

Mainframe a la  
mode:

Developer  
resources  
for porting your  
Linux app to IBM Z  
and LinuxONE

Elizabeth K. Joseph, Developer Advocate

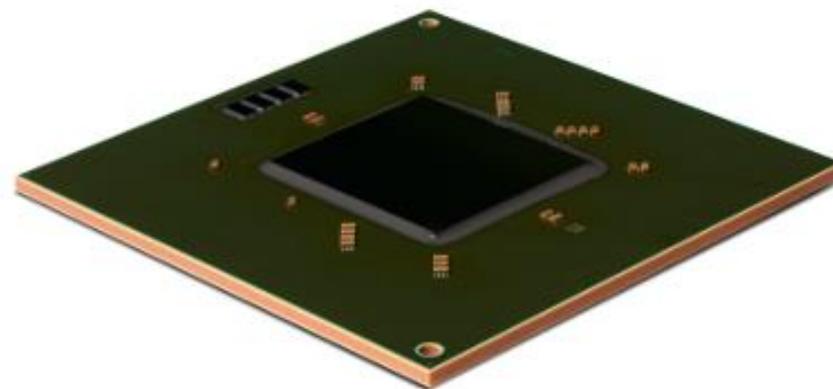
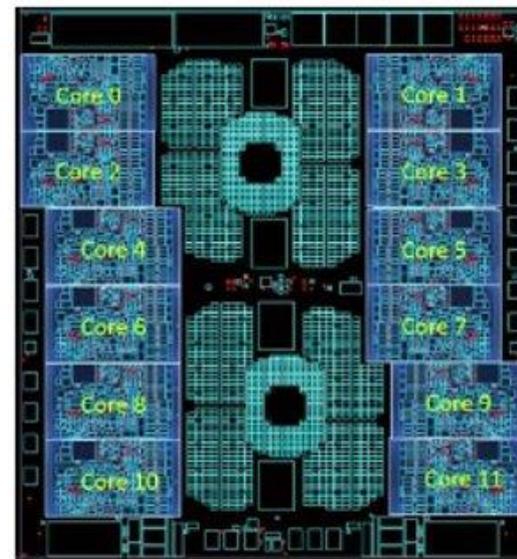
@pleia2

# IBM Z / s390x / zArchitecture

190 5.2ghz 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



# Linux on IBM Z

- Started out as the "Bigfoot" (i370) port by several community members in 1998-99.
- IBM released the first Linux kernel patches to support s390x in December 1999.
- In October 2000, SUSE Linux Enterprise Server became the first, still in production, enterprise Linux to support s390x.
- Red Hat quickly followed as the second, still in production, enterprise Linux for the mainframe.
- Ubuntu support was announced in 2016 and began with Ubuntu 16.04.

# Linux Today

Announced at the Linux Foundation's LinuxCon 2015, IBM released the first Linux-only mainframe, the IBM LinuxONE.

Today's LinuxONE is in its third iteration, with the LinuxONE III released in September 2019.



2015: LinuxONE Emperor & Rockhopper



2017: LinuxONE Emperor II & Rockhopper II



2019: LinuxONE III

# IBM Z has an open source legacy!

- In 1955, the volunteer-run SHARE Inc was founded.
- A key resource for this organization was the SHARE library of software that systems programmers would share among their peers, freely.
- In 1959, SHARE released the SHARE Operating System (SOS), one of the first true "operating systems"<sup>1</sup> and Wikipedia says of SOS:
  - *"SOS was one of the first instances of "commons-based peer production" now widely used in the development of free and open-source software such as Linux and the GNU project."*

<sup>1</sup> [https://en.wikipedia.org/wiki/SHARE\\_\(computing\)](https://en.wikipedia.org/wiki/SHARE_(computing))



Are other projects  
porting their apps?

Yep!

