



Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

A timeline walkthrough | Told by the customer himself |
Spiced up with feelings and real-life happenings



Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

Case Study

Scott McClannahan
Independent Software
Consultant
Owner of Pure Technology Inc.

Website

www.pureittech.com/
www.linkedin.com/company/pureittech/about/

Industry

Technology and Communication Services, IT Managed Services

Employees Count

15-50

About Pure Technology

Pure Technology is a USA-based MSP specializing in smallbusiness engagement and developing customized, unique solutions for its clients.

Pure Technology is a leader in IT services for small businesses serving the Atlanta and Denver metro areas since 1996.

For over two decades, Pure Technology has helped companiesinnovate and transform by leveraging its unique approach to IT.

With their "peace of mind" support plans, you can rest easy knowing we've got you covered with 24/7 monitoring and technology support so you don't have to.

"We can work hand-in-hand to support your existing technology team or take full responsibility and free up valuable resources within your business. Enjoy fixed predictable pricingso you know what you're getting and what you're payingbefore there's a problem."



*"Skip the Headache of Building a Software Installer - Use a tool that JUST Works!
My story of how I moved from using Microsoft Setup Projects to Advanced Installer ... and ended up loving it!"*

Scott McClannahan Owner of Pure Technology Inc.

Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

Since my early days of writing software, I've noticed a common trend. When a new software product begins its life, lots of time and energy are spent in conceptualization, design, planning and in most companies, in QA and Testing.

Software comes in all sizes and flavors, but it is safe to say that most solutions take several months to go from concept to market. Not to mention the inordinate amount of financial resources that are poured into a project hoping it will solve a problem, build a better mousetrap, and -of course- make a profit!

After months worth of planning, engineering, researching, coding your software, developing a reliable testing and QA process, and hopefully creating documentation and marketing materials -- you're finally ready to release your software.

At this point, you've met with all the SMEs and PMs, obtained the needed approvals, and the marketing campaign is ready to go. Oops, there's only one thing missing -- you forgot to write an installer!

The Challenge: Building an installer

Building an installer: Why do many software developers forget this step?

Let me take you back to the start. I've been a software developer for over 25 years and through my experience, I've worked on numerous large projects where my software was only a small cog in a much larger machine. In addition, I've written basic utilities and services to solve smaller business challenges and I have a bit of experience with software installers.

Now, how could I forget the installer?

Let's go through my timeline while working on this project.

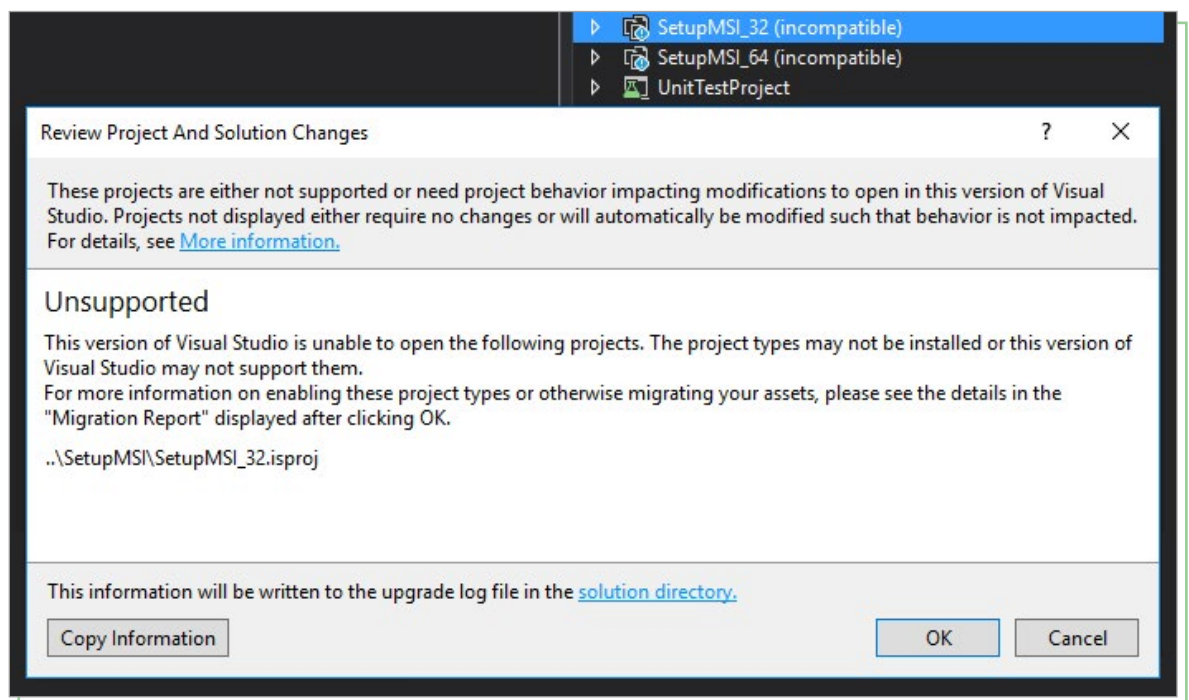
Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

