OpenInfra Days
Beijing 2018

OpenStack on ARM

李枫

hkli2013@126.com
Jun 22，2018

Agenda

I. Growing Ecosystem of ARM
- Overview
- MicroServer
- Trend

II. OpenStack on ARM
- Current Status
- Hardware Perspective
- ARM Development Boards
- My Practice

III. Accelerating Python
- Overview
- GraalVM
- GraalPython
- My Practice
- Future

IV. Wrap-up

I. Growing Ecosystem of ARM

1) Overview

ARM服务器军团全军溃败，还有人能搅动这潭水吗？

钛媒体 05-23 09:59

Anand Chandrasekher
@achandrasekher
关注

Last day at Qualcomm. In 6 years, we accomplished some great things - proved ARM could deliver better performance/watt than Intel. I leave with some great memories.
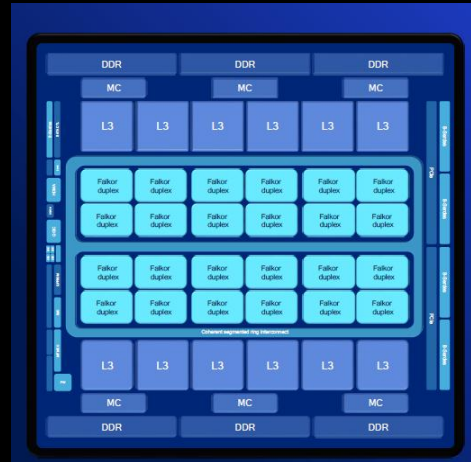
下午4:28 · 2018年5月11日

2 转推 24 喜欢

高通服务器芯片负责人离职：ARM难挑战Intel x86霸权

Qualcomm Centriq™ 2400

World's first 10nm server processor

QUALCOMM

Qualcomm Centriq™ Processor

Centriq

Qualcomm Centriq 2400: Built for cloud

Qualcomm® Falkor™ CPU
- 5th-generation custom core design
- 2.2 GHz base frequency, 2.6 GHz peak*
- Arm v8-compliant
- AArch64 only

High core count
- Up to 48 single-thread CPU cores

Large cache
- 64 KB L1 instruction cache with 24 KB single-cycle L0 cache
- 512 KB L2 per duplex
- 60 MB unified L3 cache
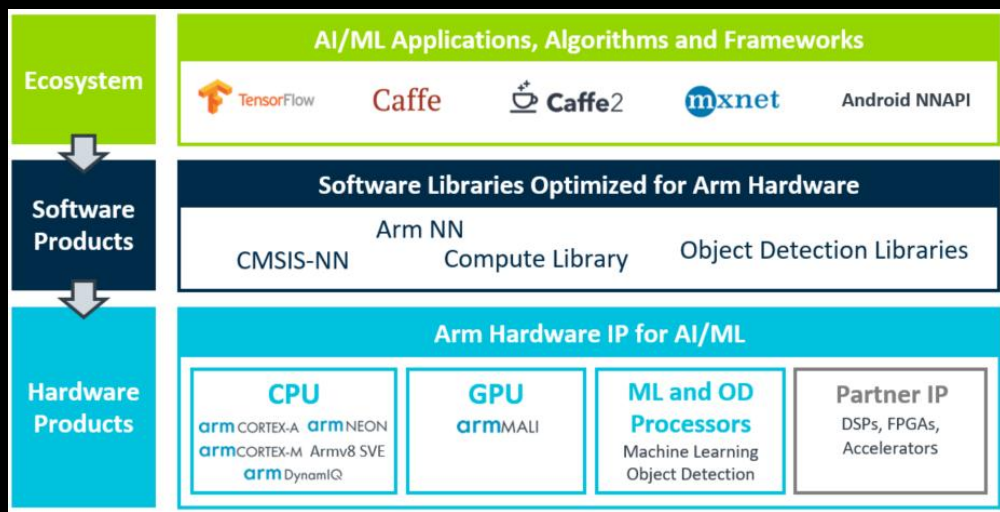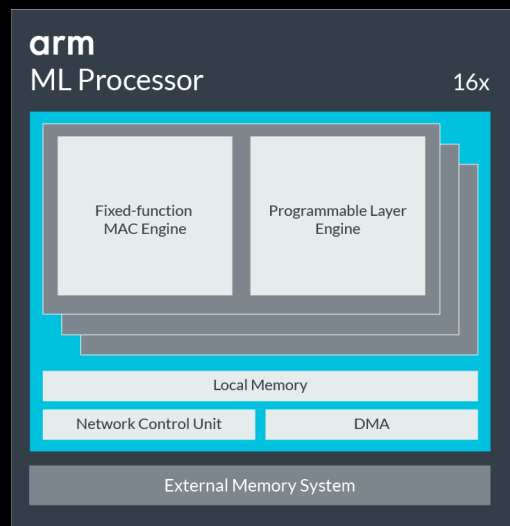
Bi-directional multi-ring
- Fully coherent
- >250GB/s aggregate bandwidth

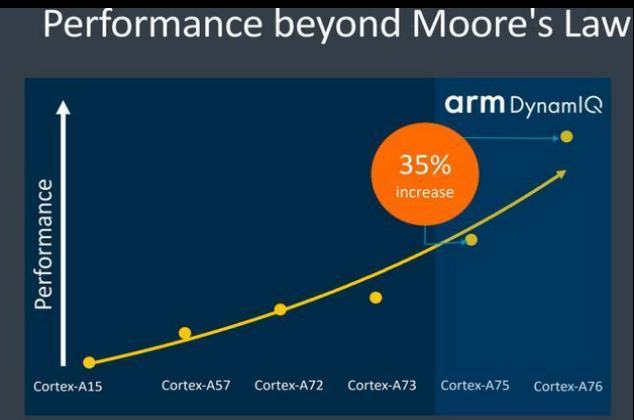https://www.qualcomm.com/media/documents/files/qualcomm-centriq-2400-media-deck.pdf

But…

AI

- **https://developer.arm.com/products/processors/machine-learning**
- **http://pages.arm.com/dynamiq-technology.html**

Cloud

- **Windows Server on ARM**
- **https://buildazure.com/2017/03/10/windows-server-running-on-arm-cpus-azure-is-next/**

packet.net
- **https://retout.co.uk/blog/2017/04/25/packet-net-arm64-servers**

Packet.net offer an ARMv8 server with 96 cores for $0.50/hour.

HPC

Scalable Vector Extension

Fujitsu High-end Supercomputers Development — FUJITSU

**K computer and PRIMEHPC FX10/FX100 in operation**

The CPU and interconnect of FX10/FX100 inherit the K computer architectural concept, featuring state-of-the-art technologies

System software "TCS" supports Fujitsu supercomputer with originally introduced technologies

Many applications are currently running and being developed for science and various industries

Post-K supercomputer

RIKEN and Fujitsu are working together to provide a successor to K computer with application R&D teams using co-design approach
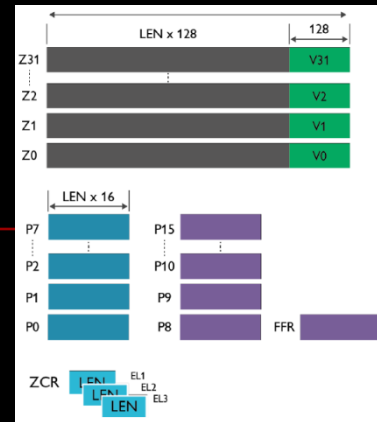
ARMv8-A SVE benefits

- Extending ARMv8-A with AArch64 extension which expands vector length up to 2048 bits

- Expands fine-grain data parallelism for HPC scientific workloads

- Better compiler target: increases vectorization opportunities; reduces software deployment effort

- Enabling open-source community and wider ARM ecosystem

Arm HPC ecosystem roadmap
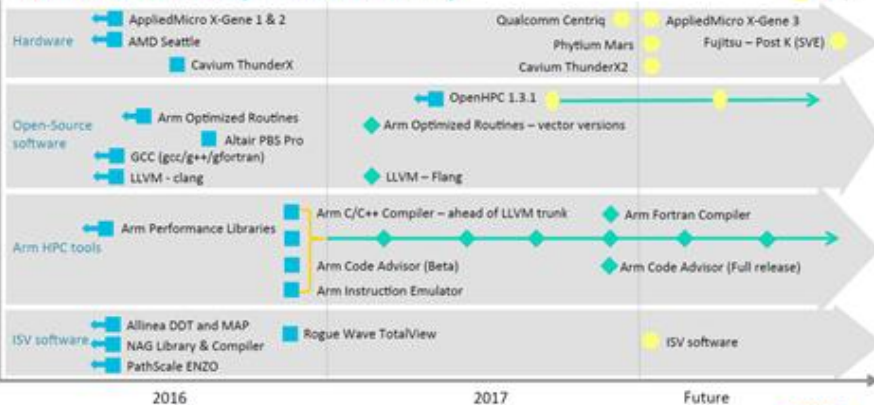
HPC benchmarks SVE vs NEON
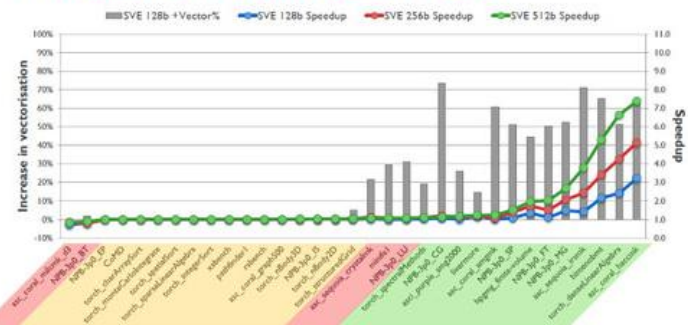
The Machine
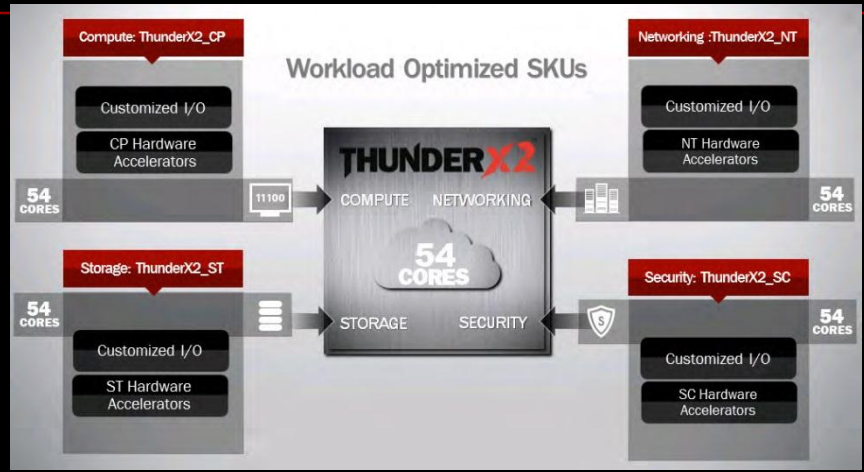
- https://www.theinquirer.net/inquirer/news/3010243/hpe-shows-off-arm-powered-the-machine-prototype-with-160tb-memory

