Microsoft

Windows 10
IoT Core Overview

Hancom MDS Inc.

Windows 10 For IoT Goals

Make Windows the optional platform for IoT devices

Increase velocity for builders of Windows devices

Build mindshare with Makers and Hobbyists

Provide turn-key services for Intelligent systems

Enable IoT devices to more easily participate in the IoT

tuned to each form factor…

Windows for PCs

Familiar desktop shell

Broad hardware ecosystem

Windows desktop application compatibility

Windows for Mobile

Familiar mobile shell
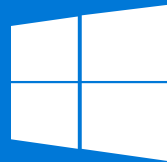
Rich telephony

Windows phone app compatibility

Windows on Xbox

10' shell experience

Shared gaming experiences

Xbox One game and app compatibility

Windows for …

Form factor–appropriate shell experience
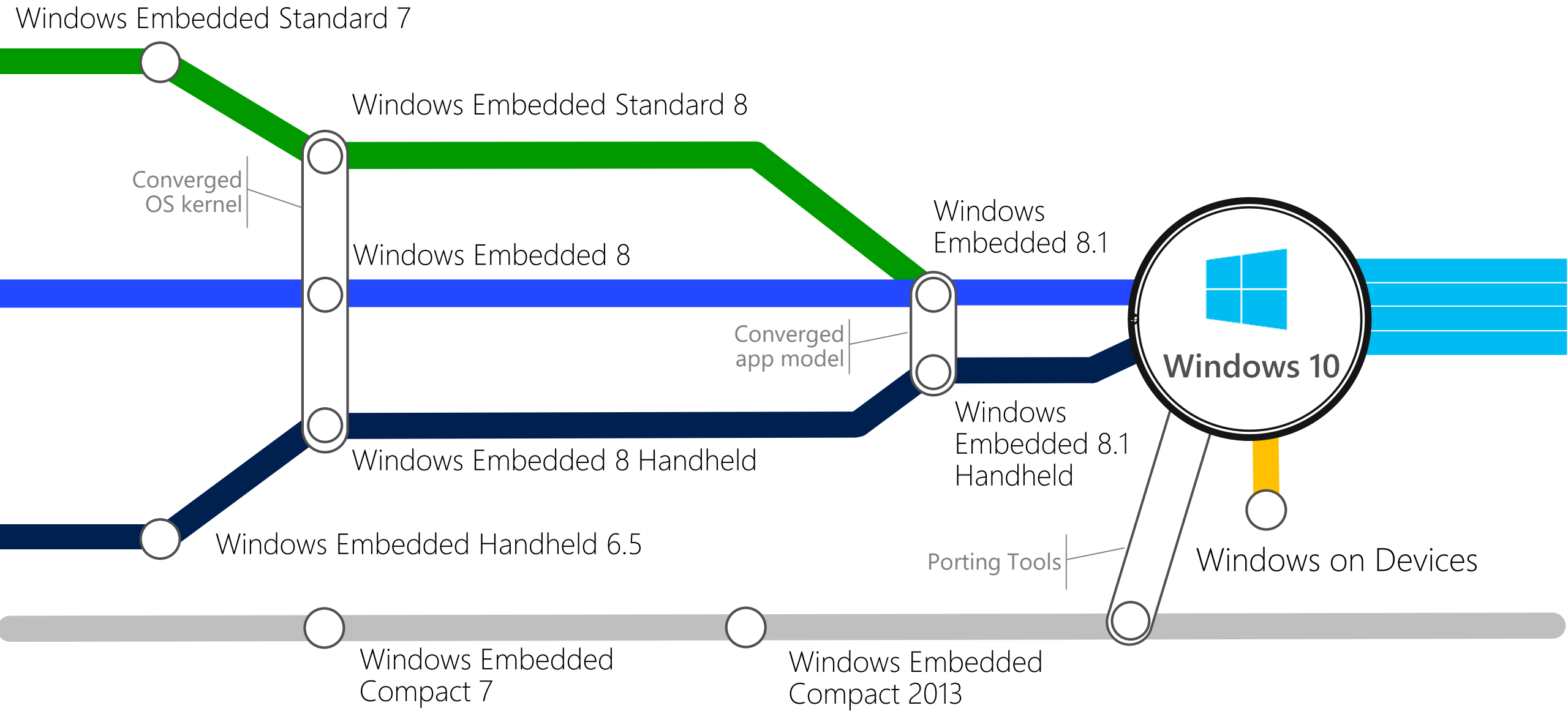
Device-specific scenario support

One Core OS

Base OS

App and Device platform

Runtimes and frameworks

...with one app platform

Mobile

PC

HoloLens

Xbox

Surface Hub

Devices +IoT

Multiple device families

Adaptive
user interface

Common APIs
and SDK

Natural
user inputs

Common
toolset

Common
store
and dev center

One App Platform

Platform Convergence Journey

Windows Embedded Standard 7

Windows Embedded Standard 8

Converged
OS kernel

Windows Embedded 8

Windows
Embedded 8.1

Converged
app model

Windows Embedded 8 Handheld

Windows
Embedded 8.1
Handheld

Windows Embedded Handheld 6.5

Windows 10

Porting Tools

Windows on Devices

Windows Embedded
Compact 7

Windows Embedded
Compact 2013

Windows 10 IoT Editions

Windows 10 IoT Enterprise

Desktop Shell, Win32 apps, Universal apps and drivers
Minimum: 1 GB RAM, 16 GB storage
X86/x64

Windows 10 IoT mobile

Modern Shell, Mobile apps, Universal apps and drivers
Minimum: 512 MB RAM, 4 GB storage
ARM

Windows 10 IoT Core

Universal Apps and Drivers
No shell or MS apps
Minimum: 256MB RAM, 2GB storage
X86/x64 or ARM

+

Windows Updates

Visual Studio & UWP

New User Interfaces

Security & Identity

Integrated Device Connectivity

Microsoft Azure IoT

Windows 10 IoT Enterprise

Bringing Windows to a Standard Class of IoT Device

Build on x86

The full version of Windows with advanced lockdown capabilities, Win32 compatibility and modern app support for standard industry devices.

Universal App Support

Windows 10 IoT Mobile Enterprise

Bringing Windows to a Mobile Class of IoT Device

Build on ARM or x86

A version of Windows for Industry Devices that need mobility, rich recognizable user experiences, excellent battery life.

Universal App Support

Windows 10 IoT Core

Bringing Windows to a Compact Class of IoT Device

Build on ARM or x86

An optimized small footprint version of Windows for resource and cost constrained devices with the full power of modern application development with Universal Apps.

