The Conference Sep 18 - 22

- Friday, Sep 16, Java Champion Summit at Oracle HQ
- Saturday, Sep 17, JavaOne4Kids, Moscone North
- Sunday, Sep 18, Sessions begin at 18 and the Keynote at 1:30, Moscone North
- Monday - Thursday, Sessions at Hilton Union Square and Parc 55
  - Community Keynote at Moscone West
Java Champion Summit
JC Summit Agenda

- Netbeans Update - Handed over to the Apache Foundation
  [https://netbeans.org/community/apache-incubator.html](https://netbeans.org/community/apache-incubator.html)
- JDK 9 (mid 2017)
- Java EE 8 (end of 2017) and Java EE 9 (end of 2018)
- Cloud PaaS
- Oracle IaaS
- Oracle Cloud
- Java Community Process
## JDK Enhancement Proposals (JEPs) for JDK 9

### New standards
- Process API Updates
- Variable Handles
- Spin-Wait Hints
- Dynamic Linking of Language-Defined Object Models
- Enhanced Method Handles
- More Concurrency Updates
- Compiler Control

### Specialized
- HTTP 2 Client
- Unicode 8.0
- UTF-8 Property Files
- Datagram Transport Layer Security (DTLS)
- OCSP Stapling for TLS
- TLS Application-Layer Protocol Negotiation Extension
- SHA-3 Hash Algorithms
- DRBG-Based SecureRandom Implementations
- Create PKCS12 keystores by Default
- Merge Selected Xerces 2.11.0 Updates into JAXP
- XML Catalogs
- HarfBuzz Font-Layout Engine
- HTML5 Javadoce

### Housekeeping
- Parser API for Nashorn
- Prepare JavaFX UI Controls & CSS APIs for Modularization
- Modular Java Application Packaging
- New Version-String Scheme
- Reserved Stack Areas for Critical Sections
- Segmented Code Cache
- Indify String Concatenation
- Unified JVM Logging
- Unified GC Logging
- Make G1 the Default Garbage Collector
- Use CLDR Locale Data by Default
- Validate JVM Command-Line Flag Arguments
- Java-Level JVM Compiler Interface
- Disable SHA-1 Certificates
- Simplified Doclet API
- Deprecated the Applet API
- Process Import Statements Correctly
- Annotations Pipeline 2.0
- Elide Deprecation Warnings on Import Statements
- Milling Project Coin
- Filter Incoming Serialization Data

### Gone
- Remove GC Combinations Deprecated in JDK 8
- Remove Launch-Time JRE Version Selection
- Remove the JVM TI hprof Agent
- Remove the jhat Tool

### Behind the scenes
- Store Interned Strings in CDS Archives
- Improve Contended Locking
- Compact Strings
- Improve Secure Application Performance
- Leverage CPU Instructions for GHASH and RSA
- Tiered Attribution for javac
- Javadoc Search
- Marlin Graphics Renderer
- HiDPI Graphics on Windows and Linux
- Enable GTK 3 on Linux
- Update JavaFX/JavaFX to Newer Version of GSreamer

### New functionality
- Module System
- The Modular JDK
- Modular Source Code
- Modular Run-Time Images
- Encapsulate Most Internal APIs
- Jlink: The Java Linker
- Enhanced Deprecation
- Stack-Walking API
- Convenience Factory Methods for Collections
- Platform Logging API and Service
- jshell: The Java Shell (Read-Eval-Print Loop)
- Compile for Older Platform Versions
- Multi-Release JAR Files
- Platform-Specific Desktop Features
- TIFF Image I/O
- Multi-Resolution Images

### Specialized
- HTTP 2 Client
- Unicode 8.0
- UTF-8 Property Files
- Datagram Transport Layer Security (DTLS)
- OCSP Stapling for TLS
- TLS Application-Layer Protocol Negotiation Extension
- SHA-3 Hash Algorithms
- DRBG-Based SecureRandom Implementations
- Create PKCS12 keystores by Default
- Merge Selected Xerces 2.11.0 Updates into JAXP
- XML Catalogs
- HarfBuzz Font-Layout Engine
- HTML5 Javadoce
Project Jigsaw
Modularize the Java Platform

