

InstallAware X12

Reviewer's Guide

June 2020



The information contained in this document represents the current view of InstallAware Software on the issues discussed as of the date of publication. Because InstallAware must respond to changing market conditions, it should not be interpreted to be a commitment on the part of InstallAware, and InstallAware cannot guarantee the accuracy of any information presented after the date of publication.

This guide is for informational purposes only. INSTALLAWARE MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE INFORMATION IN THIS DOCUMENT.

InstallAware may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from InstallAware, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2020 InstallAware Software. All rights reserved.

InstallAware, MSIcon, Genuine Scripting for Windows Installer, WebAware Installations, Web Media Blocks, Partial Web Deploy, and One-Click Patching are either registered trademarks or trademarks of InstallAware Software in the United States and/or other countries.

Microsoft, Windows, and other Microsoft product names are trademarks or registered trademarks of Microsoft Corporation in the U.S. and other countries. All other marks are the property of their respective owners.

Contents

Overview	4
Introduction	4
Product Description	4
Contact Information.....	4
InstallAware Highlights	5
New and Interesting Features of InstallAware X12	5
Unique Features and Key Benefits for Developers	7
InstallAware X12 Two-Way Integrated IDE	8
What Types of Setups are Created with InstallAware X12?	9
Setup Development.....	11
Projects in InstallAware X12	12
Native Code Setups in InstallAware X12	13
Deploying Setups.....	14
Additional Advanced Tools in InstallAware	15
Basic Introduction	20
About InstallAware Software.....	21

Overview

Introduction

Thank you for your interest in reviewing InstallAware X12, the cutting-edge tool for software installation authoring, application repackaging, and software virtualization. This guide will give you an overview of some key benefits that are available only with InstallAware.

Product Description

InstallAware X12 is the fastest and most scalable way to deliver ultra-rich, ultra-fast Windows installations. Dramatically reduce coding time and create installers tens of times faster with MSIcode based development and a fully visual two-way RAD IDE. Completely eliminate the need for Windows Installer training, save on bandwidth and support costs, and build smart installers which adapt to their environment at runtime.

InstallAware X12 also includes a vast array of application virtualization options, including MSIX, AppX, App-V, and export to InstallAware Virtualization. Last but not least, InstallAware X12 contains the fastest setup capture technology of the industry – helping you funnel your legacy code and/or third party applications into either the most modern installers, or into industry standard virtualization/containerized packages.

Contact Information

We are here to support you throughout your evaluation period – and beyond. Please do not hesitate to contact us anytime.

Technical Assistance: support@installaware.com

Sales Inquiries: sales@installaware.com

InstallAware Highlights

New and Interesting Features of InstallAware X12

- **Ethereum Blockchain Payments** – InstallAware X12 is the first and only installer to enable cryptocurrency payments. Through its **trustless Smart Contract authored in Solidity**, ETH funds are transferred **securely and discreetly** at install-time directly from end-users to developers, in 15 seconds or less. **Lookups for past purchases** is also virtually instant, facilitating re-installations of past purchases with ease.
- **Signed Web Media Blocks** – InstallAware X12 now code signs its web deployed bits, and verifies both the signature, and the signing entity. This helps **eliminate a new source of attacks where third parties obtain legitimate code signing certificates and re-sign inauthentic code!** Especially in view of the decentralized download capability introduced one version earlier via Torrent files/Magnet links, this new security verification feature is of vital import and very timely.
- **New Feature and Server Role Updates** – Our team painstakingly maintains 100% accurate inventories for each Windows platform, including **X86, AMD64, and AARCH64**; letting you configure the operating system components of all your targets with laser-guided precision to mold the end-user operating system into the exact state your setup requires. Even better, you configure everything from a single dialog mimicking the native **“Turn Windows features on or off”** dialog.
- **User Mode Installs** – New in InstallAware X12 is a **Limited User Account setup template**, which installs, repairs, maintains, and uninstalls without ever requiring elevation or administrator intervention. Also new is the facility to **register file types per-users**, in addition to per-machine.
- **Azure DevOps Integration** – InstallAware X12 enables globally distributed teams and working from home with in-IDE integration for **Team Foundation Server, Visual Studio Team Services Azure DevOps Server 2019 (including Update 1.1), and Azure DevOps Services**. InstallAware X12 now includes a brand-new **Team Project Explorer**.
- **Advanced String Manipulation** – InstallAware X12 introduces string length, substring match, and **Regular Expression** support into its scripting language for convenient user input validation.

