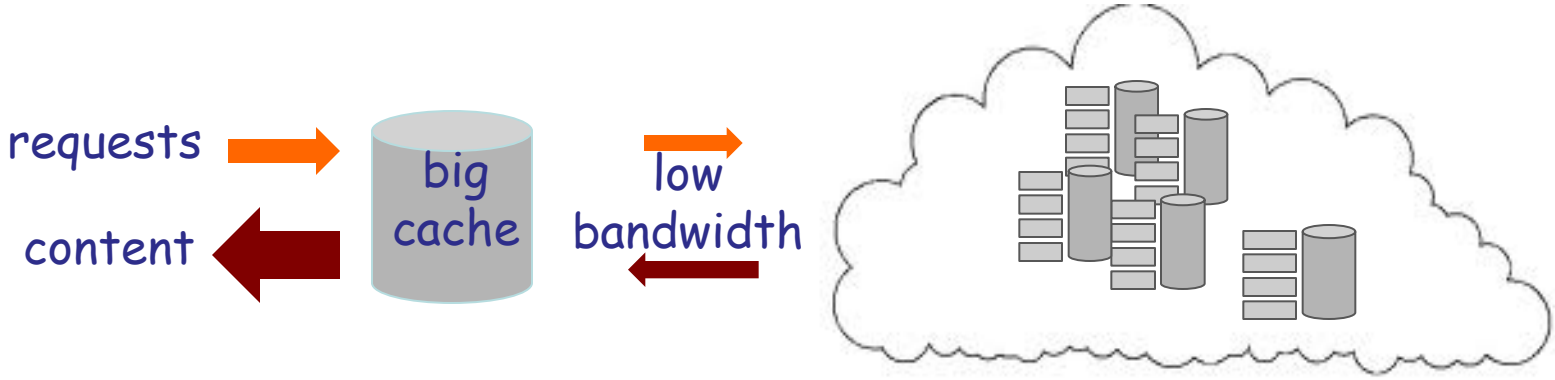
Performance and cost effectiveness of caching in mobile access networks

Jim Roberts (IRT-SystemX)
joint work with Salah Eddine Elayoubi (Orange Labs)

ICN 2015
October 2015

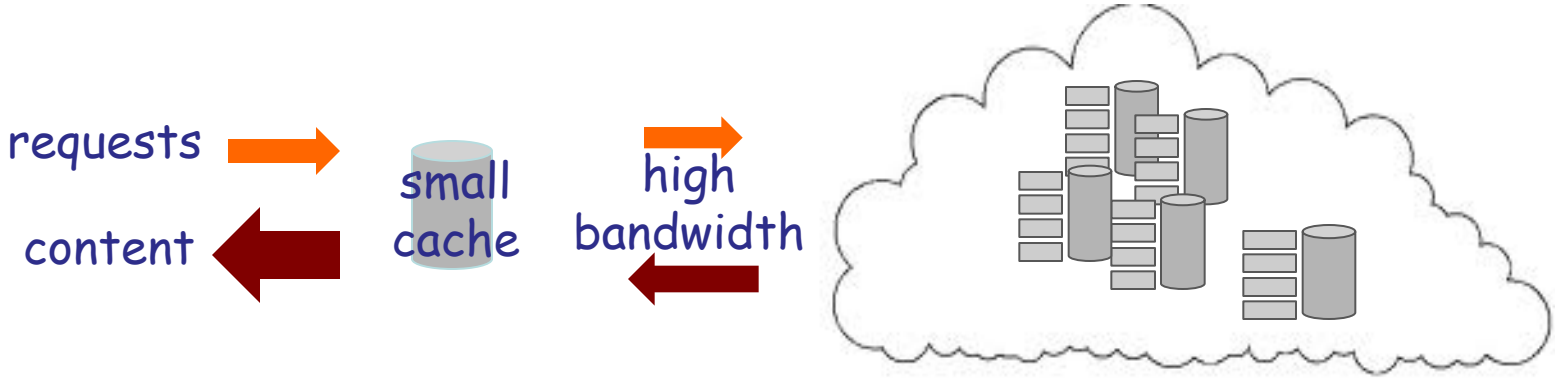
The memory-bandwidth tradeoff

- preferred cache size depends on overall cost of memory (cache capacity) and bandwidth (including routers)
  - more memory means less traffic and therefore less bandwidth



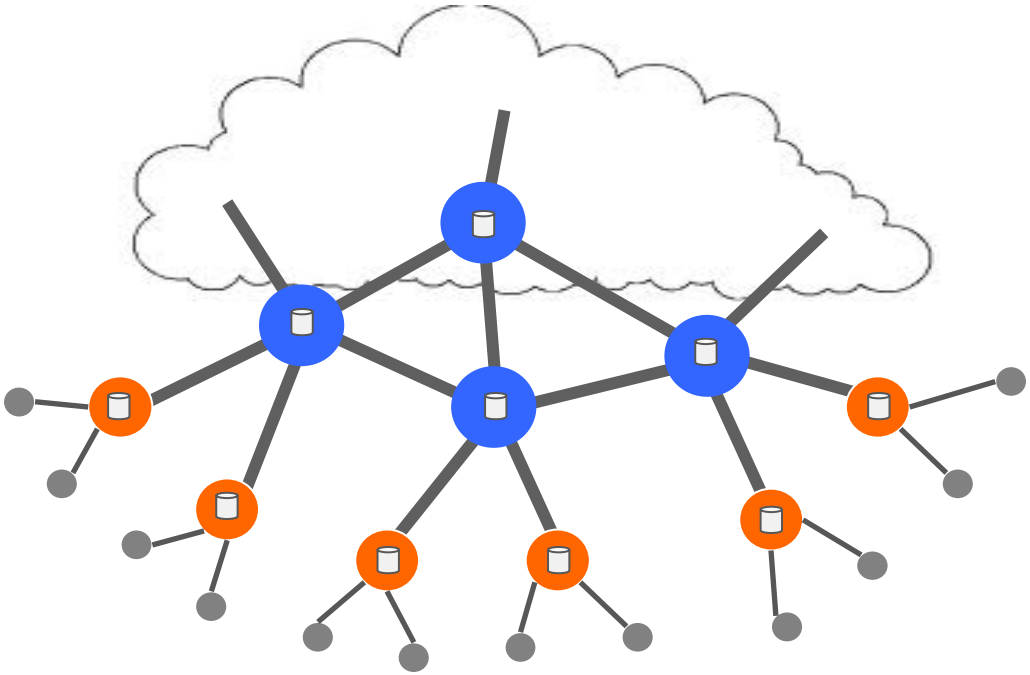
The memory-bandwidth tradeoff

- preferred cache size depends on overall cost of memory (cache capacity) and bandwidth (including routers)
  - more memory means less traffic and therefore less bandwidth



