

Device Tree Tools Status

Frank Rowand, Sony

December 2, 2016

161201_1957

What is a Device Tree?

A device tree describes hardware that can not be located by probing.

Why my Device Tree talks?

Debugging device tree problems is not easy.

- tools do not exist or are not sufficient

The above claim was from 2015

But tools are slowly improving

Today's talk is an overview of the improvements

Agenda

- Comparing device trees
 - Device Tree source location
 - Tools to focus investigation of boot time issues
 - Creating the correct kernel configuration for a Device Tree
 - Investigate issues related to properties
-

(1) Comparing Device Trees

Top level source file typically includes other source files, which can in turn include more files.

During compilation, the last seen instance of a definition replaces any previous definition.

The contents of the Device Tree can be changed by many actors, including compiler, boot image creation, boot loader, kernel Device Tree framework, other kernel code, and overlays.

DT data life cycle

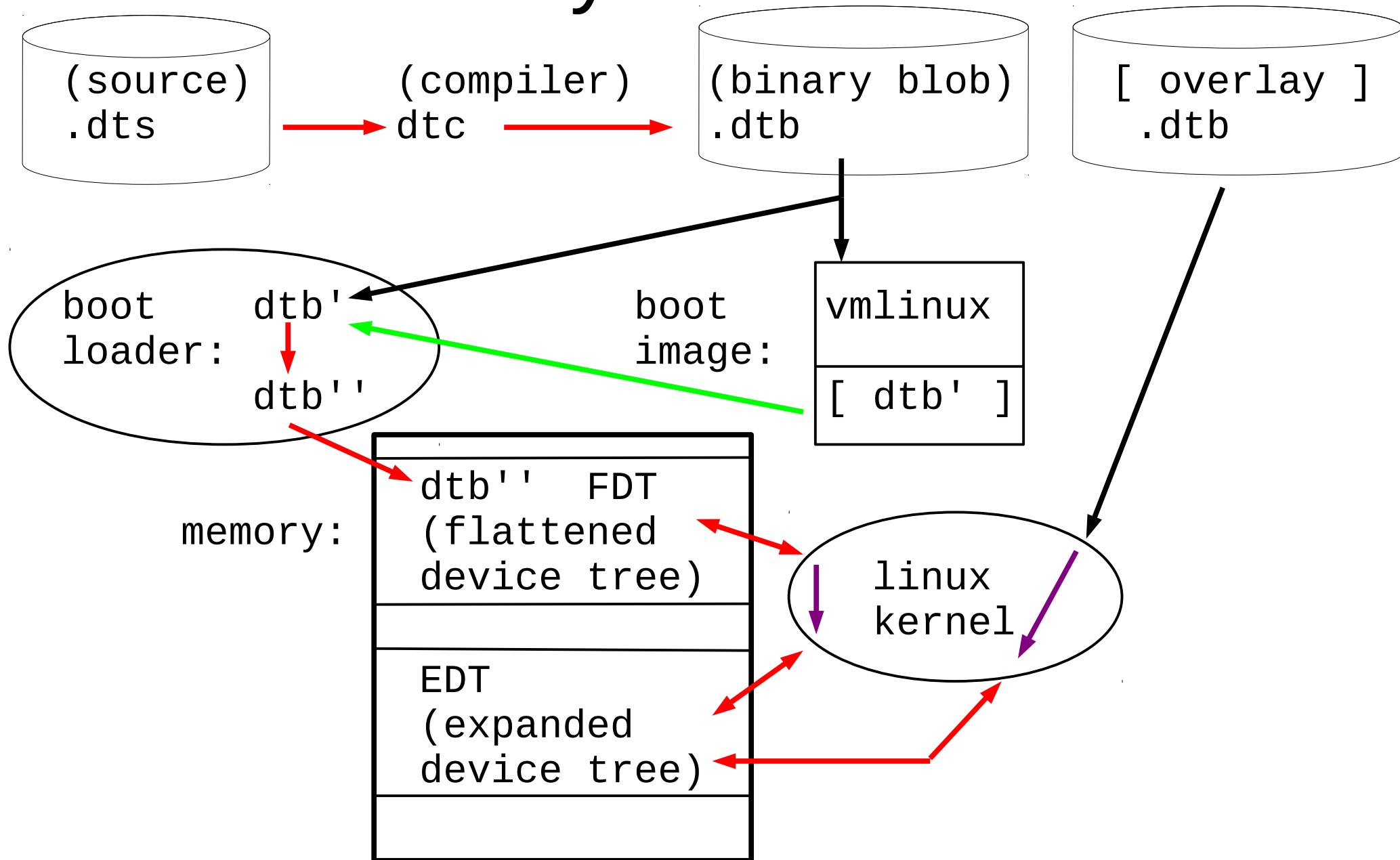
device tree source

.dts file
.dtsi files
.h files



device tree
on booted
target

DT data life cycle



scripts/dtc/dtx_diff

dtx_diff compares device trees in various formats

- source (.dts and the .dtsi includes)
- dtb (binary blob)
- file system tree

merged in Linux 4.6-rc1

extensive examples and use shown in:

Solving Device Tree Issues

elce 2015

example - two source files

```
$ cat v1/test.dts
```

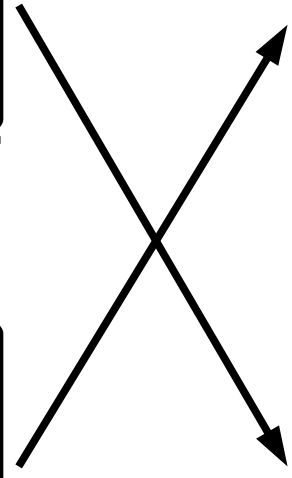
```
/dts-v1/;
```

```
/ {  
    model = "model_1";  
    compatible = "test";  
    c {  
        model = "model_c";  
    };  
};  
/ {  
    model = "model_3";  
    compatible = "test";  
    a {  
        model = "model_a";  
    };  
};
```

```
$ cat v2/test.dts
```

```
/dts-v1/;
```

```
/ {  
    model = "model_3";  
    compatible = "test";  
    a {  
        model = "model_a";  
    };  
};  
/ {  
    model = "model_1";  
    compatible = "test";  
    c {  
        model = "model_c";  
    };  
};
```



```
$ diff -u v1/test.dts v2/test.dts
```

```
@@ -1,19 +1,19 @@  
/dts-v1/;
```

```
 / {  
- model = "model_1";  
+ model = "model_3";  
  compatible = "test";  
  
- c {  
-     model = "model_c";  
+ a {  
+     model = "model_a";  
  };  
};  
  
 / {  
- model = "model_3";  
+ model = "model_1";  
  compatible = "test";  
  
- a {  
-     model = "model_a";  
+ c {  
+     model = "model_c";  
  };  
};
```

```
$ dtx_diff v1/test.dts v2/test.dts
```

```
@@ -2,7 +2,7 @@
```

```
 / {  
     compatible = "test";  
-     model = "model_3";  
+     model = "model_1";  
  
     a {  
         model = "model_a";
```

scripts/dtc/dtx_diff

Process one source device tree

- pre-process include file directives and create resulting source (that is, converts .dts files and included .dtsi files into a single .dts)

(2) Source Location

Top level source file typically includes other source files, which can in turn include more files

During compilation, the last seen instance of a definition replaces any previous definition.

Where did the source for a specific line of the final device tree come from?

dtc -O dts --annotate

```
pm8941_coincell: qcom,coincell@2800 { /* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:70 */
    compatible = "qcom,pm8941-coincell"; /* arch/arm/boot/dts/qcom-pm8941.dtsi:14 */
    reg = <0x2800>; /* arch/arm/boot/dts/qcom-pm8941.dtsi:15 */
    status = "ok"; /* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:71 */
    qcom,rset-ohms = <0x834>; /* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:72 */
    qcom,vset-millivolts = <0xbb8>; /* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:73 */
    qcom,charge-enable; /* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:74 */
}; /* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:75 */
```

dtc -O dts --annotate

```
pm8941_coincell: qcom,coincell@2800 {  
    compatible = "qcom,pm8941-coincell";  
    reg = <0x2800>;  
    status = "ok";  
    qcom,rset-ohms = <0x834>;  
    qcom,vset-millivolts = <0xbb8>;  
    qcom,charge-enable;  
};
```

dtc -O dts --annotate

```
/* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:70 */  
/* arch/arm/boot/dts/qcom-pm8941.dtsi:14 */  
/* arch/arm/boot/dts/qcom-pm8941.dtsi:15 */  
/* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:71 */  
/* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:72 */  
/* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:73 */  
/* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:74 */  
/* arch/arm/boot/dts/qcom-apq8074-dragonboard.dts:75 */
```


(3) Tools to focus investigation of boot time issues

Example - driver not bound

```
$ dt_node_info coincell
```

```
==== devices
```

```
/sys/devices/platform/soc/fc4cf000.spmi/spmi-0/0-00/
```

```
==== nodes
```

```
/soc/spmi@fc4cf000/pm8941@0/qcom,coincell@2800 qcom,
```

```
==== nodes bound to a driver
```

```
==== nodes with a device
```

```
/soc/spmi@fc4cf000/pm8941@0/qcom,coincell@2800 qcom,
```

```
==== nodes not bound to a driver
```

```
/soc/spmi@fc4cf000/pm8941@0/qcom,coincell@2800 qcom,
```

```
==== nodes without a device
```