- **Windows 10 2004 Update Eco System** – InstallAware X12 includes the latest in application runtimes and technology support, including redistributables for **.NET Framework 4.8**, SQL Server Express 2019, the **Visual C++ 15.9** runtime, IIS Express, Crystal Reports, and more. The new setup engine fully detects all Windows versions, especially the myriad releases of Windows 10 - which are **already ten in number!**
- **Acrylic Material** – InstallAware X12 **remains the only installer with support for Microsoft Fluent Design System**. The **Acrylic Theme** showcases this support with textured noise patterns and breathtaking translucency in setup dialogs. You may now create your own setup dialogs and themes based on Acrylic Material as well.
- **Dialog Editor Undo/Redo** – InstallAware X12's Dialog Editor has virtually unlimited undo/redo (limited only by available memory). Experiment freely with dialog designs and enjoy the ability to revert any changes or modifications at any time. New in InstallAware X12's Dialog Editor is the ability to directly invoke the **Intra-Dialog Relations** and **Dialog Behavior** screens.
- **Instant Extraction** – InstallAware X12 setups, even those with multi-gigabyte payloads, launch instantly, without full pre-extraction of the setup payload. **The setup script boots instantly along with the full setup user interface**. This enables you to present the setup interview in its entirety, before starting time consuming tasks such as payload extraction and data transfer – dramatically improving your end-user experience.
- **Advanced Compression Engine** – InstallAware X12 features a 64-bit compression engine with advanced heuristics, crunching data as tightly as possible using tens of gigabytes of RAM; while retaining extraction compatibility with 32-bit platforms having as little as just one gigabyte RAM!
- **Application Pinning** – InstallAware X12 is the only Windows Installer with automatic, programmatic pinning of your applications to the new Windows 10 Start Menu in the live tiles section, and/or to the Windows 10 Taskbar.
- **Visual Studio Add-Ins** – InstallAware X12's Visual Studio Add-Ins let you do even more without ever having to launch the stand-alone InstallAware IDE. Fully control the files, assemblies, and folders going in to your installation. Create registry keys and values. Perform code signing. InstallAware's Visual Studio Add-Ins integrate with Visual Studio versions **2019, 2017, 2015, 2013, 2012, 2010, 2008, 2005, and 2003**.

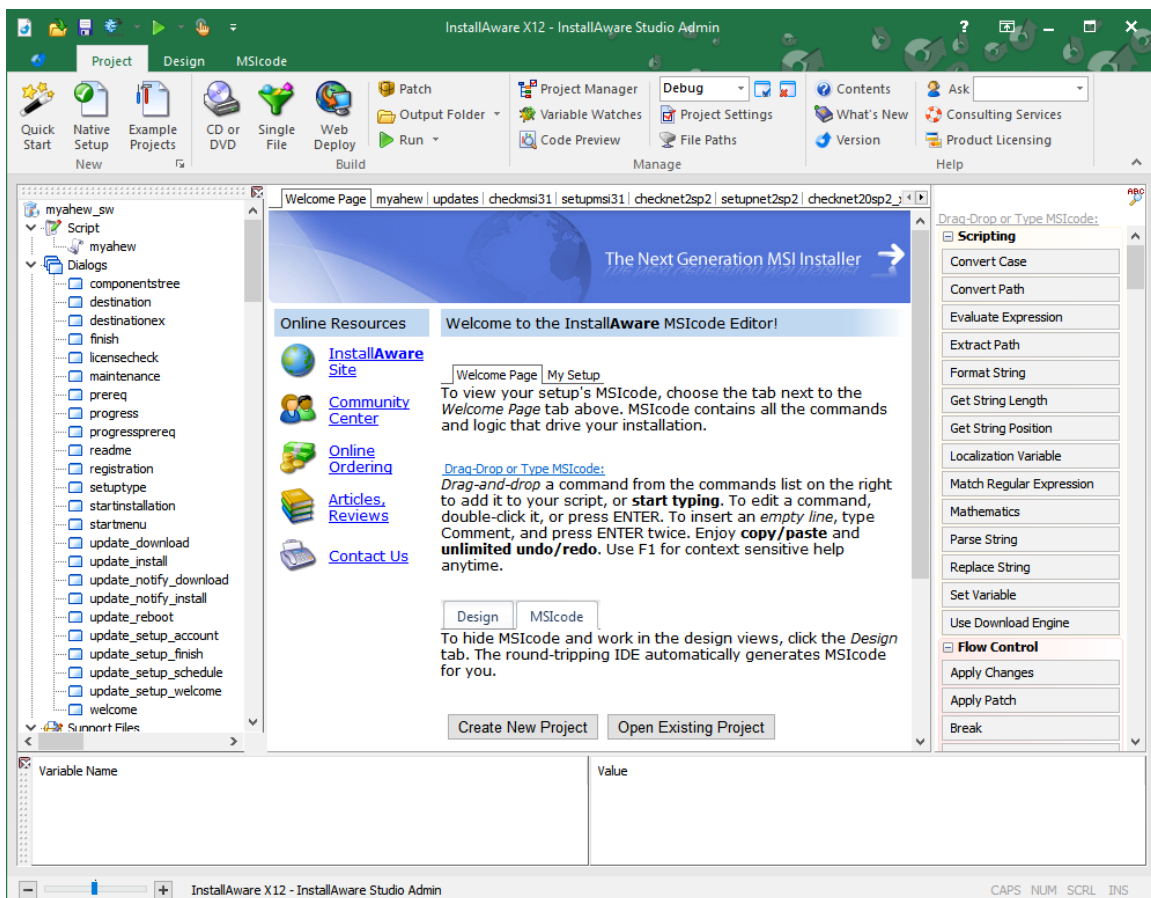
Unique Features and Key Benefits for Developers

