



Wi-Fi Installation Guide

Revision 1.1

February 2018

Intel Confidential



Table of Contents

1	About this Document.....	4
1.1	Introduction, purpose, and scope	4
1.2	Document revisions.....	4
1.3	Abbreviations and terminology	4
1.4	References.....	5
2	INF Types in Microsoft Windows 10*	6
2.1	Base INF	6
2.2	Extension INF	6
2.3	Component INF	8
2.4	PROSet – tightly coupled INF installation.....	8
3	The UWD layout for Wi-Fi	14
3.1	Tools	14
3.2	Utilities	18
3.3	Wi-Fi drivers.....	27
4	UWD-compliant Installation Methods	32
4.1	Right-click installation.....	32
4.2	PnPUtil-based installation	32
4.3	Pnputil – Install the driver from the driver store	33
5	Appendix.....	34
5.1	Wi-Fi customization table for drivers 04 and 06.....	34
5.2	Setup.xml configuration.....	34
5.3	What a correct PROSet MSI installation via INF looks like	37
5.4	Uninstalling UWD PROSet	41
5.5	Driver Update Acceptable (DUA) process	46
5.6	How to generate a GUID	48



Table of Tables

Table 1 – Document revisions	4
Table 2 – Abbreviations and terminology	4
Table 3 – Reference documents	5
Table 4 – Wi-Fi customization table for drivers 04 and 06	34
Table 5 – XML parameters and configurations	36

Table of Figures

Figure 1- Extension INF and DUA (Driver Update Acceptable) Process	8
Figure 2 – PROSet – tightly coupled INF chart	9
Figure 3 – Installation of Wi-Fi Base INF	10
Figure 4 – Installation of PROSet extension INF	10
Figure 5 – Wi-Fi + Customization INF, installation order	11
Figure 6 – Intel® PROSet/Wireless Wi-Fi Software installation order	12
Figure 7 – New installation model	13
Figure 8 – Programs and Features entry for Intel® PROSet/Wireless WiFi Software	38
Figure 9 – Intel® PROSet/Wireless icon	39
Figure 10 – Intel® PROSet/Wireless Wi-Fi Software services running	40
Figure 11 – Intel® PROSet/Wireless Wi-Fi Software folders installed	41



1 About this Document

1.1 Introduction, purpose, and scope

This document covers the installation of the following Wi-Fi ingredients in the **Microsoft Windows 10* RS3 and later** OS, which requires Universal Windows Drivers (UWD):

- Wi-Fi driver
- Intel® PROSet/Wireless Wi-Fi software
- Intel® Power Sharing Manager (PSM)

1.2 Document revisions

Table 1 lists the revisions of this document.

Table 1 – Document revisions

Rev.	Date	Owner	Details
1.0	Nov. 2017	Maor Tal	Initial release
1.1	Feb 2018	Maor Tal	<ul style="list-style-type: none"> • PROSet using InfAppRunner • PROSet tightly coupled Extension and Component INFs • OS Minimum version directive in INF [TargetOSVersion] • More info about ExtensionId in Extension INFs • More info on DUA process

1.3 Abbreviations and terminology

Table 2 lists the abbreviations and terminology used in this document.

Table 2 – Abbreviations and terminology

Term	Abbrev.	Description
Intel® Power Sharing Manager	PSM	A PSM is an ACPI BIOS object, used to discover and configure the Power Wireless components
Event Tracing for Windows	ETW	Efficient kernel-level tracing facility that lets log kernel to a log file
Advanced Configuration and Power Interface	ACPI	
Intel® Dynamic Platform and Thermal Framework	DPTF	A solution to help enable thin, quiet, and cool platform designs
Universal Windows Drivers	UWD	A driver package that contains an INF file and binaries that will install and run on UWP-based editions of Windows 10*
Universal Windows Platform	UWP	
Windows Installer XML Toolset	WiX	A free software toolset/framework that builds Windows Installer packages from XML code



1.4 References

Table 3 lists other documents referenced in this document.

Table 3 – Reference documents

Ref #	Document Name	Document Location
Ref 1	Intro to the Universal Windows Platform	https://docs.microsoft.com/en-us/windows/uwp/get-started/universal-application-platform-guide
Ref 2	Getting Started with Universal Windows Drivers	https://docs.microsoft.com/en-us/windows-hardware/drivers/develop/getting-started-with-universal-drivers



2 INF Types in Microsoft Windows 10*

INF types:

- Base INF – Associated to a real Plug-and-Play (PnP) device. The Primary INF to install the Wi-Fi driver associated with the Intel Wi-Fi device.
- Extension INF – Extends a driver package INF file's functionality.
- Component INF - A software component package that can install one or more software modules.

2.1 Base INF

Every device must have one base INF, and can optionally have one or more Extension INFs associated with it.

2.2 Extension INF

An Extension INF used for base INF customization purposes are applied after the base INF.

Each Extension INF has a unique ID that is defined by an ExtensionId (a GUID under the **[VERSION]** section in the INF).

Intel provides an Intel-specific ExtensionId in the Extension INF. Each OEM should override the Intel-specific ExtensionId (GUID) and replace it with the OEM's unique GUID. See the section [“How to Generate a GUID”](#) for instructions.

The OEM doesn't need to create a new GUID for each subsequent version of the same Extension INF. The OEM should generate a GUID once per Extension INF and then reuse that same GUID for newer versions of that same Extension INF.

Intel will update the other fields (like version number and date) in the Extension INF; the OEM should not touch these fields.

For example, for the PROSet Extension INF, the GUID that needs to be replaced by the OEM is highlighted below.

```
[version]
Signature = "$Windows NT$"
Class = Extension
ClassGUID = {e2f84ce7-8efa-411c-aa69-97454ca4cb57}
Provider = %PROVIDER_NAME%
ExtensionId = {569bd0e5-e8cc-4f8c-a53d-2fd78123be93} ; Ext. GUID For PROSet
DriverVer = 08/08/2017,1.0.0.5 ;DATE HAS TO BE IN FOLLOWING FORMAT MM/DD/YYYY
CatalogFile = ProSetExtension.cat
```

For more information about ExtensionId, See [Creating an Extension INF](#).

Or, [Hardware Dev Center now automatically registers Extension IDs](#).



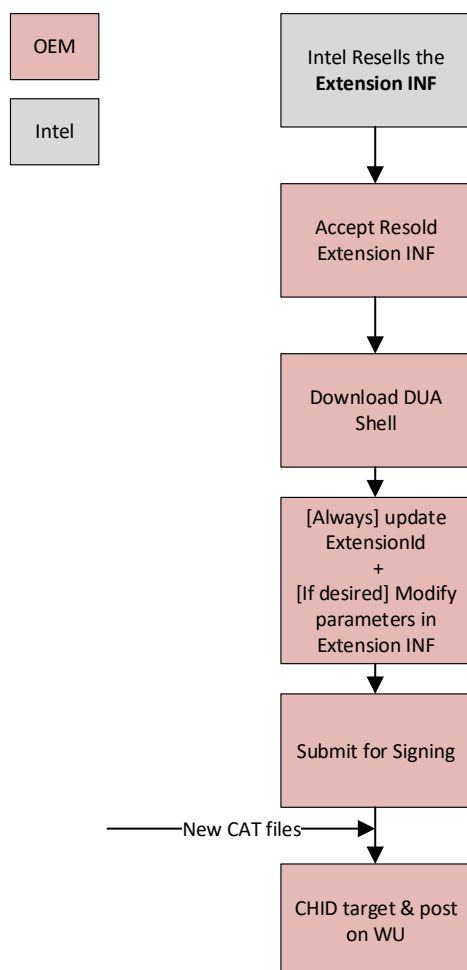
The wireless Extension INFs will limit installation on Windows 10 RS3 or newer by using the *TargetOSVersion* decoration in the INF.

```
[Manufacturer]
%COMPANY_NAME% = DeviceExtensions, NTamd64.10.0...16299
.
.
.
[DeviceExtensions.NTamd64.10.0...16299]
```

For more details, see [INF Manufacturer Section TargetOSVersion](#).

The OEM should submit the Extension INF to WU after updating the ExtensionId and (if desired) modifying the allowed parameters in the Extension INF via the DUA Process that is described below.

Figure 1- Extension INF and DUA (Driver Update Acceptable) Process



2.3 Component INF

Installation of a Component INF is applied after the Extension INF that created the child device node in the Device Manager.

For more info about “Universal and INFs types,” see [Using an Extension INF File](#).

2.4 PROSet – tightly coupled INF installation

The PROSet Extension INF and Component INF are now tightly coupled.

The user will only have to install one INF instead of two. Only the Extension INF (***prosetextension.INF***) will have to be installed, and that INF will trigger the installation of the Component INF and the PROSet package automatically as described in the “PROSet - Tightly Coupled INF Chart,” below.

For more information about the “Tightly coupled INF” mechanism, see [tightly-coupling-multiple-inf-files](#).

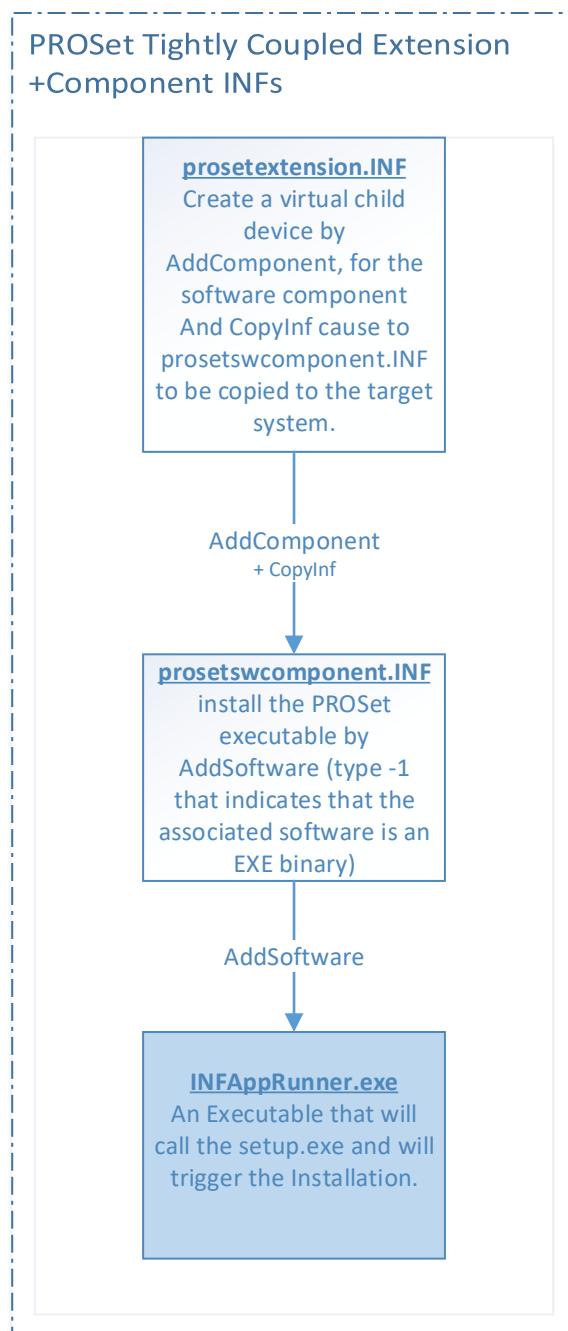
Figure 2 – PROSet – tightly coupled INF chart

