

Compiladores Just-In-Time (JIT)

Os compiladores JIT são híbridos que usam tanto os conceitos de compilação tradicional (antes da execução) quanto de interpretação (durante a execução). Para isso, o compilador gera um código intermediário para ser executado em uma máquina virtual. Por esse motivo, as aplicações se tornam mais portáteis, já que o código intermediário é neutro à arquitetura dos computadores, entretanto o tempo de execução delas se tornam mais lentas, principalmente na inicialização do programa. Java e .NET são exemplos de linguagens que utilizam compiladores JIT.

Em Java, o código fonte é transformado em bytecode (código intermediário) e esse é interpretado pela JVM. No início da execução milhares de métodos são chamados e por isso há existe o atraso significativo. Durante a execução do programa alguns métodos podem ser compilados para serem salvos na memória afim de otimizar a execução, porém não são compilados na primeira chamada. O JIT interpreta os métodos até que ultrapassem um limite de execução e depois esse método é compilado em tempo de execução (por isso o nome) para serem acessados mais facilmente na memória.

OBS: entreguei essa atividade anteriormente em dupla com Pedro Antônio de Souza, resolvi reenviar pois refiz a pesquisa e a entrega anterior foi feita no CV do Pedro, então não custava nada...