# Developing Applications using Google App Engine & the Globe Labs APIs

Greg Igaya
Head, Future Applications



Globe

# GOOGLE APP ENGINE INTRO



Globe

# What is it?

Way for you to host your application on Google's infrastructure

**Free Scalability**

# What is it?

A standardized environment in which to code (internet) based applications

**Java & Python**

# What is it?

| |
|---|
| Local development environment (SDK) |
| Deployment mechanism |
| WSGI + WebOb + Django templating system |
| Data storage |
| Other neat stuff: scheduled tasks, XMPP messaging, email, authentication, image manipulation, memcache |



Globe

# A note on data storage

- NOT A RELATIONAL DATABASE
- “ReferenceProperty” type for FKs
- Up to 500 MB of storage for free
- Querying through objects & GQL
  - Up to 1000 records at a time
  - No JOIN
  - Comparison operators are limited only to single column in GQL



Globe

# PYTHON OVERVIEW

# Python Overview

- Multi-purpose high level language
  - compile on the fly
  - Object oriented – everything is an object
  - Strongly typed but dynamic
  - **WHITESPACE IS IMPORTANT**
  - Common operations / expressions
    - Pretty standard operators (+,-,=, <, etc) plus textual logical (and/or/not), membership (in) & identity (is)

# Python Overview

- Common operations / expressions
  - Collections
    - Lists – can use splicing to extract (ex. list[0:4])
    - Dictionaries – reference item by key
    - Tuples – immutable lists
  - Functions
    - Definition uses “def” keyword

```
def doSomething(arg1, arg2):
```

    - Calling a function

```
doSomething(var1, “string”)
doSomething(arg2=var1, arg1=“string”)
```

# Python Overview

- Common operations / expressions
  - Classes
    - Definition

```
class <name> (superclass, …):
  member = value
  def method (self, …):
    self.member = value
```

    - Constructor

```
def __init__(self, value):
```



Globe

# Python Overview

- Common operations / expressions
  - Classes
    - Usage

```
car = Car()              // instantiate
car = Car("Toyota")      // with constructor argument
car.year = "2001"   // access to member
car.sell()               // run class method
```

  - Modules

```
import modulename   // where modulename is just a
                    //  py file

from modulename import classdef
```

# GLOBE LABS' APIS



Globe

# OVERVIEW AND REGISTRATION

# Getting Started

- Working knowledge of HTTP, XML REST and/or SOAP in the language of your choice [Python/Java]
- A publicly accessible web server for processing incoming HTTP requests [GAE]
- A provisioned account to access to the API [http://www.globelabs.com.ph]



