

App-V 5.0 and Citrix

Integration Overview

This whitepaper is designed to provide administrators with guidance for combining App-V 5.0 and Citrix solutions. This document discusses the benefits of App-V 5.0 and Citrix, recommendations for Citrix images, App-V cache management, App-V management with Citrix, factors that impact user experience and administrative effort, and the Citrix XenApp Connector for Configuration Manger 2012 SP1. This whitepaper should not be used to determine the compatibility of specific versions of the products mentioned inside this document. Please consult the product-specific documentation for more information.



Contents

Overview	4
Overview of Citrix and App-V	5
App-V 5 and Citrix together	5
Unified delivery of App-V applications	6
Packaging App-V Applications	6
Delivery of App-V Applications	6
Extending the Reach of App-V applications	6
Extending App-V to non-Managed Devices	6
Benefits of App-V and Citrix	7
Additional Resources	7
Building Images with App-V	8
App-V options for image creation	8
Citrix options for image creation	8
Imaging benefits with App-V	9
Reduce Storage Requirements	9
Reduced Number of Images	9
Faster Logon Times	9
Configuration of the App-V Client	10
Enabling Package Scripts	10
Pooled VDI Image Management	11
Pooled VDI Image with Pvd Management	12
Dedicated VDI Image Management	13

Shared Desktop RDS Image Management.....	14
Shared Desktop RDS Image Management.....	15
Image management with Citrix.....	16
Publishing App-V applications in Citrix XenApp 6.5	16
Publishing App-V applications in Xendesktop 7.....	16
Citrix Receiver Local Application Preference	17
App-V Cache Management	18
Local Cache	18
App-V Shared Content Store.....	19
Local cache VS Shared Content Store Mode	20
Managing App-V applications in Citrix Images	21
Scenario 1: Single RDS (Shared Desktop) Image for all Farms.....	21
Scenario 2: Single VDI image.....	21
Scenario 3: RDS (Application Publishing) Image	22
Factors that impact User experience and administrative effort	23
Scenario 1: Maximizing User experience	23
Scenario 2: Least Administrative Effort.....	23
Scenario 3: Hybrid User-Experience and Administrative effort.....	23
Citrix RDS.....	24
Citrix VDI	25
Citrix XenApp Connector for Configuration Manager 2012 SP1	26
App-V Integration with Configuration Manager 2012 SP1.....	26
Conclusion	27
Appendix a: Output App-V Executable Paths with Powershell.....	28

OVERVIEW

Citrix® XenApp™ and Microsoft® App-V™ have long complemented traditional approaches to application management, but often as discrete systems instead of an integrated solution. Often organizations have completely separate departments managing applications and the Citrix environment. This separation causes duplication in packaging, deployment and management tasks that serve the same goal of making applications available to users. This inefficiency also prohibits a user-centric model, which allows users to roam from device to device and still maintain access to the applications and data required for productivity. Combining App-V and Citrix products enables unified application lifecycle management for Windows® and non-Windows devices from anywhere.

This whitepaper will discuss the scenarios, benefits, and features available with combined Citrix and App-V solutions. The following discussion points will provide explanation on building a solution that fits organization needs while maintaining high levels of service and controlling costs.

- Overview of Citrix and App-V
- App-V and Citrix together
- Building Images with App-V
- Local versus Shared Content Store Mode
- Managing App-V applications in Citrix Images
- Factors that affect user experience and administrative effort
- Configuration Manager 2012 SP1 Connector for Citrix XenApp 6.5

Today, Citrix supports App-V 5 as a primary method of both application delivery and publication. With XenDesktop 7, Citrix has built-in native support for App-V 5. This document will focus on the individual and combined capabilities of App-V 5 with Citrix products. At the time of this writing, Citrix XenApp 6.5 and XenDesktop 7 are available in market and will be the focus of the documentation. Since these products do differ, the recommended solutions provide generic guidance with links to specific steps or documentation.

OVERVIEW OF CITRIX AND APP-V

Citrix XenApp and XenDesktop provide integration with Microsoft App-V, delivering App-V virtualized applications to any user on any device anywhere. Through this integration, organizations can combine the benefits of Citrix and App-V infrastructures to reach a broader set of devices including lightly managed and non-Windows endpoints across local or wide-area networks. This solution lowers the cost of delivering and maintaining Windows applications for all users in the enterprise.

From the enterprise perspective, App-V can greatly reduce the effort of creating and deploying Citrix solutions. App-V simplifies RDS and VDI solutions by eliminating application installations and with, Shared Content Store mode, reduces expensive storage. App-V also eliminates application conflicts with isolation, reducing testing and decreasing the number of servers required in RDS solutions. App-V and Citrix provide benefits when working together.

APP-V 5 AND CITRIX TOGETHER

This document presents different benefits and configurations for co-implementation of App-V and Citrix technologies. This section will describe some key benefits that address customer needs and impact the IT department, including:

- Creation of App-V packages, necessary for all App-V deployments
- Deploying virtual applications to the Citrix environment, enabling non-Windows based devices access to Windows based applications
- Rich roaming capabilities of App-V applications across any device in any location

This document assumes that administrators of the XenApp environment are familiar with, or have already deployed, a Citrix Web Interface to enable access to applications from web connected devices, Citrix Access Gateway™ to provide application-level policy and action controls, and Citrix NetScaler® to optimize the delivery of applications over the Internet and private networks. The administrator should also be familiar with any other Citrix or third-party supporting technologies for management of a XenApp environment. This section will focus on the pieces of technology that pertain to App-V with Citrix solutions, including:

- Unified delivery of App-V applications
- Extending the reach of App-V applications to non-managed and non-Windows devices

The benefits of App-V and Citrix together offer a solution to a common need for organizations. These benefits will lay the groundwork for common goals, with the remainder of the document explaining considerations and configurations for executing App-V and Citrix solutions together. The desired solution for a specific organization may be one or a combination of the scenarios with the details for implementation presented in this document.

