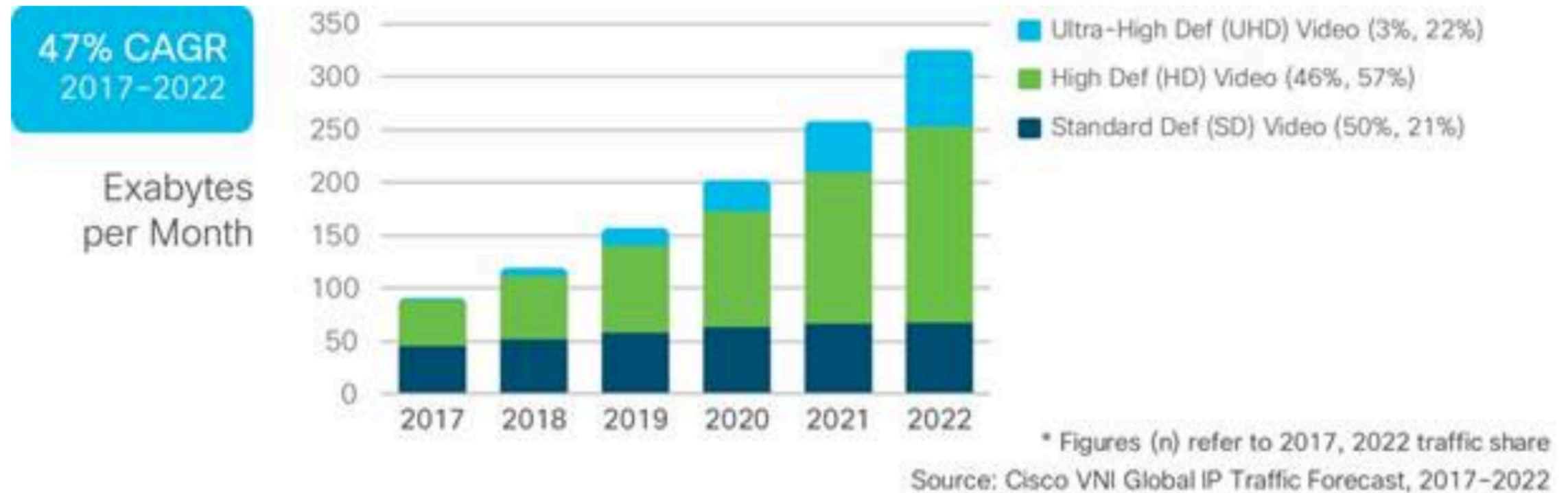# Content Delivery Networks

Ing. Tomáš Boros

# Agenda

- Introduction
- TCP / HTTP
- DNS
- Caching
- CDN
- Routing
- Demo

# Introduction

- Video content accounts for the most transferred data in consumer networks on a global scale. 6 out of 10 people prefer online video platforms to live TV

- Delivering and distributing data for a large set of users is challenging.

- **Globally, IP video traffic will be 82 percent of all IP traffic (both business and consumer) by 2022, up from 75 percent in 2017.** Global IP video traffic will grow four-fold from 2017 to 2022, a CAGR of 29 percent. Internet video traffic will grow fourfold from 2017 to 2022, a CAGR of 33 percent.

# Introduction



**Figure 7.** Global UHD IP video traffic

47% CAGR 2017-2022

Exabytes per Month

- Ultra-High Def (UHD) Video (3%, 22%)
- High Def (HD) Video (46%, 57%)
- Standard Def (SD) Video (50%, 21%)

\* Figures (n) refer to 2017, 2022 traffic share

Source: Cisco VNI Global IP Traffic Forecast, 2017–2022

# TCP / HTTP

- Early 90' 44% of Internet traffic via FTP
- Later HTTP
  - Methods (GET, POST, PUT, DELETE, HEAD etc.)
  - Status Codes. Listen on https://http.cat/
- HTTP Encapsulated in TCP
- TCP first introduced in 1981, (RFC 793)
  - TCP Connection oriented reliable protocol

# HTTP 0.9

- Only GET method
- No other headers
- Response as Clear Text (e.g. HTML)
- TCP Session Closed Immediately

# HTTP 1.0

- Additional request and response headers introduced
- GET, HEAD, POST
- Content-type response introduced
- RFC 1945
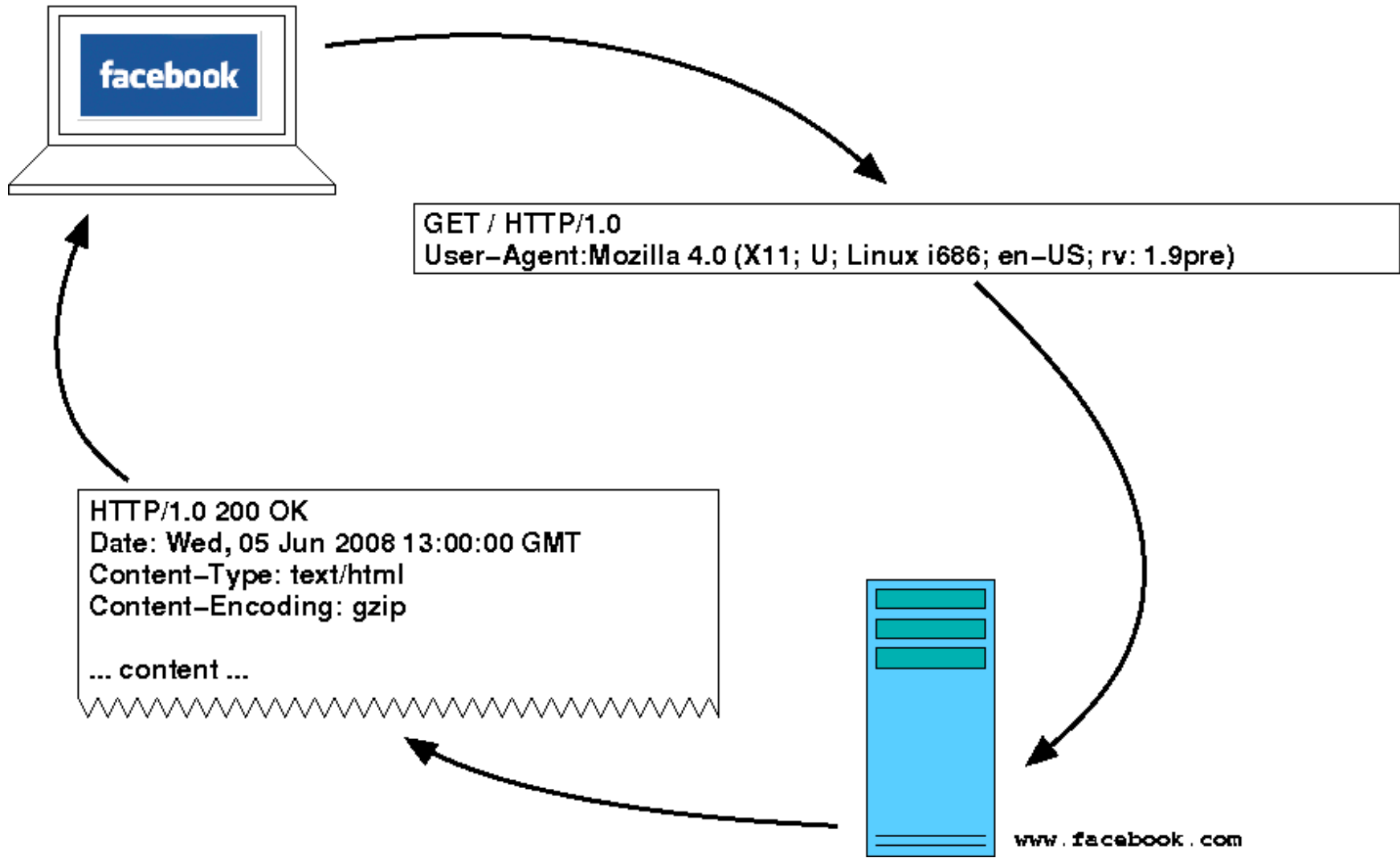- Did not became an official standard

