Web Caching and Content Distribution: A View From the Interior

Overview
- Analytical tools have evolved to predict behavior of large-scale Web caches.
  - Are results from existing large-scale caches consistent with the predictions?
    - NLANR
  - What do the models predict for Content Distribution/Delivery Networks (CDNs)?
    - Goal: answer these questions by extending models to predict interior cache behavior.

Goals and Limitations
- Focus on interior cache behavior.
  - Assume leaf caches are ubiquitous.
  - Model CDNs as interior caches.
  - Focus on hit ratio (percentage of accesses absorbed by the “cloud”).
    - Ignore push replication; at best it merely reduces some latencies by moving data earlier.
  - Focus on “typical” static Web objects.
    - Ignore streaming media and dynamic content.

Outline
- Analytical model
  - applied to interior nodes of cache hierarchies
  - applied to CDNs
- Implications of the model for CDNs in the presence of ubiquitous leaf caching
- Match model with observations from the NLANR cache hierarchy
- Conclusion
Analytical Model

- [Wolman/Voelker/Levy et. al., SOSP 1999]
  - refines [Breslau/Cao et. al., 1999], and others
  - Approximates asymptotic cache behavior assuming Zipf-like object popularity
    - caches have sufficient capacity
  - Parameters:
    - \( \lambda \) = per-client request rate
    - \( \mu \) = rate of object change
    - \( p_c \) = percentage of objects that are cacheable
    - \( \alpha \) = Zipf parameter (object popularity)

Cacheable Hit Ratio: the Formula

- \( C_N \) is the hit ratio for cacheable objects achievable by population of size \( N \) with a universe of \( n \) objects.

\[
C_N = \int_1^n \left( 1 + \frac{1}{\mu C x + \lambda N} \right) dx
\]

\[
C = \int_1^n \frac{1}{x} dx
\]

[Wolman/Voelker/Levy et. al., SOSP 99]

Inside the Hit Ratio Formula

Approximates a sum over a universe of \( n \) objects...
...of the probability of access to each object \( x \)...
...times the probability \( x \) was accessed since its last change.

\[
C = \int_1^n \frac{1}{x} dx
\]

\[ C = \int_1^n \frac{1}{x} dx \]

in [Breslau/Cao 99]

0 < \( \alpha \) < 1

An Idealized Hierarchy

Assume the trees are symmetric to simplify the math.
Ignore individual caches and solve for each level.

Hit Ratio at Interior Level \( i \)

- \( C_N \) gives us the hit ratio for a complete subtree covering population \( N \)
- The hit ratio predicted at level \( i \) or at any cache in level \( i \) is given by:

\[
\frac{h_i}{r_i} = \frac{R_{p_c}(C_N - C_{N_{i+1}})}{r_{i+1} - h_{i+1}}
\]

“the hits for \( N_i \) (at level \( i \)) minus the hits captured by level \( i+1 \), over the miss stream from level \( i+1 \)”

Root Hit Ratio

- Predicted hit ratio for cacheable objects, observed at root of a two-level cache hierarchy (i.e. where \( r_2 = R_{p_c} \)):

\[
\frac{h_1}{r_1} = \frac{C_{N_1} - C_{N_2}}{1 - C_{N_2}}
\]
Generalizing to CDNs

Symmetry assumption: $f$ is stable and "balanced".

What happens to $C_n$ if we partition the object universe?
Hit ratio in CDN caches

Given the symmetry and balance assumptions, the cacheable hit ratio at the interior (CDN) nodes is:

$$\frac{C_{N_j} - C_{N_l}}{1 - C_{N_l}}$$

$N_j$ is the covered population at each CDN cache. $N_l$ is the population at each leaf cache.

