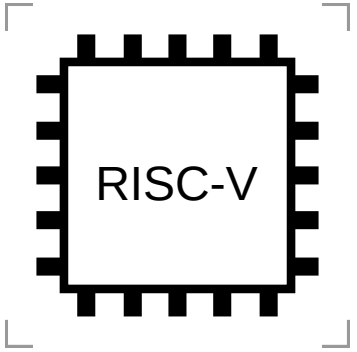
RISC-V runs Linux

Матюкевич Сергей
Syntacore
2024

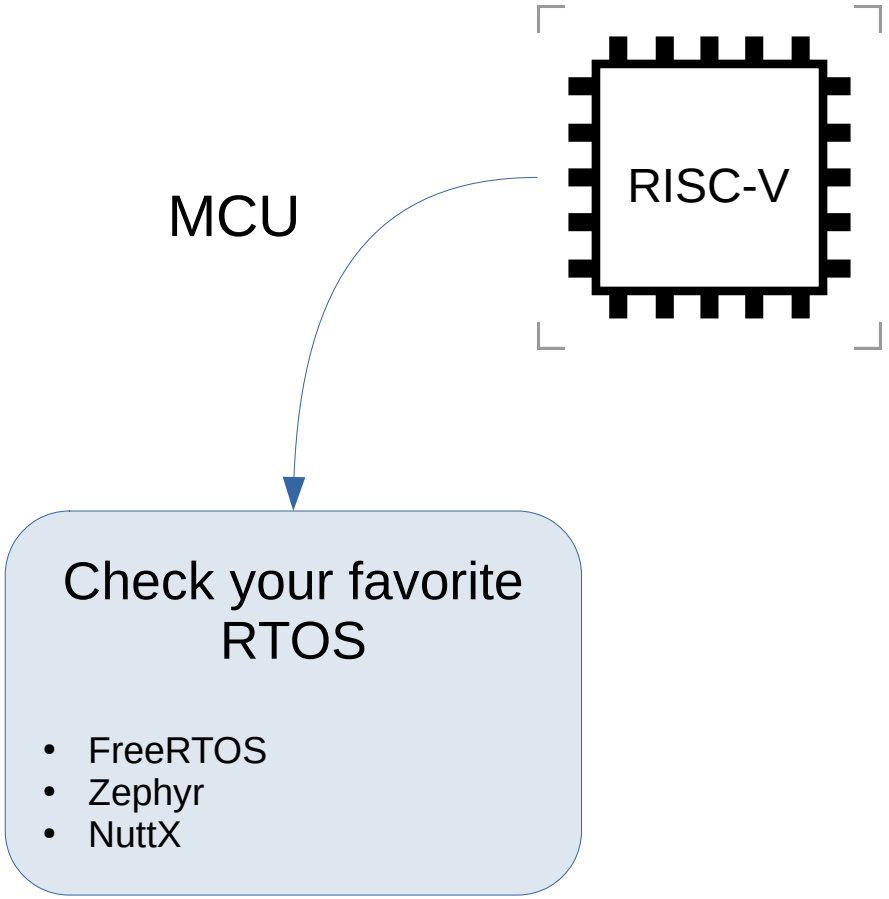
Матюкевич Сергей

- Отдел системного ПО
- Ведущий инженер-программист

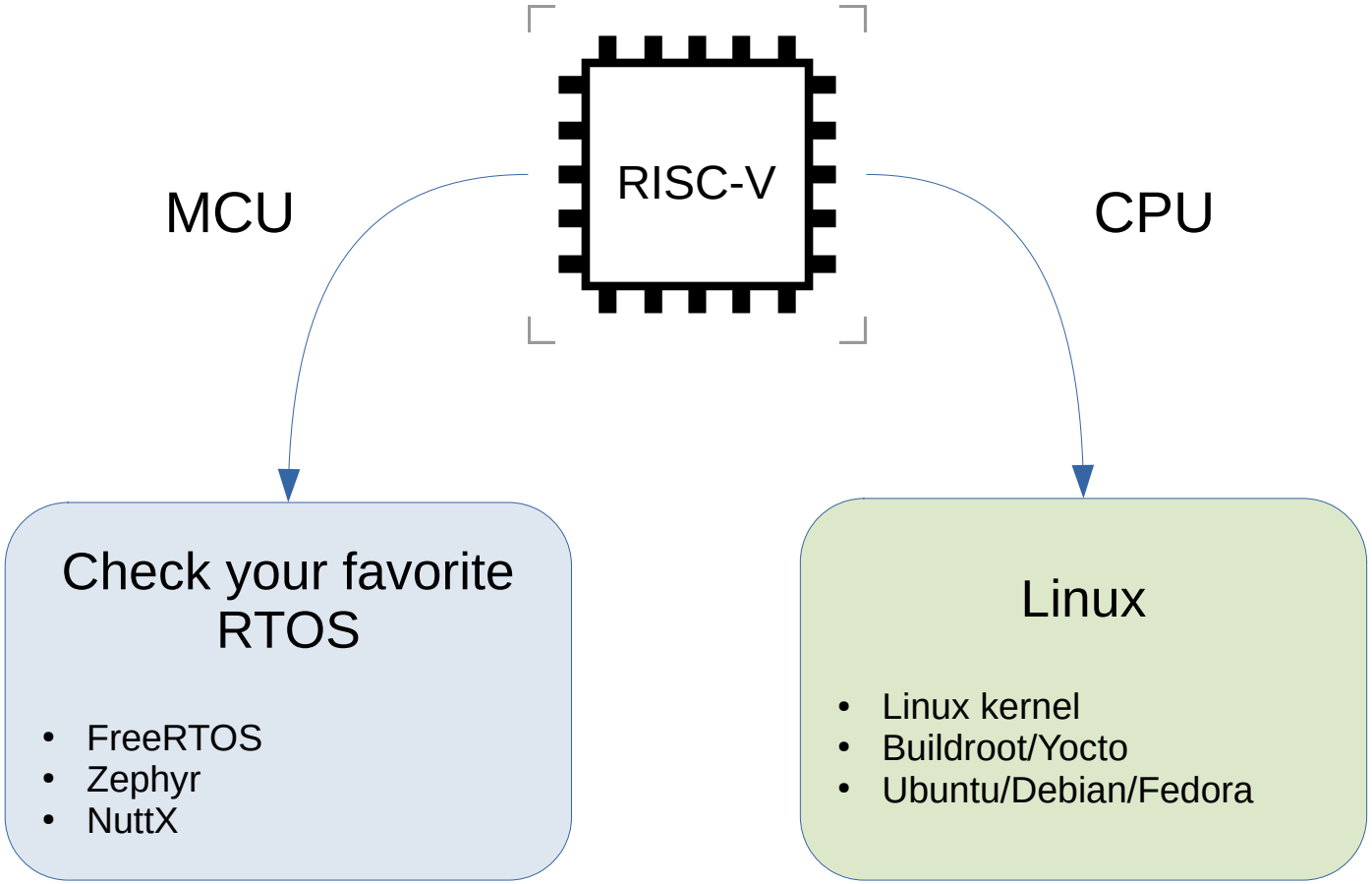
OSS Ecosystem



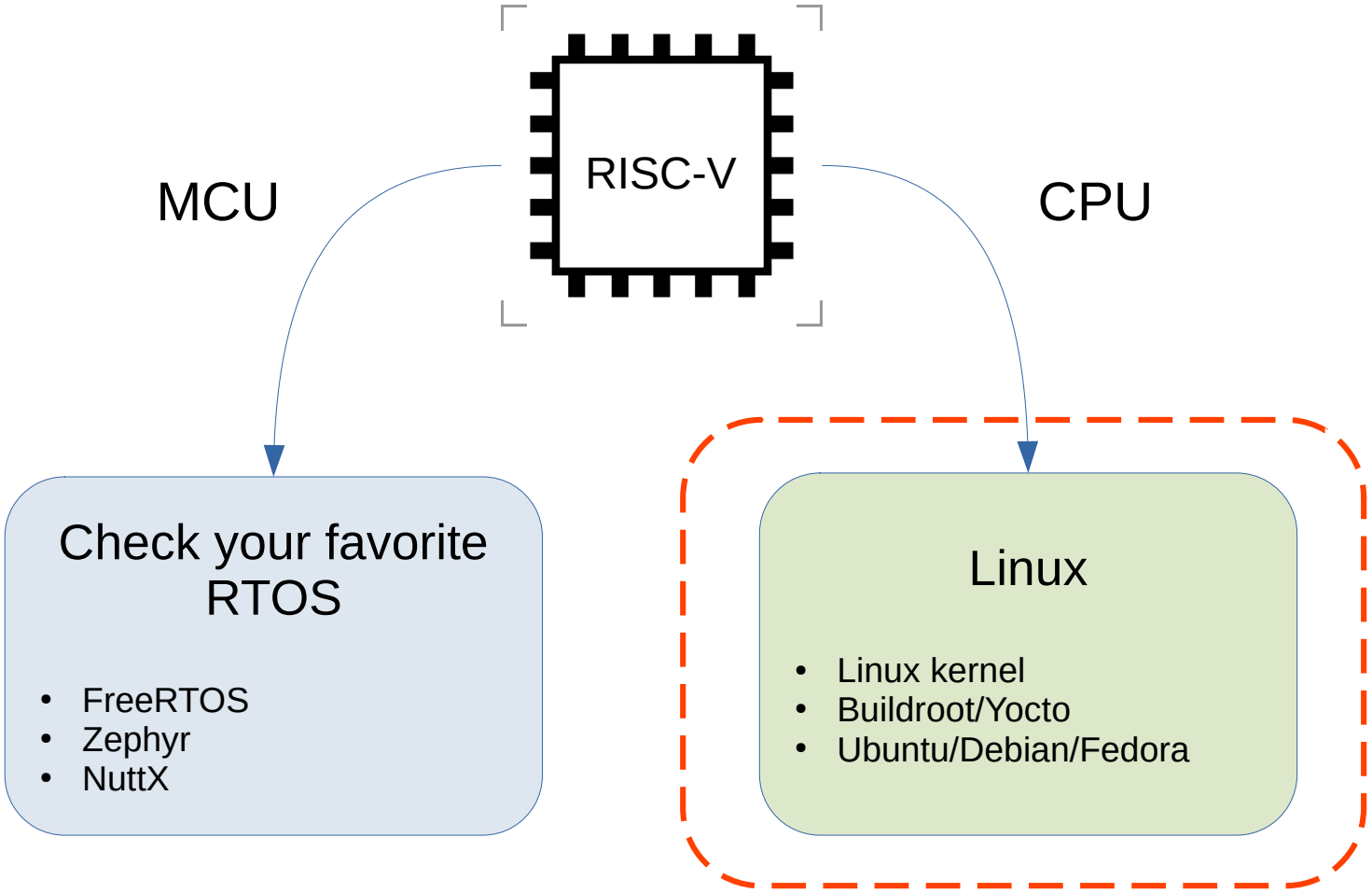
OSS Ecosystem



OSS Ecosystem

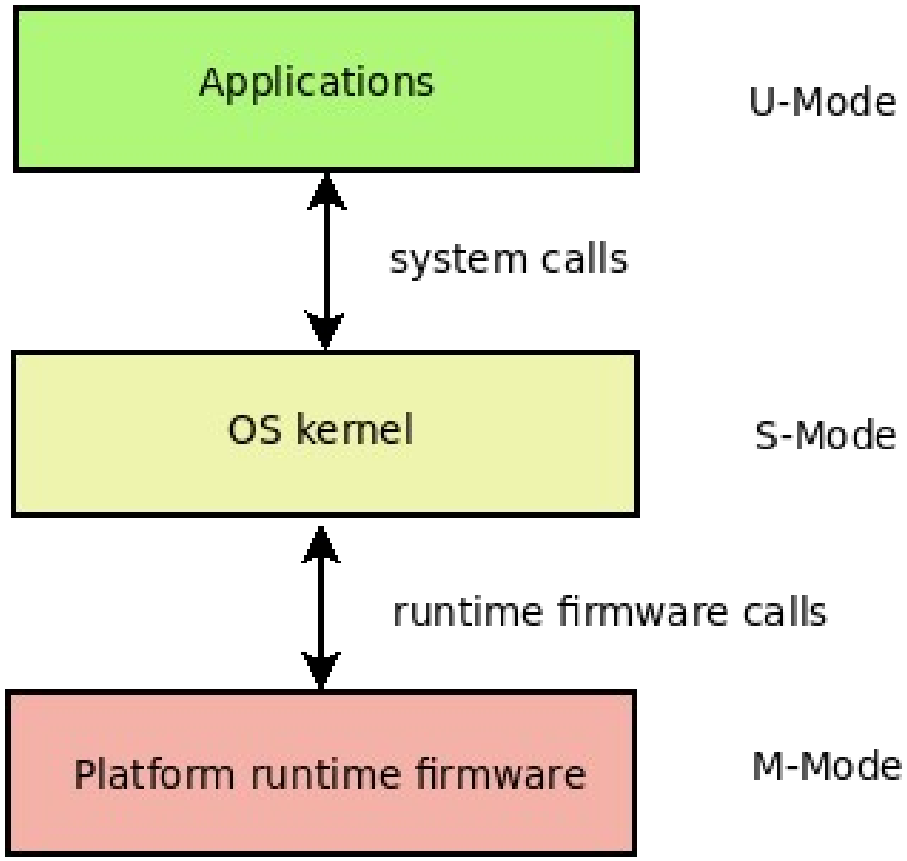


OSS Ecosystem



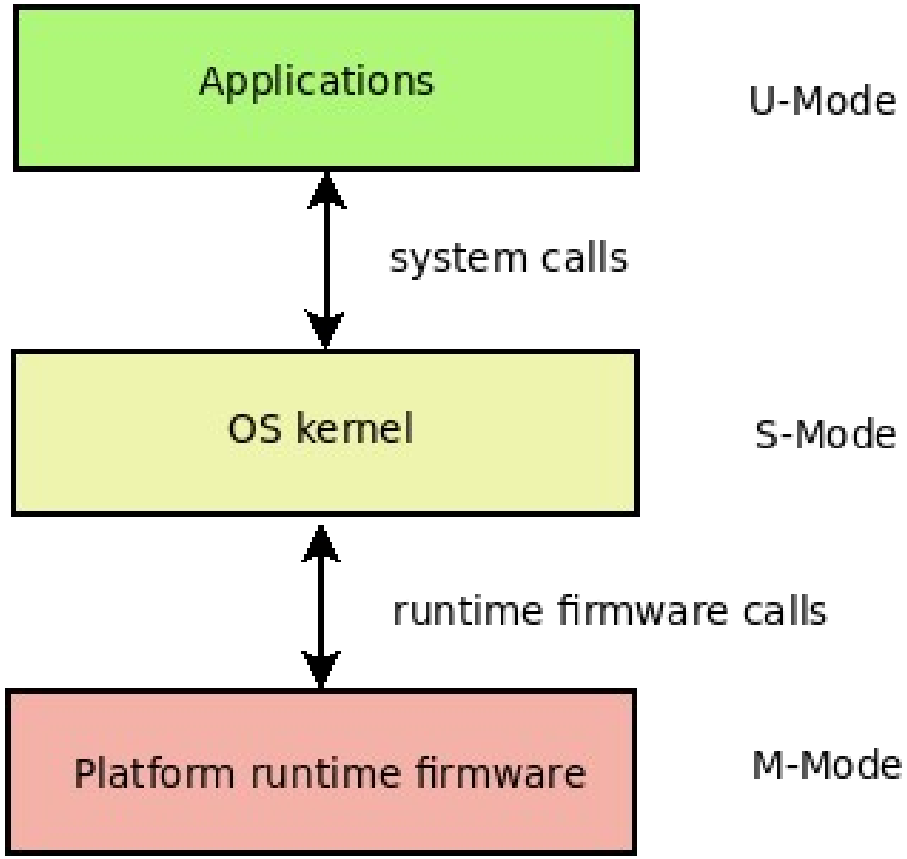
RISC-V runtime

- M-mode
