**CS9223 Cloud Computing Final Project Proposal**

**Cloud Photo Album - “Traceroute”**

**Project Overview:**

Traceroute is resemble to Google Photo in a general way. However, it also provides a platform for users to upload their photos to a personal cloud storage space and generate a road map. This space is exclusive to user from user’s point of view, however, the storage is assigned to user in the form of S3 bucket which would provide necessary isolation and shared space at the same time. The project is designed to be a Web based project locally developed and deployed on AWS when released. NoSQL would be used for DAO operations which involves DynamoDB from AWS. Besides, several open source libraries would be used to facilitate AWS services.

**Technical Aspects:**

Back-end framework: MVC model using Struts2, Spring3 (pending)

Front-end framework: Bootstrap, JQuery, GoogleMap API

Cloud services: Elastic Beanstalk, DynamoDB, CloudFormation, CloudWatch, EC2, S3

Third-party library: dynamic-dynamodb[1], jsoda[2]

**Expectation Functionality:**

*Application functionality:*

1. Photo upload. User is able to upload their local photo on their laptops (may not support mobile device for now).

2. Create album that includes new upload photos or existing photos on the cloud.

3. Photo information display, update time, size

4. A road map would be generated based on the location information from photos

*System functionality:*

1. User sign up and sign in.

2. Hierarchical S3 bucket & file structure for each user.

3. DynamoDB using for NoSQL.

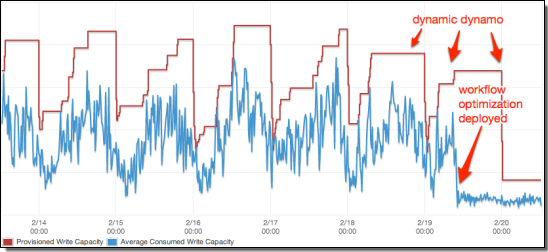
4. Dynamic-dynamodb for auto scaling.

**Cloud Service Necessity:**

1. Lower up-front investment for infrastructure and operation cost. We assume this project would be executed by a start-up company who doesn’t hope to invest up-front before anything promising delivers. So, PaaS and IaaS are perfect for such agenda. A light weight application can be instantly deployed on AWS without complicated configuration and maintenance.

2. S3 cloud storage service is another reason for using cloud service, as we don’t need to allocate and attend these user data locally which may incur information security problems. Also, we don’t need to worry about the damage of the user’s data.

3. Amazon DynamoDB is a fully-managed NoSQL database. When you create a DynamoDB table, you provision the desired amount of request capacity, taking in to account the amount of read and write traffic and the average size of each item. With Dynamic DynamoDB, an open source tool built by independent developer Sebastian Dahlgren. This flexible and highly configurable tool manages the process of scaling the provisioned throughput for DynamoDB tables. [3] We hope to analyze the actual request capacity by a figure similar to the following:



***Reference:***

[1] *dynamic-dynamodb, Github*, https://github.com/sebdah/dynamic-dynamodb/tree/master/docs

[2] *jsoda,Github,* https://github.com/williamw520/jsoda

[3] *Auto Scale DynamoDB With Dynamic DynamoDB*, https://aws.amazon.com/cn/blogs/aws/auto-scale-dynamodb-with-dynamic-dynamodb/