WHITE PAPER–JUNE 2020

Windows Application Delivery Design Considerations

vmware®

Table of contents

Introduction 3
Current Windows application platforms 3
Windows executables . . . 3
Scripted installs . . . 4
Windows installers . . . 4
Intranet apps . . . 4
SaaS apps . . . 5
Virtualized apps . . . 5
Application modernization design decisions 5
Business purpose of the application . . . 5
Internally developed apps and third-party apps . . . 6
New apps and existing apps . . . 6
Installation prerequisites and install order . . . 6
Application authentication and SSO . . . 6
App file format and platform support . . . 7
App lifecycle and updates . . . 7
Unique app requirements . . . 7
Windows application delivery decision workflow 8
Modern Windows application delivery end states 9
Strategy 1: Re-architect the application to be cloud native . . . 9
Strategy 2: Publish as a web-based application . . . 10
Strategy 3: Repackage to a modern native platform . . . 10
Strategy 4: Deliver through a virtualization solution . . . 11
Strategy 5: Migrate with no modifications . . . 11
Summary 12
Additional resources 12
Authors and contributors 12

Introduction

Windows 10 is a modern, cloud-first mobile operating system. It allows for enterprise-grade configurations, policies and applications to be deployed to Windows 10 devices on any network. IT can have full visibility into the device's health status and mandate a minimum-security requirement to enable access to line-of-business (LOB) applications.

VMware Workspace ONE® is a unified endpoint management solution that allows organizations to fully realize the benefits of the modern Windows 10 operating system. Workspace ONE allows IT professionals to create policies for their Windows fleet, provide end users with access to their applications, and maintain a secure operating environment by integrating into Windows patching and third-party security solutions.

Application delivery to the Windows device fleet is the most important factor to maximize end-user satisfaction. The inability for an end user to access the apps they need for their duties becomes a cause for frustration and friction with IT teams. There are many different types of apps and delivery models, meaning it is imperative that IT professionals choose the correct delivery model to optimize the user experience and future-proof their Workspace ONE deployment.

The purpose of this white paper is to provide IT professionals with a prescriptive guide for delivering applications to their Windows 10 fleet using the Workspace ONE platform. It details the challenges in current application frameworks and guides an IT professional to the recommended design approach to ensure their apps are best equipped to benefit from Workspace ONE delivery.

Current Windows application platforms

Windows applications are the backbone of all core processes for many organizations. They are required for most process-oriented tasks, and help organizations complete their business-to-business (B2B) and business-to-consumer (B2C) functions.

Moving to a modern platform such as Workspace ONE presents some significant changes to the current Windows application landscape. Many principles used in the past to develop and deploy Windows apps no longer apply. This is due to factors such as a re-architected operating system, the advancements in web and software-as-a-service (SaaS) technologies, and the modern cloud-based delivery methods of Workspace ONE.

The following are the six most common formats currently used for Windows applications. The purpose is to illustrate the advantages of these platforms and challenges when delivering these formats through a modern platform such as Workspace ONE.

Windows executables

Executables (EXEs) are a Windows installation file format and the most used platform for Windows programs. EXE-based apps are overwhelmingly used for internally developed and third-party-developed Windows apps.

Platform benefits

EXEs are very easy to develop and distribute, and can be installed using standard command line switches.

Considerations for Workspace ONE delivery

Installation command and success criteria are not defined in the install program, which requires the admin to determine and configure in the console. It also requires the admin to determine and specify the method to uninstall the application from an end user's machine.

Scripted installs

Windows executables are augmented by organizations to fit their appropriate needs, typically through a scripted installation or wrapper script. In most cases, the application's installation EXE is called by an associated script file that includes the installation parameters, the pre-installation checklists, installation success criteria and app-specific settings. Most organizations deliver EXE-based apps through a wrapper script because the current delivery mechanism does not provide these capabilities.

Platform benefits

Scripted installs are incredibly flexible, as an organization can augment an application to fit their specific needs. IT professionals can use standard PowerShell scripting or equivalent skills to generate scripted install packages.

Considerations for Workspace ONE delivery

Installation command and install success criteria are not defined in the configuration settings in Workspace ONE Unified Endpoint Management (UEM) in the program. Configuration settings are done within the wrapper that encapsulates the app, so there is limited visibility into the delivery of the application or individual components within the wrapper. There is also no change logging or auditing, and it usually requires uninstalling the entire package to deploy a new version of the app.

