



Introduction to Web API & Database

Manoj Reddy
PhD Candidate
UCLA Computer Science
mdareddy@cs.ucla.edu

Overview of Lecture

Part-1: Introduction to Web API

- Web API: Simple Example
- Web API vs Website
- Web API in more detail

Part 1 Summary: Web API's are part of larger ecosystem of web development. We will try to touch its basics and introduce the Web API's.

Part-2: Introduction to Databases

- Data
- Databases
- Databases: types
- MySql
- MongoDB
- Exercises

Part 2 Summary: Introduce data and ways to store and query it using databases.

Part-3: Project using Web API and MongoDB.

Web API : Simple Example

Application programming interface (API): Defines method of communication between various software components.

Simple terms: *Web API* provides ways to use computing (Query data, Store data, perform calculations etc) facilities over the internet.

Web API is an evolution of web service.



Hi, this is Mr. Sandy. Can you tell me nearby hotels ?

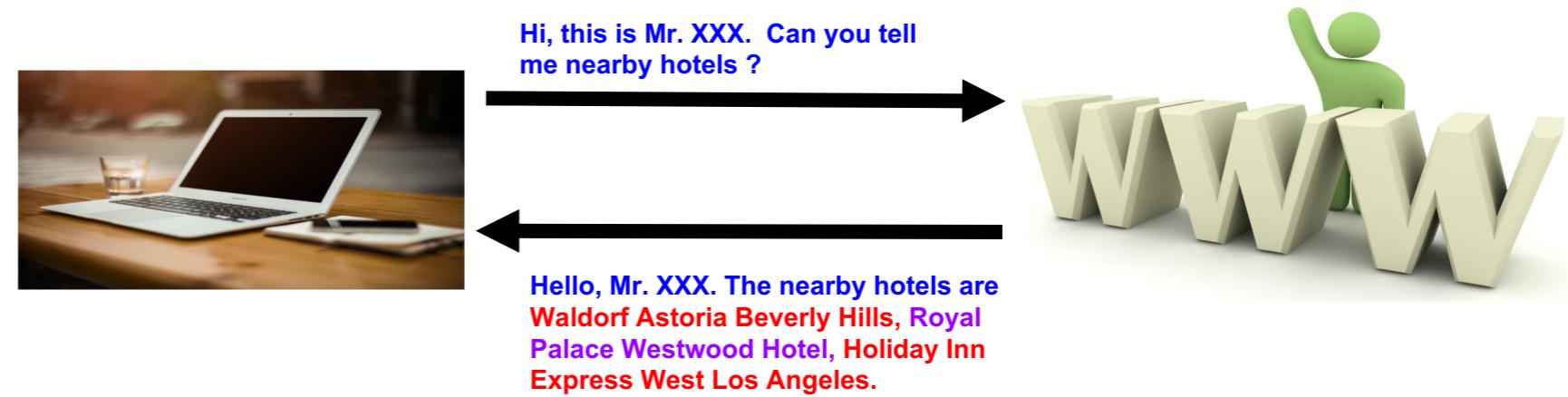


Hello, Mr. Sandy. The nearby hotels are Waldorf Astoria Beverly Hills, Royal Palace Westwood Hotel, Holiday Inn Express West Los Angeles.

Example: Web API to search nearby hotels

Web API Example

Simple terms: *Web API* provides ways to use computing (Query data, Store data, perform calculations etc) facilities over the internet.



What are different components of web api in this example ?

Hint:

How to Communicate/Contact to remote server?

How to understand what we are sending and receiving?

