

InstallShield 2026 Release Notes

Originally released in July 2026; updated to include R2 (September 2026)

Introduction	2
New Features in 2026 R1	2
Enhancements.....	4
InstallShield 2026 R2	4
InstallShield 2026 R1	5
Important Information	8
Concurrent License Users Must Update the FlexNet Licensing Server Software	8
Evaluating InstallShield	9
Obtaining the Installations for InstallShield, InstallShield Add-Ons, and the Redistributable Files.....	9
Installing More than One Edition of InstallShield.....	9
Installing More than One Version of InstallShield.....	10
Remove .NET/J# Tab and Include MSI Engine Options from the Releases View	10
InstallShield Edition Naming Change	10
Project Upgrade Alerts.....	10
General Information about Upgrading Projects that Were Created in Earlier Versions of InstallShield	11
Changes to the List of Supported Versions of Windows for Target Systems.....	11
Localized String Considerations.....	11
Resolved Issues.....	12
InstallShield 2026 R2	12
InstallShield 2026 R1	13
System Requirements.....	15
For Systems Running InstallShield	15
For Target Systems.....	16
Known Issues	17
Access Issue with Network Drive in VMware Workstation for InstallShield as Administrator	17
InstallScript Object Restriction for 64-bit InstallScript Setups	17
Legal Information	18

Introduction

InstallShield is the industry standard for authoring high quality Windows Installer, InstallScript-based installations, and MSIX packages.

InstallShield 2026 also offers new features, enhancements, and resolved issues that make it easy to use the latest technologies.

New Features in 2026 R1

InstallShield 2026 R1 includes the following new features:

- [Support for Running PowerShell Scripts in MSIX Package Using Package Support Framework](#)
- [Support for WinUI-Based Wizard Pages](#)
- [Support for Passing Command-Line Arguments to MSIX-Packaged Applications Using Custom Fixup](#)
- [Support for Microsoft Entra ID Authentication and Custom SQL Connection String](#)

Support for Running PowerShell Scripts in MSIX Package Using Package Support Framework

InstallShield 2026 R1 introduces support for running PowerShell scripts, included in the MSIX package, using the Microsoft Package Support Framework (PSF). This capability enables application customization and compatibility actions to be performed at runtime without requiring changes to the application source code.

In InstallShield, this support functionality is enabled through custom fixups, which are configured using a PSF config.json file. This file is included within the MSIX package and is read and processed by the Microsoft Package Support Framework during the application execution.

The custom fixups allow you to explicitly define:

- Which executable within the MSIX package the PowerShell script applies to.
- When the PowerShell script is executed: either before application launch (startScript) or after application exit (endScript).
- How the PowerShell script behaves during execution, such as execution scope, visibility, error handling, and execution timing.

At runtime, the Microsoft Package Support Framework intercepts the application launch lifecycle and invokes the configured PowerShell scripts at the defined execution points—before the application starts and after the application exits—ensuring that runtime configuration and cleanup tasks are performed consistently within the MSIX container.

Support for WinUI-Based Wizard Pages

InstallShield 2026 R1 introduces the WinUI framework support for the wizard pages that are displayed during Advanced UI or Suite/Advanced UI installation or uninstallation. A new drop-down option, **WinUI**, is now available for the **Installation Type** setting—under the **Wizard Options** area on the **Wizard**

Pages node in the **Wizard Interface** view. Configuring the **Installation Type** setting to the **WinUI** option also enables the following additional setting and **Wizard Format** drop-down options that help you customize the installer user experience:

- **WinUI Frameless Window**—Specify whether WinUI-based wizard pages are displayed without a standard frame. Available drop-down options are:
 - **Yes**—Displays WinUI-based wizard pages in a frameless window for a modern appearance.
 - **No**—Uses the standard window frame. By default, the **WINUI Frameless Window** setting is set to **No**.
- **Wizard Format**—Specify the required visual style to be applied to wizard pages when **Installation Type** is set to **WinUI**. Available drop-down options are:
 - **Mica**—Setting this option applies the Windows 11-style Mica material effect for a modern appearance.
 - **Mica Alt**—Setting this option applies the Mica Alt material effect, which provides a stronger background tint than Mica.
 - **Acrylic**—Setting this option applies the Windows 10-style Acrylic material effect instead of the Windows 11 Mica styling.