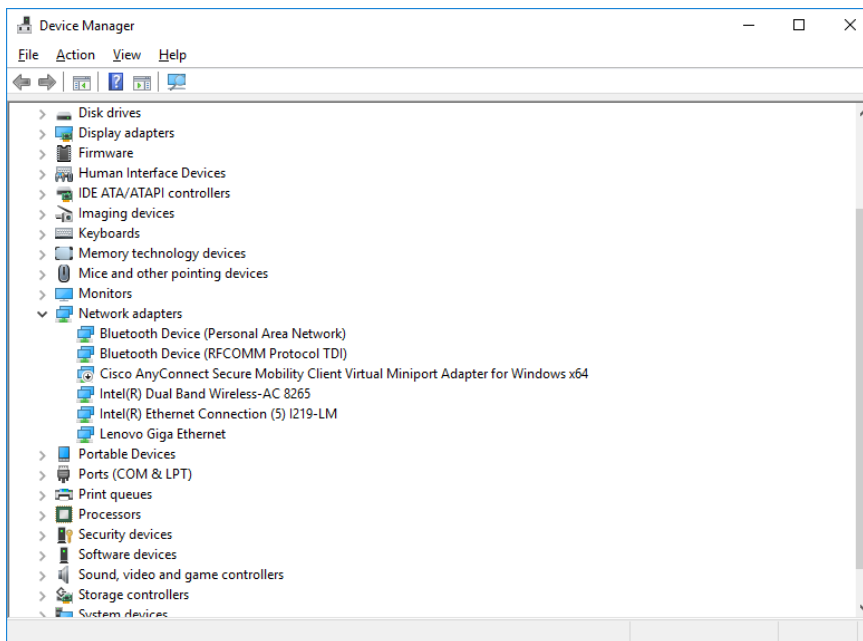
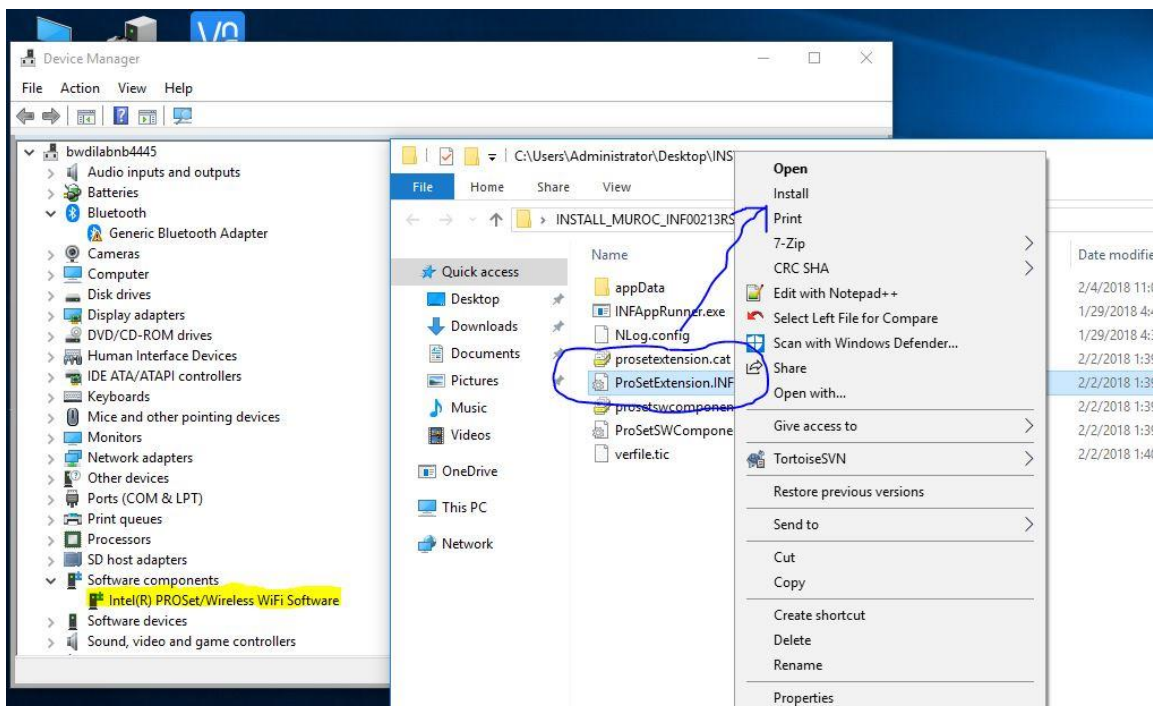
Figure 3 – Installation of Wi-Fi Base INF**Figure 4 – Installation of PROSet extension INF**

Figure 5 – Wi-Fi + Customization INF, installation order

Install Wi-Fi With Customizations

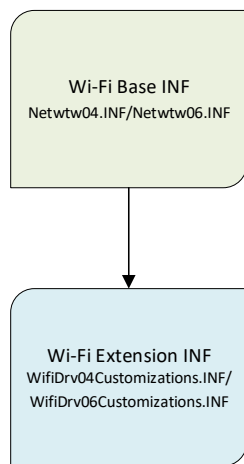


Figure 6 – Intel® PROSet/Wireless Wi-Fi Software installation order

Install Intel® PROSet/Wireless Wi-Fi Software

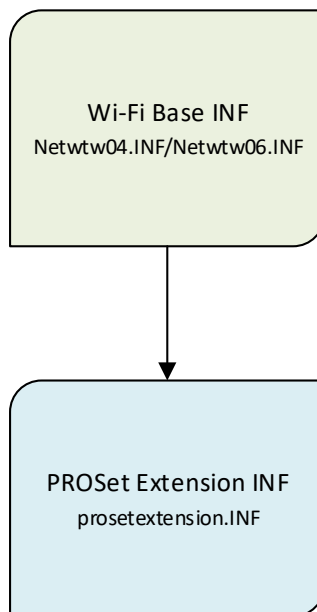
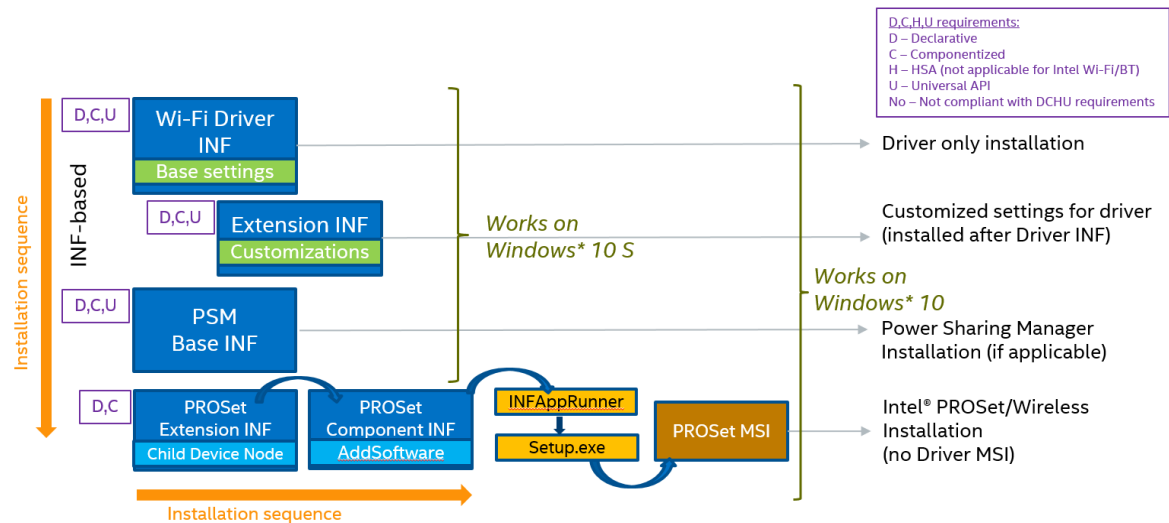


Figure 7 – New installation model

New installation model – Wi-Fi

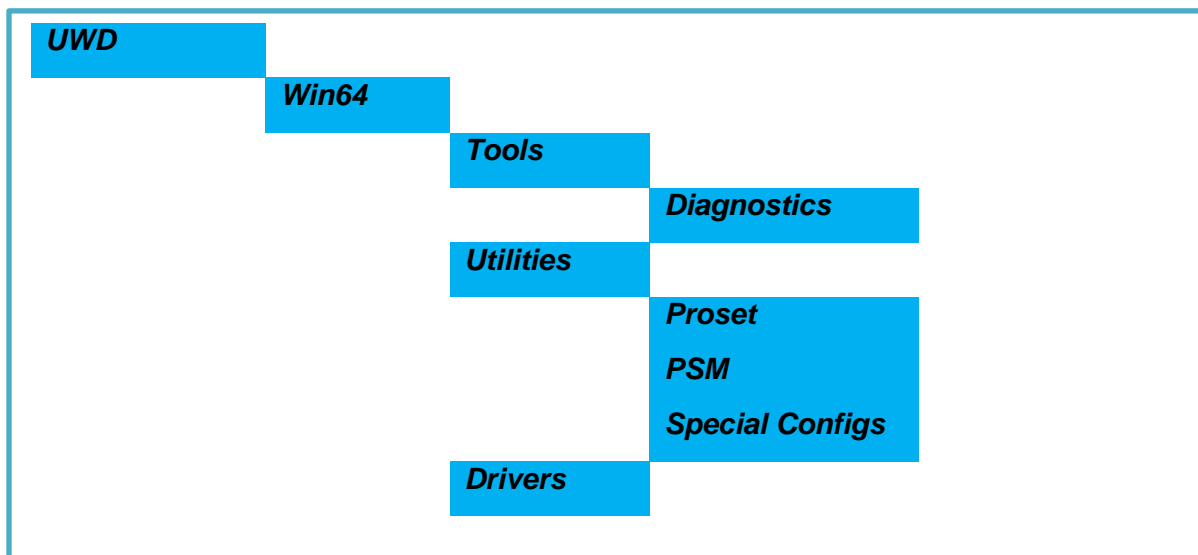


3 The UWD layout for Wi-Fi

The UWD-compliant software supports 64-bit versions of Windows 10* only. The layout contains

1. The Wi-Fi drivers
2. Extension software, applications or drivers (e.g., Intel® PROSet/Wireless, Intel® Power Sharing Manager)
3. Tools (diagnostic tools)

Layout: The folders are colored in blue.