The Process: The Timeline Walkthrough

11:00 pm **Bad Luck.**

Recently, I found myself once again at the 11th hour (actually the 23rd hour), working a late night in order to get a release of a new software project into the marketplace. Like many of you, my "installer of choice" was always the free one that came with Microsoft Visual Studio, so you can imagine my frustration when I installed the latest version of Visual Studio and opened my solution to find a big fat warning:

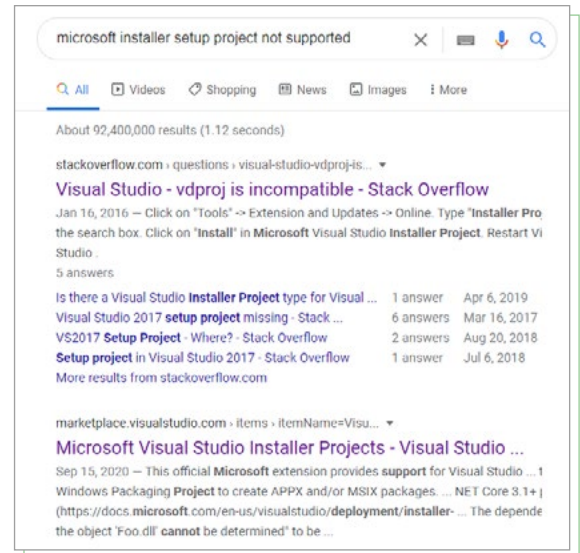
What? Surprise!. Microsoft, in it's infinite wisdom, has decided to no longer support one of the features which made my life as an independent software developer easy – the Microsoft Setup Project!



Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

11:25 pm First instinct: Denial.

My first instinct was denial: “this can’t be right – there must be a workaround”. So, I did what most of us do when faced with a problem we cannot easily solve ... I turned to Google. A quick search of “Microsoft Installer Setup Project not supported” rendered multiple results from some of my favorite developer communities:



So I started clicking and clicking -- and after about 15 minutes I learned that:

1. VDPROJ is no longer supported in Visual Studio,
2. Their “solution” is to just exclude the project (if I didn’t need the installer, I wouldn’t have built it so ...yeah), and
3. There is this new platform called “WIX” that most people have moved onto which includes an “unsupported” extension that might work but it’s laden with bugs and warnings about compatibility with newer projects.

Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

11:55 pm First hope.

I decided to take my chances. I downloaded the extension, installed it, and voilà! I reopened my solution, and the installer project opened just fine. It seemed everything was golden and I could get this release out the door to market... until I tried the installer. The first thing I noticed when building the installer was tons of build errors. After googling and solving those

problems (which required me to manually edit the .csproj files), I finally got the “duck taped” solution to build. Only to find out that the installer wouldn't run, or would only run on a very old version of Windows. Another 45 minutes lost...

12:40 am No, It didn't work!

Ok, that didn't work. Let's see what this WIX thing is all about. More googling. Turns out there is another extension (the WIX Toolset) that looks pretty easy to install. I do a little reading, their website says it “Integrates Seamlessly” (which sounds like a good sign), scrolling, reading, scrolling. After another 15-20 minutes of reading, I learned that the WIX framework

requires several lines of code around MSBUILD.exe, among others, and I signed my builds with a strong-key, plus there's a bunch of configurations you have to set manually in XML ... and argh, I was getting annoyed and realized it would take me all night just to get my first project done.

1:20 am Back to square one.

Back to square one. What else is out there? More Googling ... “Installers for Visual Studio .net projects”. The first result was “Advanced Installer”, it looks easy, but it's kind of expensive - I can find something else. I go back to Googling “Alternatives to Advanced Installer”, it yields some results: NSIS, WiX (there's that one again), INNO SETUP (it's free, that's a good sign, but it's from

1997, argh), IzPack, InstallForge (I've worked with that one, and know it's difficult and expensive ... next). Ok not a ton of options, but let's see what else is free.

Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

1:50 am Trying Inno Setup

I decided to try Inno Setup. I went to their website, downloaded the product, it's alright, but still a lot of command line stuff, and not many features -- not what I'm looking for.

2:00 am Trying NSIS

What's next on the list? NSIS? Sure why not, it seems to have good reviews and there's one of those little "thumbs up" icons -- that means it should be good, right? It's open-source, so the price is right. Downloading, oh no, more coding.

Why can't I just find a solution that is easy to use, works reliably, and doesn't require me to learn another language to get a simple installer? Aargh, here's where I get frustrated.

2:15 am I really need a break!

I made a trip to the cookie jar and took a short break to let my mind reset.

do
pre
the p
some o
shortcu

Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

The Solution

3:00 am **Advanced Installer again.**

When I first opened the application, I was presented with a dialog based on what kind of application I wanted to build/install. Ok, easy enough ... I chose .Net Application and clicked "Create Project".

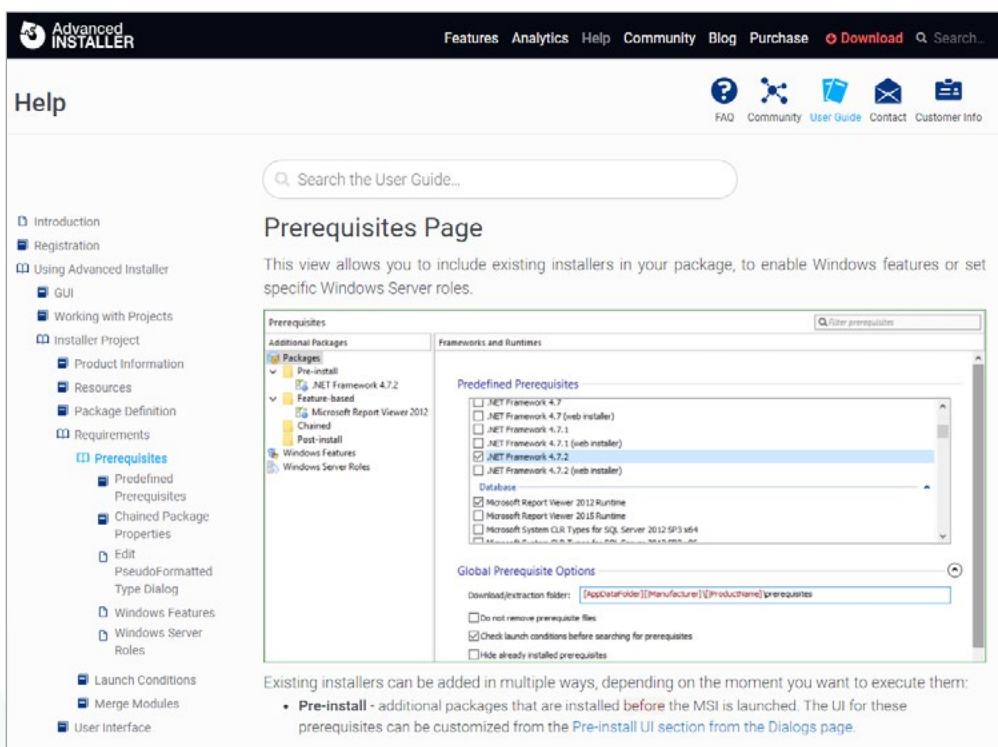
www.advancedinstaller.com > user guide > prerequisites ▾
Prerequisites Page - Advanced Installer
To make sure that the target machines can run your installation package, you can include Windows Installer as a prerequisite. Windows Installer predefined ...

www.advancedinstaller.com > tutorial-professional ▾
Creating a Professional package - Advanced Installer
Jump to **Add Prerequisites** – Select "New Package Prerequisite" from the context menu. I then select "My Application.exe" installation package from disk.

Prerequisites Page

Advanced Installer, there it is again. And right below it "tutorial-professional" - click.

"This view allows you to include existing installers ..." scrolling ... Pre-Install, Feature-Based ... scrolling ... wow this actually looks pretty easy. And look at the list on the left of other tutorials ... "Product Information", "Windows Features", "Windows Server Roles", Launch Conditions, Merge Modules. WOW! This product really seems to have a lot to offer – I'll give it a shot.



