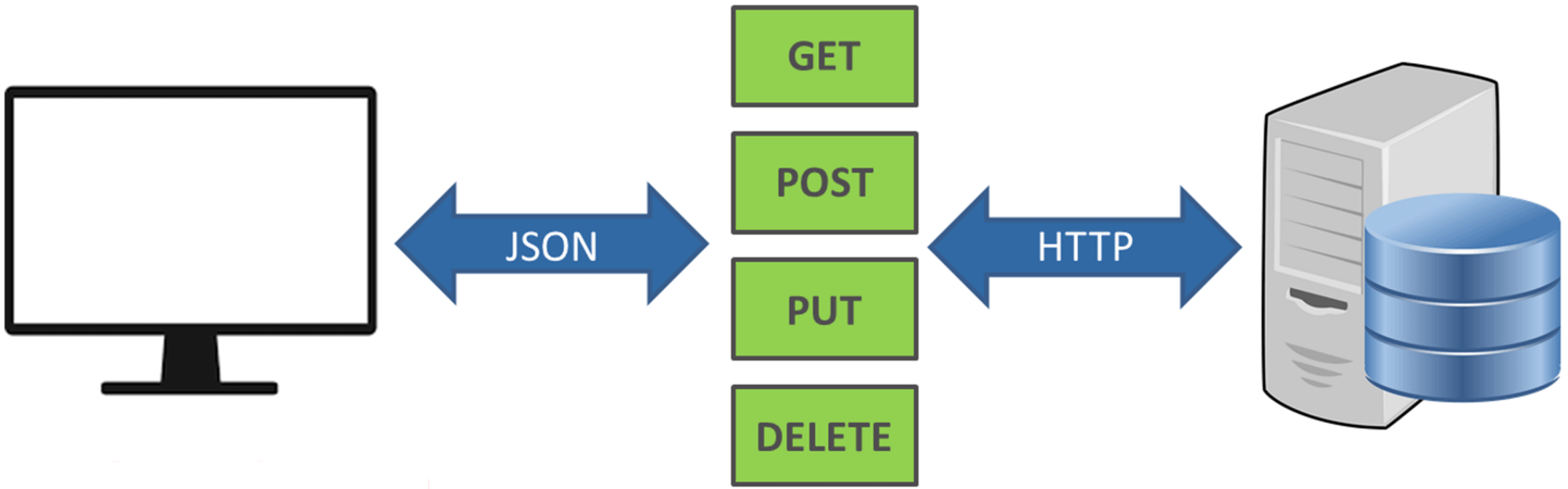
Spherical Cows of HTTP

