Tracker-based Peer Selection using ALTO Map Information

draft-yang-tracker-peer-selection-00

Y. Richard Yang
Richard Alimi, Ye Wang, David Zhang, Kai Lee

IETF 79 - Beijing, China - Nov 2010

Challenges

- Tracker Scalability
  - Many peers distributed in many torrents/channels
  - Many ISPs providing ALTO info
- Application-Network Information Fusion
  - Application requirements/policies
  - Application endpoint info
  - Network providers' ALTO info
  - Third-party database info

Simple Representation

- Peer Table

Peer ID	IP Address	Upload Capacity	Play Point	ALTO Network Info	City	…
GH4C9	128.36.233.1	512 kbps	01:19:21	**pid1.yale**	New Haven	…
J8NRE	130.132.10.2	10 Mbps	00:05:37	**pid2.yale**	Unknown	…
…	…	…	…	**…**	…	…

- Cost Tables, e.g.,
  - ALTO Cost table

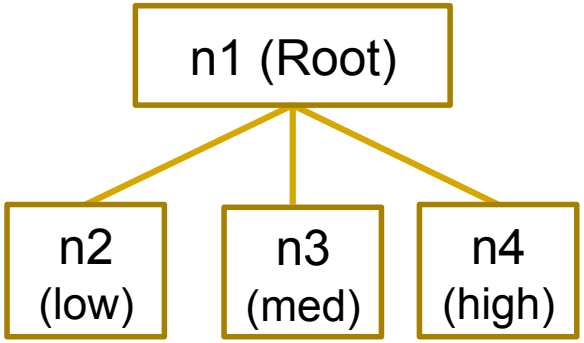
- Problem: scanning Peer Table to select peers can be inefficient

Peer Selection using Classification

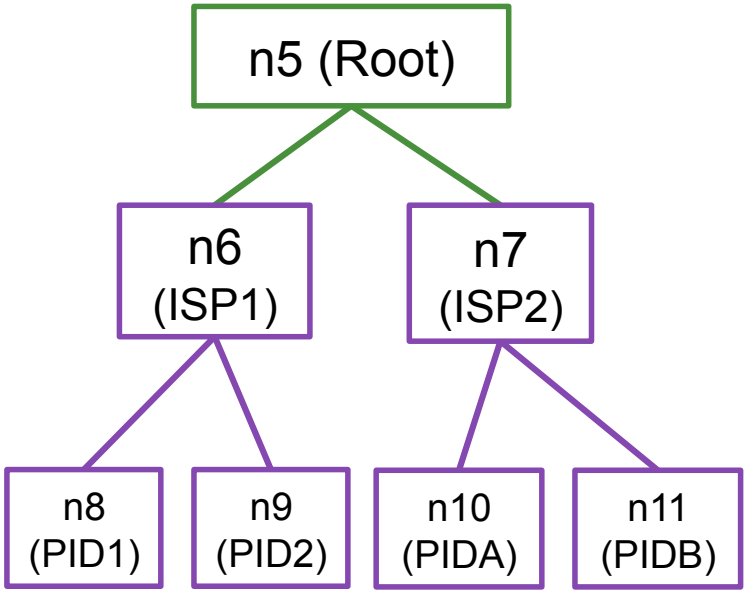
- Objective
  - Aggregation to improve scalability
    - only need to match categories → **better scalability**
- Many Classification Attributes, e.g.,
  - Upload capacity class, play point cluster, ALTO Network Map
- Multi-Dimension Classification
  - Classify peers using multiple attributes, e.g.,
    - Level 1: ALTO Network Map
    - Level 2: Upload capacity