# HTTP 1.1

- Released in 1997 as standard of HTTP
- RFC 7230-7237
- Persistent connections (keepalive)
- Pipelining (not on a single connection though)
- Cache control introduced

# HTTP 2.0

- Enables compression
- Multiplexing over a single connection
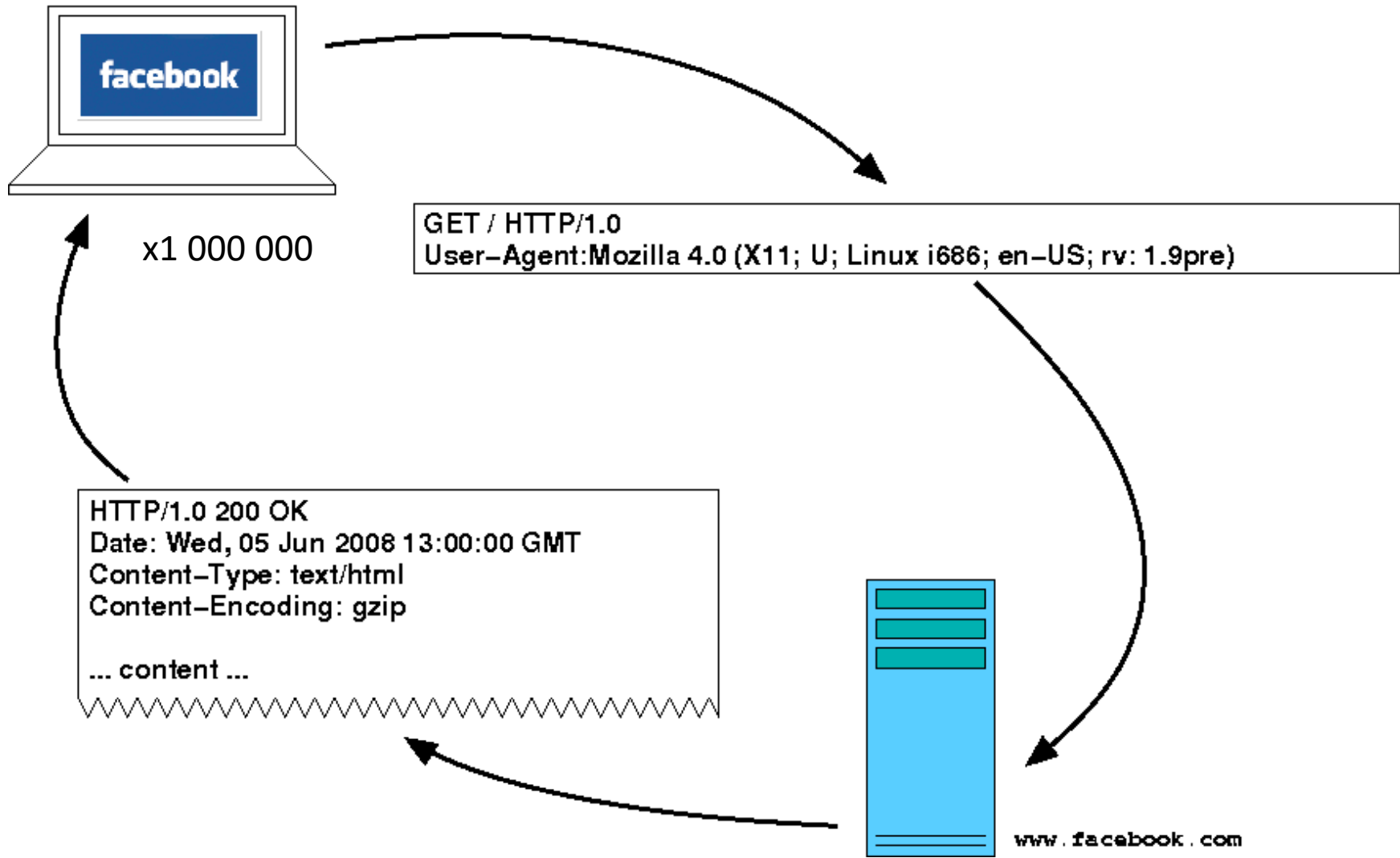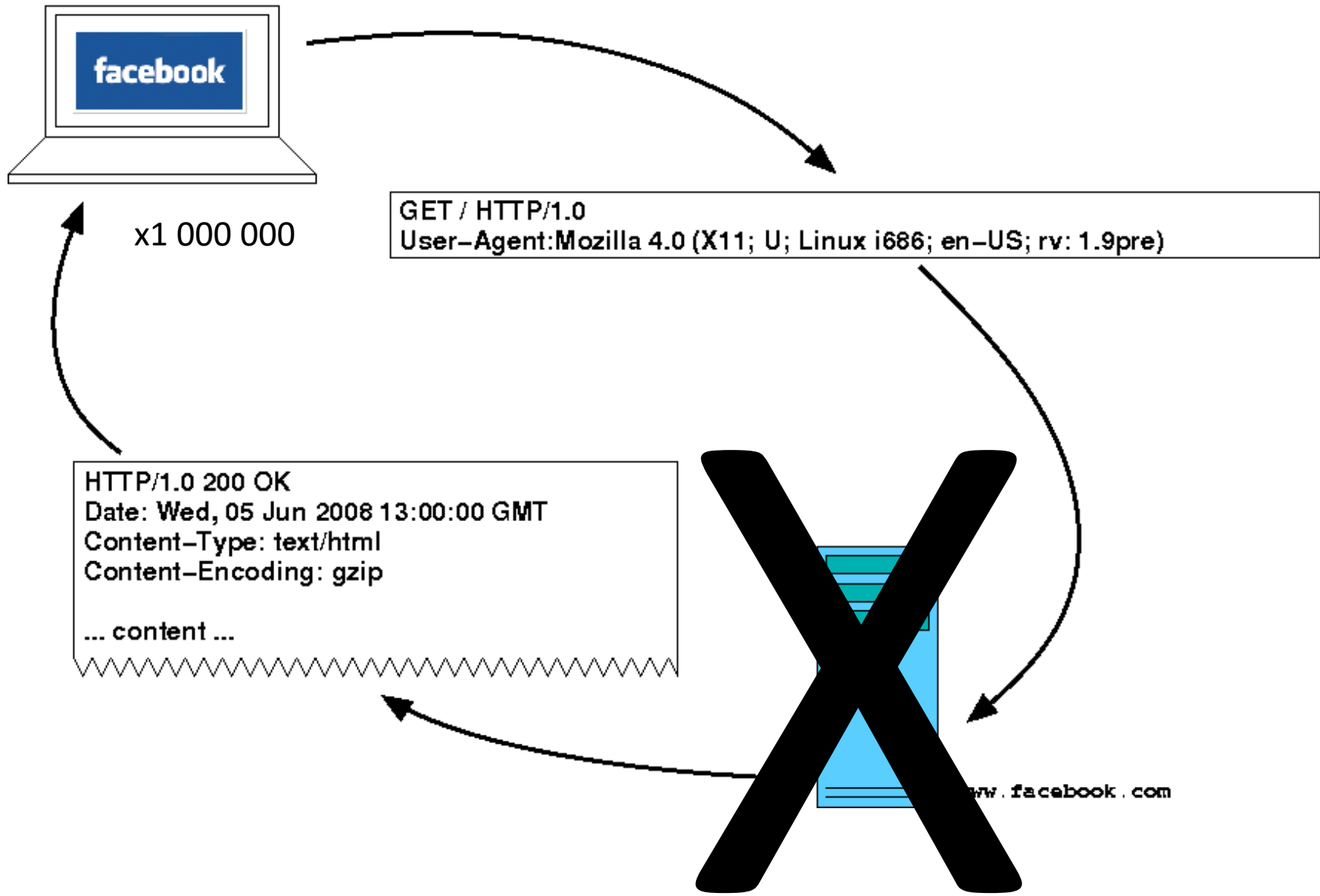- Flow control and various security updates

