

# Redes de Computadores

## Redes de Distribuição de Conteúdos

(breve motivação)

Departamento de Informática da

FCT/UNL

# Objetivo do Capítulo

- HTTP (*Hyper Text Transfer Protocol*) é o principal protocolo de aplicações distribuídas
- Hoje em dia generalizou-se a sua utilização mesmo quando o objetivo é só transferir informação multimédia
- O protocolo tem sido complementado com infraestruturas para aumentar a qualidade do acesso aos recursos informativos (*caches, proxies e reverse proxies*)
- Estas infraestruturas evoluíram para as chamadas **Redes de Distribuição de Conteúdos**

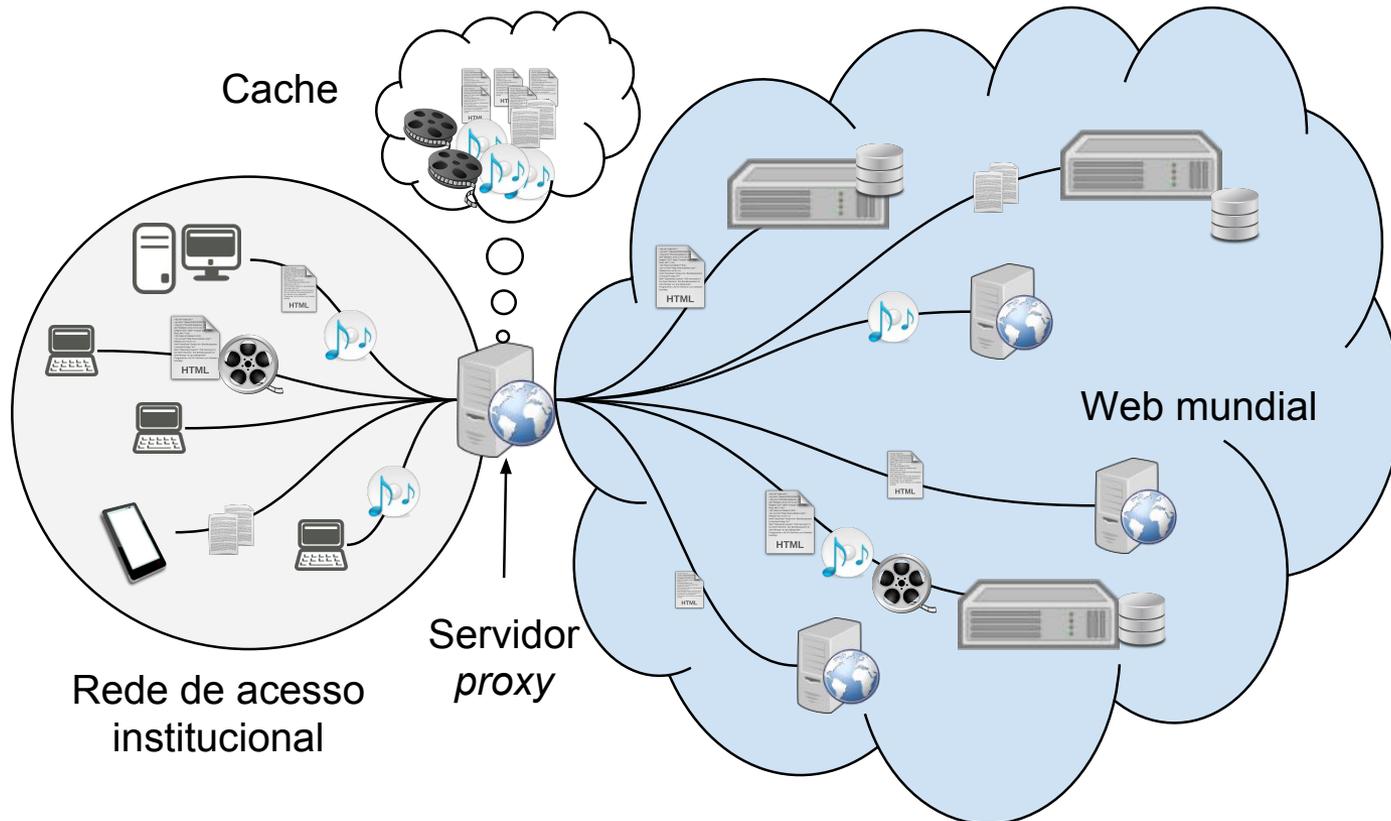
*A priori, finding similar laws governing Web page popularity, city populations, and gene copies is quite mysterious, but if one views all these as outcomes of processes exhibiting rich-get-richer effects, then the picture starts to become clearer.*

