

Google Cloud Platform



Переяславский А. В.
apereyaslavsky@naumen.ru

Google Cloud Platform

- Apps Engine
- Compute Engine
- Cloud Storage
- BigQuery
- Cloud SQL
- Дополнительные API
 - Prediction
 - Translate, Mail, etc.

GAE - мотивация

- Создавать масштабируемые веб-приложения тяжело. Особенно для небольших групп разработчиков.
 - Типичный LAMP-стек требует нетривиальной конфигурации и последующего администрирования.
 - LAMP-приложения требуют дополнительной инфраструктуры для решения таких проблем как балансировка, репликация данных и мониторинг.
 - Такие приложения тяжело масштабировать. Нередко приходится менять архитектуру приложения с ростом нагрузки.

Google App Engine



- Платформа отвечает за автоматическую масштабируемость и балансировку нагрузки
- Платформа предоставляет динамический веб-сервер с поддержкой многих распространенных веб-технологий
- Хранилище данных на основе Big Table
- Простая интеграция с аккаунтами Google через API.
- Python-only на старте в апреле 2008; Java – 2009, Go.

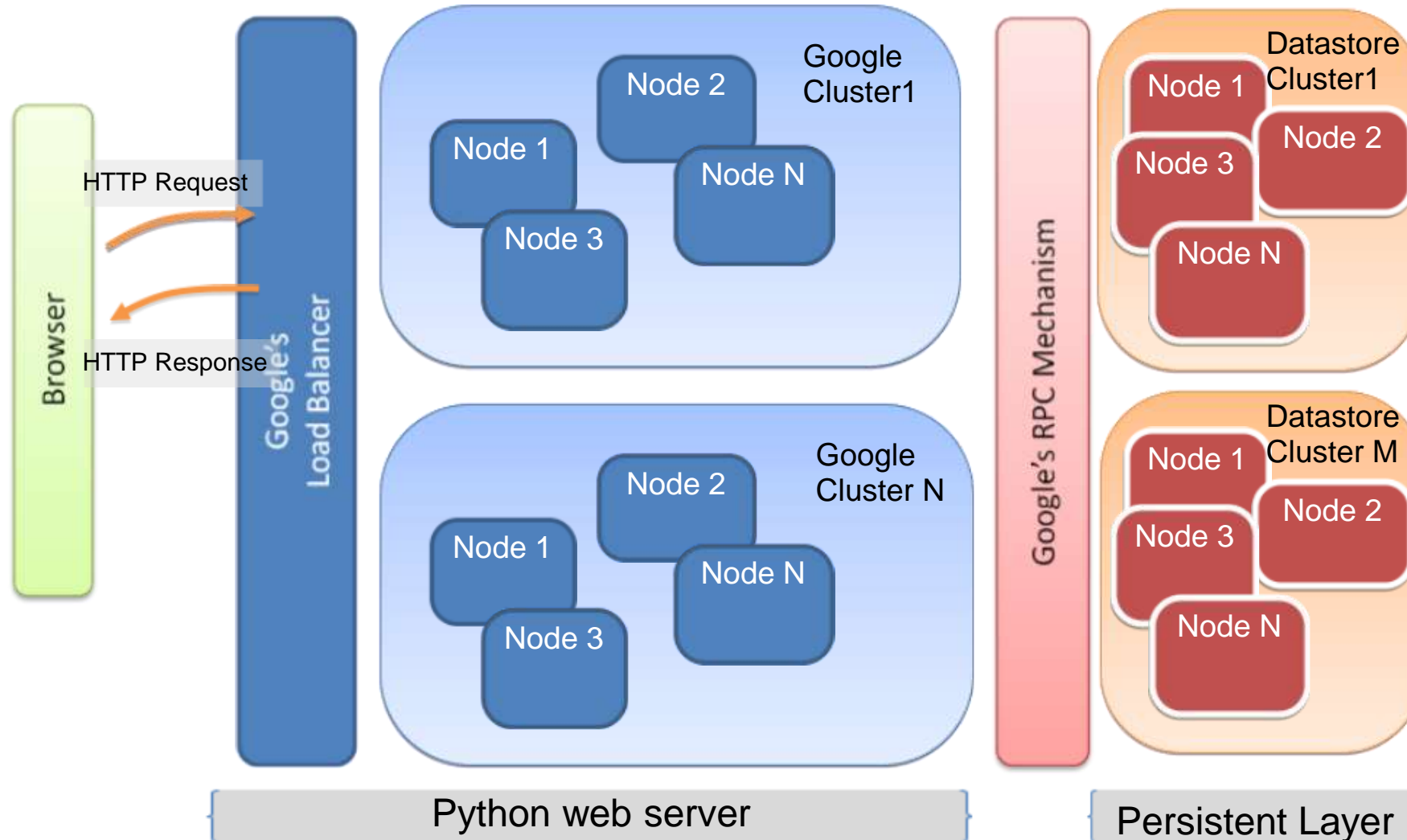
Google App Engine в числах

- 7.5 миллиардов хитов в день
- 1 миллион активных приложений
- 250,000 активных разработчиков
- Половина всех IP адресов интернета заходит на сервера Google App Engine за неделю
- 2 триллиона операций с хранилищами данных в месяц

