The Future of Java & You

Heather VanCura
Director & Chair, JCP Program
Twitter @heathervc
July 2019
Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
Heather VanCura

• Chairperson & Director of the JCP Program
• International Speaker and Java Community Leader
• Open Source & Diversity Fan
• Californian - from San Diego
• Personal Interests: Travel, Fitness, Music, Fun
• Twitter @heathervc
Continuing Growth

12+ Million Developers Run Java

38 Billion Active Virtual Machines

21 Billion Cloud Connected Virtual Machines

#1 Programming Language
Java Philosophies

- Platform Completeness
- Quality and Security
- Modernization and Innovation
- Open and Transparent Evolution
- Developer Productivity & Compatibility
- Active Ecosystem Involvement
Celebrating 20 Years of the JCP Program!
Open Evolution

• 1995: Sun Microsystems develops Java.
• 1998: Sun opens up the development process to its competitors, creating the Java Community Process.
• 2006: Sun open-sources Java SE and Java EE.
• 2007: Sun goes into decline; Java stagnates.
• 2010: Oracle acquires Sun and becomes the steward of Java.
• 2017—: Introduce Faster OpenJDK Release Cycle; Java EE migration to Eclipse.
Organization

Executive Committee

Program Management Office

JSR PARTICIPATION

Expert Group Members

Specification Lead

Contributors

JCP Members

Specification Lead

Contributors

Expert Group Members
Membership Quotes

• “Joining the JCP, especially being a member of several expert groups, has had a big impact on my career. Mostly by making it visible for the leadership and managers in my company that I am actually a part of forming the future of the platform. I also think, or hope, that it is an inspiration for my colleagues.”

  - Ivar Grimstad

• “Joining the JCP is like being a Java citizen.” - Heinz Kabutz
Collaborative Development - How does is work?

• Java Specification Requests (JSRs)
  – A JSR is a single version of a Java specification.

• JSRs are led by a community member (the Spec Lead), with a group of interested members (the Expert Group) helping with the day-to-day decisions and work.
  – Any JCP member can submit and lead a JSR.

• Each Expert Group must deliver:
  – The Specification
  – A Reference Implementation (RI)
  – A Technology Compatibility Kit (TCK)
The JSR Development Cycle

• Every project developed through the JCP follows the JSR lifecycle
• It Includes formal public reviews and votes by the Executive Committee.
• Full Members can submit & lead JSRs, serve as Expert Group members on JSRs.
• Associate Members can participate as Contributors on JSRs.
Compatibility Triangle

IS THE SPECIFICATION UNAMBIGUOUS?

IS THE TCK CORRECT? DOES THE RI CONFORM?

CAN YOU BUILD AN IMPLEMENTATION?

Technology Compatibility Kit

Reference Implementation

Specification
An International Effort
Who Are The Members?

- Corporations
- Non-Profit & Open Source
- Java User Groups (JUG)
- Individual Developers
The Executive Committee

- Corporations
- Non-Profits/Open Source
- Java User Groups
- Individual Developers
How: Open Standards & Open Source

• We need both!
Complementary

- Open source important requirement. So are Standards.
- Coop-etition. Agree on what to standardize (cooperation) & what to compete on (implementations).
- Open standards implemented in open source = easier to implement standard & for developers to understand technology.
- Results in more implementations and greater adoption of the standard.
- An effective way to develop a standard-start with an open source project that has demonstrated the need for standardization.
Organization Focus

- New revisions of platform.
- "JCP.Next" reforms
- Increased participation from Community.
Open Source Implementations

- The Reference Implementations of the Java platform are developed collaboratively and released under open-source license
- Java SE: OpenJDK
Java SE Platform

Last Major Release

Modularity

Cadence
New JDK Release Model – Starting with JDK 9, LTS every 3 yrs

- JDK 6
- JDK 7
- JDK 8
- JDK 9
- JDK 10
- JDK 11 (18.9 LTS)
- JDK 12
- JDK 13
- JDK 14
- JDK 15
- JDK 16
- JDK 17 (21.9 LTS)
From Oracle JDK to OpenJDK from Oracle
Features Open-Sourced in Java

- **Application Class Data Sharing** *(Now available in OpenJDK 10)*
  - Enables you to place classes from the standard extensions directories and the application class path in the shared archive

- **Project ZGC** *(Now available in OpenJDK 11)*
  - Low latency garbage collector to support multi-terabyte heaps

- **Flight Recorder** *(Now available in OpenJDK 11)*
  - Collects diagnostic and profiling data about a running Java application

- **Mission Control** *(Now available in OpenJDK 11)*
  - Monitor and manage Java applications with minimal performance overhead

