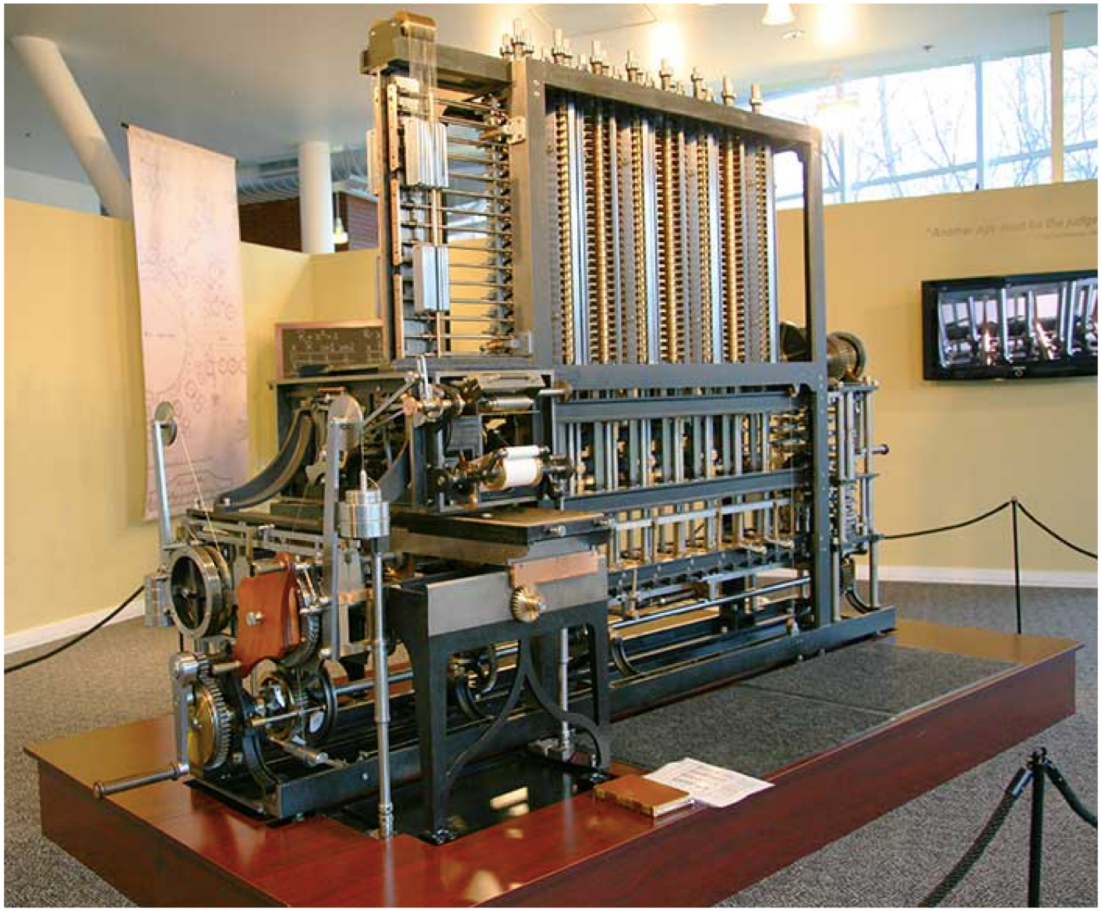
Introduction to Web Services

Assoc. Prof. Dr. Kanda Runapongsa Saikaew
Department of Computer Engineering
Faculty of Engineering
Khon Kaen University

Agenda

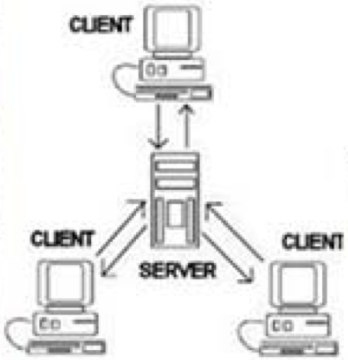
- The Evolution of Computing
- What are Web Services?
- Why Web Services?
- Web Services Architecture and Standards
- Where are Web Services?
- Web Services Development

What is This?



Courtesy of Jitze Couperus. Copyright: CC-Att-SA-2 (Creative Commons Attribution-ShareAlike 2.0 Unported).

The Computing Evolution



ENIAC (Electronic numerical integrator And Computer)

IBM Mainframe

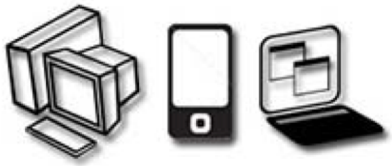
Microsoft Office, OS

World-Wide-Web Google

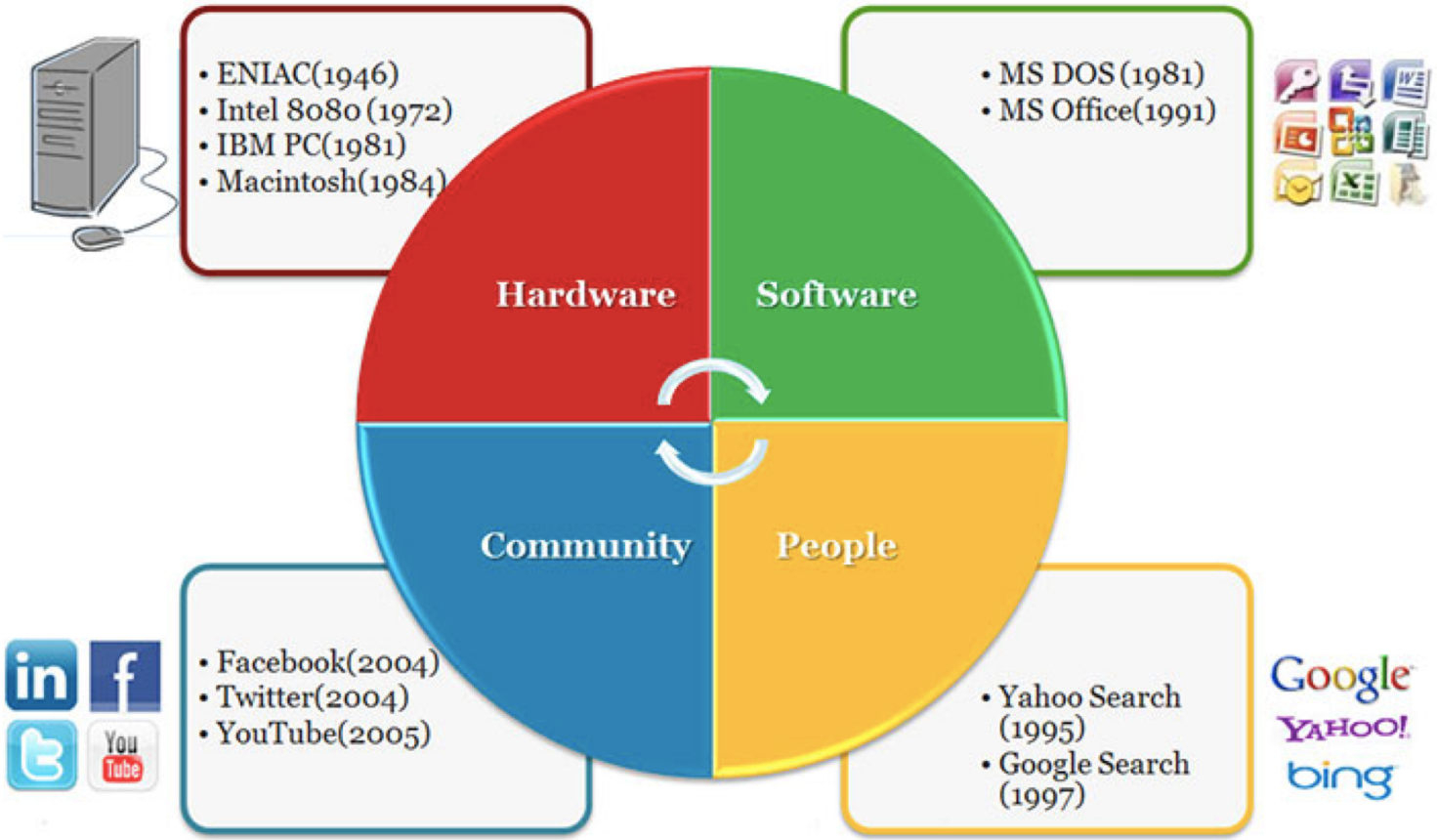
Social Media

Computer System Levels

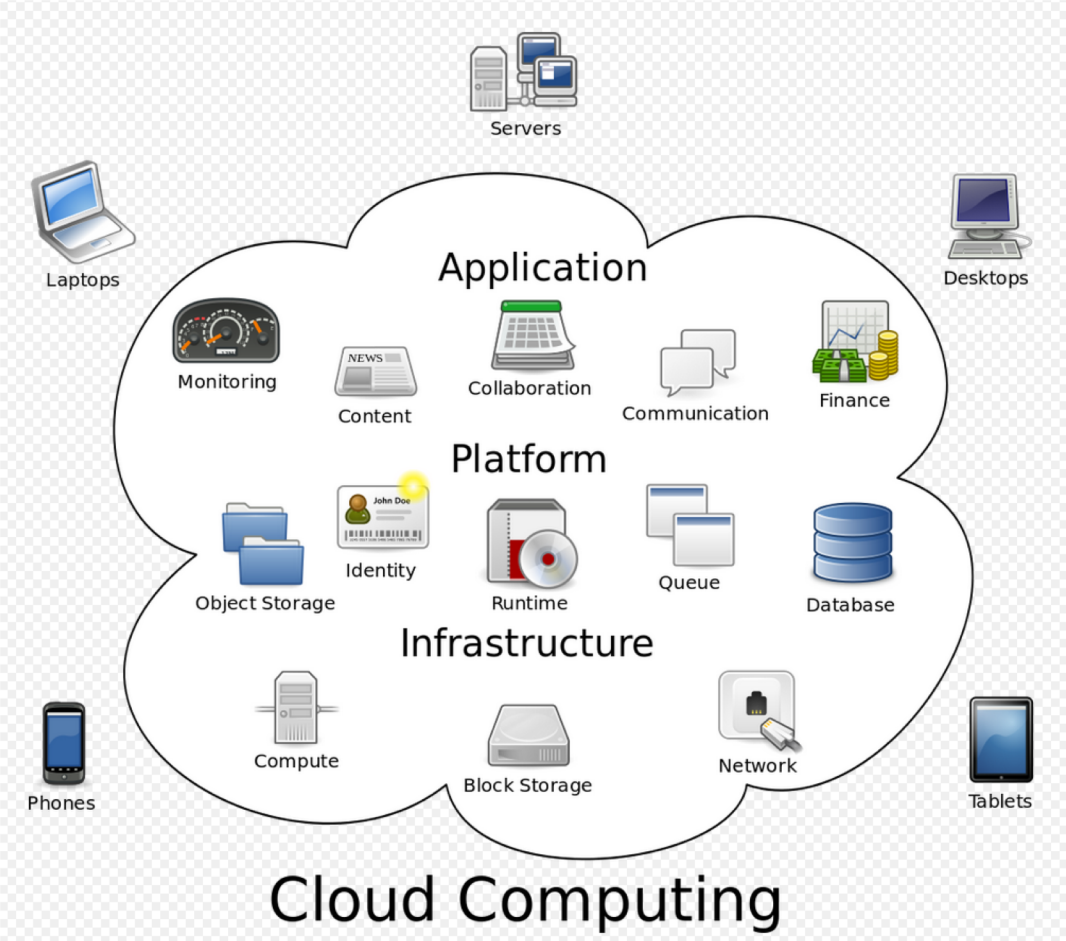
STS	Community + HCI(s)
HCI	Person + IT(s)
IT	Software + device(s)
Technology	Any device



The Four Stages of Computing



Cloud Computing



Cloud Computing

- **Cloud computing** is the delivery of computing as a service rather than a product
- Shared resources, software, and information are provided to computers and other devices as a utility (like the electricity grid) over a network (typically the Internet)
- Clouds can be classified as public, private or hybrid

Why Cloud Computing?

