The State of PTXdist

Roland Hieber <rhi@pengutronix.de>

FOSDEM 2020

What is PTXdist?

- Build system geared towards Embedded Linux
- Building blocks:
  - GNU Make
  - Kconfig
  - Bash
  - some AWK
- First version before August 2003
- Monthly releases
- License: GPL-2.0
- Online documentation: https://ptxdist.org/doc

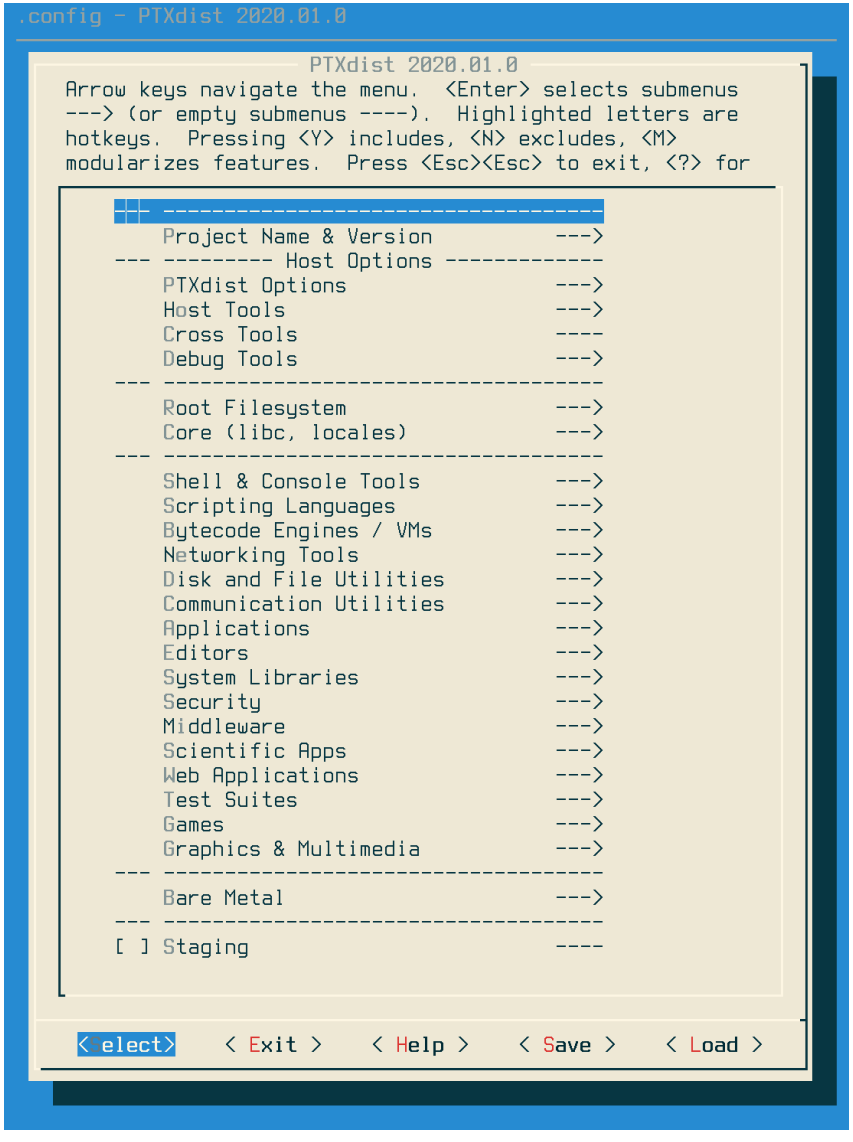
Statistics

- 914 target packages
- 184 host packages
- 18 pre-built toolchain architectures (via OSELAS.Toolchain)
  - ARM-v4/v5/ARM-v7, AArch64, x86, x86_64, MIPS, RISC-V, PowerPC 600

