Fedora on RISC-V

Status and practice 傅炜

Wei Fu <wefu@redhat.com>
Senior Software Engineer
Platform Enablement, Red Hat Software (Beijing) Co.,Ltd.

Nov 13th 2019
The 1st China RISC-V forum

AGENDA

Introduction

Fedora on RISC-V

- History
- Facility
- Status
- Supported Targets

Tools

RISC-V Development on Fedora

- Toolchain
- QEMU
- VM Tools

HowTo

Fedora Image in practice

- OpenSBI
- U-Boot
- Linux kernel
- Fedora Image

Part One

Fedora on RISC-V

History

Facility

Status

Supported Targets

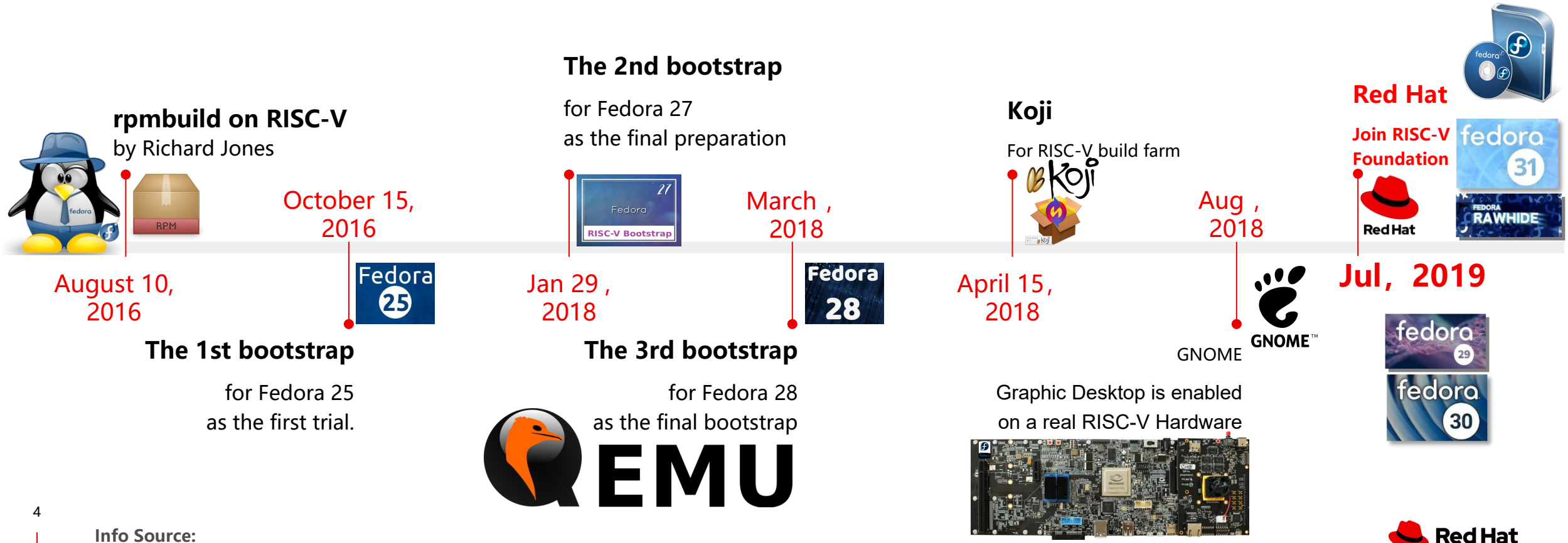
Red Hat

History

Fedora on RISC-V History

Since Fedora has an **upstream first policy** and it also applies to Fedora/RISC-V.

We need all the key patchsets for **toolchain**, **Linux kernel** and **glibc** to be merged, then we can do the final **bootstrap** on RISC-V.