Note - The WinUI framework support is available exclusively in InstallShield Premier Edition. This feature is not available in InstallShield Professional and InstallShield Express Editions.

Support for Passing Command-Line Arguments to MSIX-Packaged Applications Using Custom Fixup

InstallShield 2026 R1 introduces support for passing command-line arguments to an application when it is launched using the installed MSIX package. This capability is enabled through the Microsoft Package Support Framework (PSF). Since the MSIX-packaged application entry points do not natively allow startup arguments, this capability enables packagers to reliably supply required parameters at launch time without modifying the application itself.

To use this functionality, you need to define the desired arguments in the PSF config.json file, and then apply them using a custom fixup under the **Package Support Framework** area of the **Application Settings** view of an MSIX project type.

For more details, see [Application Settings](#) in the *InstallShield Help Library*.

At runtime, the Microsoft Package Support Framework processes the configuration and ensures that the defined arguments are passed to the application when it is launched, enabling greater flexibility and compatibility for MSIX-packaged applications.

Support for Microsoft Entra ID Authentication and Custom SQL Connection String

InstallShield 2026 R1 introduces support for the Microsoft Entra ID (formerly Azure Active Directory) authentication when connecting to Microsoft SQL Server databases. This new authentication enables secure, modern authentication mechanisms while providing full control over database connection configuration.

Additionally, InstallShield 2026 R1 introduces a new setting, **Connection String**, in the **Requirements** tab of an SQL connection within the **SQL Scripts** view. This new setting allows you to define a new connection string or customize the existing SQL Server connection string for any database server used during installation. Typically, an SQL server connection string can include essential connection details, such as the server name, database name, and authentication parameters, including Microsoft Entra ID-based credentials.

The **Connection String** setting is enabled only when the following conditions are met:

- The **Specify Custom Connection String** check box in the **Requirements** tab is selected.
- A supported and desired database server in the **Database Servers** list on the **Requirements** tab is selected.

When an SQL Server connection string is configured and the installation project is saved, the connection string information is automatically stored in the ISSQLDBMetaData table—available under the **Tables** explorer in the **Direct Editor** view—and is used during SQL script execution during installation. This ensures that the installation process consistently uses the defined authentication method and connection parameters.

The **Connection String** setting may provide the following benefits:

- Native support for Microsoft Entra ID authentication scenarios
- Greater flexibility in configuring secure SQL Server connections
- Centralized control of database connectivity within the InstallShield IDE

Enhancements

InstallShield 2026 includes the following enhancements:

- [InstallShield 2026 R2](#)
- [InstallShield 2026 R1](#)

InstallShield 2026 R2

InstallShield 2026 R2 includes the following enhancement:

- [Automation Interface Support to Configure InstallShield Prerequisites Location](#)

[Automation Interface Support to Configure InstallShield Prerequisites Location](#)

InstallShield 2026 R2 introduces support for configuring the location of InstallShield prerequisites through the automation interface, which determines the source from which the prerequisites are obtained during installation. This enhancement adds the `SetupPrerequisiteLocation` property that enables you to programmatically specify how prerequisite files are obtained for a release.

You can configure this property to obtain prerequisites from the source media, extract them from `setup.exe`, download them from the web, or use the locations already specified for individual prerequisites.

This enhancement provides greater flexibility when automating the configuration of InstallShield releases and helps you control how prerequisite files are obtained during installation.

For more information, see [ISWiRelease Object](#) in the *InstallShield Help Library*.



