InstallShield 2018
Release Notes

Originally released January 2018; updated to include SP1, released May 2018

Introduction

InstallShield is the industry standard for authoring high quality Windows Installer and InstallScript–based installations, as well as Microsoft App-V packages. InstallShield 2018 helps you mitigate the risks of OSS code with a quick and easy scan of your daily build. You'll uncover OSS and IP compliance vulnerabilities before you ship, so your build becomes the first line of defense against future OSS data breaches.

InstallShield 2018 also offers new features and enhancements that make it easy to use the latest technologies.

For the latest information about InstallShield 2018, including updates to these release notes, see the online version of the InstallShield 2018 release notes.

Important • *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 2018. For more information, see Concurrent License Users Must Update the FlexNet Licensing Server Software.*

Changes in SP1

InstallShield 2018 SP1 includes the following new features:

- Configure Downloadable Updates for a Suite/Advanced UI Installation
- Support Dual Signing Using SHA1 & SHA256 Digest
- Resolved Issues in SP1

Configure Downloadable Updates for a Suite/Advanced UI Installation

In InstallShield 2018 SP1, you can now enable and configure a software downloadable update. The end user can either download a newer version or skip and proceed with the current installation process. A new **Updates** tab has been added to the **Releases** view where you can specify these downloadable update settings.

- Enable Downloadable Update Support
- Configure a Newer Version of Downloadable Updates
- Configure a Prompt Message
- Specifying the Absolute Path URL to the Downloadable Update Location

Enable Downloadable Update Support

You can now enable updates for the end users to see the downloadable update support for your Advanced UI or Suite/Advanced UI setup launcher. To enable updates, select the **Enable Updates** option on the **Updates** tab of the **Releases** view.

Configure a Newer Version of Downloadable Updates

You can configure a newer version of a downloadable update using the **Prompt User** and **Force Install** settings on the **Updates** tab of the **Releases** view:

- **Prompt User**—Select this option to prompt end users to opt to either download a newer version or skip and proceed with the current installation process.

- **Force Install**—Select this option to enable end users to forcibly download and install the new version of the Suite package (if available).

Configure a Prompt Message

You can customize the message that is displayed during the installation to prompt end users to choose to either download a newer version or skip and proceed with the current installation process. You can enter the customized message in the **Prompt Message** field on the **Updates** tab of the **Releases** view. You can either create a new message or select a message from the localized available list of strings.

Note • *The end users will see this prompt message with **Yes** and **No** options to download a newer version or skip and proceed with the current installation process.*

Specifying the Absolute Path URL to the Downloadable Update Location

In previous releases, if you wanted to make updates available for an Advanced UI or Suite/Advanced UI setup launcher for download to target systems, you would enter the absolute path URL to that location in the **Update URL** field on the **Setup.exe** tab of the **Releases** view.

In InstallShield 2018 SP1, you now enter the absolute path URL to that location in the **Update URL** field on the new **Updates** tab of the **Releases** view.

Support Dual Signing Using SHA1 & SHA256 Digest

In previous releases, you could choose the signature digest hashing algorithm only based on:

- Certificate Hash

- SHA-1

- SHA-256

In InstallShield 2018 SP1, along with the above signature digest hashing algorithms, you can now choose:

- Dual Signing - (SHA-1 and SHA-256) digest

Resolved Issues in SP1

For descriptions of resolved issues in InstallShield 2018 SP1, refer to InstallShield 2018 SP1.

New Features

InstallShield 2018 includes the following new features:

- Perform Open Source Risk Assessment with FlexNet Code Aware
- Specify Uninstallation Order of Packages in a Suite Project
- Method to Run a Suite Installation with Minimum UI
- Conditionally Set the Visibility of a Feature at Run Time
- Perform Recursive or Non-Recursive IIS Registration
- Set Forms Authentication on Web Applications
- New Option to Control Whether to Load User Profile for an Application Pool Entity
- Add Kill Process and PowerShell Custom Actions to a Transform Project

Perform Open Source Risk Assessment with FlexNet Code Aware

InstallShield now includes full integration with FlexNet Code Aware, an automated open source risk assessment and package discovery solution that enables you to quickly scan your products for security and intellectual property (IP) compliance risk.

- Supported File Types
- Running FlexNet Code Aware
- Reading the FlexNet Code Aware Report
- More Information

Supported File Types

FlexNet Code Aware supports analysis of the following files:

- Java Packages
- Node Packages
- Nuget Packages
- RPM Packages
- Ruby Packages
- EXE & DLL Files

Security vulnerabilities are looked up against the National Vulnerability Database (NVD).

Running FlexNet Code Aware

FlexNet Code Aware is part of InstallShield and no activation ID is required to activate it.

To run FlexNet Code Aware from within InstallShield, click **Scan Project using FlexNet Code Aware** on the InstallShield **Project** menu or click the FlexNet Code Aware icon on the standard toolbar.

Figure 1: FlexNet Code Aware Icon on InstallShield Toolbar

Note • *This FlexNet Code Aware menu options are disabled out if you are not currently in an open InstallShield project.*

When FlexNet Code Aware completes the scan of your project, the Results Summary view opens, displaying the number of files scanned, and the number of open-source packages and vulnerabilities found.

Figure 2: FlexNet Code Aware Results Summary

When you click the **View Report** button, a full report is displayed.

Reading the FlexNet Code Aware Report

When you click **View Report** on the Results Summary screen, the full FlexNet Code Aware report opens, consisting of an **Initial Summary** view and a **Package Inventory** view.

Initial Summary View

The **Initial Summary** view presents the user with a scan summary, and assessments of operational risk, security vulnerability exposure, and license exposure.
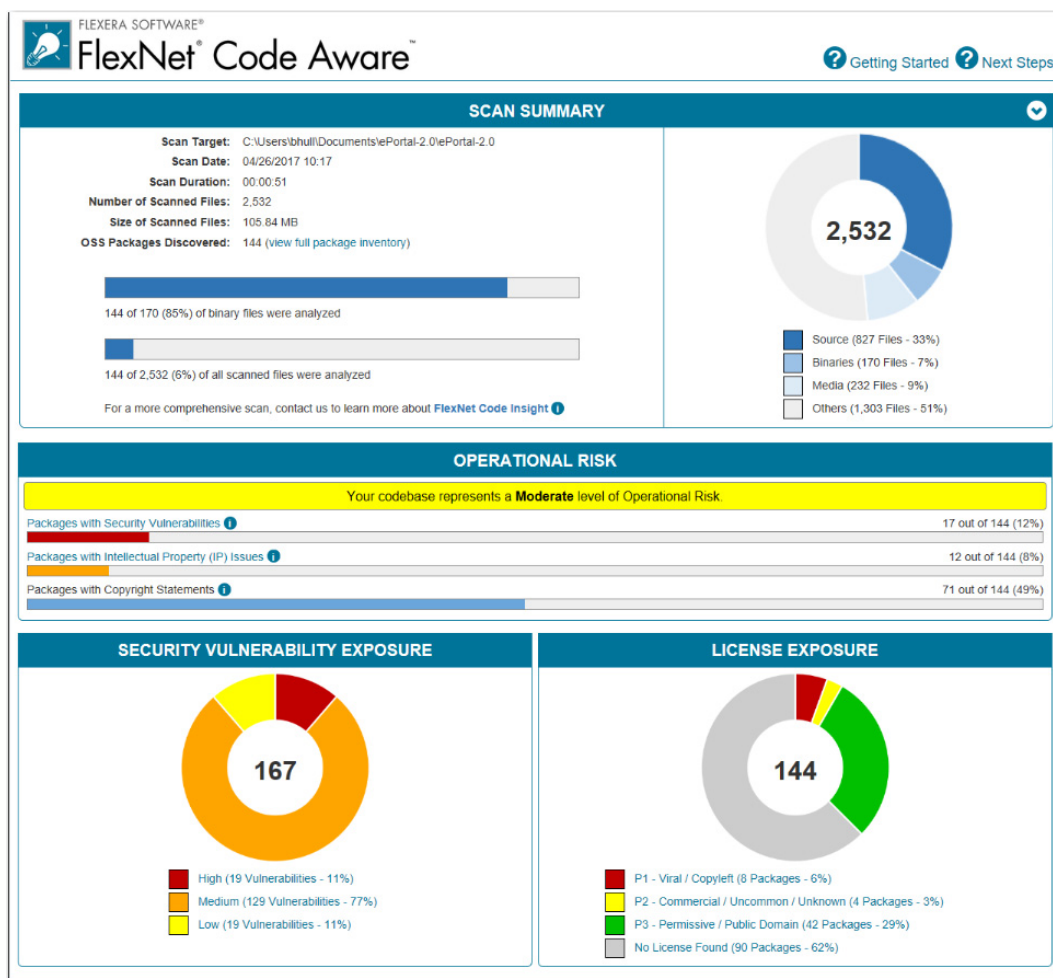
Figure 3: FlexNet Code Aware Initial Summary View

The FlexNet Code Aware Initial Summary View displays the following information:

- **Scan Summary**—This section provides details regarding the codebase that was scanned, including a breakdown of file types, percent of files analyzed, and number of findings.

- **Operational Risk**—This section provides a composite risk rating based on the combination of packages with Intellectual Property (IP) issues and packages with Security Vulnerabilities.

- **Security Vulnerability Exposure and License Exposure**—These sections provide a breakdown of the types and categories of identified issues.

Package Inventory View

The **Package Inventory** view, available by clicking **View full package inventory** in the **Scan Summary** section, provides a complete list of discovered open source and third-party packages with associated licenses, security vulnerabilities, dependencies, and detected copyright statements.

Figure 4: FlexNet Code Aware Package Inventory View

The **Package Inventory** view provides filters that you can use to execute targeted queries to refine the list to various package types of interest.

To view additional package details, click a vulnerability count listed in the **Vulnerabilities** column of the package you want to review:

Figure 5: Vulnerabilities Column

The **Vulnerabilities Detail** page opens (covering a portion of the Package Inventory view), and displays detailed information on the selected package.

struts 1.2.7

Name	struts 1.2.7
Version	1.2.7
License	▮ Apache-2.0
Vulnerabilities	▮ 7 | ▮ 2 ▮ 5 ◻ 0
Description	The core of the Struts framework is a flexible control layer based on standard technologies like Java Servlets, JavaBeans, ResourceBundles, and Extensible Markup Language (XML), as well as various Jakarta Commons packages. Struts encourages application architectures based on the Model 2 approach, a variation of the classic Model-View-Controller (MVC) design paradigm. Struts provides its own Controller component and integrates with other technologies to provide the Model and the View. For the Model, Struts can interact with any standard data access technology, including Enterprise Java Beans, JDBC, and Object Relational Bridge. For the View, Struts works well with JavaServer Pages, including JSTL and JSF, as well as Velocity Templates, XSLT, and other presentation systems. The Struts framework provides the invisible underpinnings every professional web application needs to survive. Struts helps you create an extensible development environment for your application, based on published standards and proven design patterns.
Path	C:\Users\bhull\Documents\ePortal-2.0\ePortal-2.0\extras\struts-1.2.7\contrib\struts-el\lib\struts.jar C:\Users\bhull\Documents\ePortal-2.0\ePortal-2.0\extras\struts-1.2.7\lib\struts.jar
Evidence Type	pom.xml
Maven GAV	struts:struts:1.2.7
Vendor	Apache
Copyrights	None Found

Security Vulnerabilities ⌄

CVE ID: CVE-2006-1547
Severity: ▮ High
CVSS Score: 7.8

Figure 6: Vulnerabilities Detail

More Information

For detailed information on using FlexNet Code Aware, see *Using FlexNet Code Aware to Perform Open Source Risk Assessment* in the InstallShield Help Library.

