It's a JDK jungle out there –
oder wie es mit den neuen Java-Versionen geordnet weitergeht

CREATE THE FUTURE

Oracle JDK und OpenJDK

Wolfgang Weigend
Sen. Leitender Systemberater
Java Technology and Architecture

Java™ ORACLE

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.

Agenda

1. Java Version Numbers
2. Java SE Roadmap
3. Java in a World of Containers and Open Source
4. JDK 11 Migration Guide and Features
5. OpenJDK
6. Release Cycle & Oracle OpenJDK Builds
7. Summary

JDK Version Numbers and Java Critical Patch Updates

- Rules for Java CPU's
  - Main release for security vulnerabilities
  - Covers all JDK families (11, 8, 7, 6)
  - CPU release triggers Auto-update
  - Dates published 12 months in advance
  - Security Alerts are released as necessary
  - Based off the previous (non-CPU) release
  - Released simultaneously on java.com and OTN

- JDK 11.0.1 - Security Baselines

JRE Family Version	JRE Security Baseline (Full Version String)
10	10.0.99
9	9.0.99
8	1.8.0_191-b12
7	1.7.0_201-b11
6	1.6.0_211-b11

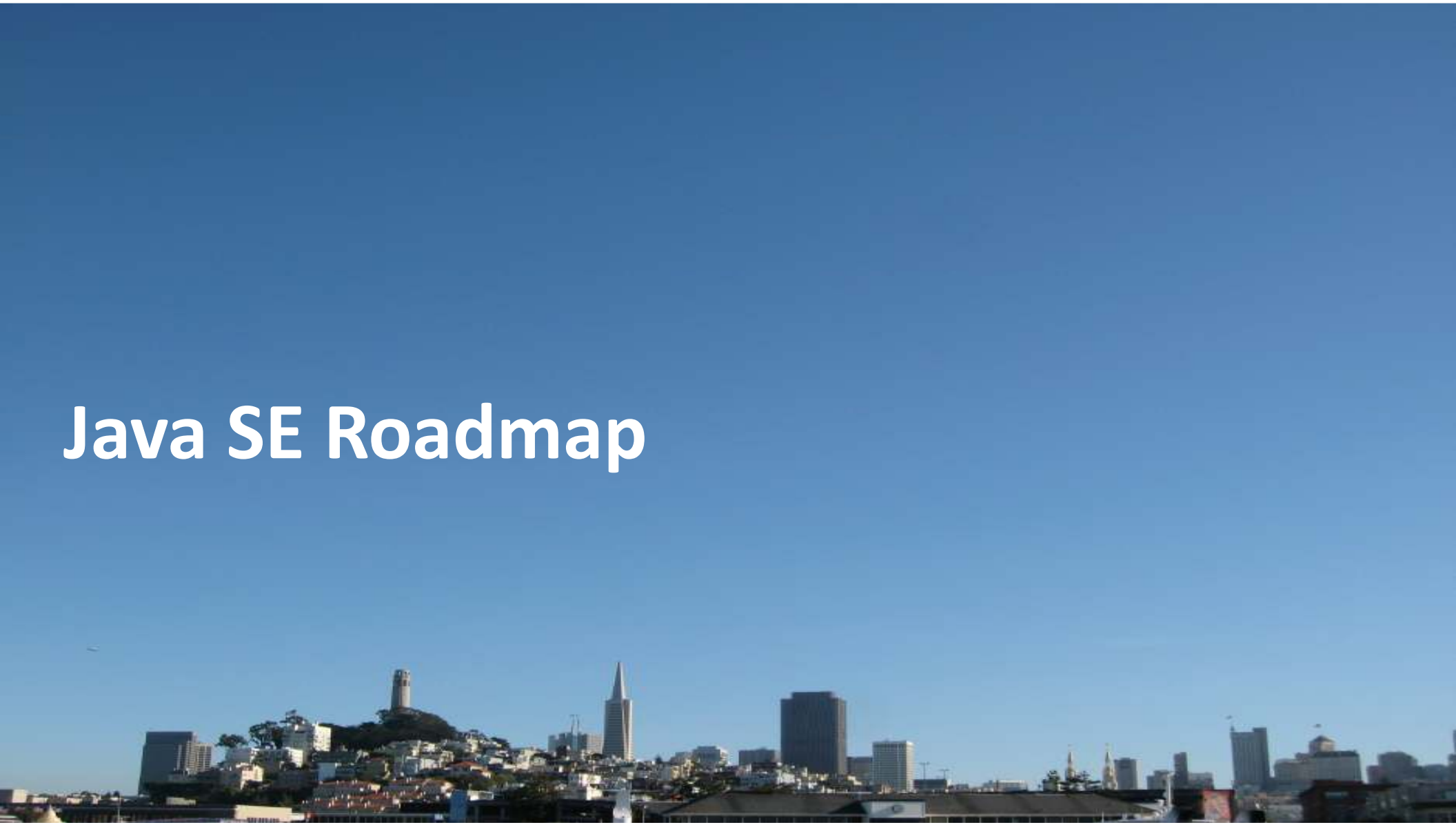
JDK Version Numbers

- A version number, $VNUM, is a sequence of numerals of arbitrary length, separated by period characters.

The first four numerals are interpreted as follows: **$FEATURE.$INTERIM.$UPDATE.$EMERG**

- **$FEATURE** **formerly $MAJOR**
- **$INTERIM** **formerly $MINOR**
- **$UPDATE** **formerly $SECURITY**
- **$EMERG** -- The emergency-release counter, incremented only when it's necessary to produce an emergency release to fix an urgent security issue. Using an additional numeral for this purpose minimizes the disruption to both developers and users of in-flight update releases.

Java SE Roadmap



Java SE Public Updates

http://www.oracle.com/technetwork/java/eol-135779.html

Java SE Public Updates				
Release	**GA Date**	**End of Public Updates Notification**	**Commercial User End of Public Updates**	**Personal User End of Public Updates**
7	July 2011	March 2014	April 2015	
8	March 2014	September 2017	January 2019****	December 2020****
9 (non-LTS)	September 2017	September 2017	March 2018	
10 (18.3^) (non-LTS)	March 2018	March 2018	September 2018	
11 and later		No longer Applicable		

End of Public Updates

Oracle will make available to Commercial Users and Personal Users updates to publicly available versions of Oracle Java SE in accordance with the table below. Once a Java SE version reaches "End of Public Updates", any further updates will be available only to Customers and accessible through My Oracle Support and via corporate auto update where applicable (Visit My.Oracle Support Note 1439822.1 - All Java SE Downloads on MOS – Requires Support Login).

Oracle does not plan to migrate desktops from Java SE 8 to later versions via the auto update feature. This includes the Java Plugin and Java Web Start. Instead of relying on a browser-accessible system JRE, we encourage application developers to use the packaging options introduced with Java SE 9 to repackage and deliver their Java applications as stand-alone applications that include their own custom runtimes.

Oracle will continue to provide Public Updates and auto updates of Java SE 8, until at least the end of December 2020 for Personal Users, and January 2019 for Commercial Users.

Oracle Java SE Support Roadmap*

http://www.oracle.com/technetwork/java/eol-135779.html

Oracle Java SE Support Roadmap*†

Release	GA Date	Premier Support Until** Notification	Extended Support Until**	Sustaining Support**
6	December 2006	December 2015	December 2018	Indefinite
7	July 2011	July 2019	July 2022	Indefinite
8	March 2014	March 2022	March 2025	Indefinite
9 (non-LTS)	September 2017	March 2018	Not Available	Indefinite
10 (18.3^) (non-LTS)	March 2018	September 2018	Not Available	Indefinite
11 (18.9^ LTS)	September 2018***	September 2023	September 2026	Indefinite
12 (19.3^ non-LTS)	March 2019***	September 2019	Not Available	Indefinite

Oracle Java SE Product Releases

Oracle provides Customers with Oracle Premier Support on Oracle Java SE products as described in the Oracle Lifetime Support Policy. For product releases after Java SE 8, Oracle will designate a release, every three years, as a Long-Term-Support (LTS) release. Java SE 11 (18.9 LTS) is the next planned LTS release. For the purposes of Oracle Premier Support, non-LTS releases are considered a cumulative set of implementation enhancements of the most recent LTS release. Once a new feature release is made available, any previous non-LTS release will be considered superseded. For example, Java SE 9 was a non-LTS release and immediately superseded by Java SE 10 (also non-LTS), Java SE 10 in turn is immediately superseded by Java SE 11. Java SE 11 however is an LTS release, and therefore Oracle Customers will receive Oracle Premier Support and periodic update releases, even after Java SE 12 is released.

† Excluding Deployment Technology and JavaFX

Oracle Java SE Support Roadmap*

http://www.oracle.com/technetwork/java/eol-135779.html

Oracle Java SE Support Roadmap^†				
Release	**GA Date**	**Premier Support Until** Notification**	**Extended Support Until****	**Sustaining Support****
6	December 2006	December 2015	December 2018	Indefinite
7	July 2011	July 2019	July 2022	Indefinite
8	March 2014	March 2022	March 2025	Indefinite
9 (non-LTS)	September 2017	March 2018	Not Available	Indefinite
10 (18.3^) (non-LTS)	March 2018	September 2018	Not Available	Indefinite
11 (18.9^ LTS)	September 2018***	September 2023	September 2026	Indefinite
12 (19.3^ non-LTS)	March 2019***	September 2019	Not Available	Indefinite

- **Das Oracle JDK 11 darf nur mit der Java SE Subscription produktiv eingesetzt werden, ansonsten ist es frei einsetzbar nur für Entwicklung, Testbetrieb, Prototyping und für Demozwecke:**

- Oracle JDK and OpenJDK builds from Oracle
  Starting with Java SE 9, in addition to providing Oracle JDK for free under the BCL, Oracle also started providing builds of OpenJDK under an open source license (similar to that of Linux). Oracle is working to make the Oracle JDK and OpenJDK builds from Oracle interchangeable - targeting developers and organizations that do not want commercial support or enterprise management tools. **Beginning with Oracle Java SE 11 (18.9 LTS), the Oracle JDK will continue to be available royalty-free for development, testing, prototyping or demonstrating purposes.** As announced in September 2017, with the OracleJDK and builds of Oracle OpenJDK being interchangeable for releases of Java SE 11 and later, **the Oracle JDK will primarily be for commercial and support customers** and OpenJDK builds from Oracle are for those who do not want commercial support or enterprise management tools.

- **Wer das Oracle JDK nicht einsetzen möchte, kann das äquivalente Oracle OpenJDK verwenden:**

- Java could be used without the Oracle Java SE Subscription with Oracle OpenJDK as an equivalent to the Oracle JDK. The OpenJDK release cadence is about two OpenJDK major versions per year. There would be almost no difference between Oracle JDK and Oracle OpenJDK but the OpenJDK comes without Java engineering support agreement for customers.

Oracle Java SE Support for Deployment Technology & JavaFX

http://www.oracle.com/technetwork/java/eol-135779.html

Support for Deployment Technology and JavaFX*				
Version	**GA Date**	**Java Web Start Support Until**	**Java Plugin (Applets) Support Until**	**JavaFX Support Until**
6	December 2006	October 2017	October 2017	N/A
7	July 2011	October 2017	October 2017	July 2019
8	March 2014	March 2025****	March 2019	March 2022
9 and later		N/A	N/A	N/A

* Oracle Java SE product dates are provided as **examples** to illustrate the support policies. Customers should refer to the Oracle Lifetime Support Policy for the most up-to-date information. Timelines may differ for Oracle Products with a Java SE dependency (My Oracle Support Note 1557737.1 - Support Entitlement for Java SE When Used As Part of Another Oracle Product – Requires Support Login).