UNIFIED DELIVERY OF APP-V APPLICATIONS

The promise of unified delivery of applications across all devices and to any location is a priority as user-centric computing has moved into the mainstream. The benefits of App-V include accelerated deployment of applications and reduced management of the application lifecycle. Extending these benefits from the traditional desktop to Citrix solutions enables the unified delivery of applications.

PACKAGING APP-V APPLICATIONS

Unified delivery of applications starts with packaging applications. For traditional MSI based delivery, applications must be modified or built to support both the desktop (traditional and VDI) and Citrix RDS based implementation. Selecting the App-V format reduces packaging effort, as most applications will work across delivery targets. Thus, packaging applications into the App-V format is the first step toward enabling unified application delivery. More information on packaging App-V applications is available in the **App-V 5 Sequencing Guide** at:

<http://www.microsoft.com/en-us/download/details.aspx?id=27760>

DELIVERY OF APP-V APPLICATIONS

After packaging applications into the App-V format, the administrator must decide how to deliver and manage the App-V applications for traditional desktops as well as XenApp servers. The App-V Client for Remote Desktop Services is required for the XenApp servers; however, there are several options for managing the client and for delivering App-V applications. Delivery of App-V applications to a Citrix environment can be achieved using the following methods:

- Configuration Manager 2012 with the Citrix Connector for Configuration Manager 2012
- App-V Management and Publishing server
- Scripted PowerShell
- App-V MSI based delivery

Depending on the organization infrastructure, one or more of the deployment methods may be required to complete a solution.

EXTENDING THE REACH OF APP-V APPLICATIONS

The next step in an App-V and Citrix solution is extending the reach of App-V applications beyond managed computers to non-managed and non-Windows based devices. App-V provides an efficient delivery mechanism for applications and Citrix enables access to applications from anywhere and from any device capable of running the Citrix Receiver client.

EXTENDING APP-V TO NON-MANAGED DEVICES

Extending App-V to non-managed and non-Windows devices requires the use of the Citrix Receiver client in conjunction with the App-V and Citrix XenDesktop infrastructure. App-V applications published to Citrix RDS environments are available to users working on devices that have the Citrix Receiver Client. Citrix VDI may deliver the App-V applications to the desktops either directly with a local App-V client or via Citrix Receiver, when applications have been published to the Citrix RDS infrastructure. This document will describe delivery of the App-V applications into both Citrix RDS and VDI-based solutions.

BENEFITS OF APP-V AND CITRIX

The combination of App-V and Citrix benefits both the user and the administrator. The combination provides the user with self-service access and a rich user-experience, while enabling ease of administration and controlling costs. The following table is a list of some of the benefits of the technologies when utilized together.

XenDesktop Features	App-V Features
<ul style="list-style-type: none">• Self-service delivery• Access from any device, anywhere• High-definition user experience with HDX• Secure architecture, secure delivery, secure by design• Single instance server and application management• Enterprise-class application management• Integrated management of App-V applications• Integration with System Center	<ul style="list-style-type: none">• Isolation: enable multiple version coexistence, increase application compatibility with XenApp execution environment, server silo consolidation• Reduce end-user impact; on-demand, accelerate deploying, upgrading, patching, and removal of applications, and eliminate reboots• Reduce Storage with Shared Content Store (SCS) mode• Integration with System Center

ADDITIONAL RESOURCES

This whitepaper assumes the reader has a working knowledge of the technologies listed below; however, detailed knowledge of each required technology may be spread throughout the organization. This document provides links throughout for reference and additional detail.

In order to proceed with the technical information contained within this document, the reader should be familiar with each of the technologies listed below. These links provide a base level of knowledge for each of the technologies.

- [App-V](#)
- [System Center Configuration Manager 2012 SP1](#)
- [Citrix XenApp](#)
- [Citrix XenDesktop](#)
- [Citrix Receiver](#)

Read the [Integrating App-V with Microsoft VDI](#) whitepaper for an overview and benefits of Shared Desktop Computing and VDI Infrastructure scenarios. This document will be referenced throughout this whitepaper.

BUILDING IMAGES WITH APP-V

App-V improves Citrix image management by decreasing the number of images required, reducing image size, and enabling faster logon times. Citrix solutions for VDIs, or where Provision Service or Machine Creation Service are implemented for server-based streaming, however, require significant image management. App-V 5.0 Shared Content Store mode changes the default behavior of the App-V Client when enabled virtual applications are not cached locally. Applications instead run from the network on launch and only cache in memory. This change in behavior enables administrators to publish applications to VDIs or RDS servers with minimal disk impact.

Building an image management solution with Citrix and App-V can be accomplished in several different ways depending on the desired outcome.

APP-V OPTIONS FOR IMAGE CREATION

- **Pre-publish App-V applications:** Application assets are not cached on the specific endpoint, but the publishing information (FTAs and shortcuts) are already present in the image.
- **Pre-Publish and Pre-cache App-V applications:** Application assets are stored locally in the App-V package store.
- **Shared Content Store Mode:** The App-V Client configuration disables streaming of package assets to the package store by default. Administrators can change on a per package basis.

CITRIX OPTIONS FOR IMAGE CREATION

- **Provisioning Service:** PVS enables streaming delivery of RDS and VDI-based images that are IOPS optimized with a separate component for large multi-site locations.
- **Machine Creation Service:** MCS enables streaming delivery of RDS (XenDesktop 7) and VDI-based images that are built in to the Citrix Studio for single-site locations.
- **Personal vDisks:** PvDs enable VDI single image management of pooled and streamed desktops while allowing users to install applications and change their desktop settings.

For details on specific App-V configurations combined with RDS and VDI solutions, review the Integrating App-V 5 with Microsoft VDI whitepaper at: http://download.microsoft.com/download/B/B/7/BB782D16-67A9-4E54-8BB3-28A898DFFBA6/Integrating_App-V_with_Microsoft_VDI.pdf. Review information on Personal vDisk at: <http://support.citrix.com/proddocs/topic/xendesktop-ibi/cds-about-personal-vdisks-ibi.html>

IMAGING BENEFITS WITH APP-V

Combining App-V with Citrix imaging solutions provides three primary benefits that improve both RDS and VDI based solutions and image management.