Specify Uninstallation Order of Packages in a Suite Project

Project • This information applies to the following project types:

- *Advanced UI*
- *Suite/Advanced UI*

In InstallShield 2018, you can now specify the uninstallation order of packages in a suite project using the new **Uninstallation Order** property on the **Setup.exe** tab of the **Releases** view.
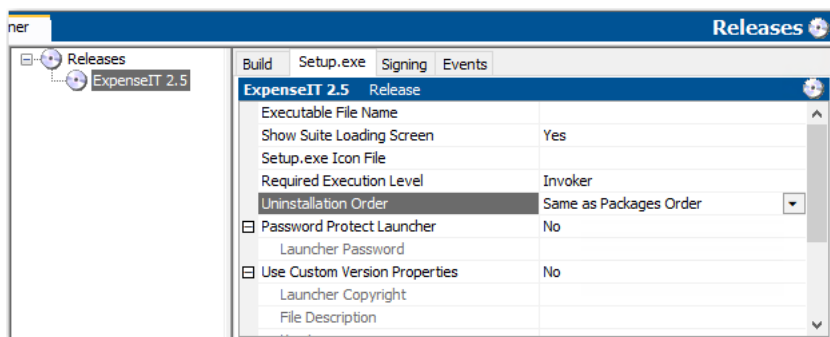
Figure 7: Uninstallation Order Setting

You can use this setting to specify the uninstallation order of the packages in a suite project by selecting one of the following options:

- **Same as Packages Order**—Uninstall the packages in the same order that packages were installed (as defined in the project).
- **Reverse of Packages Order**—Uninstall the packages in the reverse order that packages were installed (as defined in the project).

New UninstallOrder Method in Automation Interface

You can use the `UninstallOrder` method in the automation interface to set the **Uninstallation Order** property. Specify one of the following values:

- `euoForward(0)`—Uninstall the packages in the same order that packages were installed (as defined in the project).
- `euoReverse(1)`—Uninstall the packages in the reverse order that packages were installed (as defined in the project).

Note • This change was tracked in issue IOJ-1624801.

Method to Run a Suite Installation with Minimum UI

Project • *This information applies to the following project types:*

- *Advanced UI*
- *Suite/Advanced UI*

In InstallShield 2018, you can now use a new command line parameter to run a suite installation in minimum UI mode, only displaying a progress panel.

To run a suite installation in minimum UI mode, use the /passive parameter in the command line:

```
Setup.exe /passive
```

To uninstall using a minimum UI mode, use the following command:

```
Setup.exe /passive /remove
```

Note • *This change was tracked in issue IOJ-1625935.*

Conditionally Set the Visibility of a Feature at Run Time

Project • *This information applies to the following project types:*

- *Advanced UI*
- *Suite/Advanced UI*

In previous releases, you could set the **Visible** property of a feature in an Advanced UI or Suite/Advanced UI installation to **Yes** or **No** specify whether the feature should be visible on the **InstallationFeatures** wizard page of the installer.

In InstallShield 2018, you can conditionally show or hide a feature based upon a property at run time using the new **Condition** option under the **Visible** property on the **Features** view of the Installation Designer.
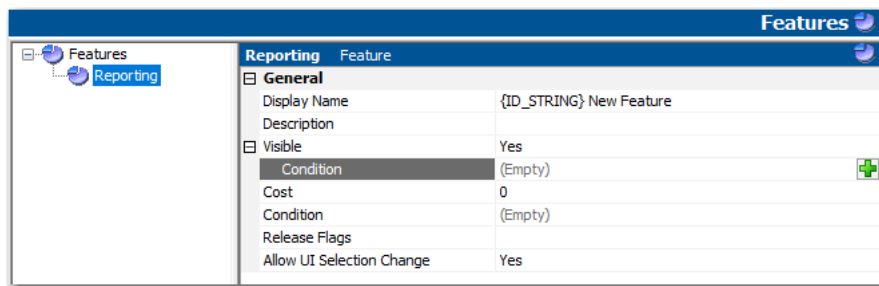
Figure 8: New Condition Option for Visible Property on Features View

You can use the **Condition** setting to specify one or more conditions that the Advanced UI or Suite/Advanced UI installation should use to evaluate whether the feature should be visible for installation by default on the **InstallationFeatures** wizard page.

For example, if you want a particular feature to be visible by default on target systems that have a particular version of Windows, you can create a condition that specifies that version of Windows.

Task   ***To conditionally display a feature in an Advanced UI or Suite/Advanced UI installation:***

1. On the **Features** view, click in the **Condition** row under the **Visible** property. A green plus sign, the **New Condition** button, appears at the end of the row.

2. Click the **New Condition** button. A new row is added under the **Condition** row.
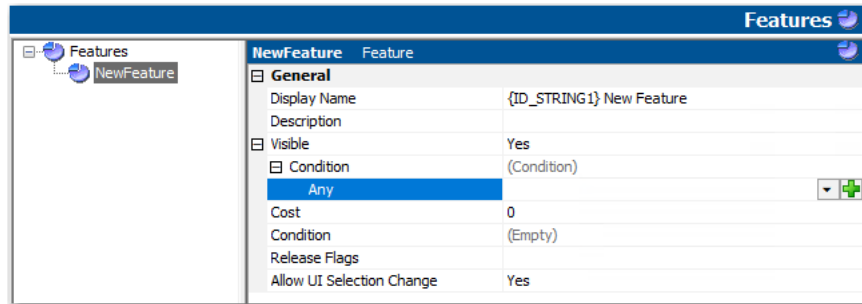
3. Click the down arrow next to the **New Condition** button and select the appropriate option—**All**, **Any**, or **None**—from the list.
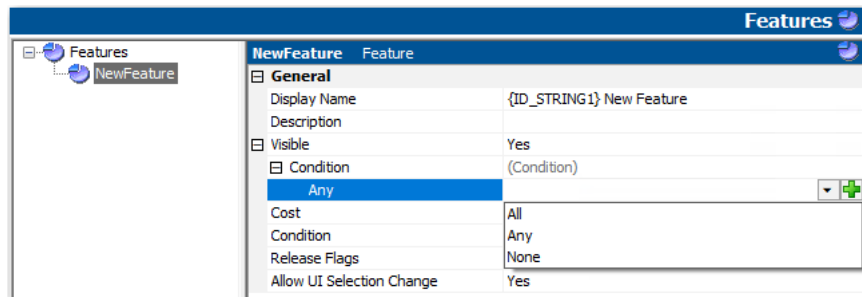
Then in the same row, click the **New Condition** button, and select the appropriate option to continue building the conditional statement.
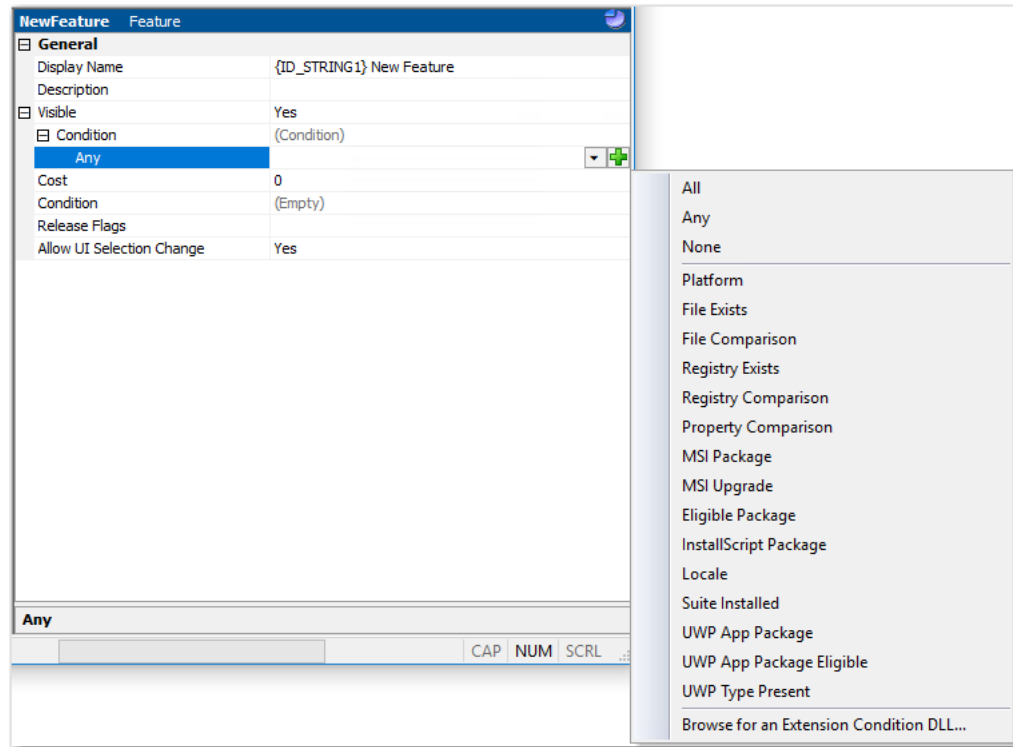
If one or more conditional statements are configured, the **Condition** property lists **(Condition)**. If none are configured, the **Condition** property lists **(Empty)**.

For more information, see Building Conditional Statements in Advanced UI and Suite/Advanced UI Projects.

