<Presentation>

1. Introduction (1 min)

* Idea of the project: Movie Map

1. Features

* Basic features (1 min)
  + Search: Type your favorate movie title
  + Find: Find an interesting movie from the movie map
  + Check: Check detailed information of the movie
* Other feature (In Demo)
  + Facebook login
  + Autocomplete
  + Trigger bing image search
  + Comment (NoSQL)

1. Database
   1. Data Cleaning? (3 min)
      1. Normalization?-> we splitted the given database to several sql tables
         1. Convert to MySQL
      2. Data Structure
         1. ER Diagram
         2. Schema (how to convert ER to Schema)
            1. keys
            2. contrains
   2. Optimization (4 min)
      1. Web
         1. Queries
         2. Indexing
      2. Score (2 min)
         1. Queries
         2. Optimization
2. Architecture & technologies (1 min)
   1. Architecture
   2. Technologies
      1. AWS Instances
         1. mysql, Ubuntu (host server)
      2. SW development
         1. Node.js, Bootstrap, Arbor.js, Facebook api, bing search
3. Demo (rest)
   1. Show all features
4. Questions

<Report>

1. Introduction and project goals

2. Basic architecture (not a dump of the classes) Hiro

3. Key technical challenges and how they were overcome Wenhe

a). Say details of how we get the score(s).

4. Performance evaluation

a). How we decrease the latency of each request.

5. Potential future extensions

(We should mention about DB, so ER diagram, queries, optimization)

Queries

<To search movie>

* Autocomplete for searching a movie title

SELECT title from Movie where title like "%'+req.query.key+'%"

req.query.key: keywords typed by a user

**OK?**

<To find movie>

* Search typed movie title

SELECT mid, date\_format(rdate, '+"'%m/%d/%Y'"+') as reldate from Movie where title="'+req.query.typeahead+'";

req.query.typeahead: Typed movie title

**OK?**

* Search a movie from the movieId (When the above query returned multiple titles)  
  (I found that this is redundant, since selecting mid using mid….)

SELECT mid from Movie where mid="'+temp\_Mid+'";

**OK? Since we need select specific movieID.**

* Explore similar movie titles from the typed movie title (expect 10 results)

SELECT CMid from Compare where Mid="'+rows[0].mid+'";

**OK? How to specify 10?**

* Explore similar movie titles from the mids obtained from the above query (so this query is executed 10 times) (limit to 2 tuples)

SELECT CMid FROM Compare WHERE Mid="'+ t +'" LIMIT 2;

**OK?**

* Search movie title from mid (Level0:1, Level1:10, Level2:20)

SELECT mid, title from Movie where Mid="'+ t +'";

**?**

<To show detailed info>

* Search title, rating, trailer, revenue, HP, budget, poster from mid

SELECT M.title, D.rating, D.trailer, D.revenue, D.HP, D.budget, D.poster

FROM Movie M INNER JOIN Details D ON M.Mid = D.Mid\

WHERE M.Mid="'+categoryKey+'";

**OK?**

* Search Studio name

SELECT S.name as studio

FROM Created\_by CB INNER JOIN Studio S ON CB.Sid = S.Sid\

WHERE CB.Mid="'+categoryKey+'";

**First select CB.Mid="'+categoryKey+'", then inner join CB and S?**

* Search director’s name, DoB, profile (image of the director)

SELECT DR.Name as dname, date\_format(DR.DoB, '+"'%m/%d/%Y'"+') as ddob,\

DR.Profile as dprofile, DR.Pid as Did\

FROM Directors DR INNER JOIN Person P ON DR.Pid = P.Pid\

WHERE DR.Mid="'+categoryKey+'";

**First select DR.Mid="'+categoryKey+'", then inner join DR and P?**

* Search Actors’ name, DoB, profile (limit to 14)

SELECT P.Name as actor, date\_format(P.DoB, '+"'%m/%d/%Y'"+') as bdate,\

P.Profile, A.Pid as Aid\

FROM Act A INNER JOIN Person P ON A.Pid = P.Pid\

WHERE A.Mid="'+categoryKey+'"\

ORDER BY A.OrderNum ASC\

LIMIT 14;

