Exploring 20 Years of Linux and Open Source on the Mainframe

Elizabeth K. Joseph, IBM

#ossummit @pleia2

THE LINUX FOUNDATION

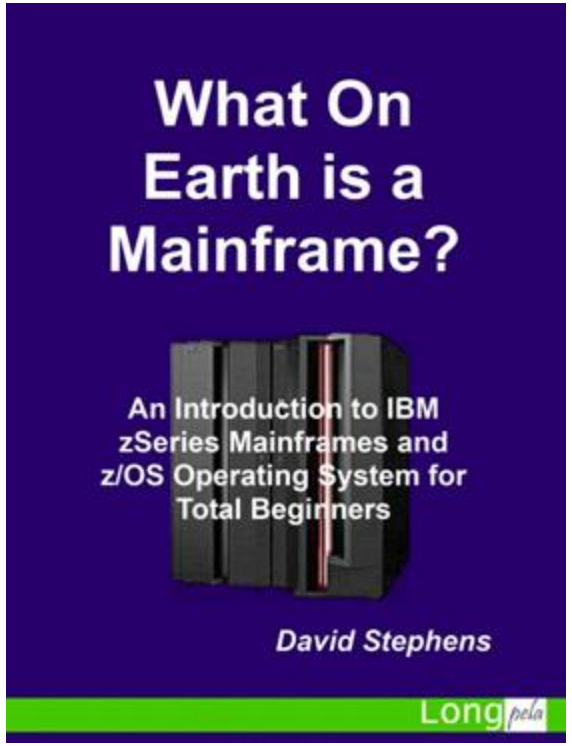
Elizabeth K. Joseph

- Debian and Ubuntu
- OpenStack
- Apache Mesos
- ...and now mainframes? Yep!

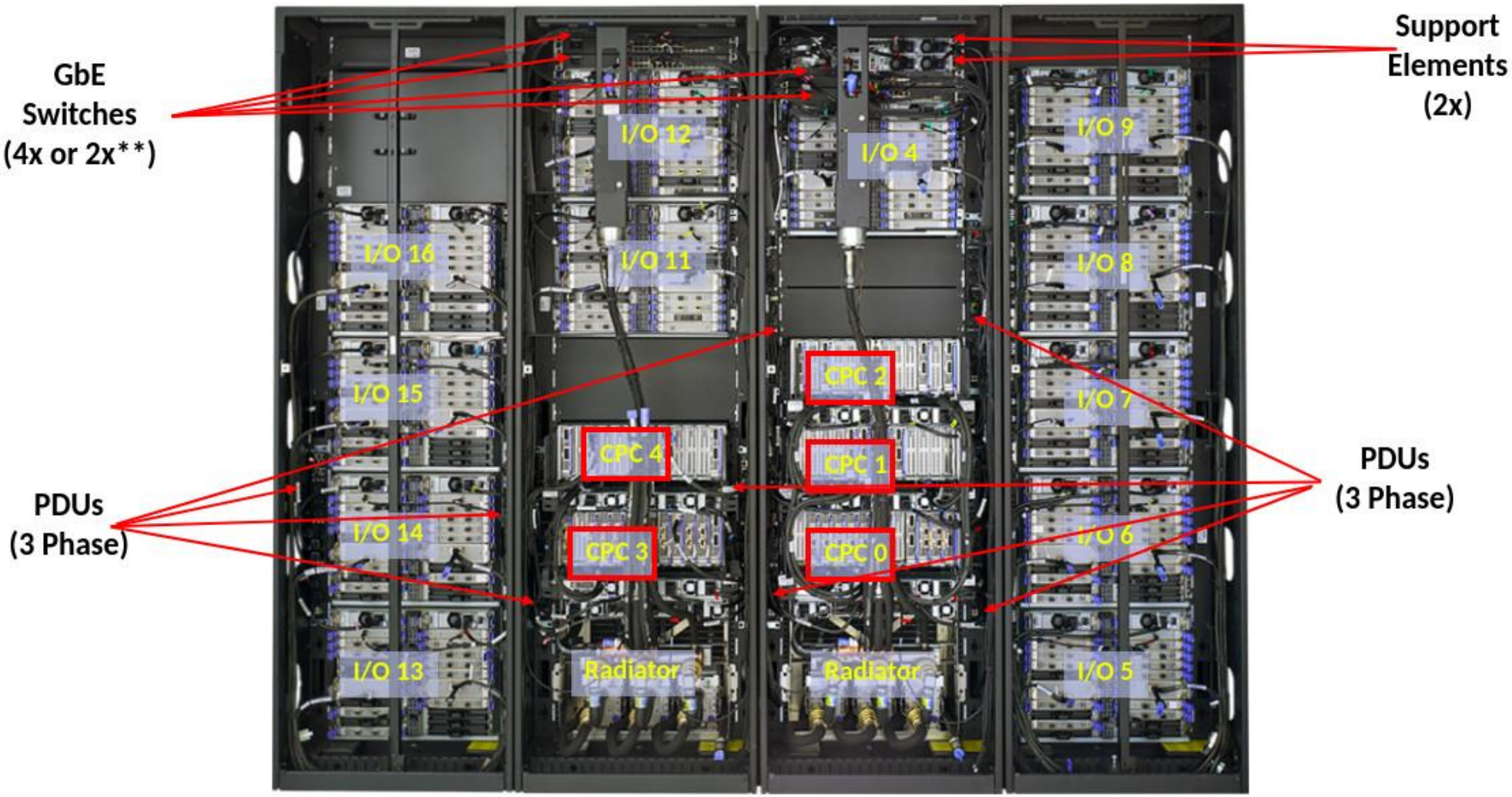
What is a mainframe?

What is a mainframe?

Data, data, data.



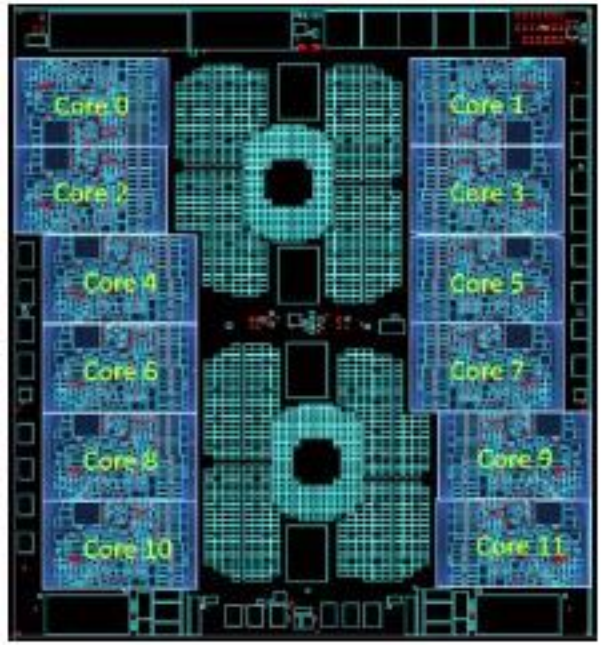
What is a mainframe?



What is a mainframe?

Not x86.

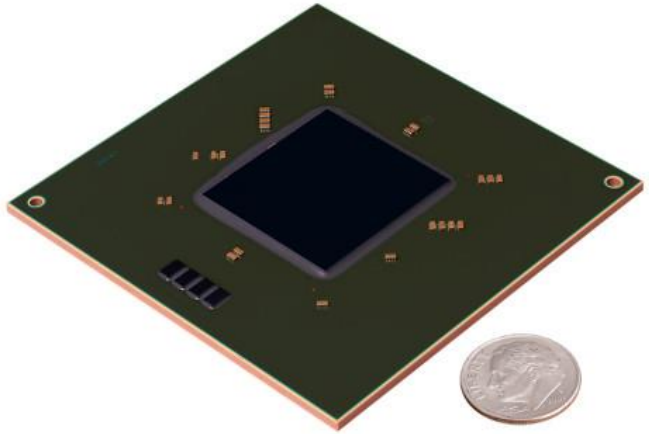
(IBM Z | zArchitecture | s390x)



190 5.2 ghz processor units (PUs), with 12 cores per chip

But also...

- 40TB of RAM
- 60 PCIe control units across 12 PCIe I/O drawers
- 22 dedicated I/O offload processors (SAPs) pre-allocated per system
  https://developer.ibm.com/blogs/systems-inside-the-new-ibm-z15/



Storage - DS8900F

The highest end model, the IBM DS8950F Model 996 has nearly 5.9 PB (5,898 TB) maximum physical capacity

But also...