# HTTP 3.0

- Experimental
- Replaces TCP with UDP
- QUIC
- Developed by Google
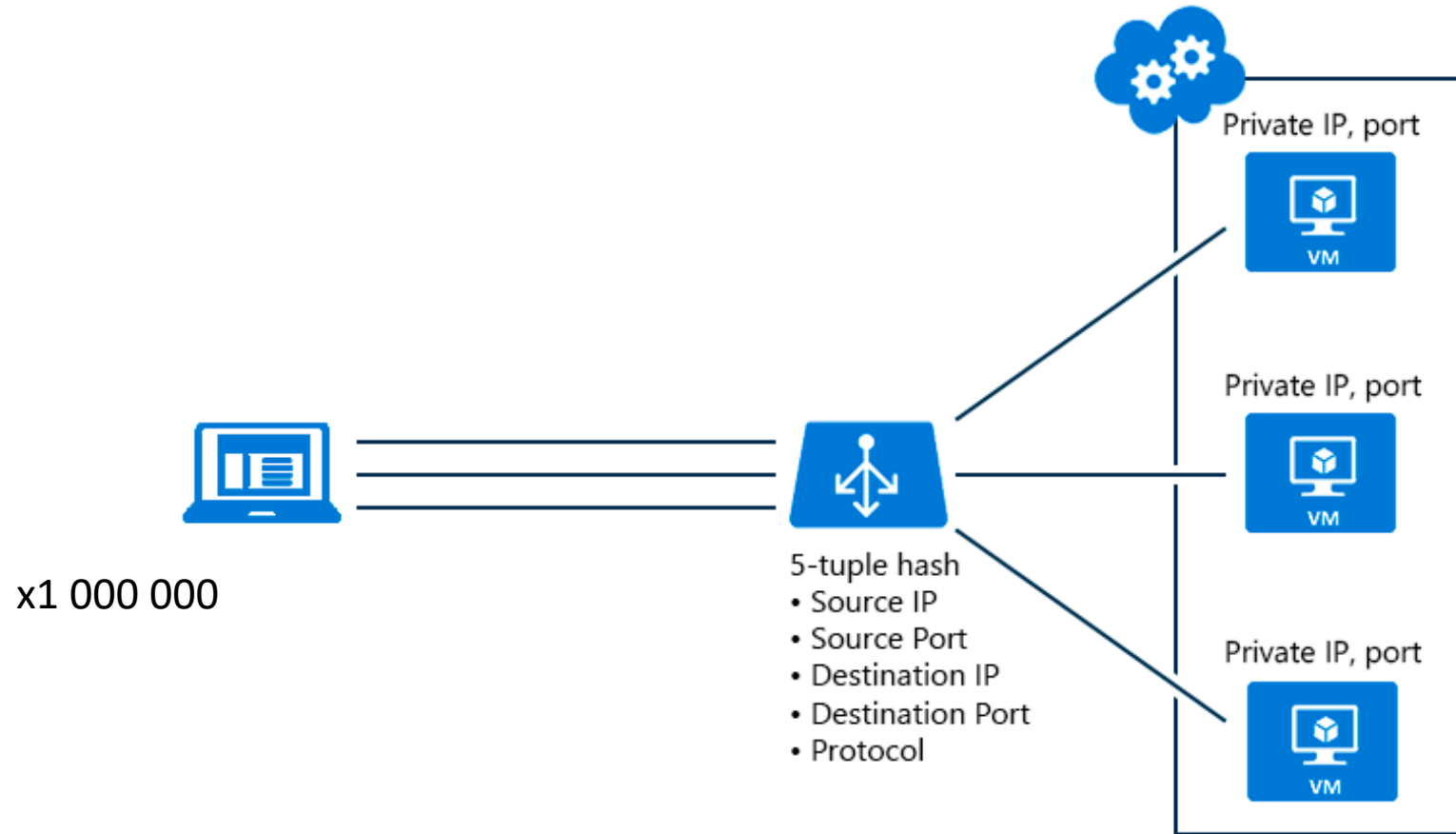- Enforced TLS
- Performance enhancements