REDUCE STORAGE REQUIREMENTS

App-V reduces the hardware requirements for both RDS and VDI solutions with the Shared Content Store feature. This feature eliminates duplicating applications installed on each server or desktop operating system. Each application can be placed on the datacenter network once and shared between and amongst servers that are included in an application publishing farm, or for VDIs, whether pooled or dedicated. The actual binaries of the application are remote from the image, which reduces the image size and storage requirements, decreases delivery time, and creates images that can be used across different scenarios. This feature decreases the amount of physical storage required in an expensive Storage Area Network (SAN) solution.

REDUCED NUMBER OF IMAGES

App-V Shared Content Store enables image creation with only the App-V client and publishing server information. App-V applications are not installed and are only published for RDS solutions, and for VDI, the publishing process is only completed once a user logs in. This reduces the number of required images, as a single image creation can be repurposed for multiple scenarios. For RDS-based solutions, a single image captured with publishing information can be used across the entire farm without the user logon process, making user logons faster.

RDS-based deployments should pre-publish applications and configure the App-V Client with a publishing server. All applications are entitled to Citrix servers or all users, including administrators. This configuration presents the shortcuts and registers file type associates (FTAs) to the server OS. For administrators, the App-V applications published on the servers can be published to end-users.

For VDI solutions configure the App-V Client with a publishing server, capturing the image prior to any publishing information. This enables the image to be re-purposed for multiple scenarios, where the App-V applications are published and presented at logon for users. For pooled scenarios, this process is repeated every time the user logs in unless a Personal vDisk has been configured to capture the changes during an session. Dedicated VDIs retain the publishing information across logons.

FASTER LOGON TIMES

With Citrix and App-V solutions for VDIs, App-V applications can be pre-published or even pre-cached, enabling users to logon and only have an update to the current publishing information. When present, PvDs will save state information across logons for pooled desktops. For RDS-based solutions, pre-publication and pre-caching improve logon performance and eliminate a full publishing refresh and streaming until the image is updated.

CONFIGURATION OF THE APP-V CLIENT

During or post-installation, there are several configurations to support Citrix for the App-V client when configured in an image. The following tables describes the recommended default settings and configurations for Citrix Images across the following scenarios:

- Pooled VDI Images
- Pooled VDI Images with PvDs
- Dedicated VDI Images
- RDS (shared desktop) images
- RDS (application publishing) images

By default, all scenarios should enables package scripts, as this option is required for deploying Microsoft Office and will be commonly used with other sequenced packages created by an organization. Enabling package scripts is performed with PowerShell. The following command should be run to configure this App-V feature as part of the installation and before capturing and image:

ENABLING PACKAGE SCRIPTS

Perform the following PowerShell commands.

```
#Import App-V Client Module
```

```
Import-Module AppvClient
```

```
#Enable Package Scripts
```

```
Set-AppvClientConfiguration -EnablePackageScripts 1
```

POOLED VDI IMAGE MANAGEMENT

Managing pooled VDI images with App-V is performed as part of image management. Administrators should evaluate how often applications update and incorporate that process with the maintenance schedule for the operating system. Administrators should publish applications in pooled VDI scenarios to the entire machine since any user could connect to any of the active sessions. Using Shared Content Store is recommended to reduce the size of the image, however individual applications can be pre-cached to improve launch performance. Pre-publishing applications is optional. Creating the image without pre-publishing applications enables the image to be used across different scenarios, however the logon time will be increased as the publishing process must be performed at each logon of the pooled VDI desktop.

Table 1: Pooled VDI Image Configuration

	PowerShell or MSI	App-V Full Infrastructure	Configuration Manager*
Shared Content Store	X	X	X
Pre-publishing	X	X	X
Pre-Cache	Optional	Optional	Optional
Publishing Server		X	
Enable Package Scripts	X	X	X
Enable Global Refresh		X	
Static Image	X	X	X
Dynamic Image			

* Only XenApp 6.5 and XenDesktop 5.6 support Configuration Manager 2012 at the time this document was created.

The table above lists the recommended settings for configuring the App-V Client and treating the image. These are not absolute recommendations and should be evaluated based on specific application performance as well as combinations of settings.

POOLED VDI IMAGE WITH PVD MANAGEMENT

Managing pooled VDI images utilizing PvDs and App-V enables dynamic images. PvDs retain App-V applications and settings between sessions. Dynamic images allow updates to occur between scheduled image maintenance windows. New applications can be published and the new publishing information is stored in the PvDs. During image maintenance, the new and updated applications should be added to the gold image and the PvDs should be reset. Using Shared Content Store is recommended to reduce the size of the image, however individual applications could be pre-cached to improve launch performance. Pre-publishing applications is optional. Creating the image without pre-publishing applications enables the image to be used across different scenarios, however the logon time will be increased as the publishing process must be performed at each logon of the pooled VDI desktop.

Table 2: Pooled VDI Image with PvD Configuration

	PowerShell and MSI	App-V Full Infrastructure	Configuration Manager*
Shared Content Store	X	X	X
Pre-publishing	X	X	X
Pre-Cache	Optional	Optional	Optional
Publishing Server		X	
Enable Package Scripts	X	X	X
Enable Global Refresh		X	
Static Image			
Dynamic Image	X	X	X

* Only XenApp 6.5 and XenDesktop 5.6 support Configuration Manager 2012 at the time this document was created.

The table above lists the recommended settings for configuring the App-V Client and treating the image. These are not absolute recommendations and should be evaluated based on specific application performance as well as combinations of settings.

DEDICATED VDI IMAGE MANAGEMENT

Managing dedicated VDI images with App-V is similar to managing traditional desktops. Dedicated VDI retains publishing, caching, and settings between sessions. Using Shared Content Store is recommended to reduce the size of the image, however individual applications could be cached to improve launch performance. Pre-publishing applications is optional. Creating the image without pre-publishing applications enables the image to be used across different scenarios, however the logon time will be increased as the publishing process must be performed at each logon of the dedicated VDI desktop.

