

# 2022 年 Java エコシステムの現状

最も人気のあるプログラミング言語の一つを徹底解説



## コンテンツ

概要	3
新しい標準である Java 11	3
Java 14 非 LTS 版で最も人気の高いバージョン	4
オラクルの人気は縮小し、アマゾンが台頭	5
周辺のものすべてを実行するコンテナ	5
コンテナでの計算設定	5
コンテナでのメモリ設定	6
ガベージイン・ガベージアウト	7
方法論	7
New Relic について	8

## 概要

現代のソフトウェア産業は非常に広範で、プログラミング言語の選択には事欠きません。Java はプラットフォームに依存せず、コンピュータシステム間で簡単に移動でき、何千ものライブラリが提供され、サポートも充実しているためソフトウェア開発者に絶大な人気を誇り、ほとんどすべての主要産業や経済分野で使用されています。

2020 年 3 月、New Relic は、パフォーマンスデータを提供する数百万のアプリケーションから収集したデータに基づき、初の [Java エコシステムの現状レポート](#) を発表しました。今回、Java 11 以来の長期サポート版 (LTS) である Java 17 がリリースされたことで、そのデータを改めて見直す機会になりました。このレポートを作成するために、New Relic は適切なデータを匿名化し、意図的に粗視化することで、Java エコシステムの全体像を把握できるようにしました。攻撃者やその他の悪意のある者の役に立つような詳細情報は、意図的にレポートから除外されています。

本レポートの目的は、今日の Java エコシステムの現状について文脈と知見を提供することです。

以下のカテゴリーについて検討しました。

- [実運用で最も使用されているバージョン](#)
- [最も人気のあるベンダー](#)
- [コンテナの台頭](#)
- [最も一般的なヒープサイズ設定](#)
- [最もよく使われるガベージコレクションアルゴリズム](#)

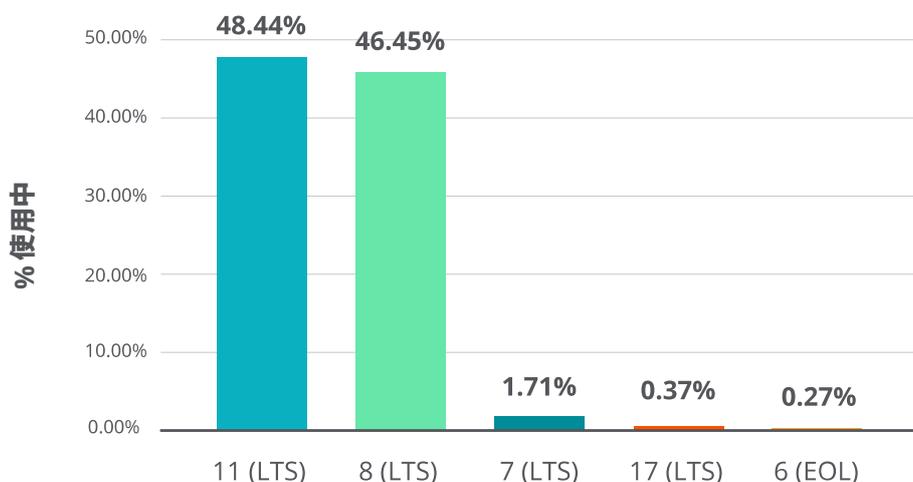
## 新しい標準である Java 11

2020 年に Java 11 が提供されて 1 年以上が経過しているにもかかわらず、大多数のアプリケーション (84.48%) は Java 8 のままでした。それ以来、この 2 つの LTS リリースバージョンの間でバランスが変化しています。現在、48% 以上のアプリケーションが実運用で Java 11 を使用しており (2020 年の 11.11% から増加)、Java 8 は僅差で 2 位、実運用でそのバージョンを使用しているアプリケーションの 46.45% を獲得しています。