**First select A.Mid="'+categoryKey+'"\, then inner join A and P?**

<To obtain person’s detailed info>

* Search director’s info (Name, DoB, Profile, Mids, Tiltes directed by the director)

SELECT P.Name, date\_format(P.DoB, '+"'%m/%d/%Y'"+') as bdate, P.Profile,\

M.Mid, M.title\

FROM Person P INNER JOIN Directors D ON D.Pid = P.Pid\

INNER JOIN Movie M ON M.Mid=D.Mid\

WHERE P.Pid="'+categoryKey+'"\

ORDER BY M.popularity DESC;

**First select P.Pid="'+categoryKey+'"\, then inner join P and D, and then D and M?**

* Search actore’s info (Name, DoB, Profile, Mids, Tiltes, Charcter’s name of the actor)

SELECT P.Name, date\_format(P.DoB, '+"'%m/%d/%Y'"+') as bdate, P.Profile,\

M.Mid, M.title, A.CharacterName\

FROM Person P INNER JOIN Act A ON A.Pid = P.Pid\

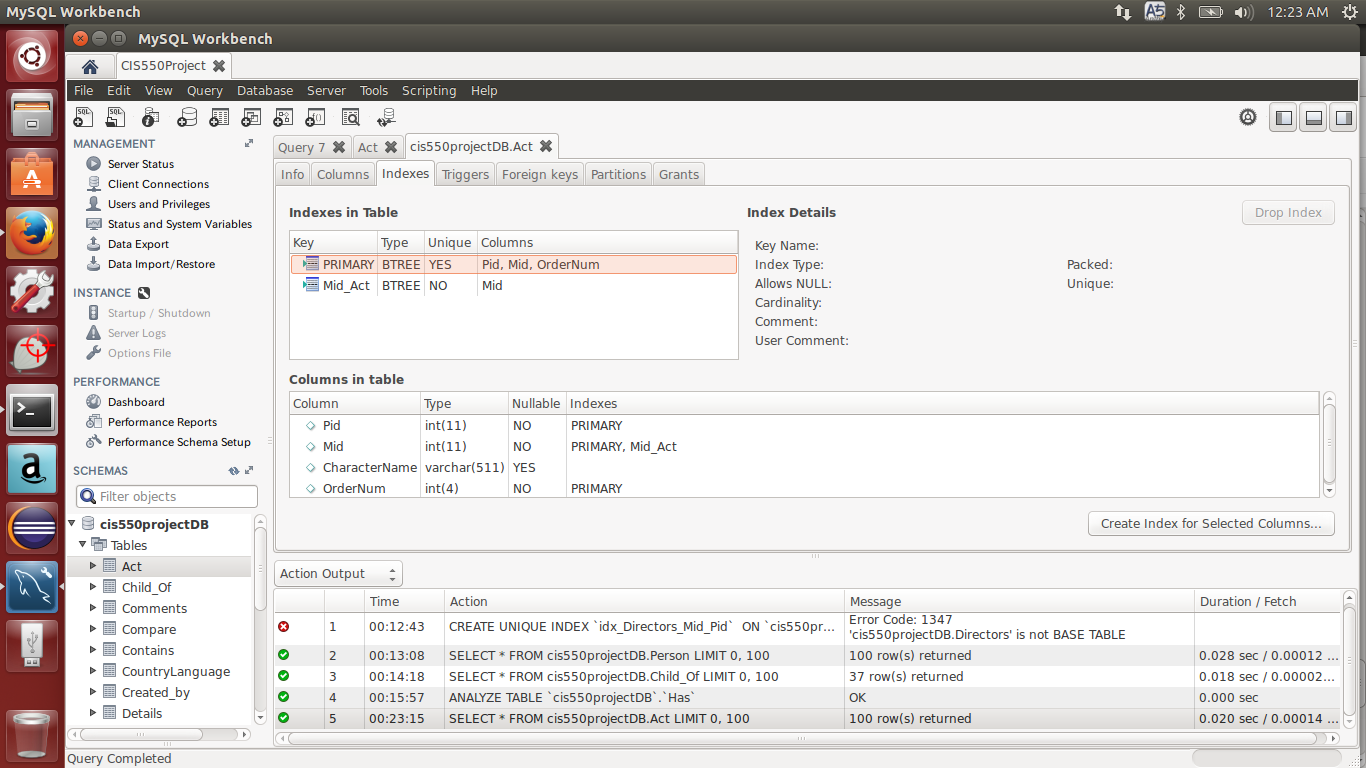
INNER JOIN Movie M ON M.Mid=A.Mid\

WHERE P.Pid="'+categoryKey+'"\

ORDER BY M.popularity DESC;

