ivanti

User Workspace Manager

Install and Configure Guide
Version 2018.3 SP1

Copyright Notice

Contents

About User Workspace Manager

User Workspace Manager Products

Ivanti User Workspace Manager product suite consists of: Application Control, Environment Manager, and Performance Manager.

User Workspace Manager Management Center is the framework that enables the User Workspace Manager products, Application Control, Environment Manager, and Performance Manager, to be used across an entire enterprise.

Consoles

User Workspace Manager consoles launch when the appropriate link is selected in the **Start** >**All Programs**> **Ivanti** menu. The graphic below uses Environment Manager as an example.

Quick Access Toolbar

Ribbon

Navigation Pane

Work Area

Configuration File

Configuration files contain the settings created using the product console.

Configurations are stored locally in the *All Users* profile and are protected by NTFS security. In Evaluation mode, configuration changes are saved in the custom Ivanti .a*mp format (*=product initial) and read by the agent.

In Advanced mode, configurations are stored in the User Workspace Manager Management Center database, and deployed using the Management Center console.

Configurations can be exported and imported to and from MSI file format using the product consoles. This is useful for creating templates or distributing configurations using third-party deployment systems.

After creating or modifying a configuration, you must save the configuration with the latest settings to ensure that they are implemented.

Agent

User Workspace Manager products are installed and run on endpoints using a lightweight agent. The agent is deployed to managed computers to implement the configuration rules. In Evaluation mode, the agent is installed directly onto the local computer. In Advanced mode, configurations are stored centrally and deployed remotely across a network to multiple controlled computers using the User Workspace ManagerManagement Center.

Agents are constructed as Windows Installer MSI packages, which allows them to be distributed using any third-party deployment system that supports the MSI format.

For more information about deploying User Workspace Manager products, see the Management Center Help.

Application Control

Use Application Control to control which applications a user receives on their physical or virtual desktop.

Application Control provides protective measures, such as automatically blocking the execution of all authorized applications, eliminating the threat of a user introducing - either intentionally or unintentionally - an executable file to a network.

The product gives you granular control so that you can decide at user level who has the authority to run specific applications.

AM Web Service

The AM Web Service is installed on any selected machine as part of the Application Control installation. It is a lightweight component that does not require typical server tools such as Internet Information Services (IIS) or SQL Server. In Evaluation mode, the service is installed on any selected machine. To install the Service as part of the Advanced mode, the Application Privilege Discovery option must be selected.

For more information about Application Control and Application Control Web Services, see the Application Control Help.

Environment Manager

Environment Manager is a user virtualization solution that ensures users always receive a consistent, predictable, and personalized working experience.

User virtualization represents a fundamental change in the way the corporate desktop is constructed, delivered, and managed.

Environment Manager enables standard desktop images to be used to deliver fully configured and personalized desktops to all employees. The user component of a desktop (user personality) is decoupled from the operating system and applications, managed independently, and applied into a desktop as required. This is achieved without scripting, group policies, or use of roaming user profiles - regardless of how the desktop is delivered.

Environment Manager provides a more efficient alternative to roaming profiles, reducing the potential for profile corruption and providing users with a consistent and seamless working experience.

Because Environment Manager applies user data on demand, you can combine delivery methods, migrate users between platforms and operating systems, and update corporate desktops without impacting user experience.

Environment Manager uses a combination of Policy Configuration and User Personalization to enable comprehensive user virtualization.

- **Policy Configuration** - Use corporate policy to set up a corporate desktop environment and specify what users can access, how they access it and what they can do with it.
- **User Personalization** - Constitutes anything a user is able to customize on their endpoint. This includes the desktop look and feel, application menus and buttons, language settings, and screen resolution.

Environment Manager enables users to have a single personality that is accessible from any location and any device. The settings save on disconnection, enabling the user to work offline. Any changes made to their personality while they are offline are synchronized with the corporate network when they reconnect.

The Environment Manager user interface provides the ability to create and modify both Policy and Personalization configurations. Both of these are created and deployed to endpoints in different ways. Policy configuration settings are stored in the a configuration file (AEMP), which must be deployed to endpoints before it can take effect. For User Personalization, the user interface maintains a live connection to the database. Changes are immediately saved to the database and take effect at each endpoint the next time it performs a configuration poll.

For more information about Environment Manager see the [Environment Manager Help](.).

Environment Manager Administrative Tools

Environment Manager is packaged with the following standalone tools, which assist administrators to create configurations and work with the Personalization Database.

- Environment Manager Logging Setup
- Environment Manager Monitor
- Personalization Server Log Viewer
- Environment Manager Log File Conversion
- EMP File Utility
- EMP Migrate Utility
- EMP Migrate Command Line Utility
- EMP Registry Utility
- File Based Registry Explorer

For more information, see [Environment Manager Administrative Tools](.).

Personalization Operations

Personalization Operations is an Ivanti Environment Manager utility that provides management of Personalization data via a web console. Depending on their role, users can manage backups and current settings for either single users or multiple users at a time. They can also search for and delete audit logs, and view the migration status of Personalization Groups.

> **i** For further information, see the [Personalization Operations Help](#).

Performance Manager

Use Performance Manager to implement rules to manage precisely the allocation and distribution of CPU and disk resources for applications and users on your system. Performance Manager includes CPU thread throttling to control demand on resources and ensure the efficient and smooth running of the system.

Performance Manager provides a fine level of granular management to allocate resources based on the state of a session, applications, or the desktop.

For further information, see the [Performance Manager Help](#).

Management Center

Management Center is a scalable multi-tier system that enables the central management and secure deployment of configuration information to thousands of endpoint devices and user environments. The Management Center incorporates comprehensive auditing and reporting, and provides failover support for server resiliency.

The Management Center, comprises of the Management Server, Database (Microsoft SQL Server), Management Console, and the Deployment Agent which must be installed on each managed endpoint.

The Deployment Agent uploads event data from managed endpoints via the Management Server to the database. Product configurations are created in the product consoles and stored in the Management Center database, from where they can be downloaded along with product agents and software updates by the Deployment Agent for installation on managed endpoints.

Management Center Components

The Management Center includes the following components:

- **Management Console** - The Management Console provides an interface to the Management Server and the other components of the Management Center, allowing you to control deployment groups, users, event data and alerts, configurations and packages, managed endpoints, and reports.

- **Management Server** - The Management Server manages communications (using Microsoft Internet Information Services - IIS) with a Microsoft SQL Server database for data access and storage, providing security control, communications for managing network discovery services and software deployment to managed endpoints, resource management, and auditing.

- **Database** - The Management Center must have access to a Microsoft SQL Server database for the storage and retrieval of User Workspace Manager software agents, configuration packages, license packages, and event and alert data. The Management Server and can be installed locally on the Management Center host computer or on a remote computer.

- **Deployment Agent on Managed Computers** - The Deployment Agent is installed on managed endpoints to manage communications between the product agents and the Management Center. You can deploy the Deployment Agent as follows:

  - Use the Install Deployment Agent functionality in the Management Console.

  - On each endpoint download the agent from the Management Server website and install it.

  - Use a third-party deployment mechanism.

For further information, see the [Management Center Help](#).

Prerequisites

Supported Languages

- English
- German
- French

Supported Operating Systems and Technologies

The Supported Operating Systems and Technologies are detailed in the Maintained Platforms Matrices available on Ivanti Support Portal.

System Requirements

The table below contains the minimum and recommended hardware requirements for running User Workspace Manager products.

Component	Requirement	
User Workspace Manager Agents	See Microsoft system requirements for the specific Windows version.	
User Workspace Manager Consoles	Processor	Minimum speed: 1 GHz (x86) or 1.4 GHz (x64) Recommended speed: 2 GHz or faster Minimum # of CPUs: 1 Recommended # CPUs: 2 or greater Refer to Windows editions documentation on support for more than 4 CPUs
	Memory	Minimum: 2 GB RAM Recommended: 4 GB RAM or greater Refer to Windows editions documentation on support for more than 4 GB RAM (x86) and 32 GB RAM (x64)
	Available Disk Space	Minimum: 1 GB

Component	Requirement	
	Minimum Resolution	1024 x 768 pixels
Personalization Server	Processor	Minimum speed: 1.4 GHz (x64) Recommended speed: 2 GHz or faster Minimum # of CPUs: 1 Recommended # CPUs: 2 or greater Refer to Windows editions documentation on support for more than 4 CPUs
	Memory	Minimum: 2 GB RAM Recommended: 4 GB RAM or greater Refer to Windows editions documentation on support for more than 32 GB RAM (x64)
	Available Disk Space	Minimum: 10 GB
Personalization Database Server	As the system requirements for the specific SQL Server version, plus 2 GB of available disk space for the initial installation.	
Application Control Web Services	Processor	Minimum speed: 1 GHz (x86 or x64)
	Memory	Minimum: 1 GB RAM (x86) or 2 GB RAM (x64)
	Available Disk Space	Minimum: 10 GB
Management Server	Processor	Minimum speed: 1.4 GHz (x64) Recommended speed: 2 GHz or faster Minimum # of CPUs: 1 Recommended # CPUs: 2 or greater

Component	Requirement	
		Refer to Windows editions documentation on support for more than 4 CPUs
	Memory	Minimum: 2 GB RAM Recommended: 4 GB RAM or greater Refer to Windows editions documentation on support for more than 32 GB RAM (x64)
	Available Disk Space	Minimum: 10 GB
Management Console	Processor	Minimum speed: 1 GHz (x86) or 1.4 GHz (x64) Recommended speed: 2 GHz or faster Minimum # of CPUs: 1 Recommended # CPUs: 2 or greater Refer to Windows editions documentation on support for more than 4 CPUs
	Memory	Minimum: 2 GB RAM Recommended: 4 GB RAM or greater Refer to Windows editions documentation on support for more than 4 GB RAM (x86) and 32 GB RAM (x64)
	Available Disk Space	Minimum: 1 GB
Database Server	As the system requirements for the specific SQL Server version, plus 2 GB of available disk space for the initial installation. See the Maintained Platforms Matrix for more details.	

Required Components

The following components are installed as part of the User Workspace Manager Installer:

- Microsoft Windows Installer 5.0
- Web Server Internet Information Services (IIS)
  - Common HTTP Features
    - Default Document
    - Directory Browsing
    - HTTP Errors
    - Static Content
    - HTTP Redirection
  - Health and Diagnostics
    - HTTP Logging
    - Logging Tools
    - Request Monitor
    - Tracing
  - Performance
    - Static Content Compression
    - Dynamic Content Compression
  - Security
    - Request Filtering
    - Basic Authentication
    - IP and Domain Restrictions
    - Windows Authentication
  - Application Development
    - .NET Extensibility 4.5
    - ASP.NET 4.5
    - SAPI Extensions
    - ISAPI Filters
  - Management Tools
    - IIS Management Console
    - IIS Management Scripts and Tools
    - IIS 6 Metabase Compatibility
- Background Intelligent Transfer Service (BITS)
  - IIS Server Extension
- Remote Server Administration Tools
  - BITS Server Extension Tools
- Windows Powershell 3.0 or above
- Microsoft .NET Framework 4.6.1

- Microsoft Visual C++ 2015 Redistributable package - Update 3 (14.0.24123.0) or later
- SQL Express 2014 SP1 - only applicable for server products
- IIS URL Rewrite Module 2 - only applicable for Personalization Server

  IIS URL Rewrite Module 2 has a dependency on IIS. IIS is automatically installed though the User Workspace Manager Installer for server products.

- XML Lite 1.0.1018.0

Database

SQL Server AlwaysOn is the preferred SQL Server technology to support High Availability/Disaster Recovery screnarios and User Workspace Manager 10.x servers have been optimized to support this technology.

SQL mirroring is available for User Workspace Manager 10.1 FR1 customers who are currently in the process of transitioning to AlwaysOn technology.

SQL AlwaysOn

Guidance on configuring SQL Server AlwaysOn Availability Groups can be found here:

- Overview of AlwaysOn Availability Groups
- Prerequisites, Restrictions, and Recommendations for AlwaysOn Availability Groups
- Configuration of a Server Instance for AlwaysOn Availability Groups
- Creation and Configuration of Availability Groups

The whitepaper on SQL Server High Availability and Disaster Recovery can be found here:

- High Availability and Disaster Recovery for User Workspace Manager SQL Server Databases

If you have configured SQL AlwaysOn with multi-subnet failover availability groups, you must configure the MultiSubnetFailover value in the database connection string for the relevant listener. This can be done using the Server Configuration Portal or Powershell cmdlets.

> For more information on configuring the database connection string, see Setting Up a New Server and Database in the Server Configuration Portal Scripting Guide.

SQL Mirroring

SQL Database mirroring is a strategy which ensures data resiliency by maintaining a real time copy of a database in a mirror SQL Instance. In the event of a failover, this standby database can be employed to provide immediate restoration of service.

The originating server is known as the principal and the standby is known as the mirror. Data is automatically synchronized between the two so the mirror is fully up to date when required.

If set up in accordance with Microsoft best practices, SQL mirroring is supported by Ivanti User Workspace Manager.

> A witness server is required for automatic failover. Without the witness a manual changeover is required.
> SQL mirroring is supported on SQL Server 2012 and 2014 but not SQL Express edition.

Using SQL mirroring with User Workspace Manager Servers

The User Workspace ManagerServer allows the user to add extra parameters to the database connection strings. This can be used to add the Failover Partner parameter so that in a mirror configuration, the User Workspace ManagerServer automatically switches over when the principal fails or is switched over.

 i   Automatic failover requires a witness server as well as the mirrored pair.

Setting up SQL mirroring with User Workspace Manager involves the following steps:

- Initial Installation of the Management Server
- Prepare the Principal and Mirror Database
- Database Mirroring Setup
- Management Server Setup

Initial Installation of the User Workspace Manager Server

The User Workspace ManagerServer should initially be configured to point to the principal database. The Server Configuration Portal (SCP) used to install the database sets up the config files to contain details of the principal database.

If the SCP is run after installation when the original principal is acting as the mirror, it will be unable to connect to the database as the SCP does not recognize the failover configuration.

Connection errors will occur resulting in database variances. If you are switching databases, the web.config must be manually edited to remove the failover partner.

To ensure the services have access to both instances of the mirror pair:

1. The service account must use windows authentication
2. A domain user must be used.

 i   For more information on the SCP see the Server Configuration Portal Help.

Prepare the Principal and Mirror Database

1. Run SQL Server Management Studio on the server hosting the principal database.
2. Right-click the management server database in the Object Explorer and select **Properties**.
3. In the Database Properties dialog, select the **Options** page.
4. Select **Full** from the Recovery mode drop-down.
5. Click **OK**.
6. Once the recovery mode has been set to full, back up the database and the transaction log.

7. Create the mirror database on the mirror server by restoring the full backup followed by the transaction log. Ensure the **RESTORE WITH NONRECOVERY** option in the Restore Database dialog is selected for each restore.

> For further information about restoring databases, refer to the SQL Server Management Studio online help or SQL Server documentation.

Database Mirroring Setup

It is recommended that each SQL Server service is running under a domain account as each database has to have a login for the other SQL Server services.

1. Run SQL Server Management Studio on the server hosting the principal database.
2. Right-click the database in the Object Explorer and select **Task** > **Mirror**.
3. Click **Configure Security** to access the Configure Database Mirroring Security Wizard.
   The wizard sets up the principal and mirror server instances.

For further information about how to set up database mirroring refer to the SQL Server Management Studio online help or SQL Server documentation

Management Server Setup

Once the mirrored pair has been setup, you need to configure the Management Server to enable it to use the failover partner when required, this is done by the following PowerShell commands:

Import-ApsInstanceModule -ProductName "Management Server" -IsDefault

$pw ="Password"|ConvertTo-SecureString -AsPlainText -force

$sc = New-Object System.Management.Automation.PSCredential ("Domain\Username",$pw)

Set-ApsServerDatabase -DatabaseConnection NameOfConnection -ServiceCredential $sc -ConnectionString "Failover Partner=NameOfFailOverServer"

> If the database exists within the default SQL instance (MSSQLServer) do not specify "Server\MSSQLServer" in the web.config.file as mirroring will not work. To use the default SQL instance, specify only the server name.
> For example, if your server "SVR_2k8_01" exists in the default instance, specify "SVR_2k8_01", not "SVR_2k8_01\MSSQLServer".

When the failover partner Powershell command has run successfully the file that gets updated is the databaseconnections.xml.
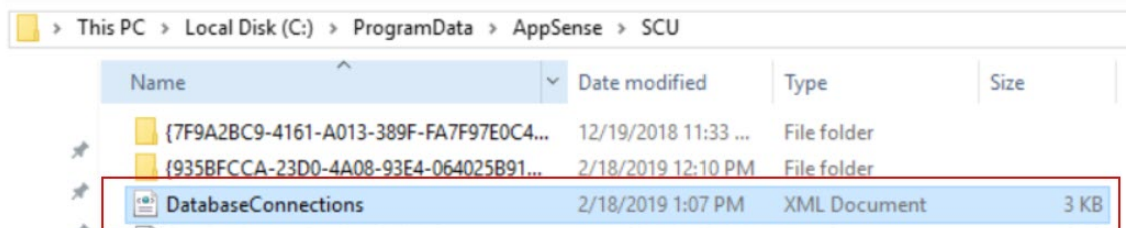
A failover connection is added when the PoSh command is run:

```xml
- <DatabaseConnection ConnectionId="fc5e891c-b76c-4339-80ad-08446c888563">
      <ServerType>935bfcca-23d0-4a08-93e4-064025b91d7b</ServerType>
      <FriendlyName>NameOfConnection</FriendlyName>
      <DatabaseInstance>MD-SQL2016-STD1</DatabaseInstance>
      <DatabaseName>MC_Mirror1</DatabaseName>
      <ConnectionString>Failover Partner=MD-SQL2016-STD2</ConnectionString>
  - <ServiceCredential>
      <UserName>test\profileserviceuser</UserName>
      <Password>AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAAaGQwyIrcME69cYVYD6vW0AQAAAAQAAAA
      <AuthenticationType>Impersonate</AuthenticationType>
    </ServiceCredential>
  - <ConfigurerCredential>
      <UserName/>
      <Password/>
      <AuthenticationType>Windows</AuthenticationType>
    </ConfigurerCredential>
      <CachedDatabaseState>UpToDate</CachedDatabaseState>
      <UpgradeStarted xsi:nil="true"/>
    </DatabaseConnection>
  </List>
</DatabaseConnectionList>
```

The Server Configuration Portal would look like this, which is expected as there is no configuration account details on the failover connection:

Personalization Server Setup

Setting up mirroring does not transfer server logins to the mirror so the service account login must be added to the mirror.

1. In the Object Explorer in SQL Server Management Studio, right-click the **Security** > **Login** node and select **New Login**.
2. Enter details of the service account login and click **OK**.

It is only necessary to add the user as the user's SID is already set in the database; when a failover occurs the user will have access.

Update the web.config File

1. Open the web.config file on the Personalization Server. This is usually located in:
   C:\Program Files\AppSense\Environment Manager\Personalization Server\PS

2. Add the failover partner as highlighted in the example below.

```xml
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <configSections>
    <section name="log4net" type="log4net.Config...." />
  </configSections>
  <appSettings>
    <add key="SqlServerInstance" value="ENC#AQAAANCMnd..." />
    <add key="SqlDatabaseName" value="ENC#AQAAANCMnd8BFd..." />
    <add key="SqlAccountName" value="ENC#AQAAANCMnd8..." />
    <add key="SqlAccountPassword" value="ENC#AQAAANCMnd8..." />
    <add key="SqlAuthenticationType" value="ENC#AQAAANCM..." />
    <add key="SqlConnectionString"
            value="Failover Partner=ServerB\InstanceB" />
  </appSettings>
  ...
```

> ℹ️ If the database exists within the default SQL instance (MSSQLServer) do not specify "Server\MSSQLServer" in the web.config.file as mirroring will not work. To use the default SQL instance, specify only the server name.
> For example, if your server "SVR_2k8_01" exists in the default instance, specify "SVR_2k8_01", not "SVR_2k8_01\MSSQLServer".