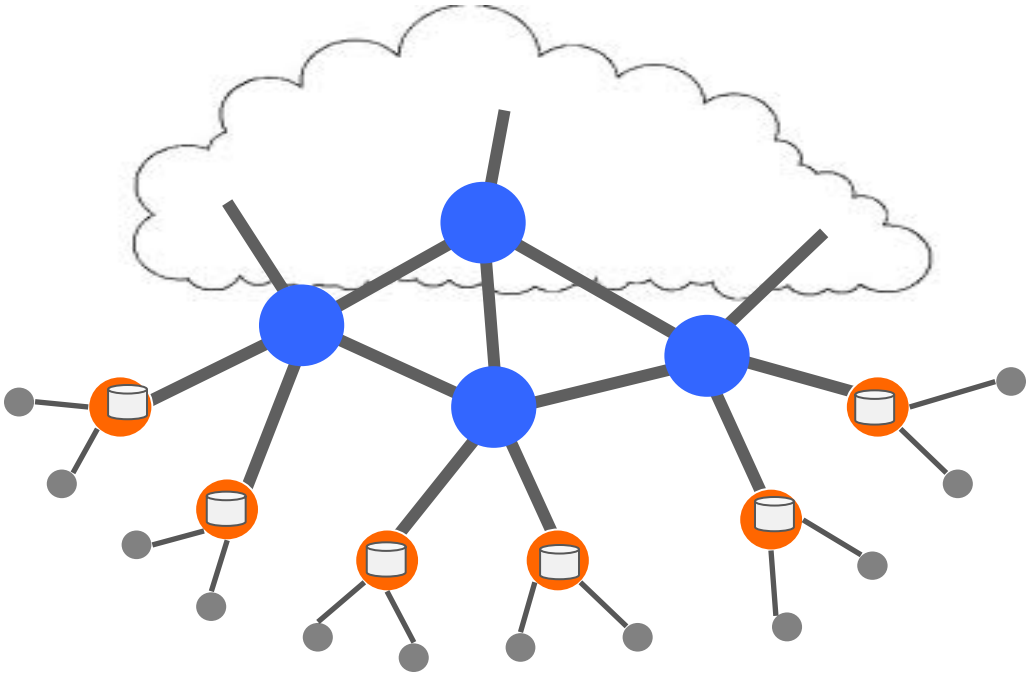
In-network caching or caches at the edge only?

- our prior work suggests caching nearly all content at the "edge" is cost effective [Roberts & Sbihi, 2013]



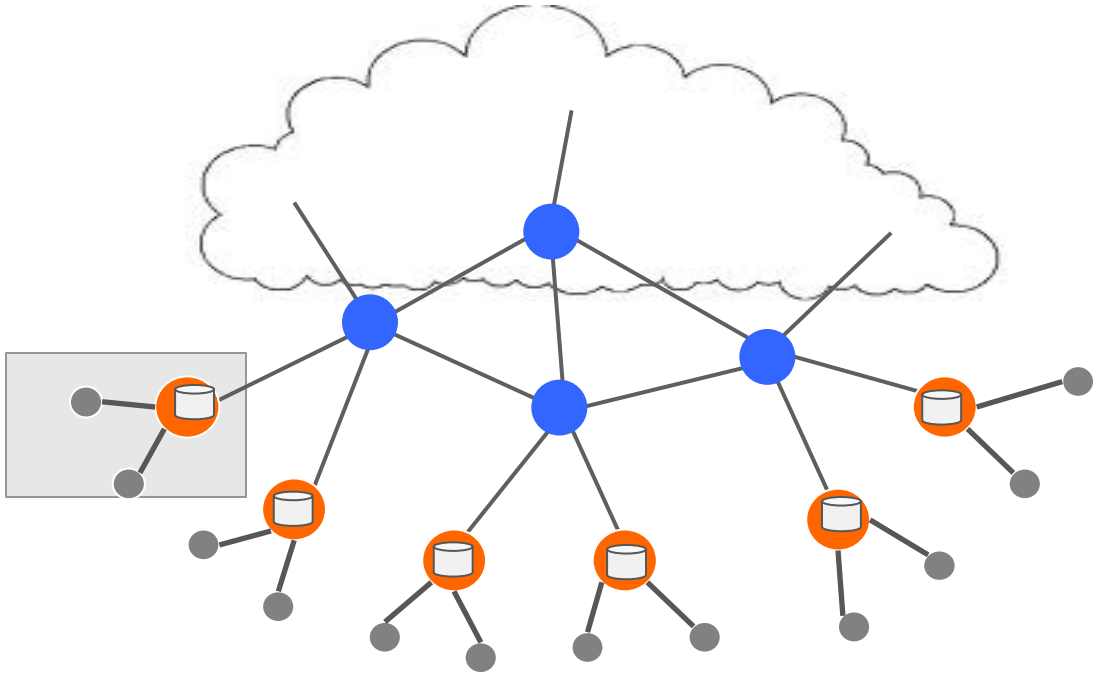
In-network caching or caches at the edge only?

- our prior work suggests caching nearly all content at the "edge" is cost effective [Roberts & Sbihi, 2013]



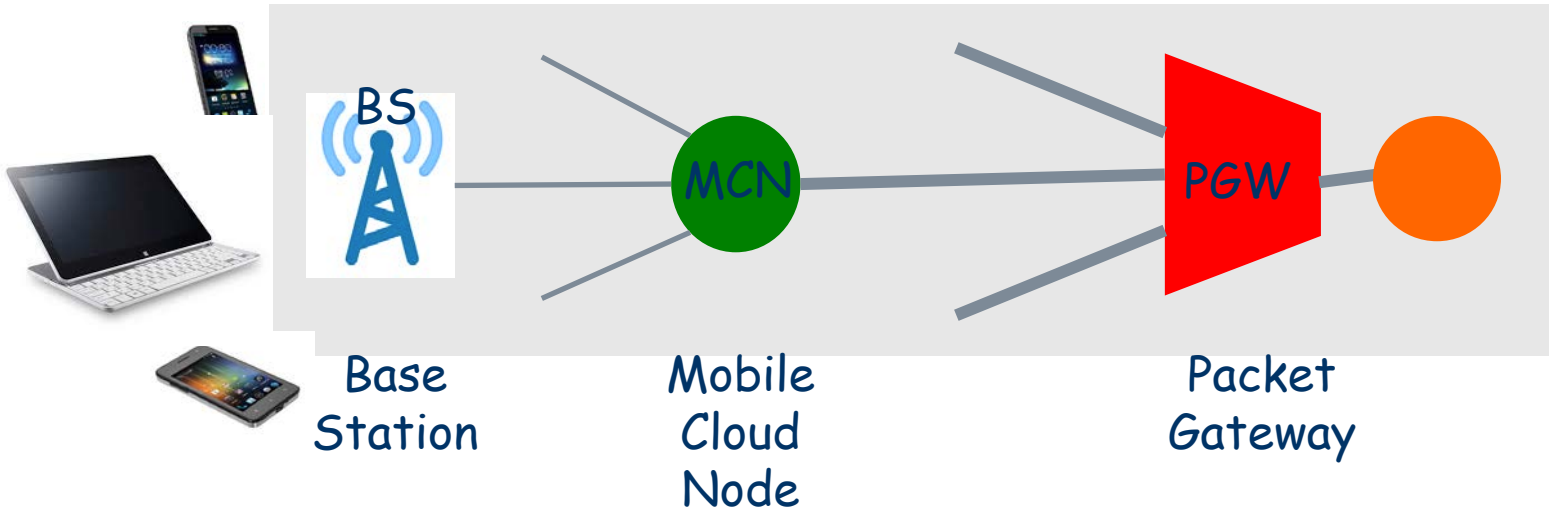
In-network caching or caches at the edge only?