3.1 Tools



Contains the folder:

- *Diagnostics* – Contains Wi-Fi Driver Event Tracing for Windows (ETW), which lets you log Wi-Fi-driver-defined events to a log file. It includes scripts that enable/disable the Wi-Fi driver ETW logging.

3.1.1 Diagnostics

All diagnostics data output from the Wi-Fi core driver is done via the Windows ETW/PCW native-OS mechanism.

Diagnostic data shall take the form of:

- ETW Events – For diagnostics events data
- PCW Performance Counters – for statistics/metrics snapshot diagnostic data

For more details how to use ETW PCW, see the document called 'Intel Wi-Fi Core Windows Diagnostics For Customers' (CDI document #575259)

Diagnostics	
RegisterCustomerEtwPcwFiles.bat	
UnRegisterCustomerEtwPcwFiles.bat	
ETW	Driver
RESOURCES	

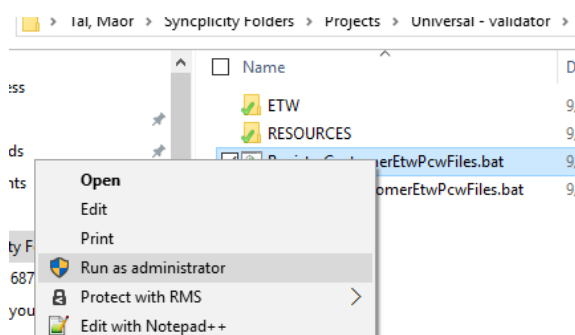
Contains the folders:

- *ETW* – Contains the scripts that enable/disable the Wi-Fi Driver ETW manifests.
- *Resources* – Contains DLLs that enable and disable the individual providers and thus the events that they generate. Another DLL is used to interpret the events.

For more info about ETW, see [About Event Tracing](#).

3.1.1.1 RegisterCustomerEtwPcwFiles.bat

Type:	Batch file
Dependencies:	Diagnostics subfolders
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise etc.)
Description:	Installs the event publishers and logs from a manifest. Call to RegisterAllManifests.bat with 'customer' parameter.
How to:	Right-click, run as Administrator (or via Command Line/CLI – as Administrator)





3.1.1.2 UnRegisterCustomerEtwPcwFiles.bat

Type:	Batch file
Dependencies:	Diagnostics subfolders
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise etc.)
Description:	Uninstalls all publishers and logs from a manifest. Call to UnRegisterAllManifests.bat with 'customer' parameter.
How to:	Right-click, run as Administrator

3.1.1.3 ETW

ETW	RegisterAllManifests.bat
	UnRegisterAllManifests.bat
	Driver

Contain the folders:

- *Driver* – Contains the scripts that enable/disable the Wi-Fi Driver ETW manifests, plus the Manifest.

3.1.1.3.1 RegisterAllManifests.bat

Type:	Batch file
Dependencies:	Diagnostics sub folders
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)
Description:	Install all publishers and logs from a manifest. Call to RegisterCustomerManifest.bat.
How to:	Internal batch file; user should not execute it directly.

3.1.1.3.2 UnRegisterAllManifests.bat

Type:	Batch file
Dependencies:	Diagnostics sub folders
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)



Description: Uninstalls all publishers and logs from a manifest. Call to UnRegisterExternalManifest.bat.

How to: Internal batch file; user should not execute it directly.

3.1.1.3.3 Driver

Driver

RegisterCustomerManifest.bat

UnRegisterCustomerManifest.bat

WlanDriverEventsCustomer.man

3.1.1.3.3.1 RegisterCustomerManifest.bat

Type: Batch file

Dependencies: Other Diagnostics sub folders

Supported OS: Windows 10* Desktop only (Home, Pro, Enterprise, etc.)

Description: Installs all publishers and logs from a manifest.

- Unregisters in case user didn't unregister.
- Copies IntelWi-FiDriverEtw_driver_customer.dll to System32
- Installs all publishers and logs from a manifest.

How to: Internal batch; user should not execute it directly.

3.1.1.3.3.2 UnRegisterCustomerManifest.bat

Type: Batch file

Dependencies: Diagnostics sub folders

Supported OS: Windows 10* Desktop only (Home, Pro, Enterprise, etc.)

Description: Does the following:

- Uninstalls all publishers and logs from a manifest.
- Deletes the IntelWi-FiDriverEtw_driver_customer.dll from System32

How to: Internal batch; user should not execute it directly.



3.1.1.3.3 WlanDriverEventsCustomer.man

Type:	Manifest file
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise etc.)
Description:	<p>Wi-Fi drivers based on the managed object format (MOF) use a schema to define the events that an ETW Consumer knows how to process.</p> <p>Customers can use it as a reference to the event structures.</p>
How to:	Internal use only.

3.1.1.4 Resources

RESOURCES

IntelWiFiDriverEtw_driver_customer.dll

3.1.1.4.1 IntelWiFiDriverEtw_driver_customer.dll

Type:	Manifest dynamic library
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)
Description:	<p>Contains macros, string text corresponding to ETW events, etc.</p> <p>Produced as part of a driver's build.</p>
How to:	Will be used later for parsing the ETL file using WPA (a Microsoft tool).

3.2 Utilities

Wi-Fi extension software and configurations.

Utilities

Proset

PSM

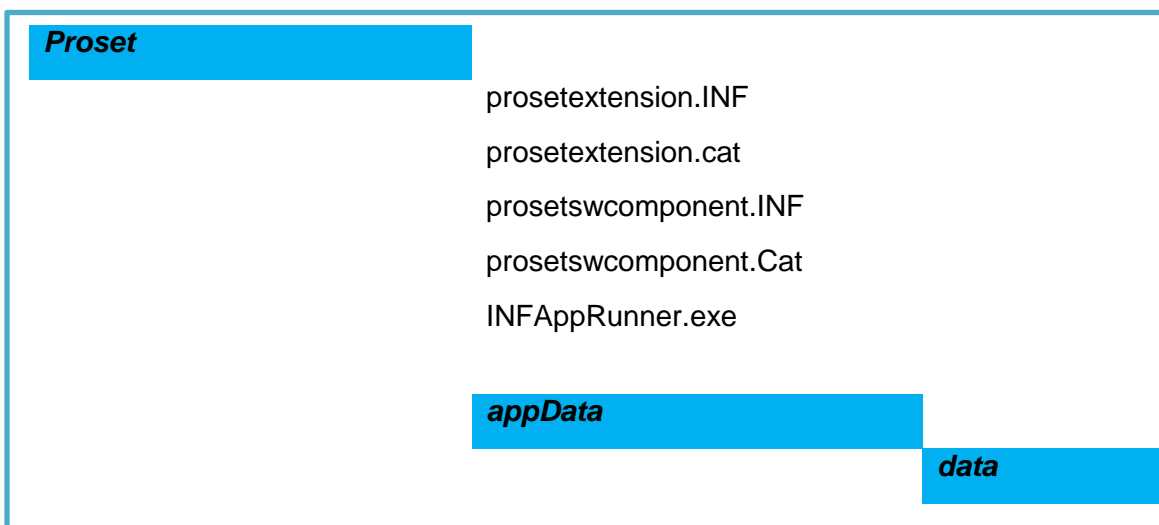
Special Configs



Sub folders:

- *Proset*
 - SDK – A set of APIs that interface with the Wi-Fi driver
 - Wi-Fi Admin tool – A tool to create custom network profile packages.
- *PSM* – Power Sharing Manager; for WiGig or DPTF platforms.
- *Special Configs* – To modify settings provided in a Wi-Fi base driver INF, modifying a hardware configuration setting.

3.2.1 Intel® PROSet/Wireless Wi-Fi software



The PROSet INF based installer.

Sub folders:

- *appData* – the Old Installation mechanism based on the setup.exe
 - *Data* – the Old Installation hierarchy

3.2.1.1 Prosetextension.inf

Type:	Extension INF
Dependencies:	Wi-Fi Base INF, and Catalog file (<i>prosetextension.cat</i>)
Supported OS:	Windows 10* RS3 Desktop only (Home, Pro, Enterprise etc.)
Description:	Creating a PROSet software component, and copy the Component INF into the driver store and trigger the installation of the Component INF.
How to:	See Section 4, “UWD-compliant Installation Methods”.



3.2.1.2 Prosetextension.Cat

Type:	A digitally-signed catalog file for the <i>prosetextension.INF</i> .
Dependencies:	
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)
Description:	Contains a collection of cryptographic hashes, or <i>thumbprints</i> . Each thumbprint corresponds to a file that is included in the collection.
How to:	Not relevant; part of the INF installation.

3.2.1.3 Prosetswcomponent.inf

Type:	Component INF
Dependencies:	<ul style="list-style-type: none"> • prosetextension.INF • Catalog file (<i>Prosetswcomponent.cat</i>) • INFAppRunner.exe • NLog.config • All the PROSet Subfolders and the content in the subfolders.
Supported OS:	Windows 10* RS3 Desktop only (Home, Pro, Enterprise, etc.)
Description:	Install one the PROSet software modules, by copying all the content into the driver store and calling ' <i>INFAppRunner.exe</i> '
How to:	Users should NOT run this file; it is part of the PROSet Extension INF ' <i>Prosetextension.inf</i> ' installation.

3.2.1.4 Prosetswcomponent.cat

Type:	A digitally-signed catalog file to the <i>Prosetswcomponent.INF</i>
Dependencies:	No dependencies
Supported OS:	Desktop
Description:	<p>Contains a collection of cryptographic hashes, or thumbprints. Each thumbprint corresponds to a file that is included in the collection:</p> <ul style="list-style-type: none"> • INFAppRunner.exe • NLog.config • All the PROSet Subfolders content. • prosetextension.inf
How to:	Not relevant; part of the INF installation.



3.2.1.5 INFAppRunner.exe

Type:	Executable
Dependencies:	NLog.config and The PROSet subfolders content.
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)
Description:	An application designed to replace the Autorun.exe to install the PROSet via a component INF. Copies the Setup.exe and the relevant files into users\appData (%temp%) folder and executes it from there.
How to:	Users should NOT run this file; it is part of the INF installation.

3.2.1.6 appData

appData	
	bsutil.exe
	bsutil.exe.config
	Microsoft.Deployment.WindowsInstaller.dll
	Setup.exe
	Setup.xml
	verfile.tic
	data

Sub folders:

- Data – Part of the WiX bundle sub folders hierarchy.

3.2.1.6.1 bsutil.exe

Type:	executable
Dependencies:	<ul style="list-style-type: none"> • bsutil.exe.config • verfile.tic
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)
Description:	An executable that update the TIC number in the "Add or Remove Programs" entry, used by the setup.exe.
How to:	Read only. Users should NOT run this file.



