Why is P2P the Most Effective Way to Deliver Internet Media Content

Xiaodong Zhang
Ohio State University

In collaborations with
Lei Guo, Yahoo!
Songqing Chen, George Mason
Enhua Tan, Ohio State
Zhen Xiao, IBM T. J. Watson Research

Media contents on the Internet

• Video applications are mainstream

• Video traffic is doubling every 3 to 4 months

No. 3
1. Yahoo
2. Google
3. YouTube
The Power of measurements and modeling

• **Media delivery on the Internet**
  – Internet is an open, complex system
  – Media traffic is user-behavior driven

• **Challenges**
  – Lack of QoS support
  – Lack of Internet management and control for media flow
  – Thousands of concurrent streams from diverse clients

• **Measurements and modeling are critical for**
  – Evaluating system performance under the Internet environment
  – Understanding user access patterns in media systems
  – Providing guidance to media system design and management
Zipf distribution is believed the general model of Internet traffic patterns

- **Zipf distribution (power law)**
  - Characterizes the property of scale invariance
  - Heavy tailed, scale free

- **80-20 rule**
  - Income distribution: 80% of social wealth owned by 20% people (Pareto law)
  - Web traffic: 80% Web requests access 20% pages (Breslau, INFOCOM’99)

- **System implications**
  - Objectively caching the working set in proxy
  - Significantly reduce network traffic

\[ y_i \propto i^{-\alpha} \]
\[ \alpha \approx 0.6-0.8 \]

\( y_i \): number of references  
\( i \): rank of objects

Does Internet media traffic follow Zipf’s law?

- **Web media systems**
  - Chesire, USITS’01: Zipf-like
  - Cherkasova, NOSSDAV’02: non-Zipf

- **VoD media systems**
  - Acharya, MMCN’00: non-Zipf
  - Yu, EUROSYS’06: Zipf-like

- **P2P media systems**
  - Gummad, SOSP’03: non-Zipf

- **Live streaming and IPTV systems**
  - Veloso, IMW’02: Zipf-like
  - Sripanidkulchai, IMC’04: non-Zipf
Inconsistent media access pattern models

- Still based on the Zipf model
  - Zipf with exponential cutoff
  - Zipf-Mandelbrot distribution
  - Generalized Zipf-like distribution
  - Two-mode Zipf distribution
  - Fetch-at-most-once effect
  - Parabolic fractal distribution
  - …

- All case studies
  - Based on one or two workloads
  - Different from or even conflict with each other

- An insightful understanding is essential to
  - Content delivery system design
  - Internet resource provisioning
  - Performance optimization

Challenges of addressing the issues

- Existing studies cannot identify a general media access pattern
  - Limited number of workloads
  - Constrained scope of media traffic
  - Biased measurements and noises in the data set

- Model should be accurate, simple, and meaningful
  - Characterize the unique properties
  - Have clear physical meanings
  - Observable and verifiable predictions
  - Impacts on system designs

- Model validation methodology
  - Goodness-of-fit test
  - Reexamination of previous observations
  - Reappraisal of other models
Research Objectives

• Discover a general distribution model of media access patterns
  – Comprehensive measurements and experiments
  – Rigorous mathematical analysis and modeling
  – Insights into media system designs

Outline

• Motivation and objectives
• Stretched exponential model of Internet media traffic
  • Dynamics of access patterns in media systems
  • Caching implications
  • Concluding remarks
Workload summary

- **16 workloads in different media systems**
  - Web, VoD, P2P, and live streaming
  - Both client side and server side
  - nearly all workloads available on the Internet

- **Different delivery techniques**
  - Downloading, streaming, pseudo streaming
  - Overlay multicast, P2P exchange, P2P swarming
  - all major delivery techniques

- **Data set characteristics**
  - Workload duration: 5 days - two years
  - Number of users: $10^3 - 10^5$
  - Number of requests: $10^4 - 10^8$
  - Number of objects: $10^2 - 10^6$
  - data sets of different scales

Stretched exponential distribution

- Media reference rank follows stretched exponential distribution (passed Chi-square test)

**Probability distribution: Weibull**

$$P(X \leq x) = 1 - \exp[-\left(\frac{x}{x_0}\right)^c]$$

$c$: stretch factor

**Rank distribution:**

- fat head and thin tail in log-log scale
- straight line in log$_{10}$-$y$ scale

