tianocore

UEFI & EDK II TRAINING

EDK II BUILD SPECIFICATION FILES

tianocore.org

tianocore

LESSON OBJECTIVE

- Explain the Build components and build text files DSC, DEC, & FDF

EDK II BUILD TEXT FILES

EDK II File Extensions

- Located on tianocore.org project edk2

.DSC file - Platform Description

.DEC file - Package Declaration

.INF file - Module Definition (define a component)

.FDF file - Flash Description File

EDK II Spec

.VFR file - Visual Forms Representation for User interface

.UNI file - String text file for ease of localization

.c & . h files - Source code files

Source

.DXS file - Dependency expression file – now [DEPEX]

.FV file - Firmware Volume image file

Output

BUILD DESCRIPTION FILE TYPES

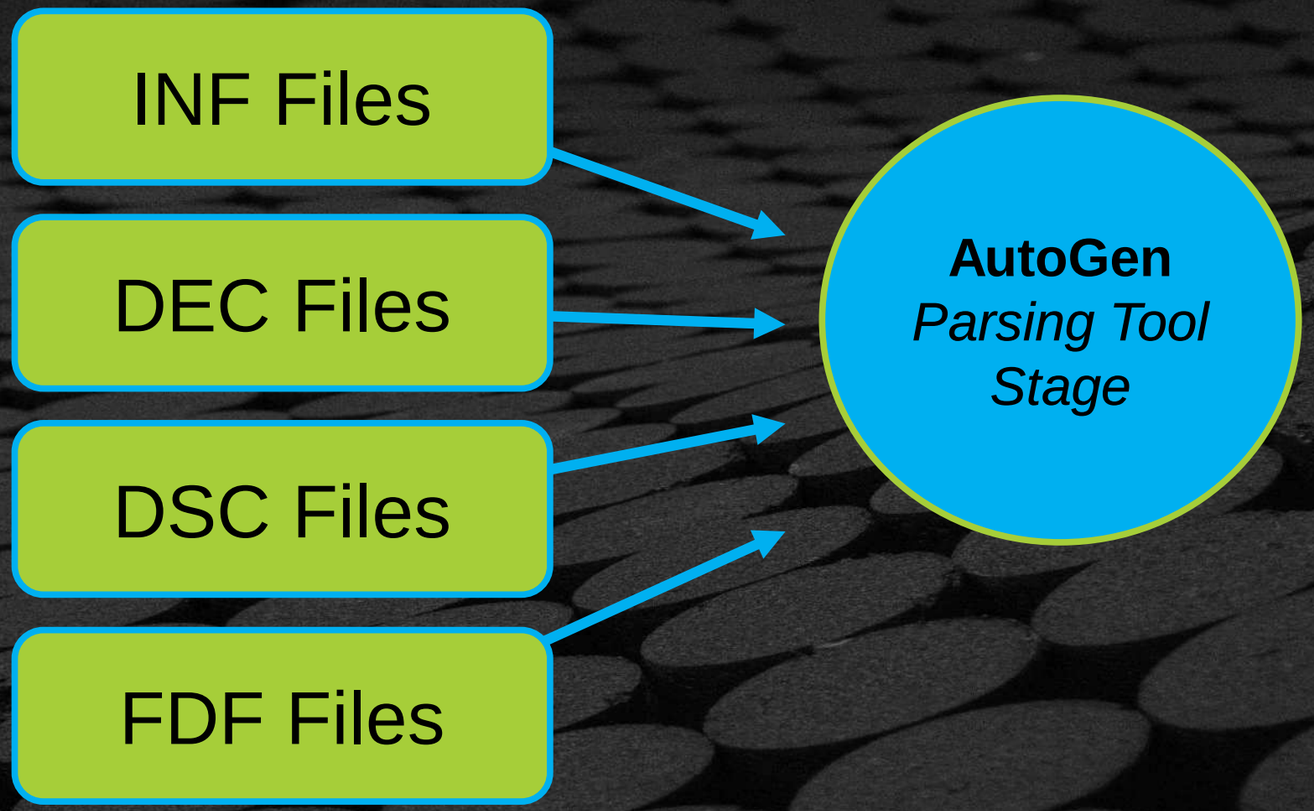
EDK II Spec

- INF Files
- DEC Files
- DSC Files
- FDF Files

Wiki Link: Build Description Files
Edk II Specifications

BUILD DESCRIPTION FILE TYPES

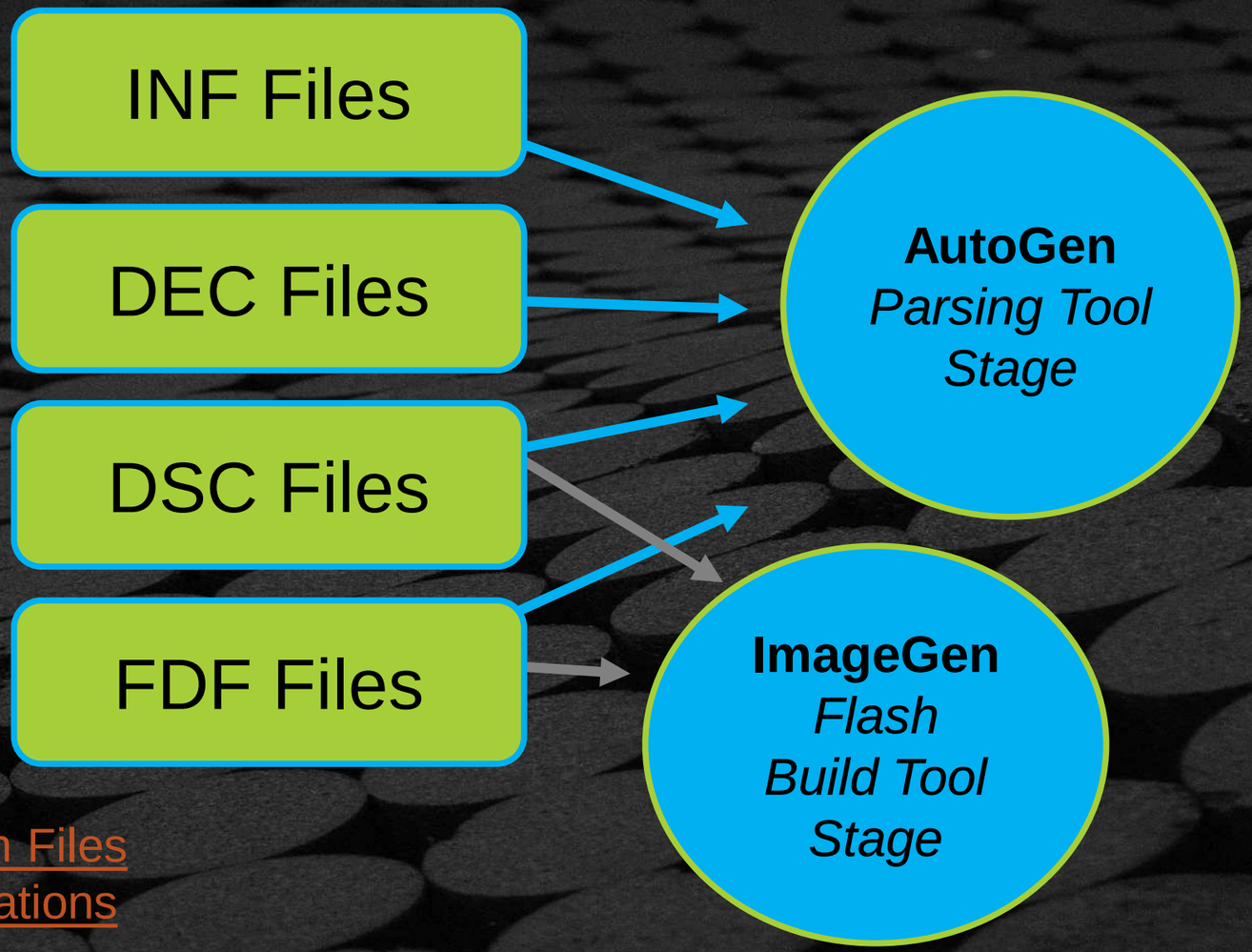
EDK II Spec



Wiki Link: Build Description Files
Edk II Specifications

BUILD DESCRIPTION FILE TYPES

EDK II Spec



Wiki Link: Build Description Files
Edk II Specifications

PACKAGE DECLARATION FILE (DEC)