Note ▪ This change was tracked in *ISDEV-44661*.

InstallShield 2026 R1

InstallShield 2026 R1 includes the following enhancements:

- [New ISCmdBld.exe Command-Line Parameters for Configuring Custom Digital Signing](#)
- [Ability to Retrieve Suite GUID Value](#)
- [Enhanced Home Menu to Control MSIX Support](#)
- [Defining Custom SQL Server Connection String for Database Import](#)
- [Support for Windows 11, version 25H2](#)
- [New Microsoft Runtime and Database Prerequisites](#)

New ISCmdBld.exe Command-Line Parameters for Configuring Custom Digital Signing

InstallShield 2026 R1 enhances digital signing by introducing new command-line parameters. These command-line parameters allow you specify the location of a custom signing tool, configure a custom signing tool, and control whether the build-generated file (setup.exe) must be digitally signed. These command-line parameters must be used with the ISCmdBld.exe command-line tool when building a release to ensure that the setup.exe executable is digitally signed using a required or desired custom signing tool. The following are the newly introduced command-line parameters:

- **-customsign_path <path to sign tool>**—Use this parameter to specify the location of a custom signing tool that will be used to digitally sign the build-generated file (setup.exe). For example:

`ISCmdBld.exe -customsign_path "C:\Users\Admin\Documents\sign.bat"`
- **-customsign_args <arguments>**—Use this parameter to specify the command-line argument for a custom signing tool's configuration. For example,

```
ISCmdBld.exe -p "C:\InstallShield 2026 Projects\MyProject1.ism" -sign_output "Y" -  
customsign_path "C:\Program Files (x86)\Windows  
Kits\10\bin\10.0.19041.0\x64\signtool.exe" -customsign_args "sign /v /a /fd SHA256 /tr  
http://timestamp.digicert.com /td SHA256 [filename]"
```



Important ▪ Specifying `-sign_output "Y"` is a prerequisite for using the `-customsign_path` and `-customsign_args` parameters.

The [filename] variable is a placeholder for the full path of the file to be signed. During build time, it resolves to the complete path of the binary being signed. By default, the file path is appended to the end of the argument list and passed to the custom signing tool. Instead of using a hard-coded path, you can use path variables, environment variables, or property names defined within your project.

- **-sign_output <Y/N>**—Use this parameter to specify whether the build-generated file (setup.exe) must be digitally signed. For example, use the following command line to enable digital signing:

```
ISCmdBld.exe -sign_output "Y"
```

These parameters are particularly beneficial for automated build pipelines in CI/CD environments, where the default InstallShield signing mechanism may not meet specific organizational or security requirements.

For more information, see [ISCmdBld.exe](#) in the *InstallShield Help Library*.



Note • This change was tracked in *ISDEV-44198*.

Ability to Retrieve Suite GUID Value

InstallShield 2026 R1 introduces a new installScript function, **SuiteGetGuid**, which enables retrieval of the Suite GUID value associated with a running Suite installer configured using an Advanced UI or Suite/Advanced UI project.

This function can be invoked from an InstallScript action within a Suite project. When executed during installation, it retrieves the Suite GUID of the running Suite installer and stores the value in a string variable for further processing. This enhancement provides greater flexibility for Suite-based installations by allowing developers to programmatically access the Suite GUID value during installation.

For more details, see 'SuiteGetGuid' in the *InstallShield Help Library*.



Note • This change was tracked in *ISDEV-43853*.

Enhanced Home Menu to Control MSIX Support

InstallShield 2026 R1 introduces a new check box named **Enable MSIX** in the **MSIX** section of the **Home** tab on the menu bar (at the top of the InstallShield user interface). The **Enable MSIX** check box allows you control the MSIX package support for the current project directly from the **Home** tab as:

- Selecting this check box enables the **Build**, **Run**, and **Uninstall** icons in the **MSIX** section simultaneously.
- Clearing this check box disables the **Build**, **Run**, and **Uninstall** icons in the **MSIX** section simultaneously.