Table 3: Dedicated VDI Image

	PowerShell and MSI	App-V Full Infrastructure	Configuration Manager*
Shared Content Store	X	X	X
Pre-publishing	Optional	Optional	Optional
Pre-Cache	Optional	Optional	Optional
Publishing Server		X	
Enable Package Scripts	X	X	X
Enable Global Refresh			
Static Image			
Dynamic Image	X	X	X

*Configuration Manager 2012 is only supported with XenApp 6.5 and XenDesktop 5.6 at the time this document was created.

The table above lists the recommended settings for configuring the App-V Client and treating the image. These are not absolute recommendations and should be evaluated based on specific application performance as well as combinations of settings.

SHARED DESKTOP RDS IMAGE MANAGEMENT

Managing Shared Desktop RDS images with App-V is similar to pooled VDI as multiple users will utilize the same configuration. Building images that support an individual farm should include pre-published applications, which will reduce logon times. Images that are built to support multiple farms will not include pre-published applications, unless they are common across all farms. Using Shared Content Store is recommended to reduce the size of the image, however individual applications could be cached to improve launch performance. Creating the image without pre-publishing applications enables the image to be used across different scenarios, however the logon time will be increased as the publishing process must be performed at each logon of the RDS session.

Table 4: Shared Desktop RDS Image

	PowerShell and MSI	App-V Full Infrastructure	Configuration Manager*
Shared Content Store	X	X	X
Pre-publishing	Optional	Optional	Optional
Pre-Cache	Optional	Optional	Optional
Publishing Server		X	
Enable Package Scripts	X	X	X
Enable Global Refresh			
Static Image			
Dynamic Image	X	X	X

* Only XenApp 6.5 and XenDesktop 5.6 support Configuration Manager 2012 at the time this document was created.

The table above lists the recommended settings for configuring the App-V Client and treating the image. These are not absolute recommendations and should be evaluated based on specific application performance as well as combinations of settings.

SHARED DESKTOP RDS IMAGE MANAGEMENT

Managing Application Publishing RDS images with App-V is similar to Shared Desktop, with the exception that all applications must first be published (App-V) and potentially pre-cached in the RDS image to support publishing from Citrix. Whether targeted for single or multiple farms, all applications are pre-published (App-V) and then only the specific applications are published to entitled users. Shared Content Store is recommended to reduce the size of the image; however, individual applications could be cached to improve launch performance.

Table 5: Application Publishing RDS Image

	PowerShell and MSI	App-V Full Infrastructure	Configuration Manager*
Shared Content Store	X	X	X
Pre-publishing	X		X
Pre-Cache			
Publishing Server		X	
Enable Package Scripts	X	X	X
Enable Global Refresh		X	
Static Image	X		
Dynamic Image		X	X

* Only XenApp 6.5 and XenDesktop 5.6 support Configuration Manager 2012 at the time this document was created.

The table above lists the recommended settings for configuring the App-V Client and treating the image. These are not absolute recommendations and should be evaluated based on specific application performance as well as combinations of settings.

For instructions on how to deploy the App-V Client manually or with Configuration Manager, review the Integrating App-V with Microsoft VDI whitepaper at: http://download.microsoft.com/download/B/B/7/BB782D16-67A9-4E54-8BB3-28A898DFFBA6/Integrating_App-V_with_Microsoft_VDI.pdf

IMAGE MANAGEMENT WITH CITRIX

Citrix provides image management with Provisioning Services (PVS) or Machine Creation Services (MCS). For more information on specific steps for these technologies visit: <http://support.citrix.com/proddocs/topic/provisioning-7/pvs-provisioning-7.html> and <http://support.citrix.com/proddocs/topic/xendesktop-7/cds-catalogs-landing-page.html>.

PUBLISHING APP-V APPLICATIONS IN CITRIX XENAPP 6.5

The focus of this document is on Citrix support for App-V 5.0, which provides a fully integrated solution together. The primary benefit of App-V 5 applications in Citrix XenApp 6.5 solutions is the ability to directly publish the EXE for the application which is available directly in the package store. Previous versions of App-V with Citrix publishing required pointing to the launching program for App-V (SFTTray.exe) and passing command line parameters (/launch "{App-V application name}). The publishing process with App-V 5 is detailed below.

After successful deployment of the App-V package, the next step is to publish the application to servers in the farm and to specific users utilizing the Citrix XenApp AppCenter. The following steps are necessary to publish App-V applications to users.

1. In the Citrix AppCenter, expand **Citrix Resources | XenApp |{Farm Name}**
2. Right-click on **Applications** and choose **Publish application**.
3. Advance through the wizard by clicking on **Next**.
4. Name the application with a name familiar to users, such as "Adobe® Acrobat Reader X."
5. On the Type page:
 - a. Select **Application**.
 - i. For application type select: **Accessed from a server**.
Note: This will only enable session-based execution of the App-V application.
6. On the **Location** page:
 - a. Find the executable in the Package Store, by default c:\programdata\App-V.
Note: The subdirectories are GUIDs that are based on the package GUID and version GUID, which can be located with PowerShell (get-AppvClientPackage).
 - b. Paste the target information, to the Command line: field.
"C:\ProgramData\App-V\5D8ECAC1-1E7E-47F2-B863-05AF9C2EA5A2\5138CEA9-8731-4F60-9455-16E09945B8F6\Root\Reader\AcroRd32.exe"
7. On the **Servers** page, **Add** appropriate servers via direct server membership, worker groups, or farm.
8. On the **Users** page, select the appropriate users who can access the application.
9. On the **Shortcut presentation** page verify the correct icon has been presented and click **Next**.
10. Finish the publishing wizard based on your corporate standards.

Note: [Appendix A](#) provides a PowerShell script that will detect all Globally published App-V applications on an App-V Client and output a text file with the package name, application name, and path to the executable. Utilize this PowerShell script in step 6a to avoid the process of finding each application executable.