Feedback

Feedback	Windows 10 IoT Core
Products are fragmented, tools are out of date. It's difficult to develop apps because of the variety of OS offerings.	Consolidating onto Threshold OS core, tools & processes and one standardized Windows 10 IoT Core image for all products
Activation is a blocker	No activation required
Must NOT have any Windows / Microsoft UI – period	period Windows 10 IoT Core is completely UI-less – No Apps, shell or branding
A significant percentage of new IoT devices will be headless and UI will be remoted to Smartphones, Tablets and web UI.	Node.JS, Python, .net, Cloud connectivity
The ability to build consistent app experiences across multiple devices	Universal App Platform, XAML, HTML5, Cloud Connectivity
OS must be secure and maintainable over time, kept current with other OS offerings	Windows security. Supports update/servicing

Windows 10 IoT Core

Small footprint version of Windows 10

o   Provides capabilities customers already love about Windows: security, update, tools, apps, manageability targeting devices

o   Uses the same development tools for all Windows 10 devices

o   Universal app support across Windows client devices – PC, Phone, Xbox, IoT Clients

Targeting a wide range of low cost IoT devices

o   Common image, common SDK: no more costly & confusing custom images and custom SDKs

o   Support for wide range of devices with different capabilities and price points

SoCs and developer boards widely available

o   Low BOM cost and small footprint enables a range of low cost SoCs/boards

o   Low cost hobbyist devices and tools

Windows 10 IoT Core

Supports both x86 and ARM chipsets

- Ships initially with chipsets supported by Mobile and Desktop Windows with more support coming online post.
- A "BSP kit" will be made available so other SV / OEMs can bring up chipsets & systems on their own supporting Windows 10 IoT Core.

Minimum Hardware Requirements:

- 400 MHz or faster. (x86 requires PAE, NX and SSE2 support.)
  - Headless
    - 256M RAM (128 MB free to OS) / 2 GB Storage
  - Headed
    - 512 M RAM (256 MB free to OS) / 2 GB Storage Support for wide range of devices with different capabilities and price points

Personal Targets for Windows 10 IoT Core

OEM / SV / Device Driver Developer

- Implement, build, test, deploy device drivers for devices/sensors using Visual Studio, Windows Driver Kit etc.
- Use Windows 10 IoT Core as a device bring up OS

Application Developer

- Implement, build, test, deploy modern device experiences and are connected to Microsoft IoT services with Universal Windows Platform App Support

System Builder / Maker / Integrator

- Create IoT devices & drivers and build applications connected to Microsoft IoT services

Personal Targets for Windows 10 IoT Core

System Administrator

- Deploy IoT devices and related applications in my enterprise
- Monitor, manage and troubleshoot devices, sensors and applications
- Update devices, sensors and applications to ensure system is current and secure.

Device Categories

Headed devices have a video display and uses the Windows Video subsystem & drivers to address it.

- Headed devices have what would be recognized as a "typical" consumer display as you might see on a laptop or tablet.

Headless IoT devices have no display.

Headed vs Headless is controlled by configuration settings (read at boot time.)

Branding

Windows 10 IoT Core based devices will not have an identifiable Windows experience or Microsoft Branding

No "inbox" / "prepopulated" Windows applications or shell.

Headed devices require custom OEM supplied application / user experience for the device exposing the specific device functionality

- Shell infrastructure provided to manage Modern application lifecycle

Targeted Boot Experience

Boot straight into desired app
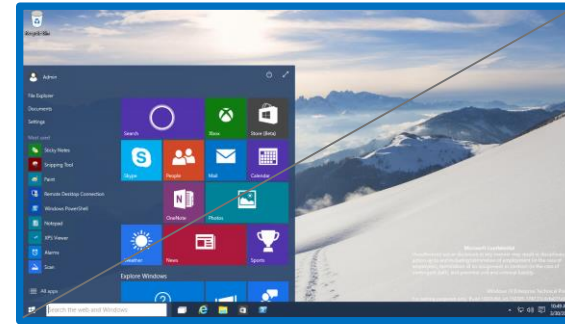
No Microsoft or Windows Branding

Easily create custom device experiences

Minimum System Requirements Comparison

Component	Windows 10 IoT for Industry Devices	Windows 10 IoT for Mobile Devices	Windows 10 IoT Core
Processor	x86	x86 and ARM	x86 and ARM, 400MHz or faster (600MHz for modern UI support)
RAM	1GB for 32-bit processors 2GB for 64-bit processors	512MB for 32-bit processors 2 GB for 64-bit processors (dependent on display type)	256MB for 32-bit processors (512MB for modern UI support)
Storage	Storage = 16GB (20 GB for 64-bit processors) SD card = Optional	Flash = 4GB Flash user partition = 1GB SD card = Optional	Flash/SD = 2GB
Display	XGA (1024 x 768) or higher with 32 bits of color per pixel (1024 x 600 scaling in driver layer)	WVGA (800x480) or higher with 16 bits of color per pixel	None or Frame buffer graphics and 2D optional (720p HDMI / 1080p+ HDMI / 3D GPU optional for modern UI support)
Audio	Optional	Primary, Loudspeaker and Earpiece required for phones and tablets. Optional for other devices	Optional
Connectors	Optional	Headphone/headset jack required for phones/phablets. Optional for other devices	Optional
Wireless	Optional	802.11b/g required for phones/phablets. Optional for other devices	Optional
Accelerometer & Proximity Sensor	Optional	Required for phones/phablets. Optional for other devices	Optional
Touch (Under Review)	Two-Finger touch Must be HID compliant Drivers and 3rd party HID miniports for the solution must be available on WU, factory image and functional in WinPE Solution must report all genuine contacts to the operating system.		Optional

Sharks Cove & MinnowBoard MAX

Both boards support Windows 10

Sharks Cove

- Intel ® Atom™ Processor Z3735G, 2M Cache, 4 Core, 1.33GHz up to 1.88GHz
- Supports Connected Standby
- 32-bit UEFI firmware
- Headers for Camera, MIPI Display, USB, I2C, SDIO, UART, GPIO, UART-to-USB for debug

