# Netdev virtual roundtable - minutes

Updates for next:

 standalone phylink PCS drivers: as part of phylink? or as standalone subsystem? or not at all?

Als:

•

### Oct 2025 (21st)

- PHY port representation
  - No major concerns so far, just general busy-ness
  - Andrew to look after the weekend
  - We'll drop from patchwork given the timelines, Andrew promises to look regardless

### Jul 2025 (29th)

- DPLL
  - Arkadiusz: The standalone DPLL driver from Ivan doesn't have a unique clock ID, which is supposed to be the ID for the whole DPLL devices. The clock ID works for time appliances. The standalone DPLL device doesn't have a network card, so no MAC address, or ID.
  - Ivan: the device can be connected to multiple controllers, and drive multiple clocks. There's no ID on the device, no serial even. We use a random number for now, devlink can be used to change it.
  - Andrew: it's an i2c device? (yes, i2c or SPI) Can we use the address?
  - Ivan: yes, but the i2c address is static, so if there are multiple buses or a mux the address will be the same for all devices in the system.
  - Andrew: we can include the full bus address (with the bus).
  - o Jakub: can we define the relationship to the netdev port in the DT?
  - Ivan: we could extend the DT note which contains the pins to include the MAC phandle.
  - o Andrew: you can use the DSA definition as the sample to steal ideas from.
  - Ivan: the DPLL can drive multiple NICs, which device to pick for the clock ID.
  - Andrew: a diagram will be very helpful; Jakub: prefer using an explicit annotation.
  - o Next step: send an RFC to the DT list?
- phylink PCS drivers: as part of phylink
  - Sean and Russell not present, let's revisit in 2 weeks

### Nov 2024 (5th)

- Netdev foundation:
  - Jesper can companies donate HW
    - Yes, but we need to discuss with LF how/if we can count that in lieu of the membership fee
  - Simon timing of the membership payment
    - Jakub expecting Q1'25
    - Mixed response, for some that's good, for others it's easier to use 2024 budget
    - We'll try to wrap up in 2024 to make both happy
  - o JT External testing reporting
    - Useful / will be supported

## Jul 2024 (2nd)

• No topics, the meeting ended without any discussion.

#### Jun 2024

- Cl update:
  - Real DB is now running, so support for KTAP break out is back
  - Runners scan for kmemleak reports
- SR-IOV ops in drivers
  - New drivers adding ndo\_set\_vf\_\* callbacks were banned for the last 4 or so years. This was expected to result in broader switchdev adoption, which didn't happen. The policy is pushing people towards out-of-tree drivers.
  - Broad support for allowing adding those ops again.
  - There is a problem with the current API supporting a limited number of VFs (100+, which is less than current HW supports). We can try to solve it in rtnetlink or add a similar API based on devlink ports.
  - No strong preferences among participants between extending devlink vs rtnetlink to expose all VFs.

# May 2024

- Cl update
  - Wiki updates
     https://github.com/linux-netdev/nipa/wiki/Setting-up-a-runner-for-a-Supported-NIC
     -driver
  - CI now uses a real DB for storing results

### Feb 2024 (13th)

- Cl update
  - good progress on stability and rooting out flakes
  - remaining flakes:
     https://netdev.bots.linux.dev/flakes.html?br-cnt=84&min-flip=0&pw-y=0
  - Mat reports that breaking out subtests from KTAP is now supported by the runner
  - Aaron reports OvS tests are almost ready for the runner
  - Paolo to reach out to Justin about ioam6 test failures
  - Patches to improve XFAIL support in selftest\_harness posted, to be able to switch expected failures from SKIP to XFAIL
  - o HW tests still need to be moved out
- Queue rate configuration
  - o devlink rate API matches HW perfectly, duplicating it buys us very little
  - o Paolo to set up a separate meeting with Jiri

