



# American Megatrends

## Shell & Application Introduce

AMI-China

Get your computer addicted to AMIBIOS

Confidential - NDA Required

Slide 1

1.800.UBUYAMI



American  
Megatrends

# Agenda

---

- ❧ EFI Shell Overview
- ❧ EFI Shell Environment
- ❧ EFI Shell Command
- ❧ EFI Shell Usage
- ❧ EFI Shell Demo
- ❧ Develop Application tools
- ❧ Run Application

# EFI Shell Overview

---

- ❧ Interactive Console Interface
- ❧ Application Launch
- ❧ Load EFI Drivers
- ❧ Scripting Capability (\*.nsh)
- ❧ Automatic execution of startup.nsh batch file
- ❧ Console redirection to files
- ❧ An EFI Application

# EFI Shell Environment

---

- ❧ 100% C Code
- ❧ Easy porting to other platform
- ❧ Command Line Environment
- ❧ Usefully internal commands
- ❧ Open Source
  - ❧ Please visit <https://efi-shell.tianocore.org/>

# EFI Shell Command

---

- ❧ Edit
- ❧ Pci
- ❧ Mem
- ❧ Mm
- ❧ Load, Unload
- ❧ Devices
- ❧ Drivers
- ❧ Devtree
- ❧ Connect, Disconnect, etc.

# EFI Shell Usage

---

- ❧ 执行 preboot 的程序
  - ❧ Diagnostics
  - ❧ Flash update
  - ❧ Operation system install ...etc
- ❧ 提供 Disk Operation System 的能力
  - ❧ Copy or move files between FDD, HDD, CDROM and so on
- ❧ 加载 preboot EFI driver
  - ❧ Tcpiip
  - ❧ USBKB ... etc

# EFI Shell Demo

---

➤ (Demo EFI Shell.....)

➤ Commands

➤ Pci

➤ Drivers

➤ Devices

➤ Map

➤ Devtree

➤ Dh

➤ Memmap

➤ Mm

# EDK - Develop Application tools

---

- ✧ The EFI Application Toolkit contains source code and documentation that enables rapid development of EFI-based applications, protocols, device drivers, EFI Shells, and OS loaders.
- ✧ Visit the following link to get the tool
  - ✧ <https://edk.tianocore.org/>



