

# UEFI Compliance Testing

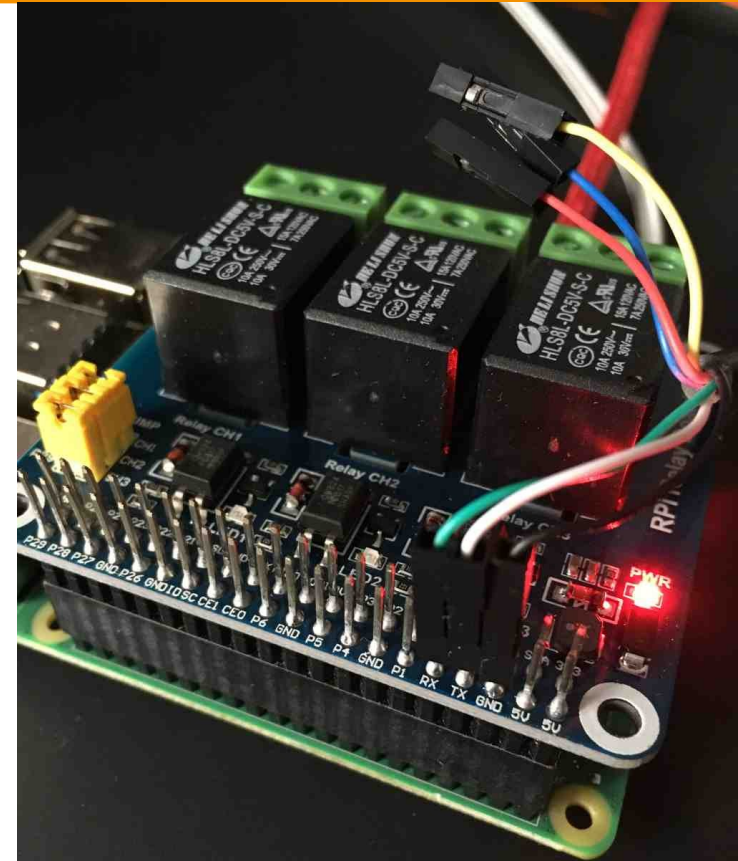
Heinrich Schuchardt  
CC BY-SA 4.0

# Agenda

- Motivation
- Open Source UEFI implementations
- UEFI test tools: SCT, FWTS
- Gaps
- Device tree validation
- Test environment in U-Boot
- Summary

# About Me

- U-Boot UEFI Maintainer
- Contributions to
  - Linux
  - GNU Linear Programming Kit (GLPK)
  - ...



# Open Source UEFI Implementations

## Tianocore EDK II

- Reference implementation
- Full UEFI scope
- SBBR compliance
- 36 configurations

```
Armada 8040 MacchiattoBin
Cortex-A72                2.00 GHz
MARVELL_EFI              16364 MB RAM

Select Language           <Standard English>   This is the option
                                                    one adjusts to change
                                                    the language for the
                                                    current system

▶ Device Manager
▶ Boot Manager
▶ Boot Maintenance Manager

Continue
Reset

↑↓=Move Highlight      <Enter>=Select Entry
```

## U-Boot

- UEFI for embedded systems
- Reduced UEFI scope
- EBBR compliance
- 786 of 1312 configurations

```
U-Boot 2020.10-00200-g0f35d96bfd (Oct 15 2020 - 19:20:09 +0200) odroid-c2

Model: Hardkernel ODROID-C2
SoC:   Amlogic Meson GXBB (S905) Revision 1f:b (0:1)
DRAM:  2 GiB
MMC:   mmc@72000: 0, mmc@74000: 1
In:    serial
Out:   serial
Err:   serial
Net:   eth_designware ethernet@c9410000: Can't get reset: -2
eth0: ethernet@c9410000
Hit any key to stop autoboot:  0
=> █
```

# Embedded Base Boot Requirements (EBBR) Specification

- Defines subset of UEFI specification
  - Targets embedded systems
  - Enough to boot operating systems
- EBBR v1.0 released in 2019  
<https://github.com/arm-software/ebbr>
- Applicable to all UEFI architectures: ARM, RISC-V, x86