## Jan 2024 (30th)

- Tour of the test runner UIs:
  - System status and test summary: <a href="https://netdev.bots.linux.dev/status.html">https://netdev.bots.linux.dev/status.html</a>
  - Example of patchwork reporting (see the netdev/contest check):
     <a href="https://patchwork.kernel.org/project/netdevbpf/patch/20240127175033.9640-1-linus.luessing@c0d3.blue/">https://patchwork.kernel.org/project/netdevbpf/patch/20240127175033.9640-1-linus.luessing@c0d3.blue/</a>
  - Clicking takes us to the list of tests run as part of the branch report: <a href="https://netdev.bots.linux.dev/contest.html?pw-n=0&branch=net-next-2024-01-29-21-00">https://netdev.bots.linux.dev/contest.html?pw-n=0&branch=net-next-2024-01-29-21-00</a>
  - Last but not least the UI for flaky tests: https://netdev.bots.linux.dev/flakes.html
- Mojatatu is running TCD on their end, so the remote executor thing is working fine. The data formats here:
  - https://docs.google.com/document/d/1TPIOOvv0GaopC3fzW-wiq8TYpI7rh8VI\_mmal0uFeJc/edit#heading=h.oax76hn06ret
    - Pedro: TCD executor code is at https://github.com/p4tc-dev/tc-executor
- What to do about slow tests? Split into a new group? Skip? Export a variable to let tests know that the perf is low?
  - Going with the export for now KSFT MACHINE SLOW=yes
  - Tests can either adjust their "acceptance criteria" down, or report XFAIL
    - Jesse: some of the perf tests can be tuned down with things like netem
    - Petr: we want them to run on HW as well, so they are kept simple
    - Matthieu: MPTCP selftests auto-detect kernel slow downs (kmemleak, lockdep, kasan, prove\_locking, etc. by looking at kallsyms). We hesitated to add a check for Qemu without KVM but we recently modified the selftests to require less resources
  - Willem: how do we deal with latency / timing sensitive tests?

- Jakub: same approach as perf tests for now, look for the export
- HW dependent tests fail for veth (ethtool, I3 stats)
  - Petr: goal was to skip cleanly on veth and fail when the env misses any tooling etc.
  - Jakub: makes sense but we do have drivers/net/ and could separate the tests since they have no chance of running
  - o Petr: with recent lib.sh changes it should be doable
- The NIPA repo has moved, Netronome has transferred the ownership to linux-netdev so we'll use <a href="https://github.com/linux-netdev/nipa">https://github.com/linux-netdev/nipa</a> going forward.

### Jan 2024 (16th)

- BQL
  - Embedded drivers missing BQL
    - Talk to the driver maintainers
    - Driver review time is best to address it
    - Some of the problems are in the vendor downstream drivers
  - Are vendors running flent and other latency tests?
    - Jesse: yes, although they struggle at DC speeds
  - o Multi-queue BQL
    - Not much interest among attendees
    - Andrew: try to keep it in the core, fewer driver changes the better
- CI
- Most of the networking selftests are now run on pending patches
  - https://netdev.bots.linux.dev/status.html
- Not reporting back to patchwork, yet because there's a bunch of pre-existing failures
- Paolo: we should extend the tests with mptcp tests
- DSA tests may also be useful Andrew to investigate
- We're looking to integrate with other people running tests themselves

## Dec 2023 (5th)

- Testing:
  - The testing repo is now fully operational, branches are combined net + net-next + all cleanly building patches from patchwork.
  - o next step for Jakub is to use it to run slow build tests (htmldocs, cocci)
  - Jesse trying to get GitHub runner targeting Intel-internal machines
  - reach out to Jesse / Jakub for access to the GitHub
- Winter Holidays shutdown:
  - Anticipate merge window to start Jan 7th
  - Initial plan 1 week starting Sat, Dec 23rd to Tue, Jan 2nd
  - Polling in the meeting: 5 votes for 1 week of shutdown, 2 votes for 2 weeks

- net-next will be closed for new features (fixes for code in net-next will be accepted)
- Dealing with interdependencies w/ other trees: what if something that landed in net-next is needed in RDMA? If it's already in, it's too late, if we're told in advance we can make a "stable branch". Andrew: note that only build time dependencies are a requirement, if the functionality doesn't work in either tree until the merge window that's fine.
- Next meeting on January 2nd.