Linaro

- **http://www.linaro.org**

96boards

- **http://www.96boards.org/**

| ROCK960 | DragonBoard™ 410c | Ultra96 |
| From $99.00 | From $75.00 | From $249.00 |

| HiKey 960 | Nitrogen | DragonBoard 820c |
| From $239.00 | From $27.95 | From $199.00 |
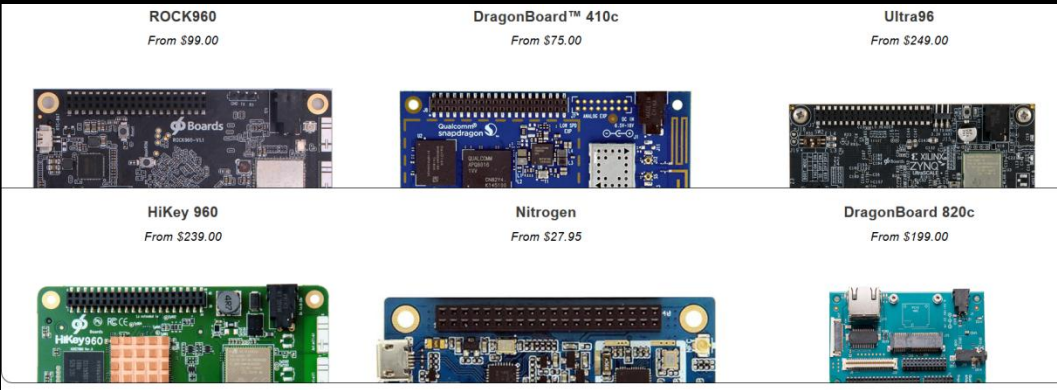
Consumer Edition (CE)

Enterprise Edition (EE)

IoT Edition (IE)

Mezzanine Products

Accessories

2) MicroServer
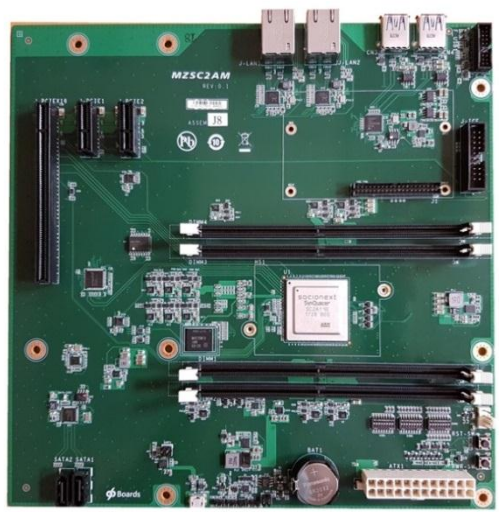
Synquacer SC2A11

- A53@1GHz x24, 5W

96Boards EE Card

- microATX format
- up to 64GB RAM
- SATA, GBE
- DeveloperBox

Processor	Cortex-A53 MPCore 24cores, 1GHz, L1 I/D=32KB/32KB, L2 =256KB, L3 =4MB
Memory I/F	DDR4-2133Mbps 64-bit + ECC
PCIe	PCI Express Gen2, Root/Endpoint select, 4 lanes (2 systems/ for SoC IF)
LAN	2ch 1Gbps with IPSec Network Offload Engine (wire-speed)
Flash I/F	HSSPI, eMMC
Serial I/F	UART, I2C, GPIO

3) Trend
Acquisitions

Desktop

■ ARM PC

Cortex-A76 / Mali-G76

- **https://developer.arm.com/products/processors/cortex-a/cortex-a76**
- **delivers laptop-class performance with mobile efficiency**
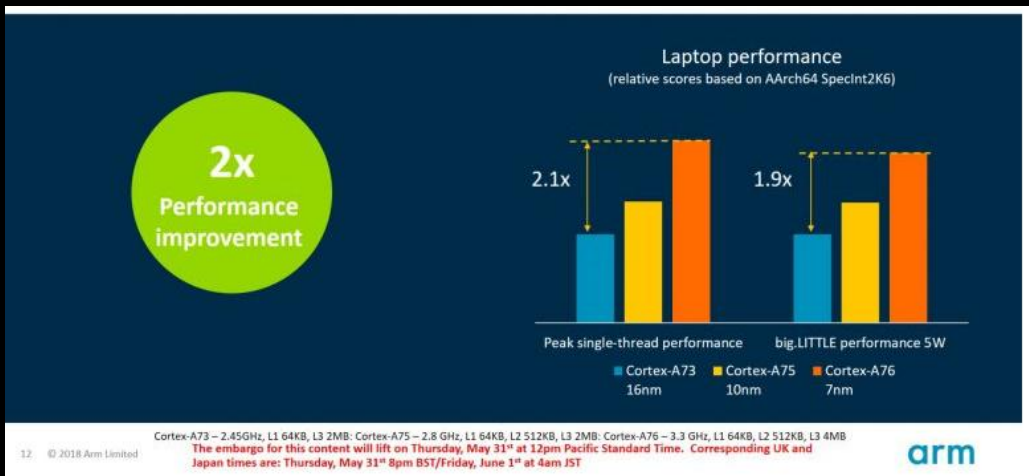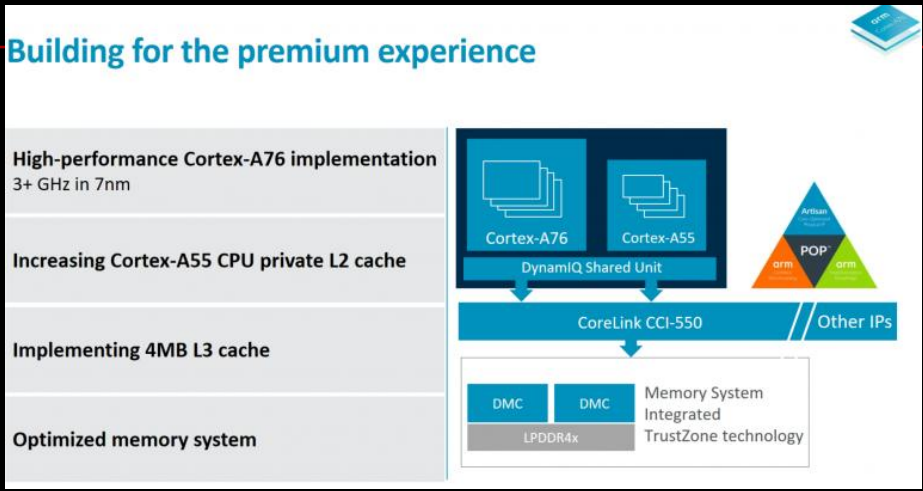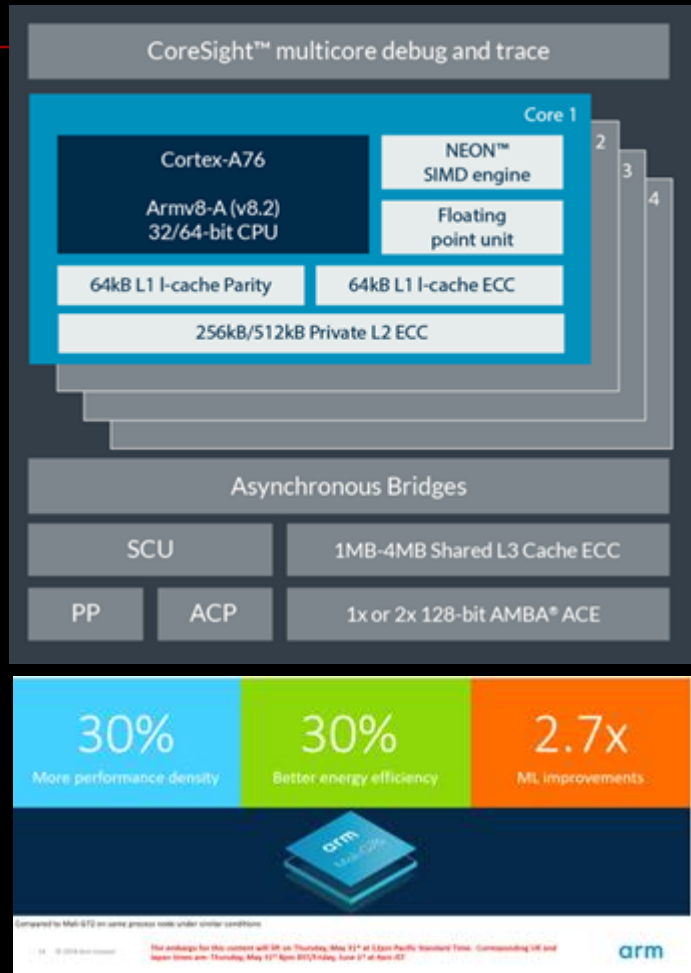
II. OpenStack on ARM

1) Current Status

- https://marcin.juszkiewicz.com.pl/2018/03/02/openstack-queens-release-done/
- https://wiki.linaro.org/OpenStack

   …

- Three ARMed OpenStack, OpenInfrastructure New York 2018
- The OpenStack on AArch64 journey, Linaro Connect HK 2018
- OpenStack on AArch64, LC3 Beijing 2017
- Build Cloud Infrastructure with ARMv8, OSSNA 2017
- ARM'ed OpenStack Farm, CloudOpen North America 2014

Enabling tech	Memcached	MongoDB	My SQL	NGINX	Hadoop	Apache Spark	Redis	Cassandra	OPNFV	Apache Tomcat
Cloud / mgmt	Kubernetes	Openstack	Mesos	Cloud Foundry						
Languages, runtimes	Java	Azul	Python	Open JDK	PHP	Node	Golang			
Tools	GCC / LLVM / Debuggers (JTAG, GDB) / Libraries (glibc, others)									
Virtualization	KVM	Xen	Docker							
OS	Canonical	Redhat	Suse	CentOS						
Firmware platform mgmt	HW Root of Trust / Trusted Execution Environment / Power Management / Secure Boot						American Megatrends			

https://www.qualcomm.com/media/documents/files/qualcomm-centriq-2400-media-deck.pdf