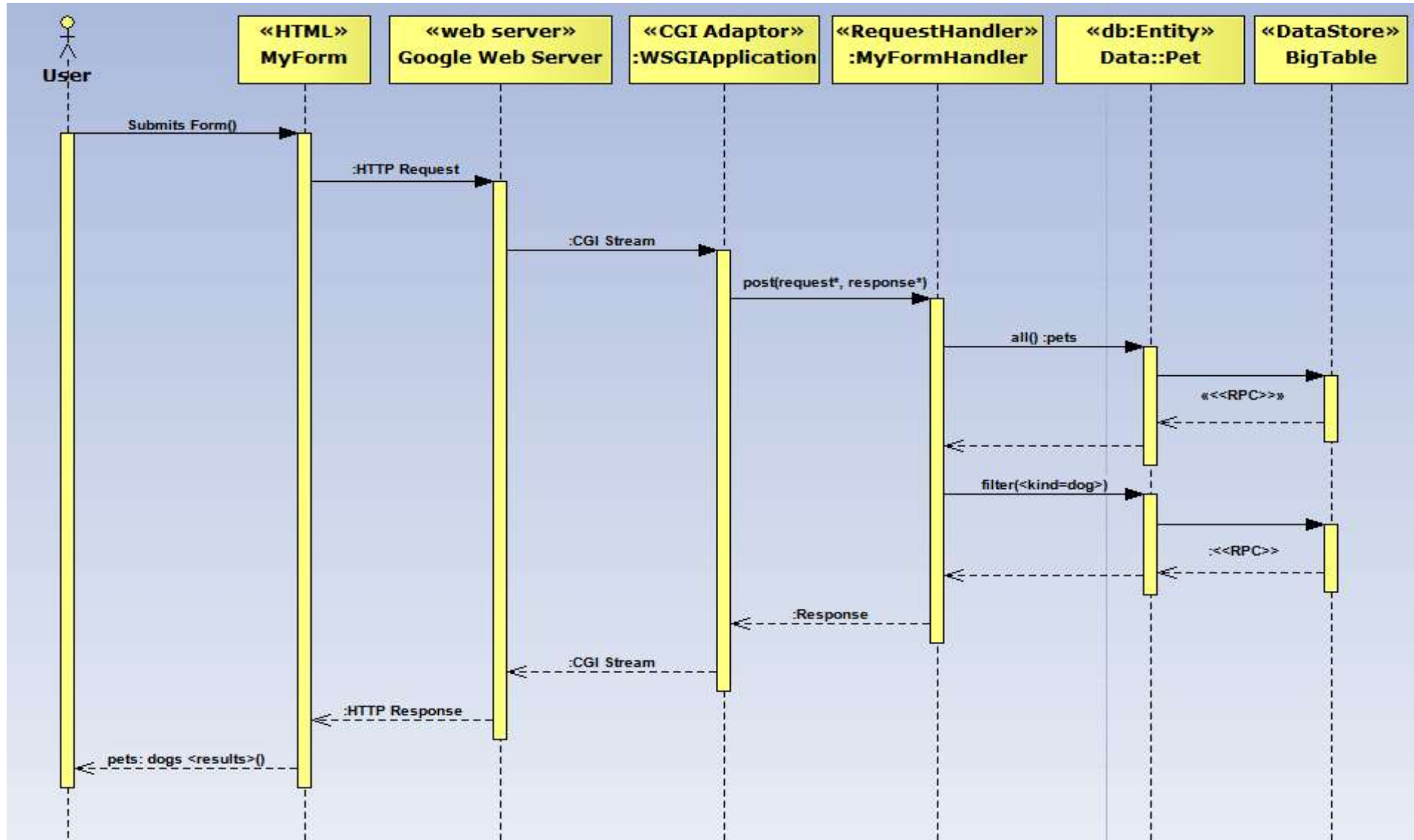
Ограничения

- ЯП, white-list разрешенных библиотек
- Нельзя создавать треды/подпроцессы, системные вызовы
- Key-value хранилище
- HTTP/S communication only
- Ограничение на количество элементов, которые можно достать за один запрос
- Лимит на время ответа сервиса, частоту выполнения задач, частоту запросов

Как работает App Engine



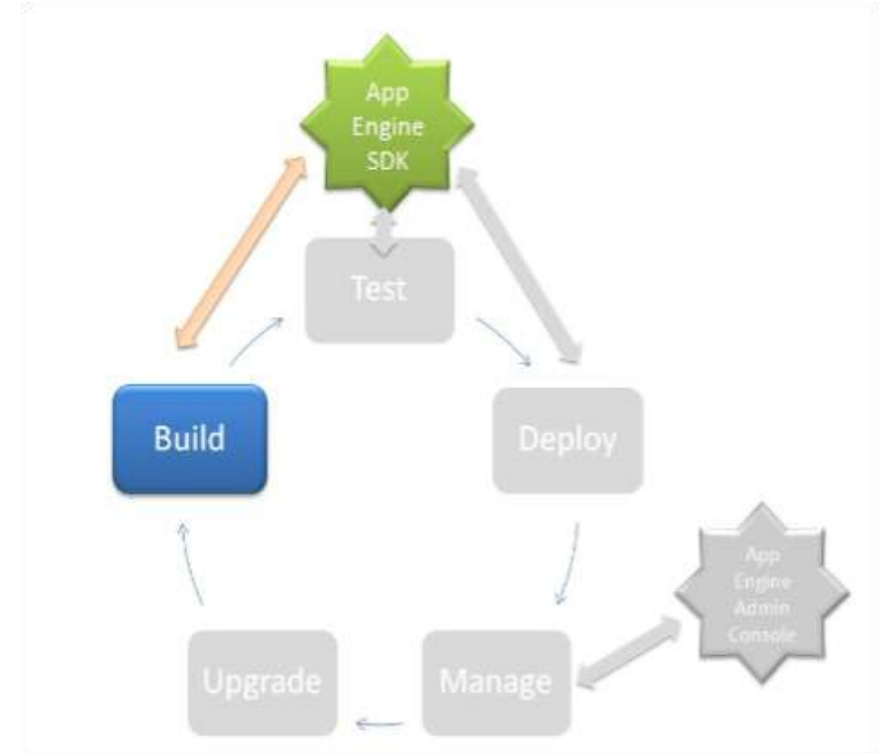
Обработка запроса к App Engine



Разработка приложения

App Engine SDK

- Веб-сервер
 - dev_appserver.py
- Утилита для загрузки приложений
 - appcfg.py
- Локальная реализация DataStore
- App Engine APIs

