You may be a Linux Kernel Maintainer - and not know it

ELC 2019
San Diego, California

Frank Rowand, Sony

August 21, 2019
190818_2058

A Real Life Story

New driver is accepted

[PATCH v4 00/12] [dt-bindings] [media] Add document file and driver for Sony CXD2880 DVB-T2/T tuner + demodulator
[PATCH v4 01/12] [dt-bindings] [media] Add document file for CXD2880 SPI I/F
[PATCH v4 02/12] [media] cxd2880-spi: Add support for CXD2880 SPI interface
[PATCH v4 03/12] [media] cxd2880: Add common files for the driver
[PATCH v4 04/12] [media] cxd2880: Add spi device IO routines
[PATCH v4 05/12] [media] cxd2880: Add tuner part of the driver
[PATCH v4 06/12] [media] cxd2880: Add integration layer for the driver
[PATCH v4 07/12] [media] cxd2880: Add top level of the driver
[PATCH v4 08/12] [media] cxd2880: Add DVB-T control functions the driver
[PATCH v4 09/12] [media] cxd2880: Add DVB-T monitor functions
[PATCH v4 10/12] [media] cxd2880: Add DVB-T2 control functions for the driver
[PATCH v4 11/12] [media] cxd2880: Add DVB-T2 monitor functions
[PATCH v4 12/12] [media] cxd2880: Add all Makefile, Kconfig files and Update MAINTAINERS file for the driver

[PATCH v4 12/12] [media] cxd2880: Add all Makefile, Kconfig files and Update MAINTAINERS file for the driver

```
diff --git a/MAINTAINERS b/MAINTAINERS

+MEDIA DRIVERS FOR CXD2880
+M:     Yasunari Takiguchi <Yasunari.Takiguchi@sony.com>

+F:     drivers/media/dvb-frontends/cxd2880/*
+F:     drivers/media/spi/cxd2880*
```

Congratulations, you are a maintainer!

What happens next?

To: Rowand, Frank <Frank.Rowand@sony.com>;
Subject: FW: [PATCH] media: cxd2880-spi: fix probe when dvb_attach fails

Hi, Frank and Tim

I received below and attached mails.

We should check them and reply confirmed or some comment for them, shouldn't we?

-----Original Message-----
From: …
Subject: [PATCH] media: cxd2880-spi: fix probe when dvb_attach fails

When dvb_attach fails, probe returns 0, and remove crashes afterwards.
This patch sets the return value to -ENODEV when attach fails.