3.2.1.6.2 bsutil.exe.config

Type:	Configuration file schema for the .NET Framework
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)
Description:	A Startup Settings Schema describes the elements that specify which version of the common language runtime to use.
How to:	Read only. Users should NOT config this file.

3.2.1.6.3 Microsoft.Deployment.WindowsInstaller.dll

Type:	Dynamic Link Library
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)
Description:	Microsoft Windows Installer Library. Part of the WiX bundle.
How to:	Read only.

3.2.1.6.4 Setup.exe

Type:	WiX bundle executable
Dependencies:	<ul style="list-style-type: none"> • setup.xml • Microsoft.Deployment.WindowsInstaller.dll • bsutil.exe
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)
Description:	Main setup/installer executable, parse the Setup.xml and call the WiFi.Msi (PROSet). Part of the WiX bundle.
How to:	Read only. Users should NOT run this file.

3.2.1.6.5 Setup.xml

Type:	XML WiX bundle configuration File
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)



Description:	WiX bundle configuration file
How to:	A custom configuration file with default installation values. The user can modify this file as needed. For more details, see the setup.xml configuration in the Appendix section.

3.2.1.7 verfile.tic

Type:	Tracking Index for an Installer Container
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)
Description:	Text file with the TIC number of the build. Used by the bsutil.exe to update the version in the “Add or Remove Programs”
How to:	Read only. Users should NOT modify this file.

3.2.1.7.1 Data

data	WiFi.msi
-------------	----------

3.2.1.7.1.1 WiFi.msi

Type:	Microsoft Installer package
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop only (Home, Pro, Enterprise, etc.)
Description:	The Wifi.MSI Installer contains information about the PROSet which is divided into features and components, and every component may hold files and registry data.
How to:	Read only. Users should NOT use this file.

3.2.2 PSM

The Intel® Power Sharing Manager, a PnP ACPI driver. The OEM should enable the PSM via the BIOS. The HW-ID is *ACPI\INT3420*.

PSM main functionalities:

- WiGig platform – Distributes the power between the WiGig and Wi-Fi adapters

- DPTF platform – Interface between integrated Connectivity (CNVi) and DPTF.

PSM

PSM.inf
psm.sys
psm.cat

3.2.2.1 PSM.inf

Type: Base INF

Dependencies: Minimum Wi-Fi driver version 20.20

Supported OS: Windows 10 Desktop (Home, Pro, Enterprise etc.) and Windows 10 S

Description: Installation of the PSM driver.

How to: See Section 4, “UWD-compliant Installation Methods”.

3.2.2.2 psm.cat

Type: A digitally-signed catalog file for the PSM.INF

Dependencies: Minimum Wi-Fi driver version 20.20

Supported OS: Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S

Description: Contains a collection of cryptographic hashes, or thumbprints. Each thumbprint corresponds to a file that is included in the collection: PSM.INF and PSM.SYS

How to: Not relevant; part of the INF installation. See Section 4, “UWD-compliant Installation Methods”.

3.2.2.3 psm.sys

Type: Driver

Dependencies: No dependencies

Supported OS: Windows 10* Desktop (Home, Pro, Enterprise etc.) and Windows 10 S

Description: PSM binary driver file



How to: See Section 4, “UWD-compliant Installation Methods”.

3.2.3 Wi-Fi Customization

Modifying settings provided in a base Wi-Fi INF, modifying a hardware or driver configuration setting.

Special Configs

WifiDrv04Customizations.INF
 WifiDrv04Customizations.cat
 WifiDrv06Customizations.INF
 WifiDrv06Customizations.cat

3.2.3.1 WifiDrv04Customizations.INF

Type: Extension INF

Dependencies: Base Wi-Fi INF, Netfw04.INF, supported by Sandy Peak and Stone Peak (D0/D1 SKUs)

Supported OS: Windows 10* Desktop (Home, Pro, Enterprise etc.) and Windows 10 S

Description: Installation of the Wi-Fi driver customizations, for details see [Wi-Fi Customization Table for Drivers 04 and 06](#).

How to:

- Installation method same as base INF
- OEM can change the basic configuration that was set in the base INF using this file

3.2.3.2 WifiDrv04Customizations.cat

Type: A digitally-signed catalog file *to WifiDrvCustomizations.INF*

Supported OS: Windows 10 Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S

Description: Contains a collection of cryptographic hashes, or *thumbprints*. Each thumbprint corresponds to a file that is included in the collection: *WifiDrv04Customizations.INF*

How to: Not relevant; part of the INF installation. See Section 4, “UWD-compliant Installation Methods”.



3.2.3.3 WifiDrv06Customizations.INF

Type:	Extension INF
Dependencies:	Base Wi-Fi INF Netfw04.INF, supported hardware: <ul style="list-style-type: none">• Snowfield Peak• Windstorm Peak• Thunder Peak 2• Jefferson Peak 1• Jefferson Peak 2
Supported OS:	Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S
Description:	Installation of the Wi-Fi driver customizations; for details see Wi-Fi Customization Table for Drivers 04 and 06 .
How to:	<ul style="list-style-type: none">• Installation method same as base INF• OEM can change the basic configuration (set in the Base INF) using this file

3.2.3.4 WifiDrv06Customizations.cat

Type:	A digitally-signed catalog file to <i>WifiDrvCustomizations.INF</i> .
Supported OS:	Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S
Description:	Contains a collection of cryptographic hashes, or thumbprints. Each thumbprint corresponds to a file that is included in the collection <i>WifiDrv06Customizations.INF</i>
How to:	Not relevant; part of the INF installation. See Section 4, “UWD-compliant Installation Methods”.



3.3 Wi-Fi drivers

The base Wi-Fi drivers:

Drivers

Netfw04.dat
 netwtw04.cat
 Netwtw04.INF
 netwtw04.sys
 Netfw06.dat
 netwtw06.cat
 Netwtw06.INF
 netwtw06.sys
 IntelWifilhv06.dll
 IntelWifilhv04.dll

3.3.1 Netwtw04.INF

Type: Base Wi-Fi INF

Dependencies: No dependencies

Supported OS: Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S

Description: Installation of the Wi-Fi driver for the following supported hardware:

- Stone Peak (D0/D1 SKUs)
- Sandy Peak

How to: Installation methods same as Base INF. See Section 4, “UWD-compliant Installation Methods”.

3.3.2 netwtw04.cat

Type: A digitally-signed catalog file to *netwtw04.INF*

Dependencies: No dependencies

Supported OS: Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S



Description:	Contains a collection of cryptographic hashes, or thumbprints. Each thumbprint corresponds to a file that is included in the collection: <ul style="list-style-type: none">• <i>netwtw04.inf</i>• <i>Netfw04.dat</i>• <i>IntelWifiIhv04.dll</i>
How to:	Not relevant; part of the INF installation. See Section 4, “UWD-compliant Installation Methods”.

3.3.3 Netfw04.dat

Type:	An internal data file
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S
Description:	Contains the micro-code image of the Wi-Fi card. The file is encrypted and should be part of the Wi-Fi installation package.
How to:	Not relevant; part of the INF installation. See Section 4, “UWD-compliant Installation Methods”.

3.3.4 IntelWifiIhv04.dll

Type:	Run-time dynamic-link library (DLL)
Dependencies:	<i>netwtw04.inf</i>
Supported OS:	Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S
Description:	<p>Native 802.11 IHV Extensions DLL. Customizes and extends the functionality of 802.11 connectivity or security properties. It is the main pipe from user mode to kernel mode. Used by the PROSet/Wireless Wi-Fi software and the WAPI application to communicate with the Wi-Fi driver.</p> <p>For more information about the IHV Extensions DLL, see Native 802.11 IHV Extensions DLL.</p>
How to:	Not relevant; part of the INF installation. See Section 4, “UWD-compliant Installation Methods”.



3.3.5 netwtw04.sys

Type:	Driver
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S
Description:	The Wi-Fi driver. Supported hardware: <ul style="list-style-type: none"> • Stone Peak (D0/D1 SKUs) • Sandy Peak
How to:	See Section 4, “UWD-compliant Installation Methods”.

3.3.6 Netwfw06.dat

Type:	An internal data file
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S
Description:	Contains the micro code image of the Wi-Fi card. The file is encrypted and should be part of the Wi-Fi installation package.
How to:	Not relevant; part of the INF installation. See Section 4, “UWD-compliant Installation Methods”.

3.3.7 netwtw06.cat

Type:	A digitally-signed catalog file to <i>netwtw04.INF</i>
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S
Description:	Contains a collection of cryptographic hashes, or thumbprints. Each thumbprint corresponds to a file that is included in the collection: <ul style="list-style-type: none"> • <i>netwtw06.inf</i> • <i>Netwfw06.dat</i> • <i>IntelWifilhv06.dll</i>
How to:	Not relevant; part of the INF installation. See Section 4, “UWD-compliant Installation Methods”.



3.3.8 Netwtw06.INF

Type:	Base Wi-Fi INF
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S
Description:	Installation of the Wi-Fi driver. Supported hardware: <ul style="list-style-type: none">• Snowfield Peak• Windstorm Peak• Thunder Peak 2• Jefferson Peak 1• Jefferson Peak 2
How to:	Installation method same as Base INF. See Section 4, “UWD-compliant Installation Methods”.

3.3.9 netwtw06.sys

Type:	Driver
Dependencies:	No dependencies
Supported OS:	Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S
Description:	The Wi-Fi driver. Supported hardware: <ul style="list-style-type: none">• Snowfield Peak• Windstorm Peak• Thunder Peak 2• Jefferson Peak 1• Jefferson Peak 2
How to:	See Section 4, “UWD-compliant Installation Methods”.

3.3.10 IntelWifiIhv06.dll

Type:	Run-time dynamic-link library (DLL)
Dependencies:	netwtw04.inf
Supported OS:	Windows 10* Desktop (Home, Pro, Enterprise, etc.) and Windows 10 S
Description:	Native 802.11 IHV Extensions DLL. Customize and extend the functionality of 802.11 connectivity or security properties. It is the



main pipe from user mode to kernel mode. Used by the PROSet application to communicate the Wi-Fi driver.

For more information about the IHV Extensions DLL, see [Native 802.11 IHV Extensions DLL](#).