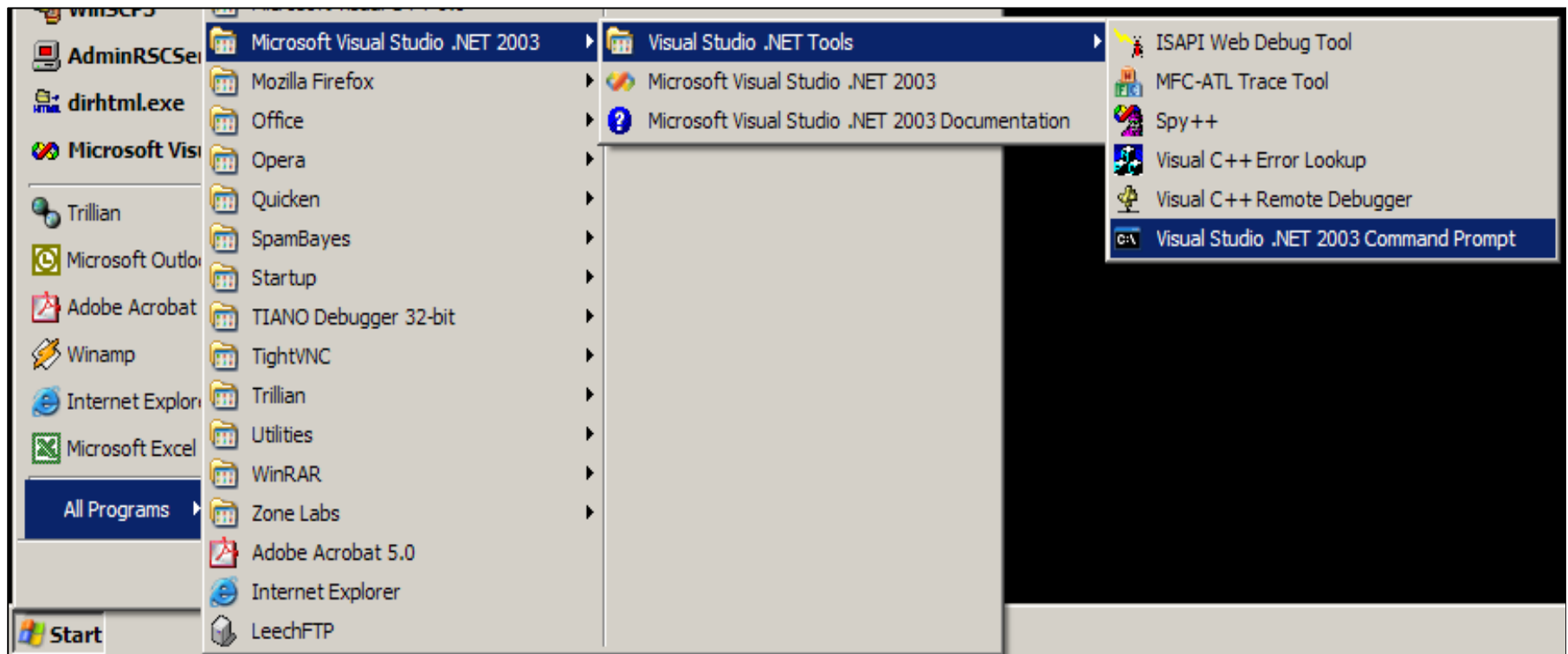
# EDK - Directory Structure

---

<b>Apps</b>	EFI sample applications.
<b>Binaries</b>	Binaries built from the source
<b>Build</b>	Build environment directories
<b>cmds</b>	Ports of FreeBSD commands and utilities
<b>Doc</b>	Documentation for the EFI Application Toolkit
<b>Include</b>	Common include files
<b>Lib</b>	Common libraries
<b>Protocols</b>	Toolkit-supplied EFI protocols and drivers

# EDK - Compiler Environment

➤ Use the Visual Studio command prompt to setup the proper compiler environment



# EDK - Setup

---

## 🌀 Local System Setup

- 🌀 Install Microsoft Visual Studio .NET. (also done for Building the Framework)
- 🌀 From Command prompt type VSVARS32
- 🌀 Set EDK\_SOURCE=C:\FW\Edk
- 🌀 The EFI Toolkit will use the Command batch file BUILD.CMD

# EDK - Install Framework NT32

---

## ❧ System requirements:

- ❧ Microsoft Windows 2000/XP/2003
- ❧ 256MB+ System Memory
- ❧ 500MB+ Free Space on Hard Drive
- ❧ Visual Studio .NET 2003 Professional
- ❧ MASM 6.15

## ❧ Build the NT32 Framework environment

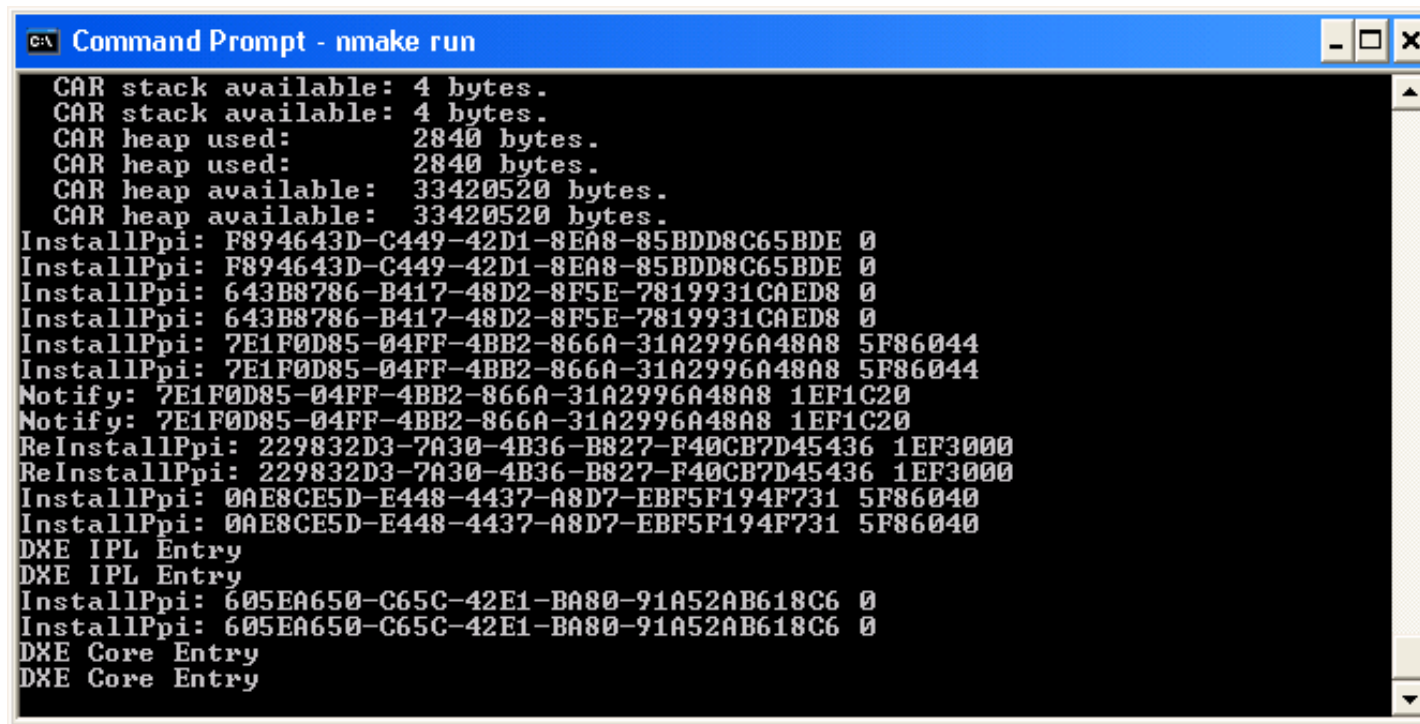
- ❧ `C:\FW\Platform\IntelSsg\Nt32\Build> set EFI_SOURCE=C:\FW`
- ❧ `C:\FW\Platform\IntelSsg\Nt32\Build> nmake`