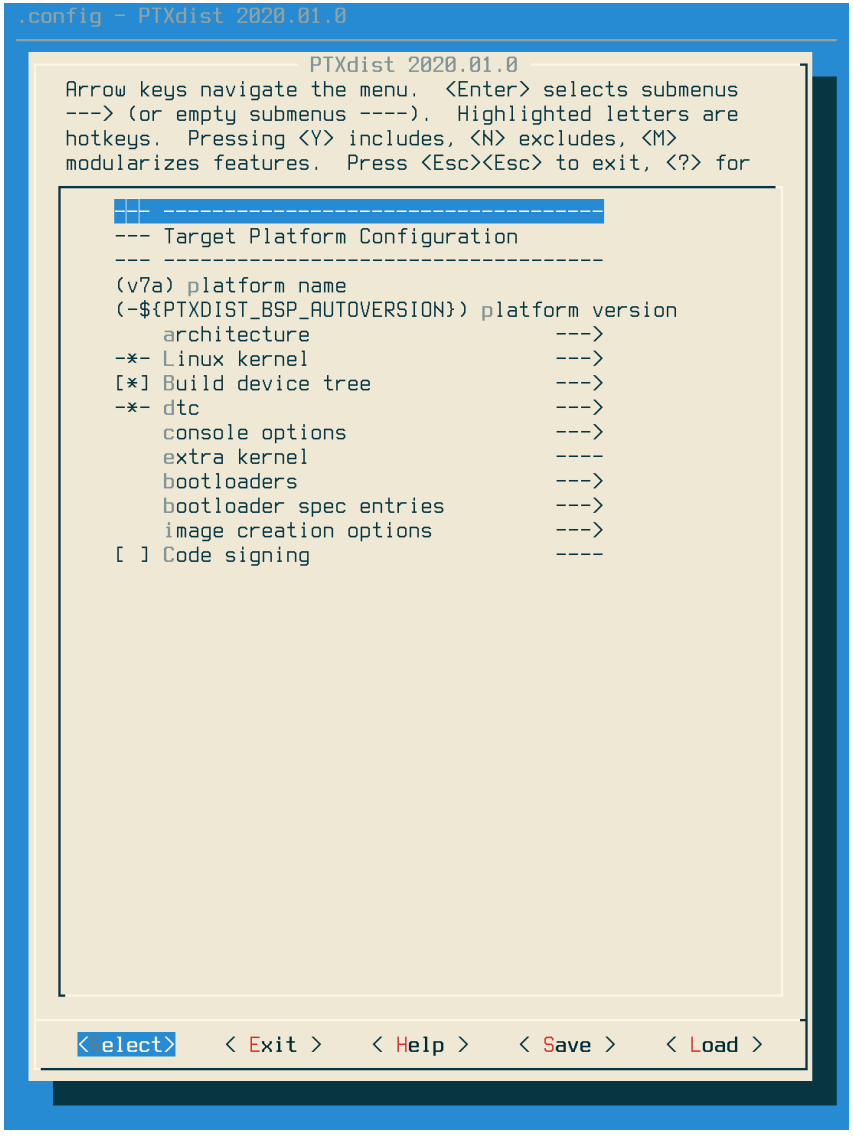
(as of PTXdist 2019.12.0)