The screenshot shows the 'Prerequisites Page' in the Advanced Installer application. The page has a dark header with the 'Advanced INSTALLER' logo and navigation links: Features, Analytics, Help, Community, Blog, Purchase, Download, and a search bar. Below the header is a 'Help' section with a search bar and a list of topics on the left: Introduction, Registration, Using Advanced Installer (GUI, Working with Projects, Installer Project, Product Information, Resources, Package Definition, Requirements), Prerequisites (Predefined Prerequisites, Chained Package Properties, Edit, PseudoFormatted, Type Dialog, Windows Features, Windows Server Roles), Launch Conditions, Merge Modules, and User Interface. The main content area is titled 'Prerequisites Page' and contains a description: 'This view allows you to include existing installers in your package, to enable Windows features or set specific Windows Server roles.' Below this is a 'Prerequisites' dialog box with two tabs: 'Additional Packages' and 'Frameworks and Runtimes'. The 'Frameworks and Runtimes' tab is active, showing a list of predefined prerequisites with checkboxes. The list includes: .NET Framework 4.7, .NET Framework 4.7 (web installer), .NET Framework 4.7.1, .NET Framework 4.7.1 (web installer), .NET Framework 4.7.2, .NET Framework 4.7.2 (web installer), Database, Microsoft Report Viewer 2012 Runtime, Microsoft Report Viewer 2015 Runtime, Microsoft System CLR Types for SQL Server 2012 SP3 x64, and Microsoft System CLR Types for SQL Server 2012 SP3 x86. Below the list are 'Global Prerequisite Options' with a text box for the download/extract folder and three checkboxes: 'Do not remove prerequisite files', 'Check launch conditions before searching for prerequisites', and 'Hide already installed prerequisites'. At the bottom, there is a note: 'Existing installers can be added in multiple ways, depending on the moment you want to execute them: • Pre-install - additional packages that are installed before the MSI is launched. The UI for these prerequisites can be customized from the Pre-install UI section from the Dialogs page.'

Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

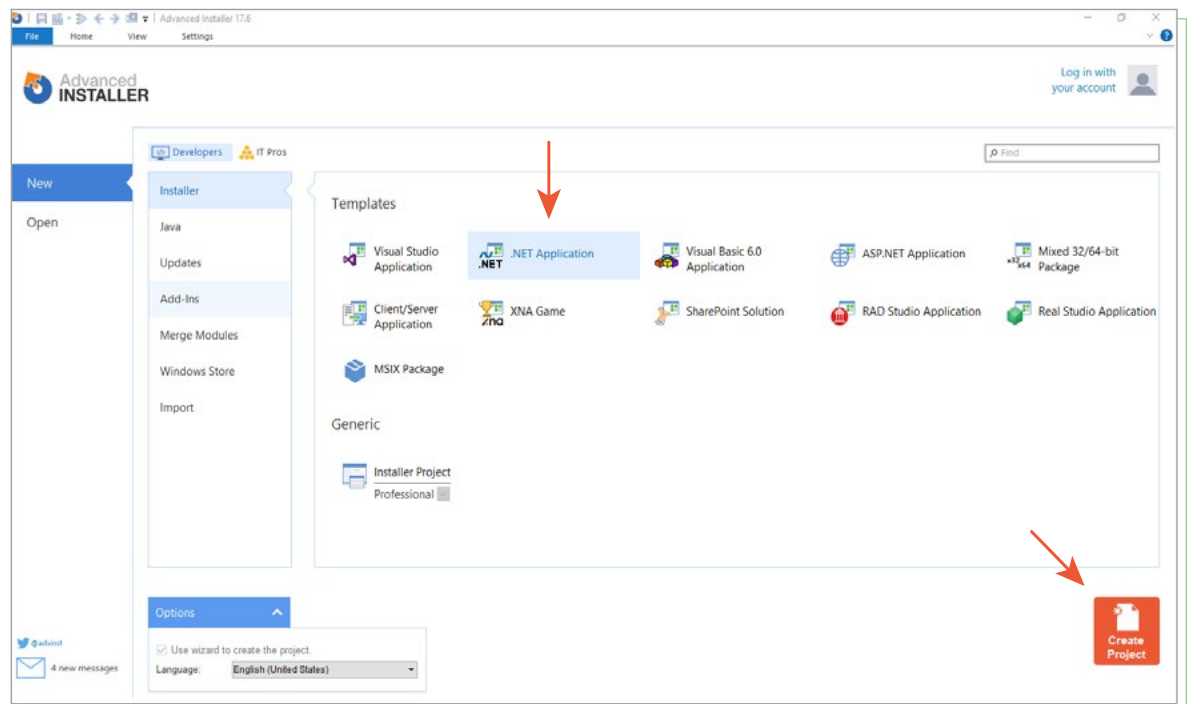
The Solution

3:30 am My First Advanced Installer Experience ... piece of cake!

