

presented by



UEFI Self-Certification Test (SCT)

UEFI Summer Plugfest – July 6-9, 2011

Presented by

Dong Wei (HP) & Harry Hsiung (Intel)

Agenda

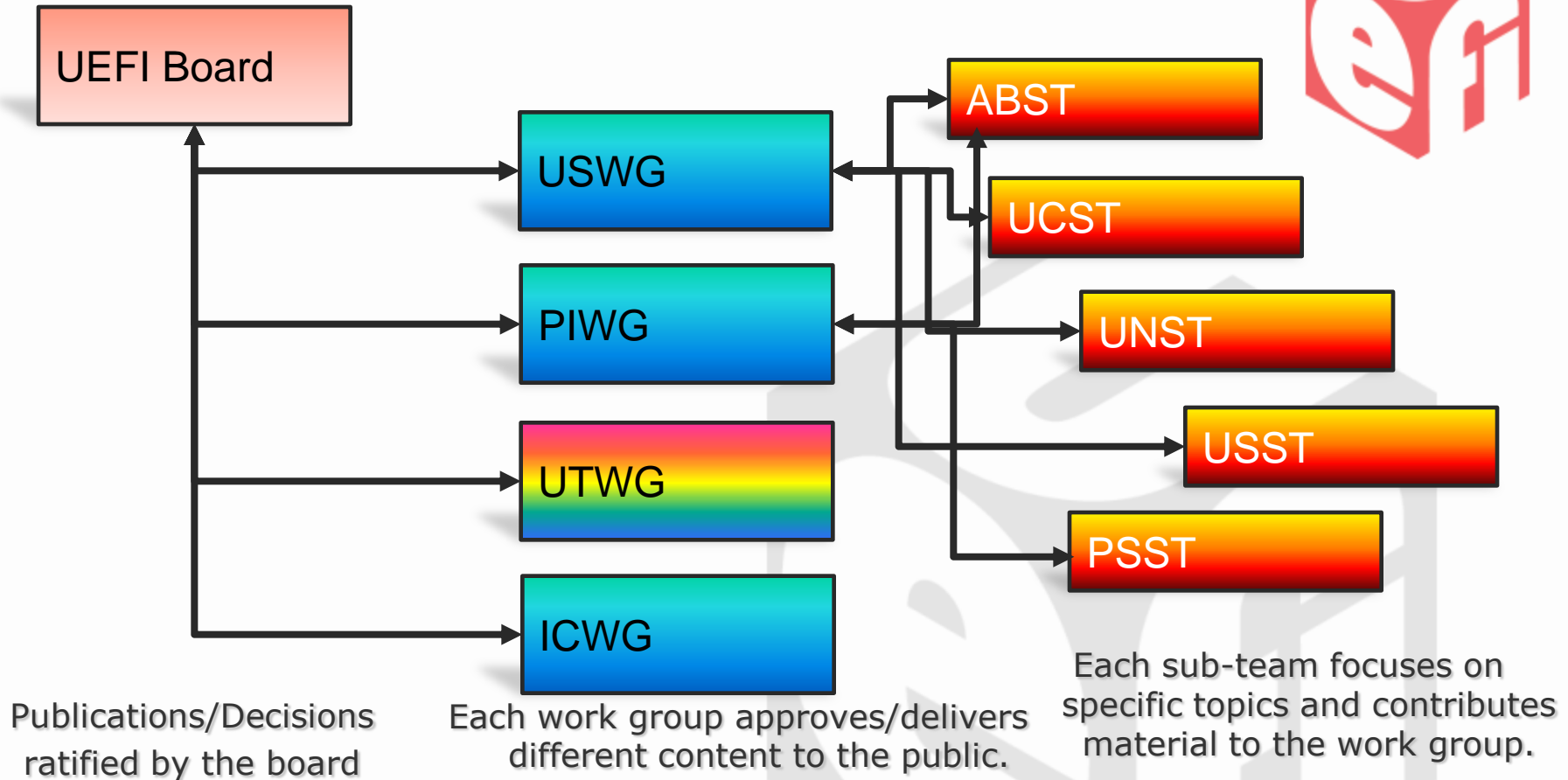


- UTWG & UEFI SCT
- UEFI SCT Details
- UEFI SCT Demo





The UEFI Forum & UTWG



UEFI Test Workgroup (UTWG) owns UEFI SCT

UEFI SCT Objectives



- UEFI SCT is to provide the computing industry with a test suite to verify the implementations in accordance with the UEFI Specification
 - UEFI Specification Requirements are specified in Section 2.6
 - Compliance is part of the UEFI logo usage guideline <http://www.uefi.org/about/logo/>

UEFI SCT