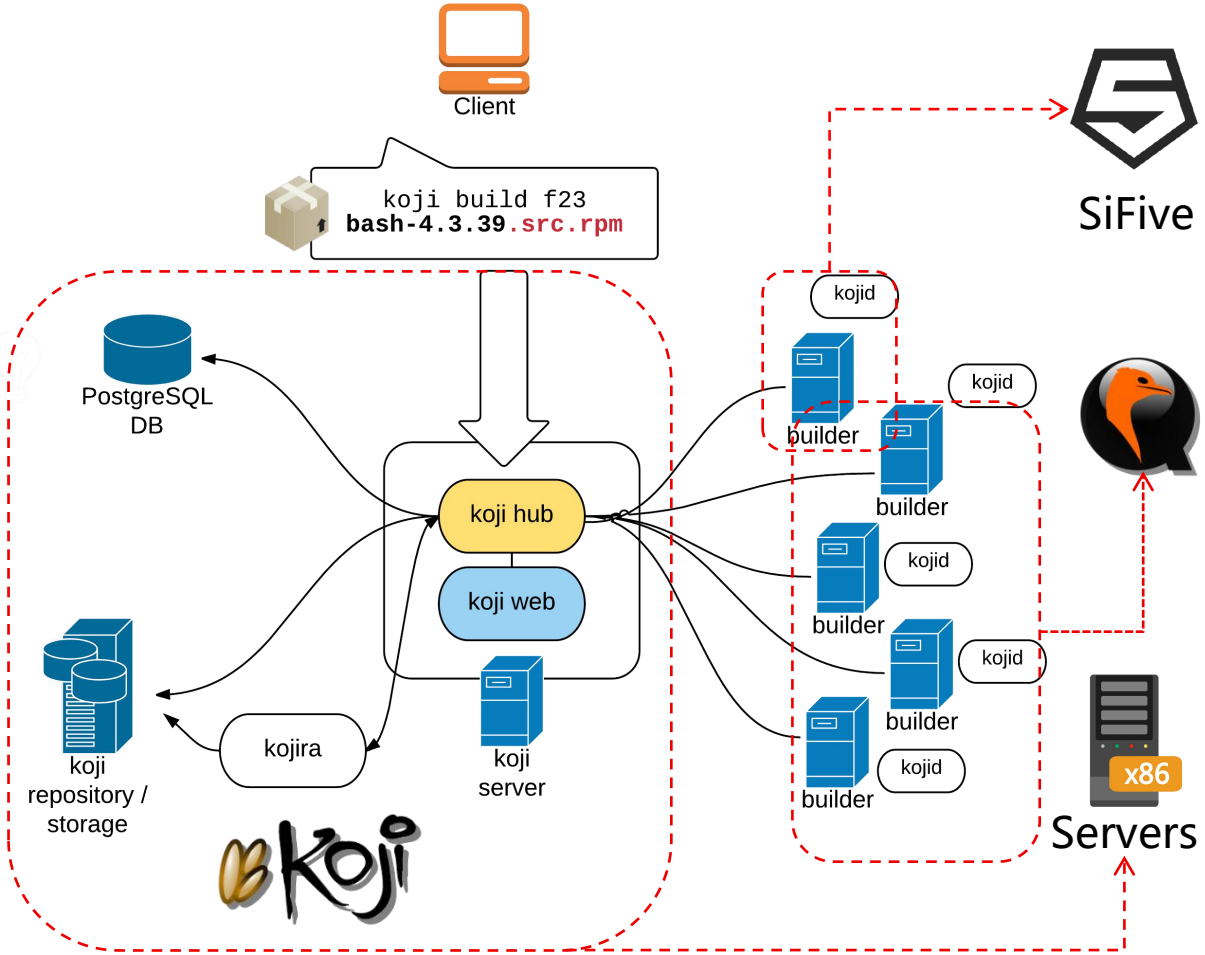


Info Source:

Most of info comes from Richard Jones and his weblog: https://rwmj.wordpress.com/

Facility: Koji Build System

Koji builds RPMs for the Fedora Project and EPEL.



3 HiFive Unleasheds

One of them connects with SSD.

142 QEMU VMs(on x86_64)

fedora-riscv-x.gcc1xx.osuosl.org

managed by libvirt

(will add more by adding more servers)

An x86_64 server for all central infrastructure

Main sever, repository creation and VMs with backup(separate NVMe).

These Koji servers for RISC-V have been moved to the hardware supplied by SiFive and WD at Fremont.

This picture is from Glaser Lo's blog

Koji build system overview

http://gklo.github.io/open-source/koji-build-systems-overview/

Red Hat

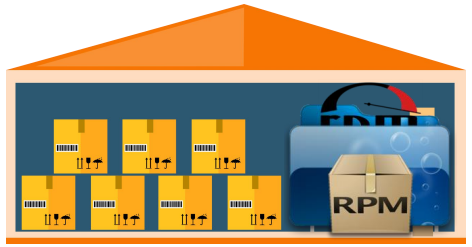
Status: Packages

Fedora for RISC-V is mirrored as
a Fedora “**alternative**” Architecture

Active projects:

Fedora 32/Rawhide

Fedora 31



Repositories

https://dl.fedoraproject.org/pub/alt/risc-v/

https://**mirror**.math.princeton.edu/pub/alt/risc-v/

https://isrc.iscas.ac.cn/mirror/fedora-riscv/

Wanna help to make a mirror?
https://fedoraproject.org/wiki/Infrastructure/Mirroring

The Koji Build System

All kinds of packages are building here, including debuginfo, debugsource and source packages.

Before we got some decent RISC-V build systems, we couldn't make RISC-V become an official Alt architecture.

Status: Images

Koji is building 3 types of disk image

Fedora Nano

smaller than Minimal, @core, kernel and no docs

Fedora Minimal

just include @core, @buildsys-build, kernel.

Fedora Developer

has extra packages installed for developers, all common editors, X11, a few small WMs, RPM tools, building tools, koji stuff, etc.

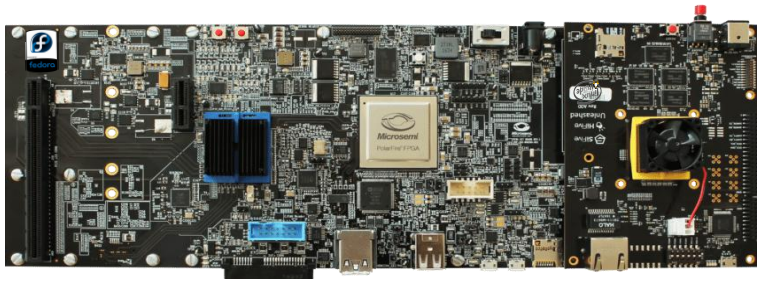
Fedora GNOME

Developer with GNOME desktop GUI support.

Supported Targets

Virtual: QEMU and libvirt/QEMU

Fedora Images can run on the libvirt/QEMU with graphics parameters (Spice).



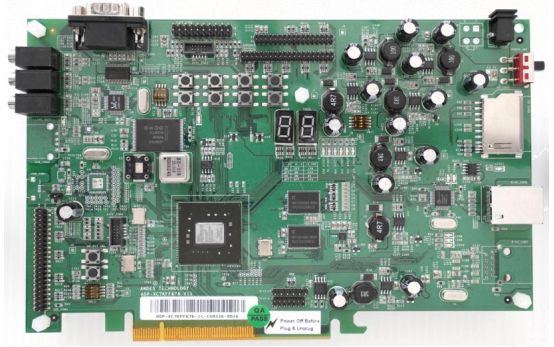
SiFive Unleashed

Fedora GNOME Image can run on SiFive Unleashed(with Expansion Board, PCI-E graphic Card & SATA SSD.)