Update the BackgroundService.exe.config

1. Open the BackgroundService.exe.config file in the Bin folder on the Personalization Server. This is usually located in:
   C:\Program Files\AppSense\Environment Manager\Personalization Server\BackgroundService

2. Update the file as highlighted in the example below:

```
<appSettings>
    <!-- Interval between attempts to connect to the database  -->
    <add key="RetryConnectionIntervalSecs" value="30" />
    <add key="MinimumExpectedSchema" value="504" />
    <add key="SchemaCommandTimeoutSecs" value="7200" />
    <add key="SchemaCommandDelayMillisecs" value="20" />
    <add key="SchemaLogIntervalSecs" value="60" />
    <add key="ArchiveScanTimeMinutes" value="2" />
    <add key="ActionRecordScanTimeMinutes" value="2" />
    <!-- Following set up by SCU -->
    <add key="SqlServerInstance" value="ENC#AQAAAN..." />
    <add key="SqlDatabaseName" value="ENC#AQAAANCMnd8BFdE..." />
    <add key="SqlAccountName"
value="ENC#AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAAkaXfnzpl..." />
    <add key="SqlAccountPassword"
value="ENC#AQAAANCMnd8BFdERjHoAwE/Cl+sBAAA...=" />
    <add key="SqlAuthenticationType" value="ENC#AQAAA... />
    <add key="SqlConnectionString"
           value="Failover Partner=ServerB\InstanceB" />
    <add key="ClientSettingsProvider.ServiceUri" value="" />
  </appSettings>
```

Install

User Workspace Manager components can be installed using either the User Workspace Manager Installer or manually by using the individual MSIs.

Products can be installed with the Management Center to create integrated enterprise-scale solutions or installed as a standalone product aimed at evaluations.

The User Workspace Manager Installer provides a comprehensive process for installing any combination of User Workspace Manager products in a single fully integrated sequence. The installer performs a complete check for system prerequisites and provides you with the option of installing required technologies automatically.

There are three types of User Workspace Manager installation:

- Evaluation - Install and configure all databases, services, and consoles automatically on one Windows server.
- Advanced - Provides control over which databases, services, and consoles install on each server.
- Clients and Consoles Only - Install selected client agents and consoles.

Alternatively, you can install each of the product components manually, by running the product installer packages for each component.

See section Manual Installation.

> **Caution:** When installing User Workspace Manager products manually, you must ensure that all required technologies and User Workspace Manager components are added. A list of required technologies and components is available in the Prerequisites.

If using Active Directory, ensure that you follow Microsoft's Active Directory Best Practices to enable it to work well with the Management Center Membership Rules and Deployment Groups.

Packages

Installer packages for each component in the User Workspace Manager product set include 32-bit and 64-bit versions of the Agent and Console, and for Environment Manager the PersonalizationServer.

Additional prerequisite third-party software components are provided with the installation media and can be installed automatically via the User Workspace Manager Installer or manually by running the relevant packages provided.

Evaluation Installation

1. Run the User Workspace Manager Installer by executing setup.exe from the installation media.

   The Welcome screen displays.

2. Click **Next** to display the License Agreement screen.

3. Read the company End User License Agreement. If you agree to the terms, select **I accept the terms in the License Agreement** and click **Next**.

   The Installation Type screen displays.
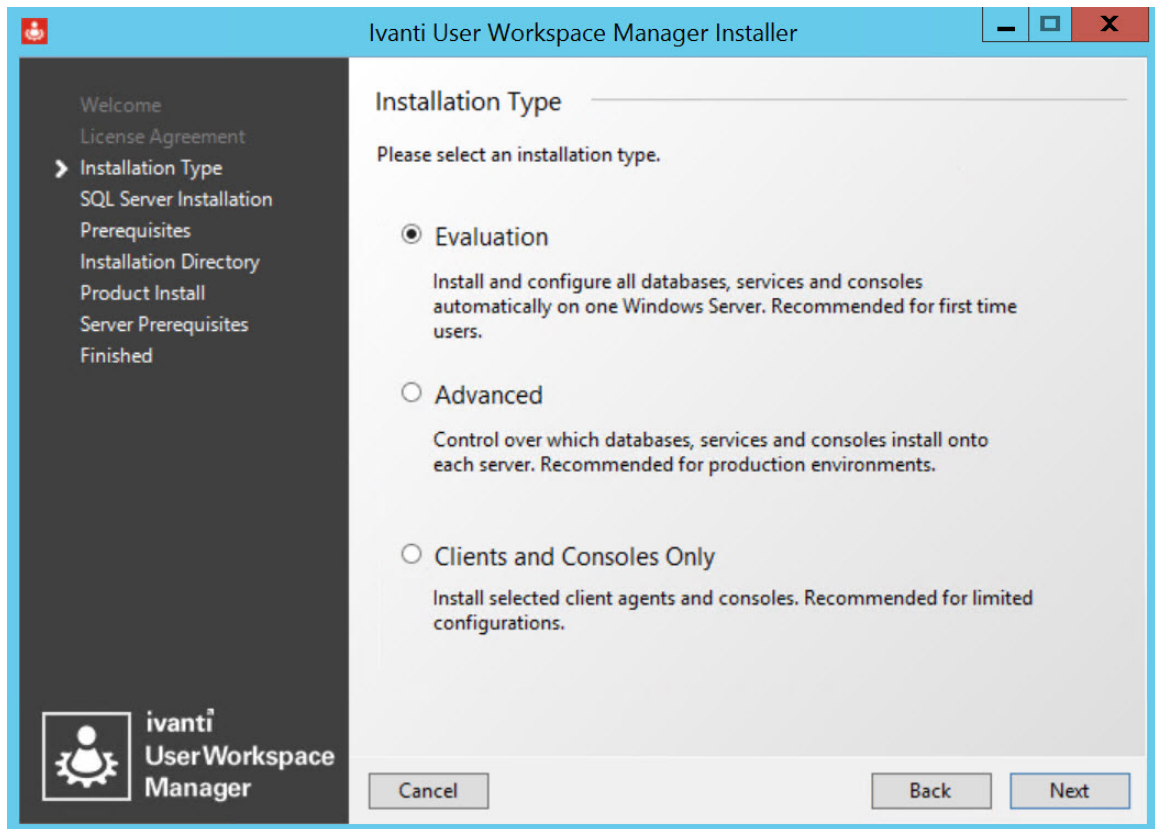
4. Select **Evaluation** and click **Next**.

   > ℹ️ If SQL Server Express is already installed, you must have sysadmin permissions and SQL authentication enabled to select Evaluation installation type.

   The SQL Server Installation screen displays.

5. Read the Microsoft Software License Terms. If you agree to the installation of SQL Server Express and the license terms, select **I accept the terms in the License Agreement**. Click **Next**.

   If there are any prerequisites missing the Prerequisites screen displays.
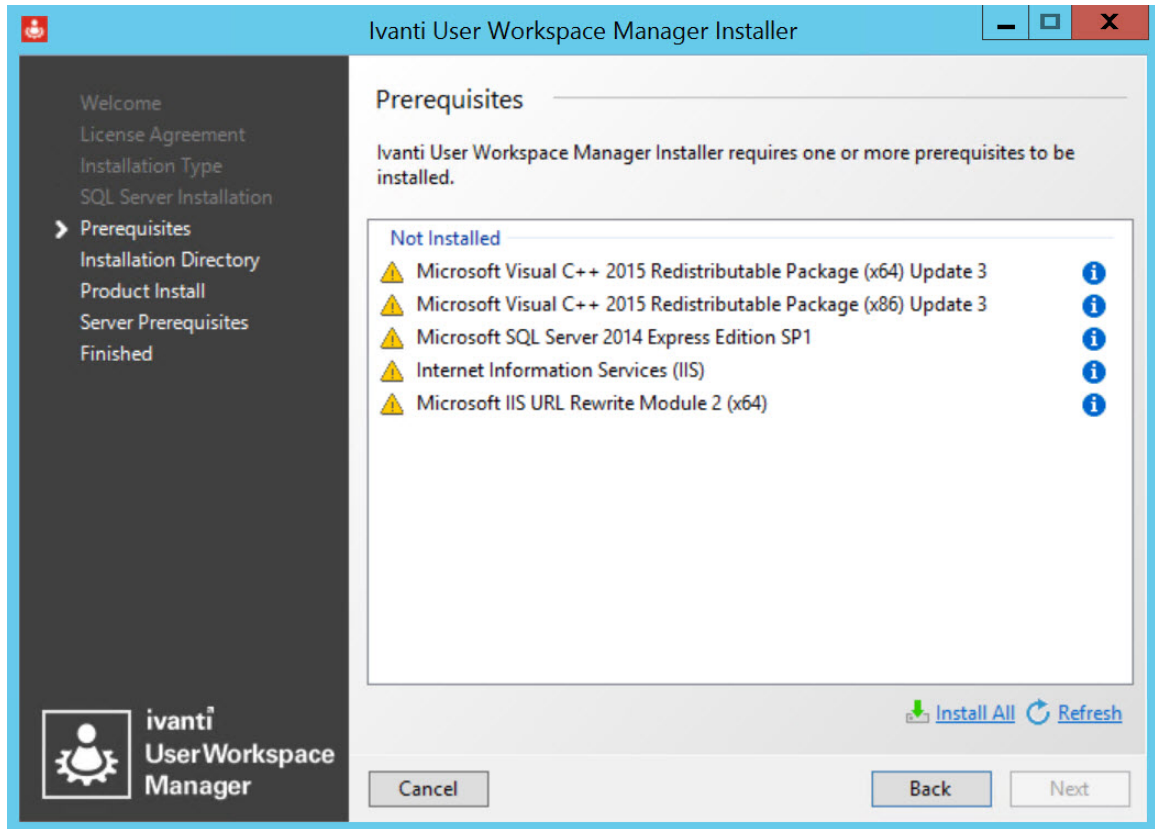
6. The Prerequisites screen lists the prerequisites that are not currently installed. Select **Install All** to automatically install all missing prerequisites.

7. Once all of the components are successfully installed, the Installation Directory screen displays.
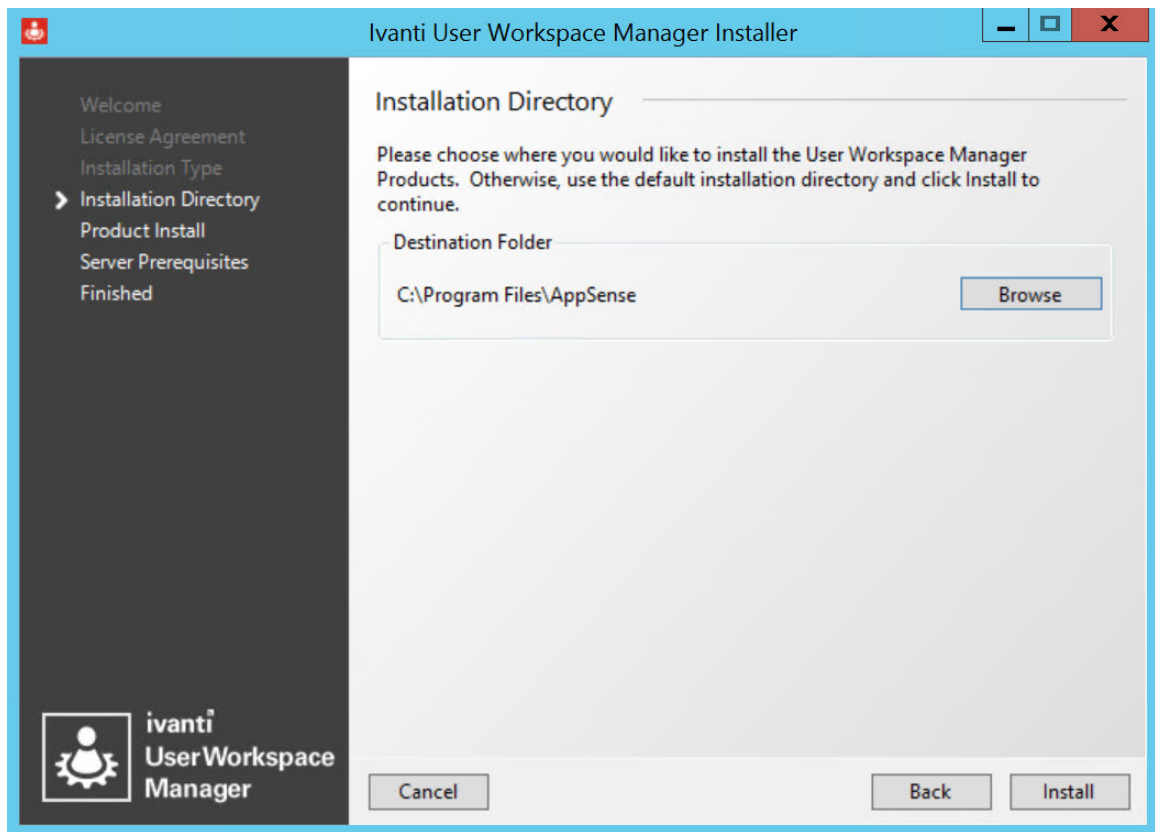
8. The default installation directory is C:\Program Files\AppSense. To continue the installation to the default location, click **Next**. Alternatively, **Browse** to select a new installation location.

9. Click **Install** to start the installation process.

    The Product Install screen displays the progress of the installation.

10. If there are missing server prerequisites the Server Prerequisites screen displays. Click **Install All** to install all the components listed.

11. Once all product components have been installed, the Install Complete screen displays.
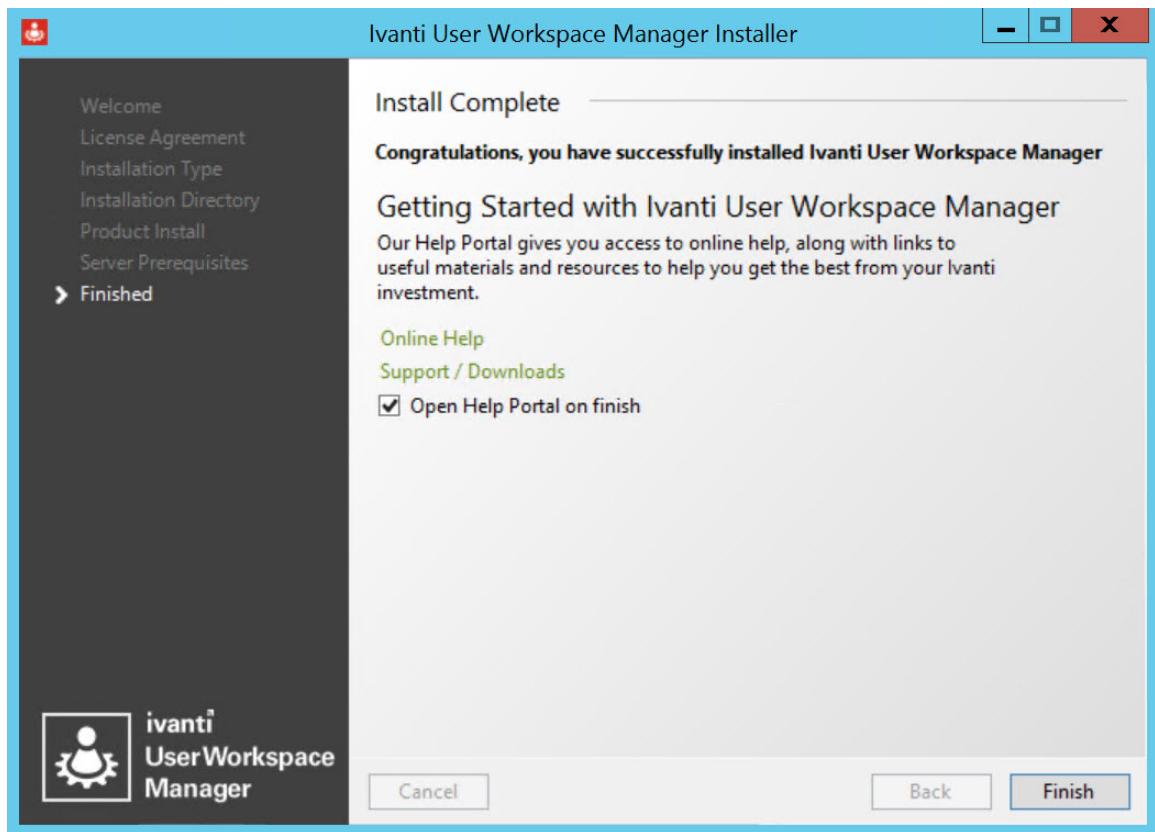
12. Click **Finish** to exit the Installer and open the Product Documentation.

If you want to exit without opening the Product Documentation, deselect **Open Help Portal on finish** and click **Finish**.

Once installation is complete a Server Configuration Portal shortcut is added to the desktop.

See Server Configuration for further details.

Advanced Installation

1. Run the User Workspace Manager Installer by executing setup.exe from the installation media.

The Welcome screen displays.

2. Click **Next**.

The License Agreement screen displays.

3. Read the company End User License Agreement. If you agree to the terms, select **I accept the terms in the License Agreement** and click **Next**.

The Installation Type screen displays.
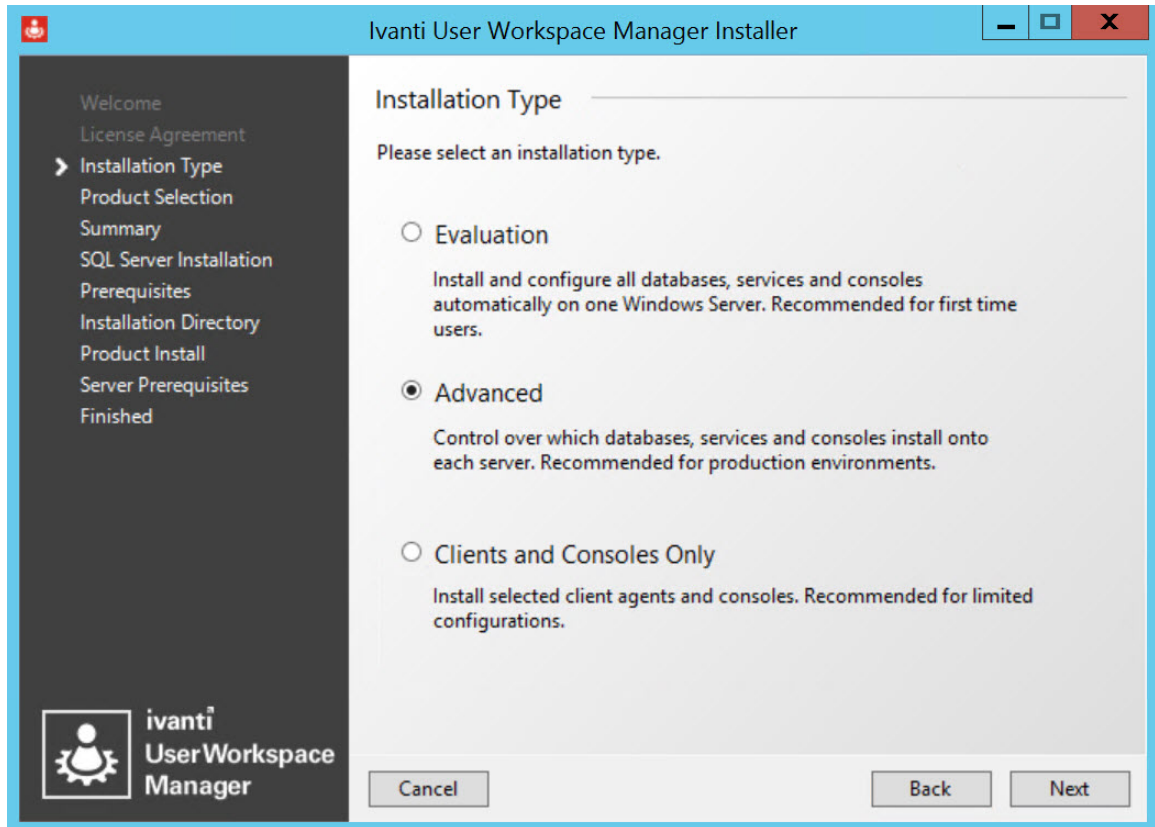
4. Select **Advanced** and click **Next**.

