**DOCUMENTAÇÃO**

*Benchmark Listando todas as cidades da França de um Banco de Dados:*

*1 CPU 386 requisições por segundo*

*4 CPU 2.904 requisições por segundo*

*4 CPU 6.382 requisições por segundo*

*Benchmark escrevendo um simples Hello World:*

*1 CPU - 3.252 requisições por segundo*

*4 CPU - 21.972 requisições por segundo*

*12 CPU - 32.361 requisições por segundo*

*Custo mensal*

*1 CPU, 512 MB - $5 / month*

*4 CPU, 8 GB - $80 / month*

*12 CPU, 32 GB - $320 / month*

**Escalabilidade**

O Node.js possui um módulo chamado cluster projetado para suportar uma alternativa de multiprocessamento.

O módulo de cluster permite gerar vários processos filhos que podem compartilhar uma porta comum, no entanto, a cada decisão de design, há compensações a serem avaliadas.

Nesse caso, nossas compensações estão associadas aos conceitos de multithreading e multiprocessamento. Threads compartilham o mesmo espaço de memória e a criação de novos threads não ocupa muito dos recursos do sistema.

Os processos são executados em alocações de memória separadas com uma cópia completa do programa, aumentando a sobrecarga de memória do aplicativo. Em situações em que a memória pode ser escassa, o Node.js pode ser a ferramenta menos eficaz para o código do aplicativo.

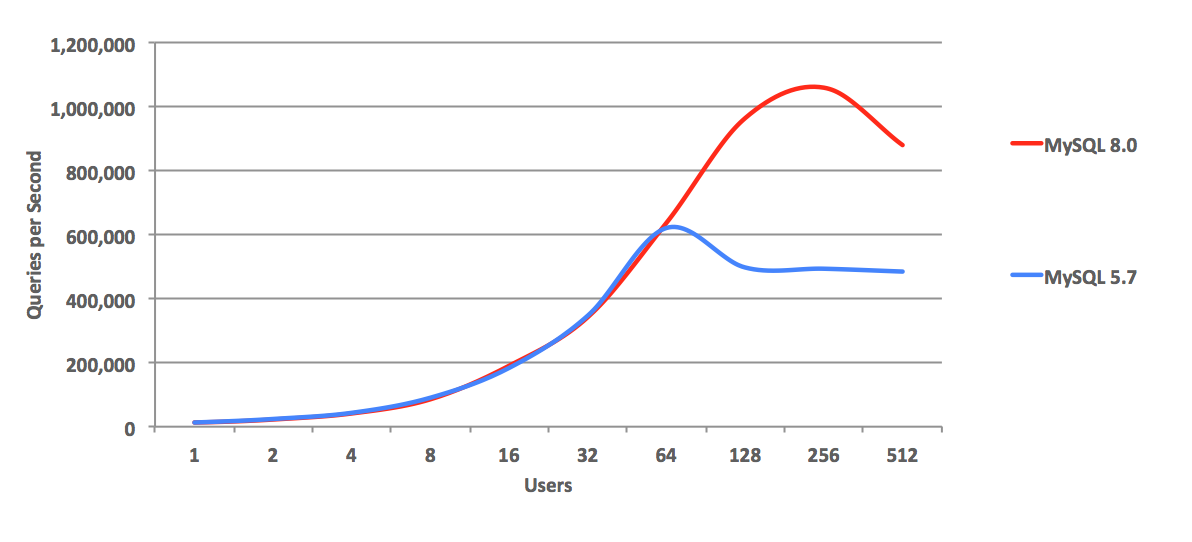
**MySQL**

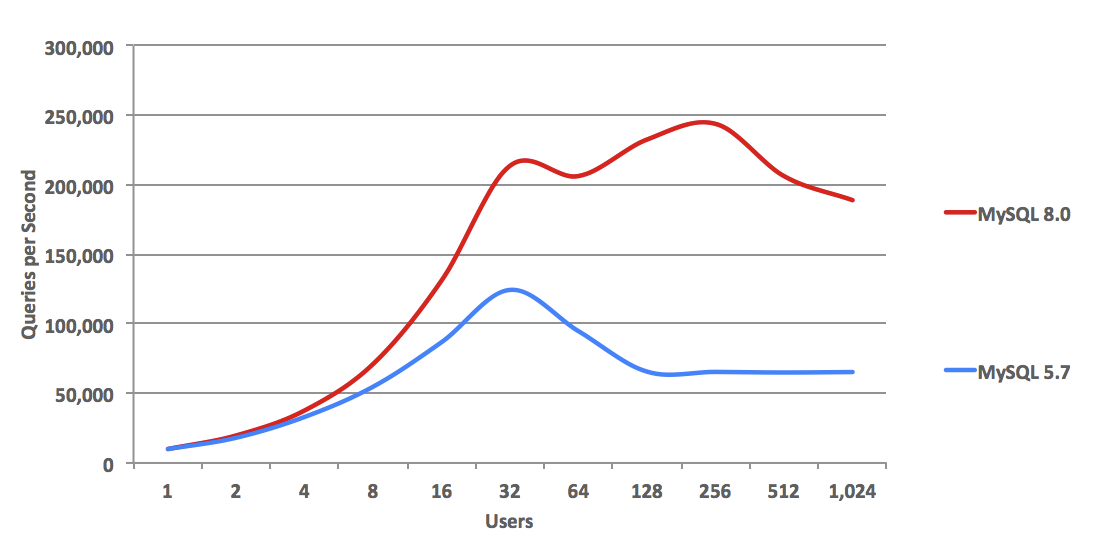
A performance do MySQL depende de 3 coisas:

* O *hardware* e sua disposição;
* Configuração
* Tipo de *query*

Com isso, não é possível medir a quantidade de requisições por segundo sem saber o espaço e a velocidade da RAM, o tipo de processador, a velocidade do processador, o espaço e a velocidade do SSD.

Além disso, uma má configuração pode arruinar tudo. A configuração deve ser otimizada para a sua exigência. É importante, também, saber a quantidade de interações com o Banco de Dados (BD), se irá envolver apenas um "SELECT" de uma tabela ou de várias, entre outras ações.

****

****