# 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**

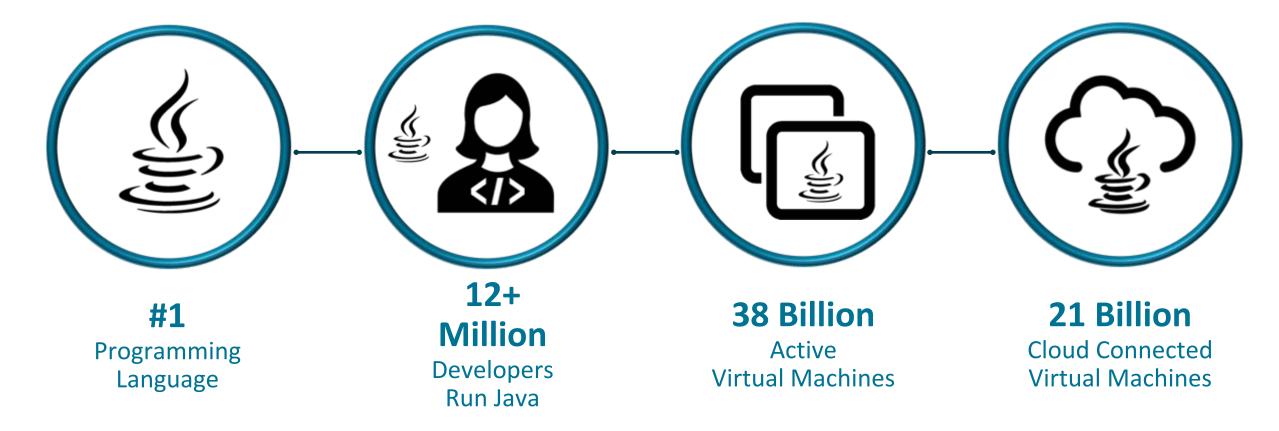
- Chairperson & Director of the JCP Program
- International Speaker and Java Community Leader
- Open Source & Diversity Fan
- Californian from San Diego
- Personal Interests: Travel, Fitness, Music, Fun
- Twitter @heathervc







# **Continuing Growth**



### **Java Philosophies**

- Platform Completeness
- Quality and Security
- Modernization and Innovation
- Open and Transparent Evolution
- Developer Productivity & Compatibility
- Active Ecosystem Involvement

<b>Java</b> <sup>™</sup>





# **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.
- 2011-2017: New Versions of Platforms Released.
- 2012—: Greater Engagement of Developer Community & User Groups.
- 2017—: Introduce Faster OpenJDK Release Cycle; Java EE migration to Eclipse.











