Original Contribution:

Mostly the open source CDNs depend on the proximity (ONLY) aware model for serving the request from the Client.
In our system, we have developed a model which is using both the proximity and Load aware while serving the request from the Client.
We have seen in our presentation that in our approach(proximity and Load aware) the time taken by the Client to get the file was substantially reduced as compared to proximity (ONLY) aware model

Setup for various components:

AWS:

We have used three services from Amazon web services(AWS), namely, S3, EC2 and Route 53.
We have used S3 as a origin server and EC2 as CDNs and Route 53 for routing the client to specific proxy based on proximity of the client.
We had chosen two regions for testing, one North America and another as Asia Pacific. Within each region, we have chosen 5 CDNs.

Please make sure that all the instances are up and running, by choosing each instance and do "start" from the "action" drop down.
Also make sure that the files are present in S3 in any "bucket".

We have also used temporarily DNS name www.dsproject.info which uses geo-location feature of Route 53 for redirecting the client to specific proxy.

CDNs:

All the CDNs should be having below mentioned JAR files present:
a) aws-java-sdk-1.9.33.jar
b) com.fasterxml.jackson.core.jar
c) com.fasterxml.jackson.databind.jar
d) commons-logging-1.2.jar
e) httpclient-4.2.3.jar
f) httpcore-4.4.1.jar
g) jackson-annotations-2.1.2.jar
h) joda-time-2.4.jar

Please make sure that the RTTfile.txt should be present in all CDNs(which is been used to get RTT) and AWS credentials should be present at ~/.aws/credentials location.

Proxy:
Proxy should be having below mentioned JAR files present.
a) jsch-0.1.52.jar

Client:
We have used two types of client, one is Rouge Client which is spawning 1000 threads and requesting proxy for the file,
while other is normal client which is also requesting for the file.

N.B: Please make sure that all the necessary files should be present for each folder(CDNs, Proxy, Client).

Once you adhere all the steps mentioned above for AWS, CDNs, Proxy and Client, please run the following sequence of operation:

Step 1: At the Proxy end:
run "launcher.sh" script.

Step 2: At the CDNs' end:
run "run.sh" script.

Step 3: At the Client end:
a) For Rouge Client: please compile and run Client_rouge.java with below commands:
javac Client_rouge.java and java Client_rouge Cover.jpg(any file present in S3)
b) At normal Client please compile and run Client.java with below commands:
javac Client.java and java Client Cover.jpg(any file present in S3)