Your Vendor's BSP Is Probably Not Built for Product Longevity

Now What?

Anna-Lena Marx, inovex GmbH

Yocto Project Summit, 2025.12

inovex

anna-lena.marx@inovex.de

+49 1523 / 31 81 26 0

@anna-lena-marx-embedded

@Allegra42

Hello,

I’m Anna-Lena Marx

Tech Lead Embedded Linux, Karlsruhe (Germany)

- with inovex since 2015
- has a Master’s degree in Embedded Systems
- ~~studies~~ studied Electrical Engineering as a hobby
- Yocto user since ~2017? Pre-LTS times!

#embeddedsystems #yocto #linux #kernel #zephyr #aosp

inovex

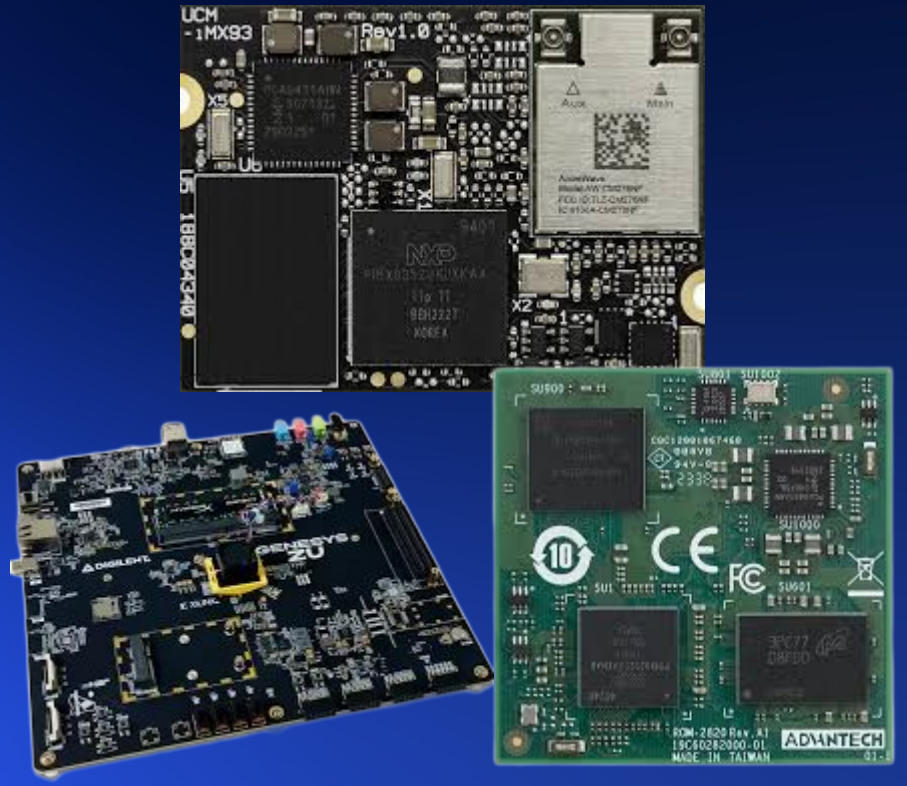
The Promise
Board Support Packages

You get a new board from your hardware vendor.

It boots with the vendor BSP.

The demos run.

It feels like you’re 90% done.



inovex

The Reality...
some years (or months) later

inovex

The Reality...
some years (or months) later

- Lots of CVEs to fix
  ... for components you can't recall why they are even part of the system
- The Yocto release just went out of support
  ... and was your BSP even based on a LTS release?
- You notice your vendor BSP utilizes a 3-year-old kernel fork
  ... unmaintained and with required binary blobs for the cool and shiny board features

And have you thought about updates and regulatory requirements at all?

inovex

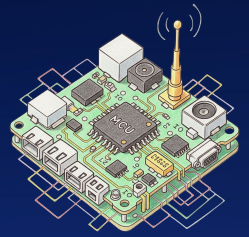
Let’s step back

Our Goals are not Aligned

Board/Silicon Vendor Focus and Goals

- Showcase **all** product features
  - CPU
  - special co-processors
  - unique selling points
- Provide an easy starting point for customers
- Provide several BSP variants for the whole product portfolio
  - often several BSP variants needed
    e.g Linux, Android, QNX, ...
  - must be easy and manageable for the vendor!
- **Sell silicon**
  ... and move to the next product