### **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)



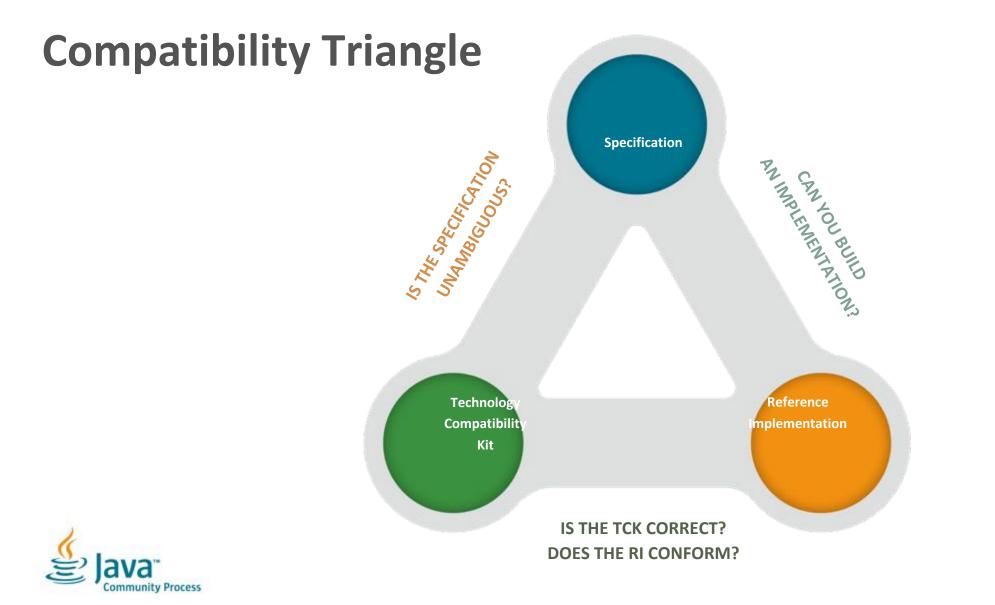


### **The JSR Development Cycle**

- 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.









### **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







# 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**

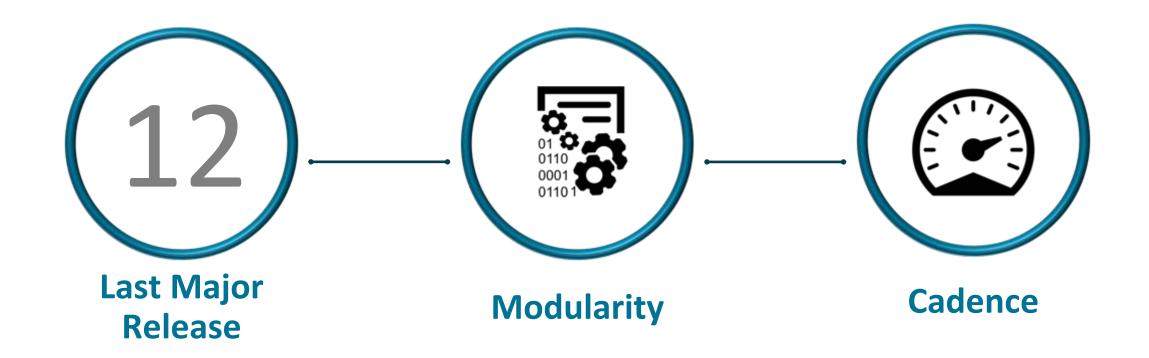
- The Reference Implementations of the Java platform are developed collaboratively and released under open-source license
- Java SE: OpenJDK