Windows installers

Windows installers (MSIs) include the installation information configuration, application files and dependency checking/initiation packaged into an installation package. MSIs allow for IT to centralize the components associated to a file install into a single entity, making it easy for repackaging when any changes are needed. MSIs also natively leverage configuration files (MSTs) to allow for separation of configuration and installer logic, enabling rapid application installation configuration without editing the installer itself.

Platform benefits

There is only a single package for installing an app. The uninstall command is standardized, and the install parameters are defined in the package.

Considerations for Workspace ONE delivery

There are minimal drawbacks. This is a preferred method for Workspace ONE because install commands, app configuration parameters, installation success criteria and uninstall commands are encapsulated in the MSI/MST, and are captured by Workspace ONE UEM.

Intranet apps

This is the most common type of web-based app used by organizations. Intranet apps provide an interface for end users to access organization-specific data and complete daily duties. These can be homegrown web apps or third-party software installed within an organization's network.

Platform benefits

Abstraction of the application from the native operating system enables delivery to all mobile operating system platforms and centralizes the lifecycle of the application.

Considerations for Workspace ONE delivery

Single sign-on (SSO) needs to be analyzed before delivering through Workspace ONE. Some intranet apps have browser dependencies that may dictate the Workspace ONE delivery method.

SaaS apps

SaaS apps are the fastest growing app platform, as organizations look to standardize on public tools to complete functions and move from an in-house development to a service consumption model. Core apps used across all organizations—such as productivity tools, payroll, HR and IT service desk—are rapidly being consumed on SaaS-based platforms.

Platform benefits

SaaS apps are built for cloud scalability and high availability, as well as for the latest web technologies. They are independent of the device operating system platform.

Considerations for Workspace ONE delivery

There are minimal drawbacks. SaaS apps are readily supported in the Workspace ONE platform and can be delivered through many different methods.

Virtualized apps

For the continued use of legacy Windows apps necessary for an organization's operations, virtualization through VMware Horizon® and Citrix Virtual Apps and Desktops can help provide access to these resources. Users can access a server-hosted instance as a Remote Desktop Services Hosted (RDSH) app, a full desktop instance (virtual desktop infrastructure [VDI]), or as a streamed application through platforms such as VMware ThinApp®.

Platform benefits

Allow Windows apps to be accessed with session-level security. Allow apps that have legacy server/browser requirements to still be available on Windows 10.

Considerations for Workspace ONE delivery

There are minimal drawbacks. Virtualized apps from Horizon, ThinApp, or Citrix Virtual Apps and Desktops can be delivered through the Workspace ONE platform. This method provides access to the app in its current form but does not address the challenges around modernizing the application for future use.

Application modernization design decisions

Workspace ONE provides an opportunity to rationalize and re-architect applications to gain the efficiencies of cloud deployment. There are eight important factors for IT professionals to consider when deciding the correct Workspace ONE delivery model to use. These are important design considerations to ensure the application lifecycle can be operationalized through the Workspace ONE platform.

Business purpose of the application

Most Windows applications used in an organization fit into two categories: apps required for user productivity, or apps that enhance the security posture of the device. The primary business purpose of apps for user productivity are to complete a business function. For these apps, it is important to analyze the user population to determine whether they should be automatically available to a user or are better delivered through a self-service mechanism.

Organizations may look to consolidate applications that provide the same business function. Commonly, consolidating security tools and functionality, reducing products/vendors, or using a function built into the operating system (e.g., migrate encryption tools from a third-party tool to the native Windows 10 BitLocker functionality) are key outcomes to many organizations.

Analyzing the role of the app and the user segments the app is deployed to are imperative to determining the correct delivery model in Workspace ONE.

Internally developed apps and third-party apps

Organizations use a mixture of homegrown-developed and public third-party applications to conduct their business operations. Third-party apps come from independent software vendors (ISVs), and are often packaged or augmented to fit the configuration requirements of the organization.

Homegrown apps are often written for a specific LOB purpose. Many of them are incredibly important to running critical systems in an organization. Unfortunately, many of these applications have limited documentation and are not architected for ever-changing operating systems and security updates.

Understanding how third-party apps are packaged and learning the architecture of homegrown apps can help with determining which is the correct delivery model in Workspace ONE.

New apps and existing apps

Organizations are often faced with the challenges of moving existing Windows apps from their current delivery model to Workspace ONE. The expectations are often set so that the same application fleet in the old system should also be accessible in the new system.