- Third-party clouds **enable organizations to focus on their core businesses** instead of expending resources on computer infrastructure and maintenance
- Advocates note that cloud computing allows companies to avoid or **minimize** up-front IT infrastructure **costs**
- Proponents also claim that cloud computing allows enterprises to get their applications up and running **faster**

Cloud Computing

- **Cloud computing** is the delivery of computing as a service rather than a product
- Shared resources, software, and information are provided to computers and other devices as a utility (like the electricity grid) over a network (typically the Internet)
- Clouds can be classified as public, private or hybrid

Cloud Computing

- **Cloud computing** is the delivery of computing as a service rather than a product
- Shared resources, software, and information are provided to computers and other devices as a utility (like the electricity grid) over a network (typically the Internet)
- Clouds can be classified as public, private or hybrid

The Cloud

Cloud Computing Players

- In 2006, Amazon Web Services
- In 2008, Google App Engine
- In 2010, Microsoft Azure
- In 2011, IBM SmartCloud
- In 2012, Oracle Cloud
- In 2012, Google Compute Engine

Cloud Computing Players

- In 2006, Amazon Web Services
- In 2008, Google App Engine
- In 2010, Microsoft Azure
- In 2011, IBM SmartCloud
- In 2012, Oracle Cloud
- In 2012, Google Compute Engine

Virtualization

- The main enabling technology for cloud computing is virtualization
- Virtualization software separates a physical computing device into one or more “virtual” devices
  - Each can be easily used and managed to perform computing tasks
- Virtualization provides the agility required to speed up IT operations, and reduces cost by increasing infrastructure utilization

Service-oriented Architecture (SOA)

- Users routinely face difficult business problems
- Cloud computing adopts concepts from Service-oriented Architecture (SOA) that can help the user break these problems into services
- Cloud computing provides all of its resources as services

Characteristics of Cloud Computing

- **Cost** reductions are claimed by cloud providers. A public-cloud delivery model converts capital expenditures (e.g., buying servers) to operational expenditure.
  - Less in-house IT skills are required for implementation of projects that use cloud computing
- **Device and location independence** enable users to access systems regardless of their location or what device they use
- **Maintenance** of cloud computing applications is easier, because they do not need to be installed on each user's computer and can be accessed from different places

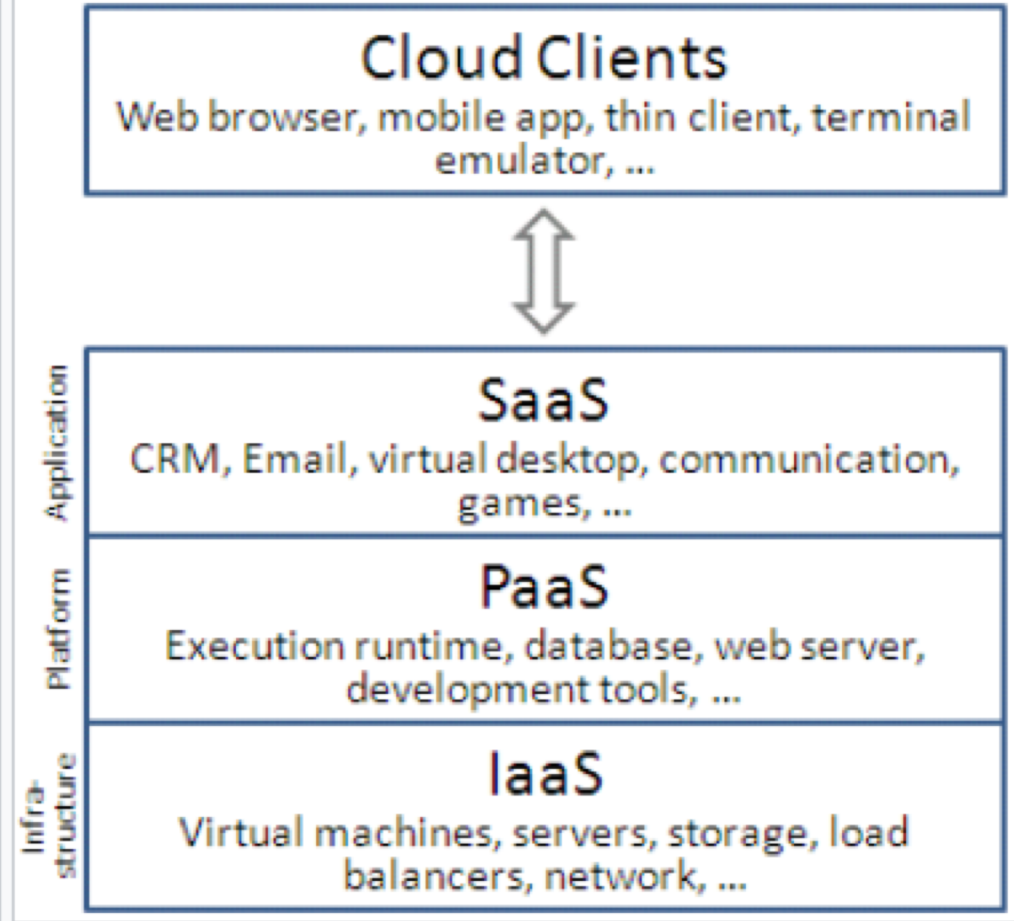
Characteristics of Cloud Computing

- **Performance i**s monitored, and consistent and loosely coupled architectures are constructed using web services as the system interface
- **Productivity** may be increased when multiple users can work on the same data simultaneously, rather than waiting for it to be saved and emailed
- **Reliability** improves with the use of multiple redundant sites, which makes well-designed cloud computing suitable for business continuity and disaster recovery

Characteristics of Cloud Computing

- **Performance i**s monitored, and consistent and loosely coupled architectures are constructed using web services as the system interface
- **Productivity** may be increased when multiple users can work on the same data simultaneously, rather than waiting for it to be saved and emailed
- **Reliability** improves with the use of multiple redundant sites, which makes well-designed cloud computing suitable for business continuity and disaster recovery

Service Models



Cloud computing service models arranged as layers in a stack

Mobile “backend” as a service (Mbaas)

- Backend as a service (BaaS) is a model for providing web app and mobile app developers with a way to link their applications to backend cloud storage and APIs
- BaaS also provides features such as user management, push notifications, and integration with social networking services
- These services are provided via the use of custom software development kits (SDKs) and application programming interfaces (APIs)

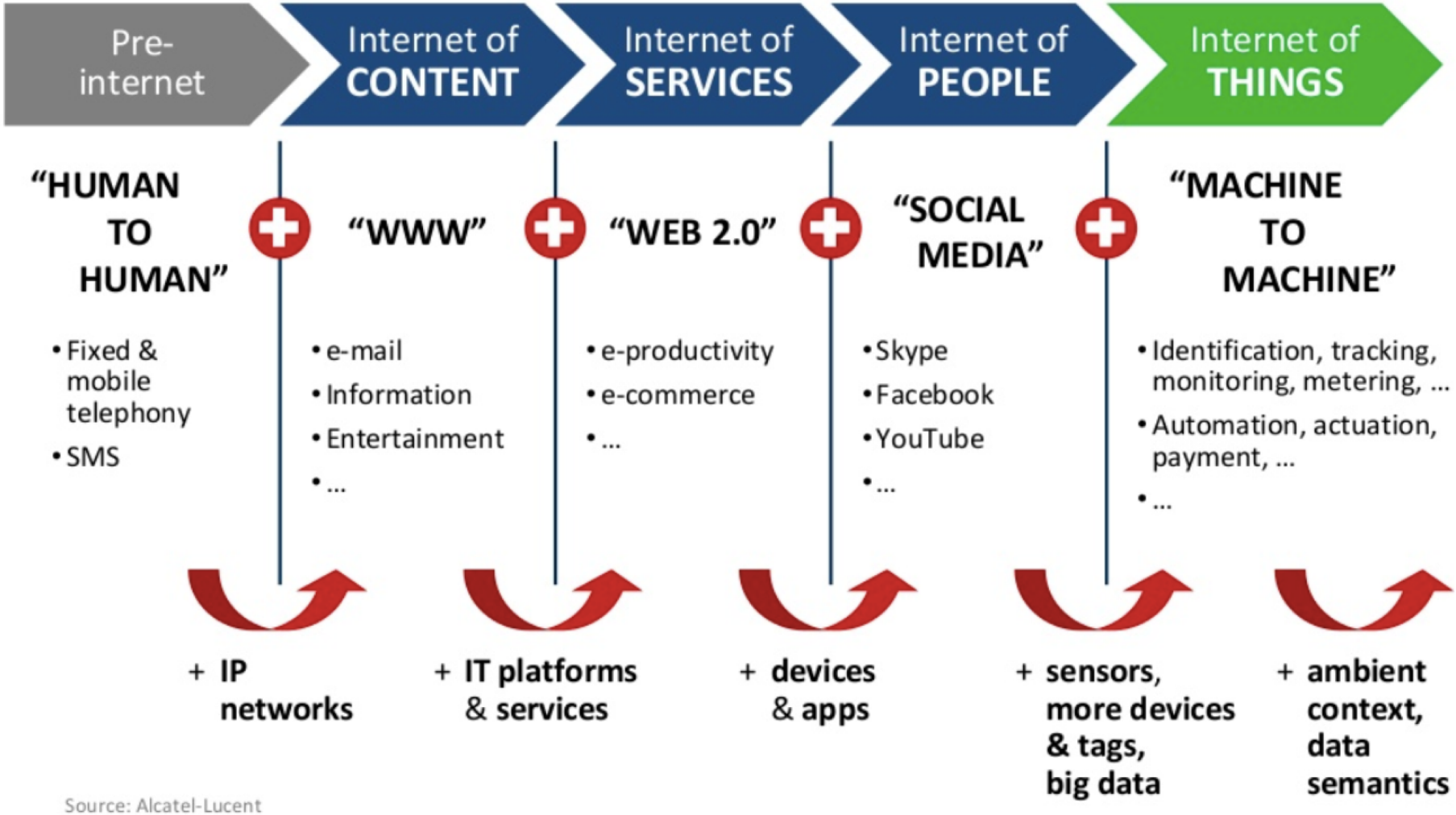
Distributed Computing Technologies

Client-server silos

Web-based computing

Web Services/Peer-to-Peer

The next step in internet evolution



Source: Alcatel-Lucent

Agenda

- The Evolution of Computing
- **What are Web Services?**
- Why Web Services?
- Web Services Architecture and Standards
- Where are Web Services?
- Web Services Development

Traditional Web Application vs. Web Services

- Traditional Web Application
  - User-to-program interaction
  - Static integration of components
  - Monolithic service
- Web Services
  - Program-to-program interaction
  - Dynamic integration of components
  - Service aggregation

What are Web Services? (1/2)

- Excerpt from http://en.wikipedia.org/wiki/Web_service
- Web services are typically application programming interfaces (API) or Web APIs that are accessed via Hypertext Transfer Protocol (HTTP) and executed on a remote system hosting the requested services
- Web services tend to fall into one of two camps: big Web services and RESTful Web services.

Big Web Services

- "Big Web services" use Extensible Markup Language (XML) messages that follow the SOAP standard and have been popular with traditional enterprise
- In such systems, there is often a machine-readable description of the operations offered by the service written in the Web Services Description Language (WSDL)
- The latter is not a requirement of a SOAP endpoint, but it is a prerequisite for automated client-side code generation in many Java and .NET SOAP frameworks

Big Web Services

- "Big Web services" use Extensible Markup Language (XML) messages that follow the SOAP standard and have been popular with traditional enterprise
- In such systems, there is often a machine-readable description of the operations offered by the service written in the Web Services Description Language (WSDL)
- The latter is not a requirement of a SOAP endpoint, but it is a prerequisite for automated client-side code generation in many Java and .NET SOAP frameworks

Characteristics of Big Web Services

- XML based everywhere
- Message-based
- Programming language independent
- Could be dynamically located
- Could be dynamically assembled or aggregated
- Accessed over the internet
- Loosely coupled
- Based on industry standards

SOAP based Web Services is a great solution when you need,

- SOAP is good for applications that require *formal contracts* between the API and consumer since it can enforce the use of formal contracts by using WSDL (Web Services Description Language)
- SOAP has built in *WS-Reliable messaging* to increase security in asynchronous execution and processing

Representational state transfer (REST)

- REST attempts to describe architectures which use HTTP or similar protocols by constraining the interface to a set of well-known, standard operations (like GET, POST, PUT, DELETE for HTTP)
- Here, the focus is on interacting with stateful resources, rather than messages or operations

REST + JSON: The Easy Way to Expose Web Services

- REST is easy to understand
  - It uses HTTP and basic CRUD operations
- REST also makes efficient use of bandwidth
  - REST reads can be cached for better performance and scalability
- REST supports many data formats, but the predominant use of JSON means better support for browser clients

Web API

- Web API is a development in Web services (in a movement called Web 2.0)
- Emphasis has been moving away from SOAP based services towards Representational State Transfer (REST) based communications
- REST services do not require XML, SOAP, or WSDL service-API definitions.