### Java SE Platform

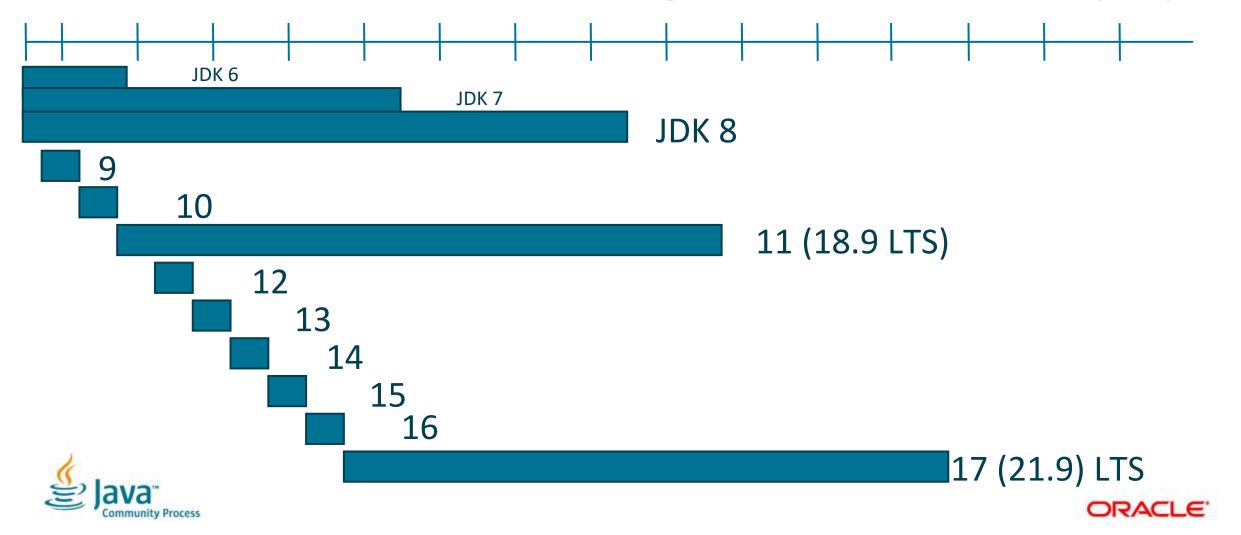




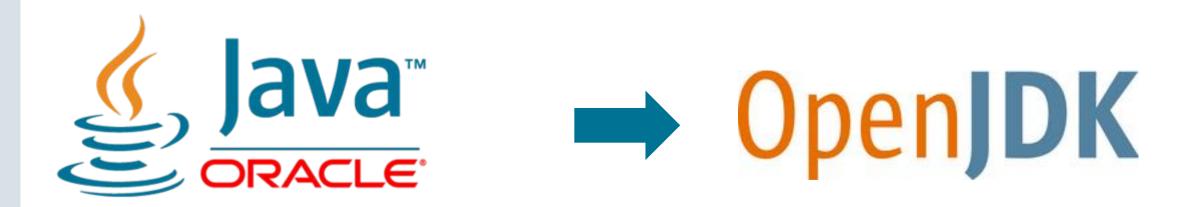


Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

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



### From Oracle JDK to OpenJDK from Oracle



Copyright © 2018, Oracle and/or its affiliates. All rights reserved.

### **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 OpenJDK11)
  - Monitor and manage Java applications with minimal performance overhead
- Usage Logger (Now available in OpenJDK11)
  - Logs how the JRE's are being used in your systems



# JDK 9

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



C Q fears 00 0 · · · · **OpenJDK** JDK 9 OpengDM RAD Instanting Contributing The goal of this Project is to produce an open-source reference implementation of the Java SE 9 Platform defined by JSR 379 in the Java Community Process. The schedule and features of this release are proposed and tracked via the JEP Desempers' Gom Mailing lass and mini-Process, as amended by the JEP 2.0 proposal. Bullevil Cernut Schedule (BP Presses 2016/05/26 Feature Complete 2016/12/22 Feature Extension Complete Source code 2017/01/05 Rampdown Start Mancurpat Journalism (8) All Tests Run 2017/02/09 2017/02/16 Zero Bug Bounce 2017/03/16 Rampdown Phase Two Adoption Adv7 Buill Compatibility & Specification Review 2017/06/22 Initial Release Candidate 2017/07/06 Final Release Candidate 2017/09/21 General Availability Status Care Librarian Governing Beam HotSpot We are now in the final phase of the release, in which we aim to fix only those bugs that are truly showstoppers to the success of the release. Please see the Release internationalis Candidate page for details. **Quick links** [candidate bugs] Release-Candidate Phase Rampdown Phase Two [candidate bugs] Rampdown Phase One [candidate bugs] Bug-deferral process (RDP 1 and later) [pending requests] Help Projects Invervented Amber Antie Engree Built intraction Cack and Code Tosts Code Tosts Code Tosts Code Tosts Fix-request process (RDP 2 and later) [pending requests] Feature-Complete extension request process [pending requests] Features 102: Process API Updates 110: HTTP 2 Client 143: Improve Contended Locking Common VM Interface Compiler Gran Desice IIO Duke Font Scaler 158: Unified JVM Logging 165: Compiler Control 193: Variable Handles 197: Segmented Code Cache Premieturiller Toolo Grad . Graders Remove 199: Smart Java Compilation, Phase Two 200: The Modular JDK Narthurs Privagest 201: Modular Source Code 211: Elide Deprecation Warnings on Import Statements IDK 7 Updates IDK 8 Java SK 8 IDK 8 Java SK 8 IDK 8 Java SK 8 IDK 10 212: Resolve Lint and Declint Warnings 213: Milling Project Coin 214: Remove GC Combinations Deprecated in JDK 8 215: Tiered Attribution for javac man frie Marri 216: Process Import Statements Correctly 217: Annotations Pipeline 2.0 219: Databram Transport Layer Security (DTLS) 220: Modular Run-Time Images Manuary Market 221: Simplified Doclet API 222: jshell: The Java Shell (Read-Eval-Print Loop) Multi-Language VM Nashori Raw I/D OpergFit Panama 223: New Version-String Scheme 224: HTML5 Javadoc 225: Javadoc Search 226: UTF-8 Property Files Panimar Port Akriki33 Rott Akriki33 Rott Akriki34 Rott Hab Port Hac OS K Rott HIPS Rott HIPS Rott Reserves/KiAki Rott Reserves/KiAki 227: Unicode 7.0 228: Add More Diagnostic Commands 229: Create PKCS12 Keystores by Default 231: Remove Launch-Time JRE Version Selection 232: Improve Secure Application Performance 233: Generate Run-Time Compiler Tests Automatically 235: Test Class-File Attributes Generated by javac 236: Parser API for Nashorn 237: Linux/AArch64 Port 238: Multi-Release JAR Files 240: Remove the JVM Ti hprof Agent. 241: Remove the that Tool 243: Java-Level JVM Compiler Interface 244: TLS Application-Layer Protocel Negotiation Extension 245: Validate IVM Command-Line Flag Arguments 246: Leverage CPU Instructions for GHASH and RSA 247: Compile for Older Platform Versi

