



# **Intel® Firmware Module Management Tool (Intel® FMMT) User Guide**

Version 1.1

9 March 2016

THIS USER GUIDE IS PROVIDED AS IS WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. Except for a limited copyright license to copy this user guide for internal use only, no license, express or implied, by estoppels or otherwise, to any intellectual property rights is granted herein.

Intel disclaims all liability, including liability for infringement of any proprietary rights, relating to implementation of information in this user guide. Intel does not warrant or represent that such implementation(s) will not infringe such rights.

Designers must not rely on the absence or characteristics of any features or instructions marked reserved or undefined. Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The Intel® Firmware Module Management Tool (Intel® FMMT) may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

This document as well as the software described in it is furnished under license and may only be used or copied in accordance with the terms of the license. The information in this manual is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Intel Corporation. Intel Corporation assumes no responsibility or liability for any errors or inaccuracies that may appear in this document or any software that may be provided in association with this document.

Intel and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

\* Other names and brands may be claimed as the property of others.

Copyright © 2011 -2016 Intel Corporation.

## Revision History

---

Revision	Revision History	Date
1.0	Initial release	Sept. 8, 2015
1.1	Added multiple files to Add, Delete & replace.	March 9, 2016

# Table of Contents

	Revision History .....	iii
	Table of Contents .....	iv
1	Introduction .....	1
	1.1 System requirements .....	1
	1.2 Terminology and conventions .....	1
	1.3 For more information .....	2
2	Setting up the Intel® FMMT .....	3
	2.1 Special considerations .....	3
	2.2 Set up the development environment .....	4
	2.2.1 Method A: If you already have a base tools project .....	4
	2.2.2 Method B: If you do not already have a base tools project .....	4
3	Using the Intel® FMMT .....	5
	3.1 Supported Intel® FMMT command directives .....	5
	3.2 Filenames and path names .....	5
	3.3 Using Intel® FMMT command directives .....	6
	3.3.1 View operation .....	7
	3.3.2 Add operation .....	7
	3.3.3 Delete operation .....	8
	3.3.4 Replace operation .....	9
	3.4 Add a UI section for an FFS file .....	9
	3.5 For more information .....	10

# 1 Introduction

---

Welcome to the user guide for the Intel® Firmware Module Management Tool (Intel® FMMT). The Intel FMMT tool enables the viewing, removal, addition, and replacement of FFS (firmware file system) files in previously built EDK-II (EFI Development Kit) firmware device (.fd) image binaries.

This introductory section includes this information:

- **System requirements** for installing the Intel FMMT
- **Special considerations** for installing and using the Intel FMMT
- **Terminology** and conventions used in this user guide
- **For more information** -- links to specifications, developer's kits, and other documentation that might be of use

## 1.1 System requirements

Before installing or using the Intel FMMT tool, make sure your system meets the following requirements. Make sure you also keep the special considerations (described in the next discussion) in mind when installing or using Intel FMMT.

To install the Intel FMMT tool, you will need a desktop or laptop PC with:

- Processor: 1 GHz Intel® Pentium processor or better
- RAM: More than 256 MB
- Hard disk: More than 100 MB of available space

Intel FMMT supports these operating systems:

- Windows XP\* for IA32-based architectures -- *recommended*
- Windows 7\* for X64-based architectures

## 1.2 Terminology and conventions

This quick start guide uses these terms and conventions:

C:\Workspace

In the example command lines, C:\Workspace is used to indicate the directory in which the platform build is launched. When entering commands on your system, make sure to replace C:\Workspace with your own directory name.

SNBCLIENT.fd	In the example command lines, SNBCLIENT.fd is used to indicate the build image. When building your own .fd image, make sure to replace SNBCLIENT.fd with your specific build name.
--------------	--

This document uses these acronyms:

EFI	Generic term that refers to one of the versions of the Extensible Firmware Interface (EFI) specification
FD	Firmware device
FDF	Flash device format
FFS	Firmware file system
FMMT	Intel® Firmware Module Management Tool
FV	Firmware volume
GUID	Globally unique identifier
UEFI	Unified Extensible Firmware Interface

## 1.3 For more information

The Intel FMMT is part of EDK II, and located in the base tools directory. The EDK II source files, specifications, and related specifications: [www.tianocore.org](http://www.tianocore.org)

UEFI specification, including the most recent UEFI platform initialization specifications: <http://www.uefi.org>.