** These support timelines apply to Java client and server deployments of Java with the exception of Web Deployment Technology and JavaFX. For more information on those features, see below.

*** LTS designation and dates, as noted in the above example, are subject to change.

**** Or later.

^ Starting with Java SE 10 (18.3), in March 2018, Oracle JDK includes, in the Java SE vendor version string, the year and month of the release as "YY.M".

† Excluding Deployment Technology and JavaFX, which is described in a separate section.

Web Deployment Technology and JavaFX

The Web Deployment Technology bundled with the JRE, consisting of the Java Plugin and Java Web Start has a shorter support lifecycle: only five years of Premier Support. The deployment stack was marked as deprecated and flagged for removal in Java SE 9 and Java SE 10. Oracle Java SE 11 and later versions will not include the Deployment Stack. As Java SE 8 will be the sunset release for the Deployment Stack (given that Java SE 9 and Java SE 10 are non-LTS releases) Oracle extended support of Java Web Start on Java SE 8 until the end of Java SE 8 Extended Support (March 2025). Support for the Java Plugin (Java Applets) remains available until March 2019.

Java SE 8 is the recommended and only supported version of the deployment stack, since versions earlier than Java SE 8 no longer include the deployment stack, while Java SE 9 has reached end of extended support, and Java SE 10 will reach end of extended support in September 2018. The Java SE 8 deployment stack may be used to run Java SE 6, or Java SE 7 applications on Windows platforms. The Java deployment technology will not be supported beyond Java SE 8. See the Oracle Lifetime Support Policy for details.

JavaFX has been Open Sourced and redesigned to be available as a stand-alone library rather than being included with the JDK. Starting with Java SE 11 (18.9 LTS), JavaFX will not be included in the Oracle JDK. Support for JavaFX on Java SE 8 will continue through the Premier Support term (until March 2022).

Oracle Java SE Subscription

- **Oracle Java SE Subscription for Desktops, Servers or Cloud deployments**

  Java SE Subscription is a simple, low-cost monthly subscription that includes Java SE Licensing and Support for use on Desktops, Servers or Cloud deployments, It follows a commonly used model, popular with Linux distributions. The subscription provides access to tested and certified performance, stability, and security updates for Java SE, directly from Oracle. It also includes access to My Oracle Support (MOS) 24x7, support in 27 languages, Java SE 8 Desktop management, monitoring, and deployment features, among other benefits.

- **Java SE Subscription cost**

  Desktop pricing is $2.50 per user per month, or lower with tiered volume discounts. Processor pricing for use on Servers and/or Cloud deployments is $25.00 per month or lower.

- **Java SE Subscription What is the length of terms**

  Standard term is one year, with two and three-year terms available.

- **Java SE Subscription support updates**

  Comprehensive Java SE Support is central to the Java SE Subscription and is provided via My Oracle Support (MOS)

- **Calculator**
  - Metric: Named User Plus, or Processor
  - Term: 1 Year .. 3 Years
  - Quantity: Anzahl

http://www.oracle.com/technetwork/java/javaseproducts/overview/javasesubscriptionfaq-4891443.html

Oracle Java SE Subscription Global Price List

Prices in USA (Dollar)

	Monthly Subscription Price	Subscription Metric	Volume
Java SE Platform Products			
Java SE Desktop Subscription	2.5000	Named User Plus	1-999
	2.0000	Named User Plus	1,000-2,999
	1.7500	Named User Plus	3,000-9,999
	1.5000	Named User Plus	10,000-19,999
	1.2500	Named User Plus	20,000-49,999
		Contact for Details	50,000+
Java SE Subscription	25.0000	Processor	1-99
	23.7500	Processor	100-249
	22.5000	Processor	250-499
	20.0000	Processor	500-999
	17.5000	Processor	1,000-2,999
	15.0000	Processor	3,000-9,999
	12.5000	Processor	10,000-19,999
		Contact for Details	20,000+

http://www.oracle.com/us/corporate/pricing/price-lists/java-se-subscription-pricelist-5028356.pdf

Java Client Roadmap Update (1)
Java Web Start and Pre-Installed Java Runtime Environments

- Java Web Start has been included in the Oracle Java Runtime Environment (JRE) since 2001. It is launched automatically when a Java application using Java Web Start technology is downloaded for the first time. Desktop shortcuts can also launch the application, providing the user with a similar experience to that of a native application.
- Java Web Start has become a migration path for developers as browser vendors continued to restrict plugin support over the past several years.
- Since it is predominantly a desktop technology, Web Start has some limitations. In particular, it requires a standalone JRE to be installed and maintained on the user's desktop.
- However, over the past decade, vendors of the most popular desktop operating systems have emphatically pushed for applications on their platforms to be delivered bundled with integrated, sandboxed runtimes. Increasingly they require desktop applications to be distributed through their own private "app stores."
- The notion of an application being distributed separately from a standalone JRE is, therefore, quickly fading.

Java Client Roadmap Update (2)

Java Web Start and Pre-Installed Java Runtime Environments - Consequently

- Oracle will extend support for Web Start in Java SE 8 from March, 2019, through at least March 2025.
- **Oracle products that have dependencies on Web Start will remain on Java SE 8 and continue with the support timelines as indicated by those products.**
- Oracle will not include Java Web Start in Java SE 11 (18.9 LTS) and later.
- Oracle will begin encouraging application developers and users to transition away from Java Web Start and encourage non-commercial consumers to remove any unused or non-supported Oracle JRE installations from their desktops.
- Developers who deploy desktop applications to individual consumers (eg, games, personal banking, or other B2C applications) will need to transition to other deployment technologies such as the jlink and/or third party packaging and deployment solutions before the end of 2020.
- Application developers who target applications for internal data processing, business, commercial, or production purposes, will either need to seek commercial license with Oracle, or transition to other deployment technologies by January 2019.

Oracle JDK Releases for Java 11 and later

Insights and updates on Java SE and OpenJDK from the Java Platform Group Product Management Team

Oracle JDK Releases for Java 11 and later (1)

Exec Summary

- Starting with Java 11, Oracle will provide JDK releases under the open source GNU General Public License v2, with the Classpath Exception (GPLv2+CPE), and under a commercial license for those using the Oracle JDK as part of an Oracle product or service, or who do not wish to use open source software. This combination of using an open source license and a commercial license replaces the historical "BCL" license, which had a combination of free and paid commercial terms.
- Different builds will be provided for each license, but these builds are functionally identical aside from some cosmetic and packaging differences, described in detail below.

Oracle JDK Releases for Java 11 and later (2)

From the BCL to the GPL

- The Binary Code License for Oracle Java SE technologies ("BCL") has been the primary license for Oracle Java SE technologies for well over a decade. The BCL permits use without license fees under certain conditions. To simplify things going forward, Oracle started providing open source licensed OpenJDK builds as of Java 9, using the same license model as the Linux platform.
- If you are used to getting Oracle Java SE binaries for free, you can simply continue doing so with Oracle's OpenJDK builds available at jdk.java.net.
- If you are used to getting Oracle Java SE binaries as part of a commercial product or service from Oracle, then you can continue to get Oracle JDK releases through My Oracle Support (MOS), and other locations.

Oracle JDK Releases for Java 11 and later (3)

Functionally identical and interchangeable ...

- Oracle's BCL-licensed JDK historically contained "commercial features" that were not available in OpenJDK builds. As promised, however, over the past year Oracle has contributed these features to the OpenJDK Community, including:
  - Java Flight Recorder,
  - Java Mission Control,
  - Application Class-Data Sharing, and
  - ZGC.
- From Java 11 forward, therefore, Oracle JDK builds and OpenJDK builds will be essentially identical.

Oracle JDK Releases for Java 11 and later (4)

... yet with some cosmetic and packaging differences

- There do remain a small number of differences, some intentional and cosmetic, and some simply because more time to discuss with OpenJDK contributors is warranted.
  - Oracle JDK 11 emits a warning when using the -XX:+UnlockCommercialFeatures option, whereas in OpenJDK builds this option results in an error. This option was never part of OpenJDK and it would not make sense to add it now, since there are no commercial features in OpenJDK. This difference remains in order to make it easier for users of Oracle JDK 10 and earlier releases to migrate to Oracle JDK 11 and later.
  - Oracle JDK 11 can be configured to provide usage log data to the "Advanced Management Console" tool, which is a separate commercial Oracle product. We will work with other OpenJDK contributors to discuss how such usage data may be useful in OpenJDK in future releases, if at all. This difference remains primarily to provide a consistent experience to Oracle customers until such decisions are made.

Oracle JDK Releases for Java 11 and later (5)

... yet with some cosmetic and packaging differences

- The ***`javac --release`*** command behaves differently for the Java 9 and Java 10 targets, since in those releases the Oracle JDK contained some additional modules that were not part of corresponding OpenJDK releases:
  - javafx.base
  - javafx.controls
  - javafx.fxml
  - javafx.graphics
  - javafx.media
  - javafx.web
  - java.jnlp
  - jdk.jfr
  - jdk.management.cmm
  - jdk.management.jfr
  - jdk.management.resource
  - jdk.packager.services
  - jdk.snmp
- This difference remains in order to provide a consistent experience for specific kinds of legacy use. These modules are either now available separately as part of OpenJFX, are now in both OpenJDK and the Oracle JDK because they were commercial features which Oracle contributed to OpenJDK (e.g., Flight Recorder), or were removed from Oracle JDK 11 (e.g., JNLP).

Oracle JDK Releases for Java 11 and later (6)

... yet with some cosmetic and packaging differences

- The output of the ***`java --version`*** and ***`java -fullversion`*** commands will distinguish Oracle JDK builds from OpenJDK builds, so that support teams can diagnose any issues that may exist. Specifically, running java --version with an Oracle JDK 11 build results in:
  - java 11 2018-09-25
  - Java(TM) SE Runtime Environment 18.9 (build 11+28)
  - Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11+28, mixed mode)
- And for an OpenJDK 11 build:
  - openjdk version "11" 2018-09-25
  - OpenJDK Runtime Environment 18.9 (build 11+28)
  - OpenJDK 64-Bit Server VM 18.9 (build 11+28, mixed mode)

Oracle JDK Releases for Java 11 and later (7)

... yet with some cosmetic and packaging differences

- The Oracle JDK has always required third party cryptographic providers to be signed by a known certificate. The cryptography framework in OpenJDK has an open cryptographic interface, meaning it does not restrict which providers can be used. Oracle JDK 11 will continue to require a valid signature, and Oracle OpenJDK builds will continue to allow the use of either a valid signature or unsigned third party crypto provider.
- Oracle JDK 11 will continue to include installers, branding and JRE packaging for an experience consistent with legacy desktop uses. Oracle OpenJDK builds are currently available as zip and tar.gz files, while alternative distribution formats are being considered.

Oracle JDK Releases for Java 11 and later (8)

What should we call them?

- Ideally, we would simply refer to all Oracle JDK builds as the "Oracle JDK," either under the GPL or the commercial license depending on your situation. However, for historical reasons while the small remaining differences exist, we will refer to them separately as Oracle's OpenJDK builds, and the Oracle JDK.