....

2018

Légal

Groups (Inveniew) 20 Groutes

jAQ Handara

National National National National National National National National Security Security Security Security Security Security National Nat

ICACINA JOK 6

Lambda

Locate Enits

Update Mobile Modulee

Port: 53904 Portola SCTP

Dune for

famal data

**ICK 7** 



### **JDK 10**

- Released March 2018
- First feature release
- 12 JEPs (Java Enhancement Proposals)



X A\* JDK 10 + G openjdk.java.net/projects/jdk/10/ --- 💟 🏠 生 🔍 JDK 10  $\rightarrow$ III\ 🖸 😣 Ξ ŵ ~  $\rightarrow$ 

#### **OpenJDK**

Open(DK FAQ

Contributing Sponsoring Developers' Guide

Mailing lists IRC · Wiki

**JEP Process** search

Source code Mercurial Bundles (6) Groups

(overview) 2D Graphics Adoption AWT Build Compatibility & Specification Review Compiler

Conformance

Legal

Bylaws · Census

Installing

This release will be the Reference Implementation of the next version of the Java SE Platform, as specified by JSR 383 in the Java Community Process.

#### Status

**JDK 10** 

JDK 10 is in the Release-Candidate Phase, in which we're fixing only those bugs that are absolutely critical to the success of the release. Please see the Release-Candidate Phase page for process details.

#### Schedule

2017/12/14	Rampdown Phase One
2018/01/11	All Tests Run
2018/01/18	Rampdown Phase Two
2018/02/08	Initial Release Candidate
2018/02/22	Final Release Candidate
2018/03/20	General Availability

#### Features

Core Libraries Governing Board HotSpot Internationalization JMX Members Networking NetBeans Projects Porters Quality Serviceability Security Serviceability Sound Swing Web Projects	286: Local-Variable Type Inference 296: Consolidate the JDK Forest into a Single Repository 304: Garbage-Collector Interface 307: Parallel Full GC for G1 310: Application Class-Data Sharing 312: Thread-Local Handshakes 313: Remove the Native-Header Generation Tool (javah) 314: Additional Unicode Language-Tag Extensions 316: Heap Allocation on Alternative Memory Devices 317: Experimental Java-Based JIT Compiler
Projects (overview)	317: Experimental Java-Based JT Compiler 319: Root Certificates
Amber Annotations Pipeline 2.0	322: Time-Based Release Versioning
Audio Engine Build Infrastructure Caciocavallo	Last update: 2018/2/9 18:21 UTC

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.