A First Look

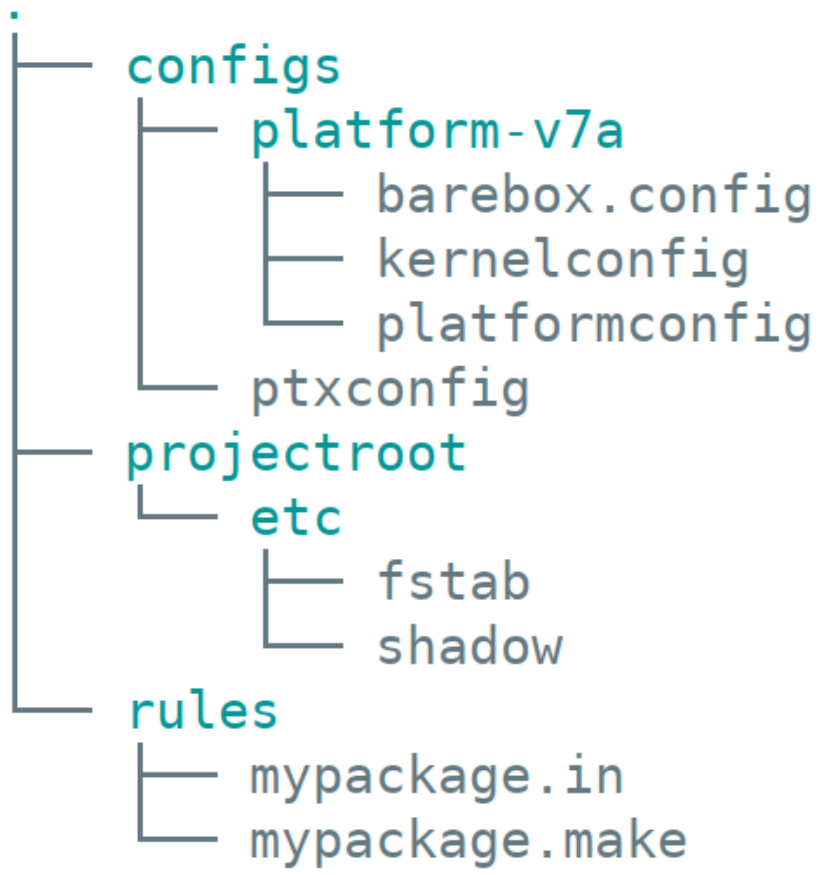
$ ptxdist menuconfig



$ ptxdist platformconfig



A Typical BSP



Idea

- BSP structure == PTXdist structure
- Extend and overwrite upstream files locally

Package Build Stages

mypackage.get

- Download upstream tarball

mypackage.extract

- Extract tarball, apply local patches

mypackage.prepare

- Run ./configure, CMake, …

mypackage.compile

- *make -j*

mypackage.install

- *make install* into isolated directory

mypackage.targetinstall

- Cherry-pick files into rootfs

Applying Patches

```
.
├── configs/platform-v7a/patches
│   └── linux-5.5
│       ├── 0001-add-frobnicated-flectrospector.patch
│       ├── 0002-fix-compilation-with-gcc-9.patch
│       └── series
└── patches
    └── busybox-1.29.3
        ├── 0001-su-c-is-not-using-bin-sh.patch
        └── series
```

- Automatically applied during the *extract* stage
- Edit patch queue with *git* or *quilt*

Package Definition: `rules/mypackage.in`

```
## SECTION=project_specific

config MYPACKAGE
    tristate
    select HOST_CMAKE
    select LIBUSB
    prompt "mypackage"
    help
      MyPackage is an example package built for FOSDEM20.
      It is built with CMake and uses libusb at runtime.
```

Package Definition: `rules/mypackage.make`

```
PACKAGES-$(PTXCONF_MYPACKAGE) += mypackage

MYPACKAGE_VERSION    := 0.1
MYPACKAGE_MD5        := 68b329da9893e34099c7d8ad5cb9c940
MYPACKAGE            := mypackage-$(MYPACKAGE_VERSION)
MYPACKAGE_SUFFIX     := tar.gz
MYPACKAGE_URL        := https://ftp.example.org/mypackage/$(MYPACKAGE).$(MYPACKAGE_SUFFIX
MYPACKAGE_SOURCE     := $(SRCDIR)/$(MYPACKAGE).$(MYPACKAGE_SUFFIX)
MYPACKAGE_DIR        := $(BUILDDIR)/$(MYPACKAGE)
MYPACKAGE_LICENSE    := 0BSD
MYPACKAGE_LICENSE_FILES := file://LICENSE;md5=60b725f10c9c85c70d97880dfe8191b3
```