  - full access to hardware
  - exceptions/interrupts



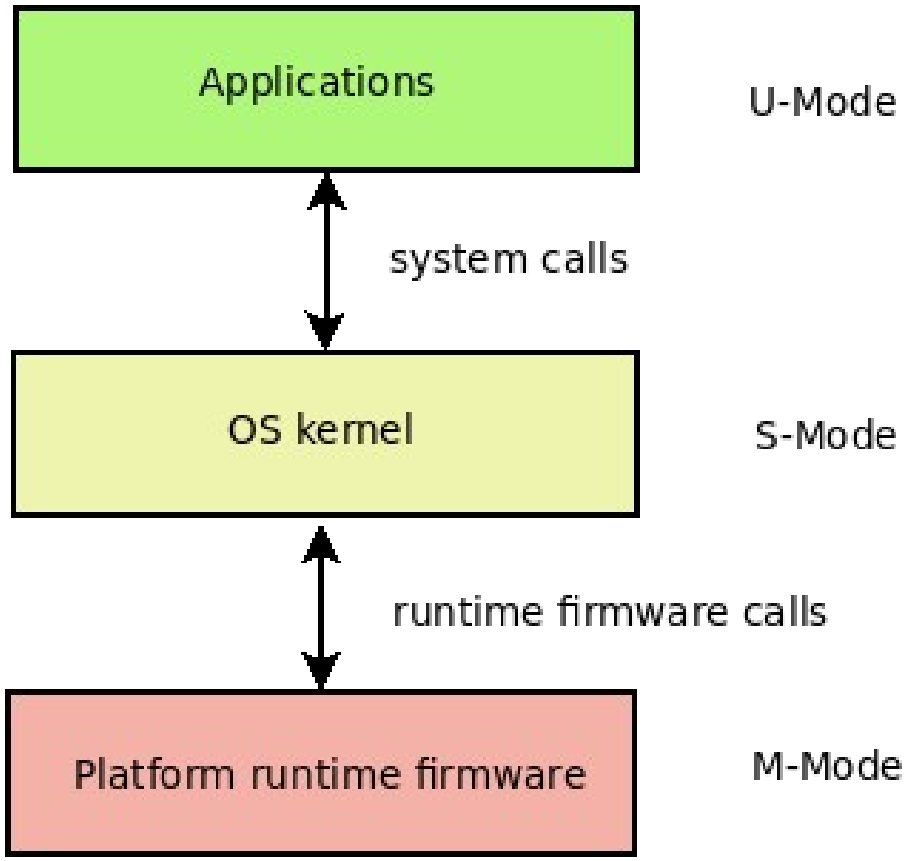
RISC-V runtime

- M-mode
  - full access to hardware
  - exceptions/interrupts
- S-mode
  - paged-based virtual memory
  - delegation (exceptions/interrupts)



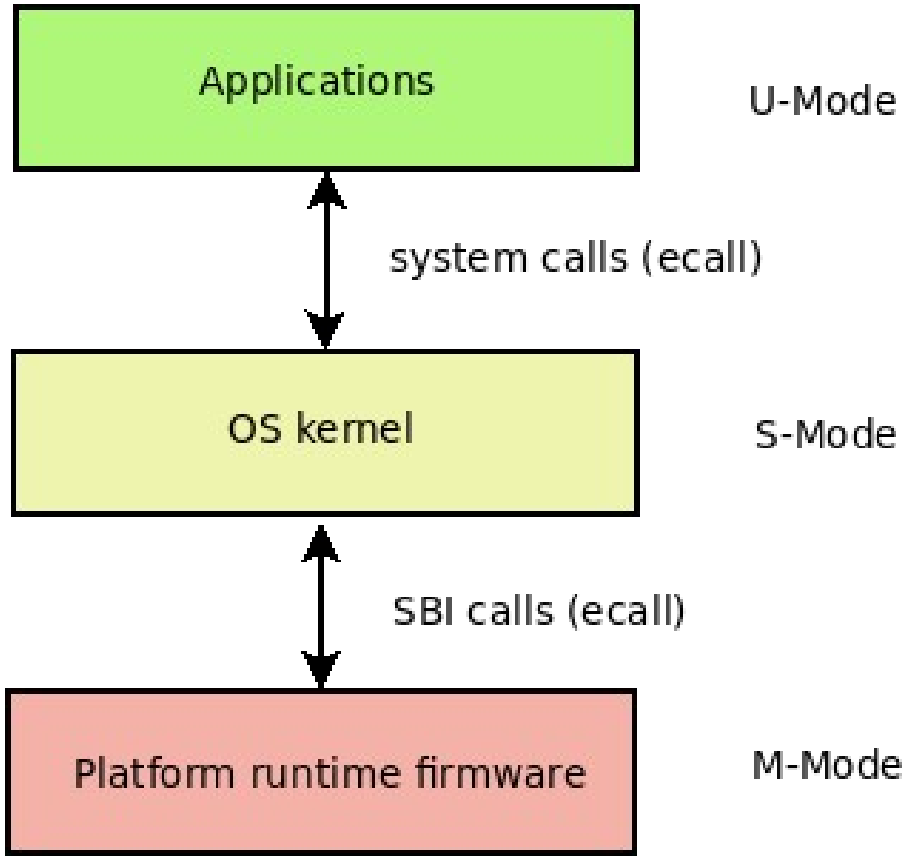
RISC-V runtime

- M-mode
  - full access to hardware
  - exceptions/interrupts
- S-mode
  - paged-based virtual memory
  - delegation (exceptions/interrupts)
- U-mode