Storwize® V5100/F

Storwize V7000

FlashSystem® 9100

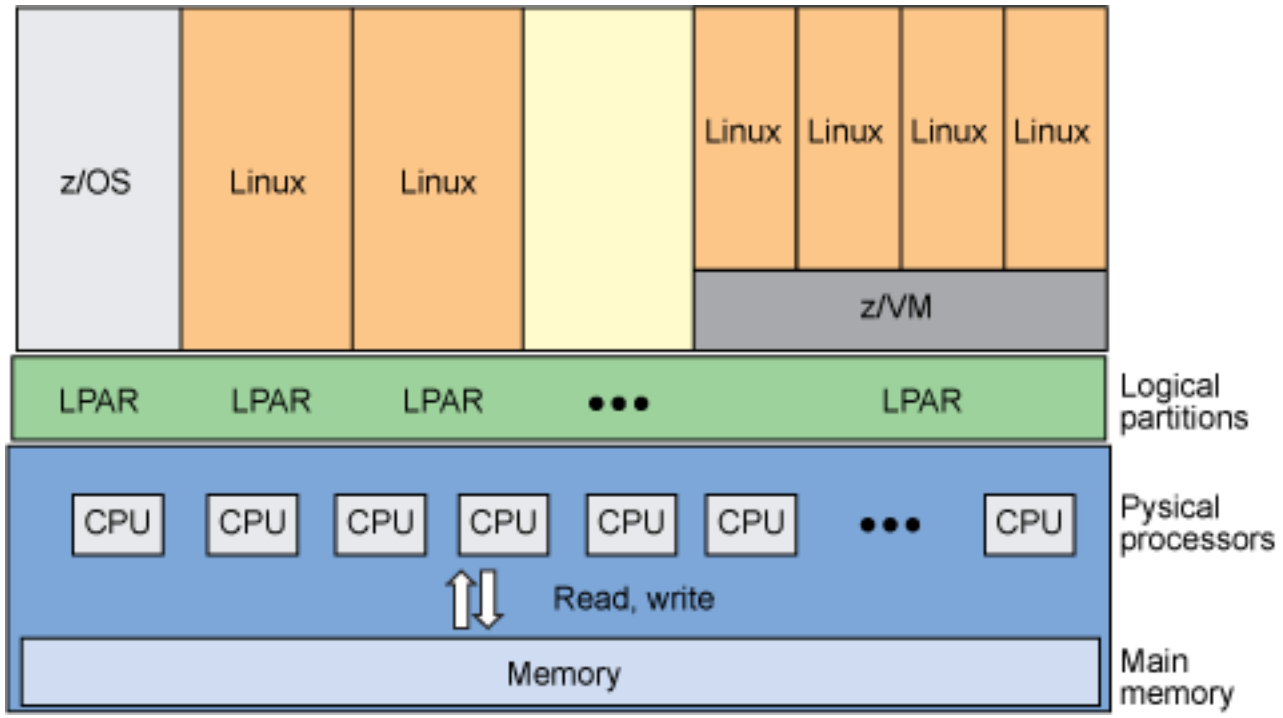
FlashSystem 900

DS8882F

Modern mainframes run Linux!

...and they have for 20+ years

How it works with Linux



There is always some kind of virtualization being used for Linux on Z.

Using z/VM (or KVM!), one or more Linux installs can be put on a single Logical Partition (LPAR).

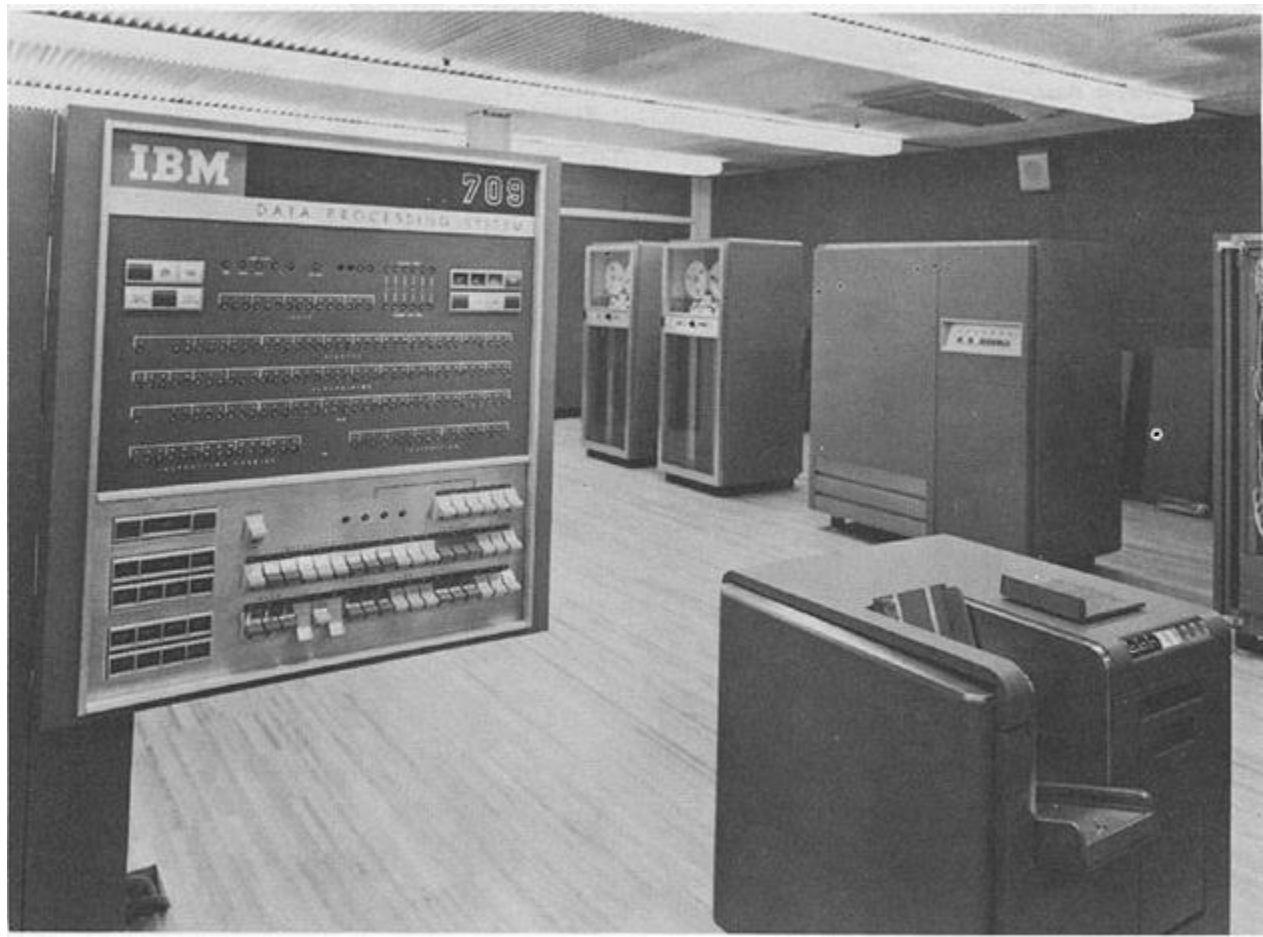
Using Processor Resource and System Manager (PR/SM) a single Linux instance can be installed on a single LPAR.

https://www.ibm.com/developerworks/library/l-systemz/

Once upon a time mainframes lacked time-sharing

Papers discussing time-sharing were published as early as 1959.

Compatible Time-Sharing System (CTSS) was first demoed by MIT on an IBM 709 in 1961.