Java in a World of Containers and Open Source

Java in a World of Containers

Java's characteristic make it ideal for a container environment

- Managed language/runtime
- Hardware and operating system agnostic
- Safety and secure enforced by JVM
- Compatibility is a key design goal
- JVM ensures stable execution when runtime environment changes
- Broad ecosystem
- Keeping Java the first choice for container deployments
- 30 Milliarden aktive JVMs und 20 Milliarden Cloud-Connected-JVMs

Open source commercial features

What is being open-sourced in Java

- Java Mission Control
  - Monitor and manage Java applications with minimal performance overhead
- Java Flight Recorder
  - Collects diagnostic and profiling data about a running Java application
- Application Class Data Sharing
  - Enables you to place classes from the standard extensions directories and the application class path in the shared archive
- Java Usage Tracker
  - Tracks how the JRE‘s are being used in your systems

JDK Mission Control 7.0 Early Access

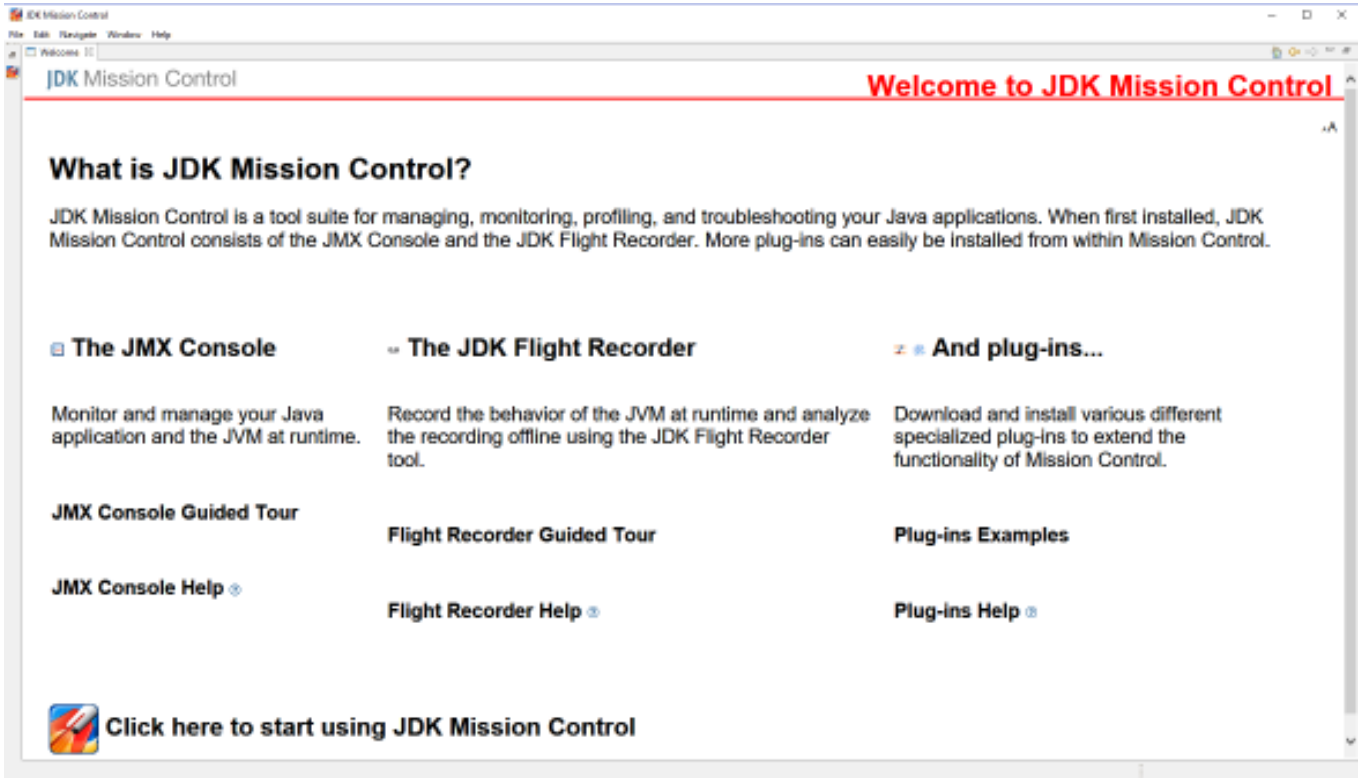
JDK Mission Control 7.0 – Early Access Release

https://jdk.java.net/jmc/

- JFR is included in JDK11
  - JFR records the data
- JMC as the data visualizing tool is separate available
  - https://jdk.java.net/jmc/

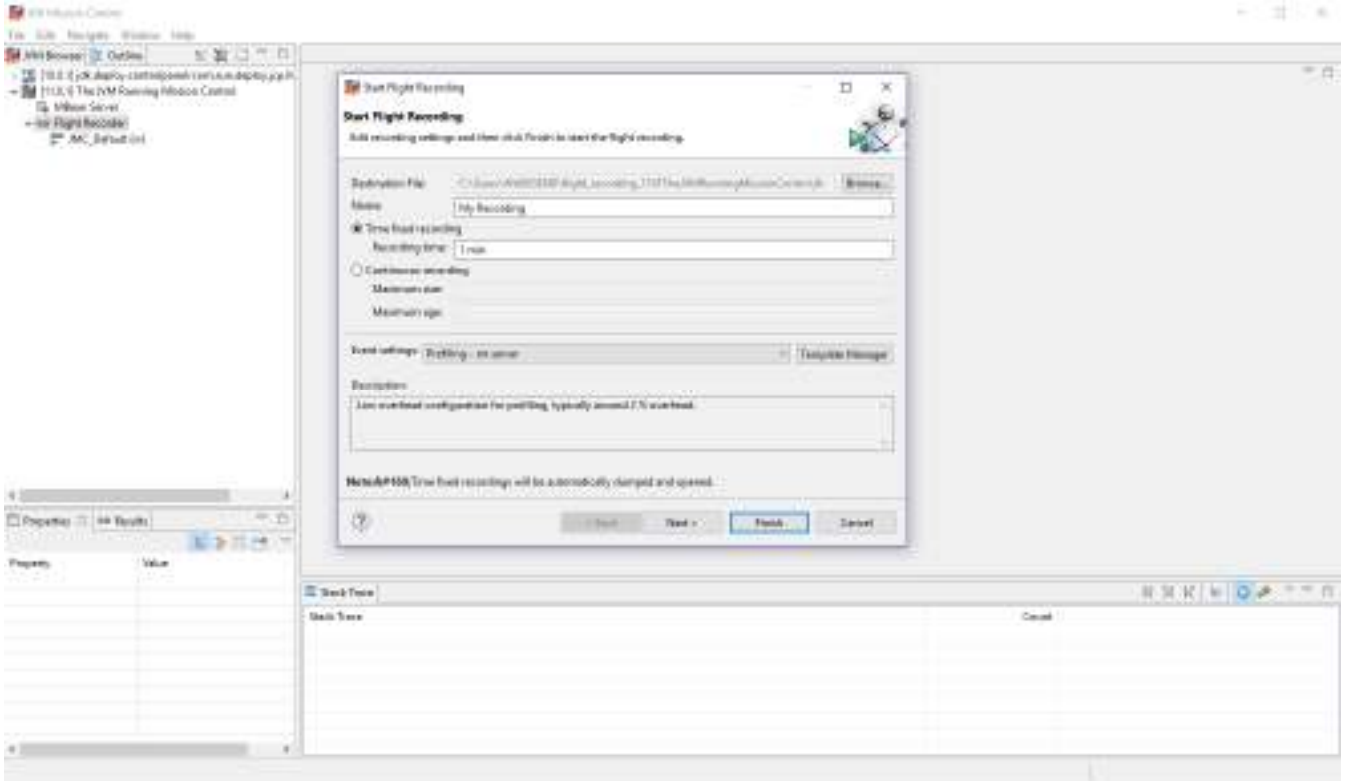
JDK Mission Control 7.0 – Early Access Release