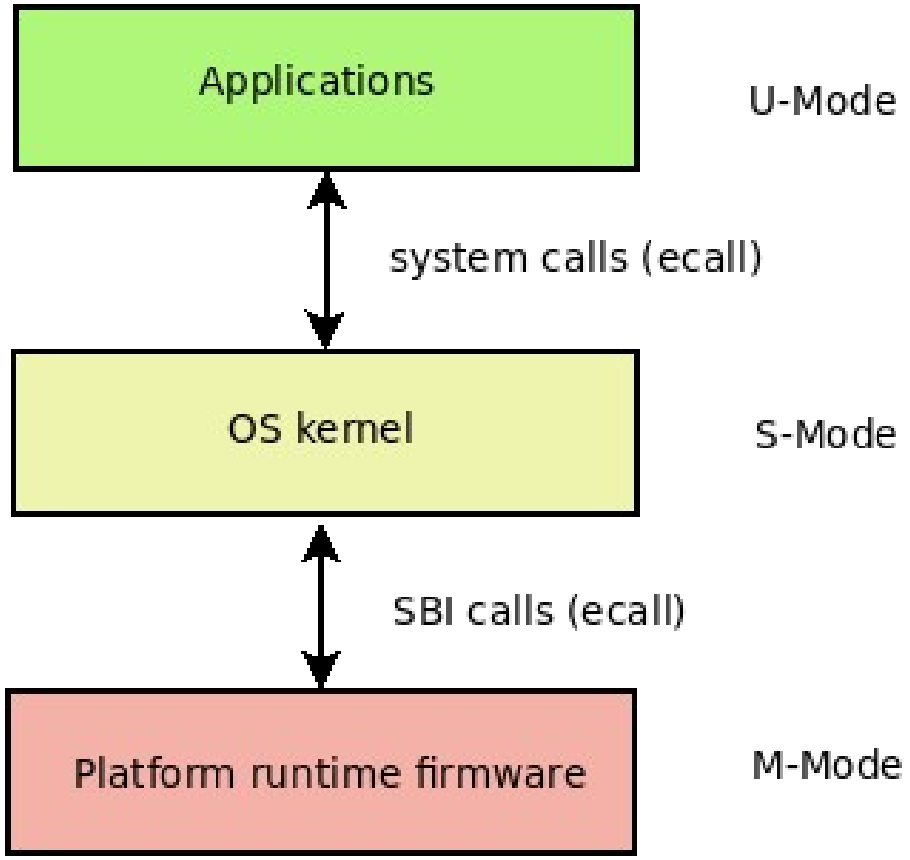
SBI: Supervisor Binary Interface

- RISC-V non-ISA specification
  - https://github.com/riscv-non-isa/riscv-sbi-doc
  - Status: v2.0 (Ratified)



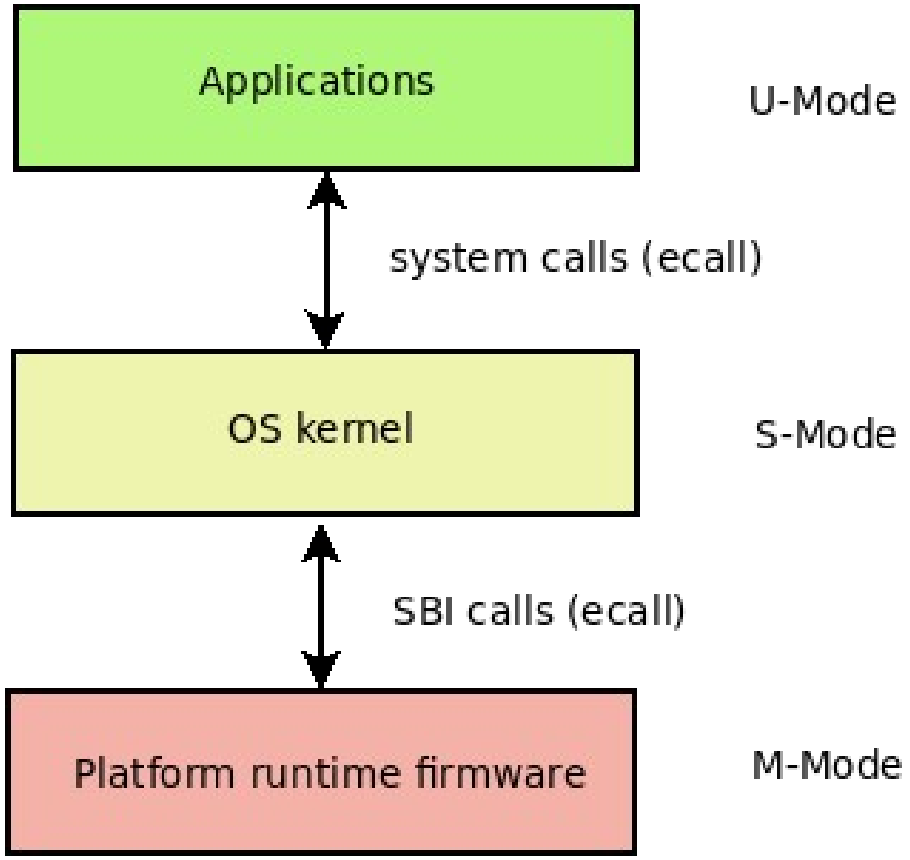
SBI: Supervisor Binary Interface

- RISC-V non-ISA specification
  - https://github.com/riscv-non-isa/riscv-sbi-doc
  - Status: v2.0 (Ratified)
  - Access to privileged resources (M-mode)
  - Platform abstraction layer

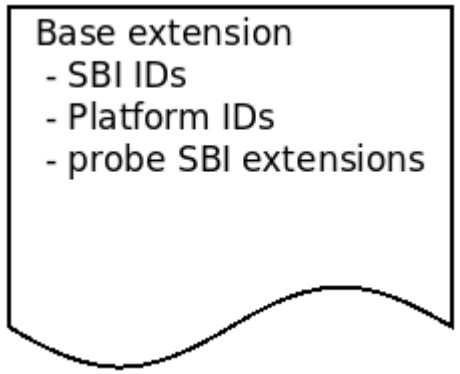


SBI: Supervisor Binary Interface

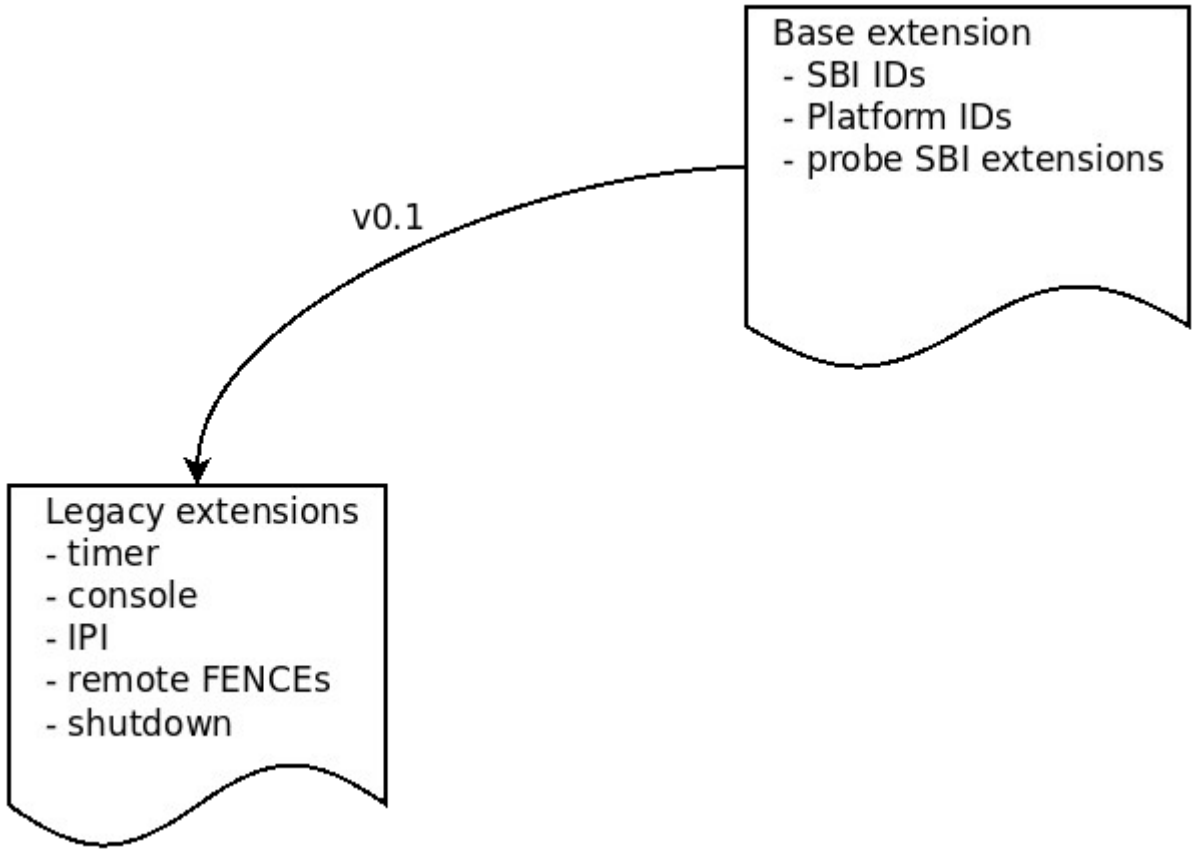
- RISC-V non-ISA specification
  - https://github.com/riscv-non-isa/riscv-sbi-doc
  - Status: v2.0 (Ratified)
  - Access to privileged resources (M-mode)
  - Platform abstraction layer
- Different implementations
  - BBL, OpenSBI, ...



