Comparing Hybrid Peer-to-Peer Systems

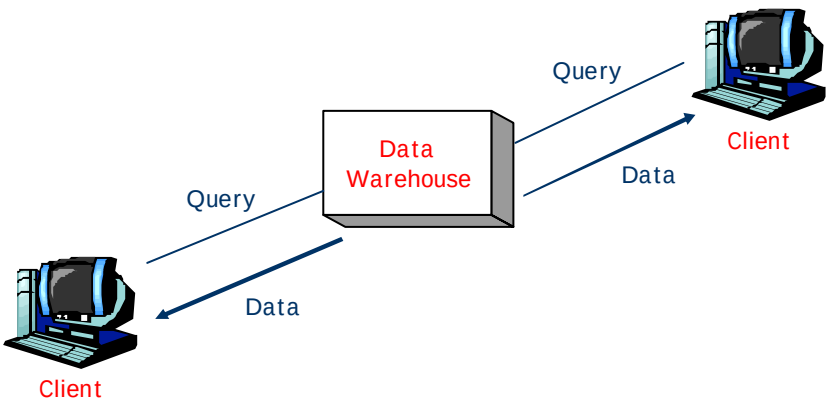
Paper by B. Yang, H. G. Molina

Presentation by Jin Xiao

Presentation Outline

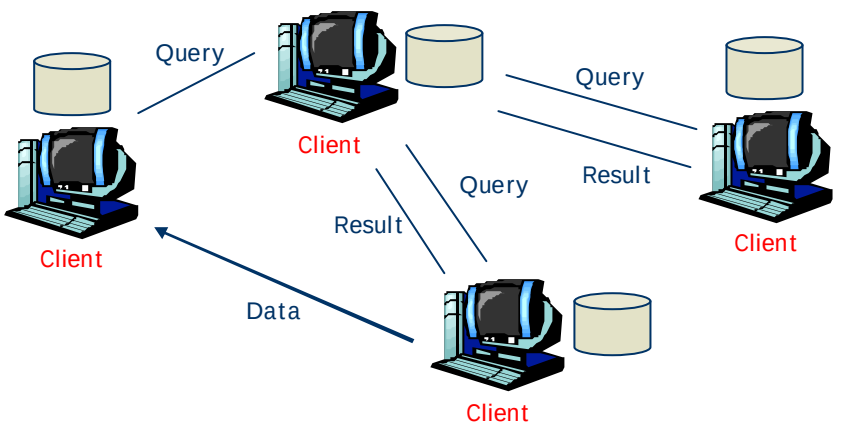
- Peer-to-Peer Systems
- Architecture of Hybrid P2P systems
- Modeling P2P Systems
- Query modeling
- Performance modeling
- Conclusion

Data-sharing: Centralized Model



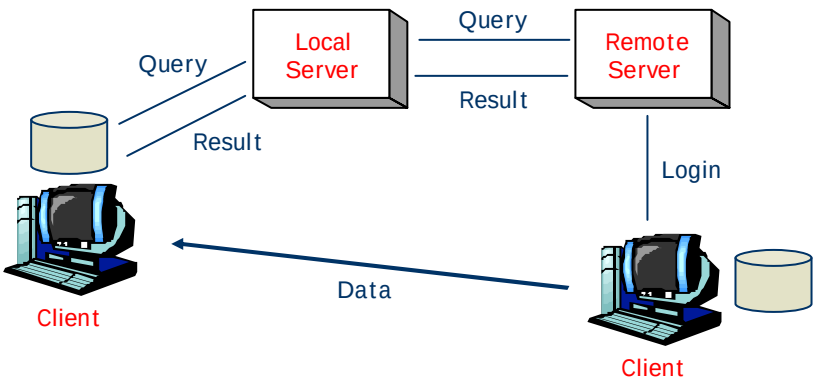
3

Data-sharing: Pure Peer-to-Peer



4

Data-sharing: Hybrid Peer-to-Peer



5

Actions in Hybrid Peer-to-Peer System

- Login
  - Index tables
  - Batch logins vs. incremental logins
- Query (Search)
  - File-level search
  - Key word matching vs. regular expression search
- Download

6

Hybrid Peer-to-Peer Architectures

- Chained
  - Large and diverse data, not many servers
- Full Replication
  - Small amount of servers, WAN connection
- Hash
  - Infrequent update, LAN connection
- Unchained
  - High data redundancy, many servers

7

Modeling Peer-to-Peer Systems

We need a method to compare these architectures.