GET / HTTP/1.0
User-Agent:Mozilla 4.0 (X11; U; Linux i686; en-US; rv: 1.9pre)

HTTP/1.0 200 OK
Date: Wed, 05 Jun 2008 13:00:00 GMT
Content-Type: text/html
Content-Encoding: gzip

... content ...

www.facebook.com

# DNS

- Resolves Domain names to IP addresses
  - Iterative vs Recursive

- DNS Record Types:
  - A
  - AAAA
  - NS
  - TXT
  - PTR

x1 000 000

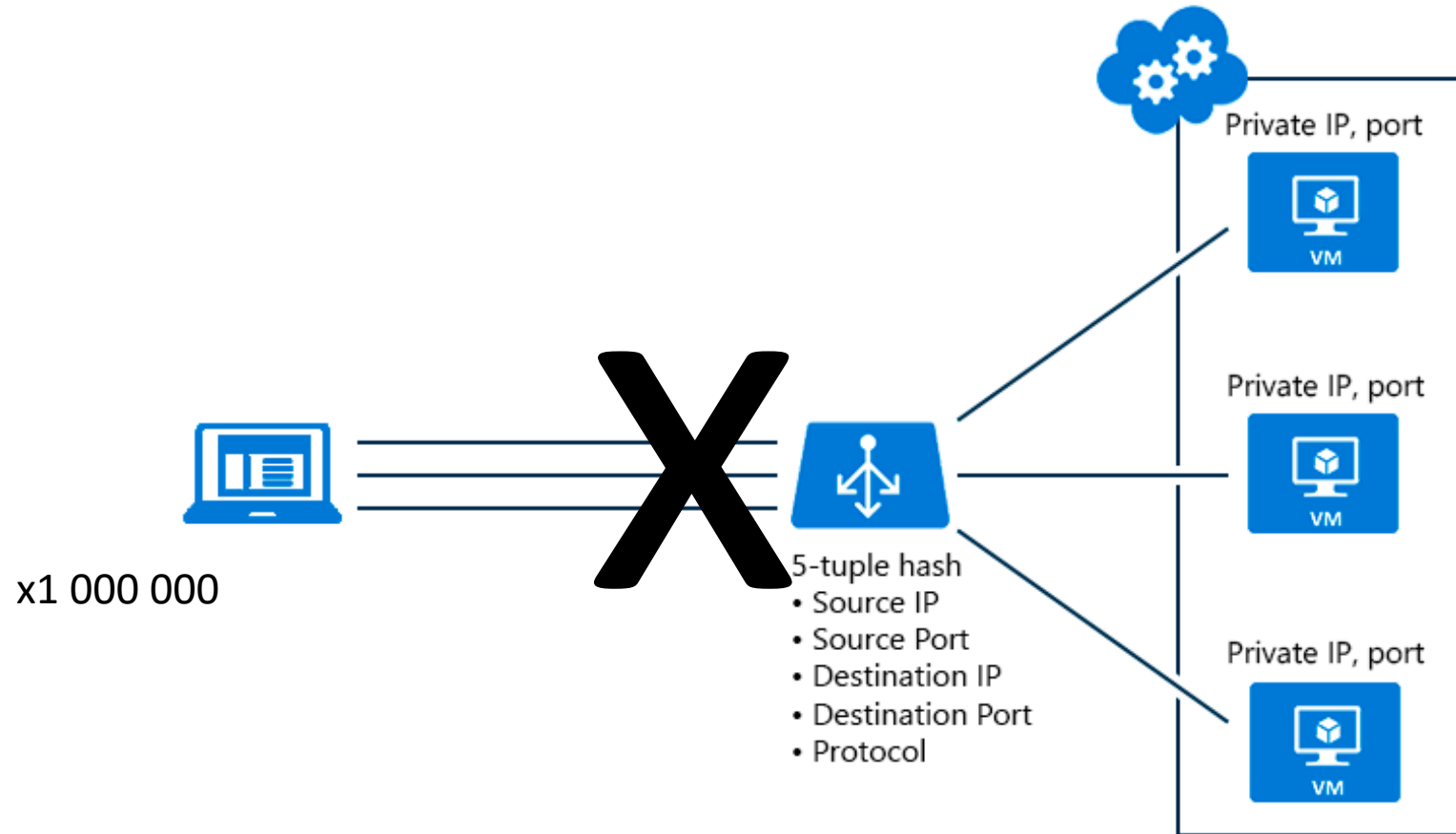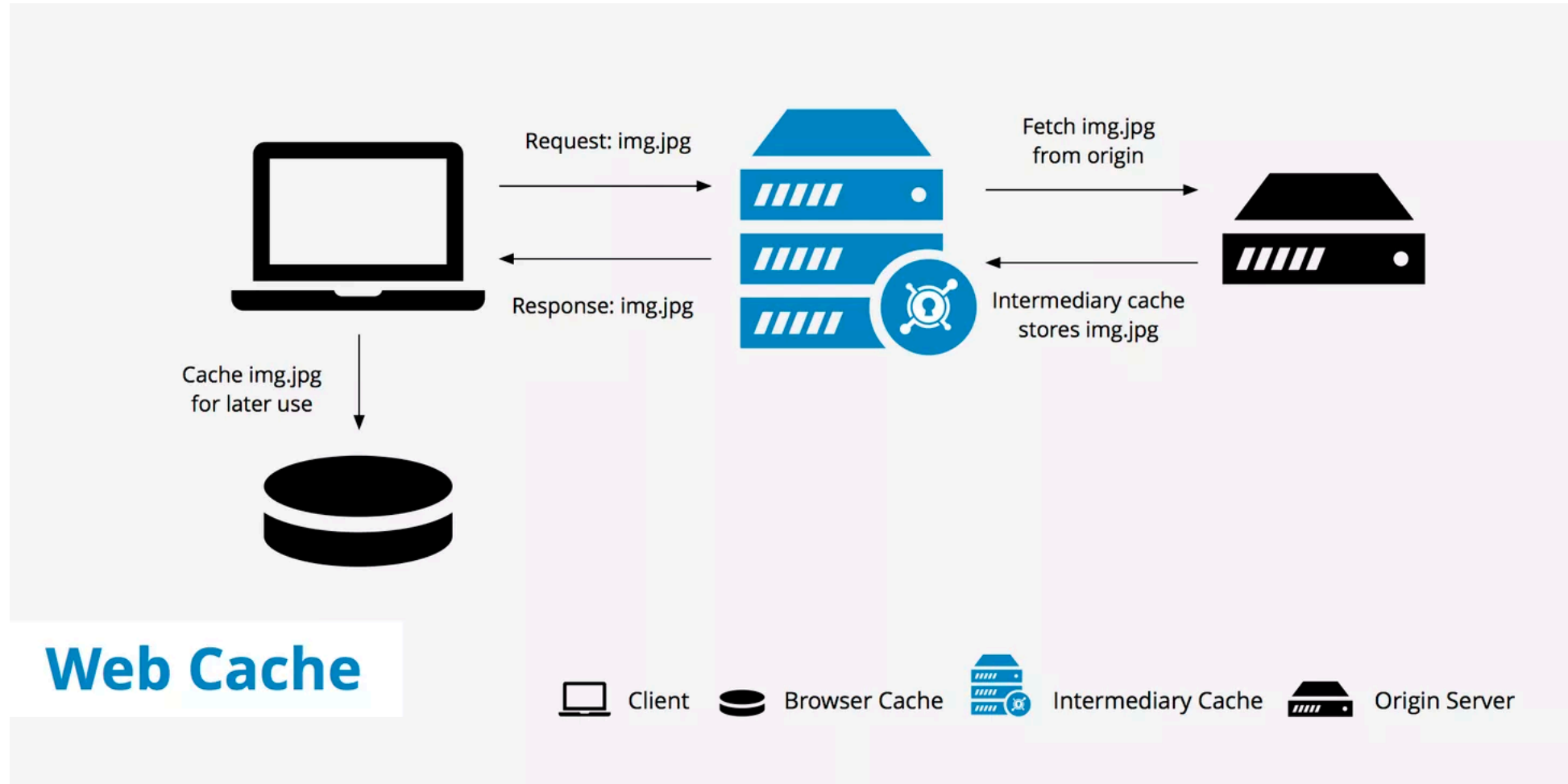GET / HTTP/1.0
User–Agent:Mozilla 4.0 (X11; U; Linux i686; en–US; rv: 1.9pre)