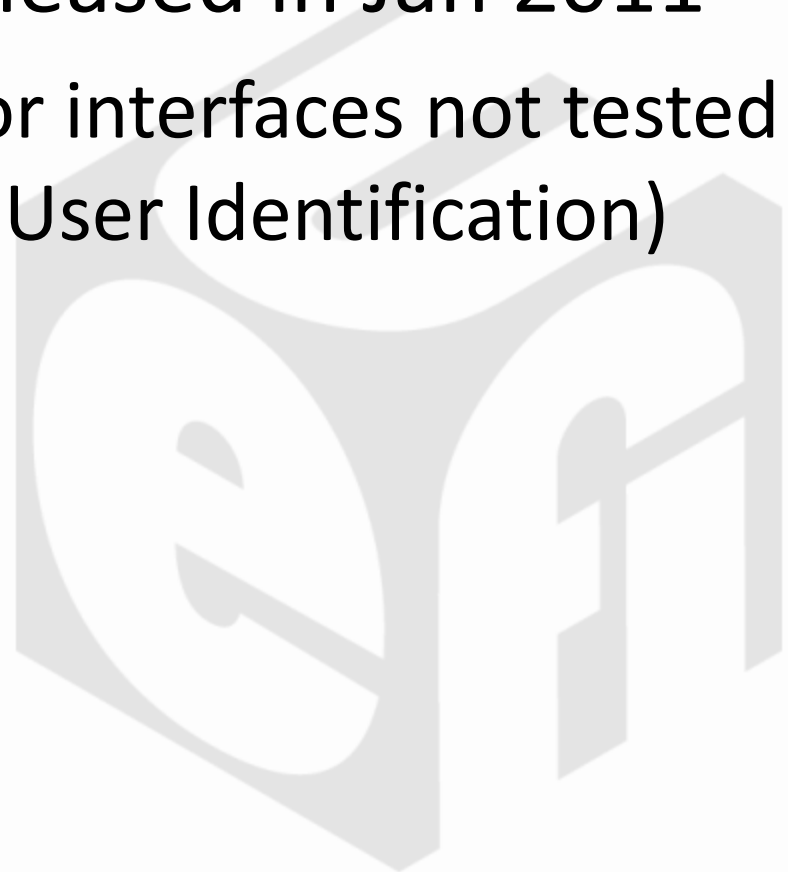


- **NOT** a **debug** environment for a UEFI implementation
- **NOT** a code coverage or performance analysis tool
- **NOT** a functional test tool
- SCT tests for existence of protocols and the input and output parameters

UEFI SCT Status



- UEFI SCT 2.3 was released in Jan 2011
 - See release notes for interfaces not tested (e.g., ARM binding, User Identification)



UEFI SCT Plan



- UEFI SCT 2.3 Patch
 - Include ARM Binding Tests
 - Target: UEFI Plugfest in Oct., 2011
- UEFI SCT 2.3.1 Release
 - Target: 12 months after the spec release (4/2012)
 - Pre-alpha package: available at this plugfest
 - Focus: Secure Boot Interfaces