Red Hat

Tested Targets

ANDES TECHNOLOGY



ICT

中国科学院计算技术研究所
INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES

QEMU for AndeStar V5 && AndeShape Development Platform ADP-XC7KFF676

Fedora Images can run on the QEMU and AndeShape FPGA board

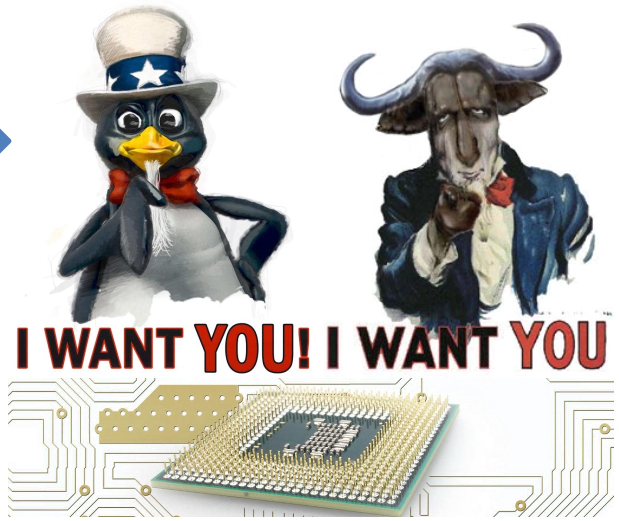
LIBERTAD AMIGOS RASGOS PRIMERO

ICT Development Platform

Fedora Developer Image can run on ICT FPGA Cloud development platform (with PCI-E SSD and graphic Card)

Red Hat

Fedora on RISC-V



From www.codasip.com

We would like to support more targets based on standard RISC-V Spec.

Red Hat

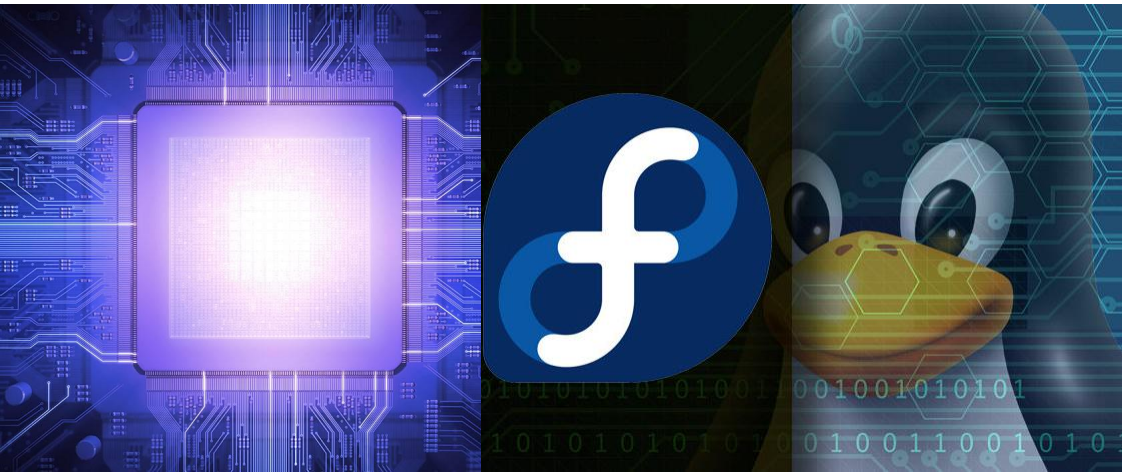
Part Two

RISC-V Development on Fedora

Toolchain

QEMU

VM Tools



Please try Fedora 31 for RISC-V development

Red Hat

Toolchain

Cross compiler for RV64:

Since Fedora 29, you can just:

“sudo dnf install gcc-riscv64-linux-gnu”

you can get the relative package list by

“dnf list *-riscv*”

Native compiler for RV64:

"Fedora Developer" Image has extra packages installed for developers, including RPM tools, building tools, koji stuff, etc.

You can use them just like on X86 machine.

For Building RPM packages and Fedora Images, we only use **native compilation**.

QEMU

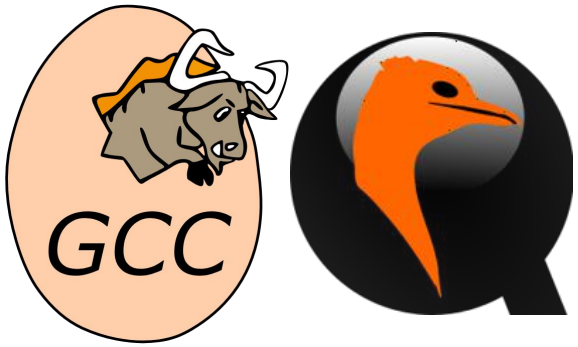
QEMU RPM for RISC-V

Since Fedora 29, you can just:

"sudo dnf install qemu-system-riscv"

But please install the latest version of them by

"sudo dnf copr enable @virtmaint-sig/virt-preview"



Build QEMU from source code

The upstream QEMU has supported most of latest RISC-V spec and can work with latest software for RISC-V.

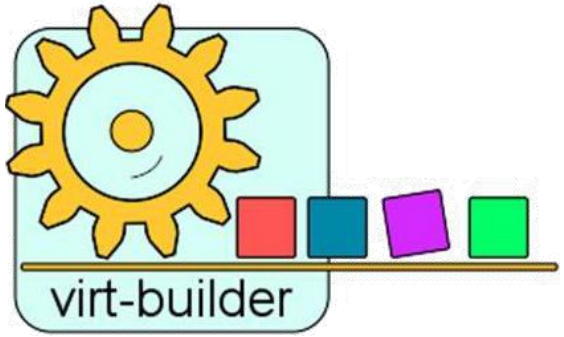
If you use dated QEMU, it will be incompatible with the latest RISC-V Software.

VM Tools

The libvirt project:

a toolkit to manage virtualization platforms, like creating new KVM, list the supported operating system variants, and start/stop/remove a VM.

sudo dnf install virt-manager libvirt



Fedora virt-builder:

You can quickly and easily build new virtual machines to practice Fedora on RISC-V .

