SFO17-503 OpenJDK Past, Present & Future

Stuart Monteith
Introduction

- Who we are
- Past Year
- Current work
- Future
  - For Linaro, Partners and OpenJDK.
- Q&A
Linaro OpenJDK

- Huawei
- Qualcomm
- ARM
- HPE
Past Year - Linaro

- OpenJDK 9 was the release in development.
  - Backporting to OpenJDK 8u as appropriate.
  - OpenJDK 10 began.
- Emphasis on bugs.
  - Benefits OpenJDK 8u, and fit in with OpenJDK 9 development schedule.
- OpenJDK 9 automation.
  - Only had JDK 8 - moved onto JDK 8u.
- JIRA: 18 bugs, 16 infrastructure, 10 performance enhancements.
  - Various misc/maintenance tasks.
- Ningsheng, Yang, Zhongwei became authors!
  - Goal for rest to become authors too.
- Felix Yang became a committer!
Past Year - Linaro (2)

- Execution of regression tests, with variety of command line options.
  - $ java -XX:+PrintFlagsFinal 2>/dev/null | wc -l
    - 643
  - For JDK8u, JDK9.
- aarch64 vs arm64 port performance investigations.
- Autovectorization investigations, improvements.
  - See “Auto vectorization support in OpenJDK9 Hotspot C2 compiler – BUD17-117”
- GC, scalability investigation.
  - Ongoing - pause times.
  - SPECjbb2015
- Bug fixing
  - JNI Floating point API
  - Register allocator
  - etc.
Timeline 2016-2017 - May 2016

Java 9 Feature Complete
Timeline 2016-2017 - December 2016

Java 9 Feature Complete

-AOT Merged-
- Oracle Unified port Merged -
Ahead of Time Compilation

- Merged December 2016.
- **Experimental** use only.
- x86_64 at this point.
- Uses the Graal backend
  - Java native compiler.
- JDK-8185505 - AOT on Aarch64 (Andrew Haley).
- Micro-services.

```
$ javac HelloWorld.java
$ jaotc --output libHelloWorld.so HelloWorld.class
$ java -XX:AOTLibrary=./libHelloWorld.so HelloWorld
Hello world!

$ jaotc --output libjava.base.so --module java.base
$ java \
-XX:AOTLibrary=./libjava.base.so,./libHelloWorld.so \
HelloWorld \\nHello world!
```
Oracle Port

- Named as the “Unified arm32/arm64 port” [JEP-297](#)
  - arm64==”Oracle Port”, aarch64==”Original OpenJDK port”
- Provides a 32-bit and 64-bit Hotspot port in a unified codebase.
  - The original Aarch64 port is unmodified.
  - Can choose between 64-bit ports with “--with-cpu-port=(arm64|aarch64) ”
- Merged in December into JDK9
  - Won’t be backported to JDK8u (32-bit or otherwise).
- The Oracle arm32 port will be the arm32 port from JDK9 onwards.
  - Oracle’s arm64 port is entangled with arm32, so will stay.
- The OpenJDK aarch64 port is the default port
  - Code sufficiently different between arm64 and aarch64, that a merge will probably be more of a reimplementaion.
  - Andrew Haley is the Aarch64 port lead, Ed Nevill the aarch32 port lead.
Timeline 2016-2017 - January 2017

Java 9 Feature Complete

- AOT Merged -
- Oracle Unified Port Merged -

Rampdown Phase 1

J F M A M J J A S O N D
Timeline 2016-2017 - March 2017

Java 9 Feature Complete

-Rampdown Phase 1-

2017

-AOT Merged-
- Oracle Unified port Merged -

-Rampdown Phase 2-

-J F M A M J J A S O N D J F M A M J J A S O N D-
Timeline 2016-2017 - May 2017

- Java 9 Feature Complete
- AOT Merged
- Oracle Unified port Merged

Rampdown Phase 1
Rampdown Phase 2
Rod Crawford joins JCP
Timeline 2016-2017 - July 2017

- Original JDK9 GA Date
- Rampdown Phase 1
- Rampdown Phase 2
- Java 9 Feature Complete
- -AOT Merged-
- - Oracle Unified port Merged -
- Rod Crawford joins JCP

Timeline 2016-2017 - July 2017

Java 9 Feature Complete
-AOT Merged-
- Oracle Unified port Merged -
Rampdown Phase 1
Rampdown Phase 2
Rod Crawford joins JCP
Original JDK9 GA Date