Package Definition: `rules/mypackage.make`

```
# ----------------------------------------------------------------------------
# Prepare
# ----------------------------------------------------------------------------

MYPACKAGE_CONF_TOOL := cmake
MYPACKAGE_CONF_OPT  := $(CROSS_CMAKE_USR) -DUSB=ON

# ----------------------------------------------------------------------------
# Target-Install
# ----------------------------------------------------------------------------

$(STATEDIR)/mypackage.targetinstall:
    @$(call targetinfo)

    @$(call install_init, mypackage)
    @$(call install_fixup, mypackage,PRIORITY,optional)
    @$(call install_fixup, mypackage,SECTION,base)
    @$(call install_fixup, mypackage,AUTHOR,"Roland Hieber <rhi@pengutronix.de>")
    @$(call install_fixup, mypackage,DESCRIPTION,missing)

    @$(call install_copy, mypackage, 0, 0, 0755, -, )

    @$(call install_finish, mypackage)

    @$(call touch)
```

Package Types

Target packages

- Built on the build host for the target architecture
  - e.g: systemd, busybox, coreutils, kernel, bootloader

Host packages

- Built on the host, executed on the host
- Compatible build environment on different host systems
  - e.g.: host-dosfstools, host-python3, host-cmake

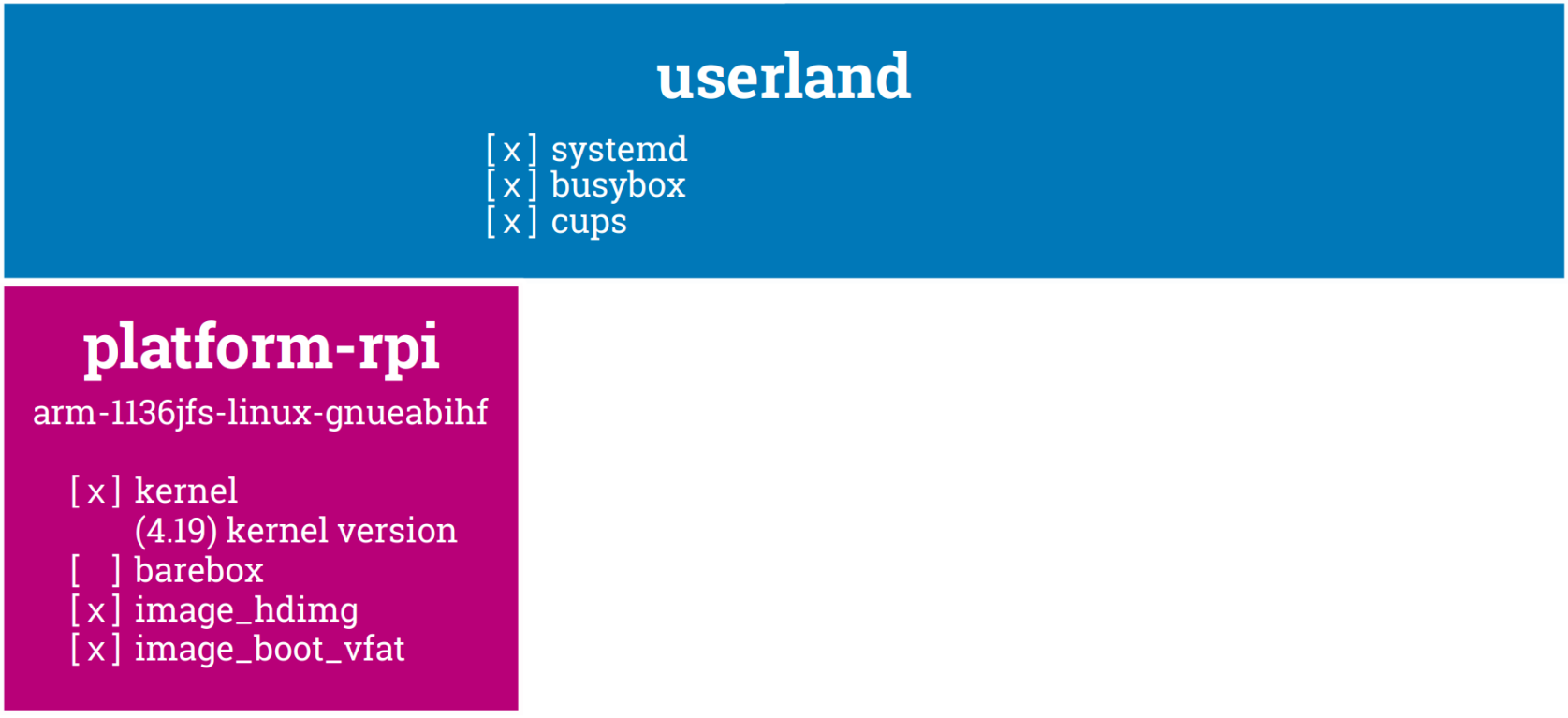
Image packages

- Determine the image format and the list of installed packages
  - e.g. hdimage, root.tgz, RAUC bundles

