WRS v4 Software

WRSv4 Status & Plans

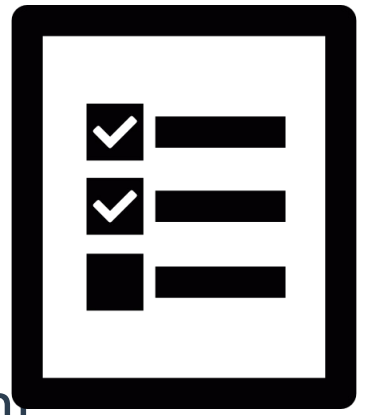
- **Received 6x WRSv4 prototypes in April**
- **Expected commercialy available in 2026+**
- **Development phase: collaboration within WRC**
  - All WR repositories to be moved to gitlab.com to allow external contributions
- **Releases: by WRC Bureau**
- **CERN specific support and integration: WR Team / IN**
- **Target: avoid duplication, reuse as much as possible**

Agenda

- **Requirements**
- **Software aspects:**
  - Device Tree
  - Boot image
  - Kernel
  - Root filesystem
  - Packages
  - WR drivers
  - WR userspace
- **Discussion**

WRSv4 OS Requirements

- **WRS Specific:**
  - Build everything from outside CERN
  - Accommodating requests and contributions by WRC members
  - Network management tools
- **Generic**
  - Everything is “Xilinx compatible”
  - PMUFW/FSBL sources generated to respect copyright
  - Easy to build with scripts
  - Debug tools



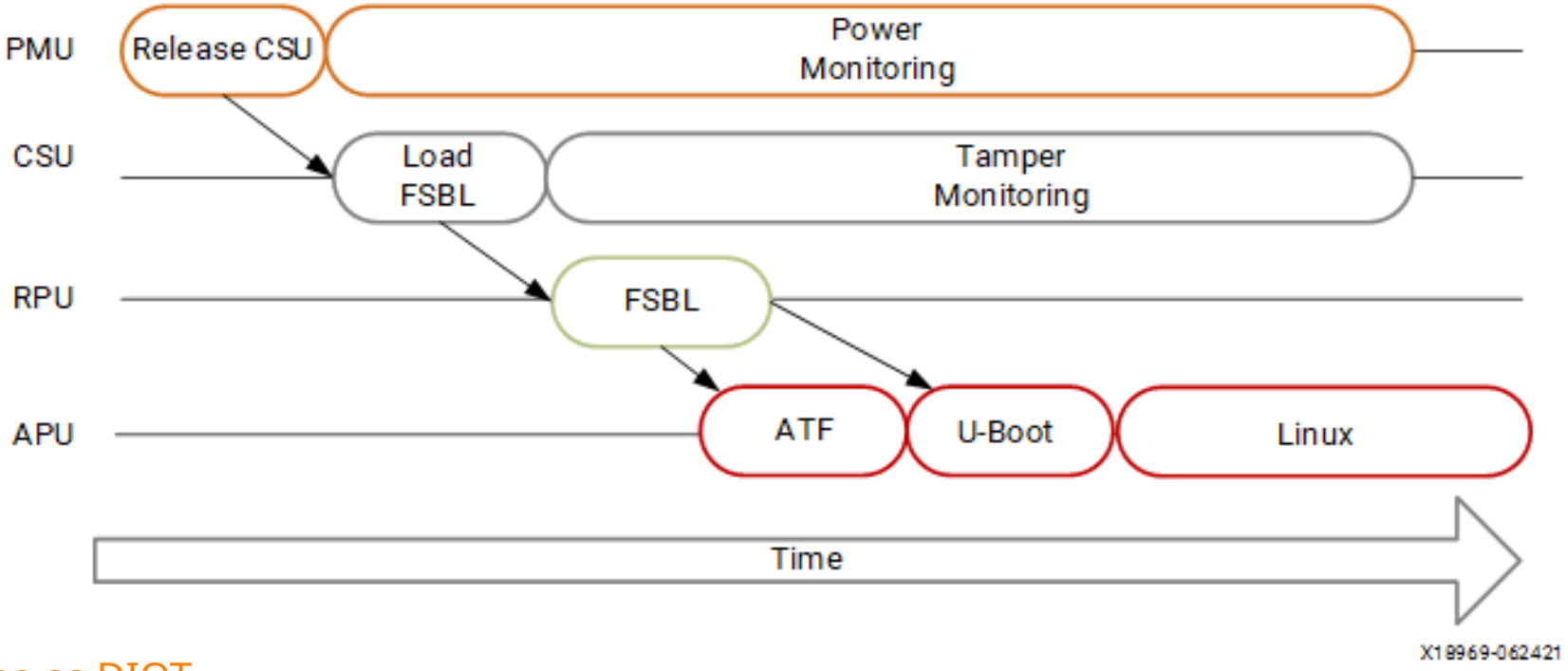
Device Tree

- **Based on Xilinx demo-board**
- **Unique to WRSv4**
- **Used by:**
  - U-Boot (network, MAC EEPROM, ...)
  - Linux kernel

Platform-specific, may be different for each application.
Can't be reused.
Handling not different than any other platform

Boot Flow

- Zynq Ultrascale+ boot flow



Same as DIOT

Boot image – description

- **What does the boot image contain:**
  - PMUFW
  - FSBL
  - ARM Trusted Firmware (xilinx-v2022.2)
  - Bitstream
  - U-Boot-xlnx (xilinx-v2022.2)