### Nov 2023 (21st)

- cc\_maintainers got some love, it will now ignore emails which we haven't heard from in 3 years (1 year if the author and missing CC emails share the domain)
- Breno's patches for rendering netlink specs as documentation are getting close. Plan is to have the netlink protocol information rendered fully automatically at docs.kernel.org.
- CI work is making slow progress:
  - Cl doc got a "MVP" design defined ☐ netdev cross-company Cl
  - Step 1 (of 5) repo / org on GH exists <a href="https://github.com/linux-netdev/testing">https://github.com/linux-netdev/testing</a>
  - Step 2 (of 5) (trivial) test branches are getting created https://netdev.bots.linux.dev/static/nipa/branches.json
  - ToDo: step 3 test something
  - ToDo: step 4 collect the results from the above into local DB
  - Step 5 (of 5) basic UI listing the outputs -https://netdev.bots.linux.dev/contest.html
  - Jesse is working on hooking a local test runner to it.
  - Simon is trying to get resources at RedHat to run the selftests.
  - Real HW testing needs more attention, we need to define expected setup / cabling. Mlxsw has some experience. We could require 2 port cards connected back-to-back. But hyperscalers have only single port cards.
  - Jamal asks if we can send notification emails to test owners when they fail.
- Simon: what's the best way to dump state from the NIC (logs, flow table state)
  - ethtool -w is an option (Netronome, Broadcom), widely available and easy to dump
  - o debugfs can be used, too, as long as it's read only.
  - devlink health is our general go-to for reporting errors and dumping health info.
  - o devlink regions work well for binary data without error states.

## Oct 2023 (24th)

Qdisc offload:

 Different qdiscs have different ideas of what offload means, with some skipping SW and others going thru both SW and HW. Discussion to happen on ML due to connectivity issues.

## Oct 2023 (10th)

- Rust drivers
  - PHY driver exists, was posted to the Rust ML and now netdev
  - o Rust lacks Spectrum / meltdown, module versioning etc.
  - Lots of abstraction work going on, kernel integration is lacking
  - Rust people want the first driver to be perfect, so they push back
  - Review speed mismatch, Rust people will not be able to match netdev
- Netconf
  - Updates to maintainer profile clarifying reviews was posted
  - Discussion about conferences
    - Timing (moving netdev.conf? Conversations with organizers started)
    - LPC tracks will be split netdev vs BPF
    - LPC is what it is since LF took over but mixing with other subsystem at at least one conf is a must
  - Queue API
    - Header-data split
      - Jesse: when to enable split? Should drivers bring the functionality back? uAPI missing
      - uAPI for enabling, uAPI for using zero-copy are both missing but worked on
    - "Queues as first class citizen" is much awaited, Intel to share finding from their previous work
  - devlink conversation (Jiri / Andy)

## Aug 2023 (29th)

- Updating patch status for vendor patches (Saeed) currently all patches which go via vendor trees are marked as Awaiting Upstream in pw, this is confusing when (a) patch was rejected / changes requested or (b) once patch made its way to netdev trees.
  - Jakub/Daniel: existing pw-bot from Konstantin only looks back 2 weeks, so we can't use it to mark as Accepted, we'd need to add Links to the patches so that we can find their original posting;
  - Jakub: for changes requested upstream maintainers have the pw-bot access so they can mark things as changes requested.
- Cl discussion:
  - High level flow was presented: patch-flow-3.png
  - Daniel: BPF CI has some of the same features / requirements Jakub to talk to Manu about reuse

- Ondrej: LNST can be leveraged for the test part, it'd be great if others can upload their results based on the trees generated by netdev; a lot of LNST testing is about perf which may be sensitive for sharing
- Jesper: XDP testing patches were posted recently for LNST, there's also work on packaging some of the existing XDP tooling, but it's more perf testing than validation