- our prior work suggests caching nearly all content at the "edge" is cost effective [Roberts & Sbihi, 2013]
- but where is the edge?



Caching in mobile access networks

- but where is the edge in the mobile access network?
  - eg, is it worth caching content in base stations or gateways?
- the tradeoff depends on hit rate performance
  - what caching policies to employ at BS, MCN, PGW?
  - eg, is LRU OK at the BS or do we need proactive caching?

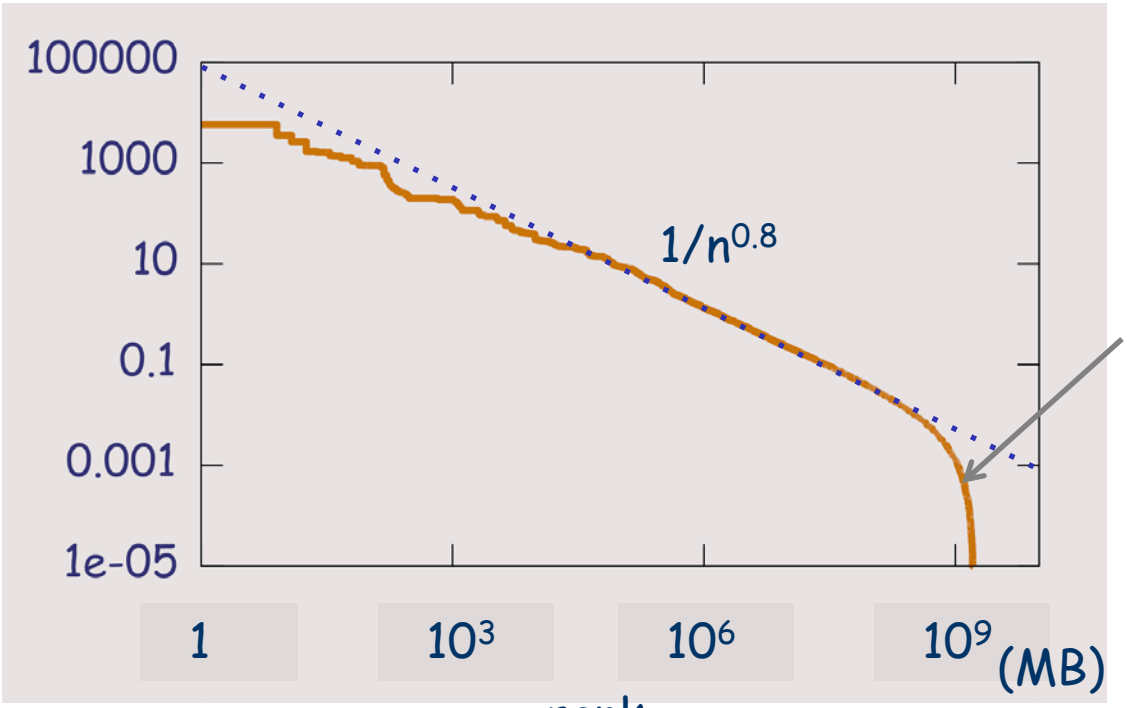


Outline

1. cache hit rate performance
2. evaluating the memory bandwidth tradeoff

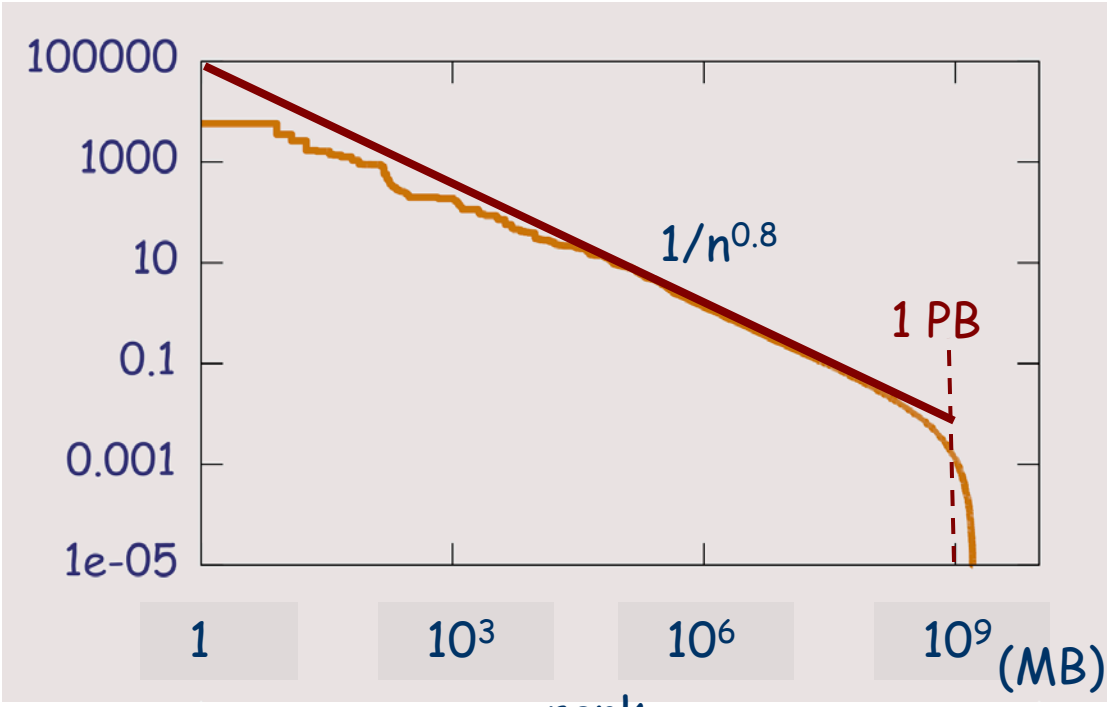
Content popularity

- popularity is measured by request arrival rate
- measurements reveal popularity decreases as a power law:
  - request rate of n^{th} most popular chunk $\propto 1/n^{\alpha}$
  - typically, $\alpha \approx 0.8$



Content popularity

- cache performance depends significantly on catalogue size
- our guesstimates
  - 1 PB for all content (YouTube, web, social networks, P2P, …)
  - 1 TB for a VoD catalogue or for a small user population