- Built on an excellent abstraction of Windows Installer called MSIcode. MSIcode is a human readable, conditionally flowing setup script, similar to a batch file. Your MSIcode script is automatically compiled into a logo compliant MSI database at build time, eliminating all need for Windows Installer training and the need to understand its complex relational database structure.
- MSIcode is more than just another descriptive language for creating setup packages: it is a layer on top of Windows Installer which directly modifies Windows Installer's runtime behavior.
- No scripting engine is required or pre-installed at runtime. Setups you build run on all 32-bit and 64-bit platforms, from the first release of Windows XP to the latest Windows Server 2019 64 bit – all with the same setup binary.
- Your MSIcode script “magically branches” according to your underlying scripted logic at runtime. You may even create/edit/delete your setup features at runtime, based on system conditions or online live license authentication. This helps you build and maintain a single installer (and a single setup project) for all of your product editions, languages, and features.
- Securely download optional application components using Partial Web Deployment. Segment your setup files into multiple online and offline Web Media Blocks. Optional Web Media Blocks are downloaded only if necessary. A partially web deployed setup does NOT mandate an Internet connection if online Web Media Blocks have not been selected for installation. Web Media Blocks may be shared by more than one setup and are highly compressed.
- One-Click Patching: No special patch project types are required to create patches. Point to the built binaries of your old setup versions to build patches in a single click. Use MSIcode scripting to customize patch behavior at runtime and perform additional custom tasks.
- Deploy an unlimited number of web updates to your installed customer base. Filter patches using MSIcode to preserve your business logic.
- “Shell” to Third Party Setups from your installations. Capture and display native, interactive installation progress in your own setup dialogs – never launching external setup windows means your dependencies are always installed with the correct parameters and as an integral part of your master installation.
- Design stunning user interfaces with the 43 built-in dialog controls. Easily pass MSIcode variables between your setup dialogs and script. Capture and submit business intelligence directly within your (un)installations, without risking a click-away on web pages.
- In addition to runtime setup customization in MSIcode, use MSIcode Compiler Variables to conditionally include/exclude files and logic from your setups at build time.

InstallAware X12 Two-Way Integrated IDE

InstallAware X12 is the only way to deliver the fastest, most scalable, and most reliable Windows installations with the richest user interfaces. The two-way integrated IDE dramatically reduces coding time. The Design tab rapidly prototypes your setup, while the IDE seamlessly emits MSIcode in the background. Switch to the MSIcode tab and inject custom code exactly where needed.

The visual and script parts of InstallAware projects are NOT disjoint, separate entities. Unlike other installation tools, the MSIcode script drives the entirety of your installation, has no hard-coded elements, and is automatically emitted for you by the IDE as you use design views. This means that when you need to drop down to the code view after doing a lot of visual design work, everything you have done so far is immediately accessible and editable. The script is not a separate entity that is divorced from the actual files, registry keys, and other elements that comprise your installation.



What Types of Setups are Created with InstallAware?

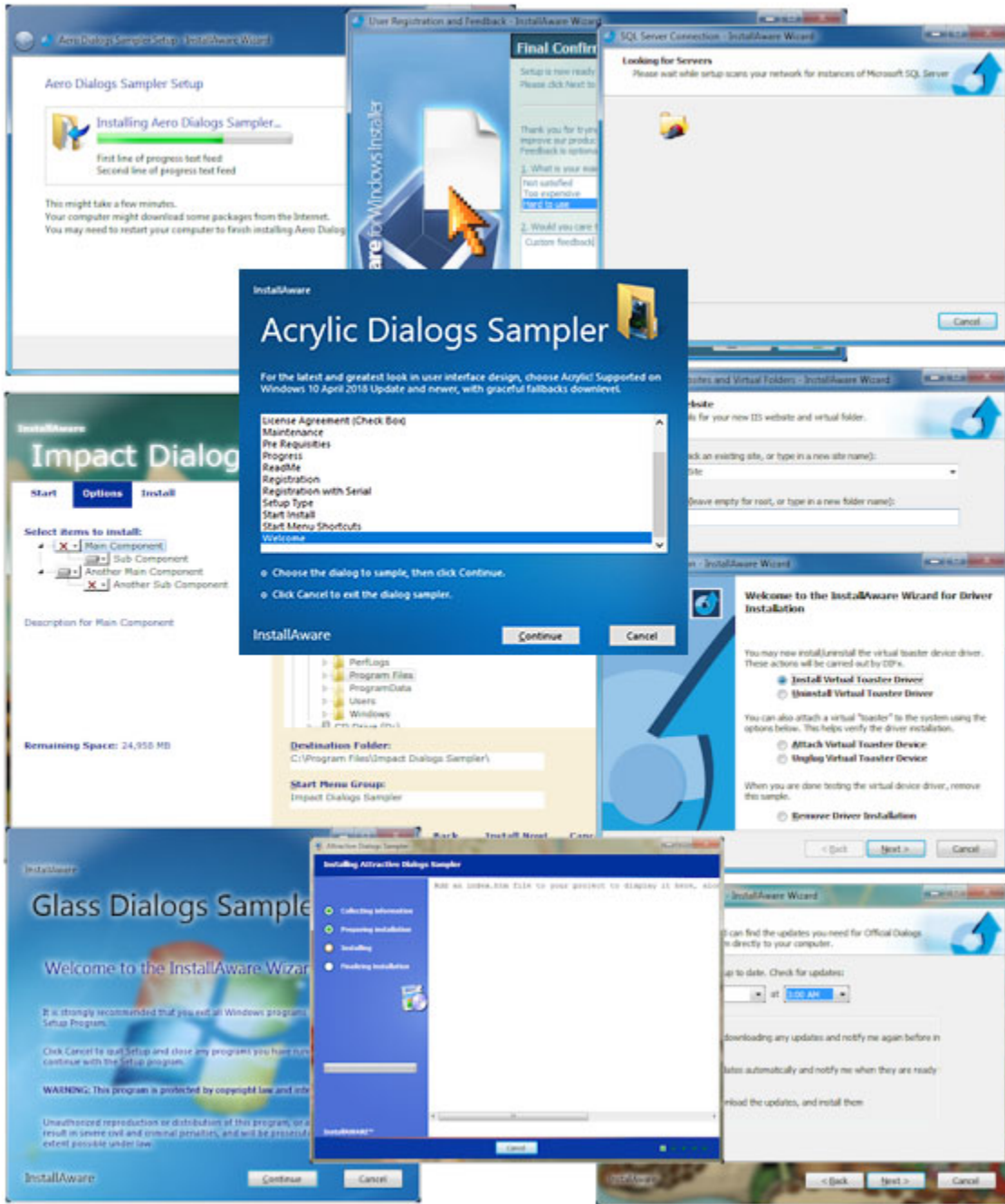
Mission-critical enterprise installs to complex desktop applications, repackaged, virtualized, or combined suite installations, driver installers, software development tools, and games; OEM, web, and database applications. Some hit commercial games like Crysis and compilers like Delphi, vertical software such as Native Instruments' Traktor, and OEM setups by Sony and BestBuy have been built using InstallAware.

