

Google App Engine Hochschule Mannheim

Peter Sutter

Fakultät für Informatik
Hochschule Mannheim
`peter.sutter@stud.hs-mannheim.de`

27.11.2009

Agenda

- Google App Engine
 - Was ist Google App Engine
 - Laufzeitumgebungen
 - Restriktionen
 - Tools
 - Konfiguration einer Anwendung
 - Versionierung
- Google App Engine Services
- Live-Demo

Was ist Google App Engine

Was ist Google App Engine

App Engine offers three distinct sets of cloud computing features to the end user:

- A PaaS that development teams [...] may use for building [...] applications that support transactions, uniform authentication, and robust scalability and availability
- Applications created with Google App Engine that are offered as SaaS, consumed directly from the end-users' web browsers
- The ability to integrate or consume third-party web services from other service platforms

Quelle: [Ciur09]

Findet jede Applikation Platz in Google's Cloud?

- Zielgruppe
 - Anwendungen mit kurzer Antwortzeit
 - Anwendungen, die große Berechnungen durchführen sind nicht geeignet

Laufzeitumgebungen

● Java

- App Engine nutzt JRE 6
- App Engine unterstützt Anwendungen die in Java 5 oder Java 6 compiliert wurden
- Java v1.4 wird nicht unterstützt
- Sprachen die auf der JVM basieren werden ebenfalls unterstützt ¹
 - Scala
 - Groovy
 - JRuby (Ruby)
 - Quercus (PHP)

● Python

- Python Laufzeitumgebung nutzt Python Release 2.5.2
- Python v3.x wird derzeit noch nicht unterstützt

¹Quelle: [PlayAppEng]

JAVA EE Unterstützung?

- Unterstützt:

- Persistenz: Java Persistence API 1.0 (JPA), Java Data Objects 2.3 (JDO)
- Java Server Faces (JSF), Java Server Pages (JSP)
- Java Servlet API 2.4

- Nicht unterstützt:

- Enterprise Java Beans (EJB)
- Java Database Connectivity (JDBC)
- Java Naming and Directory Interface (JNDI)
- Remote Method Invocation (RMI)

Restriktionen der Sandbox

- App Engine erlaubt nicht ...
 - ... das Starten von Threads
 - ... das Schreiben von Dateien auf das Filesystem (Lesen eigener Dateien ist erlaubt)
 - ... das Öffnen von Sockets
 - ... Requests die länger als 30 Sekunden laufen
 - ... Requests / Responses größer 10MB
- Es existiert eine Whitelist für alle erlaubten Klassen der JRE²
- Manche Module der Python Standard Library wurden deaktiviert, die gegen die Sandbox-Regeln verstoßen
- Third Party Librarys für Python dürfen kein C-Quellcode enthalten

²URL: <http://code.google.com/intl/de-DE/appengine/docs/java/jrewhitelist.html>

Tools

- Google App Engine SDK
 - Eclipse Plugin für Java
 - „Google App Engine Launcher“ für Python
 - Development-Server
- Admin-Console

Konfiguration einer Anwendung - Python

Python App Config - app.yaml

```
application: application-id
version: 1
runtime: python
api_version: 1
handlers:
- url: /*
  script: myApp.py
```

Konfiguration einer Anwendung - Java 1/2

Java App Config - WEB-INF/appengine-web.xml

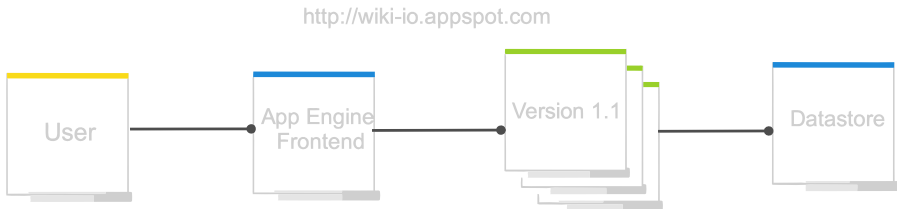
```
<?xml version="1.0" encoding="utf-8"?>
<appengine-web-app xmlns="http://appengine.google.com/ns/1.0">
  <application>application-id</application>
  <version>1</version>
</appengine-web-app>
```

Konfiguration einer Anwendung - Java 2/2

Deployment Descriptor - WEB-INF/web.xml

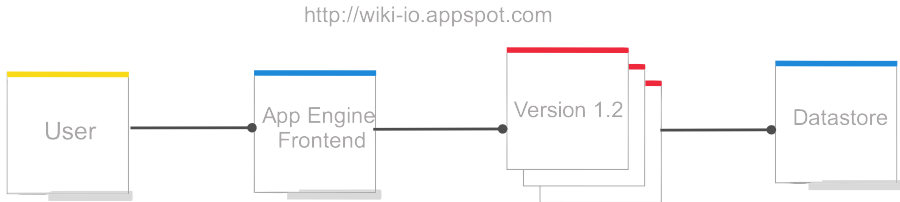
```
<web-app xmlns="http://java.sun.com/xml/ns/javaee" version="2.5">
  <servlet>
    <servlet-name>Visiter</servlet-name>
    <servlet-class>mysite.server.VisiterServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>Visiter</servlet-name>
    <url-pattern>/visiter</url-pattern>
  </servlet-mapping>
</web-app>
```