MinnowBoard MAX

- Intel® Atom™ E3800 processor
- 64-bit & 32-bit UEFI firmware
- Can also be used as an UEFI Development Kit
- PWM capable GPIO (2 pins of 8 total GPIO)
- Open Hardware Platform (Gerbers & Layout)
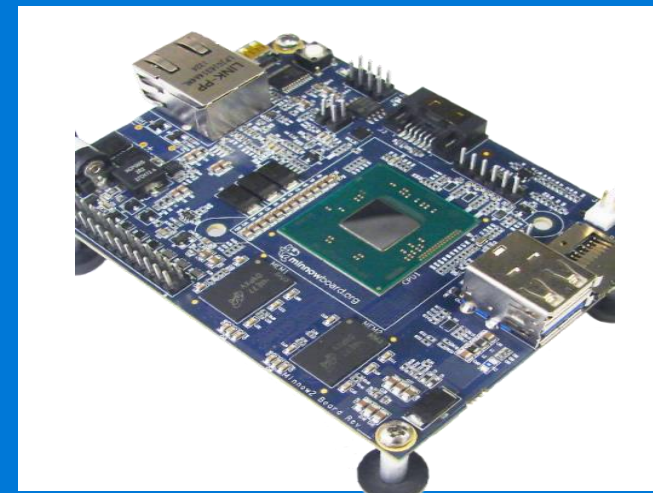
Raspberry Pi 2

Bringing the power of Windows to the Maker community

Hardware specs:

- Broadcom 2836 900MHz quad-core ARM Cortex-A7 CPU

- 1GB LPDDR2 SDRAM

- MicroSD, Ethernet, USB, HDMI

- GPIO, I2C, I2S, SPI

Attend the session on Building Devices with Windows IoT to learn more

Qualcomm DragonBoard™ 410C

QUALCOMM®

Build innovative solutions using Windows & Qualcomm Snapdragon

Hardware specs:

- Qualcomm Snapdragon 410 (APQ8016)

- 1GB LPDDR3, 4GB eMMC

- MicroSD, WiFi 802.11a/b/g/n, BT4.1 + LE, GPS

- GPIO, I2C, I2S, SPI
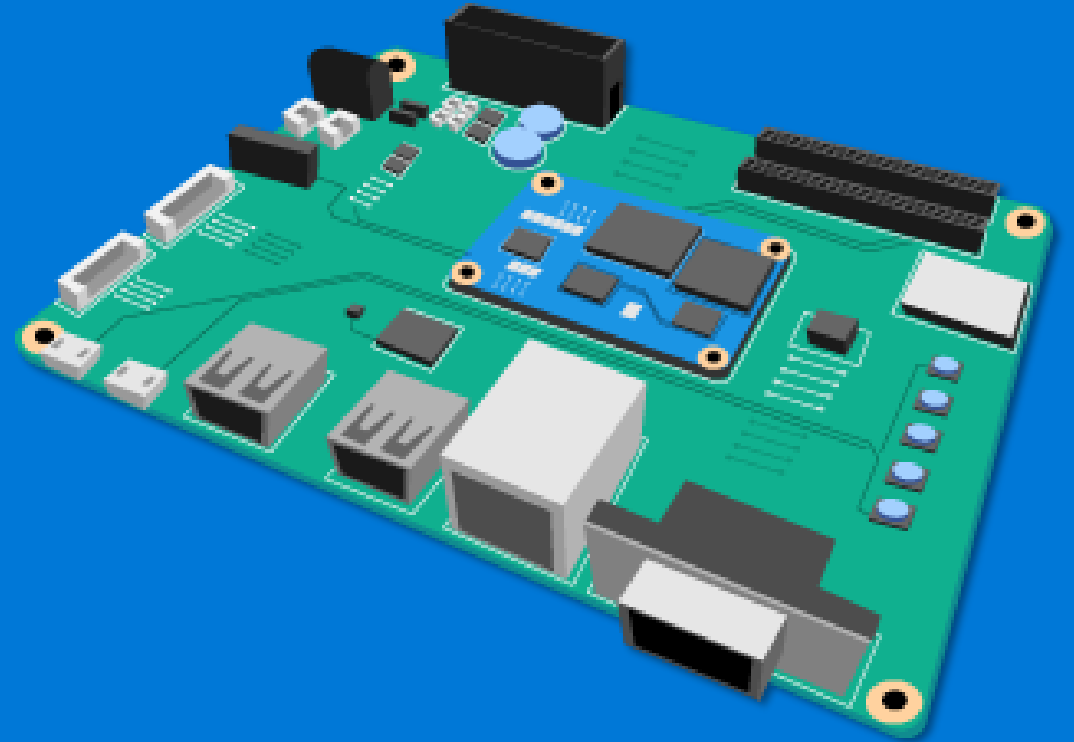
Lockdown Features

Windows 10 Lockdown

Consistent and predictable device lockdown across form factors

bilities provide
ayer of security
vice experience.

Lockdown Capabilities

Lockdown Capability	Windows 10 IoT Core	Windows 10 IoT Mobile	Windows 10 IoT Enterprise
Write Filters and Overlays	√	-	√
USB Filter	√	√	√
Dialog and Notification Filters	n/a	√	√
Input Filters	n/a	√	√
AppLocker and Layout Control	n/a	√	√
Shell and App Launcher	n/a	√	√

Windows 10 IoT Core UWP

Windows IoT Core API Surface

IoT Core supports the Universal Windows Platform APIs, including Universal Applications and Universal Drivers.

IoT Core also contains a number of IoT targeted API extensions:

- Access to Devices: GPIO, I2C, SPI, ADC, PWM, UART, Alljoyn
- System Management: Wireless network config, time zone settings, input language, system shutdown/reboot/hibernate

Universal Windows Platform APP

- **Converged** APIs, write **ONE** Universal App and target all Windows 10 editions
- **Scale** and get **higher ROI** by selling same App to all Windows 10 editions OEMs/ODMSs
- Reuse **existing development skills**

Windows Universal Platform
Common & Consistent APIs

Languages	**UI Frameworks**	**APIs**	**Deployment and Execution**	**Tools**
• C++ /CX	• HTML	• WinRT	• APPX	• Visual Studio
• C#, VB	• Xaml	• Win32	• XCopy	• PowerShell
• JS	• DirectX	• .NET	• App Isolation	
• Python		• Wiring		
• Node.js				

Easily Build Retail Line of Business Solutions
Retail Peripherals Supported Inbox

