Research on Replication

Topics and resolutions

Feng Mao
Cluster and Grid Computing Lab
Huazhong University of Sciee & Tech
fmao@hust.edu.cn
2004.3
Acknowledgments

- Group member:
  Hai Jin (PhD)
  Song Wu (PhD)
  Deqing Zou (PhD)
  Muzhou Xiong (PhD)
  Li Qi (Ms.)
  Baoli Chen (Ms.)

- China grid project
Outline

- Background on replication
- Replica location
- Replica placement
- Replica selection
- Current project
Back ground on replication

Replication vs. Caching

Cache

1. memory hierarchy $\rightarrow$ cache proxy
2. reference locality, data will be accessed again in the near further
3. Sharing mode
Back ground on replication

Sharing by one user

CPU → cache → Memory → Hard disk

Sharing by multi-users

Cache proxy
Back ground on replication

Replication vs. Caching

Replication

- Data replication → Service replication
- Speed → QoS of availability
Back ground on replication

- Cache
  - access latency / speed
  - only help on multi-sharing mode

- replication
  - concurrence, high availability, soft real-time, state, cooperation
Back ground on replication

- Cache ---the simplest replication
document, simple static data, web page

- replication
document, service, process, minor web site
Back ground on replication

- P2P (peer to peer) mode
  Cooperation between replicas
  - P2P mode

1. Symmetric communication
2. combined client and server role
Back ground on replication

Why P2P mode

- [Dynamic operability] P2P applications must keep operating transparently although hosts join and leave the network frequently.
- [Performance and scalability] P2P applications exhibit what economists call the “network effect” in which a network’s value to an individual user scales with the total number of participants.
- [Reliability] External attacks should not cause significant data or performance loss.
- [Anonymity] The application should protect the privacy of people seeking or providing sensitive information.
Outline

- Background on replication
- Replica location
- Replica placement
- Replica selection
- Current project
Replica location

- Projects in history

  Napster → Gnutella → Free net → Chord /Pastry/CAN
Napster

- Free music over the Internet
- Key idea: share the storage *and* bandwidth of individual (home) users
- centralized server
- PP, BT
Gnutella

- Idea: multicast the request
- Send request to all neighbors. Neighbors recursively multicast the request. Eventually a machine that has the file receives the request, and it sends back the answer.
- Not scalable;
Freenet

- Each file is identified by a unique identifier
- Each machine stores a set of files, and maintains a “routing table” to route the individual requests
- Balance between scalability and availability
Chord /Pastry/CAN

Ring

Hypercube

Mesh
Outline

- Background on replication
- Replica location
- Replica placement
- Replica selection
- Current project
Replica placement

- How many to replica?
- Where to replica (local or time locality)?
- Local or global benefit?
- Storage or bandwidth? (time out)
Replica placement

- Static placement (startup)
- Dynamic placement (runtime)
Outline

- Background on replication
- Replica location
- Replica placement
- Replica selection
- Current project architecture
Replica selection

- Local or remote
Outline

- Background on replication
- Replica location
- Replica placement
- Replica selection
- Current project architecture
Current project architecture
Current project architecture

- **Core services**
  - Replica Locate -- RLS (replica location service)
  - Replica placement -- RPS (replica publish service)
  - Replica selection -- RMS (replica monitor service)

- **High level services**
  - Replica
Thank You

fmao@hust.edu.cn