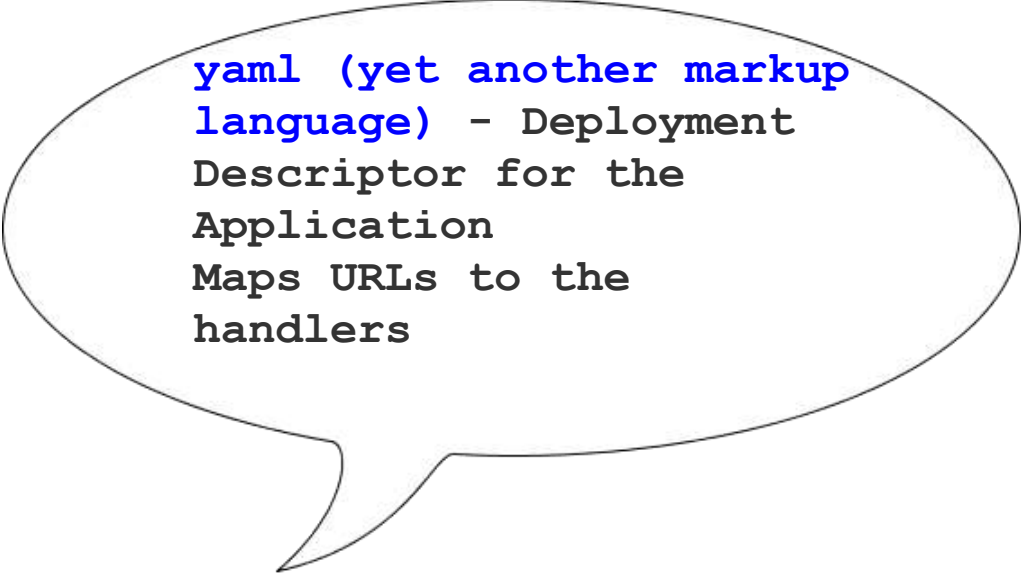


Разработка приложений с помощью
AppEngine

Конфигурация приложения - app.yaml

```
application: helloworld  
version: 1  
runtime: python  
api_version: 1
```

```
handlers:  
- url: /*  
  script: helloworld.py
```



yml (yet another markup language) - Deployment Descriptor for the Application
Maps URLs to the handlers

Request Handler - helloworld.py

```
from google.appengine.ext import webapp
from google.appengine.ext.webapp.util import run_wsgi_app

class MainPage(webapp.RequestHandler):
    def get(self):
        self.response.headers['Content-Type'] = 'text/plain'
        self.response.out.write('Hello, webapp World!')

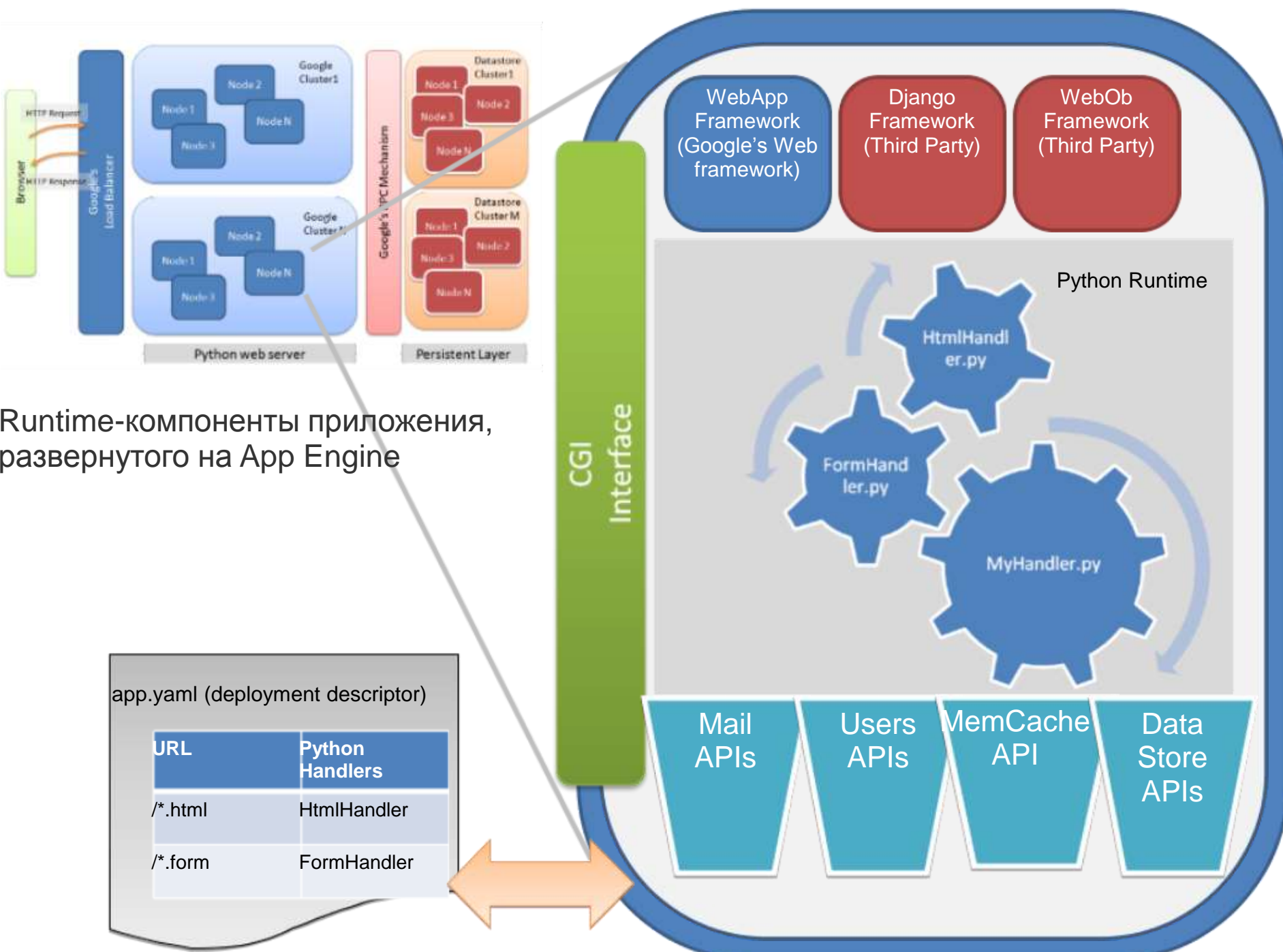
application = webapp.WSGIApplication([('/', MainPage)],
                                     debug=True)

def main():
    run_wsgi_app(application)

if __name__ == "__main__":
    main()
```