Peer Classification: an Example

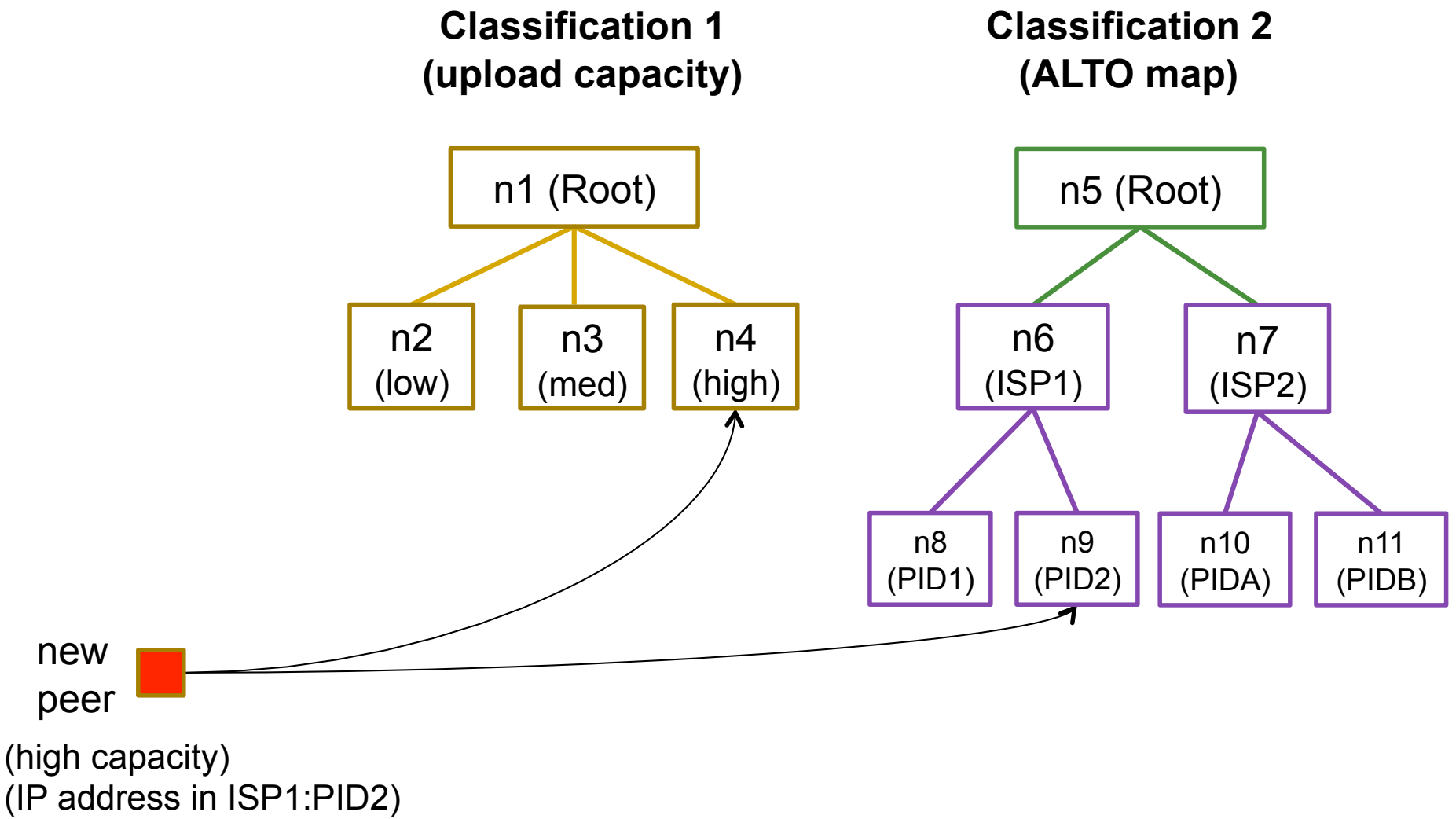
Classification 1
(upload capacity)



Classification 2
(ALTO map)