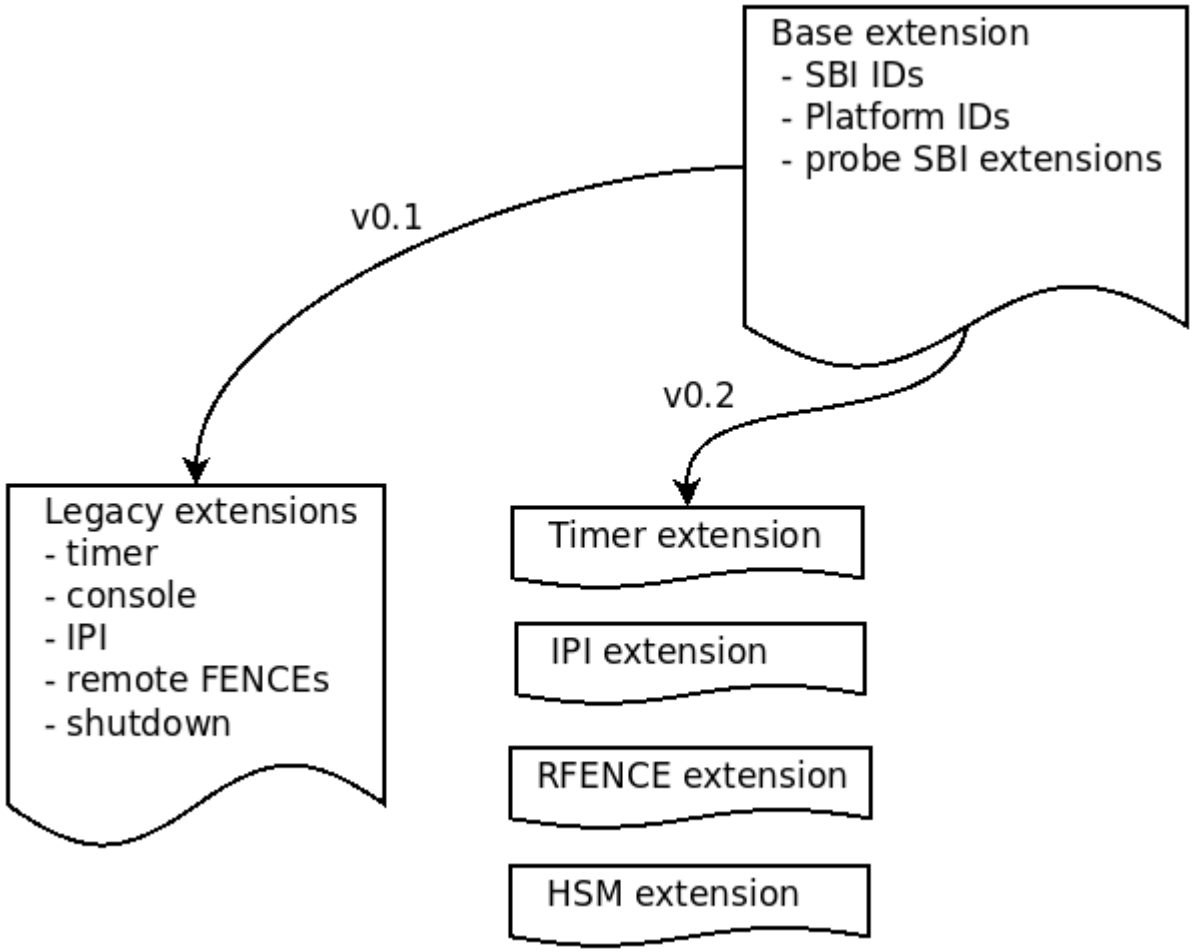
SBI: timeline



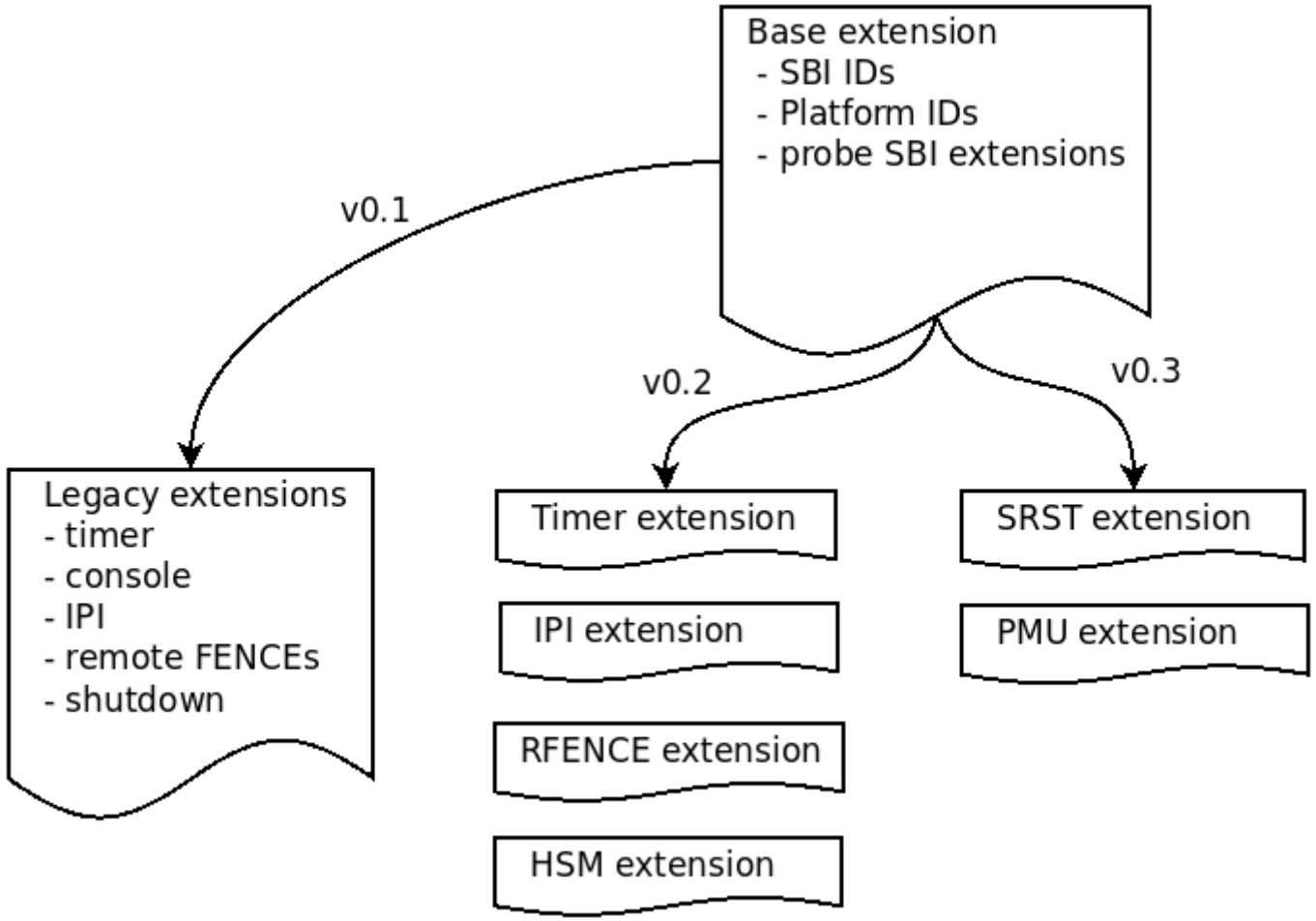
SBI: timeline



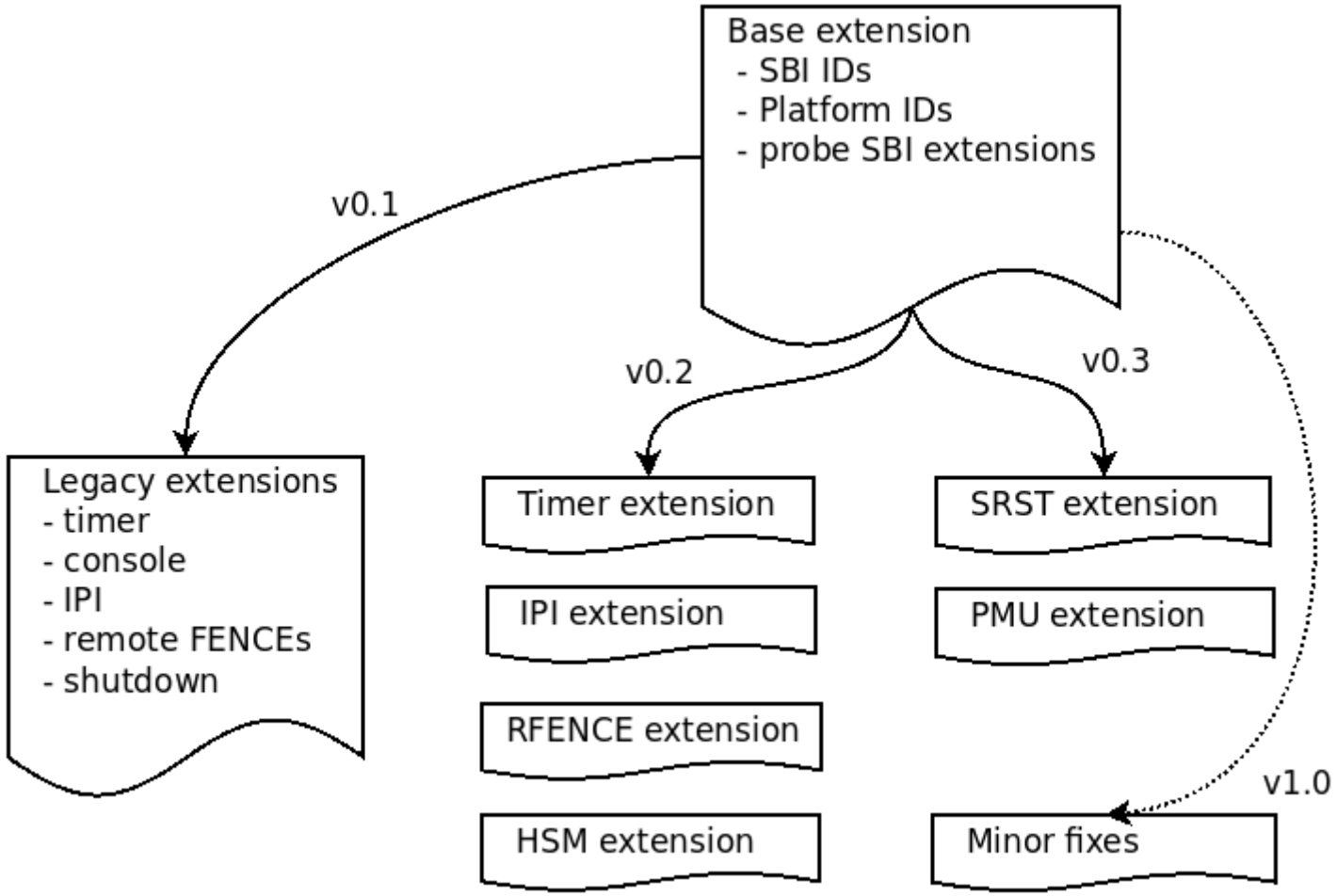
SBI: timeline



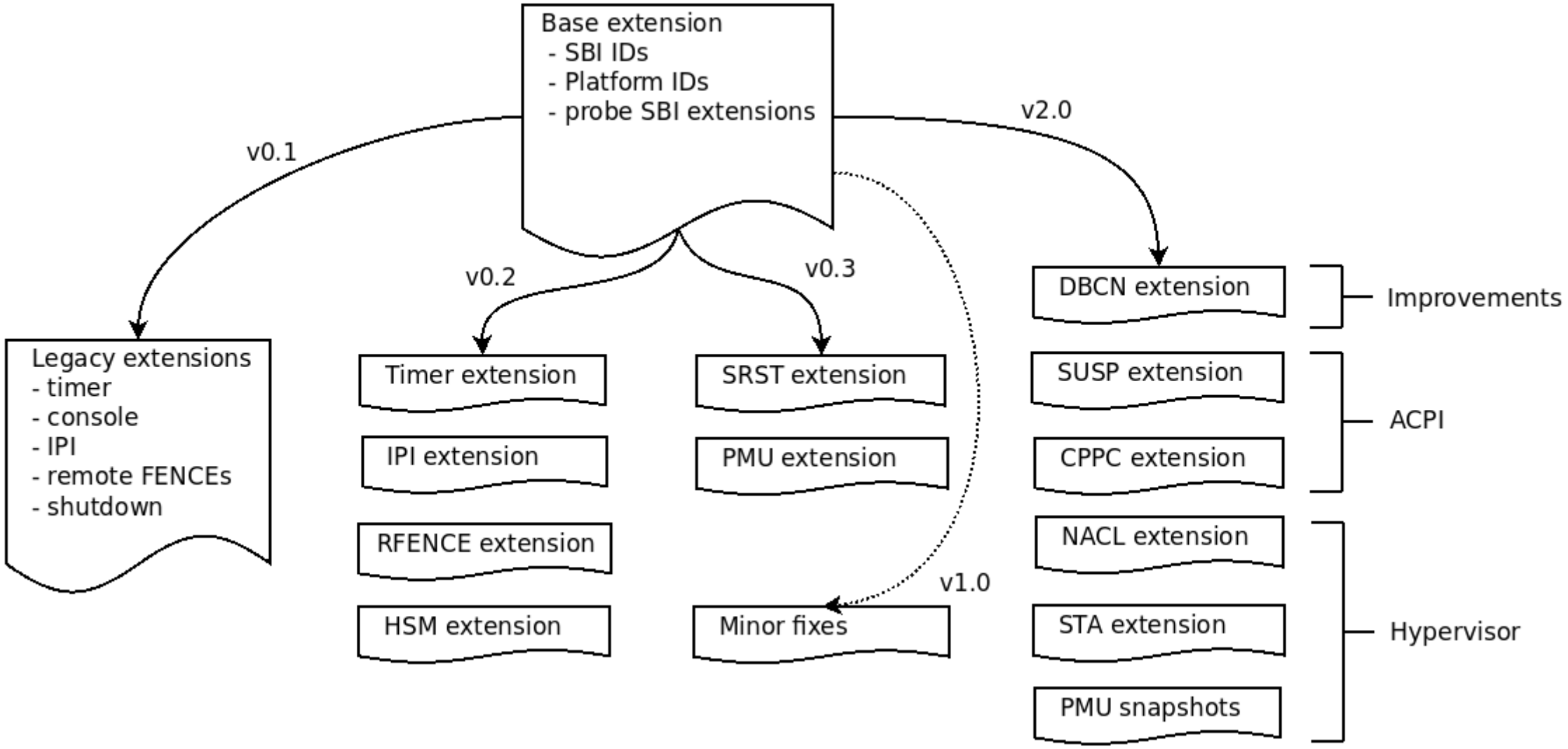
SBI: timeline



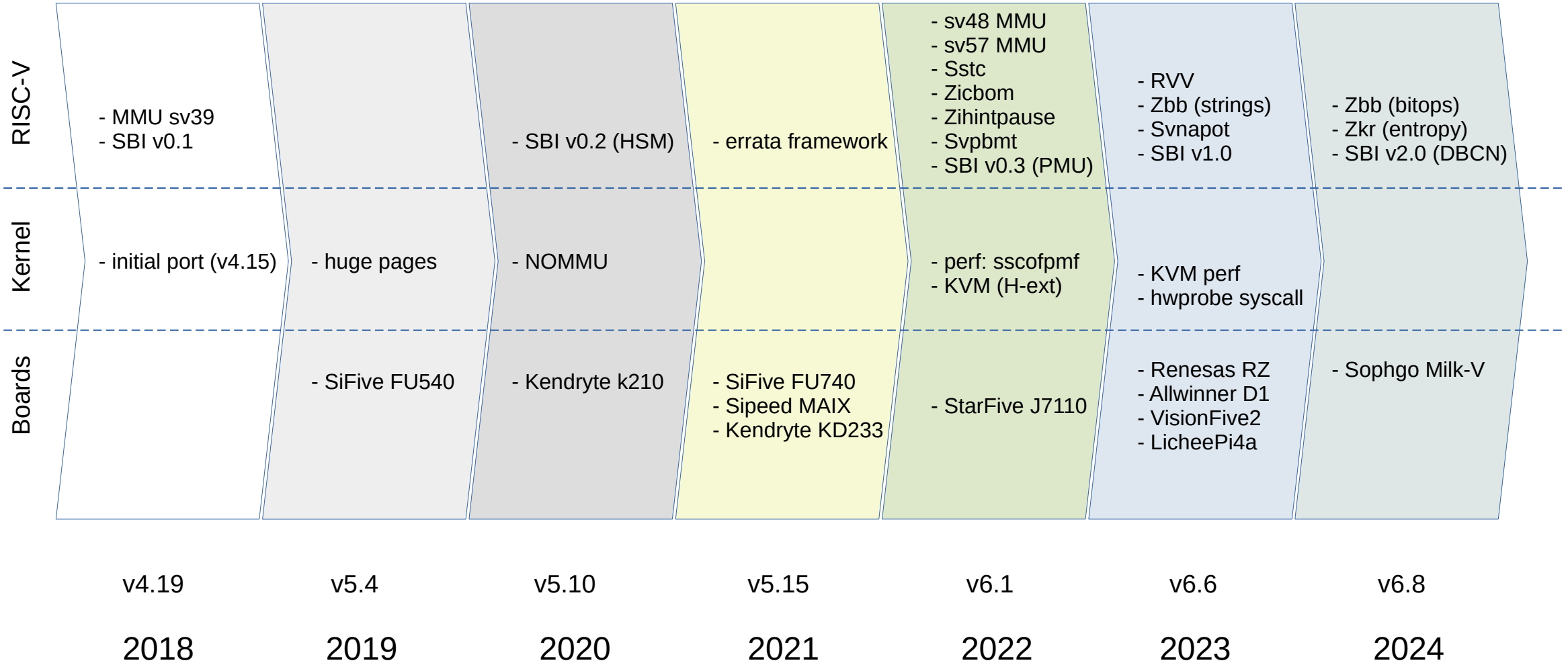
SBI: timeline



SBI: timeline



Linux: timeline



Linux: maintenance guidelines