# Growing IBM Z & LinuxONE Open Source Ecosystem

## Linux Distributions & Virtualization



## Community Versions



## Networking & Monitoring



## Cloud & Container Services



## Languages & Runtimes



## DevOps/Automation



## Middleware



## Big Data, Observability, Analytics



## Databases



# IBM Z & LinuxONE Official Docker Images

Open Source Software available in Docker Hub as Official Docker Images

[hub.docker.com](https://hub.docker.com)

## Linux Distributions



## Cloud , Web, Languages & Runtimes



## DevOps/Automation



## Big Data, Observability, Analytics



## Networking & Monitoring



## Middleware



## Databases



# Finding Open Source Software for Linux

- Go directly to the project, do they have s390x builds?
- Ask your vendor, is there a port they maintain?
- Open Mainframe Project Landscape: <https://landscape.openmainframeproject.org/>
- Verified Software List from IBM: <https://www.ibm.com/community/z/open-source-software/>
- DockerHub (IBM Z search): <https://hub.docker.com/search?type=image&architecture=s390x>
- Open Mainframe Project Software Discovery Tool (in development!)  
<https://www.openmainframeproject.org/projects/software-discovery-tool>





OPEN  
**MAINFRAME**  
PROJECT

- Full project hosting, including code, and mailing lists
- Blogs and podcasts of general interest to the open source mainframe community
- Slack and forums for communication among participants
- Project support for 3rd party open source projects seeking infrastructure (VMs, CI/CD services)
- New in 2020: Annual conference!

# IBM LinuxONE Community Cloud

*"The IBM LinuxONE Community Cloud is a no-charge, 24 x 7, enterprise-grade, open access, shared public cloud environment on IBM's LinuxONE platform. Developers, students, professors, entrepreneurs, or anyone from all over the world can sign up for 120-day access to a virtual server with full access to develop, test, or run open source applications on LinuxONE, or to access any of the other services offered."*

Visit <https://developer.ibm.com/linuxone> to get started

And join the LinuxONE Community Cloud Community at <https://www.ibm.com/community/z/linuxone-cc/>

Package in "golang-hello-world" : golang-hello-world : Elizabeth K. Joseph - Google Chrome

lp Packages in "golang-hello-world" x +

https://launchpad.net/~lyz/+archive/ubuntu/golang-hello-world/+packages

Apps NetFlix Streaming... z/OS Introduction... Expression Log (Re... Other bookmarks

1 → 2 of 2 results First • Previous • Next ▶ • Last

Source	Published	Status	Series	Section	Build Status
golang-hello-world - 0.0~git20190613.ec269b1-1ppa6 (changes file)	2019-06-13	Published	Bionic	Games	✓

### Publishing details

Published on 2019-06-13

### Changelog

```
golang-hello-world (0.0~git20190613.ec269b1-1ppa6) bionic; urgency=medium

* Update release version to latest LTS

-- lyz@princessleia.com (Elizabeth K. Joseph) Thu, 13 Jun 2019 14:05:50 -0700
```

### Builds

- ✓ amd64
- ✓ i386
- ✓ s390x

### Built packages

**golang-hello-world** Simple "Hello World" program in Go

### Package files

- [golang-hello-world\\_0.0~git20190613.ec269b1-1ppa6.debian.tar.xz](#) (1.6 KiB)
- [golang-hello-world\\_0.0~git20190613.ec269b1-1ppa6.dsc](#) (2.0 KiB)
- [golang-hello-world\\_0.0~git20190613.ec269b1-1ppa6\\_amd64.deb](#) (366.2 KiB)
- [golang-hello-world\\_0.0~git20190613.ec269b1-1ppa6\\_i386.deb](#) (353.1 KiB)
- [golang-hello-world\\_0.0~git20190613.ec269b1-1ppa6\\_s390x.deb](#) (345.1 KiB)
- [golang-hello-world\\_0.0~git20190613.ec269b1.orig.tar.xz](#) (4.0 KiB)

---

golang-hello-world - 0.0~git20190613.ec269b1-1ppa3 (changes file) 2019-06-13 Published Xenial Games ✓

1 → 2 of 2 results First • Previous • Next ▶ • Last

Ubuntu Personal Package Archives (PPAs)  
on Launchpad.net

Documentation:  
<https://help.launchpad.net/Packaging/PPA>

openSUSE build service  
at [build.opensuse.org](http://build.opensuse.org)

The screenshot shows the openSUSE Build Service web interface for the 'snappy' package. The navigation bar includes links for Downloads, Support, Community, and Development. The breadcrumb trail indicates the path: openSUSE Build Service > Projects > home:markkp:branches:openSUSE:Factory:zSystems > snappy. The main content area features a title 'A fast compressor/decompressor library' and a detailed description of the library's performance characteristics. Below the description is a warning: 'Do NOT submit it to factory without asking or the package will be yours to maintain.' A 'Source Files' section displays a table of files with columns for filename, size, and change date. The 'Latest Revision' section shows a commit by Mark Post (markkp) from 6 days ago. On the right, a 'Build Results' sidebar lists various architectures and their build statuses.