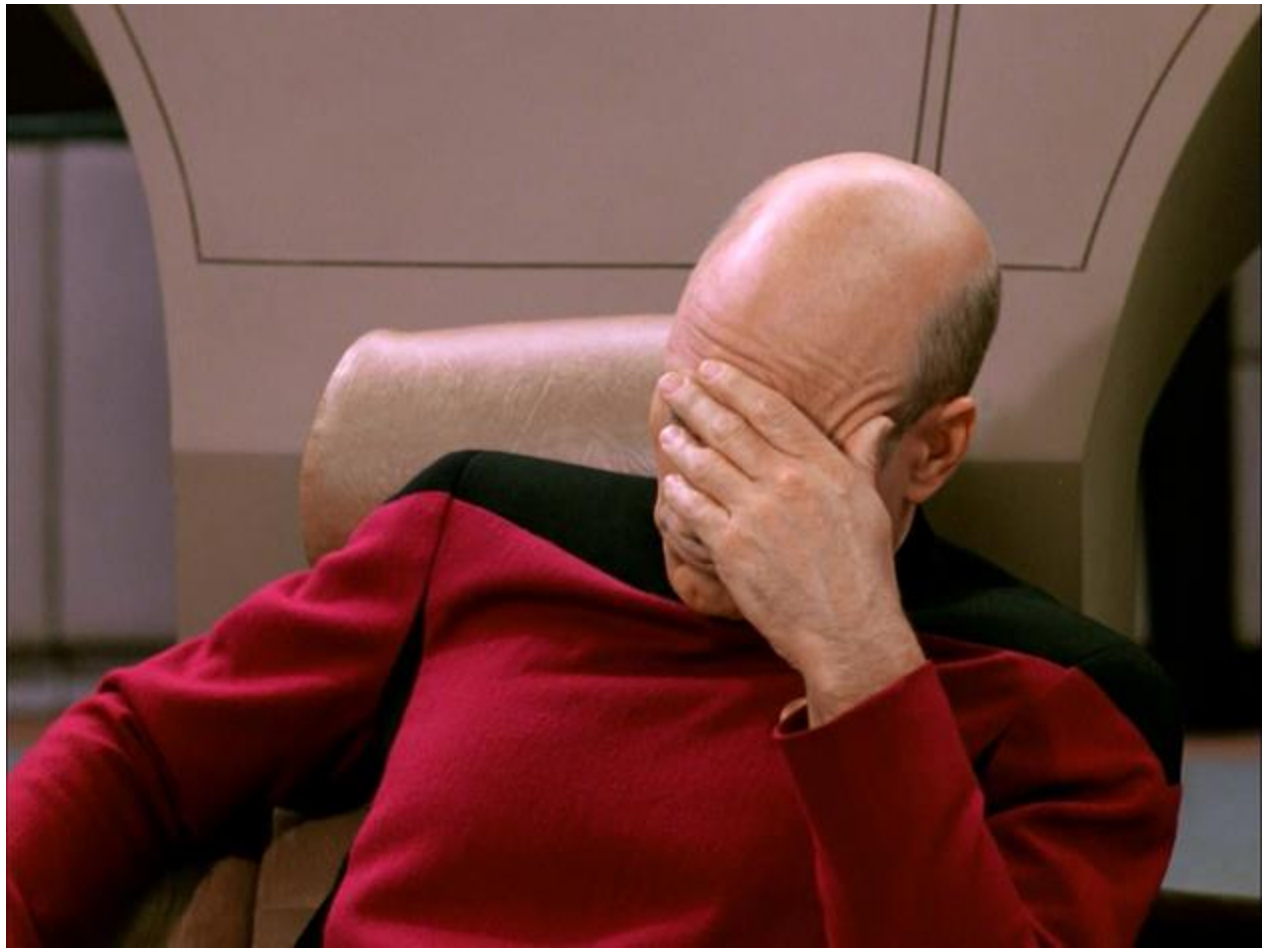
Several iterations later… VM/370, in 1972

Want to know about all those iterations? Melinda Varian has published a fascinating history, available in several formats, on her website:
http://www.leeandmelindavarian.com/Melinda/

IBM: **“I don’t think anyone needs VMs”**

(paraphrased)

The Doubtful Decade.



But it got better

The Doubtful Decade ended and VM community thrived, along with the technology and support from IBM.

In 1994 experimental TCP/IP support was added to VM, adding a key component to supporting Linux 5 years later.

This was a big step forward, previously they had been focused on the proprietary IBM Systems Network Architecture (SNA) for connecting servers.

Linux Origins: Bigfoot

Developed by Linas Vepstas in 1998-1999 as a community effort.

“the **Bigfoot (i370)** port was started first, but is currently stagnant for essentially political, social, and market reasons.”

Source: Linas Vepsta’s site on Linux on s390 https://linas.org/linux/i370.html

Why did the community want it?

“Why? Good question. One we've asked ourselves many times. Why do you do the things you do? If you think about it, you can probably find a hundred rationalizations for what your gut makes you to do. Here's some of ours:

- Stunt
- To Learn
- Because Its There
- Because Its Knarly, Duude!
- I/O
- Address Spaces and Access Lists
- VM
- The Business Model”

Source: https://linas.org/linux/i370-why.html

Linux Origins: Linux for S/390

Linux for S/390 began when “IBM published a collection of patches and additions to the Linux 2.2.13 kernel on December 18, 1999, to start today's mainline Linux on Z. Formal product announcements quickly followed in 2000”

Marist File System was the first Linux distro put together out of Marist College in Poughkeepsie, NY. Think Blue Linux by Millenux in Germany was an early distro with Red Hat packages and the IBM kernel for mainframes. Other commercial editions quickly followed.

Source: https://en.wikipedia.org/wiki/Linux_on_z_Systems

This is the current, actively developed iteration that all the major platforms are part of today.

Why did IBM want it?

IBM “Heist” commercial, 2001 https://www.youtube.com/watch?v=uxg17JlyFas

ComputerWorld (Denmark): Linux on IBM S/390 mainframe

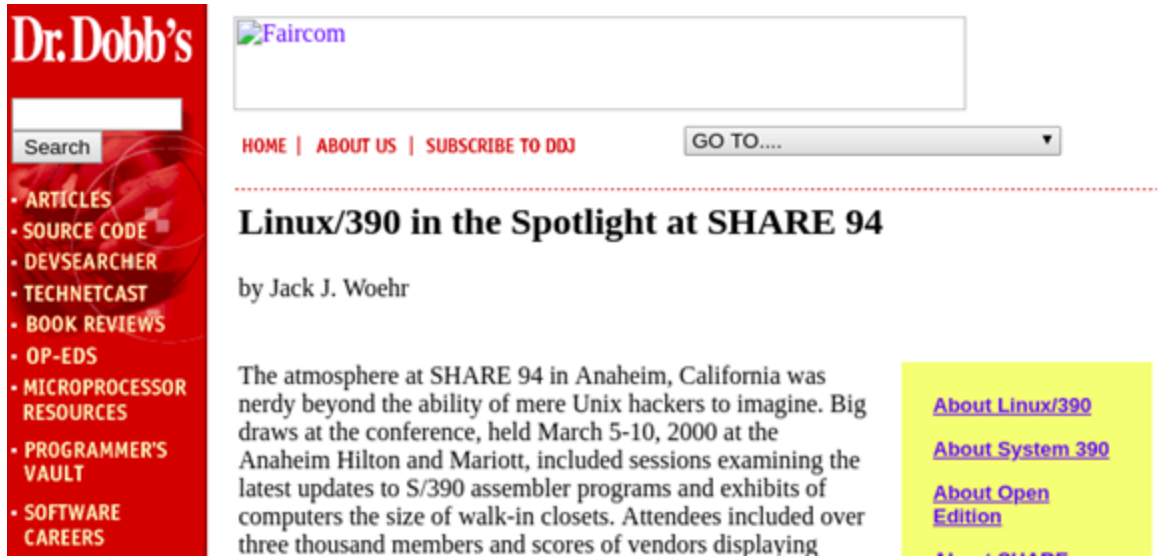
Oct 12, 1999, 01:52 UTC (19 Talkbacks) (Other stories by J.O.S. Svendsen)

[Linux Today reader Hans Schou writes:]

"Friday 8 october 1999 there was a story in the Danish Computerworld about IBM had ported Linux to the S/390 mainframe.

For some people this would not be amazing, as there was a posting to the Linux Kernel list back in march 1999, where a guy asked about DMA buffers and address space. The posting came from 3dlabs.com and the rumor began that IBM was porting Linux to mainframe.

I called IBM today and they confirmed that the development was going on, but they did



20 Years of Linux

Networking

Between Linux LPARs, HiperSocket is used for communication between VMs rather than TCP/IP for speed, responsiveness and reliability.

Storage

Linux can connect and interface with to the storage servers, like the DS8900F.

Portions of Linux rewritten to take advantage of hardware I/O capabilities reducing load from the Central Processor (CP).

