Devicetree BOF

Open Source Summit Japan 2018
Tokyo

Frank Rowand, Sony

My Goal

Do NOT show all of the slides

Agenda

- Past events

- Future events

- New since elc 2017 (February 2017)

- Tools status

- dtc compiler

- questions, comments, issues, concerns from the crowd

- commit statistics

Plumbers 2017 Summary

September 2017
Los Angeles

Was not scheduled -- not enough interest / commitment

Devicetree Workshop Oct 2017

Prague, Czech Republic

https://elinux.org/Device_tree_future
  #Kernel_Summit_2017.2C_Devicetree_Workshop

- slides
- notes

=== Validation Tools & Schema

=== Runtime usage

=== DTS maintenance issues

=== More stuff

Devicetree Workshop Oct 2017

Prague, Czech Republic

Devicetree Workshop 2017

```
 9:30 Welcome and Schedule bashing
 9:40 Encoding and Schema checking: Framing the problem
 9:45 DT YAML encoding overview
10:00 YAML encoding discussion
10:20 DT Schema format - option 1
10:35 DT Schema format - option 2
10:50 DT Schema discussion - what should go in the spec?
11:50 Code Generation from DT
12:10 Runtime memory consumption
14:30 Overlay maintenance plan
14:45 Avoiding duplicate descriptions
15:00 Criteria for accepting board files
15:15 Location for maintaining bindings - how to handle
         foreign bindings
15:30 Sharing Generic bindings
15:45 ABI Stability
16:00 [break and overflow discussion]
16:30 DT health check
16:50 devicetree.org update
17:05 EBBR Discussion
17:20 Closing and feedback
```

Plumbers 2018

November 13 - 15, 2018
Tuesday - Thursday
Vancouver, British Columbia, Canada
https://www.linuxplumbersconf.org/event/2/overview

colocated with the Linux Kernel Summit

Devicetree track proposal submitted

Email me any topics you want added
to the schedule (or talk to me here)

(Linaro) EBBR Specification

Embedded **B**ase **B**oot **R**equirements

Document basic requirements on firmware to implement a standard boot path on embedded boards

Expect primary users to be embedded & developer Arm boards using U-Boot firmware and Devicetree machine description

Details:

https://lists.linaro.org/pipermail/boot-architecture/2018-April/000416.html

(most of this email shown after the next slide)

(Linaro) EBBR Specification

Mail List

subscribe:
https://lists.linaro.org/mailman/listinfo/boot-architecture

archive:
https://lists.linaro.org/pipermail/boot-architecture/
(EBBR emails start April 2018)

Repository

https://github.com/ARM-software/ebbr

Linaro Connect Hong Kong 2018

Attendees:
  Alexander Graf (SuSE)
  Grant Likely (Arm)
  Bill Mills (TI)
  Peter Robinson (Red Hat/Fedora)
  Dong Wei (Arm)
  Yang Zhang (Linaro/96Boards)

Notes

- We discussed the purpose & intent of EBBR
  - Is intended to document the basic requirements on firmware to
    implement a 'standard' boot path on embedded boards.
  - Needed by distros (Fedora, SuSE, Debian, etc) to support boards out
    of the box
  - Needed by OpenEmbedded, Yocto, etc to get away from custom platform
    specific hacks
  - Establishes a foundation for implementing SecureBoot, A/B updates,
    and other useful boot scenarios in a consistent way.
  - We expect the primary users of EBBR will be embedded & developer Arm
    boards using U-Boot firmware and Devicetree machine description
  - We expect a distribution will be able to use the same software
    (Distro Installer, Grub, Linux UEFI stub, Shim), and the same media
    (installer images) on both embedded and server platforms

- We discussed what EBBR should contain
  - Will document interfaces and standards; not specific projects
  - Will specify a subset of the UEFI specification.
    - Boot services are in
    - Runtime services can be implemented with empty stubs
    - Need to work out what to do with runtime setting of variables
  - For the first release ("EBBR level 0"), it will track features
    available in upstream
    - In concrete terms this means EBBR can be implemented with upstream
      U-Boot or Tianocore.
  - Subsequent releases will refine the requirements as needed and as
    software improves