Integrator/Product Manufacturer Goals



- Build and ship one **product**
  - stable, secure & maintainable
  - minimal software stack
    → reduced surface for attacks and CVEs
  - low maintenance effort needed
  - expected lifespan: 5 - 20 years
- Fulfil target market regulations
  - e.g. Cyber Resilience Act,
    IEC 62304 (Medical Devices), ...
  - Security Updates
  - ...
- Sell **this product**
  ... and support it over its lifetime

inovex

Let’s have a look at some real world examples!
Evaluate BSPs

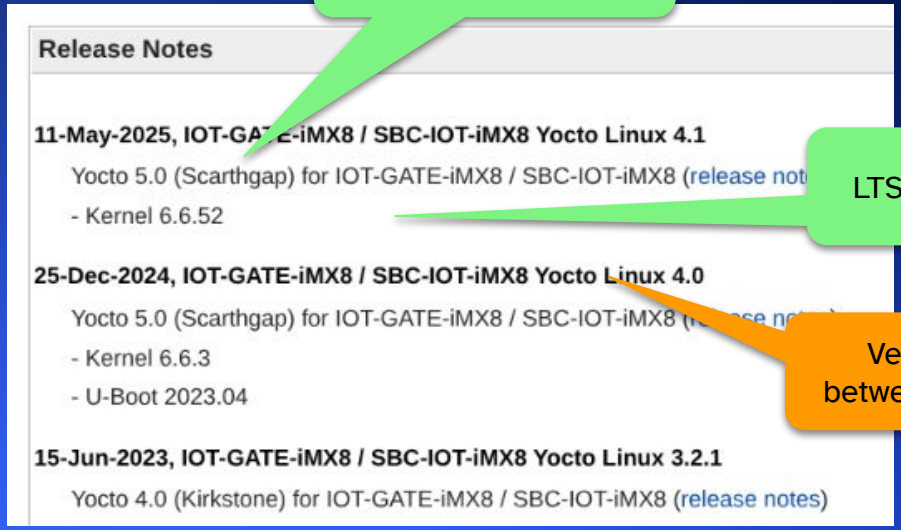
Quick reminder

~~No~~ limited vendor blaming

→ basically all (vendor) BSPs and layers contain better and worse parts

→ **this is to learn about pitfalls and what to look at when evaluating a BSP!**

Let’s just collect some points for our BSP wishlist!



Releases based on Yocto LTS branches

LTS kernel versions

Forked kernel or mainline Yocto/Linux with patches?

Very long cycles between BSP releases

Prepare NXP BSP

```
mkdir compulab-freescale-bsp && cd compulab-freescale-bsp
repo init -u https://github.com/nxp-imx/imx-manifest -b imx-linux-scarthgap -m imx-6.6.23-2.0.0.xml
```

Download CompuLab meta layer

```
mkdir -p .repo/local_manifests
wget --directory-prefix .repo/local_manifests https://raw.githubusercontent.com/compulab-yokneam/meta-bsp-imx8mm/${CL
```

NXP and board vendor repo manifest
→ but what layers are utilized?

