**通过Windows Update升级设备固件**

本主题介绍如何使用Windows Update（WU）服务更新可插拔设备或内置设备的固件。有关更新系统固件的信息，请参阅Windows UEFI固件更新平台([Windows UEFI firmware update platform](https://docs.microsoft.com/en-us/windows-hardware/drivers/bringup/windows-uefi-firmware-update-platform)).

要做到这一点，需要开发一个设备驱动程序来支持更新，其中会包含固件。

* 如果您的设备使用第三方提供的驱动程序，则可以选择**将固件更新逻辑和固件添加到现有功能驱动程序**，**或提供单独的固件更新驱动程序**。
* 如果您的设备使用Microsoft提供的驱动程序，则必须提供单独的固件更新驱动程序包。

在这两种情况下，固件更新驱动程序包必须是Universal的。有关Universal驱动程序的详细信息，请参阅Universal Windows驱动程序入门([Getting Started with Universal Windows drivers](https://docs.microsoft.com/en-us/windows-hardware/drivers/develop/getting-started-with-universal-drivers))。驱动程序二进制文件可以是[KMDF](https://docs.microsoft.com/en-us/windows-hardware/drivers/wdf/index), [UMDF 2](https://docs.microsoft.com/en-us/windows-hardware/drivers/wdf/getting-started-with-umdf-version-2) or the [WDM](https://docs.microsoft.com/windows-hardware/drivers/kernel/windows-driver-model)。

由于WU无法执行程序，因此固件更新驱动程序必须将固件交给即插即用（PnP）进行安装。

# 固件更新驱动程序操作

通常，固件更新驱动程序是一个轻量级设备驱动程序，它执行以下操作：

* + 在设备接入时或者在驱动的*[EVT\_WDF\_DRIVER\_DEVICE\_ADD](https://docs.microsoft.com/windows-hardware/drivers/ddi/content/wdfdriver/nc-wdfdriver-evt_wdf_driver_device_add)*回调函数中:
    1. 识别它所连接的设备。
    2. 确定驱动程序的固件版本是否比设备上的版本新。
    3. 如果需要更新固件，设置事件计时器以安排更新。
    4. 否则，在重新启动驱动程序之前不要执行任何操作。
  + 在系统运行时：

1. 如果更新在排队，等待直到满足相关条件。
2. 满足条件时，在设备上执行固件更新。

# 固件更新驱动程序内容

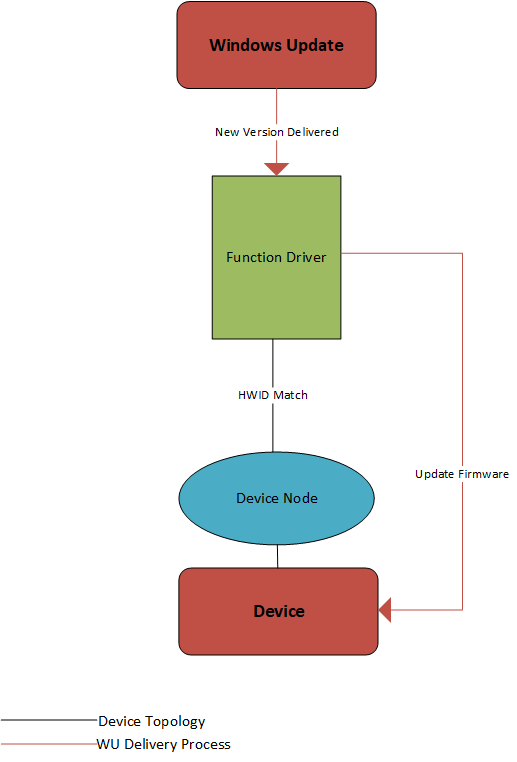
通常，固件更新驱动程序包含以下内容

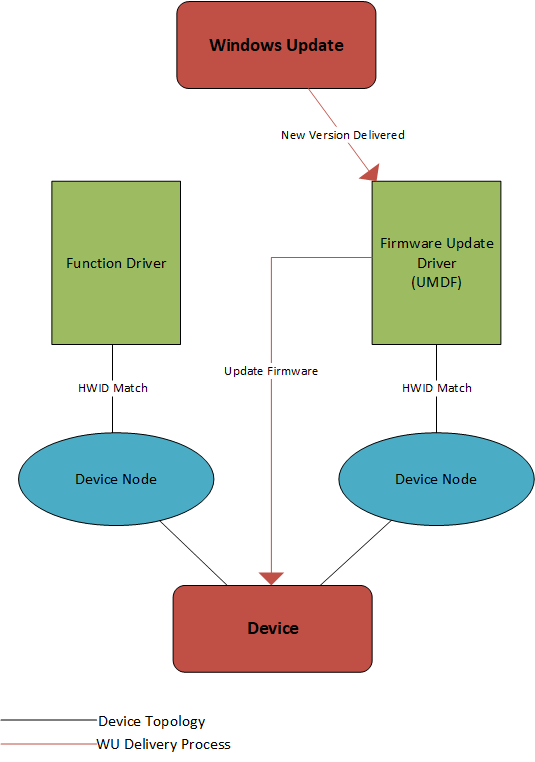
* [Universal Driver INF](https://docs.microsoft.com/en-us/windows-hardware/drivers/install/using-a-universal-inf-file)
* 驱动程序目录(Driver catalog)
* 功能驱动程序（.sys或.dll）
* 固件二进制文件