- **Usage Logger** *(Now available in OpenJDK 11)*
  - Logs how the JRE’s are being used in your systems
JDK 9

- Released September 2017
- #WorksFineOnJDK9
- Last Major Release – 100+ features
JDK 10

- Released March 2018
- First feature release
- 12 JEPs (Java Enhancement Proposals)
JDK 11 – Sep 2018

- 17 JEPs
- 4 JEPs targeted initially
- #WorksLikeHeavenonJDK11
- http://jdk.java.net/11/
JDK 12 – JSR 386

• March 2019 – available today
• 8 JEPs
• New model calls for JEPS to be targeted only when ready
• http://openjdk.java.net/projects/jdk/12/

JDK 12
This release will be the Reference Implementation of version 12 of the Java SE Platform, as specified by JSR 386 in the Java Community Process.

Status
JDK 12 is in the Release-Candidate Phase.
The overall feature set is frozen. No further JEPs will be targeted to this release.
The stabilization repository, jdk/jdk12, is open only for P1 bug fixes, with approval, per the JDK Release Process (JEP 3).
  • Release-Candidate bugs
  • Bug-Deferral Process

Schedule
2018/12/13  Rampdown Phase One (fork from main line)
2019/01/17  Rampdown Phase Two
2019/02/07  Release-Candidate Phase
2019/03/19  General Availability

Features
189: Shenandoah: A Low-Pause-Time Garbage Collector (Experimental)
230: Microbenchmark Suite
325: Switch Expressions (Preview)
334: JVM Constants API
340: One AArch64 Port, Not Two
341: Default CDS Archives
344: Abortable Mixed Collections for G1
346: Promptly Return Unused Committed Memory from G1
JDK 13 – JSR 388

• Planned Release September 2019
• 5 JEPS
• In ramp down phase
• Early Access Builds are out
JDK 14 – JSR 389

• JSR filed and posted on jcp.org
• Planned Release March 2020

JDK 14

This release will be the Reference Implementation of version 14 of the Java SE Platform, as specified by JSR 389 in the Java Community Process.

Status

The development repositories are open for bug fixes, small enhancements, and JEPs as proposed and tracked via the JEP Process.

Last update: 2019/6/18 20:55 UTC
And Beyond

- Panama
- ZGC
- Portola
- Valhalla
- Amber
- Loom
Java SE Platform Investments

• Security is #1 priority

• Improving Java developer productivity and compatibility (Amber, Panama, Loom)

• Increasing density (Valhalla)

• Improving startup time (AOT, App CDS)

• Improving predictability (zGC, Shenandoah)

• Simplifying serviceability and profiling (JFR, JMC)
Project Valhalla: Object Data layout

- Java is very good at optimizing code, less so at optimizing data
- Java’s type system gives us primitives, objects, and arrays
- But flexibility is not exactly where we need it
- The big problem: object identity
- Project Valhalla – Value Types
Project Portola

• Java in a world of Containers
• Java’s characteristics make it ideal for container deployment
  – Safe & Secure, Performant, Reliable, Rich Eco System
• We are committed to having Java remain the first choice for deployments in the cloud
Project ZGC: A Scalable Low-Latency Garbage Collector

- Open sourced earlier this year
- [http://wiki.openjdk.java.net/display/zgc/](http://wiki.openjdk.java.net/display/zgc/)
- Early Access builds available: [http://jdk.java.net/zgc/](http://jdk.java.net/zgc/)
  - Enable ZGC: `-XX:+UseZGC`
Project Panama

• Foreign Functions & Data
• Improved Java/Native Interoperability
• Simple, safe, and performant replacement for JNI
• Access to low-level hardware functionality through normal Java code
  – Vector instructions, special memory types (for example non-volatile memory)
• Big Data, Machine Learning
Project Loom

• Easier and more scalable concurrency model
  – Making blocking calls virtually free
• “Fibers” (lightweight threads) and continuations
  – Millions of fibers can be spawned in a single JVM instance
Project Amber

- Language improvements
  - Dynamic Class-File Constants
  - Raw string literals
  - Switch expressions
  - Pattern matching
Download Early Access for Panama and Valhalla

jdk.java.net

Java Development Kit builds, from Oracle

Ready for use: JDK 11

Early access: JDK 13, JDK 12, Jpackage, OpenJFX, Panama, Valhalla, & JMC

Reference implementations: Java SE 12, Java SE 11, 10, 9, 8, & 7
Get Involved

Follow on Twitter

@OpenJDK

Join and become an OpenJDK contributor

https://openjdk.java.net
Updated Java Training & Certifications

https://education.oracle.com/learn/java/pPillar_80
JCP.next: Changing the Constitution

We the People

of the United States, in Order to form a more perfect Union, establish Justice, insure domestic Tranquility, provide for the common Defence, promote the general Welfare, and secure the Blessings of Liberty to ourselves and our Posterity, do ordain and establish this CONSTITUTION for the United States of America.