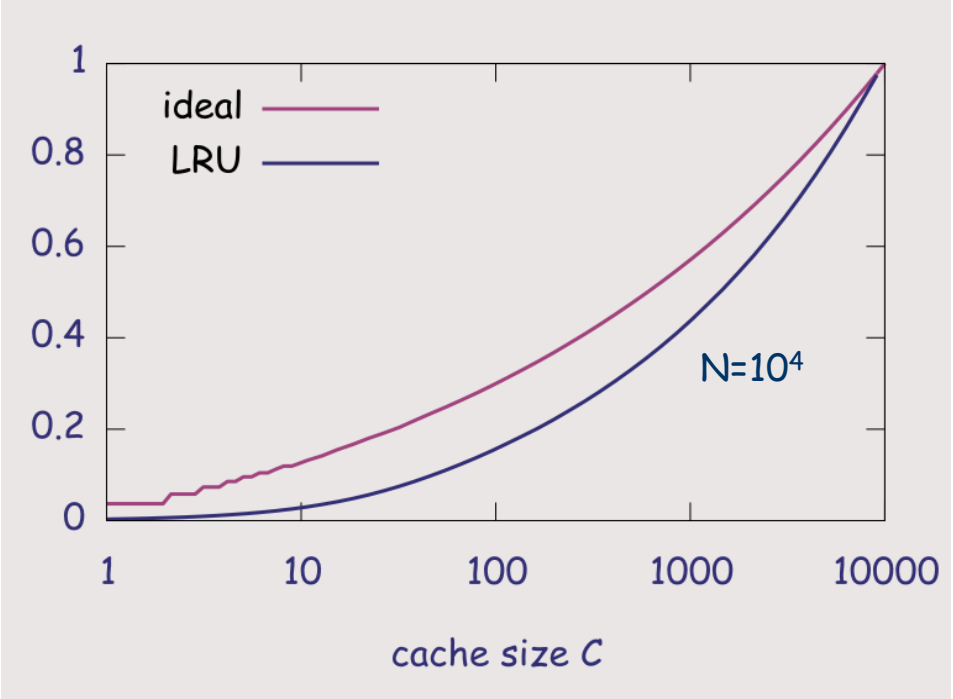


Content popularity

- cache performance depends significantly on catalogue size
- our guesstimates
  - 1 PB for all content (YouTube, web, social networks, P2P, ...)
  - 1 TB for a VoD catalogue or for a small user population
- for reproducibility, assume Zipf(.8) popularity
  - $q_i \propto 1 / i^{.8}$ and $\sum_{1 \le i \le N} q_i = 1$,
  - N and chunk size set so catalogue size is 1 TB or 1 PB
  - (for large systems, results depend on catalogue size in bytes and not on chunk size)

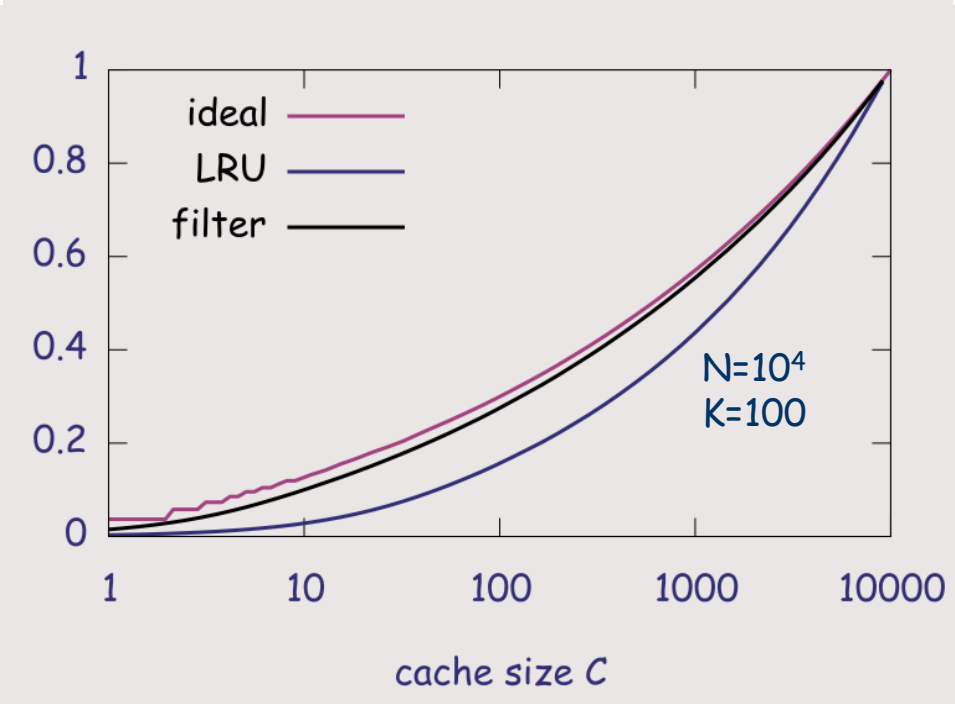
Hit rate and cache policy – stationary demand

- "ideal" cache
  - cache holds most popular items
  - hit rate, $h(C,N)$ =
    $= \sum_{i \le C} q_i \approx (C/N)^{(1-\alpha)} = h(C/N)$
- least recently used (LRU)
  - use "Che approximation":
    $h_i = 1 - \exp(-q_i t_c)$ where t_c satisfies $C = \sum h_i$
  - a significant performance penalty for small caches



Hit rate and cache policy – stationary demand

- cache with "pre-filter"
  - on cache miss, only add new item if included in previous K requests
  - $h_i^{(n+1)} = (1 - \exp(-q_i t_c)) \times (h_i^{(n)} + (1-h_i^{(n)})(1 - (1-q_i)^K))$
  - where $h_i^{(n)}$ is hit rate of n^{th} request for item i
  - for stationary demand $h_i^{(n+1)} = h_i^{(n)} = h_i$, $C = \sum h_i$ yields t_c
- but pre-filters slow reactivity to popularity changes ...



Time varying popularity

- many items are short-lived, cf. [Traverso 2013]
  - we assume the most popular have shortest lifetimes
- stationarity assumption is not appropriate when demand is low
  - eg, the first request for a new item is necessarily a miss