2) Hardware Perspective

- **https://en.wikipedia.org/wiki/ARM_architecture**
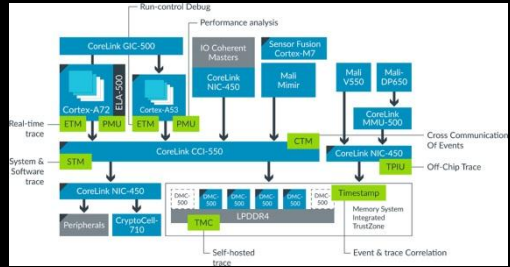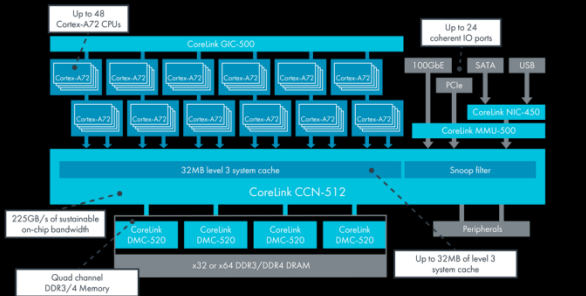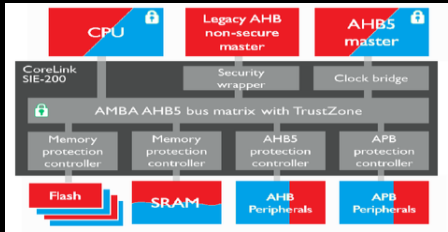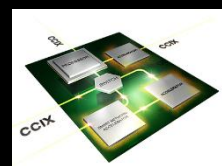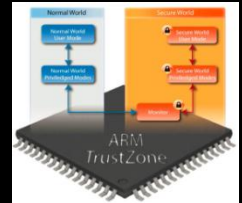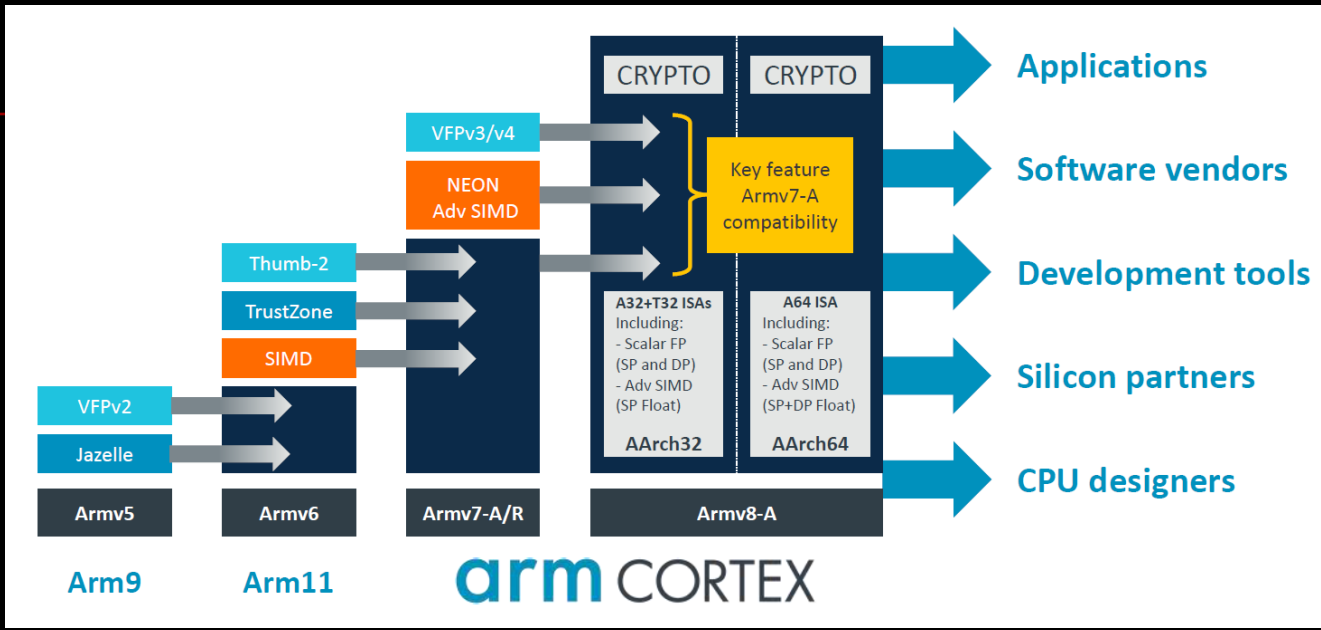
- **UEFI** on ARM
- **Please add Hardware Transactional Memory support in ARM v9 …**

Virtualization

- **https://en.wikipedia.org/wiki/Hardware-assisted_virtualization**
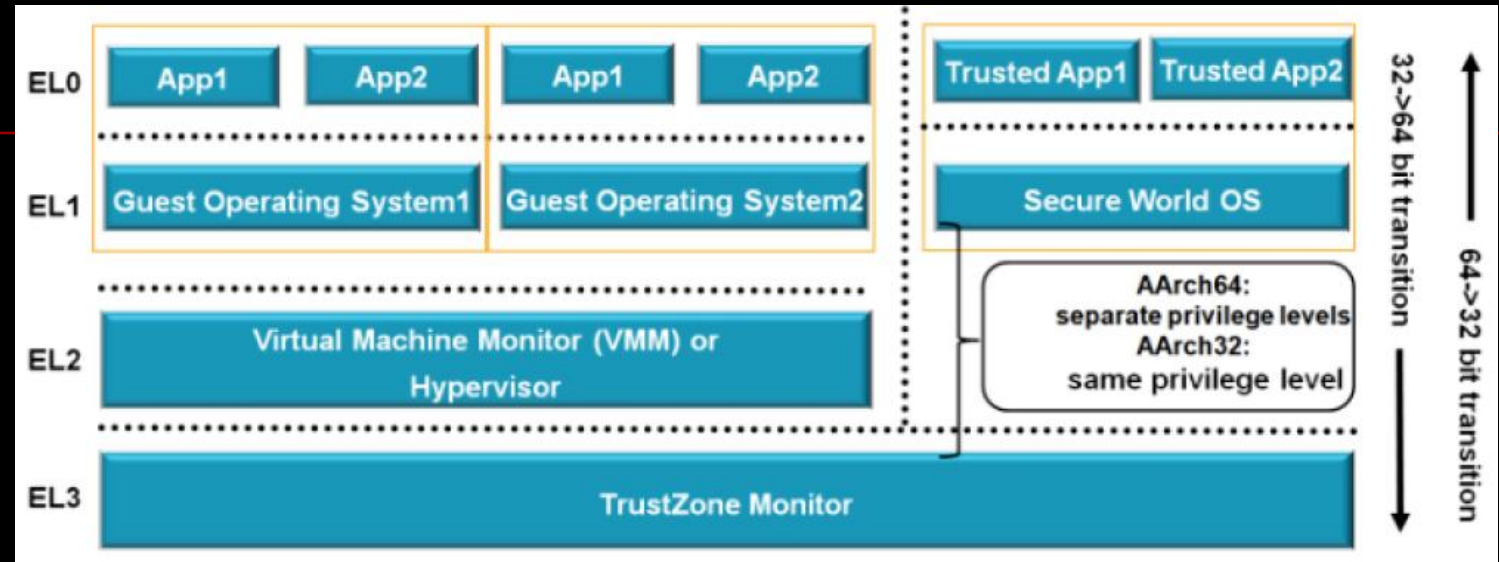
- **Virtualization Host Extensions**
- **Nested Virtualization Extensions**
- **System MMU**

Segments	Use cases	Hypervisors	Whitepaper year and related ARM CoreLink System IP
Mobile	BYOD	VMware MVP	2011 - SMMU Automotive
Automotive	ADAS/IVI ECU consolidation	Green Hills (INTEGRITY) VirtualLogix	
Server	Live migration Rapid deployment Sandboxing	Xen / KVM	2017 - SMMU & GIC

3) ARM Development Boards

Raspberry Pi

- **https://www.raspberrypi.org/**
- **https://en.wikipedia.org/wiki/Raspberry_Pi**

	Model 3 B	Model 3 B+
Release date	Feb, 2016	Mar, 2018
Arch	ARMv8-A	ARMv8-A
SoC	BCM2837	BCM2837B0
CPU	1.2 GHz 64-bit quad-core ARM Cortex-A53	1.4 GHz 64-bit quad-core ARM Cortex-A53
GPU	VideoCore IV	VideoCore IV
Memory (SDRAM)	1GB LPDDR2 RAM @900MHz (shared with GPU)	1GB LPDDR2 RAM @900MHz (shared with GPU)
Network	10/100 Mbit/s Ethernet, 802.11n wireless, Bluetooth 4.1	10/100/1000 Mbit/s Ethernet (real speed ~300 Mbit/s), 802.11ac dual band 2.4/5 GHz wireless, Bluetooth 4.2 LS BLE

- **Official release (Raspbian with Linux Kernel 4.14 currently) still does not support AArch64**

ROC-RK3328-CC

- **http://www.t-firefly.com/product/rocrk3328cc.html**
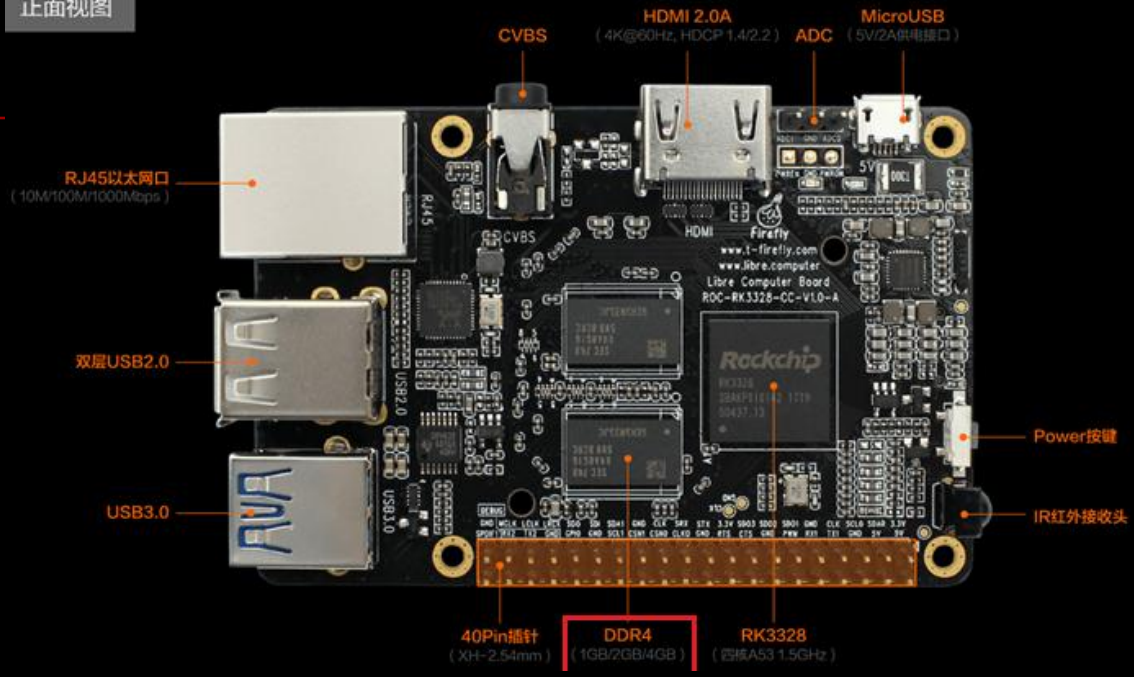
- **my test board has 4GB DDR4 @2133MHz**
- **Ubuntu 16.04/Debian 9/Android 7.1.1 for AARCH64**

My Dev Env

- **Pls refer to my presentation "eBPF in Action" at LC3 Beijing (on Jun 25, 2018)**

Can we run OpenStack on Raspberry Pi?