(2), (3) Remain proof of concept

(2) dtc --annotate

(3) dt_node_info

(3) dt_stat

Available from elinux.org:

http://elinux.org/Device_Tree_frowand

“Embedded Linux Conference Europe” section

extensive examples and use shown in:

Solving Device Tree Issues

elce 2015

(4) Linux kernel configuration for a Device Tree

What kernel configuration options need to be selected for a given Device Tree?

scripts/dtc/dt_to_config

Input: device tree

Input: kernel config (optional)

- Find drivers matching node compatible string(s)
- Find kernel config option(s) to build driver
- Check whether kernel config option is set properly
- Generate kernel config file fragments

merged in Linux 4.8-rc1

examples and use shown in

Solving Device Tree Issues - part 2
LinuxCon Japan 2016

Kernel Configuration Info

```
dt_to_config \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts \
  --short-name \
  --config ${KBUILD_OUTPUT}/.config \
  --config-format \
  > dragon_config_info
```

```
$ grep -i coincell dragon_config_info
```

```
# -d-c-----n--F : coincell@2800 : qcom,pm8941-coincell : drivers/misc/qcom-coincell.c : CONFIG_QCOM_COINCELL : n
# CONFIG_QCOM_COINCELL is not set
# CONFIG_QCOM_COINCELL=y
```

```
# -d-c-----n--F : coincell@2800 : qcom,pm8941-coincell : .....
# CONFIG_QCOM_COINCELL is not set
# CONFIG_QCOM_COINCELL=y
```

(5) Investigate issues related to properties

Properties

```
pm8941_coince11: qcom,coince11@2800 {  
    compatible = "qcom,pm8941-coince11";  
    reg = <0x2800>;  
    status = "disabled";  
    qcom,rset-ohms = <0x834>;  
    qcom,vset-millivolts = <0xbb8>;  
    qcom,charge-enable;  
};
```


Kernel Code Perspective

Is the kernel code correct?

Did my driver or a framework

- read a property value
- attempt to read a property value that did not exist
- not attempt to read a property that exists in the device tree

Device Tree Source Perspective

Is the Device Tree source correct?

Does the device tree source contain

- the necessary properties
- properties that should not be present
- properties that are not used by the kernel

scripts/dtc/dt_prop

Compare properties accessed on target system against a device tree (DTx)

prototype in final test, RFC “soon”

patches and programs available on elinux.org

examples and use shown in

Solving Device Tree Issues - part 3
elce 2016

Simple example

```
$ dt_prop --node-match coincell \
  --all-prop \
  console \
  arch/arm/boot/dts/qcom-apq8074-dragonboard.dts

# --- console
# +++ arch/arm/boot/dts/qcom-apq8074-dragonboard.dts
coincell@2800 {
    compatible = <>;
+   qcom,charge-enable;
+   qcom,rset-ohms = <>;
+   qcom,vset-millivolts = <>;
+   reg = <>;
    status = <>;
};
```

THE END

Thank you for your attention...

Questions?

Resources

http://elinux.org/Device_Tree_presentations_papers_articles

http://elinux.org/Device_Tree_Reference

devicetree: Kernel Internals and Practical Troubleshooting

Frank Rowand, ELCE 2014

http://elinux.org/ELC_Europe_2014_Presentations

Resources

dtx_diff

dtc --annotate

dt_node_info

Solving Device Tree Issues:

Frank Rowand, elce 2015

http://elinux.org/images/0/04/Dt_debugging_elce_2015_151006_0421.pdf

(In this presentation, dtx_diff was named dtdiff.)

Supporting material for: Solving Device Tree Issues:

http://elinux.org/Device_Tree_frowand

section: Embedded Linux Conference Europe (ELCE) - October 6, 2015

dt_to_config

Solving Device Tree Issues - Part 2:

Frank Rowand, LinuxCon Japan 2016

http://elinux.org/images/5/50/Dt_debugging_part_2.pdf

Resources

dt_prop

Solving Device Tree Issues - Part 3:

Frank Rowand, elce 2016

http://elinux.org/images/e/e5/Dt_debugging_part_3.pdf

Supporting material for: Solving Device Tree Issues - Part 3:

kernel patches

scripts/dtc/dts_diff

scripts/dtc/dt_prop

http://elinux.org/Device_Tree_frowand

section: Resources for "Solving Device Tree Issues - Part 3" talk