Get entire BSP tree

```
repo sync
```

imx-manifest / imx-6.6.23-2.0.0.xml

junzhuimx imx-6.6.23-2.0.0.xml: Lock demo layer revision with release tag

Code Blame 44 lines (35 loc) · 3.34 KB

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <manifest>
3
4    <default revision="scarthgap" sync-j="4"/>
5
6    <remote name="yp"          fetch="https://git.yoctoproject.org"/>
7    <remote name="oe"          fetch="https://github.com/openembedded"/>
8    <remote name="kraj"        fetch="https://github.com/kraj"/>
9
10   <remote name="community"   fetch="https://github.com/Freescale"/>
11   <remote name="ossystems"   fetch="https://github.com/OSSystems"/>
12   <remote name="qt"          fetch="https://github.com/YoeDistro"/>
13   <remote name="timesys"     fetch="https://github.com/TimesysGit"/>
14
15   <remote name="imx"         fetch="https://github.com/nxp-imx"/>
16   <remote name="imx-support" fetch="https://github.com/nxp-imx-support"/>
17
18   <project name="poky" remote="yp" path="sources/poky" revision="02eacd385eca7aa9cf25aaaa61427aa9c126e0bc" upstream="scarthgap"/>
19   <project name="meta-openembedded" remote="oe" path="sources/meta-openembedded" revision="a72010b414ee3d73888ac9cb4e310e8f05e13aea" upstream="scarthgap"/>
20   <project name="meta-clang" remote="kraj" path="sources/meta-clang" revision="66414c17ae60e489bef9a97031323e92c247fbe1" upstream="scarthgap"/>
21
22   <project name="fsl-community-bsp-base" remote="community" path="sources/base" revision="84df0b1ff416c6739472f7f7eadff04fa90d38f6" upstream="scarthgap">
23     <linkfile dest="README" src="README"/>
24     <linkfile dest="setup-environment" src="setup-environment"/>
25   </project>
```

NXP manifest
→ YoeDistro, Poky, …
Lots of potentially not needed layers for your product!

meta-bsp-imx8mm / scripts / **imx_6.6.3_1.0.0-compulab.xml**

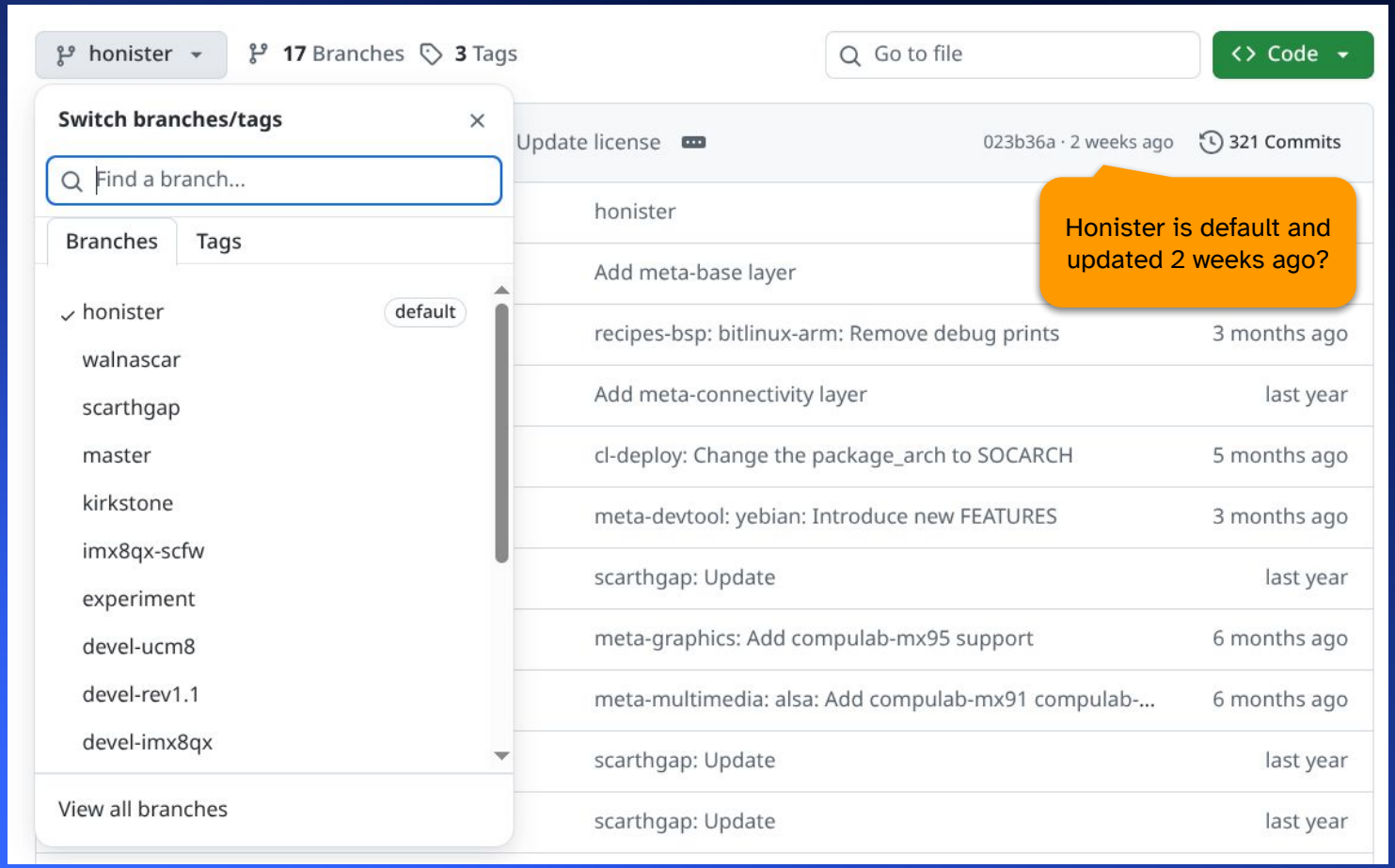
kkapranov meta: move to branch rel_imx_6.6.3_1.0.0

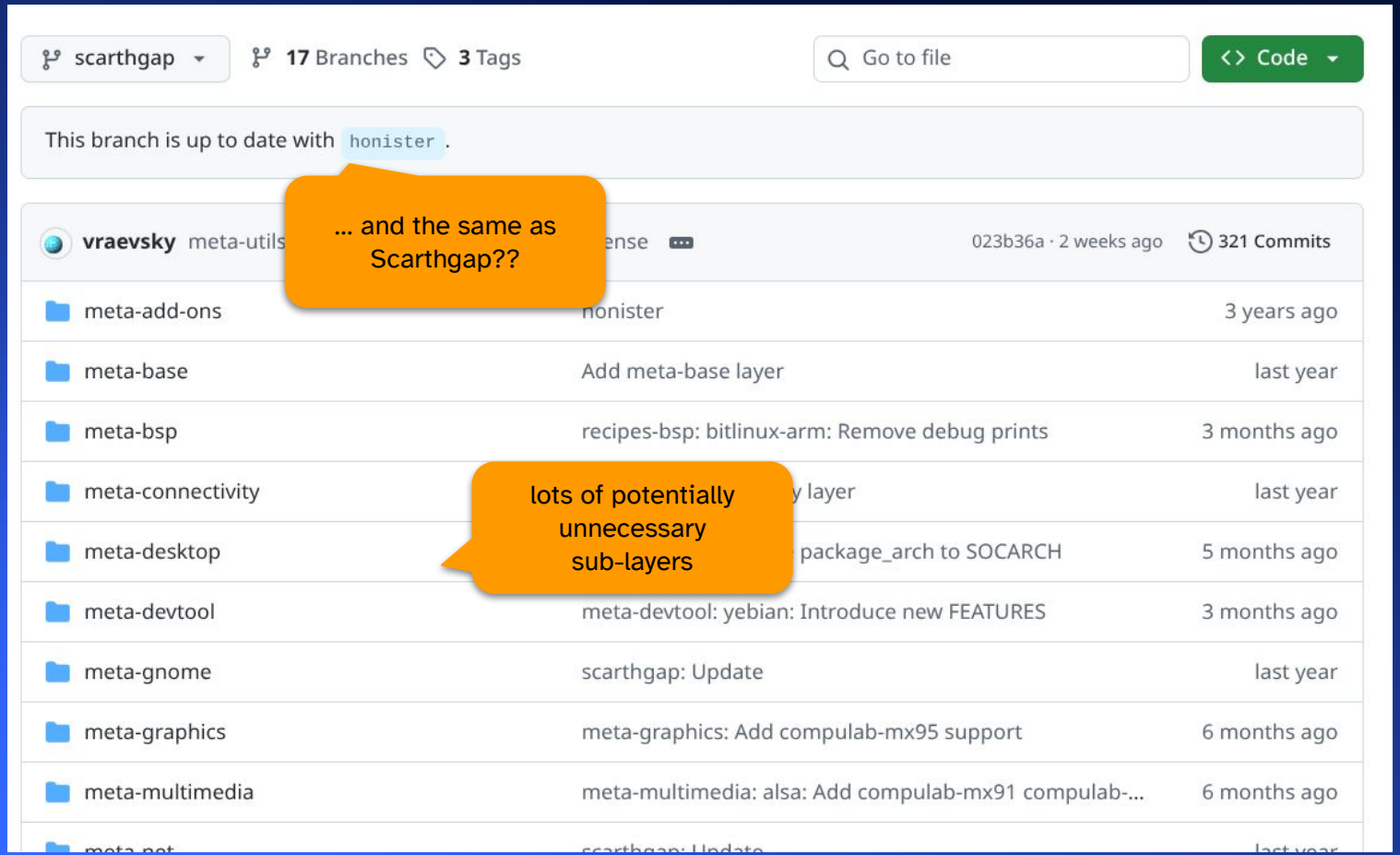
Code | Blame | 12 lines (9 loc) · 550 Bytes