HTTP/1.0 200 OK
Date: Wed, 05 Jun 2008 13:00:00 GMT
Content-Type: text/html
Content-Encoding: gzip

... content ...

www.facebook.com

x1 000 000

GET / HTTP/1.0
User-Agent:Mozilla 4.0 (X11; U; Linux i686; en-US; rv: 1.9pre)

HTTP/1.0 200 OK
Date: Wed, 05 Jun 2008 13:00:00 GMT
Content-Type: text/html
Content-Encoding: gzip

... content ...

www.facebook.com

# Add More Devices



x1 000 000

5-tuple hash
- Source IP
- Source Port
- Destination IP
- Destination Port
- Protocol

Private IP, port

Private IP, port

Private IP, port

# Still the network is a bottleneck and SPF



x1 000 000

5-tuple hash
- Source IP
- Source Port
- Destination IP
- Destination Port
- Protocol

Private IP, port

Private IP, port

Private IP, port

Môže dojsť k zahlteniu siete

# Add Cache Servers to the network

# Storing content on a Cache server



HTTP GET /image.png

HTTP GET /image.png

www.example.com

User with Proxy

200 OK
Last-Modfied: Wed, 21 Oct 2017 07:28:00 GMT
Expires: Wed, 21 Oct 2017 08:28:00 GMT

Proxy cache server
CACHE MISS

200 OK
Last-Modfied: Wed, 21 Oct 2017 07:28:00 GMT
Expires: Wed, 21 Oct 2017 08:28:00 GMT

HTTP GET /image.png

www.example.com

User with Proxy

200 OK
Last-Modfied: Wed, 21 Oct 2017 07:28:00 GMT
Expires: Wed, 21 Oct 2017 08:28:00 GMT

Proxy cache server
CACHE HIT

# Storing content on a Cache server

- Content on cache might be in different states e.g.:
  - Fresh/HIT - Content served from cache
  - Valid/EXPIRED - Validated against origin, 304 might be returned
  - Invalid/MISS - Content re-fetched from origin