Materials



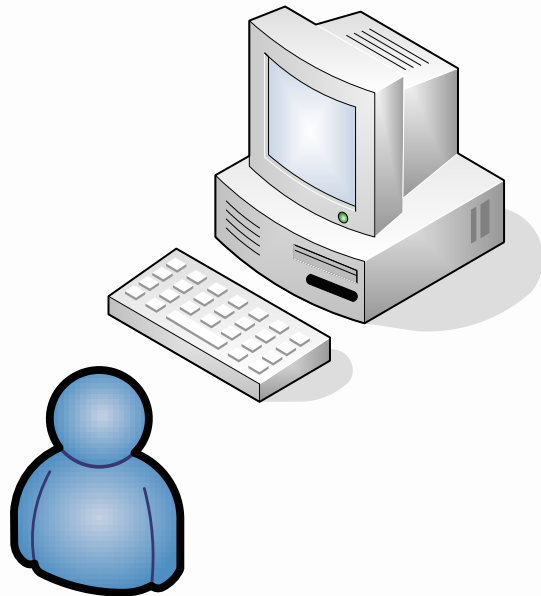
- www.uefi.org/specs/ including
 - Documents for the UEFI SCT
 - SCT2_3_ver1_1.pdf
 - UEFI_SCT2_3 User Guide_ver1_1.pdf
 - UEFI_2_3_SCT_Getting_Started_ver1_1.pdf
 - Source Code of the UEFI SCT
 - Executable images on IA32, X64\Intel® 64, and IA-64 platforms

Getting started



- Get download @ UEFI.org/specs
- Unzip to USB storage device
- Pick correct architecture (ie X64/EM64T)
- Boot to shell
- Run X64install.nsh install to usb device
- **SCT** `-u` to run in user mode and select the tests you want to run
- Generate report (.csv for MS Excel) or look at raw log files for failures

SCT Test Suite – Native Mode



```
EFI Self Certification Test (v 0.1)

Test Case Management      Description
-----
[x] Boot Services Test
[ ] Runtime Services Test
[ ] Loaded Image Protocol Test
[x] Device Path Protocol Test
[ ] Driver Model Test
[ ] Console Support Test
[ ] Bootable Image Support Test
[ ] PCI Bus Support Test
[ ] SCSI Bus Support Test
[ ] USB Support Test
[ ] Network Support Test
[ ] Debugger Support Test
[ ] Compression Test
[ ] Device IO Protocol Test
[ ] EFI Byte Code Test

Order: 0
Passes: 1
Failures: 0

Up/Dn  Select Item  Enter  Select SubMenu  F9  Run
Space  Change Status  ESC   Exit
```

UEFI-based Target Machine
Running SCT in usual mode

User interface for
local SCT validation **SCT -u**

In native mode SCTs run on the system under test in the shell

Native - Test Case selection



EFI Self Certification Test (v 0.1)

Test Case Management	Description
[x] Boot Services Test	
[] Runtime Services Test	
[] Loaded Image Protocol Test	
[X] Device Path Protocol Test	
[] Driver Model Test	
[] Console Support Test	
[] Bootable Image Support Test	
[] PCI Bus Support Test	
[] SCSI Bus Support Test	
[] USB Support Test	
[] Network Support Test	
[] Debugger Support Test	
[] Compression Test	
[] Device IO Protocol Test	
[] EFI Byte Code Test	

Order: 0
Passes: 1
Failures: 0

Up/Dn Select Item Enter Select SubMenu F9 Run
Space Change Status ESC Exit

Native - Configuration



EFI Self Certification Test (v 0.1)

Test Environment Configuration			Description
Test Case Max Run Time	[0]	Sets the maximum run time for one test case, in second (0 means unlimited)
Enable Screen Output	[True]	
Bios Id	[EFI 1.10]	
Platform Number	[0]	
Configuration Number	[0]	
Scenario String	[]	
			Value Range: Max: 400 Min: 0

Up/Dn Select Item ESC Save & Exit
F3 Set to Default

Native - Sequence



EFI Self Certification Test (v 0.1)

Main Menu	Description
▶ Test	Open File: fs0:\SCT\Sequence
▶ Test	
▶ Test	
▶ Help	

File Name: [2.seq]
Files of type: Sequence Files (*.seq)

Up/Dn Select Item F5 Load Sequence Enter Select SubMenu
F4 Reset results F6 Save Sequence ESC Exit