$i$: rank of media objects ($N$ objects)
$y$: number of references

$$P(y > y_i) = \frac{i}{N}$$

$$y_i' = -a \log i + b \quad (1 \leq i \leq N, a = x_0^c)$$

$$b = 1 + a \log N \quad \text{(assuming } y_N = 1)$$

- fat head
- thin tail

- $c$: stretch factor

- slope: $-a$
Evidences: Web media systems (server logs)

- HPC-98: enterprise streaming media server logs of HP corporation (29 months)
- HPLabs: logs of video streaming server for employees in HP Labs (21 months)
- ST-SVR-01: an enterprise streaming media server log workload like HPC-98 (4 months)

Evidences: Web media systems (req packets)

- PS-CLT-04: first IP packets of HTTP requests for media objects (downloading and pseudo streaming), 9 days
- ST-CLT-04: RTSP/MMS streaming requests (on-demand media), 9 days
- ST-CLT-05: RTSP/MMS streaming requests (on-demand media), 11 days
Evidences: VoD media systems

- **mMoD-98**: logs of a multicast Media-on-Demand video server, 194 days
- **CTVoD-04**: streaming server logs of a large VoD system by China telecom, 219 days, reported as Zipf in EUROSYS’06
- **IFILM-06**: number of web page clicks to video clips in IFILM site, 16 weeks (one week for the figure)
- **YouTube-06**: cumulative number of requests to YouTube video clips, by crawling on web pages publishing the data

Evidences: P2P media systems

- **KaZaa-02** (300 MB): large video file (> 100 MB. Files smaller than 100 MB are intensively removed) transferring in KaZaa network, collected in a campus network, 203 days.
- **KaZaa-03** (5 MB): music files, movie clips, and movie files downloading in KaZaa network, 5 days, reported as Zipf in INFOCOM’04.
- **BT-03** (636 MB): 48 days BitTorrent file downloading (large video and DVD images) recorded by two tracker sites
Evidences: Live streaming and other systems

Akamai-03: server logs of live streaming media collected from akamai CDN, 3 months, reported as two-mode Zipf in IMC’04


IMDB-06: cumulative number of votes for top 250 movies in Internet Movie Database web site

Why Zipf observed before?

- Media traffic is driven by user requests
- Intermediate systems may affect traffic pattern
  - Effect of extraneous traffic
  - Filtering effect due to caching
- Biased measurements may cause Zipf observation
Extraneous media traffic

- Ad and flag video are pushed to clients mandatorily

Effects of extraneous traffic on reference rank distributions

- Do not represent user access patterns
  - High request rate (high popularity)
  - High total number of requests
- Not necessary Zipf with extraneous traffic
  - Extraneous traffic changes
  - Always SE without extraneous traffic
- Small object sizes, small traffic volume
Caching effect

- Web workload: caching can cause a “flattened head” in log-log scale
- Stretched exponential is not caused by caching effect
- Local replay events can be traced by WM/RM streaming media protocols
  - Before replay: cache validation
  - After replay: send feed back
  - Recorded in server logs
  - Captured in our network measurement

![Graph showing Zipf distribution filtered by Web cache and stretched exponential distribution.](image)

Fetch-at-most-once effect

- SOSP’03: “flattened head” of P2P access pattern
  - Media access pattern is Zipf-like
  - Users fetch a file at most once
    Unlimited cache for all users
- Contradict with streaming media measurements
  - SE access pattern, without caching
- Small streaming media objects
  - Users fetch an object multiple times
- Large streaming media objects
  - User may fetch and view only once
- Conclusion
  - No relation with “fetch-at-most-once”
Why media access pattern is not Zipf

- "Rich-get-richer" phenomenon
  - Pareto, power law, …
  - The structure of WWW
- Web accesses are Zipf
  - Popular pages can attract more users
  - Pages update to keep popular
  - Yahoo ranks No.1 more than six years
  - Zipf-like for long duration
- Media accesses are different
  - Popularity decreases with time exponentially
  - Media objects are immutable
  - Rich-get-richer not present
  - Non-Zipf in long duration

Outline

- Motivation and objectives
- Stretched exponential model of Internet media traffic
- Dynamics of access patterns in media systems
- Caching implications
- Concluding remarks
Dynamics of Access Patterns in Media Systems

- **Media reference rank distribution in log-log scale**
  - Different systems have different access patterns
  - The distribution changes over time in a system (NOSSDAV’02)

- **All follow stretched exponential distribution**
  - Stretch factor $c$
  - Minus of slope $a$

- **Physical meanings**
  - Media file sizes
  - Aging effects of media objects
  - Deviation from the Zipf model

![Graph showing stretched factors of different systems](image)