Java 17 は、チャートには登っていませんが、リリースからわずか数カ月で、すでに Java 6、Java 10、Java 16 のリリースを上回っています。

Java 7 のサポートは 2022 年に終了しますが、まだ 1.71% のアプリケーションが実運用で Java 7 を使用しています。また、Java 6 についてはサポートが終了していますが、0.27% のアプリケーションで使用されています。Java 6 や Java 7 を使用しているアプリケーションのほとんどは、アップグレードされていないレガシーアプリケーションです。

各 Java LTS バージョンでの使用の割合



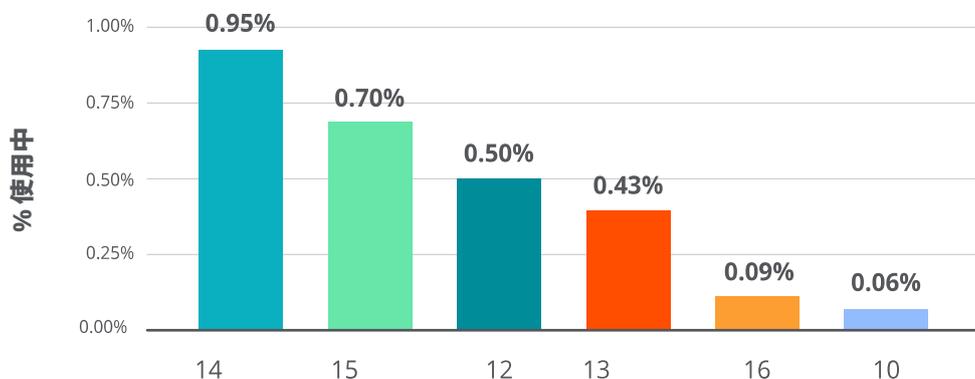
長期サポートバージョン

## Java 14 は、非 LTS バージョンで最も普及

Java 9 から、プラットフォームのリリースパターンが変更されました。Java は半年ごとに新しいバージョンが利用可能になりますが、次のリリースまでしかサポートされません。その意図は、新しい機能をより頻繁に利用できるようにするためです。

しかし、暫定的な非 LTS バージョンの Java の利用は、実運用中の LTS バージョンと比較して非常に低く、非 LTS バージョンの Java を使用しているアプリケーションはわずか 2.7% に過ぎません。Azul Systems のように、一部の非 LTS バージョンでパッチを出荷しているベンダーもありますが、ほとんどのベンダーはそうではありません。そのため、アップグレードに消極的なのだと考えられます。使用されている非 LTS の Java バージョンの中で、最も利用されているのは Java 14 で、最も利用されていないのが Java 10 と Java 16 です。