Declare

Syntax:

```
<DECfile> ::=   <Defines>
                Include
                [<LibraryClass>]
                [<Guids>]
                [<Protocols>]
                [<Ppis>]
                [<Pcd>]
                [<UserExtensions>]
```

EXAMPLE DEC FILE

```
[Defines]
  DEC_SPECIFICATION              = 0x00010005
  PACKAGE_NAME                   = OvmfPkg
  PACKAGE_GUID                   = 2daf5f34-50e5-4b9d-b8e3-5562334d87e5
  PACKAGE_VERSION                = 0.1

[Includes]
  Include

[LibraryClasses]
  ##  @libraryclass  Loads and boots a Linux kernel image
  #
  LoadLinuxLib|Include/Library/LoadLinuxLib.h

[Guids]
  gUefiOvmfPkgTokenSpaceGuid          = {0x93bb96af, 0xb9f2, 0x4eb8, {0x94, 0x62, 0xe0, 0xba, 0x74, 0x56, 0x42, 0x36}}
  gEfiXenInfoGuid                     = {0xd3b46f3b, 0xd441, 0x1244, {0x9a, 0x12, 0x0, 0x12, 0x27, 0x3f, 0xc1, 0x4d}}

[Protocols]
  gVirtioDeviceProtocolGuid           = {0xfa920010, 0x6785, 0x4941, {0xb6, 0xec, 0x49, 0x8c, 0x57, 0x9f, 0x16, 0x0a}}
  gXenBusProtocolGuid                 = {0x3d3ca290, 0xb9a5, 0x11e3, {0xb7, 0x5d, 0xb8, 0xac, 0x6f, 0x7d, 0x65, 0xe6}}

[PcdsFixedAtBuild]
  gUefiOvmfPkgTokenSpaceGuid.PcdOvmfPeiMemFvBase|0x0|UINT32|0
  gUefiOvmfPkgTokenSpaceGuid.PcdOvmfPeiMemFvSize|0x0|UINT32|1
```

Link to Gitpitch

EXAMPLE: DEC FILE DETAILS

https://gitpitch.com/Laurie0131/EDK_II_Build_Spec_Files_Pres/master#/6

PLATFORM DESCRIPTION FILE (DSC)

Description

Syntax:

```
DSCfile ::= [<Header>]
            <Defines>
            [<SkuIds>]
            [<Libraries>]
            [<LibraryClasses>]
            [<Pcds>]
            [<Components>]
            [<UserExtensions>]
```

PLATFORM DESCRIPTION FILE (DSC)

DSC file is the recipe for creating a package

Definitions for the package build

EDK libraries (for EDK Components)

EDK II Library Class Instance Mappings (for EDK II Modules)

EDK II PCD Entry Settings

EDK Component or EDK II Module INF Files

EXAMPLE: DSC FILE

```
[Defines]
  PLATFORM_NAME                  = Ovmf
  PLATFORM_GUID                  = 5a9e7754-d81b-49ea-85ad-69eaa7b1539b
  PLATFORM_VERSION               = 0.1
  DSC_SPECIFICATION              = 0x00010005
  OUTPUT_DIRECTORY               = Build/OvmfX64
  SUPPORTED_ARCHITECTURES        = X64
  BUILD_TARGETS                  = NOOPT|DEBUG|RELEASE
  SKUID_IDENTIFIER               = DEFAULT
  FLASH_DEFINITION               = OvmfPkg/OvmfPkgX64.fdf

  #
  # Defines for default states.  These can be changed on the command line.
  # -D FLAG=VALUE
. . .
[BuildOptions.common.EDKII.DXE_RUNTIME_DRIVER]
  GCC:*_*_*_DLINK_FLAGS = -z common-page-size=0x1000
  XCODE:*_*_*_DLINK_FLAGS =
[LibraryClasses]
  PcdLib|MdePkg/Library/BasePcdLibNull/BasePcdLibNull.inf
  TimerLib|OvmfPkg/Library/AcpiTimerLib/BaseAcpiTimerLib.inf
```

Link to Gitpitch

EXAMPLE: DSC FILE DETAILS

https://gitpitch.com/Laurie0131/EDK_II_Build_Spec_Files_Pres/master#/9

FLASH DESCRIPTION FILE(FDF)