将固件更新包作为单独的驱动提交。

**将固件更新逻辑添加到第三方(非微软)驱动**

如下图所示, 现有的功能驱动能实现固件升级机制:



或者，如果要单独更新功能驱动程序和固件更新驱动程序，需要创建第二个设备节点，并在其上安装固件更新驱动程序。下图显示了一个设备如何拥有两个独立的设备节点： 

在这种情况下，功能和固件设备节点必须具有不同的硬件ID才能独立定位。

有几种方法可以创建第二个设备节点。某些设备类型能够在一个物理设备上暴露第二个设备节点，例如USB。您可以使用此功能创建Windows Update可定位的设备节点，并在其上安装固件更新驱动程序。

但是，许多设备类型不允许单个物理设备枚举多个设备节点。在这种情况下，使用扩展INF指定[AddComponent](https://docs.microsoft.com/en-us/windows-hardware/drivers/install/inf-addcomponent-directive) 指令以创建可由Windows Update作为目标的设备节点，并在其上安装固件更新驱动程序。 INF文件中的以下代码段显示了如何执行此操作：

[Manufacturer]

%Contoso%=Standard,NTamd64

[Standard.NTamd64]

%DeviceName%=Device\_Install, PCI\DEVICE\_ID

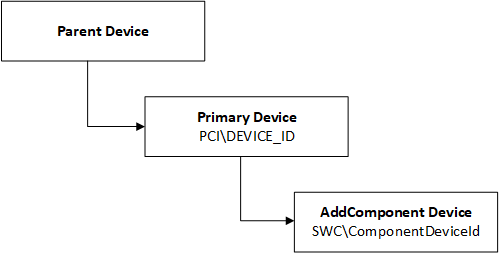
[Device\_Install.Components]

AddComponent=ComponentName,,AddComponentSection

[AddComponentSection]

ComponentIDs = ComponentDeviceId

在上面的INF示例中，ComponentIDs = ComponentDeviceId表示子设备的硬件ID为SWC\ComponentDeviceId。 安装后，此INF将创建以下设备层次结构：



对于后续的固件更新，更新包含固件的INF和二进制文件。

**将固件更新逻辑添加到Microsoft提供的驱动程序**

要更新使用Microsoft提供的驱动程序的设备的固件，您需要创建第二个设备节点，如上所示。

**最佳做法**

* 在固件更新驱动程序INF中，指定 [DIRID 13](https://docs.microsoft.com/en-us/windows-hardware/drivers/install/using-dirids) 以使PnP将文件保留在DriverStore的驱动程序中:

[Firmware\_AddReg]

; Store location of firmware payload

HKR,,FirmwareFilename,,"%13%\firmware\_payload.bin"

PnP在安装设备时解析此位置。然后，驱动程序可以打开此注册表项以确定有效内容的位置。

* 固件更新驱动程序应指定以下INF条目:

Class=Firmware

ClassGuid={f2e7dd72-6468-4e36-b6f1-6488f42c1b52}

* 要定位另一个设备节点，固件驱动程序应该查看和自身相关的设备树，而不是通过枚举所有设备节点进行匹配。用户可能已插入多个设备实例(多个同款设备)，那么固件驱动程序应仅更新与其关联的设备。通常，要定位的设备节点是安装固件驱动程序的设备节点的父节点或兄弟节点。例如，在上图中有两个设备节点的情况下，固件更新驱动程序可以查找兄弟设备以查找功能驱动程序。在上面的图中，固件驱动程序可以查找父设备以查找需要与之通信的主设备。
* 驱动程序应该对系统上的多个设备实例具有鲁棒性，包括可能具有多个不同的固件版本的情况。 例如，可能存在一个已经多次连接和更新的设备实例;也许包括一个全新的但是固件版本版本很旧的设备。这意味着必须针对设备存储状态（例如当前版本），而不是存储在全局位置。
* 如果已经存在更新firmware的方法(比如,exe或者co-installer), 可以最大程度上在UMDF驱动中重用已有的代码.