# UEFI in U-Boot

- Scope
  - Embedded Base Boot Requirements (EBBR) Specification
- Boot and Runtime Services, Secure Boot
- Protocols
  - Simple Text Input (Ex), Simple Text Output, Graphics Output
  - Block IO, Simple File System, File
  - Simple Network
  - Device Path To Text, Unicode Collation, RNG
  - Only partially: HII (enough to run SCT)

# U-Boot UEFI Usage

- Used by default for booting on embedded boards by
  - Fedora
  - Suse
  - FreeBSD
  - OpenBSD

# Motivation for Complicance Testing

- Implementation perspective
  - UEFI 2.8 Errata B specification has 2484 pages
  - Requirements are highly complex and interrelated
- Application perspective
  - Provide a reliable basis for UEFI boot flow



# UEFI Lifetime

TF-A  
Tests

PI-SCT

UEFI SCT

?

fwts

TF-A

TianoCore EDK II

iPXE

BSD

OpenSBI

*U-Boot  
Drivers  
(non-UEFI)*

U-Boot

GRUB

Linux

Security  
(SEC)

Pre EFI  
Initialization  
Environment  
(PEI)

Driver  
Execution  
Environment  
(DXE)

Boot  
Device  
Selection  
(BDS)

Transient  
System  
Load  
(TSL)

Runtime  
(RT)

After  
Life  
(AL)

# UEFI Self-Certification Test

- The UEFI SCT is an EFI shell application.
- Coverage
  - boot and runtime services
  - protocols
- Test types
  - Conformance tests for handling of invalid parameters
  - Functionals tests
- 606 EFI test cases with 7142 assertions
- 1.2 million lines of code

# SCT Test Modes

- Native mode
  - Testing API
  - Runs on single system
- Passive mode
  - Testing network protocols (DHCP, IPv4, IPv6, TCP, UDP, HTTP, ...)
  - Test controlled from Windows system
  - Requires Managed Network Protocol (MNP)

# Running the SCT

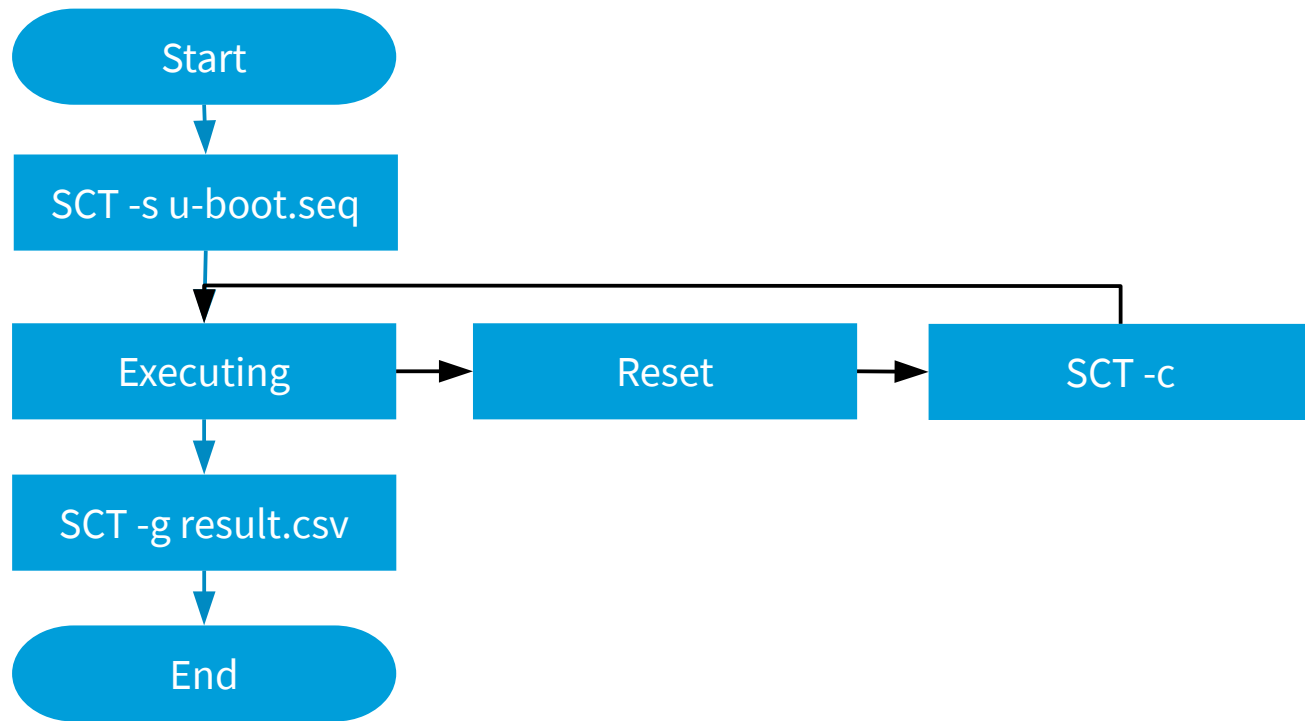
- The U-Boot Sandbox can run UEFI binaries natively
- Requires no KVM/QEMU emulation
- Fastest method of running the SCT on build system