Flash Layout

Syntax:

```
FDFfile ::= [<Header>]
    [<Defines>]
    <FD>
    <FV>
    [<Capsule>]
    [<VTF>]
    [<Rules>]
    [<OptionRom>]
    [<UserExtensions>]
```

FLASH DESCRIPTION FILE(FDF)

Describes information about flash parts

Used to create firmware images, Option ROM images or bootable images

Rules for combining binaries (Firmware Image) built from a DSC file

FLASH DEVICE CONFIGURATION COMMON LAYOUT FILE (.FDF)

FV Recovery

Used to store SEC/PEI phase code

FLASH DEVICE CONFIGURATION COMMON LAYOUT FILE (.FDF)

FV Recovery

FTW spare space

FTW working space

Used to store SEC/PEI phase code

Fault Tolerant Write (FTW) regions

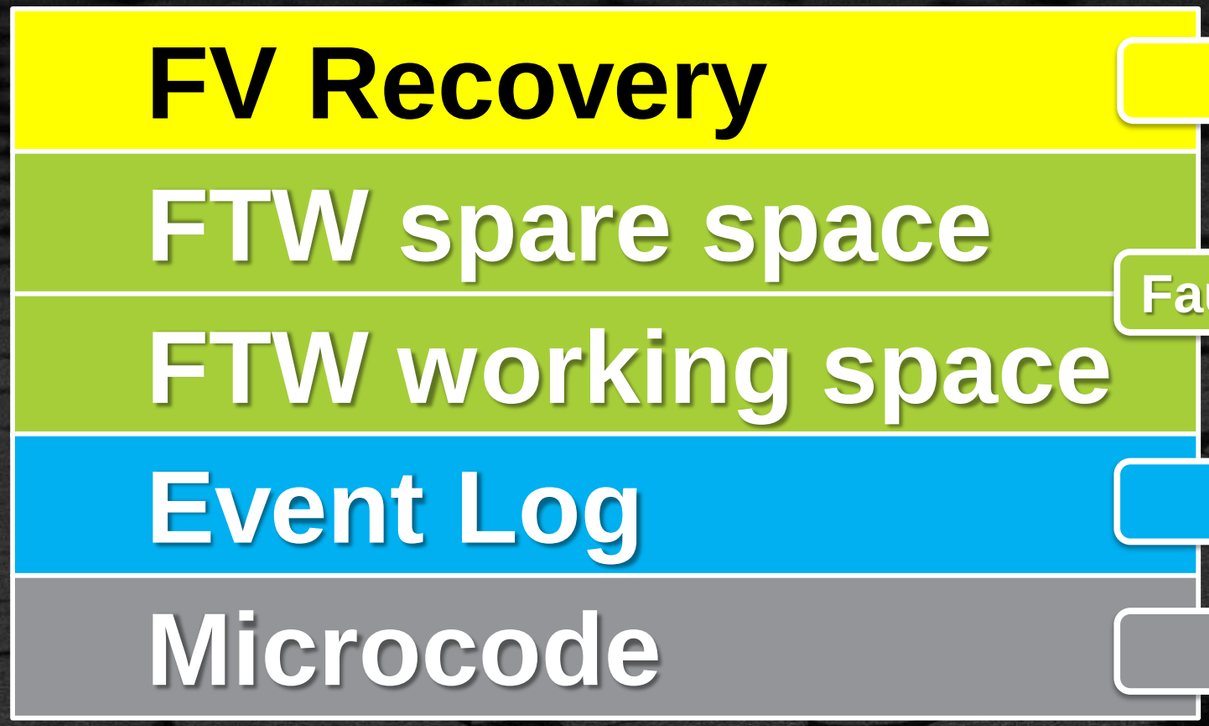
FLASH DEVICE CONFIGURATION COMMON LAYOUT FILE (.FDF)

Used to store SEC/PEI phase code

Fault Tolerant Write (FTW) regions

NVRAM storage for event logs

FLASH DEVICE CONFIGURATION COMMON LAYOUT FILE (.FDF)