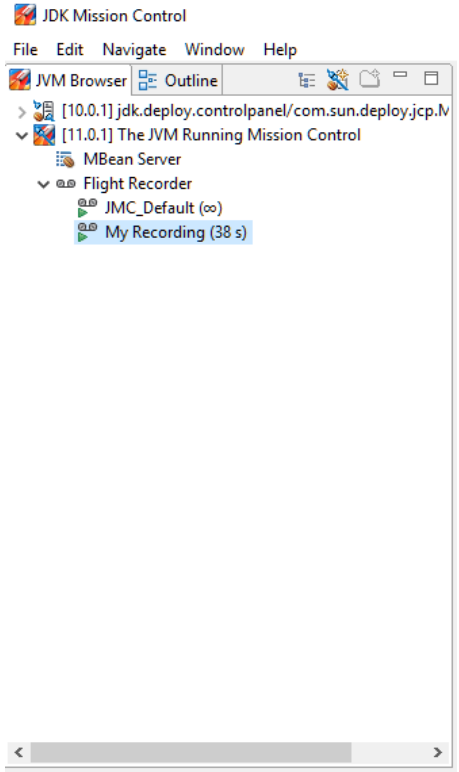
https://jdk.java.net/jmc/



JDK Mission Control 7.0 – Early Access Release

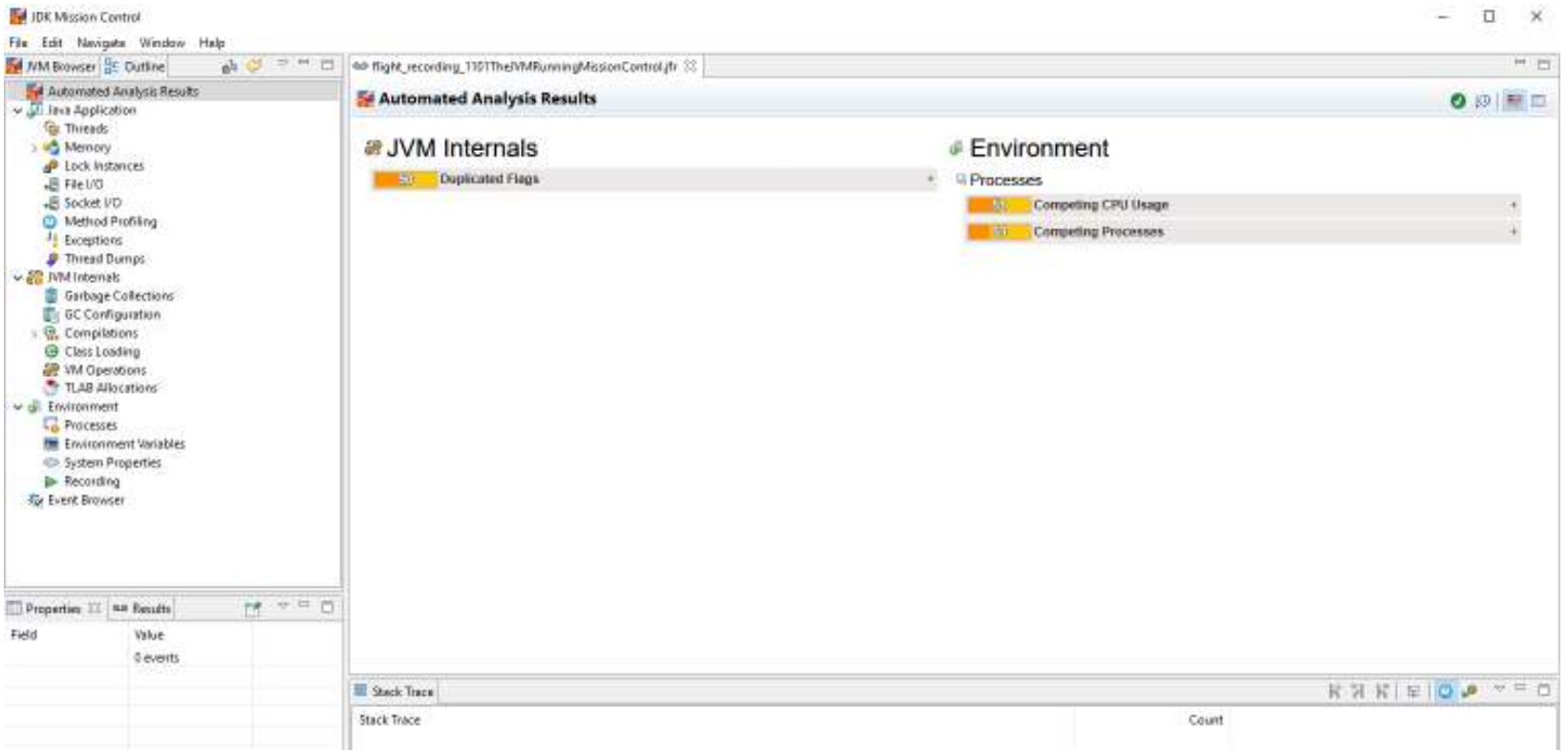
https://jdk.java.net/jmc/





JDK Mission Control 7.0 – Early Access Release

https://jdk.java.net/jmc/



JDK 11 Migration Guide

Migrating to Oracle JDK 11 - Migration Guide (1)

https://docs.oracle.com/en/java/javase/11/migrate/index.html

- **Migrating to JDK 11**
  - The purpose of this guide is to help you identify potential issues and give you suggestions on how to proceed as you migrate your existing Java application from JDK 8, or earlier version of the JDK, to JDK 11.
  - This guide is not significantly different than the JDK 10 Migration Guide.
  - Every new Java SE release introduces some binary, source, and behavioral incompatibilities with previous releases.
  - The modularization of the Java SE Platform that happened in JDK 9 brought many benefits, but also many changes.
  - Code that uses only official Java SE Platform APIs and supported JDK-specific APIs should continue to work without change.
  - Code that uses JDK-internal APIs should continue to run but should be migrated to use external APIs.

Migrating to Oracle JDK 11 - Migration Guide (2)

https://docs.oracle.com/en/java/javase/11/migrate/index.html

- **Prepare for Migration**
  - Download JDK 11
  - Run Your Program Before Recompiling
  - Update Third-Party Libraries
  - Compile Your Application if Needed
  - Run jdeps on Your Code

Migrating to Oracle JDK 11 - Migration Guide (3)

https://docs.oracle.com/en/java/javase/11/migrate/index.html

- **Migrating from JDK 8 to later JDK releases**
  - New Version-String Scheme
  - Understanding Runtime Access Warnings
  - Changes to the Installed JDK/JRE Image
  - Removed or Changed APIs
  - Deployment
  - Security Updates in JDK 9
  - Changes to Garbage Collection
  - Removed Tools and Components
  - Removed macOS-Specific Features

Issues fixed in JDK 11 per organization

https://blogs.oracle.com/java-platform-group/building-jdk-11-together

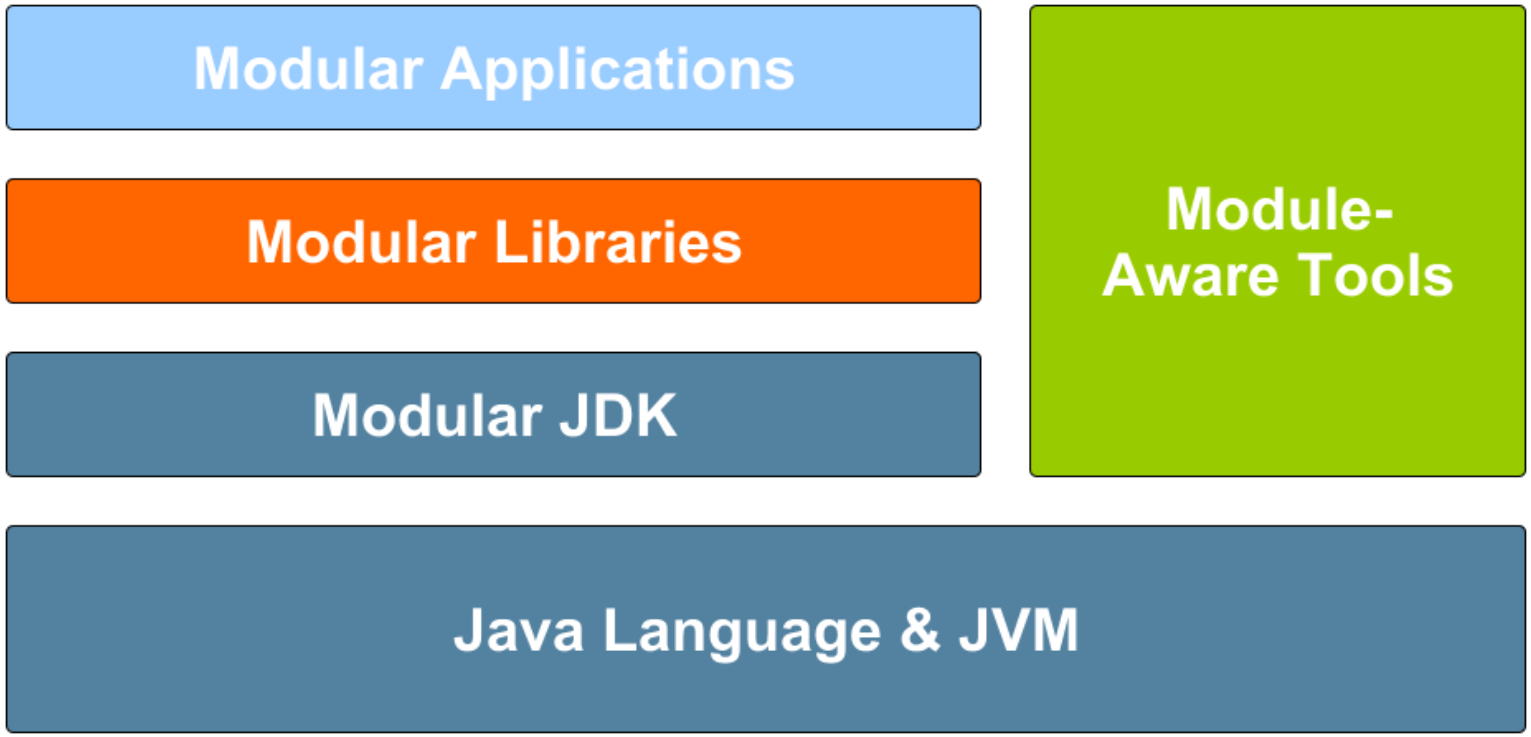
- **JDK BUG System commits**
- Overall 2468 JIRA issues marked as fixed in JDK 11
- 1963 issues were completed by Oracle
- 505 issues were contributed by individual developers and developers working for other organizations

Oracle JDK 11 – Changes in the Release Notes

https://www.oracle.com/technetwork/java/javase/11-relnote-issues-5012449.html

- The deployment stack, required for Applets and Web Start Applications, was deprecated in JDK 9 and has been removed in JDK 11.
- Without a deployment stack, the entire section of supported browsers has been removed from the list of supported configurations of JDK 11.
- Auto-update, which was available for JRE installations on Windows and macOS, is no longer available.
- In Windows and macOS, installing the JDK in previous releases optionally installed a JRE. In JDK 11, this is no longer an option.
- In this release, the JRE or Server JRE is no longer offered. Only the JDK is offered. Users can use jlink to create smaller custom runtimes.
- JavaFX is no longer included in the JDK. It is now available as a separate download from openjfx.io.
- Java Mission Control, which was shipped in JDK 7, 8, 9, and 10, is no longer included with the Oracle JDK. It is now a separate download.
- Previous releases were translated into English, Japanese, and Simplified Chinese as well as French, German, Italian, Korean, Portuguese (Brazilian), Spanish, and Swedish. However, in JDK 11 and later, French, German, Italian, Korean, Portuguese (Brazilian), Spanish, and Swedish translations are no longer provided.
- Updated packaging format for Windows has changed from tar.gz to .zip, which is more common in Windows OSs.
- Updated package format for macOS has changed from .app to .dmg, which is more in line with the standard for macOS.

Modular Development with JDK 11



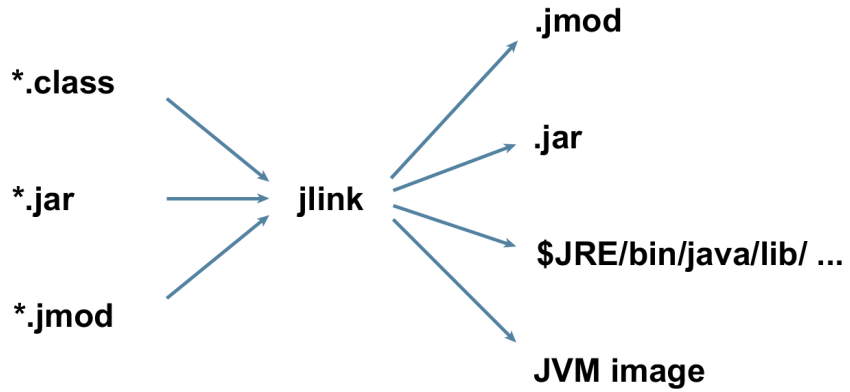
jlink - generiert JRE und Applikations-Images (1)

- Platzsparende Runtime, inklusive eigener Anwendungsmodule im frei wählbaren Verzeichnis

```
jlink <options> --module-path <modulepath> --output <path>

jlink --module-path $JDKMODS:mlib --add-modules myapp --output myimage

C:\> C:\jdk-11\bin\jlink --module-path C:\jdk-11\jmods;mlib --add-modules com.greetings --
compress=2 --verbose --output greetingsapplication
       com.greetings file:///C:/mlib/com.greetings.jar
       java.base file:///C:/jdk-11/jmods/java.base.jmod
       Providers: java.base provides java.nio.file.spi.FileSystemProvider used by java.base
```



jlink - generiert JRE und Applikations-Images (2)

- Image-Verzeichnis C:\greetingsapplication 27,5 MB

```
C:\greetingsapplication> dir

Directory of C:\greetingsapplication

20.11.2018  17:44    <DIR>          bin
20.11.2018  17:44    <DIR>          conf
20.11.2018  17:44    <DIR>          include
20.11.2018  17:44    <DIR>          legal
20.11.2018  17:44    <DIR>          lib
20.11.2018  17:44                54 release
```