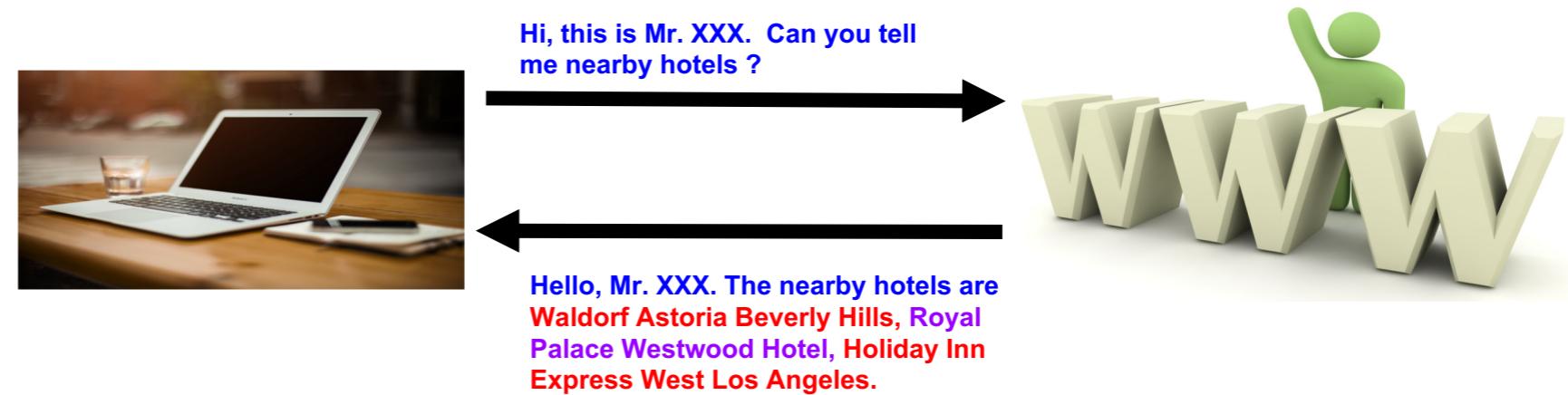
What is my query?

Few other things:

How to know we are serving right person ?

Web API Example

Simple terms: *Web API* provides ways to use computing (Query data, Store data, perform calculations etc) facilities over the internet.



What are different components of web api in this example ?

Hint:

How to Communicate/Contact to Web service? Communication protocols (HTTP)

How to understand what we send and receive? Formatted request & response (Generally: Json)

What is my query? In request: Hotels near (how much near ?) me (what is my location ?)

Few other things:

How to know we are serving right person ? Authentication

Verifying that the service requester is actually Mr. Sandy.

Web API vs Website

Website is collection of similar web pages. Web page is a document displayed by web browser.

Website may or may not use web api to do query and get data from multiple servers. *Website take care of data presentation to user but web api doesn't do that.*

hotels near me

Dates of travel: Aug 27 - Aug 28 | Deals | Price: Any price | Rating: ★★★★★

Waldorf Astoria Beverly Hills \$743
Ad 4.5 ★★★★★
5-star hotel
Luxury hotel with 360-degree city views

Hotel Angeleno \$181
Ad 3.8 ★★★★★
3-star hotel
Tower lodging with views & free parking
Free Wi-Fi
DEAL 16% off

Royal Palace Westwood Hotel \$189
3.9 ★★★★★
2-star hotel
Unassuming property with free parking
Free breakfast

About pricing | Showing results 1 - 20 | < >

Update results when map moves

The screenshot shows a search interface for "hotels near me" with filters for travel dates (Aug 27 - Aug 28), deals, price (Any price), and rating (★★★★★). It lists three hotel results:

- Waldorf Astoria Beverly Hills**: \$743, Ad, 4.5 stars, 5-star hotel, luxury hotel with 360-degree city views.
- Hotel Angeleno**: \$181, Ad, 3.8 stars, 3-star hotel, tower lodging with views & free parking, free Wi-Fi, 16% off deal.
- Royal Palace Westwood Hotel**: \$189, 3.9 stars, 2-star hotel, unassuming property with free parking, free breakfast.