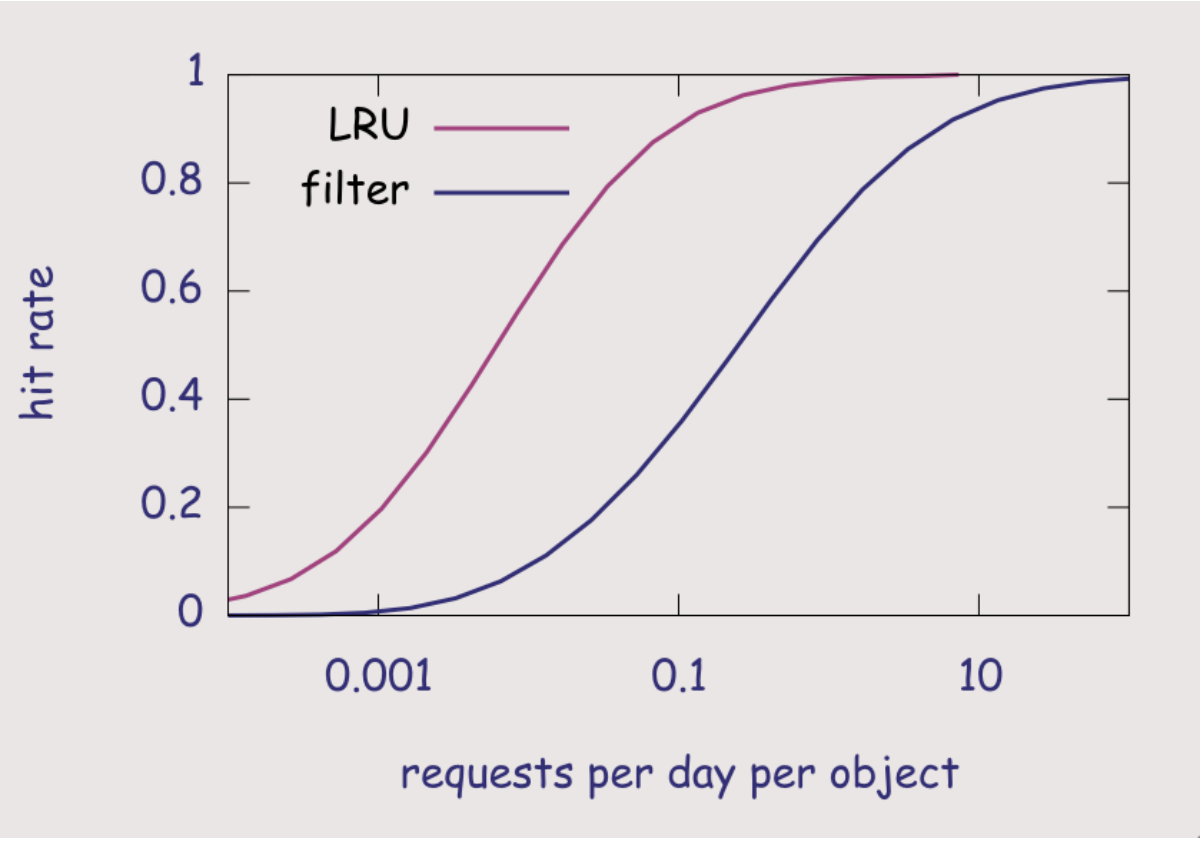
lifetime interval	proportion of items	mean lifetime
0-2 days	.5 %	1.1 days
2-5 days	.8 %	3.3 days
5-8 days	.5 %	6.4 days
8-13 days	.8 %	10.6 days
> 13 days (or < 10 reqs)	97.4 %	1 year

Hit rates with finite lifetimes

- model after [Wolman 1999]: item i always has popularity q_i but changes after each lifetime
- LRU hit rate with mean item lifetime τ_i
  - first request after change must miss
  - $h_i = (1 - \exp(-q_i t_c)) \times (q_i \tau_i / (1 + q_i \tau_i))$
- LRU hit rate with pre-filter
  - recall: $h_i^{(n+1)} = (1 - \exp(-q_i t_c)) \times (h_i^{(n)} + (1-h_i^{(n)})(1 - (1-q_i)^K))$ $\quad (*)$
  - assume item i changes after n^{th} request with probability $1 - \eta_i$ where $\eta_i = q_i \tau_i / (1 + q_i \tau_i)$
  - then, $h_i = h_i^{(1)} (1 - \eta_i) + h_i^{(2)} \eta_i (1 - \eta_i) + h_i^{(3)} \eta_i^2 (1 - \eta_i) + \cdots$
  - multiply $(*)$ by η_i^n and add eventually yields h_i

Impact of time-varying popularity

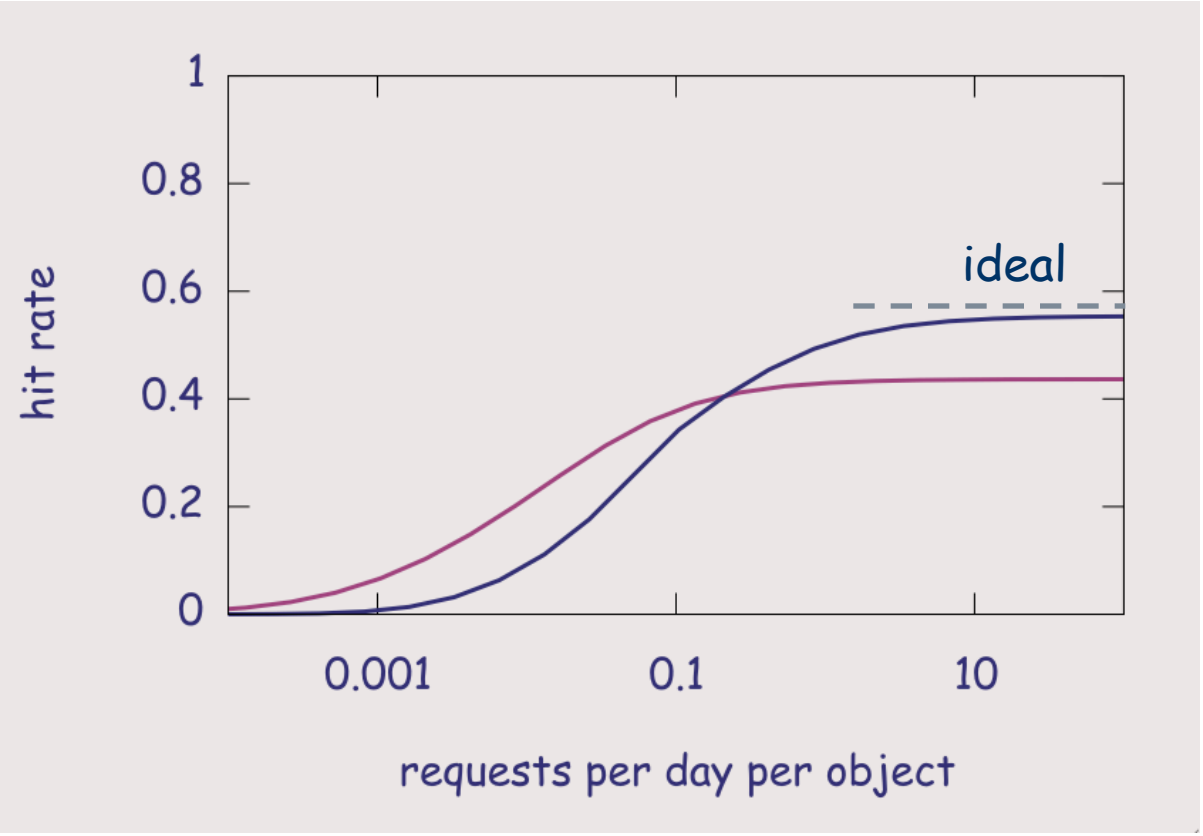
- hit rate depends on demand since first requests in lifetime always miss (≥1 for LRU, ≥2 for LRU with pre-filter)



full capacity (C=N)