- Expected target audience
  - embedded board vendors - Gives strong guidance on how to make a
    widely supported board
  - Linux distributions - Can make EBBR compliance a requirement for
    support
  - End users - EBBR will make it simpler to use embedded Arm boards
    because each board will not require special setup instructions or
    image formats

- Roadmap
  - 96Boards wants to specify EBBR compliance in an upcoming spec to be
    announced at Linaro Connect in the fall (about 6 months time)
  - Need to have general agreement on the content of EBBR well before
    that (2-3 months?)
  - Need to have a final EBBR 1.0 release before the 96Boards spec
    announcement
  - Work items:
    - Transcode existing EBBR draft into text markup and check into Git
      repo
    - Review current EBBR draft and compare with available U-Boot
      functionality
      - Identify changes required to EBBR spec
      - Identify gaps in U-Boot functionality that can reasonably be
        addressed in the EBBR v1.0 timeframe
    - Draft roadmap of goals - particularly focusing on functionality
      required by Linux distributions
    - Stand up issue tracker (GitHub?)

Open Questions:

- Can the EBBR document be drafted in public? (Dong to follow up
  internally at Arm)
- Where do the Engineering resources come from to make EBBR a reality
  - General call for engineering effort to be committed by interested
    parties
- Can we use a cut-down LuvOS or UEFI SCT as a specification conformance
  test suite?

Actions:

- Dong to have Arm internal discussion about moving EBBR draft process
  onto GitHub or similar
  - Markup candidate: Sphinx-doc with reStructuredText markup
- Grant to organize a regular weekly meeting to track EBBR drafting
  process
  - Make sure to include Tom Rini and Ard Biesheuvel
- Yang to socialize with 96Boards partners to prepare them for EBBR
  compliance
- (Unassigned) Create a hosting page with issue tracker for EBBR -- TBD
  after Dong finishes internal due diligence on moving EBBR drafting to
  a public repository
  - Probably GitHub

Devicetree Specification

Devicetree Specification 0.1 supersedes ePAPR for the Linux kernel, continues to evolve

https://www.devicetree.org/specifications/

Mail list, Build Instructions, etc

https://www.devicetree.org/collaborate/

Repository

https://github.com/devicetree-org/devicetree-specification

Tools

Kernel Configuration Info -- OLD

In tree

```
scripts/dtc/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

Debug Tools - OLD - update

dtc: dts source location annotation
  - Provide source locations from .dts & .dtsi
  - Several proof of concept versions on
     devicetree-compiler list, up to October 2015
  - ~~Stalled, awaiting some of Frank's bandwidth~~

Project picked up by Julia Lawall

```
To:      devicetree-compiler@vger.kernel.org
Subject: [PATCH 1/3 v4] annotations: check for NULL position
Date:    Fri,  2 Feb 2018 21:41:48 +0100
```

source location annotation (old)

```
----- short format  -----

sdhci@f9824900 { /* qcom-apq8074-dragonboard.dts:14 */
      compatible = "qcom,sdhci-msm-v4"; /* qcom-msm8974.dtsi:240 */
      reg = <0xf9824900 0x11c 0xf9824000 0x800>; /* qcom-msm8974.dtsi:241 */
      reg-names = "hc_mem", "core_mem"; /* qcom-msm8974.dtsi:242 */
      interrupts = <0x0 0x7b 0x0 0x0 0x8a 0x0>; /* qcom-msm8974.dtsi:243 */
      interrupt-names = "hc_irq", "pwr_irq"; /* qcom-msm8974.dtsi:244 */
      clocks = <0xd 0xd8 0xd 0xd7>; /* qcom-msm8974.dtsi:245 */
      clock-names = "core", "iface"; /* qcom-msm8974.dtsi:246 */
      status = "ok"; /* qcom-apq8074-dragonboard.dts:17 */
      bus-width = <0x8>; /* qcom-apq8074-dragonboard.dts:15 */
      non-removable; /* qcom-apq8074-dragonboard.dts:16 */
}; /* qcom-apq8074-dragonboard.dts:18 */
```

Debug Tools -- semi-OLD

scripts/dtc/dt_prop
- Compare properties accessed on target system vs a device tree (dtX)
- available on elinux.org
- ~~Plan to submit to mail list "any day now"~~

==> Stalled, awaiting some of Frank's bandwidth

dt_prop example snippets