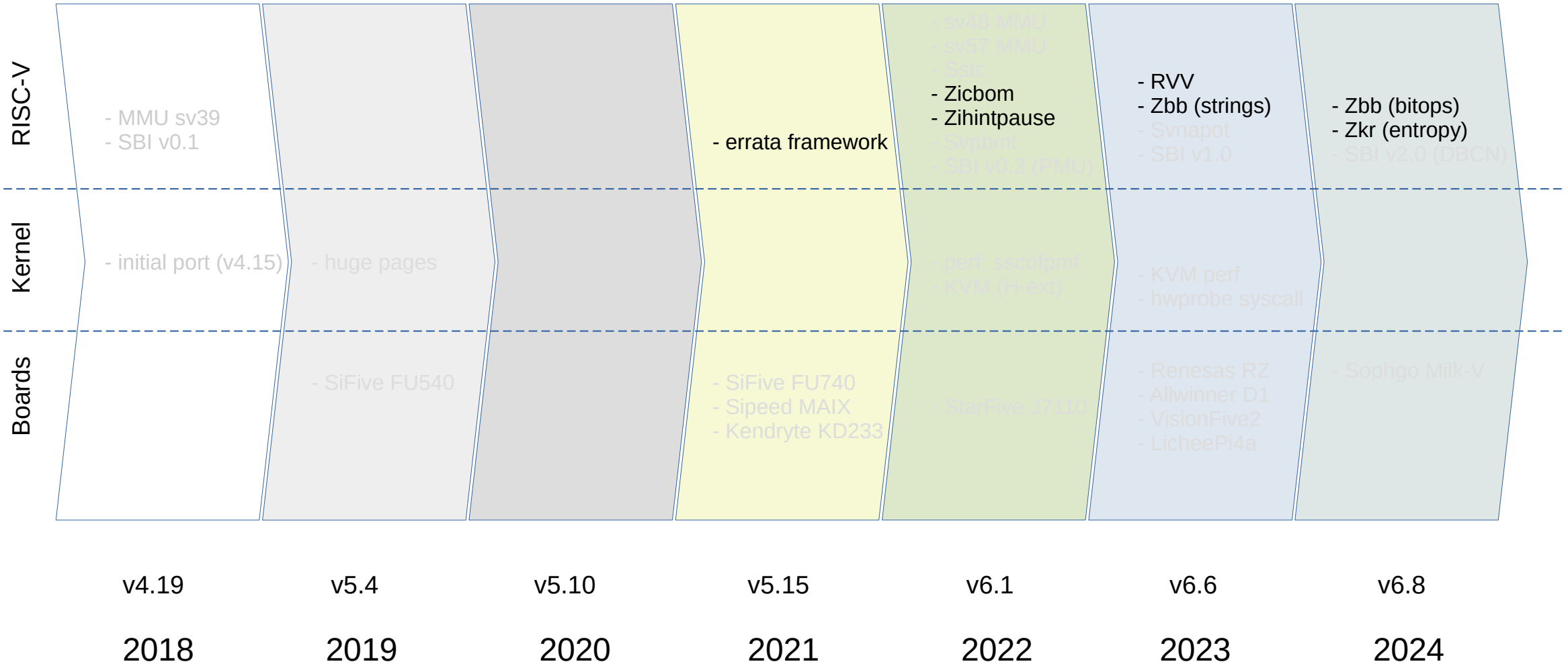
Kernel patches are accepted for extensions that:

- have been **officially frozen or ratified** by the RISC-V Foundation
- have been implemented in hardware that is widely available
  - see Documentation/riscv/patch-acceptance.rst

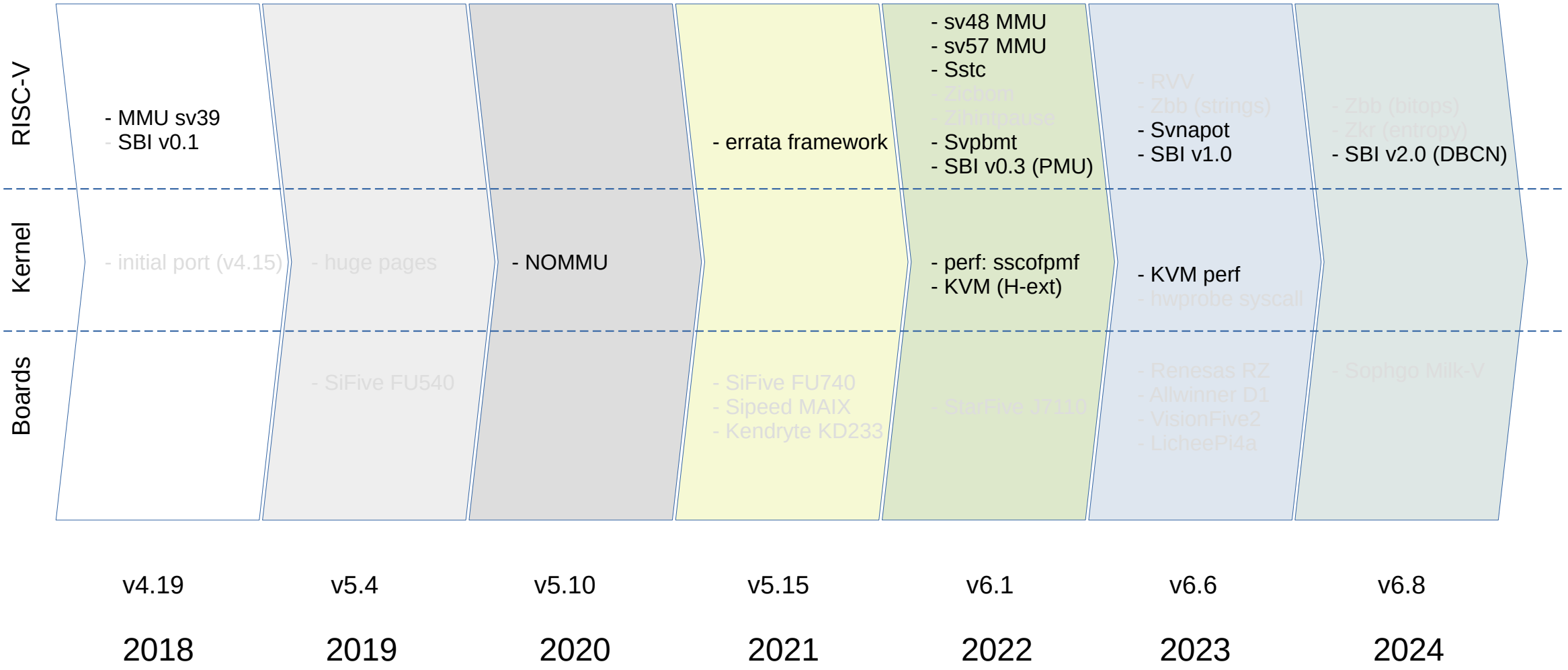
Not yet ratified functionality:

- send RFC patches for discussion and early feedback
- maintain custom Linux kernel trees

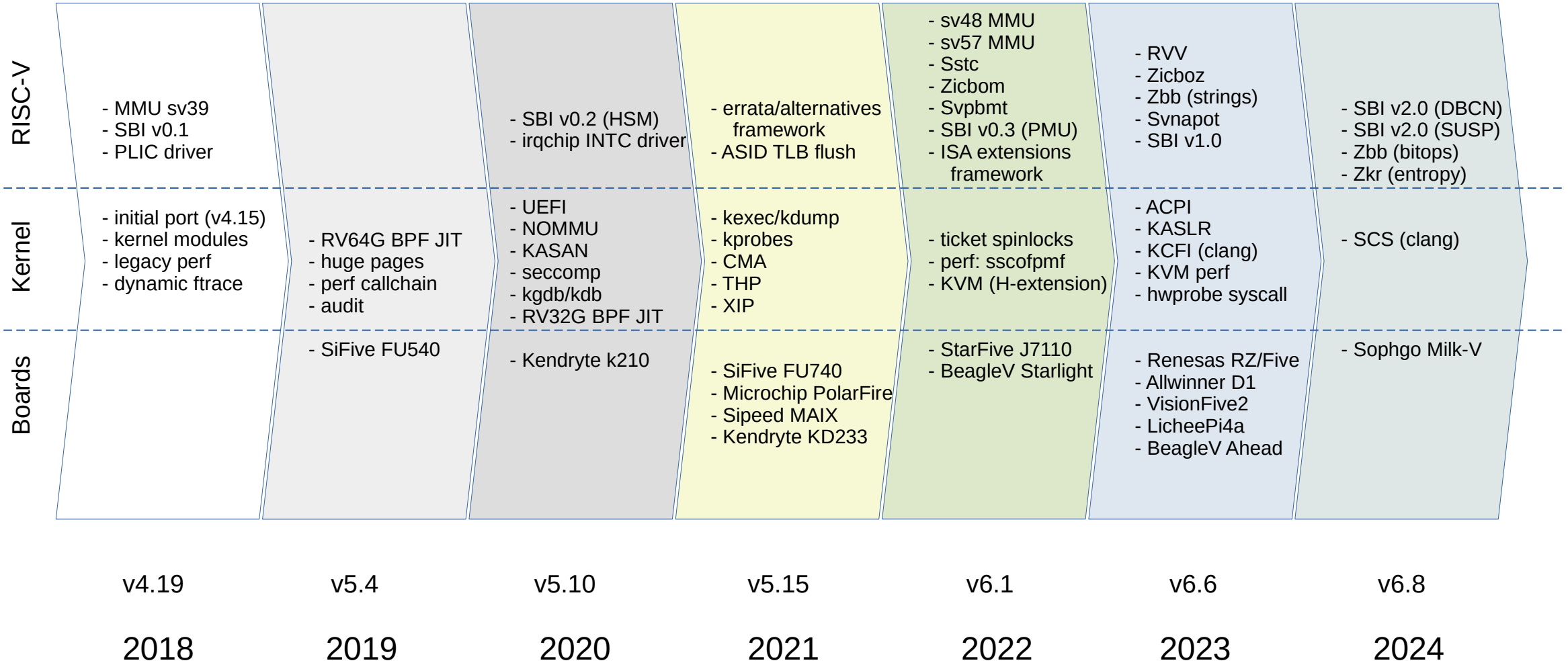
Linux: timeline



Linux: timeline



Linux: timeline



Linux: RISC-V: Great Old Ones

```
On Fri, Sep 22, 2023 at 08:56:47AM +0000, Yong-Xuan Wang wrote:
> We detect Svadu extension support from DTB and add arch_has_hw_pte_young()
> to enable optimization in MGLRU and __wp_page_copy_user() if Svadu
> extension is available.

If you're going to cc people outside the RiscV space, you should probably
explain what Svadu is.  To me, it sounds like a Great Old One, maybe a
friend of Cthulhu?
```

Linux: RISC-V: Great Old Ones in v6.8.x

- RVF/RVD, RVV
  - contexts, ptrace
- Sscofpmf
  - perf record
- Sstc
  - accessing timer from S-mode
- Bitmanip (Zbb)
  - string operations, IP/TCP/UDP checksums
- Zicbom
  - cache operations for non-coherent DMA

Linux: RISC-V: Great Old Ones in v6.8.x

- Zicboz
  - clear_page
- Zihintpause
  - cpu_relax
- Zkr
  - archrandom
- H-ext
  - KVM

Linux: RISC-V features configuration

- Configuration
  - $ make ARCH=riscv CROSS_COMPILE=riscv64-unknown-linux-gnu- defconfig
  - fixup based on toolchain support

Linux: RISC-V features configuration

- Configuration
  - $ make ARCH=riscv CROSS_COMPILE=riscv64-unknown-linux-gnu- defconfig
  - fixup based on toolchain support

```
CONFIG_RISCV_ISA_C=y
CONFIG_RISCV_ISA_SVNAPOT=y
CONFIG_RISCV_ISA_SVPBMT=y
CONFIG_TOOLCHAIN_HAS_V=y
CONFIG_RISCV_ISA_V=y
CONFIG_RISCV_ISA_V_DEFAULT_ENABLE=y
CONFIG_RISCV_ISA_V_UCOPY_THRESHOLD=768
CONFIG_TOOLCHAIN_HAS_ZBB=y
CONFIG_RISCV_ISA_ZBB=y
CONFIG_RISCV_ISA_ZICBOM=y
CONFIG_RISCV_ISA_ZICBOZ=y
CONFIG_TOOLCHAIN_HAS_ZIHINTPAUSE=y
CONFIG_TOOLCHAIN_NEEDS_EXPLICIT_ZICSR_ZIFENCEI=y
```

.config

Linux: RISC-V features configuration

- Configuration
  - $ make ARCH=riscv CROSS_COMPILE=riscv64-unknown-linux-gnu- defconfig
  - fixup based on toolchain support

```
CONFIG_RISCV_ISA_C=y
CONFIG_RISCV_ISA_SVNAPOT=y
CONFIG_RISCV_ISA_SVPBMT=y
CONFIG_TOOLCHAIN_HAS_V=y
CONFIG_RISCV_ISA_V=y
CONFIG_RISCV_ISA_V_DEFAULT_ENABLE=y
CONFIG_RISCV_ISA_V_UCOPY_THRESHOLD=768
CONFIG_TOOLCHAIN_HAS_ZBB=y
CONFIG_RISCV_ISA_ZBB=y
CONFIG_RISCV_ISA_ZICBOM=y
CONFIG_RISCV_ISA_ZICBOZ=y
CONFIG_TOOLCHAIN_HAS_ZIHINTPAUSE=y
CONFIG_TOOLCHAIN_NEEDS_EXPLICIT_ZICSR_ZIFENCEI=y
```

.config

Linux: RISC-V features configuration

- Configuration
  - $ make ARCH=riscv CROSS_COMPILE=riscv64-unknown-linux-gnu- defconfig
  - fixup based on toolchain support

```
CONFIG_RISCV_ISA_C=y
CONFIG_RISCV_ISA_SVNAPOT=y
CONFIG_RISCV_ISA_SVPBMT=y
CONFIG_TOOLCHAIN_HAS_V=y
CONFIG_RISCV_ISA_V=y
CONFIG_RISCV_ISA_V_DEFAULT_ENABLE=y
CONFIG_RISCV_ISA_V_UCOPY_THRESHOLD=768
CONFIG_TOOLCHAIN_HAS_ZBB=y
CONFIG_RISCV_ISA_ZBB=y
CONFIG_RISCV_ISA_ZICBOM=y
CONFIG_RISCV_ISA_ZICBOZ=y
CONFIG_TOOLCHAIN_HAS_ZIHINTPAUSE=y
CONFIG_TOOLCHAIN_NEEDS_EXPLICIT_ZICSR_ZIFENCEI=y
```

.config

```
static inline void cpu_relax(void)
{
#ifdef __riscv_muldiv
        int dummy;
        /* In lieu of a halt instruction, induce a long-latency stall. */
        __asm__ __volatile__ ("div %0, %0, zero" : "=r" (dummy));
#endif

#ifdef CONFIG_TOOLCHAIN_HAS_ZIHINTPAUSE
        /*
         * Reduce instruction retirement.
         * This assumes the PC changes.
         */
        __asm__ __volatile__ ("pause");
#else
        /* Encoding of the pause instruction */
        __asm__ __volatile__ (".4byte 0x100000F");
#endif
        barrier();
}
```

arch/riscv/include/asm/vdso/processor.h

Linux: RISC-V features runtime detection

Linux: RISC-V features runtime detection

- In fact it is not...

Linux: RISC-V features runtime detection

- In fact it is not...
  - mostly static DTS description

```
# allwinner D1s

riscv,isa-base = "rv64i";
riscv,isa-extensions = "i", "m", "a", "f", "d", "c",
        "zicntr", "zicsr", "zifencei", "zihpm";

# StarFive JH7110

riscv,isa-base = "rv64i";
riscv,isa-extensions = "i", "m", "a", "c",
        "zba", "zbb",
        "zicntr", "zicsr", "zifencei", "zihpm";

# RVA22

riscv,isa-base = "rv64i";
riscv,isa-extensions = "i", "m", "a", "c", "f", "d", "h", "v",
        "zba", "zbb", "zbc", "zbs",
        "zicntr", "zicsr", "zifencei", "zihpm", "zihintpause",
        "zicbom", "zicboz", "svnapot",
        "svpbmt", "svinval", "sscofpmf", "sstc";
```

Linux: RISC-V features runtime detection

- In fact it is not...
  - mostly static DTS description
- Some detection is done by kernel
  - misaligned access speed

```
# allwinner D1s

riscv,isa-base = "rv64i";
riscv,isa-extensions = "i", "m", "a", "f", "d", "c",
        "zicntr", "zicsr", "zifencei", "zihpm";

# StarFive JH7110

riscv,isa-base = "rv64i";
riscv,isa-extensions = "i", "m", "a", "c",
        "zba", "zbb",
        "zicntr", "zicsr", "zifencei", "zihpm";

# RVA22

riscv,isa-base = "rv64i";
riscv,isa-extensions = "i", "m", "a", "c", "f", "d", "h", "v",
        "zba", "zbb", "zbc", "zbs",
        "zicntr", "zicsr", "zifencei", "zihpm", "zihintpause",
        "zicbom", "zicboz", "svnapot",
        "svpbmt", "svinval", "sscofpmf", "sstc";
```