The Product Selection screen displays.
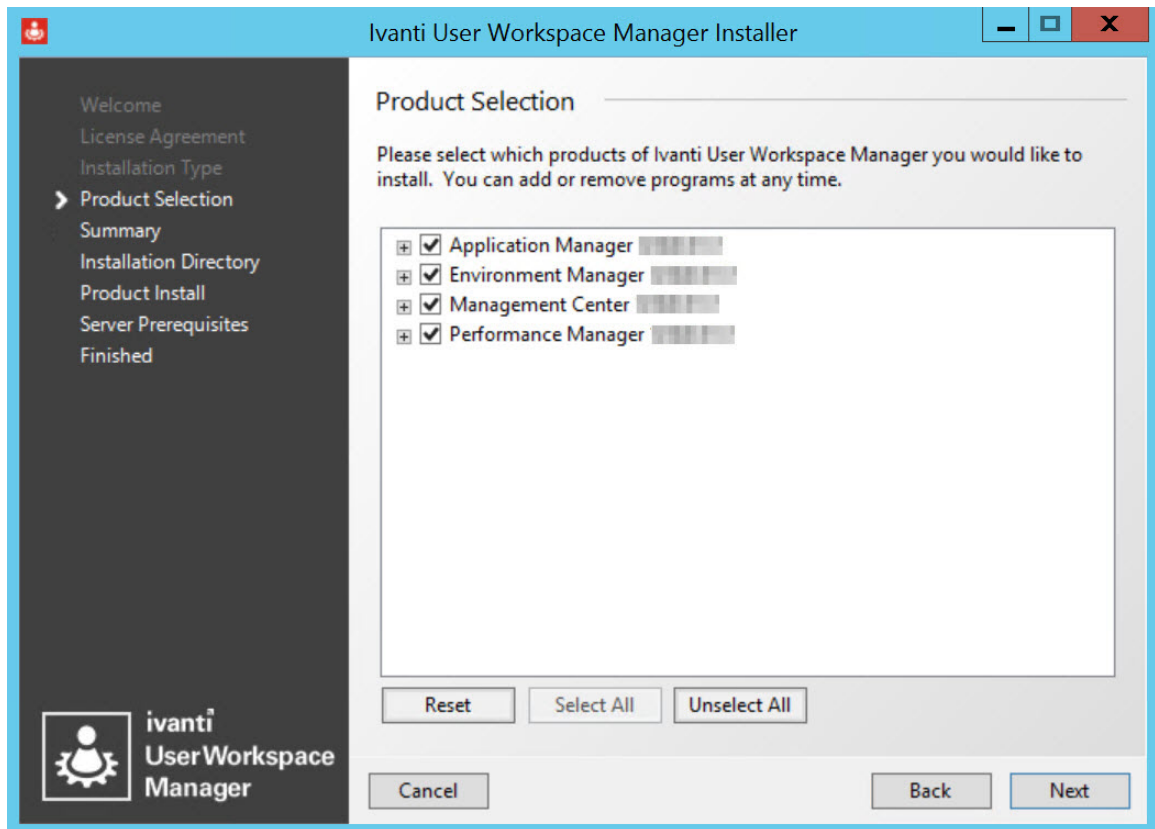
5. Expand the Products to see all the product components that can be selected for installation.

> The Server components do not display if the Operating System is not compatible. Only the Product Consoles can be selected for install.

6. Select all the product components that you want to install.

   If you want to install multiple instances of the Personalization Server or Management Server:

   1. Select **Add new Personalization/Management Server instance**.

      The Add New Instance dialog displays.

   2. Enter the name of the additional instance.

   3. Click **OK**. All instance names must be unique.

   The server instance now displays in the list of components for selection. You can add up to 16 instances.

   If you select **Back** on the Installer a message displays **Remove Pending Instance [instance name]?**. Click **OK** to continue going back and to remove the instance or click **Cancel** to remain on the Product Selection screen and keep the instance.

   For multi-instance environments, further configuration is required, see Extra Configuration for Multi-instance Personalization Servers.

7. Click **Next** to display the Summary screen.

8. A summary of the products selected to install displays. Click **Next**.

   The SQL Server Installation screen displays.

9. Select whether you want to install a local SQL instance. The default selection is No.

   If you select **Yes**, the Microsoft Software License Terms display. Read the terms and if you accept them, select **I accept the terms in the License Agreement**.

   Alternatively, to use a remote SQL instance, accept default selection **No** and click **Next**.

   If there are any prerequisites missing, the Prerequisites screen displays.
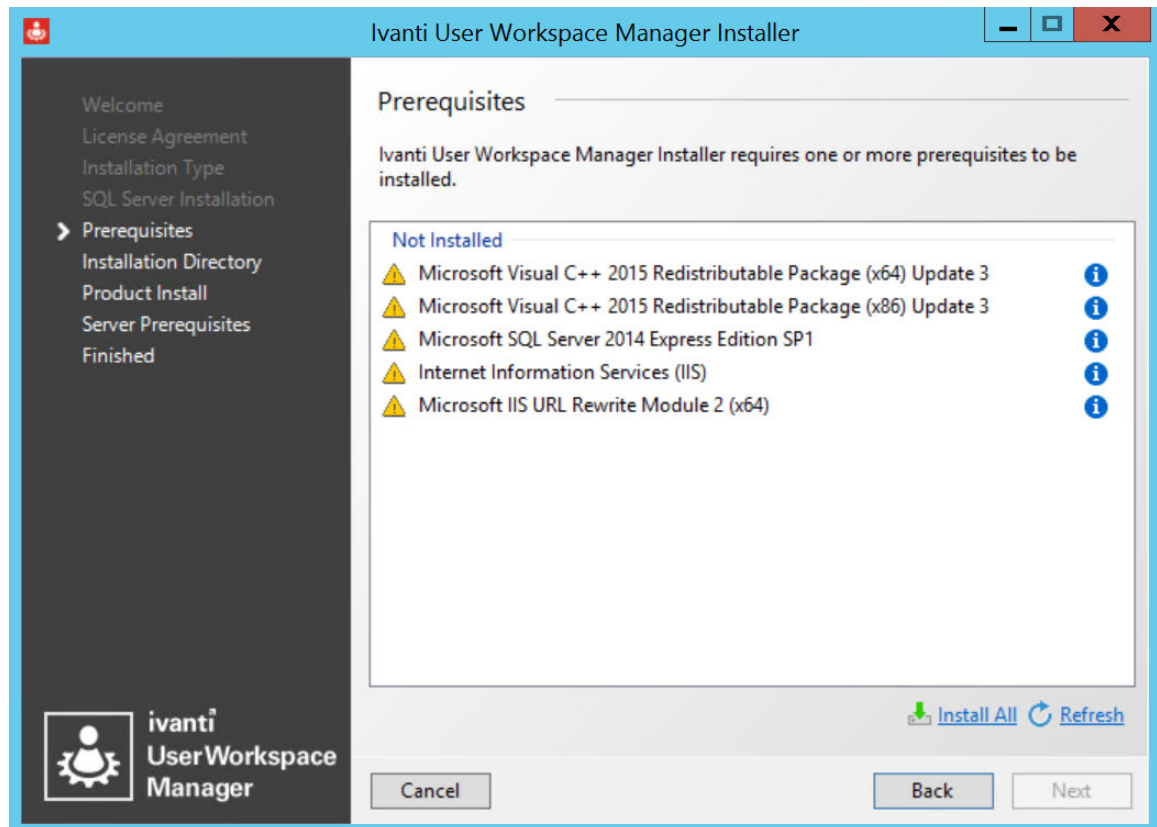
10. The Prerequisites screen lists the prerequisites that are not currently installed. Select **Install All** to automatically install all missing prerequisites.

11. Once all of the components are successfully installed, the Installation Directory screen displays.
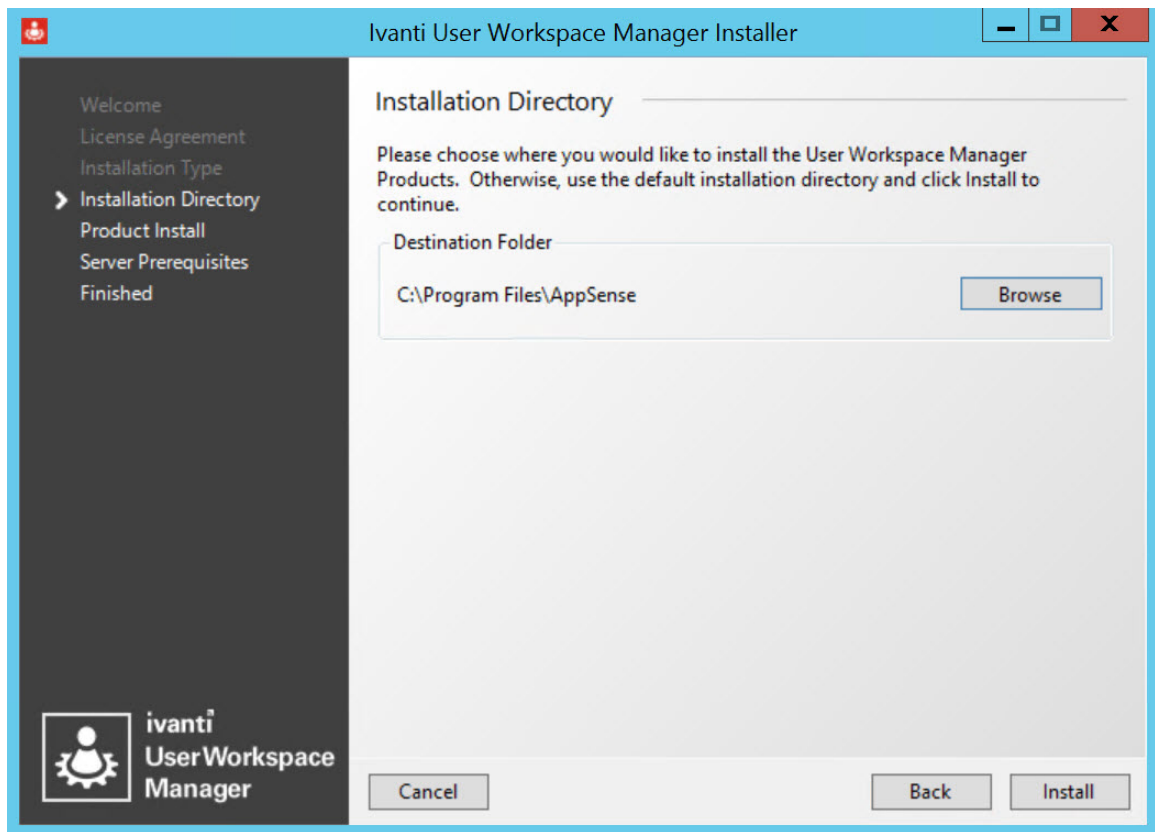
12. The default installation directory is C:\Program Files\AppSense. To continue the installation to the default location click **Install**. Alternatively, **Browse** to select a new installation location. Once you have selected the installation, click **Install**.

    The Product Install screen displays the progress of the installation.

13. If there are missing server prerequisites the Server Prerequisites screen displays. Click **Install All** to install all the components listed.

14. Once all product components have been installed, the Install Complete screen displays.
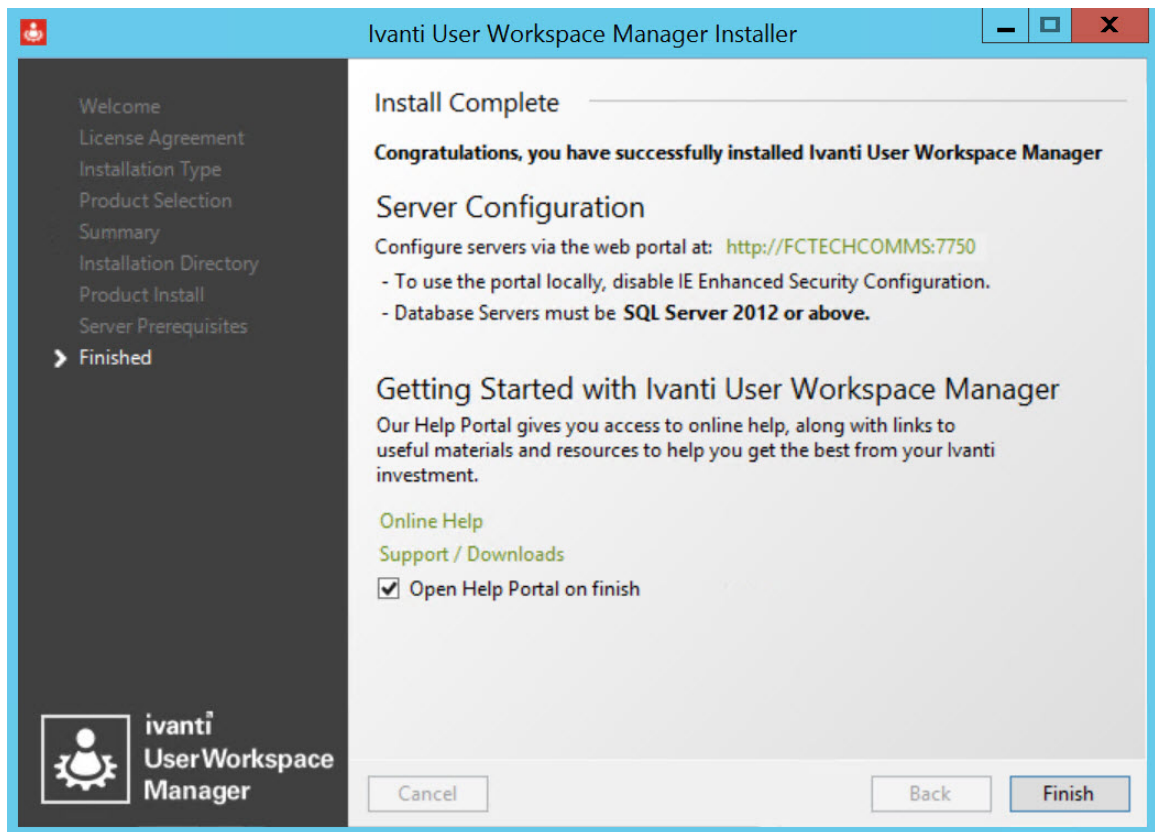
15. Click on the server hyperlink; http://<SERVER>:7750, to display the Server Configuration Portal in a web browser.

    Use the portal to configure the Management and Personalization servers, server instances and databases.

16. Click **Finish** to exit the Installer and open the Product Documentation.

    If you want to exit without opening the Product Documentation, deselect **Open Help Portal on finish** and click **Finish**.

Once installation is complete a Server Configuration Portal shortcut is added to the desktop.

For more information, see Server Configuration Portal.

Extra Configuration for Multi-instance Personalization Servers

When a named instance of the Personalization Server is installed, server self-registration is disabled for non-default instances. The new server must be manually added to the appropriate site using the Sites node in the Environment Manager Console.

For further information, see the Environment Manager Help.

Client and Console Only Installation

1. Run the User Workspace Manager Installer by executing setup.exe from the installation media.

   The Welcome screen displays.

2. Click **Next** to display the License Agreement screen.

3. Read the company End User License Agreement. If you agree to the terms, select **I accept the terms in the License Agreement** and click **Next**.

   The Installation Type screen displays.
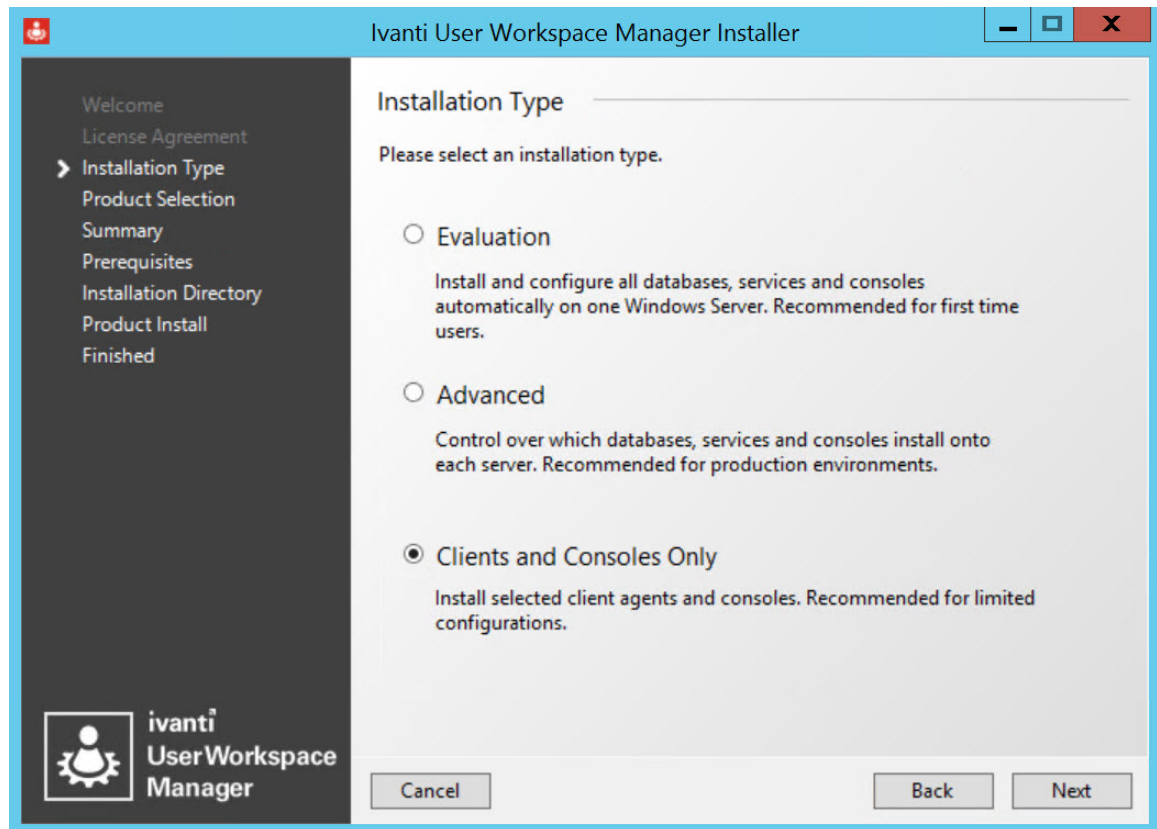
4. Select **Clients and Consoles Only** and click **Next**.

   The Product Selection screen displays.

5. Expand the Products to see all of the product components that can be selected for installation.

   > ℹ️ If you select a component that requires a reboot a warning message displays with the option to continue or cancel.

6. Once you have selected all the required products, click **Next**.

   The Summary screen displays.

7. Click **Next**.

   If there are any prerequisites missing, the Prerequisites screen displays.
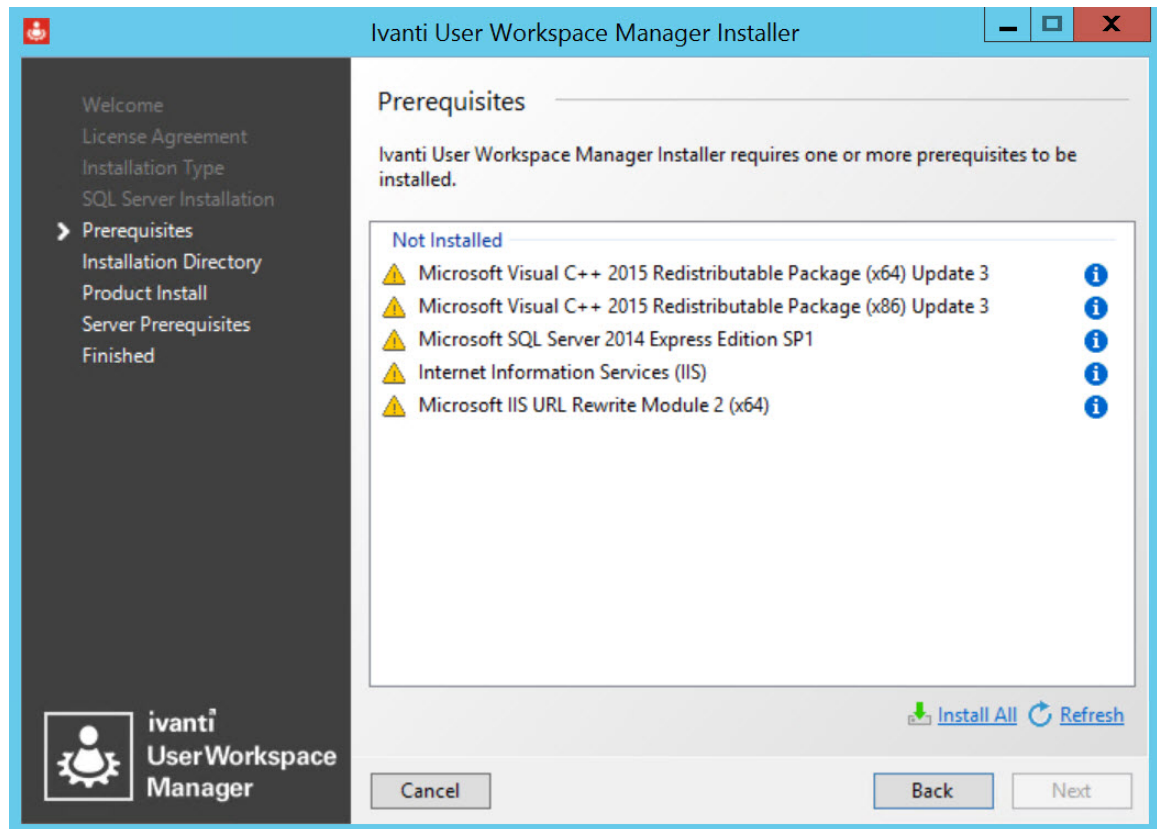
8. The Prerequisites screen lists the prerequisites that are not currently installed. Select **Install All** to automatically install all missing prerequisites.