You can use this check box when your project is configured to create an MSIX package. This feature also simplifies workflow management by allowing you to enable or disable all MSIX-related actions in a single step, ensuring consistency and ease of use.



Note - The **Enable MSIX** check box is available only for Basic MSI projects.



Note - This change was tracked in ISDEV-44874.

Defining Custom SQL Server Connection String for Database Import

InstallShield 2026 R1 allows you override or update the existing SQL Server connection string used to connect to the Microsoft SQL Server instance—specified in the **Server Name** field on the **Database Import Wizard** dialog box in the **SQL Scripts** view. This configuration is achieved by defining a custom SQL Server connection string under the <SQLScripts> node in the Settings.xml file, which is commonly stored within the project directory or under the user profile's InstallShield configuration folders. By explicitly adding or defining the custom SQL Server connection string in this file, you can control how InstallShield establishes the database connection during the import process.

This enhancement is particularly valuable for supporting modern authentication mechanisms—Microsoft Entra ID (formerly Azure Active Directory) authentication. By incorporating the appropriate connection string parameters, including those required for Microsoft Entra ID authentication, InstallShield can securely connect to SQL Server instances that enforce advanced identity-based access controls.



Note - Support for Microsoft Entra ID authentication is available only when working with Microsoft SQL Server 2022 or later versions. Earlier versions of Microsoft SQL Server do not provide the necessary capabilities required for this authentication method.



Note - This change was tracked in ISDEV-45055.

Support for Windows 11, version 25H2

InstallShield 2026 R1 allows you to configure installation conditions for Windows 11, version 25H2.



Note - This change was tracked in ISDEV-45127.

New Microsoft Runtime and Database Prerequisites

InstallShield 2026 R1 introduces the following new prerequisites, which are now available in the **Redistributables** explorer of the **Application Data** view:

- Microsoft .NET Core 10.0 Runtime
- Microsoft Visual C++ 2017-2026 Redistributable Package (x64)
- Microsoft Visual C++ 2017-2026 Redistributable Package (x86)
- Microsoft SQL Server 2025 Redistributable (x64)

These prerequisites enable your installation to install the required Microsoft technologies on target systems, if they are not already present, ensuring that your application has the necessary runtime components for successful execution.



Note ▪ These changes were tracked in [ISDEV-44897](#), [ISDEV-45107](#), and [ISDEV-44984](#).

Important Information

Note the following important information regarding the InstallShield 2026 release:

- [Concurrent License Users Must Update the FlexNet Licensing Server Software](#)
- [Evaluating InstallShield](#)
- [Obtaining the Installations for InstallShield, InstallShield Add-Ons, and the Redistributable Files](#)
- [Installing More than One Edition of InstallShield](#)
- [Installing More than One Version of InstallShield](#)
- [Remove .NET/J# Tab and Include MSI Engine Options from the Releases View](#)
- [InstallShield Edition Naming Change](#)

Concurrent License Users Must Update the FlexNet Licensing Server Software

If you are using a concurrent license of InstallShield, you must update the version of the FlexNet Licensing Server software on the licensing server before you can start using InstallShield 2026.

The version of the FlexNet Licensing Server software that shipped with previous versions of InstallShield cannot manage licenses of InstallShield 2026. InstallShield 2026 will not check out licenses from older versions of the license server even when those licenses are available.



Note ▪ The FlexNet Licensing Server software v11.19.0 can manage licenses of InstallShield 2025 and InstallShield 2026. Upgrading from InstallShield 2025 to InstallShield 2026 lets you to use the existing FlexNet Licensing Server software v11.19.0. Upgrading from the InstallShield 2021 and older versions require the upgrading of FlexNet Licensing Server software to v11.19.0.

You can download the installer for the new FlexNet Licensing Server software from the [Revenera Product and License Center](#).