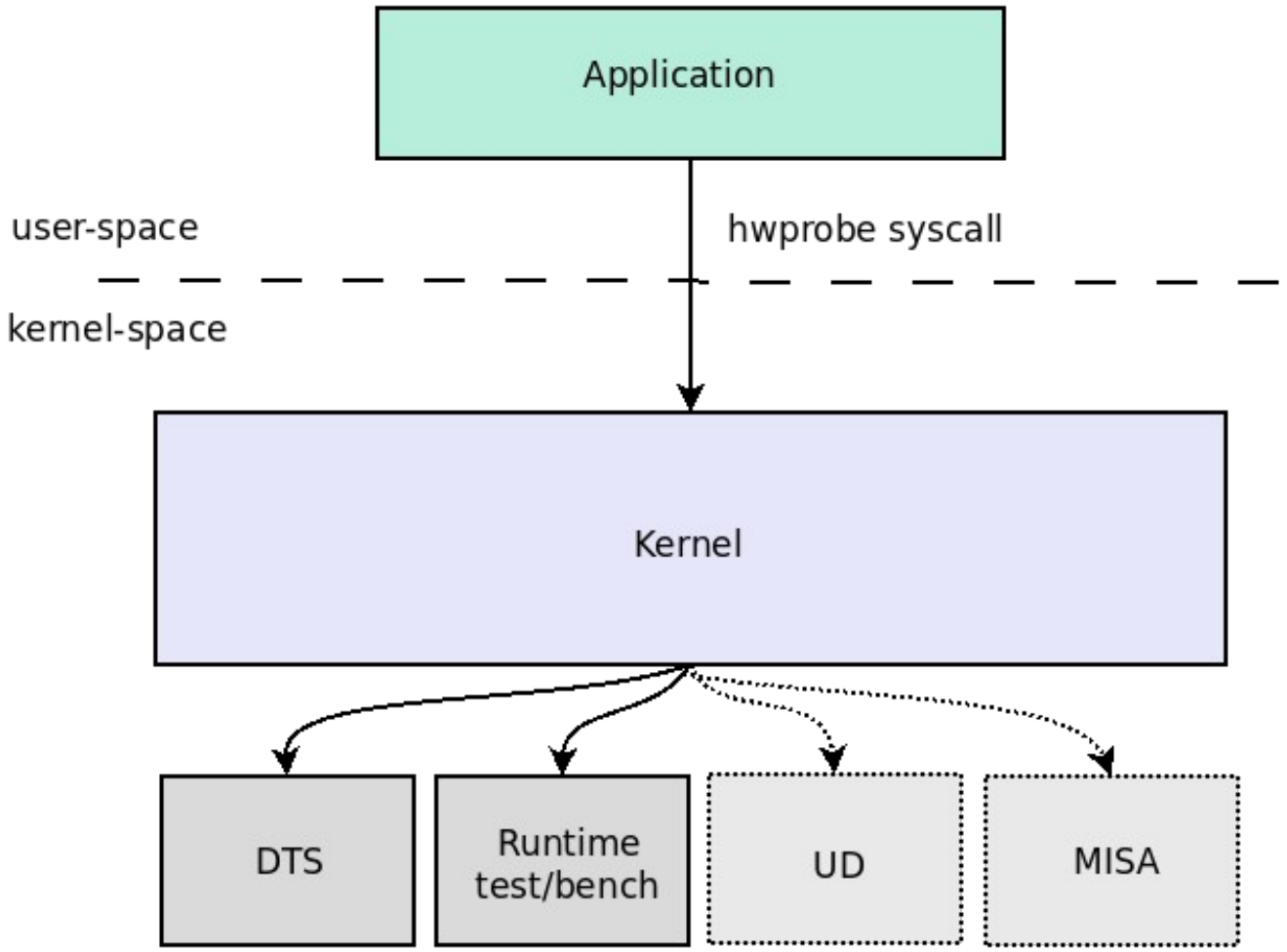
Linux: RISC-V features runtime detection

- In fact it is not...
  - mostly static DTS description
- Some detection is done by kernel
  - misaligned access speed
- MISA register ?
  - ABCDEFG...XYZ
  - not extensible
- RISC-V Unified Discovery TG ?
  - https://github.com/riscv/configuration-structure

```
# allwinner D1s

riscv,isa-base = "rv64i";
riscv,isa-extensions = "i", "m", "a", "f", "d", "c",
        "zicntr", "zicsr", "zifencei", "zihpm";

# StarFive JH7110

riscv,isa-base = "rv64i";
riscv,isa-extensions = "i", "m", "a", "c",
        "zba", "zbb",
        "zicntr", "zicsr", "zifencei", "zihpm";

# RVA22

riscv,isa-base = "rv64i";
riscv,isa-extensions = "i", "m", "a", "c", "f", "d", "h", "v",
        "zba", "zbb", "zbc", "zbs",
        "zicntr", "zicsr", "zifencei", "zihpm", "zihintpause",
        "zicbom", "zicboz", "svnapot",
        "svpbmt", "svinval", "sscofpmf", "sstc";
```

- Intro
- SBI
  - Whys and Wherefores
  - Timeline
- RISC-V support in Linux
  - Timeline
  - Kernel recipes: enable/detect/activate
  - Userpace recipes: detect

Linux: Great Old Ones in user-space



Linux: Great Old Ones in user-space

- Basic hardware info
  - mvendorid/mimpid/marchid
- CPU performance info
  - misaligned support: fast/slow/emulated/unknown
- ISA extensions
  - RVC/RVF/RVD/RVV
  - Bitmanip (Zba/Zbb/Zbc/Zbs)
  - Scalar crypto (Zbkb/Zbkc/Zbkx/Zknd/Zkne/Zknh/Zksed/Zksh/Zkt)
  - Vector ctypto (Zvbb/Zvbc/Zvkb/Zvkg/Zvkned/Zvknha/Zvknhb/Zvksed/Zvksh/Zvkt)
  - and more…

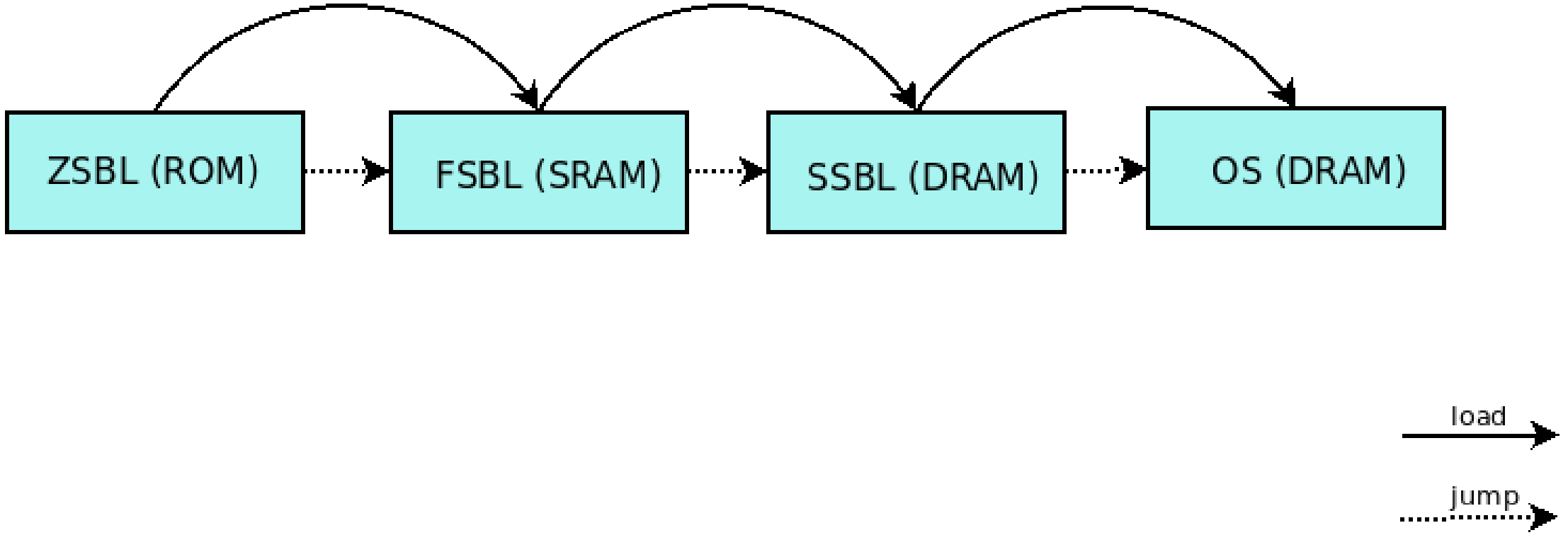
On the radar

- SBI v2.0
  - SUSP, CPPC, STA, PMU snapshots and more
- SBI v3.0
  - DBTR, RPXY and more
- Non-ISA
  - AIA, IOMMU
- ISA
  - more vector and bitmanip kernel optimizations (e.g. crc32, xor, copy),
  - accelerated cryptography using Zvk*
  - Zawrs, Zicbop