*- Autores: David Easley and John Kleinberg*

# Caching: porquê e como

- Porque fazer *caching*?
  - Muitos objetos não mudam (e.g. imagens, javascript, css)
  - Reduz o número de conexões e a carga do servidor
  - E portanto poupa a rede e aumenta a velocidade
- Podemos fazer *caching* para além do *browser*?
  - Porque não tentar?
- É mais fácil de dizer do que fazer
- É questionável se vale a pena

# Shared Web Caches (proxy server)



O utilizador parametriza o *browser* para usar o *proxy* e o *browser* envia os pedidos para o *proxy*. Se o objeto está na cache do *proxy* este envia a cópia *cached*. Senão, vai buscar o objeto ao servidor Web, faz *caching* dele e responde.

# Eficiência da Solução *Proxy*

- A eficiência da solução depende da taxa de serviço de objetos a partir da *cache* do *proxy* (*cache hit ratio*) = [ 0.2, ... 0.4, ..., 0.7 ]
- A poupança é proporcional ao *cache hit ratio*
- No entanto, caso o *hit-ratio* seja baixo, introduz maior prejuízo que benefício (uma interação suplementar pelo menos)

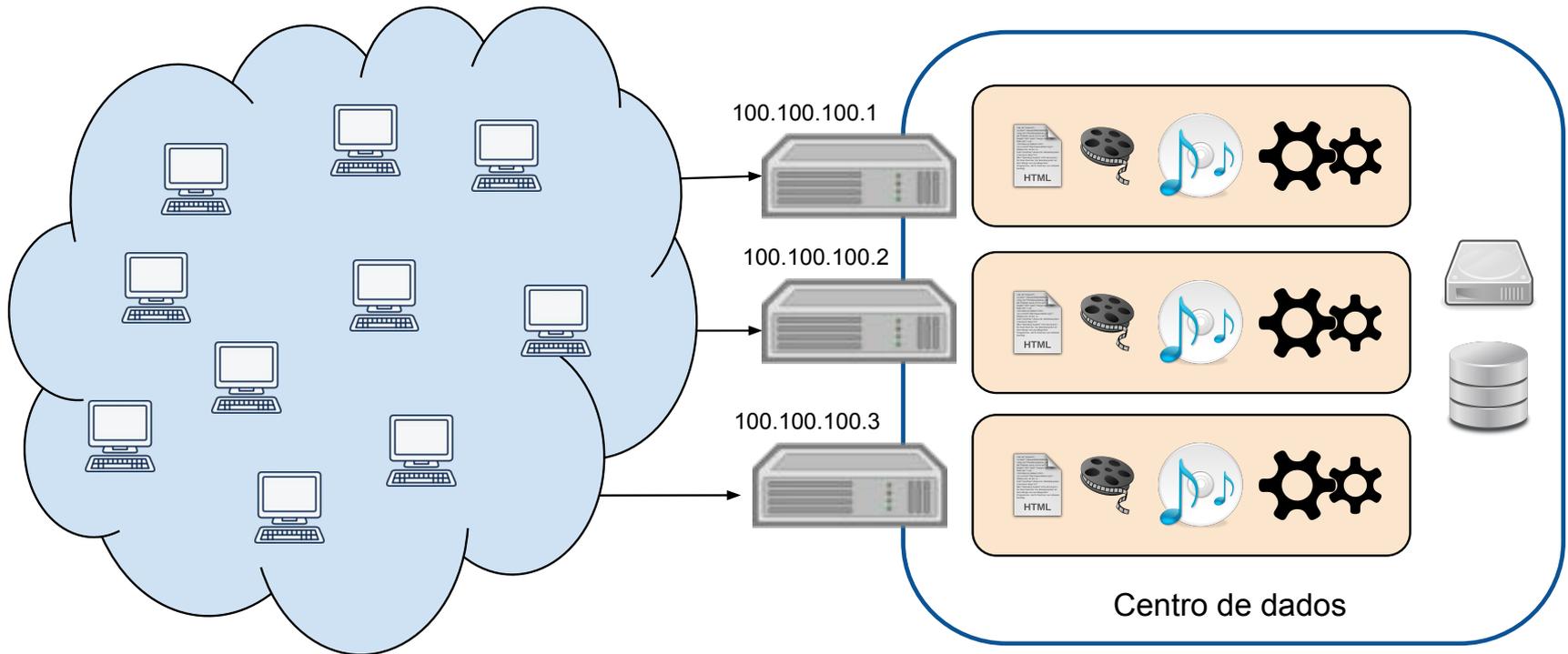
# Um saber de experiência feito

- *O caching* foi visto inicialmente como muito eficaz
  - Muitas extensões ao HTTP para o seu suporte
  - ... e sobretudo para validação das caches
- Era uma solução comum nos anos 90
  - Os ISPs e as empresas usavam *proxies* mas verificou-se que o ganho não era fantástico
  - Com efeito, se o cache *hit-ratio* for baixo, o *proxy* piora o acesso
  - A generalização do acesso por HTTPS diminui o cache hit ratio
- A generalização de *proxies* seria contra a forma intrusiva como a WEB é hoje usada do ponto de vista da privacidade (*cookies* e outras funcionalidades semelhantes)
  - cache-control: private, max-age=0, no-cache

# Aceleração do Lado do Servidor

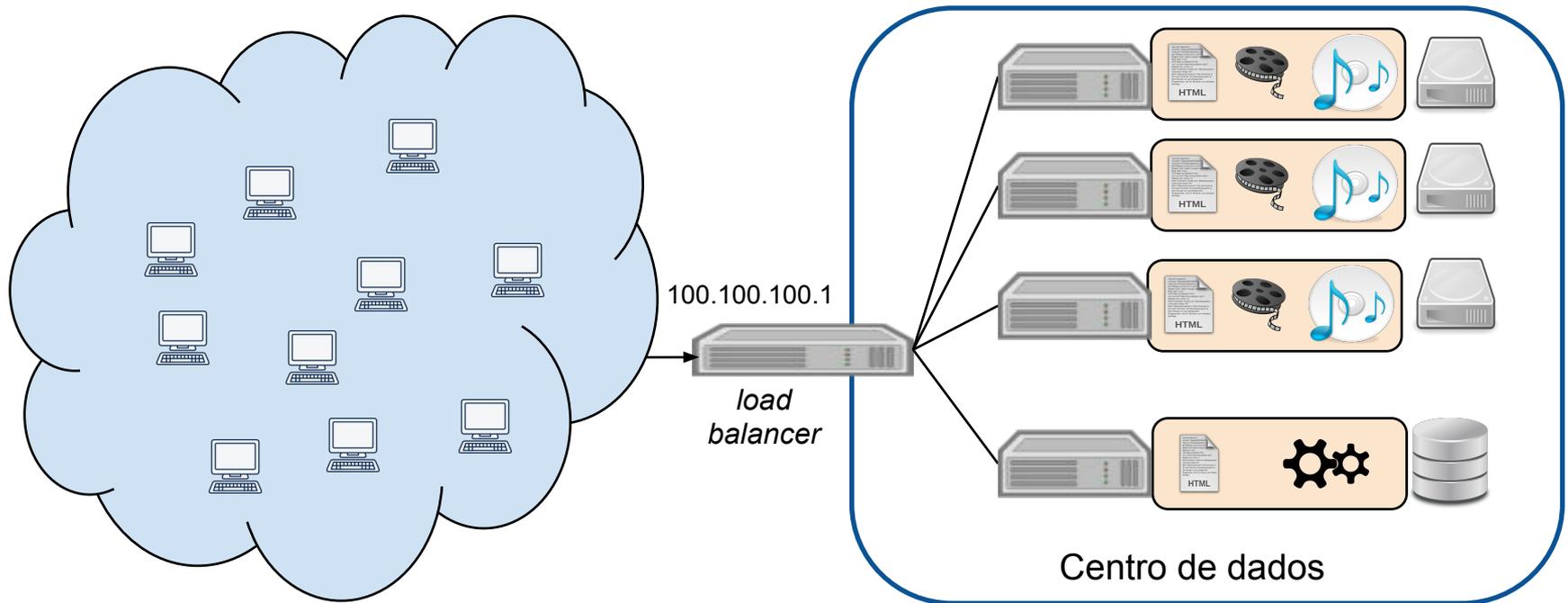
- Usar vários servidores com endereços IP distintos
- Usar distribuição de carga local mais inteligente
- Distribuição de carga em função da geo-localização
- Distribuição de carga em função da latência na rede
- Usar uma *Content Distribution Network* (CDN)

