



Fedora on RISC-V

Status and practice

Wei Fu <wefu@redhat.com>

Senior Software Engineer

Platform Enablement, Red Hat Software (Beijing) Co.,Ltd.

Nov 5th 2019

SiFive China Tech Day * 杭州

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

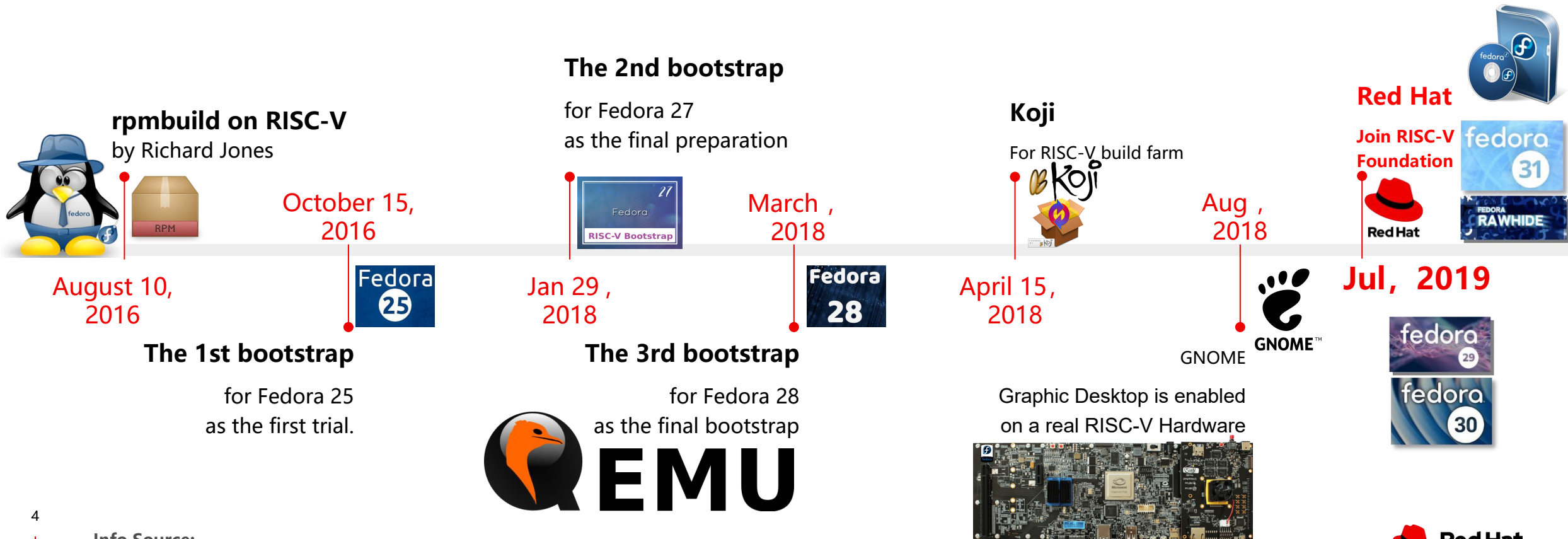
Supported Targets

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.