New Methods in Automation Interface to Support Conditional Visibility

The following new methods have been added to the automation interface to enable you to conditionally set the visibility of a feature at run time:

Method	Syntax
AddVisibleCondition	`AddVisibleCondition() As ISWiSuiteCondition`
DeleteVisibleCondition	`DeleteVisibleCondition()`
VisibleCondition	`Read-only object property`

Note • *This change was tracked in issue IOJ-1625658.*

Perform Recursive or Non-Recursive IIS Registration

Project • *This information applies to the following project types:*

- *Basic MSI*
- *DIM*
- *InstallScript*
- *InstallScript MSI*
- *Merge Module*

A new option named **ASP.NET Registration** has been added to the **Application** settings on the **Internet Information Services** view that enables you to perform recursive or non-recursive ASP.NET registration. Using this feature enables you to install both ASP.NET applications and ASP.NET Core applications to the same website.
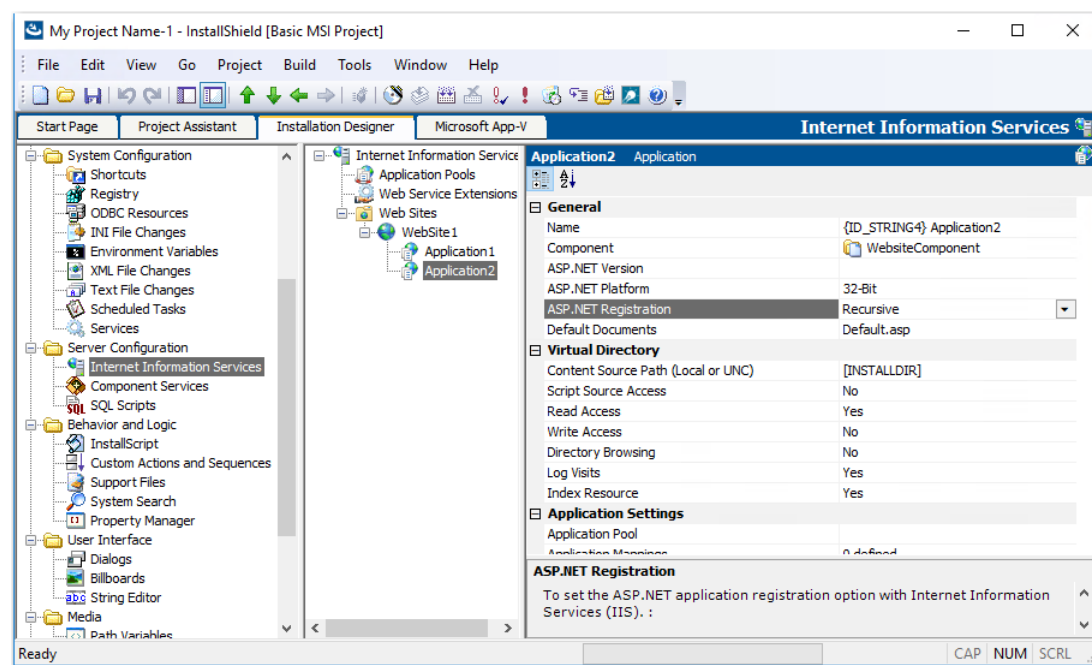
Figure 9: New ASP.NET Registration Setting

To set the ASP.NET application registration option with Internet Information Services (IIS), set the **ASP.NET Registration** property to one of the following options:

- **Recursive**—Updates script maps and application-pool assignments for the specified application and for all sub-applications.

- **Non Recursive**—Updates script maps and application-pool assignments for only the specific application. No sub-applications are changed.

Note • *This change was tracked in issue IOJ-1826269.*

Set Forms Authentication on Web Applications

Project • This information applies to the following project types:

- *Basic MSI*
- *InstallScript MSI*

InstallShield 2018 includes a new option to set forms authentication on web applications. This new option, **Forms Authentication**, is displayed under the **Authenticated Access** section of the **Internet Information Services** view for a website.
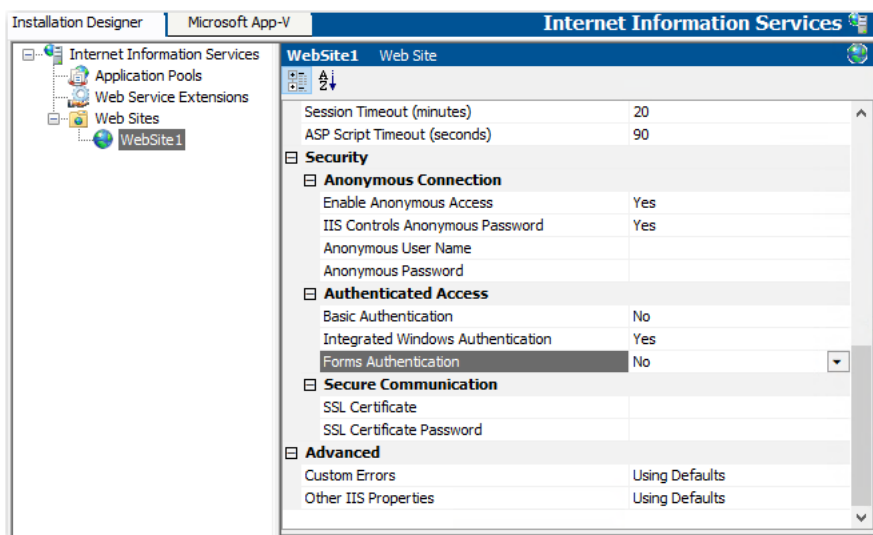
Figure 10: Forms Authentication Option on Internet Information Services View

Set the **Forms Authentication** option to **Yes** to enable forms authentication. ASP.NET forms-based authentication works well for sites or applications on public Web servers that receive many requests. This authentication mode lets you manage client registration and authentication at the application level, instead of relying on the authentication mechanisms provided by the operating system.

Important • Forms authentication sends the user name and password to the Web server as plain text. You should use Secure Sockets Layer (SSL) encryption for the Log On page and for all other pages in your application except the Home page.

Note • This change was tracked in issue IOJ-1625840.

New Option to Control Whether to Load User Profile for an Application Pool Entity

Project • *This information applies to the following project types:*

- *Basic MSI*
- *InstallScript MSI*

In InstallShield 2018, there is a new **Application Pool** settings property on the **Internet Information Services** view, named **Load User Profile**, that controls whether to load the user profile for an application pool entity.
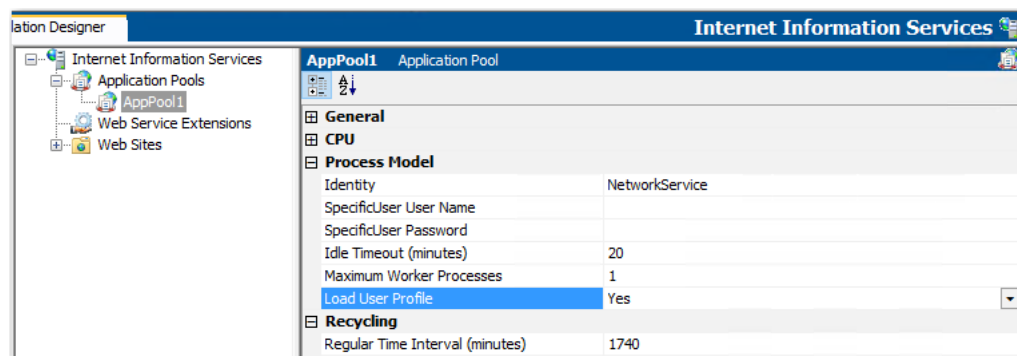
Figure 11: Load User Profile Property

Set the **Load User Profile** property to one of the following options:

- **Yes**—IIS will load the user profile for the application pool.
- **No**—IIS will not load the user profile for the application pool. This is the same behavior that occurred with IIS 6.0.

Note • *This change was tracked in issue IOJ-1659406.*

Add Kill Process and PowerShell Custom Actions to a Transform Project

Project • This information applies to the following project types:

- *Basic MSI*
- *InstallScript MSI*
- *Transform*

In previous releases, you were unable to add a Kill Process or PowerShell custom action to a Transform project. In InstallShield 2018, you can now add a **New Kill Process** or **New PowerShell** custom action to a Transform project in the **Custom Actions and Sequences** view.
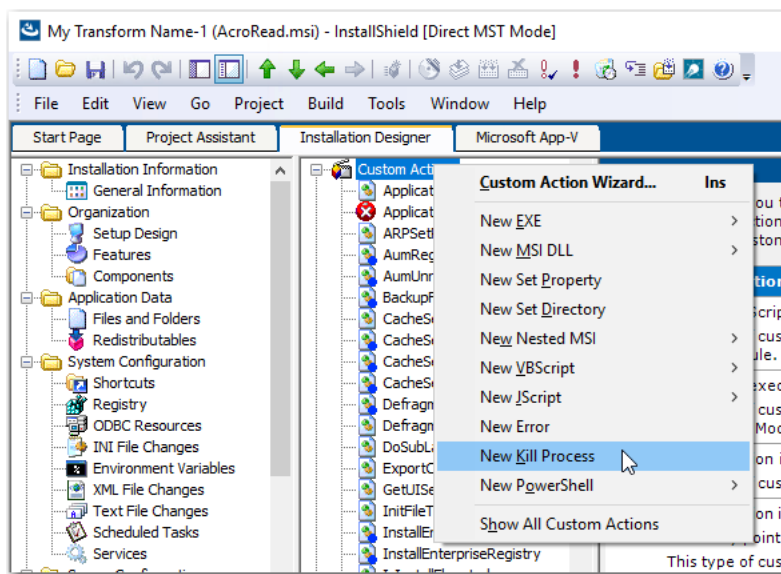
Figure 12: Adding a Kill Process or PowerShell Custom Action to a Transform Project

Note • These changes were tracked in issues IOJ-1737328 and IOJ-1625759.

Enhancements

InstallShield 2018 releases include the following enhancements:

- InstallShield 2018 SP1
- InstallShield 2018

InstallShield 2018 SP1

InstallShield 2018 SP1 includes the following enhancements:

- Configure Suite Loading Screen Message
- Customize Update Launcher Name
- Specify Character Space in PowerShell Custom Action
- Specify Predefined Public Folder

Configure Suite Loading Screen Message

Project • This information applies to the following project types:

- *Advanced UI*
- *Suite/Advanced UI*

In InstallShield 2018, you now have the ability to add a message to the Suite Loading Screen Message for your Advanced UI or Suite/Advanced UI setup launcher.

The length of the message in the Suite Loading Screen Message is limited to 35 characters.

Customize Update Launcher Name

Project • This information applies to the following project types:

- *Basic MSI*
- *InstallScript MSI*
- *QuickPatch*

In InstallShield, a new setting is introduced to customize the name of the Update Launcher. By default, InstallShield uses Update.exe is the name for the Update Launcher. Now, you can create an Update Launcher with a specified name.

Specify Character Space in PowerShell Custom Action

Project • This information applies to the following project types:

- *Advanced UI*
- *Basic MSI*
- *InstallScript*
- *InstallScript MSI*
- *InstallScript Object*
- *Merge Module*
- *Suite/Advanced UI*

In InstallShield, a new setting is introduced to specify the number of character space(s) to move forward/backward in a PowerShell custom action.

Specify Predefined Public Folder

Project • This information applies to the following project types:

- *Basic MSI*
- *DIM*
- *InstallScript MSI*
- *Merge Module*
- *MSI Database*
- *MSM Database*
- *Transform*

In InstallShield, a new predefined folder is introduced to hold the full path to the Users Public folder.