# Vários Servidores com Endereços IP distintos

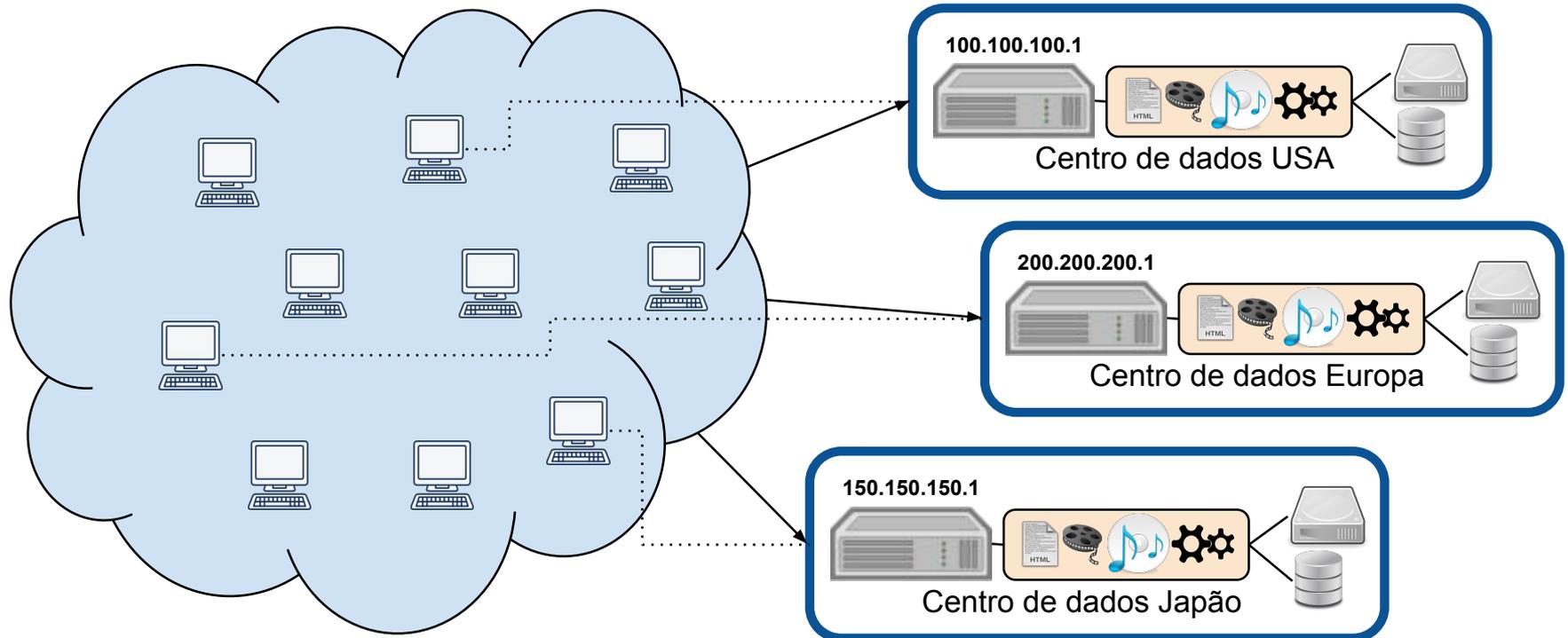


A implementação é feita com o apoio dos servidores de DNS

# Vários Servidores com Distribuição Local