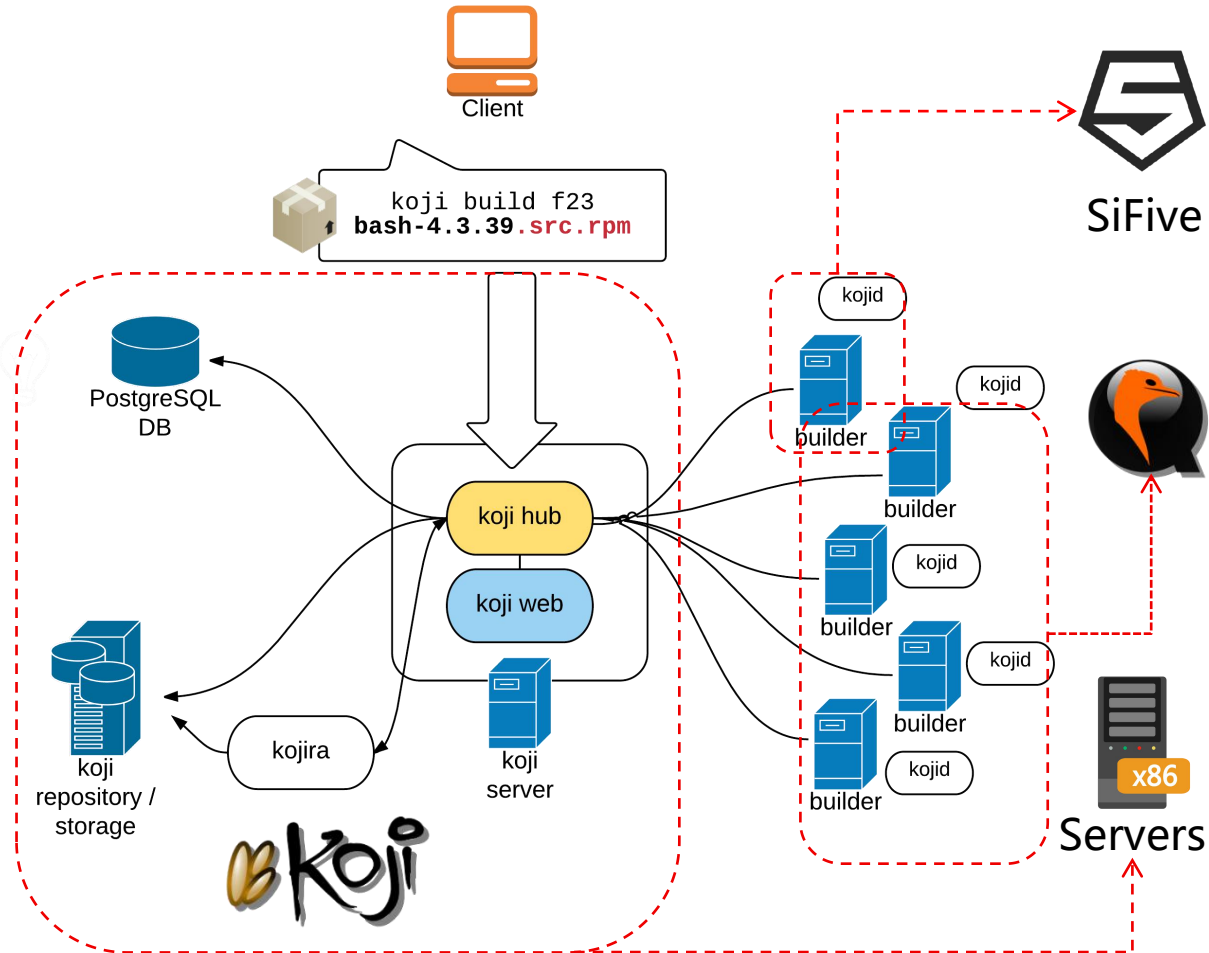


Info Source:

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).



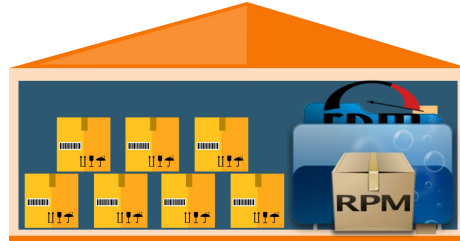
Status: Packages

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



Active projects:

Fedora 32/Rawhide
Fedora 31



Repositories



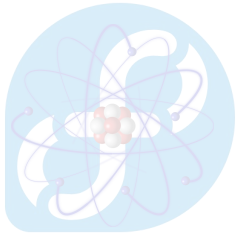
<https://dl.fedoraproject.org/pub/alt/risc-v/>
<https://mirror.math.princeton.edu/pub/alt/risc-v/>
<https://isrc.iscas.ac.cn/mirror/fedora-riscv/>

The Koji Build System

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

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

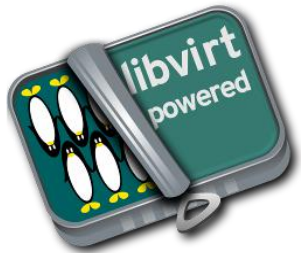
has extra packages
installed for developers,
all common editors, X11,
a few small WMs, RPM
tools, building tools, koji
stuff, etc.