## ❧ Run the NT32 Framework environment

- ❧ `C:\FW\Platform\IntelSsg\Nt32\Build> nmake run`

# EDK – Start Emulation

## 🌀 Run nmake



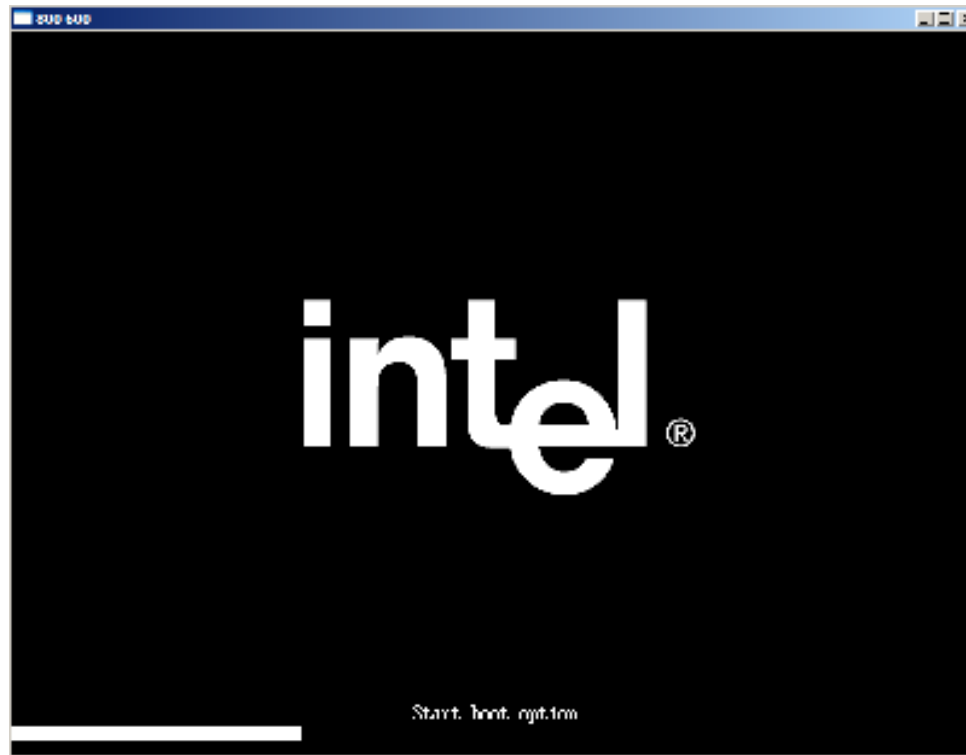
```
C:\> Command Prompt - nmake run
CAR stack available: 4 bytes.
CAR stack available: 4 bytes.
CAR heap used: 2840 bytes.
CAR heap used: 2840 bytes.
CAR heap available: 33420520 bytes.
CAR heap available: 33420520 bytes.
InstallPpi: F894643D-C449-42D1-8EA8-85BDD8C65BDE 0
InstallPpi: F894643D-C449-42D1-8EA8-85BDD8C65BDE 0
InstallPpi: 643B8786-B417-48D2-8F5E-7819931CAED8 0
InstallPpi: 643B8786-B417-48D2-8F5E-7819931CAED8 0
InstallPpi: 7E1F0D85-04FF-4BB2-866A-31A2996A48A8 5F86044
InstallPpi: 7E1F0D85-04FF-4BB2-866A-31A2996A48A8 5F86044
Notify: 7E1F0D85-04FF-4BB2-866A-31A2996A48A8 1EF1C20
Notify: 7E1F0D85-04FF-4BB2-866A-31A2996A48A8 1EF1C20
ReInstallPpi: 229832D3-7A30-4B36-B827-F40CB7D45436 1EF3000
ReInstallPpi: 229832D3-7A30-4B36-B827-F40CB7D45436 1EF3000
InstallPpi: 0AE8CE5D-E448-4437-A8D7-EBF5F194F731 5F86040
InstallPpi: 0AE8CE5D-E448-4437-A8D7-EBF5F194F731 5F86040
DXE IPL Entry
DXE IPL Entry
InstallPpi: 605EA650-C65C-42E1-BA80-91A52AB618C6 0
InstallPpi: 605EA650-C65C-42E1-BA80-91A52AB618C6 0
DXE Core Entry
DXE Core Entry
```