I decide to download and try Advanced Installer.

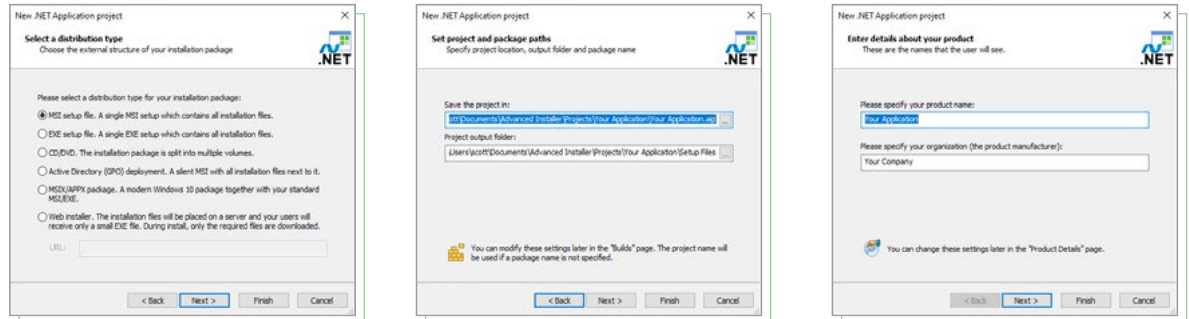
Downloading ... Installing ... (I usually look for a getting started or quick reference but this time I didn't) ... Installed.

When I first opened the application, I was presented with a dialog based on what kind of application I wanted to build/install. Ok, easy enough ... I chose .Net Application and clicked "Create Project".