How to:

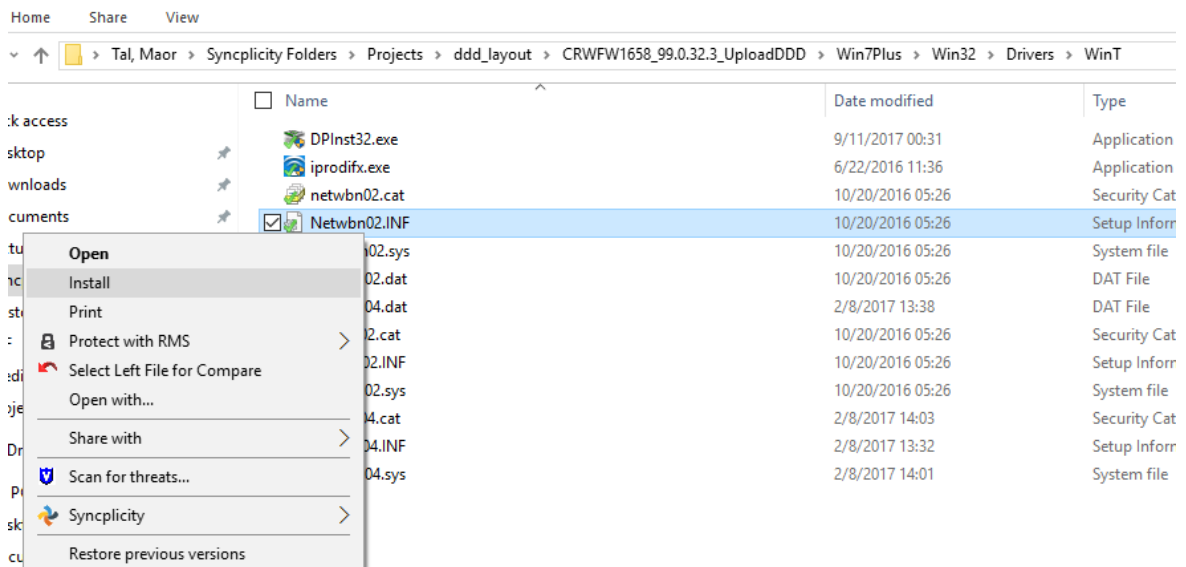
Not relevant; part of the INF installation. See Section 4, “UWD-compliant Installation Methods”.

4 UWD-compliant Installation Methods

This chapter defines the solutions for INF-based installations that are UWD-compliant.

4.1 Right-click installation

1. Open Windows Explorer.
2. Navigate to the relevant folder where the Base INF resides.
3. Right-click on the INF file.
4. Choose the 'Install' option.



4.2 PnPUtil-based installation

The PnPUtil utility (PnPUtil.exe) is included in every version of Windows.

To run PnPUtil, open a Command Prompt window (**Run as Administrator**) and enter commands using the syntax and parameters as described in [PnPUtil Command Syntax](#).

The following example adds a [driver package](#), which contains the file MyDriver.inf, to the [driver store](#):

```
C:\>pnputil /a m:MyDriver.inf
Microsoft PnP Utility

Processing inf: MyDriver.inf
Driver package added successfully.
Published name: oem22.inf
```




4.3 Pnputil – Install the driver from the driver store

The following example adds the driver package to the [driver store](#) and then installs the driver package on the computer

```
C:\>pnputil /a m:\MyDriver.inf /i  
Microsoft PnP Utility
```



5 Appendix

5.1 Wi-Fi customization table for drivers 04 and 06

Table 4 describes the customization parameters for drivers 04 and 06.

Table 4 – Wi-Fi customization table for drivers 04 and 06

INF Parameter	Description	Supported HW	Value Range	Example
Disable11ACForUkraine	Disable 11AC functionality for Ukraine	3165 – StP1 7265 - StP2 8260 - StP 18260 - DgP 17265 - MpP	1 – Disable 11AC in Ukraine All other – Enable 11AC in Ukraine	HKR,,Disable11ACForUkraine,0x00000000,"1"

5.2 Setup.xml configuration

The *Setup.xml* allows the OEMs to customize the Intel® PROSet/Wireless Wi-Fi software installation. Intel recommends keeping a copy of the original *Setup.xml* (provided in the layout). Once the XML file is customized, place it with the *Setup.exe* in the media layout.



5.2.1 XML schema

```
<?xml version="1.0" encoding="utf-8" ?>
<MWGInstallerSettings>
  <SetupConfigurations>
    <Configuration Id="default">
      <SelectableFeatures>
        <SelectableFeature Id="WIFI_Software" Selected="yes" />
        <SelectableFeature Id="WIFI_UI" Selected="yes" />
        <SelectableFeature Id="WIFI_Admin_Toolkit" Selected="no" />
      </SelectableFeatures>
      <PackageProperties>
        <Package Id="WiFi">
          <Properties>CCX_SUPPORT=FALSE</Properties>
          <Properties>EnableCpApplet=TRUE</Properties>
          <Properties>S5WOWLANEnable=FALSE</Properties>
        </Package>
      </PackageProperties>
    </Configuration>
  </SetupConfigurations>
</MWGInstallerSettings>
```



5.2.2 XML parameters and configurations

Table 5 describes the XML parameters and configurations.

Table 5 – XML parameters and configurations

Parameter	Category	Description	Values Range	Sample in the XML setting
WIFI_Software	Selectable Features under Configuration Id="default"	<p>This parameter controls installation of the WIFI_Software MSI.</p> <p>Intel® PROSet/Wireless Wi-Fi Software MSI will be installed.</p> <p>Note: please do not modify this parameter, it is not relevant to UWD</p>	<p>The Display show or not as a checkbox in the UI</p> <p>Selected – mean checkbox /radio Button Default value</p> <p>yes or no</p>	<SelectableFeature Id="WIFI_Software" Selected="yes" />
WIFI_UI	Selectable Features under Configuration Id="default"	<p>This parameter controls installation of the WIFI_UI software.</p> <p>Intel® PROSet/Wireless Configuration Utility will be installed</p> <p>Note: please do not modify this parameter, it is not relevant to UWD.</p>	<p>The Display show or not as a checkbox in the UI</p> <p>Selected – mean checkbox /radio Button Default value</p> <p>yes or no</p>	<SelectableFeature Id="WIFI_UI" Selected="yes" />
WIFI_Admin_Toolkit	Selectable Features under Configuration Id="default"	<p>This parameter controls installation of the WIFI_Admin_Toolkit.</p> <p>If set to "yes", the Administrator Toolkit will be installed.</p>	<p>to Install the Admin Tool need to change the Selected from "No" to "Yes"</p>	<SelectableFeature Id="WIFI_Admin_Toolkit" "Selected="yes" />



Parameter	Category	Description	Values Range	Sample in the XML setting
CCX_SUPPORT	Package id "Wifi" under PackageProperties	Enables or disables installation of Cisco authentication modules (EAP, PEAP, and EAP FAST).	True or False	<Properties>CCX_SUPPORT=FALSE</Properties>
EnableCpApplet	Package id "Wifi" under PackageProperties	Show the PROSet Icon in the control panel	True or False	<Properties>EnableCpApplet=TRUE</Properties>
S5WOWLANEnable	Package id "Wifi" under PackageProperties	Enable or disable S5 wake on system S5 mode. It will work only once operating system enable WOWLAN feature.	True or False	<Properties>S5WOWLANEnable=FALSE</Properties>

5.3 What a correct PROSet MSI installation via INF looks like

From an [MSDN AddSoftware article](#):

“When a software component INF file specifies **AddSoftware**, the system queues software to be installed after device installation.

There is no guarantee when or if the software will be installed.”

To verify that the software installed successfully, check for the existence of the following registry key:

```
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\DeviceSetup\
DeviceSoftware\<SoftwareName>
```

If the key doesn't exist at all, the OS never tried to run the PROSet MSI. If the key exists but there's no version information, the OS tried to run it and it failed. If the key exists and there's a version information value, the software ran successfully.

In case of error, you will need to verify the MSFT event logs. If there is any error in the execution it should be recorded in one of these log files:

- C:\Windows\System32\winevt\Logs\Microsoft-Windows-DeviceSetupManager%4Operational.evtx
- C:\Windows\System32\winevt\Logs\Microsoft-Windows-DeviceSetupManager%4Admin.evtx

After the PROSet software installation takes place, you can see an entry for PROSet under Programs and Features in the Control Panel as shown in Figure 88 and an icon in the Control Panel as shown in Figure 99.