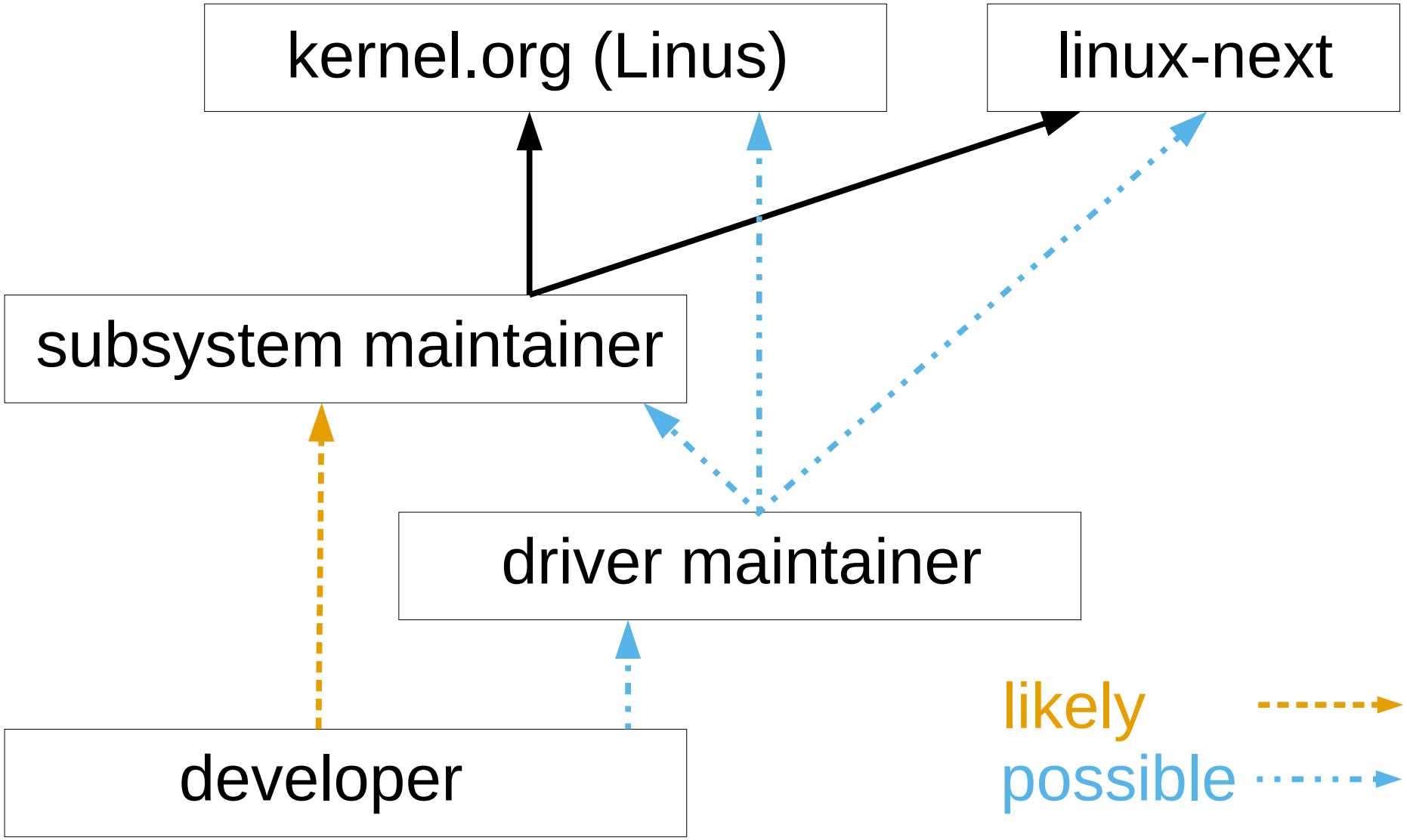
A gap in our internal OSS training

We had helped our development team learn how to submit a driver to the Linux kernel

But we had not prepared them for what was involved in being a maintainer of the driver

> What role did they have in the path a submitted patch followed?

driver maintainer - patch path?



driver maintainer - patch path?

Discuss with your subsystem maintainer

This leads into what maintainer role you choose

driver maintainer - possible roles

1) Ignore role
2) Advise upstream maintainer
3) Review patches
4) Sub-maintainer
5) … probably others that I have not thought of

1) Ignore role

- Ignore any patches to the driver

Amount of work: none

Consequence: maintainers may be less willing to accept additional drivers or major features from you

If this is what you plan to do, please send a patch that removes your name from the MAINTAINERS file

2) Advise upstream maintainer

- Reply to patch emails with Acked-by tag if you see no issues or problems, tells maintainer you have no issues
- If you see issues or problems, report them in reply (this is less rigorous than a “Reviewed-by” tag)
- At minimum, should:
  - read and analyze patch
  - test patch (you might be the only developer who has the required hardware)
- Consider providing an instance of the hardware (or remote access to the hardware in your lab) to your upstream maintainer

Amount of work: small, puts burden on upstream maintainer

3) Review patches

All the tasks for “2)”, plus:

- Review all patches to the driver
  - provide good feedback:
    - what is ok
    - what should be changed
    - suggest possible alternatives if change needed
    - what is not acceptable (and why)
- Reply to patch email with “Reviewed-by” tag if the conditions for the tag are met

Amount of work: small to moderate to large

- depends on patch volume, driver complexity, parent subsystem complexity

Reviewed-by

(a) I have carried out a technical review of this patch to evaluate its appropriateness and readiness for inclusion into the mainline kernel.

(b) Any problems, concerns, or questions relating to the patch have been communicated back to the submitter. I am satisfied with the submitter's response to my comments.

(c) While there may be things that could be improved with this submission, I believe that it is, at this time, (1) a worthwhile modification to the kernel, and (2) free of known issues which would argue against its inclusion.

(d) While I have reviewed the patch and believe it to be sound, I do not (unless explicitly stated elsewhere) make any warranties or guarantees that it will achieve its stated purpose or function properly in any given situation.

