervention or notification. Java redirection allows all required Java versions to be installed in the client environment, but only executed or visible on an as-needed basis, to maintain compatibility for specific applications and websites.

FSLogix Apps uses advanced filtering and a patent pending type of virtualization called image masking to enable Java redirection and other core functionality. This process is transparent to the client operating system and integrates seamlessly with all Windows based infrastructures.

In many instances, Java redirection will improve on your current security by permitting older versions of Java to work only with specific applications and websites approved by IT and INFOSEC. All other applications and websites will default to the most current version of Java deployed to your clients.

Java redirection is supported on Windows Vista SP1 and later, Windows Server 2008 and later, and works with Internet Explorer 8 and later. Java Redirection also works with all existing app virtualization solutions. Since it works at a higher level than application virtualization, it works with virtualized and normal applications equally.