# EDK - NT32 Emulated Boot

---

## 🌀 Splash screen displayed

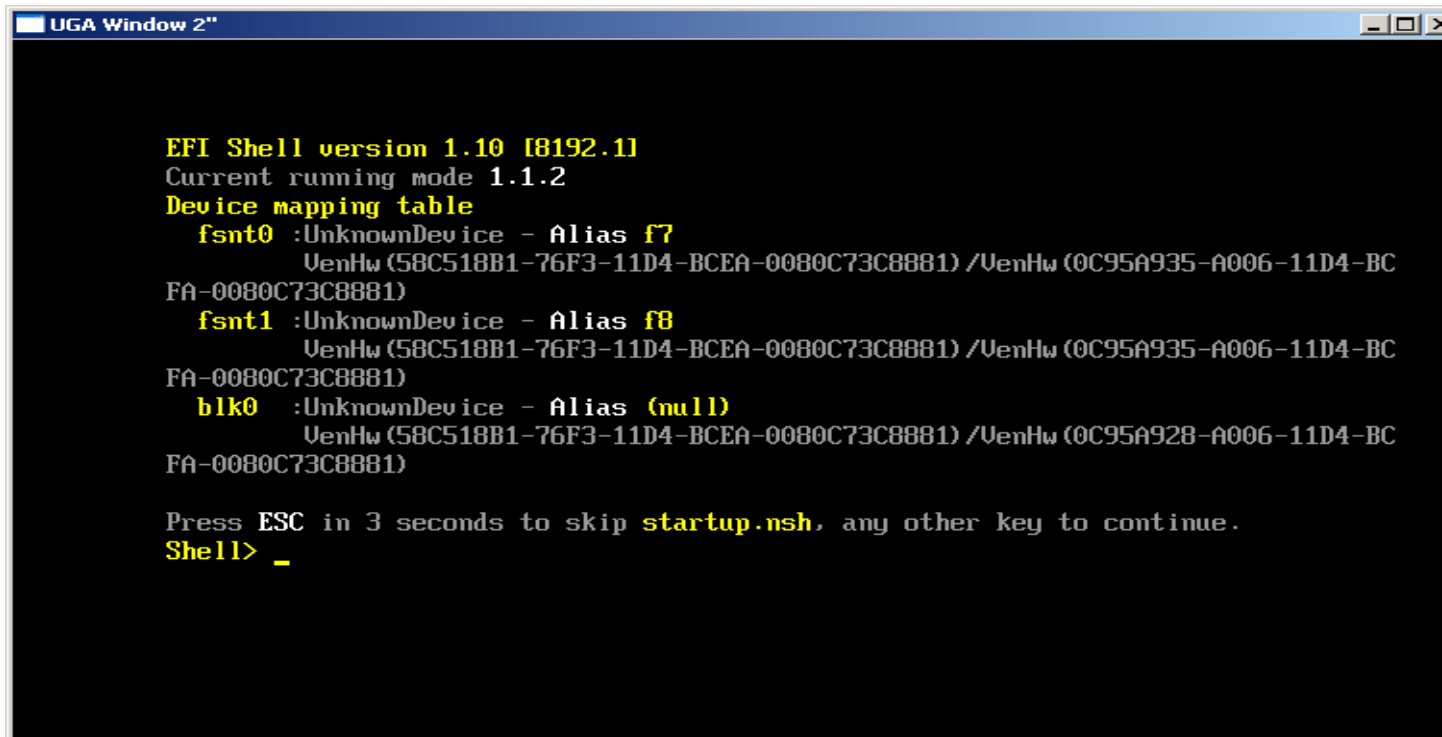
🌀 Windows drivers simulate output to screen, input from keyboard and disk operations to Hard Drive