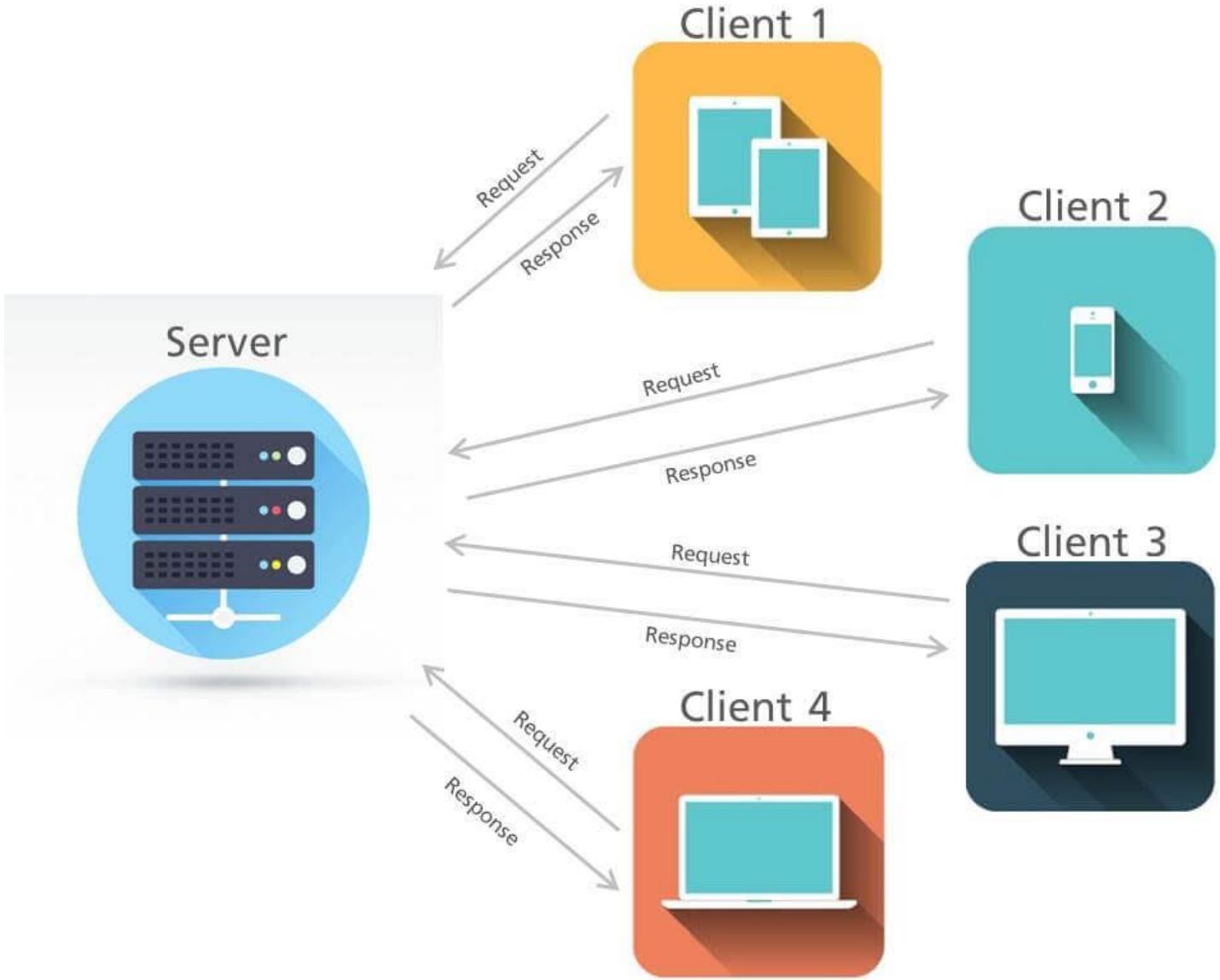
De-over-simplifying the Internet

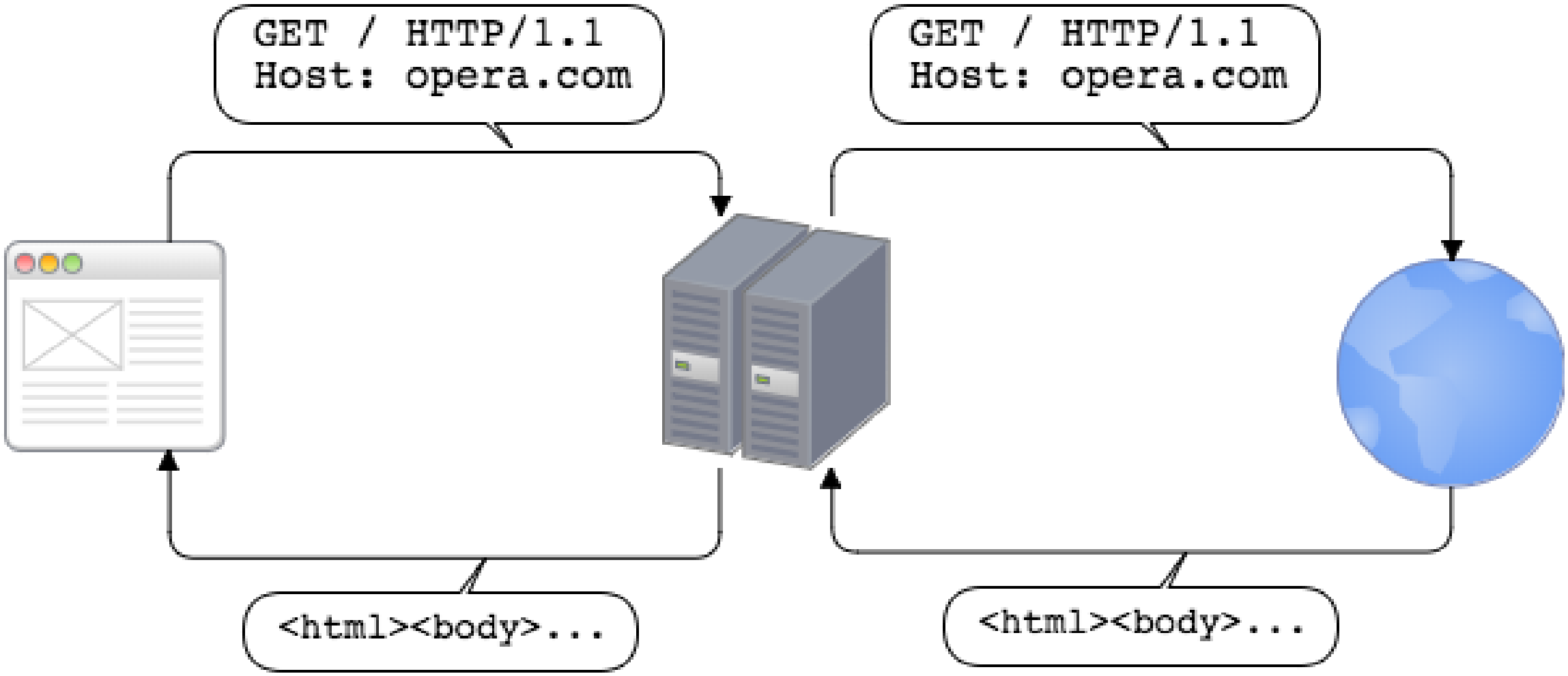