Processors

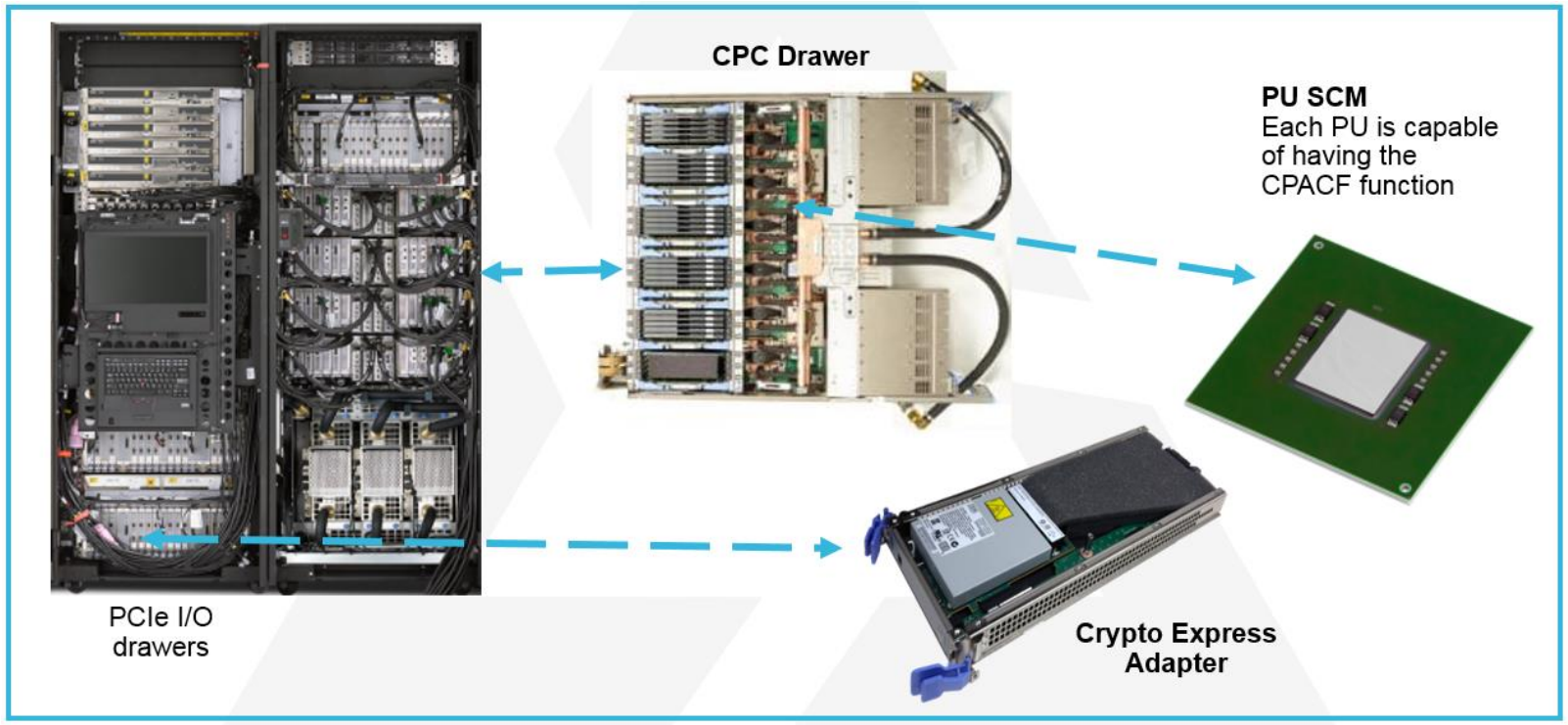
Linux can run on the traditional mainframe Central Processor (CP), but there's also an Integrated Facility for Linux (IFL) processor with some instructions disabled that are used only by z/OS.

Open Source

There are few barriers to compiling for s390x (though it is big-endian), so new open source software is being compiled for the platform every day.

Validated open source software list: https://www.ibm.com/community/z/open-source-software/

Encryption

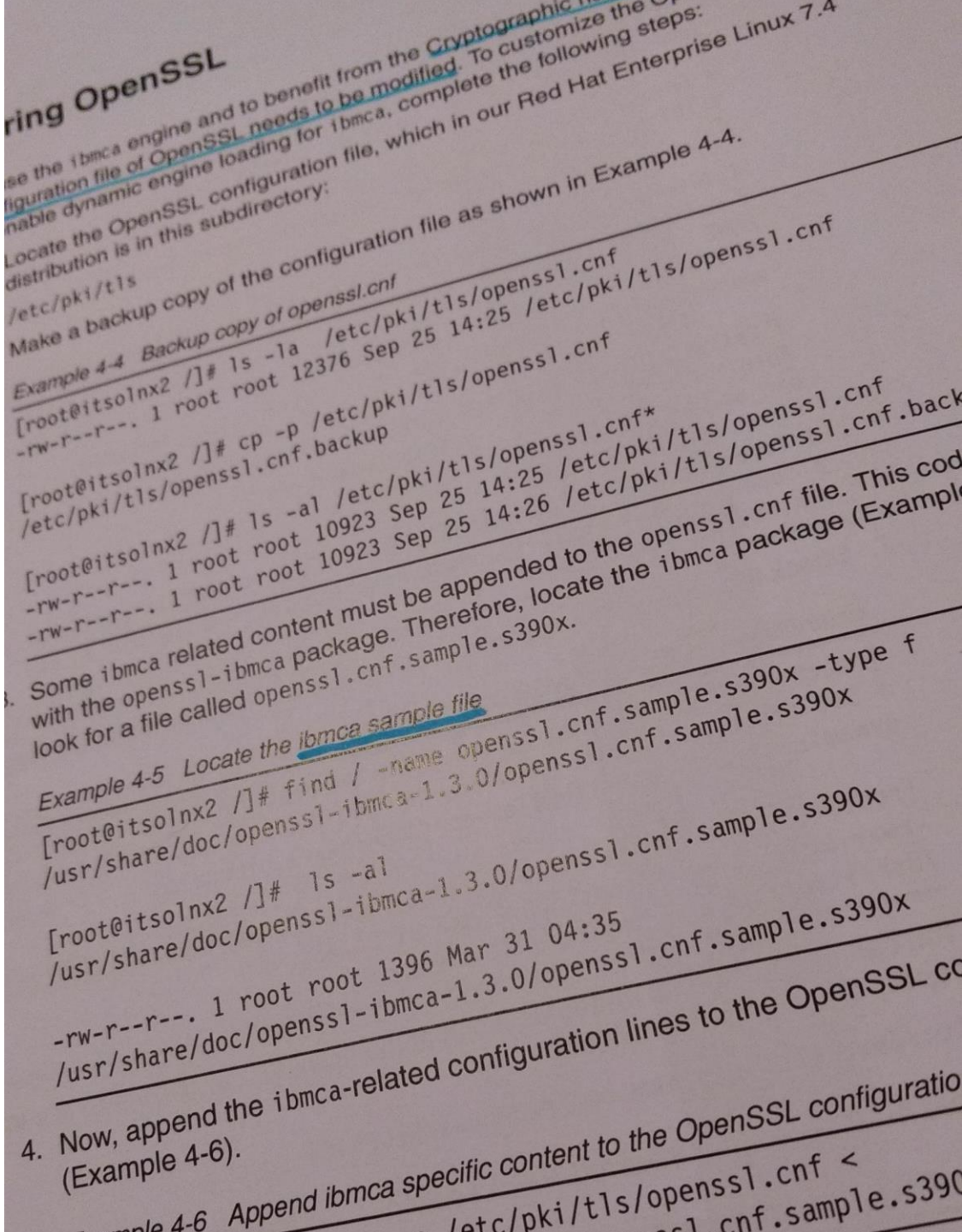

CPC Drawer
PU SCM
Each PU is capable of having the CPACF function
PCIe I/O drawers
Crypto Express Adapter

How it's used on Linux

Security for Linux on System Z
http://www.redbooks.ibm.com/abstracts/sg247728.html