Article I.

Section 1. All legislative Powers herein granted shall be vested in a Congress of the United States, which shall consist of a Senate and House of Representatives.

Section 2. The House of Representatives shall be composed of Members chosen every second Year by the People of the several States, and the Electors in each State shall have the Qualifications requisite for Electors of the most numerous Branch of the State Legislature.

No Person shall be a Representative who shall not have attained to the Age of twenty-five Years, and been seven Years a Citizen of the United States, and who shall not, when elected, be an Inhabitant of that State in which he shall be chosen.

Representatives and direct Taxes shall be apportioned among the several States which may be included within this Union, according to their respective Numbers, which shall be determined by adding to the whole Number of free Persons, including those bound to Service for a Term of Years, and excluding Indians not taxed, three fifths of all other Persons. The actual Enumeration shall be made within three Years after the first Meeting of the Congress of the United States, and within every subsequent Term of ten Years, in such Manner as they shall by Law direct. The Number of Representatives shall not exceed one for every thirty Thousand, but each State shall have at least one Representative; and until such enumeration shall be made, the State of New Hampshire shall be entitled to chuse three, Massachusetts eight, Rhode-Island and Providence Plantations one, Connecticut five, New-York six, New Jersey four, Pennslyvania eight, Delaware one, Maryland six, Virginia ten, North Carolina five, South Carolina five, and Georgia three.

When Vacancies happen in the Representation from any State, the Executive Authority thereof shall issue Writs of Election to fill such Vacancies.

The House of Representatives shall choose their Speaker and other Officers; and shall have the sole Power of Impeachment.

Section 3. The Senate of the United States shall be composed of two Senators from each State, chosen by the Legislature thereof, for six Years; and each Senator shall have one Vote.

Immediately after they shall be assembled in Consequence of the first Election, they shall be divided as equally as may be into three Classes. The Seats of the Senators of the first Class shall be vacated at the Expiration of the second Year, of the second Class at the Expiration of the fourth Year, and of the third Class at the Expiration of the sixth Year, so that one-third may be chosen every second Year; and if Vacancies happen by Resignation, or otherwise, during the Recess of the Legislature of any State, the Executive thereof may make temporary Appointments until the next Meeting of the Legislature, which shall then fill such Vacancies.

No Person shall be a Senator who shall not have attained to the Age of thirty Years, and been nine Years a Citizen of the United States, and who shall not, when elected, be an Inhabitant of that State for which he shall be chosen.

The Vice President of the United States shall be President of the Senate, but shall have no Vote, unless they be equally divided.

The Senate shall choose their other Officers, and also a President pro tempore, in the absence of the Vice President, or when he shall exercise the Office of President of the United States.

The Senate shall have the sole Power to try all Impeachments. When sitting for that Purpose, they shall be on Oath or Affirmation. When the President of the United States is tried, the Chief Justice shall preside: And no Person shall be convicted without the Concurrence of two thirds of the Members present.

Judgment in Cases of Impeachment shall not extend further than to removal from Office, and disqualification to hold and enjoy any Office of honor, Trust or Profit under the United States: but the Party convicted shall nevertheless be liable and subject to Indictment, Trial, Judgment and Punishment, according to Law.
Transparency
JCP Executive Committee (EC)

- One Java, One EC - merged ECs
- Resize the EC - 2019
Participation
Move Faster
Broadening JCP Membership

- Eliminate Barriers to participation.
- Introduce new Membership levels.
- No Membership Fees.
- Electronic Signatures.
- Add Contributors for JSR Expert Groups.
- Add Associate Seats on Executive Committee.
Membership Levels

• Associate Members are individuals who can be listed as contributors to JSRs and vote for the Executive Committee.

• Partner Members are Java User Groups and other non-profit organizations that can serve on and vote for the Executive Committee.

• Full Members can serve on Expert Groups, lead JSRs, serve on and vote for the Executive Committee
The Streamline the JSR Development Cycle
Includes formal public reviews and votes by the Executive Committee.
See the Process Document for the details.
Evolution Continues...What’s Next?

- JSR 387, Streamlining the JCP Program
  - JCP.Next Working Group
  - Final Release December 2018
- Follow the EC Summaries and discussion:
  - https://jcp.org/en/resources/EC_summaries
More Open Than Ever Before
How will you Participate?

