**# Title:**

Define DevicePath in EFI\_IMAGE\_LOAD as optional

**# Status:**

Draft

**# Document:**

UEFI Specification 2.10

**# License:**

SPDX-License-Identifier: CC-BY-4.0

**# Submitter:**

* Arm (Edhaya Chandran, Jose Marinho, Samer El-Haj-Mahmoud)
* TianoCore Community (<https://www.tianocore.org>)

**# Summary of the change**

Define the DevicePath argument of the EFI\_IMAGE\_LOAD type as optional.

**# Benefits of the change**

Create consistency between the function type declaration and the described behaviour in the UEFI spec.  
The return status EFI\_NOT\_FOUND is returned if “both SourceBuffer and DevicePath are NULL” which implies that DevicePath must be optional.

**# Impact of the change**

* **Platform FW:**
  + Changes required to the declaration of functions with EFI\_IMAGE\_LOAD type

**# Detailed description of the change [normative updates]**

* Insertions highlighted
* Removals in ~~red~~

**7.4 Image Services  
…**

**EFI\_BOOT\_SERVICES.LoadImage()**

**Summary**Loads an EFI image into memory.  
**Prototype**

**typedef  
EFI\_STATUS  
(EFIAPI \*EFI\_IMAGE\_LOAD) (**

|  |  |
| --- | --- |
| **IN BOOLEAN  IN EFI\_HANDLE  IN EFI\_DEVICE\_PATH\_PROTOCOL  IN VOID** | *BootPolicy, ParentImageHandle, \*DevicePath* **OPTIONAL***, \*SourceBuffer* **OPTIONAL,** |
| **IN UINTN  OUT EFI\_HANDLE  );** | *SourceSize, \*ImageHandle* |

**…**

Description

…

The **LoadImage()** function loads an EFI image into memory and returns a handle to the image. The image is loaded in one of two ways.

* If *SourceBuffer* is not **NULL**, the function is a memory-to-memory load in which *SourceBuffer* points to the image to be loaded and *SourceSize* indicates the image’s size in bytes. In this case, the caller has copied the image into *SourceBuffer* and can free the buffer once loading is complete. The DevicePath is optional in this case. A DevicePath should still be provided since certain portions of firmware may use it to make certain security policy decisions.
* If *SourceBuffer* is **NULL**, the function is a file copy operation that uses the EFI\_SIMPLE\_FILE\_SYSTEM\_PROTOCOL.