Angels In The Cloud

An On-Demand Peer-Assisted Content Distribution Cloud Service

Raymond Sweha remos@cs.bu.edu
Vatche Ishakian visahak@cs.bu.edu
Azer Bestavros best@cs.bu.edu

Abstract

This project develops a Cloud Service for Internet content distribution. Our system assists Seeders (content originators) with the dissemination of content (a file) to a set of nodes using Peer-to-Peer (P2P) concepts so that this dissemination is completed in the minimum time possible. Prior results of ours suggest that minimizing content distribution time may be achieved by adding nodes that are not themselves interested in downloading the content, but rather in assisting other nodes with their download in a prescribed (provably optimal) fashion. We call such nodes Angels. The emerging cloud computing architecture offers the best mechanism to allow such Angels to be created on-the-fly. As seeders request assistance with their file distribution, our service responds by spawning virtual machines that act as Angels. In this work, we describe the design and implementation of our "Angels-on-Demand" cloud service as well as the API for invoking this service.

Motivation

Barrier applications were all the peers are waiting for each other to download the same content

Example: gamers waiting for a terrain to download

Theorem[1]*:*

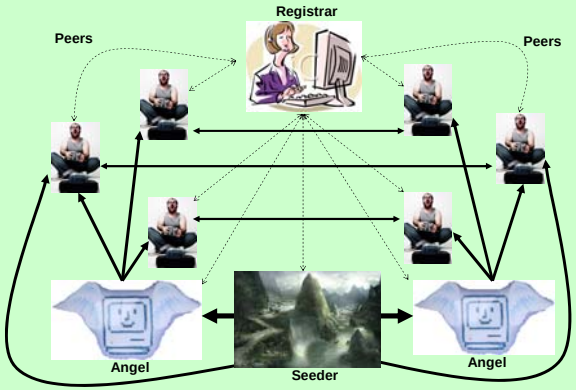
The Minimum Distribution Time of a file of F to a set of Peers L in the existence of a set of Angels H is:

$$T_{min} = \frac{F}{min\{b_{min}, a(S), \frac{a(S)+a(L)+a(H)}{|L|} - \frac{a(H)}{|L|^2}\}}$$

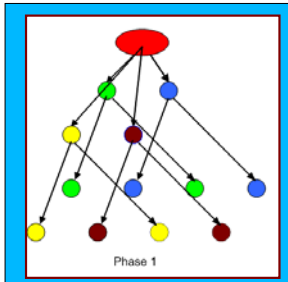
[1] Ray Sweha, Azer Bestavros, and John Byers. Angels: In-network support for minimum distribution time in p2p overlays. In *Master's Thesis, BU.*

Angels in The Cloud

System Architecture

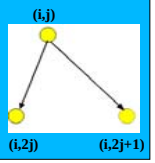


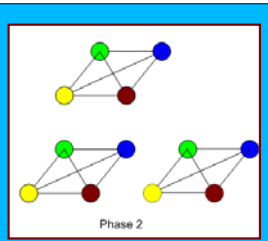
GroupTree Algorithm



Phase 1

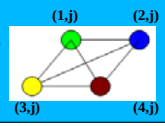
The file is divided into k segments
Each segment is decremented to a subset of nodes through a multicast tree



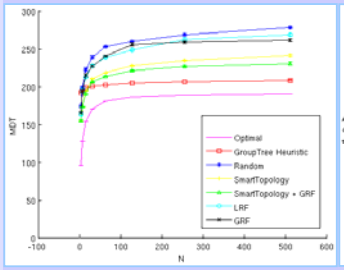


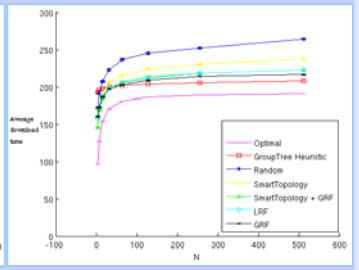
Phase 2

Each k nodes with different segments cooperate among themselves and disseminate the file



Simulation





Simulation Setup

- Discrete time simulator built in Java, providing controlled environment to reproduce experiments
- Comparing many heuristics
- Ability to run extensive experiments

BUT is simulation enough!?

System Building

- Modified Instrumented Bittorent Client as our client
  - Divide the file into segments, each segment treated as a s BT file
  - Each node gets an ID (i,j) denoting its position in the GroupTree
  - UnChoking clients depends on our algorithm
- Instantiate Angels from our Eucalyptus Cloud
- Run the Registrar on our BU infrastructure which acts as BT Tracker
- Run the clients on Emulab/planetlab nodes
  - Emulab provides semi-controlled environment with traffic shapers
  - Palnetlab provides real-world scenario verification

See Demo