8.1+	**8.1+**	**NEW**	**NEW**	**3rd Party Enabled**
Barcode Scanner	**Mag-Stripe Reader**	**Receipt Printer**	**Cash Drawer**	**Payment Terminal**
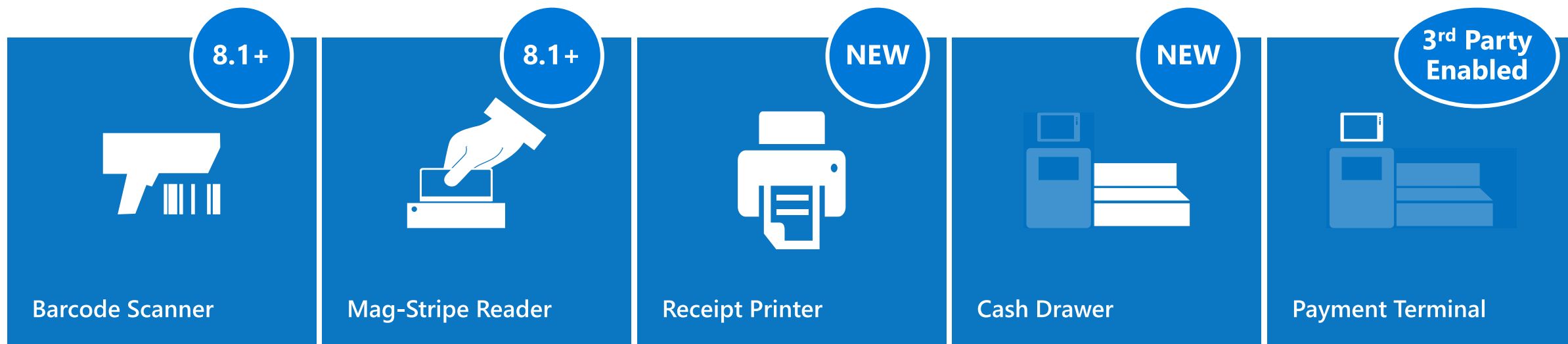
- APIs in Windows 10 SDK and DDK
- Adapted from UnifiedPOS standard

- 3rd provided libraries

Building IoT Devices with UAP

"Embedded" Mode

- Extend UAP to IoT capabilities on all Windows 10 editions

Access to system settings

- APIs to change system settings such as power state, radio control and Bluetooth.

APIs to access busses

- GPIO, I2C, SPI and easy access to custom hardware

Background Services for long running tasks

- Full control for your device, free of standard Process Lifecycle Management

Porting Apps/Drivers

- UAP/UD **API surface is rich** but smaller compared to Windows desktop
- Use **App Migration Tool** to analyze compatibility of your apps

If you are using (not supported on Small Devices) →	**Instead use**
App logic and code	
Win32/Native →	Win32/Native in OneCoreUAP.lib (subset of Desktop API surface)
.NET libraries →	.NET libraries supported in UWP App (subset of Desktop .NET APIs)
Graphic User Interface	
GDI, MFC, WinForms, WPF →	XAML, DirectX, HTML

Windows 10 IoT Core covers

Windows 10 IoT Core APIs:

- Driver Development
- Native Console App
  - **NT Service**
  - **Task Scheduler**
- UWP Applications
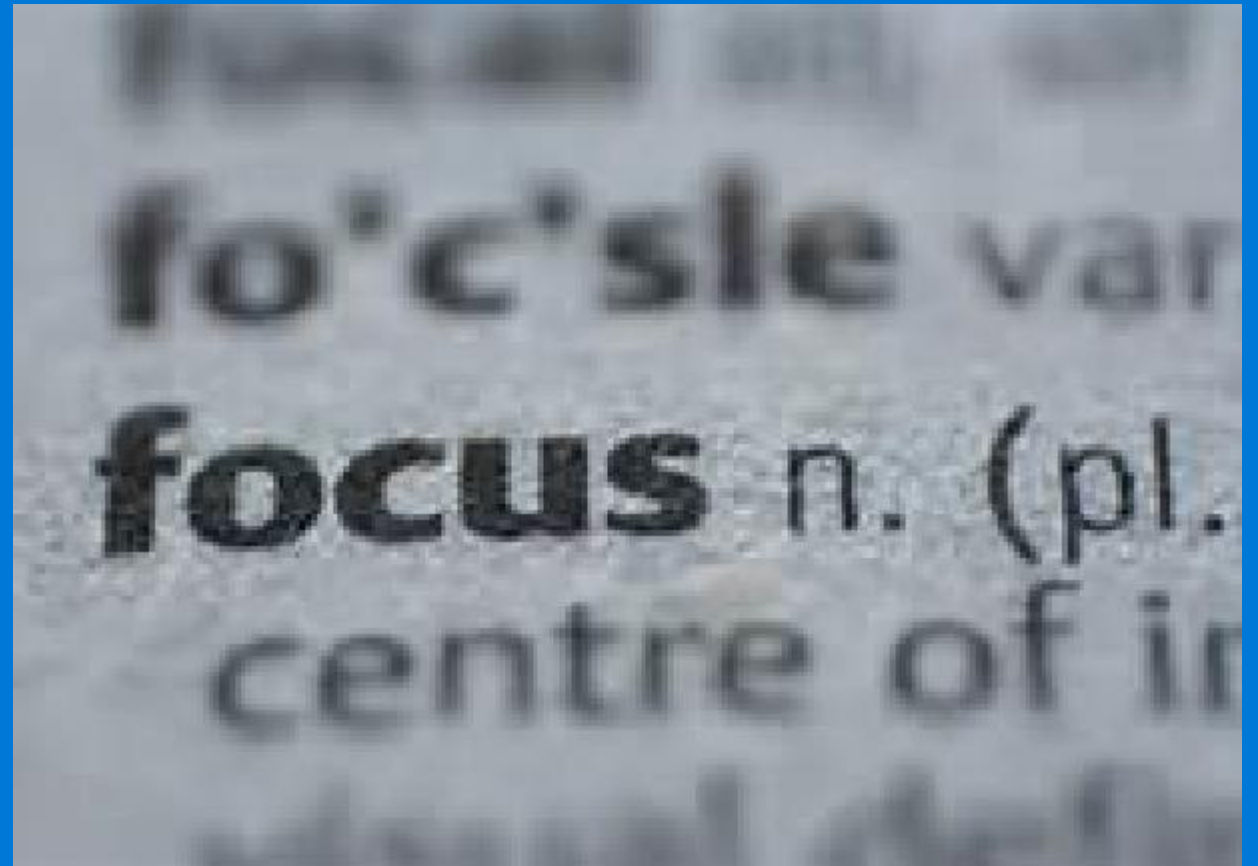
Porting Existing Applications

A few things to think about:

- Win32/GDI is not supported, use XAML, DirectX, HTML
- Win32/native is limited to OneCore.lib + additional O/S libraries
- MFC is not supported since this relies heavily on Win32/GDI – STL is fine though!
- .NET is supported (through UAP), WinForms is not supported
- WPF is not supported, XAML in UAP is supported
- Win32/native application Binary compatibility is not a focus area
  - Use Desktop/Industry + lockdown functionality