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).

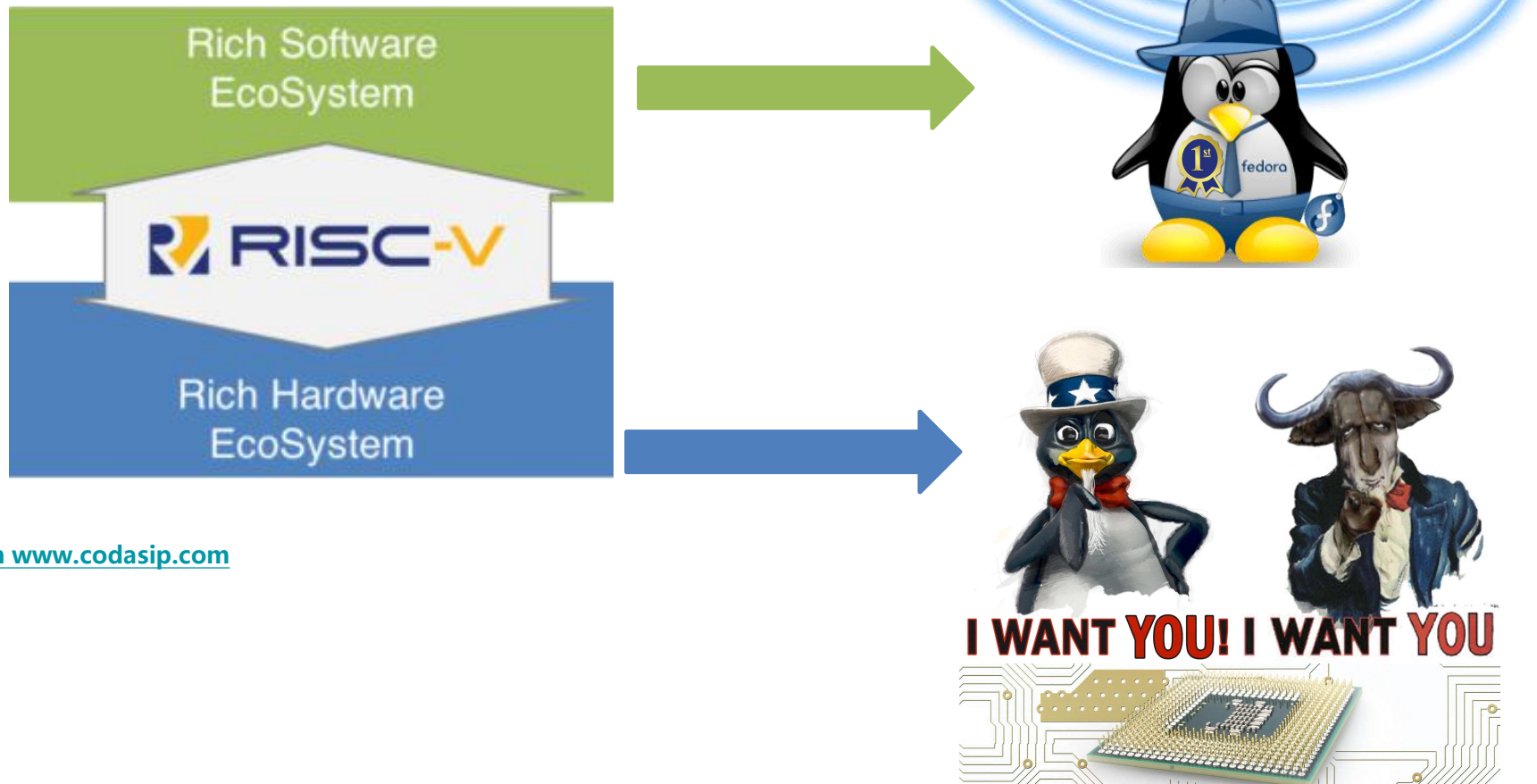


SiFive Unleashed

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



Fedora on RISC-V