Native - Report



EFI Self Certification Test (v 0.1)

Main Menu	Description
▶ Test	Save File: fs0:\SCT\Report
▶ Test	te test
▶ Test	
▶ Help	

20031108.csv

20031101.csv

File Name: [_]

Save as type: CSV Files (*.csv)

Press F2 to edit filename

Up/Dn Select Item F5 Load Sequence Enter Select SubMenu
F4 Reset results F6 Save Sequence ESC Exit

SCT log file raw test output



GetNextVariableName_Conf

Revision 0x00010000

Test Entry Point GUID: E8014C92-15C4-42A8-8B0D-6080C47D3778

Test Support Library GUIDs:

1F9C2AE7-F147-4D19-A5E8-255AD005EB3E

7FD8C38D-7C5C-42FC-B044-3A834A617476

UEFI 2.3

Test Configuration #0

Perform the consistence check of GetNextVariableName() service.

Logfile: "\\SCT\Log\Runtime Services Test\Variable Services Test\GetNextVariableName_Conf_0_0_E8014C92-15C4-42A8-8B0D-6080C47D3778.log"

Test Started: 07/01/2011 01:08

RT.GetNextVariableName - With VariableNameSize is NULL -- PASS

5826847A-9067-4F9F-8838-0BF8EC20171C

c:\uefisct\SctPkg\TestCase\UEFI\EFI\RuntimeServices\VariableServices\BlackBoxTest\VariableServicesBBTestConformance.c:1553:Status - Invalid Parameter, Expected - Invalid Parameter

RT.GetNextVariableName - With VariableName is NULL -- PASS

8E8258DC-6634-4DE1-857A-60457EFA7C21

c:\uefisct\SctPkg\TestCase\UEFI\EFI\RuntimeServices\VariableServices\BlackBoxTest\VariableServicesBBTestConformance.c:1632:Status - Invalid Parameter, Expected - Invalid Parameter

RT.GetNextVariableName - With VendorGuid is NULL -- PASS

99A357F0-B6C5-4AEC-9648-34732D2A4950

c:\uefisct\SctPkg\TestCase\UEFI\EFI\RuntimeServices\VariableServices\BlackBoxTest\VariableServicesBBTestConformance.c:1712:Status - Invalid Parameter, Expected - Invalid Parameter

SCT log file



RT.GetNextVariableName - With VariableNameSize is 2 -- PASS

51C19DBA-BAF6-4854-AC09-604547886798

c:\uefisct\SctPkg\TestCase\UEFI\EFI\RuntimeServices\VariableServices\BlackBoxTest\VariableServicesBBTestConformance.c:1833:Status - Buffer Too Small, Expected - Buffer Too Small

RT.GetNextVariableName - After entire variable list returned -- PASS

FE09FF82-B289-449F-B083-981D68D917B1

c:\uefisct\SctPkg\TestCase\UEFI\EFI\RuntimeServices\VariableServices\BlackBoxTest\VariableServicesBBTestConformance.c:2009:Status - Not Found, Expected - Not Found

Returned Status Code: Success

GetNextVariableName_Conf: [PASSED]

Passes..... 5

Warnings..... 0

Errors..... 0

UEFI 2.3

Revision 0x00010000

Test Entry Point GUID: E8014C92-15C4-42A8-8B0D-6080C47D3778

Logfile: "\SCT\Log\Runtime Services Test\Variable Services Test\GetNextVariableName_Conf_0_0_E8014C92-15C4-42A8-8B0D-6080C47D3778.log"

Test Finished: 07/01/2011 01:08

Elapsed Time: 00 Days 00:00:12-----

SCT Report in Excel .csv



Service\Protocol Name	Total	Failed	Passed							
Runtime Services Test\Variable Services Test	46	0	46							
Total service\Protocol	46	0	46							