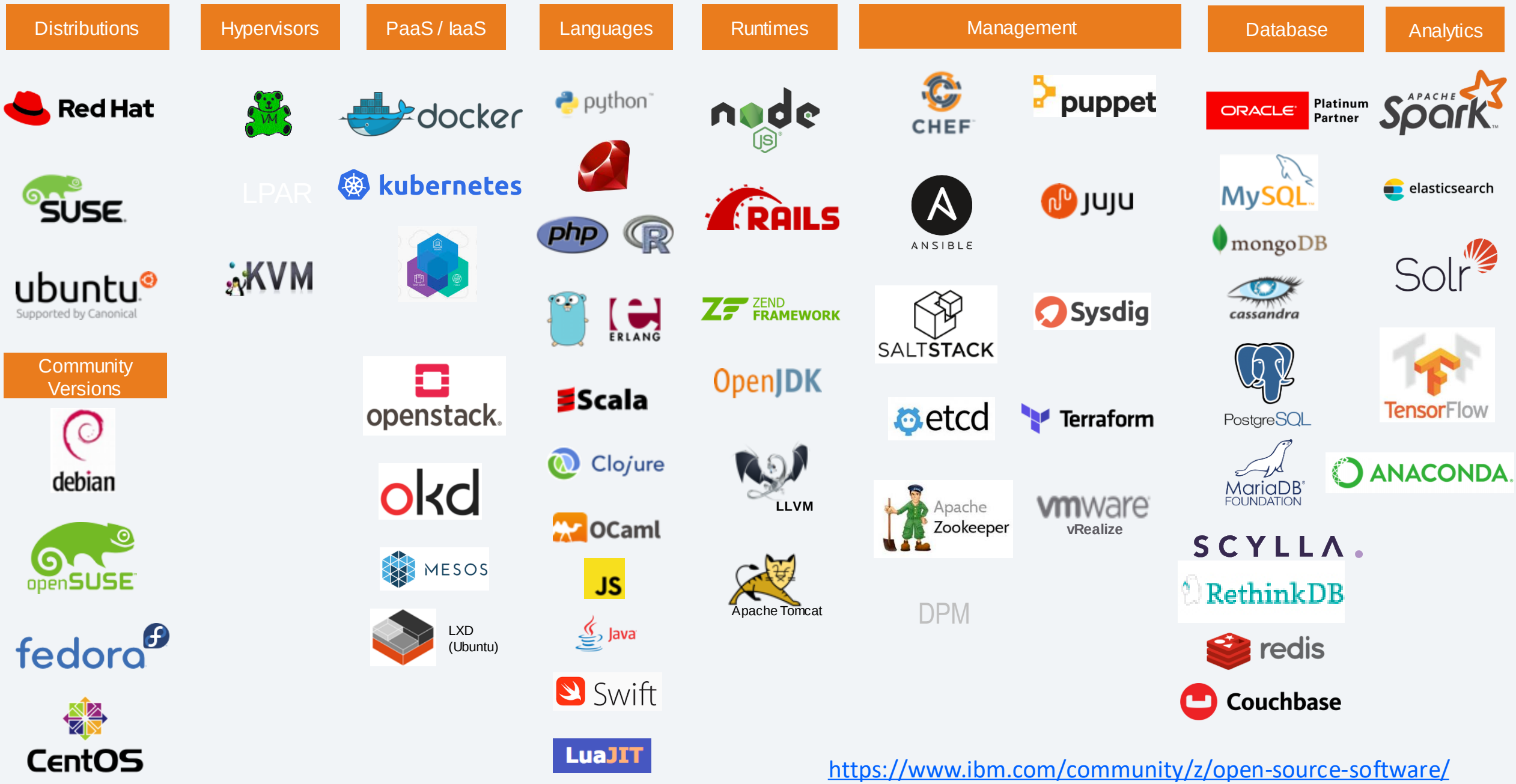
- dm-crypt
- OpenSSL and libcrypto (including for ssh, scp, sftp, Apache mod_ssl...)
- IPSec

And the open source libica crypto library for s390x https://github.com/opencryptoki/libica



LinuxONE

2019, LinuxONE III (z15 with IFLs)



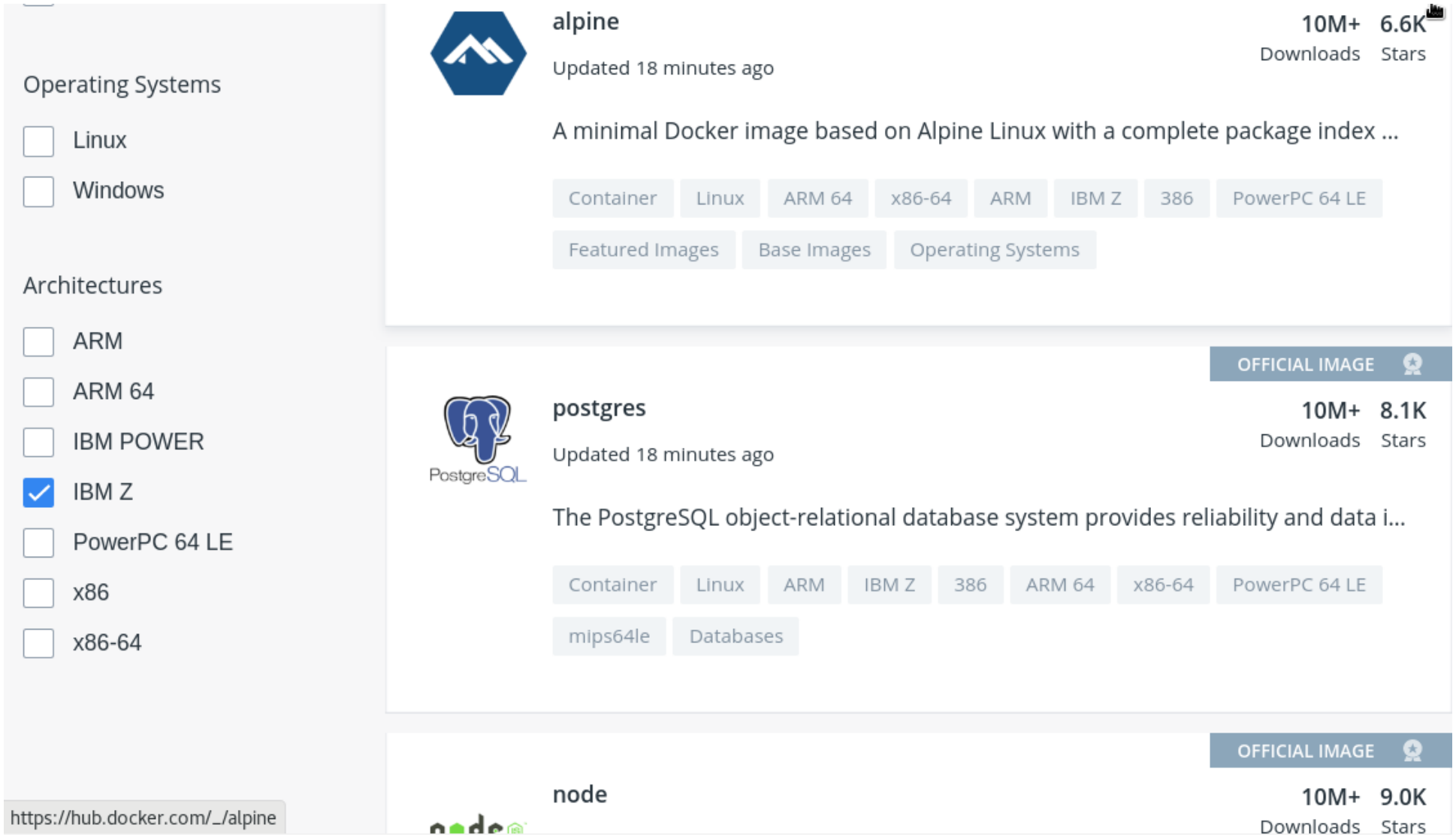
https://www.ibm.com/community/z/open-source-software/

Verified Software List

Packages	RHEL 8.x	Ubuntu 20.x	SLES 15.x	Dockerfile/Image	RHEL 7.x	Ubuntu 18.x	SLES 12.x	Ubuntu 16.x	New Test
Alfresco	5.x	NA	5.x	5.x image	5.x	5.x	5.x	5.x	NA
Ansible	Latest	Distro Latest	Latest	Image Latest	Latest	Distro Latest	Distro Latest	Distro Latest	NA
AntLR	4.x	Distro 4.x	4.x	4.x Image	3.x, 4.x	Distro 3.x, 4.x	3.x, 4.x	Distro 3.x, 4.x	NA
Apache ActiveMQ	Latest	Distro Latest	Latest	5.x Image	Latest	Distro Latest	Latest	Distro Latest	NA
Apache Camel	Download	NA	Download	NA	Download	Download	Download	Download	NA
Apache Cassandra	NA	NA	3.x	3.x Image	2.x, 3.x	2.x, 3.x	2.x, 3.x	2.x, 3.x	NA

https://www.ibm.com/community/z/open-source-software/

Docker Hub


Operating Systems
Linux
Windows
Architectures
ARM
ARM 64
IBM POWER
IBM Z
PowerPC 64 LE
x86
x86-64
alpine
Updated 18 minutes ago
10M+ Downloads
6.6K Stars
A minimal Docker image based on Alpine Linux with a complete package index ...
Container
Linux
ARM 64
x86-64
ARM
IBM Z
386
PowerPC 64 LE
Featured Images
Base Images
Operating Systems
OFFICIAL IMAGE
postgres
Updated 18 minutes ago
10M+ Downloads
8.1K Stars
The PostgreSQL object-relational database system provides reliability and data i...
Container
Linux
ARM
IBM Z
386
ARM 64
x86-64
PowerPC 64 LE
mips64le
Databases
OFFICIAL IMAGE
node
10M+ Downloads
9.0K Stars
https://hub.docker.com/_/alpine

Easy porting: Java and Go

Java has been a staple of mainframe technology for decades now.