```
1   <?xml version="1.0" encoding="UTF-8"?>
2   <manifest>
3
4     <remote fetch="https://github.com/compulab-yokneam" name="compulab"/>
5
6     <project name="meta-compulab" remote="compulab" revision="honister" path="sources/meta-compulab"/>
7     <project name="meta-compulab-bsp" remote="compulab" revision="honister" path="s                    "/>
8     <project name="meta-bsp-imx8mm" remote="compulab" revision="rel_imx_6.6.3_1.0                    imx8mm" >
9       <linkfile src="tools/compulab-setup-env" dest="compulab-setup-env" />
10    </project>
11
12  </manifest>
```

Oh... some more layers!

I thought this is an Scarthgap BSP?


honister
17 Branches
3 Tags
Go to file
Code
Switch branches/tags
Find a branch...
Branches
Tags
honister
default
walnascar
scarthgap
master
kirkstone
imx8qx-scfw
experiment
devel-ucm8
devel-rev1.1
devel-imx8qx
View all branches
Update license
023b36a · 2 weeks ago
321 Commits
honister
Add meta-base layer
recipes-bsp: bitlinux-arm: Remove debug prints
3 months ago
Add meta-connectivity layer
last year
cl-deploy: Change the package_arch to SOCARCH
5 months ago
meta-devtool: yebian: Introduce new FEATURES
3 months ago
scarthgap: Update
last year
meta-graphics: Add compulab-mx95 support
6 months ago
meta-multimedia: alsa: Add compulab-mx91 compulab-...
6 months ago
scarthgap: Update
last year
scarthgap: Update
last year
Honister is default and updated 2 weeks ago?


scarthgap
17 Branches
3 Tags
Go to file
Code
This branch is up to date with honister .
... and the same as Scarthgap??
vraevsky meta-utils
ense
023b36a · 2 weeks ago
321 Commits
meta-add-ons
honister
3 years ago
meta-base
Add meta-base layer
last year
meta-bsp
recipes-bsp: bitlinux-arm: Remove debug prints
3 months ago
meta-connectivity
y layer
last year
meta-desktop
package_arch to SOCARCH
5 months ago
lots of potentially unnecessary sub-layers
meta-devtool
meta-devtool: yebian: Introduce new FEATURES
3 months ago
meta-gnome
scarthgap: Update
last year
meta-graphics
meta-graphics: Add compulab-mx95 support
6 months ago
meta-multimedia
meta-multimedia: alsa: Add compulab-mx91 compulab-...
6 months ago

kas-dhsom / include / common-meta-dhsom-imx-bsp.yml

Marek

Code

```
1
2
3
4       - include/common-settings.yml
5
6   distro: nodistro
7
8   target:
9     - core-image-base
10
11  repos:
12    poky:
13      url: https://git.yoctoproject.org/poky
14      layers:
15        meta:
16
17    meta-mainline-common:
18      url: https://source.denx.de/denx/meta-mainline-common.git
19      branch: main
20
21    meta-dhsom-imx-bsp:
22      url: https://github.com/dh-electronics/meta-dhsom-imx-bsp.git
23      branch: main
```

Caution!
When adding additional features like OP-TEE, TF-A or proprietary binary blobs, the list will extend!

Minimal layer dependencies

ONE clear BSP layer!

kas-dhsom / include / common-settings.yml

Marek Vasut Generate SPDX SBoM list

Code Blame 24 lines (19 loc) · 935

