Will your open source project run on a mainframe? Or on a smartwatch?

...

Elizabeth K. Joseph
IBM

@pleia2@floss.social

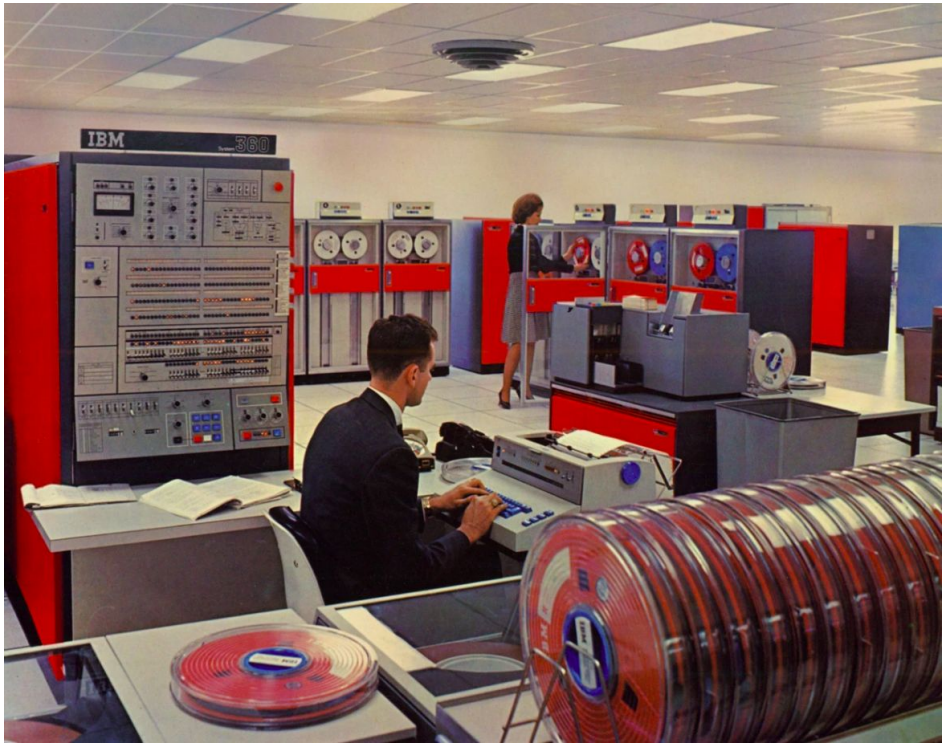
CPOSC 2024

Otherwise known as:

Open Source Software Development for non-x86 architectures

I work on mainframes

No, not those ones.



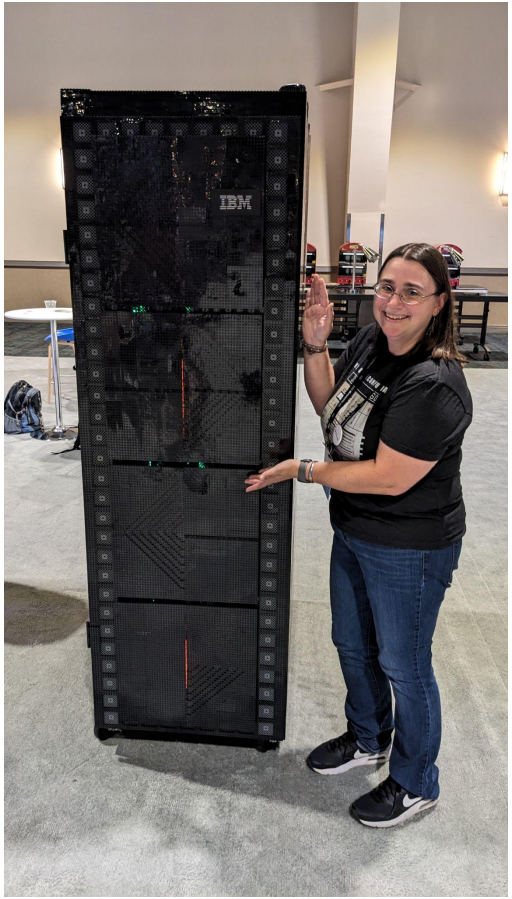
Hack the mainframe?