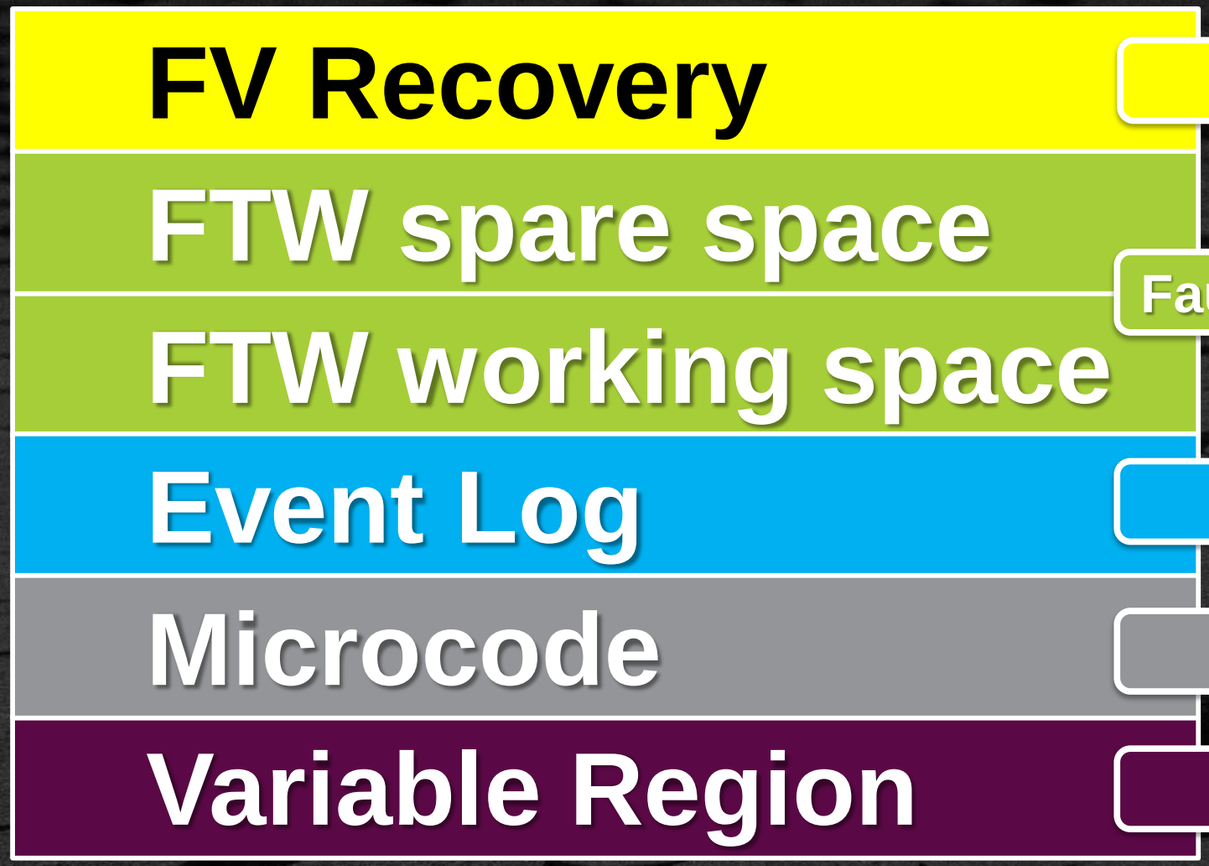
Used to store SEC/PEI phase code

Fault Tolerant Write (FTW) regions

NVRAM storage for event logs

CPU Microcode

FLASH DEVICE CONFIGURATION COMMON LAYOUT FILE (.FDF)