Downloads Support Community Development

openSUSE Build Service > Projects > home:markkp:branches:openSUSE:Factory:zSystems > snappy

Overview Repositories Revisions Requests Users Advanced

### A fast compressor/decompressor library

Links to [devel:libraries:c\\_c++ /](#)  
[Download package](#)

Snappy is a compression/decompression library. It does not aim for maximum compression, or compatibility with any other compression library; instead, it aims for very high speeds and reasonable compression. For instance, compared to the fastest mode of zlib, Snappy is an order of magnitude faster for most inputs, but the resulting compressed files are anywhere from 20% to 100% bigger. On a single core of a Core i7 processor in 64-bit mode, Snappy compresses at about 250 MB/sec or more and decompresses at about 500 MB/sec or more.

Do NOT submit it to factory without asking or the package will be yours to maintain.

#### Source Files (show unmerged sources)

Show 25 entries Search:

Filename	Size	Changed	Actions
1.1.7.tar.gz	1.04 MB	2018-02-08	
baselibs.conf	11 Bytes	2015-07-29	
snappy-pcfile.patch	2.43 KB	2018-02-08	
snappy.changes	4.53 KB	2018-02-08	
snappy.spec	3.07 KB	2018-02-08	

Showing 1 to 5 of 5 entries Previous 1 Next

#### Latest Revision

Mark Post (markkp) committed 6 days ago (revision 1)  
[Browse Source](#)

#### Build Results RpmLint Results

### snappy

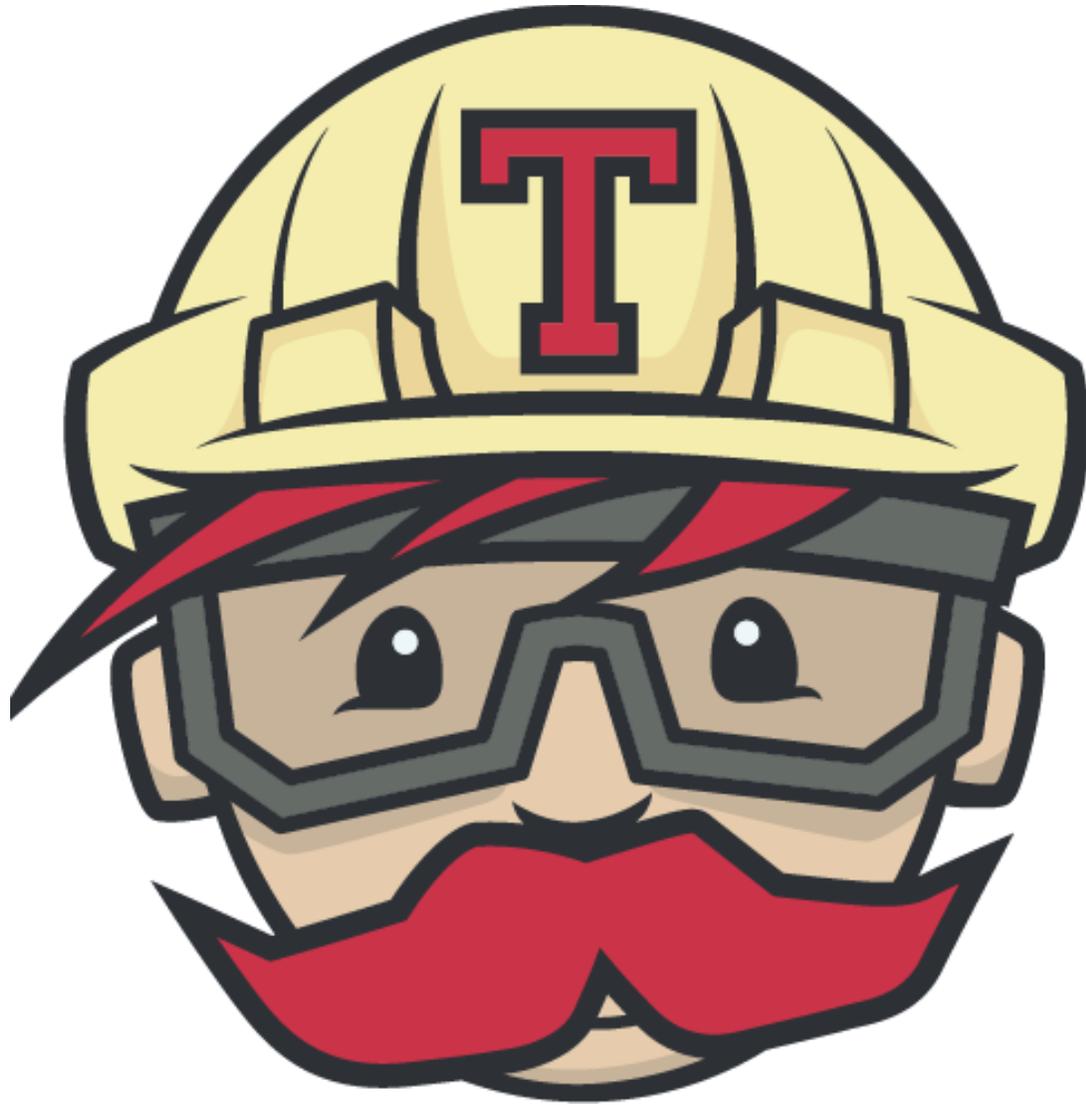
- SLE\_12\_SP2
- SLE\_12\_SP3
- SLE\_12\_SP4
- SLE\_15
- openSUSE\_Factory
- openSUSE\_Factory\_ARM
- openSUSE\_Factory\_PowerPC
- openSUSE\_Factory\_zSystems
- openSUSE\_Leap\_15.0
- openSUSE\_Leap\_15.1
- openSUSE\_Leap\_42.3