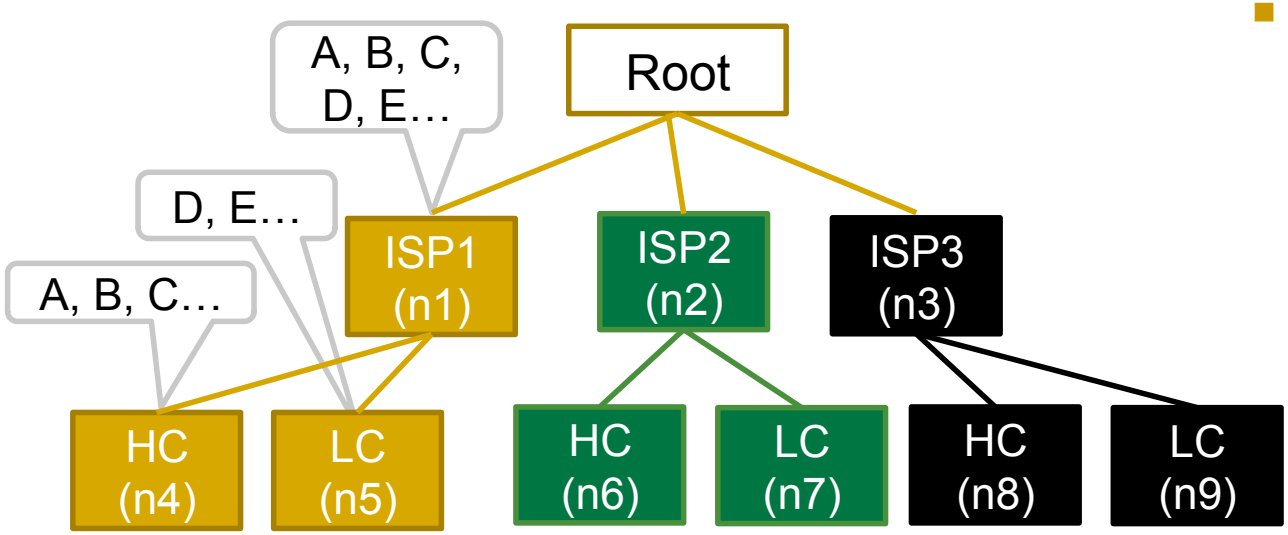
Peer Classification: an Example



Peer Selection using Classification

- Home Node
  - A *leaf* category node where the peer issuing LISTING request belongs to
- Peer Selection Sequence
  - A mapping: from a home node to a traversal *sequence* of category nodes, with a specified target fraction to be reached upon visiting each node
- Peer Selection Process
  - *Sequentially* follow the nodes in the sequence in order

An Example



Peering Selection Table

n4→[n4, 50%] [n5, 80%] [n2, 95%] [n3, 100%]
n5→[n4, 20%] [n5, 60%] [n2, 95%] [n3, 100%]
…
n9→…

- Peer A in n4 (*home leaf*) requests 50 neighbors:

  - select **up to 25 (50%)** peers from **n4**;

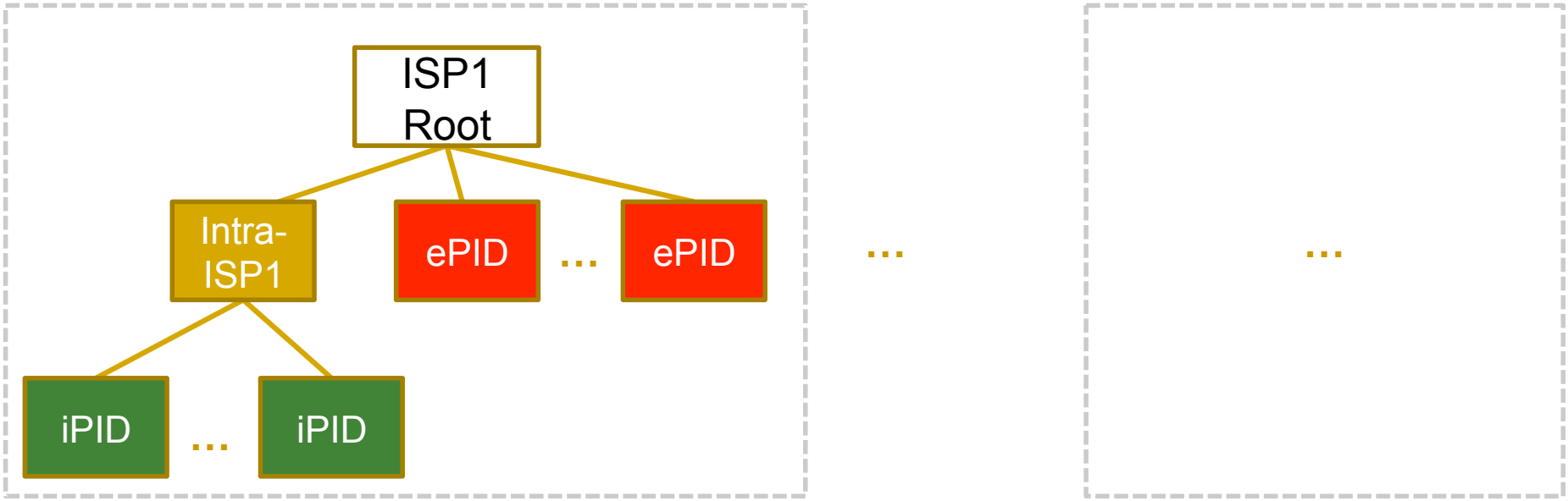
  - continue to reach **up to 40 (80%)** from **n5**;