Same as DIOT

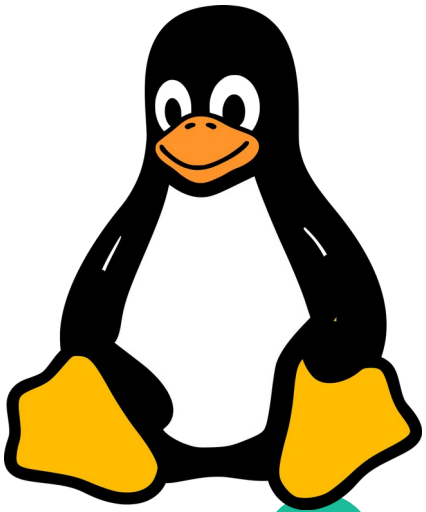
Boot image – build flow

- **Build script: build_boot.sh**
  - Requires XSA file obtained after synthesis
  - Extract from XSA:
    - PMUFW sources
    - FSBL sources
    - Bitstream
  - Compile PMUFW, FSBL, ATF, U-Boot
  - Pack everything into BOOT.bin
  - possibility to build in PMUFW/FSBL in debug
  - possibility to change default UART

Same as DIOT?
What can be reused?

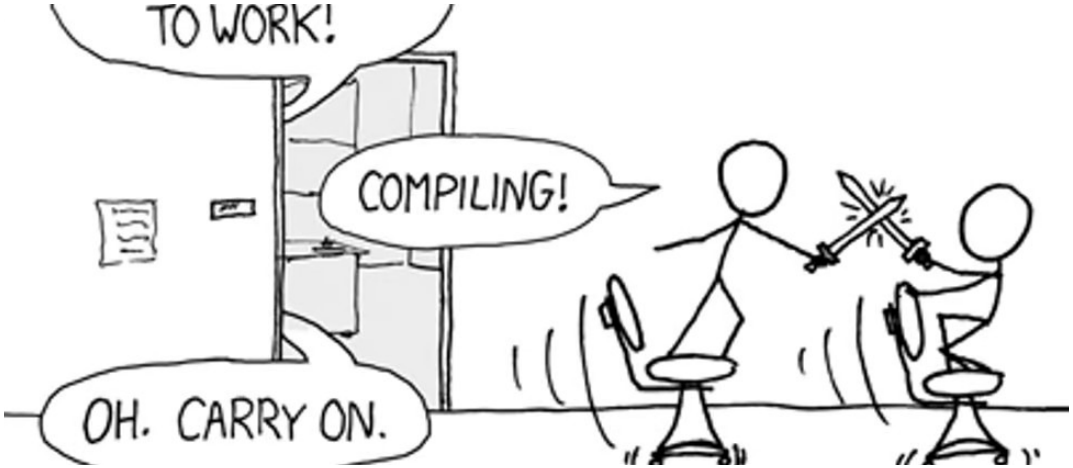
Kernel – WRSv4 history

- FEC-OS-Kernel used first
- To collaborate within WRC, sources had to be available from outside CERN
- Switch to Xilinx kernel instead: linux-xlnx (xlnx_rebase_v5.15_LTS_2022.2)
- Both working on WRSv4



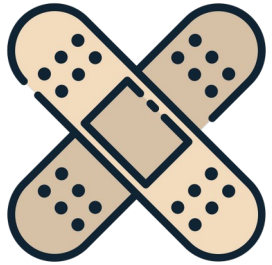
Kernel - build

- **Build script: build_kernel.sh**
  - Clone repo (either FEC-OS or linux-xlnx)
  - Copy .config
  - Build



Kernel - unknowns

- **Apply patches ?**
  - DIOT ?
  - FEC-OS-Kernel ?
- **Compiler:**
  - Clone gcc repo and build compiler ?
  - Download binary ?
  - Let user manage and set CROSS_COMPILE ?
- **Reuse build scripts ?**
  - DIOT ?
  - FEC-OS-Kernel ?



Can we make the FEC-OS available outside CERN ?
How to manage contributions? Patches policy?

Root Filesystem: OpenWRT - pros

- Open from outside of CERN
- Designed for network devices
- Lightweight
- Support HTTP server to configure the device
- Large community, good support
- Already used for WRSv4 development
- Based on buildroot (already used in WRSv3)

Root Filesystem: OpenWRT – cons

- **Not built for the same kernel version but still works:**
  - Linux-xlnx: 5.15
  - OpenWRT: 6.6
- **Package manager (opkg) not as good as apt**
- **Not really compatibile with Xilinx devices (“best effort”)**

Root Filesystem: OpenWRT – build

- **Build script: build_rootfs.sh**
  - Copy .config
  - make defconfig
  - make
  - Mount root.ext4 image and copy files:
    - */etc/inittab*
    - */etc/fstab*
    - */etc/init.d/done*

What rootfs is used on DIOT ?
How is it built ?

WR Drivers

- **Ported from WRSv3 kernel 3.16.38 to 5.15**
- **Some drivers from general-cores:**
  - htvic (cohtvic)
  - I2c-ocores (opencores i2c, with mux)
- **Run fine (OpenWRT built for kernel 6.6.35, actual kernel is 5.15 → warning incompatible)**
- **For now, one build script per driver**

How DIOT drivers are managed ?

WR Userspace

- Ported from WRSv3
- Compiled using OpenWRT compiler (gcc, built during OpenWRT build)
- For now, one script per userspace program

How DIOT userspace programs are managed ?

Summary & Discussion

- **Device Tree:**
- **Boot Image:**
  - build script: same as DIOT ?
  - Can be reused ?
- **Kernel:**
  - Can we make the FEC-OS available outside CERN ?
  - How to manage contributions?
  - Patches policy?
- **Root filesystem:**
  - What rootfs is used on DIOT ?
  - How is it built ?
- **Packages:**
- **WR drivers:**
  - how DIOT drivers are managed ?
- **WR userspace:**
  - how DIOT userspace programs are managed ?