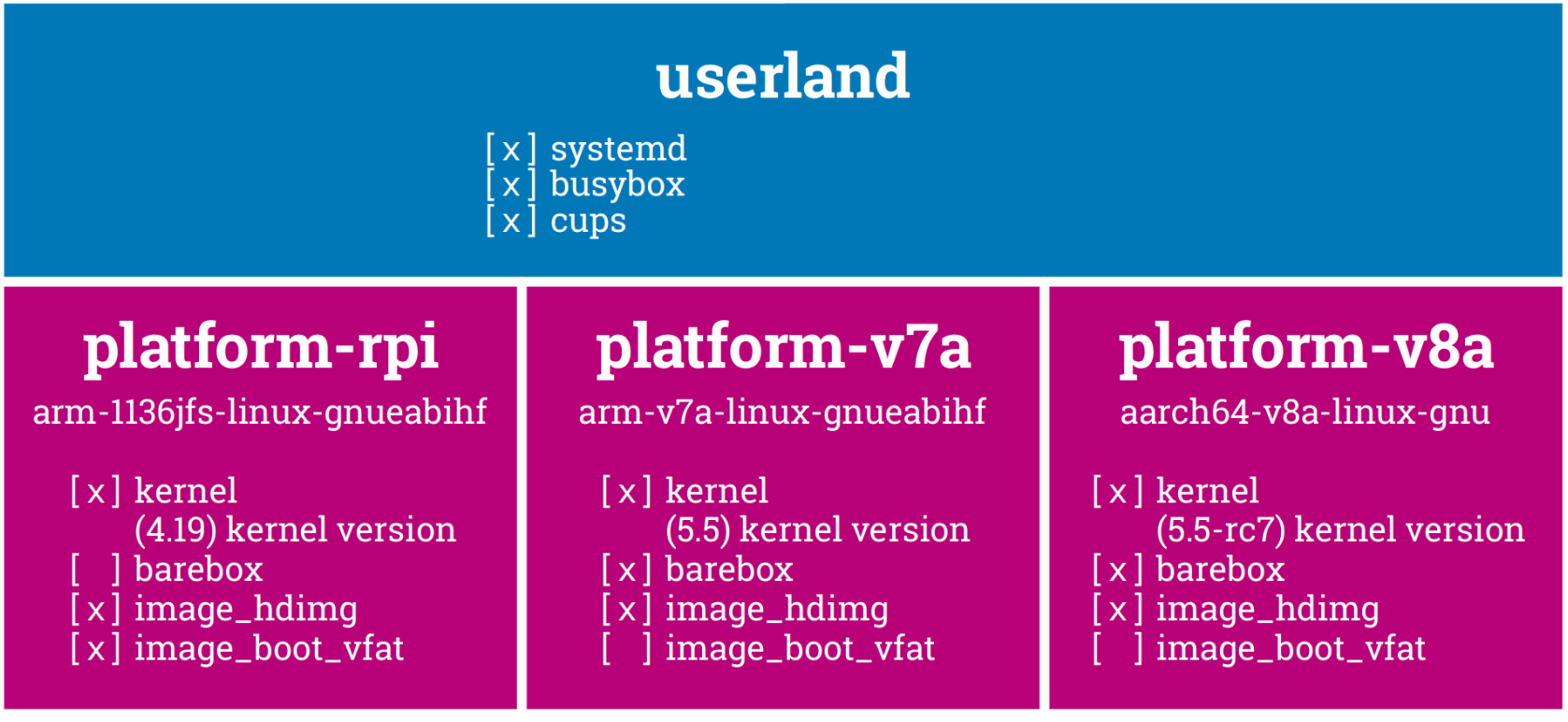
Layering Mechanisms

- Platforms
- Collections
- Alternative config files
- Kconfig diffs
- Base layers

