Devicetree MicroConference, LPC 2025 - notes

13th December 2025, Tokyo
Written by: Krzysztof Kozlowski
Video stream:https://www.youtube.com/live/2Gput2daq4c?si=o0AuWA7_ayTgpg1u&t=19448

Status of DTS validation in the Linux kernel (and how to improve it) - Krzysztof Kozlowski

1. Aspeed and Loongarch got already some fixes after OSS Japan speech
2. Konrad Dybcio needs more involvement from testing on hardware
3. Discussed briefly how DT bindings patches could be applied to help avoiding warnings in SoC trees where the DTS is applied:
    a. Krzysztof Kozlowski said DT maintainers won't take bindings to separate branch
        i. Note from Rob Herring:
        We could consider this if it helps. It's not really more work given we review/ack everything. My main concerns are if the driver and/or dts changes are abandoned or the binding changes further (not too uncommon after a R-by).
    b. Bjorn Andersson said he **builds process schema from the linux-next and then validates his SoC DTS branch against that process schema**.
    c. Bjorn Andersson mentioned that sometimes DT bindings fix is not being picked up. Krzysztof Kozlowski suggested that **SoC platform maintainer can always apply such abandoned, lonely DT binding**.

Improving fw_devlink dependency tracking - Saravana Kannan

1. Bjorn Andersson raised question if devlinks are considered optimization or a functional requirement to ensure we probe in proper order (fundamental data structure keeping dependencies)
    a. How correct do the devlinks need to be for asynchronous probing?
    b. Krzysztof Kozlowski suggested that devlinks are already functional requirement due to handling unbinding order for producer/consumers
2. Samuel Holland mentioned that existing parsing code for specific phandle properties will have to be there anyway to support existing DTS and bootloader-added properties
3. Krzysztof Kozlowski suggested that **dts-v2 could enforce having these additional discussed annotations (/plugin/ or __references__)** and Bjorn Andersson confirmed that **async-probing would have to depend on dts-v2**.

"Exclusive" resources that are not exclusive - the case of shared GPIOs in DT systems - Bartosz Golaszewski

1. Dmitry Torokhov suggested that a shared reset GPIO should be its own abstraction, because it is not a normal GPIO anymore - drivers might still expect exclusivity (e.g. reset when asserting).
    a. Bartosz Golaszewski replied that such abstractions are already there partially, e.g. regulators or reset.
2. Bjorn Andersson said that complicated shared-GPIO cases, like timing or reset sequence, need either different HW design (separate GPIOs) or a combined driver state.
3. Geert Uytterhoeven described a complicated case on a few Renesas development boards where a GPIO is connected to both button and LED. If GPIO is in output, then you can control LED. If it is in input, you can read the button state. Basically with shared GPIO proxy from Bartosz, this GPIO could be used only as LED or switch/button. Laurent Pinchart described his case of two camera sensors on the same camera bus with the same reset GPIO but having an inverter on one of the resets. This was followed by audience laughter expressing judgment on creative hardware designs.
    a. Geert Uytterhoeven later came with idea to solve his **button-LED case with an intermediate driver "wire-gpio"** (wired-OR, wired-AND) used for sharing one GPIO connected to multiple devices.
4. Wolfram Sang reminded issue from last weeks around need of enabling RESET_GPIO to have core functionality (why this is not in the core). Krzysztof Kozlowski responded that it's Philip's (reset maintainer) call to have it in the core or not and Krzysztof's earlier objection was to introduced breakage for existing platforms.
5. Srinivas Kandagatla asked if this functionality should go to subsystems/frameworks instead but Bartosz answered that not all GPIOs have such a framework (e.g. powerdown GPIO).
    a. Post-MC note: powerdown GPIO is just a variation of reset-gpios, thus should use reset-gpio driver.
6. Dmitry Torokhov asked if there is recommendation to convert existing drivers to use of reset-gpio driver for their resets?
    a. Bartosz Golaszewski answered that:
        i. Simple devices could be switched to reset-gpio - **no recommendation but there would be a benefit**
        ii. **Complex devices with reset sequences (delays etc) need more work** and he has an idea which he plans to present for Linux Plumbers Conference 2026

Power Sequencing for Enumerable Busses - What's Still Missing From the Device Tree and Kernel - Chen-Yu Tsai

1. Clocks for PHYs on MDIO:

    a. Marek Vasut mentioned that some of the phys on MDIOs require clocks and their handling is inconsistent among drivers. Chen-Yu Tsai answered that this is not being addressed and he counted five drivers supporting clocks.
2. Bartosz Golaszewski mentioned that it is annoying we cannot use MDIO autodetection because of missing resource handling. **He noted that generic stub driver would be useful in this case.**
    a. Later, Laurent Pinchart admitted lack of generic solution for MDIO was wasting him a lot of time. Also U-Boot and Linux are implemented differently with different DT properties. **Generic solution would help the system integrators and would be really nice**.
3. Srinivas Kandagatla said that Slimbus and Soundwire also include power sequencing. Soundwire solved it by separating driver probing from actual device probing (called “bind” in Soundwire).
4. Bartosz Golaszewski reminded that power sequencing bindings for SDIO are considered deprecated.