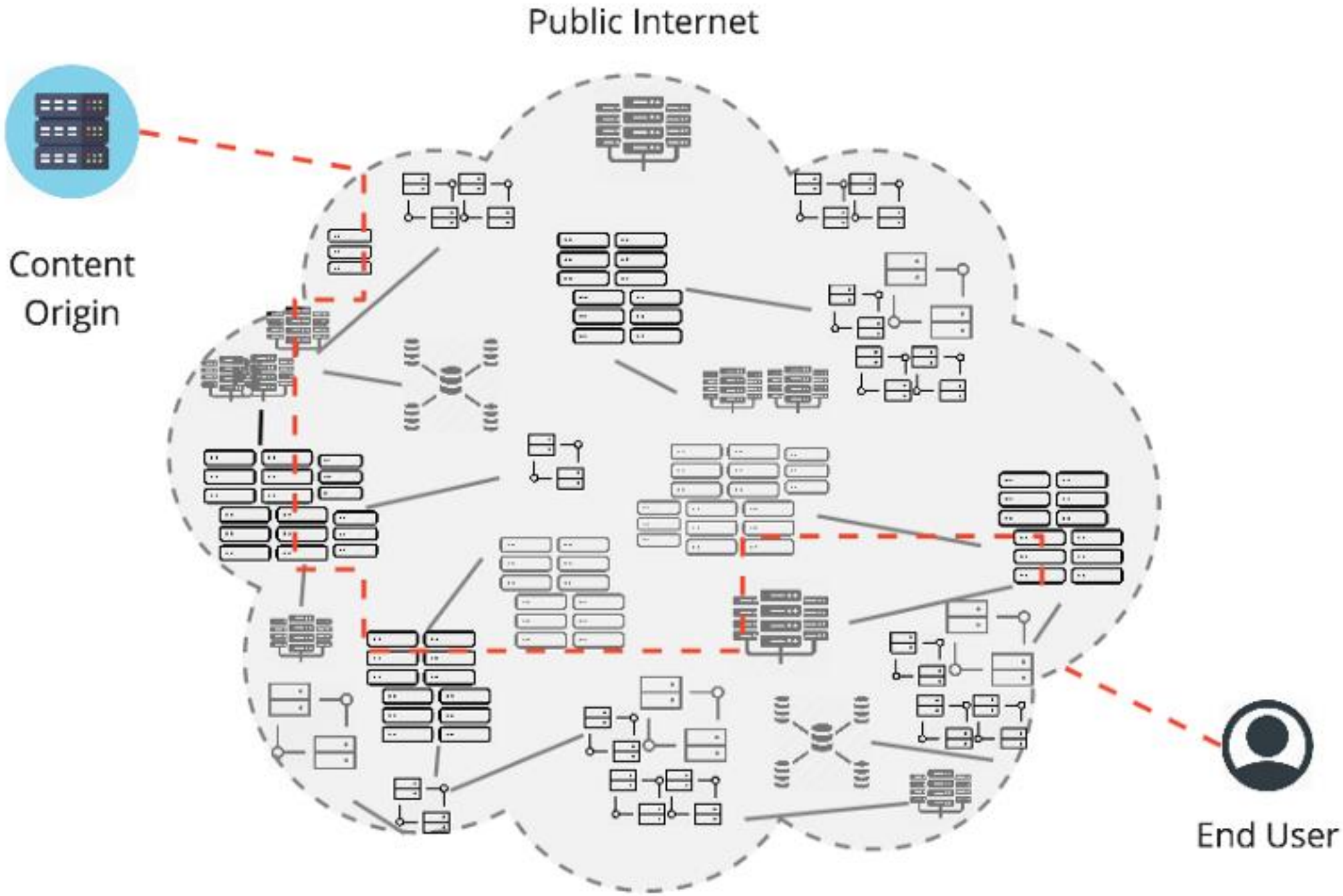
Client sends a **request**

