CLOUD COMPUTING AND M4D

Balwinder Sodhi
Indian Institute of Technology Ropar
Platform as a Service Cloud

A SERVICE MODEL BASED VARIANT
Platform as a Service (PaaS)

- NIST* definition: “... deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider”
- Consumer responsible only for writing application code
- Vendor gives sandboxed environment to develop/deploy applications
- Multiple consumers share the platform

*NIST = National Institute of Standards and Technology
PaaS Cloud Characteristics

- Allows only provider supported programming languages, tools, APIs and components for building applications
- No control of underlying infrastructure
  - Network, servers, operating systems, or storage
- Can only control deployed application and possibly its hosting environment configurations
- Effort needed to setup/management is lower than IaaS
  - But at the cost of flexibility
PaaS Architecture
Google App Engine (GAE)

- A leading PaaS cloud available to public
- Offers several services to developers
- Has faster ramp-up time to build applications
What Does It Offer

• Lets you run web applications on Google's infrastructure
  – No servers to maintain for you
  – You can focus on your application
• Supports writing apps in several programming languages
  – Java, Python, Go
• You only pay for what you use
  – No set-up costs and no recurring fees
  – Large free quotas for apps
Some GAE Features

• Serve data driven dynamic web apps
• Variety of data storage options
  – Allow queries and transactions
• Automatic scaling and load balancing
• Google Accounts APIs for authentication
• Local development environment
  – Simulates GAE locally on your machine
• Task queues and scheduled tasks
  – Perform work outside the scope of a web request
Application Hosting Environment

- Applications run in a secure sandbox environment
  - Limited access to the underlying operating system
  - Allows GAE to load balance requests for application across multiple servers, and automatically scale the servers
  - But some restrictions apply
Sandbox Restrictions

• An app can only access other computers on the Internet through the provided URL fetch and email services
• Other computers can only connect to the application via HTTP (S) requests on standard ports
• Cannot write to the file system
• Can read only files bundled with application code
• Application code only runs in response to a web request, a queued task, or a scheduled task
  – Must return response data within 60 seconds in any case
• Cannot spawn a sub-process or execute code after the response has been sent
Stateful Services Offered by GAE

• **Datastore**
  – Schemaless object datastore, with a query engine and atomic transactions
  – Java SDK includes the Java Data Objects (JDO) and Java Persistence API (JPA) interfaces, as well as a low-level datastore API

• **Blobstore**
  – Serve data objects, called blobs (binary large objects), that are much larger than the size allowed for objects in the Datastore service
  – Blobs are created by uploading a file through an HTTP request

• **Memcache**
  – Distributed in-memory data cache based on JCache
  – JCache provides a Map-like interface to cached data
Storing Data | Entities In GAE

• An entity has a **key** and some **properties**
  – **Record ≈ Entity ≈ Java/Python object**
  – **Column or Field ≈ Property**
    • Has a type
• Entities can be dynamically typed
  – Property types are recorded per Entity
• Key has either **id** or **name**
  – id is auto-assigned
  – Alternatively, the name is set by app
• Java Data Objects (JDO) or Java Persistence API (JPA)
• Quotas
  – Maximum entity size: 1 MB
  – Maximum size of a datastore API call request or response: 1MB
Stateless Service Offered By GAE

• Google Accounts
  – Can authenticate users with Google Accounts
  – Can detect whether the current user has signed in with a Google Account, and can redirect the user to sign-in/sign-up page

• URL Fetch
  – Can fetch resources and communicate with other hosts over the Internet using HTTP and HTTPS requests

• Images
  – Can resize, rotate, flip, and crop images
  – Can also enhance photographs using an predefined algorithm
Stateless Service Offered By GAE

• Mail
  – Apps can send and receive messages in the form of HTTP requests initiated by App Engine and posted to the app

• XMPP
  – Can send and receive instant messages to and from users of XMPP-compatible instant message services, including Google Talk

• Task Queues
  – Can perform background processing by inserting tasks (modeled as web hooks; i.e. a URL to a servlet) into a queue.
  – Automatically dispatch tasks for execution
GAE Architecture

GAE Java Application

Admin Console
URL Fetch
User Accounts
Memcache
Cron
Mail

Services

Google App Engine (for Java)

Java JPA/JDO
Data Store

MOOC on M4D 2013
Creating & Deploying GAE Apps

Eclipse plugin for GAE Java apps

Deploy apps from within Eclipse IDE
Manage Apps Via GAE Dashboard

- Lists GAE apps of the user
- Options for various management tasks
- Detailed app statistics
Fine-grained Data Store Stats

Display statistics for: Kind: All Entities

Statistics are updated at least once per day. Learn more

Last updated: 22:47:19 ago

<table>
<thead>
<tr>
<th>Kind</th>
<th>Entities</th>
<th>Built-in Indexes</th>
<th>Composite Indexes</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>9 KBytes</td>
<td>81 KBytes</td>
<td>0 Bytes</td>
<td>91 KBytes</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>493</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
Billing Information

Below is an event log of all billing-related events for this application.

<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
<th>Amount</th>
<th>Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012-10-26 14:53:17</td>
<td><a href="#">Usage Report for 2012-10-25</a></td>
<td>$0.00</td>
<td>$0.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource</th>
<th>Used</th>
<th>Free</th>
<th>Billable</th>
<th>Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frontend Instance Hours $0.08/ Hour</td>
<td>0.00</td>
<td>28.00</td>
<td>0.00</td>
<td>$0.00</td>
</tr>
<tr>
<td>Discounted Instance Hour $0.05/ Hour</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>$0.00</td>
</tr>
<tr>
<td>Backend Instance Hours $0.08/ Hour</td>
<td>0.00</td>
<td>9.00</td>
<td>0.00</td>
<td>$0.00</td>
</tr>
<tr>
<td>Datastore Storage $0.008/GByte-day</td>
<td>0.01</td>
<td>1.00</td>
<td>0.00</td>
<td>$0.00</td>
</tr>
<tr>
<td>Logs Storage $0.008/GByte-day</td>
<td>0.01</td>
<td>1.00</td>
<td>0.00</td>
<td>$0.00</td>
</tr>
<tr>
<td>Taskqueue Storage $0.008/GByte-day</td>
<td>0.00</td>
<td>0.49</td>
<td>0.00</td>
<td>$0.00</td>
</tr>
<tr>
<td>Blobstore Storage $0.0043/GByte-day</td>
<td>0.00</td>
<td>5.00</td>
<td>0.00</td>
<td>$0.00</td>
</tr>
</tbody>
</table>
Summary

• PaaS eases some of the tasks for developers
  – Don’t need to worry about underlying infrastructure
  – But also brings in some restrictions
• Provider gives commonly needed application services
  – Language runtimes, storage engines, identity management, async tasks etc.
• PaaS provider takes care of low level issues such as scalability, OS and other S/W patches and updates etc.
• Several players in the field
  – Google App Engine, Microsoft Azure, VMWare CloudFoundry etc.
THANK YOU