For instructions on installing the FlexNet Licensing Server software, see [Setting Up Your License Server for Concurrent Licenses](#).

Evaluating InstallShield

If you have not purchased a license for InstallShield, you can install it and use it for a limited number of days without activating it or connecting it to a license server. When you use InstallShield before activating it or connecting it to a license server, it operates in evaluation mode, and some of its functionality is not available. For details, see [Functionality Notes for the Evaluation Version of InstallShield](#). Note that the evaluation limitations are removed when you activate InstallShield or when you connect it to a license server and check out a license for it.



Note • In InstallShield 2021 to 2023, Professional Edition was renamed to “InstallShield Edition” and the Premier Edition continued to be referred to as “InstallShield Premier”. Starting with InstallShield 2024, the “InstallShield Edition” has been renamed back to “InstallShield Professional”.

Obtaining the Installations for InstallShield, InstallShield Add-Ons, and the Redistributable Files

The following installations are available for download from the Revenera Product and License Center as documented in the [InstallShield download and licensing instructions](#):

- InstallShield
- Redistributable files (for example, InstallShield prerequisites and InstallScript objects)
- Add-ons (if you are entitled to them) such as the Standalone Build and the InstallShield MSI Tools
- FlexNet Licensing Server software (if you purchased concurrent licenses and you need to set up your organization's license server)
- Skin Customization Kit
- InstallScript Object templates
- InstallShield service packs (if available)



Note • As of InstallShield 2021, the Collaboration Edition (DIM Editor) is no longer available.

Installing More than One Edition of InstallShield

Only one edition of InstallShield 2026—Premier or Professional—can be installed on a system at a time.

Microsoft Visual Studio can be integrated with only one version of InstallShield at a time. The last version of InstallShield that is installed or repaired on a system is the one that is used for Visual Studio integration.

Installing More than One Version of InstallShield

InstallShield 2026 can coexist on the same machine with other versions of InstallShield.

The InstallShield 2026 Standalone Build can coexist on the same machine with other versions of the Standalone Build. In most cases, the Standalone Build is not installed on the same machine where InstallShield is installed. If you do install both on the same machine and you want to use the automation interface, review the [Installing the Standalone Build and InstallShield on the Same Machine](#) topic in the InstallShield Help Library to learn about special registration and uninstallation considerations.

Remove .NET/J# Tab and Include MSI Engine Options from the Releases View

The **.NET/J#** tab and **Include MSI Engine** options in the **Releases** explorer on the **Media** view of the **Installation Designer** were originally provided to support .NET 1.1/2.0, Windows Installer 3.1 and J# redistributables. These technologies have become obsolete and no longer supported by the Microsoft. Therefore, starting with InstallShield 2023 all editions, the **.NET/J#** tab and **Include MSI Engine** options in the **Releases** explorer on the **Media** view of the **Installation Designer** have been removed (as well as the associated .NET 1.1/2.0 Core Language and .NET 1.1/2.0 Language Packs dialog boxes). InstallShield 2023 and later build tasks will simply ignore these options if detected in the older version of project files.

InstallShield Edition Naming Change

In InstallShield 2020 and previous releases, InstallShield was offered in two editions that were named:

- InstallShield Professional Edition
- InstallShield Premier Edition

In InstallShield 2021 to 2023, the name of the **Professional** edition was changed to **InstallShield** edition, so that the two editions were named:

- InstallShield Edition
- InstallShield Premier Edition

Starting with InstallShield 2024, the edition naming has reverted back to Professional and Premier:

- InstallShield Professional Edition
- InstallShield Premier Edition

Project Upgrade Alerts

The following information describes possible upgrade issues that may occur when you upgrade projects that were created with InstallShield 2016 and earlier to InstallShield 2026. It also alerts you to possible changes in behavior that you may notice between new InstallShield 2026 projects and projects that are upgraded from InstallShield 2016 or earlier to InstallShield 2026.