OneCoreUap.lib API List :
https://msdn.microsoft.com/en-us/library/windows/desktop/mt657573(v=vs.85).aspx

Single LoB App Model

Visual Studio 2015

Modern app dev experience

Single UWP

Multiple UWP background tasks

Win32 background tasks / Services

Universal Driver Concept

Universal Windows Platform Driver

Develop to a single API/DDI
Same API for Desktop, Tablet, Phone, IoT

Build once for all Windows editions
One binary per *instruction set*, not per *device family*

Runs on all Windows devices
Phone, Tablet, Desktop, Server, IoT
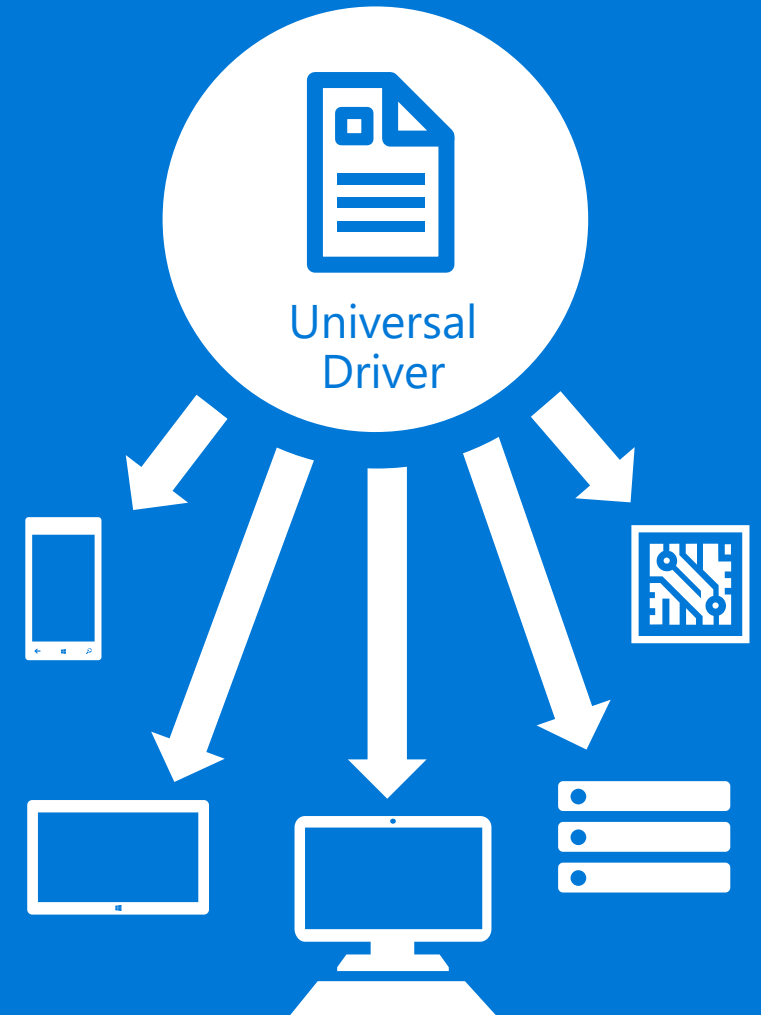
Universal
Driver

Universal Drivers

Universal Drivers (UD)

- All Drivers Included Win10IoT Core are UD.
- A newly developed driver using the UD API surface in WDK
- An existing driver that only uses the UD API set is considered UD

Windows 10 driver development tools and information will target and include IoT Core as one of the UD target SKUs.

- We will recommend IoT Core as driver bring up OS.
- Driver Development Will be part of the overall IoT Core development story

Universal Windows Platform Driver

- Write **ONE** Universal Driver and target all Windows 10 editions - Converged device areas/APIs
- **Scale** and get **higher ROI** by selling same components to all Windows 10 editions OEMs/ODMSs
- We scanned over **100k drivers** to create a **universal driver API set**
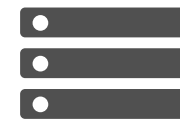
Windows Universal Platform
Common & Consistent Device Driver APIs

WDF
Audio
Bluetooth
Buses (USB, SPB)
HID(Retail), Buttons
Camera
Graphics & Display

Location
Networking - Wired
Networking - WLAN
Security - Biometrics
Security - Crypto
Security - Smartcard
Security - TPM

NFC
Sensors
Thermal
Touch
UEFI
Video

Easily Build Universal Drivers

| Download **Visual Studio** & **WDK** | Build and Debug the **Universal Driver** on PC | Test Driver using WDK Test | Validate on dev board | Submit for signing |

Universal Driver samples & templates available as a starting point

Move to Universal Driver, run on more devices

If you are using	Actions to take	Why
Inbox/Class drivers	• It just works! core device types Storage, mouse, keyboard, touch, video,…	Your device automatically leverages a large ecosystem of peripherals
Kernel Mode drivers	• High backwards-compatibility for converged device areas • Make minimal changes and test	Your driver runs on more editions
User Mode drivers and services	• Know that Windows Universal Platform Win32 API surface is smaller than desktop Windows • Use replacement APIs where available • Re-design/re-implementation if APIs are not available and test	Your driver runs on more editions

WDK with Windows 10

- **One WDK** for all Windows driver development
- Multiple versions of the WDK can exist side-by-side on disk
- VS+WDK provides a consistent driver development experience across all target Windows editions
- Build, debug & validate Windows Universal drivers easily
- Running WDK tests standalone on target machine

