

PERGUNTA 85 (PROXY)	PERGUNTA 85 (PROXY)
O que é um Proxy ?	Proxy (Procurador) , é um servidor de busca otimizada para a internet. Ele serve como um intermediário entre as máquinas locais e os servidores da internet, pois, uma vez que eles fizeram uma busca a um determinado servidor, eles guardam essa informação para ser acessada por outras máquinas que desejem fazer a mesma requisição, isso ajuda a não congestionar o tráfego de internet. Por isso também são conhecidos como Web Cache . Outro objetivo importante do proxy é proteger a rede pública de contato direto com a rede privada.
Quais as vantagens da utilização do Proxy?	<ul style="list-style-type: none"> - Diminuir o número de acessos a WEB, melhorando o desempenho da rede em velocidade, tráfego e fluidez do canal para transferência; - Acelerar aplicações que utilizem conteúdo Web, como por exemplo sistemas de CRM (Customer Relationship Management) e ERP (Enterprise Resource Planning); - Aumentar a Segurança na Internet, utilizando protocolos de controle de acesso diretamente nos Proxy's, como filtragem de conteúdo, anti-virus web, scanning SSL, entre outros;
O que é o método de Web Cache ?	Web Cache : é um método que consiste em armazenar os conteúdos de páginas Web como vídeos, imagens, arquivos estáticos, softwares e outros, na memória ram e/ou nos discos rígidos tanto em máquinas clientes como em servidores Proxy. Esse método é usado para ajudar a diminuir o tempo de acesso às páginas e transferência de arquivos, já que muitos deles são requisitados mais de uma vez.
O que é a Política de Reposição dentro do Web Cache?	Política de Reposição (Replacement Policy) : É um algoritmo existente nos servidores Proxy que determina qual objeto(s) deve ser despejado/removido do cache, a fim de abrir espaço para adicionar novos objetos mais procurados pelos usuários dos servidores. Esse método é usado para se ter um melhor aproveitamento do cache dos servidores Web proxy. Esses algoritmos de expiração podem se basear pelo tempo em que um arquivo está armazenado, no seu tamanho ou histórico de acesso.
Como funciona o Proxy baseado na política Least Recently Used ?	Least Recently Used (LRU) : É uma política baseada nos arquivos que são menos utilizado atualmente, ela vai manter no cache somente os objetos referenciados mais recentemente e remover aqueles mais antigos. O problema deste algoritmo é que ele não leva em conta o tamanho do objeto, uma vez que vários objetos pequenos podem ser substituídos por um objeto grande.
Como funciona o Proxy baseado na política Least Frequently Used ?	Least Frequently Used (LFU) : É uma política baseada nos arquivos utilizados com menor frequência, ou seja, ela vai remover os arquivos menos acessados e manter somente aqueles que tem maior número de acessos. Porém ela deixa de considerar o tempo do último acesso, mesmo que um arquivo tenha sido acessado a poucas horas, mas outro foi acessado mais vezes, ele vai priorizar aquele que foi acessado mais vezes em detrimento daquele que foi acessado poucas horas atrás.
Como funciona o Proxy baseado na política Size ?	Size : Essa política é baseada no tamanho dos objeto, ela remove aqueles arquivos que tem maior tamanho pelos de tamanho menor.
Como funciona o Proxy baseado na política GreedyDual-Size ?	GreedyDual-Size (GDS) : Essa política se baseia em duas influências sobre o arquivo, quantidade de acessos (hit rate) e o seu tamanho. Esse custo pode ser mensurado de acordo com a latência do pacote e quantas vezes os clientes estão o acessando em comparação aos outros pacotes da rede. São mantidos em cache os objetos menores que são referenciados mais vezes.
Como funciona o Proxy baseado na política FIFO ?	First in First out (FIFO) : Essa política se baseia em "o primeiro a entrar é o primeiro a sair", isto é, objetos são removidos na mesma ordem que entram. Não é muito usado em servidores cache proxy.
Desenhe como funciona o processo busca por através de um Proxy	<p>CLIENTES REQUISITANDO OBJETOS POR MEIO DE UM CACHE WEB</p> <pre> graph LR C1[Cliente] -- "requisição HTTP" --> P[Servidor proxy] P -- "requisição HTTP" --> S1[Servidor de origem] S1 -- "resposta HTTP" --> P P -- "resposta HTTP" --> C1 C2[Cliente] -- "requisição HTTP" --> P P -- "requisição HTTP" --> S2[Servidor de origem] S2 -- "resposta HTTP" --> P P -- "resposta HTTP" --> C2 </pre>