Versionierung 1/3



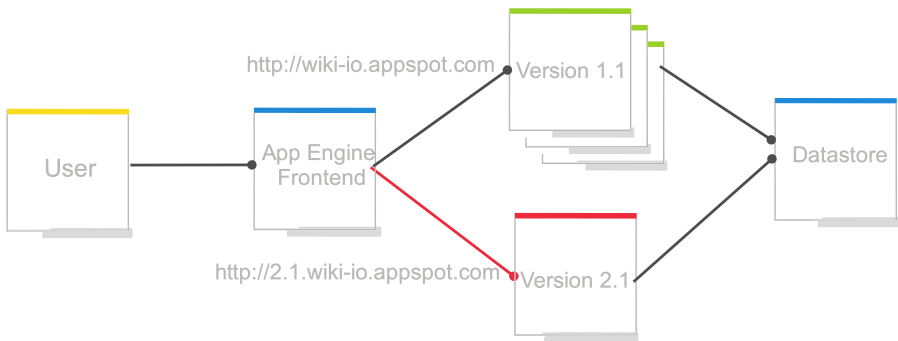
Quelle: [Vers]

Versionierung 2/3



Quelle: [Vers]

Versionierung 3/3



Quelle: [Vers]

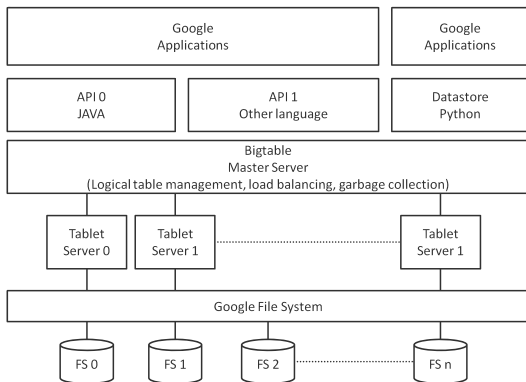
APIs

- Datastore
- Memcache
- Mail
- URL Fetch
- Google Accounts
- Task Queue
- Image

Datastore 1/2

- Persistenter Speicher
- Setzt auf BigTable auf
 - Schemalos
- Keine relationale Datenbank
- Vergleichbar mit einer verteilten sortieren geschachtelten HashMap
- Daten aller Applikationen in einer BigTable Tabelle gespeichert
 - Wird aufgeteilt in sog. Tablets und auf die Server verteilt
 - Ein Server ist für ca. 100 Tablets von einer Größe von ca. 200MB verantwortlich

Datastore 2/2

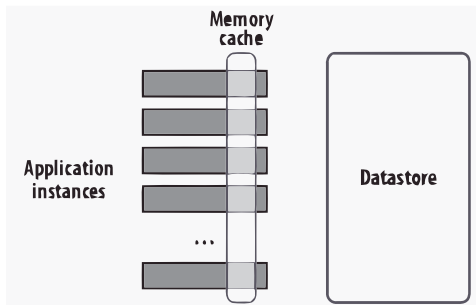


Quelle: [Ciur09]

- Auffinden eines Tablets mittels „spezieller“ Tabelle, welche den Speicherort referenziert ³

³Quelle: [BigTbl]

Memcache 1/2



Quelle: [Cha09]

- Verteilter temporärer Speicher
- Memcache wird von allen Instanzen geteilt
- Sehr schnell, da nicht von der Festplatte gelesen werden muss

Memcache 2/2

- Jeder Eintrag hat ...
 - ... einen eindeutigen Schlüssel
 - ... eine Verfallszeit (Sekunden bis Monate), kann aber auch aufgrund von Ressourcenmangel vorher gelöscht werden
- Maximal 1MB pro Item

Memcache Pattern (Python)

```
def get_data():
    data = memcache.get("key")
    if data is not None:
        return data
    else:
        data = self.query_for_data()
        memcache.add("key", data, 60)
        return data
```

Mail 1/2

- Zum Senden von E-Mails über das GMail Gateway
- Kein beliebiger Absender möglich (E-Mail-Adresse des Administrators oder des derzeit eingeloggtten Benutzers)
- Dateianhänge und mehrere Empfänger möglich

Java - Senden einer E-Mail

```
Session sess = Session.getDefaultInstance(new Properties(), null);
Message msg = new MimeMessage(sess);
msg.setFrom(new InternetAddress("loggedIn@user.com", "Username"));
msg.addRecipient(Message.RecipientType.TO, new InternetAddress(
    "recipient@example.com", "Mr. User"));
msg.setSubject("Titel");
msg.setText("Mein Text..");
Transport.send(msg);
```