各 Java 非 LTS バージョンでの使用の割合



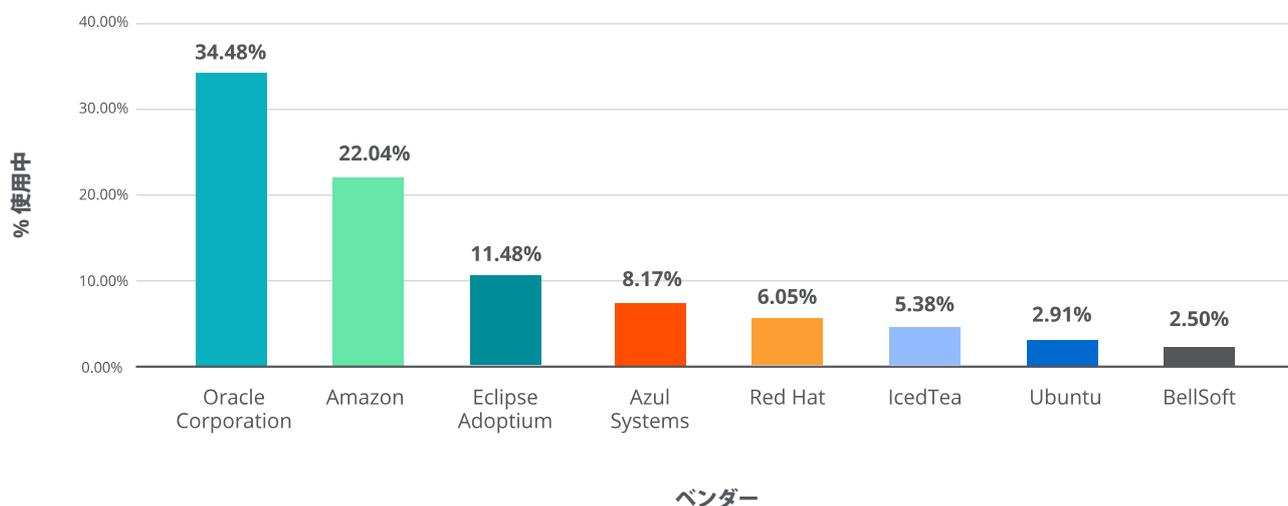
非長期サポート版

## オラクル人気は縮小し、アマゾンが台頭

近年、利用されている Java Developer Kit (JDK) ディストリビューションのソースが変化しています。かつて多くの開発者がオラクルからJDKを入手していたのに対し、OpenJDKプロジェクトでJavaがオープンソース化されたことで、豊富な選択肢が生まれました。

次の表は、オラクルのJDK 11 ディストリビューションのライセンスがより制限されるようになってからの（Java 17 によりオープンなスタンスに戻る前）オラクルのバイナリから離れる動きを示しています。2020 年には、オラクルが最も人気のあるベンダーで、Java 市場のおよそ 75% を占めていました。首位は維持しているものの、シェアは半減しています。アマゾンは 22%（2020 年の 2.18% から上昇）と劇的に上昇しました。

JDK ディストリビューションのベンダー別使用率



2021 年 11 月以降、この数字には、オラクル離れの動き以外にも、興味深い変化が見られるようになりました。Java 17 のリリース以前、Eclipse Adoptium とアマゾンは、このリストでほぼ同じ反対側に位置していました。

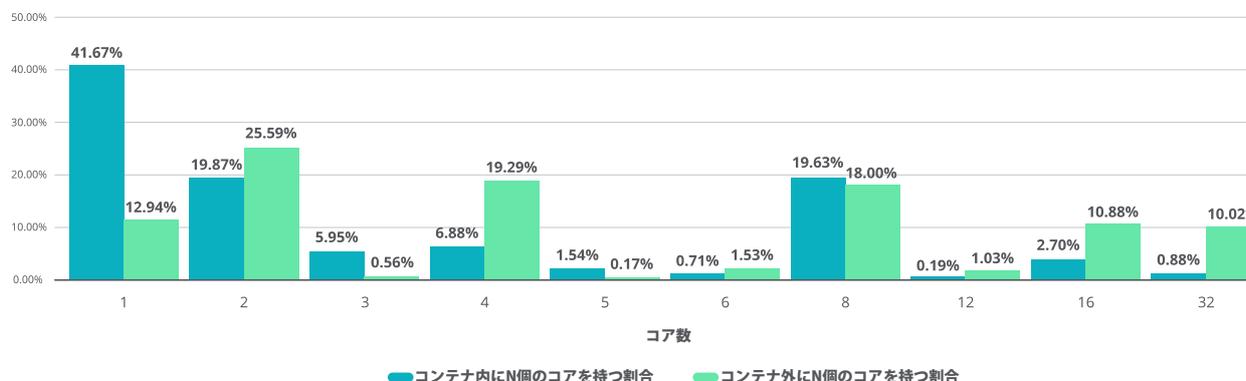
## 周辺のものすべてを実行するコンテナ

アプリケーションのコンテナ化は極めて主流になっており、New Relic の Java アプリケーションのデータもこの傾向を裏付けています。New Relic にレポートする Java アプリケーションの 70% 以上が、コンテナからレポートしています。