# Geo-Distribuição para o Centro Mais Próximo

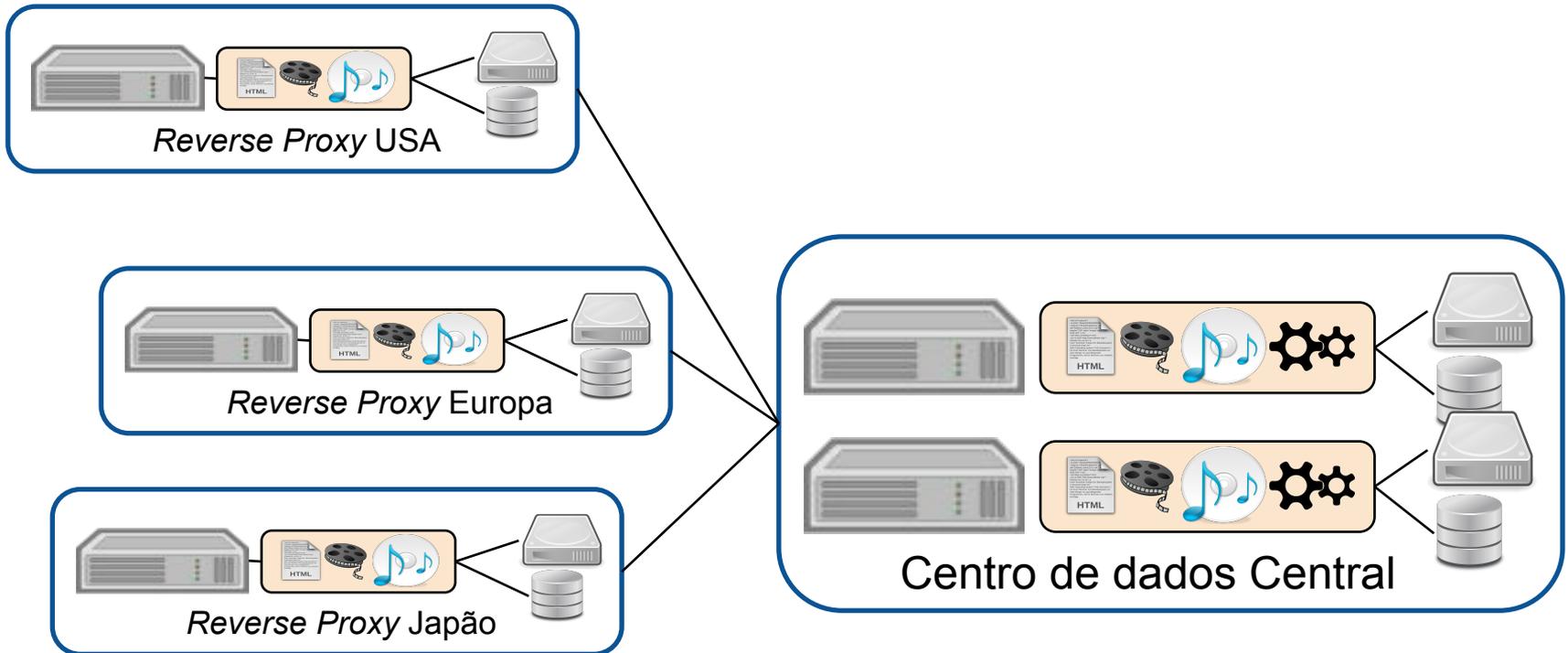


A implementação é feita com o apoio dos servidores de DNS que usam geo-localização dos endereços IP