Extends webapp.RequestHandler

WSGI CGI Adapter




Runtime-компоненты приложения, развернутого на App Engine

Тестирование и развертывание сервиса на App Engine

- Локальное тестирование
 - `dev_appserver.py helloworld/`
- Развертывание
 - `appcfg.py update helloworld/`
- Приложение «в облаке»
 - `http://<application-id>.appspot.com`

Lazy install

- Устанавливаем Eclipse.
- Устанавливаем GAE SDK и плагин GAE для Eclipse
- Жмем на появившуюся кнопку  видим выпадающее меню, жмем Google app engine deploy tool

Google Plugin for Eclipse



The collage illustrates the process of using the Google Plugin for Eclipse. It includes:

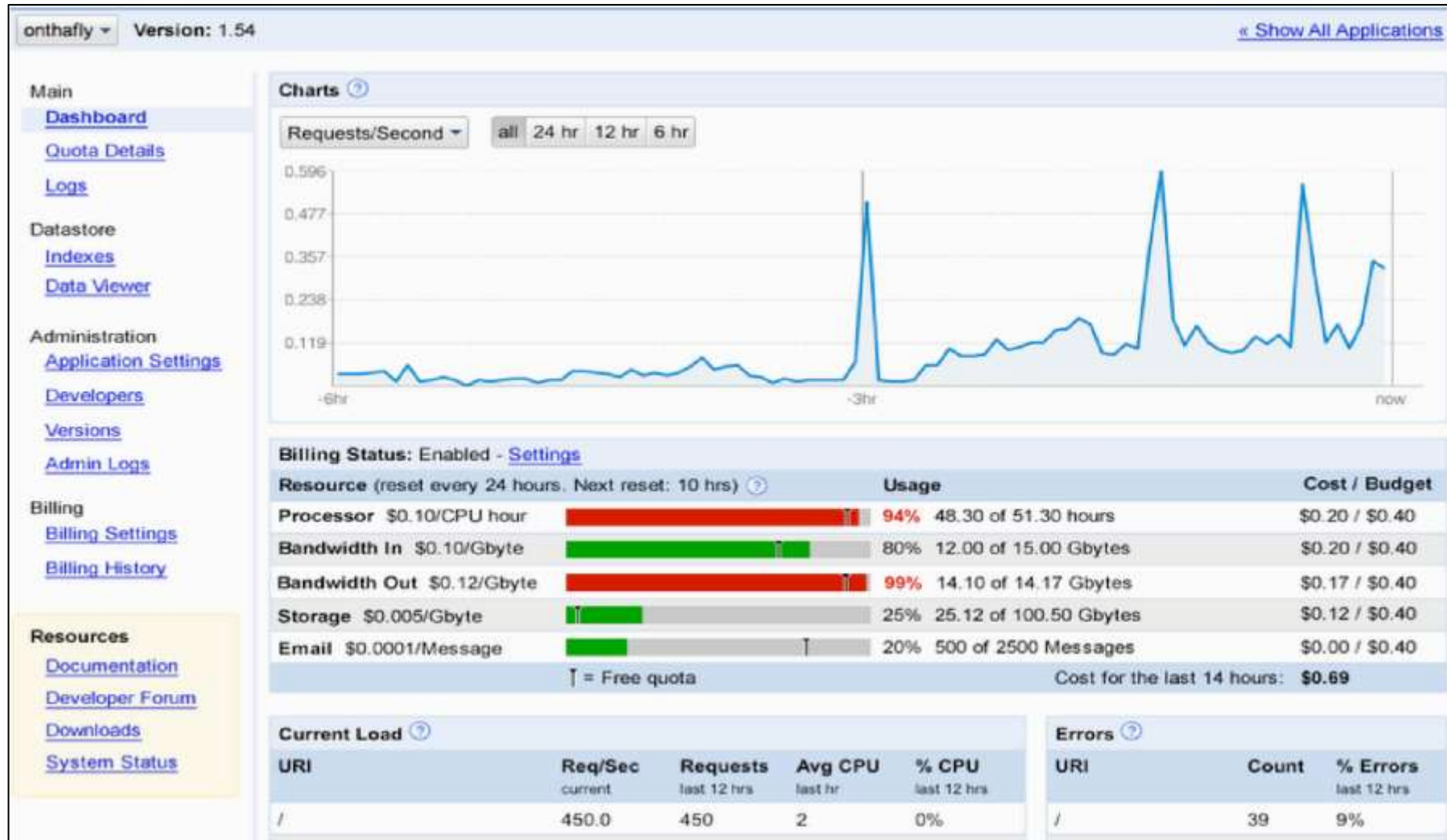
- A code editor window showing a native JavaScript method:

```
private static native void jsniMethod(boolean sayHi)/*-*/{  
    // Display a pop-up  
    if (sayHi) {  
        var name = this.@com.example.myapplication.MyApp:aname;  
        window.alert('Hello, ' + name);  
    }  
}/*-*/;
```
- A Package Explorer showing a project structure with a context menu open over the 'src' folder, with options: 'Run As', 'Run Configurations...', and 'Organize Favorites...'. The project contains files like 'MyTestProject.gwt', 'GreetingService.java', 'GreetingServiceAsync.java', and 'MyTestProject.java'.
- A 'New Web Application Project' dialog box with fields for 'Project name' (ExampleWebApp), 'Package' (com.example.myapplication), and 'Location'.
- A 'Run Configurations' dialog box for 'Run a Web Application', showing the 'MyApp' configuration with 'URL' set to 'MyApp.html' and 'Output style' set to 'OMVactivated'.
- A 'Software Updates and Add-ons' dialog box showing the 'Available Software' tab with 'Google Plugin for Eclipse 1.4' selected under the 'Google' category.