```
1   header:
2     version: 14
3
4   bblayers_conf_header:
5     standard: |
6       POKY_BBLAYERS_CONF_VERSION = "2"
7       BBPATH = "${TOPDIR}"
8       BBFILES ?= ""
9
10  local_conf_header:
11    standard: |
12      BB_HASHSERVE = "auto"
13      BB_SIGNATURE_HANDLER = "OEEquivHash"
14
15    buildinfo: |
16      INHERIT += " image-buildinfo"
17      IMAGE_BUILDINFO_VARS:append = " ARM_INSTRUCTION_SET BB_VERSION BUILD_
18
19    cve-check: |
20      INHERIT += "cve-check"
21      include conf/distro/include/cve-extra-exclusions.inc
22
23    spdx-list: |
24      INHERIT += "create-spdx"
```

Sane and minimal default configuration

meta-bsp-imx8mm / conf / **layer.conf**

```
19
20  CORE_IMAGE_EXTRA_INSTALL += " kernel-modules kernel linux-compulab-headers kernel-dev firmware-imx-sdma-imx7d"
21  CORE_IMAGE_EXTRA_INSTALL += " packagegroup-tools-bluetooth "
22  CORE_IMAGE_EXTRA_INSTALL += " grub "
23  CORE_IMAGE_EXTRA_INSTALL += " iw "
24  CORE_IMAGE_EXTRA_INSTALL += " e2fsprogs-tune2fs "
25  CORE_IMAGE_EXTRA_INSTALL += " watchdog "
26
27  HOSTTOOLS += " git-lfs "
28  HOSTTOOLS += " bison "
29
30  LAYERSERIES_COMPAT_compulab-bsp-imx8mm = "kirkstone scarthgap"
```

hidden extra installs!

meta-bsp-imx8mm / recipes-kernel / linux / **linux-compulab_6.6.52.inc**

kkapranov meta:linux: Upgrade to kernel 6.6.52

Code | Blame | 4 lines (4 loc) · 180 Bytes

```
1  SRCBRANCH = "linux-compulab_v6.6.52"
2  SRC_URI = "git://github.com/compulab-yokneam/linux-compulab;protocol=https;branch=${SRCBRANCH}"
3  SRCREV = "${AUTOREV}"
4  LINUX_VERSION = "6.6.52"
```

Forked kernel!
Actively developed, but no security patches applied!

The

About Contact us FAQ Releases Signatures Site news

Protocol	Location
HTTP	https://www.kernel.org/pub/
GIT	https://git.kernel.org/
RSYNC	rsync://rsync.kernel.org/pub/

Latest Release
6.17.7

mainline:	6.18-rc4	2025-11-02	[tarball]		[patch]	[inc. patch]	[view diff]	[browse]
stable:	6.17.7	2025-11-02	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
stable:	6.16.12 [EOL]	2025-10-12	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
longterm:	6.12.57	2025-11-02	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
longterm:	6.6.116							
longterm:	6.1.158	2025-10-29	[tarball]	[pgp]	[patch]	[inc. patch]		
longterm:	5.15.196	2025-10-29	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
longterm:	5.10.246	2025-10-29	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
longterm:	5.4.301	2025-10-29	[tarball]	[pgp]	[patch]	[inc. patch]	[view diff]	[browse] [changelog]
linux-next:	next-20251103	2025-11-03						[browse]

Current LTS Version
6.6.116

meta-dhsom-imx-bsp / recipes-kernel / linux / **linux-stable_6.6.bbappend**

Marek Vasut machine: Switch DH i.MX6ULL DHSOM based systems to Linux 6.6.y

Code | Blame | 29 lines (24 loc) · 1.19 KB