2017

Timeline:
- July 2016
- August 2016
- September 2016
- October 2016
- November 2016
- December 2016
- January 2017
- February 2017
- March 2017
- April 2017
- May 2017
- June 2017
- July 2017
Timeline 2016-2017 - July 2017

- Java 9 Feature Complete
- AOT Merged - Oracle Unified port Merged
- Rampdown Phase 1
- Rampdown Phase 2
- Rod Crawford joins JCP
- Original JDK9 GA Date
- Actual JDK9 GA Date
JDK9 Contents

- Java Platform Module System “Jigsaw”.
  - Restrictions to sun.misc.Unsafe access. (see VarHandles)
- jlink
  - Package just the Java you need.
- Compact strings
- G1 default garbage collector.
  - CMS deprecated.
- Ahead of Time compilation
  - Put in late, x86_64 only.
- Many, many other changes:
  - 91 JEPS in total: http://openjdk.java.net/projects/jdk9/
- Aarch64 enhancements, bug fixes.
OpenJDK Today

- **OpenJDK 8u**
  - Backported fixes are periodically merged.
  - Fixes from JDK9, JDK10
  - Accepting critical fixes - but essentially the stable release.
    - How many fixes?
- **OpenJDK 9**
  - Released Thursday 21st September.
  - Will continue to service in OpenJDK 9u.
- **OpenJDK 10**
  - Will continue benchmarking, GC, scalability, bug fixing.
  - Architectural features will be considered (ARMv8.1-A+).
  - arm64/aarch64.
  - C2 work will continue.
    - Graal not expected any time soon.
Java 18.3?

- September 6th Mark Reinhold announced proposal for new release process.
- Feature releases every 6 months.
  - 18.3 == March 2018
  - 18.9 == September 2018
- Update releases every 3 months.
  - Security, bug fixes, etc.
- Long-term support release every 3 years.
  - 18.9, 21.9, 24.9, etc.
- Benefits?
  - Less risk of falling behind on official releases.
  - Quicker evolution of the Java language and platform.
- **NB: This is very subject to change.**
Other Developments

- Oracle has also announced intent to bring OpenJDK and their releases closer together:
  - Making open-source the commercial tooling, over a period of time.
  - Shipping OpenJDK binaries, GPLv2
    - This opens up the “restrictions on use” part of the Oracle licenses.
- Oracle has asked about collaborating on open build-and-test infrastructure.
  - Ease publishing early-access builds for features in development.
  - Eventually for publishing authoritative builds of the JDK
- AdoptOpenJDK project doing builds
  - [https://adoptopenjdk.net](https://adoptopenjdk.net)
  - Linaro is a sponsor, as is Packet.
- Project Metropolis
  - Proposed Java-on-Java project
    - See other metacircular implementations: JikesRVM, Maxine, etc.
  - Based on Graal, Panama, Valhalla.
  - Graal expected to replace C2.
Conclusions

- OpenJDK on AArch64 looking stable.
  - I implore you to raise issues when they are encountered.
- Java becoming more open.
  - Couldn’t happen at a better time.
- Lots of work on AArch64 to be done.
  - Continued analysis of Java 9/10.
  - Scalability.
Thank You

#SFO17
BUD17 keynotes and videos on: connect.linaro.org
For further information: www.linaro.org
References

- AOT - http://openjdk.java.net/jeps/295
- Unified arm32/64 port - http://openjdk.java.net/jeps/297
- OpenJDK Vulnerability proposal - http://cr.openjdk.java.net/~mr/ojvg/
- AOT on AArch64: https://bugs.openjdk.java.net/browse/JDK-8185505
- Faster and easier use and redistribution of Java: https://blogs.oracle.com/java-platform-group/faster-and-easier-use-and-redistribution-of-java-se
- Moving Java Forward and Faster: https://mreinhold.org/blog/forward-faster
- Project Metropolis: http://cr.openjdk.java.net/~jrose/pres/201703-YourNextVM.pdf
  http://mail.openjdk.java.net/pipermail/discuss/2017-September/004336.html
  https://www.youtube.com/watch?v=OMk5KoUIOy4
- Some software compatibility: https://wiki.openjdk.java.net/display/quality/Quality+Outreach