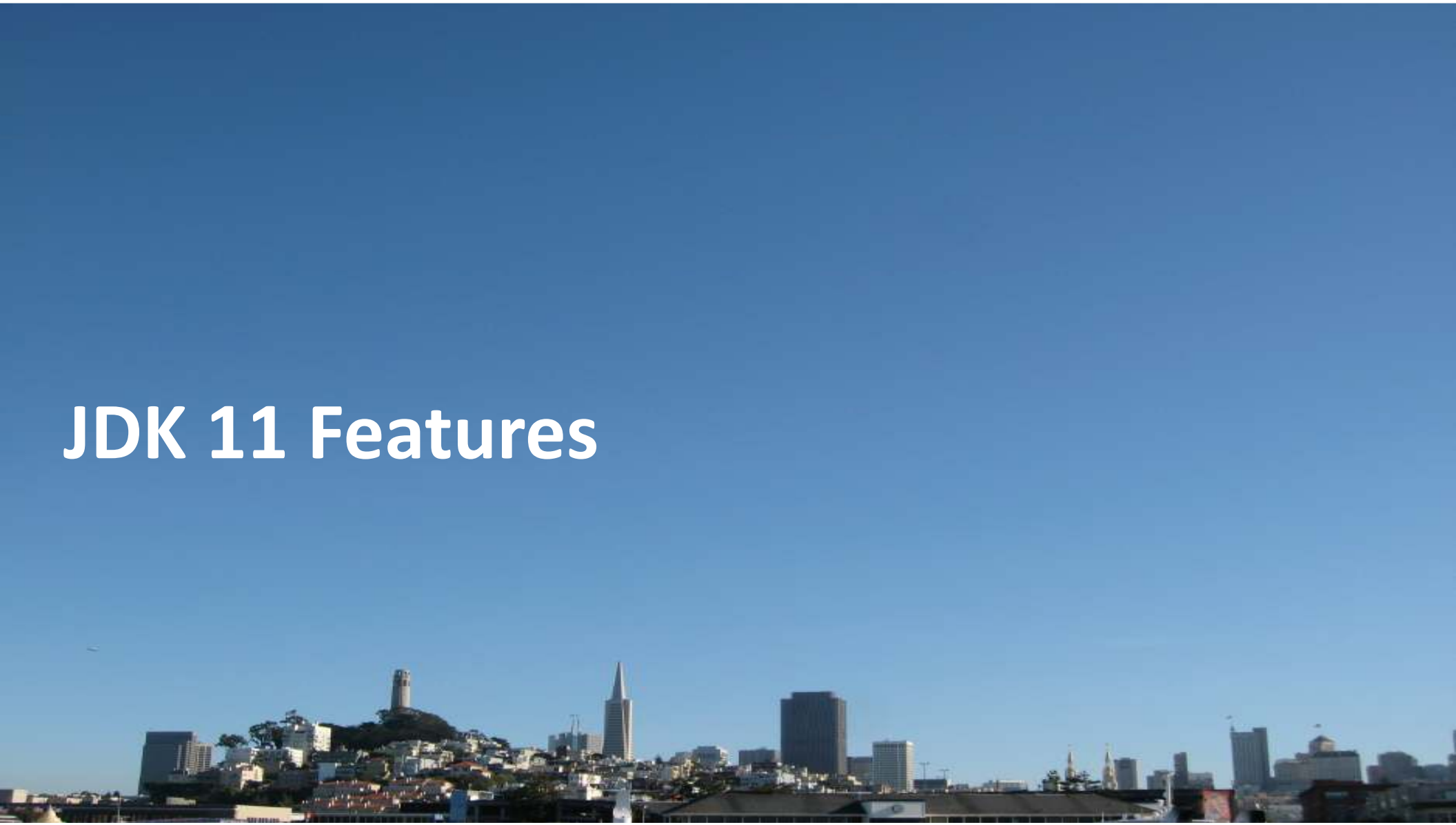
- Datei release „11“

```
JAVA_VERSION="11"
MODULES="java.base com.greetings"
```

- Datei release mit früherem „build 9-ea+142-jigsaw-nightly-h5677-20161102”

```
#Thu Mar 09 22:11:23 CET 2017
OS_NAME="Windows"
MODULES="java.base com.greetings"
OS_VERSION="5.1"
OS_ARCH="i586"
JAVA_VERSION="9"
JAVA_FULL_VERSION="9-ea"
```

JDK 11 Features



JDK 11 Features – JEP's included

- 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

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

Graal

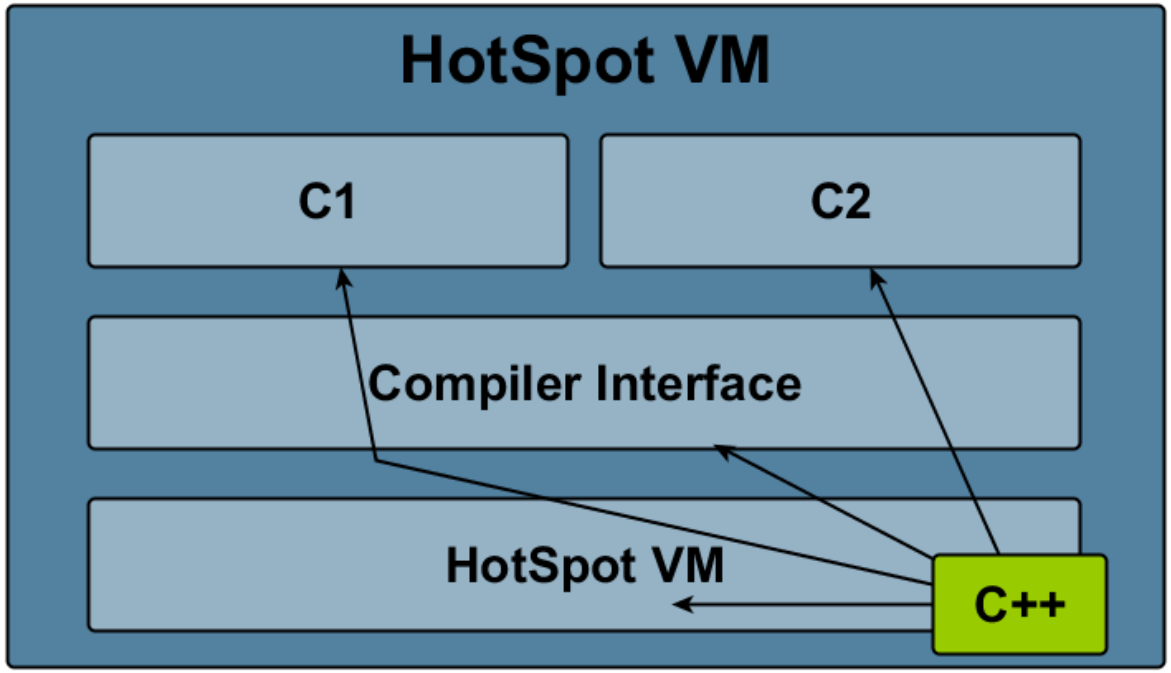
JIT Compiler

- C1 Client Compiler
  - Minimiert Startup-Zeit
- C2 Server Compiler
  - Dauerhafte Performance-Verbesserungen
  - Intensivere Analyse vom ausgeführten Code
  - Optimierungen können besser platziert werden

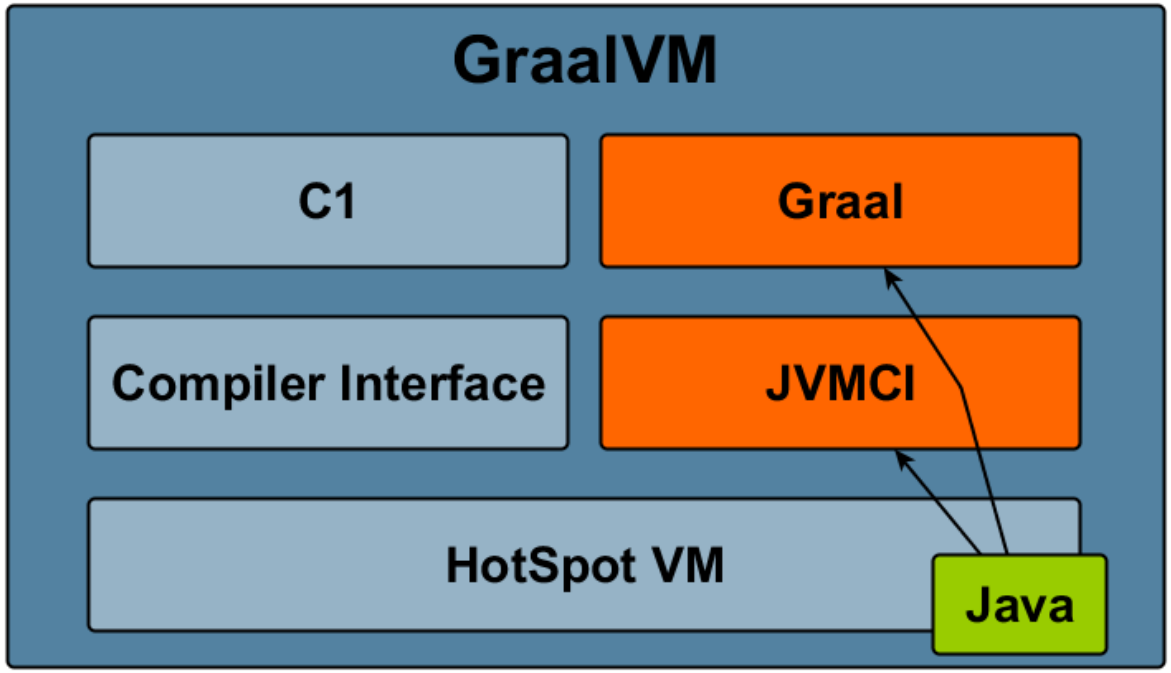
JIT Compiler with Tiered Compilation

- C1 Client Compiler
  - Minimiert Startup-Zeit
  - java -client -XX:+TieredCompilation
- C2 Server Compiler
  - Läuft mit, aber ohne Tiered Compilation
- Tiered-Compilation Ausführungs-Level
  - Level 0: interpreted code
  - Level 1: simple C1 compiled code (with no profiling)
  - Level 2: limited C1 compiled code (with light profiling)
  - Level 3: full C1 compiled code (with full profiling)
  - Level 4: C2 compiled code (uses profile data from the previous steps)

JIT Compiler written in C++



JIT Compiler written in Java



Graal

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

- The aim of this project is to expose VM functionality via Java APIs. Namely, we want to make it feasible to write in Java a dynamic compiler and interpreter for a language runtime. These components will seamlessly integrate and leverage existing VM infrastructure (e.g., HotSpot).
- The design of the dynamic compiler uses features of Java that make it highly extensible such that adding extra IR nodes and/or transformations is straightforward. At the same time, it should produce excellent code quality without compromising compile time and memory usage by the JVM.
- Building on the compiler, we aim to develop a multi-language interpreter framework. Java will be just one member in the family of supported languages. The use of partial evaluation will allow the framework to deliver competitive performance.