Lazy install

- Логинимся в GAE аккаунт
- Переходим в admin console и создаем application id:
<http://code.google.com/appengine/docs/adminconsole/>
- Важно : id глобально уникален!
- Меняем элемент application в appengine-web.xml в Eclipse на ранее созданный application id.
- Жмем «Deploy», логинимся в GAE аккаунт.
- Набираем в браузере
`http://<application_id>.appspot.com/<имя проекта>`

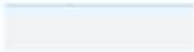


App Engine Health History



Development Tools for App Engine

Google App Engine [Help](#) | [Admin Console](#)

System Status

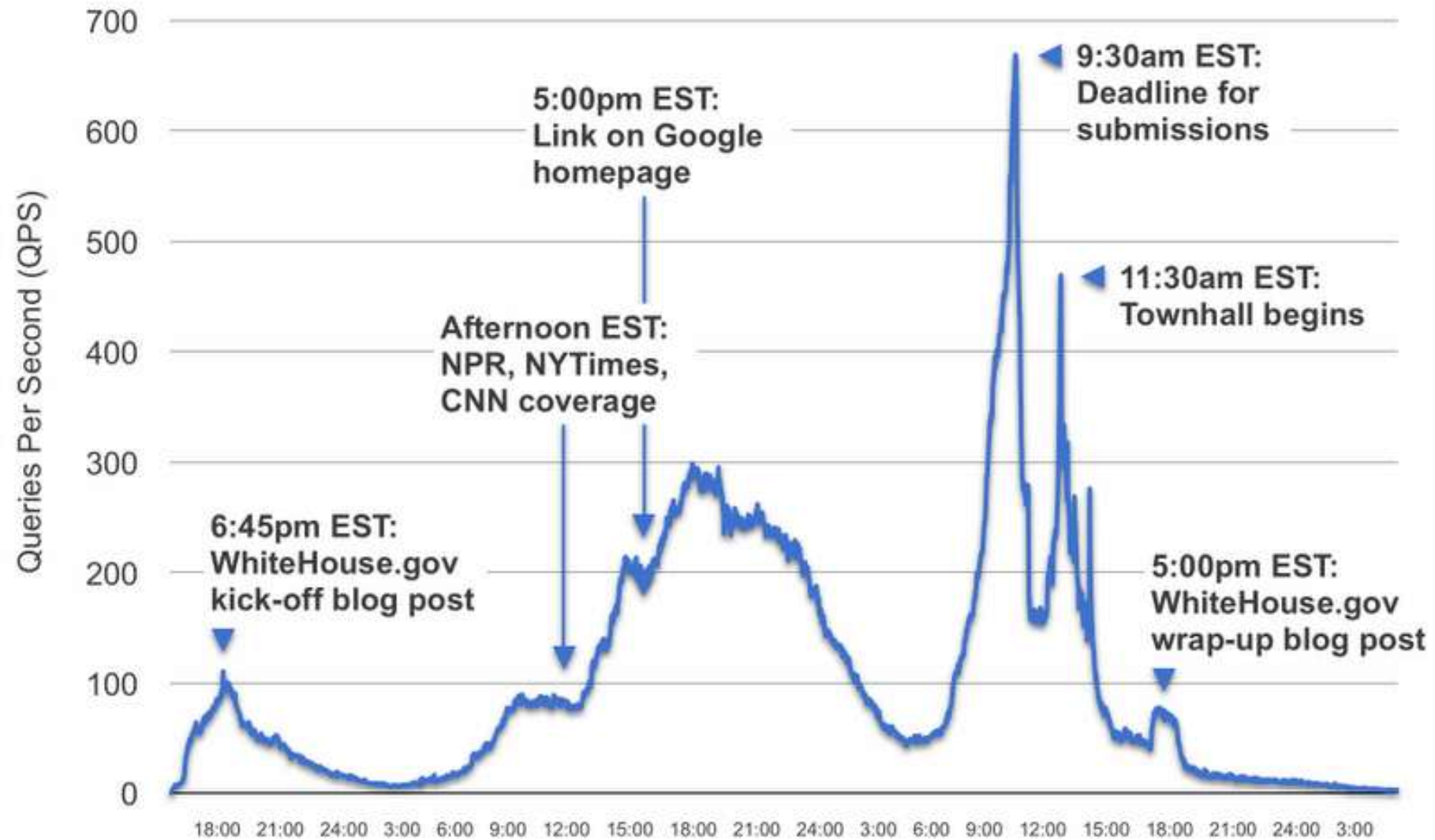
Current Availability: **100%** Uptime (last 7 days):  Read latency (today):  Write latency (today): 

	03/20/09	03/21/09	03/22/09	03/23/09	03/24/09	03/25/09	03/26/09	03/27/09	Now
Datastore	✓	✓	✓	✓	✓	✓	✓	✓	Normal
Images	✓	✓	✓	✓	✓	✓	✓	✓	Normal
Mail	✓	✓	✓	✓	✓	✓	✓	✓	Normal
Memcache	✓	✓	✓	✓	✓	✓	✓	✓	Normal
Serving	✓	✓	✓	✓	✓	✓	✓	✓	Normal
Urlfetch	✓	✓	✓	✓	✓	✓	✓	✓	Normal
Users	✓	✓	✓	✓	✓	✓	✓	✓	Normal

The following symbols signify the most severe issue (if any) encountered during that day. Click a symbol in the table above to view a day's performance graphs.