### コンテナでの計算設定

コンテナは、計算とメモリのリソースを割り当てる方法に影響を与えています。例えば、New Relic のデータでは、コンテナ内にある場合、4 コア未満で実行するアプリケーションの割合が非常に高いことを示しています。

## コンテナ内外で実行するアプリのコア数別の割合

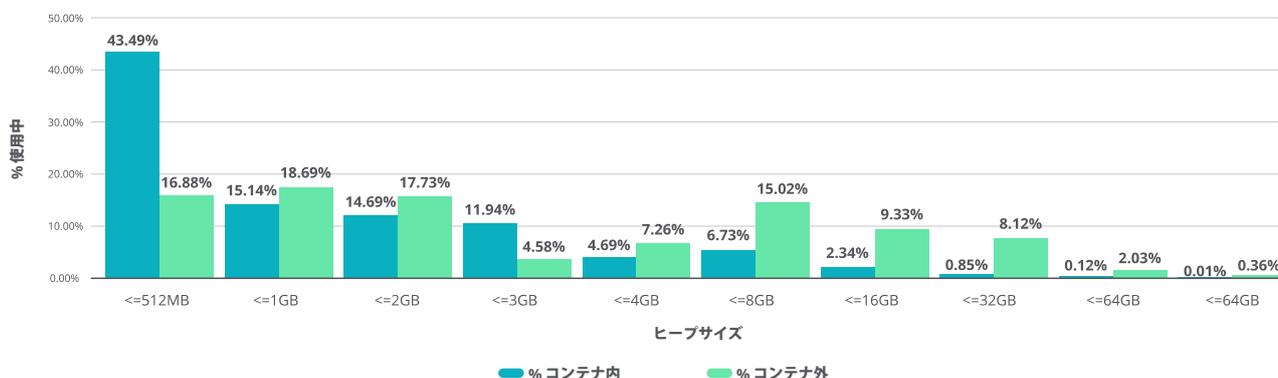


コンテナをデプロイすることが多いクラウド環境では、より小さいサイズで実行させることは非常に理にかなっています。しかし、この傾向は、アプリケーションによっては思わぬ問題を引き起こす可能性があります。特に、最近の Java 仮想マシン (JVM) のデフォルトの G1 ガベージコレクターによる同時実行のメリットの多くは、2 コア未満で実行すると消滅してしまいます。これらのシングルコアのインスタンスはすべて、シリアルコレクターを使用しており、そのパフォーマンスコストが発生している可能性があります。おそらく多くの人はそれに気づいていないでしょう。

## コンテナでのメモリ設定

メモリ設定の比較でも同様の傾向が見られ、コンテナ内のインスタンスが小さくなる傾向が見られます。New Relic のデータは、コンテナ化されたアプリケーションのうち、-Xmx または -XX:MaxRAMPercentage フラグによって JVM メモリの上限を明示的に要求しているのは約 80% に過ぎないことを示しています。バージョン 9 以降、JVM のコンテナ認識機能は、JVM が各コンテナで実行される唯一のプロセスである限り、これらのアプリケーションにとって、以前のような安全性の問題ではないことを意味します。