- [General Information about Upgrading Projects that Were Created in Earlier Versions of InstallShield](#)
- [Changes to the List of Supported Versions of Windows for Target Systems](#)
- [Localized String Considerations](#)

General Information about Upgrading Projects that Were Created in Earlier Versions of InstallShield

InstallShield creates a backup copy of the project with a file extension such as .777 (for an .ism project) or .2016 (for an .issuite project) before converting it. Delete the .777 or .2016 part from the original project's file name if you want to reopen the project in the earlier version of InstallShield.

You can upgrade projects that were created with the following versions of InstallShield to InstallShield 2026: InstallShield 2016 and earlier, InstallShield 12 and earlier, InstallShield DevStudio, InstallShield Professional 7 and earlier, and InstallShield Developer 8 and earlier.



Note - Consider the following information:

- You cannot open InstallShield 2026 projects in earlier versions of InstallShield.
- Projects that were created with InstallShield MultiPlatform or InstallShield Universal cannot be upgraded to InstallShield 2026.

Changes to the List of Supported Versions of Windows for Target Systems

For all project types, the Windows 7 and Windows Server 2008 R2 are the minimal Windows versions needed for target systems to run the installation generated in InstallShield.

Localized String Considerations

Changes to the handling and detection of localized strings were introduced starting in InstallShield 2016. For example, localized string content that includes square brackets around invalid characters can now trigger a build time warning or error. Accordingly, the following new warning and errors might occur when you are working with your installation.

Error or Warning Number	Message	Troubleshooting Information
-7355	The %4 value for string %2 does not meet validation criteria for table %1 column %3.	This warning occurs if a localized string value does not meet validation criteria for a column in the String Editor table. To resolve this warning, update the flagged value in the String Editor.

Error or Warning Number	Message	Troubleshooting Information
-7354	The %4 value for string %2 does not contain a legitimate value for table %1 column %3.	This error occurs if a localized string value does not contain a legitimate value in the named column of the String Editor table. To resolve this error, update the flagged value in the String Editor.

Resolved Issues

This section lists the customer issues that were resolved in the following versions of InstallShield:

- [InstallShield 2026 R2](#)
- [InstallShield 2026 R1](#)

InstallShield 2026 R2

The following issues have been resolved in InstallShield 2026 R2:

Issue Number	Issue Summary
ISDEV-45204	An InstallShield project built using MSBuild failed when the InstallShield.Tasks.dll file was not present in the expected installation folder. This issue has been fixed.
ISDEV-44884	Building an InstallScript project with custom signing caused signing details to be displayed in the build log during dependency checks instead of after the dependency check operations completed. This issue has been fixed.
ISDEV-43241	The EULA text failed to appear on the LicenseAgreement dialog when a project was upgraded as a minor upgrade. This issue has been fixed.
ISDEV-42946	During a Suite/Advanced UI installation, both the ISInstallProgress and ISParcelProgress properties failed to return the required values when accessed from a custom action. This issue has been fixed.
ISDEV-45221	Version-specific registry keys failed to be removed during uninstallation when a project upgraded from InstallShield 2022 R2 to InstallShield 2025 R2 or 2026 R1 versions. This issue has been fixed.
ISDEV-44631	An InstallScript project built with 'Romanian' language configured in the Setup Languages setting led the SdFeatureTree dialog to display incorrect values for the required and available disk space. This issue has been fixed.
ISDEV-44452	An InstallShield IDE crash occurred when navigating to the Setup Design view under the Organization section. This issue has been fixed.