[From www.codasip.com](http://www.codasip.com)

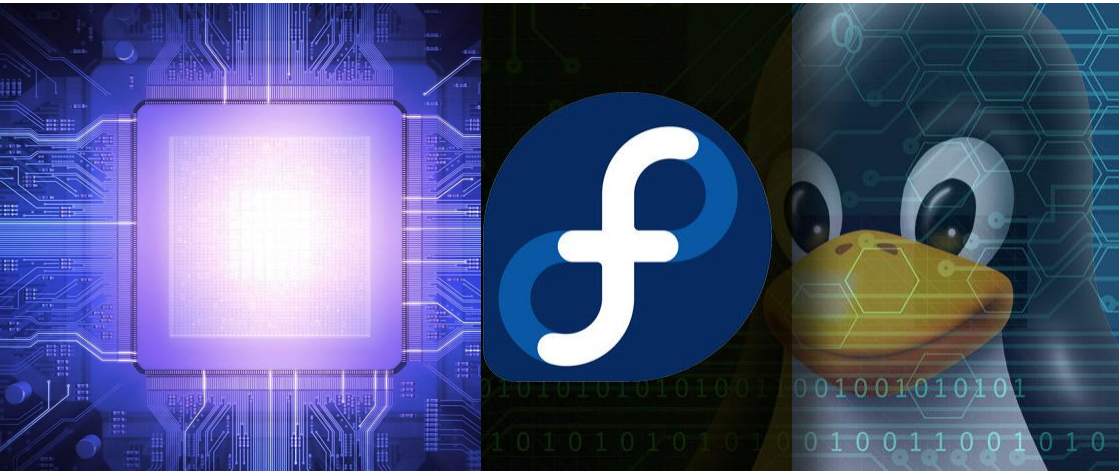
Part Two

RISC-V Development on Fedora

Toolchain

QEMU

VM Tools



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.