- Headers:
  - Cache-Control
  - If-modified-since

- https://developer.mozilla.org/en-US/docs/Web/HTTP/Caching

# Distribute them geographically
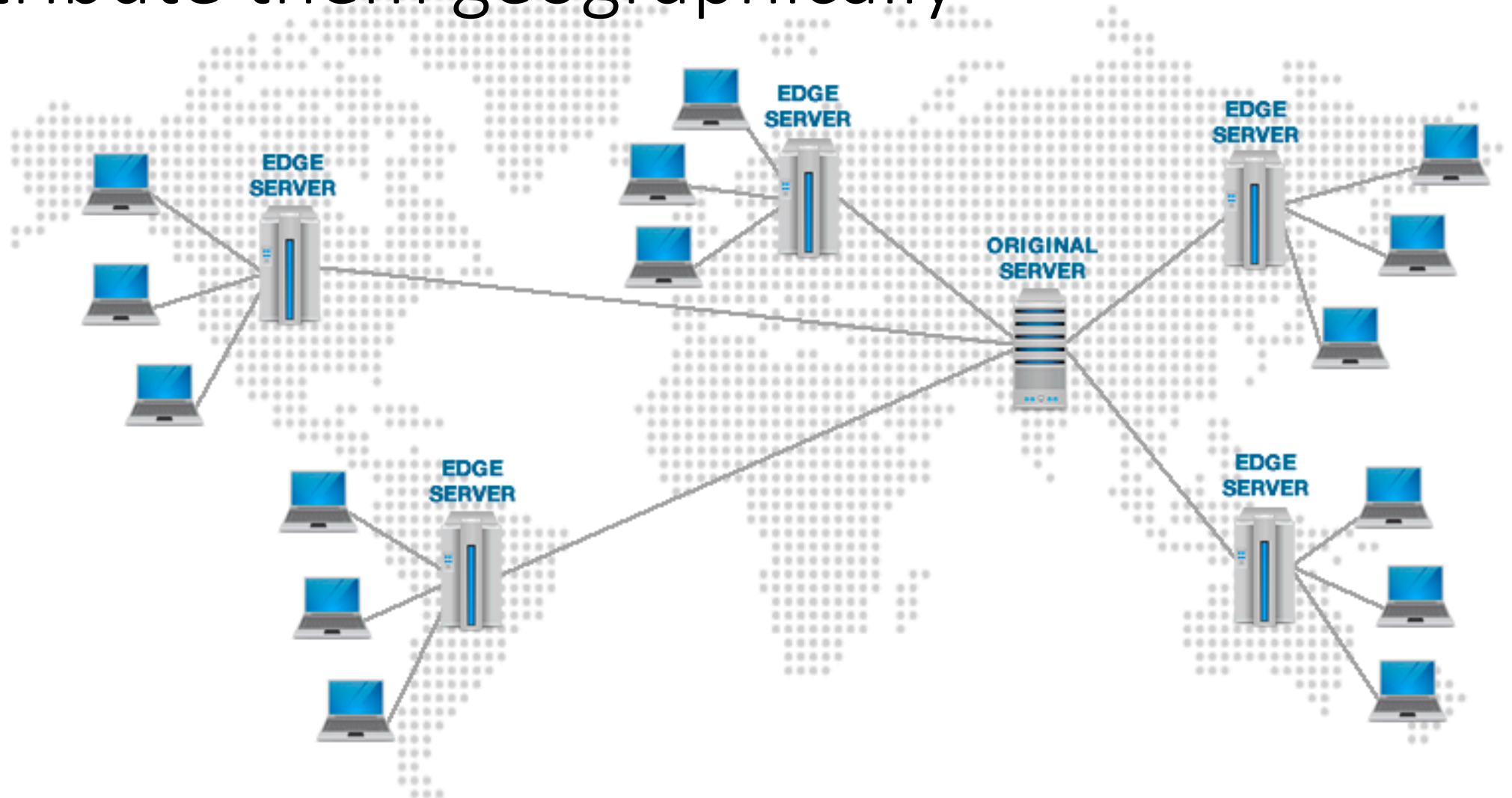
# CDN - Content Delivery Netowk

- User requests are routed to the closest / optimal cache server
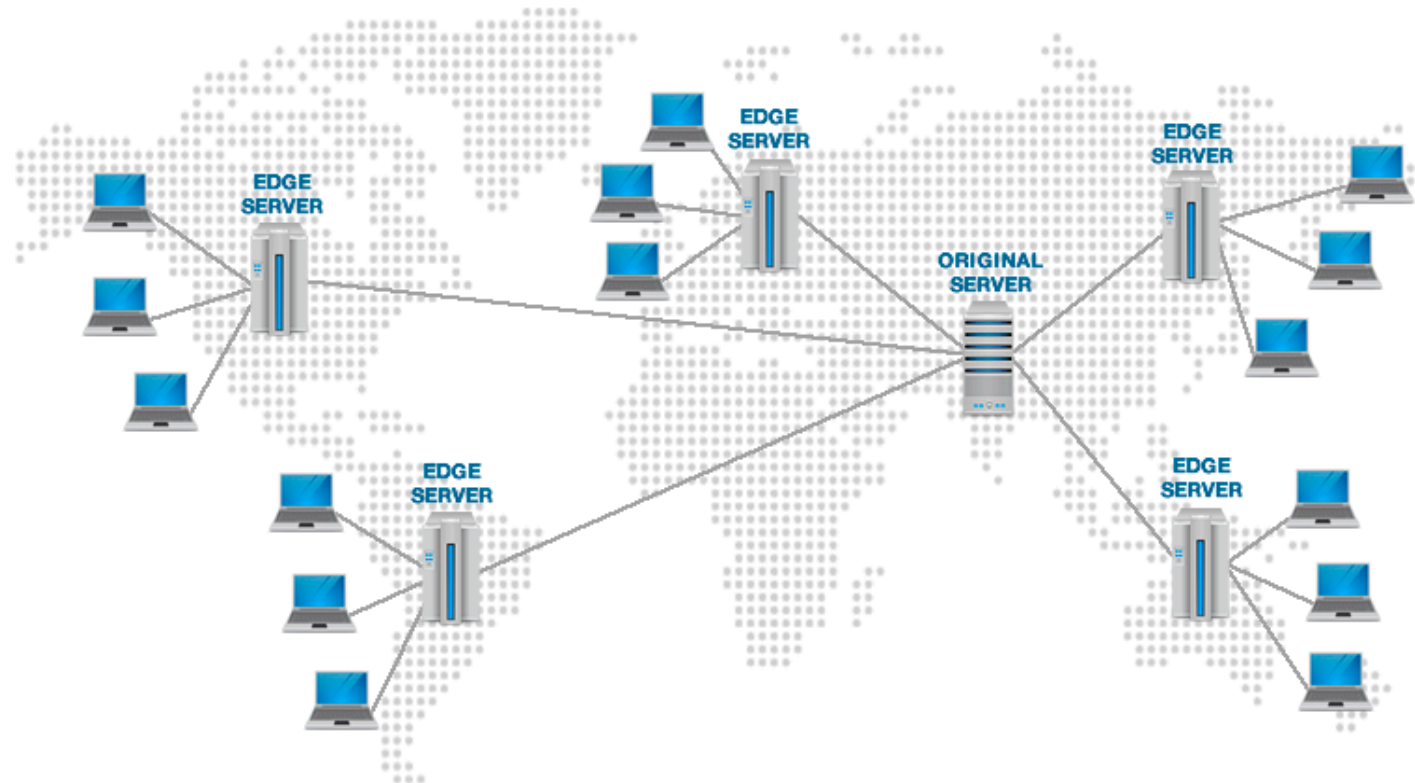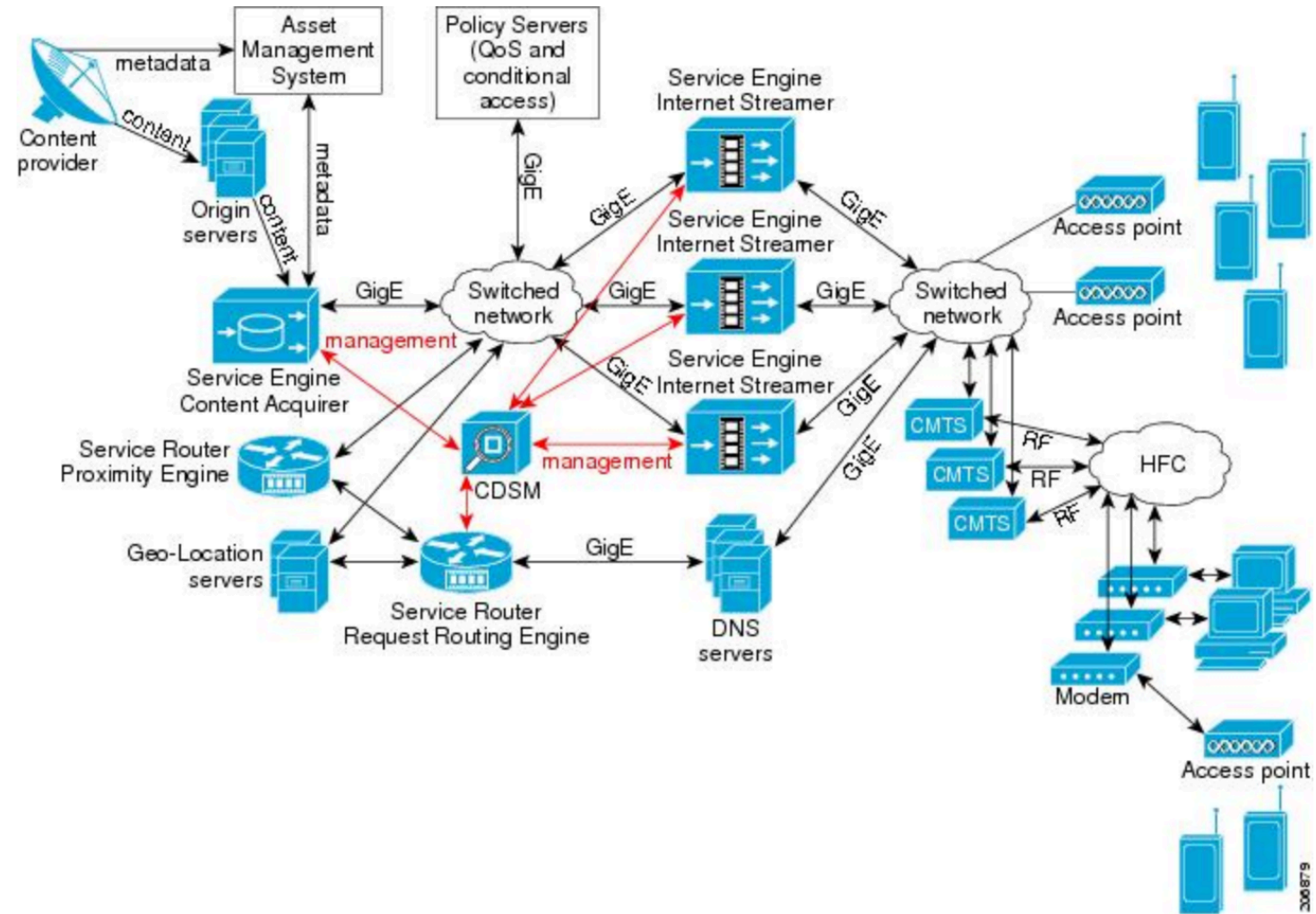
# CDN - Architecture / key components