9. Once all of the components are successfully installed, the Installation Directory screen displays.
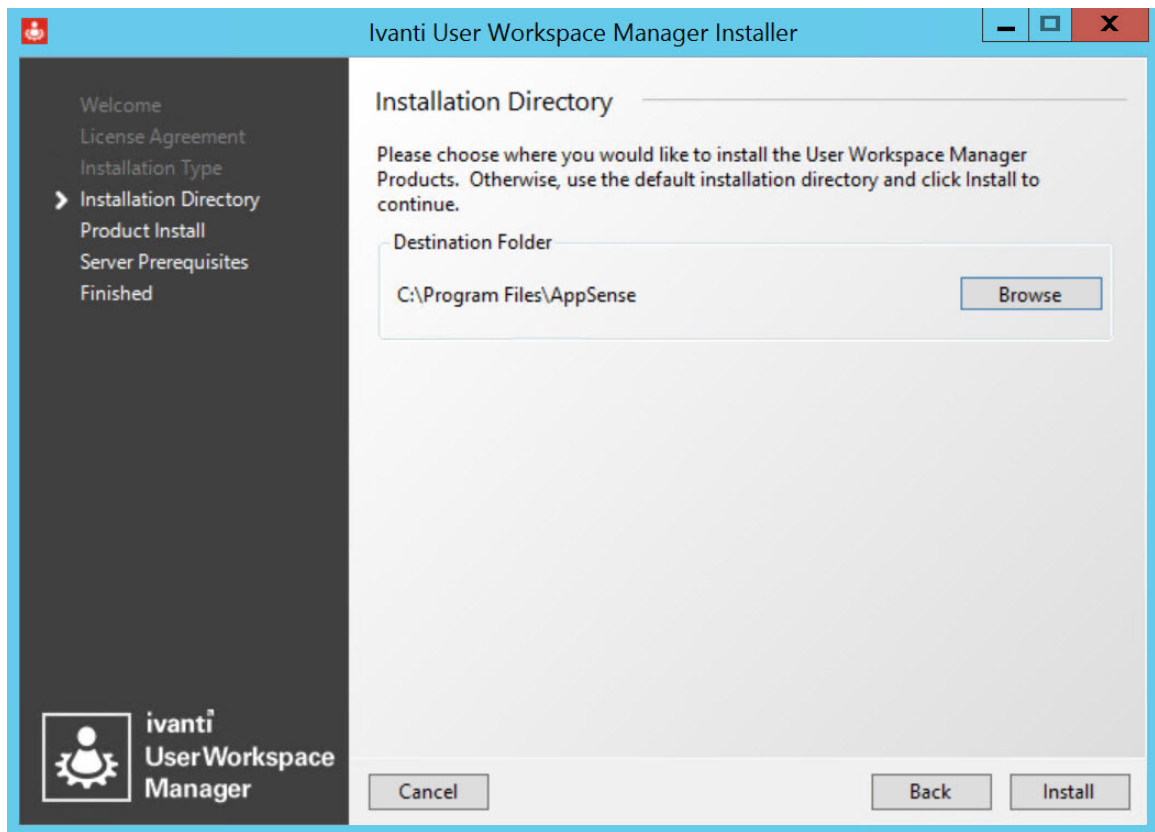
The default installation directory is C:\Program Files\AppSense. To continue the installation to the default location, click **Install**. Alternatively, **Browse** to select a new installation location. Once you have selected the location, click **Install**.

Once all product components have been installed, the Install Complete screen displays.
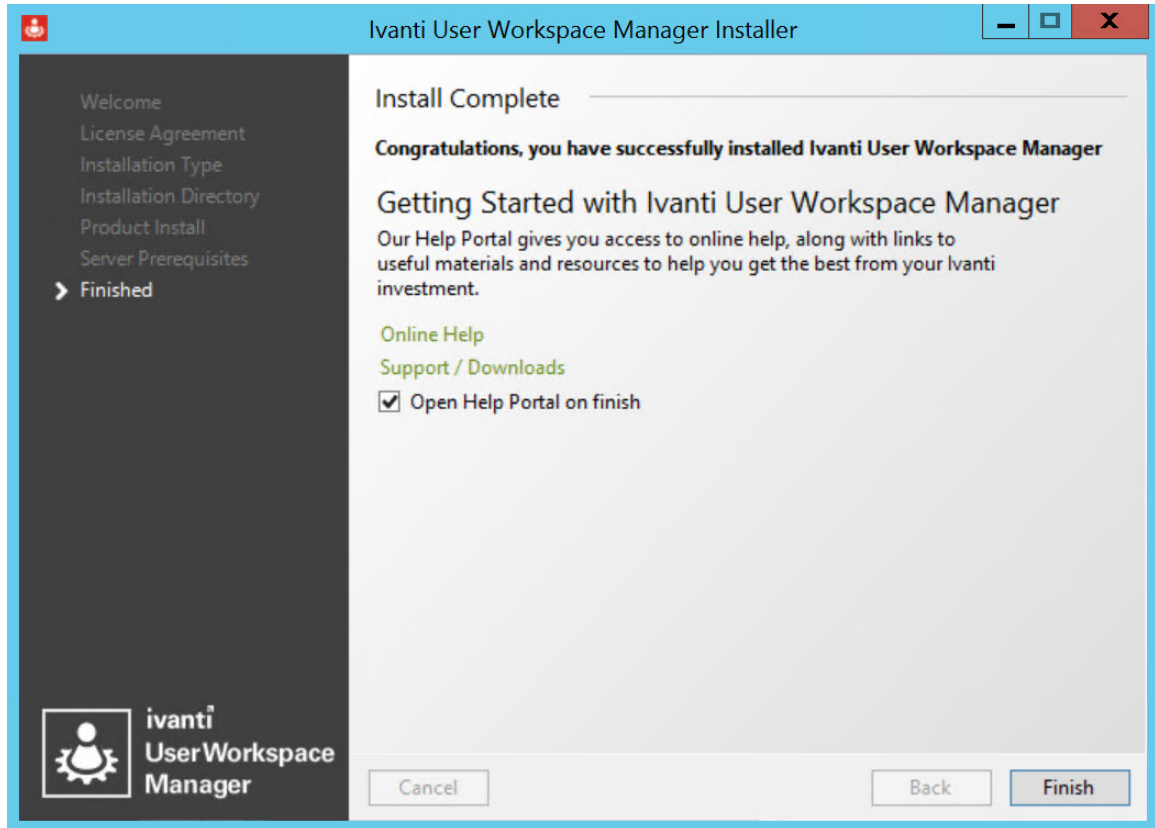
10. Click **Finish** to exit the Installer and open the Product Documentation.

   If you want to exit without opening the Product Documentation, deselect **Open Help Portal on finish** and click **Finish**.

   If you selected products that require a reboot, you need to manually reboot.

Manual Installation

The table below shows the list of the Windows Installer Packages (MSI) for each of the components in the User Workspace Manager suite, which you can run manually on the host computers. The list is organized per product and includes details about which components require a reboot of the host computer after installation.

When installing User Workspace Manager products manually, you must ensure that all required technologies and User Workspace Manager components are added. A list of required technologies and components is available in the Prerequisites.

Installation File	Description
ApplicationManagerConsole32.msi ApplicationManagerConsole64.msi	Installs the Application Control console for creating configurations to deploy to managed computers hosting the agent.
ApplicationManagerAgent32.msi ApplicationManagerAgent64.msi	Installs the Application Control agent on managed computers. When a configuration is installed, the agent implements the configuration rules.
ApplicationManagerWebServices32.msi ApplicationManagerWebServices64.msi	Installs the AM Web Service used for Privilege Discovery to monitor applications that use administrative privileges to run. When installed and configured, the service collates data and allows you to create configurations based on the Privilege Discovery Results report.
PerformanceManagerConsole32.msi PerformanceManagerConsole64.msi	Installs the Performance Manager Console for creating configurations to deploy to managed computers hosting the agent.
PerformanceManagerAgent32.msi PerformanceManagerAgent64.msi	Installs the Performance Manager agent on managed computers. When a configuration is installed, the agent implements the configuration rules.
EnvironmentManagerConsole32.msi EnvironmentManagerConsole64.msi See [Environment Manager Console Variants](#) for further console information	Installs the Environment Manager Policy and Personalization individual or combined consoles for: Creating configurations to deploy to managed computers hosting the agent Configuring the Personalization database. ⓘ For users with a Support Console role, the console opens in read-only mode. For further information, see the [Environment Manager Support Console](#) topic in the Help.
EnvironmentManagerAgent32.msi EnvironmentManagerAgent64.msi	Installs the User Virtualization Service on managed computers. When a configuration is installed, the agent implements the configuration rules.
PersonalizationServer64.msi	Installs the Personalization Server, which synchronizes user personalization settings between the SQL database and the managed computer.

Installation File	Description
	Must be configured using the Server Configuration Portal.
EnvironmentManagerTools32.msi EnvironmentManagerTools64.msi	Installs the Environment Manager Administrative Tools, which are a range of standalone tools to assist administrators when working with the Personalization Database and creating configurations. The tools run independently from Environment Manager and all other User Workspace Manager products.
EnvironmentManagerPolicyTools32.msi EnvironmentManagerPolicyTools64.msi	Installs the BatchConfig tool, so the tool can be used without the need to install the whole EM console. See the Environment Manager Policy Help for details.
ManagementConsole32.MSI ManagementConsole64.MSI	Installs the Management Center Console, which provides an interface to the Management Server and the other components of the Management Center.
ManagementServer64.MSI	Installs the Management Server, which manages data access and storage, security control, network discovery services and software deployment to managed endpoints, resource management, and auditing.
ClientCommunicationsAgent32.MSI ClientCommunicationsAgent64.MSI	Installs the Deployment Agent to manage communications between the product agents and the Management Center.

Environment Manager Console Variants

There are three variants of the Environment Manager console:

- **Personalization** - Installs only the personalization element of Environment Manager
- **Policy** - Installs only the policy element of Environment Manager
- **Both consoles** - Installs the combined console; both personalization and policy are installed.

When you use the User Workspace Manager Installer to install Environment Manager, the combined Policy and Personalization Console is installed. Administrators may not require access to both. For example, they may only be responsible for configuring personalization and have no need for the policy side of the console. If installing the console manually via the EnvironmentManagerConsole.msi you have the option to install the Personalization or Policy console.

Manually Install an Environment Manager Console

1. Double-click the installer appropriate to the operating system:
   - EnvironmentManagerConsole32.msi
   - EnvironmentManagerConsole64.msi
2. On the Welcome screen, click **Next**.
3. Read the license agreement, if you accept the terms, select **I accept...** and click **Next**.
4. On the Destination Folder screen, click **Next**.
5. On the Console Features screen, select the console you want to install:
   - Personalization
   - Policy

   Select both options to install the combined console (this is the default setting).

6. Click **Next**.
7. On the Ready to Install screen, click **Install**.
8. When the Install Complete screen displays, click **Finish** to exit the installer.

Manually Adding Product Consoles to the Management Server Downloads page

To manually add MSI or MSP files to the Management Server Downloads page, follow the following procedure.

1. Browse to C:\Program Files\AppSense\Management Center\Server\Web Site\Downloads
2. Select the appropriate product folder, for example Application Control.
3. Select the appropriate version folder, for example 8.10.0.0, or create it if it doesn't exist.
4. Copy the MSI or MSP files into the folder.

Licensing

About Licensing

License details are included in the License Agreement which is issued when an order for Ivanti software has been completed.

The License Agreement includes the following information:

- Product, Feature, and Version Details
- Issue Date
- Expiry Date
- Customer Name
- Serial ID

Together with the license agreement you will receive either a TXT file or a LIC file. Use these in the User Workspace ManagerLicensing Console to add or import the license.

> ℹ️ **Video:** How to License your DesktopNow/User Workspace Manager products

> ℹ️ For information about managing licenses within the Management Center, see Management Center Help.

When the Licensing Console is launched, all the current licenses display.

Managing Licenses

Add a License

1. Open the User Workspace Manager Licensing console.

2. Click **Add**.

   The Add License Key dialog displays.

3. Enter the License Key and click **Add**.

   If you received a TXT file from Ivanti, open the file and copy the license key, paste it in to the Add License Key dialog.

   If you received a LIC file from Ivanti, refer to the Import a License section.

Details of the license are displayed in the console and the license key is added to the following location:

%ALLUSERSPROFILE%\AppSense\Licenses

Activate a License

Once added, some licenses require activating.

1. Open the User Workspace Manager Licensing console.

2. Select a license or add one to the licensing console.

3. Click **Activate.**

4. Type or copy and paste the activation code.

5. Press **Enter** to accept the code.

   The license console saves the license key to the MS Windows registry on the local machine. The License Status field updates to show the status of the license and the license details display in the lower part of the console.

> To check that the license is active on your endpoint, search the registry for the license code. If the search finds the code, then the license is active.

Remove a License

1. Highlight the required license and click **Remove**.

   A confirmation dialog displays.

2. Click **Yes** to confirm.

   The selected license is deleted and removed from the console, and the MS Windows registry or %ALLUSERSPROFILE%\AppSense\Licenses location, whichever is applicable to the license type.

Import License Files

Import a previously exported license to an endpoint using the Licensing console.

1. Open the User Workspace Manager Licensing console.

2. Click **Import** to display the Windows Open dialog.

3. Navigate to the required LIC file.

4. Click **Open**.

Details of the license are displayed in the console and the license key is added to the following location:

%ALLUSERSPROFILE%\AppSense\Licenses

Delete this text and replace it with your own content.

Export License Files

Export licenses to an MSI or LIC file to create a backup and enable distribution to other endpoints using the Licensing console or the Management Center.

1. Open the User Workspace Manager Licensing console.
2. Highlight the license you want to export.
3. Click **Export** to display the Windows Save As dialog.
4. Browse to the required location to save the license file.
5. Enter a name for the file.
6. Select the file type: MSI or LIC.
7. Click **Save**.

A file is created and saved in the selected location. This file can be copied to any network location and loaded via the User Workspace Manager Suite Licensing console or in the Management Center console.

> For information about managing licenses in the Management Center, see Management Center Help.

Server Configuration

The Server Configuration Portal helps you configure, manage and troubleshoot User Workspace Manager servers and databases.

The installation of the User Workspace Manager Management or Personalization Servers is a two-step process. First, the User Workspace Manager Installer creates the required folders and copies the files to the correct locations. Second, the Server Configuration Portal (SCP) configures the server.

The Server Configuration Portal provides the functionality to:

- Connect Personalization Servers or Management Servers to existing databases.
- Create new databases.
- Configure the User Workspace Manager Servers.

Multiple Personalization Servers and Management Servers can be connected to the same database and managed via the Server Configuration Portal.

> ⓘ   The Server Configuration Portal can only be accessed for servers on the domain.

Database Accounts and Privileges

The Server Configuration Portal uses two SQL database accounts: the Configuration account and the Service account. Both are set up by the database administrator.

Accounts can be added, or changed. Once an account is added, it is assigned to all services.

Configuration Account

The Configuration Account is used to connect to the database to perform operations, including creating, upgrading and configuring the Management Server and database. The account is used to perform the following tasks:

- Create the database - only performed if the database does not exist, requires dbcreator rights.
- Create logins - only performed if a login does not exist, requires securityadmin rights.
- Ensure the database schema matches the version defined by the product.
- Check for variances - whether the properties of the database match the product expectations.
- Confirm the database user logins.
- Populate the initial data set into the database.

The Configuration account must have dbo rights, or be a member of the ManagementServerAdministrator role. Some additional rights may be needed for optional tasks, which are detailed in the list above.

The account can use either Windows Authentication or SQL Authentication.

Service Account

The Service account is used by the Windows services and web services that make up the Management Server. This role has access to all of the Management Server stored procedures.

The Server Configuration Portal persists the username and password of the Service account within the FileName.exe.config and web.config files.

The Service account must be a member of the ManagementServerService role and should not have any additional rights on the database of the SQL instance. The account can use either Windows Authentication or SQL Authentication.

Administrator Privileges

The user running the portal must have administrator rights on the server being administered. If the user has administrator rights to the server, but not to the SQL server, you can use PowerShell to export the SQL Scripts that need to be run to create and configure the database.

> For further details on PowerShell cmdlets refer to the Server Configuration Portal Scripting Guide.

Summary Pages

Desktop Summary

Launch the Server Configuration Portal in a web browser. The default location is
http://<servername>:7750/

The User Workspace Manager Installer automatically creates a Server Configuration Portal desktop shortcut.

The User Workspace Manager node displays the User Workspace Manager Summary page showing an overview of the Management and Personalization servers and databases represented by colored tiles.

Database Tile

Displays the total number of databases per product.

- Blue - no variances.

- Red - a database needs one of the following:

    - Upgrading - the database schema is out of date.
    - Updating - the database data is out of date.
    - Configuring - configurer details missing.

Click the red tile to go to the first database that needs attention in the tree structure. The node in the tree structure also displays in red.

Once that database variance has been fixed, go back to the User Workspace Manager Summary page, if the tile remains red there are further databases that need attention. Fix all variances until the tile turns green, which indicates all databases are up to date and configured correctly.

> ⓘ For more information, see Configuring Databases.

Servers Tile

Displays the total number of servers per product. The server tile is always blue.

Server Instances Tile

Displays the total number of instances for all servers per product. The server instance tile is always blue.

Unconfigured Server Instance Tile

Displays the number of unconfigured server instances for all servers per product.

- Green - no unconfigured instances.
- Red - one or more instances need configuring.

Click the red tile to go to the first instance that needs attention in the tree structure. The node in the tree structure also displays in red.

Once the instance has been configured, go back to the User Workspace Manager Summary page, if the tile remains red there are further instances that need configuring. Fix all variances until the tile turns green which indicates all instances are configured correctly.

> ℹ️   For more information, see [Configuring Servers](#).

Management or Personalization Summary

The Management or Personalization nodes display the product Summary page, where you can see an overview of the servers and databases for the product.

Click any tile to go to that node in the tree structure.

Database Tile

Displays the total number of databases for the product.

- Blue - no variances.
- Red - a database needs one of the following:
  - Upgrading - the database schema is out of date.
  - Updating - the database data is out of date.
  - Configuring - configurer details missing.

Click the red tile to go to the first database that needs attention in the tree structure. The node in the tree structure also displays in red.

Once that database variance has been fixed, go back to the product Summary page. If the tile remains red there are further databases that need attention. Fix all variances until the tile turns green, which indicates all database are up to date and configured correctly.

i   For more information, see [Configuring Databases](#).

Servers Tile

Displays the total number of servers for the product. The server tile is always blue.

Server Instances Tile

Displays the total number of instances for all servers for the product. The server instance tile is always blue.

Unconfigured Server Instance Tile

Displays the number of unconfigured server instances for all servers for the product.

- Green - no unconfigured instances.
- Red - one or more instances need configuring.

Click the red tile to go to the first instance that needs attention in the tree structure. The node in the tree structure also displays in red.

Once the instance has been configured, go back to the product Summary page, if the tile remains red there are further instances that need configuring. Fix all variances until the tile turns green which indicates all instances are configured correctly.

i   For more information, see [Configuring Servers](#).

Databases

Database Summary

The Database node displays the product Database Summary page, where you can see a list of all Database Connections and create or delete them.

Create a Database

You can create databases for both the Management and Personalization servers.

1. Select the required node:

   - **Management** > **Management Databases**

- **Personalization** > **Personalization Databases**

The product Database Summary page displays in the work area.

2. Click **CREATE NEW**.

The New Database Connection page displays.

3. Enter a **Friendly Name** for the database.
4. Click in the **Server Name** field to display a drop down of all known SQL servers. Select the required server.
5. Enter the **Username** and **Password** for the Configuration account.
6. To create a new database, ensure the configuration account has dbcreator server privileges and enter a unique database name.

