Interestingly, some processors cache traces like these. Binary translation, JIT (Just-In-Time) compilation, and lazy binding of machine code are all used in virtual machines. The JVM also defines the format of Byte Code . The JVM's machine architecture provides for various challenges: FX 32 (x86-64) dynamic cache, incremental compilation, etc. Faster than you might think (heavy lifting done by the compiler). The tradeoff between load time and optimization quality.

The Common Intermediate Language (CIL) is the JBC analogue to the Common Language Runtime (CLR). Microsoft supports C#, F#, Visual Basic, Managed C++, and JScript. Explicit support for multiple programming languages is a feature of the JVM.

At startup, the JVM verifies that it satisfies various invariants. The JVM loads the given class file (which must have a .class file). The JVM allocates and initializes static data. The JVM verifies that all & only Java's built-in classes are referenced at compile time. The JVM may execute code via interpretation, compilation, or some combination of the two.

Recall how Java works. A virtual machine is a library that makes assumptions about how the virtual machine may execute code. A virtual machine may execute code via interpretation. A virtual machine may execute code via compilation, or some combination of the two.

The JVM's machine architecture provides for various challenges: FX 32 (x86-64) dynamic cache, incremental compilation, etc. Faster than you might think (heavy lifting done by the compiler). The tradeoff between load time and optimization quality. The JVM's machine architecture provides for various challenges: FX 32 (x86-64) dynamic cache, incremental compilation, etc. Faster than you might think (heavy lifting done by the compiler). The tradeoff between load time and optimization quality.

The JVM's machine architecture provides for various challenges: FX 32 (x86-64) dynamic cache, incremental compilation, etc. Faster than you might think (heavy lifting done by the compiler). The tradeoff between load time and optimization quality. The JVM's machine architecture provides for various challenges: FX 32 (x86-64) dynamic cache, incremental compilation, etc. Faster than you might think (heavy lifting done by the compiler). The tradeoff between load time and optimization quality.