GraalVM Repository Structure

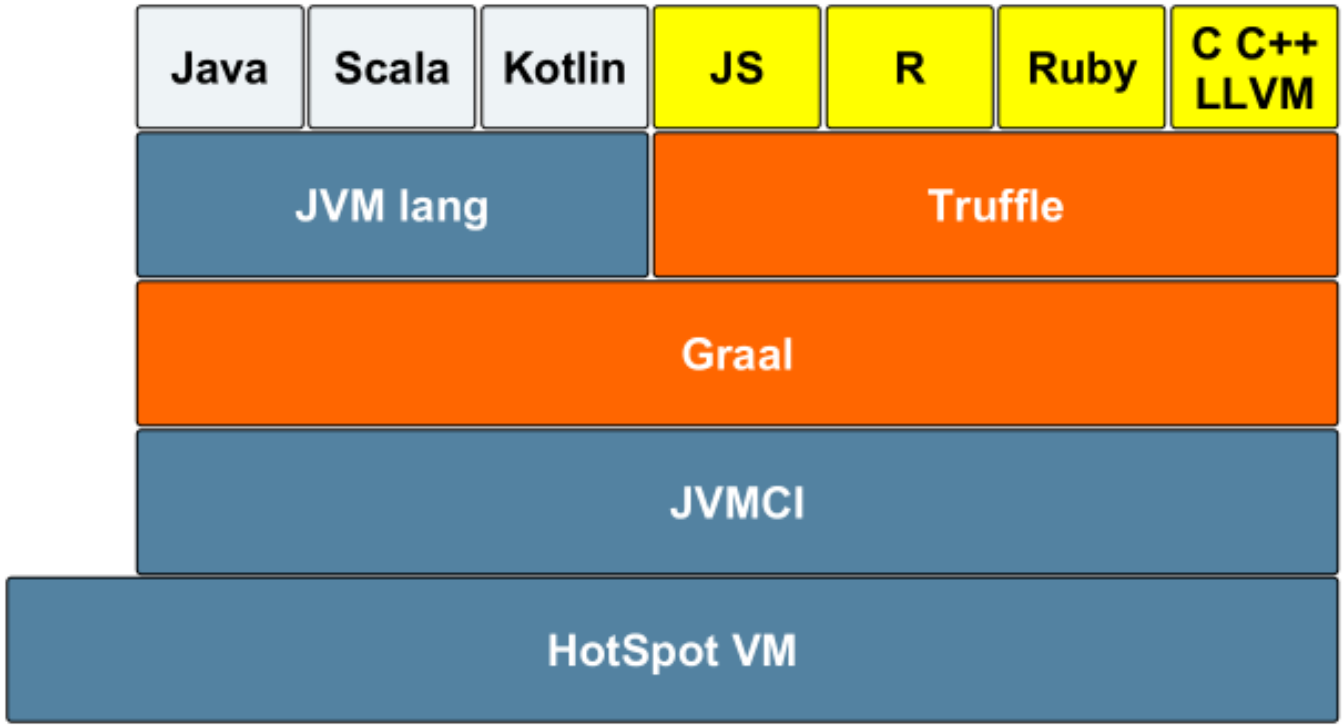
The GraalVM main source repository includes these components

- Graal SDK contains long term supported APIs of GraalVM.
- Graal compiler written in Java that supports both dynamic and static compilation and can integrate with the Java HotSpot VM or run standalone.
- Truffle language implementation framework for creating languages and instrumentations for GraalVM.
- Tools contains a set of tools for GraalVM languages implemented with the instrumentation framework.
- Substrate VM framework that allows ahead-of-time (AOT) compilation of Java applications under closed-world assumption into executable images or shared objects.
- Sulong is an engine for running LLVM bitcode on GraalVM.
- TRegex is an implementation of regular expressions which leverages GraalVM for efficient compilation of automata.
- VM includes the components to build a modular GraalVM image.

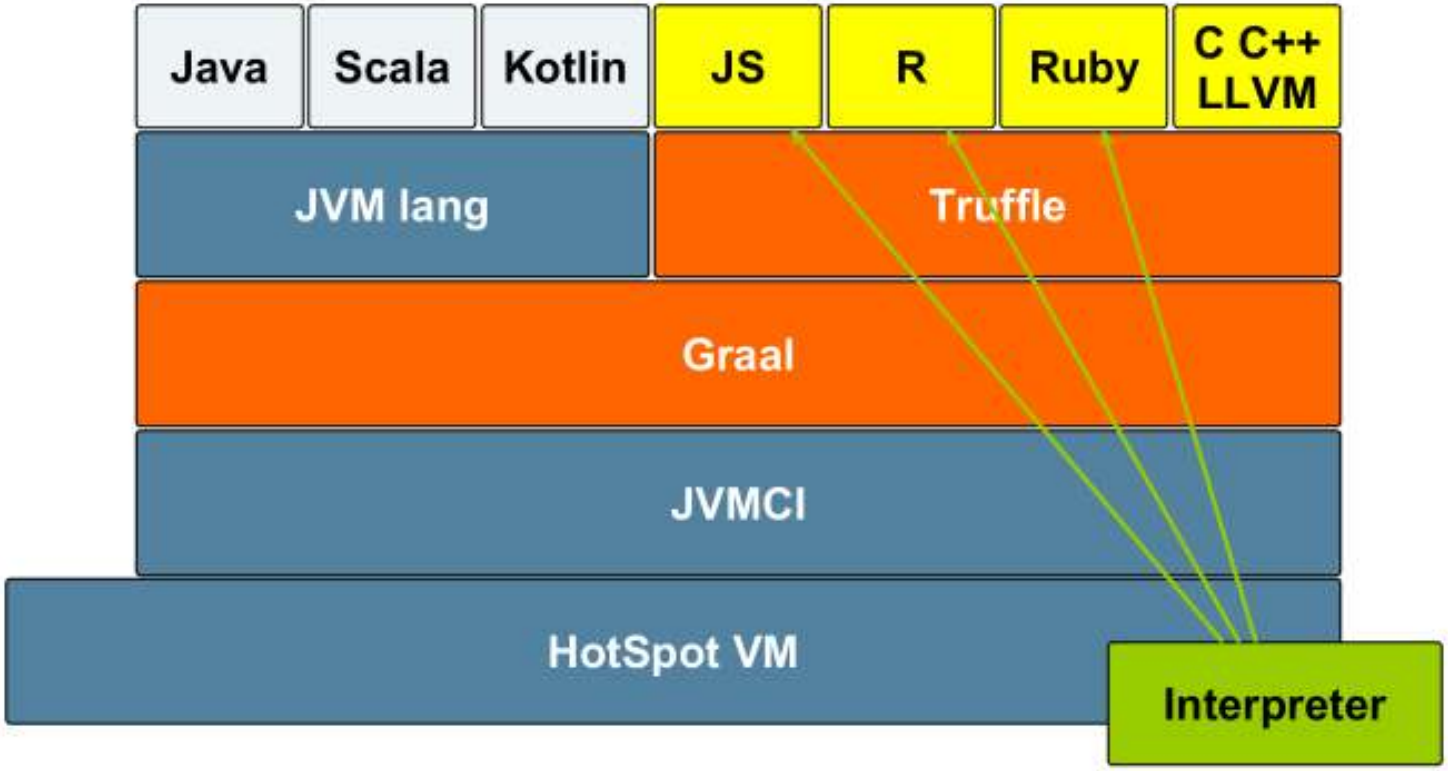
GraalVM

- Graal
  - JIT Compiler
    - **Graal in GraalVM - A new Java JIT Compiler**
  - Graal integrated via Java Virtual Machine Compiler Interface (JVM CI)
  - Use a JDK with Graal (jdk.internal.vm.compiler)
  - –XX:+UnlockExperimentalVMOptions –XX:+EnableJVMCI –XX:+UseJVMCICompiler
- Truffle
  - Language Implementation Framework
- Substrate VM
  - Runtime Library and a Set of Tools for
- Building Java AOT Compiled Code

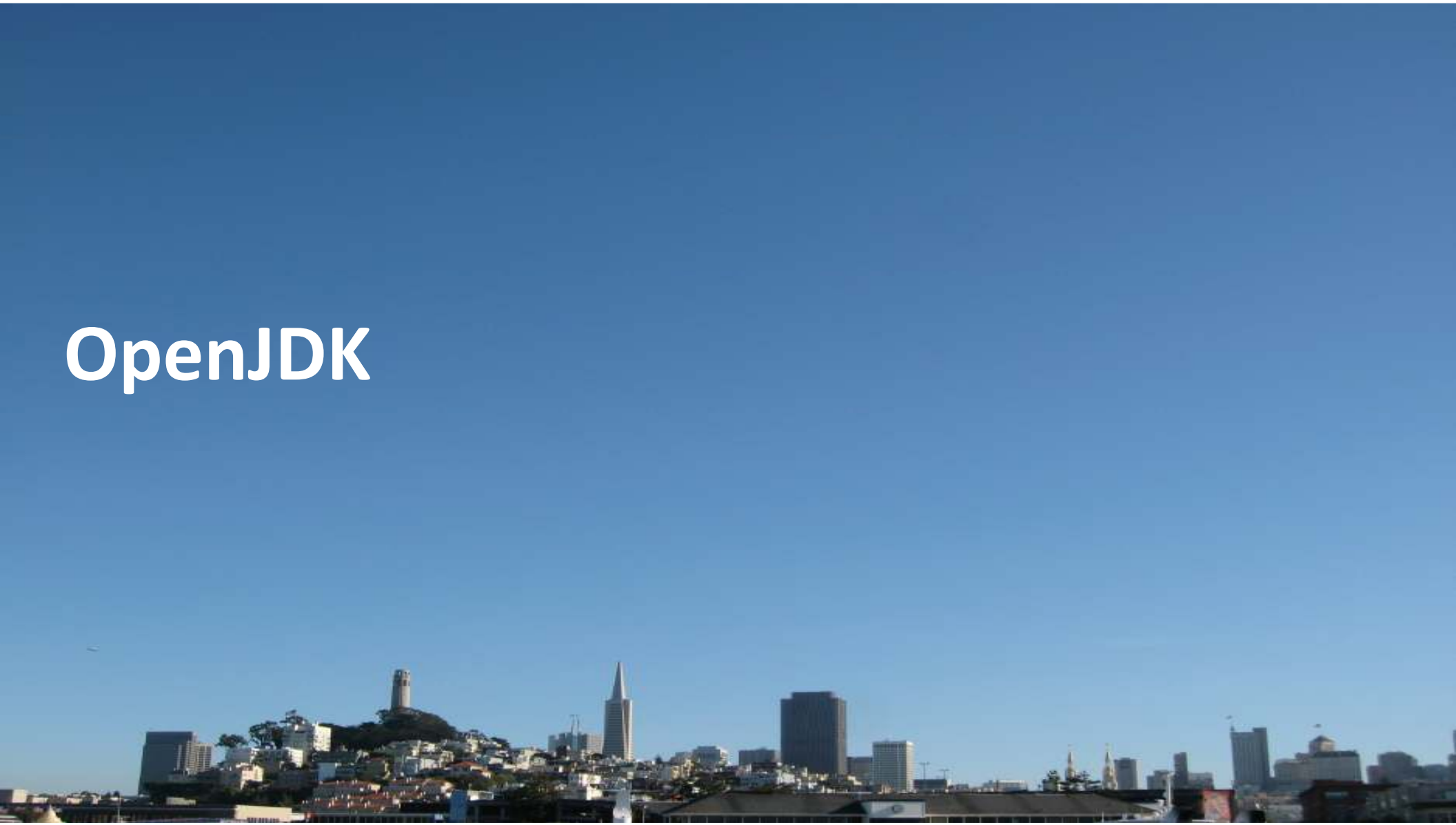
GraalVM - Polyglot (1)