# UEFI Self-Certification Test



# Shell Script - start.nsh

```
FS0:  
if exist run then  
    rm run  
    SCT -s uboot.seq  
else  
    SCT -c  
endif  
SCT -g result.csv  
reset -s
```



# Report

A	B	C	D	E	F
1 Self Certification Test Report					
2 Service\Protocol Name	Total	Failed	Passed		
3 BootServicesTest\EventTimerandPriorityServicesTest	34	0	34		
4 BootServicesTest\MemoryAllocationServicesTest	138	0	138		
5 BootServicesTest\ProtocolHandlerServicesTest	1216	6	1210		
6 BootServicesTest\ImageServicesTest	123	2	121		
7 BootServicesTest\MiscBootServicesTest	132	3	129		
8 RuntimeServicesTest\VariableServicesTest	59	8	51		
9 RuntimeServicesTest\TimeServicesTest	46	4	42		
10 RuntimeServicesTest\MiscRuntimeServicesTest	12	4	8		
11 LoadedImageProtocolTest	1492	0	1492		
12 DevicePathProtocols\DevicePathProcotolTest	12	0	12		
13 DevicePathProtocols\DevicePathUtilitiesProcotolTest	21	0	21		
14 DevicePathProtocols\DevicePathToTextProcotolTest	55	36	19		
15 NetworkSupportTest\SimpleNetworkProtocolTest	28	1	27		
16 StringServiceTest\UnicodeCollation2ProtocolTest	12	0	12		
17 ConsoleSupportTest\SimpleTextInputExProtocolTest	10	0	10		
18 ConsoleSupportTest\SimpleInputProtocolTest	4	0	4		
19 ConsoleSupportTest\SimpleOutputProtocolTest	66	0	66		
20 GenericTest\EFICompliantTest	35	27	8		
21 MediaAccessTest\SimpleFileSystemProtocolTest	147	35	112		
22 MediaAccessTest\BlockIOProtocolTest	16	0	16		
23 HII\Test\HII\DatabaseProtocolTest	22	4	18		
24 SecureTechTest\VRNGPProtocolTest	10	0	10		
25 Total service\Protocol	3690	130	3560		
26					
27 Service\Protocol Name	Index	Instance	Iteration	Guid	Result Title
28 BootServicesTest\ProtocolHandlerServicesTest	5.1.3.11.90	0	0	4643E80E-A6BF-412C-B4FF-9629282BC831	FAIL BS.ConnectController - Platform Driver Override's priority is higher than Bus Specific Driver Override's at EFI_TPL_
29 BootServicesTest\ProtocolHandlerServicesTest	5.1.3.11.91	0	0	25CFDFD5-D252-4515-AF8F-D8DB68F022C3	FAIL BS.ConnectController - Platform Driver Override's priority is higher than Bus Specific Driver Override's at EFI_TPL_
30 BootServicesTest\ProtocolHandlerServicesTest	5.1.3.11.92	0	0	555913E8-BA56-4C68-80B5-A96B8A3AFCB1	FAIL BS.ConnectController - Platform Driver Override's priority is higher than Bus Specific Driver Override's at EFI_TPL_
31 BootServicesTest\ProtocolHandlerServicesTest	5.1.3.12.43	0	0	8CD9BFBF-021F-469F-BCB3-9AFF5E90364B	FAIL BS.DisconnectController - DisconnectController() disconnects related child handles with Child is not NULL at EFI_TI