• JEP 261: Module System
• JEP 200: The Modular JDK
• JEP 201: Modular Source Code
• JEP 220: Modular Run-Time Images
• Plus
  – JEP 260: Encapsulate Most Internal APIs
  – JEP 282: jlink: The Java Linker
JEP 214: Remove GC Combinations Deprecated in JDK 8

hotspot / gc

- Remove the GC combinations that were previously deprecated in JDK 8 via JEP 173

```java
DefNew + CMS : -XX:-UseParNewGC -XX:+UseConcMarkSweepGC
ParNew + SerialOld : -XX:+UseParNewGC
ParNew + iCMS : -Xincgc
ParNew + iCMS : -XX:+CMSGenIncrementalMode -XX:+UseConcMarkSweepGC
DefNew + iCMS : -XX:+CMSGenIncrementalMode -XX:+UseConcMarkSweepGC -XX:-UseParNewGC
CMS foreground : -XX:+UseCMSCompactAtFullCollection
CMS foreground : -XX:+CMSFullGCsBeforeCompaction
CMS foreground : -XX:+UseCMSGCwithCompactingCollectionPassing
```
More Info on JEPs

http://openjdk.java.net/projects/jdk9/

JDK 9

The goal of this Project is to produce an open-source reference implementation of the Java SE 9 Platform, to be defined by a forthcoming JSR in the Java Community Process.

The schedule and features of this release are proposed and tracked via the JEP Process, as amended by the JEP 2.0 proposal.

Schedule

- 2016/05/26 Feature Complete
- 2016/08/11 All Tests Run
- 2016/09/01 Rampdown Start
- 2016/10/20 Zero Bug Bounce
- 2016/12/01 Rampdown Phase 2
- 2017/01/26 Final Release Candidate
- 2017/03/23 General Availability

The milestone definitions are the same as those for JDK 8.

Status

We are past the Feature Complete milestone, but that does not mean that the feature set is frozen. If you own a JEP or a small enhancement that is not yet complete then you can request an extension via the FC extension-request process.

Features

JEPs targeted to JDK 9, so far

- 102: Process API Updates
- 110: HTTP 2 Client
- 143: Improve Contended Locking
Java EE 8 and 9

Technical Focus Areas Summary

**Programming Model**
- Extend for reactive programming
- Unified event model
- Event messaging API
- JAX-RS, HTTP/2, Lambda, JSON-B, ...

**Key Value/Doc Store**
- Persistence and query interface for key value and document DB

**Eventual Consistency**
- Automatically event out changes to observed data structures

**Serverless**
- New spec – interfaces, packaging format, manifest
- Ephemeral instantiation

**Configuration**
- Externalize configuration
- Unified API for accessing configuration

**Resiliency**
- Extension to support client-side circuit breakers
- Resilient commands
- Standardize on client-side format for reporting health

**Packaging**
- Package applications, runtimes into services
- Standalone immutable executable binary
- Multi-artifact archives

**Multitenancy**
- Increased density
- Tenant-aware routing and deployment

**State**
- API to store externalized state

**Security**
- Secret management
- OAuth
- OpenID
Java EE 8 (Revised Proposal, 2016)
Java EE Roadmap

2016
- Engage Java EE Community
  - Feedback through Survey
  - Launch Java EE Next JSRs

2017
- Java EE 8
  - Specs, RI, TCK complete
  - Initial microservices support
  - Define Java EE 9
  - Early access implementation of Java EE 9

2018
- Java EE 9
  - Specs, RI, TCK complete
  - Modular Java EE runtime
  - Enhanced microservices support
Java EE - Summary

1. Java EE widely adopted in Cloud
2. New AppDev style for Cloud & Microservices
3. Embrace reactive programming model for distributed systems
4. Extend the benefits of standards and portability
5. Broad community support
Heather VanCura
Java Community Process (JCP) Program

- Director of the JCP Program Office
- Leader of Global Adopt-a-JSR Programs
- Open Source Fan
- Native Californian - from San Diego
- Personal Interests: Fun 😊

Bay Area, California - USA
Java Community Process (JCP) Program

- Share ideas and feedback, comment on list and public issue trackers.
- Read early versions of specifications and Javadocs.
- 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.
- https://jcp.org/en/jsr/stage?listBy=active
- https://jcp.org/en/home/index
- https://jcp.org/en/participation/membership
Influencing the Future of Java
Our Adopt-A-JSR Story

Bob Paulin
President of the Chicago JUG

Mercedes Wyss
Board Member of the Guatemala JUG
Leader of the Guatemala JDuchess Chapter