## Setting up the Intel® FMMT

---

This section explains how to set up the development environment for using the Intel® FMMT tool.

### 2.1 Special considerations

Keep these considerations in mind when installing and using the Intel FMMT:

- **FFS files:**
  - **Conform to PI 1.2.** The FFS file should be based on the platform initialization specification PI 1.2 (available from [www.uefi.org](http://www.uefi.org))
  - **Include UI section.** The FFS file must have a user interface (UI) section.
  - **Generating the FFS file.** The FFS file should *not* be generated from execute-in-place modules.
  - **Minimum requirement of one FFS file.** If the firmware volume in the firmware device file contains only one FFS file, you cannot delete that file. There must be at least one FFS file in the firmware device file.
- **Standard GUIDed tools.** The Intel FMMT supports these standard GUIDed tools:
  - a31280ad-481e-41b6-95e8-127f4c984779 TIANO TianoCompress
  - ee4e5898-3914-4259-9d6e-dc7bd79403cf LZMA LzmaCompress
  - fc1bcd0-7d31-49aa-936a-a4600d9dd083 CRC32 GenCrc32
- **Custom GUIDed tools.** Some FFS files require a custom, GUIDed tool for EFI\_SECTION\_GUID\_DEFINED sections. Intel FMMT does not currently support automatically calling a custom GUIDed tool for these EFI\_SECTION\_GUID\_DEFINED sections. However, you can update the FmmtConf.ini file to add the GUID-NAME mapping relationship. Updating the initialization file allows Intel FMMT to process the custom EFI\_SECTION\_GUID\_DEFINED section.

## 2.2 Set up the development environment

Intel FMMT uses the BaseTools project binaries GenSec/GenFfs/GenFv and LzmaCompress/TianoCompress/GenCrc32. There are two ways to set up the environment.

- If you already have BaseTools project, use method A.
- If you do not already have a BaseTools project, use method B.

### 2.2.1 Method A:

#### If you already have a base tools project

If you already have at least one base tools project, you must set the EDK tools path for Intel FMMT to an appropriate path. Follow these steps:

1. Unzip the Binaries.zip file.
2. Copy the FMMT.exe and FmmtConf.ini file to the directory %WORKSPACE%/BaseTools/Bin/Win32/

**Note:** If your SandyBridge code tree is C:\CodeTree\SandyBridge, then %WORKSPACE% = C:\CodeTree\SandyBridge

3. Open a command window and change to the directory: %WORKSPACE%/BaseTools
4. Set EDK\_TOOLS\_PATH=%WORKSPACE%/BaseTools
5. Set BASE\_TOOLS\_PATH=%WORKSPACE%/BaseTools
6. Run toolsetup.bat

### 2.2.2 Method B: If you do not already have a base tools project

If you do not already have a base tools project on your system, you can simply unzip the files and run Intel FMMT. You will not need to override any existing files. Follow these steps:

1. Verify that the configuration file FmmtConf.ini is located in your working directory.

**Caution:** Before running Intel FMMT, make sure the FmmtConf.ini file is located under your working directory. If there is no configure file in the working directory, Intel FMMT will use a hard-coded GUIDed tool to operate on any EFI\_SECTION\_GUID\_DEFINED section in any FFS file.

2. Unzip the Binaries.zip file.
3. Open the unzipped directory (to make it the active directory).
4. From within the unzipped directory, open a command window and run Intel FMMT.



## Using the Intel® FMMT

The Intel FMMT tool enables removal, addition, and replacement of FFS files in previously built EDK-II (EFI Development Kit) firmware device (.fd) image binaries.

**Caution:** *Make sure you keep the special considerations in mind (for FFS files and GUIDed tools) when installing and using the Intel FMMT tool. Special considerations are listed in the setup section of this document.*

### 3.1 Supported Intel® FMMT command directives

Intel FMMT supports the following command directives for view, add, delete, and replace operations:

- **View:** Lets you view the contents of each FV and the named files within each FV. The View command directive has this format:  
`FMMT -v <input-binary-file>`
- **Add:** Adds (inserts) a file (or files) to the firmware volume in a firmware device binary file. The Add command directive has this format:  
`FMMT -a <input-binary-file> <FV-id> <NewFilePath> [<FV-id> <NewFilePath> ...]  
 <output-binary-file>`
- **Delete:** Deletes a file (or files) from the firmware volume in a firmware device binary file. The Delete command directive has this format:  
`FMMT -d <input-binary-file> <FV-id> <File-Name> [<FV-id> <File-Name> ...]  
 <output-binary-file>`
- **Delete:** Deletes the entire FV in a firmware device binary file. The Delete command directive has this format:  
`FMMT -d <input-binary-file> <FV-id> <output-binary-file>`
- **Replace:** Deletes a file (or files) from the FV in a firmware device binary file, then inserts the new file (or files) into the FV in the firmware device binary file. The Replace command directive has this format:  
`FMMT -r <input-binary-file> <FV-id> <File-Name> <NewFilePath> [<FV-id> <File-Name> <NewFilePath> ...] <output-binary-file>`

### 3.2 Filenames and path names

The Intel FMMT uses the following file and path names:

**<Input-binary-file>** This is full FD (BIOS) image, which is created by a successful EDK-II build. Input binary files are *not* FFS files. It must be an EDK-II FD image. Firmware devices from other environments (for example, FDs from EDK-I and Aptio\*) will be rejected as not being in the required format.

**<FV-id>** The text name associated with a firmware volume (FV). Based on the sequence of the firmware volume included in the FD image, the FMMT tool names each firmware volume sequentially as FV0, FV1, FV2, and so on. It also support to use FV's file guid value. If you do not know the firmware volume's name <FV-id> or file guid value, use the FMMT -v sample.fd command to check the output.

**<File-Name>** The contents of the human-readable section in an FFS file in a firmware volume.

**<NewFilePath>** The path to an FFS file.

**<output-binary-file>** This is a full FD image created by Intel FMMT from the input FD image. The output image includes the changes you made via the Intel FMMT. (This is not an FFS file.)

**Note:** Make sure you are using the current version of EDK-II. This ensures that the output FD images will be conformant with PI version 1.2.

### 3.3 Using Intel® FMMT command directives

This discussion shows how to enter commands for the view, add, delete, and replace operations. This discussion also shows sample output from the commands.

In these examples, the workspace is the top-level directory called EDKII. Your command line prompt might be different, depending on your workspace.

**Note:** In the examples, *SNBCLIENT.fd* is used to indicate the build image, and *SNBTest.fd* and *SNBTestDel.fd* are used to indicate the input file and deleted file, respectively. When using the commands on your system, make sure to replace the example filenames with the appropriate ones for your application.

### 3.3.1 View operation

The view command directive lets you view the contents of each FV and the named files within each FV. To use the View operation, open a command window and enter the following command. Intel FMMT displays the status of the View operation as it progresses.

```
c:\EDKII> Fmmt -v NT32.fd

FV0 :

Attributes:          CFEFF
Total Volume Size:   0x00280000
Free Volume Size:    0x00006E00
FvNameGuid:          6D99E806-3D38-42C2-A095-5F4300BFD7DC

File "PeiCore"
File "PcdPeim"
File "ReportStatusCodeRouterPei"
...
```

### 3.3.2 Add operation

The Add operation adds (inserts) some file to the firmware volume in a firmware device binary file. To use the Add operation, enter the following command. In this example, the driver DriverHealthDxe.fff is being added to firmware volume 2 (FV2).

As shown below, Intel FMMT displays the status of the Add operation as it progresses. In this example, the last status line shows the newly added module, DriverHealthDxe, has been added to the FV.

```
c:\EDKII> Fmmt -a SNBCLIENT.fd FV2 DriverHealthDxe.ffs SNBTest.fd
```

```
c:\EDKII> Fmmt -v SNBTest.fd
```

FV1 :

```
Attributes:          4FEFF
Total Volume Size:   0x00670000
Free Volume Size:    0x00000E00
FvNameGuid:          1D99E806-3D38-42C2-A095-5F4300BFD7DC
```

Decoding

Child FV named FV2 of FV1

```
Attributes:          7FEFF
Total Volume Size:   0x00560000
Free Volume Size:    0x00001D00
FvNameGuid:          3D99E801-3D38-42C2-0951-F4300BFD7DC1
```

```
File "DxeCore"
...
File "EC"
File "SmmThunk"
File "DriverHealthDxe"
```

The add file operation supports to use the FvNameGuid value to specify the FV.

```
c:\EDKII> Fmmt -a SNBCLIENT.fd 3D99E801-3D38-42C2-0951-F4300BFD7DC1
DriverHealthDxe.ffs SNBTest.fd
```

The add file operation also supports adding multiple files in a single command. In the following example, we add DriverHealthDxe.ffs to the FV2, and add TestDriverDxe.ffs to the FV3.

```
c:\EDKII> Fmmt -a SNBCLIENT.fd FV2 DriverHealthDxe.ffs FV3 TestDriverDxe.ffs
SNBTest.fd
```

### 3.3.3 Delete operation

The delete operation deletes some file from the FV in a firmware device binary file or deletes entire FV in a firmware device binary file. To use the Delete operation, enter the following command. In this example, SNBTest.fd is the example input file.