PUBLISHING APP-V APPLICATIONS IN XENDESKTOP 7

XenDesktop 7 supports publishing Microsoft App-V 5 applications directly to delivery groups. This enables usage of the XenDesktop Studio to configure App-V Publishing directly from the Management and Publishing server(s). The integration enumerates the published applications from the publishing server along with master image

applications during the creation of desktops or applications delivery groups. For detailed information on configuring App-V publishing with XenDesktop 7, please review: <http://blogs.citrix.com/2013/06/28/microsoft-app-v-support-in-xendesktop-7/>

In previous versions of App-V Citrix provided the App-V Conduit that enabled scenarios where an App-V 4.6 Client can be detected on the local machine (endpoint), and based on availability of the App-V Client either stream the application directly to the endpoint, or in the absence of the App-V Client, deliver the application to the RDS server present the application to the user as a session. The App-V Conduit is not required with XenDesktop 7 and App-V 5 applications, as the combination of integrated support of App-V 5 in Delivery Groups and the new Receiver client replace the functionality. App-V 5 applications can be sourced directly from App-V Publishing servers and the App-V Client presence on the endpoint enables local delivery for offline use.

CITRIX RECEIVER LOCAL APPLICATION PREFERENCE

Citrix Receiver 3.4 supports local application detection, where preference is placed on local applications. When a user accesses applications from a StoreFront store, and based on configuration preferences, detect local versions of the application on the endpoint for launch. This allows users to launch applications presented in the store on the local machine, utilizing the locally installed copy of the software, instead of delivery or presentation from Receiver. For more information on configuration review: <http://support.citrix.com/proddocs/topic/receiver-windows-34/receiver-windows-config-xa.html>

APP-V CACHE MANAGEMENT

Choosing between the default local caching of application assets or the App-V feature, Shared Content Store mode, depends on specific conditions and desired results. Since Citrix solutions are not simple point solutions, but multiproduct and integrated solutions, organizations typically combine local and shared content store mode configurations based on different scenarios.

LOCAL CACHE

By default, App-V stores application assets locally at `c:\programdata\App-V`. After users launch applications, their assets stream to the App-V client for storage on the local disk. This benefits a typical desktop client as it provides offline support, enabling users to launch applications even when not connected to the company network. Since Citrix solutions require connection to a server for VDI or RDS, offline use is not needed.

For both RDS and VDI Citrix solutions, the use of local cache provides the best performance as application assets load directly from disk. Large-scale solutions may find the use of local cache limiting, as each application must be duplicated across each RDS server or VDI desktop. In large RDS server farms or VDI implementations in the hundreds or thousands, common applications such as Microsoft Office®, which is approximately 2 GB in size, will consume large amounts of disk space.

App-V Client

The App-V Client can be installed with or without specifying a publishing server. Since the Pvs or MCS manages the image, all updates to App-V applications will be performed during image maintenance. There are different methods to updating vDisks depending on the version of PVS or MCS: Utilize the typical method for updating the image and update the App-V applications and resetting the Pvs. This can be accomplished during the read/write mode of the endpoint image with a publishing refresh or utilizing the manual process of PowerShell.

Publish and pre-cache App-V applications

Creating vDisk or updates will require updating the App-V Client for new and updated applications. If the App-V client is not configured with a publishing server, the administrator must manually update the image. App-V provides a PowerShell module for the client. The following PowerShell command publishes and pre-caches any applications making the applications available for any users of the RDS or VDI solution.

```
Import-Module AppVClient  
Add-AppvClientPackage -path {path to location of .appv file} | Mount-  
AppVClientPackage | Publish-AppVClientPackage -global
```

Note: This is a basic PowerShell command to complete this tasks, for more automated tasks refer to PowerShell documentation that enable further automation of commands at: <http://technet.microsoft.com/en-us/scriptcenter/powershell.aspx>

APP-V SHARED CONTENT STORE

App-V 5.0 is designed to be easy and efficient to use in RDS and VDI, allowing IT to make the best use of expensive disk resources without changing the user experience. App-V 5.0 Shared Content Store turns off local virtual application storage, dramatically reducing disk requirements for RDS and VDI, while leaving the application provisioning and update process unchanged.

VDI solutions with a 500 MB application delivered across 1000 VDI desktops requires five hundred gigabytes of storage. Although storage solutions can provide benefits, App-V natively supports storing an application once on the datacenter network, while all of the VDI desktops launch the application from the shared location.

App-V Shared Content Store mode changes the default behavior of the App-V client so virtual applications are not cached on the local App-V Client. Rather, virtual applications stream into RAM on-demand at each launch of the application. Shared Content Store Mode provides flexibility where administrators can choose which applications to cache locally for highly utilized applications. Shared Content Store eliminates duplicating virtual applications across up to hundreds of RDS servers in a farm or hundreds to thousands of VDIs. Shared Content Store mode places a dependency on network bandwidth since applications download or stream to the Server (RDS-based) on the first launch after booting or, in the case of VDIs, for each instance initiated.

Configure Shared Content Store mode during setup or after installation by modifying a registry value or from the built-in App-V Client PowerShell module. The following options are available for configuring Shared Content Store mode after installation.

Registry

From the registry editor navigate to **HKLM\Software\Microsoft\AppV\Client\Streaming** and edit **SharedContentStoreMode** to a value of 1.

PowerShell

Open PowerShell as an administrator in the RDSH or VDI.

```
Import-Module AppvClient
Set-AppvClientConfiguration -SharedContentStoreMode 1
```

Note: Configuring Shared Content Store Mode after installation requires a restart of the App-V Client Service (Windows Desktop or RDSH). Review the TechNet documentation at: <http://technet.microsoft.com/en-us/library/jj713455.aspx> for configuring shared content store mode during installation.

The publishing process with Shared Content Store mode is the same as local cache with one difference. The App-V applications are added and published, but not mounted. In a typical App-V infrastructure, this is performed with a normal publishing refresh. If performing the update of applications from PowerShell, perform the following commands:

```
Import-Module AppvClient
Add-AppvClientPackage -path {path to location of .appv file} | Publish-AppvClientPackage -global
```

For RDS-based solutions, the application is streamed into memory on the first use and the Windows memory manager takes care of the in-memory cache of the application. Subsequent users will not have to stream the application again as long as the memory manager has not removed the virtual application from memory.