Java
Your
(Next)
JavaOne4Kids
<table>
<thead>
<tr>
<th>Time</th>
<th>Room 301 Cap 50</th>
<th>Room 302 Cap 50</th>
<th>Room 303 Cap 50</th>
<th>Room 304 Cap 50</th>
<th>Room 305 Cap 100</th>
<th>Room 306 Cap 100</th>
<th>Room 307 Cap 50</th>
</tr>
</thead>
<tbody>
<tr>
<td>8:00 AM - 9:00 AM</td>
<td>Registration</td>
<td>Registration</td>
<td>Registration</td>
<td>Registration</td>
<td>Registration</td>
<td>Registration</td>
<td>Registration</td>
</tr>
<tr>
<td>9:00 AM - 9:30 AM</td>
<td>8:00 AM - 9:00 AM</td>
<td>9:00 AM - 9:30 AM</td>
<td>9:45 AM - 11:45 AM</td>
<td>11:45 AM - 12:45 AM</td>
<td>12:45 PM - 2:45 PM</td>
<td>2:45 PM - 3:00 PM</td>
<td>3:00 PM - 5:00 PM</td>
</tr>
<tr>
<td></td>
<td>Room 301 Cap 50</td>
<td>Room 302 Cap 50</td>
<td>Room 303 Cap 50</td>
<td>Room 304 Cap 50</td>
<td>Room 305 Cap 100</td>
<td>Room 306 Cap 100</td>
<td>Room 307 Cap 50</td>
</tr>
<tr>
<td>9:45 AM - 11:45 AM</td>
<td>Build Your Own Website</td>
<td>Defeat the Zombies with Java</td>
<td>Introduction to Arduino</td>
<td>HTML 5, JavaScript, and Game Development</td>
<td>Make your own game with Greenfoot</td>
<td>Game Design From Scratch</td>
<td>Programming The Finch Robot in Greenfoot</td>
</tr>
<tr>
<td></td>
<td>Kevin Nilson</td>
<td>Cassandra Chin</td>
<td>Pradeep Bhattar</td>
<td>(KID7771)</td>
<td>(KID7772)</td>
<td>(KID7773)</td>
<td>(KID7775)</td>
</tr>
<tr>
<td></td>
<td>(KID7770)</td>
<td>(KID7771)</td>
<td>(KID7772)</td>
<td>(KID7773)</td>
<td>(KID7774)</td>
<td>(KID7775)</td>
<td>(KID7776)</td>
</tr>
<tr>
<td></td>
<td>Lunch</td>
<td>Lunch</td>
<td>Lunch</td>
<td>Lunch</td>
<td>Lunch</td>
<td>Lunch</td>
<td>Lunch</td>
</tr>
<tr>
<td></td>
<td>12:45 PM - 2:45 PM</td>
<td>12:45 PM - 2:45 PM</td>
<td>12:45 PM - 2:45 PM</td>
<td>12:45 PM - 2:45 PM</td>
<td>12:45 PM - 2:45 PM</td>
<td>12:45 PM - 2:45 PM</td>
<td>12:45 PM - 2:45 PM</td>
</tr>
<tr>
<td></td>
<td>Introduction to Python Level 1</td>
<td>Bringing Robot online with Robo4j Framework</td>
<td>Let's Play with a NAO Robot</td>
<td>Minecraft Modding using Forge 1.8</td>
<td>Java for Elementary and Middle School Using BlueJ</td>
<td>Use Watson to learn about your personality</td>
<td>Arcade</td>
</tr>
<tr>
<td></td>
<td>Peter Le</td>
<td>Miroslav Kopecky &amp; Nikhil Naniwadekar (KID7778)</td>
<td>Daniel De Luca</td>
<td>Aditya Gupta</td>
<td>G Venkat</td>
<td>Sandhya Kapoor</td>
<td>William McCabe (KID7783)</td>
</tr>
<tr>
<td></td>
<td>(KID7777)</td>
<td>(KID7778)</td>
<td>(KID7779)</td>
<td>(KID7782)</td>
<td>(KID7781)</td>
<td>(KID7780)</td>
<td>(KID7783)</td>
</tr>
<tr>
<td></td>
<td>Break</td>
<td>Break</td>
<td>Break</td>
<td>Break</td>
<td>Break</td>
<td>Break</td>
<td>Break</td>
</tr>
<tr>
<td></td>
<td>2:45 PM - 3:00 PM</td>
<td>2:45 PM - 3:00 PM</td>
<td>2:45 PM - 3:00 PM</td>
<td>2:45 PM - 3:00 PM</td>
<td>2:45 PM - 3:00 PM</td>
<td>2:45 PM - 3:00 PM</td>
<td>2:45 PM - 3:00 PM</td>
</tr>
<tr>
<td></td>
<td>How to send a secret message</td>
<td>Squash the Bugs!</td>
<td>Star Wars: The Programmer Awakens</td>
<td>Programming The Finch Robot in Java</td>
<td>Beginner Android programming for fun and games</td>
<td>Exploring JavaScript Venkat Subramaniam</td>
<td>Arcade</td>
</tr>
<tr>
<td></td>
<td>Brendanan Considine &amp; Nikhil Naniwadekar (KID7786)</td>
<td>David Taeb (KID7787)</td>
<td>Aditya Gupta (KID7788)</td>
<td>Bruce Regittko (KID7789)</td>
<td>Suyash Joshi (KID7790)</td>
<td>Venkat Subramaniam (KID7791)</td>
<td>William McCabe (KID7783)</td>
</tr>
<tr>
<td></td>
<td>(KID7787)</td>
<td>(KID7788)</td>
<td>(KID7789)</td>
<td>(KID7790)</td>
<td>(KID7791)</td>
<td>(KID7790)</td>
<td>(KID7791)</td>
</tr>
</tbody>
</table>