# Aug 2023 (14th)

- Change of time current time of early Monday morning is suboptimal, new poll to be sent out
- Netdev HW testing / CI discussion on getting kernels tested on vendor HW
  - Jakub: KernelCl looks like something which allows multiple sources to report test results. Can we use it to get vendors to report results from running a common set of tests?
  - Jesse: other options to consider is adding the testing to kernel build bot or reusing stuff build by the SPDK project (<a href="https://ci.spdk.io/">https://ci.spdk.io/</a>)
  - Andrew: worth considering including/learning from existing forwarding tests for switches & DSA
  - Jesse: let's get a crystalized requirements doc together so it's easier to get resources
  - Andy: UNH is running tests for DPDK, they may also be able to help
  - o Andrew: UNH also provides interop testing for standards, lots of experience there
  - Jesse: let's put a target date on the work, get something done by plumbers
  - Jakub: longest delay may be getting the tests in place to make the testing worthwhile
  - o Andrew: we can ask Jesper for any existing BPF tests to begin with
  - Jesse: ask internally for a couple of machines in DMZ as a starting point

#### June 2023 (19th)

- Fixes missing in stable
  - Andrew reports that stable occasionally misses patches based on Fixes tags alone (e.g. an SFP fix got skipped); Greg maintains Cc: stable is still a requirement for dependable backporting.
  - Should we add a NIPA check? Add CCs on everything?
  - NIPA checks sound like a good first step but we already have a lot of expectations and process quirks.
  - Adding CC tags during review is another good option (we need to align all netdev maintainers), add the support in pw scripts <AI: Jakub: code up scripts>
  - No need to specify the tree/version in the CC tag unless there's something unusual about it.

- Do some research into how widespread the problem actually is, <u>Sasha's tools</u> may help with finding missing backports <AI: Andrew?>
- Rust and netdev
  - Nobody in the community seems to know Rust
  - Rust comes with a lot of weirdness (inlines, struct definition, lack of support for basic concepts like per-cpu)
  - Developers need hand-holding in terms of netdev (Andrew has been helping so far)
  - There is also the lack of a toolchain, which makes it impossible to consider stable
  - Trying to push towards PHY drivers seems reasonable

## June 2023 (4th)

- Inquire about kernel.org's patchwork version
  - Konstantin says that upgrades take much effort so unless there's a feature we actually need we don't upgrade. We have not identified any features we'd need.
- Feedback for netdev vger -> korg migration?
  - Konstantin to update the mail distribution software to stop sending the "email could not be delivered" notifications.
  - o reviewer list is getting migrated to kernel.org servers as well.
- Jake's notmuch reviewer setup
  - described on the mailing list

#### May 2023

- 1. Review tooling discussion
  - a. GitHub gaining use in various places, integrations and auto-deploy features are neat; we're open to adopting new tools / flows if they are more efficient;
  - b. Jakub: Are the reviewers actually using patchwork as a source of truth? Jesse: yes, it's a good overview of what needs attention; Simon: slight disconnect between list and pw because of patchwork creating its own IDs.
  - c. Jake has a script to find a series which got no replies <AI: present at next meeting?>
  - d. B4 has a lot of neat features which we may benefit from, but is currently targeting individual use not team coordination; factoring out and reusing parts of B4 could help.
  - e. Jakub: what can we improve in patchwork or other tooling? Simon: kernel.org is running an old version of patchwork <Jakub AI: ask Konstantin>
  - f. Jesse: should we add a check for the imperative mood of commit messages? For instance like: Opinionated-commit-message · Actions · GitHub Marketplace Jakub: we can, but running selftests is higher prio; checkpatch may be a better integration point.
- 2. pw-bot

- a. Still only available to select reviewers.
- b. Seems to have integrated seamlessly, nobody asking about it.
- c. Simon: does it help? Jakub: yes, it cuts down on the need for maintainers to do manual bookkeeping quite a lot; no feedback from Dave or Paolo so far.
- DPLL series progress will it ever get merged? Jakub: needs higher rate of posting (1 posting a week) < Jakub AI: ask for explicit handoffs / timelines>
- 4. X-ing netdev for bluetooth and wireless
  - a. Wireless drivers and Bluetooth directories got excluded from netdev entries in MAINTAINERS to lower the list volume:
  - b. Jesse / Simon: lowering the mail volume seems like a good move, no concerns;
  - c. Simon subscribed to wireless to see the patches, the drop in netdev CCs is visible; CAN, 6lowpan should be next?
- 5. Intra-driver review rotation created a rotation for stmmac maintainers, as an experiment.

