

面试官：“你了解过 JIT 么？”

前段时间学弟们面暑期，不知道是面腾讯还是面百度的时候，面试官提到了这个问题，面试完学弟就和我分享了这个问题，我觉得相当有价值，就做了一个总结，一起来看看吧～

JIT 是什么？

IT 是 Just-In-Time 的缩写，意味着即时编译器（Just-In-Time Compiler）。

在 JavaScript 引擎中，JIT 是一个可以将动态解释的代码转换为本机代码的技术。它通过根据程序的执行情况优化代码来提高代码的执行速度。这种编译方式可以大大提高 JavaScript 在许多场景下的性能，并且可以像已编译的代码一样执行解释器。

JIT 编译器在运行时实时编译 JavaScript 代码并将其转换为本机代码，然后在下一次代码执行时直接运行编译后的本机代码，而不需要再进行解释和编译。这可以显著提高代码的性能和响应时间。

JIT 一般在哪使用？

JIT 的主要应用在 JavaScript 引擎和一些高级语言的虚拟机中，如 Java 虚拟机、.NET 虚拟机等。

在 JavaScript 引擎中，因为 JavaScript 是一种动态类型语言，这意味着变量的类型在运行时才能确定。因此，传统的静态编译器不能解决 JavaScript 性能问题。引入 JIT 技术可以通过分析运行时的数据类型和执行情况，将程序的热点代码（频繁执行的代码段）编译成本机代码，从而提高 JavaScript 的性能和响应时间。

在 Java 和 .NET 等平台中，JIT 技术可以将字节码（bytecode）编译成本机代码，从而提高程序的执行速度。与 JavaScript 引擎不同，Java 和 .NET 等平台使用了静态类型检查，因此 JIT 技术的效果可能会更加显著。

总的来说，JIT 技术可以大大提高动态语言和虚拟机的性能，使它们的执行速度趋近于编译型语言。

JIT (JS) 的原理是什么？

1. 在 JS 引擎中增加一个监视器（也叫分析器）。监视器监控着代码的运行情况，记录代码一共运行了多少次，如何运行等信息，如果同一代码运行了几次，这个代码段就被标记为 warm，如果运行了很多次，就被标记为 hot
2. (基线编译器)如果一段代码变成了“warm”，那么 JIT 就把它送到基线编译器去编译，并且把编译结果存储起来。比如，监视器监视到了，某行、某个变量执行同样的代码、使用了同样的变量类型，那么就会把编译后的版本，替换掉这一行代码，并且存储。
3. (优化编译器)如果一个代码变成“hot”，监视器会把它发送到优化编译器中。生成一个更快速和高效的代码版本出来，并且存储。例如：循环加一个对象属性，假设它是 INF 类型，有限做 INF 类型的判断

4. （去优化）我们在使用优化编译器的时候，总是会有一些假设的，比如假设对象属性总以特定的顺序出现，但是如果这些假设无效，就说明我们之前的优化操作并不可取，所以需要去优化，去掉我们需要优化的方法。

tailwind 中的 JIT

Tailwind CSS JIT（Just-In-Time Compilation）是一个全新的 Tailwind CSS 编译器，与传统的 AOT（Ahead-Of-Time Compilation）编译器不同，它可以实时编译并生成最小的 CSS 样式表，让使用 Tailwind CSS 的开发人员能更快更高效地编写样式代码。

Tailwind CSS JIT 采用增量式编译方式，只编译你需要的样式，这种方式可以避免全局搜索和解析整个 CSS 类库，提高编译速度。同时，它还可以通过缓存中间数据（如样式变量和计算结果）来进一步提高编译速度。

除了编译速度的提升，Tailwind CSS JIT 还具有其他一些优点，例如：

1. 更小的 CSS 文件：Tailwind CSS JIT 只编译你需要使用的类名，因此它生成的 CSS 文件更小，可以减少加载时间。
2. 更灵活的响应性设计：Tailwind CSS JIT 可以根据屏幕宽度等参数动态生成不同的样式类，从而支持更灵活的响应性设计。
3. 更少的重复代码：Tailwind CSS JIT 可以通过样式组合和提取重复的样式属性来减少冗余代码。

总的来说，Tailwind CSS JIT 可以让使用 Tailwind CSS 的开发人员更快、更高效地编写样式代码，并提高网站的性能和响应速度。