Service\Protocol Name	Index	Instance	Iteration	Guid	Result	Title	Runtime Ir	Case Revis	Case GUID	Device Pat
Service\Protocol Name	Index	Instance	Iteration	Guid	Result	Title	Runtime Ir	Case Revis	Case GUID	
Runtime Services Test\Variable Services Test	5.2.1.2.1	0	0	5826847A	PASS	RT.GetNextVariableName - GetNextVariableName() returns EFI_INVA c:\uefisct\0x000100(E8014C92-15C4-42A8				
Runtime Services Test\Variable Services Test	5.2.1.2.2	0	0	8E8258DC	PASS	RT.GetNextVariableName - GetNextVariableName() returns EFI_INVA c:\uefisct\0x000100(E8014C92-15C4-42A8				
Runtime Services Test\Variable Services Test	5.2.1.2.3	0	0	99A357F0	PASS	RT.GetNextVariableName - GetNextVariableName() returns EFI_INVA c:\uefisct\0x000100(E8014C92-15C4-42A8				
Runtime Services Test\Variable Services Test	5.2.1.2.4	0	0	51C19DBA	PASS	RT.GetNextVariableName - GetNextVariableName() returns EFI_BUFF c:\uefisct\0x000100(E8014C92-15C4-42A8				
Runtime Services Test\Variable Services Test	5.2.1.2.5	0	0	FE09FF82	PASS	RT.GetNextVariableName - GetNextVariableName() returns EFI_NOT c:\uefisct\0x000100(E8014C92-15C4-42A8				
Runtime Services Test\Variable Services Test	5.2.1.2.6	0	0	12071508	PASS	RT.GetNextVariableName - GetNextVariableName() gets the exist var c:\uefisct\0x000100(66A7216F-A855-47D3				
Runtime Services Test\Variable Services Test	5.2.1.2.7	0	0	A85043BC	PASS	RT.GetNextVariableName - GetNextVariableName() gets the exist var c:\uefisct\0x000100(66A7216F-A855-47D3				
Runtime Services Test\Variable Services Test	5.2.1.1.1	0	0	B0D54FEE	PASS	RT.GetVariable - GetVariable() returns EFI_INVALID_PARAMETER with c:\uefisct\0x000100(D90941AA-B626-4666				
Runtime Services Test\Variable Services Test	5.2.1.1.2	0	0	390C5E26	PASS	RT.GetVariable - GetVariable() returns EFI_INVALID_PARAMETER with c:\uefisct\0x000100(D90941AA-B626-4666				
Runtime Services Test\Variable Services Test	5.2.1.1.3	0	0	176354A6	PASS	RT.GetVariable - GetVariable() returns EFI_INVALID_PARAMETER with c:\uefisct\0x000100(D90941AA-B626-4666				

Look up Index for each testcase (i.e. ones that result is FAILED).

Check SCT2_3_ver1_1.pdf for testcase or GUID of test and look to see what the test case is doing.

Look in .log file for specific test case for what failed.

Lookup in source code file to see exactly what UEFI protocol is being called and what parameters are being used .

UefiSctEdkII-Dev.zip under SctPkg\TestCase\UEFI

SCT for IHV (plugin cards)



- Make sure the plugin card driver is started before SCT testcases are run
- Go to the Test Device Configuration menu to select your driver in the handle database for the test harness to connect it before running the testcases.

SCT for IHV



```
H          - Print this help information
I <Handle> - Insert one device into the configuration file
L          - List all devices in the configuration file
Q          - Quit
R <Index>  - Remove one device from the configuration file
S <Type>   - Scan devices in the system
            (Type 0: All, Type 1: With Option ROM)
U <Index>  - List one device in the configuration file in verbose mode
```

For new users, select 'S' first to get the devices with Option ROM, and then use 'I <Handle>' to insert the devices which should be tested into the configuration file. For more information of this handle, please refer to the 'DH' command in Shell environment.