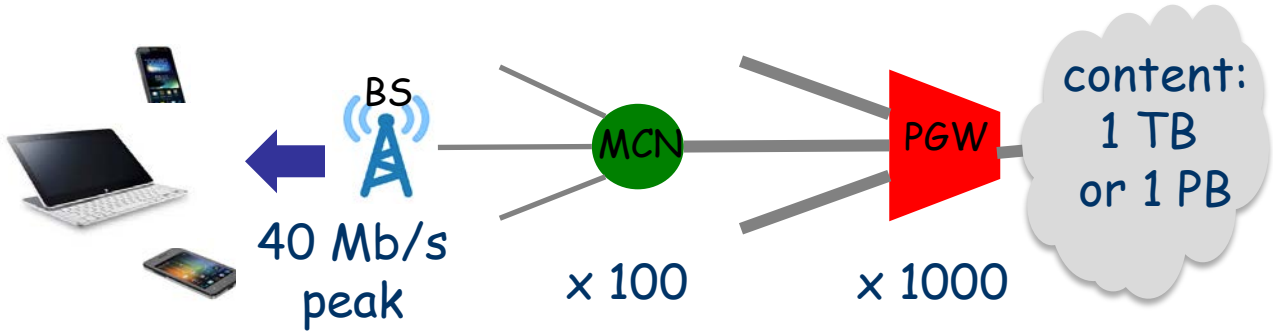
Impact of time-varying popularity

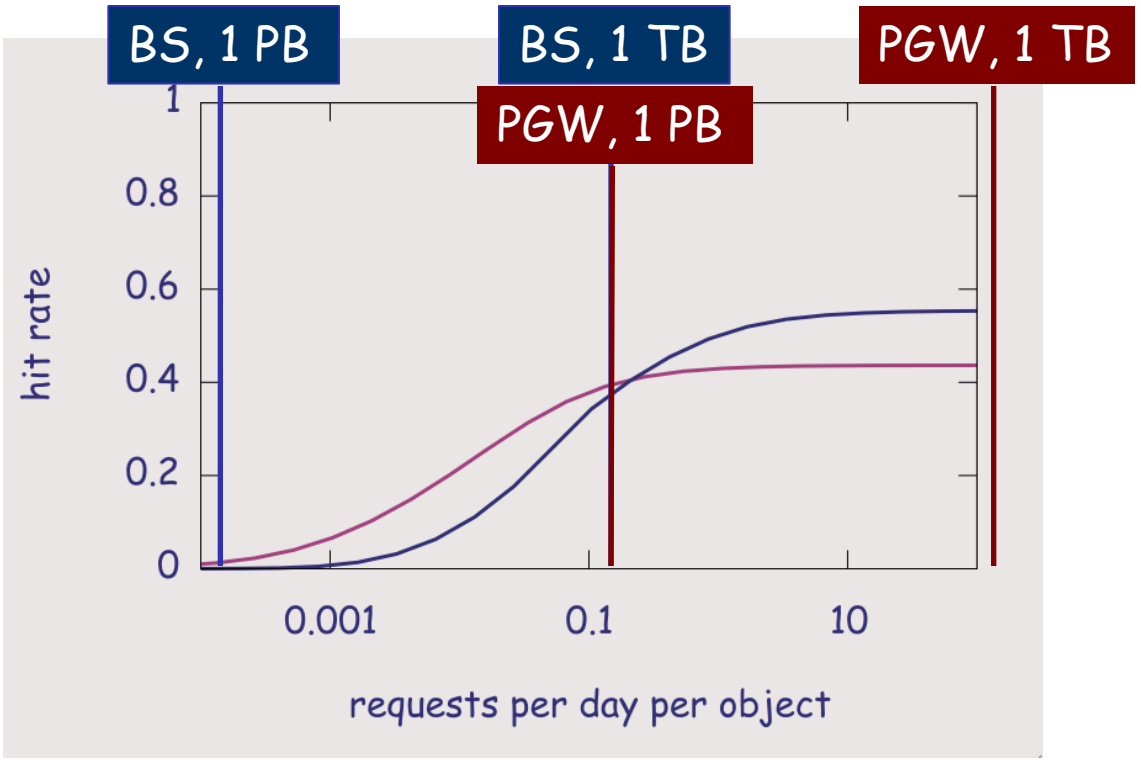
- hit rate depends on demand since first requests in lifetime always miss (≥1 for LRU, ≥2 for LRU with pre-filter)



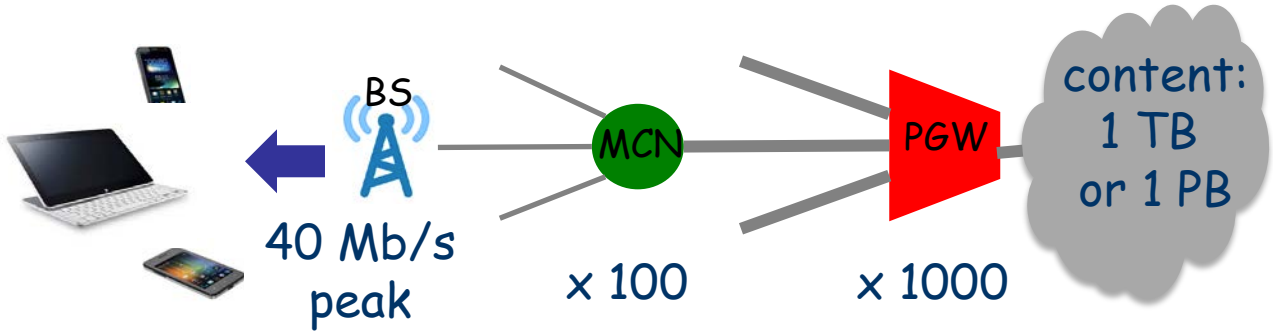
C=N/10

Application to mobile access





Implications



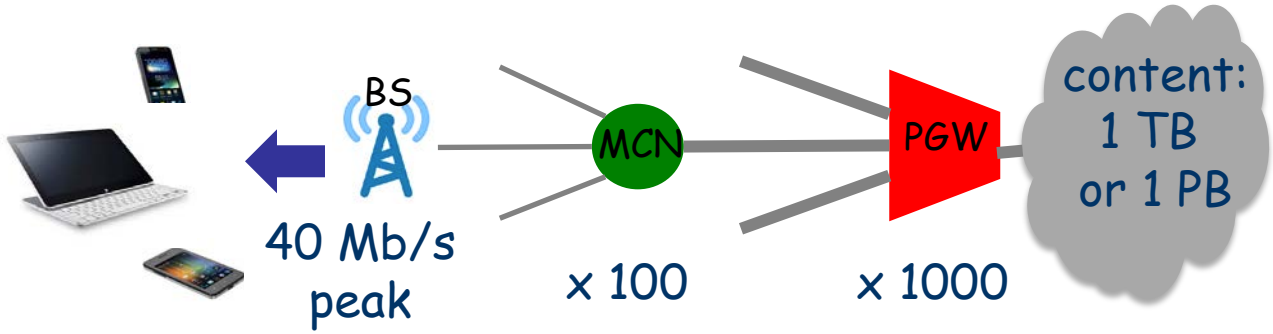
- we need proactive caching at BS (and MCN)
  - ie, network must proactively upload the most popular items
- even PGW may not concentrate enough traffic to make reactive caching effective
  - edge cache shared by multiple access networks makes more sense
- proactive caching needs some function to predict popularity
  - by being informed of requests from a large user population