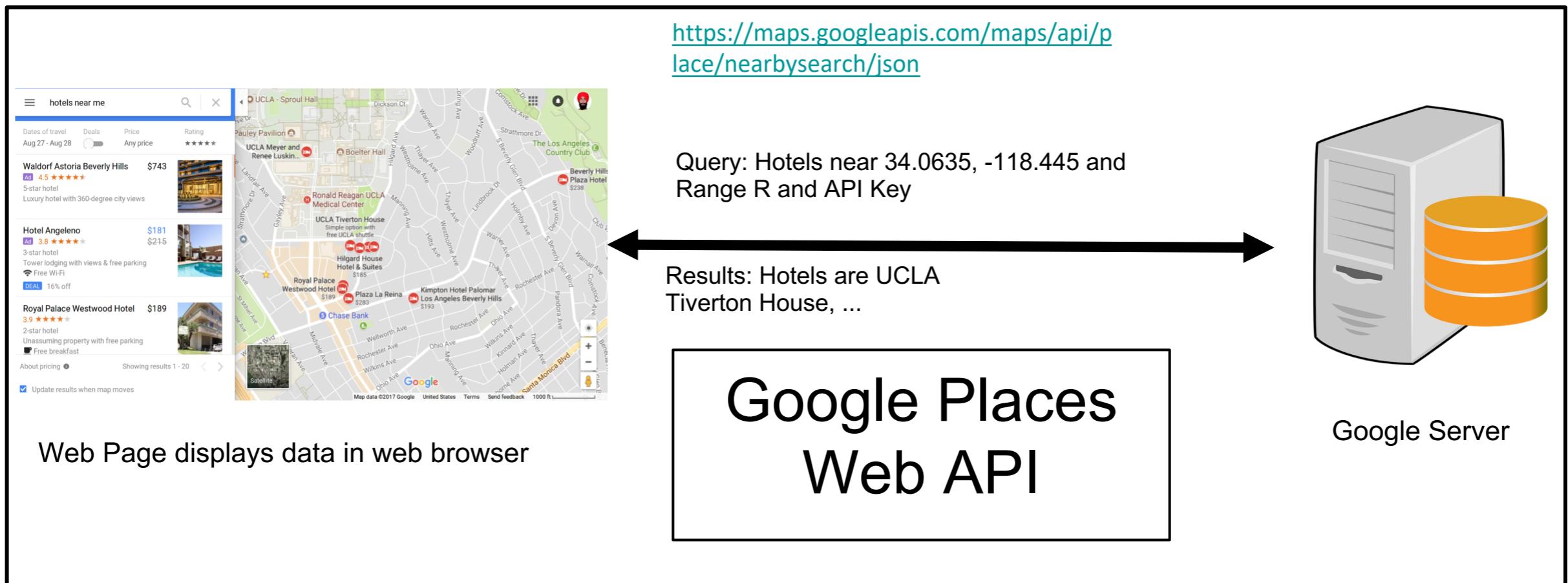
Below the results is a map of the UCLA and Beverly Hills area, showing the locations of various hotels and landmarks. The map includes labels for UCLA-Sprout Hall, Pauley Pavilion, Boelter Hall, Ronald Reagan UCLA Medical Center, UCLA Tiverton House, Hilgard House Hotel & Suites, Royal Palace Westwood Hotel, Plaza La Reina, Kimpton Hotel Palomar Los Angeles Beverly Hills, Chase Bank, and The Los Angeles Country Club. The map also shows streets like Westholme Ave, Manning Ave, Warner Ave, Thayer Ave, Holmby Ave, S. Beverly Glen Blvd, Lindbrook Dr, Club Dr, and Santa Monica Blvd.

Web API vs Website

Website take care of data presentation to user but web api doesn't do that.

WebPage: <https://www.google.com/maps/search/hotels+near+me/@34.0635363,-118.4455592,15z>

Query: Hotels near me, what is my location (Where to find hotels) & my zoom level (In how much area to search).