- Nodes
  - Origin Server
  - Delivery Server
  - Request Router
  - Management
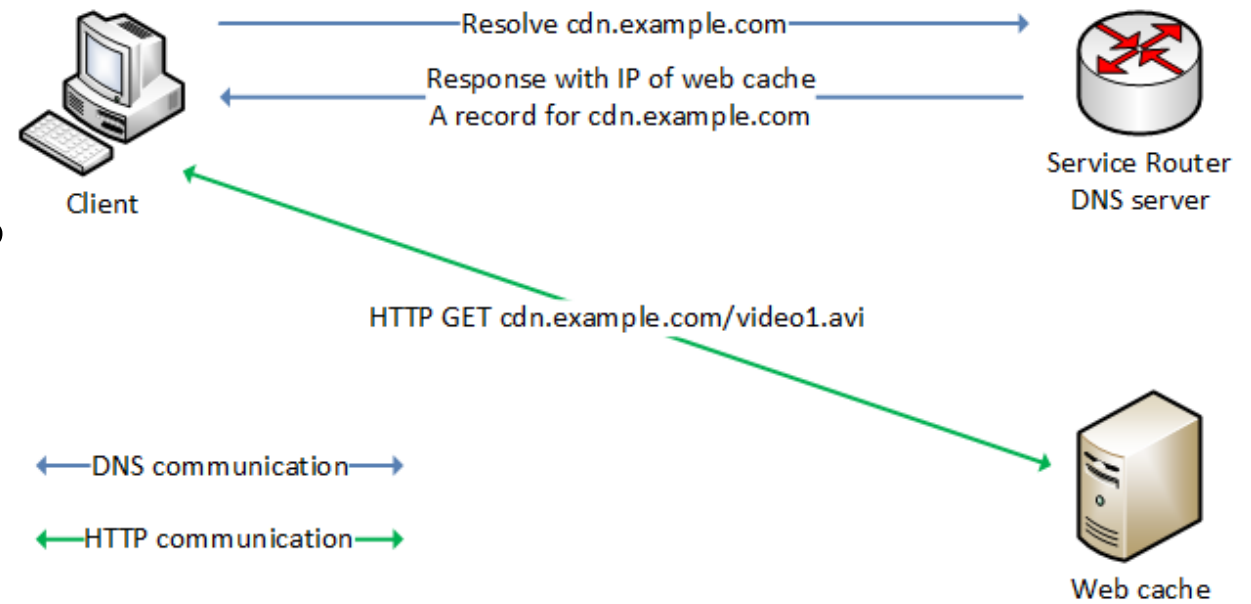


**Figure 1-1 High-Level View of the Cisco CDS**

# CDN - Request Routing

- Request Router Redirects requests to the cache servers
- RFC 3568:
  - DNS Based Request Routing
  - Application Layer Request Routing
  - Transport Layer Request Routing

# DNS Redirection

- Request Router as DNS server
- Request Router returns IP address of cache server directly