An analysis of existing applications to determine their usage and role in the organization is a key step in the movement to modern app delivery through Workspace ONE. Many existing apps may have been deployed but not actively used or not decommissioned correctly after their usage period. This is a great opportunity for IT to consolidate the app fleet and move only those necessary to the organization to Workspace ONE.

When new apps are onboarded into the organization, IT has the luxury of architecting the app to best fit the modern Workspace ONE platform. IT can define the optimal delivery model for the app based on the users, access scenarios and associated app data.

Installation prerequisites and install order

Windows LOB apps may have dependencies or prerequisite applications that typically require app components to be installed in the correct sequence, with dependencies and reboots being honored to guarantee success. This knowledge may be owned by a dedicated team within IT whose function is to package applications. This knowledge of the application installation and configuration logic is required for future deployment of the app.

Apps with complexities that make full lifecycle management difficult or rely on internal-only applications and services may be better suited to a remote application delivery model.

Application authentication and SSO

Authentication and SSO are some of the most important factors for application access. Because modern LOB apps come from a variety of sources (SaaS, public cloud, private cloud), authentication to these apps can occur through many different techniques.

The friction associated with multiple password prompts or discrete system-based passwords is a source of frustration for end users. It should be a core objective of IT to ensure that end users can access LOB apps with SSO. Apps onboarded into Workspace ONE need to be examined to best understand their authentication methods and assessed to ensure SSO in the modern Workspace ONE platform.

SSO methods—such as Security Assertion Markup Language (SAML), Open Authorization (OAuth) and certificates—are most preferred, but Integrated Windows Authentication (IWA) and Kerberos-based authentication can also be supported through the Workspace ONE platform.

App file format and platform support

For existing apps, it is important for IT to know whether the app is already in a modern format or if it may need development work to modernize it. Native Windows apps in the MSI format can be easily migrated into Workspace ONE. Design decisions need be made about native Windows apps that are not MSIs and the best way to deliver them through the modern app delivery framework.

It is also imperative to understand whether the app is only for Windows or whether there is an equivalent app for other mobile operating system platforms, such as iOS and Android. In most cases, third-party apps available across multiple platforms are updated regularly to ensure easy delivery for organizations across their employees. For example, some apps, such as the Microsoft Office suite and Adobe Reader, have well-documented enterprise delivery models that can be followed when delivering in Workspace ONE.

App lifecycle and updates

Because the Windows operating system changes frequently with feature and quality updates, Windows apps consistently need to be tested and updated accordingly, so there is no loss of functionality. In addition, new features are built into updated versions of apps, and these need to be deployed to end users in a timely manner.

The lifecycle of an app is a key consideration when determining the correct Workspace ONE delivery model. Some frequently updated apps may be better suited for a centralized app packaging and automated distribution approach, such as Flexera AdminStudio integration with Workspace ONE to distribute MSIs. In some cases, delivery of these apps can be achieved through a virtualization solution and made available to the end user through Workspace ONE Intelligent Hub.

Unique app requirements

While a smaller use case, some apps may have highly sensitive data that can only be accessed by the end user when connected to the company network. In such cases, a virtualization solution can help with delivering that app to a user on a Windows 10 device.

Virtualization solutions, such as remote desktops or app streaming, can help deliver those niche applications with unique requirements to users. However, it should not be the primary delivery method for Windows apps. Native app delivery is the recommended approach as it will ensure the best end-user experience.

Windows application delivery decision workflow

Figure 1 is a decision tree intended to help an organization select the appropriate delivery mechanism for an application through Workspace ONE for access on a Windows 10 device.