Setting up VS to build Windows Universal Driver

- Access Driver Setting option under Driver Properties in Visual Studio
- Set TargetPlatform = Universal
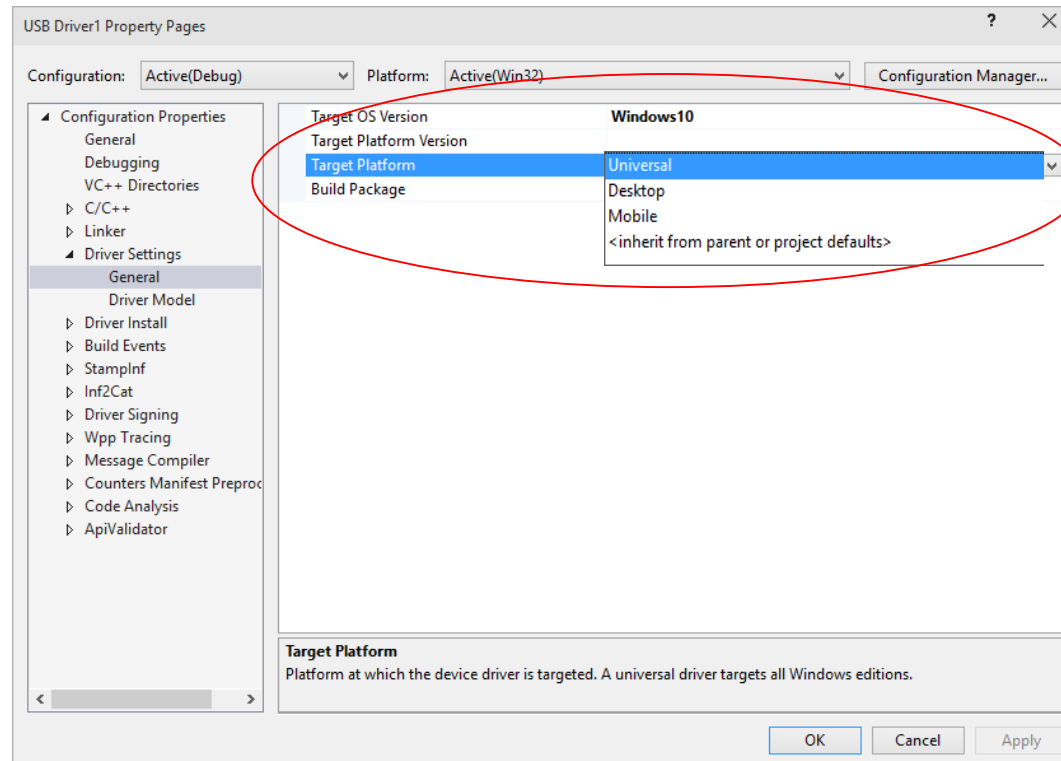
Windows Universal driver validation tool

APIValidator.exe tool

- Runs as a post build process for Windows Universal Drivers
  - Also can be run on command line
    - "C:\Program Files (x86)\Windows Kits\8.2\bin\x86\apivalidator.exe"
- Flags APIs used in the driver project that aren't part of the Windows Universal Platform
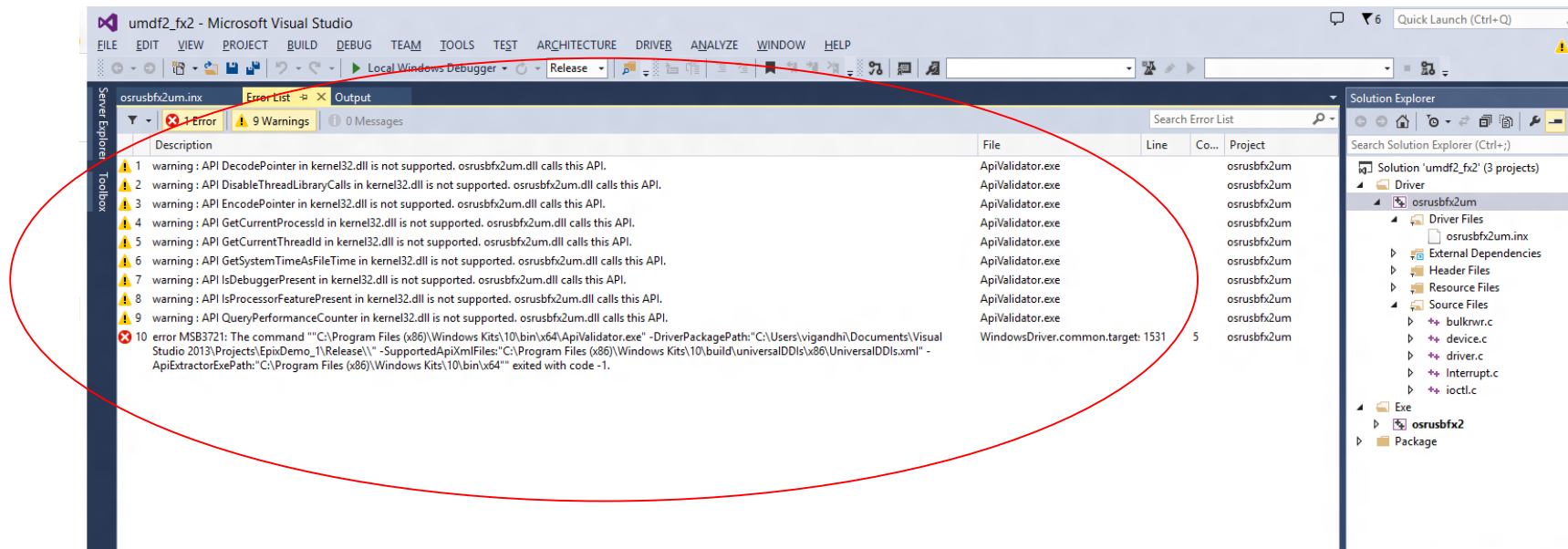
Demo: Universal apps, drivers and dev boards

Sensor app

Windows API	Windows API	Windows API
Windows for PCs	Windows for IoT	Windows for Phones
Windows DDI	Windows DDI	Windows DDI
Sharks Cove	**MinnowBoard Max**	**Dragonboard**