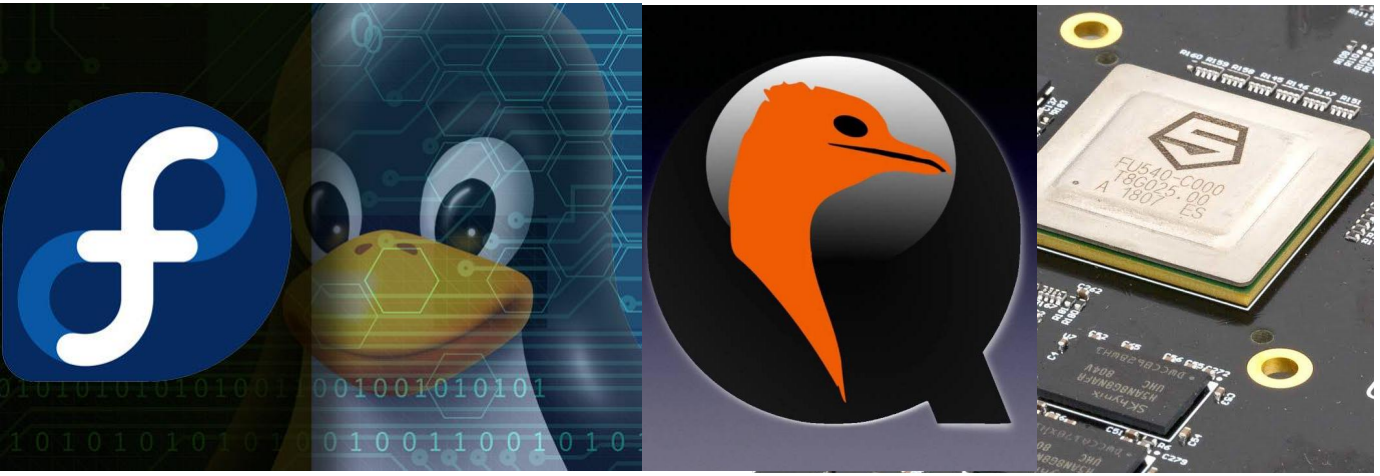
sudo dnf install libguestfs-tools-c

https://dl.fedoraproject.org/pub/alt/risc-v/repo/virt-builder-images/images/

Red Hat

Part Three

Fedora Image in practice



OpenSBI

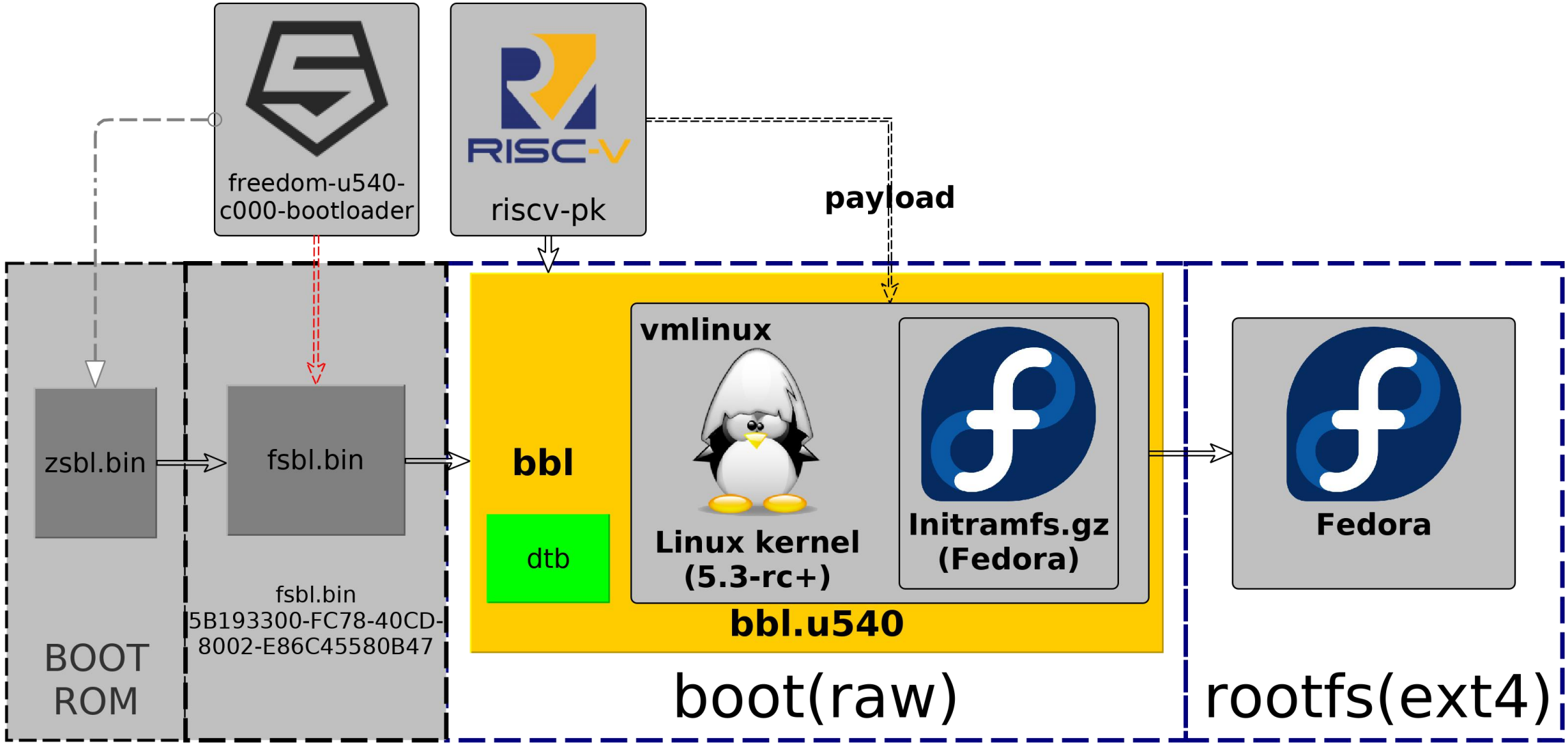
U-Boot

Linux kernel

Fedora Image

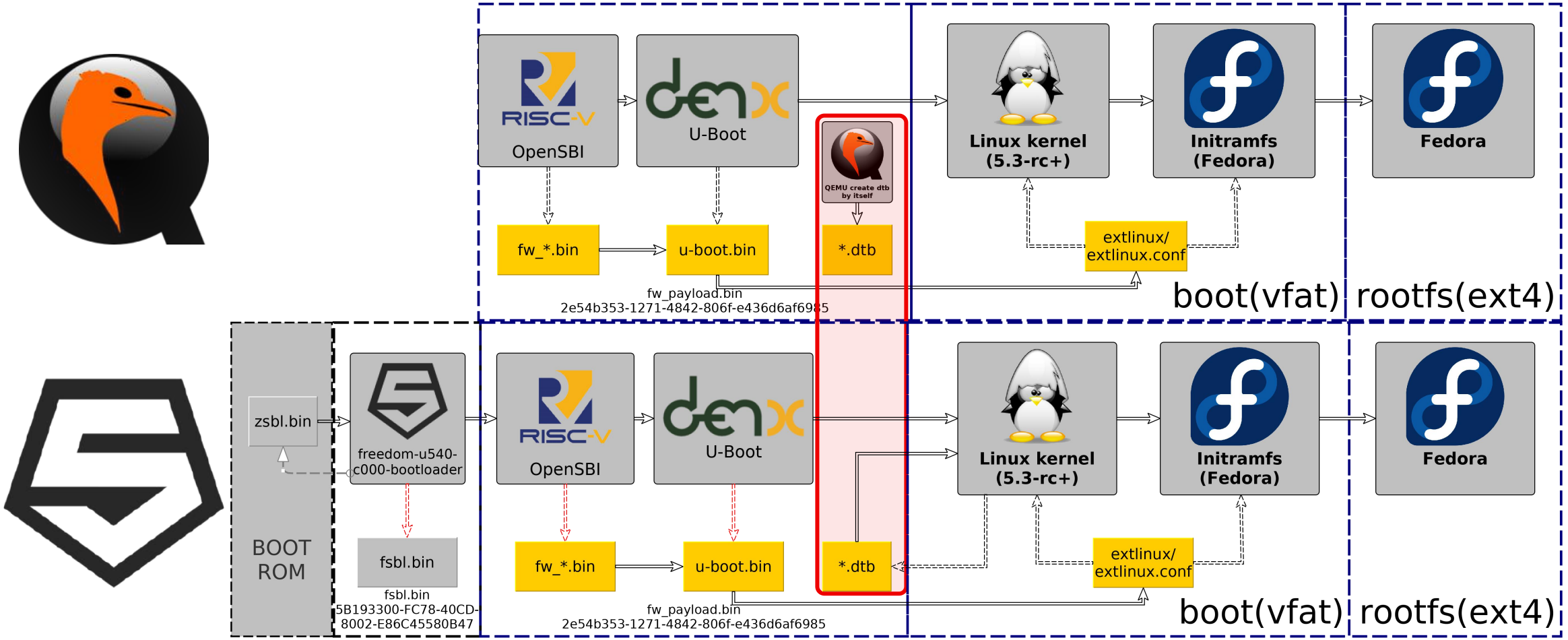
Red Hat

The dated boot flow for Fedora on RISC-V



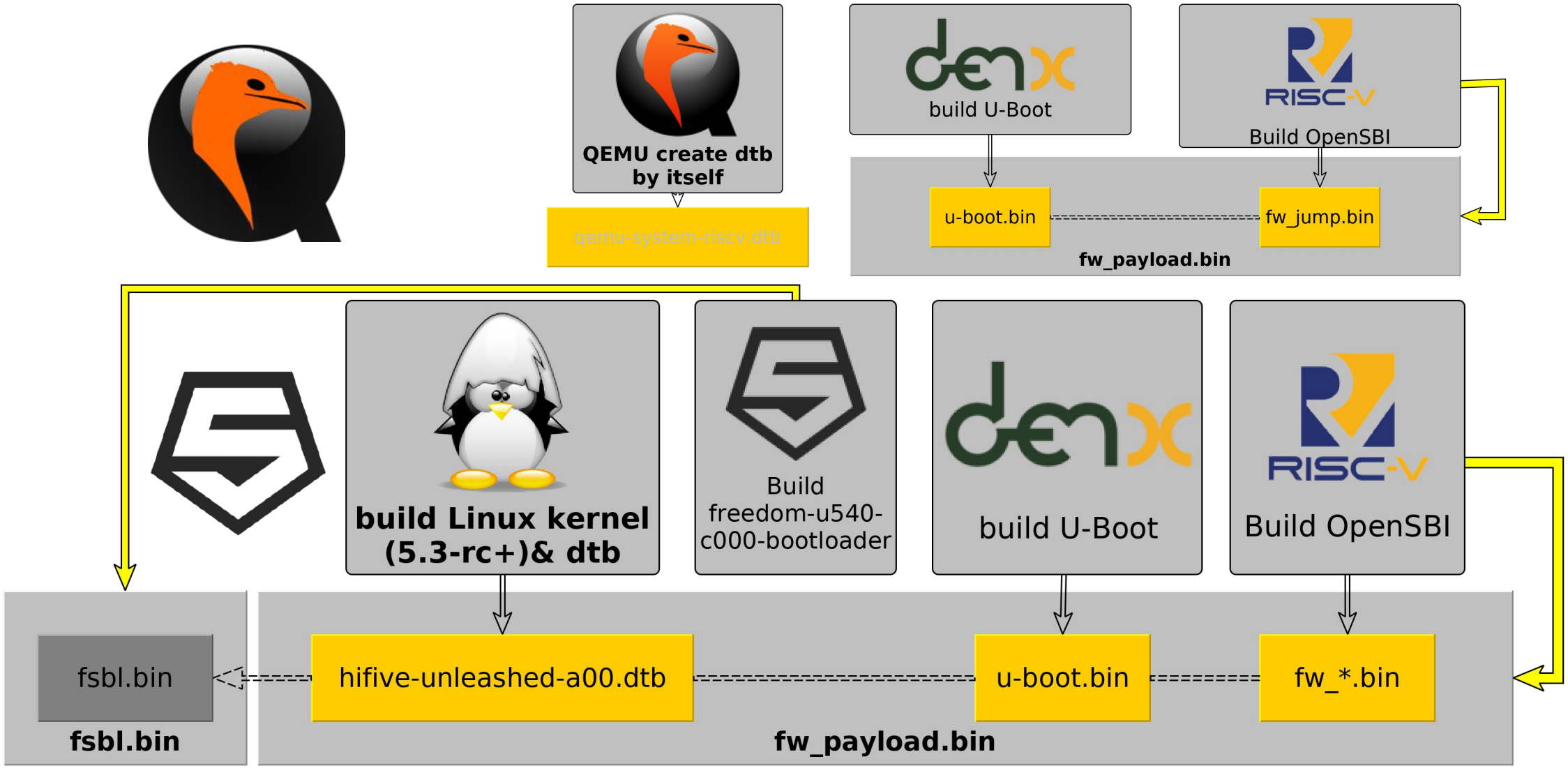
Red Hat

The current boot flow for Fedora on RISC-V



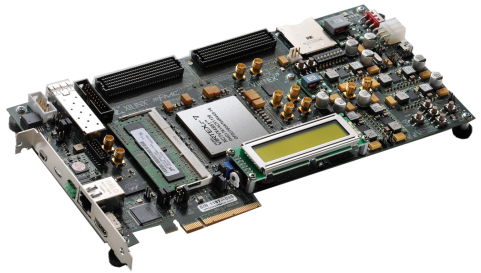
Red Hat

The current Build flow for firmware on RISC-V



Red Hat

The new progress of UEFI on RISC-V



tianocore

- Last year, HPE engineers have made Tianocore successfully boot on SiFive Freedom U500 VC707 FPGA Dev Kit with OpenSBI integrated in edk2 RISC-V port.