# SCT Results for U-Boot

2020-10-24.csv - LibreOffice Calc

File Edit View Insert Format Styles Sheet Data Tools Window Help

Liberation Sans 10 pt B I U A % 0.0 0.00

A1 fx Σ Self Certification Test Report

A				B	C	D	E	F
1	Self Certification Test Report				Total	Failed	Passed	
2	Service\Protocol Name							
3	BootServicesTest\EventTimerandPriorityServicesTest				34	0	34	
4	BootServicesTest\MemoryAllocationServicesTest				138	0	138	
5	BootServicesTest\ProtocolHandlerServicesTest				1216	6	1210	
6	BootServicesTest\ImageServicesTest				123	2	121	
7	BootServicesTest\MiscBootServicesTest				132	3	129	
8	Runtime							
9	Runtime							
10	Runtime							
11	Load							
12	Device							
13	Device							
14	Device							
15	Network							
16	String							
17	Console							
18	Console							
19	ConsoleSupportTest\SimpleOutputProtocolTest				66	0	66	
20	GenericTest\EfiCompliantTest				35	27	8	
21	MediaAccessTest\SimpleFileSystemProtocolTest				147	35	112	
22	MediaAccessTest\BlockIOProtocolTest				16	0	16	
23	HiiTest\HiiDatabaseProtocolTest				22	4	18	
24	SecureTechTest\RNGProtocolTest				10	0	10	
25	Total service\Protocol				3690	130	3560	
26								
27	Service\Protocol Name				Index	Instance	Iteration	Guid
28	BootServicesTest\ProtocolHandlerServicesTest				5.1.3.11.90	0	0	4643E80E-A6BF-412C-B4FF-9
29	BootServicesTest\ProtocolHandlerServicesTest				5.1.3.11.91	0	0	25CFFDF5-D252-4515-AF8F-D
30	BootServicesTest\ProtocolHandlerServicesTest				5.1.3.11.92	0	0	555913E8-BA56-4C68-80B5-A
31	BootServicesTest\ProtocolHandlerServicesTest				5.1.3.12.43	0	0	8CD9BFBF-021F-469F-BCB3-9

2020-10-24

Find Find All Formatted Display Match Case

Sheet 1 of 1 Default Page Style English (USA) Average: ; Sum: 0 100%

Due to missing implemented features



# Pros & Cons

- Conformance well covered
- Functionality of individual API services well tested
- Long runtime (hours)
- Gaps in coverage
  - Test for image authentication missing (Bug #2230)
- RISC-V support missing
- Missing clean separation between SCT and EDK II

# Example

## UnicodeCollationProtocol2 StrUpr()

- U-Boot's conversion

"0P[A-D]@ 0U@ " -> "0P[A-D]@ 0U@ "

- SCT claimed this was an error
  - EDK does not support uppercasing Unicode
  - SCT used the EDK library (sic!).

# UEFI SCT Status

- Development has nearly stopped
- Stuck on path to UEFI 2.7 (current spec is 2.8 B)
- Building for X86\_64 fails with GCC and VS2019

# UEFI Lifetime

TF-A  
Tests

PI-SCT

UEFI SCT

?

fwts

TF-A

TianoCore EDK II

iPXE

BSD

OpenSBI

*U-Boot  
Drivers  
(non-UEFI)*

U-Boot

GRUB

Linux

Security  
(SEC)

Pre EFI  
Initialization  
Environment  
(PEI)

Driver  
Execution  
Environment  
(DXE)

Boot  
Device  
Selection  
(BDS)

Transient  
System  
Load  
(TSL)

Runtime  
(RT)

After  
Life  
(AL)

# Firmware Test Suite

- Check firmware from Linux
- Scope
  - ACPI tables
  - SMBIOS table
  - UEFI runtime services