No, not movie mainframes.

(but An Ode to Movie Mainframes is a lot of fun, and includes that scene from *Tomorrow Never Dies*)

Yeah! Well.

This one is made out of LEGO®



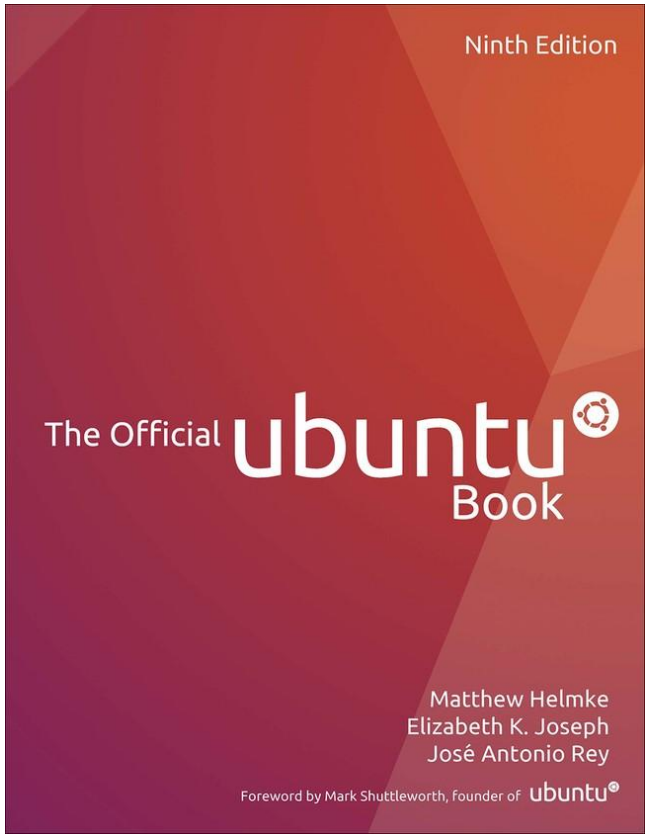
There we go!

IBM z16 and LinuxONE 4

I like Linux

I’m focusing on Linux development 🐧

```
20:33 < cfowler> I think its safe to say that once a new weird
platform gets full linux support then its mature enough to use
```



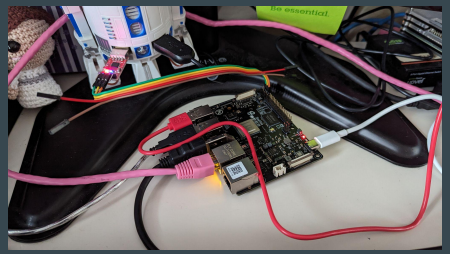
Also, I spoke at CPOSC in 2009 - hello again!

Plus a lightning talk in 2023.

Non-x86_64 architectures you'll likely encounter today

Name	Known as…	Common form factors
ARM	aarch64 / arm64	Microcontrollers, embedded, desktops, servers
IBM Z	s390x	Servers
Power	ppc64le	Servers
RISC-V	riscv64	Microcontrollers, embedded, desktops, servers

Why do various hardware architectures exist?



Differing needs and priorities

- Speed
- Reliability
- Cost
- Size
- Tooling
- Power
- ...

What does it mean for Linux to run on these architectures?

Thousands of packages make up a Linux distribution.

Does that mean they all need to be recompiled?

Yes.

All the packages are recompiled.

👀

So, my software is already ported then? Job done!
Ta-da!

Ok, maybe, but...

Your software must meet the following criteria

- Be part of the Linux distribution, and in a repository where they're building for other architectures
- Be important enough for the distribution developers to fix if it breaks on a specific architecture (instead of just being removed)
- Be simple enough for the distribution developers to fix

So HOW do you test it yourself? Glad you asked!