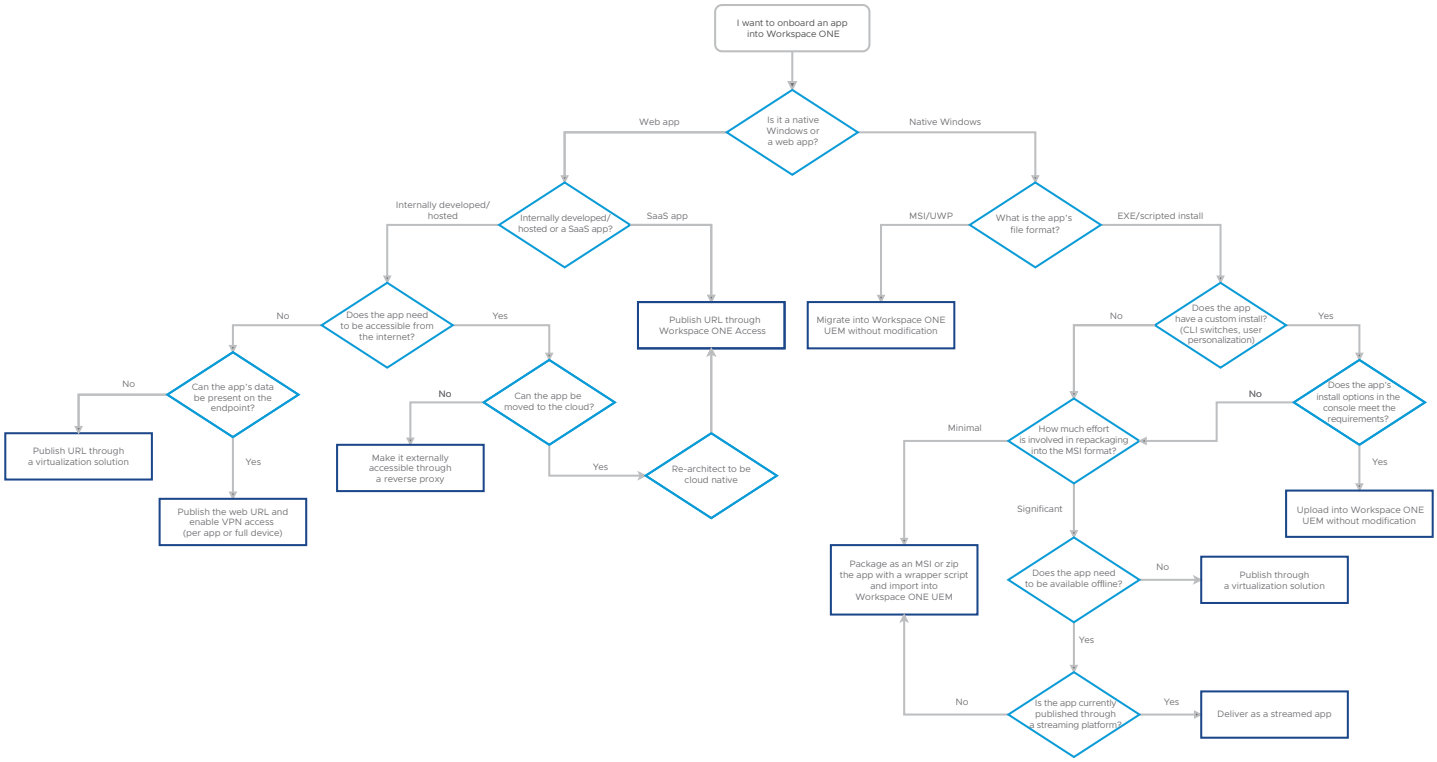

FIGURE 1: Windows application delivery decision tree.

Modern Windows application delivery end states

The following app delivery methods ensure streamlined delivery through the Workspace ONE platform:

- MSI
- Universal Windows Platform (UWP)
- Web hosted, either internally or in the public cloud
- Virtualized through Horizon or Citrix Virtual Apps and Desktops

While there will always be exceptions, standardizing the application fleet to these methods will future-proof an organization's Workspace ONE deployment for their Windows 10 devices.

To achieve this standardization, there are five strategies for an organization's apps that can be followed:

1. Re-architect the application to be cloud native
2. Publish as a web-based application
3. Repackage the app to a modern native platform
4. Deliver the app through a virtualization solution
5. Migrate as it stands, with no modifications

These strategies guide the augmentation required to enable effective delivery in the Workspace ONE platform. Many applications in an organization's environment may already be at these end states, while others may need to go through significant development/ packaging work in preparation to onboard into Workspace ONE.

The aim of these strategies is to use Workspace ONE to successfully import, patch, update and deliver apps to managed Windows 10 devices. IT organizations will be able to follow a defined change management process to modify the app in Workspace ONE.

Strategy 1: Re-architect the application to be cloud native

This strategy recommends that an organization transform their internal web-based applications to follow the principles of cloud application design.

Cloud-first, SaaS-based solutions—such as Office 365, Salesforce and Workday—have forced organizations to rethink the way they build their internal web applications and host them in cloud-scalable platforms. As cloud-based technologies continue to grow and become pivotal to operating an organization, IT organizations will continue to heavily invest in a multi-cloud strategy, and that should extend to internal web-based applications.