# Firmware Test Suite

```
> # fwts dmichk uefirtvariable --stdout-summary
Running 2 tests, results appended to results.log
Test: DMI/SMBIOS table tests.
    Find and test SMBIOS Table Entry Points.                6 passed
    Test DMI/SMBIOS tables for errors.                       6 passed, 2 failed
    Test DMI/SMBIOS3 tables for errors.                      1 skipped
    Test ARM SBBR SMBIOS structure requirements.
FAILED_LOW
Test: UEFI Runtime service variable interface tests.
    Test UEFI RT service get variable interface.            1 skipped
    Test UEFI RT service get next variable name interface.  1 skipped
    Test UEFI RT service set variable interface.            1 skipped
    Test UEFI RT service query variable info interface.     1 skipped
    Test UEFI RT service variable interface stress test.    1 skipped
    Test UEFI RT service set variable interface stress t..  1 skipped
    Test UEFI RT service query variable info interface s..  1 skipped
    Test UEFI RT service get variable interface, invalid..  1 skipped
    Test UEFI RT variable services supported status.        2 passed, 2 failed
FAILED_HIGH
# █
```

# FWTS: results.log

High failures: 2

uefirtvariable: Get the Setvariable runtime service supported via RuntimeServicesSupported variable. But actually is not supported by firmware.

uefirtvariable: Get the QueryVarInfo runtime service supported via RuntimeServicesSupported variable. But actually is not supported by firmware.

Medium failures: NONE

Low failures: 2

dmicheck: String index 0x01 in table entry 'Processor Information (Type 4)' @ 0x7aee5114, field 'Processor Manufacturer', offset 0x07 has a default value 'Unknown' and probably has not been updated by the BIOS vendor.

dmicheck: String index 0x01 in table entry 'Processor Information (Type 4)' @ 0x7aee5114, field 'Processor Version', offset 0x10 has a default value 'Unknown' and probably has not been updated by the BIOS vendor.

Other failures: NONE

Test	Pass	Fail	Abort	Warn	Skip	Info
dmicheck	12	2			1	
uefirtvariable	2	2			8	
Total:	14	4	0	0	9	0

# UEFI Lifetime

TF-A  
Tests

PI-SCT

UEFI SCT

?

fwts

TF-A

TianoCore EDK II

iPXE

BSD

OpenSBI

*U-Boot  
Drivers  
(non-UEFI)*

U-Boot

GRUB

Linux

Security  
(SEC)

Pre EFI  
Initialization  
Environment  
(PEI)

Driver  
Execution  
Environment  
(DXE)

Boot  
Device  
Selection  
(BDS)

Transient  
System  
Load  
(TSL)

Runtime  
(RT)

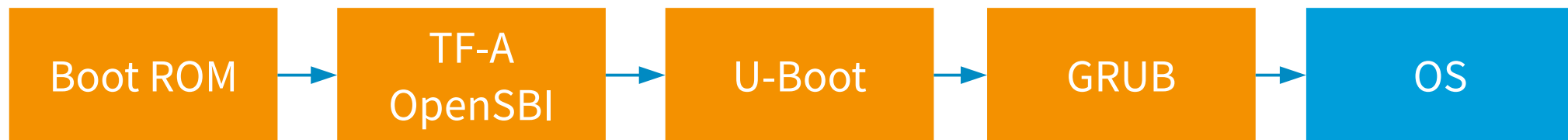
After  
Life  
(AL)