Issue Number	Issue Summary
ISDEV-45220	Adding IgnoreTmpRename="1" to the prerequisite definitions failed to suppress reboot detection, triggered by the PendingFileRenameOperations registry value, during execution of custom .NET 10.0.5 prerequisites. This issue has been fixed.
ISDEV-44543	Building a project in Visual Studio 2022 failed to display the build output in the 'Output' window. This issue has been fixed.
ISDEV-42616	Changing a product version caused the exit condition in a 64-bit Suite/Advanced UI installation to be skipped, even though the same exit condition executed and triggered successfully in the prior product version. This issue has been fixed.
ISDEV-31146	Adding file(s) to a project using Dynamic File Linking failed when a UNC path with a trailing backslash was specified. This issue has been fixed.
ISDEV-45274	Building a Suite/Advanced UI project through the automation interface caused the build process to hang or terminate with the 0xC0000409 error during process exit. This issue has been fixed.

InstallShield 2026 R1

The following issues have been resolved in InstallShield 2026 R1:

Issue Number	Issue Summary
ISDEV-44875	Using the setup.exe/tempextractpath"C:\IS_temp" command failed to extract and add the PowerShell script files to the specified custom directory. Instead, those files were extracted to the Windows default temp directory. This issue has been fixed.
ISDEV-44914	A dialog created using the SdAskOptions function failed to display the installation options when the Architecture setting, on the Build tab of a release in the Releases explorer, was set to the x64 option. This issue has been fixed.
ISDEV-44975	The RegDBQueryKey function failed to access a registry key for its subkeys and returned -1 when the Architecture setting, on the Build tab of a release in the Releases explorer, was set to the x64 option. This issue has been fixed.
ISDEV-44951	A log file generated after running an InstallScript installation contained truncated strings. This issue has been fixed.
ISDEV-44692	A crash was observed when launching a 64-bit InstallScript setup.exe that included a LaunchApplication function. This issue has been fixed.

Issue Number	Issue Summary
ISDEV-44981	A security vulnerability was observed in the InstallShield 2023 R2 kit. The issue caused a standard user to execute actions that resulted into elevated privileges (SYSTEM-level access), thereby compromising the security of the target system. This issue has been fixed.
ISDEV-44951	String truncated issues were identified in the InstallScript log file. This issue has been fixed.
ISDEV-44943	Running an InstallScript project that was built using the SdLicense2Ex function caused the following error to be displayed during installation: In function 'Sdlicense2Rtf': Unable to create dialog. This issue has been fixed.
ISDEV-44932	A failure was observed when launching a 64-bit InstallScript custom action using the LaunchAppAndWait function. This issue has been fixed.
ISDEV-44926	Configuring an InstallScript project—using an array data type along with the ServiceGetServiceState function—to x64 architecture resulted in an installation failure. This issue has been fixed.
ISDEV-45046	When launching a 64-bit InstallScript installer—including the SdAskDestPath function—with administrative privileges, the SdAskDestPath dialog failed to accept the valid destination folder and displayed an error indicating that the selected path is invalid, incomplete, or write-protected. This issue has been fixed.
ISDEV-44748	On a Window 11 24H2 machine, an issue was observed where printing the License Agreement failed with the following error message displayed: This file does not have an app associated with it for performing this action. Please install an app or, if one is already installed, create an association in the Default Apps Settings page. This issue has been fixed.
ISDEV-45170	An issue was observed where setup.exe failed to recognize and process command-line arguments when launched from a mapped network drive (for example, X:\). Consequently, installation parameters provided through the command line, such as silent installation switches and custom configuration options, were ignored. This issue has been fixed.
ISDEV-44759	Reordering subfeatures of a feature in the Setup Design view failed to save when the subfeatures were configured with the Display setting configured to Not Visible and the Advertised setting configured to Disallow Advertise . This issue has been fixed.
ISDEV-45022	An installer failed to execute a prerequisite from the /temp directory where the installer files were extracted and instead attempted to execute it from a hard-coded path. This issue has been fixed.

System Requirements

This section contains the minimum requirements for systems that run InstallShield (the authoring environment), as well as for target systems that run the installations created with InstallShield (the run-time environment).

- [For Systems Running InstallShield](#)
- [For Target Systems](#)

For Systems Running InstallShield

InstallShield runs on the latest versions of these operating systems, fully updated with the most recent patches and service packs.