InstallShield 2018

InstallShield 2018 includes the following enhancements:

- Save QuickPatch Projects in XML Format
- Localize Product Name Property in Suite Projects
- Include the Value of a Property in a Product Configuration's Setup File Name
- New MSBuild Parameters to Set Summary Information Stream Comments and to Set Package File Name
- Specify Line Break and Tab Characters in Text File Changes
- Remove or Hide the Suite Loading Screen
- Setting to Always Create Debug Logs for Suite Installers
- New Out-of-the-Box Dialog to Set the IIS Certificate File for SSL Certificate at Runtime

- Specify Absolute or Relative Path When Creating New Child Elements in an XML File

- Setting the Default Keyboard Focus for Dialog Box Controls in Suite Projects

- PowerShell Script Editor in Basic MSI Projects

- New Option to Open Existing Transform File in InstallShield Transform Wizard

- Additional Prerequisites Included

Save QuickPatch Projects in XML Format

Project • This information applies to the following project types:

- *Basic MSI*
- *QuickPatch*

In InstallShield 2018, you can now save a QuickPatch project in XML format, and you can also create a QuickPatch project from projects saved in XML format. In previous releases, QuickPatch projects could only be saved in binary format.
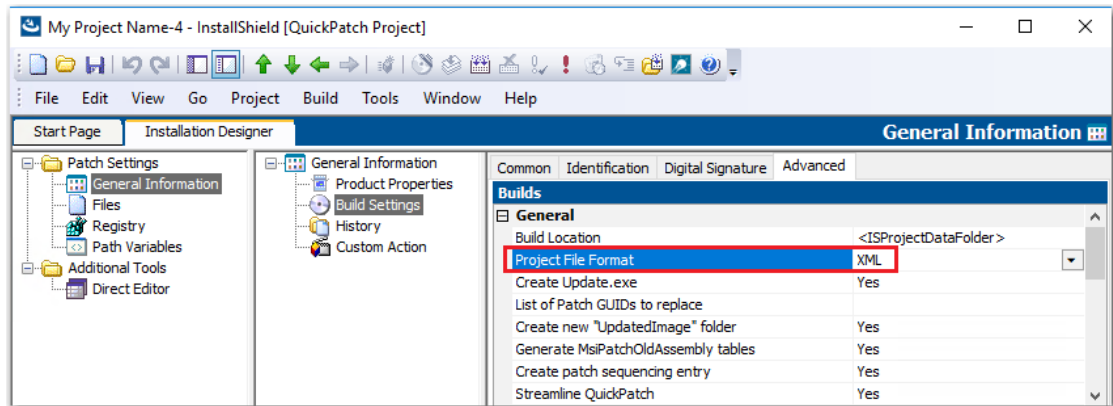
Figure 13: QuickPatch Project with XML File Format

Note • This change was tracked in issues IOJ-1624951 and IOJ-1591277.

Localize Product Name Property in Suite Projects

Project • *This information applies to the following project types:*

- *Advanced UI*
- *Suite/Advanced UI*

In InstallShield 2018, suite projects now support localizing the **Product Name** property.

To localize the Product Name property in a suite project, perform the following steps.

Task   **To localize a property in a suite project:**

1. Open a suite project and go to the **User Interface > String Editor** view.

    Create a new string that contains the localized text for each of the languages supported by your suite project, such as ID_STRING2.

English (United States)	ID_STRING2	Corporation
French (France): français	ID_STRING2	Société
Japanese: 日本語	ID_STRING2	株式会社
Spanish: español	ID_STRING2	Corporación

2. Open the **Installation Information > General Information** view.

3. Click the browse button next to the **Product Name** field to open the **Select String** dialog box.

4. Select the name of the string that you created that contains the localized text.

Note • *This change was tracked in issue IOJ-1735025.*

Include the Value of a Property in a Product Configuration's Setup File Name

⬡

Project • This information applies to the following project types:

- *Basic MSI*
- *InstallScript MSI*

In InstallShield 2018, you can now include the value of a property from the Property Table in product release configuration setup and package file names.
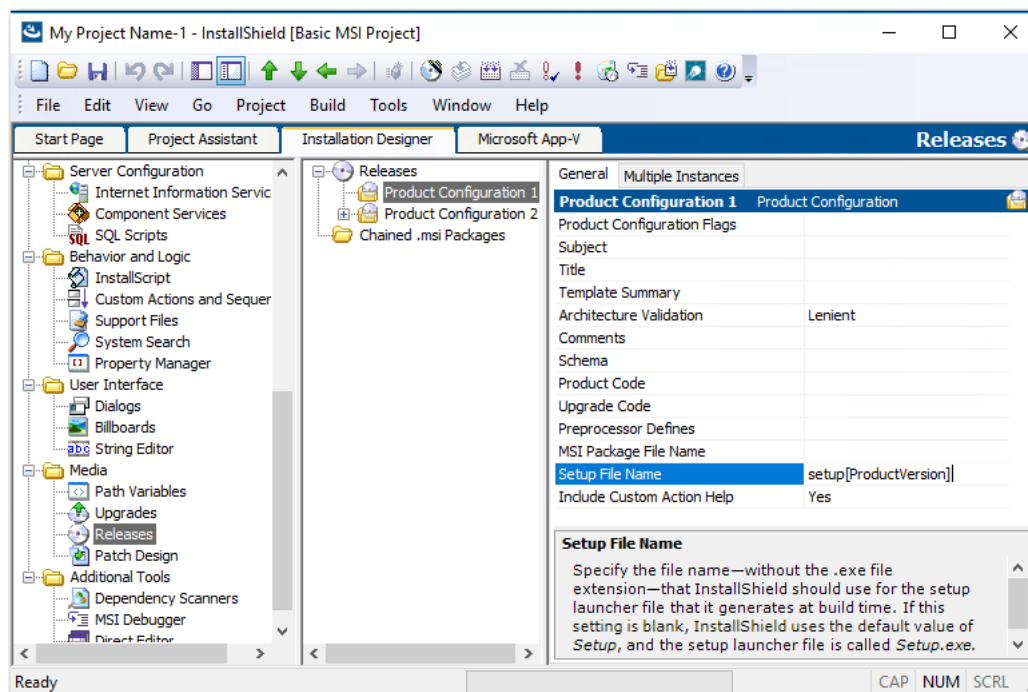
Figure 14: Entering a Property in the Setup File Name Field on the Releases View

For example, you could enter any of the following properties in the **Setup File Name** or **MSI Package File Name** field on the **General** tab of the **Releases > Product Configuration** view:

```
setup[ProductVersion]
setup[CustomVersion]
setup[ProductCode]
setup[ProductCode][ProductVersion]
```

If you entered **setup[ProductVersion]** in the **Setup File Name** field, it would result in a setup named setup14.10.1234.exe, for example.

📄

Note • This change was tracked in issue IOJ-1764179.

New MSBuild Parameters to Set Summary Information Stream Comments and to Set Package File Name

In InstallShield 2018, new MSBuild parameters were added to enable you to set add comments to an installer and to set the package file name of an installer.

- New Parameter to Set Summary Information Stream Comments

- New Parameter to Set Package File Name

New Parameter to Set Summary Information Stream Comments

Project • This information applies to the following project types:

- *Basic MSI*
- *InstallScript*
- *InstallScript MSI*
- *Merge Module*

You can add comments to an installer in the **Summary Information Stream Comments** field on the **General Information** view.

In InstallShield 2018, you also have the option of entering comments at build time. A new parameter has been added to the MSBuild.exe task, named SummaryInfoComments, to set the **Summary Information Stream** comments at build time, such as including the build number, as shown in the following example:

MSBuild.exe c:\installers\Setup.sln /Property:SummaryInfoComments="Insert Comments Here"

The comments that are added using the `SummaryInfoComments` property can be viewed on the **Properties** dialog box of the built installer.
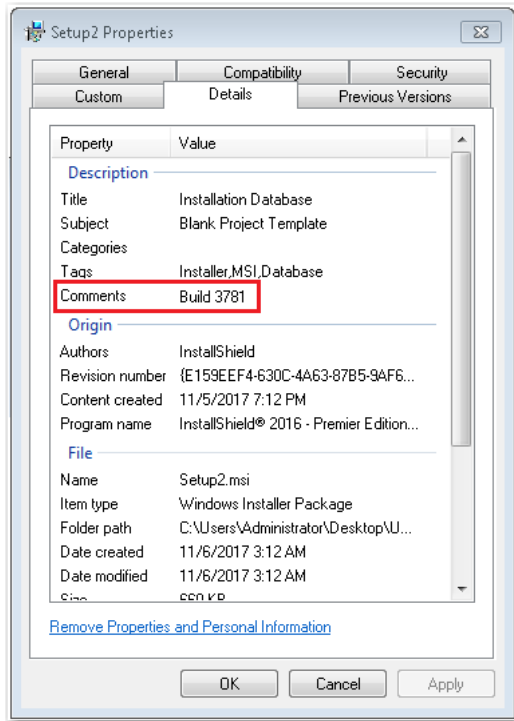
Figure 15: Comments on Properties Dialog Box

Note • *This change was tracked in issue IOJ-1735932.*

New Parameter to Set Package File Name

Project • *This information applies to the following project types:*

- *Basic MSI*
- *InstallScript MSI*

You can specify the package file name of an installer in the **MSI Package File Name** field on the **General** tab for a **Product Configuration** field on the **Releases** view.

In InstallShield 2018, you also have the option of setting the package file name at build time. A new parameter has been added to the `MSBuild.exe` task, named `MSIPackageFileName`, to set the package file name of the built installer at build time, as shown in the following example:

```
MSBuild.exe c:\installers\Setup.isproj /Property:MSIPackageFileName="MySetup"
```

When entering the value for the `MSIPackageFileName` parameter, you need to enter the file name—without the period or the file extension—that InstallShield should use for the `.msi` file.

Note • This change was tracked in issue IOJ-1735520.

Specify Line Break and Tab Characters in Text File Changes

📦

Project • This information applies to the following project types:

- *Basic MSI*
- *DIM*
- *InstallScript MSI*
- *Merge Module*
- *MSI Database*
- *Transform*

In your installer, you can configure search-and-replace behavior for content in text files that you want to modify at run time on the target system. To do this, you open the **System Configuration > Text File Changes** view and add a text **Change Set** that identifies the text files that will be searched at runtime, and also specifies the text to search for (**Find What**) and the text to replace it with (**Replace With**).

In InstallShield 2018, when adding a text Change Set, you can now enter escape sequence characters in the **Replace What** field to specify a line break or a tab.

Character	Escape Sequence
Line break	\r\n
Tab	\t

📄

*Note • For the Windows operating system, you must enter both **\r\n** to insert a line break.*

When the search-and-replace action is taken at runtime, a line break will be inserted where **\r\n** was entered in the **Replace With** field, and a tab will be entered where **\t** was entered.
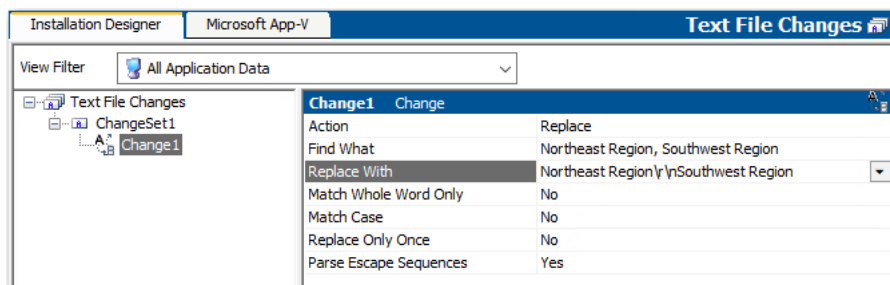
Figure 16: Adding a Line Break in Replace Text on the Text File Changes View

For these characters to be recognized as escape sequences, you also have to set the **Parse Escape Sequences** option to **Yes**.