```
$ dt_prop --td dmesg_4.5-rc5_160307_2100 qcom-apq8074-dragonboard.dts

# --- dmesg_4.5-rc5_160307_2100
# +++ qcom-apq8074-dragonboard.dts
 /dts-v1/;


                // *****   i2c@f9964000 disabled  *****
                i2c@f9964000 {
+                       #address-cells;
+                       #size-cells;
+                       clock-names;
+                       clocks;
                        compatible;
+                       interrupts;
+                       reg;
                        status;
                };
```

Debug Tools - OLD

dt_node_info, dt_stat

- Aids boot (or module load) debugging

- Status of device creation, devicetree nodes, driver discovery, driver binding

- proof of concept on elinux.org

- Stalled, awaiting some of Frank's bandwidth

dt_node_info example 1

```
$ dt_node_info coincell
===== devices

===== nodes
/soc/spmi@fc4cf000/pm8941@0/qcom,coincell@2800 qcom,

===== nodes bound to a driver

===== nodes with a device

===== nodes not bound to a driver
/soc/spmi@fc4cf000/pm8941@0/qcom,coincell@2800 qcom,

===== nodes without a device
/soc/spmi@fc4cf000/pm8941@0/qcom,coincell@2800 qcom,
```

dt_node_info example 2

```
$ 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
```

dtc Compiler

dtc - Devicetree Build Checks

Rob has been enhancing dtc error checks

Enabled for "W=1" builds

$ make V=0 W=1 qcom-apq8074-dragonboard.dtb

```
make[1]: Entering directory `/local/frowand_nobackup/src/git_linus/build/dragon_linus_4.10'
  DTC     arch/arm/boot/dts/qcom-apq8074-dragonboard.dtb
Warning (unit_address_vs_reg): Node /memory has a reg or ranges property, but no unit name
Warning (unit_address_vs_reg): Node /soc/spmi@fc4cf000/pm8941@0/vadc@3100/die_temp
Warning (unit_address_vs_reg): Node /soc/spmi@fc4cf000/pm8941@0/vadc@3100/ref_625m
Warning (unit_address_vs_reg): Node /soc/spmi@fc4cf000/pm8941@0/vadc@3100/ref_1250v
Warning (unit_address_vs_reg): Node /soc/spmi@fc4cf000/pm8941@0/vadc@3100/ref_gnd ha
Warning (unit_address_vs_reg): Node /soc/spmi@fc4cf000/pm8941@0/vadc@3100/ref_vdd ha
make[1]: Leaving directory `/local/frowand_nobackup/src/git_linus/build/dragon_linus_4.10'
```

dtc - Devicetree Build Checks

commits from February 10, 2017 to date in dtc repo

add graph binding checks
add a check for duplicate unit-addresses of child nodes
add chosen node checks
add aliases node checks
check for #{size,address}-cells without child nodes
add string list check for *-names properties
add string list check
add a string check for 'label' property
add interrupts property check
add gpio binding properties check
add phandle with arg property checks
Warn on node name unit-addresses with '0x' or leading 0s
Add bus checks for simple-bus buses
Add bus checks for PCI buses
Add Warning for stricter node name character checking
Add Warning for stricter property name character checking

dtc - overlays - Linux v4.15

dtc creates the .dtb **OVERLAY INTERNAL DATA ("metadata")**

Do not hand code overlay internal data nodes in DTS source:

```
fragment@
__overlay__
__fixup__
__local_fixup__
__symbols__
```

Metadata

How should the metadata required by overlays be encoded in the FDT?

Discussion in progress on devicetree-compiler list

Subject: [RFC] devicetree: new FDT format version
Message-ID: <b96829f9-2e8b-fdc5-5090-58591e2260cf@gmail.com>
Date: Mon, 22 Jan 2018 00:09:18 -0800

Metadata - base FDT overhead

Metadata overhead measured for arch/arm/boot/dts/*

row	dtb no symbols	delta symbols	delta new fmt	bytes saved
99%	90531	42721	15766	26955
83%	44302	14582	5163	9419
66%	26277	11662	4628	7034
49%	21047	7328	2754	4574
33%	12864	4305	1705	2600
16%	12009	2929	1520	1409
0%	1220	68	149	-81

- "**delta symbols**" is added size from '**dtc -@**'
- "**new fmt**" is added size from '**dtc -@**' **for first proposal in the thread**

dtc - overlays - example - old.dts

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

/ {
        fragment@0 {
                target-path = "/soc/base_fpga_region";
                #address-cells = <1>;
                #size-cells = <1>;

                __overlay__ {
                        ranges = <0x00000000 0x00000000 0xc0000000 0x00040000>,
                                 <0x00000001 0x00000000 0xff200000 0x00001000>;

                        external-fpga-config;

                        #address-cells = <2>;
                        #size-cells = <1>;

                        fpga_pr_region0 {
                                compatible = "fpga-region";
                                fpga-bridges = <&freeze_controller_0>;
                                ranges;
                        };

                        freeze_controller_0: freeze_controller@100000450 {
                                compatible = "altr,freeze-bridge-controller";
                                reg = <0x00000001 0x00000450 0x00000010>;
                                interrupt-parent = <&intc>;
                                interrupts = <0 21 4>;
                        };
                };
        };
};
```

dtc - overlays - example - new.dts

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

&fpga_region {
        ranges = <0x00000000 0x00000000 0xc0000000 0x00040000>,
                 <0x00000001 0x00000000 0xff200000 0x00001000>;

        external-fpga-config;

        #address-cells = <2>;
        #size-cells = <1>;

        fpga_pr_region0 {
                compatible = "fpga-region";
                fpga-bridges = <&freeze_controller_0>;
                ranges;
        };

        freeze_controller_0: freeze_controller@100000450 {
                compatible = "altr,freeze-bridge-controller";
                reg = <0x00000001 0x00000450 0x00000010>;
                interrupt-parent = <&intc>;
                interrupts = <0 21 4>;
        };
};
```

dtc - overlays - example

```
$ diff -b -u old.dts new.dts
--- old.dts
+++ new.dts
@@ -1,13 +1,7 @@
 /dts-v1/;
 /plugin/;

-/ {
-        fragment@0 {
-                target-path = "/soc/base_fpga_region";
-                #address-cells = <1>;
-                #size-cells = <1>;
-
-                __overlay__ {
+&fpga_region {
                        ranges = <0x00000000 0x00000000 0xc0000000 0x00040000>,
                                 <0x00000001 0x00000000 0xff200000 0x00001000>;

@@ -28,6 +22,4 @@

                                interrupt-parent = <&intc>;
                                interrupts = <0 21 4>;
                        };
-                };
-        };
 };
```

dtc - overlays - new.dts - no label?

(1) What if there is no label for the
    overlay target in the base devicetree?

(2) What if the overlay target is the root
    node (dtc does not allow a label on
    the root node)?

Support is in upstream dtc project, but not yet
imported into Linux as of v4.17

dtc - overlays - no label? (1)

```
$ cat path_sugar_abs_path_multi_node.dts

/dts-v1/;
/plugin/;
&{/node_level_1/node_level_2/} {
          #address-cells = <2>;
          #size-cells = <2>;

          my_node@feb90000 {
               compatible = "vendor,device";
               reg = <0 0xfeb90000 0 0x1c>;

          };

};

$ dtc -O dts path_sugar_abs_path_multi_node.dts
/dts-v1/;