**Stretched factors of different systems**

KaZaa systems, different file sizes

- Stretch factor $c$
- Median file size

Streaming systems, different file sizes

- Stretch factor $c$
- Median file size

Different systems, similar file sizes

- Stretch factor $c$
- Median file size
Stretched factor and media file sizes

- Other factors besides file size
  - Different encoding rates and compression ratios
  - Video and audio are different
  - Different content type: entertainment, educational, business

![File size vs. stretch factor c]

- 0 – 5 MB: \( c \leq 0.2 \)
- 5 – 100 MB: 0.2 ~ 0.3
- > 100 MB: \( c \geq 0.3 \)

\( c \) increases with file size

Conservations in dynamic media systems

- Media requests over time
  - Constant media request rate \( \lambda_{\text{req}} \)
  - Constant object birth rate \( \lambda_{\text{obj}} \)

\[ N(t) = \lambda_{\text{req}} t + \sum_{0 < \tau} N'(\tau) + O(\log t) \]

Number of accessed objects:

- Objects created in [0, t)
- Objects created in (-\( \infty \), 0]

![Conservations in dynamic media systems graphs]
Stretched exponential parameters

- **In a media system**
  - Constant request rate
  - Constant object birth rate
  - Constant median file size
- Stretch factor $c$ is a time invariant constant
- Parameter $a$ increases with time

$$a = \left[ \frac{\lambda_{req}}{\lambda_{obj}} \frac{1}{1 + \frac{\lambda_{obj}}{\lambda_{req}}} \frac{1}{\Gamma(1 + \frac{1}{c})} \right]^c$$

Evolution of media reference rank distribution

Reference rank distribution
Deviation from the Zipf model

\[
\frac{|EF|}{|OE|} \rightarrow 1 \text{ when } a \log N \rightarrow \infty
\]

\[
a = \left[ \frac{\lambda_{req}}{\lambda_{obj}} \frac{1}{1 + \frac{N^{0.1}}{\lambda_{req}}} \frac{1}{\Gamma(1 + \frac{1}{c})} \right]
\]

- \(a\) increases with \(c\) (\(c < 2\))
- \(a\) increases with \(\frac{\lambda_{req}}{\lambda_{obj}}\)
- \(a\) increases with \(t\)

Big media files have large deviation
Deviation increases with time

Example: YouTube Video Measurements in IMC'07

Campus users: small request rate \(\lambda_{req}\)
Old object dominant
\(a = \left[ \frac{\lambda_{req}}{\lambda_{obj}} \frac{1}{1 + \frac{N^{0.1}}{\lambda_{req}}} \frac{1}{\Gamma(1 + \frac{1}{c})} \right]\)
Small \(a\), small deviation

Global users: large request rate \(\lambda_{req}\)
No old objects
Large \(a\), large deviation
Outline

• Motivation and objectives
• Stretched exponential model of Internet media traffic
• Dynamics of access patterns in media systems
• Caching implications
• Conclusion

Caching analysis methodology

• Analyze caching with reference rank distribution
  – Requests are independent
  – Objects occupy unit storage volume
• Optimal hit ratio
  – Unlimited cache
    \[ H_{opt} = \frac{\text{\# of hits}}{\text{\# of reqs}} = \frac{N(y) - N}{N(y)} = 1 - \frac{1}{\langle y \rangle} \]
    – \( N \) objects, cache size is \( \eta N \)
  • Zipf-like distribution \( H_{Zf}(\eta) = \eta^{1-\alpha} - \eta(1-\alpha) \) for \( \alpha < 1 \)
  • Stretched exponential \( H_{se}(\eta) = \frac{\Gamma(1+\frac{1}{\alpha} - \gamma(1+\frac{1}{\alpha} - \log \eta))}{\Gamma(1+\frac{1}{\alpha} - \gamma(1+\frac{1}{\alpha} - \log \eta))} - \frac{\eta}{\langle y \rangle} \)
Modeling caching performance

Asymptotic analysis for small cache size $k$ ($k \ll N$)

- **Zipf**
  \[
  H_{\text{Zipf}}(k, N) = \sum_{i=1}^{\min(k, N)} \frac{1 - \alpha}{P_i} \times \frac{1}{N^{1 - \alpha}}
  \]

- **SE**
  \[
  H_{\text{SE}}(k, N) = \frac{k}{N} \times \frac{(\log N)^a}{N}
  \]

\[
\lim_{N \to \infty} H_{\text{SE}}(k, N) = \lim_{N \to \infty} c_i \frac{(\log N)^a}{N^\alpha} = 0
\]