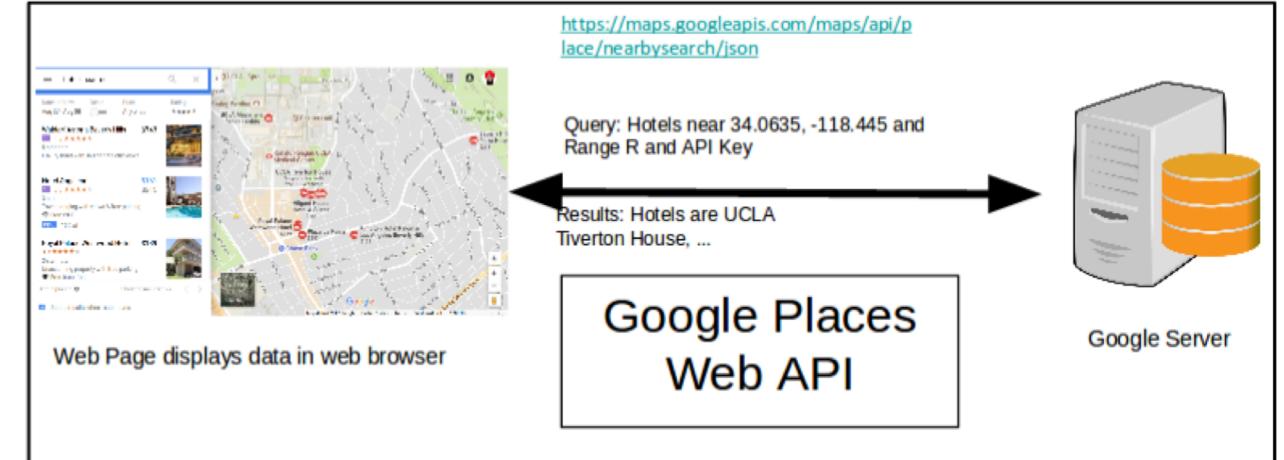


Web API vs Website

Website take care of data presentation to user but web service doesn't do that.

WebPage: <https://www.google.com/maps/search/hotels+near+me/@34.0635363,-118.4455592,15z>

```
{
  "candidates" : [
    {
      "formatted_address" : "140 George St, The Rocks NSW 2000, Australia",
      "geometry" : {
        "location" : {
          "lat" : -33.8599358,
          "lng" : 151.2090295
        },
        "viewport" : {
          "northeast" : {
            "lat" : -33.85824767010727,
            "lng" : 151.2102470798928
          },
          "southwest" : {
            "lat" : -33.86094732989272,
            "lng" : 151.2075474201073
          }
        }
      },
      "name" : "Museum of Contemporary Art Australia",
      "opening_hours" : {
        "open_now" : false,
        "weekday_text" : []
      },
      "photos" : [
        {
          "height" : 2268,
```

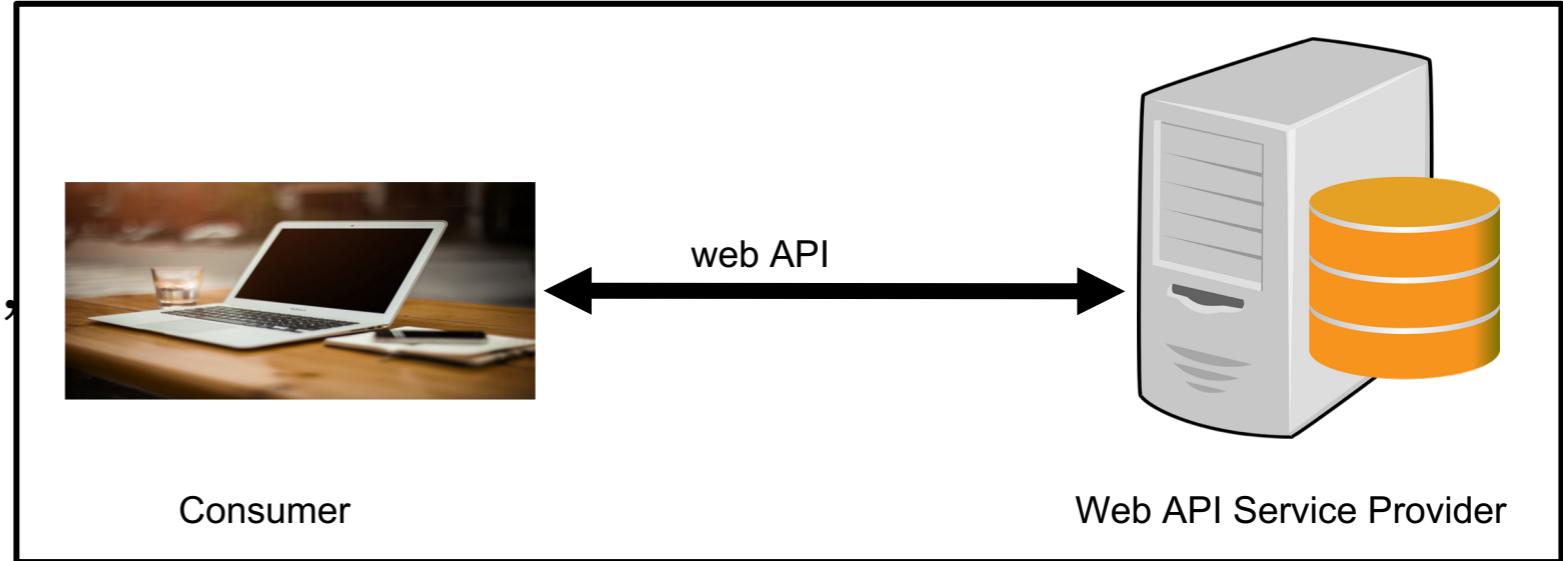


Defining Web API

Simple terms: *Web API* provides ways to use computing (Query data, Store data, perform calculations etc) facilities over the internet. A Computer/Cluster/Cloud away from user, may provide different functionalities to user by offering a *Web API*. It is a concept not a technology.