Reviewed-by

A Reviewed-by tag is a statement of opinion that the patch is an appropriate modification of the kernel without any remaining serious technical issues.

Documentation/process/submitting-patches.rst

4) Sub-maintainer

All the tasks for “3)”, plus:

- Take on all of the responsibilities of a maintainer
- Occasional review of additional drivers that are similar to your driver

Amount of work: small to moderate to large
  - depends on patch volume, driver complexity, parent subsystem complexity

Some Maintainer Tasks

Higher level of responsibility than for a developer

Takes a broader view of the impact of patches

Some Maintainer Tasks

- maintain architectural integrity
- ensure all use cases and classes of use are considered, for example, from small embedded to super computer
- ensure maintainability of driver and subsystem
  - avoid unjustified complexity
  - enforce kernel coding style
  - focus on readability
- ensure use case and/or problem statement is provided for proposed solutions or implementations
- ensure testing was done
- ensure static checks were completed

Some Maintainer Tasks

- ensure compatibility where compatibility is required
  - user space API
  - devicetree
  - etc
  - for kernel API, see: stable-api-nonsense.rst
- avoid performance regresssions
- assist developers

Communicate with your Upstream

Discuss with your subsystem maintainer

Inform him/her of the role that you would like to assume

You can move back and forth between the different roles as needed

Your Role summarized

Participate in the flow of a patch from developers to the upstream maintainer

Make your upstream maintainer’s job easier

Help contributors

The world of tags

Looking at tags provides some insights
into the role of a driver maintainer
in the flow of a patch

What tags should you require?

What tags will you add?

Signed-off-by:

- Developer’s tag must exist in patch
- If the patch flows through the driver maintainer tree, with a pull request to the subsystem maintainer, then the driver maintainer adds his/her Signed-off-by: tag

> …, I certify that:
>
> …
>
> (c) The contribution was provided directly to me by some other person who certified (a), (b) or (c) and I have not modified it.
>
> ...

Cc: stable@vger.kernel.org

(Not a tag…)

[optional]

Add into the sign-off area of the patch if it fixes a severe bug in a released kernel

> The maintainers of the stable trees will use this information as a suggestion to include the patch in stable branches

Fixes:

[nice to have]

- If patch is expected to be added to stable, provides stable tree maintainers information about which stable branches are impacted

- For all patches:
    - makes reviewers, testers job easier
    - reduces risk that the bug will re-appear in a subsequent patch

Tested-by:

How should patches be tested?
What does your subsystem maintainer want?

Should be mandatory:

- checkpatch
- no new compile warnings (without “w=123”)
- if bug fix, verify the bug is fixed
- subsystem specific tests or test suite
- if new feature, feature matches documentation
- the kernel boots

Tested-by:

How should patches be tested?

Suggested:

- make W=1 ----- no new warnings
- make W=23 ----- justifiable new warnings
- sparse ----- prefer no new warnings
- smatch ----- prefer no new warnings

Tested-by:

How should patches be tested?

Sometimes required:

- performance data

Reviewed-by:

[previously discussed]

Time expectations

Allocate some time in your work schedule to allow responses to submitted patches

Your manager should be aware of your need to be responsive to the community

Time expectations

If you will be on vacation, or otherwise unavailable for an extended period, your upstream maintainer may want to be aware of your absence

If possible, try to arrange for a knowledgeable coworker to monitor the mail list to answer questions about the driver or hardware, test patches, and review patches

Bonus Points

Monitor or skim your subsystem mail list
- keep up with subsystem trends and direction
- catch emails which fail to include you in the To: list

Keep up with trends in kernel maintainer practices
- subscribe to ksummit-discuss
  Ksummit-discuss-request@lists.linuxfoundation.org

Review

Your name in your driver entry in MAINTAINERS makes you a maintainer

You can choose your level of involvement

Some maintainer tasks

Your upstream maintainer is a resource to help you

THE END

Thank you for your attention...

Questions?

How to get a copy of the slides

1) leave a business card with me

2) frank.rowand@sony.com

3) http://events.linuxfoundation.org

Resources

If your upstream maintainer pulls from you
- git repository
- pgp key in the kernel web of trust

https://korg.wiki.kernel.org/