Scan for UEFI IHV driver in handle Database using S command

SCT for IHV



```
-S 0
6: MemMap (11:950000-BCFFFF)
7: MemMap (11:BD0000-BEFFFF)
52: VenHw (58C518B1-76F3-11D4-BCEA-0080C73C8881)
    Managed by driver <Windows Bus Driver>
53: VenHw (58C518B1-76F3-11D4-BCEA-0080C73C8881) /VenHw (AB248E99-ABE1-11D4-BD0D-00
80C73C8881)
54: VenHw (58C518B1-76F3-11D4-BCEA-0080C73C8881) /VenHw (4E11E955-CCCA-11D4-BD0D-00
80C73C8881)
    Managed by driver <Console Splitter Driver>
    Managed by driver <Platform Console Management Driver>
    Managed by driver <Console Splitter Driver>
    Managed by driver <Platform Console Management Driver>
    Managed by driver <UGA Console Driver>
    Managed by driver <Windows GOP Driver>
55: VenHw (58C518B1-76F3-11D4-BCEA-0080C73C8881) /VenHw (0C95A93D-A006-11D4-BCFA-00
80C73C8881)
```

Use “I” command to select the specific handle of the device for the SCT to run on (i.e. if GOP video device was to be selected the command would be “I 54”)

SCT for IHV



```
Managed by driver <Console Splitter Driver>
Managed by driver <Platform Console Management Driver>
Managed by driver <UGA Console Driver>
Managed by driver <Windows GOP Driver>
55: VenHw (58C518B1-76F3-11D4-BCEA-0080C73C8881) /VenHw (0C95A93D-A006-11D4-BCFA-00
80C73C8881)
  Managed by driver <Windows Serial I/O Driver>
56: VenHw (58C518B1-76F3-11D4-BCEA-0080C73C8881) /VenHw (0C95A935-A006-11D4-BCFA-00
80C73C8881)
  Managed by driver <Windows Simple File System Driver>
57: VenHw (58C518B1-76F3-11D4-BCEA-0080C73C8881) /VenHw (0C95A935-A006-11D4-BCFA-00
80C73C8881)
  Managed by driver <Windows Simple File System Driver>
Press 'q' to exit, any other key to continue
-I 54
Select the device type:
R - SCSI Raid
F - SCSI Fiber
N - NIC
U - USB
V - Video
S - Serial
P - PCI
O - Other
Enter choice: _
```

Select device type: in our GOP video example V would be selected

SCT for IHV



```
Select the device type:
```

```
R - SCSI Raid
```

```
F - SCSI Fiber
```

```
N - NIC
```

```
U - USB
```

```
V - Video
```

```
S - Serial
```

```
P - PCI
```

```
O - Other
```

```
Enter choice: U
```

```
Should test the Driver Binding Protocol? [Y(Default)/N]: Y
```

```
Should test the Driver Diagnostics Protocol? [Y(Default)/N]: Y
```

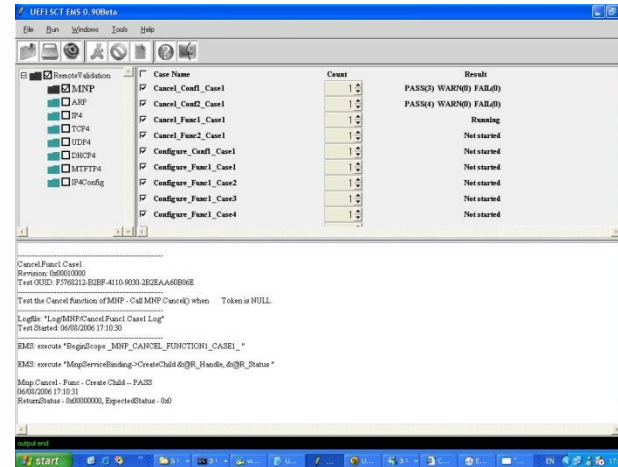
```
Should test the Driver Configuration Protocol? [Y(Default)/N]: Y
```

```
Should test the Unload Supported function? [Y(Default)/N]: Y
```

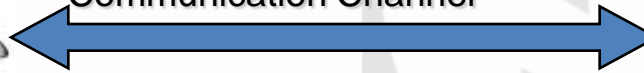
```
Should test the Runtime Supported function? [Y(Default)/N]: Y
```

Tell SCT the what protocol features are enabled in the UEFI driver:

SCT Test Suite – Passive Mode



Communication Channel



UEFI-based Target Side
Running SCT in passive mode

OS-based Management Side
User can easily finish validation
job with friendly GUI

Passive mode



- Test UEFI network stack
- Utilizes two UEFI systems where one is the target and one is the host running the agent monitoring software

UEFI 2.3 Compliance Tests



UEFI SCT EMS 0.90Beta

File Run Windows Tools Help

RemoteExecution

- Generic Test
- EFI Compliant Test
- Boot Services Test
- Runtime Services Test
 - Loaded Image Protocol Test
- Driver Model Test
 - Platform Driver Override Protocol Test
 - Bus Specific Driver Override Protocol Test
 - Component Name Protocol Test
- Console Support Test
- Bootable Image Support Test
 - Simple File System Protocol Test
 - Disk IO Protocol Test
 - Block IO Protocol Test
 - Unicode Collation Protocol Test
- PCI Bus Support Test
- SCSI Bus Support Test
- USB Support Test
- Network Support Test
- Debugger Support Test
- Compression Test
- Device IO Protocol Test
- EFI Byte Code Test
 - EBC Interpreter Protocol Test
 - Hash Testing
 - Authentication Testing
 - TapeloProtocol Testing

Case Name	Count
CheckEvent_Conf	1
CheckEvent_Func	1
CloseEvent_Func	1
CreateEventEx_Conf	1
CreateEventEx_Func	1
CreateEvent_Conf	1
CreateEvent_Func	1
RaiseTPL_Func	1
RestoreTPL_Func	1
SetTimer_Conf	1
SetTimer_Func	1
SignalEvent_Func	1
WaitForEvent_Conf	1
WaitForEvent_Func	1

UEFI SCT EMS 0.90Beta

File Run Windows Tools Help

RemoteValidation

- MNP
- ARP
- IP4
- TCP4
- UDP4
- DHCP4
- MTFTP4
- IP4Config

Case Name	Count
Cancel_Confl_Case1	1
Cancel_Conf2_Case1	1
Cancel_Func1_Case1	1
Cancel_Func2_Case1	1
Configure_Confl_Case1	1
Configure_Func1_Case1	1
Configure_Func1_Case2	1
Configure_Func1_Case3	1
Configure_Func1_Case4	1
CreateChild_Confl_Case1	1
CreateChild_Confl_Case2	1
DestroyChild_Confl_Case1	1
DestroyChild_Func1_Case1	1
GetModeData_Confl_Case1	1
GetModeData_Func1_Case1	1
GetModeData_Func2_Case1	1
GetModeData_Func2_Case2	1
GetModeData_Func2_Case3	1
Groups_Confl_Case1	1
Groups_Conf2_Case1	1
Groups_Conf2_Case2	1
Groups_Conf3_Case1	1
Groups_Conf4_Case1	1
Groups_Func1_Case1	1

Demo

- DEMO SCT



Call to action



- Run SCTs on your UEFI implementation
- Update SCT to 2.3.1 when it completed
- Looking for any contributions to the UEFI 2.3.1 SCT effort

Thanks for attending the
UEFI Summer Plugfest 2011



For more information on
the Unified EFI Forum and
UEFI Specifications, visit
<http://www.uefi.org>



But wait, there's more ...

Wed
(July 6)

- UEFI State of the Union (10:30am, Intel)
- Implementing a Secure Boot Path with UEFI 2.3.1 (1:00pm, Insyde)
- UEFI SCT Overview (2:30pm, HP/Intel)

Thu
(July 7)

- Replacing VGA: GOP Implementation in UEFI (10:30am, AMD)
- UEFI prototyping using a Windows-hosted UEFI environment (1:00pm, Phoenix)
- EFI Shell Lab (2:00-4:00pm, “Thunder”, Intel)
- GOP Enabling & Testing Lab (4:30—5:30pm, “Thunder”, Intel)

Fri
(July 8)

- Best Practices for UEFI Option ROM Developers (10:30am, AMI)

Download presentations after the plugfest at www.uefi.org