Web APIs and Mashups

- Web APIs allow the combination of multiple Web services into new applications known as mashups
- When used in the context of Web development, Web API is typically a defined set of Hypertext Transfer Protocol (HTTP) request messages along with a definition of the structure of response messages, usually expressed in two formats
  - An Extensible Markup Language (XML)
  - JavaScript Object Notation (JSON)

Agenda

- Web Technologies
- What is a Web Service?
- Styles of Web Services
- **Why Web Services?**
- Where are Web Services?
- Web Service Architecture and Standards
- Web Service Development

Why Web Services? (1/2)

- Exposing the Existing Function on the network
  - A web service is a unit of managed code that can be remotely invoked using HTTP
- Interoperability
  - Web services allow various applications to talk to each other and share data and services among themselves
  - For example, a VB or .NET application can talk to Java web services and vice versa

Why Web Services? (1/2)

- Standardized Protocol
  - Web services use standardized industry standard protocol for the communication (HTTP, XML, JSON)
- Low Cost Communication
  - Reuse existing APIs
  - No need to develop ourselves
    - Google Maps

Agenda

- The Evolution of Computing
- What are Web Services?
- Styles of Web Services
- Why Web Services?
- **Web Services Architecture and Standards**
- Where are Web Services?
- Web Services Development

Web Services Standards

- XML (Extensible Markup Language)
- SOAP (Simple Object Access Protocol)
- WSDL (Web Services Description Language)
- JSON (JavaScript Object Notation)

Extensible Markup Language

- Text-based Markup Language
- Markup is the extra information for describing and formatting data
- Standard language for exchanging and representing data on the Internet
- Both XML and HTML are markup languages
  - NSC
  - <event>NSC</event>

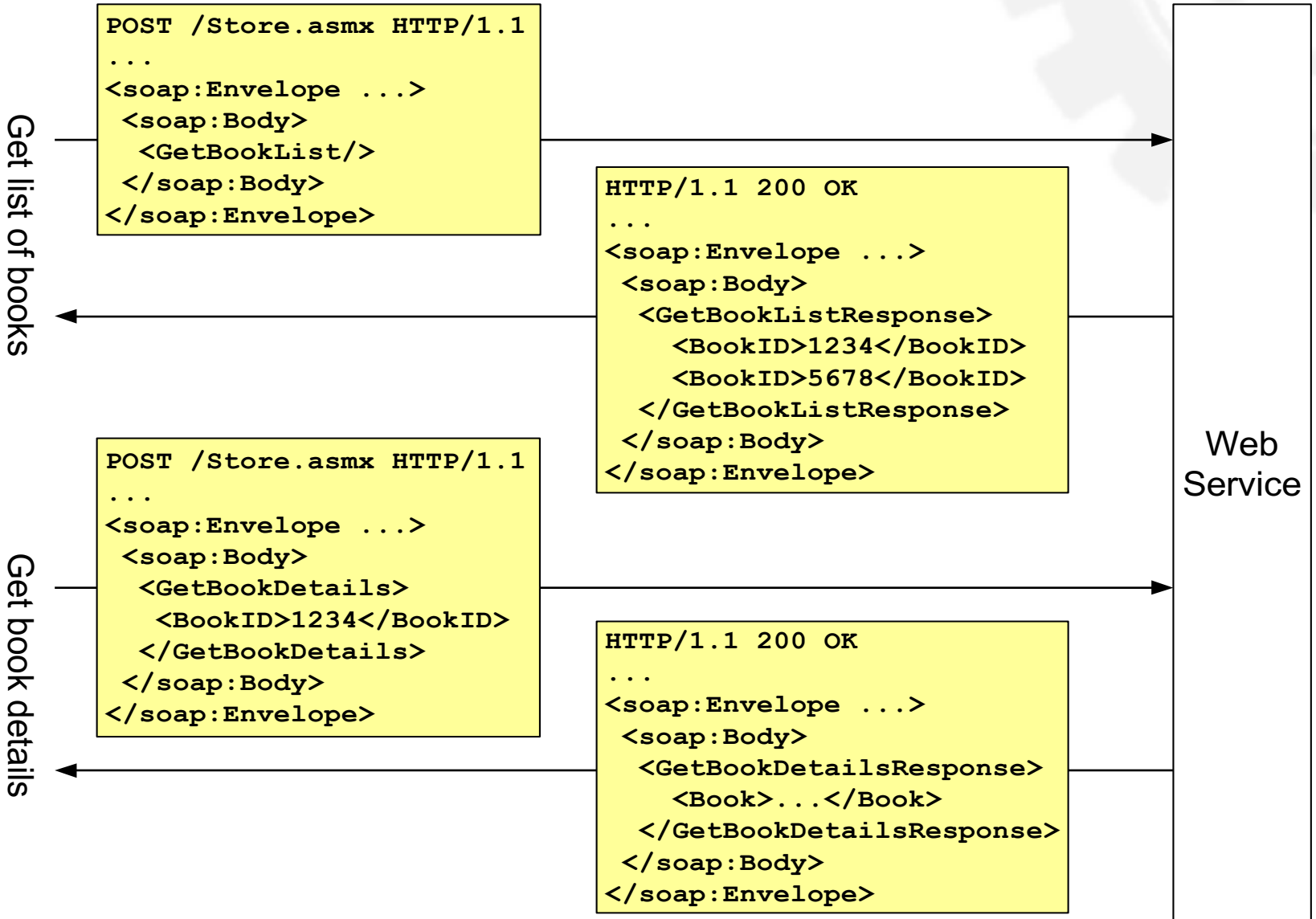
Sample XML Document

```
<?xml version="1.0"?>
<nation>
    <name>Thailand</name>
    <location>Southeast Asia</location>
</nation>
```

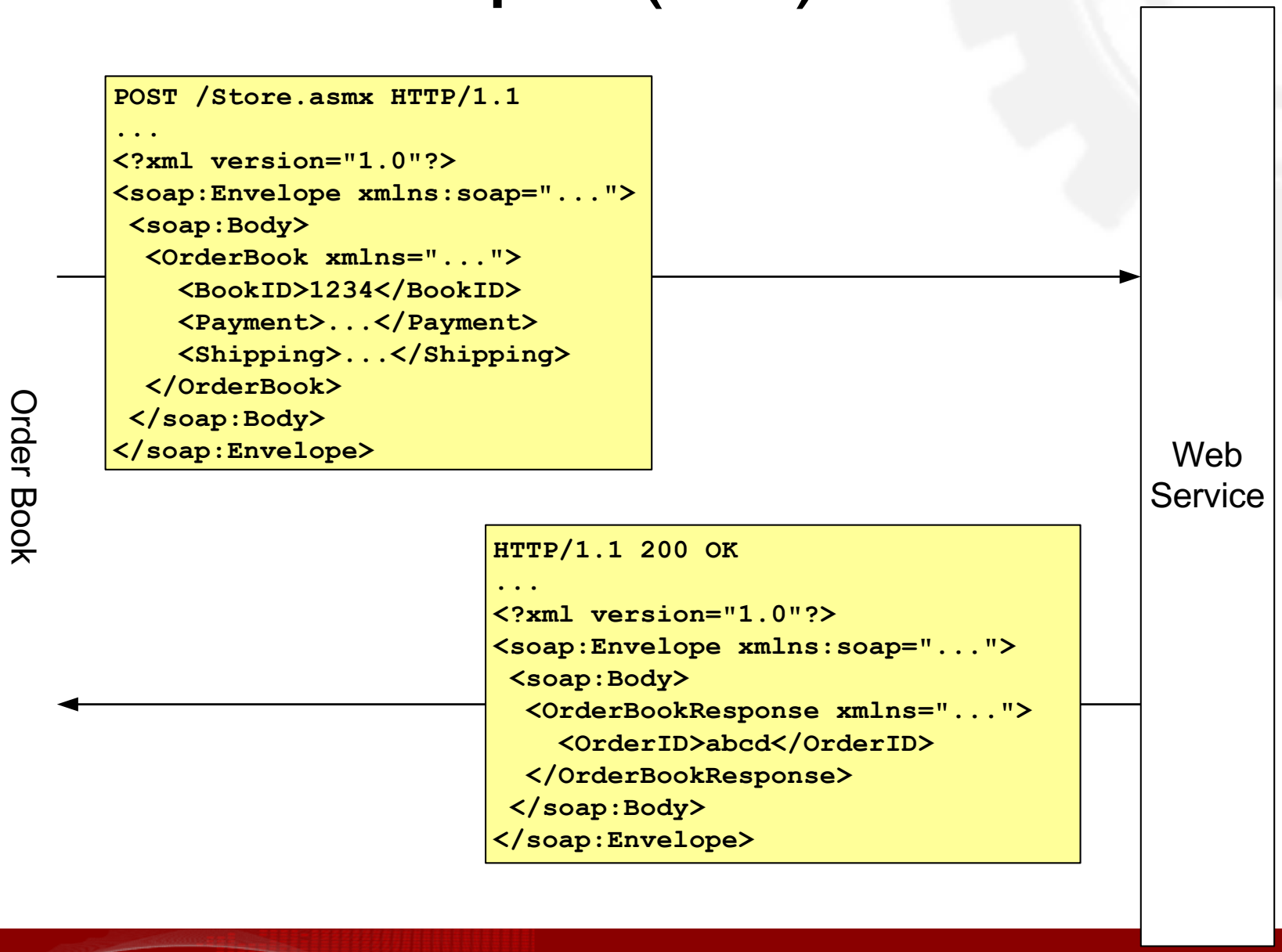
SOAP

- SOAP stands for Simple Object Access Protocol
- SOAP is a lightweight protocol intended for **exchanging structured information**
- SOAP uses XML technologies to define an **extensible messaging framework**
- The framework has been designed to be **independent of any particular programming model** and other implementation specific semantics

SOAP Example (1/2)



SOAP Example (2/2)



WSDL

- WSDL stands for Web Services Description Language
- XML language for describing web services
- XML service is described as
  - A set of communication endpoints (ports)
- Endpoint is made of two parts
  - Abstract definition of operations and messages
  - Concrete binding to networking protocol and message format

Sample WSDL

www.xignite.com/xGlobalQu

www.xignite.com/xGlobalQuotes.asmx?wsdl

Docs Gmail Facebook YouTube Dropbox ScholarOne Manusc บทเรียนของ GotoKnow User-Centered Desig Online College Class

