



LINUX PLUMBERS CONFERENCE 2026 · DEVICETREE MC

Streamlining DT Bindings to Protect Upstreaming Momentum

Amit Kucheria, Dr. Daniel Lezcano
Qualcomm Technologies, Inc.

6 Oct 2026, 11:10 · Club A, Prague Congress Centre
lpc.events/event/20/contributions/2470

2026 Linux Plumbers Conference, Prague



Slowing down upstreaming

- Only a problem for **new** bindings
- Bindings are usually a small percentage of the total amount of code for a new type of device
- Driver and DTS changes aren't reviewed until the bindings are accepted
- For early platform upstreaming, months of back and forth can kill upstreaming momentum

Not a lower bar for review

- NOT a way to land unreviewed or unfinished driver code. The driver goes through the same review it always has.
- NOT a way around subsystem or DT maintainer sign-off on the binding itself.
- NOT a change to compatible-string matching or how drivers bind at runtime. The kernel matches devices the same way it does today.
- NOT a permanent state. An experimental binding gets a real, checked lifetime, not an indefinite grace period.
- Proposal is **bookkeeping for bindings that have already been reviewed and accepted, but whose property surface is still being discovered**. It's not a way around review.

PREVIOUS ATTEMPTS AT SOLVING THIS PROBLEM

2013

Thierry Reding

- Mark experimental bindings with a ! prefix on the compatible string, gated by CONFIG_OF_EXPERIMENTAL.
- Rejected on the mailing list and at that year's Kernel Summit — it polluted DTS and driver source, and turned “promote to stable” into a flag day change (removing the marker)
- Led directly to ABI.rst rule II.4: “don't submit bindings for staging or unstable.”

2018

Krzysztof Kozlowski

- Mark a batch of Samsung Exynos bindings “unstable” in the binding doc, no enforcement attached.
- Rob Herring:
 - “The real question is how many have gone from unstable to stable. My guess is 0.”
 - “There's no carrot and I don't want to be the stick.”

PREVIOUS ATTEMPTS AT SOLVING THIS PROBLEM

2013

Thierry Reding

- Mark experimental bindings with a ! prefix on the compatible string, gated by CONFIG_OF_EXPERIMENTAL.
- Rejected on the mailing list and at that year's Kernel Summit — it polluted DTS and driver source, and
- Led

Same objection both times: lack of enforcement mechanisms puts onus on maintainers to nag bindings authors

2018

Krzysztof Kozlowski

- Mark a batch of Samsung Exynos bindings “unstable” in the binding doc, no enforcement attached.
- Rob Herring:
 - “The real question is how many have gone from unstable to stable. My guess is 0.”
 - “There's no carrot and I don't want to be the stick.”

One property, nothing to keep in sync

- **maturity: experimental**
 - one line in the binding schema itself, not the compatible string. Nothing else to write.
 - Mark a binding experimental by adding that line. Promote it to stable by deleting the same line.
- Expiry comes from git history.
 - **dt-maturity-check** finds the commit that added the property, counts how many real releases have shipped since, and fails **dt_binding_check** once that's too many. It points back at that commit, so there's always a trail to the original discussion.
- Add a matching dtschema meta-schema, **experimental.yaml**, that lets a binding skip `additionalProperties` checks while its shape is still settling. It's opt-in and only kicks in once a binding already carries the tag.
- Binding expiry can lead to removal, TBD by maintainers.
- [Linux changes](#)
- [Dtschema changes](#)

DISCUSSION POINTS

- Multi-stage timeout for bindings? WARNING → ERROR?
- What is a good amount of time to allow bindings to mature?
- How will the kernel ABI footprint be handled?
 - Does timeout mean promotion to stable or removal?
 - Does removal of an experimental binding mean ABI regression for users?

For submitters, maintainers, and CI

Submitter

- Add maturity: experimental when you post a binding whose shape isn't final yet.
- Nothing to date, link, or bump. Ever.
- Promoting it later is a one-line deletion patch.

Subsystem / DT maintainer

- Review and ack the binding exactly like you do today. Who signs off doesn't change.
- No tracking issue or deadline to babysit. Git history already has the pointer.
- Once the grace period runs out, the build just breaks. Nobody has to remember to go check.

CI / dt_binding_check

- One more check, but it's grepped first, so it costs about 0.05s on a tree with no experimental bindings.
- An unrecognised maturity value or an expired binding fails the build. It's not a warning.

Thank you

Nothing in these materials is an offer to sell any of the components or devices referenced herein.

© Qualcomm Technologies, Inc. and/or its affiliated companies. All Rights Reserved.

Qualcomm and Snapdragon are trademarks or registered trademarks of Qualcomm Incorporated.
Other products and brand names may be trademarks or registered trademarks of their respective owners.

References in this presentation to “Qualcomm” may mean Qualcomm Incorporated, Qualcomm Technologies, Inc., and/or other subsidiaries or business units within the Qualcomm corporate structure, as applicable. Qualcomm Incorporated includes our licensing business, QTL, and the vast majority of our patent portfolio. Qualcomm Technologies, Inc., a subsidiary of Qualcomm Incorporated, operates, along with its subsidiaries, substantially all of our engineering, research and development functions, and substantially all of our products and services businesses, including our QCT semiconductor business.

Snapdragon and Qualcomm branded products are products of Qualcomm Technologies, Inc. and/or its subsidiaries. Qualcomm patented technologies are licensed by Qualcomm Incorporated.

Follow us on:     

For more information, visit us at [qualcomm.com](https://www.qualcomm.com) & [qualcomm.com/blog](https://www.qualcomm.com/blog)