   - To set up the schema on a new empty database, ensure the configuration account is the database owner or a member of the db_owner role, and select the database from the list.
   - To upgrade an existing database, the configuration account must have dbo privilege, and the database must be selected from the list.

   > **i**   Always backup your database before performing an upgrade.

   - To use an existing database, the configuration account must be a member of the ManagementServerAdministrator or dbo database roles.

   For further details on the Configuration and Service accounts, see [Database Accounts and Privileges](#).

7. Select the Authentication Type:

   - **Windows Authentication** - A Windows account and password must be specified to access the database.
   - **SQL Authentication** - An SQL Authentication account must be specified to access the database. Accounts, including both username and password, are created in the SQL Server itself rather than making use of existing Windows domain accounts.

8. Click **CHECK** to validate the credentials.
9. Click in **SQL Database Name** to display a dropdown of all known databases. Select the required database. Or enter a new name to create a new one.
10. Enter the **Username** and **Password** for the Service Account.

    The Web Services and Windows Services use these credentials for the database connection.

11. Select the Authentication Type:

    ○ **Windows Authentication**: A Windows username and password must be supplied each time access to the database is required.
    Note that local Windows accounts are not supported - a domain account must be used.

    ○ **SQL Authentication**: Specify an SQL Authentication account to provide access to the database.

    Accounts, including both username and password, are created in the SQL Server itself rather than making use of existing Windows domain accounts.

    If the Service account does not already exist in the SQL Server and the Configuration account has securityadmin server privileges, a new account is created.

12. Click **CHECK** to validate the credentials.

13. Click **CREATE** to start the database creation.

    Once created, the database displays in the tree structure under the relevant Database node.

Edit a Database Connection
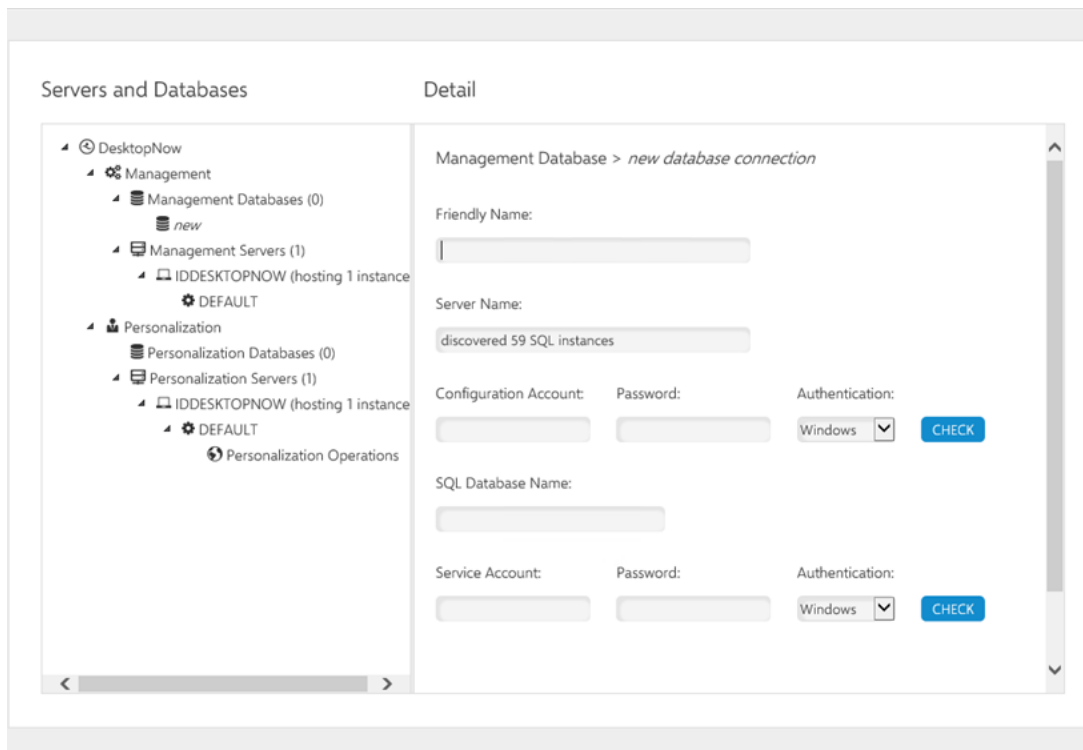
You can edit databases for both the Management and Personalization servers.

1. Select the required node:

    ○ **Management** > **Management Databases**
    ○ **Personalization** > **Personalization Databases**

    The product Database Summary page displays in the work area.

2. Select the database node that you want to change. The database Detail page displays in the work area.

3. Amend the user name for the Configuration Account.

4. Enter the **Password**.

5. Select the Authentication Type:

    ○ **Windows Authentication**: A Windows account and password must be specified to access the database.
    ○ **SQL Authentication**: An SQL Authentication account must be specified to access the database.

    Accounts, including both username and password, are created in the SQL Server itself rather than making use of existing Windows domain accounts.

    To upgrade an existing database, the configuration account must have dbo privilege, and the database must be selected from the list. Always back up your database before performing an upgrade.

    To use an existing database, the configuration account must be a member of the ManagementServerAdministrator or dbo database roles.

6. Click **CHECK** to validate the credentials.

7. Amend the username and for the Service Account.

8. Enter the **Password**.

9. Select the Authentication Type:

    ○ **Windows Authentication**: A Windows username and password must be supplied each time access to the database is required.
    ○ **SQL Authentication**: Specify an SQL Authentication account to provide access to the database.

    Accounts, including both username and password are created within the SQL Server itself rather than making use of existing Windows domain accounts.

    If the Service account does not already exist in the SQL Server and the Configuration account has securityadmin server privileges, select Create SQL Account and a new account is created.

10. Click **CHECK** to validate the credentials.

11. Click **SAVE CHANGES** to save the details.

Configure Personalization Databases with Low SQL Privileges

Personalization Server uses an SQL Server database to store personalization data. The installation procedure requires sysadmin access to the SQL Server instance in order to create and initialize the personalization database. When the person installing the Personalization Server does not have sysadmin access, scripts can be exported to enable the database to be set up. It is assumed that the SQL Server instance is on a separate machine to the Personalization Server.

> Scripts can only be exported using PowerShell and not from the Server Configuration Portal. For more information on PowerShell scripts, see the Server Configuration Portal Scripting Guide.

Servers

Server Instances

The product Servers node displays the product Server Summary page, where you can see a list of all servers for the product. Click a server name to go to that server node in the tree structure and display the product Server Summary page.

Servers and Databases | Detail

```
▲ ⊙ DesktopNow
   ▲ ⚙ Management
      ▲ ≣ Management Databases (2)
           ≣ (local)\SQLEXPRESS.ManagementServer_o0Udv4Px_1
           ≣ ManagementDatabase
      ▷ 🖵 Management Servers (1)
   ▲ 🛦 Personalization
      ▲ ≣ Personalization Databases (1)
           ≣ (local)\SQLEXPRESS.PersonalizationServer_v8PgBIM3_1
      ▷ 🖵 Personalization Servers (1)
```

Management Server Summary

PHYSICAL SERVER NAME	INSTANCES HOSTED	LOCAL / REMOTE
DESKTOPNOWSUITE	1	Local

The Server node in the tree structure displays the Server Summary page in the work area where you can see a list of all instances for the selected server:

Servers and Databases | Detail

```
▲ ⊙ DesktopNow
   ▲ ⚙ Management
      ▲ ≣ Management Databases (2)
           ≣ (local)\SQLEXPRESS.ManagementServer_o0Udv4Px_1
           ≣ ManagementDatabase
      ▲ 🖵 Management Servers (1)
         ▷ 🖳 DESKTOPNOWSUITE (hosting 1 instance)
   ▲ 🛦 Personalization
      ▲ ≣ Personalization Databases (1)
           ≣ (local)\SQLEXPRESS.PersonalizationServer_v8PgBIM3_1
      ▷ 🖵 Personalization Servers (1)
```

DESKTOPNOWSUITE (Local) Summary

INSTANCE NAME	STATUS	LOGGING	VARIANCES
DEFAULT	Started	Disabled	None Detected

The status of each instance is listed together with whether logging is set and if there are any variances.

If there are no variances 'None Detected' displays. If there are variances the word 'Detected', displays in the Variances column. Click **RECHECK** to display the list of variances.

DESKTOPNOWSUITE > Management > Variances

	ISSUE
☐	WebSite\ SiteRunning: The Web Site must be running to be operable (the World Wide Web Publishing Service may not be running. This must be started manually to fix this variance)
☐	WebSite\ WebDirectories:ManagementServer\ WebDirectories:ManagementServer/Deployment\ DirectoryUserAccountFileAccess: null
☐	WebSite\ WebDirectories:ManagementServer\ WebDirectories:ManagementServer/Deployment\ WebDirectories:ManagementServer/Deployment/Packages\ DirectoryUserAccountFileAccess: null
☐	WebSite\ WebDirectories:ManagementServer\ WebDirectories:ManagementServer/Deployment\ WebDirectories:ManagementServer/Deployment/Events\ DirectoryUserAccountFileAccess: null

FIX ALL   FIX SELECTED   REFRESH VIEW   DONE

You can select to **FIX ALL** or **FIX SELECTED** variances. Click **REFRESH VIEW** to update the list and **DONE** when finished.

You can click any Instance name to display the instance configuration or Detail page. The appearance of the page depends on whether the instance has been configured:

- For instances that are not configured, the Detail page displays the options to configure a new instance.
- For instances that have been configured, the Detail page displays the configured settings and options such status, whether logging is enabled, and the option to recheck variances.

Configure a New Instance
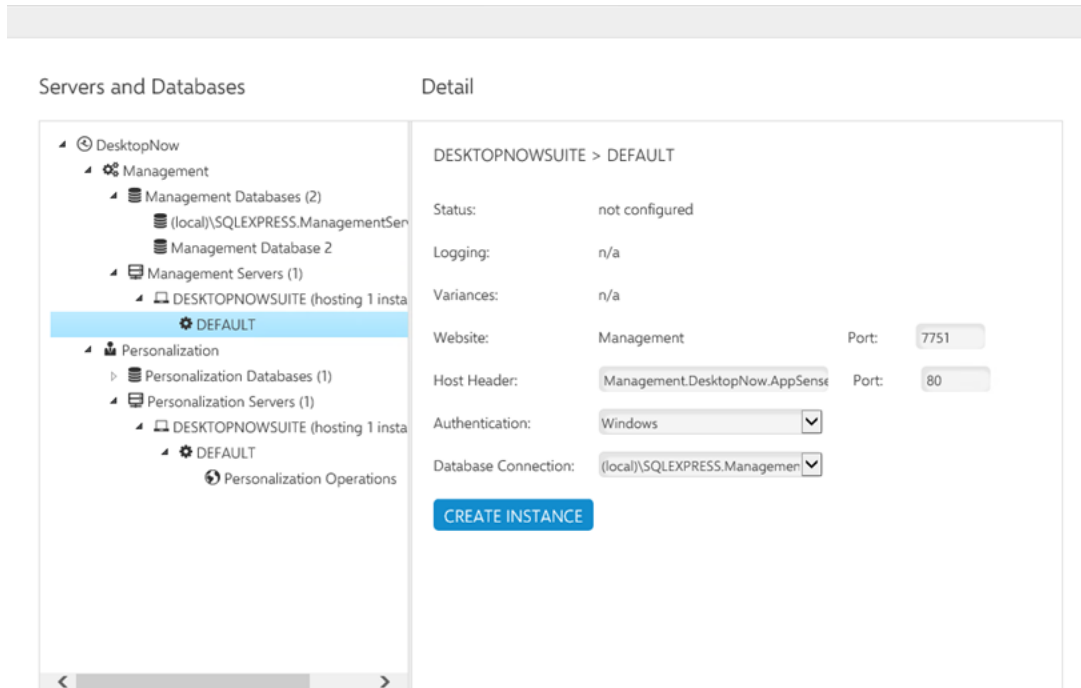
1. Click a server instance to display the Configure Instance page.

   The Website name is either Management or Personalization.

2. You can edit the website **Port** number, the default port numbers are dependent on which installation method was selected in the User Workspace Manager Installer.

   If you installed via Evaluation mode the default ports are:

   - Management - 80
   - Personalization - 80
   - Configuration - 7750

   If you installed via Advanced mode the default ports are:

   - Management - 7751
   - Personalization - 7771
   - Configuration - 7750

3. Edit the **Host Header** and **Port** number or leave as the default.

   The host header should contain your domain name only if you intend to add a CNAME record to DNS.

4. Select the Authentication type:

   - **Windows authentication** - Deployment Agents must authenticate with the server using Windows Authentication. This increases the security of the server, ensuring that only computers in the domain can access the server.
   - **Anonymous authentication** - Deployment Agents can access the server unchallenged.

5. Select the **Database Connection** from the drop-down list. If the database does not exist, a **CREATE NEW** option displays. Click it to create a new database connection.

6. Click **CREATE INSTANCE** to start the configuration process.

   Once the instance is configured, any red tiles relating to this instance are updated and the settings can be specified.

Specify Settings for an Instance

1. Click a configured server instance in the tree structure.

   The Instance Settings page displays.

2. Set the instance Status.

   The instance can be **Online** or **Offline**.

3. Set Logging to **Enabled** or **Disabled**.

   Logging is disabled by default.

   Management Server log files are written to %ProgramData%\AppSense\Management_
   [InstanceName]

   > ℹ️ If Logging is enabled you must restart the Management Server Services for the change
   > to take effect.

   Personalization Server log files are written to
   %ProgramData%\AppSense\PersonalizationServer\Personalization_[InstanceName]

4. Click **RECHECK** to run a variance check.

   If the instance has variances that need fixing, the text 'Detected' displays. Click **Detected** to
   display the variances.

5. If there are no variances the text 'None Detected' displays.

6. Click the URL to go to the website.

7. Set the Authentication type:

   - **Windows authentication**:Deployment Agents must authenticate with the server using
     Windows Authentication. This increases the security of the server, ensuring only
     computers in the domain can access the server.
   - **Anonymous authentication**:Deployment Agents can access the server unchallenged.

8. To set the **Database Connection**, select from the dropdown list.

9. Click **UPDATE** to save any changes.

Encryption

If multiple Management Servers are being used in a failover scenario, the Encryption node is used to
share the encryption key between each Management Server. Any encryption that is required uses the
Microsoft Windows Cryptographic Service Provider. It is also used to back up the key securely in the
database.

If failover servers are being used, all the servers must use the same public-private key pair.

First, a transfer key needs to be made available by the master server. The transfer key contains both the public and private keys. Click **STORE** to save the key in the database in a password protected format.

Once the password has been stored, the transfer key is shown as present and can now be retrieved by other servers to create the correct public-private key pair.

Click **RETRIEVE** for each of your servers and re-enter the password to decrypt the transfer key.

Personalization Operations

Personalization Operations is an Environment Manager utility that provides management of Personalization data via a web console. Depending on their role, users can manage backups and current settings for either single users or multiple users at a time. They can also search for and delete audit logs, and view the migration status of Personalization Groups.

Personalization Operations web console is accessed via the URLs displayed in the **User Workspace Manager** > **Personalization** > **Personalization Servers** > [Server] > [Instance] Detail page.

Personalization Operations is configured in the **Personalization Operations** node for a Personalization server instance.

To navigate to the Personalization Operations node in the tree structure select, **User Workspace Manager** > **Personalization** > **Personalization Servers** > [Server] > [Instance] > **Personalization Operations**. The Settings page displays.

You can update the settings so that the website is Online or Offline, enable or disable logging, and specify the authentication type.

If you are the administrator configuring the Personalization server and database, you have administrator rights in the Personalization Operations utility. Before other users can use Personalization Operations, they must be added as Personalization Operations users and assigned roles and Personalization Groups. You can do this in the User Roles page of the Personalization Operations web console.

> ⓘ For more information, see the [Managing User Roles](#) topic in the Personalizations Operations Help.

Export Scripts

User Workspace Manager Servers uses an SQL Server database to store data. The installation procedure requires sysadmin access to the SQL Server instance in order to create and initialize the database. When the person installing the Server does not have sysadmin access, scripts can be exported to enable the database to be set up. It is assumed that the SQL Server instance is on a separate machine to the User Workspace Manager Server.

Export the Scripts to Send to the SQL Administrator

1. Open an elevated PowerShell window.

2. Run the commands to select an instance and export the scripts, for example:

   ○ Import-ApsInstanceModule -Product Management

   ○ Export-ApsDatabaseScript -all -path c:\scripts

3. Send the exported scripts to the SQL Administrator.

Actions for SQL Administrator to Perform

Using SQL Server Management Studio the following steps must be carried out by the database administrator under sysadmin privilege.

1. In SQL Server Management Studio, open the Create Database script, change the database name in the SET line.

2. Save and Execute.

3. Open the Create Schema script and ensure the newly created database is selected in the drop-down list before executing.

4. Open and run the Create Login script as follows:

- Uncomment the declarations at the top of the script.

- To create a Configuration, or administration, account using a SQL account ("AmcAdmin" in this example) set the default values to:

@userName = 'AmcAdmin'

@password = 'Password123'

@isSql2005 = 1

@enabled = 1

@forcePswdPolicy = 1

@forcePswdExpire = 0

@mustChange = 0;

- To create a Service account using a Windows authenticated account ("DOMAIN\admin" in this example) set the default values to:

@userName = 'DOMAIN\admin'

@password = 'Password123'

@isWindowsAuth = 1

@issql2005 = 1

@enabled = 1

@forcePswdPolicy = 1

@forcePswdExpire = 0

@mustChange - 0;

Run the script once for the Configurer account and once for the Service account. Set @isWindowsAuth to 1 for a Windows account, or 0 for SQL authentication.

5. Give the Configuration account name db_owner and ManagementServerAdministrator rights on the database.

6. Give the Service account ManagementServerService rights on the database.

ℹ Refer to the <u>Server Configuration Portal Scripting Guide</u> for further PowerShell guidance.

Upgrade

In-place Server Upgrades

An in-place upgrade allows for the minimum service interruption, requires no change to the client configuration for either the Management Server or the Personalization Server, and allows for staged upgrades of either the Deployment Agent or the EM Agent.

In-place upgrades of either the Management Server or the Personalization Server install into the default website or whichever website in which the Management Server or the Personalization Server is installed.

> **Caution:** Personalization Server requires access to the root of the website in which it is installed. If installed on the default website, it overwrites any existing applications that are currently using the root of the default website.

If you already have an application installed into the root of the default website, you should attempt an in-place upgrade of the Management Server only. The Personalization Server should be installed as a new installation into the Personalization website; refer to the New Server Clean Install Upgrades section.

New Server/Clean Install Upgrades

Uninstalling the existing User Workspace Manager servers and reinstalling or provisioning a new server provides an alternative upgrade strategy, but does include caveats. You can use a number of methods to upgrade the Management server or Personalization server. However, you must plan carefully because you may need to update the connection URLs for clients and console. The most common options following a new installation and database upgrade are:

- Remove the host header and make either the Management Server or the Personalization Server the default website.

    For more information, see Alternative IIS Configuration.

- Update DNS records, Load Balancer, or Virtual IP to direct HTTP traffic to the new website.

    For information on DNS and the use of host headers, see User Workspace Manager Server Component Websites.

Management Server only

- Having upgraded the Management Server and database, deploy the Deployment Agent. This ensures the Deployment Agent URL is updated. This method is particularly advantageous if you also intend to deploy a new license.

- For staged upgrades of the Management Center, add the new Management Server URL to the Failover Server List prior to bringing the Management Server online. The Deployment Agent continues to communicate with the previous version of the Management Server until it is taken offline and the new version of Management Server comes online.

Personalization Server only

- Deploy a new Environment Manager Policy file containing the updated Personalization Server URL.
- For staged upgrades of Environment Manager Personalization, it is recommended to make use of the Virtual Hosts feature of Personalization along with updated DNS records prior to any upgrade.

Replication Servers

Upgrades cannot be performed while replication is enabled. Replication must be disabled on the master and subscribers prior to upgrading all servers to the same version.

Upgrading to a newer release includes a database schema change. Therefore all personalization servers are locked out of the database until the upgrade process is complete.

Load Balanced Servers

When servers are load balanced and operate as individual entities using a common IP address, it is recommended that all servers are taken offline before upgrading. When the upgrade is complete, servers can be brought back online and added back to the network load-balanced configuration.

When upgrading servers, the first server to be upgraded also upgrades the database. This ensures that the server and database versions match after the schema change. During an update of a server and the database, all other servers are locked out.