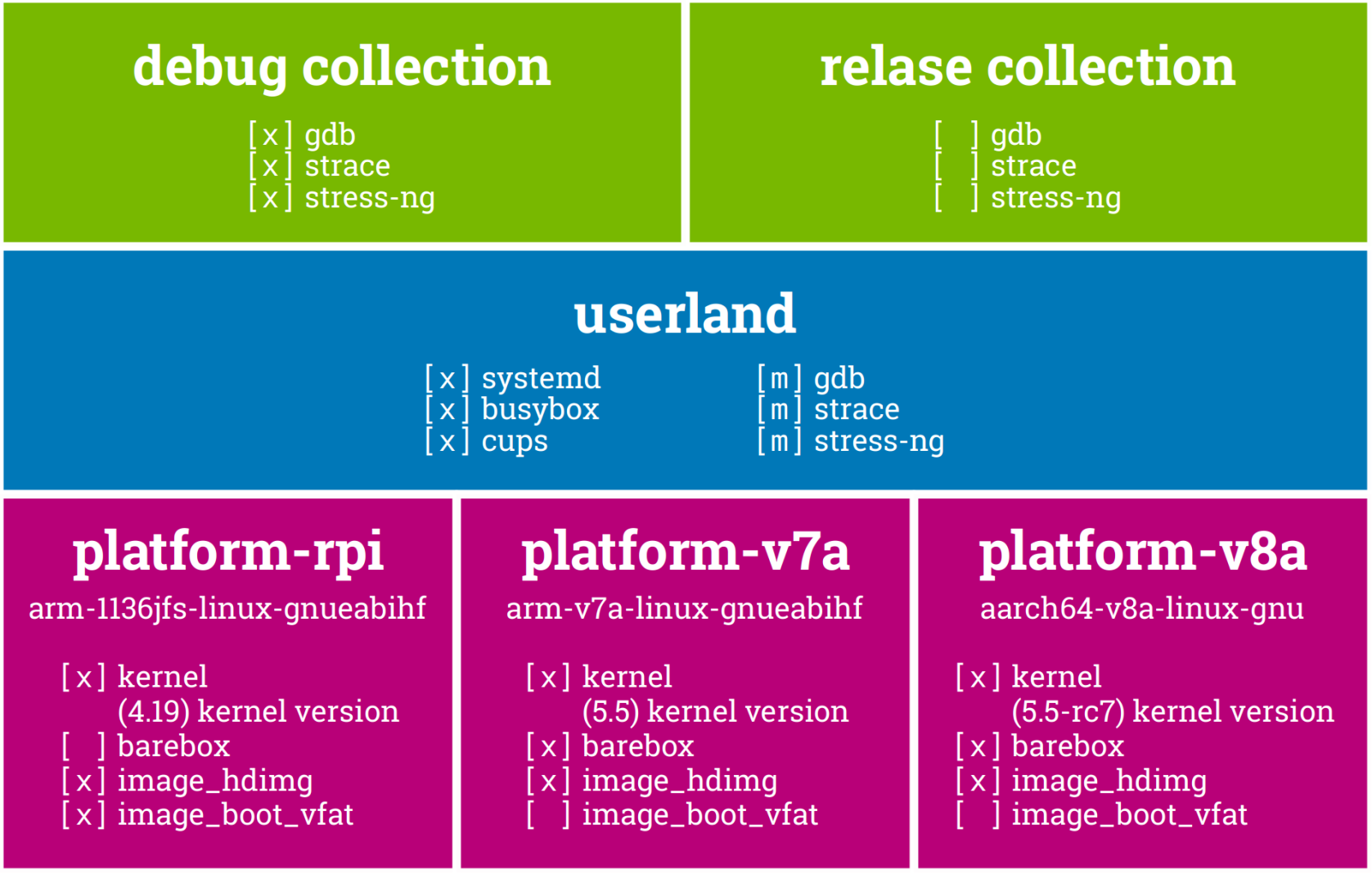
Layering: Platforms



Layering: Platforms



Layering: Collections



Layering: Alternative Config Files

```
# ptxdist/rules/cups.make
$(STATEDIR)/cups.targetinstall:
    # ...
    @$(call install_alternative, cups, daemon, lp, 0640, /etc/printcap)
```

With platform *v7a*:

1. my-bsp/projectroot.v7a/etc/printcap
2. my-bsp/projectroot/etc/printcap.v7a
3. my-bsp/configs/platform-v7a/projectroot/etc/printcap.v7a
4. my-bsp/projectroot/etc/printcap
5. my-bsp/configs/platform-v7a/projectroot/etc/printcap
6. projectroot/etc/printcap from PTXdist install dir
7. /etc/printcap from *cups* install dir
8. /etc/printcap from *cups* build dir

Layering: Kconfig Diffs

```
# configs/platform-v7a/rules/kernel-debug.make
PACKAGES-$(PTXCONF_KERNEL_DEBUG) += kernel-debug
KERNEL_DEBUG_VERSION    := 5.4
# ...
KERNEL_DEBUG_CONFIG     := $(call ptx/in-platformconfigdir, kernelconfig-debug)
KERNEL_DEBUG_REF_CONFIG := $(call ptx/in-platformconfigdir, kernelconfig)
```

```
~/projects/my-bsp $ ptxdist menuconfig kernel-debug