InstallAware X12 unlocks the potential of the developer's imagination and allows them to focus on the solution without having to worry about the underlying confines and limitations of Windows Installer. InstallAware saves developers countless hours of Windows Installer training, and especially the frustration of having to work around the straitjacket setup authoring experience that is found with other environments.

The Native Code Setup Engine in InstallAware X12 exponentially increases the runtime capabilities of setups, completely side-stepping all limitations of Windows Installer, whilst not requiring platform exclusivity – setups may even switch between Native Code and Windows Installer setup engines at runtime, as many times as needed. InstallAware's Native Engine supports extra-long file system paths, reads from and writes to protected file system and registry locations while retaining their security settings, runs in Windows Safe Mode where Windows Installer is disabled, performs setups an order of magnitude faster than Windows Installer itself, and even supports multiple simultaneous installations (which is, again, something Windows Installer itself is not capable of doing).

Because InstallAware generates standalone Windows executables without large and cumbersome runtime dependencies like Java or .NET, or script engines like InstallScript that need to be pre-installed (often failing with cryptic error messages), InstallAware setups always work on the largest possible number of target systems, out of the box, making sure your relationship with your customers starts off on the right footing. Now with the new wizard transition special effects in InstallAware X12, your setups are as visually pleasing as they are solid under the hood.

New in InstallAware X12 is the industry's sole answer to the problem plaguing PC users since time perennial: Having to click through endless Next dialogs to install software. The new Single-Click Setup Template installs in literally a single click and is modeled after Microsoft's MSIX Installer user interface, for the ultimate convenience in software installation and removal.



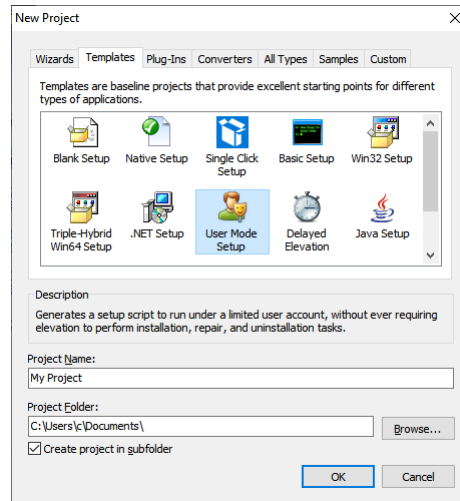
Setup Development

After installing InstallAware, you will have the opportunity to decide what type of development you want to do, and whether you wish to self-publish your solutions, or submit them to the Windows Store.

- Setup repackaging based on system snapshots or direct capture
- Setup transformation based on installation tailoring with MST creation
- Direct import of Windows Installer setups and merge modules
- Direct import of InstallShield, Wise, and other popular setup projects
- Highly custom setups started with a completely empty project
- Setups that use the InstallAware Native Code Setup Engine
- Setups that use the Windows Installer Setup Engine
- Setups that use both engines with HIT (Hybrid Installation Technology)
- Setups delivering payloads to native X86, AMD64, and AARCH64 locations from a single binary
- .NET and Java application installers, with optional download and installation of their frameworks
- Web update clients
- Multiple instance setups (installing more than one copy of the same application simultaneously)
- Obtaining end-user feedback, collecting and submitting business intelligence
- Validating serial numbers, both offline and online
- Localized setups
- Database installations that configure a named database instance and run a SQL script against it
- Web server installations that configure a web server and create websites/virtual folders on it
- Driver installations
- Setups which optionally download features from the Internet while NOT mandating a connection
- Setups dynamically defining and deleting setup features at runtime, on end-user machines
- Binary byte level patches
- Setups which securely and discreetly process cryptocurrency payments at installation-time
- Custom plug-ins which enhance the existing capabilities of the InstallAware MSIcode language
- Calling functions from any system DLL or custom DLL directly
- Bootstrappers which seamlessly install third party application frameworks and runtimes
- Checking for and removing old product versions, including those installed without InstallAware
- Checking for and installing technology frameworks, runtimes, and database/web servers
- Delivering dynamically generated product bundle offers with your applications
- And many others

Projects in InstallAware X12

Once you start InstallAware X12, the first thing you will need to do is create a project.



This can be accomplished by clicking the InstallAware button and clicking the *New Project | Other* menu item. This is the setup solution gallery and it displays all of the different types of solutions that can be created. If you are not reviewing the Studio Admin edition with all features installed, some solutions may not be available – just run InstallAware setup again and choose to do a full install of Studio Admin.