Testing Server Upgrades

On completion of the Personalization Server upgrade, test that the server is functioning by visiting the PersonalizationServer website at:

http://localhost/PersonalizationServer/status.aspx

or

https://localhost/PersonalizationServer/status.aspx

If you are running the test from a remote location to the Personalization Server, replace *localhost* with the server name and port number. After approximately 30 seconds, a page displays to confirm successful connection. Check the details are correct and the tests are successful. When connection is complete, the Personalization Server can accept requests from managed endpoint devices.

Upgrading Databases

If the database is out of date, the Database Updates Required page automatically displays when you open the Server Configuration Portal.

1. Select the databases you want to upgrade and click **UPDATE SELECTED**.

   Alternatively, select **UPDATE ALL** to upgrade all databases.

2. Once the Status changes to Upgrade Successful, click **CLOSE** to display the User Workspace Manager Summary page.

Technical Reference

Services

There are four associated services with the Management Server:

- Ivanti Alerts Service - responsible for creating alerts based on events for the Management Server, and dispatches associated actions.
- Ivanti Events Dispatcher Service - responsible for monitoring for new event files being uploaded and adds the events to the Management Server database.
- Ivanti Scheduler Service - responsible for managing all scheduled tasks associated with the Management Server. This includes discovery and offline machine detection.
- Ivanti Deployment Service - responsible for managing the installation of the Deployment Agent when chosen by the user from the Management Console.

To generate diagnostic logs for Management Server Services set Logging to Enabled on the Instance Detail page in the Server Configuration portal. The log files are stored in %ProgramData%\AppSense\Management_[Instancename]

There is one associated service with the Personalization Server:

AppSense Personalization Background Service - performs batch operations requested by the Personalization Operations console and does daily archiving and cleanup on the database.

Websites

Management Server

The ManagementServer root web directory hosts the Downloads web page for downloading the Management Console, Deployment Agent, User Workspace Manager products, and documentation.

A diagnostics log can also be generated from this page. It is stored at %ProgramData%\AppSense\Management_[Instancename] by default.

ManagementServer/Deployment

The ManagementServer/Deployment web directory provides the Management Server web services that the Deployment Agent uses to access the Management Center database. These hosted web services are:

- Polling - Managed endpoints receive settings such as poll periods and installation schedule during a poll.
- Prerequisite checking & installation - Managed endpoints download agents, configurations, and prerequisites using BITS.
- Event Collection - Managed endpoints upload the majority of events using BITS.
- Server Diagnostics - Managed endpoints send high priority events.

A diagnostics log, DeploymentDirectory.log, can also be generated from this page which is stored at %ProgramData%\AppSense\Management_[Instancename]

ManagementServer/Data Access

The ManagementServer/DataAccess web directory provides the interface to the Data Access Services. All communication from the Management Console comes here.

ManagementServer/PackageManagement

The ManagementServer/PackageManagement web directory provides an interface to the Package Management Services. All communication from the Application Control, Environment Manager and Performance Manager consoles come here.

Securing Communications Using SSL

You can optionally configure the Management Server website to support Secure Socket Layers (SSL) to provide secure communications using Active Directory.

SSL provides confidentiality and integrity of communications to ensure sensitive data is accessible only by authorized users, including:

- Event data
- Agents and agent configuration data

> **ℹ** If you are setting up SSL certificates on web servers using other supported operating systems and other versions of Microsoft SQL Server, see the following for further information: http://msdn.microsoft.com/library/

Setup SSL on IIS

This section provides information about setting up the website for SSL by creating a self-signed certificate.

> **ℹ** Other types of certificate issued by a trusted Certification Authority are also supported.

1. In **Start** > **All Programs** > **Administrative Tools** > **Internet Information Services (IIS) Manager**, select the [**ServerName**] node and in the IIS section click **Server Certificates**.

2. Ensure a suitable certificate is listed. If not, create or import a certificate using the options in the Actions panel.

3. Select the **Website** for the product and click **Edit Bindings** in the shortcut menu.

4. Click **Add** and in the **Type** drop-down list select **HTTPS**.

5. In the **SSL Certificate** drop-down list, select the certificate.

6. Click **OK** and **Close**.

Deployment Agent

Deployment Agent on Managed Computers

The Deployment Agent is installed on managed computers to manage communications between the product agent and the Management Center.

The Deployment Agent communicates with the Management Server to manage the download and installation of agents, configurations, and software package updates, and also sends event data generated by the product agents to the Management Server.

Install the Deployment Agent

Install the Deployment Agent on all computers to be managed by the Management Center. The Deployment Agent can be distributed using the integrated **Install Deployment Agent** functionality in the Management Console, by downloading the ClientCommunicationsAgent32/64.msi package from the Management Server website or by third-party deployment mechanisms.

It is recommended you set up Membership Rules and Deployment Groups in the Management Console before installing the Deployment Agent. Refer to the *Management Center Help* for further details.

The **Install Deployment Agent** functionality in the Management Console can be run in small and medium scale enterprise environments to deploy the Deployment Agent to multiple computers, or to repair or modify the URL path for currently deployed Deployment Agents to change the http or https prefix and port number.

IT administrators in organizations often create master images that include the operating system with all the required software and updates required for a new computer, as a labor-saving approach to setting up multiple computers. It is recommended to install the Deployment Agent on a master image prior to rolling out to computers in your organization.

Integrated Install Deployment Agent Functionality

The Management Console provides an Install Deployment Agent function that allows you to deploy the Deployment Agent to multiple computers that match the Management Center Deployment Group and Membership Rules. The agent can be deployed on a Microsoft Active Directory network in small or medium scale environments.

Workflow

Install Deployment Agent functionality detects the Management Center deployment groups and uses group membership rules to provide the list of computers to which the Deployment Agent can be deployed. Active Directory is queried for active directory types for membership rules by computer, groups, and containers. Alternatively, you can manually include or exclude computers from the list by NetBIOS Name.

The Deployment Agent can only deploy the Deployment Agent to computers that are members of Deployment Groups configured in the Management Center console.

The software requirements for the target client computers are detected and the latest 32-bit or 64-bit version of the Deployment Agent installation package is downloaded.

Packages are distributed to the target computers and installed silently with the correct URL of the Management Server.

The basic steps required to install the Deployment Agent are as follows:

Step 1 Set Up Client Access Credentials

1. Navigate to **Home** > **Global Settings** > **Access Credentials tab**.

2. Enter the user credentials (user name and password) for an account on the computer on which the Deployment Agent is being installed. The account must have local administrator privileges.

   You can add multiple accounts. They will be attempted in the order in which you list them.

   You will not be able to install the Deployment Agent on any endpoint using the integrated Install Deployment Agent functionality if the Access Credentials have not been set up.

Step 2 Configure Settings for the Deployment Group

1. Navigate to **Home** > **Deployment Groups**.
2. Create a **New Deployment Group**.
3. Select **Settings** for the new deployment group. The Settings work area opens on the General tab.
4. In the Server Polling and Downloads section, specify how often the client computers check for and download new agents or configurations. Use the slider to set the poll variance. The poll variance reduces the impact of multiple clients polling and downloading at the same time.
5. In the Event Data Uploads section, specify how often event data is uploaded to the Management Server. Use the slider to set the upload variance. The upload variance reduces the impact of multiple clients uploading data at the same time.
6. In the Deployment Agent Permissions, specify whether the deployment agent can self-register, unregister, or make agent and configuration updates outside of the set installation schedule.
7. Select the **Installation** tab.

8. Set up the agent and configuration installation schedules for the deployment group.

Step 3 Create Membership Rules for the Deployment Group

1. Navigate to**Home** > **Deployment Groups** > **[Deployment Group]** > **Configure Membership Rules** button.

   Each Deployment Group has a one to one relationship with a set of Membership Rules. The Membership Rules act like a filter to discover computers in Active Directory.

2. In the Actions panel, select **Edit Conditions** to add a new condition based on NetBIOS Name or Active Directory.

3. Select **Submit** from the Membership Rules work area.

4. In the Actions panel, select **Discover**.

The discovered computers that match the Membership Rules are listed in the relevant **Deployment Group** > **Computers** node.

> For the computers discovered by Membership Rules, the Computer Status should initially display: *No Deployment Agent deployed*.

Step 4 Install Deployment Agent

1. Navigate to **Home** > **Deployment Groups** > **[Deployment Group]** > **Computers**

2. Select the computer or computers on which you want to install the Deployment Agent.

3. In the Actions panel, select **Install Deployment Agent**.

The Client Access Log provides details on the installation progress. The Deployed (%) column indicates the percentage of the package deployed.

Install Deployment Agent Manually

You can manually install the Deployment Agent on a managed computer by downloading and running the ClientCommunicationsAgent32/64 installation package on a client computer.

> All prerequisites should be installed before manually installing the Deployment Agent. See [Prerequisites](#) for further details.

1. Launch a web browser and navigate to the Management Server website at the following address:

   https://[computer name]/ManagementServer

   If you have not configured SSL communications, use the HTTP prefix for the Management Server website: http://<computer name>/ManagementServer/

The Management Center download page displays.

> ℹ The Downloads page is best viewed in Internet Explorer 8 or higher.

2. Make a note of the Management Server URL displayed in the download page.

   You can also download: Management Console, EULA, Release Notes, Application Control, Environment Manager and Performance Manager Consoles, and Prerequisites Software.

3. Download and run the appropriate 32-bit or 64-bit ClientCommunicationsAgent installation MSI package.

4. In the Deployment Agent installation Welcome screen, click **Next**.

5. In the License Agreement screen, read the license agreement. If you accept the terms, select and click **Next**.

6. In the Installation Directory screen, leave the default installation directory unchanged, and click **Next**.

   The Settings screen displays.

   Enter the Management Server computer name, http://<Computer Name>/
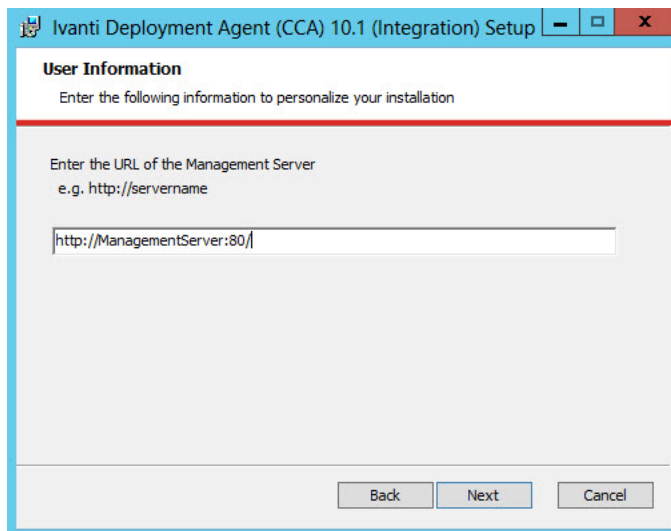
7. If you have configured SSL communications, use the HTTPS prefix for the Management Server website: https://<computer name>/

8. Click **Next** to proceed.

9. When the installation is complete, click **Finish** to exit the installation wizard.

You have now successfully installed the Deployment Agent. The host computer is able to connect to your Management Server, ready to download product agents, license, and configuration software packages according to the settings configured for the deployment group to which the current computer belongs.

Install Deployment Agent in Silent Mode

You can install the Deployment Agent silently via a third party deployment mechanism or from a command prompt.

> Use the HTTPS prefix for the Management Server website only if you intend to install an SSL certificate on the server computer and managed computers are located in the same Active Directory domain as the Management Server.

msiexec.exe /qn /i "<MSI file path>\CommunicationsAgent.msi" WEB_SITE="https://<Management Server Name>/" GROUP_NAME="<DeploymentGroup>"

- `/i` - Install
- `/qn` - Quiet mode install without the user interface.
- `WEB_SITE` - Enter the Management Server website address using the name of the host computer.
- `GROUP_NAME` (optional) - Enter the Deployment Group name to which the Deployment Agent should register. The Deployment Agent can only register with a group that is set up to allow the agent to self-register. Otherwise, the agent attempts to register with the Management Server deployment groups according to group membership precedence:
    - GROUP_NAME self-register
    - Deployment Groups - membership rules
    - (Default)Group – if no match is found

Allow self-registration with this group

You can set up a deployment group to allow the Deployment Agent to self-register with a specific group when installed using the command line.

This option is disabled by default but provides an alternative method for installing the deployment agent on managed endpoints to register with a specific Deployment Group on the Management Center rather than predefining the group membership in the Management Console.

Enable **Allow self-registration** in the Deployment Agent Permissions section on the General tab in the Settings work area for the relevant Deployment Group.

Service Packs

Patching

User Workspace Manager products can be patched using a Windows Installer patch. A patch is an MSP file that, when installed, updates files and registry keys on an existing installed product. Installing an MSP can reduce system downtime because reboots are not always required. User Workspace Manager product patching gives all of the usual benefits associated with Windows Installer Patching, including ease of deployment and the ability to roll back to an earlier version. Patches include the following:

- **Public Hotfix** - Issued publicly on to address a widely reported issue and should only be installed to address the specific problem. Public Hotfixes are cumulative in that they contain all previous hotfixes. Public Hotfixes are distributed as an MSP.

- **Service Pack** - Contains all of the fixes from the last Private or Public Hotfix and any Service Packs, plus any fixes that have been found for which a Private or Public Hotfix was not issued. Service Packs are cumulative in that they contain all previous Service Packs. Service Packs are distributed as an MSP.

  If a Service Pack is part of the product release media, the installer automatically installs them. Service Packs can also be installed or deployed using the same technology and techniques used when installing MSIs. Both Microsoft System Center and the Management Center can deploy MSPs. If neither of these products are available, service packs can be installed using the command line interface.

Installation Order and Dependencies

It is recommended that all components of a service pack are installed and that the PersonalizationServerXX.MSP is installed first. All other components have no required install order.

To view previously installed patches, navigate to **Control Panel** > **Programs** > **Programs and Features** > **Installed Updates**.

Manage Patches with the Management Center

To install a patch using Management Center you must first upload the MSP and then assign it to a deployment group for deployment to the endpoints.

Upload an MSP

1. Open the Management Console and in the navigation pane select **Package Library**.

2. Click the required product, for example Environment Manager.

   The package library for the selected product displays in the work area.

3. From the Actions panel, select **Add Package**.

   The Browse for Package dialog displays.

4. Locate the required MSP file and click **Open**.

   > ⓘ If uploading a Service Pack the base MSI package must have been uploaded previously. If uploading a Hotfix the base MSI package must have been uploaded. Additionally, if the Hotfix is applicable to a Service Pack, the Service Pack must also have been uploaded.

   The Package Upload wizard displays.

5. Check the details of the selected package and optionally enter a description.

6. Click **Next** to start the upload.

7. When the upload has competed successfully, click **Finish**.

The MSP can now be seen in the package library.

Deploy an MSP to an Endpoint

1. In the Management Console, click **Home**. In the navigation pane, expand **Deployment Groups**, and then expand the group you want to deploy to and select **Packages**.

   The Packages work area displays a list of User Workspace Manager products and the assigned packages.

2. Highlight the required product package and from the Actions panel select **Change Agent Version**.

   A dialog to change the packages used by the group displays.

3. Select the required patch package. For example, 8.9 SP2 HF3.

   The Management Center ensures that any dependencies for the selected patch are deployed to the endpoint.

   > ⓘ Deployment Agent 8.6 may be required to support deployment for the selected patch, if this is the case a warning message displays at the top of the work area.

4. Click **Finish**.

   The patch can now be seen in the Assigned Packages list.

5. Once all changes have been made, from the bottom of the work area, click **Review and Submit**.

   The Submit Changes For [Group Name] dialog displays.

6. Review the changes to be made to the deployment groups and click **Submit**.

   A warning may display informing you that changes to an agent can cause reboots at the times defined by the installation schedule. If the warning message displays, click **Yes** to assign the changes. Alternatively, click **Submit** to assign the changes.

The patch is deployed in accordance to the Deployment Group Installation Schedule.

Uninstall an MSP

To uninstall a Service Pack or Hotfix using the Management Center:

1. In the Management Console, click **Home**. In the navigation pane, expand **Deployment Groups**, and then expand the group you want to deploy to and select **Packages**.

   The Packages work area displays a list of all the User Workspace Manager products and their associated packages.

2. Highlight the required Service Pack or Hotfix and click **Change Agent Version** from the Actions menu.

   The Change the packages used by this group dialog displays.

3. Select an alternative version and click **Finish**.

4. On the Assigned Packages work area click **Review and Submit**.

   The Submit Changes dialog displays.

5. Check the details are correct and click **Submit**.

The uninstallation takes place in-line with the Deployment Group Installation Schedule.

Manage Patches Using the Command Line

User Workspace Manager patches can be installed from the command line as well as from the Management Center.

It is recommended that logging is switched on when using the following commands. To enable Logging add `/l*vx Patch.log` immediately after the `/i` or `/p`. For example: `msiexec.exe /i Agent.msi /l*vx Patch.log`

Command Line Install

To install or upgrade an MSI:

```
msiexec.exe /i Agent.msi
```

To silently install or upgrade an MSI:

```
msiexec.exe /i Agent.msi /qn
```

To install an MSP:

```
msiexec.exe /p Agent.msp
```

To install an MSI and MSP in a single operation:

```
msiexec.exe /i Agent.msi PATCH=C:\FullPath\Agent.msp
```

Example:

```
msiexec.exe /p EnvironmentManagerAgent64.msp
```

installs any files that have been amended as part of the patch for just Environment Manager 64-bit agent.

The following command installs the base version of the Environment Manager Agent (MSI) and the Environment Manager patch file (MSP) simultaneously:

```
msiexec.exe /i EnvironmentManagerAgent64.msi PATCH=c:\fullpath\EnvironmentManagerAgent64.msp
```

ⓘ   A base version must be installed before the patch file can be applied.

If the patch file contains driver or hook files that are currently in use on the machine the patch is being applied to, you are informed that a reboot is required. If you chose to continue, the system is restarted when the patch has been applied.

Command Line Uninstall

To uninstall an MSI and all associated MSP files:

```
msiexec.exe /x Agent.msi
```

To remove an MSP:

```
msiexec.exe /i Agent.msi MSIPATCHREMOVE=C:\FullPath\Agent.msp
```

Roll Back Service Packs

There are two ways to roll back, or uninstall, User Workspace Manager Service Packs:

- Using the Windows Control Panel
- Using Management Center

If a service pack is uninstalled, the installation reverts to the previous latest build, whether a service pack or base version. All agent and console service pack components except the Personalization Server component patch file (PersonalizationServerXX.msp) can be uninstalled

Rolling Back Service Packs Using Windows Control Panel

The procedure used to roll back service packs varies depending on the Operating System:

For Windows 7

Navigate to **Control Panel** > **Programs** > **Programs and Features** > **Installed Updates**.

Highlight the selected patch and click **Uninstall**.

Rolling Back Service Packs Using Management Center

1. In the Management Center console, select **Home**.

2. Expand Deployment Groups, highlight the Deployment Group and select **Packages**.

    The Assigned Packages work area displays a list of all the User Workspace Manager products and their associated packages.

3. Highlight the required Service Pack or Hotfix and click **Change Agent Version** from the Actions menu.

    The Change the packages used by this group dialog displays.

4. Select an alternative version and click **Finish**.

5. On the Assigned Packages work area click **Review and Submit**.

    The Submit Changes dialog displays.

6. Check the details are correct and click **Submit**.

The uninstallation takes place in accordance with the Deployment Group Installation Schedule.

Product Upgrades

Prepare to Upgrade

Existing User Workspace Manager software packages upgrade automatically during the installation process, including database schemas, agents, and configurations. Prior to an upgrade, it is recommended that you do the following:

- Back up all databases
- Save all product configuration packages from the console as MSI files.
- If necessary, save all earlier versions of the product agent software that you would like to maintain.
- You must upgrade the Management Server before you upgrade the Deployment Agent. The Management Server must be of the same version number or later than the installed Deployment Agent.
- Disconnect all users from the Personalization Servers.
- Take all Personalization Servers offline until the upgrade is complete.
- Run the whole suite installer to upgrade components.