✓ No issues or minor performance issues ? Investigating ⚠ Service disruption ? Unknown

App Engine в действии : Whitehouse.gov!



Google Compute Engine



- Google Compute Engine предлагает масштабируемые и гибкие вычислительные возможности виртуальных машин в облаке.
- С Google Compute Engine, вы можете решить задачи обработки и анализа данных в сетевой инфраструктуре Google.



Google Compute Engine

- + : targets high performance/throughput clusters, advertises 50% more CPU per \$ compared to EC2
- - : limited preview, нет поддержки libcloud ☹, пока не во всех странах

How-to

- Логинимся, выбираем подписку, платим :)
- Выбираем проект
 - `gcutil getproject --project=<projectID>`
- Запускаем инстанс и устанавливаем софт
 - `gcutil addinstance help`
 - Пишем shell-скрипт с установкой нужных пакетов (`sudo apt-get install...`)
 - ```
for node in $nodes; do
 gcutil push $node install.sh .
 gcutil ssh $node "sudo /bin/bash ./install.sh >& install.log.$node" &
done
```
- Запускаем работы через gcutil
  - <http://www.stat.berkeley.edu/scf/scf-usingGCE.pdf>

# Google Storage



- Хранение данных в облаке
  - any format, any amount, any time
- Вы можете контролировать доступ к своим данным
  - private, shared, or public
- Доступ с помощью Google APIs, либо сторонние инструменты/библиотеки



# Достоинства Google Storage



Высокие показатели производительности и масштабирования

Безопасность и контроль доступа к данным



Легкость в использовании

# Варианты использования

- Хостинг статического контента
  - e.g. html, images, music, video
- Бэкапы
- Shared content (вместо Dropbox 😊)
- Хранилище данных для приложений
  - e.g. storage backend для Android, AppEngine, Cloud приложений
- Хранилище данных для вычислений
  - e.g. BigQuery, Prediction API

# Google Storage

- **RESTful API**
  - GET, PUT, POST, HEAD, DELETE
  - Ресурс: идентификация по URI
  - Реализовано как в Amazon S3
- **Объекты**
  - Любые, неограниченное количество
  - Размеры: 100 GB / объект
- **Buckets**
  - Обычные контейнеры, 1000 / аккаунт
- **Контроль доступа по google-аккаунтам**
- **Два варианта авторизации запроса**
  - Подпись запроса с помощью ключа доступа
  - Авторизация в браузере

# Google Storage

- Все данные реплицируются в несколько датацентров
- Для передачи данных между серверами датацентров используется высокоскоростное оборудование Google (Fiber?)
- Read-your-writes data consistency

# Инструменты

## GSUtil

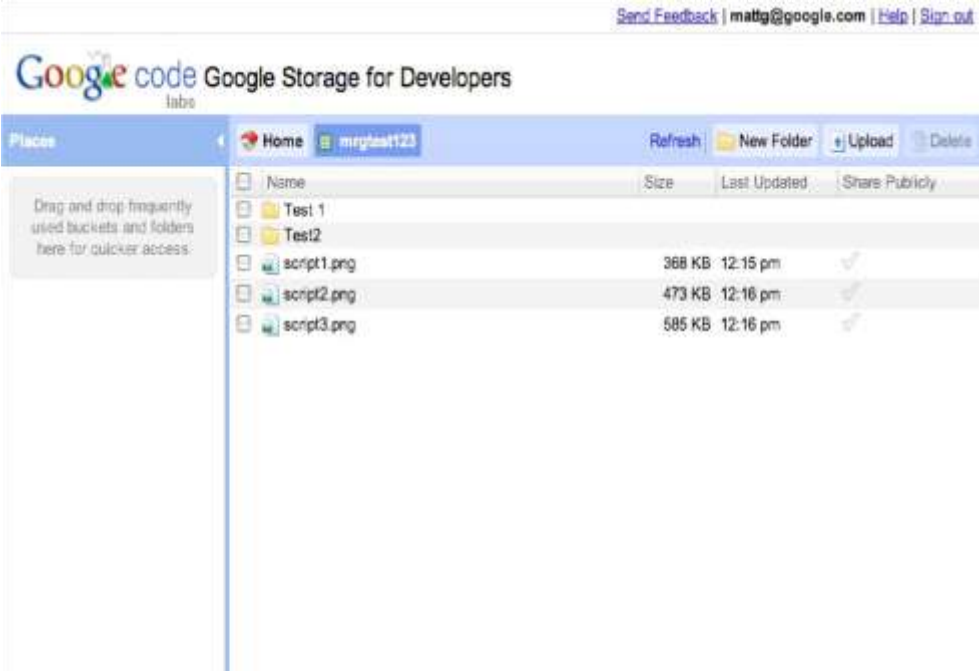
```
dhcp-172-19-3-109:~ wferrell$ gsutil
SYNOPSIS
gsutil [-d] [-h header]... command args

-d option shows HTTP protocol detail.

-h option allows you to specify additional HTTP headers, for example:
gsutil -h "Cache-Control:public,max-age=3600" -h "Content-Type:zip" cp * g
s://bucket

Commands:
Concatenate object content to stdout:
cat [-h] uri...
-h Prints short header for each object.
Copy objects:
cp [-a canned_acl] [-t] [-z ext1,ext2,...] src_uri dst_uri
- or -
cp [-a canned_acl] [-t] [-z extensions] uri... dst_uri
-a Sets named canned_acl when uploaded objects created (list below).
-t Sets MIME type based on file extension.
-z 'txt,html' Compresses file uploads with the given extensions.
Get ACL XML for a bucket or object (save and edit for "setacl" command):
```