- **Wizards** – A list of standard wizards which guide you step by step through setup project creation, including wizards which let you convert pre-existing setups (without access to their source codes).
- **Templates** – InstallAware projects which can be fleshed out in the IDE, including Win32, Win64, .NET, and Java application support; as well as highly custom applications such as Autorun apps.
- **Plug-Ins** – Create plug-ins that integrate with InstallAware’s MSIcode script, including rendering a custom line of script and participating in the setup build process. You may also use managed languages such as .NET to create InstallAware MSIcode plug-ins.
- **Converters** – Upgrade existing setups (for which you do have the source codes) into ready to build InstallAware projects in a single click.
- **Samples** – Various sample projects illustrating useful InstallAware programming concepts.
- **Custom** – Promote any existing InstallAware project into a generic template, maximizing code reuse and enforcing your organization’s installer standards. Inherit new projects from your self-made setup templates.

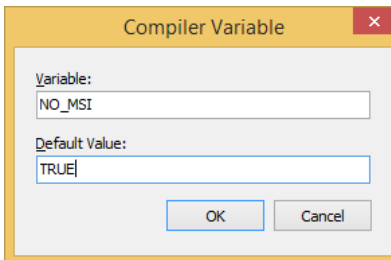
Native Code Setups in InstallAware X12

If you have previously used InstallAware, you will be pleased that all it takes is a single line of code to upgrade your existing projects to use the Native Code Setup Engine instead of Windows Installer:

```
1  
2  Comment: Enable Native Engine  
3  Set Variable NATIVE_ENGINE to TRUE  
4  
5  Comment: Enable Windows Installer Engine  
6  Set Variable NATIVE_ENGINE to FALSE  
7
```

As in the code example above, changing the value of the **NATIVE_ENGINE** variable between **TRUE** and **FALSE** will toggle any installation between the Native Code and Windows Installer setup engines. This process may be repeated as many times as required in your installation.

InstallAware does not lock you into any particular setup authoring platform and gives you the power of choosing whichever setup engine is best suited for your requirements. Best of all, your existing scripts – all of them – are upgraded to work with the Native Engine in just one line of code!

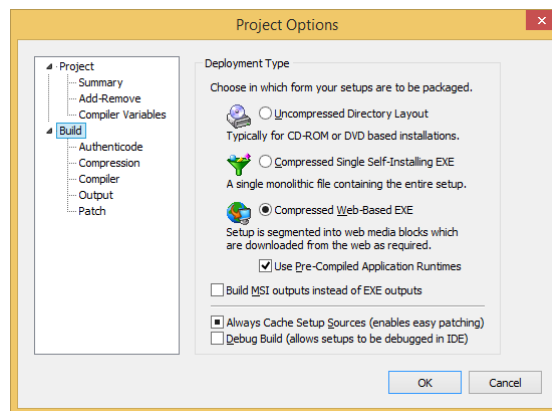


If you are sure that you will not be using the Windows Installer engine at all, you may prefer to skip building MSI files as part of your build process. This will reduce the size of your setup payload and speed up the build process significantly. Just set the **NO_MSI** compiler variable to **TRUE**.

Please see the help file topics *Pre-Defined Compiler Variables* and *Pre-Defined Variables* for many additional build time and runtime tweaks that can be applied to your setups for increasing performance and reliability. Remember, nothing is hard-coded in InstallAware!

Deploying Setups

InstallAware offers a variety of build modes for your existing projects, all based on the same source files. These settings are accessible in the *Project Options* window.

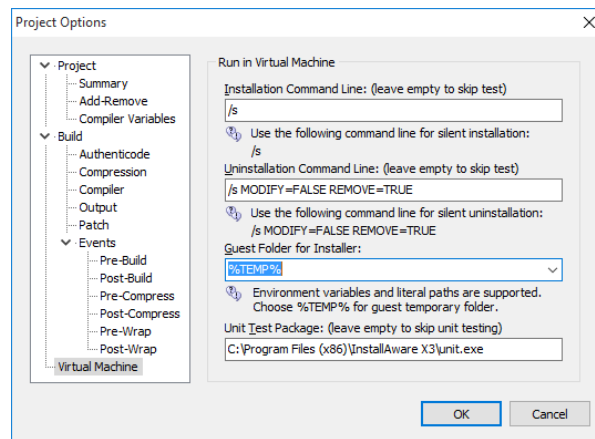


- **Uncompressed Directory Layout** is the fastest build option and ideal to use when you are coding and debugging your setups. It is also the best build mode to use if you will be distributing your setups on optical media or thumb drives, or other medium which can contain multiple folders.
- **Compressed Single Self-Installing EXE** takes the uncompressed output and packs it all inside a single, self-extracting executable. This is ideal for cases when a single monolithic file is required. All files are compressed to the smallest possible size.
- **Compressed Web Based EXE** is ideal for online distribution. The output is a compressed self-extracting executable, as well as several 7zip files which are the web media blocks containing your optional setup features. Unlike other web deployment schemes, the 7zip files are not mandatory and will be downloaded only if they are selected by the end-user for installation (and have not been previously downloaded). You get to precisely choose which source files are placed inside web media blocks, and which go inside the self-extracting “offline” setup executable.
- **Patch** is used to build your existing setup project as a patch, without having to create a new patch project type. Through the use of compiler variables, parts of your MSIcode script are conditionally included or excluded when building setups as patches. This helps you inject additional business and setup logic to the patch process whilst avoiding the need to create a new patch project.
- **Build MSI outputs instead of EXE outputs** converts any of the above EXE outputs into a Group Policy/Active Directory compatible MSI file, even when your original setup is not based on Windows Installer technology and uses the Native Engine instead.