/ {

     fragment@0 {
          target-path = "/node_level_1/node_level_2/";

          __overlay__ {
               #address-cells = <0x2>;
               #size-cells = <0x2>;

               my_node@feb90000 {
                    compatible = "vendor,device";
                    reg = <0x0 0xfeb90000 0x0 0x1c>;
               };
          };
     };
};
```

dtc - overlays - no label? (2)

```
$ cat path_sugar_abs_path_root.dts

/dts-v1/;
/plugin/;
&{/} {
        #address-cells = <2>;
        #size-cells = <2>;

        my_node@feb90000 {
            compatible = "vendor,device";
            reg = <0 0xfeb90000 0 0x1c>;

        };

};/dts-v1/;

$ dtc -O dts path_sugar_abs_path_root.dts
/dts-v1/;

/ {

    fragment@0 {
        target-path = [2f 00];

        __overlay__ {
            #address-cells = <0x2>;
            #size-cells = <0x2>;

            my_node@feb90000 {
                compatible = "vendor,device";
                reg = <0x0 0xfeb90000 0x0 0x1c>;
            };
        };
    };
};
```

.dtsi source vs overlay .dtsi

(More slides available after the 'END' slide)

What do you want to talk about?

questions

comments

issues

concerns

Devicetree Development History

Some random statistics

v4.2.. arch/*/boot/dts/ commits

```
  550  arc
54641  arm
13170  arm64
    4  c6x
   44  cris
   93  h8300
   33  metag
    2  microblaze
 1054  mips
   31  nios2
    7  openrisc
  797  powerpc
   25  sh
  115  xtensa
```

arch/*/boot/dts/ commits

v4.2   638
v4.3   592
v4.4   666
v4.5   725
v4.6   682
v4.7   722
v4.8   674
v4.9   719   2017 02 19
v4.10   768   2017 04 30
v4.11   632
v4.12   658
v4.13   687
v4.14   725
v4.15   762

arch/*/boot/dts/ commits

	arm	arm64			
v4.2	568	36			
v4.3	535	46			
v4.4	553	67			
v4.5	571	137			
v4.6	511	149			
v4.7	559	117			
v4.8	498	156			
v4.9	489	195			
v4.10	519	235	2017	02	19
v4.11	456	151	2017	04	30
v4.12	434	217			
v4.13	381	283			
v4.14	401	294			
v4.15	503	219			

.../bindings/ commits

v4.2	488	
v4.3	411	
v4.4	418	
v4.5	467	
v4.6	428	
v4.7	445	
v4.8	418	
v4.9	453	
v4.10	491	2017 02 19
v4.11	421	2017 04 30
v4.12	468	
v4.13	460	
v4.14	494	
v4.15	438	

scripts/dtc/ commits

v4.2	2	
v4.3	0	
v4.4	0	
v4.5	0	
v4.6	3	
v4.7	1	
v4.8	1	
v4.9	0	
v4.10	0	2017 02 19
v4.11	1	2017 04 30
v4.12	6	
v4.13	6	
v4.14	2	
v4.15	4	

drivers/of/ commits

```
v4.2        29
v4.3        16
v4.4        38
v4.5        22
v4.6        29
v4.7        40
v4.8        30
v4.9        24
v4.10       28       2017 02 19
v4.11       19       2017 04 30
v4.12       32
v4.13       36
v4.14       38
v4.15       52
```

v4.10.. top drivers/of/ commits

(February 19, 2017 ..)

What have patch topics been?

Very imprecise topic count (do not take the list too seriously...)

Based mostly on leading fields of patch subject

v4.10.. top drivers/of/ commits

(loosely based by patch comment tag)

```
23 overlay
10 device property
12 test
 7 of_mdio
 7 irq
 6 fdt
 6 of_graph
 6 device
 5 pci
 4 platform
 3 base
```

THE END

Thank you for your participation...

Questions?

Comments?

.dtsi source vs overlay .dtsi

With the new dtc --

Overlay .dts file contains directives:
/dts-v1/;
/plugin/;

.dtsi include file does not

Use include as .dtsi or overlay

With sugar syntax, the syntax used by an overlay is now compatible with the syntax used by an include file, if the include file uses labels as paths instead of using explicit paths.

- This may be convenient for development workflows

- Do not become dependent on this for overlays that will be long lived -- current thinking is that we want many / most overlays to use the connector model

Use include as .dtsi or overlay