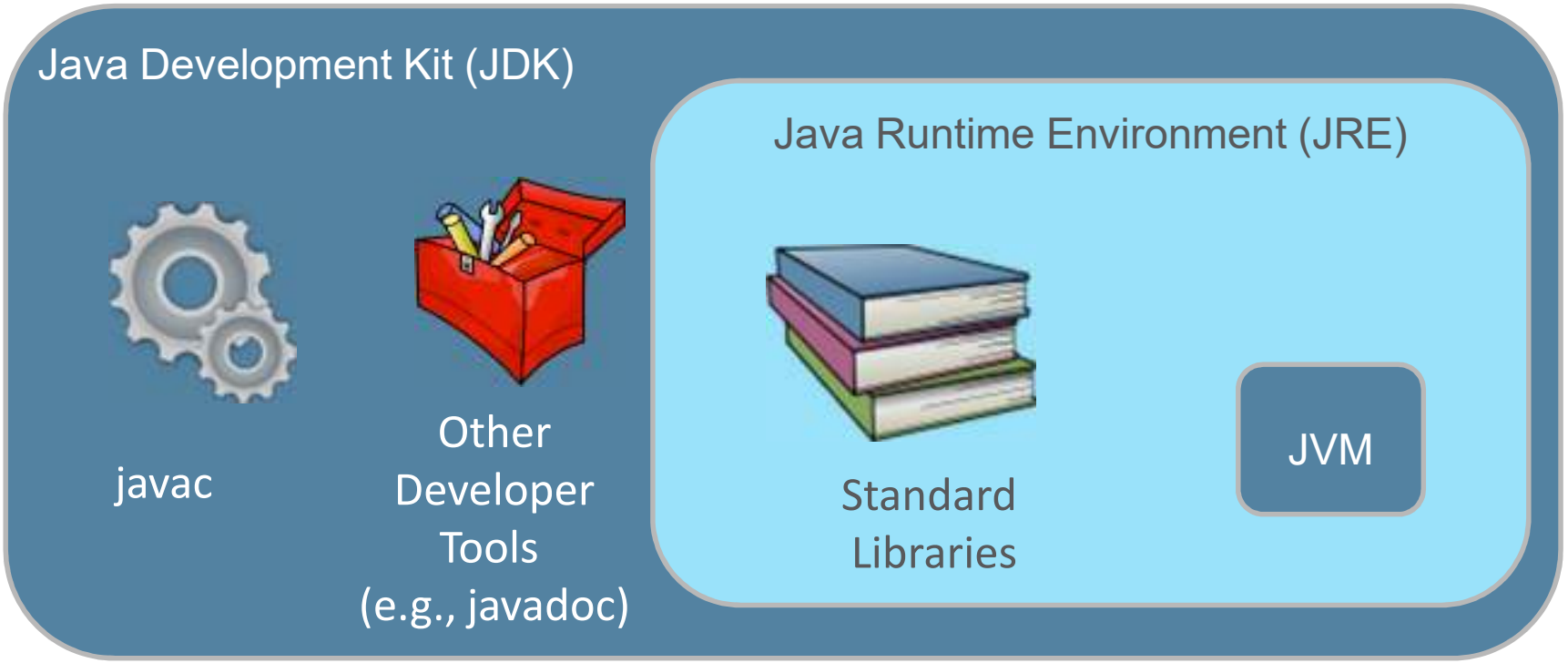
GraalVM - Polyglot (2)



OpenJDK



JDK, JRE and JVM



OpenJDK Cycle

Open Community – GPL v2 + Classpath Exception

OpenJDK

Dozens of Projects

Java SE RI Binaries

Source contributions

Companies AND Individuals

12 Million Developers using Java

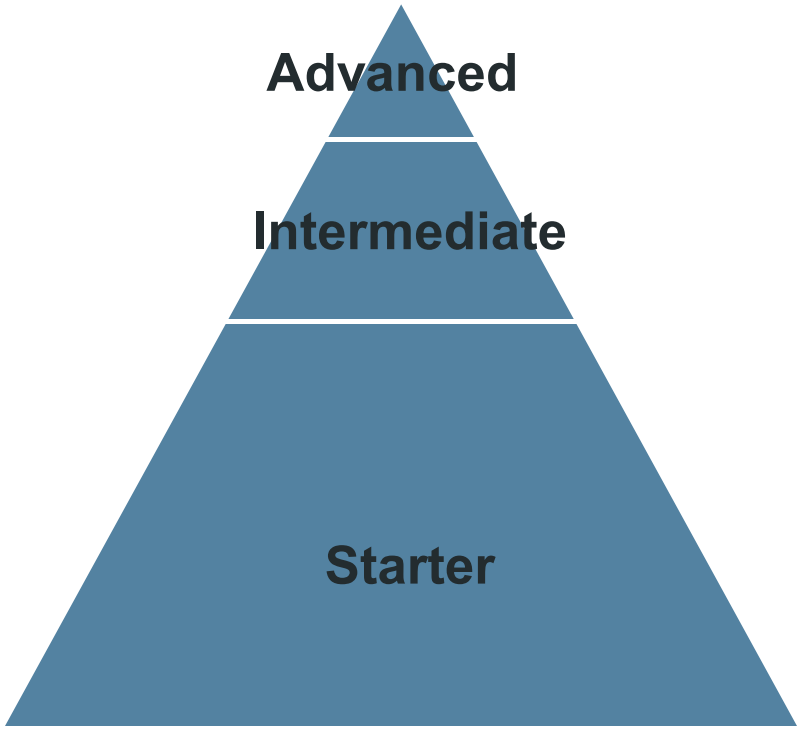
OpenJDK Source is consumed by organizations and individuals. Components are added/removed to taste, and binaries are produced, such as Oracle JDK.

Both Gratis and Commercial offerings (Including Support, Enterprise tooling, etc, available)

Customized VM for Internal Use

...and many more

Community Engagement



Solving non-trivial bugs

Non-complex bugs, review, test porting

Fix build warnings, trivial bugs,
test and report regressions

http://openjdk.java.net/census#members

OpenJDK Project

- Open source implementation of Java SE
- Licensed under GPLv2 with classpath exception
- Reference implementation for Java SE 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, ..
- Different vendors/implementers working in common space
  - Oracle, IBM, Intel, Twitter, Azul, Google, RedHat, SAP
- Many Java SE related projects

Oracle Contributor Agreement - OCA

- Copyright and patent sharing agreement with regards to your contribution
  - Simple dual license model
- You and Oracle could treat the contribution as being the sole owner
- Oracle guarantees to always publish any contribution under a suitable Free Software Foundation (FSF) or Open Source Initiative (OSI) approved license

Oracle Contributor Agreement - OCTLA Signatories List

- The following organizations and individuals have signed the OpenJDK Community TCK License Agreement (OCTLA) and been granted access to the JCK
  - Signatories for Java SE 9, or later
    - Amazon.com Services, Inc.
    - Azul Systems, Inc.
    - BellSoft
    - Canonical
    - Fujitsu Technology Solutions GmbH
    - London Jamocha Community
    - Loongson Technology Co., Ltd.
    - MicroDoc Software GmbH
    - Red Hat
    - SAP
    - SUSE Linux GmbH
    - Twitter

http://openjdk.java.net/groups/conformance/JckAccess/jck-access.html

OpenJDK and Java Ownership

- Java is a registered trademark of Oracle
  - OpenJDK is named idependently
  - Access to the Technology Compatibility Kit (TCK) is restricted
- TCK access requires to sign the OpenJDK Community TCK License Agreement (OCTLA)
  - OpenJDK or substantially projects only
  - OCTLA did not allow to disclose TCK information to Non-OCTLA licensees
- OCTLA is not a trademark license
- OpenJDK could be used according to the OpenJDK Trademark Notice
  - http://openjdk.java.net/legal/openjdk-trademark-notice.html

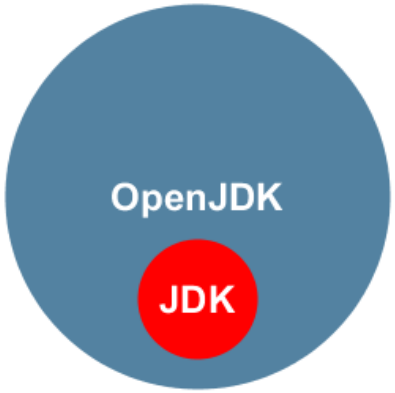
OpenJDK Infrastructure

- OpenJDK homepage
- Mercurial repositories
  - Forests of nested sub-repositories
- Mailing lists
- JDK Bug System
  - https://bugs.openjdk.java.net
- Code review server cr.openjdk.java.net
  - Review requests in Webrev format
  - Access only for Author or higher roles
- OpenJDK Wiki for group member only to get write access

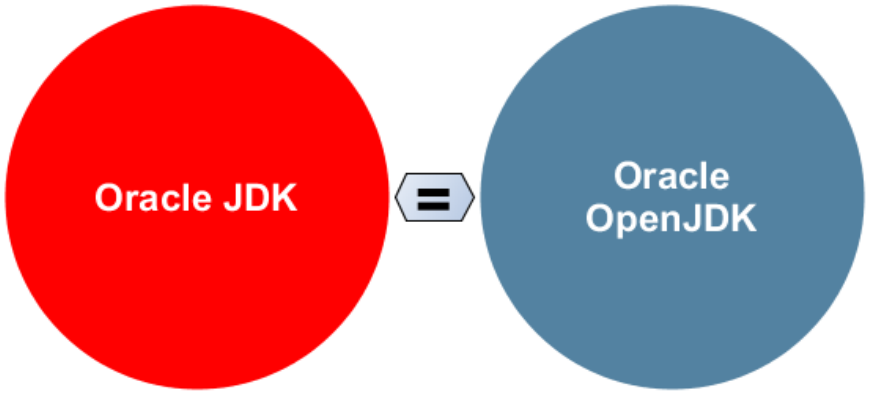
OpenJDK Bug Reporting

- Bug should be reproduced with the latest builds
- Debug build is appropriate
- Simple test case desirable (Jtreg test, ..)
- JDK Bug System search for similar bugs
  - Public visible bugs
  - Invisible bugs like security bugs and customer bugs
- Non-Authors bug submit only with Java Bug Database and temp bug ID
  - https://bugs.java.com/
- Use to write directly to the mailing list for simple reproducible bugs

Difference between Oracle OpenJDK & Oracle JDK with Java SE 8 .. and the change since Oracle JDK 11



- **Java SE 8**
- Security
- Performance
- Support



- **Since Oracle JDK 11 it‘s technically interchangeable**
- Security
- Performance
- Support only with Oracle JDK