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<wsdl:definitions xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
 xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/" xmlns:tns="http://www.xignite.com/services/" xmlns:soap="http://schemas.xmlsoap.
 xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/" xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
 xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://www
  ▼<wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
     This web service provides global delayed stock quotes and for U.S. and international equities.
   </wsdl:documentation>
  ▼<wsdl:types>
    ▼<s:schema elementFormDefault="qualified" targetNamespace="http://www.xignite.com/services/">
      ▼<s:element name="GetGlobalDelayedQuote">
        ▼<s:complexType>
          ▼<s:sequence>
             <s:element minOccurs="0" maxOccurs="1" name="Identifier" type="s:string"/>
             <s:element minOccurs="1" maxOccurs="1" name="IdentifierType" type="tns:IdentifierTypes"/>
           </s:sequence>
         </s:complexType>
       </s:element>
      ▼<s:simpleType name="IdentifierTypes">
        ▼<s:restriction base="s:string">
           <s:enumeration value="Symbol"/>
           <s:enumeration value="CIK"/>
           <s:enumeration value="CUSIP"/>
           <s:enumeration value="ISIN"/>
           <s:enumeration value="Valoren"/>
           <s:enumeration value="SEDOL"/>
         </s:restriction>
       </s:simpleType>
      ▼<s:element name="GetGlobalDelayedQuoteResponse">
        ▼<s:complexType>
          ▼<s:sequence>