Figure 8 – Programs and Features entry for Intel® PROSet/Wireless WiFi Software

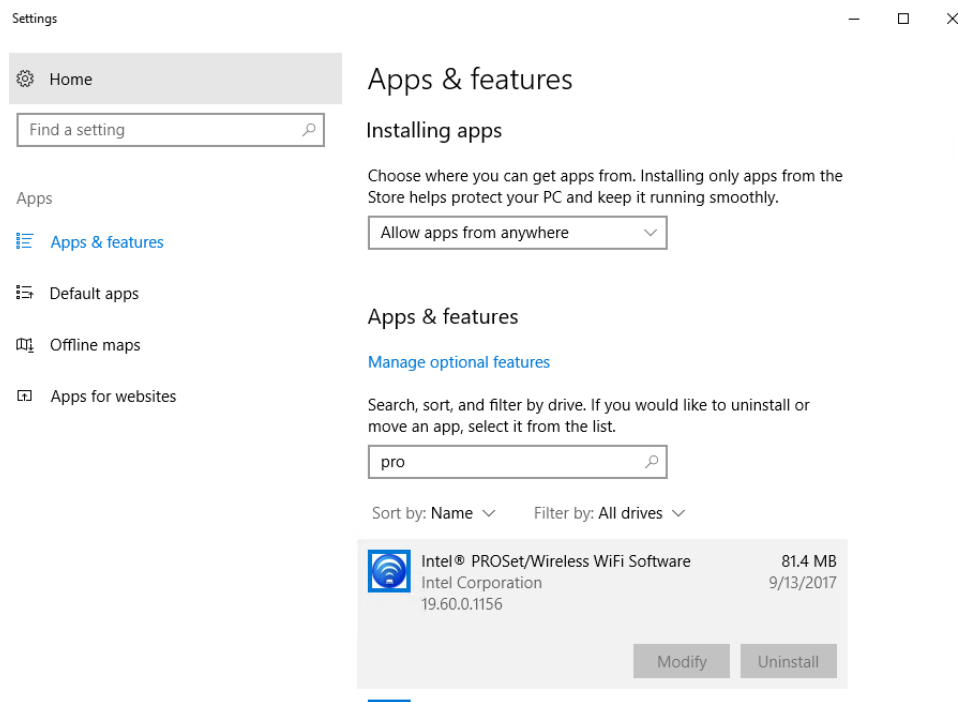
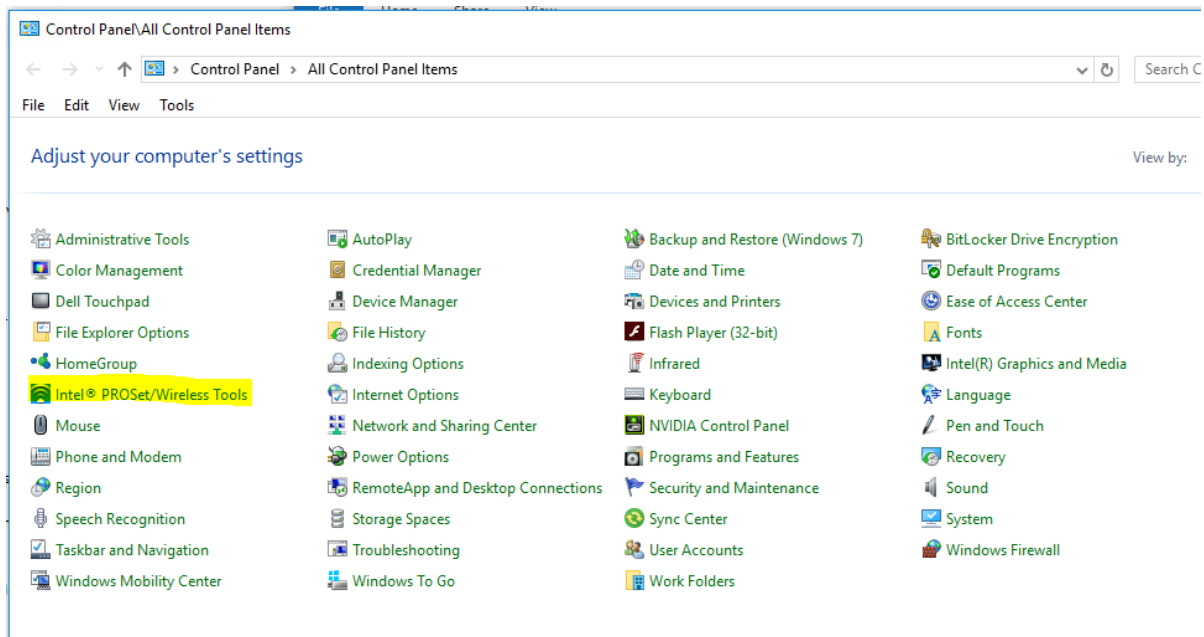
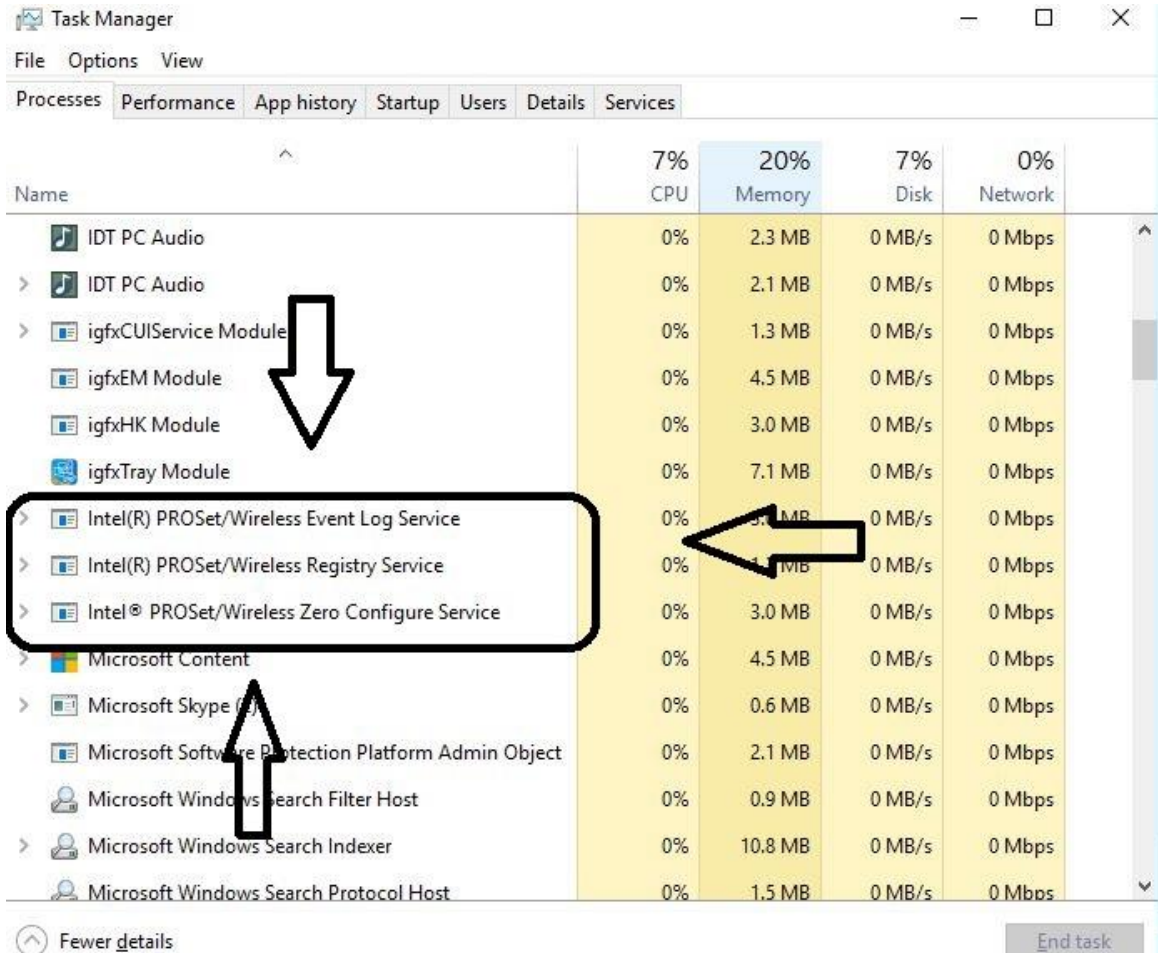


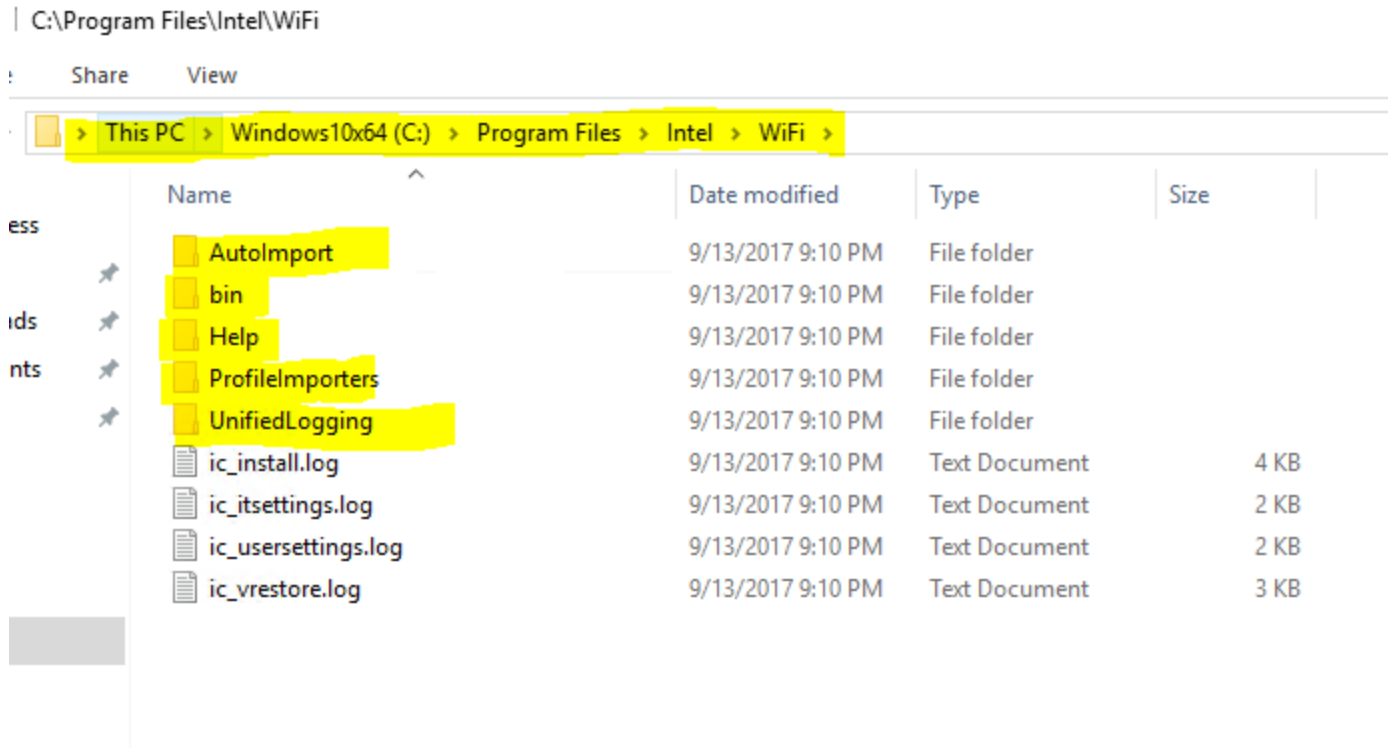
Figure 9 – Intel® PROSet/Wireless icon

Additionally, you can see the PROSet services running in the Windows* Task Manager, as shown in Figure 1010, and the PROSet folders installed in C:\Program Files\Intel\WiFi, as shown in Figure 111.

Figure 10 – Intel® PROSet/Wireless Wi-Fi Software services running



Name	CPU	Memory	Disk	Network
IDT PC Audio	0%	2.3 MB	0 MB/s	0 Mbps
IDT PC Audio	0%	2.1 MB	0 MB/s	0 Mbps
igfxCUIService Module	0%	1.3 MB	0 MB/s	0 Mbps
igfxEM Module	0%	4.5 MB	0 MB/s	0 Mbps
igfxHK Module	0%	3.0 MB	0 MB/s	0 Mbps
igfxTray Module	0%	7.1 MB	0 MB/s	0 Mbps
Intel(R) PROSet/Wireless Event Log Service	0%	3.0 MB	0 MB/s	0 Mbps
Intel(R) PROSet/Wireless Registry Service	0%	1.1 MB	0 MB/s	0 Mbps
Intel® PROSet/Wireless Zero Configure Service	0%	3.0 MB	0 MB/s	0 Mbps
Microsoft Content	0%	4.5 MB	0 MB/s	0 Mbps
Microsoft Skype (S)	0%	0.6 MB	0 MB/s	0 Mbps
Microsoft Software Protection Platform Admin Object	0%	2.1 MB	0 MB/s	0 Mbps
Microsoft Windows Search Filter Host	0%	0.9 MB	0 MB/s	0 Mbps
Microsoft Windows Search Indexer	0%	10.8 MB	0 MB/s	0 Mbps
Microsoft Windows Search Protocol Host	0%	1.5 MB	0 MB/s	0 Mbps

Figure 11 – Intel® PROSet/Wireless Wi-Fi Software folders installed


5.4 Uninstalling UWD PROSet

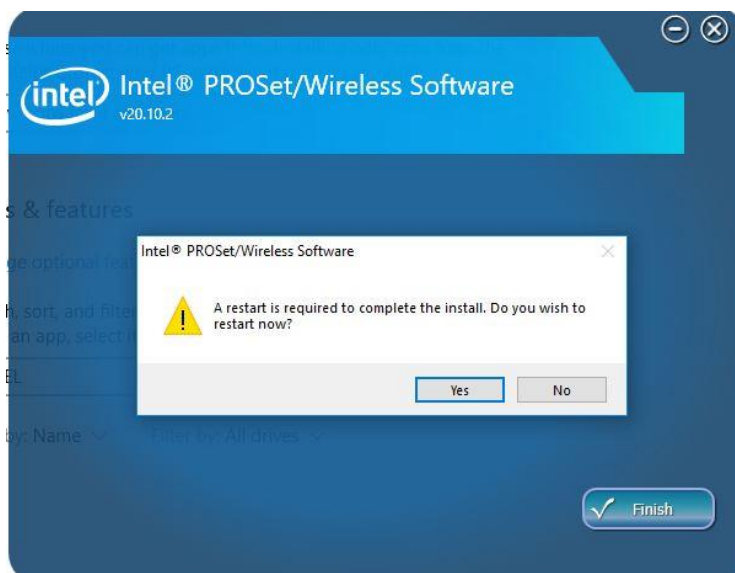
PROSet uninstallation consists of two phases; to enable a full removal and to enable a successful re-install, follow these steps:

1. Uninstall via the PROSet GUI, from the Control Panel, or from “Programs and Features.”
2. Uninstall via the Device Manager.