  - Fast
  - But not accurate
  - Redirection only based on Source IP

Resolve cdn.example.com

Response with IP of web cache
A record for cdn.example.com

Client

Service Router
DNS server

HTTP GET cdn.example.com/video1.avi

←DNS communication→

←HTTP communication→
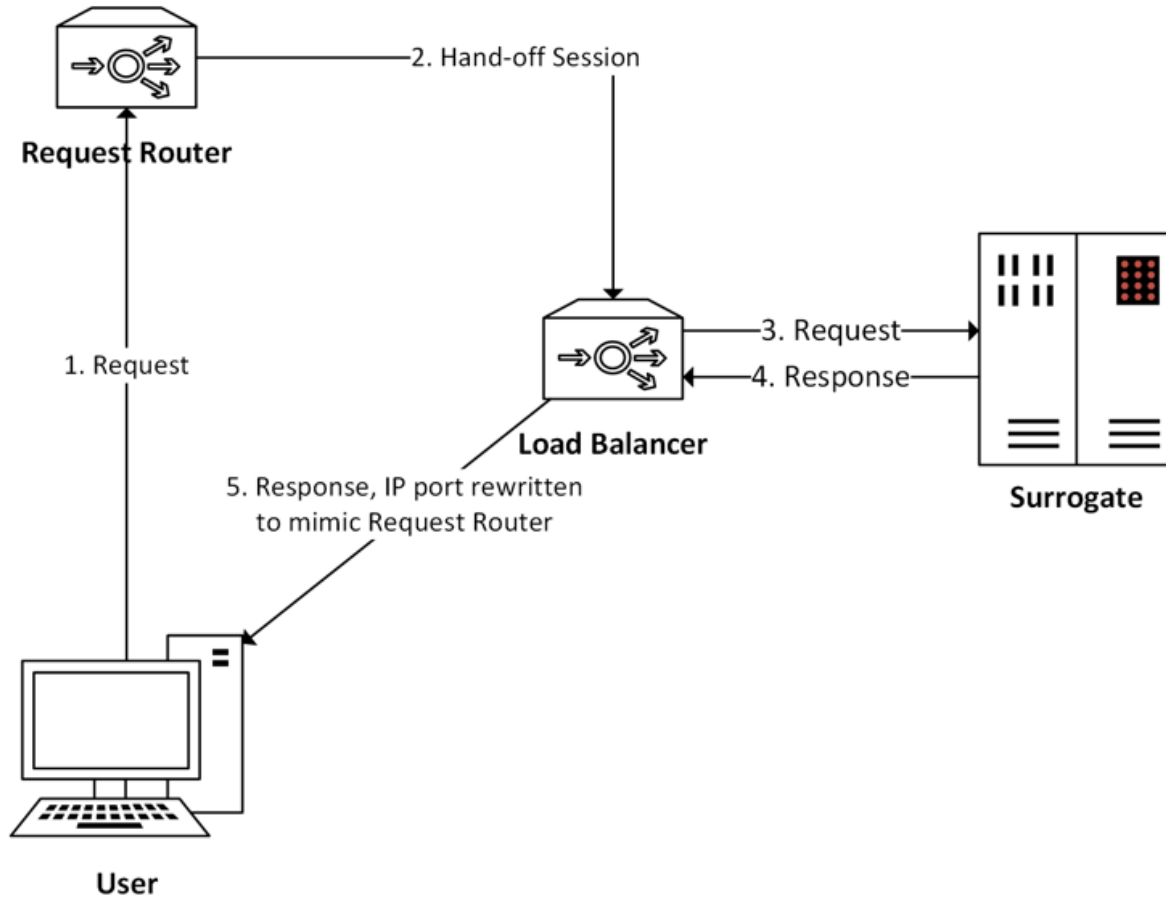
Web cache

# Application Layer Redirection

- Redirection occurs on Application Layer
- For HTTP it is 3xx status code
- URL Rewriting

  - Slower, required client to handle redirection message
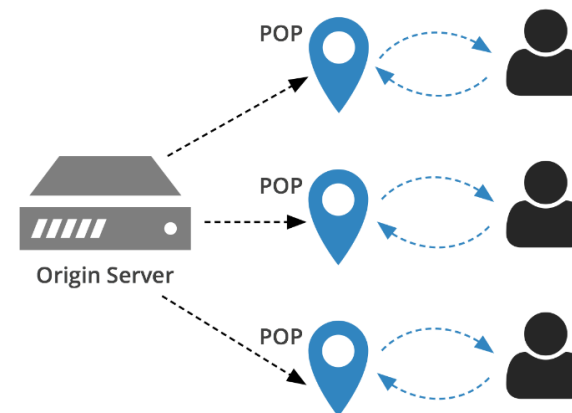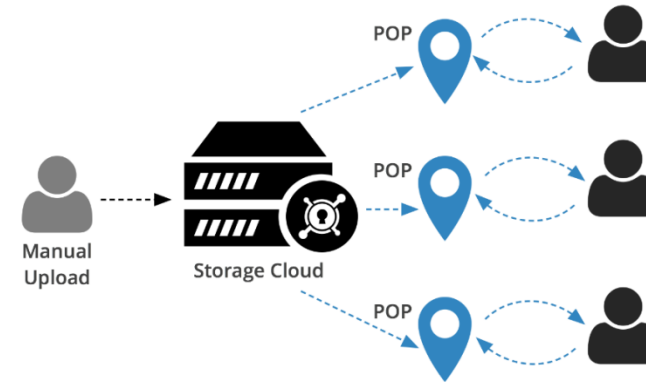  - More accurate, more parameters might be considered
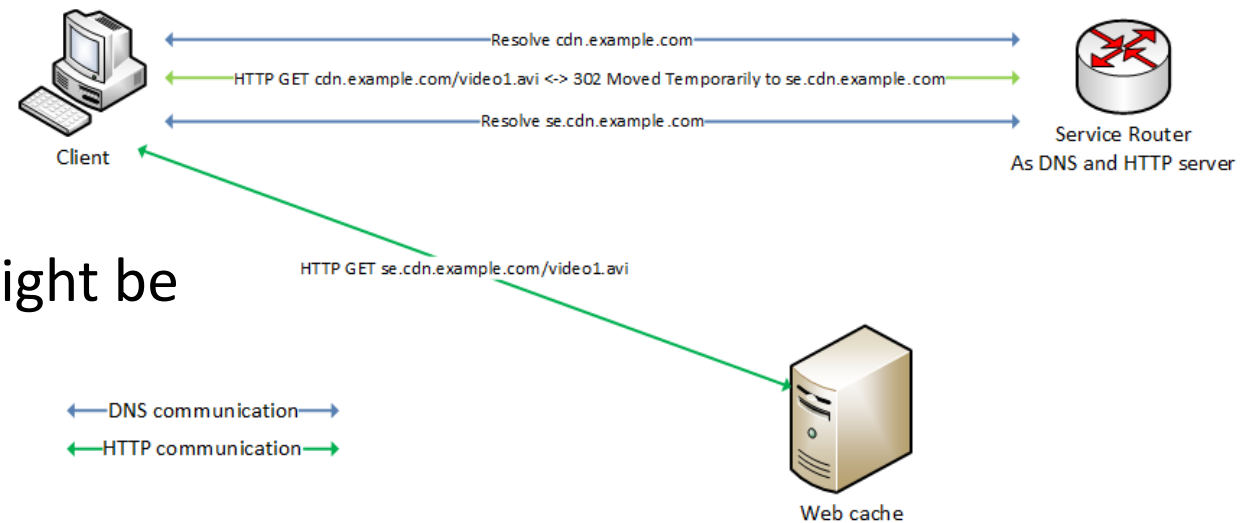
# Transport Layer Redirection

- Not worth mentioning

# Content Distribution

- Content Push
  - Content is pre fetched before any requests
  - Good for flash crowd (e.g. updates)
  - Takes up more space

- Content Pull
  - More common
  - Content fetched from origin on demand

# Case Study

- Magio TV

# Thank you for your attention

**Ing. Tomáš Boros**

tomas.boros@stuba.sk