             <s:element minOccurs="0" maxOccurs="1" name="GetGlobalDelayedQuoteResult" type="tns:GlobalQuote"/>
           </s:sequence>
         </s:complexType>
       </s:element>
```

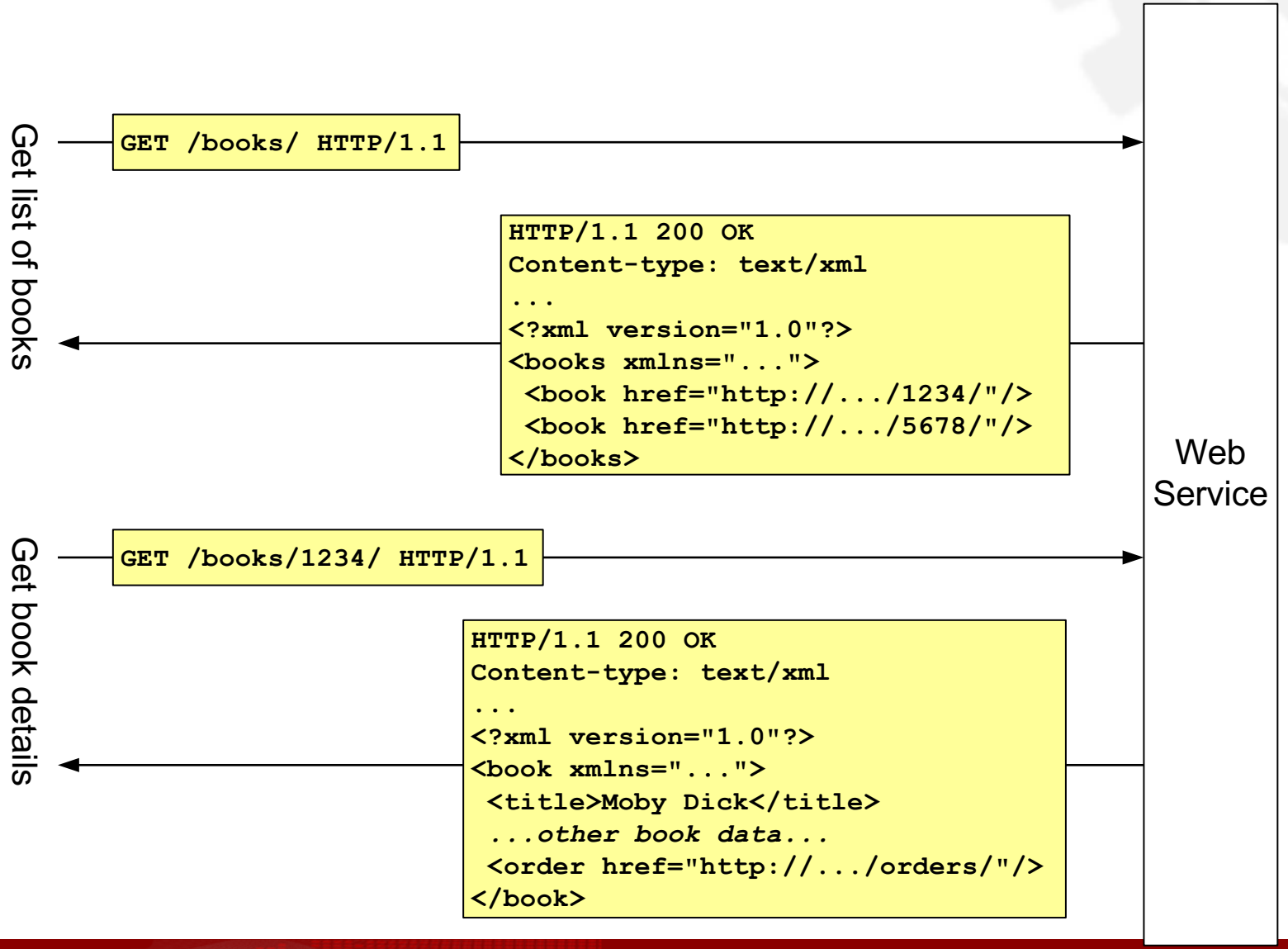
What is REST? (1/2)

- **Representational State Transfer** (**REST**) is a style of software architecture for distributed hypermedia systems such as the World Wide Web
- The term was introduced in the doctoral dissertation of Roy Fielding in 2000, one of the principal authors of the Hypertext Transfer Protocol (HTTP) specification

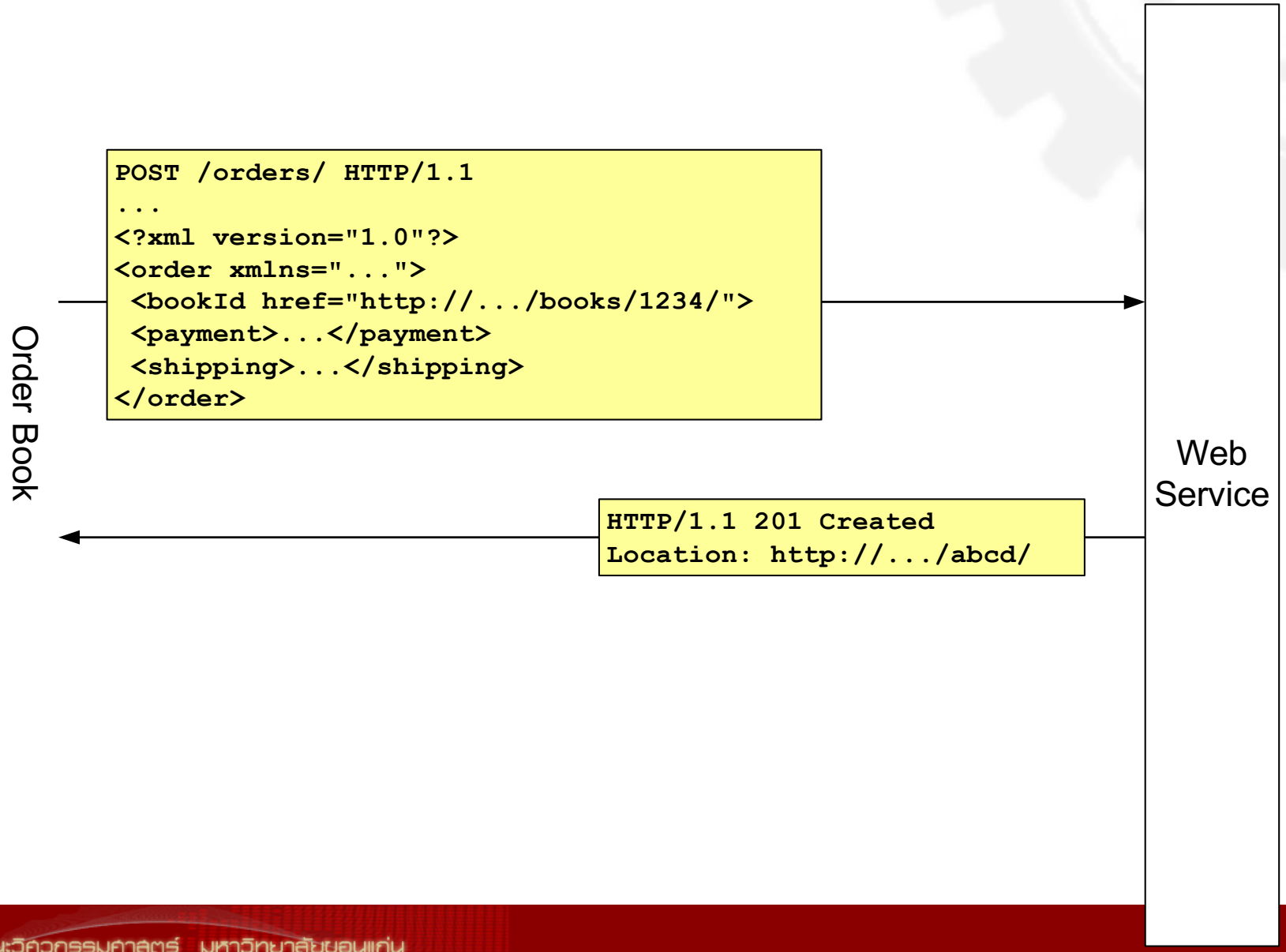
What is REST? (2/2)

- REST strictly refers to a collection of network architecture principles that outline how resources are defined and addressed
- The term is often used in a looser sense to describe any simple interface that transmits domain-specific data over HTTP without an additional messaging layer such as SOAP or session tracking via HTTP cookies.

REST (Using XML) Example



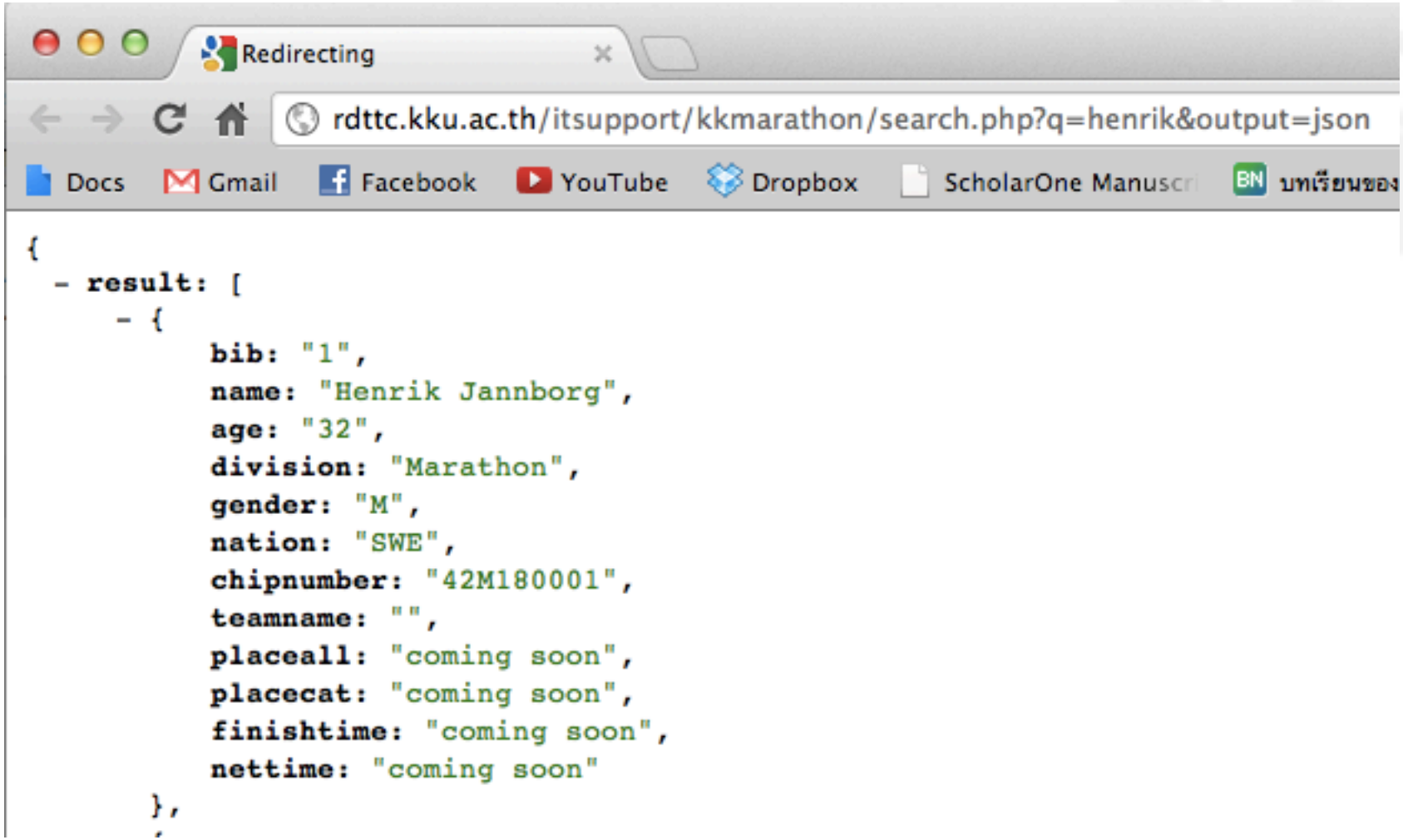
REST (Using XML) Example



JSON

- JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write
- It is easy for machines to parse and generate.
- JSON is a text format that is completely language independent
- JSON is built on two structures:
  - A collection of name/value pairs
  - An ordered list of values

REST (Using JSON) Example

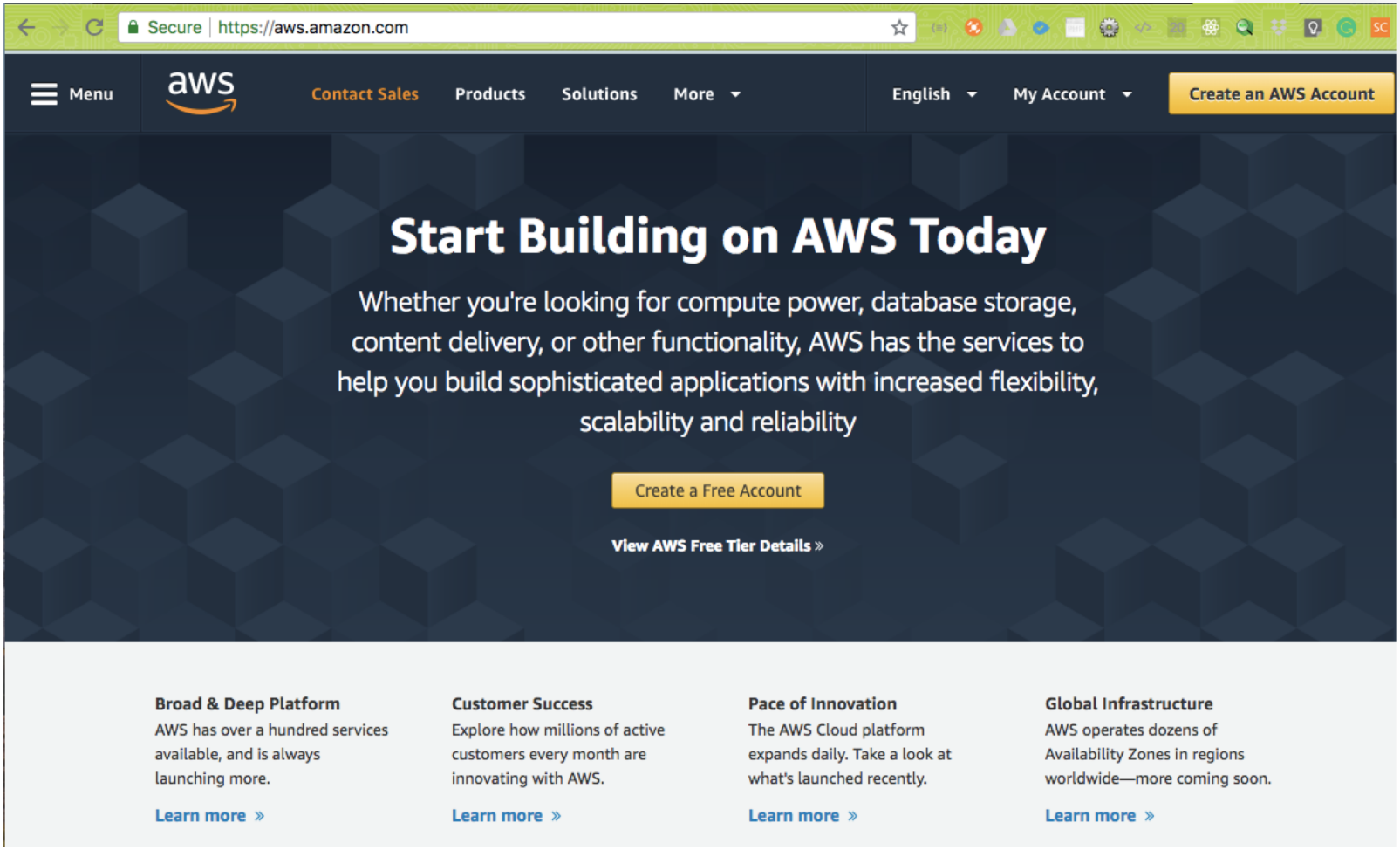


Agenda

- The Evolution of Computing
- What are Web Services?
- Why Web Services?
- Web Services Architecture and Standards
- Where are Web Services?
- Web Services Development

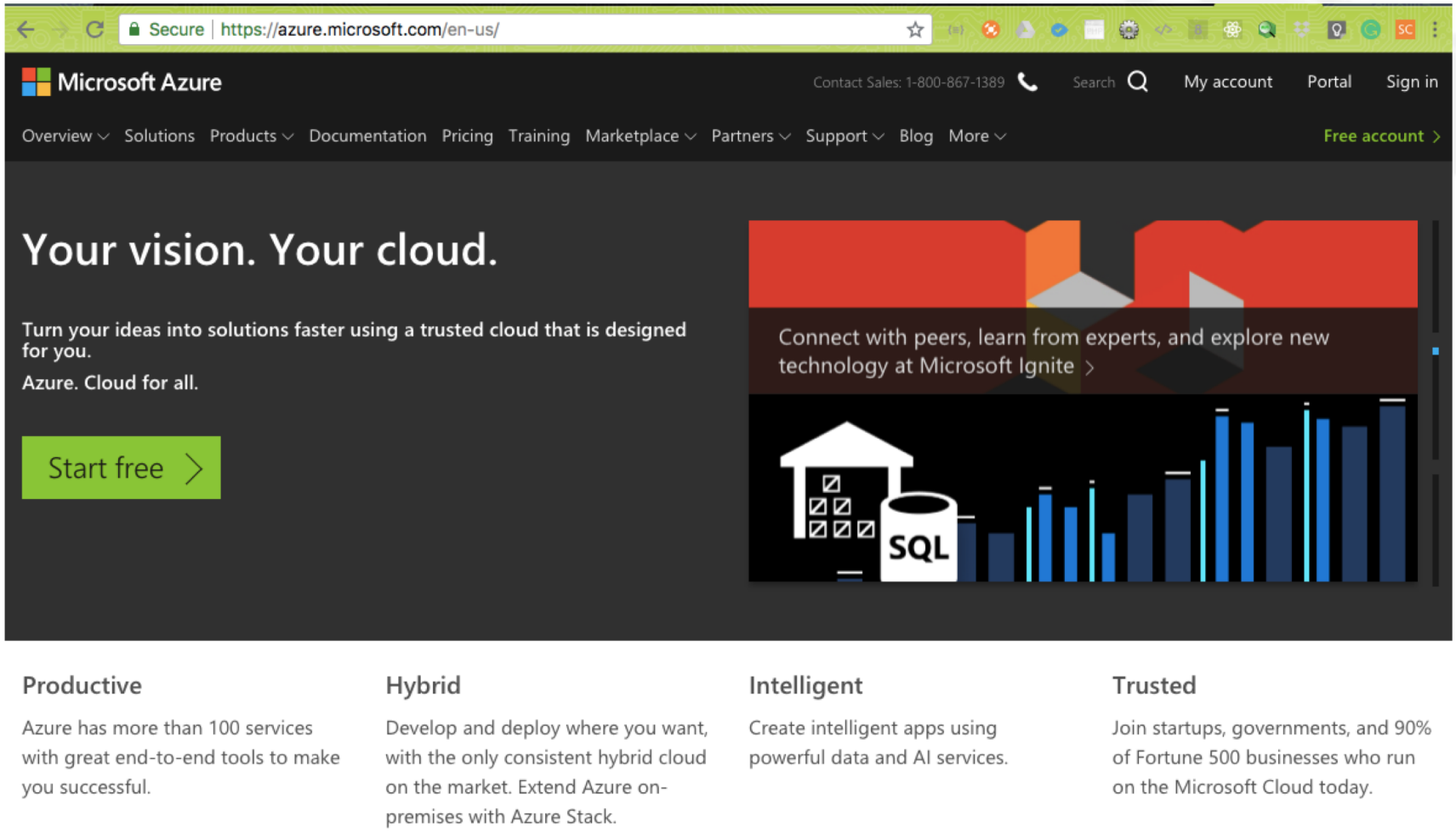