Upgrade with the User Workspace Manager Installer

1. Run the Installer by executing the **setup.exe** file in the installation media.

   The Welcome screen displays.
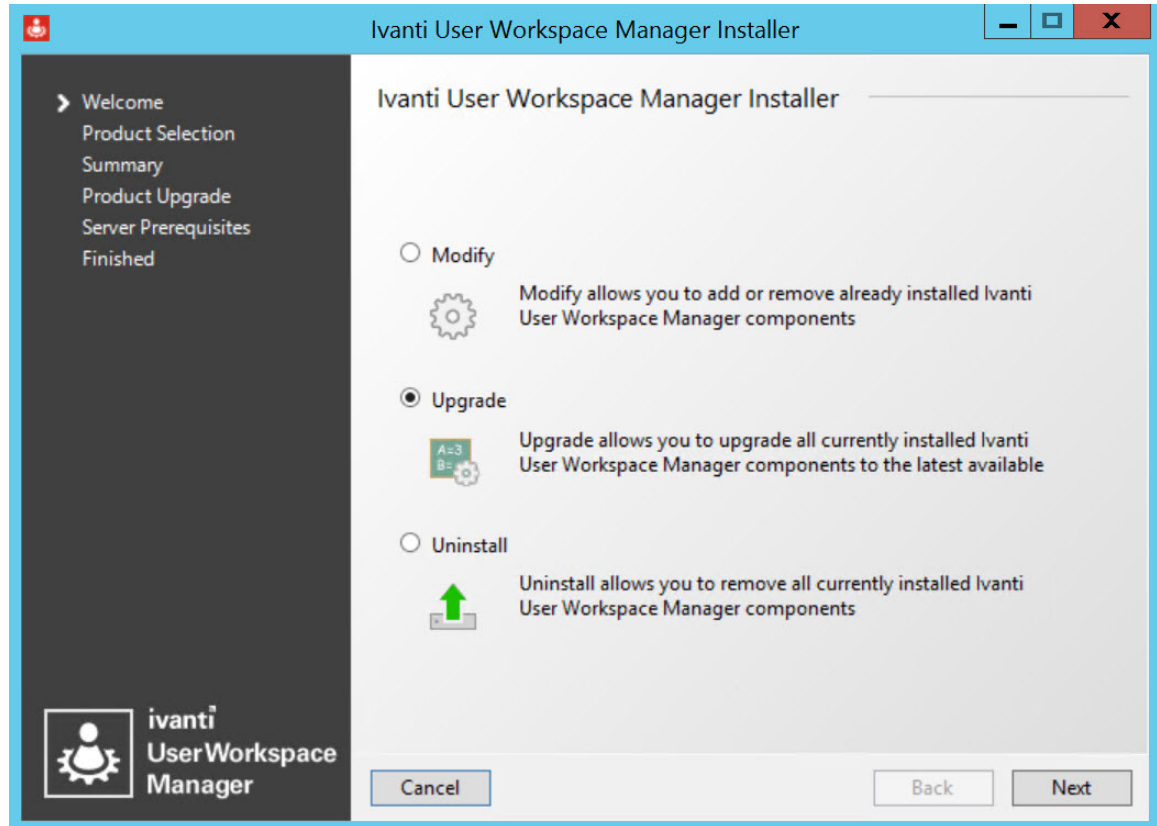
2. Select **Upgrade** and click **Next**.

    The Product Selection screen displays.
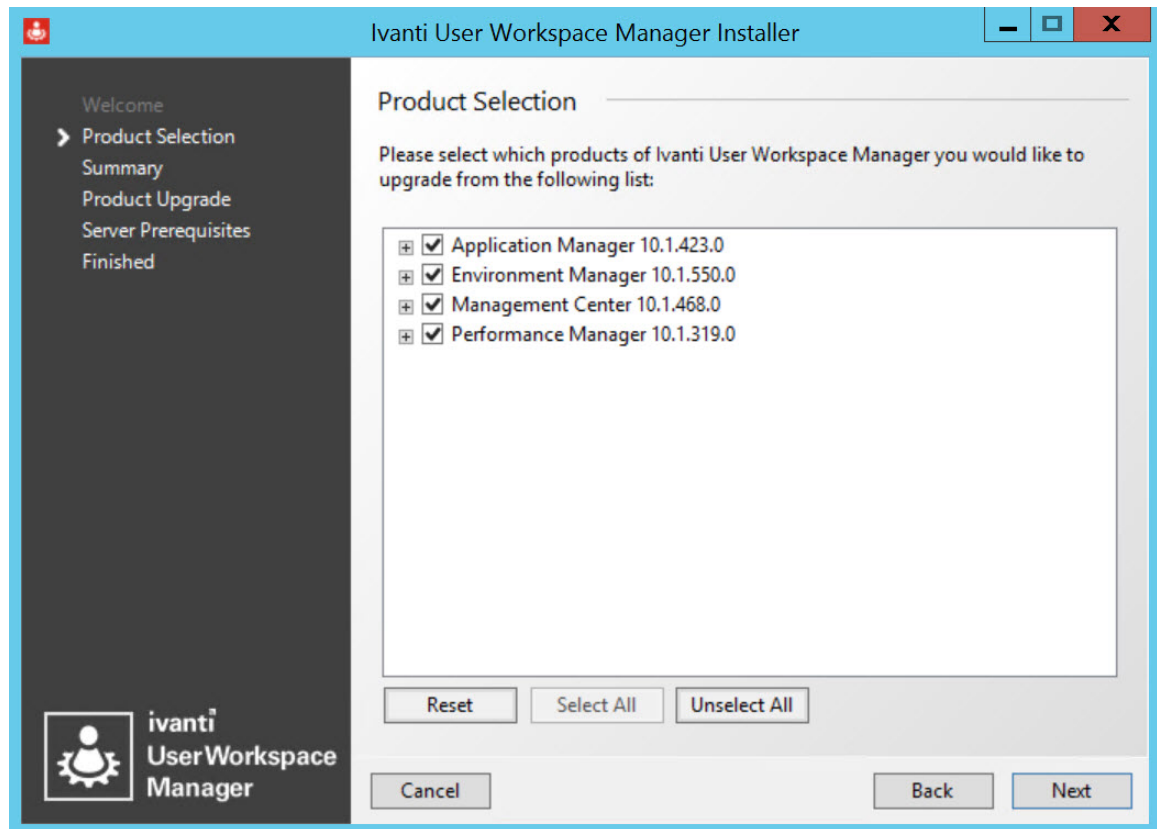
Only the products currently installed display for you to select to upgrade. If you have server products installed, each instance is listed. Select the ones to upgrade.

3. Select the product components that you want to upgrade and click **Next**.

    The Summary screen displays listing the products to be upgraded.

4. Click **Next**.

    The next step depends on whether all prerequisites are installed:

    ○ If any prerequisites are missing the Prerequisites screen displays.

       Select **InstallAll** to install all missing prerequisites. Once all components are installed the upgrade process starts immediately.

    ○ If there are no missing prerequisites the upgrade process starts.

5. If there are missing server prerequisites the Server Prerequisites screen displays, select **Install All** to install all missing components.

6. On completion the Upgrade Complete screen displays.
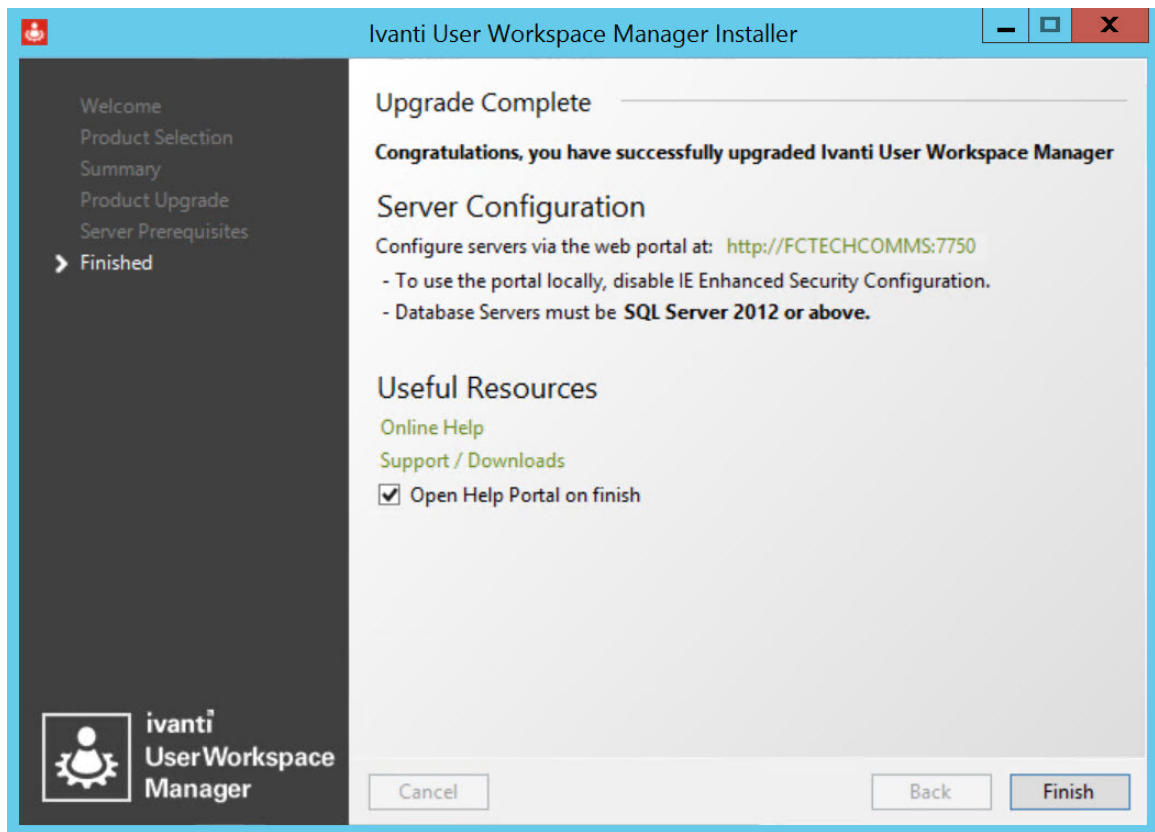
7. Click **Finish** to exit the Installer and open the Help Portal.

Upgrade Application Control

If you are upgrading configurations used in previous versions of Application Control, the introduction of the Process Rules and Group Management functionality may render the following parts of the configuration redundant:

- Trusted Applications
- Signature Groups
- Network Connection Groups
- URL Redirection and Custom Rules

Upgrades and Process Rules

If the Application Control configuration contains Trusted Application rules, the upgrade will preserve the Trusted Applications feature's behavior although some functionality regarding the three Trusted Applications options may be lost.

The table below shows how the various Trusted Application states will be converted to Process rules during a configuration upgrade.

Trusted Application State	Process Rules
Off	No Process rules added.
Disable Trusted Applications Checking	No Process rules added.
Only when blocked by Trusted Ownership Always	For each Trusted Application defined: A new Process rule is created with the name *Upgraded Trusted Application Rule (*)*.Where * represents a number automatically incremented from 1 to the number of Trusted Application rules present in the configuration being upgraded. A new Process Identifier is added to the newly created Process rule. If the Trusted Application rule was defined using a full file path then the process identifier list has one file name entry with the exact same text. If the Trusted Application rule was defined using a digital signature then the process identifier has one digital signature entry with the same digital signature. Any file name information is preserved. For each of the trusted content entries for the Trusted Application rule, a new Allowed Item is added. The Trusted Ownership setting is set to **Off**, for all added entries.

Upgrades and Group Management

If the Application Control configuration contains Signature Groups and Network Connection Groups, the upgrade directly converts them to Group Management and renames them Groups. The name of the Signature or Network Connection Group remains the same and the contents of the Signature or Network Connection Group remain the same.

To avoid any problems that may be encountered if the upgrade produces any duplicate names, each upgraded Group will be suffixed with its origin and that it was an upgrade.

Example

A Signature Group called *A*, becomes a Group called *A - Upgraded Signature Group*.

A Network Connection Group called *B*, becomes a Group called *B - Upgraded Network Connection Group*.

URL Redirection and Custom Rules

Custom rules and URL Redirection in Application Manager 10.0 and later (Application Control from 10.1 FR1 and later) differ considerably from versions 8.8 and 8.9. You can upgrade version 8.8 and 8.9 configurations by opening them in a version 10.0 or later console and saving them. This changes the configurations as follows:

- Custom rules are recreated using the new version 10.n conditions, matching the behavior of the earlier version rules.
- URL Redirections are converted to Custom rules that contain:
- Matching conditions for connection types, IP addresses, and port numbers.
  - Browser Control items for the sensitive URLs.
  - If you don't upgrade the configuration, the Application Control Agent version 10.n still reads the configuration, but the URL Redirection and Custom rules are ignored. The rest of the application still applies.

Upgrade Environment Manager

Environment Manager components must be upgraded in the following order:

Order	Components	Details
1	All Personalization Servers and Databases	All Personalization Servers and Databases must be upgraded together - if you are using SQL replication then see separate best practice guides on upgrading the database and server.
2	All Consoles	Personalization Server is only compatible with the matching console version, so the consoles will need to be upgraded immediately after the servers. Note that the Policy configuration should not be upgraded until step 4.
3	All Agents	The Personalization components in the Environment Manager Agent are compatible with all Personalization Server versions. If you have configurations created with older consoles then you might need to upgrade agents and configurations simultaneously, one group at a time.
4	Configuration Files	When new agents have been deployed to all endpoints and are working successfully then any legacy policy configurations in use can be upgraded by the latest console and deployed.

Endpoint Configuration Merging

If you are using the Endpoint Configuration Merging function in Environment Manager, all configurations must be of the same EM product version. Upgrade all configurations before merging. For more information see Configuration Endpoint Merging in the Environment Manager help.

Conversion Rules

- Personalization Groups that used Global Desktop Settings that were Shared are placed in a GlobalShared Windows Settings Group.
- Personalization Groups that used Separate Global Desktop Settings are placed in the Windows Settings Groups to match the appropriate operating systems:
  - GlobalXP
  - GlobalVista
  - GlobalWin7
  - GlobalWin8
- Personalization Groups that used specific shared and separate Desktop Settings are placed in either of the following Windows Settings Groups:
  - [Personalization Group Name]_Shared
  - [Personalization Group Name]_OS
- Session Data is placed in a SessionData Windows Settings Group.
- Certificates and Credentials are placed in the Security Windows Settings Group.

These groups are created to ensure backward compatibility during the upgrade process. They are applied to Personalization Groups that are using Desktop Settings when the Environment Manager agent is upgraded to 8.6. It is recommended that they are replaced using the default 8.6 Windows Settings Groups wherever possible.

Upgrade The Logon Trigger

In Environment Manager 8.5, a new Logon trigger structure was introduced replacing the single Logon trigger with three sub-triggers. This increases efficiency and speeds up login times, as Environment Manager actions can be configured to run at their most appropriate point during the user logon process:

- **Pre-Session** - Actions take effect before terminal services is notified of the logon. Registry, Group Policy, and Environment actions are compatible with this sub-trigger. During the upgrade, actions that were previously in the Logon Environment tab are moved here.
- **Pre-Desktop** - Actions take effect when the user logs on to the system but before the desktop shell has started. During the upgrade, actions that were previously in the Logon trigger are moved here.

- **Desktop Created**- Actions take effect after the desktop shell and Explorer has started. To improve efficiency and logon times, any non-critical Logon actions should be added to this trigger, for example, mapping drives and printers.

Example

The graphics below show a single configuration before and after Logon trigger upgrade:
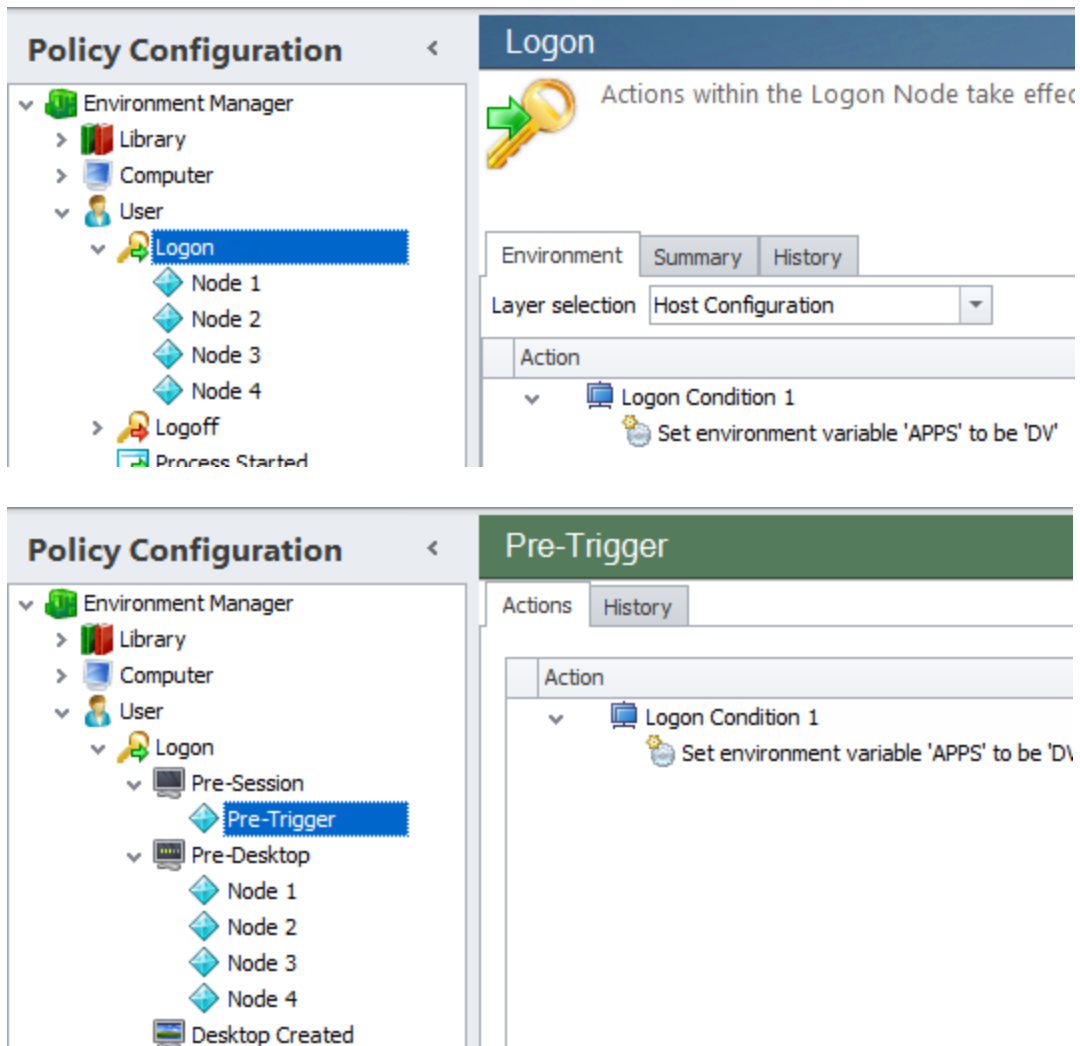
After upgrading the configuration:

Logon Condition 1 has been moved from the Logon Environment tab to the Pre-Session trigger

Nodes 1, 2, 3 and 4 have been moved from the Logon node to the Pre-Desktop trigger

The Desktop Created sub-trigger has been added

If you do not upgrade the Logon trigger, the upgraded configuration will open with the single Logon trigger. You will be prompted to upgrade each time you open the configuration.

The Logon trigger method can be changed at any time using the Advanced Settings in the Environment Manager console. See Configuration Settings in the Environment Manager Policy Help for further details.

Upgrade Certificates

The client maintains both the legacy format and the saved certificates and credentials in the same profile. When a new client logs on with an old profile, the client uses the old format to restore the certificates if there is no new-format data, and saves it in the new format. From then on there is no roaming between legacy and new clients.

Personalization Operations Bulk Operations User Selection

The Personalization Operations utility, introduced in Environment Manager 10.n, is a web-based console for managing personalization data. You can create bulk data operations for multiple users that apply to Personalization Groups, Active Directory (AD) groups, or specific users. The Active Directory (AD) group information in the database comes from the endpoints. When a user logs on using a version 10.n endpoint, the endpoint provides a list of the AD groups to which the user belongs. Endpoints running earlier versions of Environment Manager do not provide this information, so in a newly-upgraded system there is no AD group information at all, and selecting users by AD group is not possible. As the endpoints are upgraded to version 10.n and users log on, the database receives AD group information about the users and searching for AD groups works.

Upgrade Configurations

User Workspace Manager product configurations must be upgraded sequentially by major product version.

To upgrade a configuration, open it in the latest product console. The console detects that the configuration was created in an earlier version and prompts you to upgrade. When the configuration is subsequently saved, it is saved as the latest version and it is ready to be deployed using a deployment mechanism.