```
---------- base tree -------------------------------

$ expand fpga_tree.dts
/dts-v1/;

/* labels used by overlay are in the base tree */

/ {
        soc {
                intc: interrupt_ctrl {
                };
                fpga_region: base_fpga_region {
                };
        };
};

/include/ "fpga_plugin_or_dtsi.dts"

---------- overlay --------------------------------

$ expand fpga_overlay.dts
/dts-v1/;
/plugin/;

/include/ "fpga_plugin_or_dtsi.dts"
```

The .dtsi

```
$ expand fpga_plugin_or_dtsi.dts
&fpga_region {
        ranges = <0x00000000 0x00000000 0xc0000000 0x00040000>,
                 <0x00000001 0x00000000 0xff200000 0x00001000>;

        external-fpga-config;

        #address-cells = <2>;
        #size-cells = <1>;

        fpga_pr_region0 {
                compatible = "fpga-region";
                fpga-bridges = <&freeze_controller_0>;
                ranges;
        };

        freeze_controller_0: freeze_controller@100000450 {
                compatible = "altr,freeze-bridge-controller";
                reg = <0x00000001 0x00000450 0x00000010>;
                interrupt-parent = <&intc>;
                interrupts = <0 21 4>;
        };
};
```

Overlays -- more stuff...

Overlays

- U-Boot overlay support

   - enhancements have been added

Overlays - some use cases

- Expansion slots / external connectors
    - beaglebone
    - raspberry pi
    - minnowboard
    - C.H.I.P.
    - Arduino
    - seeedstudios Grove 4 pin connectors
    - others?

- FPGA

Overlays - some use cases

- Combinatorial explosion of .dts / .dtb files

    example:
      Devicetree Hardware Autoconfiguration
      Hans de Goede
      ELC Europe 2016

Overlays, one of the gating factors

```
On 10/18/17 14:46, Frank Rowand wrote:

> On Wed, 2017-10-18 at 10:44 -0500, Rob Herring wrote:

>> The issue remains that the kernel is not really setup to deal with any
>> random property or node to be changed at any point in run-time. I
>> think there needs to be some restrictions around what the overlays can
>> touch. We can't have it be wide open and then lock things down later
>> and break users.

> That paragraph is key to any discussion of accepting code to apply overlays.
> Solving that issue has been stated to be a gating factor for such code from
> the beginning of overlay development.
```

Not the only remaining issue -- see "Frank's Evolving Overlay Thoughts":

https://elinux.org/Frank%27s_Evolving_Overlay_Thoughts

Overlays, Linux kernel progress

(Not a complete list)

- overlay.c refactored
- resolver.c refactored
- hand coded overlays in devicetree source files
    - resolved by sugar syntax if base tree has required labels
    - continued dtc enhancements underway
- papered over issue: free FDT or expanded devicetree while pointers
    into them still exist
- unable to free overlay FDT and overlay expanded device tree after
    overlay removal (memory leak)
- FDT format update up in the air
- of locking architecturally broken
- pre-removal checks needs to ensure relevent driver(s) unbind
- connectors architecture up in the air
- overlay semantics not fully specified
- overlay manager (do not accept until gating requirements are resolved)

Resources

http://elinux.org/Device_Tree_presentations_papers_articles
http://elinux.org/Device_Tree_presentations_papers_articles#debug

http://elinux.org/Device_Tree_Reference

Resources
Devicetree Documentation

elinux.org/Device_Tree_Reference

- becoming more complete
- contributions and comments welcome

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

How to get a copy of the slides

1) frank.rowand@sony.com

2) http://elinux.org/Device_Tree

3) http://events.linuxfoundation.org