# JDK 11 – Sep 2018

- 17 JEPs
- 4 JEPs targeted initially
- #WorksLikeHeavenonJDK11
- http://jdk.java.net/11/





 $( \rightarrow ) \rightarrow$ G Open | DK Workshop Open(DK FAQ installing. Contributing Sponsoring openid Developers' Guide Mailing lists IRC - Wiki Bylaws - Census Legal **JEP Process** search Source code Mercurial Bundles (6) Groups (overview) 2D Graphics Adoption AWT Build Compatibility & Specification Review Complier Conformance Core Libraries **Governing Board** HotSpot Internationalization IMX: Members Networking NetBeans Projects Porters JEPs tar Quality Security Serviceability 309: Dyn Sound Swing Winerability 318: Epsi Web Projects 320: Rem (overview) Amber 323: Loca Annotations Pipeline 2.0 Audio Engine Build Infrastructure Caciocavallo Closures Code 3ools Coin Common VM Interface Compiler Grammar Detroit Device VO Duke Font Scaler **Pramebuffer Toolkit** 

ළ JDK 11 ල බ

. . .

×

(i) openjdk.java.net/projects/jdk/11/

+

#### K JDK 11

This release will be the Reference Implementation of version 11 of the Java SE Platform, as specified by JSR 384 in the Java Community Process.

#### Status

JDK 11 is in Rampdown Phase One.

The overall feature set is frozen. No further JEPs will be targeted to this release.

The stabilization repository, jdk/jdk11, is open for select bug fixes and, with approval, late enhancements per the JDK Release Process (JEP 3).

- RDP 1 candidate bugs
- Bug-Deferral Process

Late-Enhancement Request Process

#### Schedule

2018/06/28	Rampdown Phase One (fork from main line)
2018/07/19	All Tests Run
2018/07/26	Rampdown Phase Two
2018/08/16	Initial Release Candidate
2018/08/30	Final Release Candidate
2018/09/25	General Availability

#### Features

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 332: Transport Layer Security (TLS) 1.3 333: ZGC: A Scalable Low-Latency Garbage Collector (Experimental) 335: Deprecate the Nashorn JavaScript Engine 336: Deprecate the Pack200 Tools and API

### ORACLE

# 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

### ORACLE

### JDK 13 – JSR 388

- Planned Release
   September 2019
- 5 JEPS
- In ramp down phase
- Early Access Builds are out



#### JDK 13

This release will be the Reference Implementation of version 13 of the Java SE Platform, as specified by JSR 388 in the Java Community Process.

#### Status

#### JDK 13 is in Rampdown Phase One.

The overall feature set is frozen. No further JEPs will be targeted to this release.

The stabilization repository, jdk/jdk13, is open for select bug fixes and, with approval, late enhancements per the JDK Release Process (JEP 3).

- RDP 1 candidate bugs
- Bug-Deferral Process
- Late-Enhancement Request Process

#### Schedule

2019/06/13	Rampdown Phase One (fork from main line)
2019/07/18	Rampdown Phase Two
2019/08/08	Initial Release Candidate
2019/08/22	Final Release Candidate
2019/09/17	General Availability

#### Features

350: Dynamic CDS Archives
351: ZGC: Uncommit Unused Memory
353: Reimplement the Legacy Socket API
354: Switch Expressions (Preview)
355: Text Blocks (Preview)



### JDK 14 – JSR 389

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

### **JDK 14**

k

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.