Environment Manager policy configuration files are upgraded when opened in a combined or policy console. When you open a configuration created in a more recent version of the console, you are asked if you want to upgrade the configuration. Click **Yes** to upgrade, click **No** to open the console with an empty configuration.

Save the configuration to complete the upgrade and ensure compatibility with the latest version of the agent, server, and console. Once the configuration has been saved it is ready to be deployed.

> ⓘ Policy configurations cannot be upgraded in the Personalization only console. They can only be upgraded if opened in the policy or combined console.

Caution: Any work in progress configurations are deleted during the upgrade process, so they must be saved before upgrading.

Upgrade Servers

All servers connected to a database must be upgraded at the same time.

For further information on upgrading servers, see Upgrade.

Uninstall

To uninstall a User Workspace Manager product use the User Workspace Manager Installer.

1. Run the User Workspace Manager Installer by executing **setup.exe** on the installation media.

2. In the Welcome screen, select **Uninstall** and click **Next**.

   The Summary screen lists the product selected to uninstall, and whether a reboot will be required.
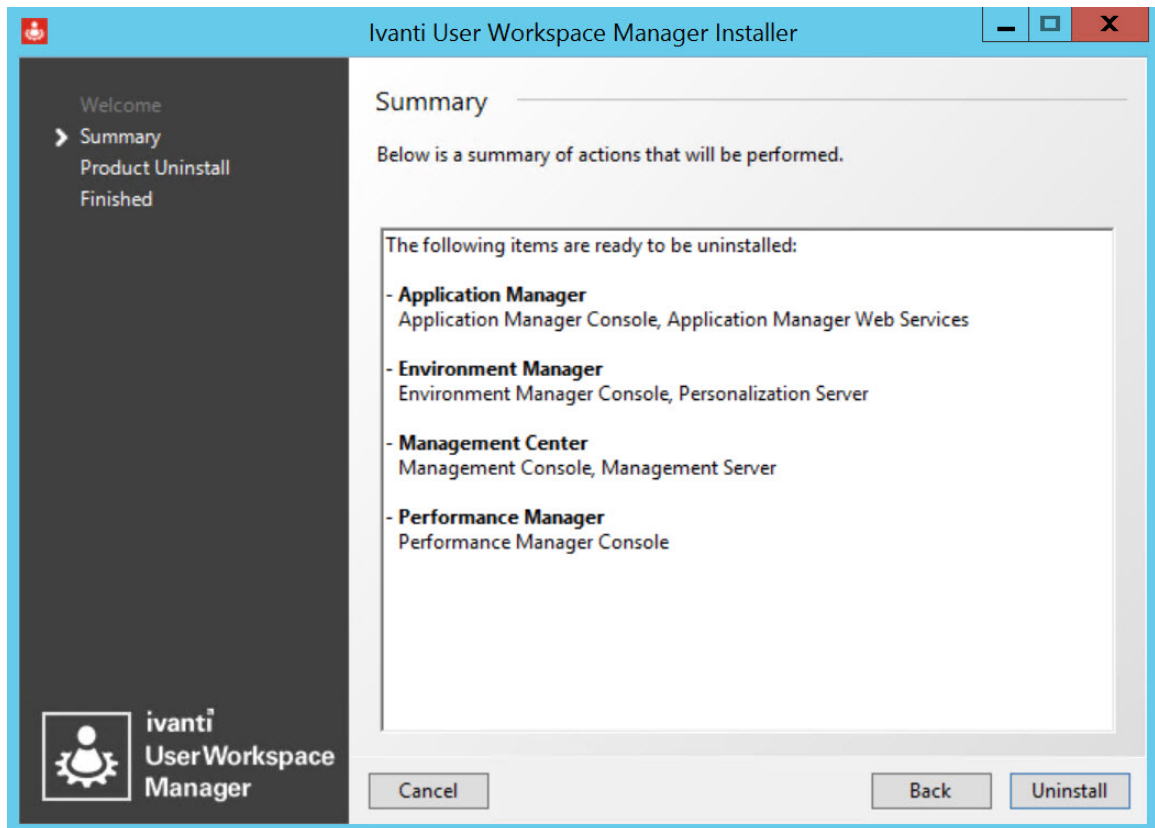
3. Click **Uninstall** to start the uninstallation process.

   The Product Uninstall screen displays - the progress of the uninstallation can be seen at the bottom of the screen. Once all products have been uninstalled the Uninstall Complete screen displays.

4. Click **Finish** to close.

Any user created configurations will not be uninstalled with the product, the A*MP files must be manually deleted from where they were saved (default location: C:\ProgramData\AppSense\<product>).

The Deployment Agent and any Packages will not automatically be removed from the managed endpoints.

If Microsoft SQL Server Express was installed during the installation process, this will not be uninstalled as part of the uninstallation process.

If User Workspace Manager is installed in Evaluation mode then you must use **Control Panel** > **Add\Remove Programs** applet to uninstall. If using this method ensure you uninstall each component of User Workspace Manager.

Multi Instance Command Line Installer

The User Workspace Manager multi instance command line installer, InstallerCmd.exe, allows you to script installation of named instances.

The InstallerCmd.exe can be run from the Bin folder on the User Workspace Manager release media.

The instance installer can be used to do the following actions:

- Install product server instances by name
- Patch specific product server instances
- Uninstall specific product server instances
- Display a list of available and installed product server instances

> The command line generated is specific to the machine it is run on. It must not be transported to another machine.

Install Options

`/i <Product.msi> <Instance Name> [Optional Parameter]`

Launches the Windows Installer to install an instance with the specified name. Additional parameters understood by the Windows Installer can be passed at the end of this command.

`/is <Product.msi> <Instance Name> [Optional Parameter]`

Displays (but does not run) the Windows Installer command required to install an instance with the specified name. Additional parameters understood by the Windows Installer can be passed at the end of this command.

Patch Options

`/p <Patch.msp> <Instance Name> [Optional Parameter]`

Launches the Windows Installer to patch an instance with the specified name. Additional parameters understood by the Windows Installer can be passed at the end of this command.

`/ps <Patch.msp> <Instance Name> [Optional Parameter]`

Displays (but does not run) the Windows Installer command required to patch an instance with the specified name. Additional parameters understood by the Windows Installer can be passed at the end of this command.

Uninstall Options

`/x <Product.msi> <Instance Name> [Optional Parameter]`

Launches the Windows Installer to uninstall an instance with the specified name. Additional parameters understood by the Windows Installer can be passed at the end of this command.

```
/xs <Product.msi> <Instance Name> [Optional Parameter]
```

Displays (but does not run) the Windows Installer command required to uninstall an instance with the specified name. Additional parameters understood by the Windows Installer can be passed at the end of this command.

Display Options

```
/e <Product.msi>
```

Lists available and installed instances associated with the specified product. The instance name is displayed for installed instances.

Details of Optional Parameters can be found in the help for Windows Installer.

Run: `MSIEXEC /?`

InstallerCmd.exe Examples

The following are examples of how to use InstallerCmd.exe.

Install Example

```
InstallerCmd.exe /i <path>\ManagementServer64.msi TestInstance /q
```

Installs an instance, named TestInstance, of the Management Server and passes the /q parameter to the Windows Installer resulting in no user interface being displayed during installation.

Uninstall Example

```
InstallerCmd.exe /x <path>\PersonalizationServer64.msi TestInstance /l* uninstall.log
```

Uninstalls the Personalization Server instance named TestInstance and saves the log to uninstall.log

Management Center MSI Custom Actions

MSI Custom Actions

There are a number of Custom Actions used within the Management Center that provide enhanced capabilities to the standard action. Each Custom Action relates to one of the following MSI's used in the installation of the Management center:

- ManagementConsole32.msi or ManagementConsole64.msi
- ManagementServer64.msi
- ClientCommunicationAgent32.msi or ClientCommunicationAgent64.msi

Management Console Custom Actions

Name	doGetManagementServerURL
Type	1 – Assembly: InstallerActions, Method: GetManagementServerURL
Description	Reads the last used server from ServerSettings.xml and sets the MANAGEMENTSERVERURL msi property to the most recently used server.
	This is run during a GUI install to pre-populate the management server dialog with a default management server when not upgrading an old version.
	Reads: [%APPDATA%\AppSense\ServerSettings.xml]
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	doStripServerPath
Type	1 – Assembly: InstallerActions, Method: StripServerPath
Description	Removes the "/ManagementServer" path from the MSI property MANAGEMENTSERVERURL. Reads property MANAGEMENTSERVERURL. Sets property MANAGEMENTSERVERURL.
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	EnterServerWarning
Type	4134 – Embedded VBScript
Description	Displays a message box stating "Please enter the Management Server URL" if the user forgets.
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	N/A
Mis-uses MSI Prefixes	No

Set Registry Keys	No
GacUtil	No

Name	CheckURL
Type	1 – Assembly: InstallerActions, Method: TestURL
Description	Checks whether a given URL is well-formed. Reads property MANAGEMENTSERVERURL, Sets MANAGEMENTSERVERURL to canonical version if URL is good. Set URL_OK to either "1" if good or empty string if bad URL
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	BadURL
Type	4134 – Embedded VBScript
Description	Displays a message box saying "URL is invalid" if the user types a bad URL. Run if property URL_OK is empty.
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Immediate

Has Rollback	N/A
MSIProcessMessage	N/A
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	ExecuteManagementConsoleUninstall
Type	210 –executable: Console.exe
Description	Invokes the console executable in silent mode to perform uninstall tasks. Deletes: [%LOCALAPPDATA%\AppSense\ManagementConsole]
Occurs During	Uninstall
Changes System State	Yes
Reversible	N/A
Execute	In a script
Has Rollback	No
MSIProcessMessage	N/A
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	AddServer
Type	1025 –assembly: InstallerActions, Method: AddServer
Description	Adds the value in [MANAGEMENTSERVERURL] into the list of known management servers in ManagementServers.xml, creating it if needed. If a new entry is added into this file, then it is marked as new with a special tag. This tag is used to remove the entry on rollback, and removed during the commit phase.

Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	deferred
Has Rollback	Yes
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	RemoveServer
Type	1281 –assembly: InstallerActions, Method: RemoveServer
Description	Removes [MANAGEMENTSERVERURL] from the ManagementServers.xml file if it was marked as new
Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	rollback
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	RemoveMarkerTags
Type	1537 –assembly: InstallerActions, Method: RemoveMarkerTags

Description	Removes the 'new' tags from the ManagementServers.xml file
Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	commit
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	TurnOffGeneratePublisherEvidence
Type	1025 – assembly: InstallerActions, method TurnOffGeneratePublisherEvidence
Description	This action reads the ManagementConsole.exe.config file and ensures that the generatePublisherEvidence element exists and is set to false. This will not have any effect during a normal install as the installed file has this set. Writes to: [CONSOLEINSTALLDIR]ManagementConsole.exe.config [CONSOLEINSTALLDIR]ManagementConsole.exe.config.tmp
Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	deferred
Has Rollback	Yes
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No

Set Registry Keys	No
GacUtil	No

Name	TurnOffGeneratePublisherEvidenceCommit
Type	1025 – assembly: InstallerActions, method TurnOffGeneratePublisherEvidenceCommit
Description	This action deletes any tmp files created by the TurnOffGeneratePublisheEvidence action Deletes: [CONSOLEINSTALLDIR]ManagementConsole.exe.config.tmp
Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	commit
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	TurnOffGeneratePublisherEvidenceRollback
Type	1025 – assembly: InstallerActions, method TurnOffGeneratePublisherEvidenceRollback
Description	This action copies any tmp files made by the TurnOffGeneratePublisheEvidence action over the original files, then deletes the tmp files Writes To: [CONSOLEINSTALLDIR]ManagementConsole.exe.config Deletes: [CONSOLEINSTALLDIR]ManagementConsole.exe.config.tmp
Occurs During	Install

Changes System State	Yes
Reversible	N/A
Execute	Rollback
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	WixFailWhenDeferred
Type	1025 – assembly: WixCA, method WixFailWhenDeferred
Description	Part of the WiX library. http://wix.sourceforge.net/manual-wix3/wixfailwhendeferred.htm This action simply fails the install during the deferred execution stage. Set the property WIXFAILWHENDEFERRED=1 to cause the failure.
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Deferred
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	FolderToDelete

Type	1 – assembly: InstallerActions, method:
Description	This action deletes the specified directory
Occurs During	Uninstall
Changes System State	No
Reversible	N/A
Execute	Deferred
Has Rollback	No
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Management Server Custom Actions

Name	WixUIValidatePath
Type	65 – 3rd party assembly: WixUIWixca, Method: ValidatePath
Description	Part of the Wix library.
Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	WixUIPrintEula
Type	65 – 3rd party assembly: WixUIWixca, Method: PrintEula
Description	Part of the Wix library.
Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	GetPlatform
Type	1 –assembly: InstallerActions, Method: GetPlatform
Description	Sets PLATFORM with either "32-bit" or "64-bit" depending on the platform being installed on.
Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	N/A
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	TurnOffGeneratePublisherEvidence1 & 2
Type	1025 –assembly: InstallerActions, Method: TurnOffGeneratePublisherEvidence
Description	This action reads the ManagementConsole.exe.config file and ensures that the generatePublisherEvidence element exists and is set to false. This will not have any effect during a normal install as the installed file has this set. This task is split into 2 due to limitations in the amount of information that can be passed in a single windows installer property

Writes to:

[BINDIR]AlertsServices.exe.config

[BINDIR]DeploymentService.exe.config

[BINDIR]EventsDispatcher.exe.config

[BINDIR]SchedulerService.exe.config

[BINDIR]ServerConfiguration.exe.config

[BINDIR]AlertsServices.exe.config.tmp

[BINDIR]DeploymentService.exe.config.tmp

[BINDIR]EventsDispatcher.exe.config.tmp

[BINDIR]SchedulerService.exe.config.tmp

[BINDIR]ServerConfiguration.exe.config.tmp |
Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	Deferred
Has Rollback	Yes
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	TurnOffGeneratePublisherEvidenceCommit1 & 2
Type	1025 –assembly: InstallerActions, Method: TurnOffGeneratePublisherEvidenceCommit
Description	This action deletes any .tmp files created by TurnOffGeneratePublisherEvidence1 & 2 Deletes: [BINDIR]AlertsServices.exe.config.tmp [BINDIR]DeploymentService.exe.config.tmp [BINDIR]EventsDispatcher.exe.config.tmp [BINDIR]SchedulerService.exe.config.tmp [BINDIR]ServerConfiguration.exe.config.tmp
Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	Deferred
Has Rollback	Yes
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	LaunchServerUninstall
Type	3154 – executable: ServerConfig.exe
Description	This action launches the SCU with the /uninstall flag to perform generic uninstall actions: Stops all includes services Unregisters all services Stops all IIS AppPools

	Deletes all web directories & website filesystem directories
	Removes all AppPools
Occurs During	UnInstall
Changes System State	Yes
Reversible	N/A
Execute	Deferred
Has Rollback	No
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	LaunchServerBeginUpgrade
Type	3154 – executable: ServerConfig.exe
Description	This action launches the SCU with the /beginupgrade flag to perform generic upgrade actions:
	Stops all installed services
Occurs During	UnInstall
Changes System State	Yes
Reversible	N/A
Execute	Deferred
Has Rollback	No
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	LaunchServerEndUpgrade
Type	3154 – executable: ServerConfig.exe
Description	This action launches the SCU with the /endupgrade flag to perform generic upgrade actions:
	Writes to the following webconfigs:
	[BINDIR]ActiveProductDefinition\Management Server.xml
	[BINDIR]AlertsServices.exe.config
	[BINDIR]DeploymentService.exe.config
	[BINDIR]EventsDispatcher.exe.config
	[BINDIR]SchedulerService.exe.config
	[BINDIR]ServerConfiguration.exe.config
Occurs During	Install
Changes System State	Yes
Reversible	N/A
Execute	Deferred
Has Rollback	No
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	RestartServicesPostRepair
Type	1 - Assembly: InstallerActions, Method: RestartServicesPostRepair
Description	This action restarts the services at the end of a repair operation.
	This custom action does not require a corresponding commit custom action.
Occurs During	Repair

Changes System State	Yes
Reversible	N/A
Execute	Deferred
Has Rollback	Yes
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	SetAppSenseServicesToRestart
Type	1 - Assembly: InstallerActions, Method: SetAppSenseServicesToRestart
Description	This action reads a CustomActionData property listing all the AppSense Windows Services and determines which are running and which are currently stopped. This information is recorded and used if a rollback occurs to ensure that on completion of a rollback the service system state is the same.
Occurs During	Repair
Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	No
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	RestartServicesPostRepairRollBack
Type	1 – Assembly: InstallerActions, Method `RestartServicesPostRepairRollBack`
Description	This action ensures that if a rollback occurs during a repair operation, the AppSense windows services have their initial service status as determined by the SetAppSenseServicesToRestart custom action. This custom action does not require a corresponding commit custom action.
Occurs During	Repair
Changes System State	Yes
Reversible	N/A
Execute	Deferred
Has Rollback	Yes
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Client Communications Agent Custom Actions

Name	AppendAppSenseInstallMgrCM
Type	1 - Assembly: InstallerActions, Method: AppendAppSenseInstallMgrCM
Description	Reads the registry string at HKLM\System\CurrentControlSet\Control\NetworkProvider\Order\ProviderOrder and then appends a comma separated string at the end which is the name of the Appsense credential manager hook dll.
Occurs During	Install
Changes System State	Yes
Reversible	Yes
Execute	Immediate

Has Rollback	Yes
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	Yes
GacUtil	No

Name	RemoveAppSenseInstallMgrCM
Type	1 - Assembly: InstallerActions, Method: RemoveAppSenseInstallMgrCM
Description	Reads the registry string at HKLM\System\CurrentControlSet\Control\NetworkProvider\Order\ProviderOrder and then removes the AppSense credential manager hook dll name from the string without corrupting the current network provider order.
Occurs During	UnInstall, Repair
Changes System State	Yes
Reversible	N/A
Execute	Deferred
Has Rollback	No
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	Yes
GacUtil	No

Name	EnterServerWarning
Type	4134 – Embedded VBScript
Description	Displays a message box saying "Please enter the Management Server URL" if the user forgets during a manual install
Occurs During	Install

Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	CheckURL
Type	4134 – Embedded VBScript
Description	Checks whether a given URL is well-formed. Reads property MANAGEMENTSERVERURL Sets MANAGEMENTSERVERURL to canonical version if URL is good Set URL_OK to either "1" if good or empty string if bad URL
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	Yes
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	BadURL

Type	4134 – Embedded VBScript
Description	Displays a message box saying "URL is invalid" if the user types a bad URL. Run if property URL_OK is empty.
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	N/A
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	WixUIPrintEula
Type	65 – 3rd party assembly: WixUIWixca, Method: PrintEula
Description	Provides a button which allows the user to print the current EULA document from a 'Select Printer' dialog.
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	N/A
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	CheckBITS
Type	1 –assembly: InstallerActions, Method: CheckBITS
Description	Checks if BITS is installed
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	N/A
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Name	FormatWebSiteURL
Type	1 –assembly: InstallerActions, Method: FormatWebSiteURL
Description	Appends a forward slash (/) to URL if none present and adds port number.
Occurs During	Install
Changes System State	No
Reversible	N/A
Execute	Immediate
Has Rollback	N/A
MSIProcessMessage	N/A
Mis-uses MSI Prefixes	No
Set Registry Keys	No
GacUtil	No

Management Center Third Party Public Symbols Usage

Public symbols are available for all Management Server binaries by contacting Support. The Management Server also uses a number of third-party binaries which may be available by contacting the vendor; these binaries are listed below.

- DevExpress_XtraCharts_<version>.dll
- DevExpress_Charts_<version>_Core.dll
- DevExpress_XtraReports_<version>.dll
- DevExpress_XtraGrid_<version>.dll
- DevExpress_XtraTreeList_<version>.dll
- DevExpress_Data_<version>.dll
- DevExpress_XtraBars_<version>.dll
- DevExpress_OfficeSkins_<version>.dll
- DevExpress_BonusSkins_<version>.dll
- DevExpress_XtraEditors_<version>.dll
- DevExpress_Utils_<version>.dll
- DevExpress_XtraNavBar_<version>.dll
- DevExpress_XtraLayout_<version>.dll
- DevExpress_Scheduler_<version>.dll
- DevExpress_XtraRichEdit_<version>.dll
- DevExpress_XtraPrinting_<version>.dll
- log4net.dll

Where <version> is the version number of the library.