**Caution:** *Delete is a nonrecoverable command directive. There is no confirmation before delete executes. There is also no undo; the deleted file is irretrievable gone. You should always keep a copy of your files until you are certain you no longer need them.*

```
c:\EDKII> Fmmt -d SNBTest.fd FV2 DriverHealthDxe SNBTestDel.fd
```

After the operation has completed, use the Fmmt command with the -v switch to check the SNBTestDel.fd file. The view command directive will display a list of drivers, without the DriverHealthDxe driver.

```
c:\EDKII> Fmmt -v SNBTestDel.fd
```

The delete file operation supports to use the FvNameGuid value to specify the FV.

The delete file operation also supports deleting multiple files in a single command. In the following example, we delete DriverHealthDxe from FV2 and delete TestDriverDxe from FV3.

```
c:\EDKII> Fmmt -d SNBTest.fd FV2 DriverHealthDxe FV3 TestDriverDxe
SNBTestDel.fd
```

### 3.3.4 Replace operation

The Replace operation deletes some file from a FV in a firmware device binary file, then inserts some new file into the FV in the firmware device binary file.

To use the Replace operation, enter the following command. Intel FMMT displays the status of the replace operation as it progresses.

```
c:\EDKII> Fmmt -r SNBCLIENT.fd FV2 HelloWorld DriverHealthDxe.ffs
SNBTestReplace.fd
```

```
c:\EDKII> Fmmt -v SNBTestReplace.fd
```

FV1 :

Attributes:	4FEFF
Total Volume Size:	0x00670000
Free Volume Size:	0x00000E00
FvNameGuid:	1D99E806-3F17-42C2-A095-5F4300BFD7DC

Decoding

Child FV named FV2 of FV1

Attributes:	7FEFF
Total Volume Size:	0x00560000
Free Volume Size:	0x00000E00
FvNameGuid:	1D99E806-3F17-42C2-A095-5F4300BFD7DC

File "DxeCore"

...

File "EC"

File "DriverHealthDxe" <- Original is SmmThunk module.

The replace file operation supports to use the FvNameGuid value to specify the FV.

The replace file operation also supports replacing multiple files in a single command. In the following example, we replace HelloWorld with DriverHealthDxe in the FV2, and replace TestDriverDxe with DriverSample in the FV3.

```
c:\EDKII> Fmmt -r SNBCLIENT.fd FV2 HelloWorld DriverHealthDxe.ffs FV3
TestDriverDxe DriverSample.ffs SNBTestReplace.fd
```

## 3.4 Add a UI section for an FFS file

If you want to add a user interface section to an FFS file, you must modify the flash device format (FDF) file, which is a firmware device map that specifies where various elements are located. The specific modification depends on whether the platform is a desktop PC or laptop PC. (The examples shown here use the Sandy Bridge platform.)

If your video option ROM is consumed by a desktop PC:

```
FILE FREEFORM = E880F38E-EC5E-4952-B17D-0CA527B2A8AC {
    SECTION RAW = ..\Board\NB\LegacyVideoRom\LegacyVideoRomDT.bin
    SECTION UI = "LegacyVideoRomDT" <- New added line
}
```

If your video option ROM is consumed by a laptop PC:

```
FILE FREEFORM = 6A5F1EF9-C478-42ba-A4DB-C7846CC7DB57 {
    SECTION RAW = R8SnbClientAptioPkg\Board\NB\LegacyVideoRom\LegacyVideoRomMB.bin
    SECTION UI = "LegacyVideoRomMB" <- New added line
}
```

## 3.5 For more information

The Intel FMMT tool is part of EDK II, and located in the base tools directory. You can download the EDK II source files, specifications, and related specifications from [www.tianocore.org](http://www.tianocore.org)

The UEFI specification, including the most recent UEFI platform initialization specifications can be located at: <http://www.uefi.org>.