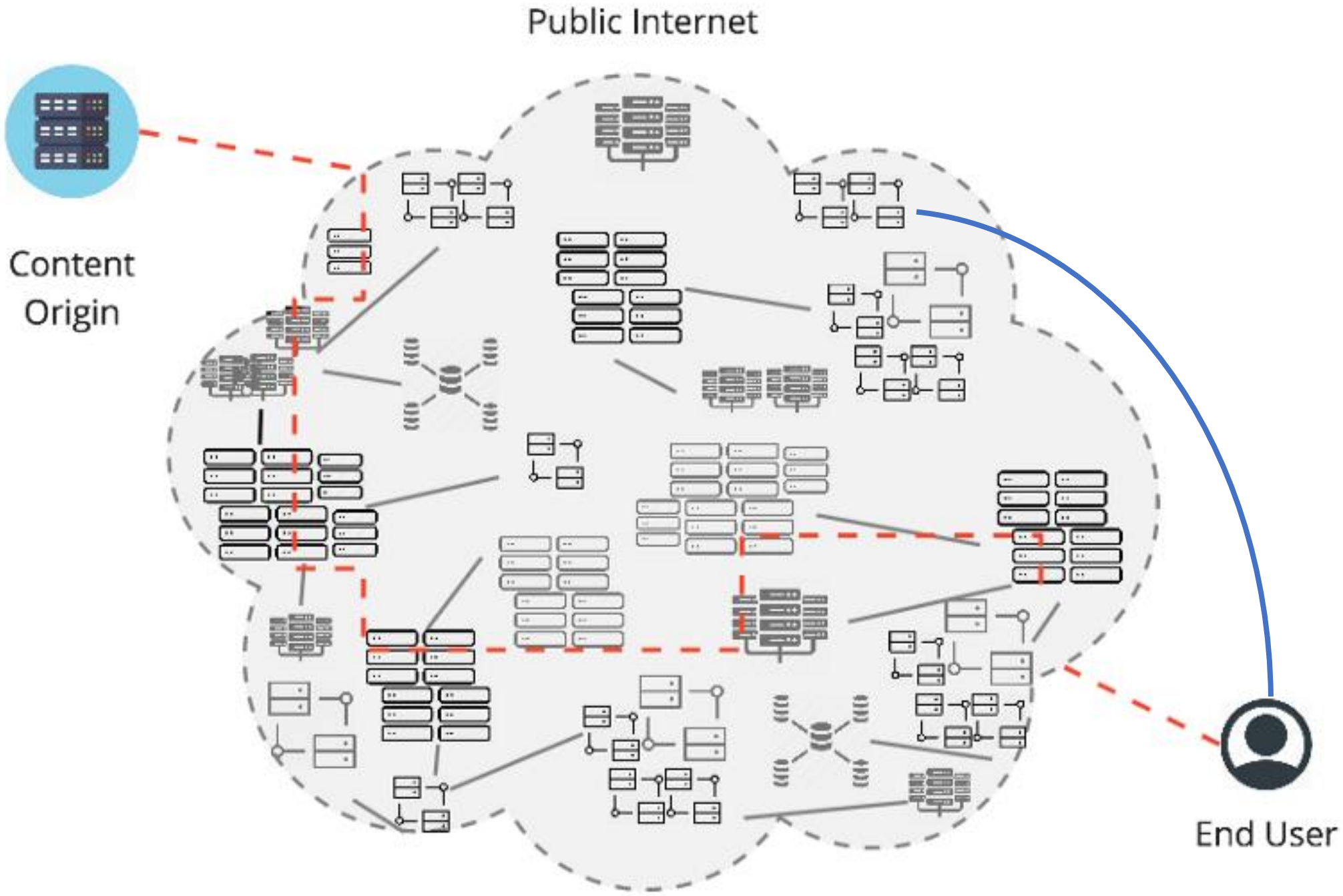
HTTP methods

Server sends a **response**