Globe

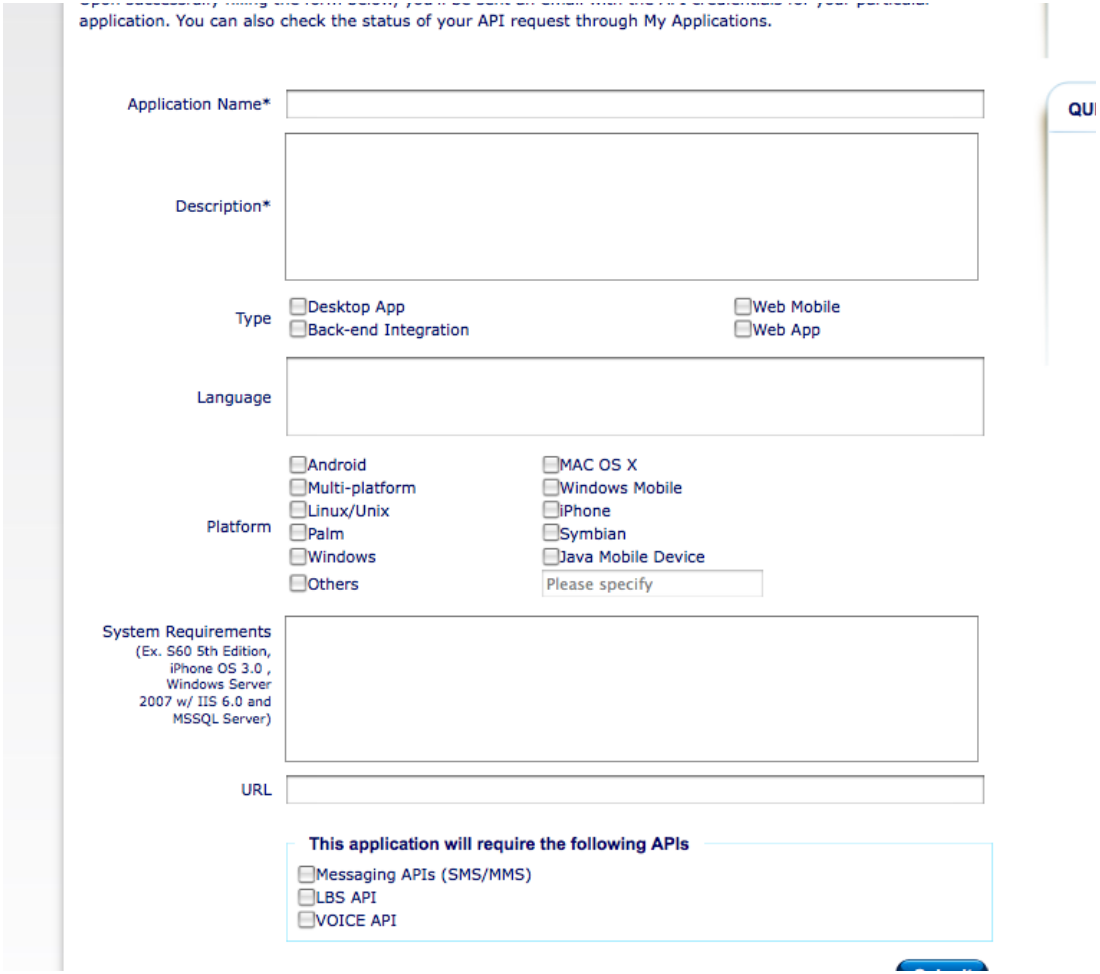
# Getting Started

- Both SOAP and REST interfaces are supported
  - For SOAP, get the appropriate library for your platform and utilize the WSDL to generate the proxy class
    - WSDL: http://iplaypen.globelabs.com.ph:1881/axis2/services/Platform?wsdl
  - For REST calls, simply make an HTTP call to the appropriate URL along with the paramters
    - http://iplaypen.globelabs.com.ph:1881/axis2/services/Platform/methodCall



Globe

# Provisioning: Go to the Portal & Request for API Access






Globe

# Provisioning: Receive email to get authentication details & access number

Hi globedemo1

You are receiving this email because you requested access to the Globe Labs APIs. If you did not make this request, contact us at globelabs@globetel.com.ph with "API Request Error" as the subject line.

Please take note of the following information:

WSDL: http://iplaypen.globelabs.com.ph:1881/axis2/services/Platform?wsdl
URL endpoint: http://iplaypen.globelabs.com.ph:1881/axis2/services/Platform
Access Number or Short-code: 2373
SMS and MMS Suffix (for receiving messages): 0303
Username (uName): [redacted]
Password (uPin): [redacted]
URL CALLBACK: http://www.dummy.com

Before using the API, make sure to define your URL call back and registered Globe numbers. To do this, go to http://202.126.34.119:1888/login.aspx, then login using the username(uName) and password(uPin) contained in this email. You'll be presented with a webtool that you can use to define / edit your callback URL and registered Globe numbers.

Remember, check the Globe Labs website for updates.

Thank you.

Globe Labs



Globe

# Provisioning: Define your URL callback & your test numbers

## SMS/MMS API

Please enter the following information below for us to complete your request.

Application URL:

This is the URL our system will access in order to send your application messages.

http://freeformsoftware.org/globe/lbs.php

(If you're not sure yet what your application URL will be, you may fill up this field later).

Allowed Mobile Numbers: These are the mobile numbers your application is allowed to send messages to and receive messages from.

(You may add numbers at a future time but cannot remove or edit previously entered numbers)

Submit

| Mobile No(s): | | | |
|---|---|---|---|
| | 09175889615 | | |
| | 09178306730 | | |
| | | | |
| | | | |
| | | | |