This functionality may be:

- Saving data.
- Running computations (eg. query, transformations, calculations).
- Returning results (data).
- or multiple of above.

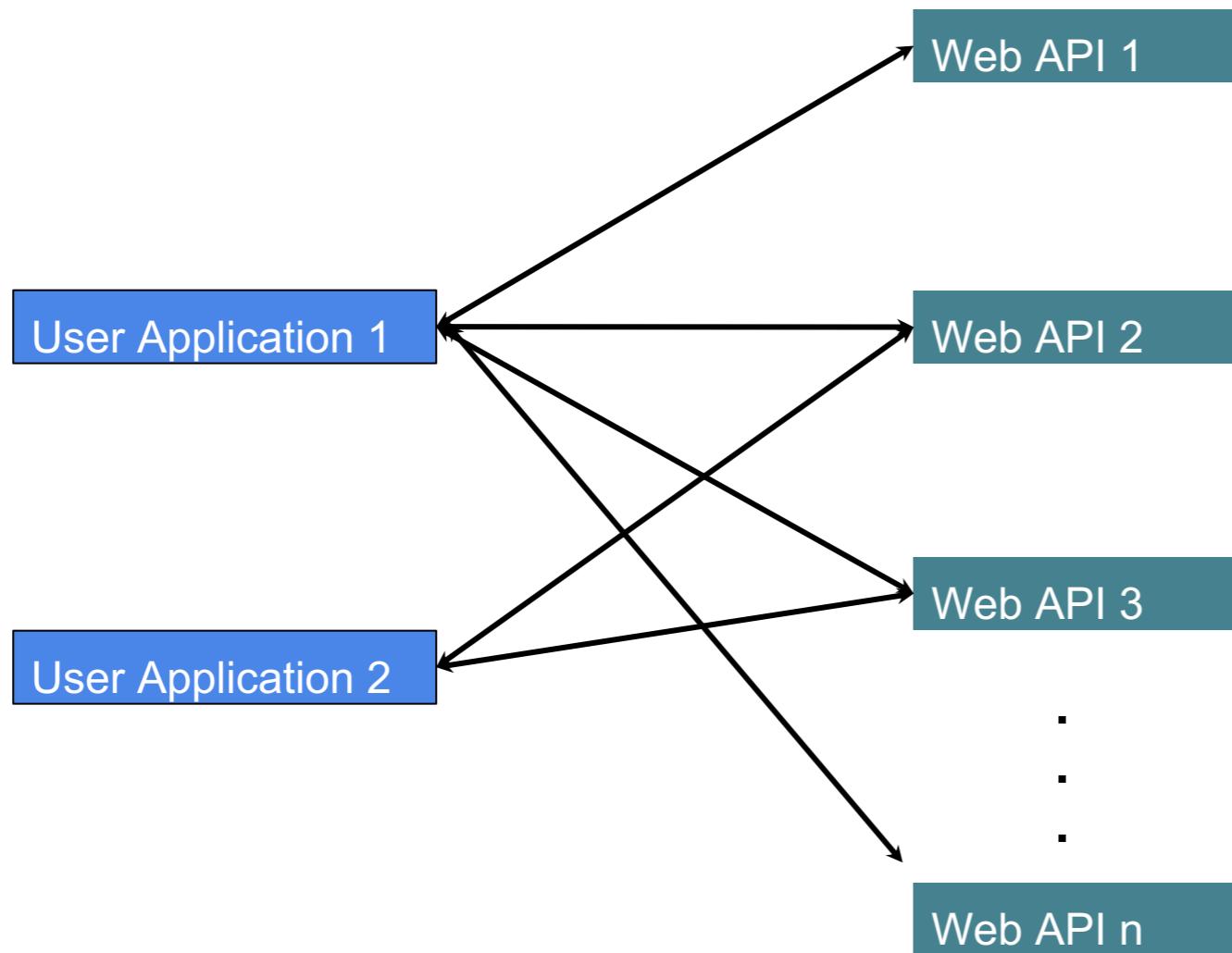


Definition:

Web API is an application programming interface for a web server. Web API doesn't include web server implementation details.

