CSCI 5105

Instructor: Abhishek Chandra

Today

- Data Replication and Consistency
  - Replica Management

Replica Management

- Where, when and by whom should replicas be placed?
- Replica-server placement: Where to place replica servers
- Content placement: Which servers to place what content on

Replica-server Placement

- Where to place replica servers?
- Factors:
  - Client locations
  - Network topology and properties
  - Workload type

Client Location-based Replica-server Placement

- Place servers at the best K out of N locations
- Based on distance between clients and locations
  - Distance can be latency or bandwidth
- Pick one server at a time:
  - Minimize average distance between server and clients

5

Network Topology-based Replica-server Placement

- Considers the Internet topology
  - Consisting of Autonomous Systems (ASs)
- Assumes clients are uniformly distributed
- Assigns replicas to ASs in order of their size
  - To the most connected servers

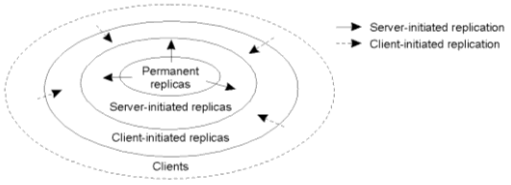
6

Content Replication and Placement

- Where to replicate and place the content?
- When and how to propagate updates?

7

Content Replicas



- Permanent replicas: Server replication, mirroring
- Server-initiated replicas: Push caching, CDNs
- Client-initiated replicas: Proxies, client caches

8

Permanent Replicas

- Initial set of replicas
- Server replication
  - Multiple servers hosting a datastore
  - Typically hosted on a LAN/cluster
  - Requests are distributed among the servers
- Mirroring
  - Popular Websites have multiple fixed servers across locations
  - Users select one of the mirror sites

9

Server-Initiated Replicas

- Servers pushing updates to replicas
  - Primarily for performance reasons
- Content-Distribution Networks (CDNs)
  - Geographically-distributed replicas
  - Each replica typically holds static content
  - Users are directed to closest replica
- Dynamic replication: Determine placement of content based on request traffic, clients locations. Factors:
  - Server load
  - Request distribution for files

10

Client-Initiated Replicas

- Client caches:
  - Local cache maintained on the client machine
- Proxy caches:
  - Cache shared by multiple browsers/users
  - Can be placed at multiple levels: On a LAN, department-wide, institute-wide
- Cooperative caches:
  - Each cache holds subset of Web pages
  - Share pages among each other
  - Can have hierarchy or overlay network of proxies

11

Content Distribution

- What happens when content changes?
- What is sent in the updates?
  - Notification vs. new data
- Who initiates the updates?
  - Push vs. pull

12

Update State

- What is sent in the updates?
- Invalidation:
  - Notification of change
  - Actual update will have to be done upon an operation
- Modified data:
  - Can also send the log of changes
- Active replication:
  - Send operations, replicas execute the operations

Update Dissemination

- Who initiates the updates?
- Push vs. pull protocols

Push-based Protocols

- Server sends updates to replicas whenever content changes
- Which replicas would this be suitable for?
- What kind of workload would this be useful for?

Push-based Protocols: Pros and Cons

- Pros:
  - Maintains high consistency
  - Replicas can be passive/dumb
  - Useful for time-sensitive data
- Cons:
  - Need to maintain a lot of state at the server
    - Susceptible to server crashes
  - Too much network bandwidth for updates

Pull-based Protocols

- Client cache pulls in updates from the server
- Can poll the server periodically
  - Get fresh data if changed, else serve from cache
- What should be the polling frequency?
  - Client-initiated: Poll server whenever a client makes a request
  - Time-To-Live: Content may have a TTL, after which replica needs to poll
  - Adaptive: Maintain a history of update times to estimate TTL for different data items

17

Pull-based Protocols: Pros and Cons

- Pros:
  - Server remains stateless
  - Resilient to both server and replica failures
  - Demand-driven
- Cons:
  - Weaker consistency guarantees
  - Active replicas required (not passive caches)
  - High overhead if high polling frequency
  - Response time higher on misses

18

Leases

- Hybrid approach between push and pull
- Lease: Limited time during which server pushes updates to replica
- Lease expiry: Replica has to pull updates or renew lease
- Generalization of push-pull approaches:
  - What is the lease duration for pure push and pure pull approaches?

19

Types of Leases

- Age-based
  - Based on object age
  - Larger the expected lifetime, longer the lease
  - Expected lifetime based on last modification time
- Renewal-frequency based
  - Based on client request frequency
  - More frequently a cache accesses a page, the longer its lease
- State-based
  - Based on server load
  - Shorter leases during heavy server load

20