Patch workflow

Eng. Mauro Carvalho Chehab, CISSP, MSc
Kernel General Specialist
Patch workflow

- Patches sent to linux-media@vger.kernel.org are stored at patchwork.linuxtv.org, tagged as “New”.
  - Mangled patches are silently discarded;
- Patches are converted locally into a quilt tree;
- After reviewing a patch, they're tagged to reflect its review status;
- Currently, patchwork “delegation” feature is not used.
Patch workflow (2)

• Main patch status:
  • **New**: Patches not yet reviewed;
  • **Under review**: someone is expected to be reviewing it (driver's maintainer).
    - Unfortunately, patchwork doesn't have a field to indicate who will be reviewing it. It does have a field that indicates to whom a patch got delegated.
    - There's no process to warn the driver's maintainer about that patch: it is all manual;
  • **Superseded**: when the same patch is sent twice, or a new version/git pull was sent; the old ones are targeted as superseded;
  • **Obsoleted**: patch doesn't apply anymore or the patched driver doesn't exist anymore.
  • **Changes requested**: when someone requests changes at the patch;
  • **Rejected**: When the patch is wrong or doesn't apply;
  • **RFC**: patches where the maintainer doesn't need to handle;
  • **Not Applicable**: patches that aren't meant to be applicable via the media-tree.git;
  • **Accepted**: when patch is merged; when driver maintainer says that the patch will be applied via his tree;
    - Fixes go through a different tree, and only sent upstream when they're about to get merged.
Patch merge scripts

- All patches come from Patchwork
- They got converted into a quilt tree
- Script assisted:
  - `get_linux_media`
    - Uses pwclient to list New/Under Review patches;
    - Syncs my quilt tree from patchwork
  - `show_next` - very simple script:
    - `cat `quilt next`` | colordiff
    - `kompare `quilt next` `
Patch merge scripts (2)

- test_patch - checks several things at the patch
  - Display, the relevant patch headers, SOB's, ACK's;
  - Tests if patch applies (and if it is at -p1 or -p2) format, or if it was already applied;
  - Runs scripts/checkpatch.pl;
  - Checks similar patches at quilt;
  - Search for message replies;
  - Search messages on my local IMAP server with the same subject;
  - Get the driver maintainers, as listed at MAINTAINERS (excluding myself)
Examples

v4l@pedra ~/v4l/patchwork $ test_patch
testing if patches/lmml_15124 omap3isp_preview_add_support_for_8_bit_formats_at_the_sink_pad.patch applies
patch -p1 -i patches/lmml_15124 omap3isp_preview_add_support_for_8_bit_formats_at_the_sink_pad.patch --dry-run -t -N
patching file drivers/media/platform/omap3isp/isppreview.c
drivers/media/platform/omap3isp/isppreview.c | 37 +++++++++++++++++++++++++++++++++++
1 file changed, 26 insertions(+), 11 deletions(-)
Subject: omap3isp: preview: Add support for 8-bit formats at the sink pad
From: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
Patchwork: http://patchwork.linuxtv.org/patch/15124/
Date: Tue, 23 Oct 2012 10:23:39 -0000
Signed-off-by: Laurent Pinchart <laurent.pinchart@ideasonboard.com>
Patch applies OK
total: 0 errors, 0 warnings, 94 lines checked

patches/lmml_15124 omap3isp_preview_add_support_for_8_bit_formats_at_the_sink_pad.patch has no obvious style problems and is ready for submission.
Found message(s) 141686 141764
Subject: [PATCH v2] omap3isp: preview: Add support for 8-bit formats at the sink pad
<1350997862-18880-1-git-send-email-laurent.pinchart@ideasonboard.com>
Subject: Re: [PATCH v2] omap3isp: preview: Add support for 8-bit formats at
<20121024171208.GB23933@valkosipuli.retiisi.org.uk>
Found message(s) 141684 141686 141764
Subject: [PATCH] omap3isp: preview: Add support for 8-bit formats at the sink pad
<1350991419-23028-1-git-send-email-laurent.pinchart@ideasonboard.com>
Subject: [PATCH v2] omap3isp: preview: Add support for 8-bit formats at the sink pad
<1350997862-18880-1-git-send-email-laurent.pinchart@ideasonboard.com>
Subject: Re: [PATCH v2] omap3isp: preview: Add support for 8-bit formats at
<20121024171208.GB23933@valkosipuli.retiisi.org.uk>
Laurent Pinchart <laurent.pinchart@ideasonboard.com> (maintainer:OMAP IMAGE SIGNAL...)
Duplicated md5sum patches
Likely duplicated patches (need manual check)
Patch merge scripts (3)

- **import_patch** – applies the first patch from quilt
  - Checks if patch got tested;
  - Preserve patch timestamps/authorship;
  - Do a few other consistency checks;
  - Adds [media] prefix;
  - Prepare to update patchwork

- **change_status** – changes pathwork status
  - Runs in “local mode”: updates a local file
    - There is a “flush” parameter to update main server

- **gitimport** – handles a “git pull” email

- **ancillary scripts** (githead.pl; search_msg.pl)
Patch merge scripts (4)

- quick_compile.sh
  - Compiles all media drivers, including staging ones
    - By default: i386. Also supports x86_64
- auto_compile
  - Detects a new git merge and runs quick_compile.sh
  - Plays audio and prints error in RED when bad things happen.
Simplified process

New patch on git?

- Yes: Compile
- No: Buzz

Compile

Error?

- Yes: Buzz
- No: Check patch, helped by show_patch and test_patch

Check patch, helped by show_patch and test_patch

- OK?
  - Yes: Import patch
  - No: Buzz
- No: Import patch

Import patch

Change status
Patch workflow discussions

Eng. Mauro Carvalho Chehab, CISSP, MSc
Kernel General Specialist
Ideas for brainstorm

• Have 2 or 3 sub-maintainers for drivers
• Have more SOB's at core changes
• Use delegation at patchwork
• Having someone adding more features at patchwork
  • Fixing delegation to associate it with an ACL;
  • Improve “under review” to store/notify about who is reviewing a patch;
  • Fix it to either notify when a new patch arrives, and/or when a bad patch got discarded;
• Use branch topics;
• Comments? What else?