Some of the first open source tooling brought to the mainframe was around Java, as Trevor Eddolls writess in a recent blog post:

"If you are working with IMS and like using Java, you can use Spring Boot, Hibernate, JPA, and ORM."

And continues:

"They [IBM] also created a Maven plugin for authoring CICS bundles, which can be used directly in a Java application build toolchain. And they created a Gradle plugin for authoring CICS bundles, which can be used directly in a Java application build toolchain."

https://it.toolbox.com/blogs/trevoreddolls/mainframes-and-open-source-tools-061420

Support for s390x was added as experimental in Go 1.7 https://golang.org/doc/go1.7 (2016)

Today, if your application is written in Go, there's a pretty good chance it'll compile and run fine on Linux on Z.

The crypto/aes package in Go uses hardware-based encryption: "The AES operations in this package are not implemented using constant-time algorithms. An exception is when running on systems with enabled hardware support for AES that makes these operations constant-time. Examples include amd64 systems using AES-NI extensions and **s390x** systems using Message-Security-Assist extensions."

Integration Layers

Swagger is being used extensively as we tie traditional z/OS workloads with modern front-ends.

COBOL has native support for XML and JSON, allowing for Enterprise COBOL being run on the mainframe to directly interact with front-ends.

https://developer.ibm.com/technologies/cobol/blogs/cobol-programming-past-present-future/

A Linux Foundation Project, the Open Mainframe Project:

"is intended to serve as a focal point for deployment and use of Linux and Open Source in a mainframe computing environment. The Project intends to increase collaboration across the mainframe community and to develop shared tool sets and resources. Furthermore, the Project seeks to involve the participation of academic institutions to assist in teaching and educating the mainframe engineers and developers of tomorrow."

Open Mainframe Project projects:
https://www.openmainframeproject.org/projects

Zowe website: https://www.zowe.org/

Zowe Overview (November 2018):

https://www.youtube.com/watch?v=NX20ZMRoTtk

Zowe Webinar (22 February 2019):

https://www.youtube.com/watch?v=XixEltbRmds

Here at the summit, Joe Winchster did a session on **"How Open Mainframe Project's Zowe is Opening up the Mainframe. Who Says You Can't Teach Old Dogs New Tricks."** keep an eye out for the recording!

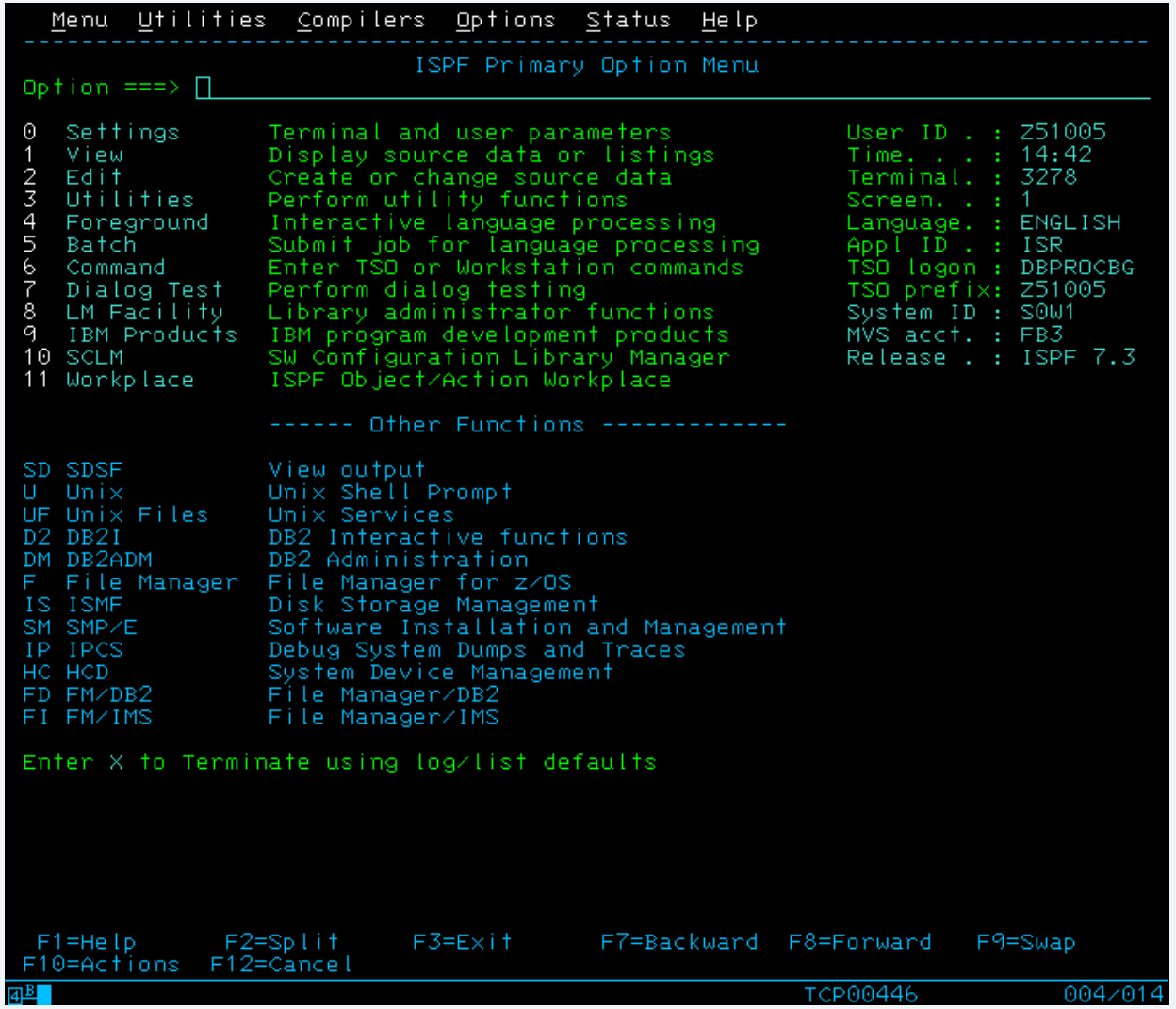
https://ossna2020.sched.com/event/c3PD/how-open-mainframe-projects-zowe-is-opening-up-the-mainframe-who-says-you-cant-teach-old-dogs-new-tricks-joseph-winchester-ibm