■ **Seems to be Mission Impossible**

■ **My testbed: Fedora Minimal 28 AARCH64 on RPi 3B/3B+**

```
[stack@promote boot]$ uname -a
Linux promote.cache-dns.local 4.16.14-300.fc28.aarch64 #1 SMP Tue Jun 5 16:00:29 UTC 2018 aarch64 aarch64 aarch64 GNU
/Linux
[stack@promote boot]$ cat /boot/config-4.16.14-300.fc28.aarch64 |grep -i kvm
CONFIG_HAVE_KVM_IRQCHIP=y
CONFIG_HAVE_KVM_IRQFD=y
CONFIG_HAVE_KVM_IRQ_ROUTING=y
CONFIG_HAVE_KVM_EVENTFD=y
CONFIG_KVM_MMIO=y
CONFIG_HAVE_KVM_MSI=y
CONFIG_HAVE_KVM_CPU_RELAX_INTERCEPT=y
CONFIG_KVM_VFIO=y
CONFIG_HAVE_KVM_ARCH_TLB_FLUSH_ALL=y
CONFIG_KVM_GENERIC_DIRTYLOG_READ_PROTECT=y
CONFIG_HAVE_KVM_IRQ_BYPASS=y
CONFIG_KVM=y
CONFIG_KVM_ARM_HOST=y
CONFIG_KVM_ARM_PMU=y
[stack@promote boot]$
```

```
[stack@promote boot]$ dmesg | grep -i kvm
[    1.611383] kvm [1]: 8-bit VMID
[    1.613639] kvm [1]: Hyp mode initialized successfully
[stack@promote boot]$
```

4) My Practice
DevStack

- **https://github.com/openstack-dev/devstack**
- **tested on Fedora Minimal 28 AARCH64**
- **Django issue**

```
2018-06-12 20:50:30 Package conntrack-tools-1.4.4-7.fc28.aarch64 is already installed, skipping.
2018-06-12 20:50:30 Package keepalived-1.4.3-1.fc28.aarch64 is already installed, skipping.
2018-06-12 20:50:30 No match for argument: Django
2018-06-12 20:50:30 Package python2-pyxattr-0.5.3-16.fc28.aarch64 is already installed, skipping.
2018-06-12 20:50:30 Package dstat-0.7.3-4.fc28.noarch is already installed, skipping.
2018-06-12 20:50:30 Package python2-psutil-5.4.3-4.fc28.aarch64 is already installed, skipping.
2018-06-12 20:50:30 Error: Unable to find a match
2018-06-12 20:50:30 YUM_FAILED 1
2018-06-12 20:50:30 ++ functions-common:yum_install:1352        :    result=1
2018-06-12 20:50:30 ++ functions-common:yum_install:1354        :    time_stop yum_install
2018-06-12 20:50:30 ++ functions-common:time_stop:2254          :    local name
```

- **uwsgi issue**
  https://github.com/unbit/uwsgi/issues/1763
- **my patch:**

```
diff --git a/files/rpms/horizon b/files/rpms/horizon
index fa5601a9..155f4de6 100644
--- a/files/rpms/horizon
+++ b/files/rpms/horizon
@@ -1,4 +1,3 @@
-Django
 httpd # NOPRIME
 mod_wsgi  # NOPRIME
 pyxattr
diff --git a/stack.sh b/stack.sh
index 56e00bfb..5dec71c8 100755
--- a/stack.sh
+++ b/stack.sh
@@ -829,7 +829,7 @@ echo_summary "Installing OpenStack project source"
 install_libs

 # Install uwsgi
-install_apache_uwsgi
+#install_apache_uwsgi

 # Install client libraries
 install_keystoneauth
diff --git a/stackrc b/stackrc
index 6c4d7d68..27496997 100644
--- a/stackrc
+++ b/stackrc
@@ -230,7 +230,7 @@ GIT_TIMEOUT=${GIT_TIMEOUT:-0}
 # to allow for 2 modes, which is "uwsgi" which runs with an apache
 # proxy uwsgi in front of it, or "mod_wsgi", which runs in
 # apache. mod_wsgi is deprecated, don't use it.
-WSGI_MODE=${WSGI_MODE:-"uwsgi"}
+#WSGI_MODE=${WSGI_MODE:-"uwsgi"}
```

result:

```
atd.service                        loaded active running   Job spooling tools
auditd.service                     loaded active running   Security Auditing Service
chronyd.service                    loaded active running   NTP client/server
crond.service                      loaded active running   Command Scheduler
dbus.service                       loaded active running   D-Bus System Message Bus
dbxtool.service                    loaded active exited    Secure Boot DBX (blacklis>
devstack@dstat.service             loaded active running   Devstack devstack@dstat.s>
● devstack@etcd.service            loaded failed failed    Devstack devstack@etcd.se>
devstack@g-api.service             loaded active running   Devstack devstack@g-api.s>
devstack@g-reg.service             loaded active running   Devstack devstack@g-reg.s>
dracut-shutdown.service            loaded active exited    Restore /run/initramfs on>
epmd@0.0.0.0.service                loaded active running   Erlang Port Mapper Daemon
fedora-readonly.service            loaded active exited    Configure read-only root >
getty@tty1.service                 loaded active running   Getty on tty1
gssproxy.service                   loaded active running   GSSAPI Proxy Daemon
iptables.service                   loaded active exited    IPv4 firewall with iptabl>
iscsi-shutdown.service             loaded active running   Logout off all iSCSI sess>
kmod-static-nodes.service          loaded active exited    Create list of required s>
libvirtd.service                   loaded active running   Virtualization daemon
lvm2-lvmetad.service               loaded active running   LVM2 metadata daemon
lvm2-monitor.service               loaded active exited    Monitoring of LVM2 mirror>
mariadb.service                    loaded active running   MariaDB 10.2 database ser>
netcf-transaction.service          loaded active exited    Rollback uncommitted netc>
NetworkManager-wait-online.service loaded active exited    Network Manager Wait Onli>
NetworkManager.service             loaded active running   Network Manager
openvswitch.service                loaded active exited    Open vSwitch
ovs-vswitchd.service               loaded active running   Open vSwitch Forwarding U>
ovsdb-server.service               loaded active running   Open vSwitch Database Unit
polkit.service                     loaded active running   Authorization Manager
rabbitmq-server.service            loaded active running   RabbitMQ broker
rngd.service                       loaded active running   Hardware RNG Entropy Gath>
rpc-statd-notify.service           loaded active exited    Notify NFS peers of a res>
```

```
top - 14:29:26 up 42 min,  3 users,  load average: 0.61, 0.65, 0.77
Tasks: 137 total,   1 running,  88 sleeping,   0 stopped,   0 zombie
%Cpu(s):  4.2 us,  1.7 sy,  0.0 ni, 92.7 id,  0.2 wa,  1.1 hi,  0.1 si,  0.0 st
KiB Mem :   983876 total,   154312 free,   634836 used,   194728 buff/cache
KiB Swap:  6815736 total,  6688504 free,   127232 used.   322204 avail Mem

 PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 636 openvsw+  10 -10   89444  87624  11496 S   0.7  8.9   0:16.01 ovs-vswitchd
 455 stack     20   0  181528  84704   7208 S   2.3  8.6   1:32.89 glance-registry
1387 stack     20   0  181528  79740   2320 S   0.0  8.1   0:00.03 glance-registry
1386 stack     20   0  181528  79540   2320 S   0.0  8.1   0:00.04 glance-registry
 960 mysql     20   0  969704  73520   5664 S   0.3  7.5   0:06.94 mysqld
 452 stack     20   0  198416  69644   8000 S   2.6  7.1   1:37.47 glance-api
1389 stack     20   0  198416  63788   2192 S   0.0  6.5   0:00.04 glance-api
1388 stack     20   0  198416  63504   2192 S   0.0  6.5   0:00.02 glance-api
 687 rabbitmq  20   0 1249956  43640   5660 S   0.0  4.4   0:31.67 beam.smp
 320 root      20   0   76092  16896  15804 S   0.3  1.7   0:10.10 systemd-journal
 692 root      20   0  430860  16584   7700 S   0.0  1.7   0:02.57 libvirtd
 575 polkitd   20   0 1254156  15188   9768 S   0.0  1.5   0:00.72 polkitd
 477 root      20   0   96952  10612   8856 S   0.0  1.1   0:01.53 NetworkManager
   1 root      20   0   52912   8520   5332 S   0.0  0.9   0:11.20 systemd
 479 stack     20   0   24184   8020   3468 S  15.7  0.8   9:56.92 dstat
```

Kata Containers

- **https://katacontainers.io**
- **https://github.com/kata-containers/**
- **The speed of containers, the security of VMs**
-

Intel® Clear Containers

h_ HYPER.SH *run*V

→ katacontainers

May 2015   Dec 2017

source: https://katacontainers.io/media/uploads/katacontainers/uploads/katacontainers/kata-
containers-on-boarding-deck-for-website01022018.pdf
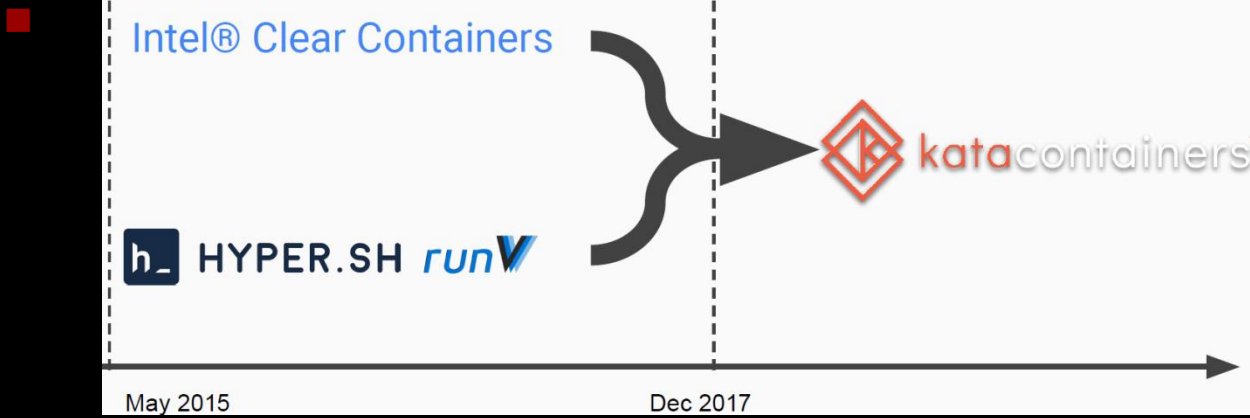
https://github.com/kata-containers/runtime