Mail 2/2

- Mail API erlaubt ebenfalls das Empfangen von E-Mails (string@myappid.appspotmail.com)
- Empfangene E-Mails werden als POST Request an `/_ah/mail/<senderemailadress>` gesendet
- URL kann wie gewohnt im Deployment Deskriptor auf ein Servlet gemappt werden, welches die E-Mail entgegen nimmt

URL Fetch

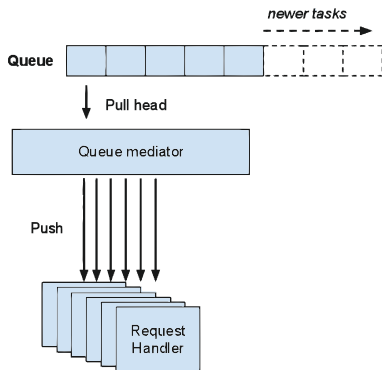
- Kommunikation mit anderen Webseiten
- Erlaubt das Senden von HTTP(S)-Requests
 - URL muss standard Ports für HTTP (80) bzw. HTTPS (443) verwenden
 - GET, POST, PUT, DELETE und HEADER Request Methoden werden unterstützt
- Vorsicht vor Endlos-Rekursionen bei Aufruf der eigenen URL
- HTTP Request kann Header und Body enthalten
- HTTP Response enthält Header, Body und Status-Code (z.B. 404 Not Found)
- Request und Response sind jeweils auf 1MB beschränkt

Google Accounts

- Mit diesem Service können sich Benutzer über ihren Google Account einloggen
- Man muss keine eigene Login-Logik implementieren
- Benutzer wird auf eine standard Login-Page von Google weitergeleitet (nicht individualisierbar)
- Login jedoch nur mit Google Account möglich
- Man kann erkennen ob der eingeloggte Benutzer ein Administrator ist

Task Queue 1/4

- Asynchrone Abarbeitung der Tasks
- Arbeitet nach dem FIFO Prinzip



Quelle: [OffPro]

Task Queue 2/4

- App Engine setzt den Task bei fehlerhafter Ausführung (Status-Code \neq 200 OK) zurück in die Queue
- Task Abarbeitungszeit beschränkt auf 30 Sekunden
- Maximal 100.000 Tasks / Tag

Task Queue 3/4

WEB-INF/queue.xml

```
<queue-entries>
  <queue>
    <name>visiter-queue</name>
    <rate>1000/d</rate>
    <bucket-size>10</bucket-size>
  </queue>
</queue-entries>
```

Starten der Tasks

```
Queue queue = QueueFactory.getQueue("visiter-queue");
TaskOptions to = TaskOptions.Builder.url("/visiterTaskProcessor");
to.param("dummyVisiter", "Worker");
to.method(Method.GET);
queue.add(to);
```

Task Queue 4/4

Log

```
„GET /visiterTaskProcessor?dummyVisiter=Worker HTTP/1.1“ 200 0  
„http://myapp.appspot.com/visiter?taskQueue=start+Visiter+Task+Queue“  
„AppEngine-Google; (+http://code.google.com/appengine)“
```

Image

- Serverseitige Bild-Manipulation
 - Drehen
 - Spiegeln
 - Ausschneiden
 - Größe ändern
 - Automatische „I'm Feeling Lucky“-Funktion zum Verbessern der Helligkeit, Kontrast und Farben

Java - Image

```
byte[] imgData; //...
Image oldImg = ImageServiceFactory.makeImage(imgData);
Transform enhance = ImageServiceFactory.makeImFeelingLucky();
Image newImage = imgService.applyTransform(enhance, oldImg);
imageData = newImage.getImageData();
```

Demo

- Erstellen einer Beispiel-Applikation
- Testen im Development-Server
- Anwendung auf Google App Engine deployen
- Anwendung im Browser ausführen
- Admin-Console

Vielen Dank für Ihre Aufmerksamkeit

Quellen

- Codesnippets von <http://code.google.com/intl/de-DE/appengine/docs/>
- [Ciur09] Eugene Ciurana. *Developing with Google App Engine*. Apress, 1. Auflage 2009. ISBN: 978-1-4302-1831-9
- [OffPro] <http://code.google.com/intl/de-DE/events/io/2009/sessions/OfflineProcessingAppEngine.html>
- [PlayAppEng] <http://groups.google.de/group/google-appengine-java/web/will-it-play-in-app-engine>
- [Cha09] Charles Severance. *Using Google App Engine*. O'Reilly Media, First Edition May 2009. ISBN: 978-0-596-80069-7
- [Vers] <http://sites.google.com/site/io/best-practices—building-a-production-quality-application-on-google-app-engine>
- [BigTbl] http://code.google.com/intl/de-DE/edu/parallel/index.html#_distrib_storage