Zowe

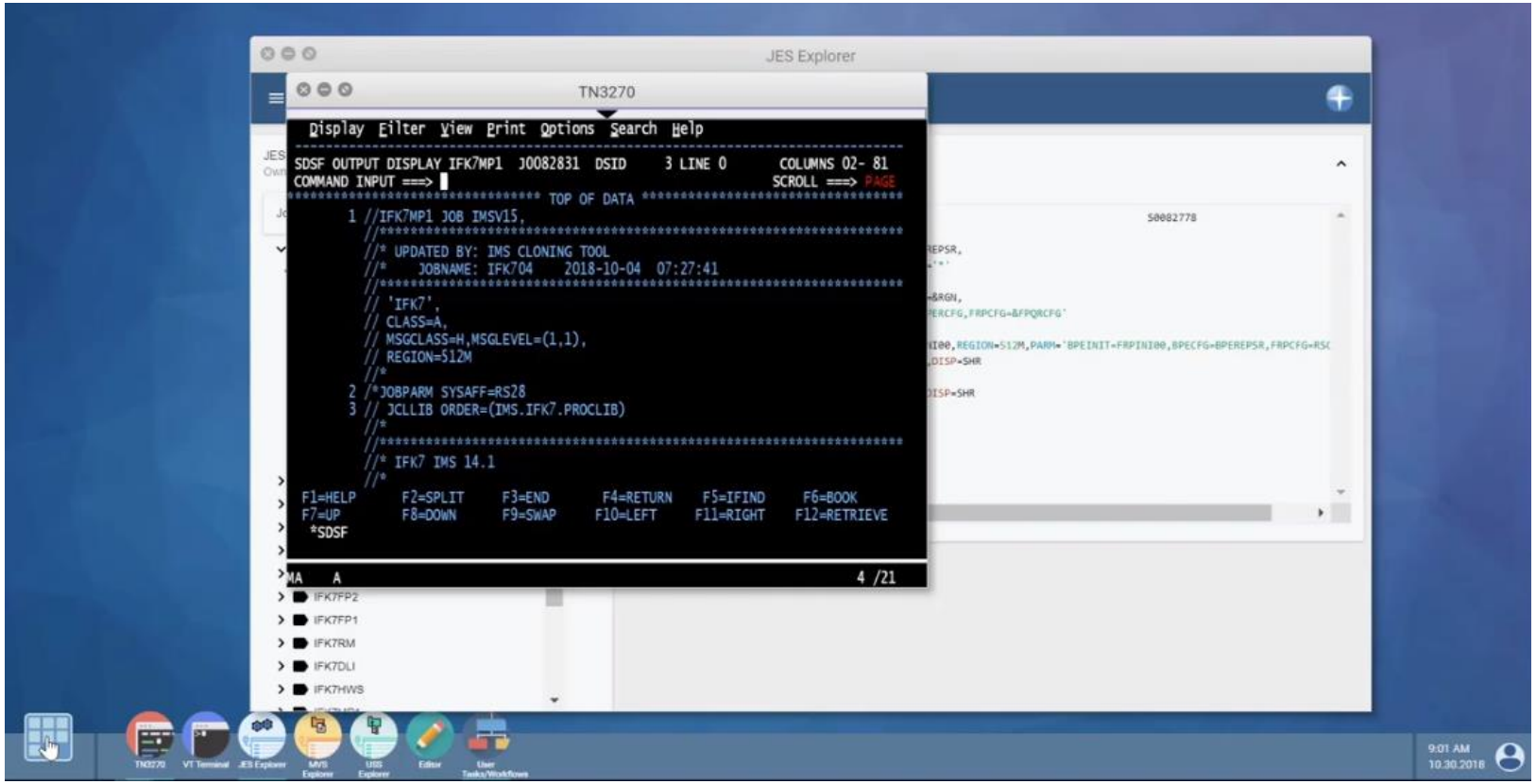
Zowe is a new open source software framework that provides solutions that allow development and operations teams to securely, manage, control, script and develop on the Mainframe like any other cloud platform. Zowe is the first open source project based on z/OS.

Traditional interaction: ISPF

(Interactive System Productivity Facility)



Modern interaction: Zowe Web Desktop



- Desktop-like environment, in your web browser
- Includes a 3270 terminal
- Provides GUI-based explorers for files and datasets with ways to manipulate data

Modern interaction: Zowe CLI

Modern command line tool

- Execute zowe commands from standard MacOS/Linux terminal
- Includes core commands for interacting with mainframe-specific needs (TSO, batch processing)
- Extendable for your specific needs, or those of vendors who wish to offer support for their tooling

Can build IDE extensions, a Visual Studio Code extension already exists.

Automation and scripting, including CI/CD pipelines!

Example usage, Using the Zowe CLI to edit a data set:

https://docs.zowe.org/stable/user-guide/zowe-getting-started-tutorial.html#using-the-zowe-cli-to-edit-a-data-set

Modern interaction: Zowe API

API gateway that can abstract out the various number of services running on the mainframe that you may wish to interact with: job services, dataset services, and more

Open source and vendor products are now leveraging the API for new products that interact with the mainframe

Other Open Mainframe Projects

Ambitus

Ambitus fosters a community that will help educate developers about all open source technologies on z/OS

Anomaly Detection Engine for Linux Logs (ADE)

ADE detects anomalous time slices and messages in Linux logs (either RFC3164 or RFC5424 format) using statistical learning.

Atom language syntax highlighting for z/VM

Helping connect the next generation with mainframe and open source, the internship program has help students over the past several years become contributors to open source on mainframe, as well as develop the skills for a long career in technology.

Other Open Mainframe Projects

Feilong

Feilong is an open source z/VM cloud connector project under the Open Mainframe Project umbrella that will accelerate the z/VM adoption, extending its ecosystem and its user experience.

Mentorship Program

Helping connect the next generation with mainframe and open source, the mentorship program has helped students over the past several years become contributors to open source on mainframe, as well as develop the skills for a long career in technology.

Polycephaly

Polycephaly enables developers to build z/OS source code files with Jenkins and Git

Other Open Mainframe Projects

And more new projects!

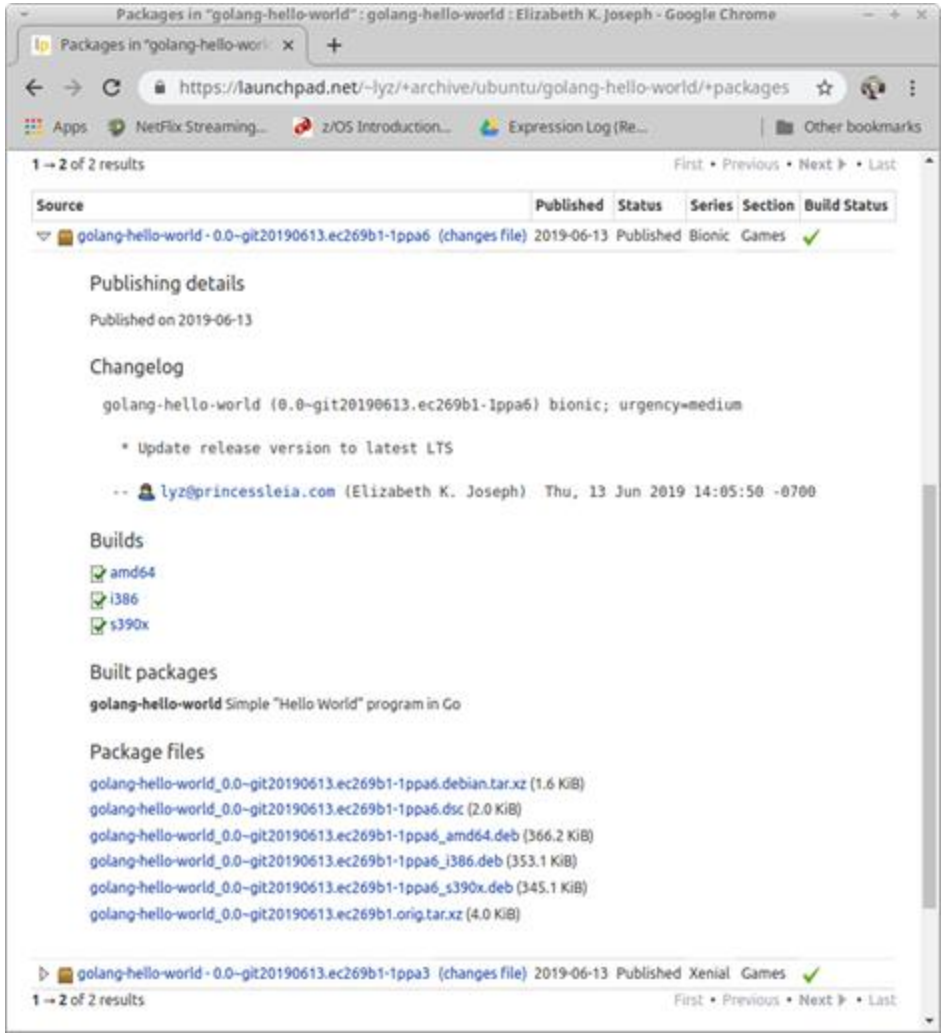
- cobol-programming-course
- software-discovery-tool
- omp-education

GitHub Project:https://github.com/openmainframeproject

Your software?