Kata Containers currently works on systems supporting the following technologies:

- Intel's VT-x technology.
- ARM's Hyp mode (virtualization extension).
- IBM's Power Systems.

```
arch/
├── amd64-options.mk
├── arm64-options.mk
└── ppc64le-options.mk
```

build Kata on Raspberry Pi 3B+ with Fedora Minimal 28 AARCH64 & go 1.10.3

```
2018-06-19 00:36:39      CLEAN    clean
2018-06-19 00:36:42      GENERATE cli/config-generated.go
2018-06-19 00:36:42      CONFIG   data/kata-collect-data.sh
2018-06-19 00:36:42 kata-runtime - version 1.0.0 (commit 42821b7c0a572bb6b1497e6a2b9a3ad6301c09bb)
2018-06-19 00:36:42
2018-06-19 00:36:42 • architecture:
2018-06-19 00:36:42      Host: aarch64
2018-06-19 00:36:42      golang: arm64
2018-06-19 00:36:42      Build: arm64
2018-06-19 00:36:42
2018-06-19 00:36:42 • golang:
2018-06-19 00:36:42      go version go1.10.3 linux/arm64
2018-06-19 00:36:42
2018-06-19 00:36:42 • Summary:
2018-06-19 00:36:42
2018-06-19 00:36:42      binary install path (DESTTARGET)     : /usr/local/bin/kata-runtime
2018-06-19 00:36:42      config install path (DESTCONFIG)     : /usr/share/defaults/kata-containers/configuration.toml
2018-06-19 00:36:42      alternate config path (DESTSYSCONFIG) : /etc/kata-containers/configuration.toml
2018-06-19 00:36:42      hypervisor path (QEMUPATH)           : /usr/bin/qemu-system-aarch64
2018-06-19 00:36:42      assets path (PKGDATADIR)              : /usr/share/kata-containers
2018-06-19 00:36:42      proxy+shim path (PKGLIBEXECDIR)       : /usr/libexec/kata-containers
2018-06-19 00:36:42
2018-06-19 00:36:42      BUILD    /opt/MyWorkSpace/MyProjs/Virtual/VM-Container/Kata/src/github.com/kata-containers/runtime
/kata-runtime
2018-06-19 00:38:11      CONFIG   cli/config/configuration.toml
```

check

```
[myrpi4@promote runtime]$ sudo /usr/local/bin/kata-runtime kata-check
INFO[0000] Unable to know if the system is running inside a VM
INFO[0000] kernel property found                        description="Host kernel accelerator for virtio" name=vhost pid=2
5195 source=runtime type=module
INFO[0000] kernel property found                        description="Host kernel accelerator for virtio network" name=vho
st_net pid=25195 source=runtime type=module
INFO[0000] kernel property found                        description="Kernel-based Virtual Machine" name=kvm pid=25195 sou
rce=runtime type=module
INFO[0000] System is capable of running Kata Containers name=kata-runtime pid=25195 source=runtime
INFO[0000] device available                             check-type=full device=/dev/kvm name=kata-runtime pid=25195 sourc
e=runtime
INFO[0000] feature available                            check-type=full feature=create-vm name=kata-runtime pid=25195 sou
rce=runtime
INFO[0000] System can currently create Kata Containers  name=kata-runtime pid=25195 source=runtime
```

III. Acceleration Python

1) Overview

Why Python

■ **https://www.tiobe.com/tiobe-index/**

May 2018	May 2017	Change	Programming Language	Ratings	Change
1	1		Java	16.380%	+1.74%
2	2		C	14.000%	+7.00%
3	3		C++	7.668%	+2.92%
4	4		Python	5.192%	+1.64%
5	5		C#	4.402%	+0.95%

■ **http://pypl.github.io/PYPL.html**

■ **https://spectrum.ieee.org/computing/software/the-2017-top-programming-languages**

Language Rank	Types	Spectrum Ranking
1. Python		100.0
2. C		99.7
3. Java		99.5
4. C++		97.1
5. C#		87.7
6. R		87.7
7. JavaScript		85.6
8. PHP		81.2
9. Go		75.1
10. Swift		73.7

- **https://www.kdnuggets.com/2018/05/poll-tools-analytics-data-science-machine-learning-results.html**

KDnuggets Analytics, Data Science, Machine Learning Software Poll, 2016-2018

Tool	2018 %share
Python	65.6%
RapidMiner	52.7%
R	48.5%
SQL	39.6%
Excel	39.1%
Anaconda	33.4%
Tensorflow	29.9%
Tableau	26.4%
scikit-learn	24.4%
Keras	22.2%
Apache Spark	21.5%

Legend: 2018 %share, 2017 %share, 2016 %share

- **Other Python projects**
  Build: Meson, SCons…   DevOps: Ansible, SaltStack…
  Web: Django, web2py, Flask, Tornado, TurboGears…
  AI: PyTorch, Theano… Big Data: PyData, PySpark…
  Science: Scipy, Sage…
  Cloud/DataCenter: OpenStack
  Security: a swiss knife for hackers…

  …

Why Python is Slow

- **dynamically typed**
- **no JIT support in the official CPython**
- **GIL (Global Interpreter Lock)**

Python 3 programs versus Java

vs C vs C++ vs Go **vs Java**

by benchmark task performance

pidigits

source	secs	mem	gz	cpu	cpu load
Python 3	3.43	12,716	386	3.43	100% 1% 1% 0%
Java	3.13	36,984	938	3.36	4% 4% 99% 3%

regex-redux

source	secs	mem	gz	cpu	cpu load
Python 3	15.22	447,324	512	27.44	25% 33% 32% 91%
Java	10.51	573,972	929	31.30	70% 73% 70% 86%

reverse-complement

source	secs	mem	gz	cpu	cpu load
Python 3	18.79	1,008,868	814	19.73	9% 69% 35% 30%
Java	3.15	680,424	2183	7.07	52% 70% 43% 63%

k-nucleotide

source	secs	mem	gz	cpu	cpu load
Python 3	77.65	182,700	1967	302.86	97% 99% 97% 98%
Java	8.75	385,056	1812	27.09	85% 72% 70% 85%

binary-trees

source	secs	mem	gz	cpu	cpu load
Python 3	93.55	280,624	589	337.74	92% 89% 87% 93%
Java	8.39	933,808	835	28.28	82% 86% 84% 88%

fasta

source	secs	mem	gz	cpu	cpu load
Python 3	59.47	15,996	1947	138.97	55% 55% 63% 66%
Java	2.27	43,628	2473	5.93	51% 75% 57% 81%

fannkuch-redux

source	secs	mem	gz	cpu	cpu load
Python 3	565.97	15,528	950	2,172.63	95% 94% 95% 100%
Java	18.27	31,820	1282	72.06	99% 99% 98% 98%

mandelbrot

source	secs	mem	gz	cpu	cpu load
Python 3	225.24	15,736	688	899.25	100% 100% 100% 100%
Java	6.10	76,520	796	23.59	97% 98% 98% 96%

n-body

source	secs	mem	gz	cpu	cpu load
Python 3	838.39	10,324	1196	838.20	95% 1% 5% 0%
Java	22.17	33,040	1489	22.27	100% 1% 0% 1%

spectral-norm

source	secs	mem	gz	cpu	cpu load
Python 3	180.97	15,876	443	720.51	100% 100% 100% 100%
Java	4.38	35,388	950	16.80	96% 96% 95% 97%

Python 3	Python 3.6.3
Java	java 10 2018-03-20 Java(TM) SE Runtime Environment 18.3 (build 10+46) Java HotSpot(TM) 64-Bit Server VM 18.3 (build 10+46, mixed mode)

https://benchmarksgame-team.pages.debian.net/benchmarksgame/faster/python.html

Runtimes

- **LLVM-based (VMKit, MCJIT,)**

up to 10x speedups Google

PySton

Dropbox

- pypy

RPython
Meta-tracing
…

How has PyPy performance evolved over time?

Source: http://speed.pypy.org/

2) GraalVM

- **https://www.graalvm.org/**
- **http://www.oracle.com/technetwork/oracle-labs/ program-languages/overview/index.html**
- **https://blogs.oracle.com/developers/announcing-graalvm**

- **High-Performance Polyglot VM**
- **A meta-runtime for Language-Level Virtualization**
- **Currently base an Oracle Labs JDK 8 with JVMCI support**
- **http://openjdk.java.net/jeps/243(JVMCI): experimental in JDK 9**