Sensor driver
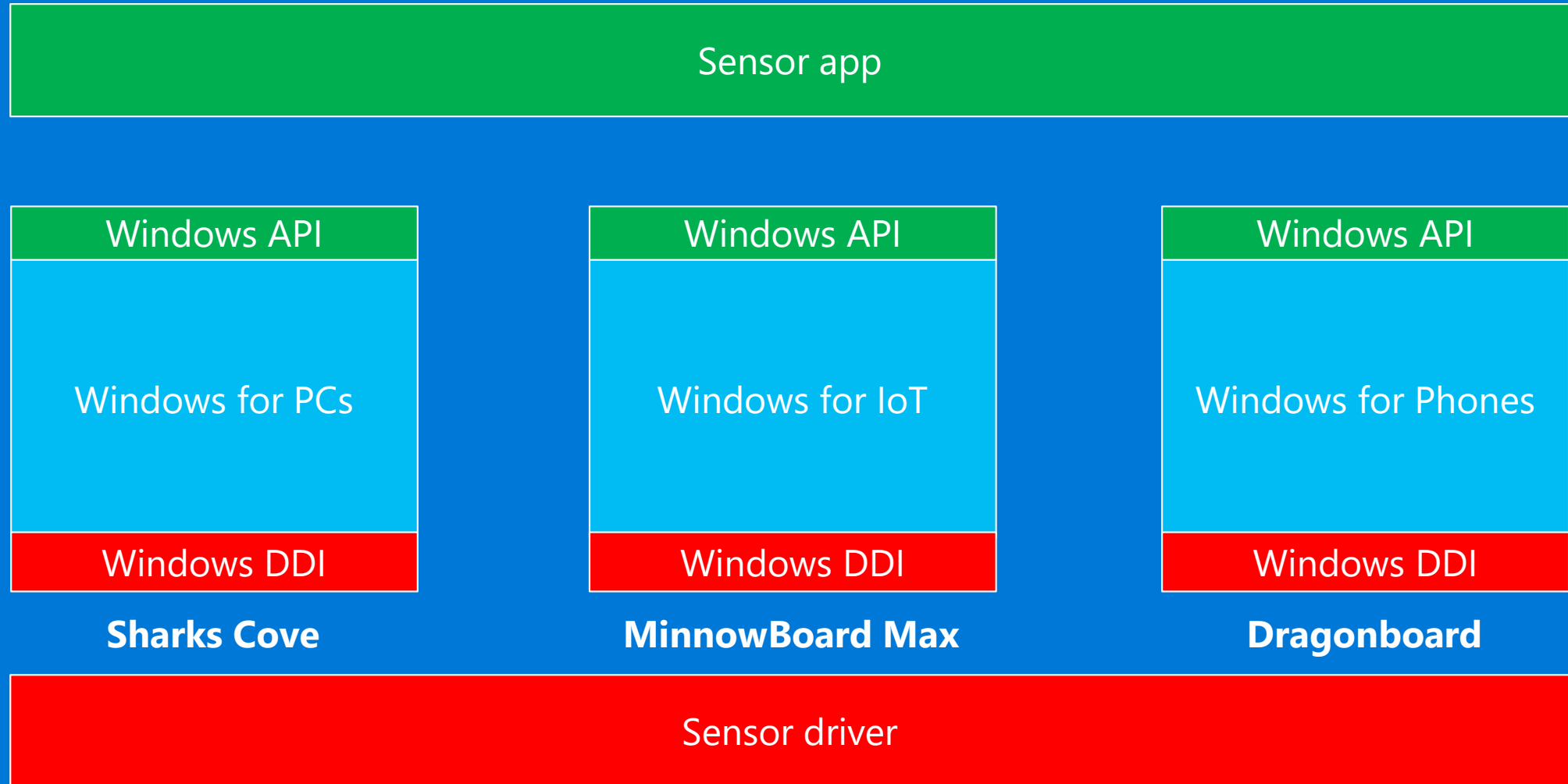
Others

Shell UI

With IoT Core, the target devices are custom devices

- The user experience is defined and implemented by the OEM/Maker as a Universal App

- Windows 10 IoT Core contains the OS infrastructure with which the UWP Application can be launched as the device shell / UI

- Windows 10 IoT Core contains no inbox Windows UI or applications

Implementation Summary

- IoT Core is a single image that can be configured to boot with (headed) or without (headless) display support
  - Configured at image creation time with ICD

- In headed mode Windows video stack & UI components will be launched

- Multiple headed and headless application can be installed on the device.

- The shell infrastructure
  - Will launch all configured Startup Tasks from all installed headed and headless apps
  - Will, in headed mode only, launch the one UWP application configured to provide the OEM user experience

- The Shell will monitor launched apps and tasks and auto restart any that have unexpectedly terminated.

Configuration

- A developer / maker
  - Can deploy applications to the device
  - Can designate a sing UWP app to launch at boot

- An OEM / Maker can configure their image with ICD, and can inject and configure multiple headed and headless apps.

- Multiple headed and headless application can be installed on the device.

- An IT Pro can install applications and configure settings with device management

Image Development

Device Imaging: Full Flash Update

All Windows 10 device families support imaging and manufacturing based on *Full Flash Update* (FFU)

- Existing workflow/tools still supported (e.g., production media, WIM) for PC

FFU image format is sector-based and describes all partitions on disk

FFU images created using Windows Imaging and Configuration Designer (WICD) or command-line tools (imageapp.exe)

FFU images flashed to up to 8 USB-tethered devices using ffutool.exe or directly to disk using WICD or dism.exe

FFU Device Imaging

FFU Device Imaging

Device-Image.ffu

FFUTOOL

WICD or DISM

Flash to up-to 8 tethered Windows devices

...