QEMU



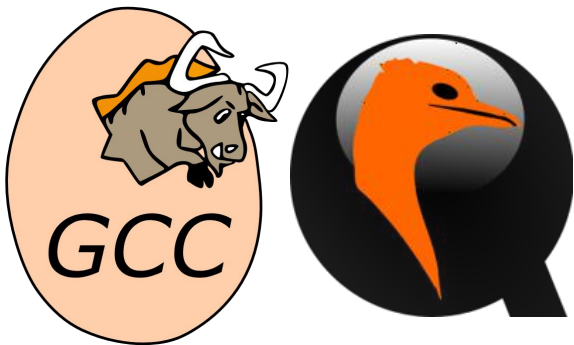
QEMU RPM for RISC-V

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"`

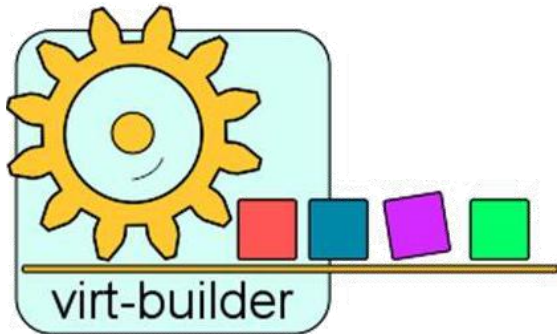


Build QEMU from source code

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