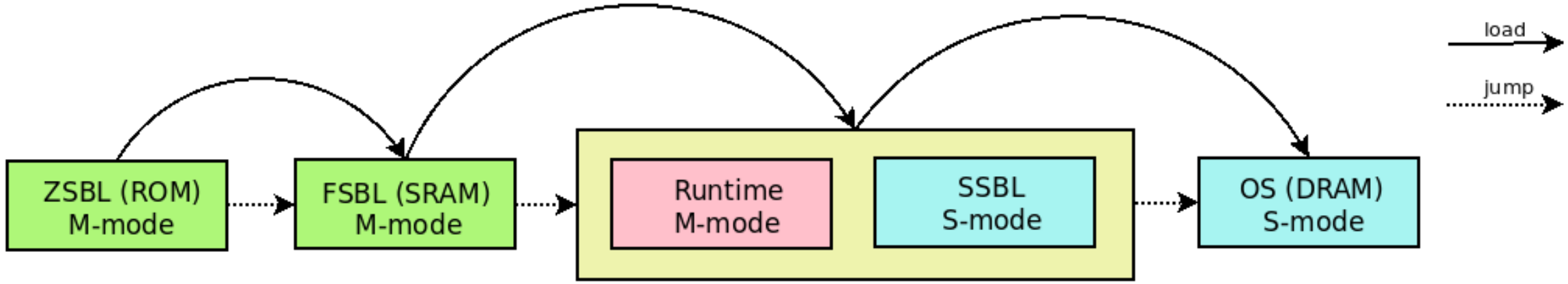
БУДУЩЕЕ
В НАШИХ
РУКАХ

Backup

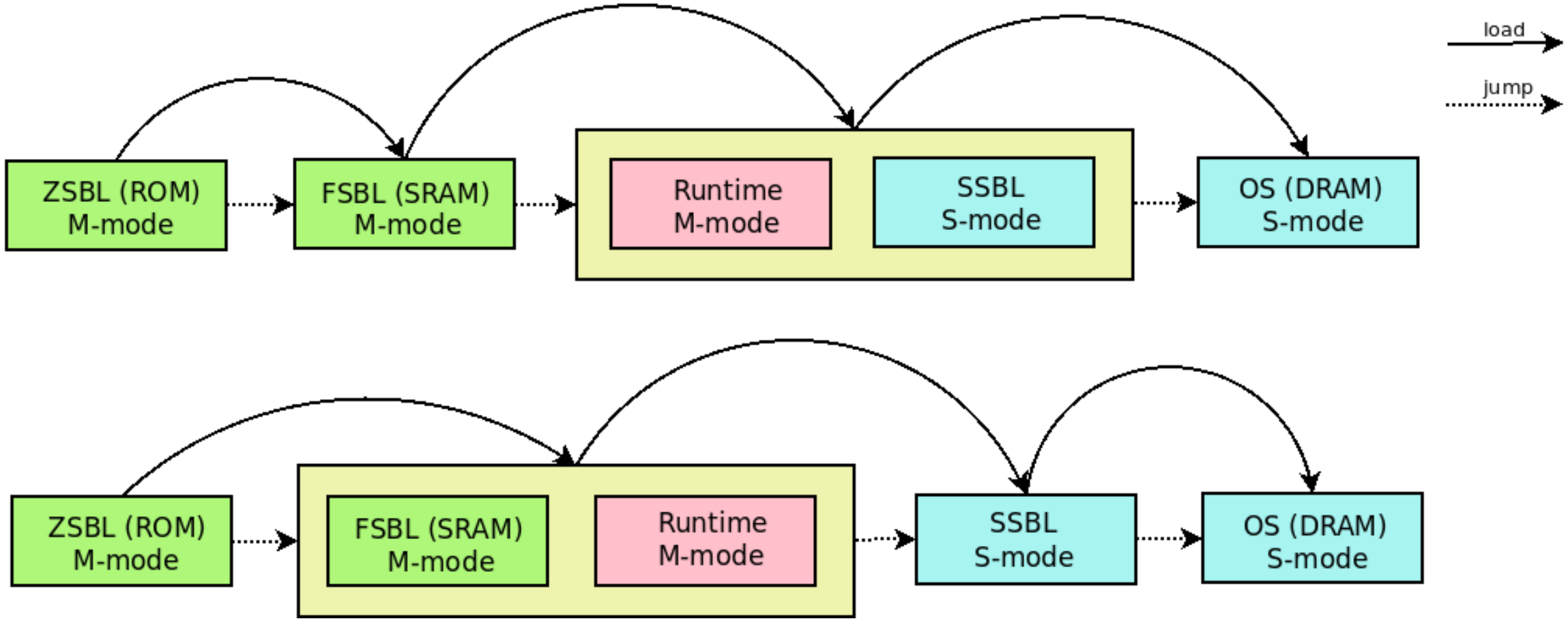
Boot flow: basic



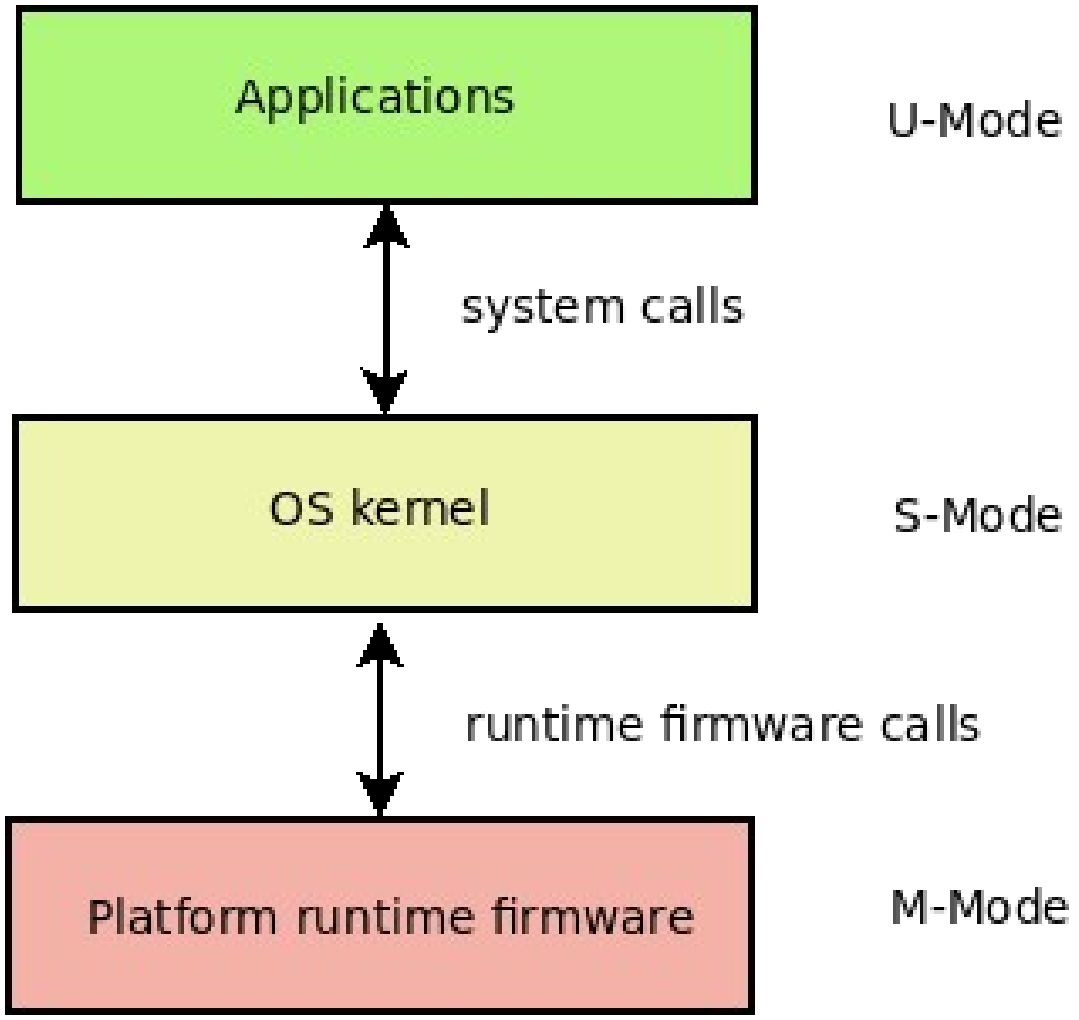
Boot flow: RISC-V



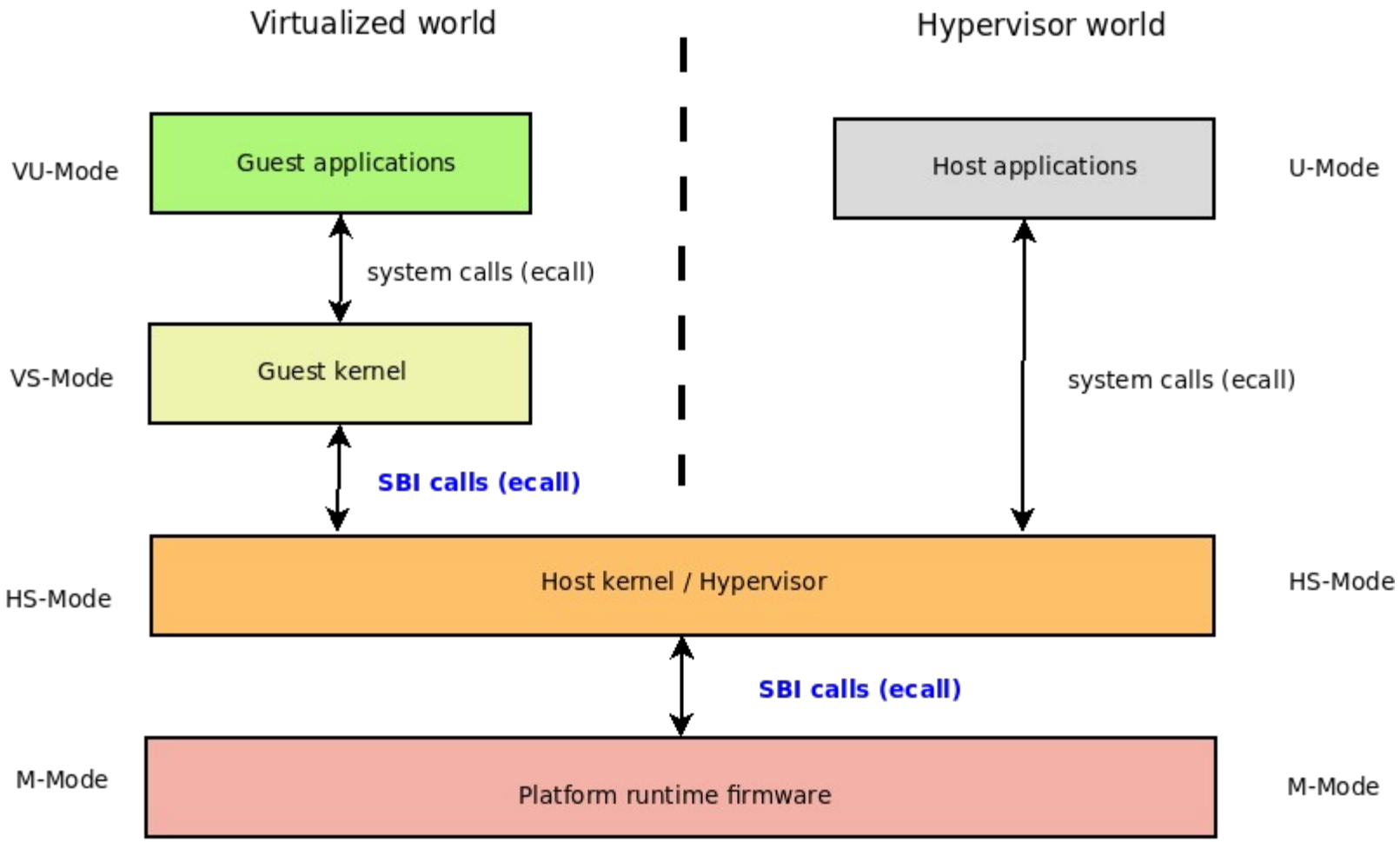
Boot flow: RISC-V



RISC-V Runtime



SBI


Virtualized world
Hypervisor world
VU-Mode
Guest applications
Host applications
U-Mode
system calls (ecall)
VS-Mode
Guest kernel
system calls (ecall)
SBI calls (ecall)
HS-Mode
Host kernel / Hypervisor
HS-Mode
SBI calls (ecall)
M-Mode
Platform runtime firmware
M-Mode

SBI: timeline

Bridge ISA Gap	Higher Privilege Assistance	Virtualization	Platform Abstraction	Confidential Computing	SBI spec version/status
TIME RFENCE (remote fence) IPI (S-mode inter-processor irq) PMU (performance monitoring)	HSM (hart state management)		SRST (system reset)		v1.0 (ratified)
	SUSP (system suspend)	STA (steal time accounting) NACL(nested acceleration) PMU snapshot	CPPC (collab. cpu perf ctrl) DBCN (debug console)		v2.0 (ratified)
DBTR (debug trigger)	FWFT (firmware feature) SSE (supervisor s/w event)		RPXY (RPMI proxy)	COVH COVG COVI	v3.0 (planned)

Source: “The SBI v2.0 and beyond”, RISC-V summit 2023, A. Patil and A.Patra