PlanetP: Using Gossiping to Build Content Addressable Peer-to-Peer Information Sharing Communities

HPDC’03
IEEE Computer Society

J. L. Chiang, Sept. 10, 2003
Outline

• Introduction
• PlanetP
• Gossiping
• Content Search and Retrieval
• Performance
• Conclusions
Introduction

• Advantages of P2P:
  – Distributed shared data
  – Incremental scalability
  – Member size scalability

• Infrastructures of P2P:
  – Unstructured
  – Structured (cluster / tree / ring / hypercube)
PlanetP Overview

- PlanetP is a content addressable publish/subscribe service that uses gossiping to replicate global state across *unstructured* communities.
- PlanetP is comprised of two components:
  - an infrastructural gossiping layer
  - a content search and ranking algorithm
PlanetP Features

• PlanetP is simple:
  – Each peer only perform a periodic, randomized, point-to-point message exchange with other peers, rather than collaborate to correctly and consistently maintain a complex distributed data structure.

• PlanetP is powerful:
  – It maintains a globally content-ranked data collection w/o depending on centralized resources or the on-line presence of specific peers.
Gossiping

• Epidemic algorithms for replicated database maintenance [4]
• NNTP (Network News Transfer Protocol)
• Distributed system monitoring, management, and data mining
• Probabilistic reliable multicast in ad hoc networks
• Probabilistic reliable dissemination
• Peer-to-peer (group) membership management
Gossiping - Push

(1) Every $T_g$ seconds, $x$ would push this change (rumor) to a peer chosen randomly from its directory.

(2) If $y$ has not seen this rumor, it records the change and also starts to push the rumor just like $x$.

(3) $x$ stops pushing the rumor after it has contacted $n$ consecutive peers that have already heard the rumor.

A change in shared data
Gossiping – Pull (anti-entropy)

(1) Every $Tr$ rounds, $x$ would attempt to pull information from a random peer instead of pushing.

(2) $z$ would reply a summary of its version of the shared data.

(3) $x$ would update any new information that it does not have.
Questions

- How long will it take that all peers learn the new information?
  - It is proven that the time for gossip to disseminate in a flat population grows logarithmically with the size of the population, even in the face of network links and participants failing with a certain probability. [4]

- How to deal with rapid changes in the directory in a dynamic P2P environments?
  - Increase pull rate?
  - Partial pull (anti-entropy)
(1) Every $T_g$ seconds, $x$ would push this change (rumor) to a peer chosen randomly from its directory.

(3) $x$ stops pushing after it has contacted $n$ consecutive peers that have already heard the rumor.

(2a) If $y$ has not seen this rumor, it records the change and also starts to push the rumor just like $x$.

(2b) $y$ also piggybacks the identifiers of a small number $m$ of the most recent rumors that $y$ learned about but is no longer actively spreading onto its reply to $x$. 
Content Search and Retrieval

- PlanetP uses a two-stage search process to perform exhaustive searches while limiting the size of the globally replicated index.
- Global index: PlanetP uses gossiping to replicate a term-to-peer (t->p) index everywhere for communal search and retrieval.
- Local index: PlanetP uses vector space ranking model to find highly relevant documents.
Vector Space Ranking Model

• Document/query $\rightarrow$ Vector

$\text{Vector} = \{ \text{term + weight} \}^*$

$\text{Sim}(Q, D) = \frac{\sum_{t \in Q} w_{Q,t} \times w_{D,t}}{\sqrt{|Q| \times |D|}}$

• $Q$: Query; $D$: Document
• $|Q|$ and $|D|$: # of terms in $Q$ and $D$
• $w_{Q,t}$: the weight of term $t$ for query $Q$
• $w_{D,t}$: the weight of term $t$ for query $D$
TFxIDF

- TFxIDF is a popular method for assigning term weights.

\[ IDF_t = \log(1 + \frac{N_c}{f_t}) \]

\[ w_{D,t} = 1 + \log(f_{D,t}) \quad w_{Q,t} = IDF_t \]

- \( N_c \): # of documents in the collection
- \( f_t \): # of times that term \( t \) appears in the collection
- \( f_{D,t} \): # of time term \( t \) appears in document \( D \)
Ranking Peers

- **IPF (inverse peer frequency):** $\log(1 + \frac{N}{N_t})$

$$R_p(Q) = \sum_{\{t \in Q | (t \rightarrow p) \in I\}} IPF_t$$

- **N:** # of entries in the directory
- **N_t:** # of “t->*” entries in the global index
- **I:** the global index
Retrieving Documents

• Given a pair \((Q,k)\), \(k\) is the required # of potential documents for a query \(Q\).

1. Rank peers for \(Q\).

2. Contact peers in groups of \(m\) from top to bottom of the ranking.

3. Each contacted peer returns a set of document with their relevance.

4. Stop contacting peers when the documents identified by \(p\) consecutive peers fail to contribute to the top \(k\) ranked documents.
Global Index

- Bloom filter [1] is an array of bits used to represent a set of strings.
Search Efficacy

- Comparison with CENT (centralized TFxIDF)

\[ R(Q) = \frac{\text{no. relevant docs. presented to the user}}{\text{total no. relevant docs. in collection}} \]

\[ P(Q) = \frac{\text{no. relevant docs. presented to the user}}{\text{total no. docs. presented to the user}} \]
Gossiping Performance

- Propagation time is a logarithmic function of community size ➔ Scalable
- Bandwidth requirements is modest ➔ Scalable
Gossiping Performance

- Time required to reach consistency when new members simultaneously join the community.
Gossiping Performance

- To required to convergence when nodes rejoin the community.
Conclusions

- PlanetP provides powerful information sharing infrastructure while avoiding cost, privacy, and safety concerns in centralized scheme.
- PlanetP supports distributed content search, ranking and retrieval, and uses gossiping robustly disseminate new information even in a dynamic P2P environment.
- Future work focus on grouping to support vary large communities and building applications to validate the utility of PlanetP.
P2P Researches

- Multimedia streaming services
  - Fault-tolerant, layered media, content-based retrieval
- Performance improvement
  - Cache, bandwidth aware transmission
- Security of P2P system
  - Trust model, secure group communication
- Reliable multicast in Ad hoc P2P network
- P2P e-commerce