Globule: A collaborative Content Delivery Network

Guillaume Pierre
Maarten van Steen

Presented By: Marwa K. Elteir

Outline

- Introduction
- Related work
- Collaborative Content Delivery Network
  - Requirements
  - Model
  - Content Distribution
  - Management
- Handling dynamic content
- Comments
Introduction

For a user or an organization to be able to publish a web site, they either do that through paying a monthly fee to a hosting server or deploying their own servers.

Problems appear when the web sites owners require some sort of guarantees about the availability and performance of their web sites.

One solution is to use a commercial Content Delivery Network (CDN) which provides resources to be allocated dynamically to serve certain web sites.

Another solution is to build a personal CDN composed of the resources available locally in the user's side.
Introduction

The work of this paper extends the idea of personal CDN into what is called Collaborative CDN (CCDN).

In CCDN, an overlay network of end-user machines cooperate through a wide area network in a peer-to-peer fashion.

They provide some of their resources and in return be able to replicate their web content across the CCDN which guarantee the required performance.

Related Work

Kazaa and BitTorrent
Coral- CoDeeN
- Replicas
- Reasonable performance
- No efficient replica placement scheme
- No efficient handle for dynamic content

ACDN
- Dynamic content
- Assume no data modification

DotSlash
- Dynamic content
- Assume the database is not bottleneck
CCDN Requirements

- Transparency to the clients.
- Automatic redirection to the target replica.
- Network proximity between clients and their replicas should be used as a factor to distribute the replicas.
- Consistency management.

CCDN Model

![CCDN Model Diagram](Image)

Figure 1: Globule Model
The locations of replicas are determined based on their proximity to the clients. They chose internode latency as their proximity measure. This measure is also used to redirect the client to the most appropriate replica.

Estimating the distance between every client machine to the replica servers is impossible. An alternative is to assume that the nodes are placed in M-dimensional geometry space. (M = 6) The coordinates of a node X is calculated based on the latency between this node and m landmark nodes.
Measuring proximity

90% of the latency estimation fall in the interval \([2/3 L, 3/2L]\), where \(L\) is the actual latency between two nodes.

Replica Placement

- The clients are localized. (Dynamic process?)
- The \(M\)-dimensional space is partitioned into cells of equal size.
- Each cell is ranked according to the number of clients located in this cell.
- Put the replica servers in the \(k\) highest ranked cells.
- The results of this algorithm is near optimal. In addition, its computational complexity is lower than the optimal alternative.
Client Redirection

- **HTTP based redirection**
  - Per-page basis decision
  - No transparency .. URL is returned

- **DNS based redirection**
  - Per-site basis decision
  - Transparent .. IP is returned

Another redirection policy can be used other than the estimated latency like for example the **average load** in each replica server.

### Content availability

- **Redirector availability**: DNS protocol allows multiple redundant servers to be registered for the same name.

- **Replica server availability**: Redirector server should periodically check the availability of replica servers.

- **Fresh copy availability**: Every replica server should be able to contact the origin server or the backup server to retrieve the up-to-date documents.
CCDN Management

Security
- Malicious content .. Sandboxing technique
- Malicious replica server .. Client solution or Origin server solution

Brokerage
- Allows any server to perform automatic resource discovery and allocation.
- Important especially during the registration process.

Supporting Web Applications (1/2)
Supporting Web Applications (2/2)

Comments

The main idea is based on deploying a peer-to-peer overlay network. It is adopted a lot as a solution to several problems in distributed systems arena.

- How they handle the dynamic nature of the web clients and replicas?
- How the assignments of the different servers to the participating nodes are achieved?
- Performance evaluation
Thank you