📄

Note • This change was tracked in issue IOJ-1766354.

Remove or Hide the Suite Loading Screen

◇

Project • This information applies to the following project types:

● *Advanced UI*

● *Suite/Advanced UI*

In InstallShield 2018, you now have the ability to control whether or not the Suite Loading Screen is displayed during installation.

Figure 17: Suite Loading Screen

To control whether this screen is displayed, a new property has been added to the **Setup.exe** tab of the **Releases** view named **Show Suite Loading Screen**. If you want to hide the Suite Loading Screen for your Advanced UI or Suite/Advanced UI setup launcher, set this property to **No**.
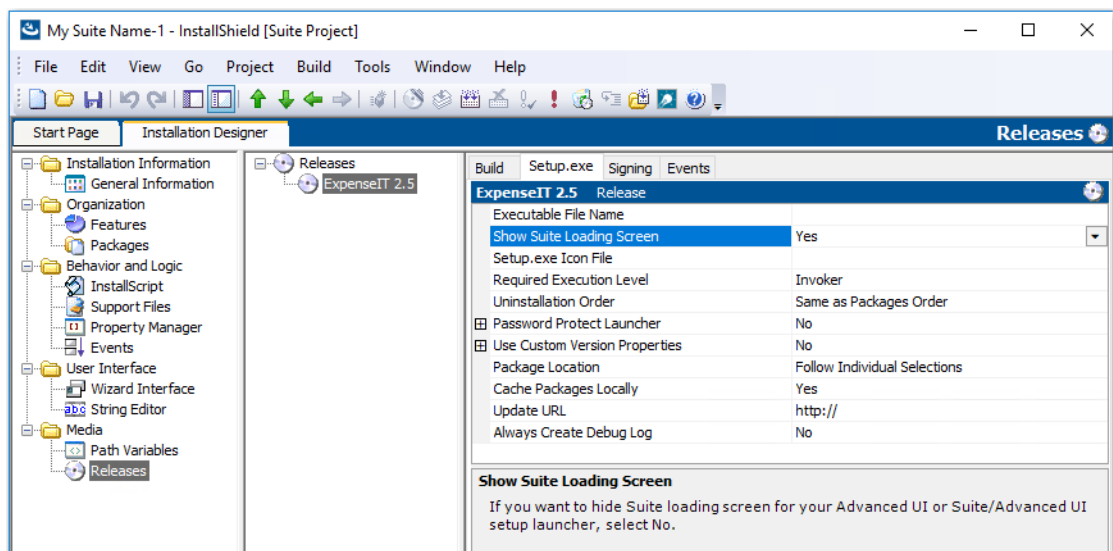
Figure 18: New Show Suite Loading Screen Property

New ShowSuiteLoadingScreen Method in Automation Interface

You can use the ShowSuiteLoadingScreen method in the automation interface to set the **Show Suite Loading Screen** setting on the **Setup.exe** tab of the **Releases** view. The default value is True.

📄

Note • This change was tracked in issue IOJ-1833774.

Setting to Always Create Debug Logs for Suite Installers

⬡

Project • This information applies to the following project types:

- *Advanced UI*
- *Suite/Advanced UI*

In InstallShield 2018, you can now select an option to turn on logging for a suite project without passing debuglog through the command line.

A new option, **Always Create Debug Log**, has been added to the **Setup.exe** tab of the **Releases** view for Advanced UI and Suite/Advanced UI projects.
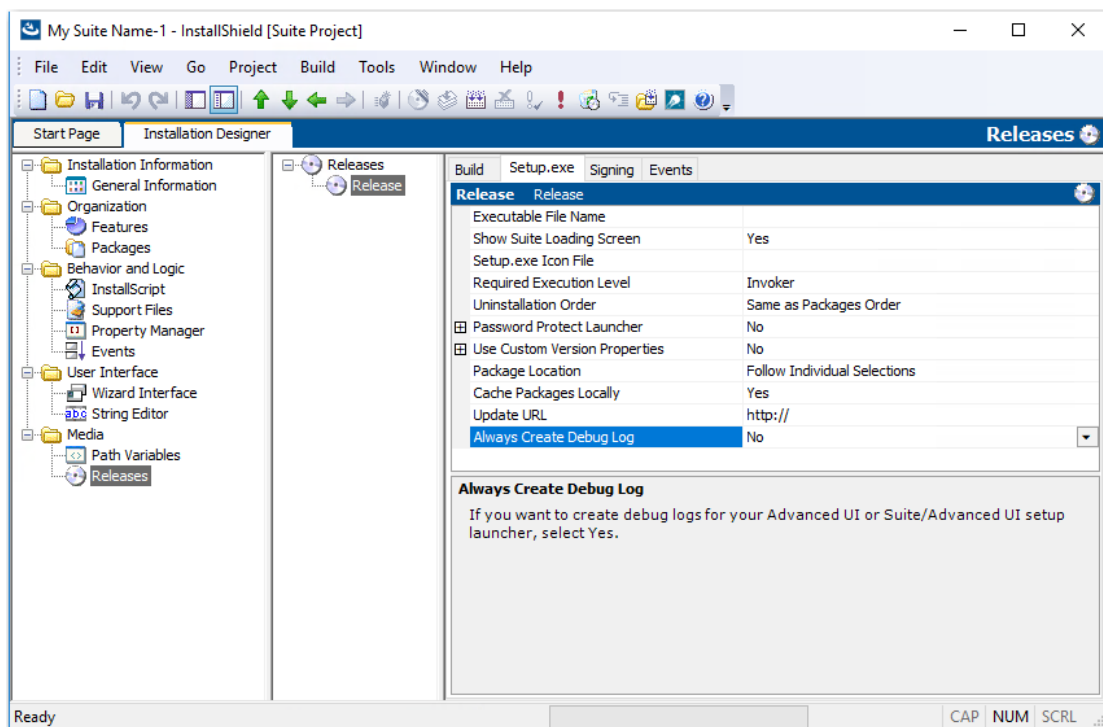
Figure 19: Always Create Debug Log Property on Setup.exe Tab of Releases View

If you want to always create debug logs for your Advanced UI or Suite/Advanced UI setup launcher, set the **Always Create Debug Log** option to **Yes**.

New CreateDebugLog Method in Automation Interface

You can use the `CreateDebugLog` method in the automation interface to set the **Always Create Debug Log** setting on the **Setup.exe** tab in the **Releases** view. The default value is `False`.

📄

Note • This change was tracked in issue IOJ-1664877.

New Out-of-the-Box Dialog to Set the IIS Certificate File for SSL Certificate at Runtime

📦

Project • This information applies to the following project types:

● *Basic MSI*

InstallShield 2018 includes a new out-of-the-box dialog (`IISBrowseSSLCertificate`) for the installer that enables the end user to browse to a IIS certificate file that they provide for the SSL Certificate and enter a password at installation runtime.
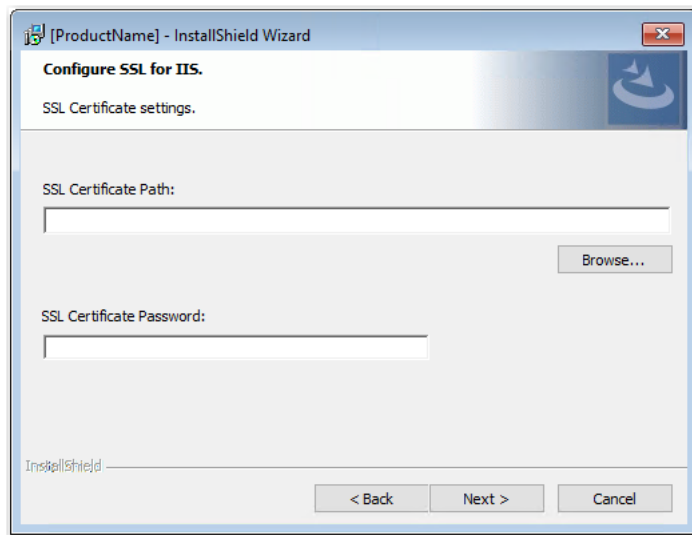
Figure 20: Configure SSL for IIS Dialog

To add a **Configure SSL for IIS** dialog to your installer, perform the following steps:

To add a "Configure SSL for IIS" dialog to your installer:

1. In the **View List** under **Server Configuration**, click **Internet Information Services**.

2. Right-click the **Web Sites** explorer and click **Add Web Site**. InstallShield adds a new Web site.

3. Select the new web site and locate the **SSL Certificate** and **SSL Certificate Password** properties under **Security > Secure Communication**.
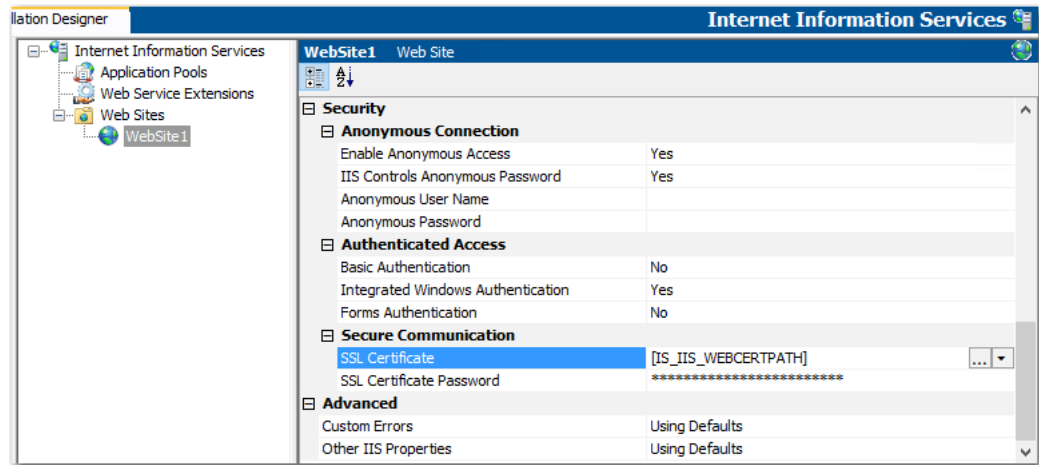
4. Set the **SSL Certificate** and **SSL Certificate Password** properties to the following values:

Property	Value
SSL Certificate	[IS_IIS_WEBCERTPATH]
SSL Certificate Password	[IS_IIS_WEBCERTPASSWORD]

5. Open the **User Interface > Dialogs** view and add the **IISBrowseSSLCertificate** dialog to the dialog sequences.
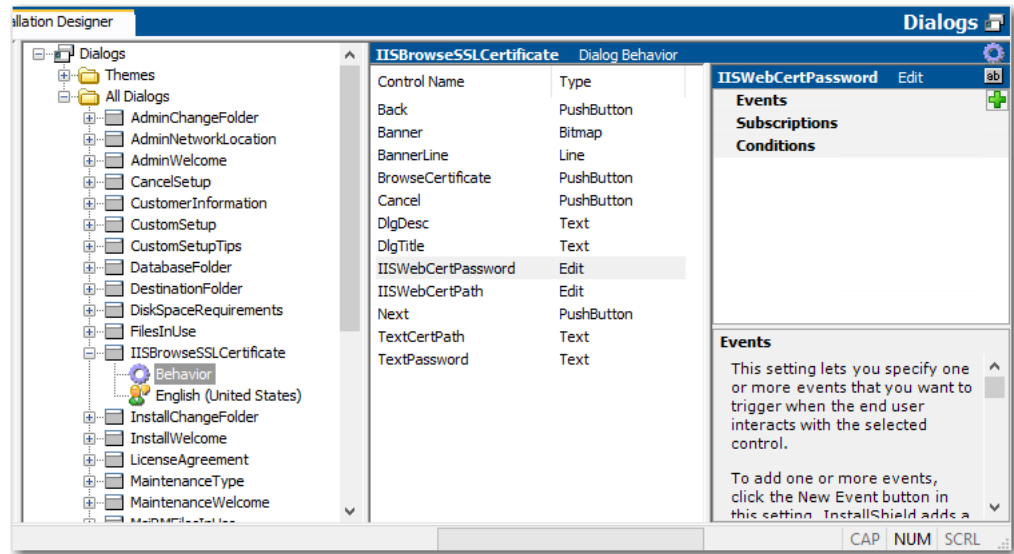
The property name for the SSL Certificate and password configured by the user is required to update in the **IISBrowseSSLCertificate** dialog for the **Edit** boxes (**IISWebCertPassword** and **IISWebCertPath**) and the push button (**BrowseCertificate**) events.