  - continue to reach **up to 49 (95%)** from **n2**;

  - continue to reach **up to 50 (100%)** from **n3**

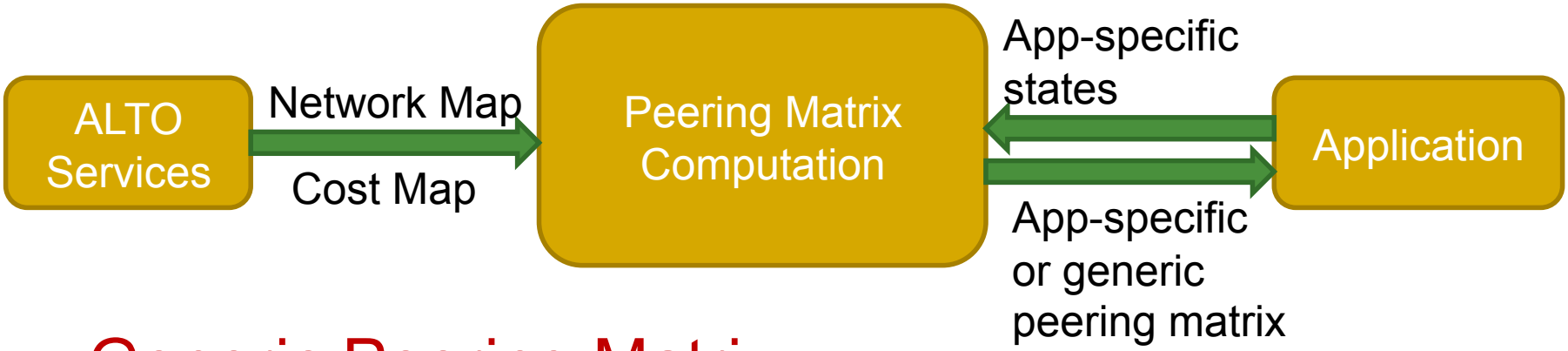
Simple Peer Classification using ALTO Network Maps

- One three-layer classification tree using the ALTO Network Map of each ISP
  - Used in P4P trials
  - Can be used with distributed trackers (one tracker per ISP)



Peering Matrix Computation

- Bandwidth Matching
  - Consider both application requirements and ALTO info
    - for each PID, tracker periodically estimates aggregated upload capacity and download demand
    - use bandwidth matching algorithm to compute weights



- Generic Peering Matrix
  - bandwidth matching, assuming uniform supply and demand across PIDs

ALTO/P4P Library for Tracker Peer Selection

Peer Selection:
-Upon peer LISTING request, selects peers according to classification

Application Optimization Engine (peering matrix computation; a separate thread or machine)

Peer Classification:

-Upon peer arrival, looks up new address in each ALTO Network Map

Peer Update:
-Upon peer keep alive, update statistics of classification

ALTO Info update (run in its own thread)

Benchmarking

1,000,000 peers; Network Map/ Cost Map with 10 to 30 PIDs

Peer Selection:

- **Join rate (classification + peer selection): 25,000 peers/sec in single thread**

Peer Classification:

-Lookup rate: ~2,000,000 lookups/sec using Patricia tree

Peer Update

Application Optimization Engine (if state dependent, 5 sec)

ALTO Info update (run in its own thread; potentially slowest)