5.4.1 Uninstalling via the PROSet GUI

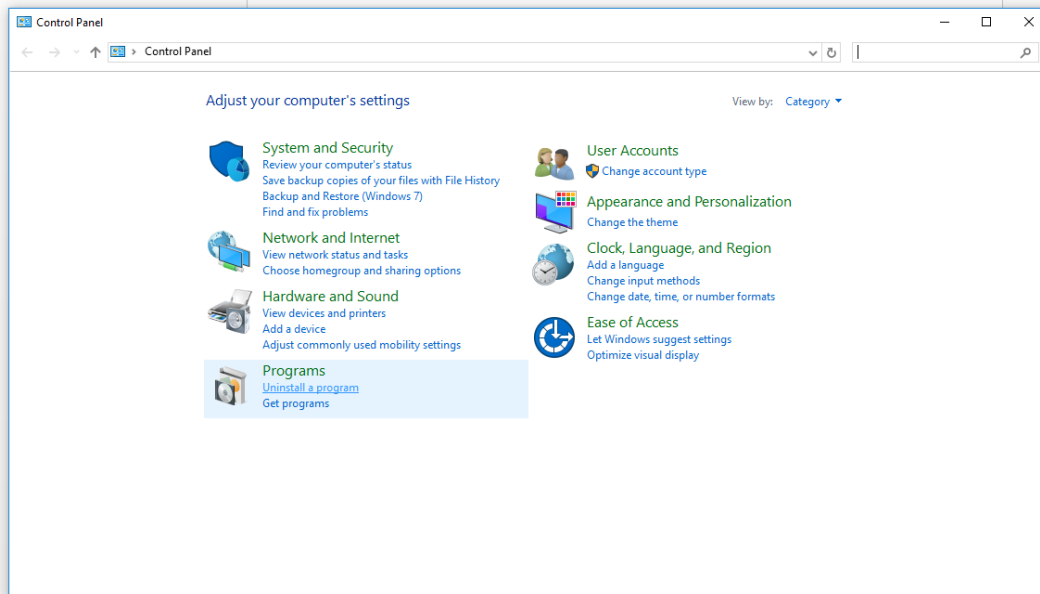


Optionally, a reboot might be required.