# Apr 2023 (24th)

Virtio DMA API debacle - Alex D agrees that normal DMA API should be used; VirtIO 1
does not use DMA mapping, version 2 does; Olek says other VirtIO drivers use the DMA
API but networking is special. Nobody knows DMA bufs well.

## Mar 2023 (27th)

- 1. Jesse reached out to Jon for BBB changes, an additional system would have to be set up, LWN folks are short on time.
- 2. Andrew reached out to Alan @NXP asked for setting up internal support structure for netdev participants, will try but it's different groups within the company.

# Mar 2023 (13th)

- 1. Jesse: BBB can be made to fall back to using HTTPS which should resolve the connectivity issues at Intel, Jakub to follow up with the LWN team :
- 2. Microchip/NXP call out for help (Andrew forwarded to the reviewer list)
  - a. Jakub: trying to push them to build internal expertise, they contribute a lot
- 3. Routing Intel and BPF driver patches
  - a. Most patches should go via intel-wired-lan, the postings with [PATCH net-next] are mostly by mistake
  - b. Exceptions will be clearly explained and marked
  - c. bpf-next takes patches which touch BPF / XDP core, pure driver patches via netdev

#### Feb 2023

- 1. (Jakub) Conferences
  - a. Can we try to get the reviewer team to converge on a particular conference? (attendees: LPC: 4, netdev: 4, devconf: 2);
  - b. Are we thinking about restarting netconf? possibly, depending on how well we can organize either netconf or just a get together;
  - c. The industry slowdown makes it harder for folks to travel, especially intercontinentally;
  - d. (Maciej) travel for a more structured meeting would be easier to justify.
- 2. (Leon) Question about how to handle negotiations between upstream and 3rd party companies.

#### Dec 2022

- 1. Meeting meta-notes:
  - a. the invite included a Google Meet which caused some confusion, it has been removed now
  - b. Intel folks had to join over the phone due to corp network / VPN issues
  - c. Andy reported issues with the video not working with magic background

Overall not too bad, BBB served its purpose, we're keeping it for now [please complain if you disagree]

- 2. DaveM happy with our impact so far, the review count definitely increased.
- Pavan/Andy and Jake asked about identifying the best patches to review no silver bullet so far, Leon suggests reading the whole list during on-call days (which is also nice from the educational perspective).
- 4. We should encourage people more to add links to previous postings and make sure they CC previous maintainers.
  - a. This can be automated [ / actually we'll need to reach out to Konstantin, I don't see the linking to previous version now in pw state]
  - b. Leon: RDMA is requiring this already
  - c. Jakub: extending the CI is encouraged
- 5. Not enough people on the Intel review circle. Maintainers are happy with the current review coverage but more people could be better, up to Intel.
- 6. Inter-delegation of patches will be needed for a larger group
  - a. Intel can use their internal messaging channel in the meantime
  - b. Patchwork is lacking in this space
  - c. We should push for patchwork improvements
    - i. 

      Jakub to reach out to Konstantin and keep pushing LF

- 7. Winter Holidays (2 weeks from Dec 19th until Jan 1st)
  - a. There is no expectation of reviews during obvious shutdown periods like Xmas
  - b. Reviewers can switch shifts if they want to, Leon happy to step up since his local calendar is different
  - c. We will keep the shifts as is with the understanding that there will be disruptions
  - d. Merge window will last until Dec 25th so the first part will be chill anyway

#### Nov 2022

- 1. we will revive the netdev-driver-reviewers mailing list, and use that for team communication
- 2. whatsapp group chat might be used for some communications (but isn't preferred by some)
- 3. Intel will take next week's shift.
- 4. we're setting up the google calendar and I have sent out a bunch of invites to everyone as an editor of the calendar.
- 5. the document was mentioned to be fine ( netdev driver reviewer group ).