Amazon Web Services

http://aws.amazon.com



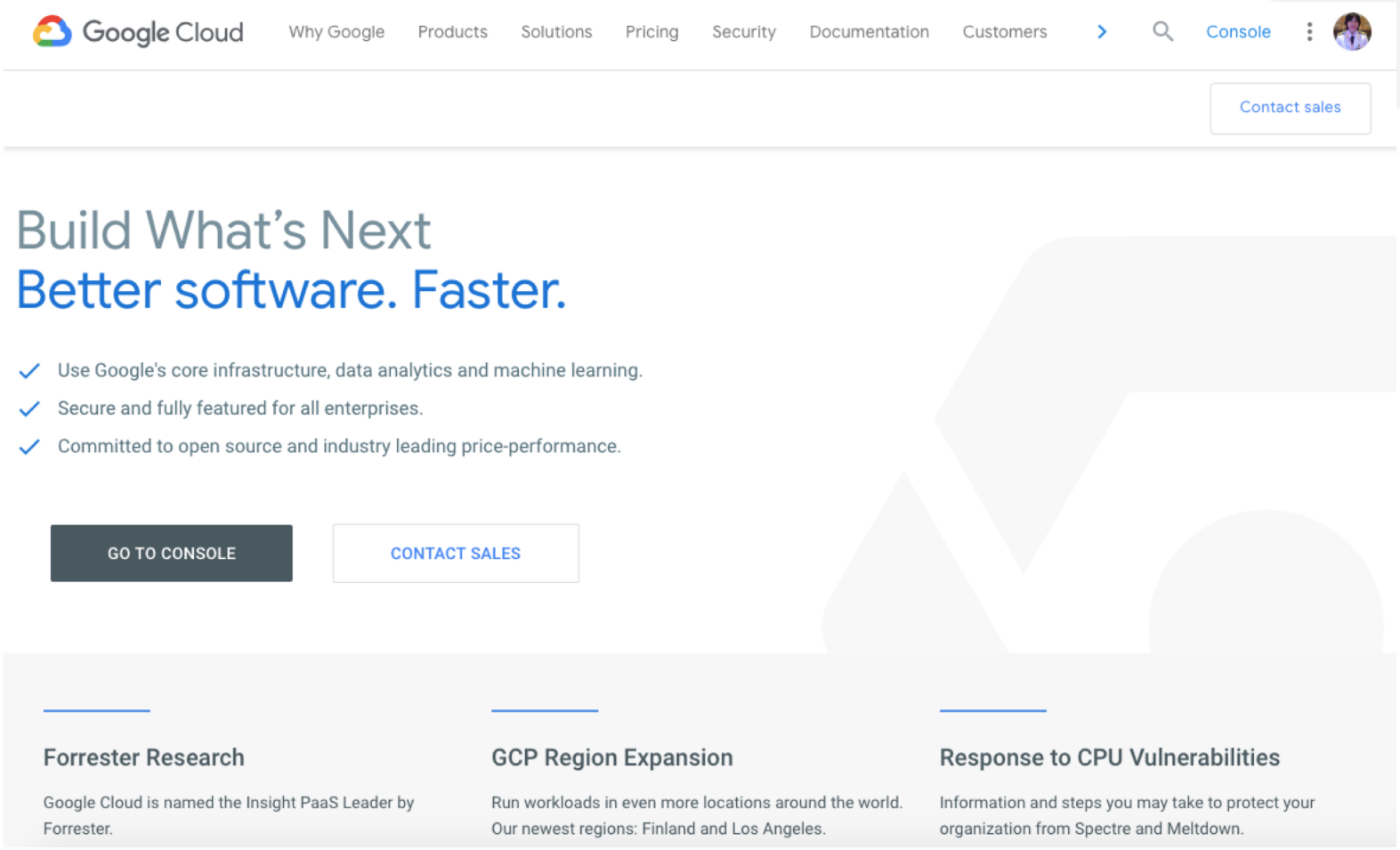
Microsoft Azure

https://azure.microsoft.com/en-us/

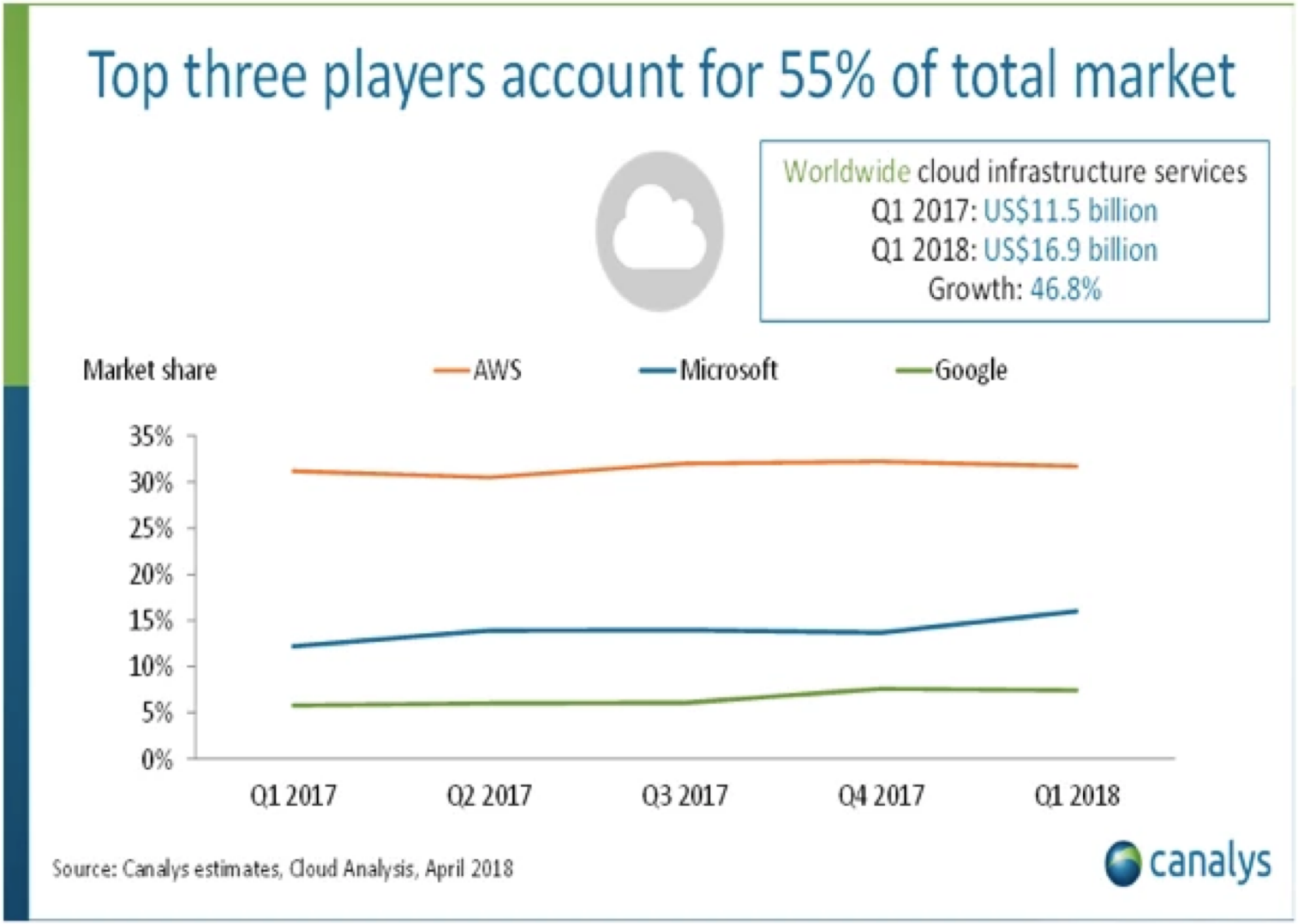


Google Developers

https://cloud.google.com/



Top Three Players



Search for Web APIs and Mashups

Secure | https://www.programmableweb.com

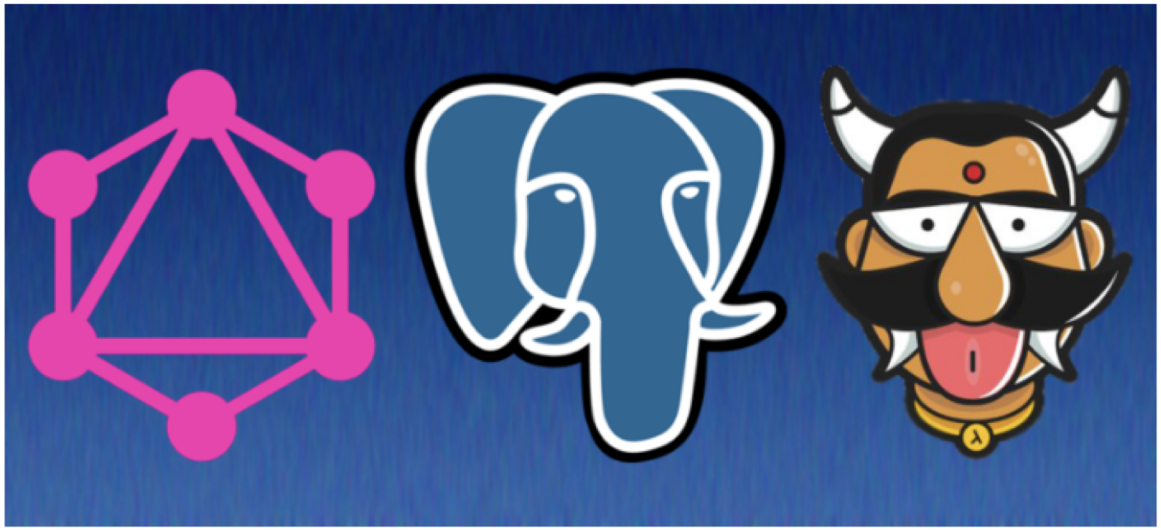
WRITE FOR US | BECOME MEMBER | LOGIN

ProgrammableWeb

API DIRECTORY API NEWS

Search over 19,925 APIs and much more

LEARN ABOUT APIS | WHAT IS AN API ? | TUTORIALS | ADD APIs & MORE



Add GraphQL-as-a-Service to Postgres-Based Apps With Hasura's Open Source GraphQL Engine

Today in APIs

Latest news about the API economy and newest APIs, delivered daily:

Search for Web APIs and Mashups

Secure | https://www.programmableweb.com

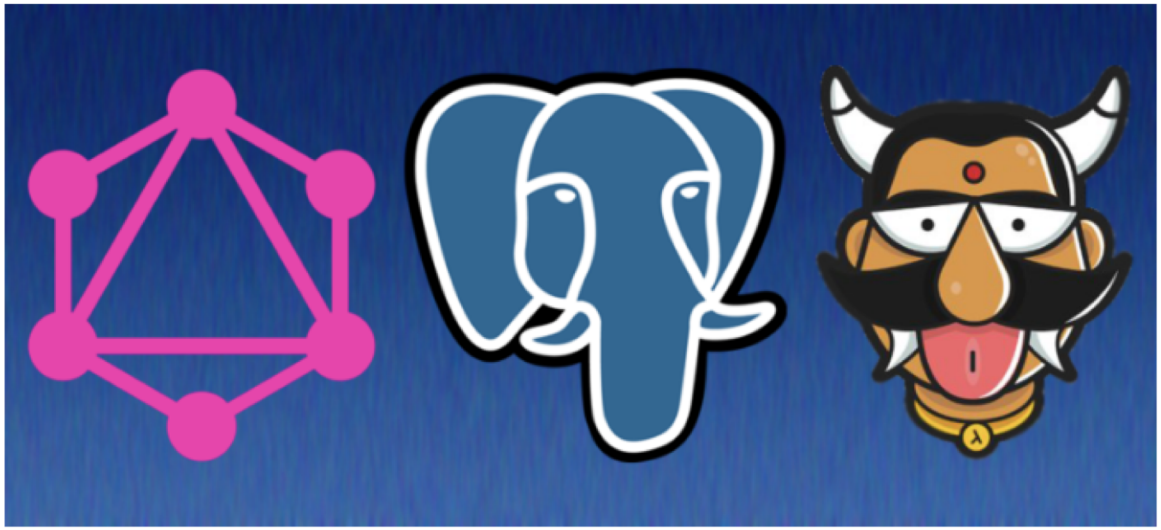
WRITE FOR US | BECOME MEMBER | LOGIN

ProgrammableWeb

API DIRECTORY API NEWS

Search over 19,925 APIs and much more

LEARN ABOUT APIS | WHAT IS AN API ? | TUTORIALS | ADD APIs & MORE



Add GraphQL-as-a-Service to Postgres-Based Apps With Hasura's Open Source GraphQL Engine

Today in APIs

Latest news about the API economy and newest APIs, delivered daily:

Most Clicked Throughs APIs in 2017

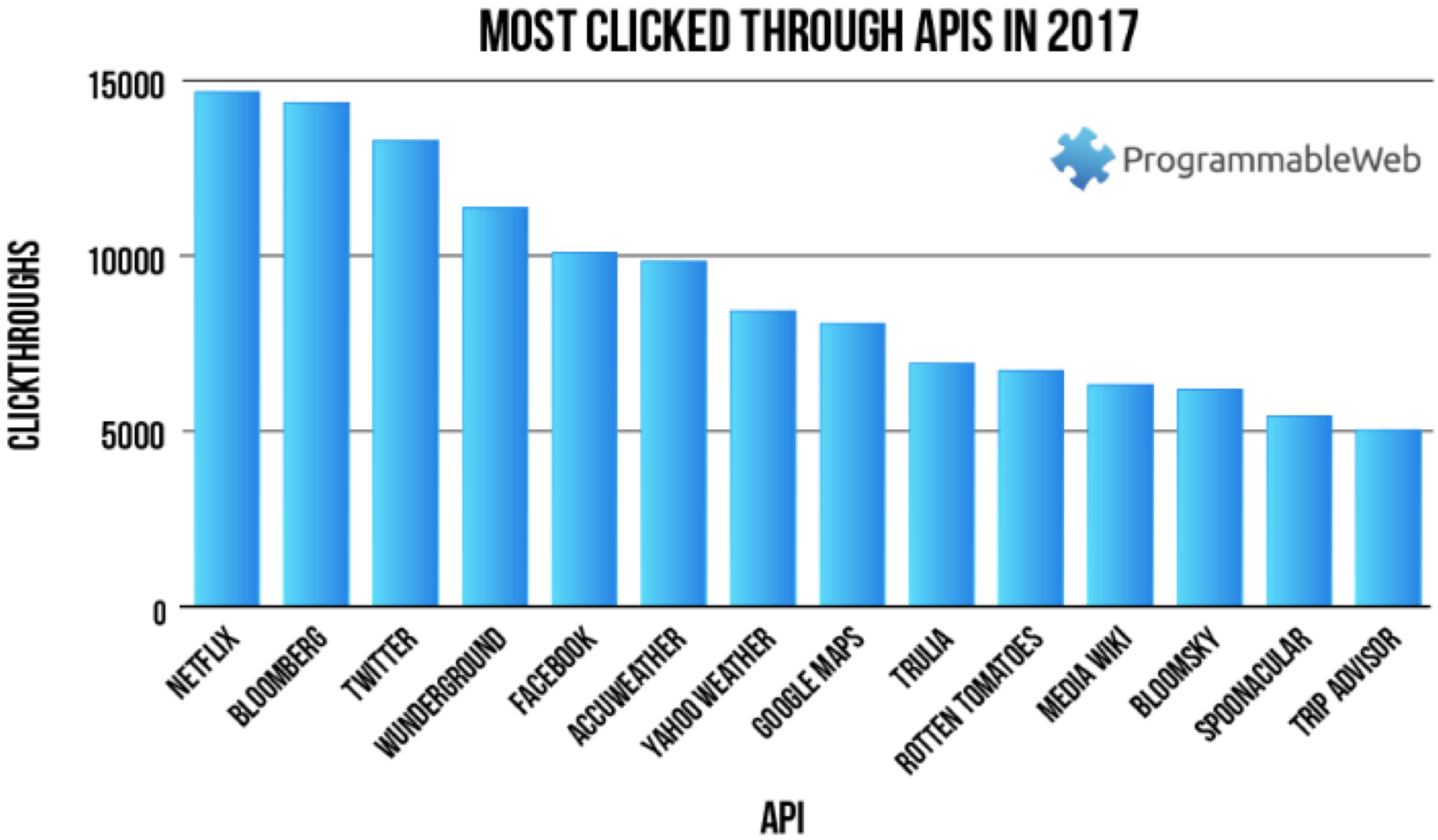
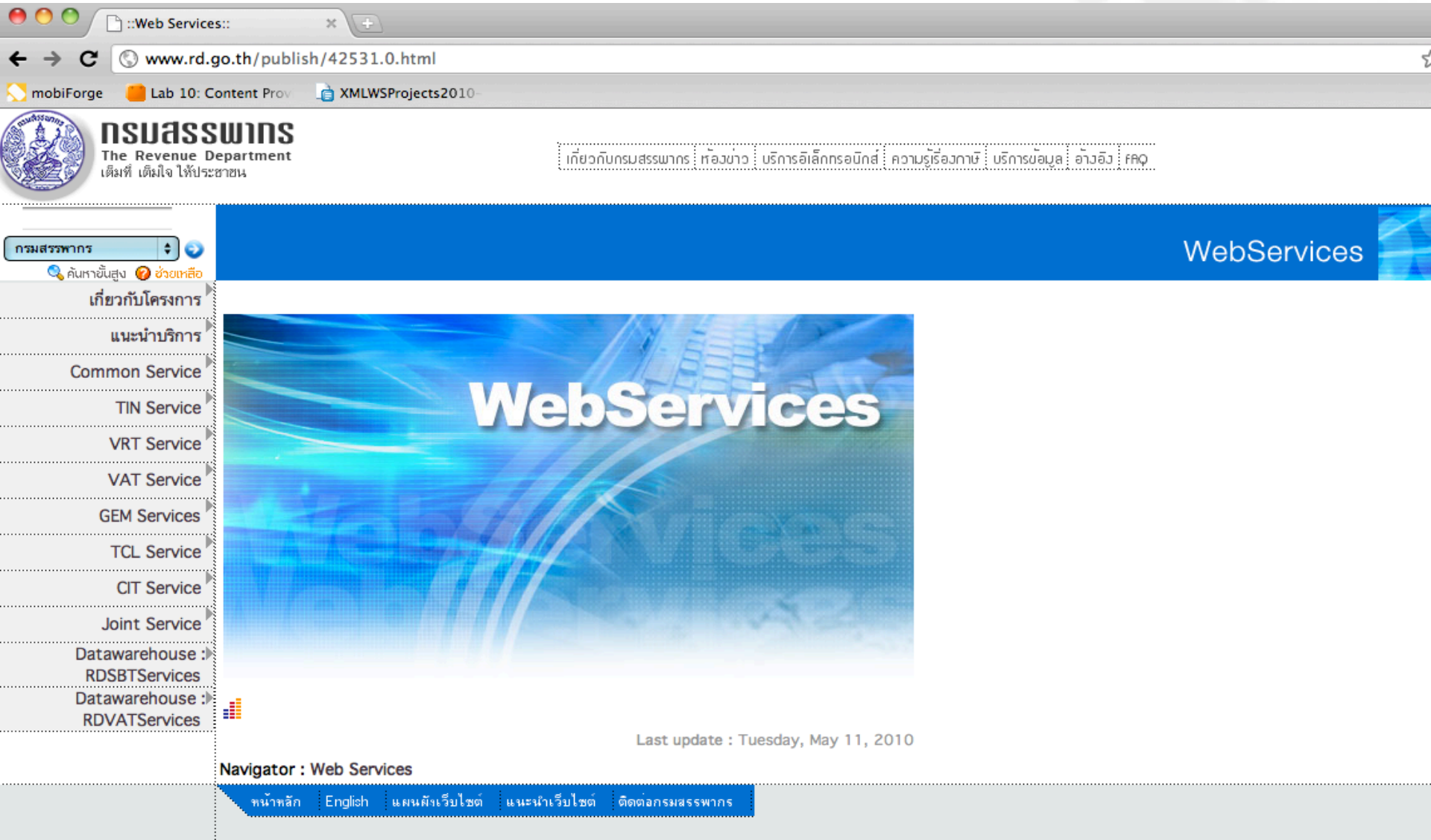


Figure 1: APIs that drove the most click throughs from ProgrammableWeb API profiles in 2017

e-Revenue Web Services (1/2)

- http://www.rd.go.th/publish/42531.0.html

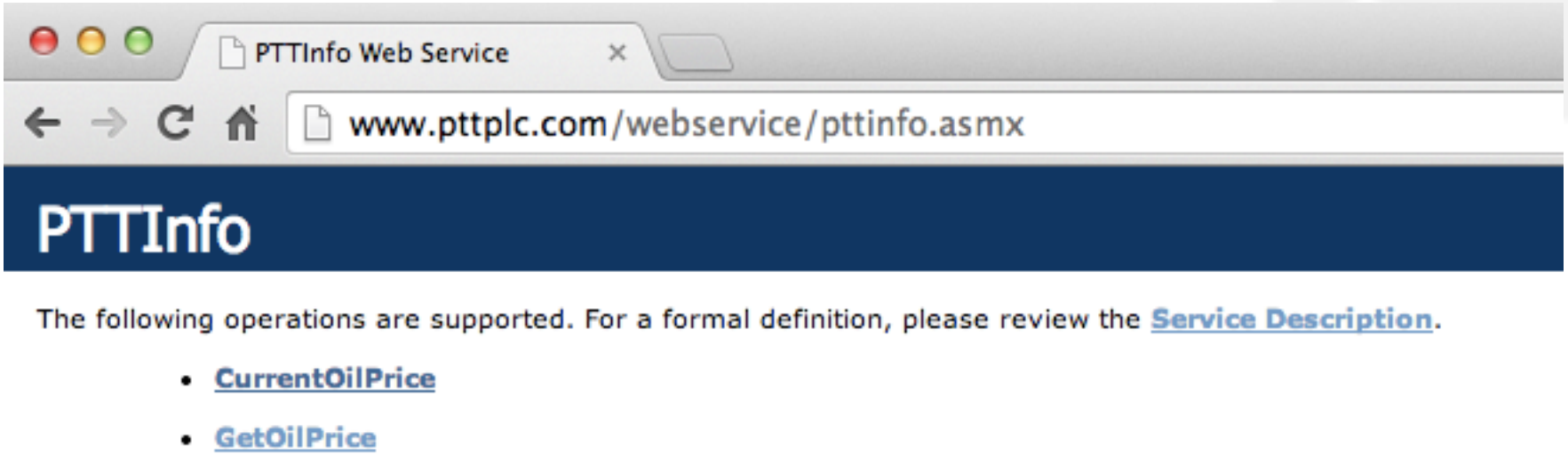


e-Revenue Web Services (2/2)