Benefits of Web API

- Loosely Coupled
- Ease of Integration
- Service Reuse



Public APIs

- **Curated at:**
 - <https://github.com/public-apis/public-apis/blob/master/README.md>

Components of Web API

In order to get functionality we should know:

Required:

1. From where to get service?

Service url of the API.

Eg. <https://maps.googleapis.com/maps/api/place/nearbysearch/json>

2. What is the format of the request?

Specifying query and input data etc. |

Eg. location=34.0635363,-118.4455592&radius=1000&type=hotels

3. What is the response format?

Response can be string, xml or Json

Eg. Google places API returns result in Json format.

Optional:

1. How to differentiate between different users or restrict service access?

Allowed users vs not allowed. Authentication. Eg: key

2. Error codes in the request?

Inform user if anything is missing in the request. Error codes.

Web API Example

- Weather API (<https://www.metaweather.com/api/location/2442047/>)
- <https://www.metaweather.com/api/location/search/?query={placename}>

```
{"consolidated_weather":[{"id":4600897889894400,"weather_state_name":"Light Cloud","weather_state_abbr":"lc","wind_direction_compass":"SW","created":"2019-07-14T18:07:26.105127Z","applicable_date":"2019-07-14","min_temp":18.77000000000003,"max_temp":27.07500000000003,"the_temp":26.68,"wind_speed":4.766953356470214,"wind_direction":218.3462382308425,"air_pressure":1014.339999999999,"humidity":59,"visibility":12.44575429491768,"predictability":70},{"id":5568470940909568,"weather_state_name":"Clear","weather_state_abbr":"c","wind_direction_compass":"SW","created":"2019-07-14T18:07:29.009809Z","applicable_date":"2019-07-15","min_temp":20.105,"max_temp":29.68999999999998,"the_temp":28.975,"wind_speed":4.548633621349983,"wind_direction":214.6537617691575,"air_pressure":1012.204999999999,"humidity":48,"visibility":10.29456722739203,"predictability":68},{"id":6722281021636608,"weather_state_name":"Clear","weather_state_abbr":"c","wind_direction_compass":"SSW","created":"2019-07-14T18:07:32.286054Z","applicable_date":"2019-07-16","min_temp":21.215,"max_temp":28.275,"the_temp":28.68999999999998,"wind_speed":5.080530108780722,"wind_direction":206.25006469587407,"air_pressure":1009.19,"humidity":45,"visibility":13.132058776743817,"predictability":68},{"id":6042753911750656,"weather_state_name":"Clear","weather_state_abbr":"c","wind_direction_compass":"SSE","created":"2019-07-14T18:07:35.276288Z","applicable_date":"2019-07-17","min_temp":19.045,"max_temp":26.505,"the_temp":26.5,"wind_speed":6.10168713157825,"wind_direction":164.2888544336482,"air_pressure":1011.245,"humidity":57,"visibility":15.095591744213792,"predictability":68},{"id":5568669952245760,"weather_state_name":"Light Cloud","weather_state_abbr":"lc","wind_direction_compass":"SSW","created":"2019-07-14T18:07:38.103907Z","applicable_date":"2019-07-18","min_temp":16.86500000000002,"max_temp":23.545,"the_temp":23.725,"wind_speed":5.468079871319495,"wind_direction":191.48804182014615,"air_pressure":1013.694999999999,"humidity":62,"visibility":15.20308896047085,"predictability":70},{"id":6365747162906624,"weather_state_name":"Clear","weather_state_abbr":"c","wind_direction_compass":"SSW","created":"2019-07-14T18:07:40.967087Z","applicable_date":"2019-07-19","min_temp":16.645,"max_temp":23.655,"the_temp":20.05,"wind_speed":3.442445915851428,"wind_direction":212.99999999999997,"air_pressure":1013.47,"humidity":64,"visibility":9.999726596675416,"predictability":68}], "time": "2019-07-14T11:52:06.385529-07:00", "sun_rise": "2019-07-14T05:52:01.894991-07:00", "sun_set": "2019-07-14T20:05:49.136713-07:00", "timezone_name": "LMT", "parent": {"title": "California", "location_type": "Region / State / Province", "woeid": 2347563, "latt_long": "37.271881,-119.270233"}, "sources": [{"title": "BBC", "slug": "bbc", "url": "http://www.bbc.co.uk/weather/", "crawl_rate": 360}, {"title": "Forecast.io", "slug": "forecast-io", "url": "http://forecast.io/", "crawl_rate": 480}, {"title": "HAMweather", "slug": "hamweather", "url": "http://www.hamweather.com/", "crawl_rate": 360}, {"title": "Met Office", "slug": "met-office", "url": "http://www.metoffice.gov.uk/", "crawl_rate": 180}, {"title": "OpenWeatherMap", "slug": "openweathermap", "url": "http://openweathermap.org/", "crawl_rate": 360}], "title": "Weather Underground", "slug": "wunderground", "url": "https://www.wunderground.com/?apiref=fc30dc3cd224e19b", "crawl_rate": 720}, {"title": "World Weather Online", "slug": "world-weather-online", "url": "http://www.worldweatheronline.com/", "crawl_rate": 360}], "title": "Los Angeles", "location_type": "City", "woeid": 2442047, "latt_long": "34.053490,-118.245323", "timezone": "US/Pacific"}
```