**First select P.Pid="'+categoryKey+'"\, then inner join P and A, and then A and M?**

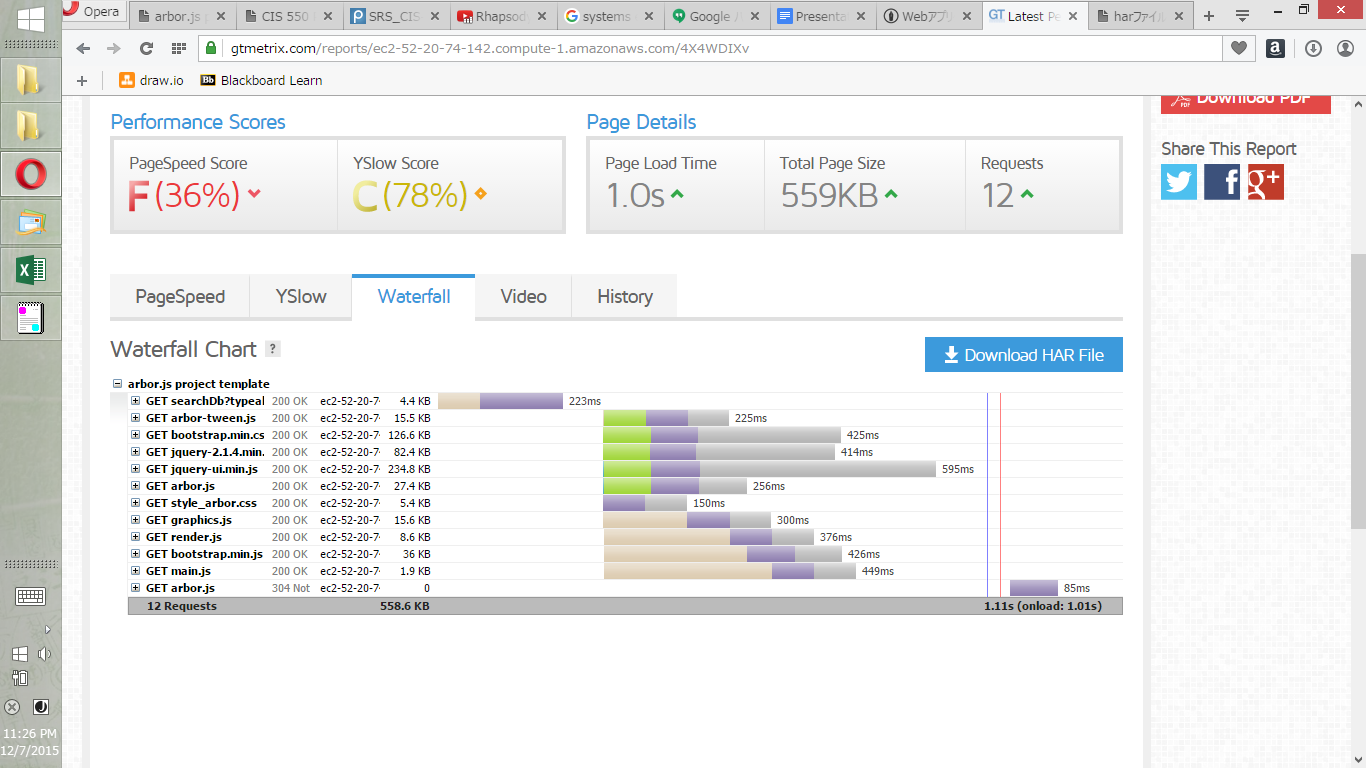
**Indexing**



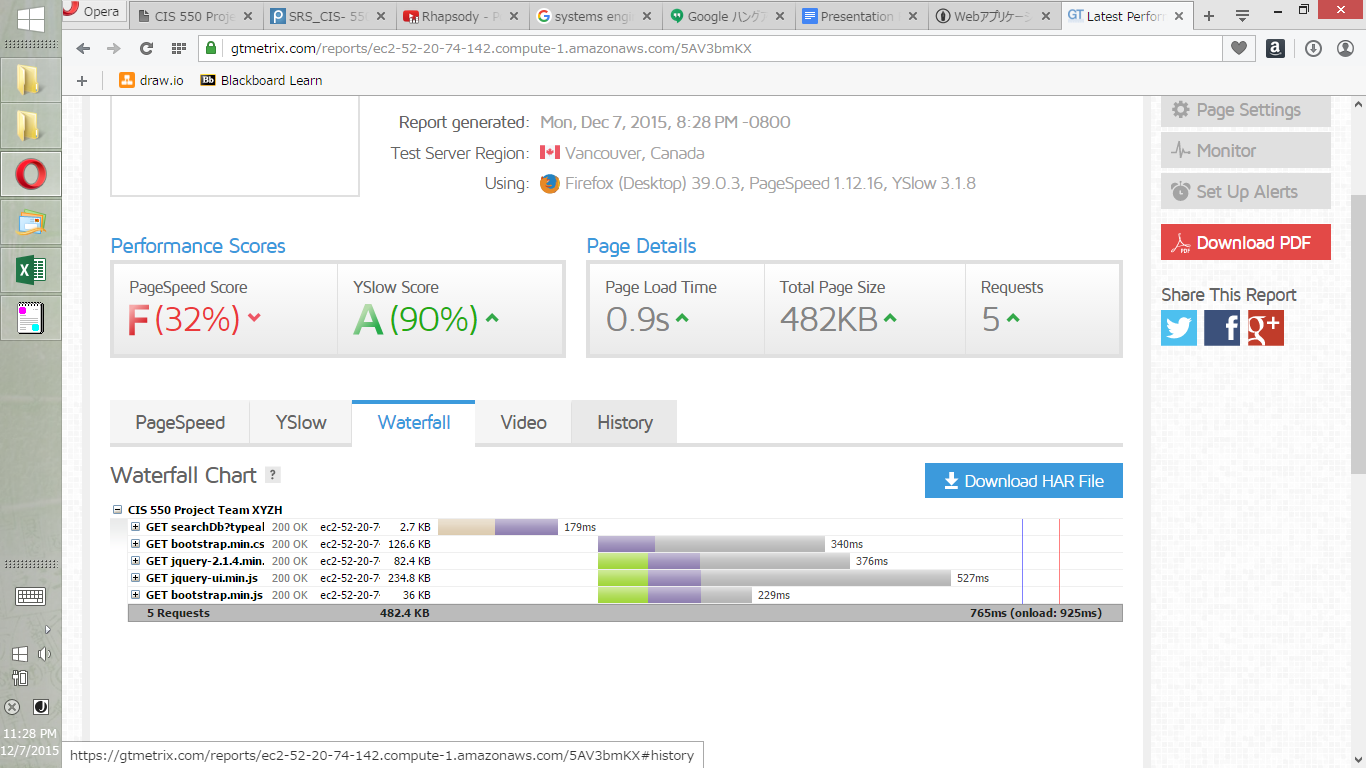