📄

Note • This change was tracked in issue IOJ-1621833.

Specify Absolute or Relative Path When Creating New Child Elements in an XML File

📦

Project • This information applies to the following project types:

- *Basic MSI*
- *DIM*
- *InstallScript*
- *InstallScript MSI*
- *Merge Module*
- *MSI Database*
- *Transform*

In previous releases, when using the **System Configuration > XML File Changes** view to add a new child element to an XML file that has the same name as a child element in an existing parent element, the XML file change would fail.

The path of a node in an XML document can be either absolute or relative. Absolute paths start at the root. When adding a new child element to an XML file that has the same name as a child element in an existing parent element, it is necessary to use the absolute path.

In InstallShield 2018, a new setting has been added to the **XML File Changes** view named **Use Absolute XPath** to enable you to specify that you want to use an absolute path when creating child elements.
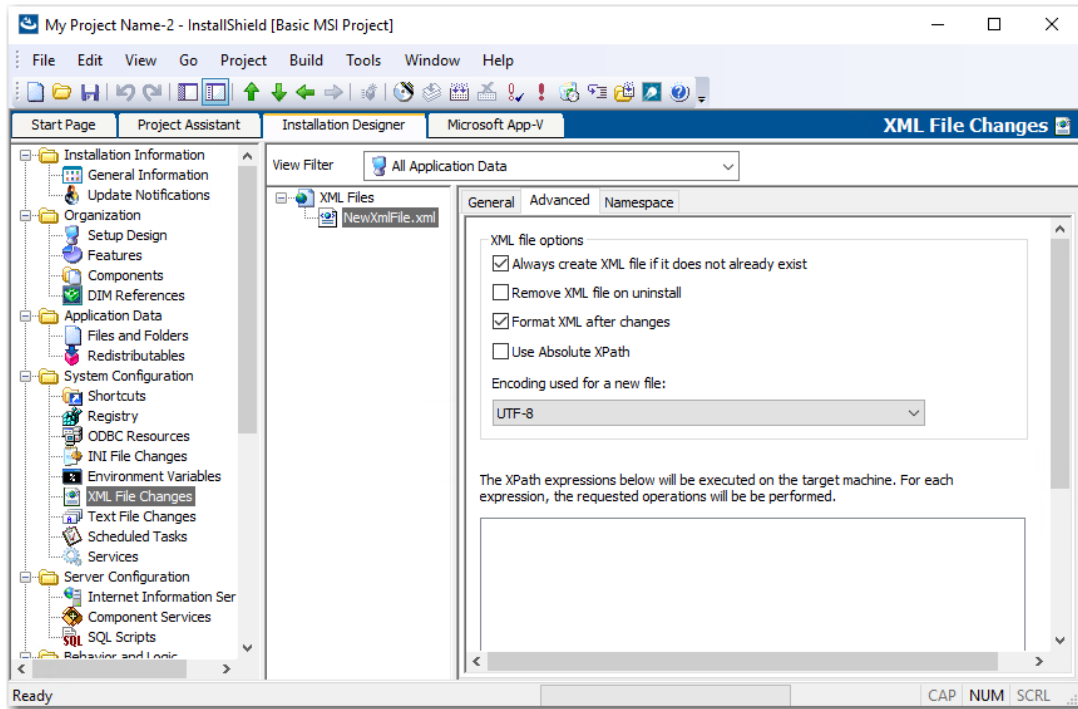
Figure 21: Use Absolute Path Option on XML File Changes View

The behavior used when creating a child element depends upon the **Use Absolute XPath** option setting:

- **Selected**—If this option is selected, Absolute XPath will be used when adding a child element.

- **Not selected**—if this option is not selected, Generic XPath will be used when adding a child element. By default, the **Use Absolute XPath** option is not selected.

Note • This change was tracked in issue IOJ-1817081.

Setting the Default Keyboard Focus for Dialog Box Controls in Suite Projects

Project • *This information applies to the following project types:*

- *Advanced UI*
- *Suite/Advanced UI*

In InstallShield 2018, when defining the wizard pages for a Suite project, you can now specify which control on a wizard page will have the default keyboard focus.
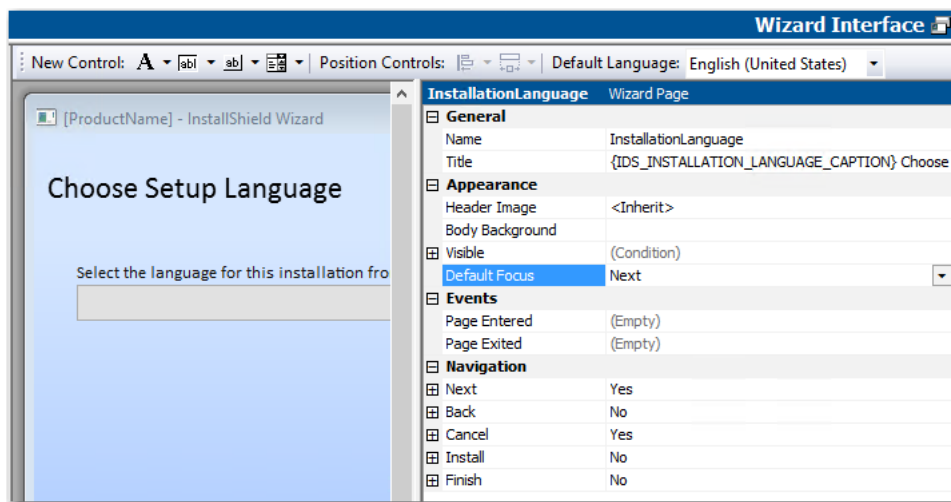
Figure 22: Default Focus Property

On the **Wizard Interface** view, there is a new property under **Appearance** named **Default Focus**, which lists all of the controls defined on that wizard page. Select a control to set the default keyboard focus to that control.

Note • *This change was tracked in issue IOJ-1626057.*

PowerShell Script Editor in Basic MSI Projects

Project • *This information applies to the following project types:*

● *Basic MSI*

● *InstallScript MSI*

In InstallShield 2018, the PowerShell script editor is available on the **Custom Actions and Sequences > Custom Actions** view for Basic MSI projects, on the new **Script** tab. In previous releases, the PowerShell Script Editor was only available for Suite/Advanced UI projects.
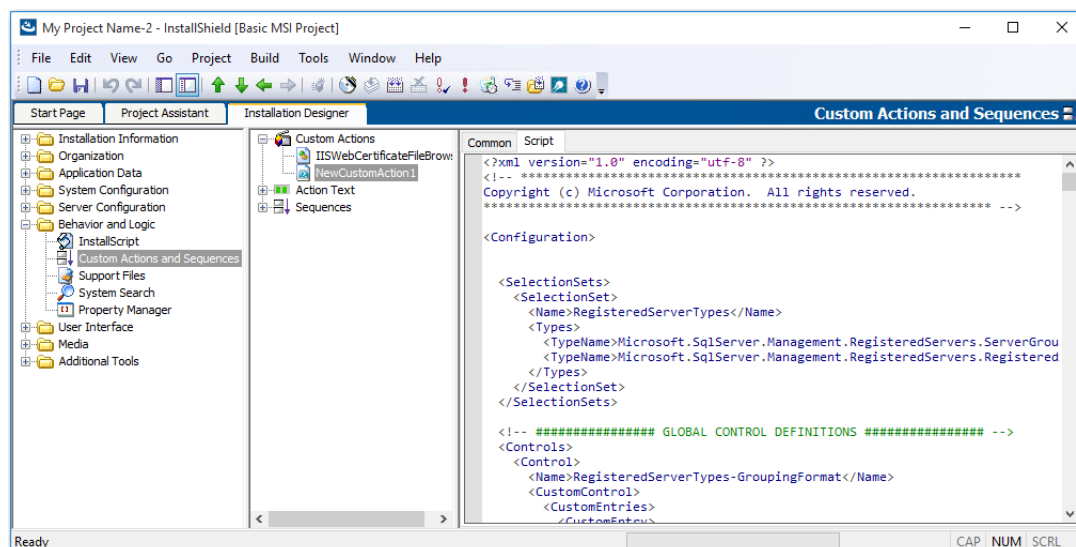
Figure 23: Script Editor on Custom Actions and Sequences View of Basic MSI Project

Note • *This change was tracked in issue IOJ-1662280.*

New Option to Open Existing Transform File in InstallShield Transform Wizard

Project • *This information applies to the following project types:*

● *Transform*

In InstallShield 2018, you can now open an existing transform file in the InstallShield Transform Wizard (as if it were being opened in the Transform Wizard for the first time), where you will be prompted to select a base MSI package for the transform file. This enables you to use the same generic transform file for multiple MSI packages.

To open an existing transform file in the InstallShield Transform Wizard, right-click on the transform file in Windows Explorer and select **Open in InstallShield Transform Wizard** from the context menu.
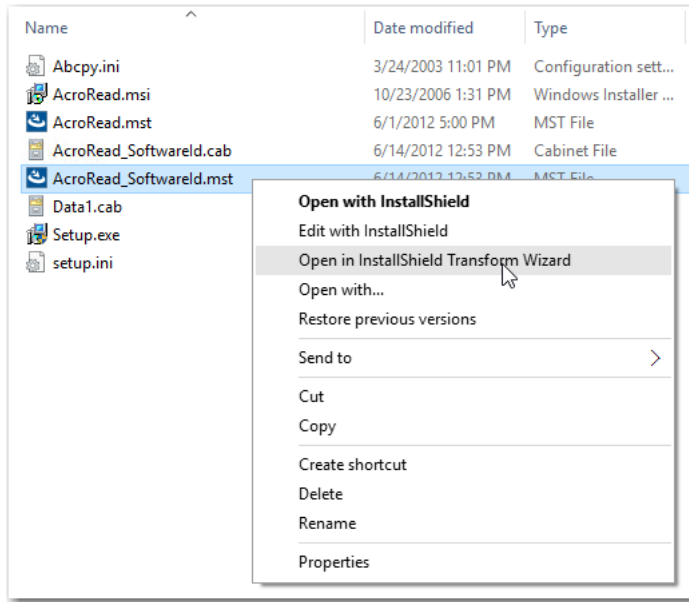
Figure 24: Open in InstallShield Transform Wizard Option

Note • This change was tracked in issue IOJ-1725656.