- Then they were busy on standardizing firmware spec: SMBIOS 3.3.0 already released with new record type (type 44) added, CIM works were done as well with RISC-V processor definitions.

DMTF

- HPE has posted their **V3** patchset for review.
- For Now, with V3 patchset, EDK2(+ OpenSBI) can run on QEMU(>V4.1.5, -machine sifive_u -cpu sifive-u54) and **Real Hardware SiFive Unleashed.**

Hewlett Packard Enterprise

Red Hat

The Firmwares on RISC-V

keep updating below specs to reflect the latest RISC-V specs.

- UEFI spec
- Platform Initialization spec

Also working on below specs:

- ACPI tables for RISC-V processor
- Evaluate the works done in RISC-V TEE WG for drafting EFI Management Mode spec of RISC-V processor.

Acknowledgments

Hewlett Packard Enterprise

Abner Chang
Gilbert Chen

Red Hat

Al Stone
Andrea Bolognani
Charles Wei
DJ Delorie
John Feeney
Richard Jones

SiFive

David Abdurachmanov

WD

Alistair Francis
Anup Patel
Atish Kumar Patra

Mikael Frykholm
Stefan O'Rear

TBSI 清华-伯克利深圳研究院 Tsinghua-Berkeley Shenzhen Institute

RISC-V 中国委员会

RVA 中国开放指令生态（RISC-V）联盟 China RISC-V Alliance

ICT 中国科学院计算技术研究所 INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES

facebook

TRANQUILLITY

OSL OPEN SOURCE LAB

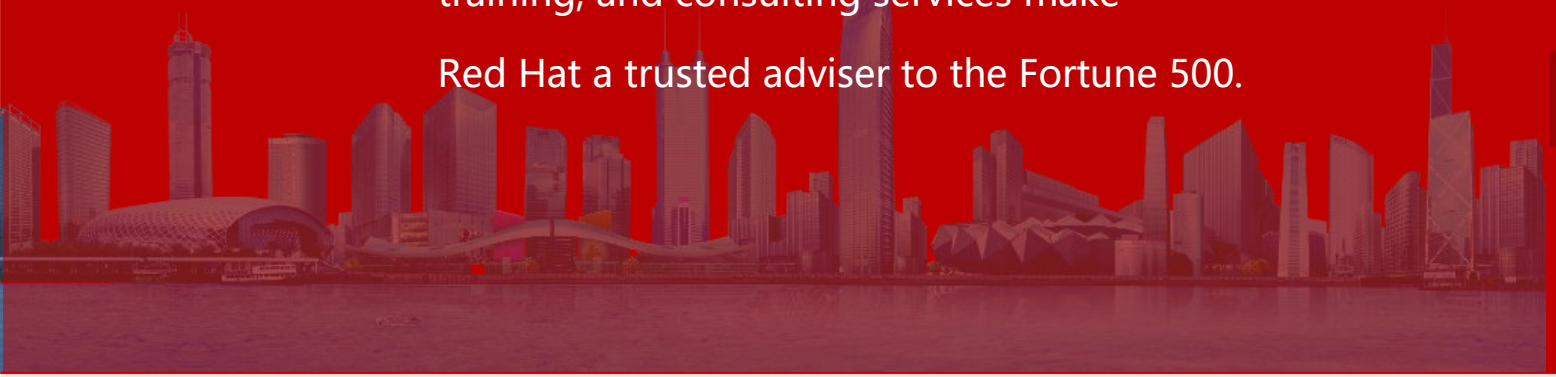
Academic Computer Club Umeå University

... and countless other individuals and companies, who have contributed to RISC-V specifications and software eco-system!