## GS Manager



Send Feedback | mattg@google.com | Help | Sign out

Google code Google Storage for Developers  
labs

Places Home mytest123 Refresh New Folder Upload Delete

Drag and drop frequently used buckets and folders here for quicker access.

| Name        | Size   | Last Updated | Share Publicly |
|-------------|--------|--------------|----------------|
| Test1       |        |              |                |
| Test2       |        |              |                |
| script1.png | 368 KB | 12:15 pm     | ✓              |
| script2.png | 473 KB | 12:16 pm     | ✓              |
| script3.png | 585 KB | 12:16 pm     | ✓              |

# Google Storage - Цены

- Storage
  - \$0.17/GB/Month
- Network
  - Upload - \$0.10/GB
  - Download \$0.15/GB Americas / EMEA \$ \$0.30/GB APAC
- Requests
  - PUT, POST, LIST - \$0.01 / 1000 Requests
  - GET, HEAD - \$0.01 / 10000 Requests
- **Limited preview in US currently**
  - 100GB free storage and network from Google per account
- **Note: Non US preview available on case-by-case basis**

# Google BigQuery

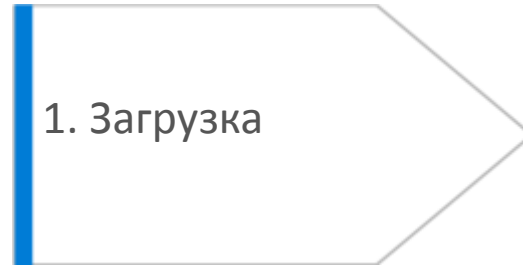
- Инструмент для интерактивного анализа больших датасетов из Google Storage
- Простой язык запросов GQL (SQL-like)
- Гибкий доступ по REST API, JSON-RPC, Google Apps Script

# Ключевые возможности BigQuery

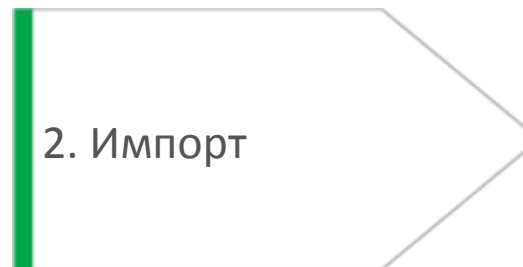
- Миллиарды записей
- Ответ – несколько секунд



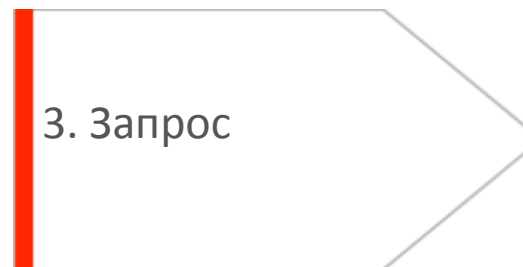
# Использование BigQuery



1. Загрузка



2. Импорт



3. Запрос

# BigQuery via REST

Таблица:

GET /bigquery/v1/tables/{table name}

Запрос:

GET /bigquery/v1/query?q={query}

JSON Reply:

```
{ "results": { "fields": [
 {"id": "COUNT(*)", "type": "uint64"}, ...]
}, "rows": [
 {"f": [{"v": "2949"}, ...]}, {"f": [{"v": "5387"}, ...]}, ...]
}
```

Поддерживает JSON-RPC

# Безопасность и privacy

- Стандартные варианты аутентификации Google
  - OAuth
- Поддержка HTTPS

# GQL

```
SELECT [* | __key__]
FROM <kind>
WHERE <condition> [AND <condition> ...]
ORDER BY <property> [ASC | DESC] [, <condition> [ASC | DESC]...]
LIMIT [<offset>,] <count>]
OFFSET <offset>
```

- DATETIME(*year, month, day, hour, minute, second*)
- DATE(*year, month, day*)
- TIME(*hour, minute, second*)
- KEY ('encoded key')
- USER ('email address')
- GEOPT ('lat', 'long')

# Пример анализа BigData

## Wikimedia Revision History

BigQuery snaidu@google.com [Help](#) [Report a problem](#) [Send feedback](#) [Sign out](#)

Enter SQL query

```
SELECT TOP(title, 5), COUNT(*)
FROM [wikipedia]
WHERE wp_namespace = 0;
```

Run

| TOP(title, 5)                               | COUNT(*) |
|---------------------------------------------|----------|
| George W. Bush                              | 43652    |
| List of World Wrestling Entertainment emplo | 30571    |
| Wikipedia                                   | 29722    |
| United States                               | 27432    |
| TOP(title, 5)                               | COUNT(*) |