Overview of Lecture

Part-1: Introduction to Web API

- Web API: Simple Example
- Web API vs Website
- Web API in more detail

Part 1 Summary: Web API's are part of larger ecosystem of web development. We will try to touch its basics and introduce the Web API's.

Part-2: Introduction to Databases

- Data
- Databases
- Databases: types
- MySql
- MongoDB
- Exercises

Part 2 Summary: Introduce data and ways to store and query it using databases.

Part-3: Project using Web API and MongoDB.

What is Data?

Data: A piece of information.

Anything on which operations can be performed by computer, can be stored and transmitted.

Data is least abstract. Information is next and knowledge is most abstract.

Data is Future? World is driven by data..?



Types of Data :

At higher level data is of following three types depending on its structure:

1. Structured data: Expressed using Tables.
2. Semi-structured data: Expressed using XML or Json.
3. Un-structured data: Expressed as plain text

Other types: quantitative data vs qualitative data

Which type of data is **easy to use** ?

which type of data exist in **abundance** ?

```
<dataitem>
  <city>"Los Angeles"</city>
  <name>"Sandeep"</name>
  <id>20</id>
</dataitem>
```

```
{
  "City": "Los Angeles",
  "Name": "Sandeep",
  "id": 20
}
```

| | | | | | | |
|----|----|---|--------|--------|--------|-------|
| 5 | 23 | 2 | 0.913 | 0.0588 | 0.8049 | 1 |
| 8 | 21 | 2 | 0.8261 | 0.079 | 0.6848 | 0.996 |
| 9 | 19 | 1 | 0.7826 | 0.086 | 0.631 | 0.971 |
| 12 | 18 | 1 | 0.7391 | 0.0916 | 0.5798 | 0.942 |
| 13 | 17 | 1 | 0.6957 | 0.0959 | 0.5309 | 0.912 |
| 18 | 14 | 1 | 0.646 | 0.1011 | 0.4753 | 0.878 |
| 23 | 13 | 2 | 0.5466 | 0.1073 | 0.3721 | 0.803 |
| 27 | 11 | 1 | 0.4969 | 0.1084 | 0.324 | 0.762 |
| 30 | 9 | 1 | 0.4417 | 0.1095 | 0.2717 | 0.718 |
| 31 | 8 | 1 | 0.3865 | 0.1089 | 0.2225 | 0.671 |
| 33 | 7 | 1 | 0.3313 | 0.1064 | 0.1765 | 0.622 |
| 34 | 6 | 1 | 0.2761 | 0.102 | 0.1338 | 0.569 |
| 43 | 5 | 1 | 0.2208 | 0.0954 | 0.0947 | 0.515 |
| 45 | 4 | 1 | 0.1656 | 0.086 | 0.0598 | 0.458 |
| 48 | 2 | 1 | 0.0828 | 0.0727 | 0.0148 | 0.462 |

<xml />

{JSON}

Exercise: Parsing Json

JSON (JavaScript Object Notation) is a lightweight data-interchange format.