# EDK - Boot to EFI Shell

---

- Load the EFI Shell
- Switch to fsnt0: for the local disk
  - Mapped to **EFI\_WIN\_NT\_FILE\_SYSTEM**



```
UGA Window 2"

EFI Shell version 1.10 [8192.1]
Current running mode 1.1.2
Device mapping table
  fsnt0 :UnknownDevice - Alias f7
        VenHw (58C518B1-76F3-11D4-BCEA-0080C73C8881) /VenHw (0C95A935-A006-11D4-BC
        FA-0080C73C8881)
  fsnt1 :UnknownDevice - Alias f8
        VenHw (58C518B1-76F3-11D4-BCEA-0080C73C8881) /VenHw (0C95A935-A006-11D4-BC
        FA-0080C73C8881)
  blk0  :UnknownDevice - Alias (null)
        VenHw (58C518B1-76F3-11D4-BCEA-0080C73C8881) /VenHw (0C95A928-A006-11D4-BC
        FA-0080C73C8881)

Press ESC in 3 seconds to skip startup.nsh, any other key to continue.
Shell> _
```

# EFI Application Sample - Hello.efi

---

```
//write program with EFI service
#include "efi.h"
```

```
EFI_STATUS
InitializeHelloApplication (
    IN EFI_HANDLE      ImageHandle,
    IN EFI_SYSTEM_TABLE *SystemTable
)
{
    UINTN Index;

    SystemTable->ConOut->OutputString(SystemTable->ConOut,
        L"Hello application started\n");
    SystemTable->ConOut->OutputString(SystemTable->ConOut,
        L"\n\r\n\r\n\rHit any key to exit this image\n\r");
    SystemTable->BootServices->WaitForEvent(
        1, &(SystemTable->ConIn->WaitForKey), &Index);
    SystemTable->ConOut->OutputString(SystemTable->ConOut,
        L"\n\r\n\r");

    return EFI_SUCCESS;
}
```



# EFI Application Sample - Hello.efi

---

```
//Write program with EFI Library
#include "efi.h"
#include "efilib.h"

EFI_STATUS
InitializeHelloLibApplication (
    IN EFI_HANDLE          ImageHandle,
    IN EFI_SYSTEM_TABLE    *SystemTable
)
{
    InitializeLib (ImageHandle, SystemTable);

    Print(L"\n\n\nHelloLib application started\n\n\n");
    Print(L"\nHit any key to exit this image\n");
    WaitForSingleEvent(ST->ConIn->WaitForKey,0);
    ST->ConOut->OutputString (ST->ConOut, L"\n\r\n\r");

    return EFI_SUCCESS;
}
```

# Run EFI Application

---

- ❧ 直接输入应用程序的名称
- ❧ 可以使用 **path** 来做环境参数
- ❧ **EFI Shell** 会自动加上 **".efi"**

**Path=fs0:\;.**

**Fs0:> hello1.efi**

**Fs0:> hello2.efi**

- ❧ 可以使用 **.nsh** 结尾的批处理文件, 使用与 **DOS** 下的 **.bat** 文件相似

# Appendix

---

## ❧ AMI

❧ <http://www.ami.com>

## ❧ Tiano Web Site

❧ <http://www.tianocore.org>

## ❧ Intel Efi Web Site

❧ <http://www.intel.com/technology/efi/>



American  
Megatrends

Get your computer addicted to AMIBIOS