While this re-architecture will usually initiate an internal project to complete this work, an organization can reap the benefits of cloud scale, resiliency and improved user experience with an initial step of moving traditional business application workloads from internal servers to cloud native platforms. VMware is in a unique position to assist organizations with the initial step to hybrid cloud, and the ultimate goal of cloud native, through the VMware Tanzu™ portfolio.

Once moved into a cloud-scale platform, an IT administrator can publish the URL through Workspace ONE Access™ to the end user and enable SSO through authentication methods, such as SAML. The user can launch the URL using a native browser by clicking on the app icon in Workspace ONE Intelligent Hub.

Strategy 2: Publish as a web-based application

Web-based applications have been a long-standing method for delivering LOB apps to end users. Web apps generally fall into two broad categories:

- Intranet apps – This includes both internally developed websites and third-party products hosted on an intranet web server.
- SaaS apps – Cloud native applications developed by third parties and often consumed by organizations through a subscription.

Intranet applications have many factors that determine the correct delivery model. A design decision should be made on whether the application needs to be accessible from the internet or not. It is the responsibility of IT and Information Security to assess the relevant risk profile of the application data to determine whether VDI or native application access is the most appropriate.

If possible, modernizing web applications should also include a seamless authentication framework for SSO. Enabling web-based applications to support SSO can be achieved by introducing a password and access management (PAM) provider into the application web server. Some web-based applications support this capability natively, while others require an authentication plug-in, such as a SAML PAM plug-in for Apache Web Server.

Apps defined as internal only can be made available in Workspace ONE Intelligent Hub through an RDSH app workload in Horizon, or published natively on Windows 10 using per-app or a full-device virtual private network (VPN). A VPN solution allows for the web-based intranet app to be accessible on a device on any network. The web URL for the resource can be published through Workspace ONE Access, and it is also suggested that the Workspace ONE UEM admin include the URL to the allowlist in the VPN profile.

Intranet apps deemed internet accessible can be delivered using the reverse proxy function of VMware Unified Access Gateway™. Unified Access Gateway can be enabled to load balance requests to the internal website and facilitate credential translation for SSO into the web app. Apps front-ended by Unified Access Gateway can be published as a web URL through Workspace ONE Access, and configured for SAML-based authentication and conditional access.

In some cases, browser-based applications have browser version/type dependencies (e.g., must use Internet Explorer 11), and that requirement may be best suited to app delivery through a virtualization solution such as Horizon or Citrix Virtual Apps and Desktops.

In the case of an existing SaaS application, publishing them through Workspace ONE Access allows the administrator to define assignments and conditional access rules based on identity, access medium (device) and network location.

Strategy 3: Repackage to a modern native platform

Applications natively installed on Windows desktop clients are the primary way end users interact with business systems. Many organizations invest in internal development teams responsible for creating specific LOB Windows applications or augmenting third-party app software to fit their business needs.

Most native Windows 10 apps exist as EXEs, MSIs or scripted installations. All of these app formats can be migrated to Workspace ONE. However, formats such as MSI and UWP are preferred for modern Windows app delivery. The major benefits are that MSI and UWP apps include:

- The install and uninstall commands for the app
- The application bundle ID or a developer code, which can be used as part of the installation detection criteria and application inventory management
- A simple and modular approach to application configuration
- Standardized reboot and success exit codes

Where possible, move Windows apps to the MSI and UWP formats for delivery through Workspace ONE. Moving the app to an MSI allows organizations to take advantage of MSI transform configuration files and command line parameters. Transform files help modularize the application components, and allow IT to build a lifecycle management process for the app.

There are many public tools available to help repackage EXEs into the MSI and UWP formats. Tools such as *Flexera InstallShield* and *WiX* allow for the capture of the app installation components (such as file, folder and registry key creation), which is used in the creation of the MSI installation package.

MSI and UWP applications can be delivered to Windows 10 devices managed by Workspace ONE by using the software distribution platform in Workspace ONE UEM. An IT admin can upload the MSI or UWP app into the console, and Workspace ONE UEM will extract the application file and associated installation parameters. The admin can assign the uploaded app to a group of users, and then specify for the app to be installed automatically or make it available through Workspace ONE Intelligent Hub.

In addition, some third-party apps are packaged in the UWP format and delivered through the Microsoft Store for Business. An admin can integrate Workspace ONE UEM into this platform to deliver LOB apps through the Microsoft Store. For those apps not released through the Microsoft Store, an admin can upload those UWP apps through the Workspace ONE UEM console.