**Total:** 2

| Page 1 of 1 |

**SMS:** 3 / 5 available today
**MMS:** 5 / 5 available today
**LBS:** 0 / 0 available today



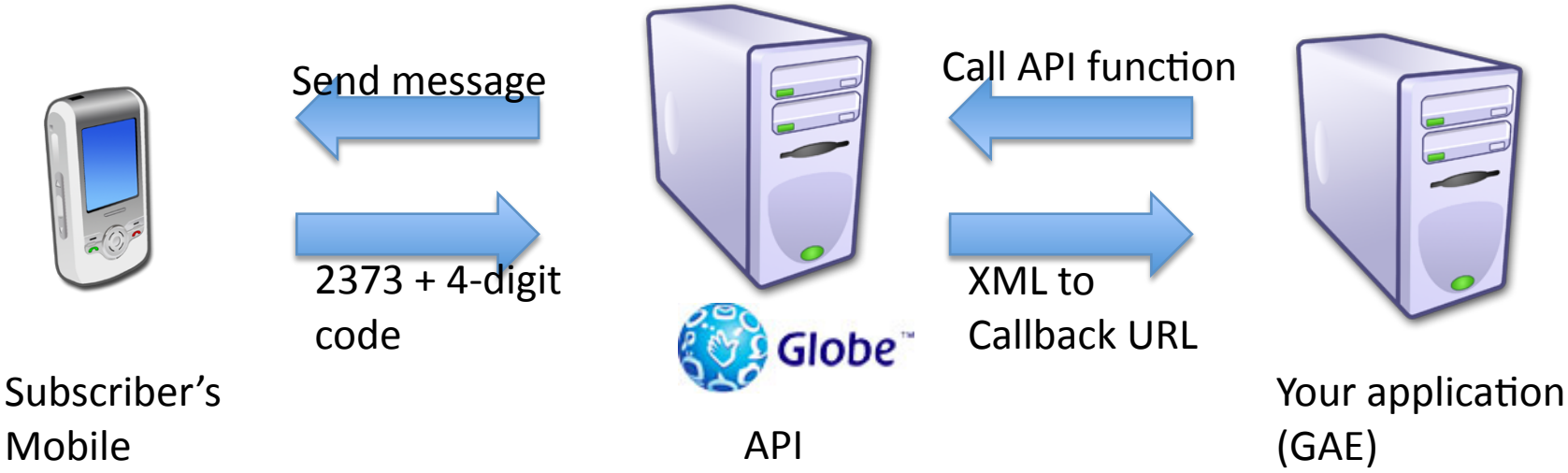
Globe

# MESSAGING APIS (SMS/MMS)



Globe

# Overview




- Send messages to subscribers via the appropriate method (sendSMS/sendMMS), they'll receive the message from your access code. Each request has a return code to indicate success/failure
- Receive messages from subscribers that they send to your access code. API forwards the information as XML



Globe

# Sending a message (SMS)

Make a POST request to

http://iplaypen.globelabs.com.ph:1881/axis2/services/Platform/**sendSMS**

With the following parameter arguments

| sendSMS | |
|---|---|
| uName | API username |
| uPin | API password |
| MSISDN | Subscriber number |
| messageString | Message to be sent |
| Display | Notification / Inbox |
| udh | User data header |
| mwi | Message waiting indicator |
| coding | Bit encoding |



Globe

```
<?xml version='1.0' encoding='UTF-8'?>

<soapenv:Envelope xmlns:soapenv='http://schemas.xmlsoap.org/soap/envelope/'>
  <soapenv:Body>
    <ns:sendSMSResponse xmlns:ns='http://ESCPlatform/xsd'>
      <ns:return>201</ns:return>
    </ns:sendSMSResponse>
  </soapenv:Body>
</soapenv:Envelope>
```



Globe

# Sending a message (MMS)

Make a POST request to

http://iplaypen.globelabs.com.ph:1881/axis2/services/Platform/**sendMMS**

With the following parameter arguments

| sendMMS | |
|---|---|
| uName | API username |
| uPin | API password |
| MSISDN | Subscriber number |
| subject | The subject line for the message |
| smil | SMIL formatted message |



Globe

# A note on SMIL

- Synchronized Multimedia Integration Language – similar to HTML
- MMS messages have their display area divided into different sections
- http://open.movilforum.com/files/documentacion/documentacion31_2.pdf

```
<smil>
<head>
 <meta name="title" content="My Message" />
 <meta name="author" content="Greg Igaya" />
 <layout>
  <root-layout width="160" height="120"/>
  <region id="Image" width="100%"
    height="80" left="0" top="0" />
  <region id="Text" width="100%"
    height="40" left="0" top="80" />
 </layout>
</head>
<body>
<par dur="8s">
 <img src="FirstImage.jpg" region="Image" />
 <text src="FirstText.txt" region="Text" />
 <audio src="FirstSound.amr"/>
</par>
<par dur="7s">
 <img src="SecondImage.jpg" region="Image" />
 <text src="SecondText.txt" region="Text" />
 <audio src="SecondSound.amr" />
</par>
</body>
</smil>
```