# Caching e replicação proactiva

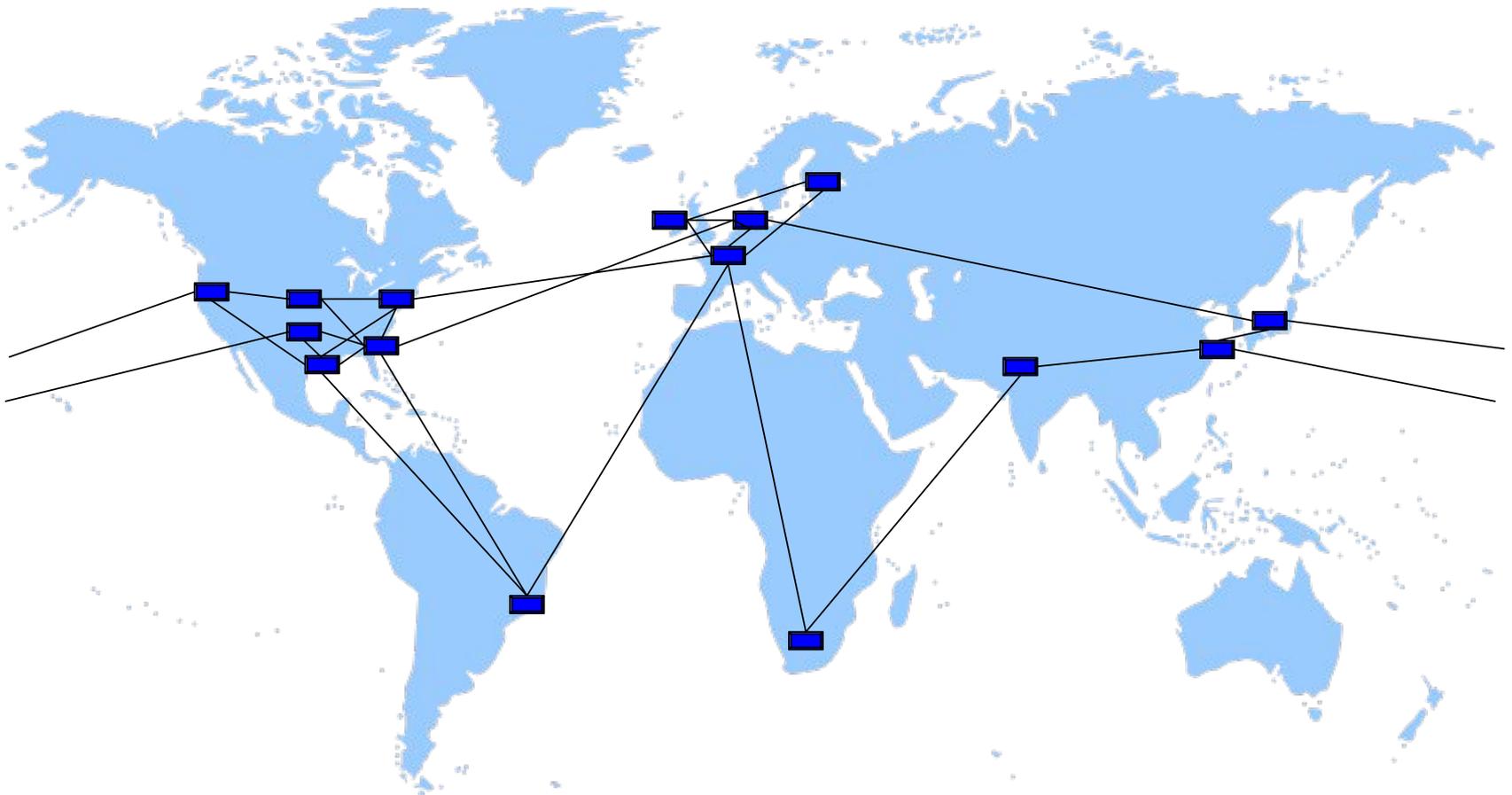
- **Motivações para por o conteúdo junto dos clientes**
  - Reduz o tempo de trânsito e o tempo de transferência
  - Reduz a carga sobre a rede e o servidor
  - Reduz o custo da transferência dos dados sobre a rede
- **Caching**
  - Necessita de verificar se o conteúdo foi alterado e alguns conteúdos não são *cacheable*
- **Replicação proactiva**
  - Replica os conteúdos de forma planeada e a priori
  - A atualização dos conteúdos não depende do HTTP nem de estampilhas temporais e passa a ser controlada pelos seus gestores
  - Pode replicar as aplicações que geram as respostas dinâmicas
- **Poder-se-ia implementar nos *proxies* dos ISPs e das empresas ?**

# Reverse Proxies do Gestor dos Conteúdos



A implementação é feita com o apoio dos servidores de DNS que usam geo-localização dos endereços IP. É ideal para conteúdos estáticos (e.g. filmes)