Analysis (cont’d)

- Fixed parameters (unless noted otherwise):
  - $\lambda$ (client request rate) = 590 reqs./day
  - $\mu$ (rate of object change) =
    - once every 14 days (popular objects, 0.3%)
    - once every 186 days (unpopular objects)
  - $p_c$ (percent of requests cacheable) = 60%
  - $\alpha$ (Zipf parameter - object popularity) = 0.8

Analysis

- We apply the model to gain insight into interior cache behavior with:
  - varying leaf cache populations ($N_l$)
    - e.g., bigger leaf caches
  - varying ratio of interior to leaf cache populations ($N_i/N_l$)
    - e.g., more specialized interior caches
  - Zipf $\alpha$ parameter changes
    - e.g., more concentrated popularity
Cacheable interior hit ratio observed at interior level fixing interior/leaf population ratio

Increasing $N_I$ and $N_L$ →

Cacheable interior hit ratio as percentage of all cacheable requests, fixing interior/leaf population ratio

Increasing $N_I$ and $N_L$ →

Cacheable interior hit ratio fixing leaf population

Increasing “bushiness” →

Cacheable interior hit ratio as percentage of all requests fixing leaf population

Increasing “bushiness” →

Cacheable interior hit ratio as percentage of all requests varying Zipf $\alpha$ parameter

$N_L$ fixed at 1024 clients

Cacheable interior hit ratio as percentage of all requests varying Zipf $\alpha$ parameter

$N_I/N_L$ fixed at 64K
Conclusions (I)

- Interior hit ratio captures effectiveness of upstream caches at reducing access traffic filtered by leaf/edge caches.
  - Hit ratios grow rapidly with covered population.
- Edge cache populations ($N_L$) are key: is it one thousand or one million?
  - With large $N_L$, interior ratios are deceptive.
  - At $N_L = 10^5$, interior hit ratios might be 90%, but the CDN sees less than 20% of the requests.

Correlating with NLANR Observations

- Do the predictions match observations from existing large-scale caches?
- Observations made from traces provided by NLANR (10/12/99).
  - Observed total hit ratio at (unified) root is 32%
  - 200 of the 914 leaf caches in the trace account for 95% of requests
  - Daily request rate indicates population is on the order of tens of thousands
  - What is the predicted $N$?

Model vs. Reality

- NLANR roots cooperate; we filter the traces to determine the unified root hit ratio.
- NLANR caches are bounded; traces imply that capacity misses are low at 16GB.
- Analysis assumes the population is balanced across the 200 leaves of consequence.
- Analysis must compensate for objects determined to be uncacheable at a leaf.

Conclusions (II)

- NLANR root effectiveness is around 32% today; it is serving its users well.
- NLANR experiment could validate the model, but more data from the experiment is needed.
  - E.g., covered populations, leaf summaries
- The model suggests that the population covered by NLANR is relatively small.
  - With larger $N$ and $N_L$, higher root hit ratios are expected, with lower marginal benefit.
Modeling CDNs

- If the routing function satisfies three properties:
  - an interior cache sees all requests for each assigned object \( x \) from a population of size \( N_i \)
  - every interior cache sees an equivalent object popularity distribution \( \left( \frac{n}{\lambda} \right) \) held constant
  - all requests are routed through leaf caches that serve \( N_L \) clients

- Then interior cacheable hit ratio is:

\[
\frac{C_{N_i} - C_{N_L}}{1 - C_{N_L}}
\]

Hit ratio with detected uncachable documents

- \( p_u \) is the percentage of uncachable requests detected at request time (and not forwarded to parents):

\[
\begin{align*}
\frac{h_1}{r_1} &= \frac{RP_{\bar{c}}(C_{N_i} - C_{N_L})}{R - R_{m} - (1 - p_u)(1 - p_c)r_{m,1}} \\
\frac{h_2}{r_2} &= \frac{H_{m} - H_{m,1}}{1 - H_{m} - (1 - p_u)(1 - p_c)}
\end{align*}
\]

Cache Hierarchies

- As introduced by the Harvest project
  - \( k \) levels of demand-side caches arranged in a tree (for now)
  - clients are bound to leaves
  - each node’s miss stream routes to its parent

- As extended by NLANR (Squid)
  - NLANR-operated root caches cooperate by partitioning URL space

Cache Hierarchies Illustrated