Additional Prerequisites Included

InstallShield 2018 includes the following additional prerequisites:

- Visual C++ 2017 x86 and x64 Prerequisites

- Microsoft SQL Server 2014 SP1 and SP2 Prerequisites

- Microsoft .NET Framework 4.7 Prerequisite

Visual C++ 2017 x86 and x64 Prerequisites

Because Microsoft Visual Studio 2017 has been released, InstallShield now includes the prerequisites for Visual C++ 2017 x86 and x64.

Note • This change was tracked in issue IOJ-1832110.

Microsoft SQL Server 2014 SP1 and SP2 Prerequisites

Because Microsoft SQL Server 2014 has had 2 Service Packs released, InstallShield now includes the prerequisites for both Microsoft SQL Server 2014 SP1 and SP2.

Note • This change was tracked in issue IOJ-1832297.

Microsoft .NET Framework 4.7 Prerequisite

InstallShield now includes a prerequisite for Microsoft .NET Framework 4.7.

Note • This change was tracked in issue IOJ-1834933.

Important Information

Note the following important information regarding the InstallShield 2018 release:

- 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

- Removal of .NET/J# Tab from the Releases View

- Current Directory Removed from DLL Search Path to Safeguard Against DLL Preloading Attacks

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 2018.

The version of the FlexNet Licensing Server software that shipped with previous versions of InstallShield cannot manage licenses of InstallShield 2018. InstallShield 2018 will not check out licenses from older version of the license server even when those licenses are available.

Note • The new version of the FlexNet Licensing Server software can manage licenses of both InstallShield 2018 and previous releases of InstallShield.

You can download the installer for the new FlexNet Licensing Server software from the Flexera Product and License Center.

For instructions on installing the FlexNet Licensing Server software, see Setting Up Your License Server for Concurrent Licenses of InstallShield 2016 and Later.

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.

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

The following installations are available for download from the Flexera Software 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 2018, the Collaboration Edition (DIM Editor) is no longer available.

Installing More than One Edition of InstallShield

Only one edition of InstallShield 2018—Premier, Professional, or Express—can be installed on a system at a time. In addition, the InstallShield 2018 DIM Editor cannot be installed on the same machine with any edition of InstallShield 2018.

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 2018 can coexist on the same machine with other versions of InstallShield.

The InstallShield 2018 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.

Removal of .NET/J# Tab from the Releases View

Project • This information applies to the following project types:

- *Basic MSI*
- *InstallScript MSI*

The **.NET./J#** tab in the **Releases** view of the Installation Designer was originally provided to support .NET 1.1/2.0 and J# redistributables. Those technologies have become obsolete and no longer supported by Microsoft. Therefore, in InstallShield 2018, the **.NET/J#** tab of the **Releases** view has been removed (as well as the associated .NET 1.1/2.0 Core Language and .NET 1.1/2.0 Language Packs dialog boxes).

Current Directory Removed from DLL Search Path to Safeguard Against DLL Preloading Attacks

To safeguard installations against DLL preloading attacks, InstallShield has removed the current directory from the standard DLL search path by calling the `SetDllDirectory` Windows API with an empty string (`""`).

If a DLL links implicitly to the other DLLs, or loads them dynamically using `LoadLibrary()` without specifying a fully qualified path name, the `UseDLL()` InstallScript function cannot load the dependencies from the current working directory.

You can work around this issue by prototyping `SetDllDirectoryW` (prototype number `kernel32.SetDllDirectoryW(wstring);`) and call it with `SUPPORTDIR` to get the support folder to be in the DLL load search path.

In InstallShield 2018, the following changes have been made:

- The `DLL_DIRECTORY_SUPPORTDIR` constant was added to the **Enable** function so that customers can explicitly opt-in to using `SUPPORTDIR` as a DLL directory.

- The `DLL_DIRECTORY_SUPPORTDIR` constant was added to the **Disable** function so that customers can explicitly opt-out to using `SUPPORTDIR` as a DLL directory.

- The `SetDllDirectory (szPathName)` wrapper function was added so that customers can explicitly opt-in to using any directory as a DLL directory. If the parameter is an empty string (`""`), the call removes the current directory from the default DLL search order.

Note • *This change was tracked in issue IOJ-1839017.*

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 2018. It also alerts you to possible changes in behavior that you may notice between new InstallShield 2018 projects and projects that are upgraded from InstallShield 2016 or earlier to InstallShield 2018.

- 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

If you use InstallShield 2018 to open an project that was created with an earlier version, InstallShield 2018 displays a message box that asks you if you want to convert the project to the new version. If you reply that you do want to convert it, 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. Note that you cannot open InstallShield 2018 projects in earlier versions of InstallShield.

You can upgrade projects that were created with the following versions of InstallShield to InstallShield 2018: InstallShield 2016 and earlier, InstallShield 12 and earlier, InstallShield DevStudio, InstallShield Professional 7 and earlier, and InstallShield Developer 8 and earlier. Note that projects that were created with InstallShield MultiPlatform or InstallShield Universal cannot be upgraded to InstallShield 2018.

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

For all project types except for Suites, Windows XP SP3 and Windows Server 2003 SP2 are the minimum versions of Windows that are required for target systems that run the installations that are created in InstallShield. For suites (Advanced UI, and Suite/Advanced UI project types), Windows Vista and Windows Server 2008 are the minimum versions of Windows that are required for target systems.

Localized String Considerations

Changes to the handing 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.
-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 2018 SP1

- InstallShield 2018

InstallShield 2018 SP1

The following issues have been resolved in InstallShield 2018 SP1.

Issue Number	Issue Summary
IOJ-1845413	An additional command line option (/clone_wait) is required for setup.exe to wait for the completion of the installation process.
IOJ-1868710	Upgrading QuickPatch project from InstallShield 2016 to InstallShield 2018, the project fails at the point of upgrading table InstallShield.
IOJ-1865668	Selecting the Upgrade Windows Installer Setup option, InstallShield crashes when the Upgrade Code Property is NULL.
IOJ-1864017	InstallShield Script Debugger window does not close, remains open after stepping through the debugger of the InstallScript custom action.
IOJ-1864015	Unable to upgrade to InstallShield 2018 that has a custom dialog theme and an IIS website associated with the project.

Issue Number	Issue Summary
IOJ-1863602	Unable to call the build method of the ISWiSuiteReleases object as the application crashes while the Suite contains an InstallShield Project file as a package.
IOJ-1862793	The InstallShield 2018 Skin Customization Kit downloaded from the Product and License Center does not work. The installer will work as the 2015 version, if InstallShield 2015 is installed on client.
IOJ-1862582	Unable to change the font (color, style, or size) in InstallShield 2018 Express. The ability to change the font (color, style, or size) of the Setup window does not work as expected.
IOJ-1861264	The project specific IDE crash occurs in InstallShield 2016 SP2 and InstallShield 2018.
IOJ-1860906	With or without a project open, if selected Tools > Options > Prerequisites tab, and then tries to add a new folder to the list to be searched for prerequisites, the field is blank.
IOJ-1860620	Adding an InstallShield project to a VS solution, receives an error when trying to build the solution with MSBuild.
IOJ-1860548	When components Key File is a DLL and .NET Scan in the build is set to Dependencies and Properties, the automation ISWiRelease build method crashes in PowerShell and VBScripts.
IOJ-1860335	In an InstallScript MSI project, when setting an MSI property with the MsiSetProperty function, with a value that contains a double quotation ("), the property ends up containing a value where the double quote appears twice in a row.
IOJ-1853555	Using a Store Certificate to sign swidtag ends in a build error.
IOJ-1851448	Including a dynamic link in Suite/Advanced UI, the static link for the selected package is not added to the project and thus receive a build error in InstallShield.
IOJ-1850455	If you delete a Key File from a component and then re-add the same file from a different location, Installshield will continue to show it as a Key File. After saving and reopening the project, Installshield shows that the file isn't a Key File.
IOJ-1845946	When using a 4k monitor, the default check box controls in a Basic MSI project do not display correctly.
IOJ-1844706	While launching InstallShield, setup.exe does not support using relative paths.
IOJ-1844374	Using "Required Features" Ellipses on a feature in Installscript project, Installshield freezes indefinitely.
IOJ-1844013	InstallScript Compiler crashes when the script assigns array values to Data Type Not Declared as an array.
IOJ-1843628	When you run a compressed InstallScript project, the copied installer is named setup.exe; regardless the name of the launched installer.

Issue Number	Issue Summary
IOJ-1843590	After adding a PowerShell Custom Action to a Basic MSI project and sequencing the action, if no script file is specified for the Custom Action, InstallShield will build the project without an error. The built installer fails at run-time.
IOJ-1842874	While the setup is completed, all the files are cleaned in TEMP folder; Except for a single Basic MSI media with feature prerequisite leaves two setup.exe in the temp.
IOJ-1842166	The DialogSetInfo() function is used with the DLG_INFO_ALTIMAGE parameter to display an alternative logo image on UI dialogs. When upgrading to InstallShield 2018 and rebuilding the installation, the logo images are displayed improperly.
IOJ-1835755	Both the parent installer and feature prerequisites fail to install when the parent includes an install condition, checking for pending reboot property on the system.
IOJ-1830023	While the setup is completed, all the files are cleaned in TEMP folder; Except for a single Basic MSI media leaves two setup.exe in the temp.
IOJ-1813407	Running parallel builds simultaneously for merge module projects, an error is thrown - Unable to delete a directory <project path>\<project name>\Build\Interm\MergeModules.
IOJ-1731438	The registry view in the InstallShield IDE has a limit on what it will display as the value for a DWORD. Entering anything over 2147483647, hexadecimal 0x7FFFFFFF is displayed as 0x7FFFFFFF in the IDE.
IOJ-1665230	Setting INSTALLDIR from the Registry does not search 64-bit portion of registry and needs to be read from x64 location.
IOJ-1623965	If an InstallScript MSI project contains any file, ISBEW64.exe is extracted to a temp. Even after the installation or uninstallation of the product, the file is not cleaned from the TEMP folder.
IOJ-1596681	When a function is deleted from the Script Editor pane, the Script Editor in an InstallScript project retains the deleted functions as available in the setup.rul.

InstallShield 2018

The following issues have been resolved in InstallShield 2018.