# [... enable debug options ... ]

~/projects/my-bsp $ ls -1 configs/platform-v7a/kernelconfig*
configs/platform-v7a/kernelconfig
configs/platform-v7a/kernelconfig-debug
configs/platform-v7a/kernelconfig-debug.diff

~/projects/my-bsp $ cat configs/platform-v7a/kernelconfig-debug.diff
1cdfdb2da785381a41fdd7320b37cd3d
CONFIG_DMA_API_DEBUG=y
CONFIG_DMA_API_DEBUG_SG=y
```

Layering: Base Layers

Remember this?:

Idea

- BSP structure == PTXdist structure
- Extend and overwrite upstream files locally

Layering: Base Layers

Remember this?:

Idea

- BSP structure == PTXdist structure
- Extend and overwrite upstream files locally

Better Idea

- “All problems in computer science can be solved by another level of indirection.” (David Wheeler)

Layering: Base Layers

```
distrokit/
├── configs
│   ├── platform-v7a
│   │   ├── kernelconfig
│   │   └── platformconfig
│   └── ptxconfig
└── rules
    ├── datapartition.in
    └── datapartition.make

my-bsp/
├── base -> ../distrokit/
├── configs
│   ├── ptxconfig
│   └── ptxconfig.diff
└── rules
    ├── mypackage.in
    └── mypackage.make
```