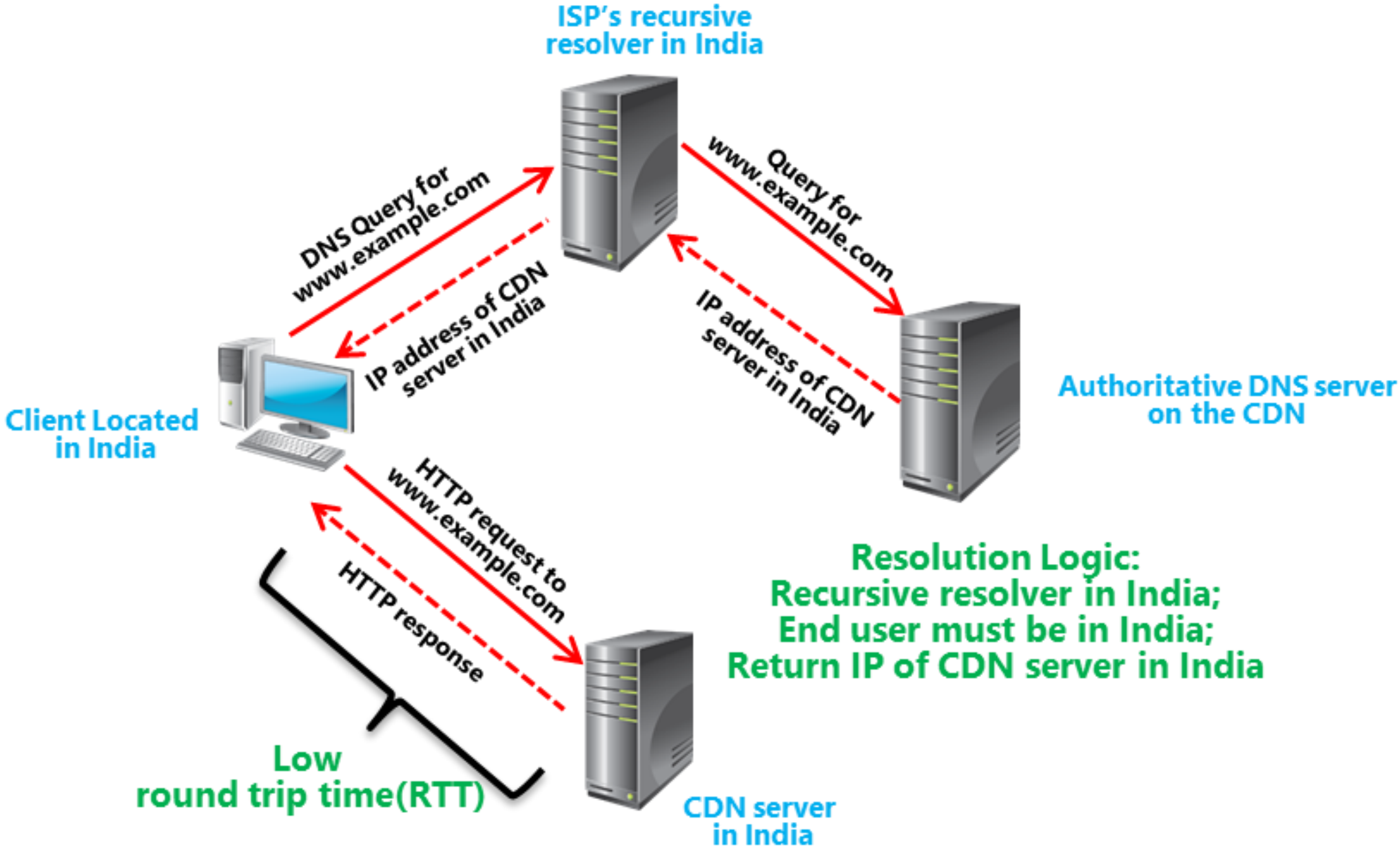

Client 1
Request
Response
Client 2
Server
Request
Response
Request
Client 3
Response
Client 4
Request
Response