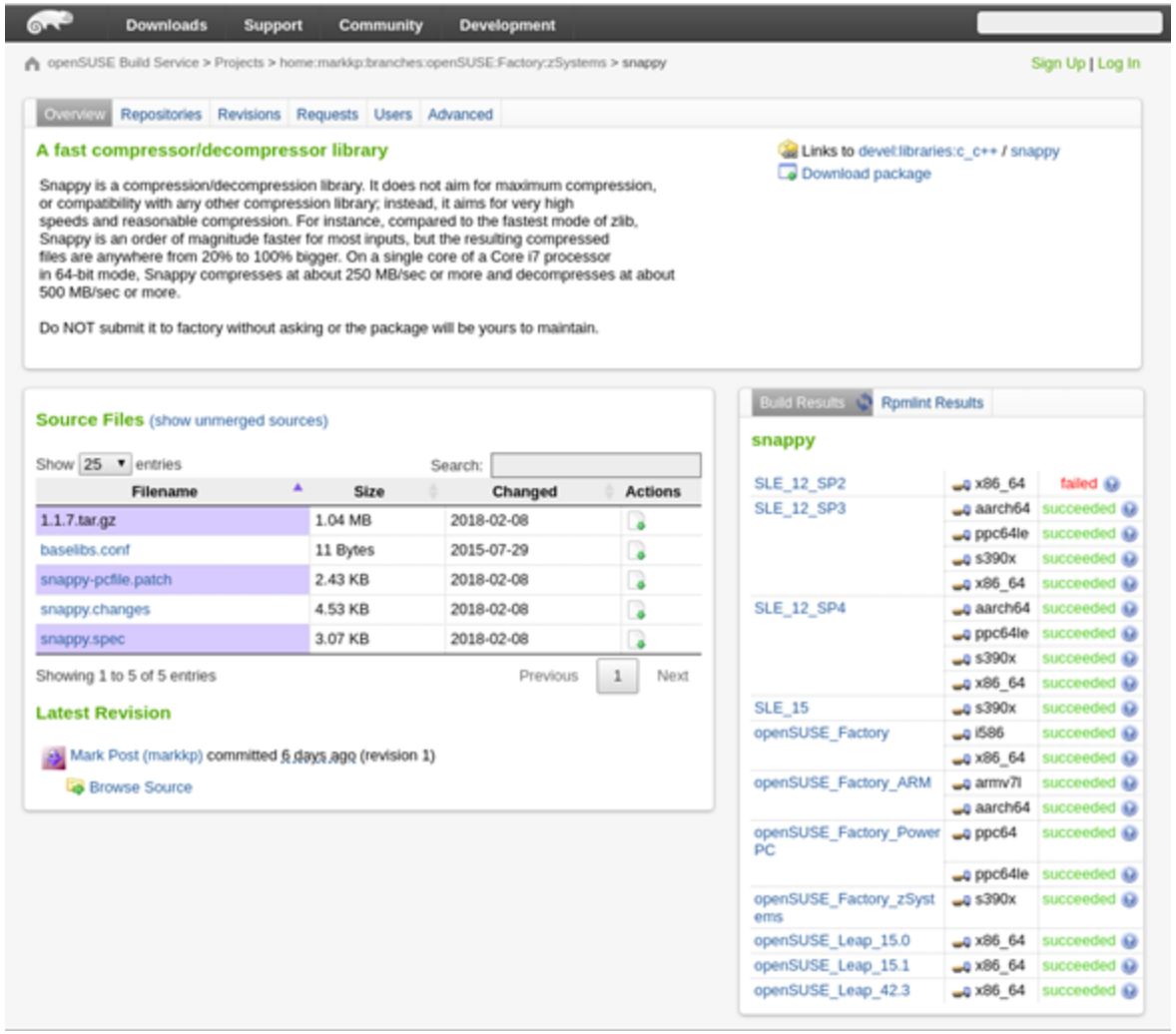
Build it for the mainframe!

Self-Service s390x: Ubuntu PPAs



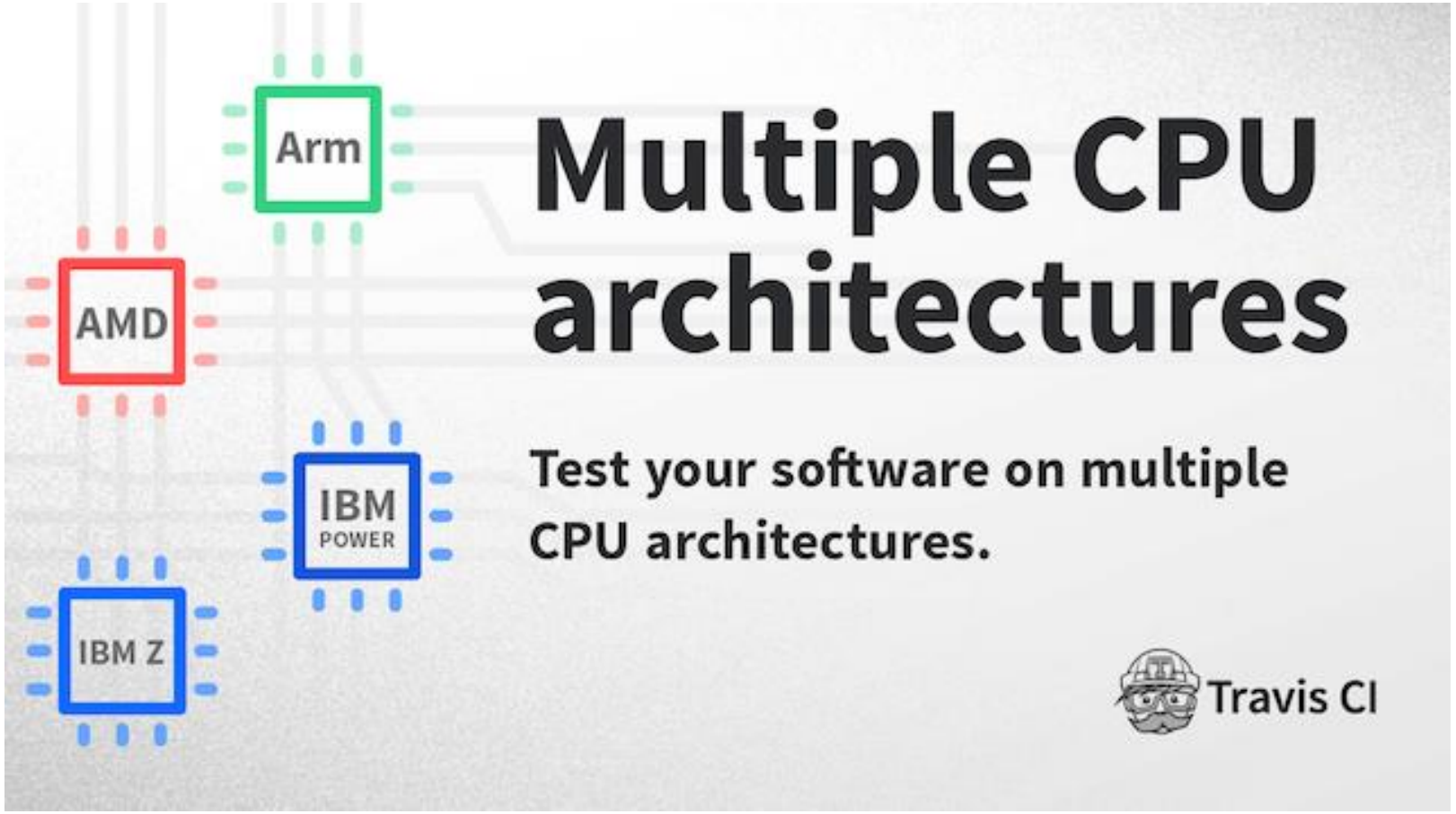
https://princessleia.com/journal/2019/06/building-a-ppa-for-s390x/

Self-Service s390x: openSUSE Build Service



https://build.opensuse.org/

Self-Service s390x: Travis CI



https://blog.travis-ci.com/2019-11-12-multi-cpu-architecture-ibm-power-ibm-z

Unleash the power to innovate with IBM **LinuxONE**™ Community Cloud

The IBM LinuxONE Community Cloud provides a no charge, self-provisioned SUSE or Red Hat virtual machine on an IBM LinuxONE Enterprise Server (s390x architecture) to develop, test and run your apps.

https://developer.ibm.com/linuxone

Thank you!

Elizabeth K. Joseph lyz@ibm.com | lyz@princessleia.com

@pleia2

THE LINUX FOUNDATION

THE LINUX FOUNDATION

OPEN SOURCE SUMMIT

NORTH AMERICA