# Not Covered by SCT and FWTS

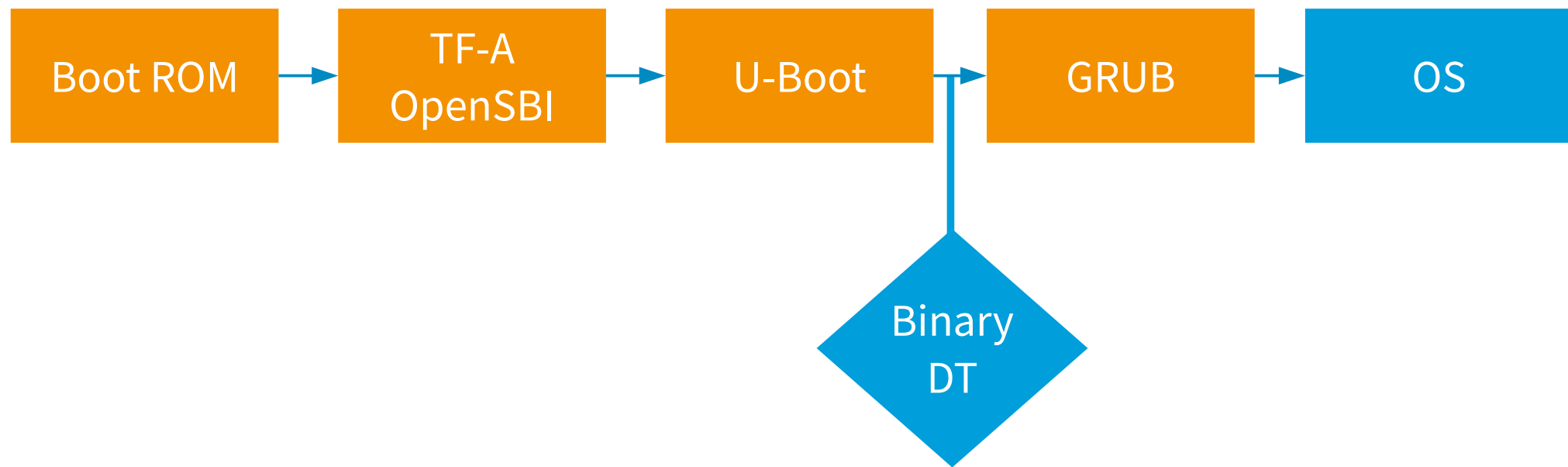
- Boot device selection
  - Boot manager, boot variables
- Transient system load
  - ExitBootServices()
    - Release of non-runtime memory
    - MMU, cache, and interrupts state
  - SetVirtualAddressMap(), ConvertPointer()
  - Device tree

# Sources of Device Tree



- Device trees can be provided on different stages.
- Device trees are fixed up on all stages.
- Device trees can be generated on the fly (e.g. QEMU)

# Where to Validate Device Tree



# Device Tree Bindings

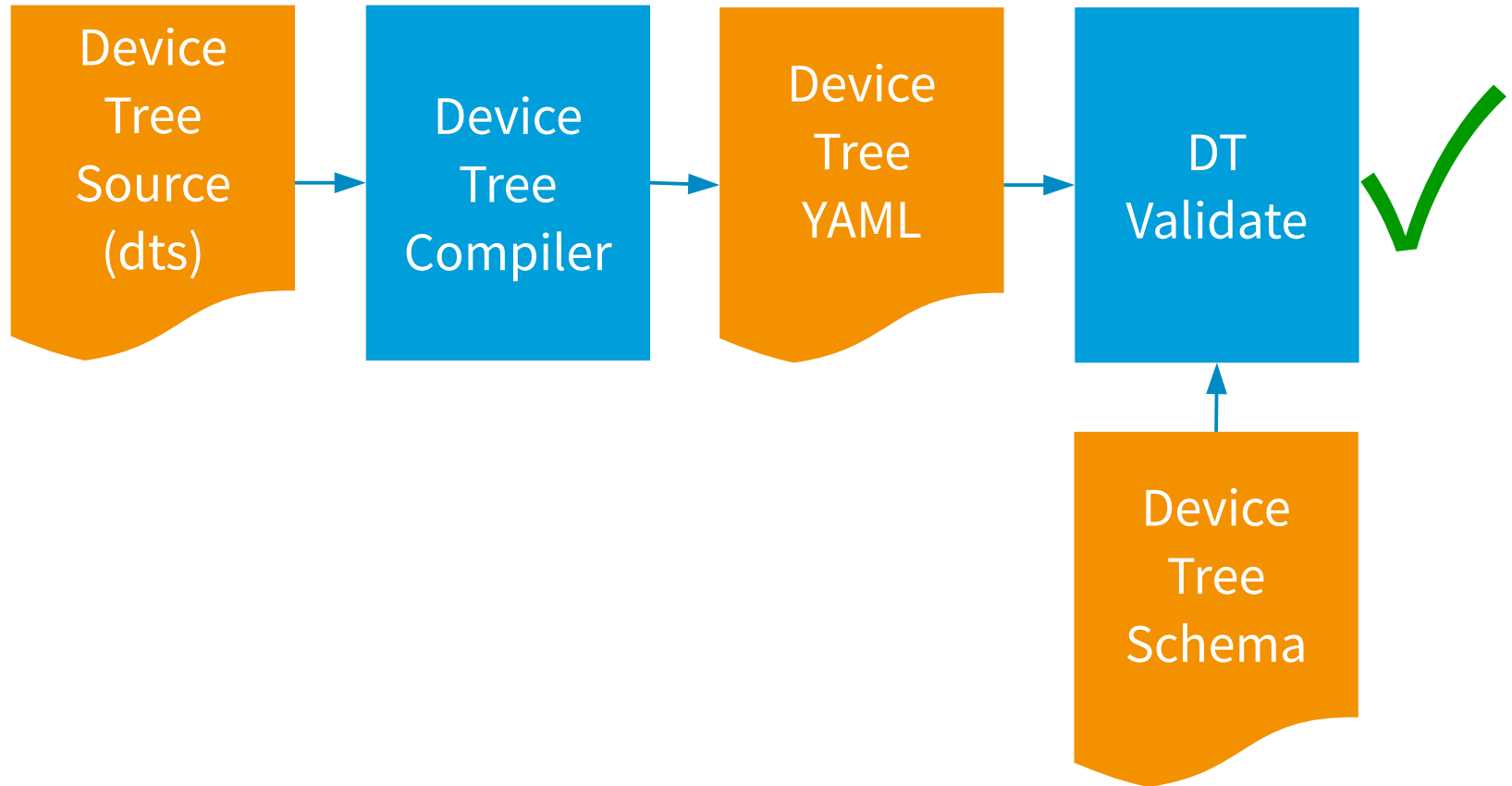
- On the move from text to YAML schema
- Definitions can be validated
- Device trees can be validated