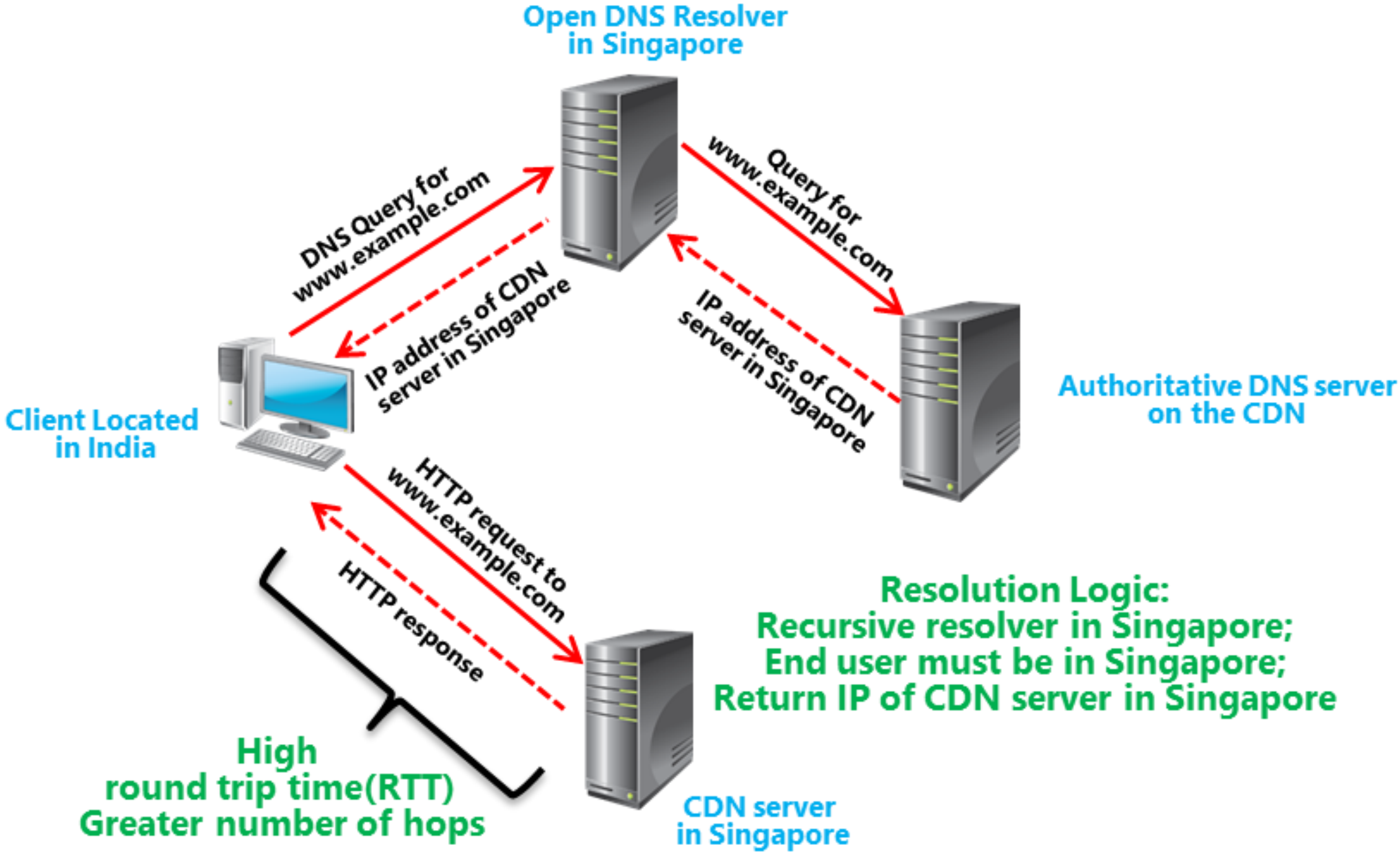

GET / HTTP/1.1
Host: opera.com
GET / HTTP/1.1
Host: opera.com
<html><body>...
<html><body>...


Public Internet
Content Origin
End User


Public Internet
Content Origin
End User

		Am I the right one to serve it?	
		Yes	No
Can I find the right one to serve it?	Yes	2XX	3XX
	No	404	421


ISP's recursive resolver in India
DNS Query for www.example.com
IP address of CDN server in India
Query for www.example.com
IP address of CDN server in India
Client Located in India
Authoritative DNS server on the CDN
HTTP request to www.example.com
HTTP response
Low round trip time(RTT)
CDN server in India
Resolution Logic:
Recursive resolver in India;
End user must be in India;
Return IP of CDN server in India


Open DNS Resolver
in Singapore
DNS Query for
www.example.com
IP address of CDN
server in Singapore
Query for
www.example.com
IP address of CDN
server in Singapore
Client Located
in India
Authoritative DNS server
on the CDN
HTTP request to
www.example.com
HTTP response
Resolution Logic:
Recursive resolver in Singapore;
End user must be in Singapore;
Return IP of CDN server in Singapore
High
round trip time(RTT)
Greater number of hops
CDN server
in Singapore

		Am I the right one to serve it?		
		Yes	Maybe or Kinda	No
Can I find the right one to serve it?	Yes	2XX		3XX
	No	404		421

When might this happen?