- Create service oriented organization by providing professional services to citizens to improve efficiency and fairness in tax collection
- Serve as a catalyst in driving e-services / e-commerce take up via business partnership
- Sample services: PIN/TIN Verification Info, VAT Refund for Tourist info

PTT Information Web Services (1/2)

http://www.pttplc.com/webservice/pttinfo.asmx



PTT Information Web Services (2/2)

- The users can get the information about oil price and news related to oil
- Support these five operations
  - GetOilPrice
  - CurrentOilPrice

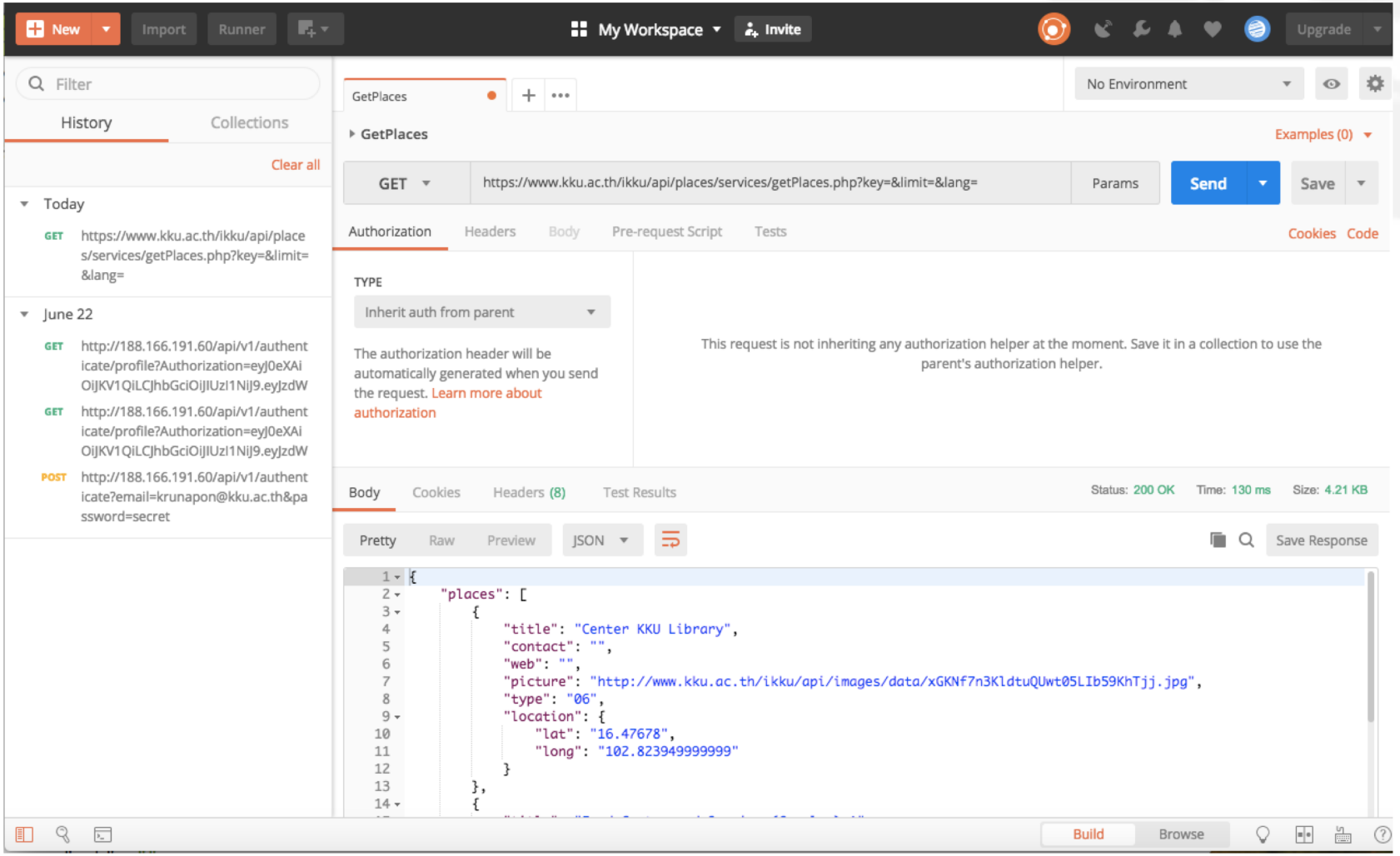
Agenda

- The Evolution of Computing
- What is a Web Service?
- Why Web Services?
- Web Services Architecture and Standards
- Where are Web Services?
- **Web Services Development**

Postman

- Postman is a powerful HTTP client for testing web services
- Created by Abhinav Asthana, a programmer and designer based in Bangalore, India
- Postman makes it easy to test, develop and document APIs by allowing users to quickly put together both simple and complex HTTP requests

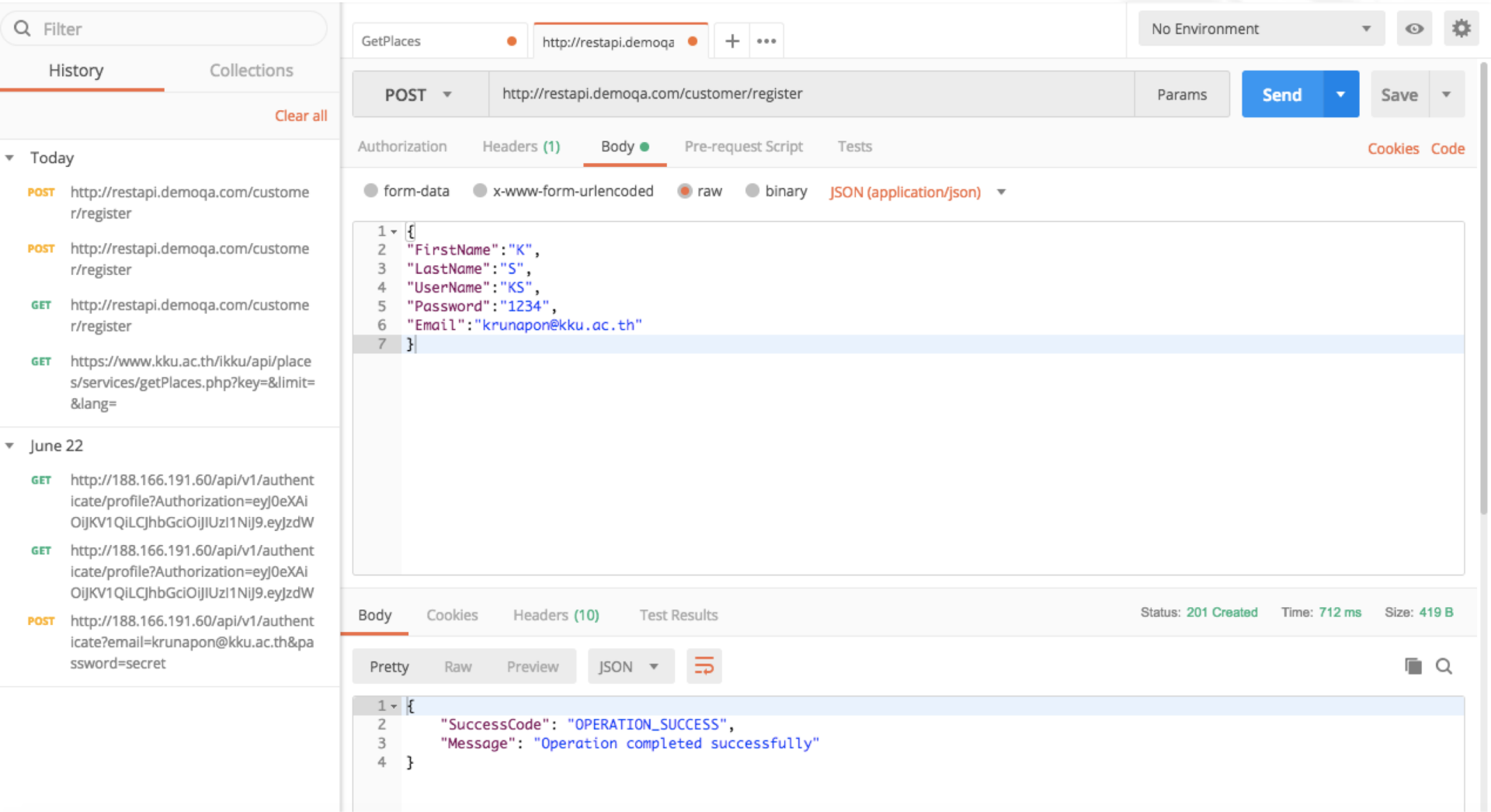
Postman HTTP Get Usage Example



What is a POST Request?

- A POST is an HTTP Verb similar to a GET request, this specifies that a client is posting data on the given Endpoint
- A POST request is a method which is used when we need to send some additional information inside the body of the request to the server
- One of the classic example of a POST request is the Login page

Postman HTTP Post Usage Example



What is Authorization?

- The meaning of authorization can be seen as a question which is, are we eligible to access a secured resource on the Server?