For details see

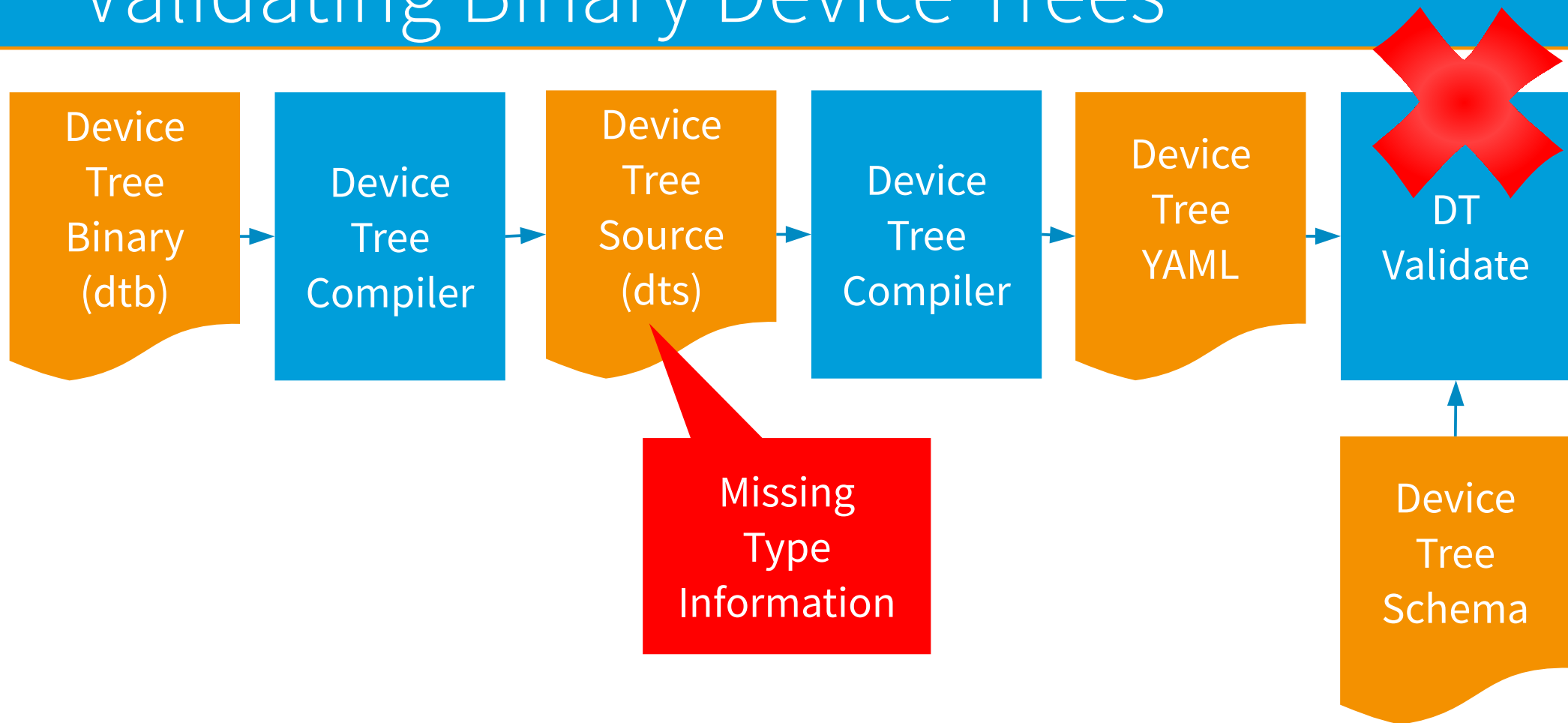
<https://github.com/devicetree-org/dt-schema>