Strategy 4: Deliver through a virtualization solution

Another popular solution for delivering Windows applications is through a virtualization or streaming solution. Technologies such as Horizon and Citrix Virtual Apps and Desktops provide remote app access through a remoting client running on the Windows 10 device.

Virtualizing apps is primarily used for native Windows apps and web applications that should only be accessible internally to the organization, or have browser dependencies that are difficult to maintain or satisfy, such as a specific browser and plug-in version combination. Virtualizing apps is also especially useful for apps and web URLs that have high-risk data or information that should not leave the organization's data center.

Another use case for this technology is if apps have dependencies that aren't supported on Windows 10 (e.g., an older version of the .NET Framework, Java, etc.). An end user can access the app from a virtualized instance of a supporting operating system remotely delivered from Horizon or Citrix Virtual Apps and Desktops through Workspace ONE Intelligent Hub, and can be integrated with their SSO solution for seamless access.

The output of an app rationalization exercise may lead organizations to move some of their native Windows apps into a virtualization solution. This will reduce some of the complexity around app delivery and updating, as this will be done in the central source in the virtualization platform.

Streaming technologies such as ThinApp are another option to make the app available in Workspace ONE. A ThinApp resource can be advertised to the end user through Workspace ONE Intelligent Hub and streamed to the endpoint device once they select the app in the catalog. Applications that already reside in the ThinApp platform can continue to be leveraged when moving to modern Windows application delivery.

Strategy 5: Migrate with no modifications

In some cases, there is a huge amount of work effort involved in re-architecting, repackaging or moving the app from its current platform to one of the four end states previously mentioned. For legacy Windows applications, this generally applies to complicated scripted installations or EXEs in the organization's environment.

For apps that exist in the Microsoft Endpoint Configuration Manager platform (formerly known as Microsoft System Center Configuration Manager [ConfigMgr]), Workspace ONE AirLift can assist with this migration process. Workspace ONE AirLift connects to ConfigMgr and an admin can review the apps, determine the validity of the apps to migrate, select the appropriate applications that need to be migrated to Workspace ONE UEM, and migrate those apps with the push of a button.

The software distribution platform in Workspace ONE UEM provides an admin with some flexibility to upload and configure LOB applications. These include defining the application version, the installation commands, how to call the installation complete, and the exit codes for reboot and successful execution. App ordering can be used to define the sequence in which applications are installed in the platform, and which dependency apps are present/installed before the LOB app install is initiated.

An admin can assign the uploaded app to a group of users and then specify if the app is automatically installed or made available for self-service through Workspace ONE Intelligent Hub.

Summary

Moving to Workspace ONE presents an organization with many benefits, such as cloud application delivery, cloud scalability and insights to the device across any network. When moving or onboarding new apps into the Workspace ONE platform, many factors determine the optimal delivery model to provide end users with the best experience.

The design decision tree enables IT professionals to determine which delivery method is recommended for their application. There are five strategies to ensure an organization can deliver apps to their Windows 10 fleet successfully. This will allow end users to perform their duties, and provide IT with the tools to maintain the app through its lifecycle in the organization.

Additional resources

For more information, please see the following resources:

- *Understanding Windows 10 management learning path*
- *Modernizing Windows 10 Management: VMware Workspace ONE Operational Tutorial*
- *Hands-on Lab: Desktop Management with Workspace ONE UEM*
- *Deploying Win32 Applications tutorial*
- *Workspace ONE UEM integration with Flexera AdminStudio*
- *Integrating Microsoft Store for Business tutorial*

Authors and contributors

Adarsh Kesari, staff solutions architect in VMware End-User Computing (EUC), authored this white paper. Adarsh's primary function is to provide transformational guidance to help organizations adopt Windows 10 modern management using VMware Workspace ONE.

Contributors to this document include:

- Mike Nelson, senior solutions architect, Windows Product Engineering, VMware
- Josue Negron, EUC staff architect, End-User Computing Technical Marketing, VMware
- Phil Helmling, APJ staff solutions architect EUC, EUC Office of the CTO, VMware
- Pim van de Vis, senior solutions architect, Windows Product Engineering, VMware
- Brooks Peppin, senior solutions architect, Windows Product Engineering, VMware
- James Murray, senior staff solutions engineer, End-User Computing Systems Engineering, VMware
- Bryan Garmon, senior solutions engineer, AMER End-User Computing Systems Engineering, VMware

VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 vmware.com Item No: 537869aq-wp-win-app-dlvry-dsgn-uslet-Final2 6/20