Shared Content Store mode is not an absolute setting; by default, it enables application publishing and launching without storing the assets locally for applications. However, administrators may choose to use the benefit of shared content store for most applications while selecting a few applications and storing the application assets in the local cache. For more information on scenarios supporting Shared Content Store review:

<http://blogs.technet.com/b/appv/archive/2013/07/22/shared-content-store-in-microsoft-app-v-5-0-behind-the-scenes.aspx>

LOCAL CACHE VS SHARED CONTENT STORE MODE

There is no absolute reason to use local cache versus shared content store mode. The basic difference is that local caching of applications stores App-V applications on each App-V Client. This storage could be a local hard disk in the example of an RDS solution, which is relatively cheap, or a SAN location that is more expensive in nature. With local storage, each RDS server or VDI desktop has to have an individual copy of all the applications targeted to the device or users of the device. Shared content store mode allows the RDS and VDI endpoints to share one copy of the applications across the entire network. This option greatly reduces the total storage across the enterprise and is the recommended option in most solutions.

Some specific considerations when choosing between local vs. shared content store mode.

Considerations

- Shared Content Store mode is recommended when using SAN storage for RDS and VDI based desktops to reduce the storage footprint. Since SAN-based storage is always across a network, there is no benefit to multiple copies unless they are zoned separately, and therefore Shared Content Store mode reduces the number or read IOPS to a specific zone.
- Direct attached storage solutions can be implemented either way, where local storage is typically cheaper than SAN and will provide the most consistent and, in most cases, the best performance. This scenario is only common for RDS environments, where there are several physical servers.
- Shared Content Store is recommended for use with Virtual Machine based RDS as it will reduce the size of the images.
- Shared Content Store is recommended when using Streamed Image Deployment to reduce the size of the images and decrease delivery time.

MANAGING APP-V APPLICATIONS IN CITRIX IMAGES

Managing App-V applications in Citrix image based solutions can be accomplished manually (scripted), with App-V infrastructure, or with Configuration Manager. Each of the different types of image (RDS or VDI), as well as optional configurations, have specific differences in managing applications. Since the process of publishing App-V applications is well known, this section will focus on some unique scenarios where the inclusion of App-V in Citrix images offers additional choices. This section will focus on common scenarios that have been deployed in the field and explain them in-depth in terms of App-V application lifecycle management. This set of scenarios represents a subset of possibilities when using App-V with Citrix based image management.

Note: Each scenario assumes that the primary goal of Shared Content Store mode is to reduce the size of images for both storage and streaming on the network.

SCENARIO 1: SINGLE RDS (SHARED DESKTOP) IMAGE FOR ALL FARMS

Organizations that would like to get to a single image across all Citrix RDS farms can achieve this goal with App-V and Citrix. The App-V Client for Remote Desktop Services is installed in Shared Content Store mode in the base image and configured for the appropriate delivery method. The App-V Full Infrastructure requires the configuration of a Publishing Server and Configuration Manager requires the Configuration Manager Client. The important part of this image creation is not capturing any publishing refresh information (Shortcuts, File Type Associations (FTAs) and Package Store population) prior to capturing the image. With this scenario, when users logon to the Citrix RDS based implementation the publishing refresh will occur. For manual (PowerShell or MSI) based delivery, an appropriate scripting method or Group Policy should be utilized to perform the publishing process.

Setting up a Citrix RDS (Shared Desktop) image with these configurations will enable administrators to utilize the image across all of their RDS (Shared Desktop) Farms. This also eliminates updating the base image for App-V specifics, unless there are specific App-V applications that have been cached locally to improve launch speed. Only App-V applications that are considered part of the base build should be pre-published and pre-cached in this scenario, and will require updates to pre-cached applications in the base image during normal operating system maintenance windows. The drawback to this scenario is that the first logon and any subsequent logons are dependent on how long the publishing refresh operation takes before all applications become available.

SCENARIO 2: SINGLE VDI IMAGE

Similar to the above, the goal of Scenario 2 is to have a single image that will work for different VDI based delivery groups. Since there are two types of VDI desktops, pooled and dedicated, there are some differences between the two. Pooled VDI desktops configured with the App-V Client, set for the specific delivery scenario, and enabled for Shared Content Store, provide administrators with the ability to have a single image. This configuration ensures that every logon by a user will perform a publishing refresh (scripted, App-V Full-Infrastructure, or Configuration Manager). As the amount of applications or the size and complexity of applications increases, the logon time for users will increase. Utilizing PvDs will improve login time as publishing information will be retained between sessions and only new applications will be published on subsequent logons.

For Dedicated VDI, the single VDI image has the initial expense of performing a publishing refresh. Unlike the Pooled VDI image, dedicated VDI desktops retain publishing information across sessions giving a traditional desktop App-V delivery behavior.

Neither type of VDI desktop requires specific image maintenance for App-V unless specific Line of Business or corporate standard software has been pre-published and pre-cached into the image.

SCENARIO 3: RDS (APPLICATION PUBLISHING) IMAGE

Citrix RDS adds an additional layer for application publishing. With XenApp6.5, the process of publishing the App-V applications from the console for XenApp 6.5 requires that all applications that might be entitled to users are delivered to the machine (global refresh) so they are available to all users, including administrators, for Citrix publishing. The App-V Client for Remote Desktop Services is configured with Shared Content Store mode and all App-V applications are pre-published to the machine prior to capturing the image. This scenario enables delivery via PVS or MCS with a fully configured App-V Client with Applications with only one step, configuration of the Citrix Publishing from the XenApp 6.5 console.

XenDesktop 7 has fully integrated support for App-V 5 allowing administrators to configure a publishing and streaming server from the Citrix Studio. This configuration enumerates the list of applications from the App-V Publishing server for direct publishing to application and desktop Delivery Groups. This eliminates publishing steps from the previous versions, but requires an App-V Full-Infrastructure.