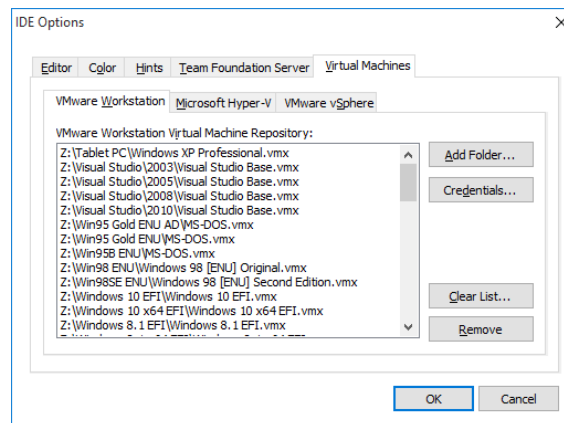
Additional Advanced Tools in InstallAware

InstallAware X12 includes many features beyond those covered in this Reviewer’s Guide. There are a few that deserve special mention, just to remind you, the reviewer, of the power of these tools and the value they add to the InstallAware setup authoring environment.

- Automated Virtual Machine Testing** – Just double-click any target virtual machine snapshot in the Project Manager window, and InstallAware launches the selected guest operating system and initiates a unit test with your desired command line. Launch any additional number of tests and watch them take place simultaneously.

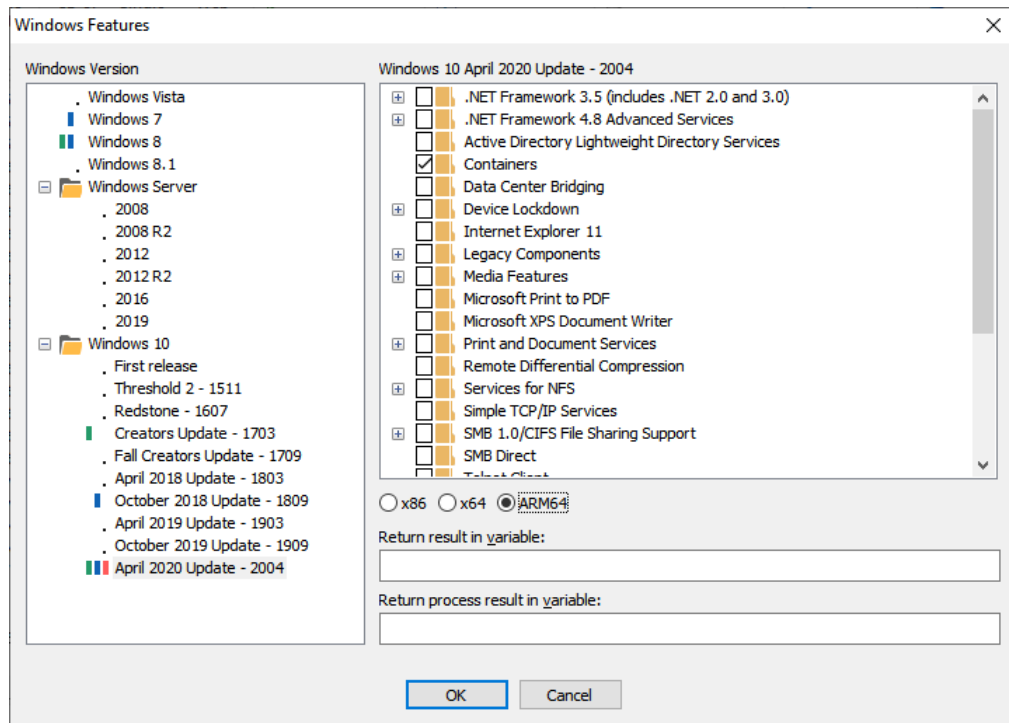


Manage your repository of VMware Workstation, VMware vSphere, and Microsoft Hyper-V virtual machines in the InstallAware IDE. Testing has never been so convenient before.



If you would like to monitor for your own custom installation targets, such as successful creation of particular services, files, registry keys, and any other measurable items; simply edit the default unit test script that ships with InstallAware. Use your custom unit test to objectively prove your installations on any chosen virtual machine snapshots.

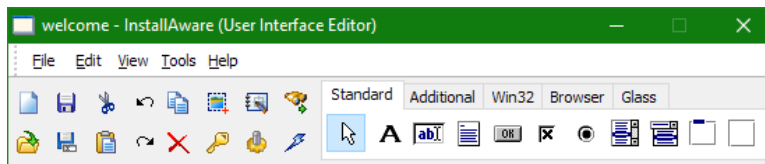
- **ETH Payments** – InstallAware X12’s brand-new **ETH Transaction** plug-in either makes a first time Ethereum transaction, or looks up the record of a past transaction stored on the blockchain, facilitating re-installations of past purchases. The **Smart Contract** has been authored in Ethereum’s native **Solidity** programming language, enabling a trust-less, secure, and discreet payment; directly from your end-users to your ETH wallet, in fifteen seconds or less. No separate browser or bulky user interface is launched at any time – everything happens in your setup script!
- **Windows Features** – Hand-pick features to add to your target Windows client or server operating system, visually configuring each version and platform as precisely as required by your application. Literally every version of Windows is unique, and that includes the myriad releases of Windows 10 which all have a custom set of evolving features. InstallAware is the only installer to accurately inventory 100% of all features and server roles ever released in all Windows versions and platforms, including X86, AMD64, and now, **AARCH64**!