**Performance Test**

<Each Request>

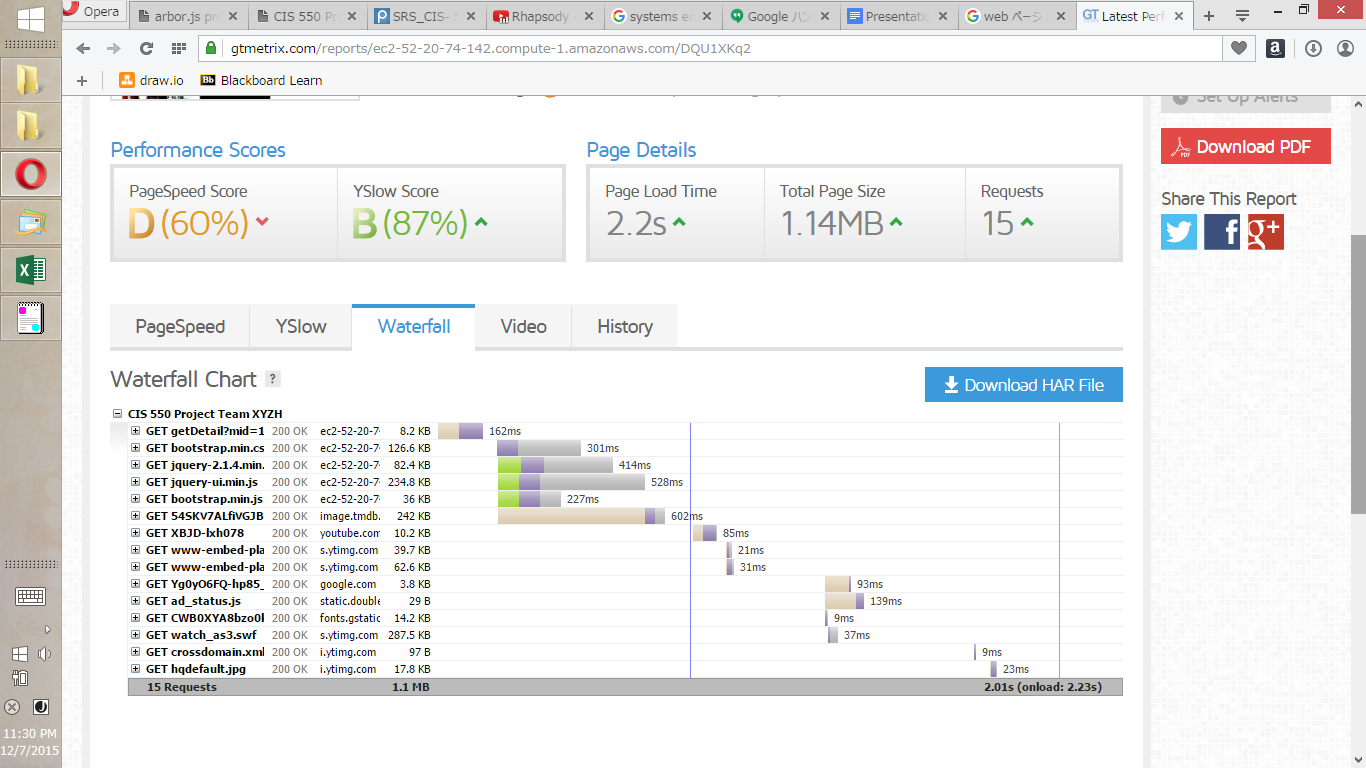
* Serach Movie (show result)