• As an Individual – OK
• As part of a team – better
• Work through JUG or employer
• Help each other
• Teach other
• Work with each other
Working Together – We Achieve More
JUG Members & Adopt-a-JSR Global Adoption Efforts

Abdijan JUG (Ivory Coast)  Detroit JUG (USA)
Alpes JUG (France)  Duchess (Women)
Austin JUG (USA)  ESPRIT Tunisian JUG (Tunisia)
BeJUG (Belgium)  FASOJUG (Burkina Faso)
BreizhJUG (Brittany)  Guatemala Java User Group
CEJUG (Brazil)  Green Tea JUG (China)
Central Ohio JUG (USA)  Houston JUG (USA)
Chicago JUG  iJUG e.V. (Germany)
ChinaNanjingJUG  IndiJava (India)
Connecticut JUG (USA)  ITP_JAVA (Peru)
Istanbul JUG  Japan JUG
JUG Frankfurt [JUGF] (Germany)  JUG EG (Egypt)
Java Hellenic User Group (Greece)  JUG Indonesia
Java Student User Group (Vienna)  JUG JogLoSemar (Indonesia)
Java Web User Group (London)  JUG-MK (Macedonia)
Joji JUG (South Africa)  JUG-RU (Russia)
JUG-AFRICA  JUG-USA
JUG Chennai (India)  Lagos JUG (Nigeria)
JUG-Cologne (Germany)  London Java Community (UK)
JUG Dortmund (Germany)  Malaysia-JUG
JUG Members & Adopt-a-JSR Global Adoption Efforts
JUG Members & Adopt-a-JSR Global Adoption Efforts

Madras JUG (India)
Nashville JUG (United States)
Oklahoma City JUG (USA)
Philly JUG (USA)
Rio JUG (Brazil)
Riviera JUG (France)
Ruhrjug (Germany)
Silicon Valley JavaFX User Group
SeneJUG (Senegal)
SouJava (Brazil)
Utah JUG (USA)
Vancouver Island JUG (Canada)
JUGs around the World - Driving Adoption
1) Pick JSR - New JSRs

• Recently Submitted:
  – MVC 1.0 (JSR 371)
  – Visual Rec (JSR 381)
  – Desktop Application API (JSR 377)
  – Portlet 3.0 Bridge for JSF 2.2 (JSR 378)
  – Units of Measurement 2.0 (JSR 385)
  – Java SE 13 (JSR 388)

[https://jcp.org/en/jsr/stage?listBy=active]
List of Active JSRs (posting in last 12 months):
http://jcp.org/en/jsr/all?status=Active&activeMonths=12
2) Communication: Two Way Street
3) Decide on Steps: Take Action

- Share ideas and feedback, comment on list and public issue trackers.
- Read early versions and share feedback on specifications and Javadocs.
- Download and provide feedback on early access reference implementation.
- Try writing sample applications using early builds of reference implementation.
- Write or speak about the technology and encourage others to participate. Translate into your native language.
- Evangelize the JSR - social media, blogging or lightning talks.
- Help with documentation.
4) Follow Public Discussions & Comment - Issue Tracker
5) Participate in Hack Days & Workshops – Have Fun!
Early Access

• In adoption group, you can discuss Early Access Builds – provide comments
Participate in OpenJDK

• Download Early Access Builds (13 available now)
• Adoption Group & Quality Outreach
• Join mail list prior to sending feedback: http://mail.openjdk.java.net/mailman/listinfo/adoption-discuss
• https://wiki.openjdk.java.net/display/quality/Quality+Outreach
Quality Outreach part of Adoption Group

- Test FOSS projects on EA OpenJDK Builds
- Open Source Projects - Quality Outreach examples
  - Eclipse Collections - Nikhil Nanivadekar
  - Apache Maven - Robert Scholte
We need you!

• Follow the JCP online: http://JCP.org
  – Join the JCP: https://jcp.org/en/participation/membership
  – Twitter @jcp_org #JCP, @heathervc
  – blogs.oracle.com/jcp
  – Facebook: Java Community Process
Oracle Code One

• JavaOne expansion and renaming - more Tracks, Languages and Communities - adding Go, Rust, Python, JavaScript, and R in addition to Java – join us!

• September 16-19 2019 San Francisco CA USA

• https://www.oracle.com/code-one/index.html
thank you, ευχαριστώ, shukriya, dziekuje, धन्यवाद,
tak, tack, grazie, gracias, danke, Благодаря, teşekkür ederim
merci, obrigado, bedankt, kiitos, xie xie, 謝謝。

Questions?
Find me on Twitter: @heathervc
email: heather at jcp dot org