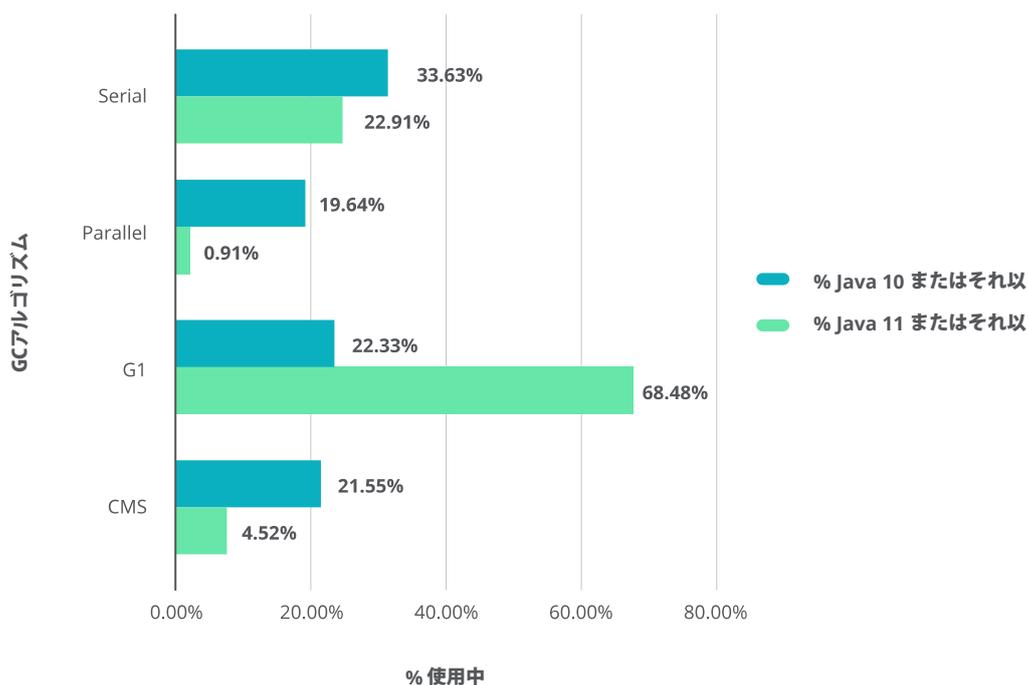
## コンテナ内外で実行するメモリ設定ヒープサイズの割合



## ガベージイン・ガベージアウト

JVM のパフォーマンスにおいて中心的な役割を考えると、果たすガベージコレクション (GC) は、依然としてコミュニティで多くの議論を呼んでいます。New Relic のデータでは、Java 8 以降ガベージコレクターの使用状況に明確な変化が見られています。これは、Java 11 以降で G1 コレクターのデフォルトが更新され、より高いパフォーマンスが得られることを考えれば驚くべきことではありません。

Java 10 以前と Java 11 以降で使用されている GC アルゴリズムの割合



G1 は、Java 8 から離れたユーザに好まれています。Java 8 以降に登場した他の実験的なコレクター (ZGC と Shenandoah) は、実運用システムでの使用は多くありません。どちらも最近まで実稼働状態ではなかったため、これはあくまでも予測です。

## 手法論

本レポートのデータは、2022 年 1 月に New Relic にレポートしたアプリケーションのみから取り出したものであり、Java の使用状況の全体像を示すものではありません。New Relic は、適切なデータを匿名化し、意図的に粗視化することで、Java エコシステムの一般的な概要を提供します。攻撃者やその他の悪意のある者の役に立つような詳細情報は、意図的にレポートから除外されています。

## New Relic について

New Relic は、オブザーバビリティのリーダーとして、優れたソフトウェアの計画、構築、デプロイ、実行に対するデータ駆動型のアプローチでエンジニアを支援しています。New Relic オブザーバビリティ プラットフォームは、エンジニアがすべての遠隔測定（メトリクス、イベント、ログ、トレース）を取得できる唯一の統合データプラットフォームを提供し、最も強力なフルスタック分析ツールとの組み合わせにより、エンジニアが「何か」を超えて「なぜ」を明らかにできるよう支援します。New Relic は、直感的で予測可能な業界唯一の使用ベース価格によって提供され、計画サイクルタイムの改善、変更失敗率の低減、リリース頻度の加速、解決までの平均時間の短縮を支援することにより、エンジニアに高い費用対効果をもたらします。これにより、AB InBev、Banco Internacional、Chegg、Gojek、Signify Health、TopGolf、World Fuel Services (WFS)、Zalora などの世界有数のブランドが稼働時間と信頼性を向上し、運用効率を高め、革新と成長を促す優れた顧客体験を提供できるようになりました。

今すぐ Java データのモニターを開始しましょう。[Java クイックスタートをインストールします。](#)