---

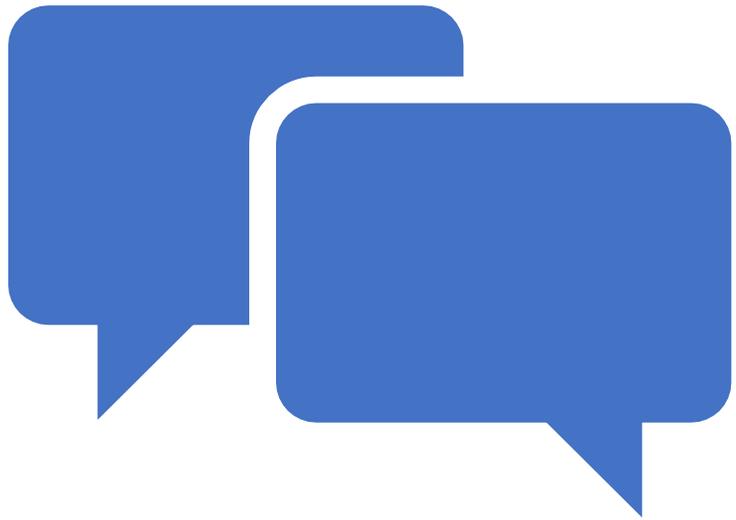
Jenkins instance for s390x maintained by the **Oregon State University Open Source Lab** (OSU OSL)

<https://osuosl.org/services/ibm-z/>



TravisCI build service for s390x (Beta trial for open source projects)

Documentation: <https://docs.travis-ci.com/user/multi-cpu-architectures/>



So, any programming language?

# Programming Language Tips

- Source code across architectures will generally be identical, but it needs to be compiled (C, C++), or otherwise interpreted (Python, Node.js) for this architecture.
- That means you need a compiler or an interpreter built for the mainframe.
- The mainframe architecture is big-endian, but most of the supported architectures today are little-endian.
- Your code will probably build and run, give it a try!
- If not, note that the higher level a language is, the more luck you'll have. Lower level languages like C do more hardware-specific operations, higher level languages like Node.js have much of that abstracted away.