Maintainer: Rob Herring

# Device Tree Validation



# Validating Binary Device Trees



# U-Boot UEFI Selftests

- Originally needed as U-Boot could not run SCT
- Covers Transient System Load
  - SetVirtualAddressMap(), ConvertPointer()
  - Device Tree (present, well formed)
  - Boot Hart ID
- Completely separated code base (except headers)
  - Only EFI API used to call into U-Boot
- Compiled into U-Boot for easy debugging
- 43 test case, 597 assertions, 10576 lines

# Testing U-Boot

Gitlab CI, Travis CI, Amazon CI

Python Tests

U-Boot

C unit tests  
=> ut all

UEFI selftest  
=> bootefti selftest



# U-Boot UEFI Selftests

```
denx : u-boot — Konsole
Executing 'text output'

Color palette
0000000000000000 0000000000000010 0000000000000020 0000000000000030 0000000000000040 0000000000000050 0000000000000060 0000000000000070
0000000000000001 0000000000000011 0000000000000021 0000000000000031 0000000000000041 0000000000000051 0000000000000061 0000000000000071
0000000000000002 0000000000000012 0000000000000022 0000000000000032 0000000000000042 0000000000000052 0000000000000062 0000000000000072
0000000000000003 0000000000000013 0000000000000023 0000000000000033 0000000000000043 0000000000000053 0000000000000063 0000000000000073
0000000000000004 0000000000000014 0000000000000024 0000000000000034 0000000000000044 0000000000000054 0000000000000064 0000000000000074
0000000000000005 0000000000000015 0000000000000025 0000000000000035 0000000000000045 0000000000000055 0000000000000065 0000000000000075
0000000000000006 0000000000000016 0000000000000026 0000000000000036 0000000000000046 0000000000000056 0000000000000066 0000000000000076
0000000000000007 0000000000000017 0000000000000027 0000000000000037 0000000000000047 0000000000000057 0000000000000067 0000000000000077
0000000000000008 0000000000000018 0000000000000028 0000000000000038 0000000000000048 0000000000000058 0000000000000068 0000000000000078
0000000000000009 0000000000000019 0000000000000029 0000000000000039 0000000000000049 0000000000000059 0000000000000069 0000000000000079
000000000000000a 000000000000001a 000000000000002a 000000000000003a 000000000000004a 000000000000005a 000000000000006a 000000000000007a
000000000000000b 000000000000001b 000000000000002b 000000000000003b 000000000000004b 000000000000005b 000000000000006b 000000000000007b
000000000000000c 000000000000001c 000000000000002c 000000000000003c 000000000000004c 000000000000005c 000000000000006c 000000000000007c
000000000000000d 000000000000001d 000000000000002d 000000000000003d 000000000000004d 000000000000005d 000000000000006d 000000000000007d
000000000000000e 000000000000001e 000000000000002e 000000000000003e 000000000000004e 000000000000005e 000000000000006e 000000000000007e
000000000000000f 000000000000001f 000000000000002f 000000000000003f 000000000000004f 000000000000005f 000000000000006f 000000000000007f

Testing cursor column update

HA3
Executing 'text output' succeeded

Setting up 'task priority levels'
Setting up 'task priority levels' succeeded

Executing 'task priority levels'
Executing 'task priority levels' succeeded
```

# Take Aways

- UEFI Self-Certification Tests needs more developers
- Missing test coverage for boot device selection and transient system load
- Compliance testing for binary device trees missing

**Q&A**