**An Analysis of Facebook Photo Caching**

### Instrumented Stack

**Deep and Distributed**
1. 4 layers of cache and storage.
2. ~12M user IPs, ~20 Point-of-Presence, 4 Datacenters.

- **Browser (millions)**
  - 77.2M user reqs
  - 65.5% hit ratio
  - 65.5% reqs share

- **Edge (dozens)**
  - 26.6M reqs
  - 58% hit ratio
  - 20% reqs share
  - Routing factors:
    - Latency
    - Edge capacity
    - Peering cost

- **Origin (one)**
  - 11.2M reqs
  - 31.8% hit ratio
  - 4.6% reqs share
  - Routed by consistent hashing

- **Haystack**
  - 7.6M reqs
  - 9.9% reqs share
  - Prefers local Haystack

### Workload

At top layers, req popularity follows a power-law dist., but curve flattens as reqs tunnels deeper.

- **Browser Cache**
  - Popularity dist.: Exponential
  - Hit ratio: 37%
  - Traffic: 99.93%
  - L1 miss rate: 0.07%

- **Edge Cache**
  - Popularity dist.: Stretched Exponential
  - Hit ratio: 58%
  - Traffic: 99.87%
  - L1 miss rate: 0.17%

- **Origin Cache**
  - Popularity dist.: Stretched Exponential
  - Hit ratio: 31.8%
  - Traffic: 99.75%
  - L1 miss rate: 0.27%

- **Haystack**
  - Popularity dist.: Consistent hashing
  - Hit ratio: 31.8%
  - Traffic: 99.75%
  - L1 miss rate: 0.27%

### Cache Performance

**Traffic Share by Photo Popularity**

Cache traffic share drop for less popular items.
1. Top 1K photos attract 25% traffic.
2. Cache serves 99.93% reqs for them.
3. Haystack handles the tail.

**Browser Caching**

1. Clients with <10 reqs send 37% traffic.
2. Active clients have higher hit ratio.
3. Increasing cache size helps.

**Edge Caching & Origin Caching**

1. Request from clients are often routed to remote Edges.
2. Collaborative Edges (collab bar) increases hit ratio by 17%.

3. S4LRU increases hit ratio significantly both at Edge and Origin.