

Paralelismo no HTTP/1.1

As páginas Web têm cada vez mais dependências. Diversas imagens, diversos JavaScripts, CSSs etc. Dezenas, talvez centenas.

Por que o HTTP/1.1 tem um sério gargalo de performance ao baixar esses recursos em paralelo?

Selecione uma alternativa

A

A questão principal é que, no HTTP/1.1, o Keep-Alive não é obrigatório. Se habilitamos o Keep-Alive, os gargalos de paralelismo acabam e conseguimos fazer requests sem problemas. É uma boa prática habilitar o Keep-Alive sempre.

B

São diversos problemas. Como não temos gzip obrigatório, um grande volume de dados é trafegado, atrapalhando os demais requests. Outra questão é o limite de conexões de 6 que não pode ser aumentado. Além disso, por ser um protocolo em texto, os dados não podem ser enviados em paralelo.

C

As conexões HTTP só conseguem baixar sequencialmente. Envia-se um request, espera-se o response e o segundo request só pode ser enviado quando a resposta do primeiro terminar de chegar. Cria-se uma fila que tentamos aliviar abrindo 6 ou 8 conexões paralelas. Mas como as páginas têm dezenas de recursos, ainda temos uma fila.