Arch

- ## **A hybrid of static & dynamic runtimes**

Substrate VM

Source: https://ics.psu.edu/wp-content/uploads/2017/02/GraalVM-PSU.pptx

Source: https://www.slideshare.net/jyukutyo/jvmgraalopenj9
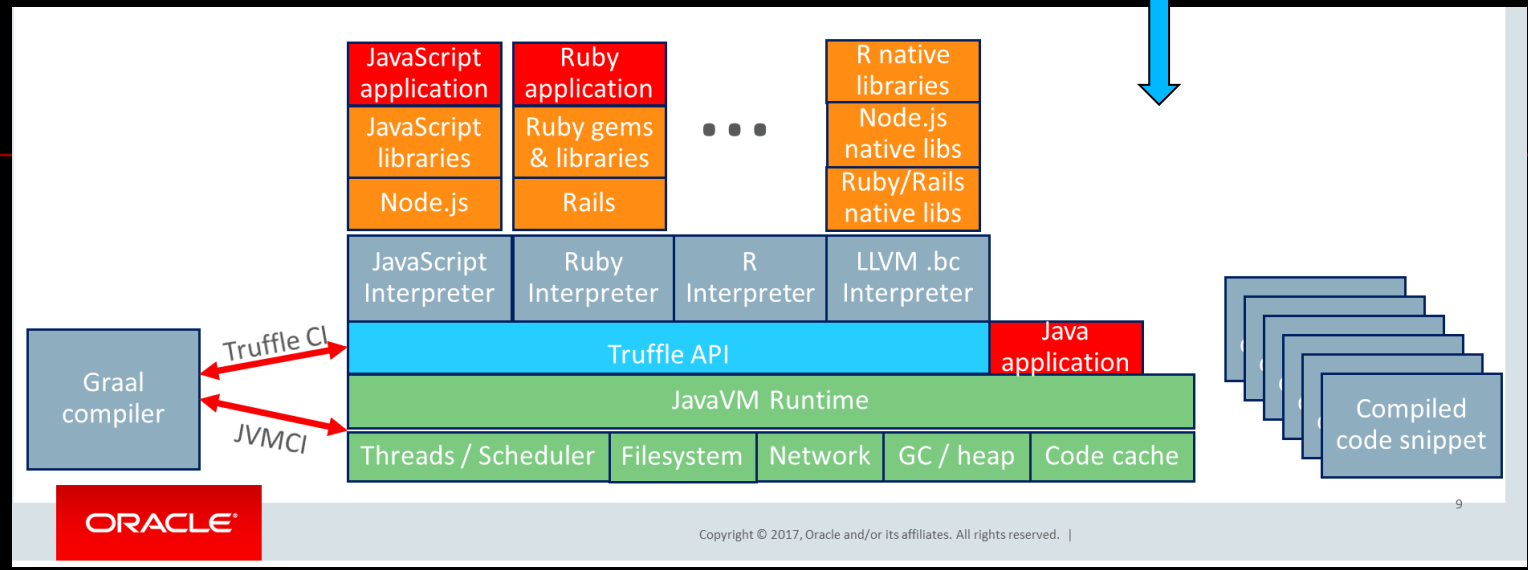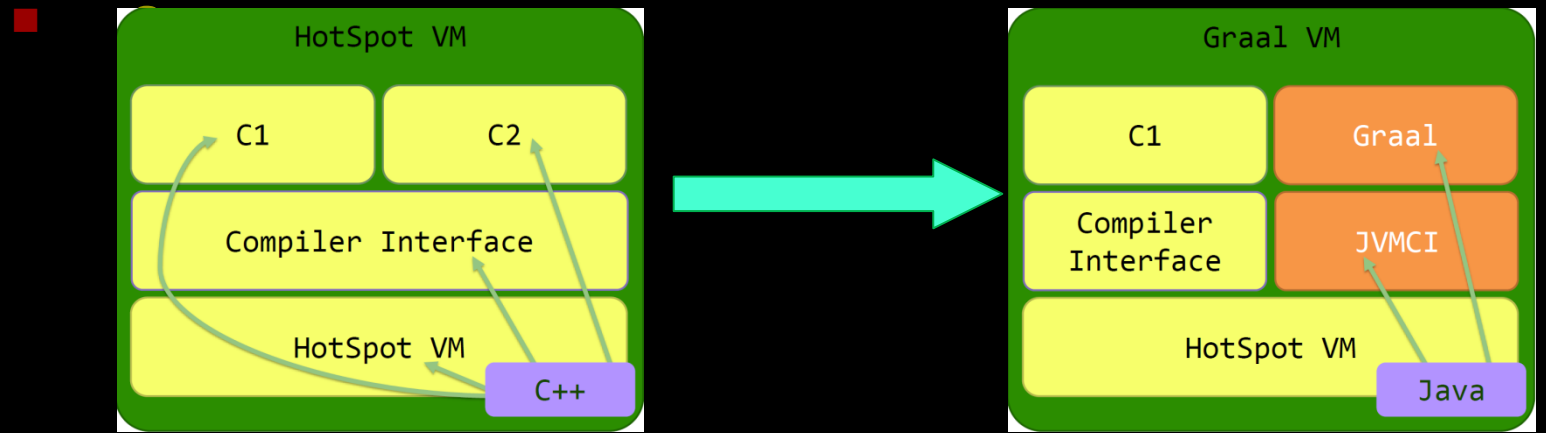
Performance
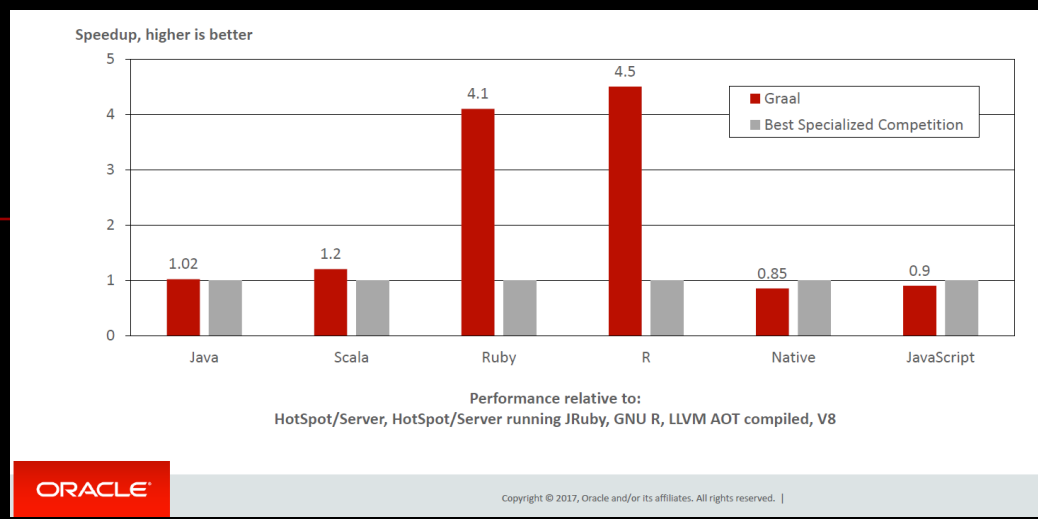
Source: http://lafo.ssw.uni-linz.ac.at/papers/2017_PLDI_GraalTutorial.pdf

Source: http://dbpl2017.org/slides/DBPL-2017-s2.pdf
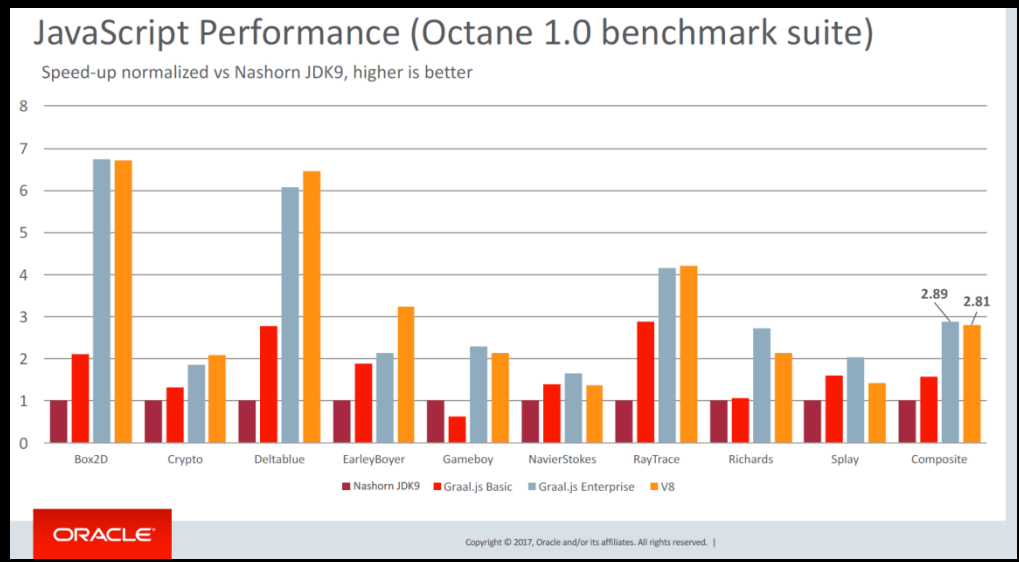
■ **but for GraalVM 1.0.0 RC1**

	GRAALVM	ORACLE JDK 8	ORACLE JDK 9
AVERAGE OPS/S	6.795 ±(99.9%) 0.016	6.727 ±(99.9%) 0.017	7,136 ±(99.9%) 0,026
MIN	6.477	6.466	6,464
MAX	6.967	6.899	7,443
STD DEV	0.068	0.070	0,111
CI (99.9%) (ASSUMES NORMAL DISTRIBUTION)	[6.778, 6.811]	[6.710, 6.743]	[7,110, 7,162]

Source: https://blog.frankel.ch/first-impressions-graalvm

■ **Real World Apps:**
Using GraalVM to run Native Java in AWS Lambda with Golang
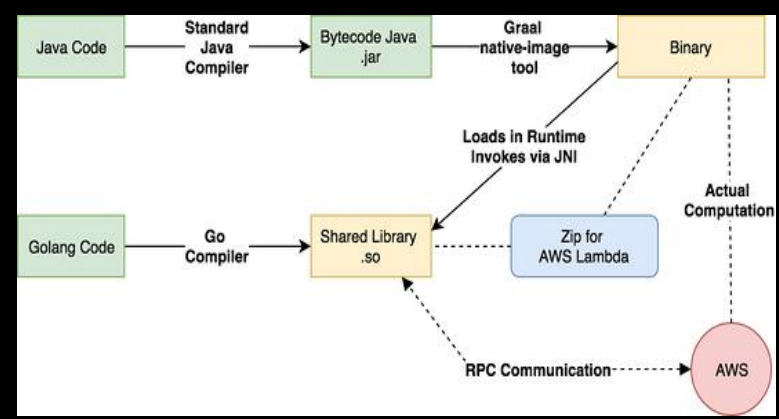
Memory (MB)	Avg Duration (ms)	Max Duration (ms) java	Avg Graal + Go (ms)	Max Graal + Go (ms)
256	489	3179	992	1011
512	235	1426	486	529
1024	123	652	243	266
1536	85	443	162	173
2048	78	371	143	153

Source: https://engineering.opsgenie.com/run-native-java-using-graalvm -in-aws-lambda-with-golang-ba86e27930bf

3) GraalPython

Graal/Truffle-based implementation of Python

GraalVM provides an early-stage experimental implementation of Python. A primary goal is to support SciPy and its constituent libraries. This Python implementation currently aims to be compatible with Python 3.7, but it is a long way from there, and it is very likely that any Python program that requires any imports at all will hit something unsupported. At this point, the Python implementation is made available for experimentation and curious end-users.

- **https://github.com/graalvm/graalpython**
- **https://www.graalvm.org/docs/reference-manual/languages/python/**

	Java 10.0.1	CPython 3.6.5	GraalPython ee-1.0.0-rc2
n-body	9.676s	11m56.642s	15m57.543s

Test on Dell XPS 15z: i5-2410M@2.3Ghz, 6G RAM, Fedora 28 for X64 with Kernel 4.16.14

```
[mydev@myfedora Python]$ graalpython -V
Graal Python 3.7.0 (GraalVM 1.0.0-rc2)
[mydev@myfedora Python]$
[mydev@myfedora Python]$ graalpython knucleotide.py 0 < knucleotide-input1000.txt
Please note: This Python implementation is in the very early stages, and can run little more than basic benchmarks at this point.
Traceback (most recent call last):
  File "knucleotide.py", line 20, in <module>
    b'from os import cpu_count'
ImportError: cannot import name 'cpu_count'
```

ZipPy

ZipPy is a fast and lightweight Python 3 implementation built using the Truffle framework. ZipPy leverages the underlying Java JIT compiler and compiles Python programs to highly optimized machine code at runtime. Repository on Bitbucket.