Memory: < 150 MB

PlanetLab Experiments

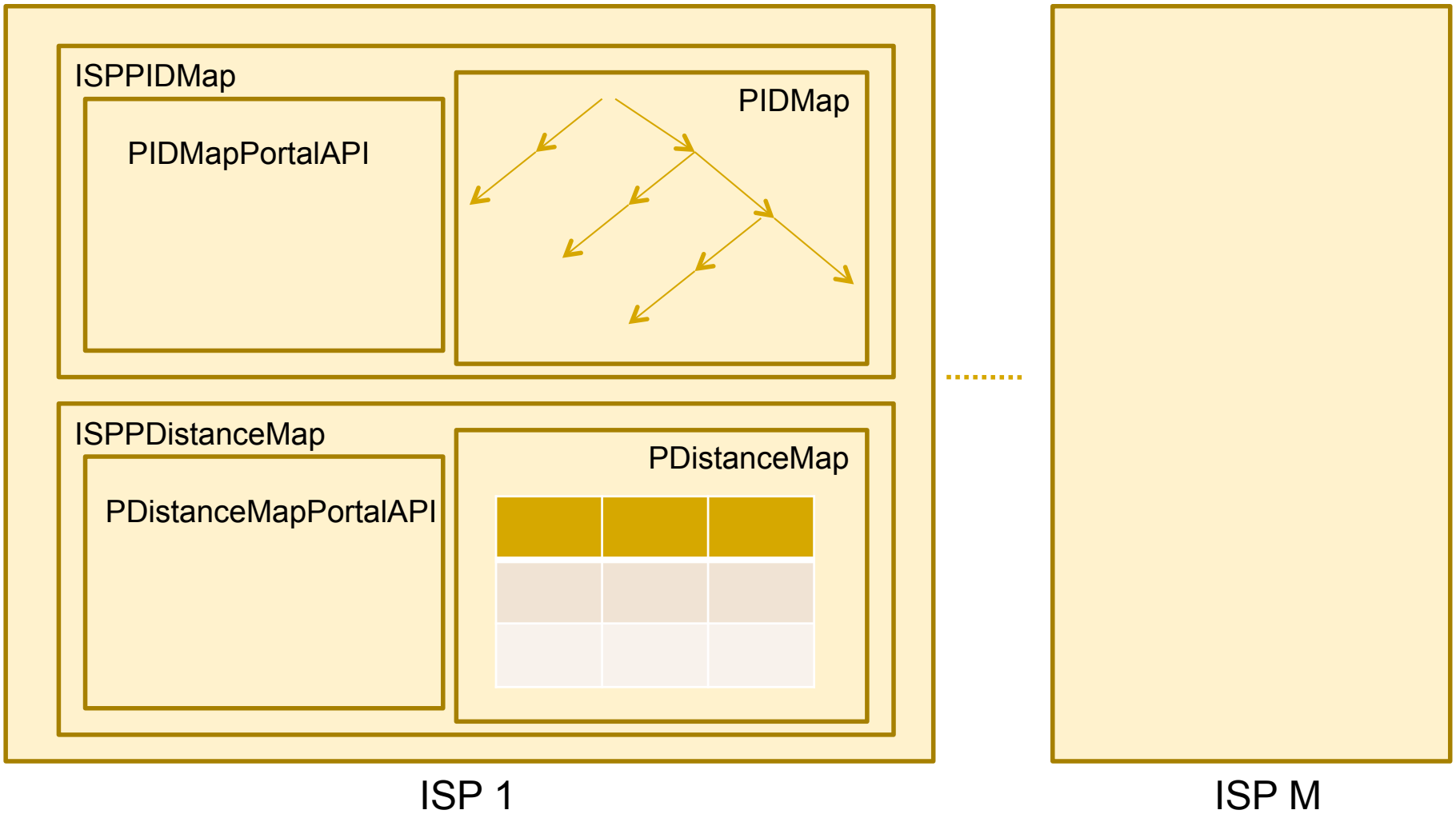
- Experiment Setup
  - ~2500 P2P live streaming clients
    - 4 instances running on each PlanetLab node
  - Three emulated ALTO servers
    - US, Europe, Asia
  - Generic Peering Matrix using the library
- Results

	Metric	w/o ALTO	w/ ALTO
Network Efficiency	Intra-US supply ratio	25.9 GB (/44.1 GB) **58.7%**	40.9 GB (/52.6 GB) **77.8%**
	Intra-PID supply ratio	6.9 GB **15.6%**	22.7 GB **43.2%**
Application Performance	Avg. Playback Startup Delay	**31.1** seconds	**26.9** seconds
	#Playback Freezes	**106** for all clients	**52** for all clients