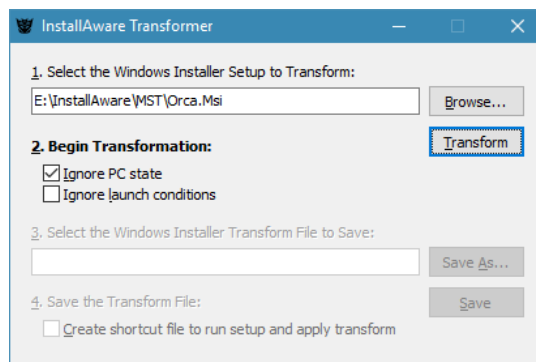
- **Dialog Designer** – Just as nothing has been hard-coded in InstallAware, setup dialogs too are fully editable and customizable. While InstallAware ships with 18 pre-built themes and even more custom-made dialogs in sample projects, nothing prevents you from creating highly customized, richly branded user interfaces that offer your customers a compelling installation experience.

As illustrated in the *Impact* theme, you may even combine controls from multiple dialogs inside a single dialog to provide a streamlined workflow for your customers, just like Visual Studio’s own setup routine! And remember, since InstallAware captures the native installation progress of setups you spawn from your master setup, the progress feedback of these third-party installations will be displayed interactively, inside the setup dialogs that you have designed!

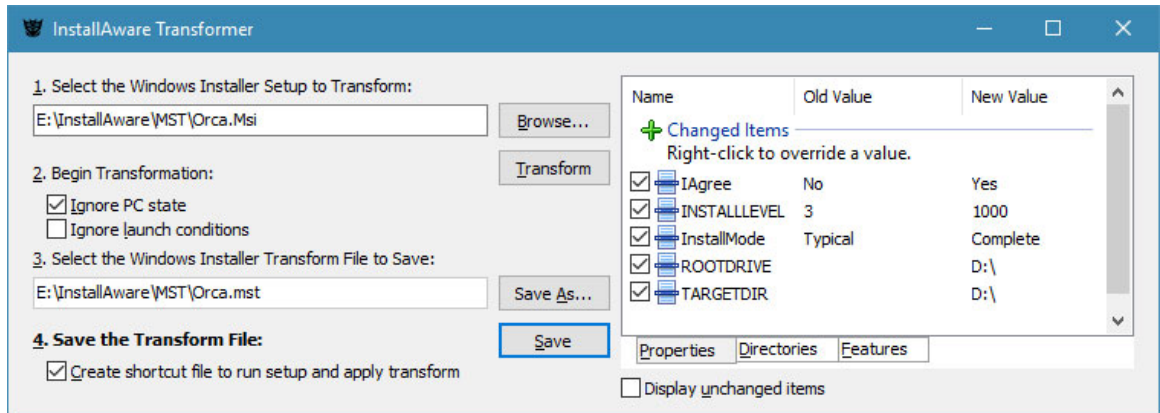
With InstallAware X12, you may also load dialog bitmaps programmatically at runtime to re-brand your installation on end-user systems without rebuilding anything! Custom dialogs are also a great place to collect and submit user data directly within your setup user interface, without ever launching external web browsers and risking the possibility of a click-away.



- **InstallAware Transformer** – InstallAware’s new installation tailoring tool walks you through an MSI’s installation user interface, as if you were normally installing the product.

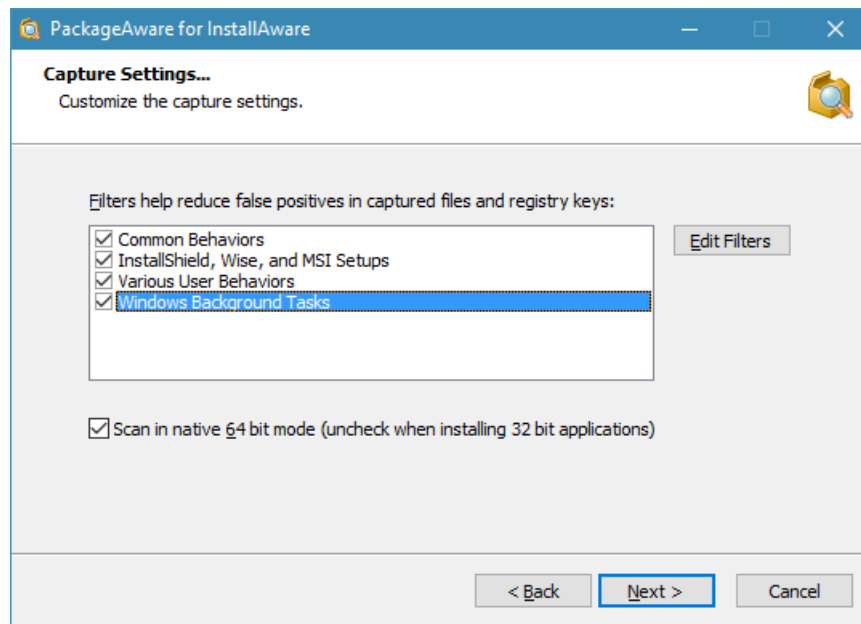


However, instead of installing the package, the MST creator collects **all changed properties, folders, and features** as you clicked your way through the setup wizard steps.



You may then review and edit all captured changes or opt out of including some changes in your final **MST transform file** at all. When you save, the InstallAware Transformer can also create a handy shortcut (.LNK file) for you, to install your MSI with your newly created MST pre-applied. Of course, the resulting MST file may be used together with the original MSI in Group Policy deployments across an Active Directory network, or with InstallAware’s Direct Deploy.