Peer-to-Peer system is complex to analyze, a simplified model could work.

- Query modeling
- Performance modeling
- Validation with real data (OpenNap)

8

Query Modeling

f – query "selection power"

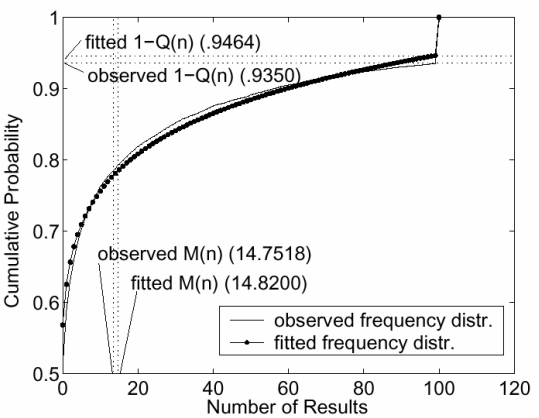
g – query popularity

- Correlate f and g with exponential distribution

$$g(i) = \frac{1}{\lambda_g} e^{-\frac{i}{\lambda_g}}$$

9

Query Modeling



10

Comparing P2P Systems using Query Modeling

- Expected number of servers (ExServ)
  - Chained: $ExServ = k - \sum_{s=1}^{k-1} Q(s \cdot UsersPerServer \cdot FilesPerUser)$
  - Full replication & Unchained: 1
- Expected Results (ExLocal + ExRemote)
  - Chained: M(y) + M((k-1) × y)
  - Full replication: M(N) + 0
  - Unchained: M(n) + 0
  - Hash: M(N) + 0

$$\sum_{i=0}^{\infty} g(i)\left(R - \sum_{m=0}^{R-1} \binom{n}{m} (f(i))^m (1-f(i))^{N-m} (R-m)\right)$$

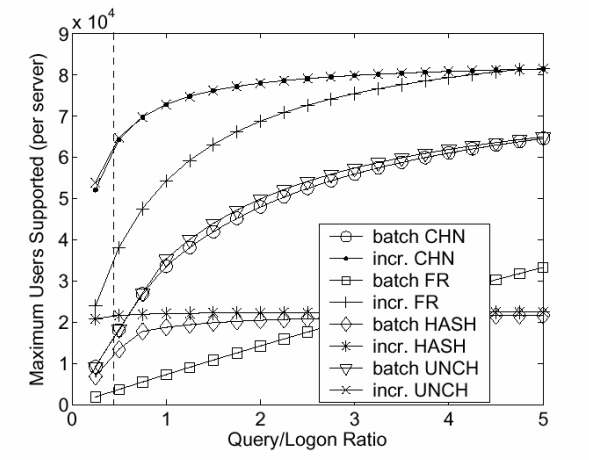
11

Performance Modeling

- Defined metrics for the following aspects:
  - CPU consumption
  - Network usage
  - Memory requirement
- The actions being considered are:
  - Login, query, and download

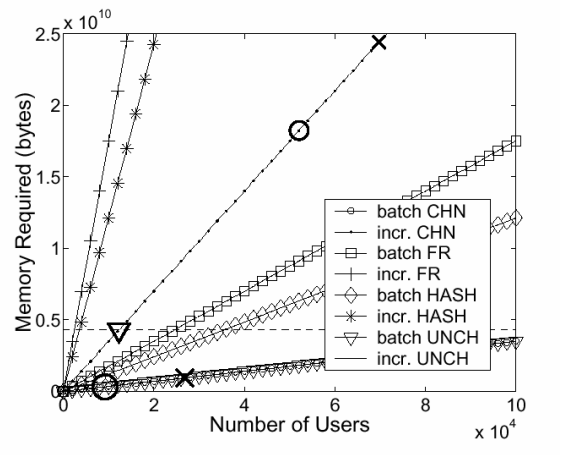
12

Comparing P2P Systems using Performance Modeling



13

Comparing P2P Systems using Performance Modeling



14

Conclusion

- Incremental login scales better than batch login
- Unchained architecture results in lower number of results returned
- Hash architecture is resource intensive
- Full replication works well when result sets are large
- Chained architecture is well-suited for music-sharing, but suffers from poor query performance