Cross-compiling

"A cross compiler is a compiler capable of creating executable code for a platform other than the one on which the compiler is running." (wikipedia)

QEMU is the most popular, most broadly supported open source emulation software, and it's built into a lot of tools, CI systems, and instructions for building for other architectures.

You may also come across **Unicorn**, which is based on QEMU but has a focus on emulating CPU operations (rather than the full environment).

Specific architecture tooling collections often also have their own emulation tools and environments.

Cross-compiling: Limitations

- Doesn’t include all features of an architecture
- Lacks full environment (including boot environment, disk, network)
- Limited ability to test interoperability with peripheral devices
- May run slowly

Architecture-native environment

Depending on the architecture, this may be:

- A Single Board Computer (SBC) owned by your project for testing (like a Raspberry Pi, or a VisionFive 2)
- A virtual machine hosted by a provider that has servers made available to community members (programs exist for IBM Z, Power)

Architecture-native environment: Limitations

- Have to get access to one (though there are many programs that make this free for open source projects)
- It's helpful to have a well-connected hosting environment
- Must configure your Linux environment, and keep it up to date
- Still may not have access to every feature and peripheral device you want to test
  - A Raspberry Pi is just one of many types of ARM processor
  - A virtual machine provided to the community for IBM Z may not have all the hardware permutations available to test (though you can ask!)

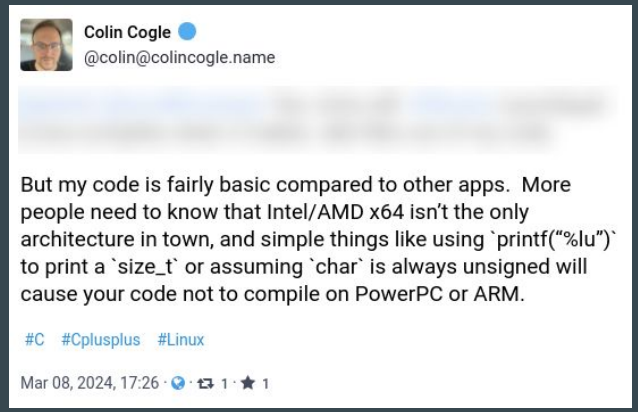
Programming Languages

Confirm support for the compiler/interpreter on the architecture (most are these days)

High level vs. low level languages

Are you taking advantage of any hardware-specific features in the language? Compression? Cryptography? AI/ML features?

Numbers may trip you up: assumptions around signed vs. unsigned integers, memory addressing



(source)

Let’s build a pipeline!

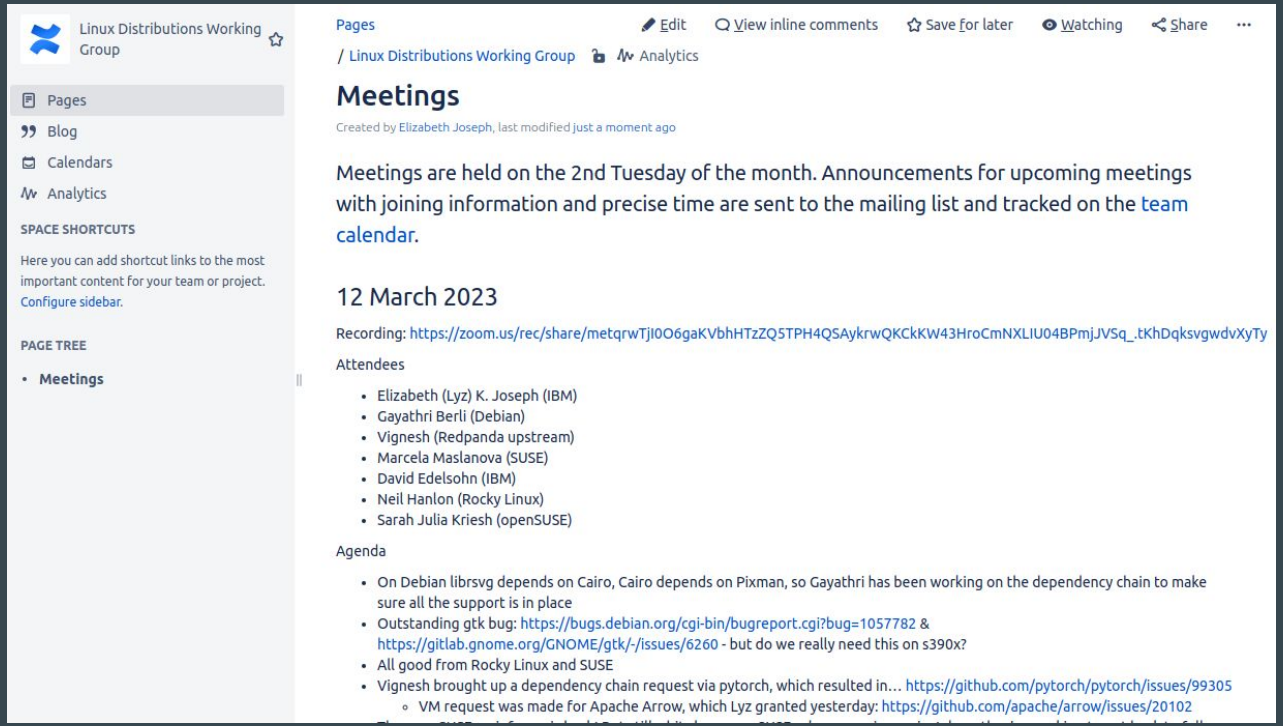
Free-to-Open-Source-Projects CI Systems (2024)