Outline

1. cache hit rate performance
2. evaluating the memory bandwidth tradeoff

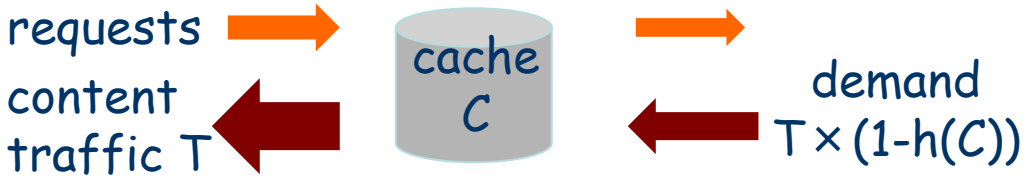
Evaluating the tradeoff at the PGW

- the packet gateway hosts a small data center with modular cache capacity
- caches have ideal performance (eg, proactive or pre-filter)
- popularity is Zipf(.8) with a catalogue of 1 TB or 1 PB



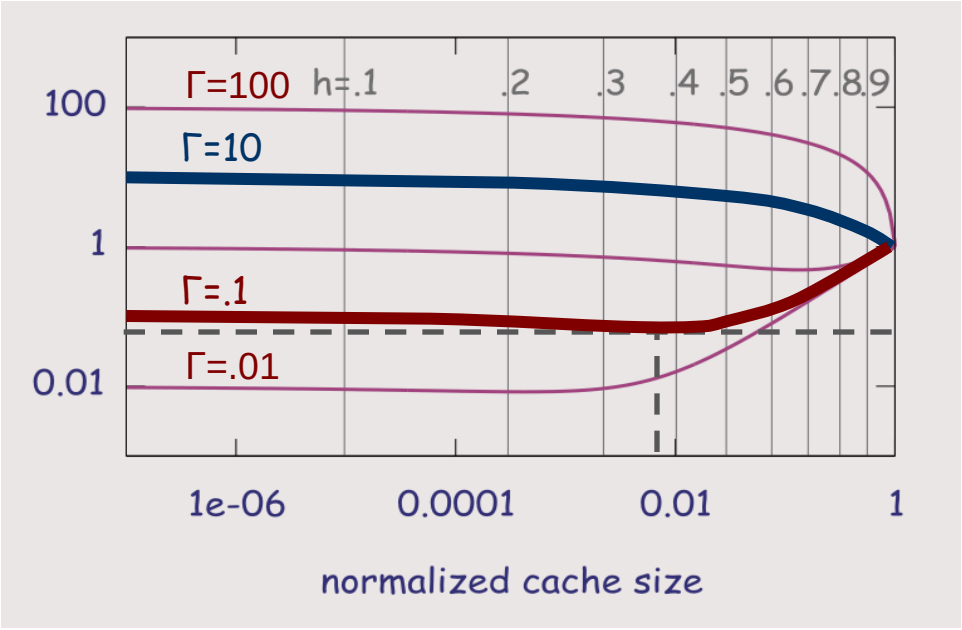
Evaluating the tradeoff at the PGW

- overall cost of cache and bandwidth is
  - $\Delta(C) = K_b(T \times (1\text{-}h(C))) + K_m(C)$
  - where T is download traffic, $h(C)$ is hit rate, $K_b(D)$ and $K_m(C)$ are cost functions for bandwidth D and cache C
- to simplify, assume linear cost functions
  - $K_b(D) = k_b \times D, \quad K_m(C) = k_m \times C$
  - where k_b and k_m are marginal costs of bandwidth and memory
- consider normalized cost $\delta(c)$ for relative cache size $c = C/N$
  - $\delta(c) = \Gamma \times (1\text{-}h(c)) + c$ (ie, $\delta(1) = 1$ and $\delta(0) = \Gamma$)
  - where $\Gamma = k_b T/k_m N$ is ratio of max bandwidth cost to max cache cost



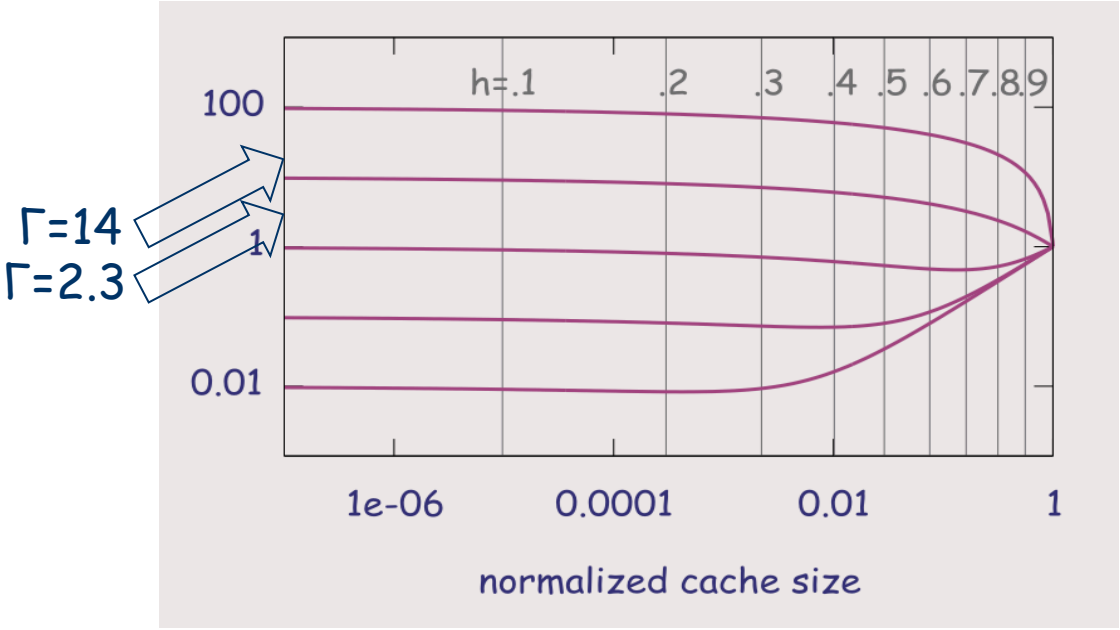
Normalized cost v normalized cache size

- normalized cost $\delta(c) = \Gamma \times (1-h(c)) + c = \Gamma \times (1-c^{0.2}) + c$
- where $\Gamma = k_b T/k_m N$ is max bandwidth cost / max cache cost
- if $\Gamma \geq 5$, max cache is optimal (c=1, ie, C=N)
- if $\Gamma < 5$, there is optimum cache size for 0<c<1 but gain is limited
  - eg, for $\Gamma = .1$, min cost for c=.008, h(c)=.37 but gain < 30%