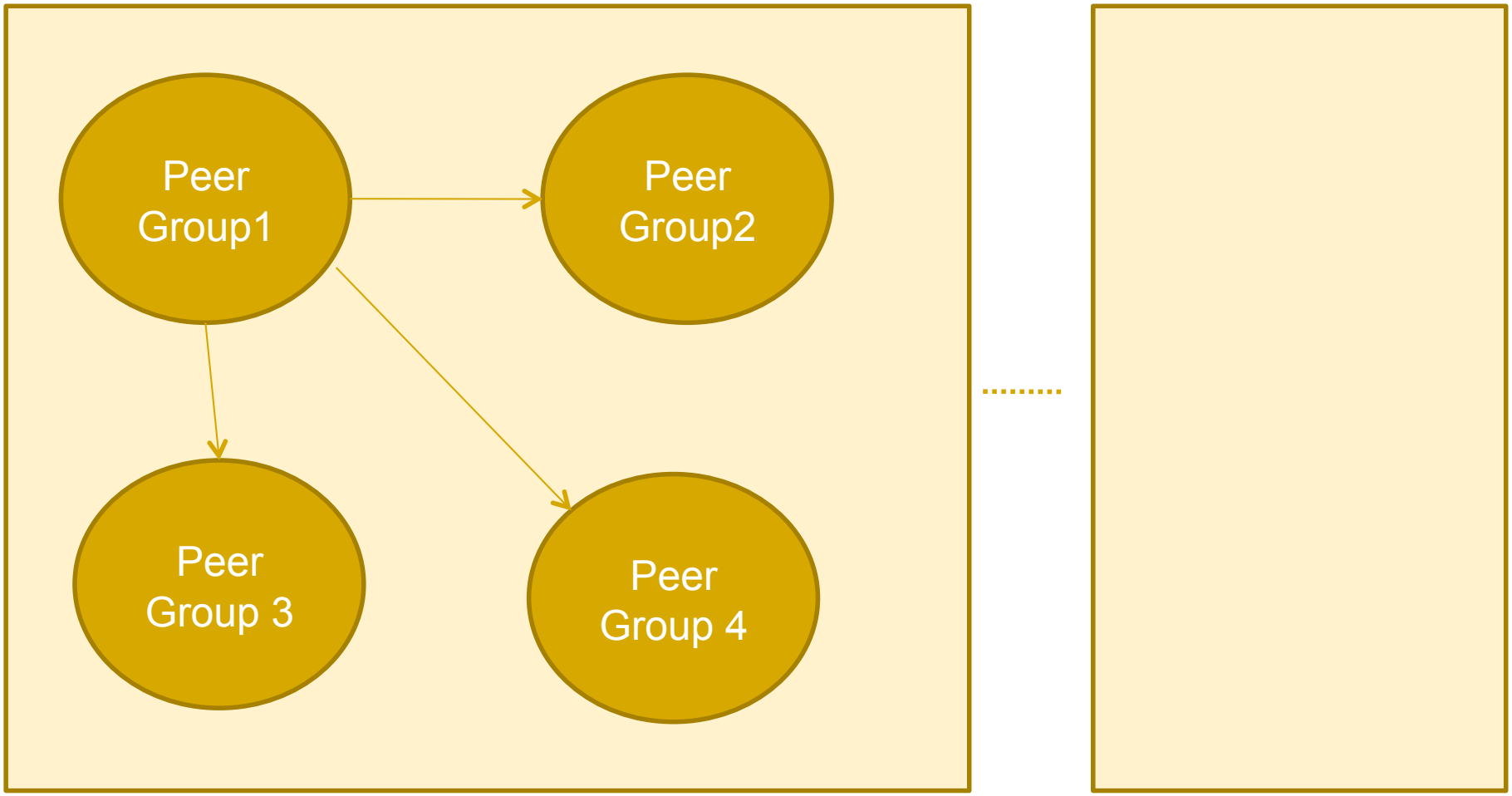
Thank you!

Backup Slides

Tracker ISP Data Structures



Per Channel Data Structures



Peering Matrix

- A data structure to implement peering weights of per-ISP classification tree
  - each entry in matrix encodes peering weight from row to column
  - complexity: O(N)

	iPID1	iPID2	iPID3	iPID4	intra ISP	ePID1	ePID2	ePID3
iPID1								
iPID2								
iPID3								
iPID4								

Implementation Experience

- Processing Complexity
  - Peer IP lookup in peer classification
    - We use Patricia Trie for IP address lookup: > **2,000,000/second**
      - extended LC-trie can be more efficient
      - hash map is slow
- Multi-thread Processing
  - ALTO info update should run in a thread
    - periodically refresh ALTO maps
    - Network Map update triggers Cost Map update and peer classification update
      - slowest part
    - Cost Map update triggers Peering Matrix update
      - e.g., by calling AOE
  - Can run multi-thread workers for peer classification
    - if peer arrival/departure rate is high