For both versions of RDS (application publishing) image management, it is possible to publish all of the applications that impact the entire organization, since there is no impact on the disk with Shared Content Store mode. This will enable more of a single image across the organization. However, if App-V applications are pre-cached that are unique for a specific use case, it is better to create multiple images, since the size of the images will grow with updates and additional pre-cached applications. Updating the App-V applications is different depending on the version, since XenApp 6.5 requires performing a publishing refresh and Citrix publishing process, whereas the XenDesktop 7 environment only requires updating the Delivery Group with new App-V applications prior to capturing an image for streaming.

FACTORS THAT IMPACT USER EXPERIENCE AND ADMINISTRATIVE EFFORT

Centralized desktop and application management impacts the overall user experience. Since all of the desktops and applications reside in the data center in the form of VDI and RDS based desktops, the logon and application launch load is concentrated to a single or a few points. The typical RDS scenario will have many users logging in at the same time, and with VDI scenarios, several machines are started and logged into at the same time. All will require application launch immediately after the user logon. The combination of App-V and Citrix can be configured to improve the logon and launch performance, but will be a tradeoff of additional disk space, network traffic, and maintenance. The information provided in this section outlines the factors that impact performance, but will not provide absolute configurations or recommendations. Organizations should test any configuration and determine if the end results are acceptable prior moving into their production environment. Here is a list of factors that impact logon experience with Citrix RDS and VDI scenarios with App-V.

- Streamed Image: PVS and MCS based solutions
- Shared Content Store: Applications are not stored locally in the VDI or RDS local package store.
- Pre-publishing App-V: The process of performing a publishing refresh prior to capturing an image.
- Pre-caching App-V: Placing App-V applications in the local package store, either selectively or completely.
- Pre-loading App-V applications in memory: Providing an automated method for pre-loading the application in memory.

SCENARIO 1: MAXIMIZING USER EXPERIENCE

To optimize the configurations of Citrix and App-V to provide the best user experience, pre-publish and pre-cache all the App-V applications into the base image. This configuration will have the shortest logon and launch times as all the off-machine operations do not complete. This scenario works well for RDS and Pooled VDI where the list of applications published is static, with no add-and-remove and no updates. For this scenario to work, the image must update every time there is a new application added or updated. This scenario has the highest costs of administration.

SCENARIO 2: LEAST ADMINISTRATIVE EFFORT

Most organizations have limited IT Resources and look to minimize the impact of any new process on the administrators. The previous discussion of single image management provides a framework for achieving the solution with the least administrative effort. This scenario has a single image with the App-V Client, but no pre-published or pre-cached applications. Delivering the single image with Citrix PVS or MCS completes the solution with a high level of automation.

This least administrative effort solution provides the user with longer logon times and application launch times than other scenarios presented in this section.

SCENARIO 3: HYBRID USER-EXPERIENCE AND ADMINISTRATIVE EFFORT

This scenario is the middle-ground where typical organizations reside and requires testing across the stack of components to ensure the tradeoffs of user experience and administrative effort are acceptable. This scenario offers the best blend of user-experience with controlled administrative efforts and should be planned and tested prior to implementation, ensuring user satisfaction without uncontrolled costs of administrative efforts. The bulleted list from above describes factors that affect the user experience as a whole. This scenario will address

each factor and discuss its influence on the user experience. Thus, organizations can choose which factors to test and implement those that provide the greatest benefit.

CITRIX RDS

The creation of and management of Citrix RDS and VDI images, and especially their contents, has a large effect on the user experience and administrative efforts. This section discusses these topics, but an understanding of the impact of image streaming end-result improvement remains the focus here.

For example, an organization has created a single image for their entire RDS farm, which includes the App-V Client and Shared Content Store Mode, but nothing else. When this image is streamed to a target and starts up, it is lacking any publishing information and any local application assets. The image has been targeted with 20 App-V applications. When the RDS image first boots up and becomes available, the first user that logs on has to perform a publishing refresh and after that operation is complete, will need to start an application. Since the applications are stored on the network and not on the local disk, each application launch will stream the necessary assets into the RDS servers' memory. Subsequent users will gain an advantage as the initial publishing refresh has updated the server image with some of the applications that the specific user needs and some of the applications will be loaded into memory. Therefore, over time, this RDS server will have most applications published and the most commonly used applications loaded into memory.

This hypothetical organization has a normal maintenance policy to reboot their servers periodically. When this occurs, the process of publishing App-V applications and loading them into memory has to occur again. This will provide an inconsistent user experience, but represents a minimal effort from an administrative standpoint.

Changing the configuration of the streamed image can improve and stabilize the user experience. The list below represents some of the common ways to improve and stabilize the user experience without dramatically impacting administrative efforts:

- **Global Publishing:** Configure the App-V Client to perform Global Publishing Refresh and target common applications to the machine rather than to users. Common applications will then be refreshed on the reboot and not wait for a user logon. This can also be accomplished by creating a script (PowerShell, batch, scheduled task, GPO) that performs the operation to minimize the number of applications being published.
- **Pre-Publish:** Capturing the base image for the RDS solution with pre-published App-V applications will reduce logon time since only updates to the currently published applications are required. This requires updating the base image periodically with new and updated application publishing information.
- **Pre-Load Applications:** Create a script that pre-loads applications into memory and utilize (Run line, Schedule task, GPO) during startup so that commonly used applications will already be loaded into the Windows Memory Manager.
- **Shared Content Store and Pre-Caching:** This reduces the storage requirements, which improves the image delivery time, but places the limiting factor for performance once delivered on the network and remote storage. Selectively, pre-caching applications that perform poorly in Shared Content Store mode can dramatically increase the user experience, but requires the testing of applications, as there is not a specific formula for determining which ones work well in this configuration and which ones do not.

The user's experience (being able to perform required tasks) and the performance of the VDI-based solution (speed of the virtual desktop and applications) are the primary factors in user acceptance of the solution. As VDI solutions have much greater associated disk usage costs compared with RDS, Shared Content Store mode should be enabled unless an additional requirement makes it not possible. This configuration places the bulk of the performance or user experience factors on network and storage.

Pooled VDI is not persisted and has to be very generic, because of this requirement, impacting the user experience is difficult to do with specific settings and is more dependent on the underlying hardware.