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`



Part Three

Fedora Image in practice

OpenSBI

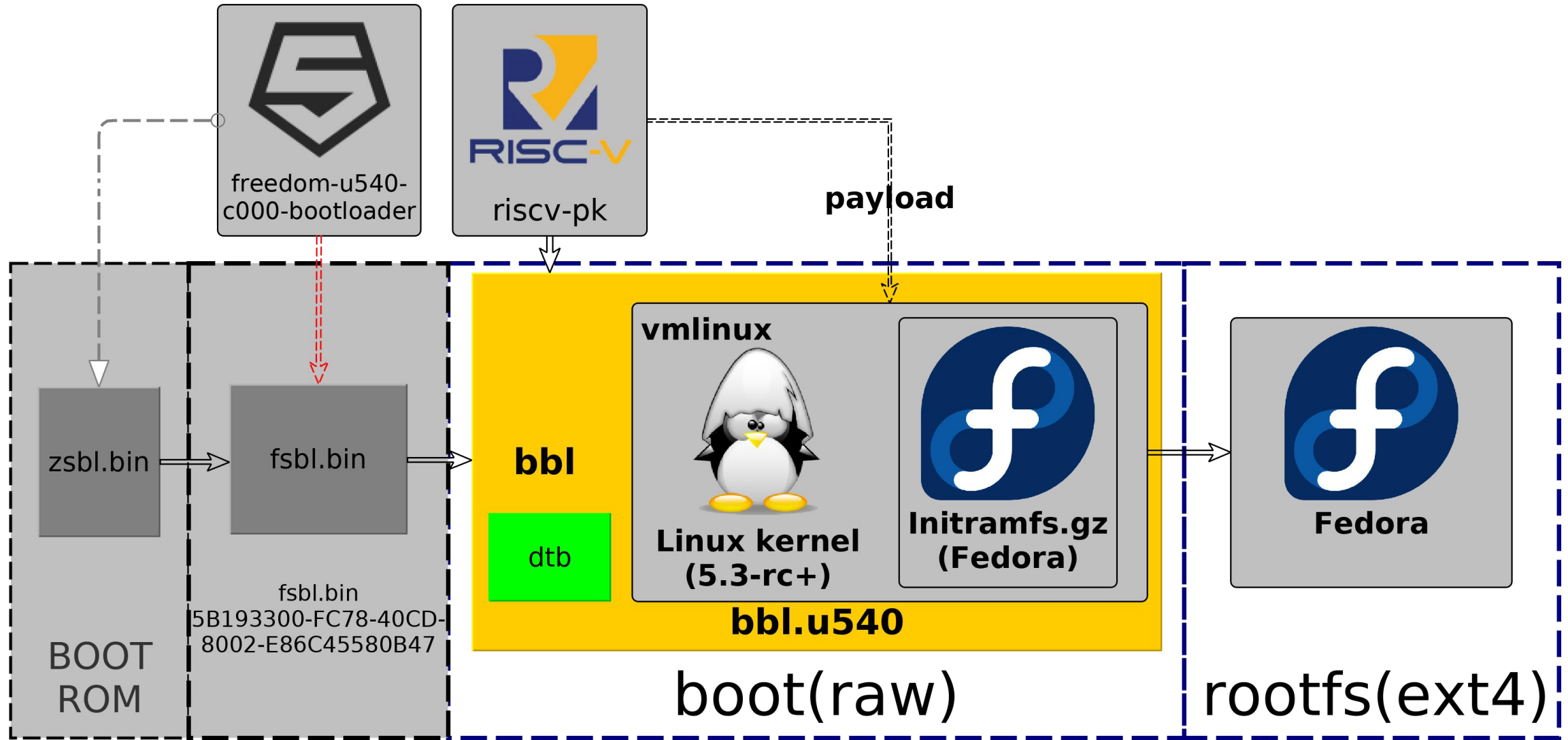
U-Boot

Linux kernel

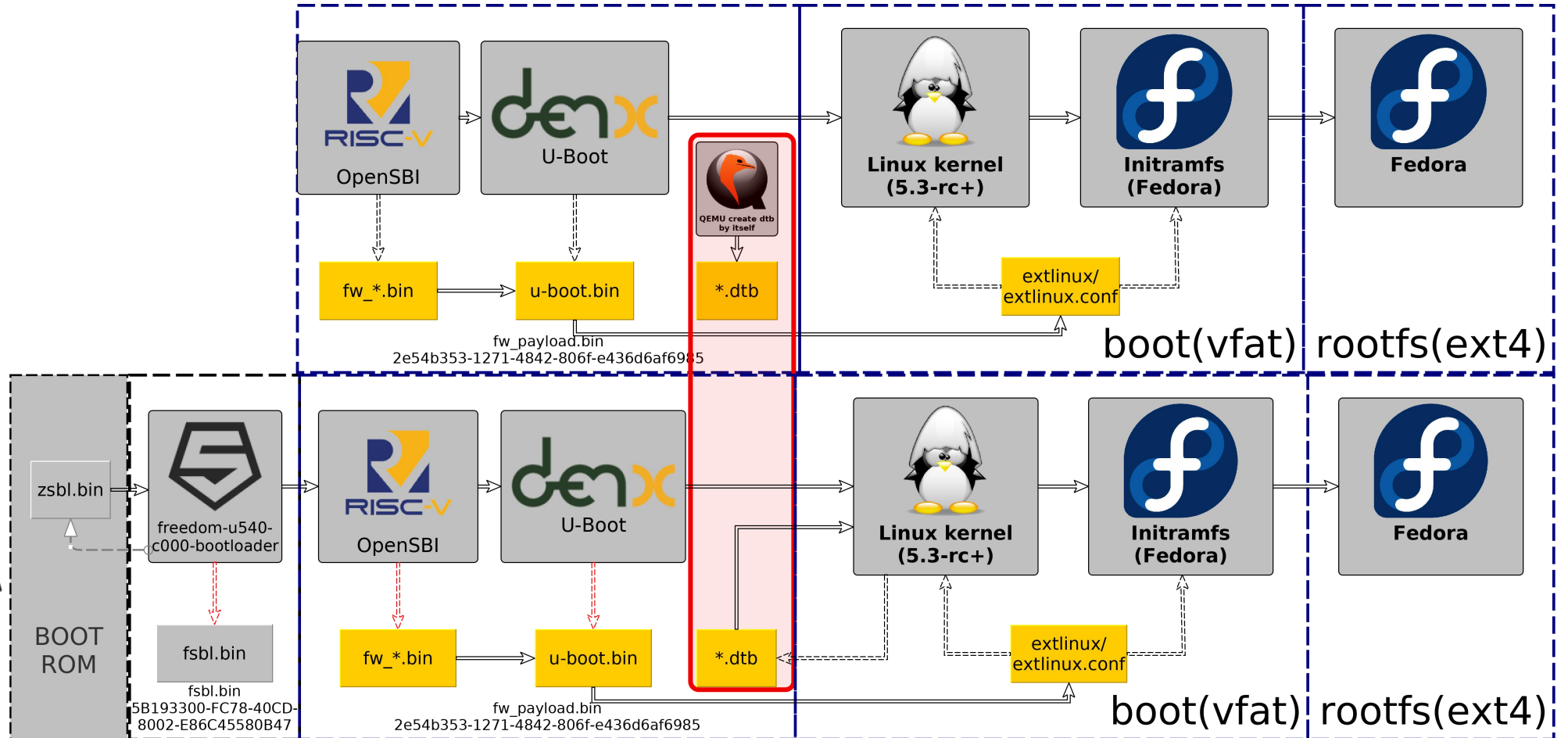
Fedora Image



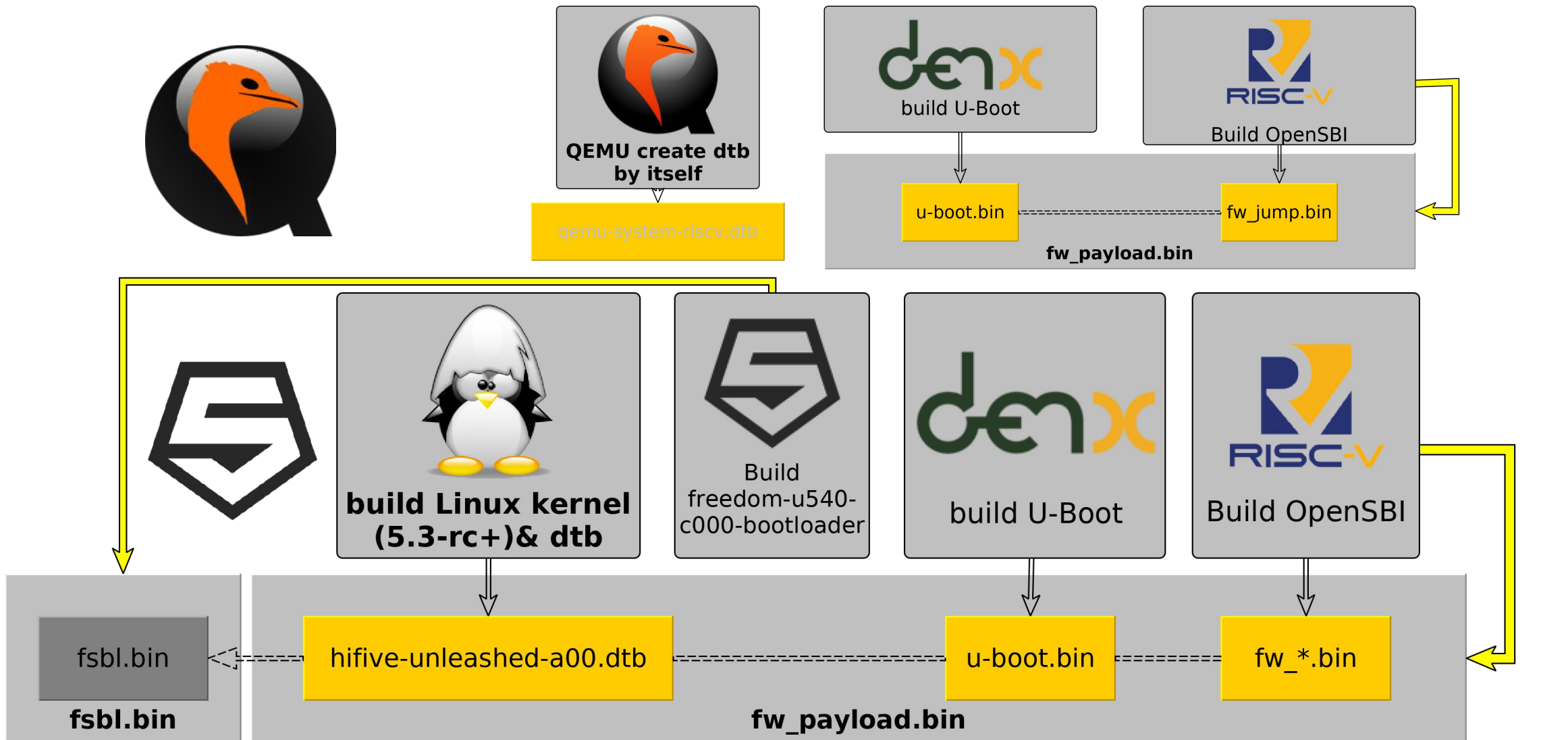
The dated boot flow for Fedora on RISC-V



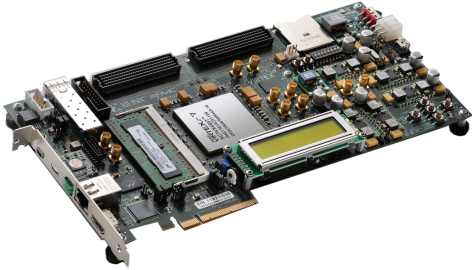
The current boot flow for Fedora on RISC-V



The current Build flow for firmware on RISC-V



The new progress of UEFI on RISC-V



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.