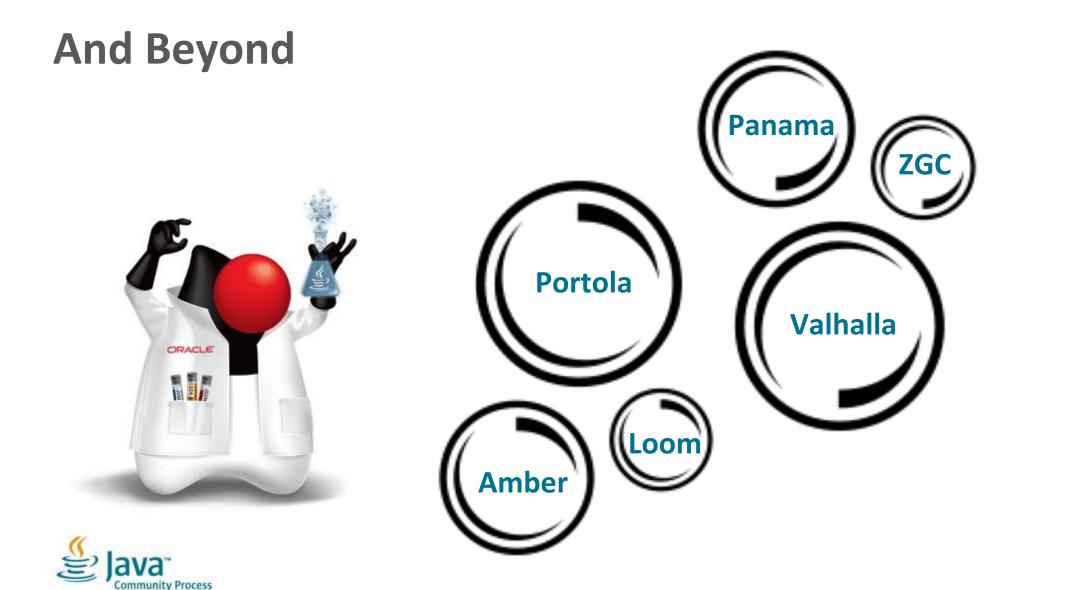
#### Status

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









### 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
- <u>http://wiki.openjdk.java.net/display/zgc/</u>
- Early Access builds available: <a href="http://jdk.java.net/zgc/">http://jdk.java.net/zgc/</a>
  - 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**



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





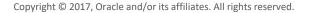


Follow on Twitter **@OpenJDK** 



Join and become an OpenJDK contributor https://openjdk.java.net

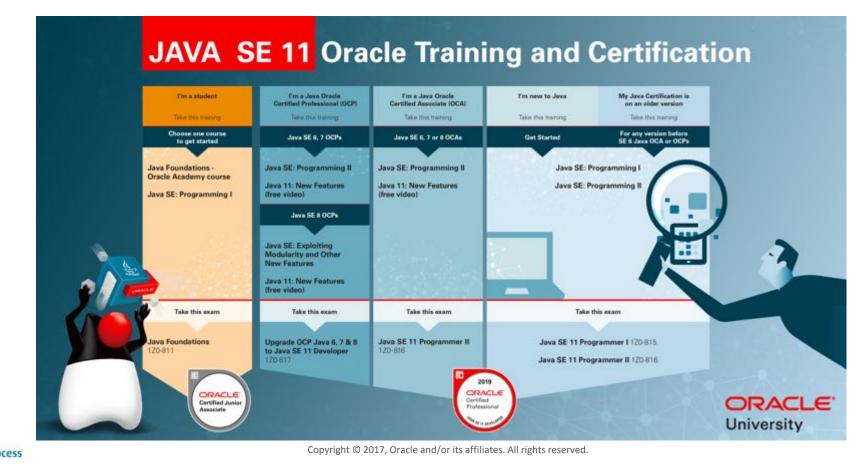






## Updated Java Training & Certifications

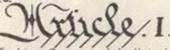
https://education.oracle.com/learn/java/pPillar\_80



ORACLE

#### JCP.next: Changing the Constitution

mestic 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.



SECTION 1. All legislative Powers berein 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, Pennsylvania 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 chuse 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 chuze their other Officers, and also a President pro tempore, in the absence of the Vice President, or uben 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.