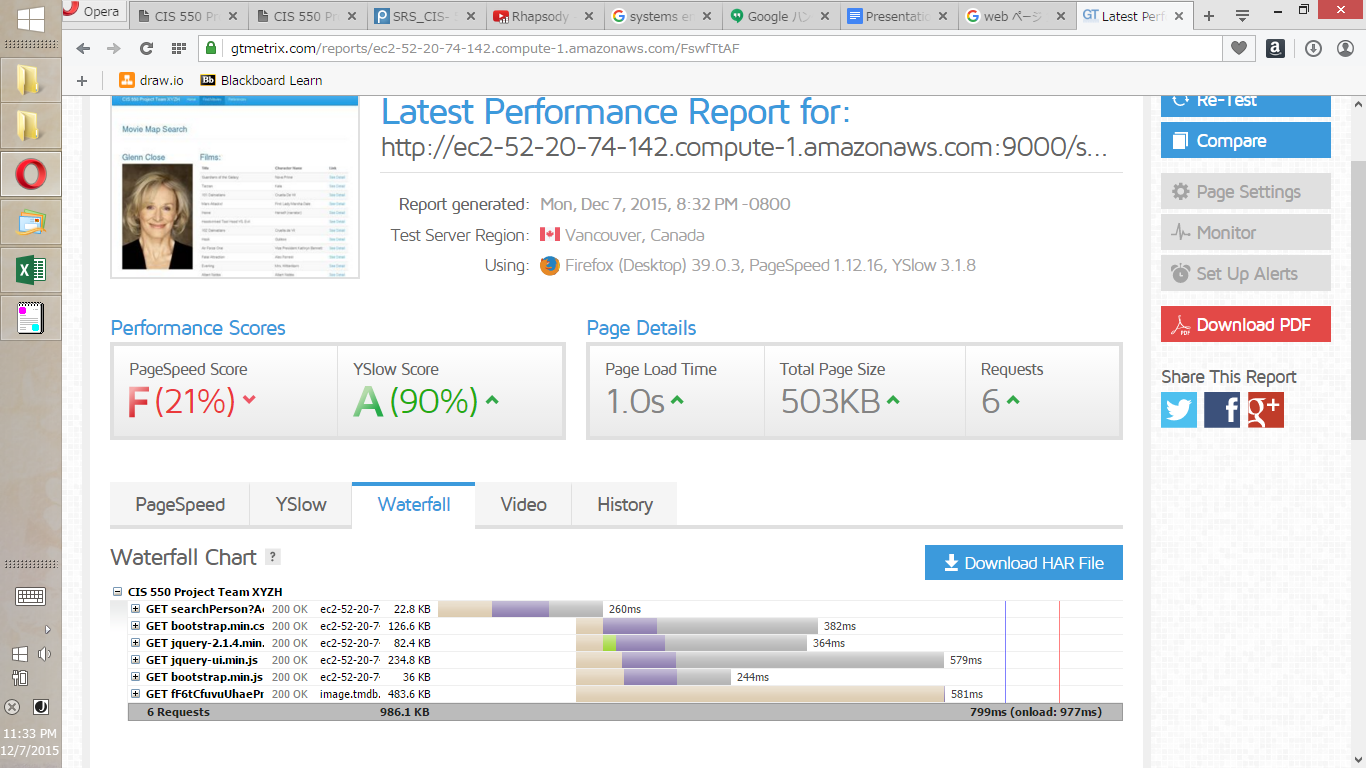
* Seach Movie (obtain multiple results)



* Find Movie



* Obtain person info



<Load Test>

* Search Movie (show result)

siege -d10 -c20 http://ec2-52-20-74-142.compute-1.amazonaws.com:9000/searchDb?typeahead=The+Day+After+Tomorrow

\*\* SIEGE 3.0.5

\*\* Preparing 20 concurrent users for battle.

The server is now under siege...

^C

Lifting the server siege... done.

Transactions: 5286 hits

Availability: 100.00 %

Elapsed time: 1354.52 secs

Data transferred: 19.74 MB

Response time: 0.15 secs

Transaction rate: 3.90 trans/sec

Throughput: 0.01 MB/sec

Concurrency: 0.59

Successful transactions: 5286

Failed transactions: 0

Longest transaction: 1.04

Shortest transaction: 0.11

* Search Movie (obtain multiple results)

siege -d10 -c20 http://ec2-52-20-74-142.compute-1.amazonaws.com:9000/searchDb?typeahead=101+Dalmatians

\*\* SIEGE 3.0.5

\*\* Preparing 20 concurrent users for battle.

The server is now under siege...^C

Lifting the server siege... done.

Transactions: 1657 hits

Availability: 100.00 %

Elapsed time: 415.21 secs

Data transferred: 4.36 MB

Response time: 0.07 secs

Transaction rate: 3.99 trans/sec

Throughput: 0.01 MB/sec

Concurrency: 0.29

Successful transactions: 1657

Failed transactions: 0

Longest transaction: 5.07

Shortest transaction: 0.05