or

Write directly to disk
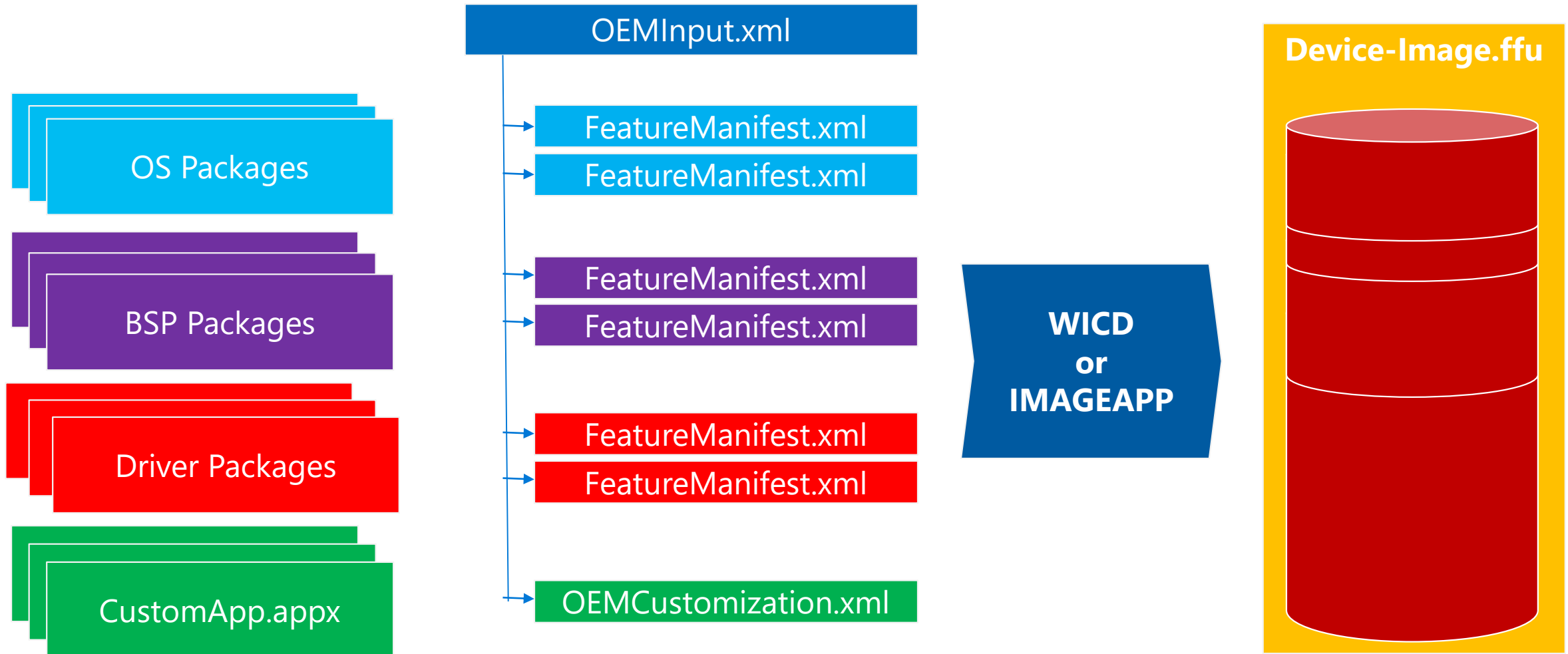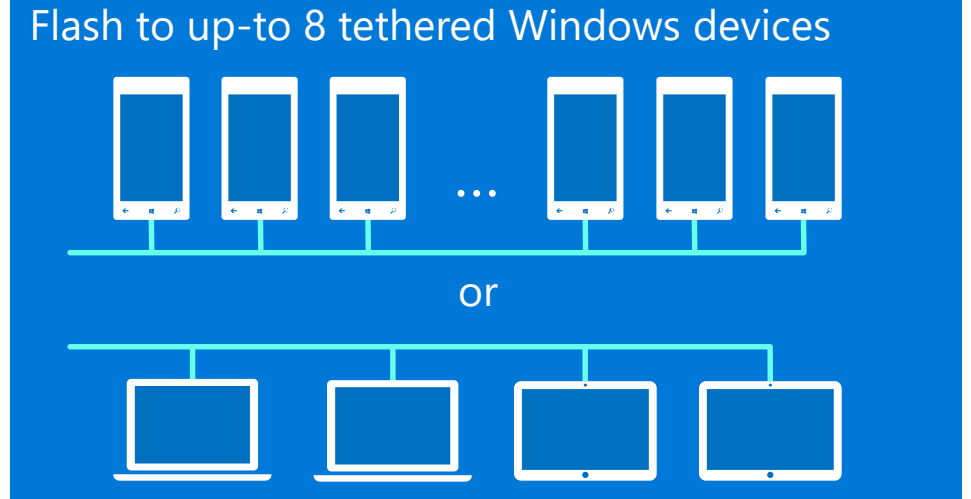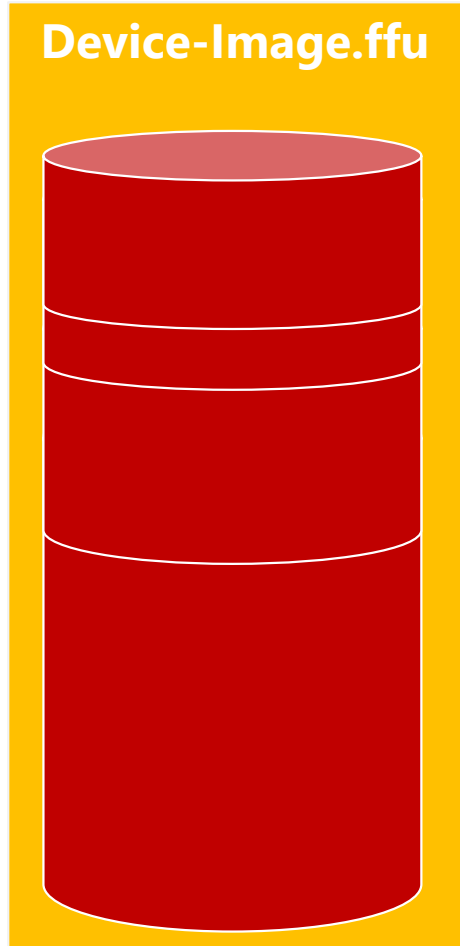
Windows 10 IoT Core Summary

- Based on the Threshold common core with UWP application support
    - Rich OS capabilities to develop Modern, immersive, user experiences
- Supports both ARM and x86
- Single image configurable to run in both a headed and headless mode
    - Configured with a reg key
- Contains No Microsoft branding or Windows UI
    - No Windows shell or apps, No explorer, No IE, No MS in-box Apps
    - Custom branded, end to end device experiences
- No Certification requirements.
- Low hardware requirements

Windows 10 IoT Core Summary

- Shell "hosting" Infrastructure
- Appx Lifecycle management – install, start, stop, recover, uninstall
- Access the functionality of hardware from Modern Applications
- Background app capabilities to host services and servers
  - E.g. monitor sensors, provide remote access, etc
- Servicing model & device mgmt.
- Supporting Makers, hobbyists in addition to traditional Embedded Partners

Windows 10 IoT Core Summary

Productive	Trusted	Connected
Windows Update and Store app servicing Familiar dev tools Windows manageability 64-bit support UWP IoT Remote Client RPi3 & Intel Joule support Computer vision	Secure/Trusted Boot Bitlocker TPM 2.0 Unified Write Filter HORM	Alljoyn/OCF OPC UA for industrial scenarios Bridge and connect other ecosystems (Modbus, BACNet, Zigbee) Ubiquitous connectivity stack Seamless Connectivity to Azure Support for cloud protocols - MQTT AMQP TPM based secure connectivity
Deploy and manage your IoT devices with the commercial ready MDM tools [Additional CSPs, auto provisioning] and with **Azure IoT DM** **Cortana** enables you to bring immersive and interactive user experiences to your devices. Increase flexibility with **new Intel silicon**	Protect your device identities with "**Device guard for IoT**" improving threat resistance by allowing only known/trusted code.	Optimize your cloud resources bringing **edge intelligence** to your IoT solutions [Azure Gateway SDK] OCF merged with AllJoyn brings the broadest ecosystem of IoT interoperability to your Windows devices [Windows 10 UWP compatible with IoTivity 1.2]

RS2

Web Site

https://dev.windows.com/en-us/iot

HOL 1-1 OS Installation

HOL 1-2 Win10IoTCore Tools