Item	Description
Processor	Pentium III-class PC (500 MHz or higher recommended)
RAM	1 GB of RAM (2 GB preferred)
Hard Disk	1 GB free space
Display	Designed for XGA resolution at 1024 × 768 or higher
Operating System	• Windows Server 2012 R2 • Windows 10 • Windows Server 2016 • Windows Server 2019 • Windows 11 • Windows Server 2022 • Windows Server 2025
Privileges	Administrative privileges on the system
Mouse	Microsoft IntelliMouse or other compatible pointing device

Item	Description
Optional Integration with Visual Studio	The following versions of Microsoft Visual Studio can be integrated with InstallShield Premier or Professional Editions: • Visual Studio 2012 • Visual Studio 2013 • Visual Studio 2015 • Visual Studio 2017 • Visual Studio 2019 • Visual Studio 2022 The following editions of these versions of Visual Studio can be integrated with InstallShield Premier or Professional Editions: • Professional • Premium • Ultimate • Enterprise

For Target Systems

Target systems must meet the following minimum operating system requirements:

- Windows 7
- Windows Server 2008 R2
- Windows 8
- Windows Server 2012
- Windows Server 8.1
- Windows Server 2012 R2
- Windows 10
- Windows Server 2016
- Windows Server 2019
- Windows 11
- Windows Server 2022
- Windows Server 2025

Target systems must also support the SSE2 instruction set.

Known Issues

There are the following known issues in InstallShield 2026:

- [Access Issue with Network Drive in VMware Workstation for InstallShield as Administrator](#)
- [InstallScript Object Restriction for 64-bit InstallScript Setups](#)

Access Issue with Network Drive in VMware Workstation for InstallShield as Administrator

A mapped network drive in VMware Workstation, configured using VMware Tools, is accessible to the standard user but becomes inaccessible when accessed by InstallShield running with administrator privileges.

For more information, see [A network drive mapped in VMware Tools is accessible only to Standard users, blocking access for InstallShield running as Administrator](#) in the Revenera Community.



Note ▪ This issue was tracked in ISDEV-44315.

InstallScript Object Restriction for 64-bit InstallScript Setups

InstallScript object projects are not supported in 64-bit InstallScript setups. Attempting to build an Installscript project—targeting the 64-bit (x64) architecture—with an InstallScript object included will result in a build time error.



Note ▪ This issue was tracked in ISDEV-44640.

Legal Information

Copyright Notice

Copyright © 2026 Flexera Software. All Rights Reserved.

This publication contains proprietary and confidential information and creative works owned by Flexera Software and its licensors, if any. Any use, copying, publication, distribution, display, modification, or transmission of such publication in whole or in part in any form or by any means without the prior express written permission of Flexera Software is strictly prohibited. Except where expressly provided by Flexera Software in writing, possession of this publication shall not be construed to confer any license or rights under any Flexera Software intellectual property rights, whether by estoppel, implication, or otherwise.

All copies of the technology and related information, if allowed by Flexera Software, must display this notice of copyright and ownership in full.

Intellectual Property

For a list of trademarks and patents that are owned by Flexera Software, see <https://www.revenera.com/legal/intellectual-property.html>. All other brand and product names mentioned in Flexera Software products, product documentation, and marketing materials are the trademarks and registered trademarks of their respective owners.

Restricted Rights Legend

The Software is commercial computer software. If the user or licensee of the Software is an agency, department, or other entity of the United States Government, the use, duplication, reproduction, release, modification, disclosure, or transfer of the Software, or any related documentation of any kind, including technical data and manuals, is restricted by a license agreement or by the terms of this Agreement in accordance with Federal Acquisition Regulation 12.212 for civilian purposes and Defense Federal Acquisition Regulation Supplement 227.7202 for military purposes. The Software was developed fully at private expense. All other use is prohibited.