Lectures 9/10: Feb 23.

1. Last class: motivation for RE service
2. RE algorithms (MAXP and MODP for identifying redundancy)
3. Empirical study and cache size
4. ~~Cache~~ size Fast and effective RE.
   - avoid duplicate storage
   - avoid expensive encoding at every hop
   - decouple decoding from its immediate reliance on encoding.

Network-wide optimization using a central routing platform

- Collect traffic and redundancy properties
- Router computation & memory capacities (only for decoding)
- Compute fraction of traffic on a path that a router should cache such that
  - cache occupancy is minimized
  - lookups per second matches router constraint
  - overall network footprint is minimized

- Translate solution in encoding and decoding hash ranges that are non-overlapping

- Bucketize each routers hash so that packet eviction only happens in a bucket corresponding to a router / engine pair.

- Look at overlap of paths to know if a packet is encodable

Speed — 2.5 Gbps for encoding
12 Gbps for decoding.

Applications — - partial / incremental deployment.
- data-centers (servers can also participate)
- multi-hop wireless

Let us now look at the redundancy-awareness paper.

- Implications of universal RE on network protocols.
- Initial ideas on fast stand-alone RE implementation.

~~Impact on~~

RE-awareness impacts both intra and inter-domain routing.

Both cases: estimate redundancy profiles
compute routes that maximize potential for compression.

Inter-domain case: < egress selection
peer selection

Redundancy awareness: (1) for the same peer, pick exit point in redundancy aware manner than based on latency. [compare redundancy with internally-destined traffic)

Allows you to control load due to peer's traffic

② keep egress point selection fixed, but pick peer in a redundancy aware manner than based on BGP policies. : trade-off external and internal costs
could do it on-demand.

③ adapt both egress and peer selection to redundancy very fine grained trade-offs such as the above.

④ greater benefits if neighbors coordinate in their redundancy aware route selection.

Network vs. end-host RE: context: enterprise - DC link or even more generally speaking.

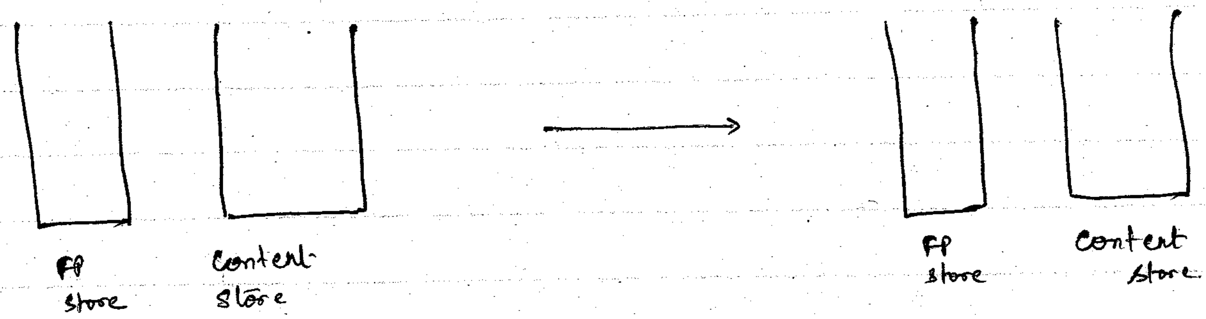
Downsides of network-based approaches
encryption / no last hop benefits

Why not push as high up into end-hosts as possible.
- socket layer as it avoids the need for reinventing RE.
- question: is there enough redundancy?
  ↳ Our measurements says so.

Challenges: optimal use of memory and storage on servers and clients.

Take a look at RE algorithms and data structures again.

Server Client



S → size of packet   w → window   p → "sampling rate"

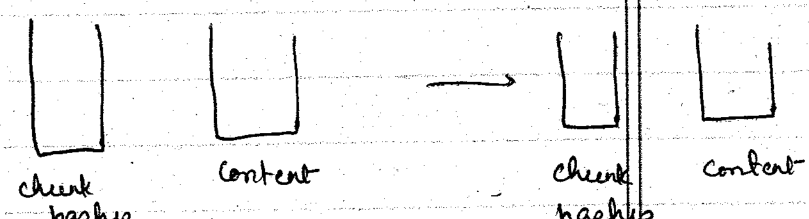
	FIXED	MODP	MAXP	SB
Marker Bytes	Every pth byte		Local maxima over p bytes	look for fixed bytes
Fingerprints	Hash of w-byte region around marker	Rolling R-K hash	Hash of w-byte region around marker	→
Cost	$H\left(\frac{S}{p}\right)$	$RK(w) + H(S-w)$	$A(S) + H\left(\frac{S}{p}\right)$	$A'(S) + H\left(\frac{S}{p}\right)$
Effectiveness rank	4	2	1	3

Chunk Match

↓

Compute (SHA-1) Hashes of byte regions between markers.

Server Client



Cost → overridden by chunk hash cost

but interesting memory savings are possible.

① client does not need chunk hash store. Server can simply send exact client memory address and emulate client eviction policies.

(2) Server does not need to store content as content is stored at the client
Server only needs to do FIFO eviction in CH-store and assume client does similar eviction in his content cache.

↓

control client cache insertion and eviction behavior.