Issue Number	Issue Summary
IOJ-1600400	When creating a new QuickPatch project based on a previous QuickPatch project, and selecting a QuickPatch ISM that is in XML format, the Create New QuickPatch wizard displays the following error message: `<ProjectName>.ism is not a valid ism file. Please specify a valid QuickPatch project (.ism file) in the Existing QuickPatch Project field.`

Issue Number	Issue Summary
IOJ-1625736	After using the New Language Wizard to add a new language to an InstallScript project, building with the InstallShield IDE successfully adds the new language to the media. However, performing the build on the Standalone Build System fails to add the language to the media, even after adding the new language to the Standalone Build.
IOJ-1627091	Dependent files are not being included in the InstallShield package when building an InstallShield project with MSBUILD. However, dependent files are included when building the same project using the Visual Studio interface.
IOJ-1749409	When you attempt to use COM extraction in InstallShield on a poorly written DLL, it will get stuck in an infinite loop during registration. InstallShield will hang indefinitely instead of stopping the process and displaying an error message after a certain amount of time has elapsed.
IOJ-1773142	The features that install JRE50 and JRE7 in the InstallShield 2016 Prerequisite installer display incorrect names and descriptions. The feature name for JRE50 is displayed as `ID_STRING102` and the feature name for JRE7 is displayed as `ID_STRING94`. The names of the features to be installed should contain a descriptive name reflective of the files that will be installed for that feature.
IOJ-1777320	Specifying a Wizard Lite Wizard Format for the Suite UI displays a white strip along the top of each wizard page. The title and header image appears to be removed.
IOJ-1807823	When customer tries to use a PowerShell custom action that uses the Import-Certificate cmdlet to install a certificate into the certificate store, the script works as expected outside of InstallShield, but when used in a PowerShell custom action, this cmdlet appears to do nothing.
IOJ-1808966	In an InstallShield 2016 project, `XCopyFile` throws error 0x80070020. In an InstallShield 2014 project, `XCopyFile` returns 0.
IOJ-1812514	When installing the FlexNet License Server (11.13.1.5) for use with a concurrent license, if you specify different port numbers for use with the HTTP port and the License Server Manager port, the ports are not saved and are reverted back to the default ports (8090 for HTTP and 20071-79 for License Server Manager port).
IOJ-1813258	Microsoft SQL Server 2016 Express prerequisites fail to download.
IOJ-1815650	When using **Text Replacement** in the **SQL Scripts** view, you are unable to replace a string with the same value but different case sensitivity. For example, finding the string `testvalue` and replacing it with `TestValue` fails; the script retains the value of `testvalue`. This occurs with the **Preserve case** option unselected. This behavior would be expected if **Preserve case** was selected.

Issue Number	Issue Summary
IOJ-1817063	InstallShield 2016 SP1 setup.exe installer can use `LAAW_OPTION_WAIT_INCL_CHILD` as a parameter for the original setup to wait for the termination of the clone process. For more information, see the following article: LaunchAppAndWait nOptions Parameter When Using InstallShield 2016 SP1
IOJ-1817718	The Direct Editor displays a red signature column in the `RegLocator` table. When a second system search is configured that uses the `RegLocator` table instead of the Signature table, the **Signature** column in the `RegLocator` table is red.
IOJ-1822716	When a new shortcut is created from the Project Assistant view, InstallShield automatically generates a short file name for the newly generated shortcut. If the newly generated shortcut is renamed from the Project Assistant view, a short file name is not generated for the renamed shortcut.
IOJ-1823277	Calling a Basic MSI setup launcher from a command prompt and including parameters where the entire command exceeds 752 characters results in a "Setup Launcher Unicode Has Stopped Working" error. Although the error is seen, the MSI still launches and does contain all of the properties passed on the command line.
IOJ-1823375	After applying a hot fix to an older version of InstallShield, using a relative path on the command line causes the setup to fail to launch.
IOJ-1823621	When adding a scheduled task to a merge module project and including it in a Basic MSI project, build warning -6524 occurs.
IOJ-1823819	Opening an InstallShield 2015 SP1 Basic MSI project in a Visual Studio 2015 solution causes hang and/or problems with the Visual Studio interface. The Solution Explorer becomes unresponsive.
IOJ-1824096	The `InstallShieldPrereqs_SQLServer.exe` installer installs the SQL 2016 Express prerequisites to the SQL 2014 Express location, replacing the 2014 prerequisites.
IOJ-1825341	When an MSI file is converted to an InstallShield project, the resulting `.ism` file does not reflect the **Always Overwrite** file settings that were set in the original MSI file.
IOJ-1826101	When a suite project is run on Windows Server 2016, an error is encountered when attempting to enable an IIS Windows feature: runtime error 0x800f080c.
IOJ-1828662	The `XCopyFile()` function is unable to copy a locked file when the `LOCKEDFILE` constant is passed. The message "Access is denied" is displayed.
IOJ-1829226	If the `ntmarta.dll` file is located in the same folder as an InstallScript single .exe installer, it will cause a Unicode crash during setup initialization.
IOJ-1830045	Checking for updates in InstallShield 2016 SP2 causes Software Manager to display "The product version is not registered on the server" instead of displaying the message that no updates are available.

Issue Number	Issue Summary
IOJ-1830667	Changing the brush color for the footer brush in a Suite project does not update the footer color for dialogs displayed in the InstallShield interface after a dialog has been viewed. However, the correct color is displayed: 1) if the user saves the project, then closes InstallShield and opens the project again, and 2) if the user changes the brush color prior to viewing a Wizard Page dialog in the IDE. The correct color is displayed at runtime; this is an interface problem.
IOJ-1831676	A PowerShell custom action that uses the `New-SelfSignedCertificate` cmdlet to create a certificate in the certificate store of the local machine does not behave as expected. The script works as expected outside of InstallShield, but when used in a Suite PowerShell custom action, the cmdlet does not create the certificate.
IOJ-1833474	A suite installer that includes the Microsoft VC++ 2015 Update 3 prerequisite will crash on Microsoft Server 2008 if Microsoft VC++ 2015 Update 3 is already installed on the system.
IOJ-1833599	When you include a right-to-left language and run a patch or minor upgrade (setup.exe), runtime error 2721 occurs based on binary table data.
IOJ-1834589	InstallShield crash at build exception ACCESS VIOLATION (0xc0000005) at address `<foo>` trying to read address `<dat>`.
IOJ-1835866	After applying an InstallShield hot fix, InstallShield MSI installers do not cache all language files, causing panels to display in the default language when launched from Add/Remove programs.
IOJ-1836734	InstallScript MSI projects that use the new style InstallScript user interface type fail at runtime. The actual behavior varies slightly depending on if a setup launcher is included or not. In both cases the package does get installed.
IOJ-1838239	Starting with InstallShield 2016, the Build Report Summary does not update any of the numbers for features, merge modules, prerequisites, dynamic links, or files. These all display 0. The number of components was also removed.
IOJ-1839017	Change in InstallShield 2015 SP2 / 2016 is causing an issue with the `UseDLL()` function.
IOJ-1839820	When using a differential media release for an upgrade, it can cause the "InstallShield Installation Information" folder to be missing the cached package, causing problems with repairs of missing files.
IOJ-1842270	SETUPEXEDIR does not resolve to the expected value when passed to an InstallShield prerequisite.
IOJ-1842577	Upgrading a Suite project from a previous version of InstallShield will remove the value from the string `IDS_SUITEERROR_ABORTCONDITION`.
IOJ-1843630	If a compressed Suite is larger than 1.5GB, and has Windows features associated with a package, the Suite install will fail when run on Windows 7 or Windows Server 2008 R2. If the Suite is not compressed, or is smaller in size, this issue will not occur.

Issue Number	Issue Summary
IOJ-1844034	Suite run using system context results in MSI parcel launch result: 80070003.
IOJ-1844673	If the `cryptbase.dll` file is located in the same folder as a `setup.exe` installer file, it will be loaded during setup initialization.

IOJ-1846227	Request that a registry convention be used as a condition for the Microsoft Visual C++ 2015 Update 3 redistributable instead of the `msvcp140.dll` file, which could be unreliable.
	
	The currently installed version is stored in this key:
	
	`HKEY_LOCAL_MACHINE\SOFTWARE[\Wow6432Node]\Microsoft\VisualStudio\`
	`    vs-version\VC\Runtimes{x86|x64|ARM}`
	
	For example, C++ 2015 Update 3 x86 is stored in:
	
	`HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432\NodeMicrosoft\VisualStudio\14.0\VC\`
	`    Runtimes\x86\Version=v14.0.24212.00`
	
	For more information, see Redistributing Visual C++ Files.

IOJ-1849090	Enabling package logging support in a Suite installer when parcel is a patch (`*.msp`) does not work as expected; it causes a GUID ID name problem.
IOJ-1854157	Inline help for **Display Name** setting in a **Shortcuts** folder incorrectly suggests using `<SHELL_OBJECT_FOLDER>` in InstallScript MSI projects.

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

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	256 MB of RAM (512 MB preferred)
Hard Disk	750 MB free space
Display	Designed for XGA resolution at 1024 × 768 or higher

Item	Description
Operating System	- Windows Vista - Windows Server 2008 - Windows 7 - Windows Server 2008 R2 - Windows 8 - Windows Server 2012 - Windows 8.1 - Windows Server 2012 R2 - Windows 10 - Windows Server 2016
Privileges	Administrative privileges on the system
Mouse	Microsoft IntelliMouse or other compatible pointing device
Optional Integration with Visual Studio	The following versions of Microsoft Visual Studio can be integrated with InstallShield Premier or Professional Editions: - Visual Studio 2008 - Visual Studio 2010 - Visual Studio 2012 - Visual Studio 2013 - Visual Studio 2015 - Visual Studio 2017 The following editions of these versions of Visual Studio can be integrated with InstallShield Premier or Professional Editions: - Community - Professional - Premium - Ultimate

For Target Systems

For all project types except for Suites (Advanced UI, and Suite/Advanced UI project types), Windows XP SP3 and Windows Server 2003 SP2 are the minimum versions of Windows that are required for target systems that run the installations that are created in InstallShield. For Suites, Windows Vista and Windows Server 2008 are the minimum versions of Windows that are required for target systems.

Target systems must meet the following minimum operating system requirement:

- Windows XP SP3

- Windows Server 2003 SP2

- Windows Vista

- Windows Server 2008

- Windows 7

- Windows Server 2008 R2

- Windows 8

- Windows Server 2012

- Windows 8.1

- Windows Server 2012 R2

- Windows 10

- Windows Server 2016

Target systems must also support the SSE2 instruction set.

Known Issues

For a list of known issues, see https://flexeracommunity.force.com/customer/articles/en_US/INFO/InstallShield-2018-Known-Issues.

Legal Information

Copyright Notice

Copyright © 2018 Flexera. All Rights Reserved.

This publication contains proprietary and confidential information and creative works owned by Flexera 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 is strictly prohibited. Except where expressly provided by Flexera in writing, possession of this publication shall not be construed to confer any license or rights under any Flexera intellectual property rights, whether by estoppel, implication, or otherwise.

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

Intellectual Property

For a list of trademarks and patents that are owned by Flexera, see https://www.flexera.com/producer/company/about/intellectual-property/. All other brand and product names mentioned in Flexera 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.