```
1  FILESEXTRAPATHS:prepend := "${THISDIR}/${PN}:"
2
3  KBRANCH:dh-imx-dhsom ?= "linux-6.6.y"
4  KMACHINE:dh-imx6-dhsom ?= "dh-imx6-dhsom"
5  KMACHINE:dh-imx6ull-dhsom ?= "dh-imx6ull-dhsom"
6  KMACHINE:dh-imx8mp-dhsom ?= "dh-imx8mp-dhsom"
7  COMPATIBLE_MACHINE:dh-imx6-dhsom = "(dh-imx6-dhsom)"
8  COMPATIBLE_MACHINE:dh-imx6ull-dhsom = "(dh-imx6ull-dhsom)"
9  COMPATIBLE_MACHINE:dh-imx8mp-dhsom = "(dh-imx8mp-dhsom)"
10
11 DEPENDS:append:dh-imx-dhsom = " lzop-native "
12
13 SRC_URI:append:dh-imx6-dhsom = " \
14     file://${BPV}/dh-imx6-dhsom;type=kmeta;destsuffix=${BPV}/dh-imx6-dhsom \
15     file://common/dh-imx6-dhsom;type=kmeta;destsuffix=common/dh-imx6-dhsom \
16     "
17 KERNEL_FEATURES:dh-imx6-dhsom = " dh-imx6-dhsom-standard.scc "
18
19 SRC_URI:append:dh-imx6ull-dhsom = " \
20     file://${BPV}/dh-imx6ull-dhsom;type=kmeta;destsuffix=${BPV}/dh-imx6ull-dhsom \
21     file://common/dh-imx6ull-dhsom;type=kmeta;destsuffix=common/dh-imx6ull-dhsom \
22     "
23 KERNEL_FEATURES:dh-imx6ull-dhsom = " dh-imx6ull-dhsom-standard.scc "
24
25 SRC_URI:append:dh-imx8mp-dhsom = " \
26     file://${BPV}/dh-imx8mp-dhsom;type=kmeta;destsuffix=${BPV}/dh-imx8mp-dhsom \
27     file://common/dh-imx8mp-dhsom;type=kmeta;destsuffix=common/dh-imx8mp-dhsom \
28     "
29 KERNEL_FEATURES:dh-imx8mp-dhsom = " dh-imx8mp-dhsom-standard.scc "
```

Appends a mainline kernel
(meta-mainline-common)
No fork!

Also appends for
bootloader and other
firmware components

meta-raspberrypi / conf / machine / include / **rpi-default-versions.inc**

shr-project and **kraj** Revert "rpi-default-versions: Switch default kernel to 6.12"

This reverts commit 8c916b6.

Change like this doesn't belong to stable branch, people who want to use 6.12 can easily switch the P_V in their config.

As reported in:
#1483

lttng-modules version in scarthgap fails to build with 6.1

lttng-modules/2.13.12/lttng-modules-2.13.12/src/../inclu
lttng-modules-2.13.12/src/../include/wrapper/uprobes.h
declaration]
34 | uprobe_unregister(inode, offset, uc);
| ^~~~~~~~~~~~~~~~

Signed-off-by: Martin Jansa <martin2.jansa@lgepartner.c

Code | Blame | 4 lines (3 loc) · 168 Bytes

