A WWW caching system
Definition

• In RFC 2616
• Local store of response messages

Hot topic:
THE MOST WIDELY STUDIED APPLICATION ON THE WEB !!
The **GOALS** of Caching

- Reduce user-experienced latency
- Reduce load on the network
- Reduce load on the origin server
Why cache?

- All participants in the Web message exchange can benefit from caching

![Bar chart showing the size of distinct, static pages (in millions) for different months: Jun-97 with 125, Nov-97 with 200, Mar-98 with 275, May-99 with 600.]
Different caching architectures

- Proxy caching
- Adaptive web caching
- Push caching
- Active caching
Proxy caching

1. Reverse proxy caching
2. Transparent caching
   - Intercepting HTTP requests and redirecting them to web cache servers or cache clusters
Figures for transparent proxy
Delay factors

- Network connectivity
- Network congestion and bandwidth issues
- Origin server load
- Time to generate response
- Browser rendering of response
What is cacheable?

Two factors:
1. HTTP protocol-related requirements
2. Content-specific considerations
Protocol-specific consideration (what to cache)

- Abide by HTTP 1.1
  - The request method, header fields, response status and headers have to indicate that the response is cacheable. A constraint on the part of any of them makes the response uncacheable.
  - For example, responses to OPTIONS, PUT and DELETE methods are not cacheable.
Content-specific consideration (what to cache)

- A cache may have its own set of rules for deciding what to cache.
- For example: business thoughts, storage overhead or responses to scripts, the rate of change of resources
Common criteria for cache to decide what to cache

- Is the content typically uncachable (dynamic data or particular content)
- Is the cached response likely to be reused
- Will the decision to cache a particular response lead to replacement of one or more resources
Where to build cache?

- Multiple locations
- 1.browser cache
- 2.proxy cache
- 3.reverse cache
A cache chain

Origin server plus server-side cache

Server's ISP or CDN reverse proxy cache

Client's ISP forward proxy cache

Organization forward proxy cache

Browser plus cache
Cooperative caching
How is caching done?

- Deciding whether the message is cacheable
- Cache replacement and storing the response in cache
- Returning a cached response
- Maintaining a cache
Cache Coherency

What is coherency?
A cache must ensure that the cached response is still fresh before returning it to a client requesting the resource.

Strong consistency
Weak consistency
Two heuristics for cache

- 1. Lease-based approach
- Lease period
- 2. TTL (time to live) approach
- Response has a cache expiration time
Rate of change of resources

- The rate at which different resources on the Web change varies widely.
- The rate raises important questions about cache. (what to cache, how long to cache or is there a relation between the rate and the frequency of its access?)
How to categorize resources to the rate?

- The content type
- The frequency and periodicity of the changes
- The size of the resources
Methods to evaluate cache mechanism

- Hit ratio (HR)
- Byte hit ratio (BHR)
- HR ranges from 30–49 percent
- BHR ranges from 14–36 percent
- (Why?)
One small question

• Why HR and BHR can’t be increased to a very high value?
Possible reasons:

1. Uncachable resources
2. ZIPF's law: less popular, less reference
3. \( \frac{1}{k^{1-a}} \) is a constant
4. High rate of change
Some extended mechanisms about cache

- Clues about future reference
- Filling in or replacing gaps
- Alias discovery
Prefabeching

Local-based or server-hint-based

Different prefetching scenarios:
1. between clients and servers
2. between clients and proxies
3. between proxies and servers