Globe

# RECEIVING MESSAGES



Globe

# Receiving a message (SMS)

| parameters | |
|---|---|
| messageType | Type of message (SMS) |
| source | Subscriber's number |
| target | 2373 + 4 digit access code |
| msg | Message sent |
| id | Message identifier |
| udh | User data header |



Globe

# Receiving a message (SMS)

```
<?xml version="1.0" encoding="utf-8"?>
<message>
  <param>
    <name>messageType</name>
    <value>SMS</value>
  </param>
  <param>
    <name>id</name>
    <value>xxxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>source</name>
    <value>xxxxxxxxxxx</value>
  </param>
  <param>
    <name>target</name>
    <value>xxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>msg</name>
    <value>xxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>udh</name>
    <value></value>
  </param>
</message>
```



Globe

# Receiving a message (SMS)

```
<?xml version="1.0" encoding="utf-8"?>
<message>
  <param>
    <name>messageType</name>
    <value>SMS</value>
  </param>
  <param>
    <name>id</name>
    <value>xxxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>source</name>
    <value>xxxxxxxxxxx</value>
  </param>
  <param>
    <name>target</name>
    <value>xxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>msg</name>
    <value>xxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>udh</name>
    <value></value>
  </param>
</message>
```



Globe

# Receiving a message (SMS)

```
<?xml version="1.0" encoding="utf-8"?>
<message>
  <param>
    <name>messageType</name>
    <value>SMS</value>
  </param>
  <param>
    <name>id</name>
    <value>xxxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>source</name>
    <value>xxxxxxxxxxx</value>
  </param>
  <param>
    <name>target</name>
    <value>xxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>msg</name>
    <value>xxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>udh</name>
    <value></value>
  </param>
</message>
```



Globe

# Receiving a message (SMS)

```
<?xml version="1.0" encoding="utf-8"?>
<message>
  <param>
    <name>messageType</name>
    <value>SMS</value>
  </param>
  <param>
    <name>id</name>
    <value>xxxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>source</name>
    <value>xxxxxxxxxxx</value>
  </param>
  <param>
    <name>target</name>
    <value>xxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>msg</name>
    <value>xxxxxxxxxxxxx</value>
  </param>
  <param>
    <name>udh</name>
    <value></value>
  </param>
</message>
```



Globe

# Receiving a message (MMS)

| parameters | |
|---|---|
| messageType | Type of message (SMS) |
| source | Subscriber's number |
| target | 2373 + 4 digit access code |
| file | File list (more file nodes) |
| subject | Subject used by subscriber |

```
<?xml version="1.0" encoding="utf-8"?>
<message>
  <param>
    <name>messageType</name>
    <value>MMS</value>
  </param>
  <param>
    <name>subject</name>
    <value>subject123</value>
  </param>
  <param>
    <name>source</name>
    <value>123</value>
  </param>
  <param>
    <name>target</name>
    <value>123</value>
  </param>
  <param>
    <name>file</name>
    <value>
      <file>http://localhost:1234/testing.jpg</file>
      <file>http://localhost:1234/testing.txt</file>
    </value>
  </param>
</message>
```



Globe

# Using CURL to test

- cURL groks URLs
- curl –d @sms.xml –H "Content–type: text/xml" –v http://localhost:8888/catch.php