- **http://thezhangwei.com/**
- **https://github.com/securesystemslab/zippy**
- **Optimizations**
  - **Numeric Types, Type Specializations, Efficient Data Representation**
  - **Control Flow Specializations, Generator Peeling, Optimizing Object Model and Calls**

benchmmark	CPython3	CPython	Jython	PyPy	PyPy3	ZipPy
binarytrees	1.00	0.94	1.99	2.60	2.70	7.31
fannkuchredux	1.00	0.97	0.51	44.53	47.29	87.50
fasta	1.00	1.04	1.55	11.73	11.24	15.57
mandelbrot	1.00	1.08	0.34	10.91	10.82	11.69
meteor	1.00	1.02	0.77	2.64	2.62	2.13
nbody	1.00	0.97	0.73	12.13	12.06	6.17
pidigits	1.00	1.00	0.62	0.98	0.95	0.60
spectralnorm	1.00	1.33	1.89	127.33	127.25	128.10
float	1.00	0.95	1.05	8.64	8.67	17.71
richards	1.00	0.94	1.21	29.53	29.25	50.13
chaos	1.00	1.17	1.55	40.88	25.69	68.28
deltablue	1.00	0.85	1.33	30.08	29.14	23.46
go	1.00	1.08	1.99	6.79	6.66	15.41
mean	1.00	1.02	1.05	12.15	11.68	15.34

Python program — Hosted VM
Jython — Hosting VM
Modular VM (extended from Maxine VM)
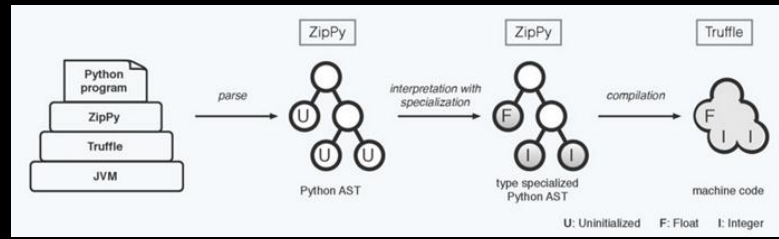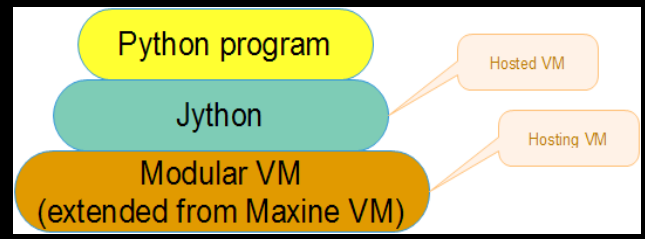
Jython

- **http://www.jython.org**   **//No new release since 2015…**

VOC
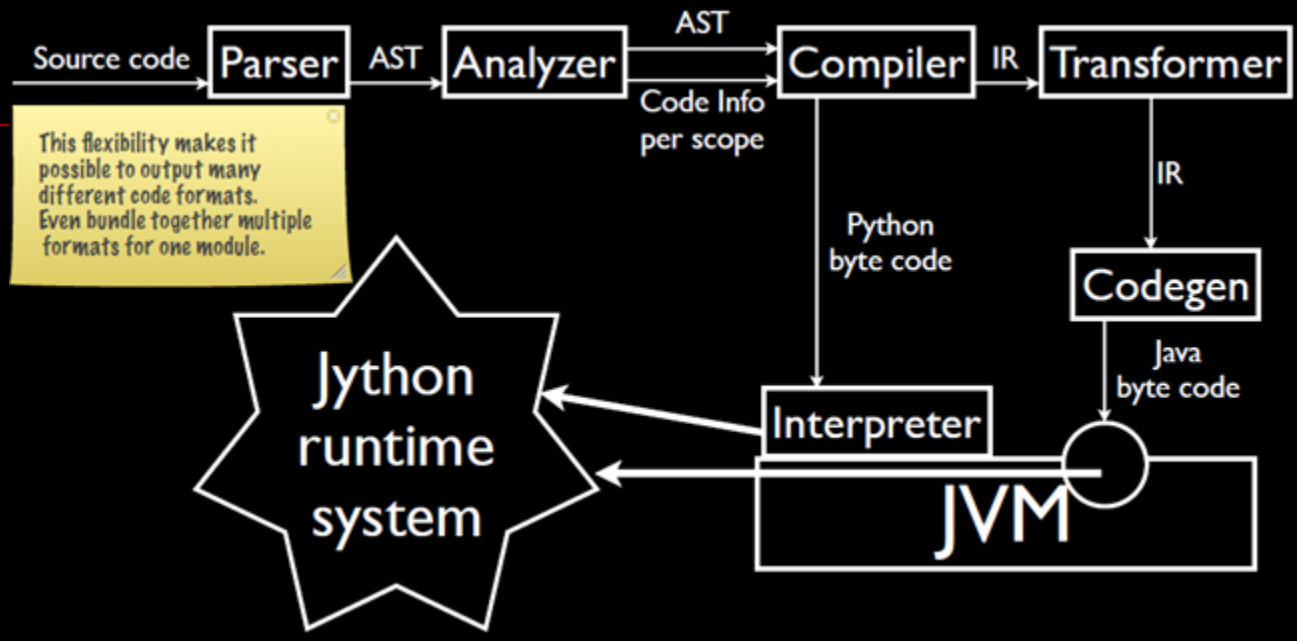
- **https://github.com/pybee/voc/**
- **A transpiler that converts Python code into Java bytecode**

…

4) My Practice
Technical Solution

■ **http://openjdk.java.net/projects/jdk/11/**

Features

JEPs proposed to target JDK 11

review ends

332: Transport Layer Security (TLS) 1.3   2018/06/21

JEPs targeted to JDK 11, so far

181: Nest-Based Access Control
309: Dynamic Class-File Constants
315: Improve Aarch64 Intrinsics
318: Epsilon: A No-Op Garbage Collector
320: Remove the Java EE and CORBA Modules
321: HTTP Client (Standard)

323: Local-Variable Syntax for Lambda Parameters
324: Key Agreement with Curve25519 and Curve448
327: Unicode 10
328: Flight Recorder
329: ChaCha20 and Poly1305 Cryptographic Algorithms
330: Launch Single-File Source-Code Programs
331: Low-Overhead Heap Profiling
333: ZGC: A Scalable Low-Latency Garbage Collector
       (Experimental)
336: Deprecate the Pack200 Tools and API

■ **http://openjdk.java.net/projects/metropolis/**

- Experimental clone of JDK 11 (*not* for immediate release)
- Hosting work on AOT and the Graal compiler
- Definition of "System Java" for implementing HotSpot modules.
  - Experimentation with SVM-style deployment.
- Translation of discrete HotSpot modules into System Java.
- The Big One: Compilation of Graal as System Java for JIT
  - Replacement for C2, then C1, then stub and interpreter generators.
  - This will take a long time, but it's a necessary technology refresh.
- *Tomorrow's reference implementation!*

Source: http://cr.openjdk.java.net/~jrose/pres/201801-JIT-Metropolis.pdf

_MX_

- **https://github.com/graalvm/mx**
- **command-line tool used for the development of Graal projects**
- **missing…**

- **How about integrate MX into Meson?**
- **http://mesonbuild.com/**

OpenJDK 11 on ARM

- **https://github.com/AdoptOpenJDK/openjdk-jdk //OpenJDK11 src**
- **https://github.com/AdoptOpenJDK/openjdk10-nightly/releases**
  https://ci.adoptopenjdk.net/job/openjdk10_build_arm64_linux/

```
Project openjdk10_build_arm64_linux

Last Successful Artifacts
    OpenJDK10_aarch64_Linux_201812061552.tar.gz   346.10 MB  view
Recent Changes
```

- **export JDK_BOOT_DIR=$YOUR_OpenJDK10-AARCH64_HOME**
- **reserve at least 6GB disk space**
- **on ROC-RK3328-CC with Debian 9 + Kernel 4.4.114 + GCC 7.3.0-19 + jemalloc 5.1.0 + 6GB Memory (4GB DDR4 + 2GB Swap)**

```
firefly@firefly:/$ free -m
              total      used      free    shared  buff/cache   available
Mem:           3927       193      3533        49         200        3655
Swap:          2047_        0      2047
```

- **cd $YOUR_OPENJDK11_SRCHOME and run the commands:**
  bash configure --disable-warnings-as-errors
  make JOBS=4 images

```
firefly@firefly:/usr/bin$ ls -l /usr/bin/ld
lrwxrwxrwx 1 root root 7 Jun  9 13:34 /usr/bin/ld -> ld.gold
```

  ~2h2m   for a full build with GNU ld linker 2.30
  ~1h55m for a full build with GNU gold linker 1.15

Integration
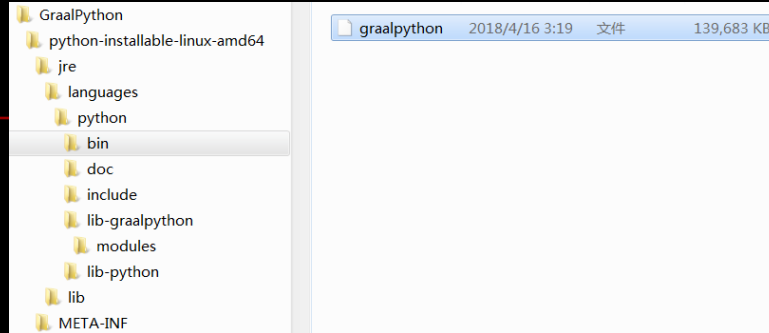
- **https://github.com/graalvm/graalpython/releases/download/vm-1.0.0-rc2/python-installable-linux-amd64.jar**