# BigQuery Shell

```
Editor
title STRING NULL
id INT64 NULL
is_bot BOOL NULL
comment STRING NULL
num_characters INT32 NULL
is_minor BOOL NULL

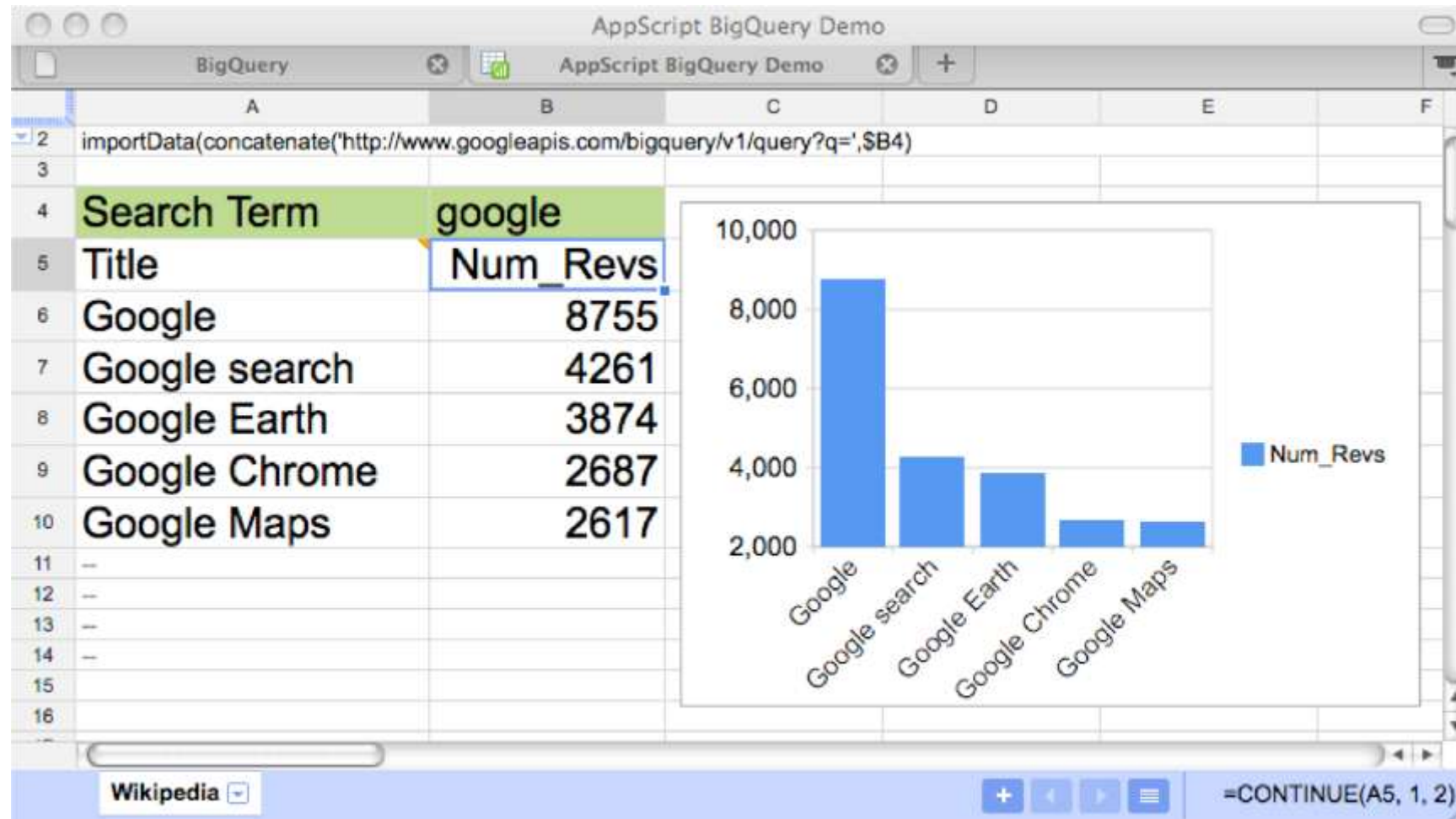
? SELECT TOP(title, 5), COUNT(*) FROM [bigquery.test.001/tables/wikipedia]
> WHERE wp_namespace = 0;
Execution time: 10.953 seconds
5 rows

TOP(title, 5) COUNT(*)

George W. Bush 43652
List of World Wrestling Entertainment employees 30572
Wikipedia 29726
United States 27433
Michael Jackson 23245

? |
```

# BigQuery из Spreadsheet



# Google Prediction API

- Технология machine learning от Google
- Доступен как on-demand RESTful HTTP веб-сервис





# Как оно работает?

- Prediction API находит некоторые особенности в данных во время обучения.
- Prediction API ищет по этим особенностям во время предсказания.

|           |                                                                    |
|-----------|--------------------------------------------------------------------|
| "english" | The quick brown fox jumped over the lazy dog.                      |
| "english" | To err is human, but to really foul things up you need a computer. |
| "spanish" | No hay mal que por bien no venga.                                  |
| "spanish" | La tercera es la vencida.                                          |

# Пример использования Prediction API

- Автоматически распознавать язык письма и отвечать на советующем языке
- Данные для обучения : много писем , язык которых уже известен.

# Использование Prediction API

1. Загрузка

2. Обучение

3. Предсказание

# Шаг 1

- Шаг 1: Загрузка
- Загрузите данные для обучения в Google Storage
  - Формат данных: CSV (e.g. english, Some text in english)
  - `gsutil cp ${data} gs://yourbucket/${data}`

# Шаг 2

- Шаг 2: Обучение
  - Создание модели обучения:
    - POST `prediction/v1.1/training?data=mybucket%2Fmydata`
  - Обучение асинхронно. Посмотреть статус:  
GET `prediction/v1.1/training/mybucket%2Fmydata`  

```
{"data":{
 "data":"mybucket/mydata",
 "modelinfo":"estimated accuracy: 0.xx"}}
```

# Шаг 3

- Шаг 3: Предсказание
  - Применить тренированную модель на новых данных  
POST prediction/v1.1/query/mybucket%2Fmydata/predict  
{ "data": {  
 "input": { "text" : [  
 "J'aime X! C'est le meilleur" ]}}}

---

```
{ data : {
 "kind" : "prediction#output",
 "outputLabel": "French",
 "outputMulti" : [
 {"label": "French", "score": x.xx} {"label": "English", "score": x.xx}
 {"label": "Spanish", "score": x.xx}]]}
```

# Возможности Prediction API

- Данные
  - Входные данные: числа, неструктурированный текст
  - Выходные данные: данные по дискретным категориям
- Обучение
  - Используется множество алгоритмов обучения
  - Выбираются автоматически
  - Асинхронный доступ
    - Приложения Google App Engine
    - Apps Script (e.g. из Google Spreadsheet)
    - Desktopные приложения

# Q&A