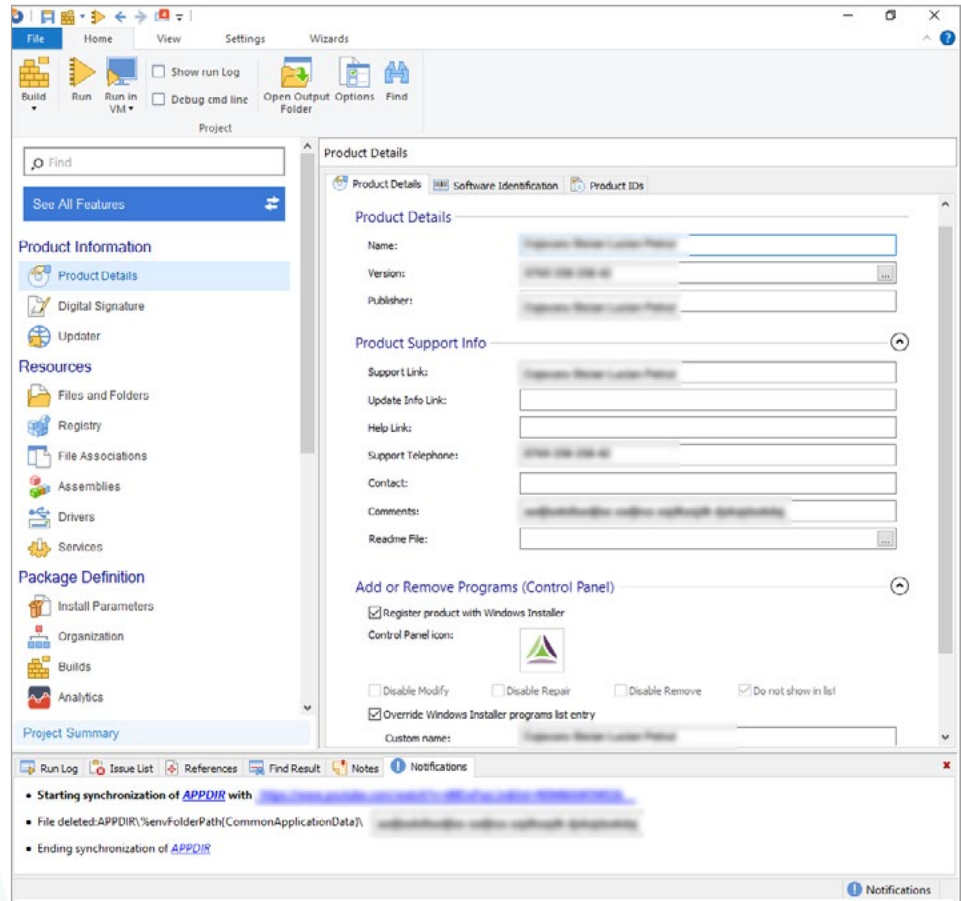


The next few pages were very basic steps asking about the product name, my company, what kind of distribution media I wanted (MSI, EXE, CD/DVD, Web Installer, and even support for a Windows 10 MSIX/APPX package). Wow, they've thought of everything! And finally, where do I want to save myproject file and setup files and where does my source project (bin folder) live.

Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.



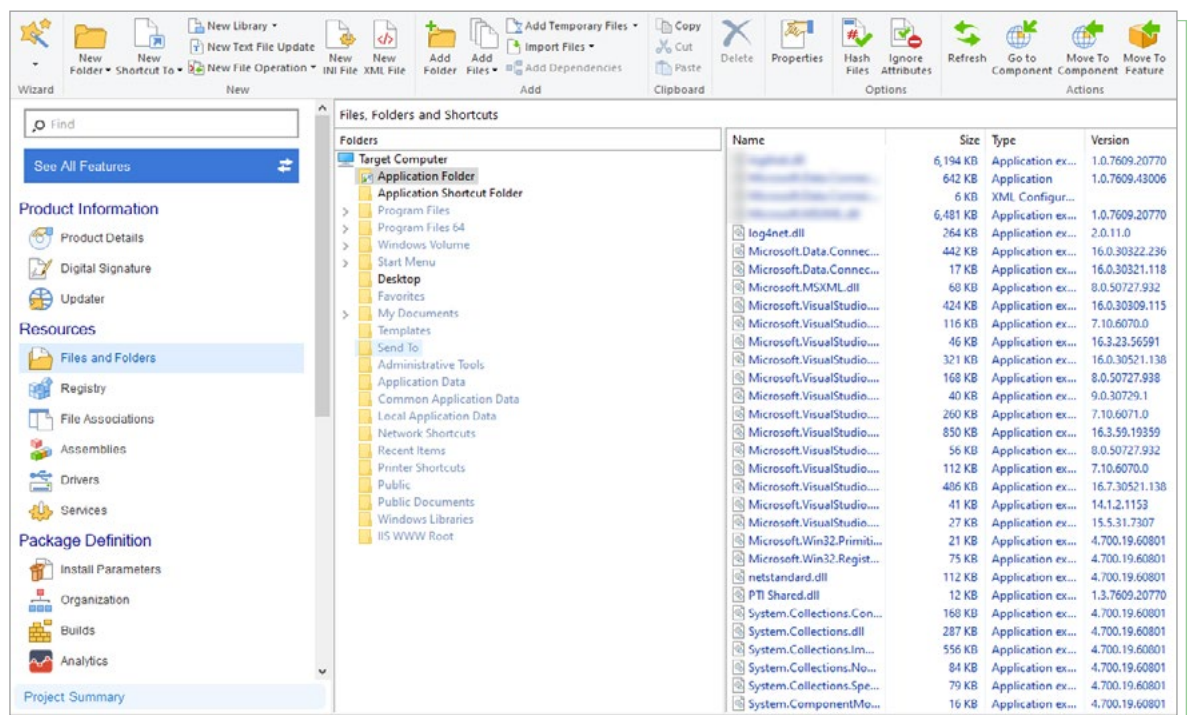
Easy ... Peasy. Once the “wizard” was completed, I was redirected to the main ‘Project’ on the ‘Product Details’ page where I could easily add additional details such as my software Version, Publisher, Support Link, Help Link, Telephone, Comment, icons, etc. You’ll also notice in the screenshot below that the system is smart enough to know where I want to put things like my APPDIR (where the application gets installed) and it figured out that I had a “log” file that would need to go in the [Common Application Data] folder. Again, pretty sweet!



Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

3:45 am 15 minutes later.

