PROP: a Scalable and Reliable P2P Assisted Proxy Streaming System

Presented by 江文德
Outline

- Introduction
- System Design
- Performance Evaluation
- Conclusions
Introduction

- Three representative technologies for multimedia streaming:
  - Content Delivery Network (CDN)
  - Server-based Proxy
  - Client-based P2P
Introduction (Cont.)

Advantages and Limitations of the Three Technologies

<table>
<thead>
<tr>
<th></th>
<th>Advantages</th>
<th>Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDN</td>
<td>Performance-Effective</td>
<td>Very Expensive</td>
</tr>
<tr>
<td>Proxy</td>
<td>Cost-Effective</td>
<td>Not Scalable (limited bandwidths and storages)</td>
</tr>
<tr>
<td>P2P</td>
<td>Highly Cost-Effective</td>
<td>Not Guarantee the QoS</td>
</tr>
</tbody>
</table>
Introduction (Cont.)

- A Further Approach
  - Segment-based Proxy Caching
  - Limitations
    - The Limited storage capacity
    - The reservation of continuous bandwidths will limit the number of clients
    - A proxy easily becomes a system bottleneck and forms a single point of failure
Introduction (Cont.)

- PROP
  - Collaborating and coordinating PROxy and its P2P clients
  - Building scalable and reliable media proxy system in a cost-effective way
System Design

- Infrastructure Overview
- P2P Routing and Media Streaming
- Replacement Policies
Infrastructure Overview

- Two Main Components
  - The Proxy
    - The bootstrap site of the P2P overlay network
    - The interface between the P2P system and media servers
  - The Client Peers Connected by a P2P Overlay Network
Each Peer in the PROP System Has Three Functionalities:

- A peer is a client that requests media data
- A peer is a streaming server
- A peer is an index server that maintains a subset of indices of media segments in the system for content locating

P2P Overlay in the System: CAN
The Media Segments and Their Corresponding Indices are Decoupled

The Segment Locating is Conducted in Two Steps:

- Route the request to the peer maintaining the index of demanded segment
- Select a peer that caches a copy of the segment
P2P Routing and Media Streaming

- The DHT Stores *(key, value)* Maps
  - *key* : a GUID (globally unique identifier) hashed from the URL
  - *value* : the index of the segment

- Joining and Leaving P2P Routing
  - Getting a key space zone and take over the corresponding indices
  - Handing over the segment indices and merge the key space zone to a neighbor
Publishing and Unpublishing Media Segments

- `publish( seg_id, location )`
- `unpublish( seg_id, location )`

- `seg_id`: the segment identifier
- `location`: IP address and port number of the peer that caches the segment copy
P2P Routing and Media Streaming (Cont.)

- Requesting and Serving Media Segments
  - \textit{request}(\textit{seg\_id}, \textit{URL})
P2P Routing and Media Streaming (Cont.)

- Requesting and Serving Media Segments

![Diagram of P2P overlay network](image)
P2P Routing and Media Streaming (Cont.)

 Updating Segment Popularity and Utility Values

- PROP uses the *popularity* and *utility* values of segments to manage cached data
- *update*(seg_id, access_info)
- *notify*(peerset, seg_id, value)
  - *peerset* is the peers in the location list of the segment index
  - *value* is the popularity or utility value of the segment
Replacement Policies

- Popularity-based Proxy Replacement Policy
  - The proxy should hold those popular media objects to minimize the performance degradation due to peer failure

- Utility-based Peer Replacement Policy
Popularity-based Proxy Replacement Policy

\[ p = \frac{S_{\text{sum}}}{S_0} \times \min(1, \frac{T_r - T_0}{t - T_r}), \]

- \( T_0 \), the time when the segment is accessed for the first time;
- \( T_r \), the most recent access time of the segment;
- \( S_{\text{sum}} \), the cumulative bytes that the segment has been accessed;
- \( S_0 \), the size of the segment in bytes;
- \( n \), the number of requests for this segment;
Utility-based Peer Replacement Policy

Three Considerations

- Keeping those unnecessary copies of popular objects degrades the cache efficiency
- The cached data is prone to be flushed in a long stream session if LRU replacement is used
- The segments of a media object may be cached in a single peer, thus the availability is sensitive to the peer failure and leaving

\[ u = \frac{(\log p - \log p_{\text{min}}) \times (\log p_{\text{max}} - \log p)}{r^{\alpha+\beta}} \]

- \( r \), the number of replicas of the segment in the system.
Performance Evaluation

- Metrics in the Evaluation
  - Streaming jitter byte ratio
  - Delayed start request ratio
  - Byte hit ratio
Performance Evaluation (Cont.)

Figure 2. Performance evaluation on REAL workload. Left: Streaming jitter byte ratio; Middle: Delayed start request ratio; Right: Byte hit ratio.
Performance Evaluation (Cont.)

- Proxy Load Change
Performance Evaluation (Cont.)

- Replacement Policy Comparisons

Figure 6. Replacement policy comparisons on REAL workload. Left: Streaming jitter byte ratio; Middle: Delayed start request ratio; Right: Byte hit ratio.
Conclusions

- The collaboration and coordination between the proxy and its P2P clients address the *scalability* problem of proxy-based technique, and also *eliminate* the concern of *unstable quality of services* by only relying on self-organized clients.

- The proposed content location mechanism is *efficient* and *cost-effective*.

- The load balance and data locality in the PROP system are determined by the segment replacement policies.