	ARM	IBM Z	Power	RISC-V
CircleCI	Hosted	Self-hosted	Self-hosted	
GitLab (CE)	Hosted	Self-hosted	Self-hosted	
GitHub Actions	Hosted	Self-hosted (unofficial)	Self-hosted (unofficial)	
Jenkins	Self-hosted	Hosted (OSU)	Hosted (OSU)	Self-hosted
Launchpad PPAs	Hosted	Hosted	Hosted	Hosted
openSUSE Build Service (OBS)	Hosted	Hosted	Hosted	Hosted
Travis CI	Hosted	Hosted	Hosted	

Linux Distributions WG (find your people)

Feel like the only one who works on or cares about a specific architecture?

We created a Working Group with the Linux Foundation's Open Mainframe Project.



Developer Resources: ARM

- Documentation
- Education (learning paths, training)
- Community (blogs, forums, chat)
- Tools (Arm Development Studio, Hardware/Software success kits)

https://www.arm.com/developer-hub

Developer Resources: IBM Z

- Free access to s390x-native Virtual Machines
- Links to the various CI systems that offer free access to open source software projects

https://openmainframeproject.org/news/developer-resources-for-linux-on-s390x/

Further community resources for various aspects of IBM Z/LinuxONE at https://community.ibm.com/z

Developer Resources: Power

- Various programs for different members of the community (Partners, companies, individuals, etc)
- Documentation around QEMU usage for your development efforts
- *“The OpenPOWER hub providers ... provide free access to OpenPOWER hardware for development and testing on the platform. Each provider has specific configurations available.”*

https://community.ibm.com/community/user/powerdeveloper/blogs/linda-alkire-kinnunen/2022/08/08/accelerate-your-open-source-development-with-acces

Learn more about the OpenPower community which features working groups, chat, documentation, forums and more at https://openpowerfoundation.org/

Developer Resources: RISC-V

Application for free developer boards to projects, criteria for application include:

- Have clear impact – contributing patches, testing, or documentation – to a significant, established upstream community
- Are proposed by proven open-source software contributors who provide a link to their documented track record in GitHub, GitLab, etc.

https://riscv.org/risc-v-developer-boards/details/

- The RISC-V community also has technical forums, chat, mailing lists, documentation

Questions?

Contact me: Elizabeth K. Joseph lyz@princessleia.com | lyz@ibm.com

Special thanks to:

- Michael Hall, Arm
- Linda Alkire, IBM Power
- James Kulina, OpenPOWER Foundation
- Drew Fustini, RISC-V Ambassador