```
1  # RaspberryPi BSP default versions
2
3  PREFERRED_VERSION_linux-raspberrypi ??= "6.6.%"
4  PREFERRED_VERSION_linux-raspberrypi-v7 ??= "${PREFERRED_VERSION_linux-raspberrypi}"
```

Always do the deep dive!
- meta-raspberrypi is well maintained
- the default kernel version for scarthgap is 6.6.x → LTS

BUT

linux Public

rpi-6.6.y | **104** Branches | **108** Tags | Go to file t | Add file

This branch is 16614 commits ahead of and 108013 commits behind `rpi-6.12.y`.

pelwell dts: rp1: Don't use DMA with UARTs bba53a1 · 8 months ago

.github Update the Issue template

inovex

Now What?

inovex

So what do we actually need from a BSP?
A Wishlist

- One single BSP Layer **based on a Yocto LTS release**, including
  - a **mainline Linux** or **Yocto based LTS kernel** with needed patches
    → the less patches the better!
  - same for the **bootloader**
  - a **device tree** describing the board (ARM machines)
    - for kernel and bootloader
  - a Yocto **machine.conf** configuration
    → focused on the essentials, clean, understandable and reusable
  - additional **firmware** for the board features
  - if needed binary blobs
  - board specific overlays if needed

And that's it!

inovex

So what do we actually need from a BSP?
A Wishlist

Further points I wish for

- Standard tools like **KAS** or **bitbake-setup**
  → avoid the need to maintain yet another build and layer management tool, also as a vendor!
- No implicit enforcement of vendor specific workflows and tooling
- A full featured demo layer or demo configuration is fine!
  - if it is separated from the actual BSP
  - and optional

As a vendor, please do not install additional packages from obscure and incomprehensible locations!

inovex

When is the time to build your own BSP Layer?

Consider setting up your own BSP if

- the vendor's BSP is based on an outdated Yocto release or a non-LTS version
- kernel or bootloader are forked and do not get security updates
- you do continuously find packages you did not add into the image
- the BSP is a mess and hard to understand and use

Setting up your own Yocto BSP layer is not that hard!
It's a valid approach and you are in control!

inovex

When it’s time to build your own BSP Layer

1. Start with a mainline Yocto LTS release

2. Create your own meta-<cpu>-bsp layer

3. Get the device tree and defconfig for kernel as well as the bootloader from the vendor’s BSP

4. Define your bootloader and kernel
   a. Are board specific patches needed?
      → keep as close as possible on a mainline LTS release!
      → cherry-pick needed patches
   b. Are specific firmware blobs needed?

5. Create your machine.conf configuration
   → copy&paste only the relevant parts from the vendor’s definition → adjust as needed
   → make sure the machine is picked up by the kernel recipe’s `COMPATIBLE_MACHINE` variable

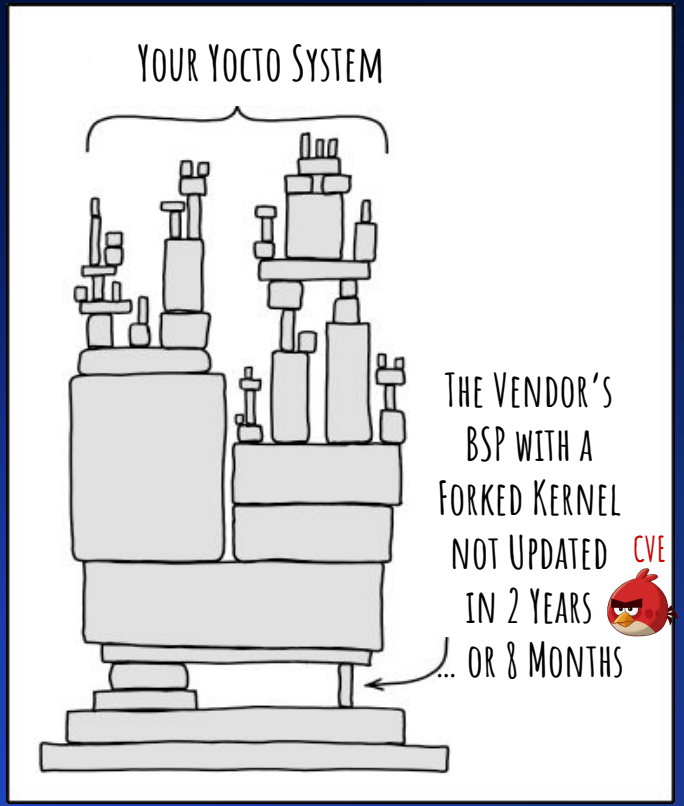
6. Integrate additional firmware components
   e.g. trusted firmware, OPTEE, ...

7. Add hardware specific appends if needed

inovex

The Takeaway
Your BSP is a Foundation, Not a Demo!

- A well-maintained, up-to-date BSP is essential for security
  → a forked vendor kernel can showcase features
  → but fast and reliable security patches are crucial in production
- Most often, very few demo features are actually needed in your product
  → but demo features and additional packages increase the attack surface
- Investing in a solid and maintainable base helps on the long run
  → and lowers the ongoing effort during a product's lifecycle
- Regulations like CRA require security updates
  → have a well-tested update mechanism in place
  → establish a BSP foundation and workflow that support you to meet regulatory requirements with ease!



Thank you!

Anna-Lena Marx

Tech Lead Embedded Linux

+49 1523 / 3181260

anna-lena.marx@inovex.de

yocto PROJECT | THE LINUX FOUNDATION