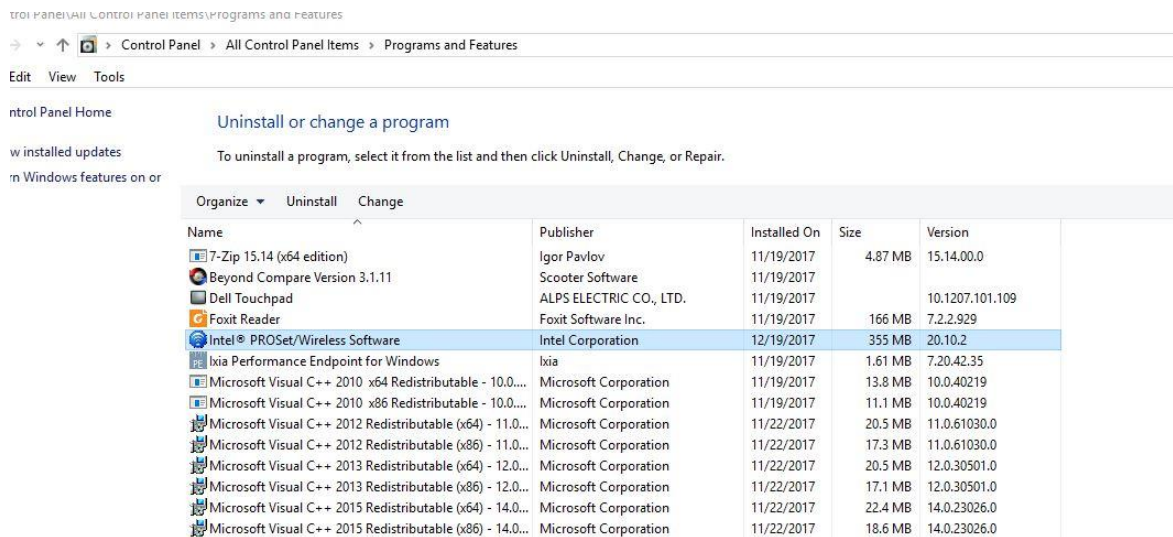


5.4.2 Uninstalling from “Programs and Features”

1. Open the Control Panel and choose “Uninstall a program.”

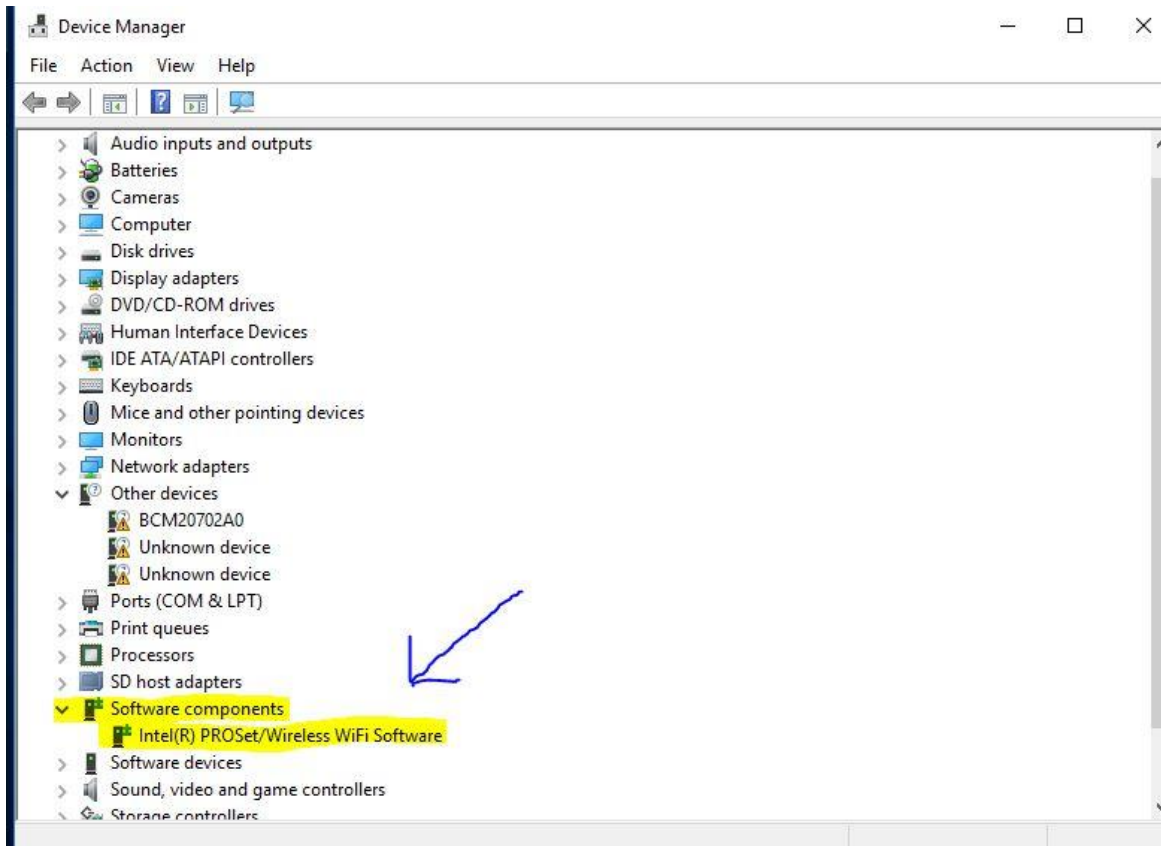


2. Choose Intel PROSet in the list. Then, right-click and select “Uninstall.”

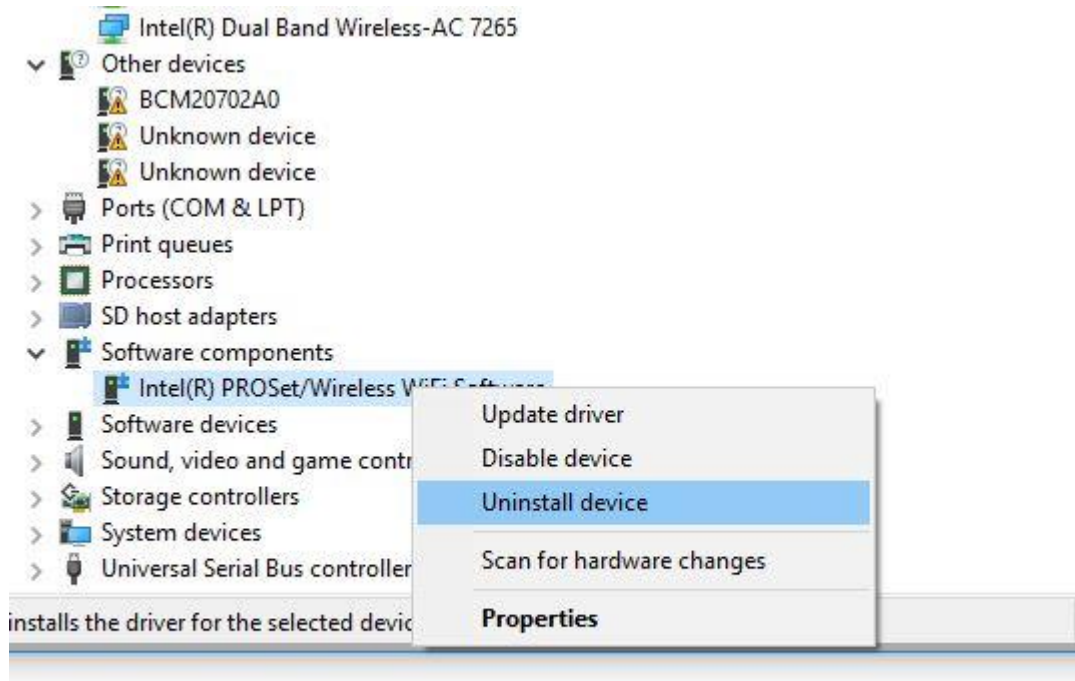


5.4.3 Uninstalling from Device Manager

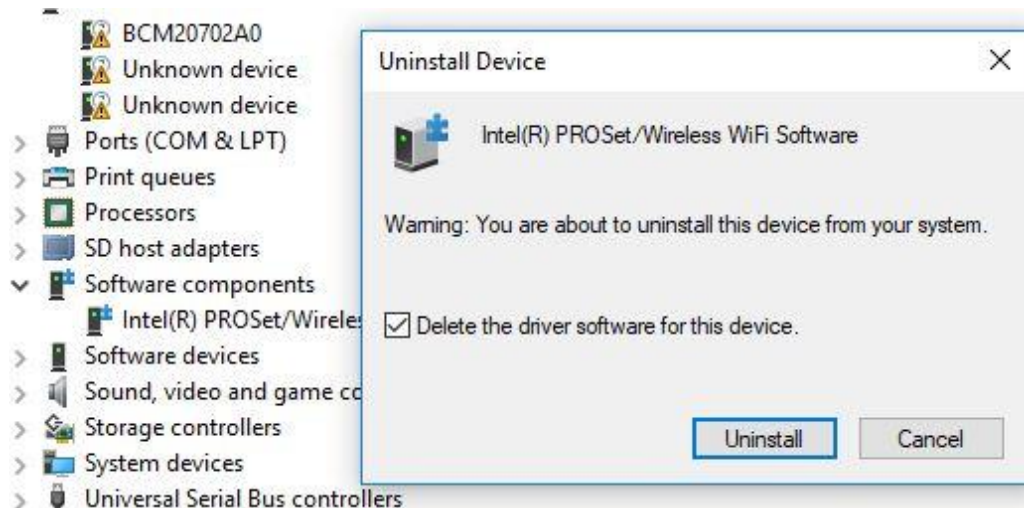
1. Open the Device Manager and locate the PROSet component under “Software components.”



2. Right-click on the “PROSet” node and choose “Uninstall device.”



3. Select the check box to select “Delete the driver software for this device.” Then, click the “Uninstall” button.



After uninstalling, the ‘Generic software component’ under ‘Software components’ will still appear in Device Manager. There seems to be no way to remove this entry.

5.5 Driver Update Acceptable (DUA) process

This DUA process is available on Microsoft's DevCenter portal. For more details see [Manage Hardware Submissions](#).

1. From **Manage submissions**, click the appropriate submission ID.
2. Under the **Download** heading, click **DUA shell package**, and then click **Save**.

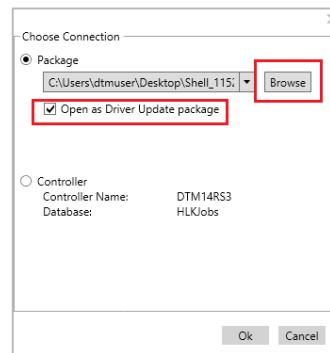
Packages and signing properties

Upload new Download DUA shell

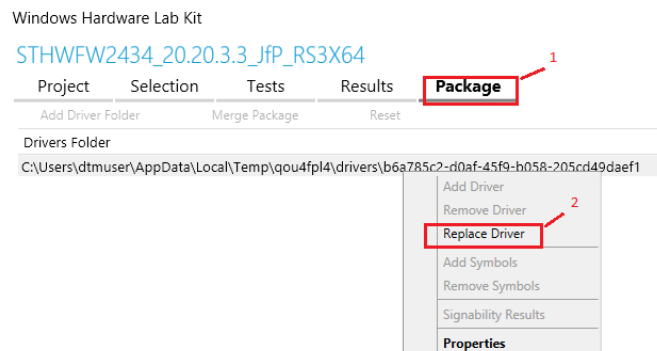
3. From HCK Studio, click **Connect**.

Help | Configuration **Connect** | About

4. Select **Package**, and then click **Browse** to open the existing DUA shell package.
5. Under Packaging Options, select Driver Update.
6. Click **OK**.



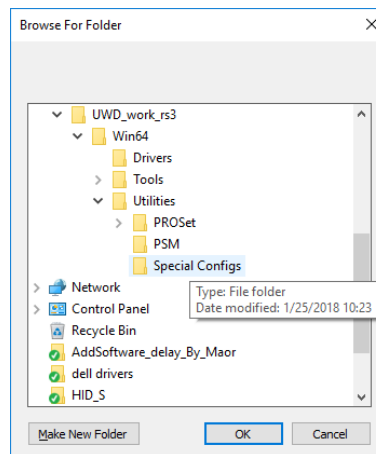
7. On the **Package** tab, right-click the appropriate driver folder, and then click **Replace Driver**.



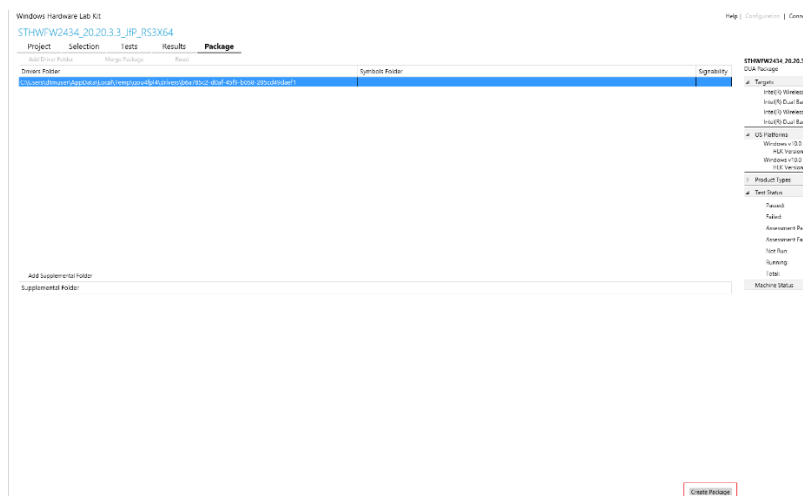


8. Navigate to the relevant folder that contains the Extension INF that needs to be customized, and click **OK**:

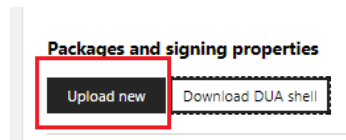
- “Utilities\PROSet”
 - to update the ExtensionId in the ‘ProSetExtension.INF’ file
 - and—if desired—to modify any of the fields in the Setup.xml
- “Utilities\Special Configs”
 - to update the ExtensionId in the WifiDrv04Customizations.INF or in the WifiDrv06Customizations.INF
 - and—if desired—to update any of the values in the WifiDrv04Customizations.INF or WifiDrv06Customizations.INF



9. Click **Create Package**.



10. Go back to **Manage submissions** in the dashboard, search by ID, and click **Upload driver update (DUA)**.

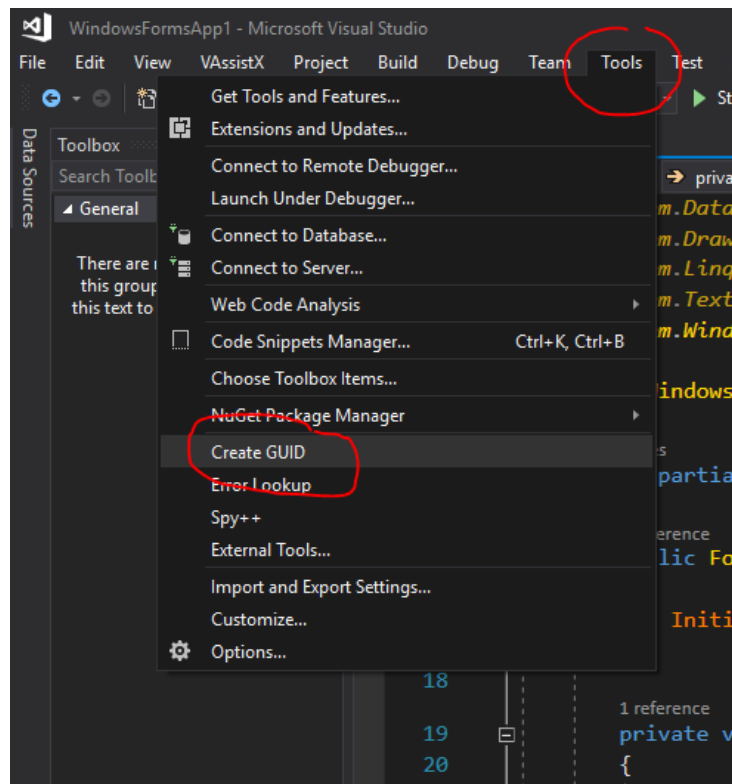


11. Use the new **.hckx** file that was created by HCK Studio and complete the DUA wizard.
12. Once all stages are marked with a green check mark, the DUA package should be ready for download.

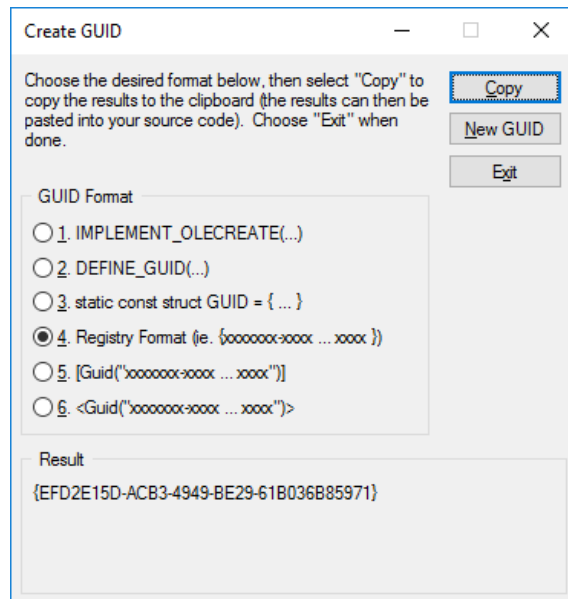
5.6 How to generate a GUID

5.6.1 Via MSFT Visual Studio

1. Open Visual Studio.
2. Select **Tools** → **Create GUID**.



3. In the resulting Create GUID dialog box, Select “4. Registry format” and click **Copy**.



5.6.2 Online generator

You can use the following online GUID generators:

- <https://www.guidgenerator.com/>
- <https://www.guidgen.com/>