Everything looks good and it's taken me less than 15 minutes. I did a little clicking around. Under the "Files and Folders", I discovered that all the files my application would need, including the Microsoft DLLs and others I had linked to my project were already included. It was easy to then add and rearrange specific items such as documentation, links, settings files (which would go in the Common Application Data folder) and shortcuts for both Desktop and Start Menu.



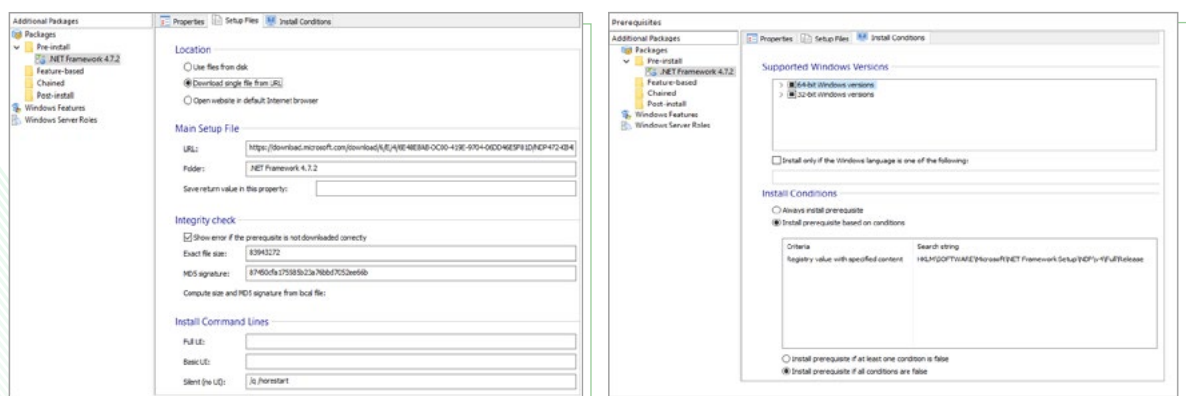
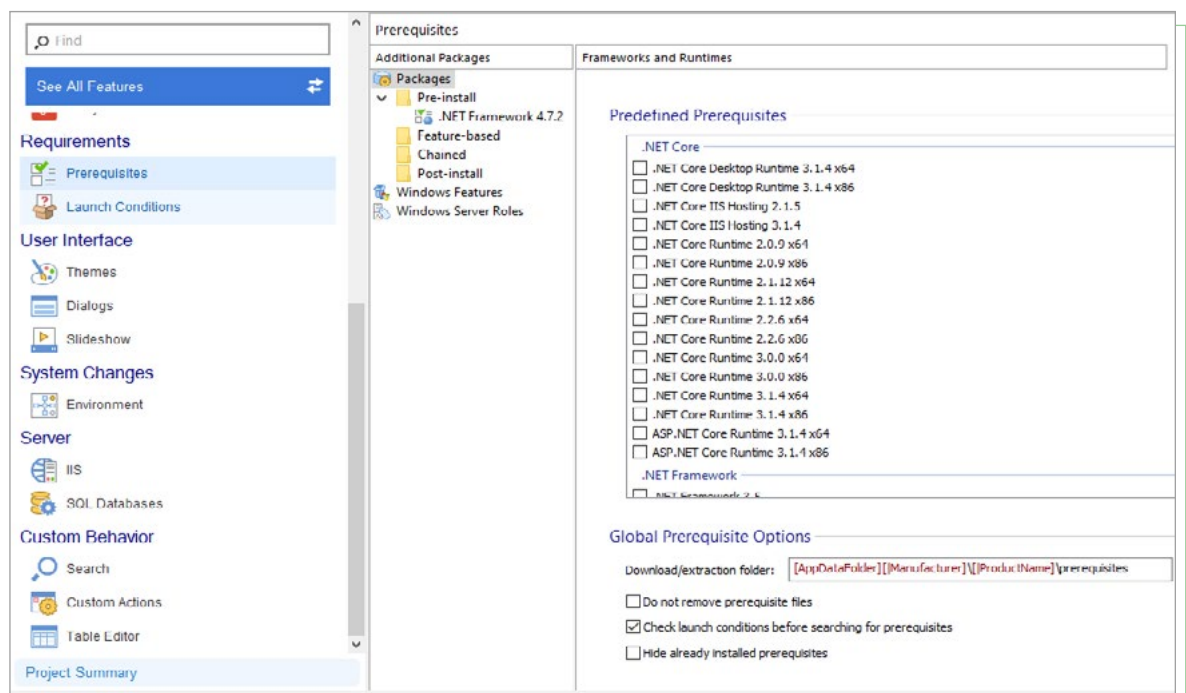
If I had needed them, I could also have easily added Registry Keys, File Associations, and Windows Services with a few clicks ... and I haven't even read the first "tutorial" or "documentation" on their website!

Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

3:50 am Now, what about those prerequisites I need?

Remember, I had found what looked like some really good documentation when I was still searching for an installer solution but it turns out I don't need it. If you scroll down just below "Package Definition" there is a section called "Requirements". If you click on that, you get a simple checklist of common packages which your solution might require.

You also have the option to do things like "Reboot after installing this prerequisite", "Continue even if the prerequisite is not installed", and it also has the option to either include the required files or have the installer download them from a public site (i.e.microsoft.com) at install time to keep your packages small.



Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

I didn't need them, but Advanced Installer offers the ability to require and/or turn-on embedded Windows features and server roles (such as IIS), should your software require these. And, the screenshots don't do it justice, because the full list of "Predefined Prerequisites" range from about 40 versions of .NET Framework and Runtimes, everything SQL related from Report Viewer to express and full versions SQL SERVER, and other third party products such as Adobe Flash, Acrobat, Java, Silverlight (I know, who uses that anymore?), Internet Explorer, DirectX, multiple versions of Office, and even advanced administration tools such as PowerShell and Remote Server Admin.

“ *Note: This ONE feature alone, saved me hours of troubleshooting, googling, and complex XML which I would likely have had to conquer with the alternative solutions mentioned above.* ”

Oh and another "cool" feature about Advanced Installer, under "Package Definition", you can easily configure Install Parameters and Features (under Organization). That way, you can allow your users to install only part of your project (i.e. if they only need the client utility vs. the full server package) and you can easily tell Advanced Installer which DLLs and other files are included in each package and where they should get installed.

3:55 am I have my installer in less than 30 minutes.

It's been less than 30 minutes and I now have my installer, complete with the needed prerequisites, all ready to go.

I click the "BUILD" button and bam -- installer done!

A couple of quick tests on our lab machines to make sure everything installs correctly and we're done.

4:05 am Uploading the installer for my customers.

I uploaded our new installer to the website where it will be distributed to our customer base tomorrow.

Time for bed and I can still get a little sleep before I start my day with meetings and phone calls at 8 am!

Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

Some extra notes about Advanced Installer

After I got those few hours of much-needed rest, I spent a little more time with Advanced Installer. In just a few hours, I was able to easily add features to our installer, including:

- *User Agreement and EULA*
- *User Registration and Licensing*
- *Automatic Updates downloaded from our server*
- *Windows File association for our project files*
- *Custom Dialogs with additional options (i.e. Run after install or Display Release Notes)*
- *Install Parameters such as logging, ability to reboot, and support for logging*
- *And my personal favorite: pre-built "skins" with our logos and color scheme to make the installation package match our branding (marketing loves this, by the way). You can also add a "slideshow" of images to display to the user during the installation process, but we haven't tackled that one just yet.*

Pure Technology's story of moving from Microsoft Setup Projects to Advanced Installer.

My Takeaways

Like many of us, my daily activities consist of multiple meetings, phone calls, Webex, etc. and I often find myself doing my real “work” late at night in the quiet wee hours. Too often, we are met with last-minute deadlines that don't allow time for adequate research and evaluation when we need a new tool, so we turn to Google. In an effort to save a few bucks (and not wanting to learn a new product), I wasted nearly 4 hours of my precious time (and valuable hours I could have been asleep) trying to solve my problem with open source and cobbled together freeware solutions.

Key Point

You get what you pay for every single time.

If you want a quality installer product that “just works” and is easy to use and packed with features, I strongly recommend Advanced Installer. In less than 15 minutes, I had an installation package that was ready to go. And in another 10 (only because I poked around a bit), I had a fully deployable package, complete with all my needed prerequisites ready for distribution.

Take it from me – don't waste valuable time trying to find another solution – use Advanced Installer and spend your hours doing something more productive with your life or enjoy that time with family, friends, or just sleeping.



Note from Advanced Installer

When we read Scott's story we were absolutely mesmerized by the level of details, the emotions and the level of depth for real-life stories he shared with us. So we decided to make minimal interventions - most of them in terms of content formatting and organization to make it easier to follow.

We hope you enjoyed the candor of this story as much as we did.