**sms.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<message>
<param>
<name>id</name>
<value>2373017320090210204843</value>
</param>
<param>
<name>messageType</name>
<value>SMS</value>
</param>
<param>
<name>target</name>
<value>23730173</value>
</param>
<param>
<name>source</name>
<value>09175889615</value>
</param>
….
```



Globe

# LOCATION BASED SERVICES



Globe

# Location Based Services API

- Allows an application to determine the approximate location of subscriber through the network



Globe

# Enhanced Cell-ID

Timing Advance

Estimate distance to tower

Cell ID Only

Full coverage of cell site 500m to several km

Known coordinate:
14.343231, 121.343543



Globe

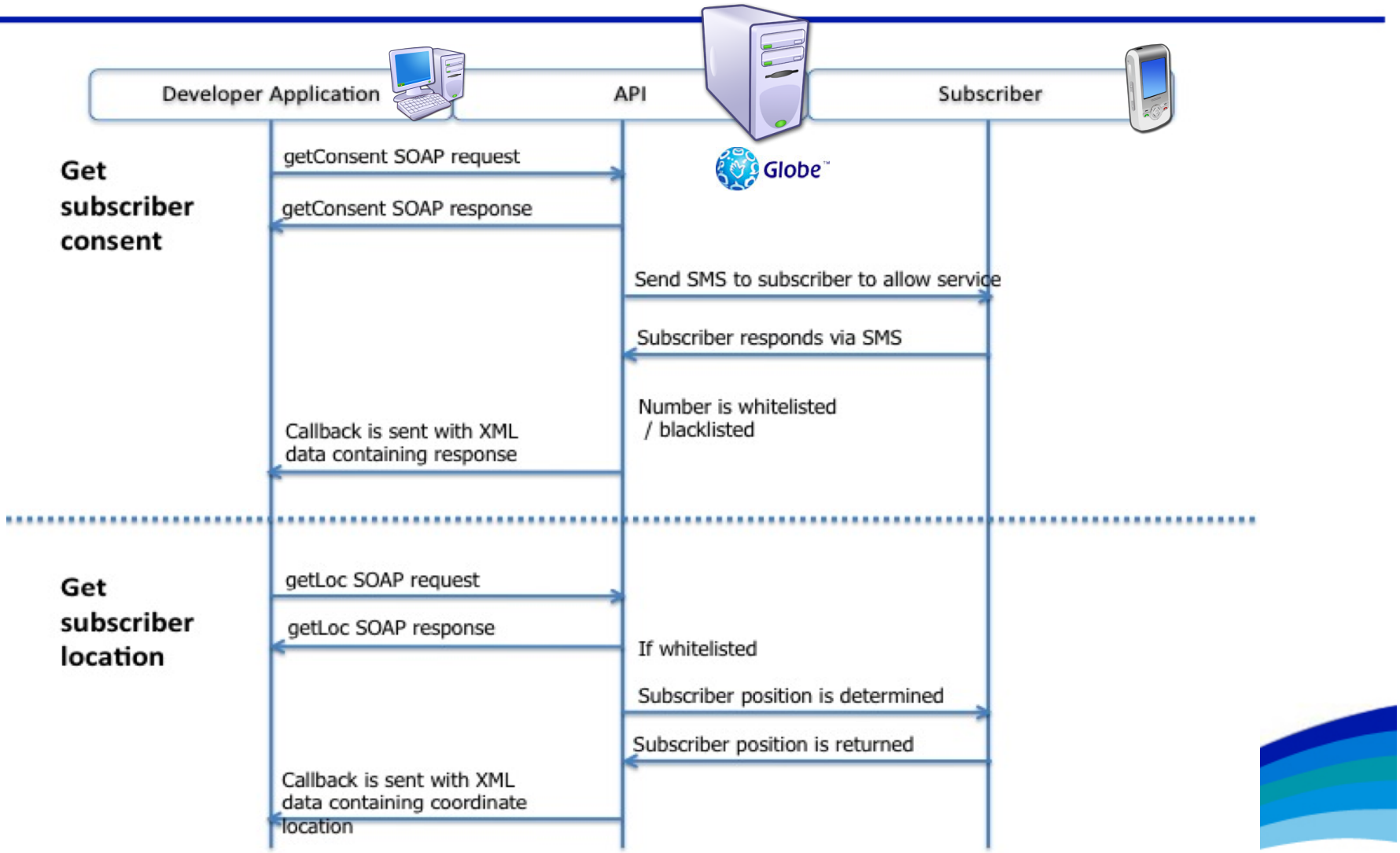
# Enhanced Cell-ID

- Accuracy varies on cell size, but could be 300 meters to several kilometers. Accuracy improves on density.
- no speed or direction of travel is available



Globe

# LBS API - Overview






Globe

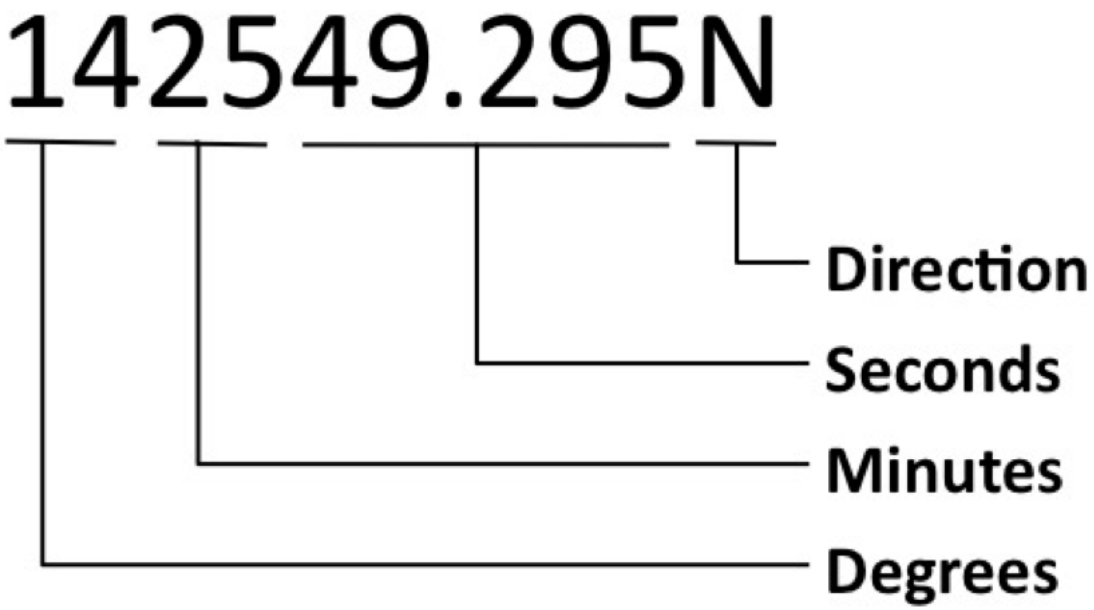
# LBS Notes

- WGS-84 spatial reference system
- Returns coordinate data to application – Degrees Minutes Seconds
- Known Limitations
  - Accuracy of 300 meters in dense cell coverage areas up to several kilometers in rural areas
  - 2G / 3G network divergence
  - 1 TPS limit, 6PM – 2AM restriction



Globe

- X – Latitude – parallel to equator
- Y – Longitude – perpendicular to equator
- How to read returned DMS data



# Application Possibilities

| 300-1000 meters | < 50 meters | < 10 meters |
|---|---|---|
| ✓Location sensitive call routing<br>✓Fleet Management<br>✓City Guide / Yellow Pages / Location sensitive Info Directory or Stream<br>✓Traffic Alerts<br>✓Find your friend<br>✓Remote workforce monitoring<br>✓Points of Interest / Tourist Attraction<br>✓Location contextual entertainment<br>✓Location aware advertising – pull (user initiated)<br>✓Dating / meet-up<br>✓Geo-tagging | ✓Emergency services<br>✓Asset tracking<br>✓Finding<br>✓Location sensitive charging and billing | ✓Safetly and medical monitoring<br>✓Step-by-step navigation<br>✓Location sensitive charging and billing<br>✓Stolen vehicle recovery<br>✓Location aware advertising – push (telco initiated) |

To be compelling, applications built on top of the LBS API need to integrate other proprietary data

- Road maps and POI
- Coordinate locations of restaurants / venues
- Existing social networks



Globe

# GAE & GLOBE LABS APIS



Globe

# What you need to know

- Urlfetch python API to make external requests

```
import urllib

form_fields = {
  "first_name": "Albert",
  "last_name": "Johnson",
  "email_address": "Albert.Johnson@example.com"
}
form_data = urllib.urlencode(form_fields)
result = urlfetch.fetch(url=url,
                        payload=form_data,
                        method=urlfetch.POST,
                        headers={'Content-Type':
'application/x-www-form-urlencoded'})
```



Globe

# What you need to know

- minidom Python API library to process XML

```
from xml.dom import minidom
…

doc = minidom.parseString(body)
 params = doc.getElementsByTagName( "param" )
 for param in params:
      name = param.getElementsByTagName( "name" )[0].firstChild.nodeValue
      test_value = param.getElementsByTagName( "value" )[0].firstChild
      if (test_value != None) :
           value = param.getElementsByTagName( "value" )[0].firstChild.nodeValue
      else:
           value = None
      data[name] = value
```



Globe

# What you need to know

- Extracting the raw HTTP post data

```
class MainHandler(webapp.RequestHandler):

def post(self):
 body = self.request.body
```



Globe

# CODE REVIEW

- http://www.globelabs.com.ph
- globelabs@globetel.com.ph
- http://code.google.com/appengine/



Globe