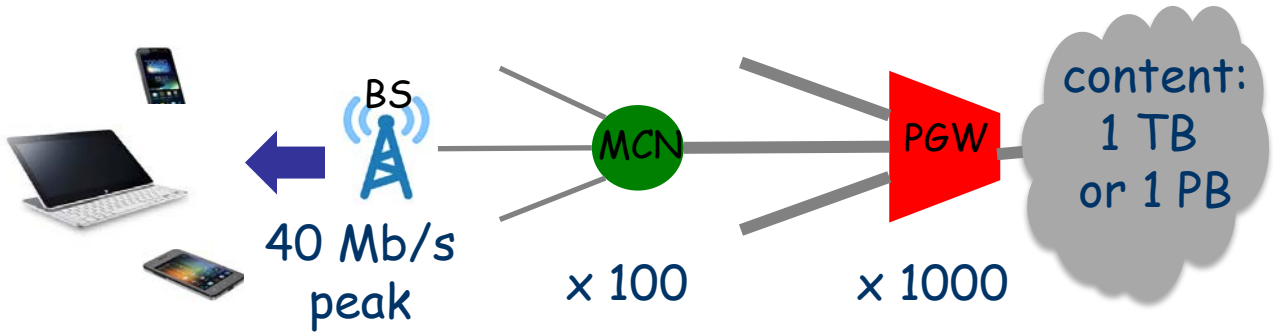
Cost and demand guesstimates

- cost of bandwidth: k_b = \$2 per Mb/s per month
- cost of memory: k_m = \$.03 per GB per month
- if N = 1 PB and T = 40 Gb/s, $\Gamma = k_b T / k_m N = 2.3$
- if N = 1 PB and T = 240 Gb/s, $\Gamma = 14$
- if N = 1 TB and T = 40 Gb/s, Γ 2300

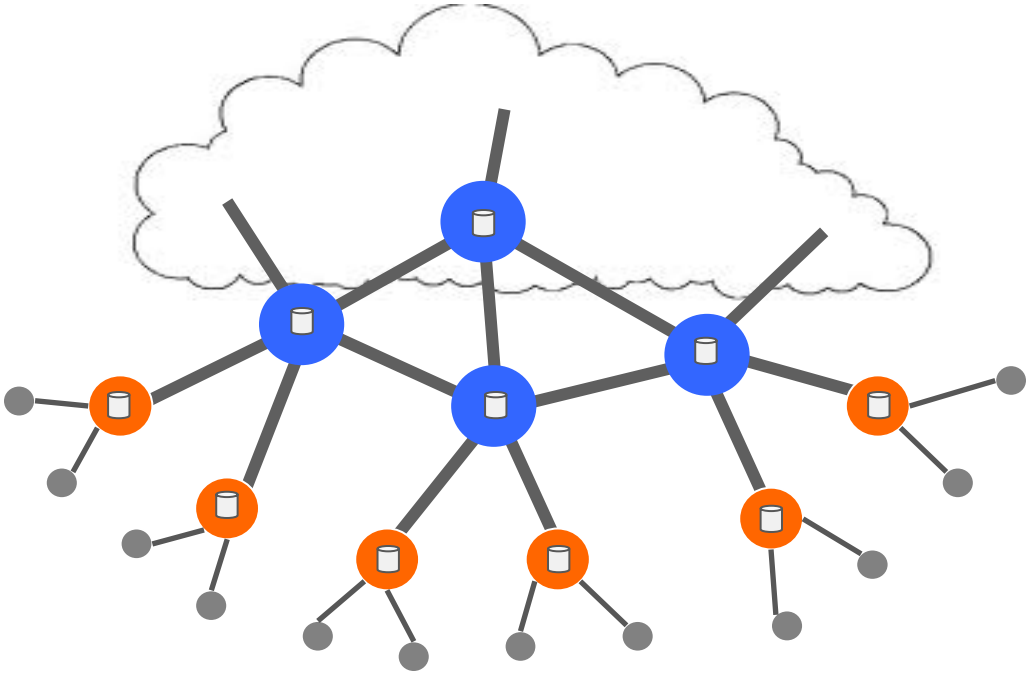


Remarks on tradeoff

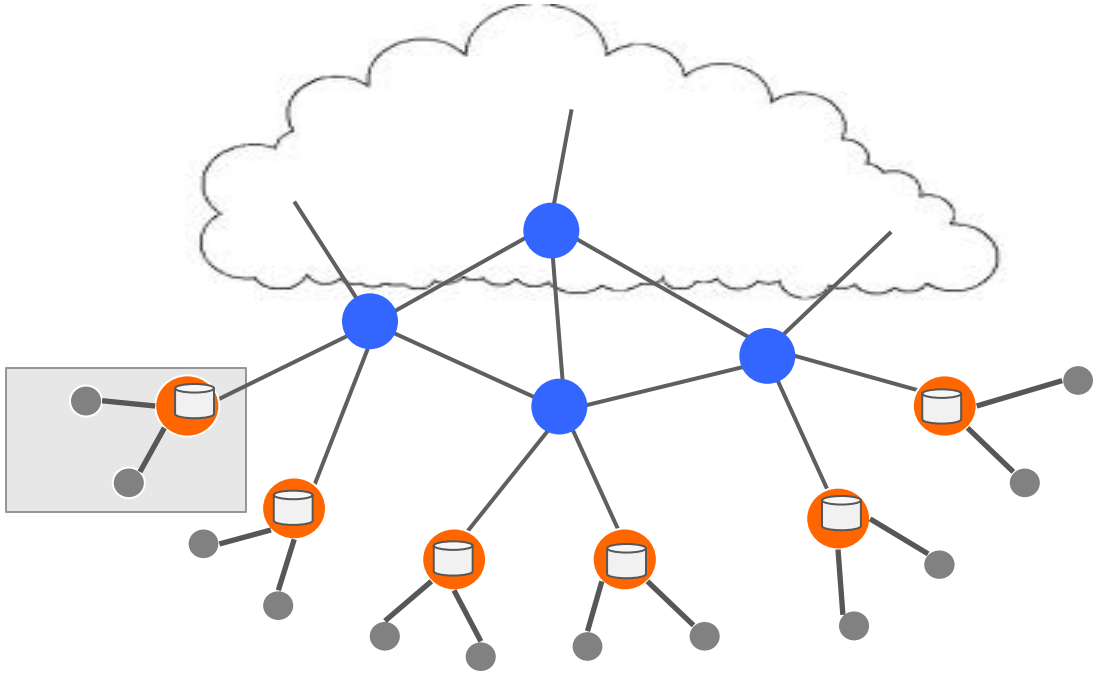
- key factor is $\Gamma = Tk_b / Nk_m$ where N is catalogue size
  - Γ = max bandwidth cost / max storage cost
- cost trends ⇒ Γ is increasing with time
  - k_m decreases by 40% each year, k_b decreases by 20% each year
- tradeoff is favourable at PGW
  - but even more so at "central office" concentrating demand of multiple access networks
- tradeoff at BS or MCN is favourable if N = 1 TB but hardly so if N = 1 PB (see paper...)



Conclusions



Conclusions



Conclusions

- rather than a cache per PGW (and a cache for other access networks), prefer a consolidated large-scale cache at the edge
- proactively cache most popular items lower in the network, as determined by analysis of requests reported to edge node
- proposed methodology and formulas allow repeated evaluation with better guesstimates…