Used to store SEC/PEI phase code

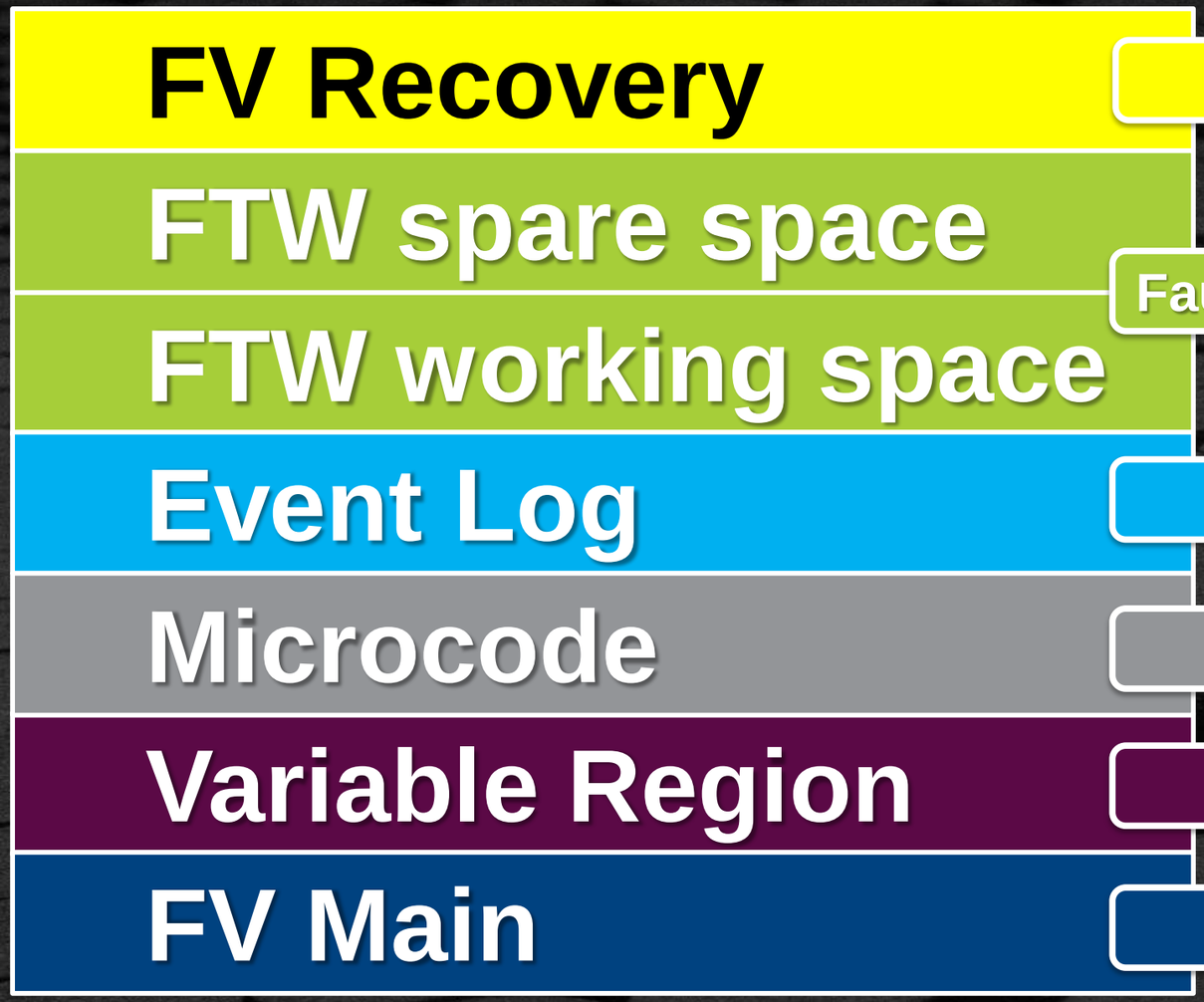
Fault Tolerant Write (FTW) regions

NVRAM storage for event logs

CPU Microcode

Variables & platform settings

FLASH DEVICE CONFIGURATION COMMON LAYOUT FILE (.FDF)



Used to store SEC/PEI phase code

Fault Tolerant Write (FTW) regions

NVRAM storage for event logs

CPU Microcode

Variables & platform settings

Contains DXE phase drivers

EXAMPLE: FDF FILE

```
[Defines]
!include OvmfPkg.fdf.inc

#
# Build the variable store and the firmware code as one unified flash device
# image.
#
[FD.OVMF]
BaseAddress   = $(FW_BASE_ADDRESS)
Size          = $(FW_SIZE)
ErasePolarity = 1
BlockSize     = $(BLOCK_SIZE)
NumBlocks     = $(FW_BLOCKS)


$(VARS_SIZE)|$(FVMAIN_SIZE)
FV = FVMAIN_COMPACT

$(SECFV_OFFSET)|$(SECFV_SIZE)
FV = SECFV
```

Link to Gitpitch

EXAMPLE: DEC FILE DETAILS

https://gitpitch.com/Laurie0131/EDK_II_Build_Spec_Files_Pres/master#/13

SUMMARY

- Explain the Build components and build text files DSC, DEC, & FDF

tianocore

Questions?


tianocore