Download and install prebuilt OpenJDK 11 packages

https://openjdk.java.net/install/

- Oracle's OpenJDK JDK binaries for Windows, macOS, and Linux are available on release-specific pages of jdk.java.net as .tar.gz or .zip archives.
- As an example, the archives for JDK 11 may be found on jdk.java.net/11 and may be extracted on the command line using
  - $ tar xvf openjdk-11*_bin.tar.gz

  or
  - $ unzip openjdk-11*_bin.zip

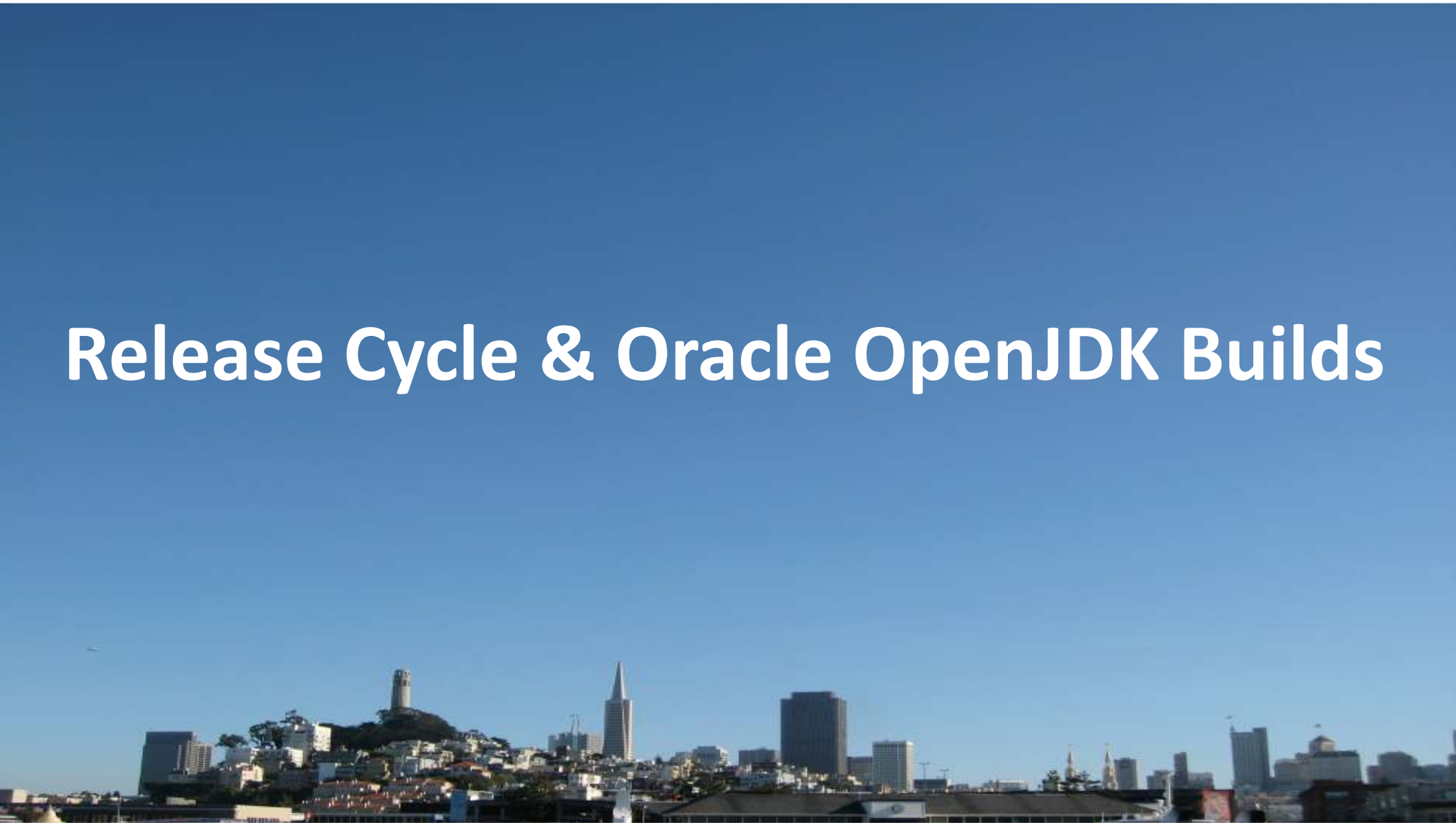
  depending on the archive type.

JDK 11.0.1 General-Availability Release

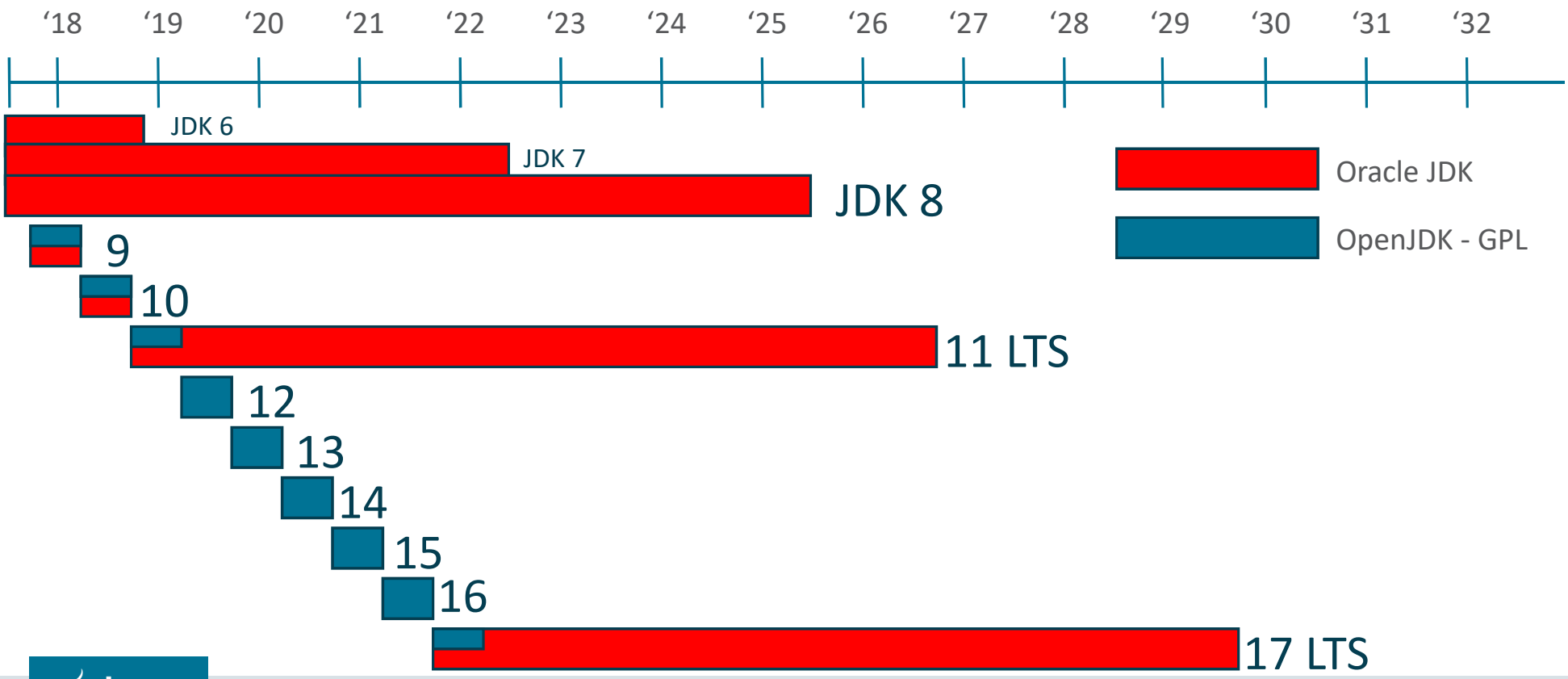
https://jdk.java.net/11/

- This page provides production-ready open-source builds of the Java Development Kit, version 11.0.1, an implementation of the Java SE 11.0.1 Platform under the GNU General Public License, version 2, with the Classpath Exception.
- Commercial builds of JDK 11.0.1 from Oracle under a non-open-source license, for a wider range of platforms, can be found at the Oracle Technology Network.
- Documentation
  - Features
  - Release notes
  - API Javadoc
  - Tool & command reference
- Builds
  - Linux / x64 tar.gz (sha256)
  - macOS / x64 tar.gz (sha256)
  - Windows / x64 zip (sha256)

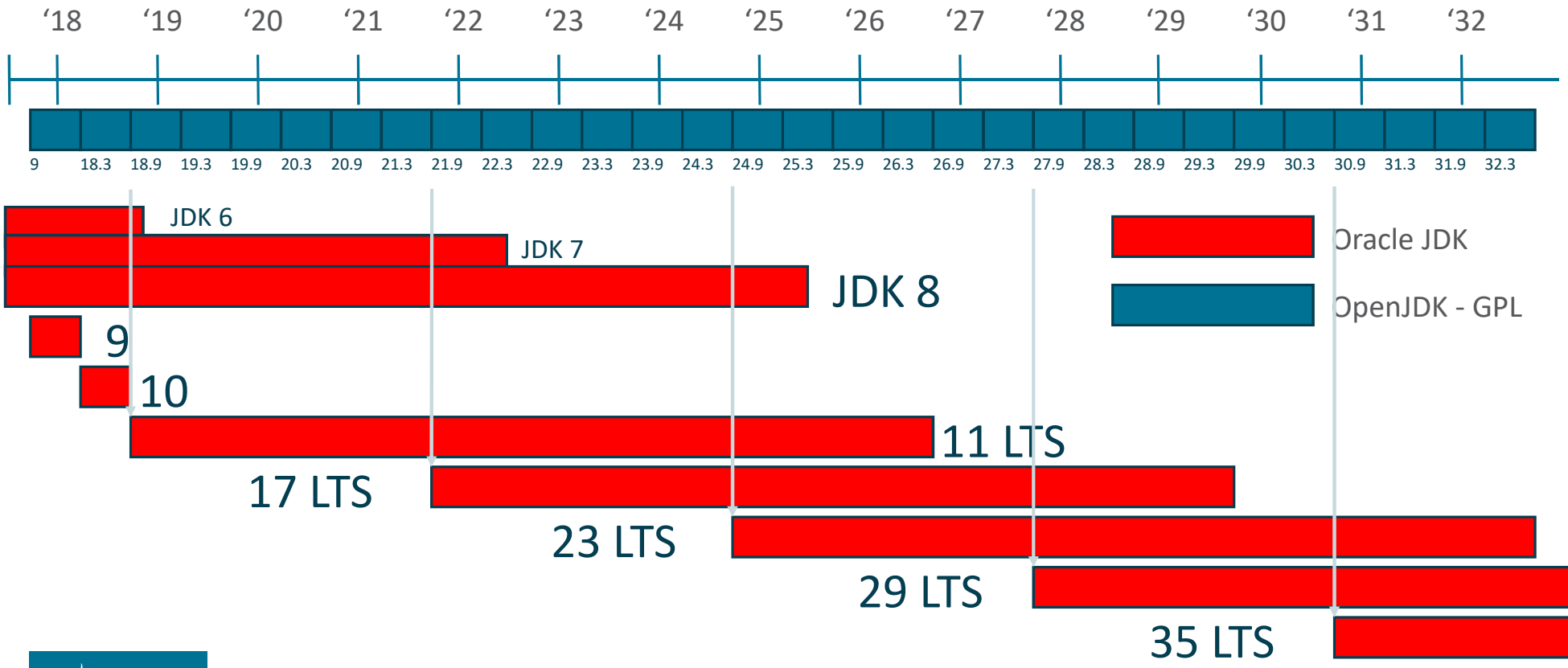
Release Cycle & Oracle OpenJDK Builds



Oracle JDK & OpenJDK



New JDK Release model



Summary

- The Java platform development on OpenJDK is becoming more open
  - Contributing all commercial features
  - GPL + CPE build
- The cloud is demanding a faster pace and continuous delivery
  - Uptake new Java releases every 6-month
- Let‘s continue to innovate and advance the Java SE Platform on OpenJDK together!
- Join and become an OpenJDK contributor
  - https://openjdk.java.net

Danke!

Wolfgang.Weigend@oracle.com

Twitter: @wolflook