Media caching is far less efficient than Web caching

Parameter selection
- **Zipf**: typical Web workload ($\alpha = 0.8$)
- **SE**: typical streaming workload ($c = 0.2$, $a = 0.25$, same as ST-CLT-05)

Potential of long term media caching

- **Short term**
  - Requests dominated by old objects $N'(t) >> \lambda_{\text{obj}}t$
  
- **Long term**
  - Requests dominated by new objects $\lambda_{\text{obj}}t >> N'(t)$

- Optimal hit ratio of caching 10% objects
  - PS-CLT-04: 0.52 for 9 days, 0.85 maximal
  - ST-CLT-04: 0.48 for 9 days, 0.84 maximal
  - ST-CLT-05: 0.54 for 11 days, 0.85 maximal

- Request correlation can be further exploited
  - Object popularity decreases with time

Great improvement when $\lambda_{\text{obj}}t >> N'(t)$
Long time to reach optimal

- Media objects have long lifespan
  - Most requested objects are created long time ago
  - Most requests are for objects created long time ago
- To achieve maximal concentration
  - Very long time (months to years)
  - Huge amount of storage
  - Only peer-to-peer systems provide such a huge space with a long time

Summary

- Media access patterns do not fit Zipf model
- We give reasons why previous results were confusing
- Media access patterns are stretched exponential
- Our findings imply that
  - Client-server based proxy systems are not effective to deliver media contents
  - P2P systems are most suitable for this purpose
- We provide an analytical basis for the effectiveness of a P2P media content delivery infrastructure
Stretched Exponential Distribution:  
Decentralized Content Delivery in Internet

• Centralized Internet accesses follows zipf

• Decentralized Internet accesses (in an organized way, such as P2P) follow SE

• Other P2P-like accesses fitting SE reported since PODC’08
  – IPTV, user channel selection distribution (SIGMETRICS’09)
  – PPLive, P2P streaming request distribution (ICDCS’09)
  – Wikipedia, Yahoo answers, social network posting distribution (KDD’09)
  – Access distribution in PPStream is converting from zipf (2007) to stretched exponential (2009) (a report from Nanjing Statistical Institute)

References

- The stretched exponential distribution, PODC’08
- Social network contributors’ distribution, KDD’09
- PSM-throttling, streaming in WLAN with low power, ICNP’07
- SCAP, wireless AP caching for streaming, ICDCS’07.
- Quality and resource utilization of Internet streaming, IMC’06
- Internet streaming workload analysis, WWW’05
- Measuring and modeling BitTorrent, IMC’05
- Sproxy, caching for streaming, INFOCOM’04
Caching effect on SE distribution

- Pages can be cached by web browser and proxy
- Page reload events can be accumulated over time
  - Trivial in one week
  - Increase with time gradually
- Number of affected objects is small
  - Movie replay events are not common

Page clicks of movie trailers, published in IFILM Web site

Client requests with time

In short duration, media reference rank distribution is stationary
Segment-based streaming media caching

- Streaming media are often partially accessed
  - Segment caching is efficient
- "Ideal" segment reference rank distribution
  - M segments per object, no partial access
  - Two mode SE distribution

```
<table>
<thead>
<tr>
<th>Object reference rank (log scale)</th>
<th>Number of references (y scale)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10^0</td>
<td>10</td>
</tr>
<tr>
<td>10^1</td>
<td>100</td>
</tr>
<tr>
<td>10^2</td>
<td>1000</td>
</tr>
<tr>
<td>10^3</td>
<td>10000</td>
</tr>
<tr>
<td>10^4</td>
<td>100000</td>
</tr>
<tr>
<td>10^5</td>
<td>1000000</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>Segment reference rank (log scale)</th>
<th>Number of references (y scale)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10^0</td>
<td>10</td>
</tr>
<tr>
<td>10^1</td>
<td>100</td>
</tr>
<tr>
<td>10^2</td>
<td>1000</td>
</tr>
<tr>
<td>10^3</td>
<td>10000</td>
</tr>
<tr>
<td>10^4</td>
<td>100000</td>
</tr>
<tr>
<td>10^5</td>
<td>1000000</td>
</tr>
</tbody>
</table>
```

Segmentation: 5 seconds of media data
- Segments rank distribution: two-mode SE
  - Same stretch factor $c$
  - Smaller $a$ than object rank distribution
- Less temporal locality
Segment caching performance

- Segment hit ratio ≈ byte hit ratio
  - Much lower than object hit ratio
- LRU is less efficient for media caching
  - Less request concentration
  - Larger working set
- Segment LFU is even worse
  - Sequential order in an object not captured

Conventional replacement policies are not efficient