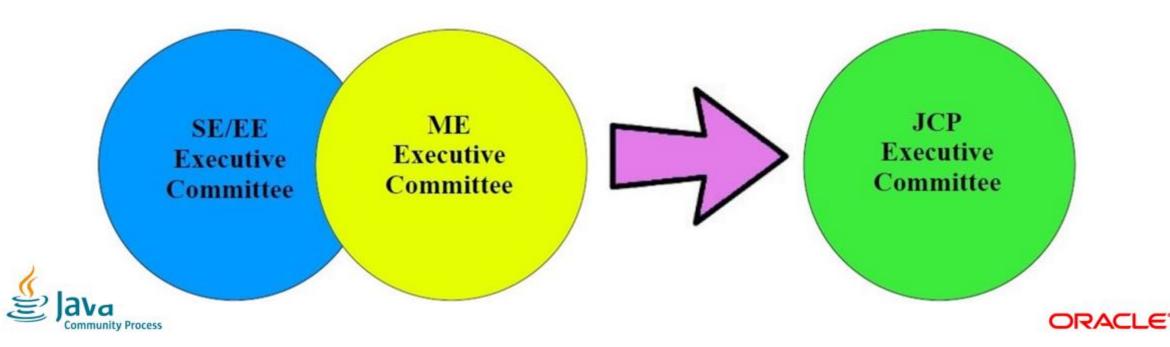
Judgment in Cases of Impeachment shall not extend further than to removal from Office, and disqualification to hold and enjoy any Office of honor, Trust or Profit under the United States: but the Party convicted shall nevertheless be liable and subject to Indictment, Trial, Judgment and Punishment, according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







## 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**

**Community Process** 





#### 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)

JUG-EG (Egypt) JUG Frankfurt [JUGF] (Germany) JUG Indonesia JUG JogLoSemar (Indonesia) JUG-MK (Macedonia) JUG-RU (Russia) JUG-USA Lagos JUG (Nigeria) London Java Community (UK) Malaysia-JUG MoroccoJUG

Japan JUG Java Hellenic User Group (Greece) Java Student User Group (Vienna) Java Web User Group (London) Jozi JUG (South Africa) JUG-AFRICA JUG Chennai (India) JUG-Cologne (Germany) JUG Dortmund (Germany)

Detroit JUG (USA) Duchess (Women) ESPRIT Tunisian JUG (Tunisia) FASOJUG (Burkina Faso) Guatemala Java User Group Green Tea JUG (China) Houston JUG (USA) iJUG e.V. (Germany) IndiJava (India) ITP\_JAVA (Peru) Istanbul JUG

Abdijan JUG (Ivory Coast) Alpes JUG (France)

Austin JUG (USA)

BeJUG (Belgium)

BreizhJUG (Brittany)

CEJUG (Brazil)

Central Ohio JUG (USA)

Chicago JUG

ChinaNanjingJUG

Connecticut JUG (USA)





#### 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



#### JSR example page

) ( jcp.org/en/jsr/detail?id=	310	t <u>∩</u> ~ C	Contraction of the second s		Q 🕈 🖸		
Java Java Community Process	Community I	Development of Java Technology Specificat		Room   Get Java Here   Search JSRs			
SRs	JSR Community Expert Group	union of S					
earch JSRs 🥏	Summary   Proposal   Detail (Summary & Proposal)  JSRs: Java Specification Requests JSR 310: Date and Time API						
JSRs by Platform							
JSRs by Technology	Stage	Access	Start	Finish			
JSRs by Stage	Early Draft Review 2	Download page	14 Sep. 2012	14 Oct, 2012			
JSRs by Committee	Early Draft Review	Download page	26 Feb, 2010	28 Mar, 2010			
List of All JSRs	Expert Group Formation		13 Feb, 2007				
y JCP	JSR Review Ballot	View results	30 Jan, 2007	12 Feb, 2007			
iser ID: username	Status: Active JCP version in use: 2.9 Java Specification Participation Agreement ver	sion in use: 2.0					
legister for Site laving problems logging in?	Description: This JSR will provide a new and improved date an	d time API for Java.					
CP Info	Expert Group Transparency:						
About JCP .	Public Communications Issue Tracking						
Get Involved							
Community Resources +	Team						
Community News >	Specification Leads						

List of Active JSRs (posting in last 12 months): <u>http://jcp.org/en/jsr/all?status=Active&activeMonths=12</u>

va

**Community Process** 

a

ORACLE'