# Geo-Replicação



# Exemplo

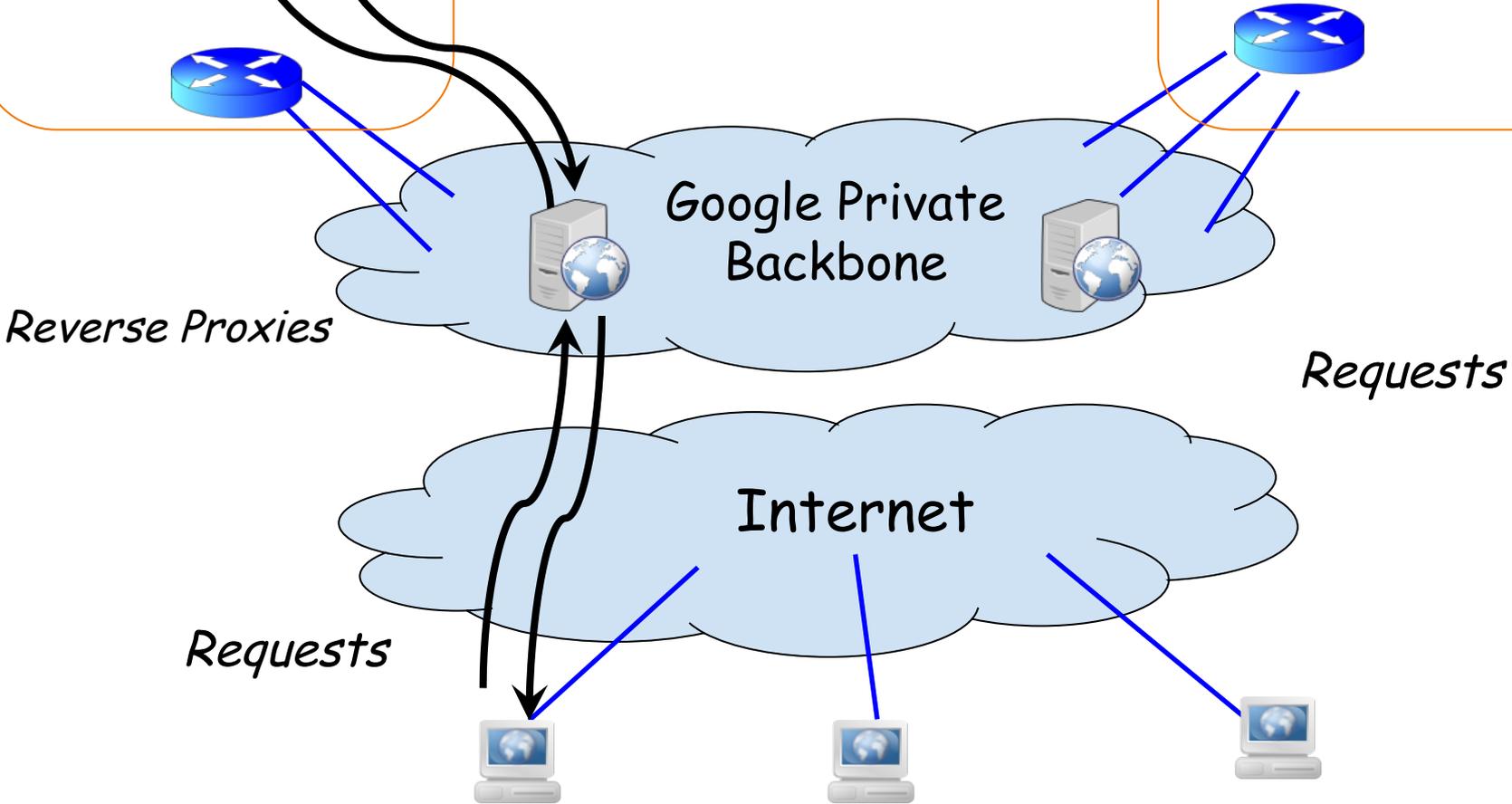
World Wide Data Centers



Servers



Servers

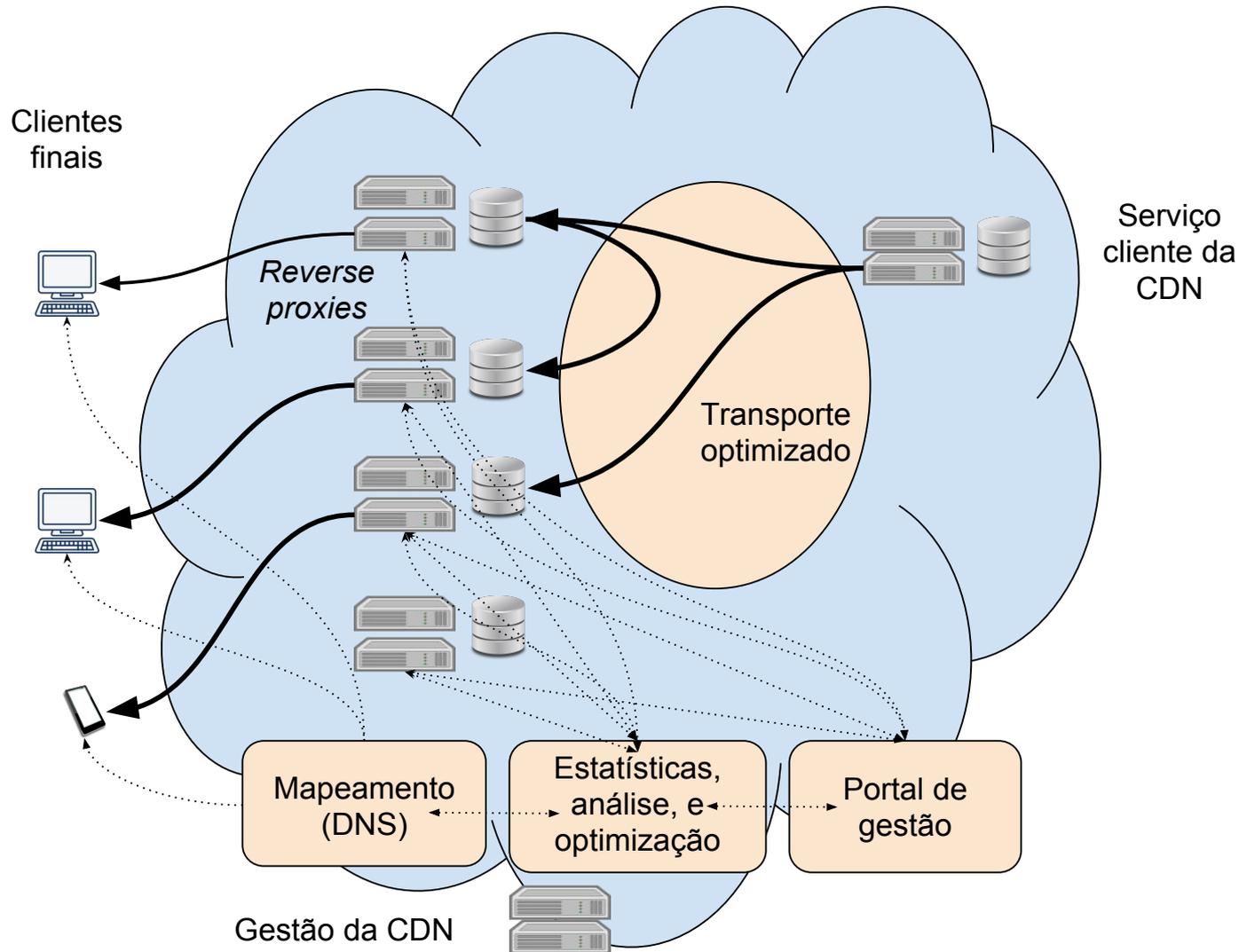


*Reverse Proxies*

*Requests*

*Requests*

# Content Distribution Networks (CDNs)



# Conclusões

- HTTP - Hyper Text Transfer Protocol é o principal protocolo que permite o acesso a recursos informativos e a aplicações na Internet e é o protocolo em que está baseada a WWW (*World Wide Web*)
- Dado ser idempotente (a operação pode repetir-se até ao infinito) ou sem estado, contempla facilmente *caching*, replicação, etc. Sobretudo quando os objetos são estáticos (imutáveis)
- As aplicações com grande número de utilizadores levaram ao desenvolvimento de CDNs dedicadas ou partilhadas, que podem incluir ou não grandes centros de dados, mas que contém sempre *reverse proxies*.