- **GraalVM EE 1.0.0 RC2**

build GraalPython & GraalVM

- **setup mx**
- **patching for avoid javaCompliance limitation**
- **failed to build GraalVM with OpenJDK 10 & 11**

```
Compiling org.graalvm.compiler.serviceprovider with javac-daemon(JDK 10) failed
Shutting down
  File "/opt/MyWorkSpace/DevSW/Tools/Build/MX/mx/mx.py", line 17693, in <module>
    main()
  File "/opt/MyWorkSpace/DevSW/Tools/Build/MX/mx/mx.py", line 17674, in main
    retcode = c(command_args)
  File "/opt/MyWorkSpace/MyProjs/Java/JDK/GraalVM/graal/substratevm/mx.substratevm/mx_substratevm.py", line 100,
 in build
    orig_command_build(args, vm)
  File "/opt/MyWorkSpace/DevSW/Tools/Build/MX/mx/mx.py", line 11725, in build
    abort('{0} build tasks failed'.format(len(failed)))
  File "/opt/MyWorkSpace/DevSW/Tools/Build/MX/mx/mx.py", line 11251, in abort
    traceback.print_stack()
1 build tasks failed

Compiling org.graalvm.compiler.serviceprovider with javac-daemon(JDK 11) failed
Shutting down
  File "/opt/MyWorkSpace/DevSW/Tools/Build/MX/mx/mx.py", line 17693, in <module>
    main()
  File "/opt/MyWorkSpace/DevSW/Tools/Build/MX/mx/mx.py", line 17674, in main
    retcode = c(command_args)
  File "/opt/MyWorkSpace/MyProjs/Java/JDK/GraalVM/graal/substratevm/mx.substratevm/mx_substratevm.py", line 100, in b
uild
    orig_command_build(args, vm)
  File "/opt/MyWorkSpace/DevSW/Tools/Build/MX/mx/mx.py", line 11725, in build
    abort('{0} build tasks failed'.format(len(failed)))
  File "/opt/MyWorkSpace/DevSW/Tools/Build/MX/mx/mx.py", line 11251, in abort
    traceback.print_stack()
1 build tasks failed
WorkSpace/MyProjs/Java/JDK/GraalVM/graal/compiler/mxbuild/src/org.graalvm.compiler.serviceprovider/javafilelist.txt -
Xlint:all,-auxiliaryclass,-processing,-options -XDignore.symbol.file -encoding UTF-8
```
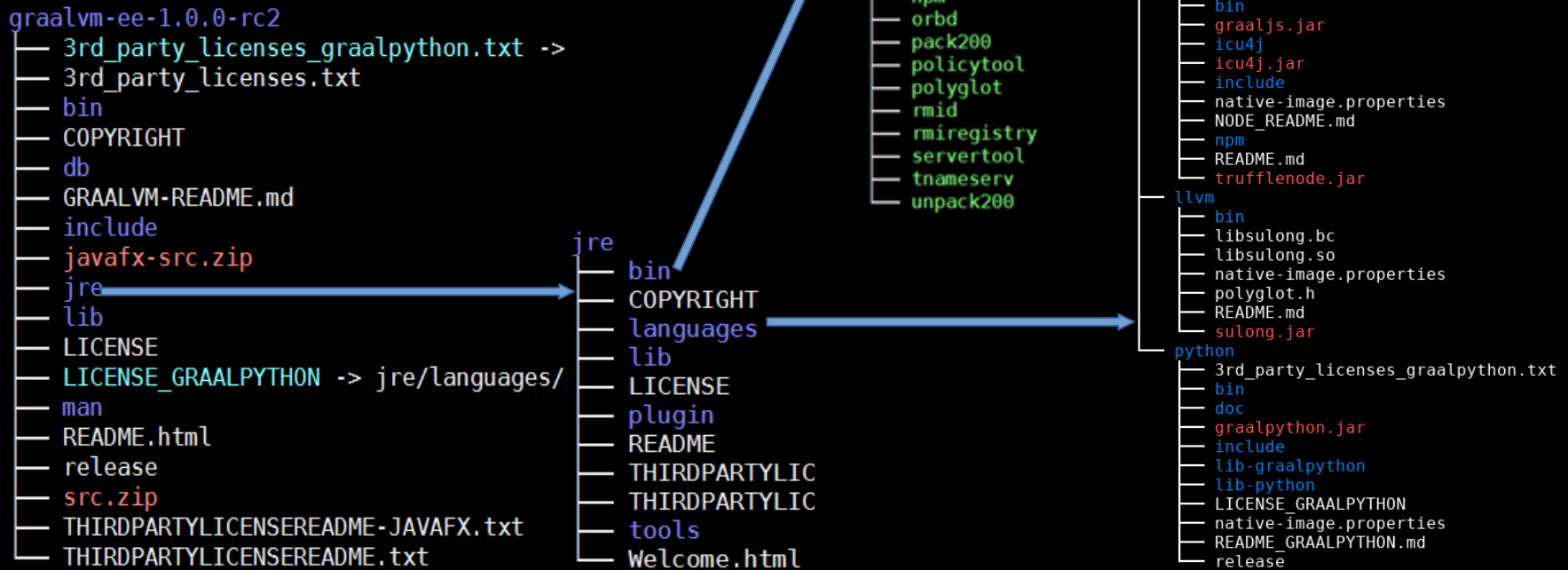
failed to build GraalPython with OpenJDK 11 & LLVM 6.0.0

```
/opt/MyWorkSpace/MyProjs/Java/JDK/GraalVM/sulong/tests/com.oracle.truffle.llvm.tests.sulong/c/stdlib/atexit005.In file in
cluded from /opt/MyWorkSpace/MyProjs/Java/JDK/GraalVM/sulong/tests/com.oracle.truffle.llvm.tests.sulong/c/intrinsics/move
mask.c:1:
In file included from /opt/MyWorkSpace/DevSW/Toolchain/LLVM/clang-llvm-6.0.0-aarch64-linux-gnu/lib/clang/6.0.0/include/em
mintrin.h:27:
In file included from /opt/MyWorkSpace/DevSW/Toolchain/LLVM/clang-llvm-6.0.0-aarch64-linux-gnu/lib/clang/6.0.0/include/xm
mintrin.h:27:
/opt/MyWorkSpace/DevSW/Toolchain/LLVM/clang-llvm-6.0.0-aarch64-linux-gnu/lib/clang/6.0.0/include/mmintrin.h:64:12: error:
 invalid conversion between vector type '__m64' (vector of 1 'long long' value) and integer type 'int' of different size
    return (__m64)__builtin_ia32_vec_init_v2si(__i, 0);

...

/opt/MyWorkSpace/DevSW/Toolchain/LLVM/clang-llvm-6.0.0-aarch64-linux-gnu/lib/clang/6.0.0/include/mmintrin.h:539:12: error
: invalid conversion between vector type '__m64' (vector of 1 'long long' value) and integer type 'int' of different size
    return (__m64)__builtin_ia32_psubw((__v4hi)__m1, (__v4hi)__m2);
                 ^~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
fatal error: too many errors emitted, stopping now [-ferror-limit=]
20 errors generated.
make: *** [c/intrinsics/movemask/00.bc] Error 1
```

challenges

- **prone to break build**

- **customize GraalPython to meet our need**

- **dynamically enable or reload Graal compiler at runtime**
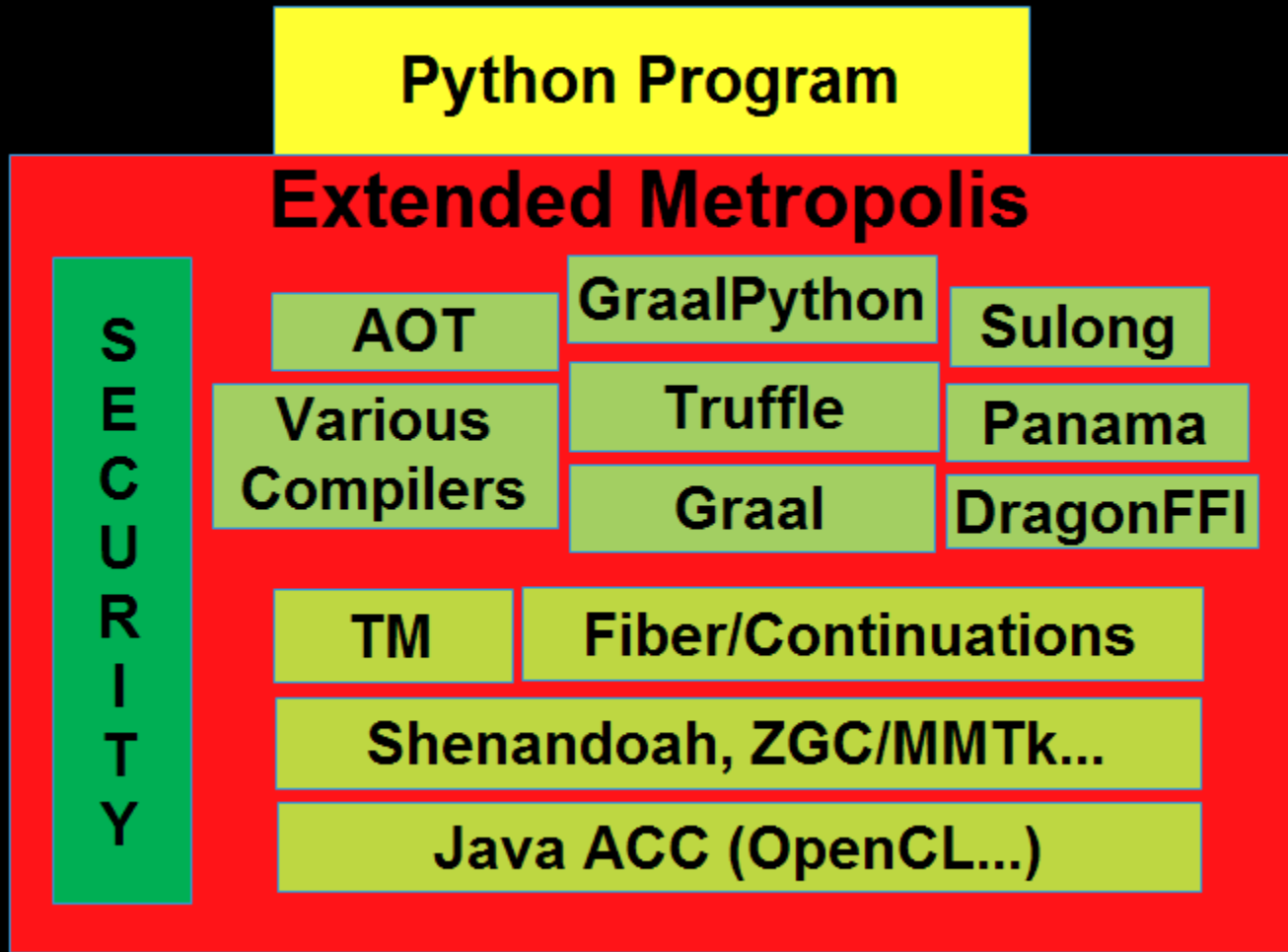
- **deal with JDK, Truffle/Graal, LLVM, Python…**

- **…**

Rethinking of Python Runtime

■ **from my point of view, various Runtime Frameworks for Python implementation:**

	OMR	LLVM	PyPy	GraalVM
Pros	easily leverage new hardware features low-maturity	high efficiency; high-maturity	productivity(RPython); high-maturity	combine continually improved JVM and LLVM techs; productivity(Java);
Cons	productivity (C++/C)?	death of VMKit...	mainly for dynamic language; PyPy3	low-maturity; memory footprint
Performance	experimental/not sure	not enough	not enough	not enough
Native		DragonFFI	CFFI, CPPYY	GNFI (Graal Native Function Interface)
Related Projects	JBM J9/OpenJ9	Unladen Swallow, PySton	Psyco	ZipPy
License	EPL v2.0	LLVM	MIT	GPL v2

5) Future

- extend **Project Metropolis** and customize it for Python

Python Program

Extended Metropolis

S E C U R I T Y

AOT

GraalPython

Sulong

Various Compilers

Truffle

Panama

Graal

DragonFFI

TM

Fiber/Continuations

Shenandoah, ZGC/MMTk...

Java ACC (OpenCL...)

IV. Wrap-up

- **My first ARM-based MicroServer will come in 2019**

- **Different design & architecture from OpenStack**

- **Rethinking of App runtime…**

Thanks!

Reference

Slides/materials from many and varied sources:

- **http://en.wikipedia.org/wiki/**
- **http://www.slideshare.net/**
- **https://www.python.org**
- **http://llvm.org**
- **https://en.wikipedia.org/wiki/Just-in-time_compilation**
- **https://github.com/dropbox/pyston**
- **…**