Goodies: Package Info

```
~/projects/my-bsp $ ptxdist package-info coreutils

------------------------------
target: coreutils.package-info
------------------------------

package:      coreutils
version:      8.31

license:      GPL-3.0-or-later
  files:      file://COPYING;md5=d32239bcb673463ab874e80d47fae504

source:       /usr/src/coreutils-8.31.tar.xz
md5:          0009a224d8e288e8ec406ef0161f9293
url:          http://ftp.uni-kl.de/pub/gnu/coreutils/coreutils-8.31.tar.xz

src dir:      my-bsp/platform-v7a/build-target/coreutils-8.31
build dir:    my-bsp/platform-v7a/build-target/coreutils-8.31
pkg dir:      my-bsp/platform-v7a/packages/coreutils-8.31

rule file:    ptxdist-2020.01.0/rules/coreutils.make

finished target coreutils.package-info
```

Goodies: BSP Info

```
~/projects/my-bsp $ ptxdist bsp-info

----------------
target: bsp-info
----------------

vendor:           ACME
project:          MyBSP
version:          -

platform:         v7a
platform version: -

BSP:              /home/rohieb/projects/my-bsp
PTXdist:          /usr/local/lib/ptxdist-2020.01.0

ptxconfig:        my-bsp/configs/ptxconfig
platformconfig:   my-bsp/configs/platform-v7a/platformconfig

images:           image-hdimg
                  image-kernel
                  image-root-ext
                  image-root-tgz

finished target bsp-info
```

Goodies: License Report

```
$ ptxdist make license-report
```

Contents

20 dosfstools **42**
20.1 COPYING . . . 42
20.2 src_fsck.fat.c . . . 54

21 e2fsprogs **55**
21.1 NOTICE . . . 55
21.2 lib_uuid_gen_uuid.c . . . 71

22 expat **72**
22.1 COPYING[automatically found] . . . 72

23 gcclibs **73**
23.1 COPYING3 . . . 73
23.2 COPYING.RUNTIME . . . 85

24 gdbserver **87**
24.1 COPYING . . . 87
24.2 COPYING3 . . . 93
24.3 COPYING.LIB . . . 105
24.4 COPYING3.LIB . . . 114

25 glib **118**
25.1 COPYING[automatically found] . . . 118

26 glibc **128**
26.1 COPYING . . . 128
26.2 COPYING.LIB . . . 134
26.3 LICENSES . . . 143

21 e2fsprogs

Package: e2fsprogs 1.44.4

License: GPL-2.0-or-later AND LGPL-2.0-or-later AND BSD-3-Clause AND MIT

Flags: attribution

URL: http://downloads.sourceforge.net/sourceforge/e2fsprogs/e2fsprogs/v1.44.4/e2fsprogs-1.44.4.tar.gz

MD5: `156e94a6169ca1fa3f0c6749ae5921b9`

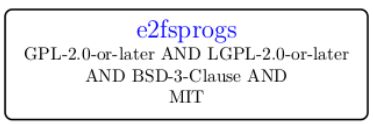


Figure 21.1: Dependency tree for e2fsprogs

21.1 NOTICE

```
This package, the EXT2 filesystem utilities, are made available under
the GNU Public License version 2, with the exception of the lib/ext2fs
and lib/e2p libraries, which are made available under the GNU Library
General Public License Version 2, the lib/uuid library which is made
available under a BSD-style license and the lib/et and lib/ss
libraries which are made available under an MIT-style license.  Please
see lib/uuid/COPYING for more details for the license for the files
comprising the libuuid library, and the source file headers of the
libet and libss libraries for more information.
```

Trying it all out

DistroKit

- preconfigured BSP for a variety of dev boards (and qemu-arm)
- https://git.pengutronix.de/cgit/DistroKit

Contributing

- Currently: ~85% Pengutronix contributions…
- Patches welcome:
  https://www.ptxdist.org/doc/contributing.html

Contributing

- Currently: ~85% Pengutronix contributions…
- Patches welcome:
  https://www.ptxdist.org/doc/contributing.html

Questions?