Copyright © 2016, Oracle and/or its affiliates. All rights reserved.
# JavaOne - The Conference

<table>
<thead>
<tr>
<th>Sessions and Tracks</th>
<th>Sunday</th>
<th>Monday</th>
<th>Tuesday</th>
<th>Wednesday</th>
<th>Thursday</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core Java Platform</td>
<td>12</td>
<td>23</td>
<td>25</td>
<td>24</td>
<td>16</td>
<td>100</td>
</tr>
<tr>
<td>Emerging Languages</td>
<td>0</td>
<td>12</td>
<td>9</td>
<td>13</td>
<td>6</td>
<td>40</td>
</tr>
<tr>
<td>Java and Devices</td>
<td>3</td>
<td>11</td>
<td>11</td>
<td>9</td>
<td>5</td>
<td>39</td>
</tr>
<tr>
<td>Java, Big Data, and the Cloud</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Java Clients and User Interfaces</td>
<td>0</td>
<td>14</td>
<td>15</td>
<td>12</td>
<td>3</td>
<td>44</td>
</tr>
<tr>
<td>Java, Cloud, and Server Side Development</td>
<td>3</td>
<td>33</td>
<td>28</td>
<td>27</td>
<td>17</td>
<td>108</td>
</tr>
<tr>
<td>Java Development Tools</td>
<td>5</td>
<td>23</td>
<td>24</td>
<td>17</td>
<td>11</td>
<td>80</td>
</tr>
<tr>
<td>Java, DevOps, and Methodologies</td>
<td>1</td>
<td>11</td>
<td>19</td>
<td>9</td>
<td>5</td>
<td>45</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>25</td>
<td>128</td>
<td>131</td>
<td>112</td>
<td>63</td>
<td>459</td>
</tr>
</tbody>
</table>

444 Total Unique Sessions (439 last year)
Resources

YouTube - 85 Videos - JavaOne 2016 Playlist:
https://www.youtube.com/playlist?list=PLPIzp-E1msrYicmovyeyuOABO4HxVPlhEA

JavaOne Session List - search JavaOne 2016 - Session Catalog:
Sessions
Sessions

Prepare for JDK 9

Alan Bateman
Java Platform Group, Oracle
September 2016
Sessions

More Information

OpenJDK Project Jigsaw page, this has links to all the JEPs
http://openjdk.java.net/projects/jigsaw/
mailto:jigsaw-dev@openjdk.java.net

Early Access Builds
https://jdk9.java.net/download

Java Dependency Analysis Tool
https://wiki.openjdk.java.net/display/JDK8/Java+Dependency+Analysis+Tool

JEP 223: New Version-String Scheme
http://openjdk.java.net/jeps/223

JEP 238: Multi-Release JAR files
http://openjdk.java.net/jeps/238
Sessions

JUnit 5
Features, Architecture, and Extensibility

Steve Moyer
Steve Seltzer
Niraja Ramesh

The Pennsylvania State University
Sessions

How to Initiate Kids Into Java Programming

Félix Roberge
@felixroberge
Sessions

Metrics-Driven Continuous Delivery

Stop Bad Code and Bad Ideas Early!

Andreas Grabner: @grabnerandi, andreas.grabner@dynatrace.com
Slides: http://www.slideshare.net/grabnerandi
Podcast: https://www.spreaker.com/show/pureperformance
Sessions

Arun Gupta
Theresa Nguyen
Roy van Rijn
Siamak Ashrafi
Vinicius Senger
Ray Tsang
Tori Wieldt
Reza Rahman
Tim Berglund
Stephen Chin
Bert Jan Schrijver
Sessions
Sessions

http://trishagee.github.io/presentation/refactoring_to_java_8/
Sessions
Questions?

Thank You!