HPE has posted their **V3** patchset for review.

For Now, with V3 patchset, EDK2(+ OpenSBI) can run on QEMU(SiFive U540 machine) and **Real Hardware(SiFive Unleashed)!!!**

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



Abner Chang
Gilbert Chen

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

David Abdurachmanov

Alistair Francis
Anup Patel
Atish Kumar Patra

Mikael Frykholm
Stefan O'Rear



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



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



FYI

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

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
```



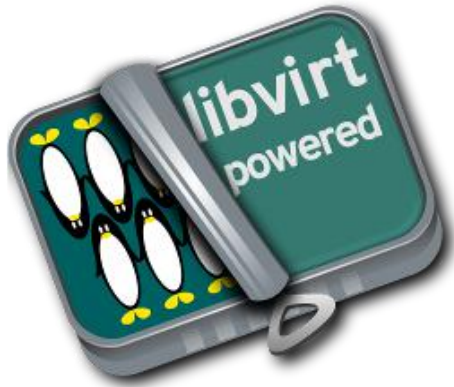
Test on QEMU



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-20190703.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
```

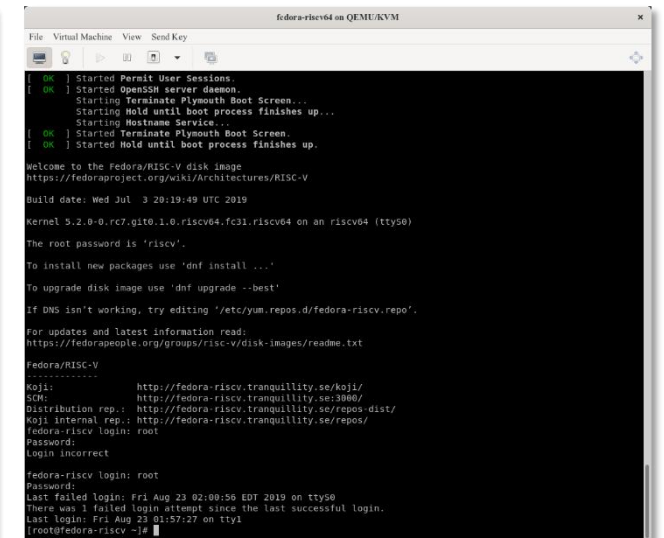
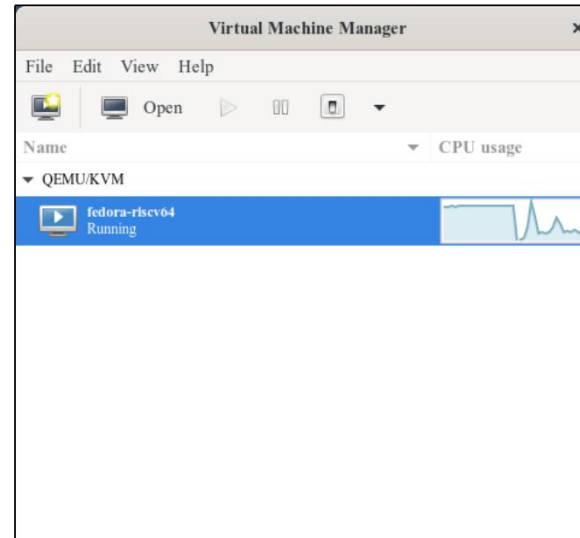
Test with Libvirt



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-20190703.n.0-sda.raw \
--network network=default \
--graphics spice
```

virt-manager



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

<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
```

HiFive Unleashed: Flash into uSD



```
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
```

boot

/(Rootfs)

dd & resize2fs

dd & resize2fs



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}
```