- **PackageAware** – Also included in InstallAware X12 is the industry’s fastest setup capture utility. Optimized with proprietary, driver-less direct capture technologies, PackageAware performs both immediate captures, and snapshot-based system scans. Save and compare any arbitrary two system snapshots to obtain the delta on any type of system change, in addition to software installations.

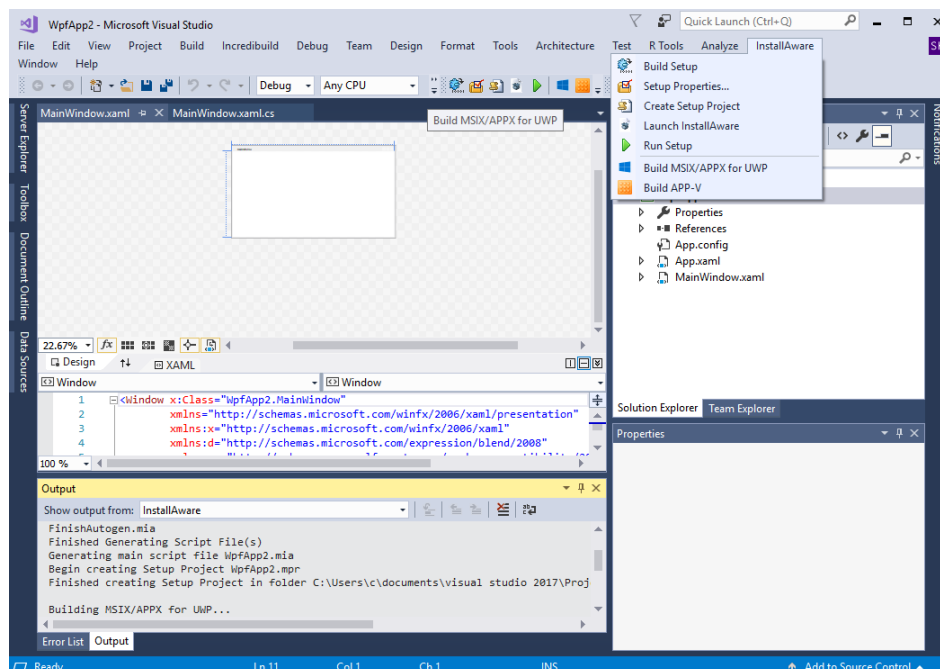


PackageAware is fully Unicode, offers the concept of stackable setup capture filters to be used during installation monitoring, and is based on a large memory model to ensure you never run out of memory, even when capturing on systems with millions and millions of files and registry keys.

Three new capture filters are available which eliminate various sources of capture noise on a system. You may combine multiple filters, create your own filters, and edit existing filters.

- **Visual Studio Integration** – InstallAware X12 integrates with Microsoft Visual Studio versions 2019, 2017, 2015, 2013, 2012, 2010, 2008, 2005, and 2003.

A single click on the InstallAware toolbar or menu within Visual Studio instantly builds your active project as a ready-to-install setup, together with its host of dependencies, such as Microsoft .NET Framework 4.8. An InstallAware setup project is also automatically created as part of this process, which you may further customize either inside InstallAware's Visual Studio Extension, or using the full-fledged InstallAware IDE, while keeping in sync with changes you continue to make in Visual Studio. You may even build your Visual Studio projects as MSIX Universal Windows Platform (**Desktop Bridge**) packages with a single click on the Visual Studio toolbar. A single-click build button is also available for Microsoft App-V Application Virtualization.



Basic Introduction

Keep in mind that this guide barely scratches the surface of what is possible with InstallAware. Please refer to the following additional resources that help you get the most out of the product:

MSIcode Scripting:

http://www.installaware.com/msicode_scripting_technology.pdf

Partial Web Deployment (Web Media Blocks):

http://www.installaware.com/webaware_installation_technology.pdf

Building Custom Application Runtimes:

http://www.installaware.com/installaware_extending_application_runtimes_view.pdf

Patching:

http://www.installaware.com/installaware_patching.pdf

Web Updates:

http://www.installaware.com/installaware_web_updates.pdf

Frequently Asked Questions:

http://www.installaware.com/installaware_faq.ppt

InstallAware Edition Comparison Matrix:

<http://www.installaware.com/right-edition.htm>

Community Forums:

<http://www.installaware.com/forum/>

Procurement:

<http://www.installaware.com/buydirect.asp>

InstallAware Blog:

<http://www.installaware.com/blog/>

About InstallAware Software

InstallAware Software, founded in 2003, is the leading Cloud Infrastructure Company with its laser sharp focus on bullet-proof enterprise software deployment. InstallAware has been recognized by multiple awards coming from Microsoft, SDTimes "**Leader of the Software Development Industry**", Visual Studio Magazine **Reader's Choice**, ComponentSource, WindowsITPro, among other recognition. InstallAware X12 is available in a free edition for all Visual Studio users and paid editions with prices starting at \$329. For a fully functional 30-day trial and more information, visit www.installaware.com.

Copyright© 1996-2020 InstallAware Software. All rights reserved. All InstallAware brand and product names are trademarks or registered trademarks of InstallAware Software in the United States and other countries. Microsoft, Windows, and other Microsoft product names are trademarks or registered trademarks of Microsoft Corporation in the U.S. and other countries. All other marks are the property of their respective owners.

www.installaware.com · sales@installaware.com