DeviceTrees - MIPI SoundWire® Device Class for Audio (SDCA) and classic ACPI-DT problem - Srini Kandagatla

1. Option – 1: Should we add exactly same device tree bindings for SDCA
    a. Krzysztof Kozlowski doubted we want register init sequences in DT.
2. Option – 2: Ignore SDCA and add bindings as required
    a. Krzysztof Kozlowski accepted that solution from DT point of view. Srini Kandagatla said this will mean we do not use generic-drivers.
3. Option – 3: Use Secondary fwnode
    a. Srini Kandagatla clarified that driver would be looking for secondary FW node after probing from DT with basic compatible.
    b. Konrad Dybcio said this is a special case where we have ACPI but asked what to do with platforms not having ACPI? Manivannan Sadhasivam re-iterated the question. Bjorn Andersson said firmware people should not add ACPI tables just to full fill this on DT-only platforms.
    c. Srini Kandagatla reminded that SDCA spec requires description of the device to come from the hardware.
    d. Krzysztof Kozlowski asked if a firmware file (linux-firmware) would fulfil that need.
    e. Konrad Dybcio asked if init sequences differ between boards having the same device. Mark Brown doubted same init sequence would fit multiple boards (vendors will probably want system-specific tuning).
    f. Bjorn Andersson said that we need to be sure if device description is here static between devices. If it is static, it could be put into a dedicated driver. Krzysztof Kozlowski reminded that compatible implies features of the device, thus init sequences would not be suitable to DT.
    g. **Solution would be to have C code (driver) which will provide necessary structures to the generic driver, e.g. via providing software nodes**.

DT formatter - a maintainer's and contributor's elusive dream? - Konrad Dybcio

1. Krzysztof Kozlowski brought the problem of different styles and lack of acceptance for one DTS coding style. His view on the tool was that each SoC platform would have its own config file in DTS sources determining the style.
    a. Krzysztof Kozlowski followed up with a question if maintainer will agree on common DTS style. **Answer from many people was that yes, people want a common style.** (Later though Krzysztof reminded that Nvidia will probably not back off from their full node path override style in DTS).
2. Marek Vasut suggested that merge-resulted breakages could be solved by comparing decompiled DTS before and after merge. Bjorn Andersson said this is not a solution because it would mean the SoC maintainer has one more manual step in his workflow.
3. The audience expressed their wish of auto-correction (or auto-crashing?)
4. Samuel Holland added to the wishlist for the language server to match DTS being checked against bindings.
5. Bartosz Golaszewski wants to query devicetree based on some search query, like find nodes by addresses.
    a. Note from Rob Herring:
    I had AI write a tool to dump out a memory map from the DT. It worked surprisingly well. My goal with it was to find overlapping memory regions. I've also thought about some sort of search/query tool for the schema. There's already a tool that can dump all properties and their type(s). Not sure what all would be helpful there?
6. Tooling currently in development
    a. Marek Vasut reminded Peng Fan's tool. (**<can anyone find it?>**)
    b. Krzysztof Kozlowski said this is written in C, but now he thinks he probably mixed it up with Zhi Li's dt-format: https://github.com/lznuaa/dt-format.git
7. Konrad Dybcio suggested **finalizing the DTS coding style to be much more stricter and not allow per-SoC platform differences, which would then be used as real base for the formatting tool**.

Device Tree Metadata: How Bootloaders Pick DTBs and Apply Overlays - Chen-Yu Tsai

1. Doug Anderson asked what the top compatible is for.
    a. Bjorn Andersson mentioned that it might be useful for reporting the final hardware in bug reports.
    b. Geert Uytterhoeven said that it is for last-resort of fixups, therefore each of SKUs and board-combinations should have their representation in the top–level compatible.
2. Geert Uytterhoeven suggested that DT overlay could insert new top-level compatible value in front of existing top-level compatibles.

3. Marek Vasut proposed new top-level “config” node with name of the DT overlay and then final DTB would have concatenated result. He mentioned there is a patch for this for U-Boot already.