Video Streaming with Peer-to-Peer Multicast

Alex Bikfalvi
For streaming, the overlay is usually a tree rooted at the source.
The nodes of the tree are the peers rather than the routers.
The challenge is building a reliable multicast tree.
Usually multiple overlay trees are used.
## Issues and Requirements

<table>
<thead>
<tr>
<th>Issues</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peer behavior is unpredictable, they can join and leave at any time</td>
<td>NAT traversal, especially for a service that targets home users</td>
</tr>
<tr>
<td>The users may experience a high delay, since the traffic is routed through several peers</td>
<td>Consider the network heterogeneity</td>
</tr>
<tr>
<td>Requires complex protocol exchanges between peers in order to maintain a reliable overlay fabric</td>
<td>Maintain a quality of experience since QoS is not usually considered in P2P</td>
</tr>
<tr>
<td></td>
<td>Real time transmission, especially in communication services</td>
</tr>
</tbody>
</table>
Routing and packet replication shifted to the network edge
Same traffic traverses some links several times
Optimized tree construction is very important
Paramount criteria: parent selection and loop avoidance
Multicast Trees

Depth, Spread and Fan-Out

Depth: 3

Depth: the number of levels in the tree

Spread: ratio between leaf and interior nodes

Fan-out: number of children for interior nodes

Maximum Depth

Maximum Spread

Maximum Latency

Maximum Source Bandwidth
Uplink required bandwidth can be higher than for downlink.

Downstream peer experience depends on upstream peers.
Multicast Trees

Balanced Trees

**Interior Nodes**

\[
\frac{f^d - 1}{f - 1}
\]

**Leaf Nodes**

\[
f^d
\]

**Small Fan-Out**

Fan-Out of 2

Around 50%

**Large Fan-Out**

Fan-Out of 100

Around 1%
SplitStream
Use a different multicast tree for each stripe
Distribute uniformly the peer as interior nodes in each tree
Ideally, interior nodes in one tree are leaf nodes in all others
The main tree where a node is interior is called proper tree

Downlink-uplink ratio controlled by stripes-fan-out ratio
Assign to each peer an identifier within a range (hash space)
Distance between two peers is the difference between IDs

Routing Table

<table>
<thead>
<tr>
<th>Peer ID</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Many entries for closer peers</td>
</tr>
<tr>
<td>Large</td>
<td>Fewer entries for distant peers</td>
</tr>
</tbody>
</table>

The Idea

- Assign to resources key identifiers in the hash space
- Publish the resources at peers closest to the keys
- Locate those peers using DHT routing

Use a DHT-based P2P protocol: Pastry and Stripe
Routing in Pastry

Initiator

Hash space

Group tree root node

Resource key
Groups are assigned unique hash keys.
The node closer to each group key is the root for that group.

How to create interior-node disjoint multicast trees?

- Create group keys different in most significant digits
- A group interior node should have an ID with the same significant digits
Example: Hash space of 4 hexadecimal digits (16 bits) with 16 groups
Joining and Rejection

How to join a group?

- Lookup the group key
- The routing should stop at first peer along the route that belongs to the group

What happens when a node inside a group is found?

- If the fan-out limit of the candidate parent is not reached, **adopt the new node**
- Otherwise... **reject one node**
What node to reject?

- Match criteria in the following order:
  - Children belonging to other groups
  - Children belonging to the same group and having the greatest distance to the group key

ID: 089x  Gr: 0800
ID: 08Bx  Gr: 0800
ID: 081x  Gr: 0800
ID: 9xxx  Gr: 1800
ID: 08xx  Gr: 0800
Joining and Rejection

What node to reject?

- Match criteria in the following order:
  - Children belonging to other groups
  - Children belonging to the same group and having the greatest distance to the group key

- ID: 08xx  Gr: 0800
- ID: 089x  Gr: 0800
- ID: 08Bx  Gr: 0800
- ID: 081x  Gr: 0800
- ID: 001x  Gr: 0800
- ID: 9xxx  Gr: 1800

Fan-out limit: 4

Orphan
Match criteria in the following order:
- Children belonging to other groups
- Children belonging to the same group and having the greatest distance to the group key

Fan-out limit: 4

ID: 085x  Gr: 0800

ID: 089x  Gr: 0800
ID: 08Bx  Gr: 0800
ID: 081x  Gr: 0800
ID: 001x  Gr: 0800
What node to reject?

- Match criteria in the following order:
  - Children belonging to other groups
  - Children belonging to the same group and having the greatest distance to the group key

ID: 08xx
Gr: 0800

Fan-out limit: 4

Orphan

ID: 001x
Gr: 0800

ID: 089x
Gr: 0800
ID: 08Bx
Gr: 0800
ID: 081x
Gr: 0800
ID: 085x
Gr: 0800
Joining and Rejection

What do orphans do?

- Search for another parent
- If none is found, search for parent in the spare capacity tree
- If none is found... bad luck

Spare Capacity Tree

- A tree with all nodes that have not reached their fan-out limit
P2P Cast
Introduction

Same principles like in SplitStream

What is Different?

- Classify the nodes in
  - Incomplete: fan-out not reached in the proper tree
  - Complete: fan-out limit reached in the proper tree
  - Only-child: a leaf node

Let’s assume an overlay with 2 tree groups and fan-out 2 for all peers
Joining

Joining an incomplete node

Joining a complete node
Departures & Failures

Departure of an incomplete node
Proximity Awareness

In usual DHT P2P networks...
Proximity Awareness

PeerCast proposal...

Landmark points

Landmark vectors
Multicast Management

- Service (group ID) advertised on an off-band channel
- Peer with closest ID becomes a rendezvous point
- New peers will lookup the neighbors
- Only if none of the neighbors are in the group, a lookup towards the group ID is performed