Example Json Data:

In python: using json library

```
{  
    "City": "Los Angeles",  
    "Name": "Sandeep",  
    "id": 20  
}
```

```
>>> import json  
>>> print(json.dumps({'Name': 'Sandeep', 'City': "Los Angeles",'id':20}, sort_keys=True, indent=4))  
{  
    "City": "Los Angeles",  
    "Name": "Sandeep",  
    "id": 20  
}
```

Parsing Json Data:

```
import json  
data = '{"Name" : "Sandeep", "City" : "Los Angeles", "id" : 20}'  
j = json.loads(data)  
print(j['Name'])  
Sandeep
```

Database

A Database is structured collection of data.

Example: Telephone directory, Dictionary and many more.

Databases can be stored in computer and analyzed by program. Programs are often called *database management systems* or in short *databases*. We call the programs which help us to analyze data as databases too.

Different types of databases to store different type of data.

Normally we have two types of datastores:

1. **Relational database.** (SQL)
2. **Non-Relational database.** (NoSQL)

Relational Databases

Databases whose organization is based on relations. We have tables and data items are inserted in the form of rows.

Also called SQL databases.

Various relational databases: MySQL, Oracle, Microsoft SQL, IBM DB2.

SQL: **Structured query language.** Used to insert, query and update data items.

Most of early data stored in tables. eg. Data of big banks in Oracle database.

SQL: Queries

Creating Table:

```
CREATE TABLE table_name (
    column1 datatype,
    column2 datatype,
    column3 datatype,
    ...
);
```

Inserting Data:

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

Query Data:

```
SELECT column1, column2, ... FROM
table_name WHERE condition;
```

Database language, similar to python, but designed to work with database. Can do many complex things, which we left for simplicity.

SQL Tutorial

- Resource
 - <https://www.w3schools.com/sql/default.asp>
- Database can have multiple tables
- Schema defines the columns and their types of a table

Index in Databases

- Makes retrieval much faster
- Example: Building a search engine
 - Web pages contains text (collection of words)
 - User should be able to issue a query (couple of words)
 - Search engine needs to efficiently return a set of webpages relevant to that query
- How would you implement it using a traditional database like MySQL?
- Can you make it more efficient?

NoSQL Databases: MongoDB

NoSQL: Used to Store data which is not in tabular form and is semi-structured.
Eg: Json data.

Also called not only SQL.

Today, used increasingly to store big data in web applications.

MongoDB: MongoDB is a **document database** with the scalability and flexibility that you want with the querying and indexing that you need.

Json data items are stored in documents. Documents are similar to tables.
Every Json data item is equivalent to row.

Exercise 2: Using MongoDB in Python

1. Start MongoDB service on your machine. (First install MongoDB)
2. Python: Connect to MongoDB and insert few data items.
3. Query

Demo 1:

```
import pymongo
from pymongo import MongoClient
import json

client = MongoClient()
db = client.test_database
collection = db.test_collection

data = '{"Name" : "Sandeep", "City" : "Los Angeles", "id" : 20}'
j = json.loads(data)
data_id = collection.insert_one(j).inserted_id
collection.find_one()
```

Overview of Lecture

Part-1: Introduction to Web API

- Web API: Simple Example
- Web API vs Website
- Web API in more detail

Part 1 Summary: Web API's are part of larger ecosystem of web development. We will try to touch its basics and introduce the Web API's.

Part-2: Introduction to Databases

- Data
- Databases
- Databases: types
- MySql
- MongoDB
- Exercises

Part 2 Summary: Introduce data and ways to store and query it using databases.

Part-3: Project using Web API and MongoDB.

Project

- Individual or Groups of 2 members
- Explore any freely available public APIs (No Key Required)
- Build an interesting and useful Application
 - Can use HTML/CSS/JS etc.
- Examples:
 - Build a Library of Books using the Google Books API
 - Crawl top 10 books of following genres
 - Science Fiction
 - Autobiography
 - Mystery
 - History
 - Store metadata into the MongoDB database
 - Author name
 - Publisher
 - Year
- Be Creative!!!