Red Hat

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.



linkedin.com/company/red-hat

youtube.com/user/RedHatVideos

facebook.com/redhatinc

twitter.com/RedHat

Red Hat

FYI

Steps to build firmware(OpenSBI/U-boot) for Fedora Image on RISC-V platform

Red Hat

Development Info:

IRC
#fedora-riscv (FreeNode)

Fedora Wiki pages For RISC-V

- **Main Entrance:**
  https://fedoraproject.org/wiki/Architectures/RISC-V
- **Instruction of installation:**
  https://fedoraproject.org/wiki/Architectures/RISC-V/Installing

Fedora Main REPO for RISC-V:
https://dl.fedoraproject.org/pub/alt/risc-v/

Koji for RISC-V:
Domain Name: fedora.riscv.rocks

- **Nightly build images:** http://fedora.riscv.rocks/koji/**tasks**?order=-completion_time&state=closed&view=flat&**method=createAppliance**
- **dist-repos:** http://fedora.riscv.rocks/repos-dist/
- **SCM:** http://fedora.riscv.rocks:3000/

QEMU：u-boot.bin & fw_payload.bin

U-boot:
git://git.denx.de/u-boot.git

make qemu-riscv64_smode_defconfig

make

<u-boot>/u-boot.bin

OpenSBI:
https://github.com/riscv/opensbi.git

make PLATFORM=qemu/virt \
FW_PAYLOAD_PATH=<u-boot_source>/u-boot.bin

<opensbi>/build/platform/qemu/virt/firmware/fw_payload.bin

Cross compiler:

ARCH=riscv CROSS_COMPILE=riscv64-linux-gnu-

Test on QEMU

QEMU

qemu-system-riscv64 \

-smp 8 -m 2G -machine virt -nographic \

-bios fw_payload.bin \

**-device virtio-blk-device,drive=hd0 **

-drive file=Fedora-Developer-Rawhide-20191030.n.0-sda.raw,format=raw,id=hd0 \

-object rng-random,filename=/dev/urandom,id=rng0 \

-device virtio-rng-device,rng=rng0 \

-device virtio-net-device,netdev=usernet \

-**netdev** tap,id=usernet,ifname=tap0,script=no,downscript=no \

-serial telnet:localhost:7000,server

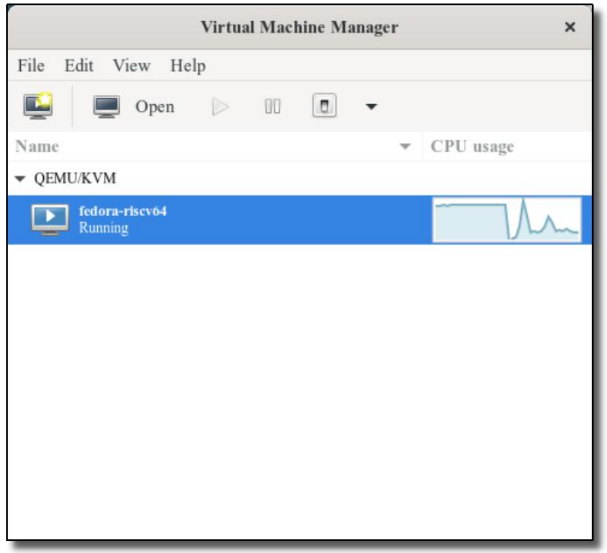
Please set up the network on your host machine correctly for "-netdev"

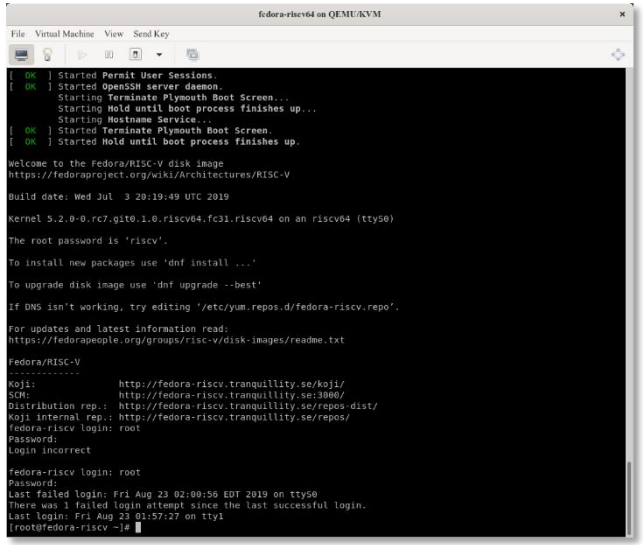
Test with Libvirt

Libvirt

virt-install --name fedora-riscv64 --arch riscv64 --vcpus 8 --memory 4096 \
--os-variant fedora30 \
--boot **loader=/var/lib/libvirt/images/fw_payload.bin** \
--import **--disk path=/var/lib/libvirt/images/Fedora-Developer-Rawhide-20191030.n.0-sda.raw** \
--network network=default \
--graphics spice

virt-manager





Please copy the firmware and image to the **right directory** and set up the **correct permission** of these files

Red Hat

HiFive Unleashed：u-boot.bin & hifive-unleashed-a00.dtb

DTB

#in Linux kernel tree (5.3-rc+)

make defconfig

make dtbs

arch/riscv/boot/dts/sifive/hifive-unleashed-a00.dtb

freedom-u540-c000-bootloader(Native build on QEMU, currently)

https://github.com/sifive/freedom-u540-c000-bootloader

make CROSSCOMPILE=

/fsbl.bin

U-boot

make sifive_fu540_defconfig

make

/u-boot.bin

Cross compiler:

ARCH=riscv CROSS_COMPILE=riscv64-linux-gnu-

HiFive Unleashed：fw_payload.bin

OpenSBI

make PLATFORM=**sifive/fu540** \
FW_PAYLOAD_PATH=u-boot.bin \
FW_PAYLOAD_FDT_PATH=<linux source>/arch/riscv/boot/dts/sifive/hifive-unleashed-a00.dtb

/build/platform/sifive/fu540/firmware/fw_payload.bin

Cross compiler:
ARCH=riscv CROSS_COMPILE=riscv64-linux-gnu-

HiFive Unleashed: Flash into uSD(fsbl/u-boot)


```
unzstd        Fedora-*$BUILD_DATE.n.0-sda.raw.zst
guestfish -a Fedora-*$BUILD_DATE.n.0-sda.raw run : download /dev/sda1 boot.raw
guestfish -a Fedora-*$BUILD_DATE.n.0-sda.raw run : download /dev/sda2 rootfs.raw
```

dd if=fsbl.bin \
of=/dev/sdx2

**dd if=fw_payload.bin \
of=/dev/sdx3 bs=1024**

```
sgdisk --clear                                                                    \
  --new 1::+500M --typecode=1:0FC63DAF-8483-4772-8E79-3D69D8477DE4 --change-name=1:'_/boot' \
  --new 2::+32K --typecode=2:5B193300-FC78-40CD-8002-E86C45580B47    --change-name=2:'_/fsbl' \
  --new 3::+8M --typecode=3:2E54B353-1271-4842-806F-E436D6AF6985     --change-name=3:'_/opensbi-uboot' \
  --new 4::-0 --typecode=4:0FC63DAF-8483-4772-8E79-3D69D8477DE4      --change-name=4:'_/' \
      ${DISK}
```

SiFive U540：EDK2 Soruce

EDK2

REPO: https://github.com/changab/edk2-staging-riscv

branch: RISC-V-V2-v3

edk2-platform(in edk2 dir)

REPO: https://github.com/gilbert225/edk2-platforms

branch: devel-riscv-v2-PATCHv5

patch for serial port baudrate:

https://github.com/tekkamanninja/edk2-platforms/commits/RISC-V_TN

SiFive U540: EDK2 build procedure

- The crosstools in Fedora doesn't work on this code, we need a special version of gcc:
  REPO: https://github.com/riscv/riscv-gnu-toolchain
  Commit: 64879b24

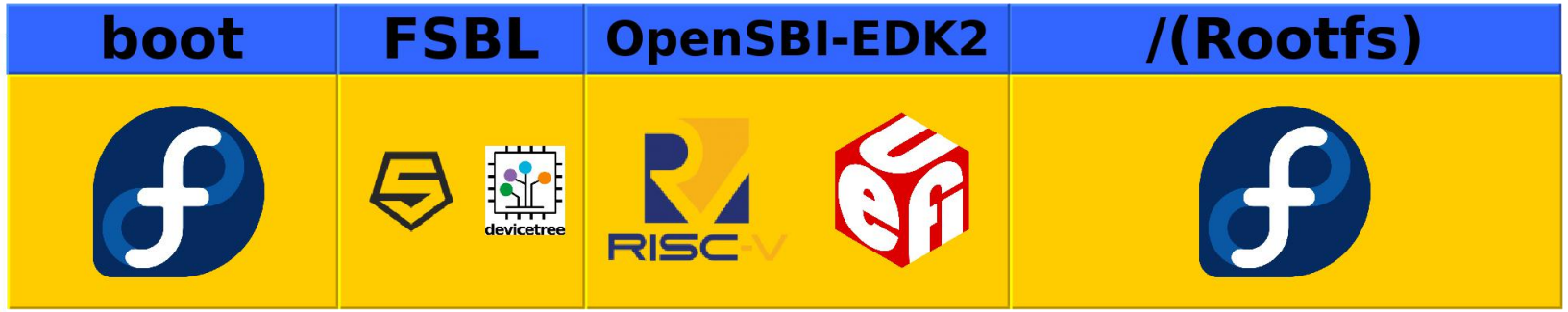
- **Build commands:**

```
cd $(UEFI_SRC_DIR)
git submodule init ;  git submodule update
#make sure that you got opensbi submodule
export PATH=$(CROSS_TOOL_DIR_RV64):${PATH}
export GCC5_RISCV64_PREFIX=riscv64-unknown-linux-gnu-
source ./edksetup.sh --reconfig
make -C BaseTools/
build -a RISCV64 -t GCC5  \
-p Platform/SiFive/U5SeriesPkg/FreedomU540HiFiveUnleashedBoard/U540.dsc
```

HiFive Unleashed: Flash into uSD(fsbl/edk2)

QEMU

qemu-system-riscv64 -cpu sifive-u54 -smp cpus=5,maxcpus=5 -m 4096 -machine sifive_u -nographic -bios **U540.fd** -serial telnet:localhost:7000,server



dd if=fsbl.bin \
of=/dev/sdx2

**dd if=U540.fd \
of=/dev/sdx3 bs=1024**

```
sgdisk --clear                                                    \
  --new 1::+1G   --typecode=1:0FC63DAF-8483-4772-8E79-3D69D8477DE4   --change-name=1:'_/boot'      \
  --new 2::+32K --typecode=2:5B193300-FC78-40CD-8002-E86C45580B47    --change-name=2:'_/fsbl'      \
  --new 3::+16M --typecode=3:2E54B353-1271-4842-806F-E436D6AF6985    --change-name=3:'_/opensbi-edk2' \
  --new 4::-0      --typecode=4:0FC63DAF-8483-4772-8E79-3D69D8477DE4   --change-name=4:'_/'      \
     ${DISK}
```