DNS mis-resolution	Anycast misrouting	Controlled endpoints	Protocol availability
• Resolver is far from client • Resolver doesn't forward Client Subnet to DNS authoritative	• Anycast reached a suboptimal endpoint	• Some server endpoints aren't public, but you're eligible for them (right network, right capabilities, etc.)	• Server supports more preferred protocol than client used, on this or a different endpoint

Before the request

During the request

For future requests

Ways to Redirect

Before the Request – HTTPS Records

```
; This zone contains/returns different CNAME records
; at different points-in-time.  The RRset for "www" can
; only ever contain a single CNAME.

; Sometimes the zone has:
$ORIGIN customer.example.  ; A Multi-CDN customer domain
www 900 IN CNAME cdn1.svc1.example.

; and other times it contains:
$ORIGIN customer.example.
www 900 IN CNAME customer.svc2.example.

; and yet other times it contains:
$ORIGIN customer.example.
www 900 IN CNAME cdn3.svc3.example.

; With the following remaining constant and always included:
$ORIGIN customer.example.  ; A Multi-CDN customer domain
; The apex is also aliased to www to match its configuration
@     7200 IN HTTPS 0 www
; Non-HTTPS-aware clients use non-CDN IPs
              A    203.0.113.82
              AAAA 2001:db8:203::2
```

```
; Resolutions following the cdn1.svc1.example
; path use these records.
; This CDN uses a different alternative service for HTTP/3.
$ORIGIN svc1.example.  ; domain for CDN 1
cdn1     1800 IN HTTPS 1 h3pool alpn=h3 ech="123..."
                 HTTPS 2 . alpn=h2 ech="123..."
                 A    192.0.2.2
                 AAAA 2001:db8:192::4
h3pool 300 IN A 192.0.2.3
           AAAA 2001:db8:192:7::3

; Resolutions following the customer.svc2.example
; path use these records.
; Note that this CDN only supports HTTP/2.
$ORIGIN svc2.example. ; domain operated by CDN 2
customer 300 IN HTTPS 1 . alpn=h2 ech="xyz..."
          60 IN A    198.51.100.2
                A    198.51.100.3
                A    198.51.100.4
                AAAA 2001:db8:198::7
                AAAA 2001:db8:198::12

; Resolutions following the cdn3.svc3.example
; path use these records.
; Note that this CDN has no HTTPS records
; and thus no ECH support.
$ORIGIN svc3.example. ; domain operated by CDN 3
cdn3      60 IN A    203.0.113.8
                AAAA 2001:db8:113::8
```

Future Requests – Alt-Svc

```
HTTP/1.1 200 OK
Content-Type: text/html
Cache-Control: max-age=600
Age: 30
Alt-Svc: h2=":8000"; ma=60
```

Together?

Clients that implement support for both Alt-Svc and HTTPS records and are making a connection based on a cached Alt-Svc response SHOULD retrieve any HTTPS records for the Alt-Svc alt-authority, and ensure that their connection attempts are consistent with both the Alt-Svc parameters and any received HTTPS SvcParams. If present, the HTTPS record's TargetName and port are used for connection establishment (as in Section 3). For example, suppose that "https://example.com" sends an Alt-Svc field value of:

```
Alt-Svc: h2="alt.example:443", h2="alt2.example:443", h3=":8443"
```

The client would retrieve the following HTTPS records:

```
alt.example.                IN HTTPS 1 . alpn=h2,h3 ech=...
alt2.example.               IN HTTPS 1 alt2b.example. alpn=h3 ech=...
_8443._https.example.com.  IN HTTPS 1 alt3.example. (
    port=9443 alpn=h2,h3 ech=... )
```

Troubles with Alt-Svc

Inability to verify still valid

- Supposed to clear on network change (with exceptions), but clients don’t always know when network changes
- Permits (accidental?) capture by CDN provider even after traffic has shifted

Possibility of disagreement

- DNS is the better source of truth about current network configuration
- Headers received directly from origin are more trusted

Replacement? Delegate to SVCB/HTTPS

Alt-SvcB: “oxford.svc2.example”

Open Debate: Stickiness vs. Disclosure

- If client doesn’t remember the Alt-Svc or clears it too soon, it will get the same redirection from the origin and flip-flop between origin and alternative.
- If client remembers Alt-Svc too long, it will continue using an endpoint which might no longer be in service.
- Current design for stickiness relies on publishing all still-valid alternatives in the origin’s HTTPS record
  - Some providers might not want to publish all endpoints