

# GraalVM

**Official Documentation:** GraalVM is a high-performance runtime that provides significant improvements in application performance and efficiency which is ideal for microservices. It is designed for applications written in Java, JavaScript, LLVM-based languages such as C and C++, and other dynamic languages. It removes the isolation between programming languages and enables interoperability in a shared runtime.

- یک ران تایم با پرفورمنس بالا که بهبود عملکرد برنامه ها را ارائه می دهد.
- برای مایکرو سرویس ایده آل است.
- از جاوا، جاوا اسکریپت، C، C++، پایتون، R، رابی و ... پشتیبانی میکند.
- سعی دارد ایزوله بودن زبانهای برنامه نویسی را از بین ببرد و بین این زبان ها همکاری ایجاد کند.

پس گرال وی ام یک رانتایم است، اما:

- رانتایم چیست؟
- گرال چه جور رانتایی است؟
- چگونه عملکرد بهینه تری از رانتایم های دیگر دارد؟
- چطور بین زبان های مختلف برنامه نویسی همکاری ایجاد می کند؟

- رانتایم یک محیط اجرا است؟! یا محیطی که برنامه مدت زمان اجرای خود را روی آن سپری می کند!
- گرال برای هر زبان برنامه نویسی، رانتایم پیشفرض زبان را شبیه سازی می کند.
- هسته اصلی گرال کامپایلر است. یک JIT کامپایلر است که به زبان جاوا نوشته شده است. و با کمک JVMCI با کامپایلر جاوا یا همان Hotspot Vm ادغام می شود. امکاناتی همچون کامپایل AOT را برای ترجمه زبان های خانواده جاوا به زبان ماشین دارد.
- با استفاده از ترجمه میانی! ( intermediate representation (IR ) بهبود های مستقل از زبان را انجام می دهد و برنامه را به زبان ماشین ترجمه می کند.

# امکانات و اجزای گرال

- **GraalVM compiler** : به زبان جاوا نوشته شده است. می تواند با HotSpot VM ادغام شود یا بطور مستقل اجرا شود.
- **Substrate VM** : با استفاده از کامپایل جلوتر از زمان (AOT) (JEP-295) برنامه های جاوا را به executable images تبدیل می کند.
- **Truffle** : فریمورک ایجاد زبان برای گرال
- **Sulong** : یک انجین برای اجرای زبان های خانواده LLVM بر روی گرال
- **GraalWasm** : یک انجین برای اجرای برنامه های WebAssembly بر روی گرال
- **Tools** : مجموعه ای از ابزارها برای دیباگ و مانیتور برنامه ها

## تأثیرات بر روی جاوا

- کامپایلر جاوا یا همان HotSpot VM درون خود کامپایلر بایت کد C2 را دارد. که به زبان C++ نوشته شده است. متأسفانه به مرور زمان بسیار پیچیده شده و نگهداری آن سخت است.
- بنابراین در آینده، گرال می تواند نسل بعدی کامپایلر برای HotSpot VM و جایگزینی برای کامپایلر C2 باشد

## نسخه های مختلف Graal Vm

دو نسخه Community و Enterprise را دارد. که در حال حاضر هر دو نسخه با جاوا 11 یا جاوا 17 هستند. در گذشته جاوا 8 نیز بود.

- نسخه Community براساس OpenJDK است.

- نسخه Enterprise براساس Oracle JDK است. (نسخه پولی)

- از جاوا 11 به بعد، کاپایلر گرال به عنوان یک ویژگی آزمایش به JDK اضافه شده است. برای فعال شدن آن باید دستور زیر را به دستورات استارآپ JVM اضافه کرد:

`-XX:+UnlockExperimentalVMOptions -XX:+UseJVMCICompiler`

```
<dependency>  
  <groupId>org.graalvm.js</groupId>  
  <artifactId>js-scriptengine</artifactId>  
  <version>20.2.0</version>  
</dependency>
```

```
String condition = "1 < 3";  
ScriptEngine engine = new ScriptEngineManager().getEngineByName("graal.js");  
Boolean result = (Boolean) engine.eval(condition);
```

# Reference

- <https://www.javacodegeeks.com/2020/12/introduction-to-graalvm-community-edition-revolution-or-evolution.html>
- <https://www.javacodegeeks.com/introduction-into-graalvm-community-edition-graalvm-for-jvm-applications.html>
- <https://www.graalvm.org/22.2/docs/getting-started/>