



北京时间 2018 年 9 月 26 日，Oracle 官方宣布 Java 11 正式发布。这是 Java 大版本周期变化后的第一个长期支持版本，非常值得关注。从官网即可下载，最新发布的 Java11 将带来 ZGC、Http Client 等重要特性，一共包含 17 个 JEP (JDK Enhancement Proposals，JDK 增强提案)。

- 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

JDK 11 将是一个 **企业不可忽视**的版本。从时间节点来看，JDK 11 的发布正好处在 JDK 8 免费更新到期的前夕，同时 JDK 9、10 也陆续成为“**历史版本**”，下面是 Oracle JDK 支持路线图：



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

JDK 更新很重要吗？非常重要，在过去的很多年中，Oracle 和 OpenJDK 社区提供了接近免费的午餐，导致人们忽略了其背后的海量工作和价值，这其中包括但不限于：最新的安全更新，如，安全协议等基础设施的升级和维护，安全漏洞的及时修补，这是 Java 成为企业核心设施的基础之一。大量的新特性、Bug 修复，例如，容器环境支持，GC 等基础领域的增强。很多生产开发中的 Hack，其实升级 JDK 就能解决了。不断改进的 JVM，提供接近零成本的性能优化...

JDK 11 是一个长期支持版本 (LTS, Long-Term-Support)

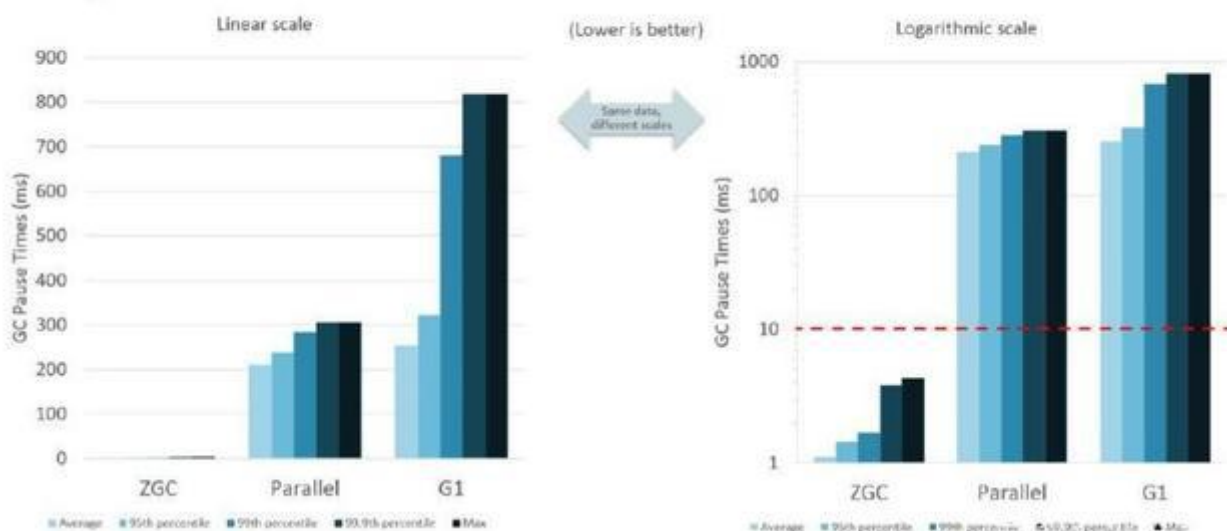
对于企业来说，选择 11 将意味着长期的、可靠的、可预测的技术路线图。其中免费的 OpenJDK11 确定将得到 OpenJDK 社区的长期支持，LTS 版本将是可以放心选择的版本。

从 JVM GC 的角度，JDK11 引入了两种新的 GC，其中包括也许是划时代意义的 ZGC，虽然其目前还是实验特性，但是从能力上来看，这是 JDK 的一个巨



大突破，为特定生产环境的苛刻需求提供了一个可能的选择。例如，对部分企业核心存储等产品，如果能够保证不超过 10ms 的 GC 暂停，可靠性会上一个大的台阶，这是过去我们进行 GC 调优几乎做不到的，是能与不能的问题。

SPECjbb® 2015 – Pause Times



对于 G1 GC ,相比于 JDK 8 ,升级到 JDK 11 即可免费享受到 :并行的 Full GC ,快速的 CardTable 扫描，自适应的堆占用比例调整（IHOP），在并发标记阶段的类型卸载等等。这些都是针对 G1 的不断增强，其中串行 Full GC 等甚至是曾经被广泛诟病的短板，你会发现 GC 配置和调优在 JDK11 中越来越方便。

云计算时代的监控、诊断和 **Profiling** 能力，这个是相比 ZGC 更具生产实践意义的特性。

Java 的应用场景跨度很大，从单机长时间运行的 Java 应用，发展成为分布式、大的单体应用或小的 function、瞬时或长时间运行等，应用场景非常复杂。

我们用什么工具诊断 Java 应用？



JDK 11 为我们提供了更加强大的基础能力，主要是两部分：

JEP 328: Flight Recorder (JFR) 是 Oracle 刚刚开源的强大特性。



JFR 是一套集成进入 JDK、JVM 内部的事件机制框架，通过良好架构和设计的框架，硬件层面的极致优化，生产环境的广泛验证，它可以做到极致的可靠和低开销。在 SPECjbb2015 等基准测试中，JFR 的性能开销最大不超过 1%，所以，工程师可以基本没有心理负担地在大规模分布式的生产系统使用，这意味着，我们既可以随时主动开启 JFR 进行特定诊断，也可以让系统长期运行 JFR，用以在复杂环境中进行“After-the-fact”分析。

在保证低开销的基础上，JFR 提供的能力可以应用在对锁竞争、阻塞、延迟，JVM GC、SafePoint 等领域，进行非常细粒度分析。甚至深入 JIT Compiler 内部，全面把握热点方法、内联、逆优化等等。JFR 提供了标准的 Java、C++ 等扩展 API，可以与各种层面的应用进行定制、集成，为企业应用栈或者复杂的分布式应用，提供 All-in-One 解决方案。



而这一切都是内建在 JDK 和 JVM 内部的，并不需要额外的依赖，开箱即用。

JEP 331: Low-Overhead Heap Profiling。

它来源于 Google 等业界前沿厂商的一线实践，通过获取对象分配细节，为 JDK 补足了对对象分配诊断方面的一些短板，工程师可以通过 JVMTI 使用这个能力增强自身的工具。

从 Java 类库发展的角度来看，JDK 11 最大的进步也是两个方面：

第一，HTTP/2 Client API，新的 HTTP API 提供了对 HTTP/2 等业界前沿标准的支持，精简而又友好的 API 接口，与主流开源 API（如，Apache HttpClient，Jetty，OkHttp 等）对等甚至更高的性能。与此同时它是 JDK 在 Reactive-Stream 方面的第一个生产实践，广泛使用了 Java Flow API 等，终于让 Java 标准 HTTP 类库在扩展能力等方面，满足了现代互联网的需求。

第二，就是安全类库、标准等方面的大范围升级，其中特别是 JEP 332: Transport Layer Security (TLS) 1.3，除了在安全领域的重要价值，它还是中国安全专家范雪雷所领导的 JDK 项目，完全不同于以往的修修补补，是个非常大规模的工程。

除此之外，JDK 还在逐渐进行瘦身工作，或者偿还 JVM、Java 规范等历史欠账，例如

335: Deprecate the Nashorn JavaScript Engine



它进一步明确了 Graal 很有可能将成为 JVM 向前演进的核心选择，
Java-on-Java 正在一步步的成为现实。