#### 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

000		[#ADOPT/	JSR-2] Javadoc	7 help for JSF	- Java.net JIRA				
M ljClarity M Search r	JClarity   🛄 Google G	🛃 jPDM	Scratch	Scratch I	Google A	Class G	Class Lo	Adopt a	₩ [#AD ×
Java.net/Jira/browse/ADOF	PTAJSR-2					11 - C	(B + macports re	lease notes	Q) (m) (E)
X JIRA							karianna	- Quick Searc	h
	ssues - Agile - Admin	nistration   -							+ Create Issue
ssue Details (XML   Word   Printable)									
Key: ADOPTAJSR-2 Sype: Task	adoptaisr Javadoc 7 help for JSF Created: 12/Oct/12 06:16 PM Updated: 14/Nov/12 10:29 PM						<pre>ssue 1 of 1 issue(s) &lt;&lt; Previous   ADOPTAJSR-2   Next &gt;&gt;</pre>		
Status: 🥠 Open Priority: 🗣 Major Assignee: <u>Thomas Modeneis</u> Reporter: <u>karianna</u>	Component/s:     JSR-344 (JSF 2.2)       Affects Version/s:     None       Fix Version/s:     None								
/otes: <u>1</u> Vatchers: <u>0</u>	Time Tracking:	Not Specified							
Available Workflow Actions	Tags:	javadoc EDIT							
Resolve Issue	Participants:		nna and Thomas	Modeneis					
Operations									
Go to Planning Board	Description								
Assign this issue ( <u>to me</u> ) Clone this issue Comment on this issue	I sure could use some help i The task would be very hand Ed							-a-JSR program he	elp me out here?
Create <u>sub-task</u>	[1] https://maven.java.net/service/local/repositories/snapshots/archive/javax/faces/javax.faces-api/2.2-SNAPSHOT/javax.faces-api-2.2-20121011.174655-72-javadoc.jar ///javadocs/index.html								
Edit this issue	This is not a permalink so it will probably not resolve once a new nightly gets published. If it does not resolve, just go to http://maven.java.net/ <sup>®</sup> and put in javax.faces:javax.faces-api:2.2-SNAPSHOT into the search box, select the -javadoc.jar, and view it in the archive browser. Instructions are here:								
Move this issue	http://weblogs.java.net/blog/	/edburns/archive/2012/	05/17/how-view-a	absolute-latest-j	sf-22-spec-snaps	shot			

#### ORACLE

#### 5) Participate in Hack Days & Workshops – Have Fun!









## **Early Access**

 In adoption group, you can discuss Early Access Builds – provide comments

#### jdk.java.net

**GA Releases** JDK 11 Early-Access Releases IDK 13 **IDK 12 Jpackage Open/FX** Panama Valhalla IMC Reference Implementations lava SE 12 lava SE 11 lava SE 10 Java SE 9 Java SE 8 Java SE 7 Feedback Report a bug

Archive

JDK 13 Early-Access Builds

Schedule, status, & features (OpenJDK)

#### Documentation

- Release notes
- Test results
- API Javadoc

#### Latest build: 11 (2019/3/7)

- Changes in this build
- Issues addressed in this build

#### Builds

These early-access, open-source builds are provided under the GNU General Public License, version 2, with the Classpath Exception.

Linux/x64	tar.gz (sha256)	194685134 bytes
macOS/x64	tar.gz (sha256)	187112270
Windows/x64	zip (sha256)	193205088
Alpine Linux/x64	tar.gz (sha256)	197120161





## **Participate in OpenJDK**

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







#### 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: <u>http://JCP.org</u>
  - -Join the JCP: <a href="https://jcp.org/en/participation/membership">https://jcp.org/en/participation/membership</a>
  - -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 , **நன்றி**, tak, tack, grazie, gracias, danke, Благодаря, teşekkür ederim merci, obrigado, bedankt, kiitos, xie xie,ありがとう.

**Questions?** 

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









Confidential – Oracle Internal/Res