Dedicated VDI is persisted, however since user information of applications, settings, and data need to be stored somewhere, the use of specific configurations can benefit a specific dedicated VDI solution.

Several of the factors can be evaluated to improve the user experience with minimal impact to administrative efforts:

- **Global Publishing: Pooled VDI:** Configure the App-V Client to perform Global Publishing Refresh and target common applications to the machine rather than to users. Common applications will then be refreshed on the reboot and not wait for a user logon. This can also be accomplished by creating a script (PowerShell, batch, scheduled task, GPO) that performs the operation to minimize the number of applications being published. **Dedicated VDI:** Only globally publish base applications that will be used by all users of the dedicated VDI virtual machines, target department, or user specific applications to the individual users as the desktop is persisted between sessions.
- **Pre-publishing:** : Capturing the base image for the RDS solution with pre-published App-V applications will reduce the logon time, as only updates to the currently published applications are required. This requires updating the base image periodically with new- and updated-application publishing information.
- **Shared Content Store and Pre-Caching:** This reduces storage requirements, which improves the image delivery time, but places the limiting factor for performance, once delivered, on the network and remote storage. Selectively pre-caching applications that perform poorly in Shared Content Store mode can dramatically increase the user experience, but requires the testing of applications, as there is not a specific formula for determining which ones work well in this configuration and which ones do not.

CITRIX XENAPP CONNECTOR FOR CONFIGURATION MANAGER 2012 SP1

The Citrix Connector for Configuration Manager is an add-on to Configuration Manager developed by Citrix, that provides integrated management of application deployment to Citrix XenApp servers and the publishing of XenApp 6.5 hosted applications from System Center Configuration Manager. The Connector enhances the built-in capabilities of Configuration Manager to perform the series of tasks required to deploy applications to XenApp Servers with the same ease and familiarity used for desktops.

The Citrix Connector enables administrators to create orchestrated tasks required to deliver applications (traditional and virtual) to XenApp 6.5 Servers and publish hosted applications with the familiar Configuration Manager console. Prior to the Citrix Connector for Configuration Manager, application deployment across a XenApp 6.5 Farm from Configuration Manager was not well-managed end-to-end. Installing applications on XenApp 6.5 servers requires draining user connections, installing applications, and potentially rebooting each server to complete the installation tasks. Configuration Manager could not tell how many users were connected to a server before installing an application, and could not force disconnection of users from XenApp 6.5 server. Therefore, the solution at the time was to deliver applications to XenApp servers in “In Maintenance Mode,” where user connections are disabled. Because this required communication and coordination between two separate administrative units, the technologies were managed separately.

For traditional applications, the Citrix Connector orchestrates the tasks (draining users, putting the server in “install mode,” rebooting the servers and returning them to service) necessary for a traditional installation. Since App-V applications are not installed and reboots are not necessary, the Citrix Connector provides the ability to publish App-V applications from the Configuration Manager Management Console. The Connector relies on Configuration Manager 2012 SP1’s App-V integration to deliver the App-V apps prior to publishing.

Complete details of the Citrix XenApp Connector for Configuration Manager 2012 SP1 are available in the following whitepapers: XenApp Connector Enterprise Setup: <https://citrix.sharefile.com/d/s4744d4a7b064566a> and XenApp Connector In-depth Technical Reference: <https://citrix.sharefile.com/d/sed0b509414240019>.

APP-V INTEGRATION WITH CONFIGURATION MANAGER 2012 SP1

Integrating Configuration Manager and App-V provides a comprehensive deployment and update service for virtual applications. Configuration Manager 2012 SP1 reduces the App-V footprint to the App-V Sequencer and Client. Configuration Manager takes the place of the publishing and streaming components in a typical App-V full infrastructure. This eliminates the need for two separate infrastructures to support application deployment. Configuration Manager can be used to deploy, maintain, inventory and meter both traditional and virtual applications.

Enabling delivery of App-V applications to XenApp begins with integrating App-V and Configuration Manager 2012 SP1. The details for planning and implementing App-V with Configuration Manager 2007 are available in the **Virtual Application Management with Microsoft Application Virtualization 5.0 and System Center Configuration Manager 2012 SP1** whitepaper available at: http://download.microsoft.com/download/C/0/1/C0199142-8A8F-42A3-9D3F-6E807CB248F9/Managing_AppV5_with_Configuration_Manager_2012SP1.pdf.

CONCLUSION

Citrix and App-V represent a fully integrated solution across RDS and VDI-based solutions. This document has presented the benefits of building solutions with a combination of App-V and Citrix. The goal was to provide information to enable an administrator to make decisions about the App-V configuration in Citrix based images, minimize the storage impacts, and improve the user experience without negatively affecting the administrative effort.

APPENDIX A: OUTPUT APP-V EXECUTABLE PATHS WITH POWERSHELL

As an example, this PowerShell command compiles a list of all applications published to a specific App-V Client for the purposes of publishing App-V applications in XenApp 6.5. This example may not work in every environment.

PowerShell Example for extracting executable paths

This PowerShell shows the published packages to the machine (global) and the path to the application executables. Some executables for paths will be tokenized, however most are ones that are embedded DLLs and EXEs that are not published in a Citrix XenApp 6.5 environment.

```
"PackageName,Application Name,ApplicationPath"|out-file .\AppPath.txt -Append
foreach($pkg in (gwmi -Namespace root\Appv -Class AppVClientPackage -Filter
"IsPublishedGlobally = true"))
{
    $apps=gwmi -Namespace root\Appv -Class AppVClientApplication -Filter
"PackageID = '$($pkg.PackageId)' and PackageVersionID = '$($pkg.VersionID)'"
    foreach($app in $apps)
    {
        $rootFolder= (Get-ItemProperty "HKLM:\Software\Microsoft\AppV\Client
\packages\"+($app.PackageId)\versions\"+($app.PackageVersionID)\Catalog").Folder
+"Root"
        $appPath=$app.TargetPath.Replace("[AppVPackageRoot]", $rootFolder)
        "$($pkg.Name), $($app.Name), $($appPath)"|out-file .\AppPath.txt -Append
    }
}
}
```