- If the answer is yes, then in technical terms we can say that we are Authorized to access the resource
- Similarly while there could be many ***APIs*** in a company or a project. It is not necessary that everyone will have access on all the APIs
- Only authorized people can access the secured APIs.

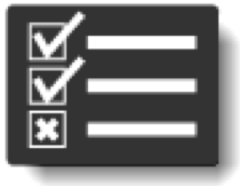
Authentication

- Authentication is a process of presenting your credentials to the system and the system validating your credentials
- These credentials tell the system about who you are which enables the system to ensures and confirms a user's identity
- Authentication tells who you are while Authorization tells what you can do.

Authentication vs Authorization

Authentication

Who you are

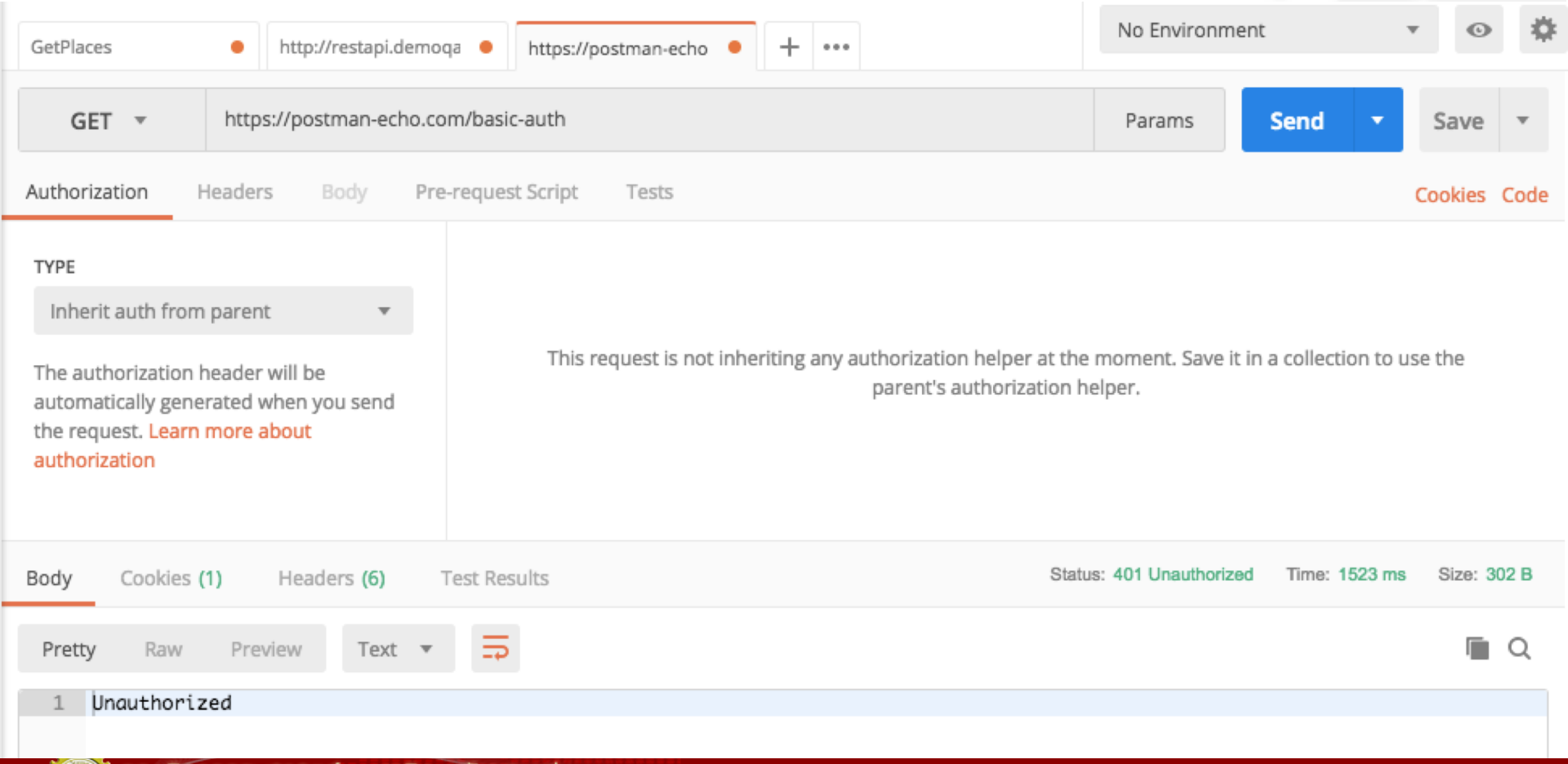


Authorization

What you can do

Authorization using Postman

- Create a *GET* request and enter the endpoint as ***https://postman-echo.com/basic-auth***



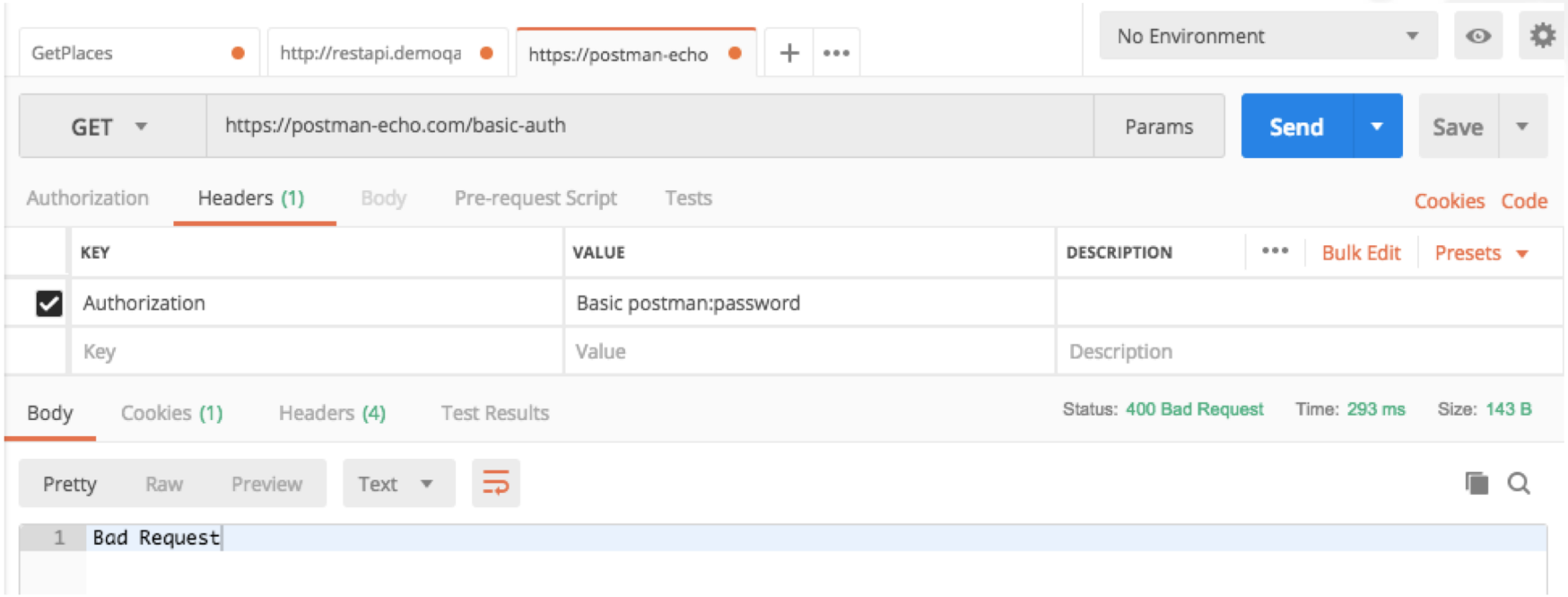
HTTP Basic Authentication

- A Basic Access Authentication is the most simple and basic type of authorization available
- It requires just a username and password for checking the authorization of any person (That is why we say basic access authentication)
- The username and password are sent as header values in the Authorization header

400, Bad Request

If we use these key value pairs in Header

Authorization : Basic postman:password

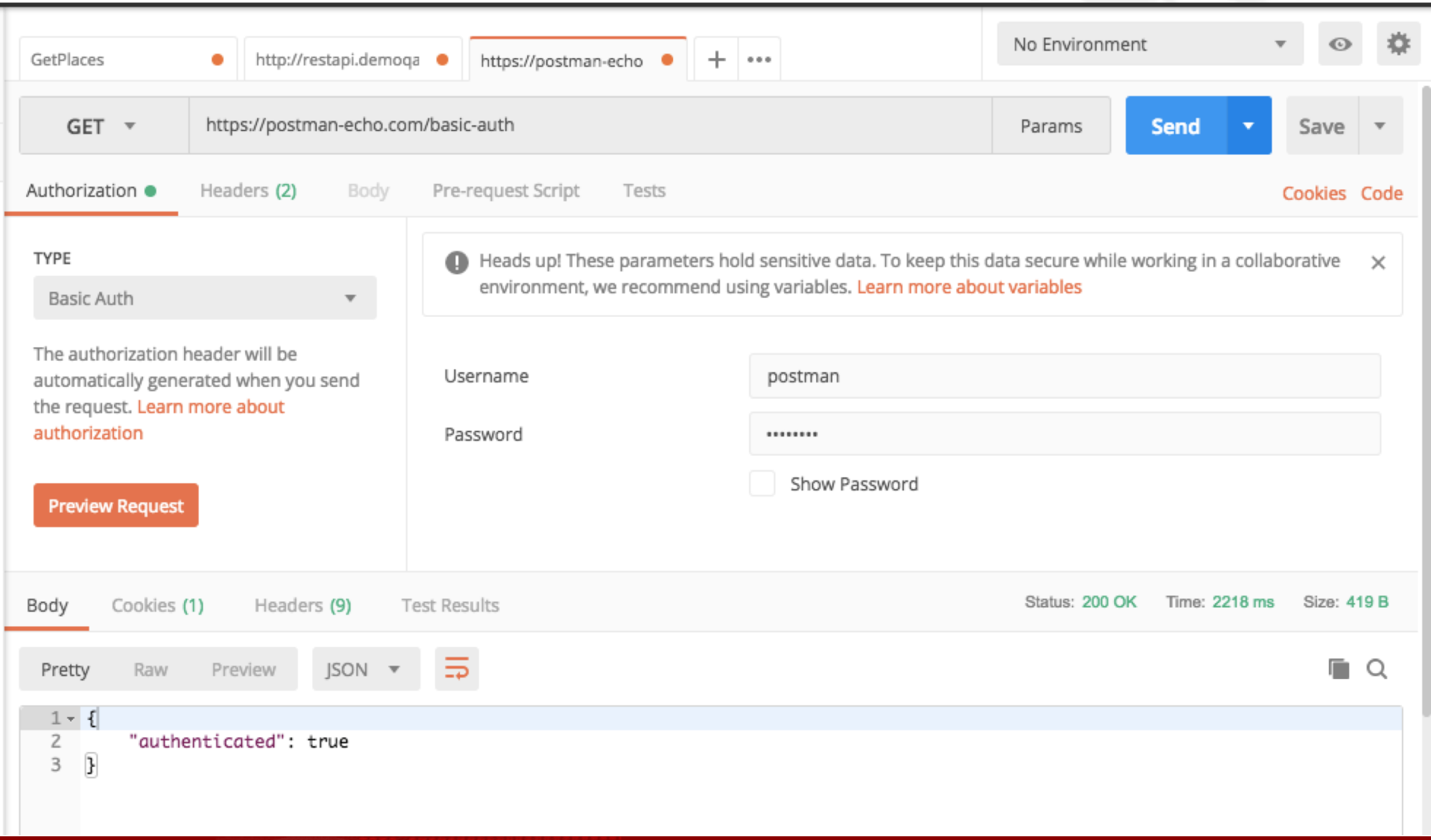


A basic access authentication requires username and password to be encoded in Base64

What is Base64 encoding?

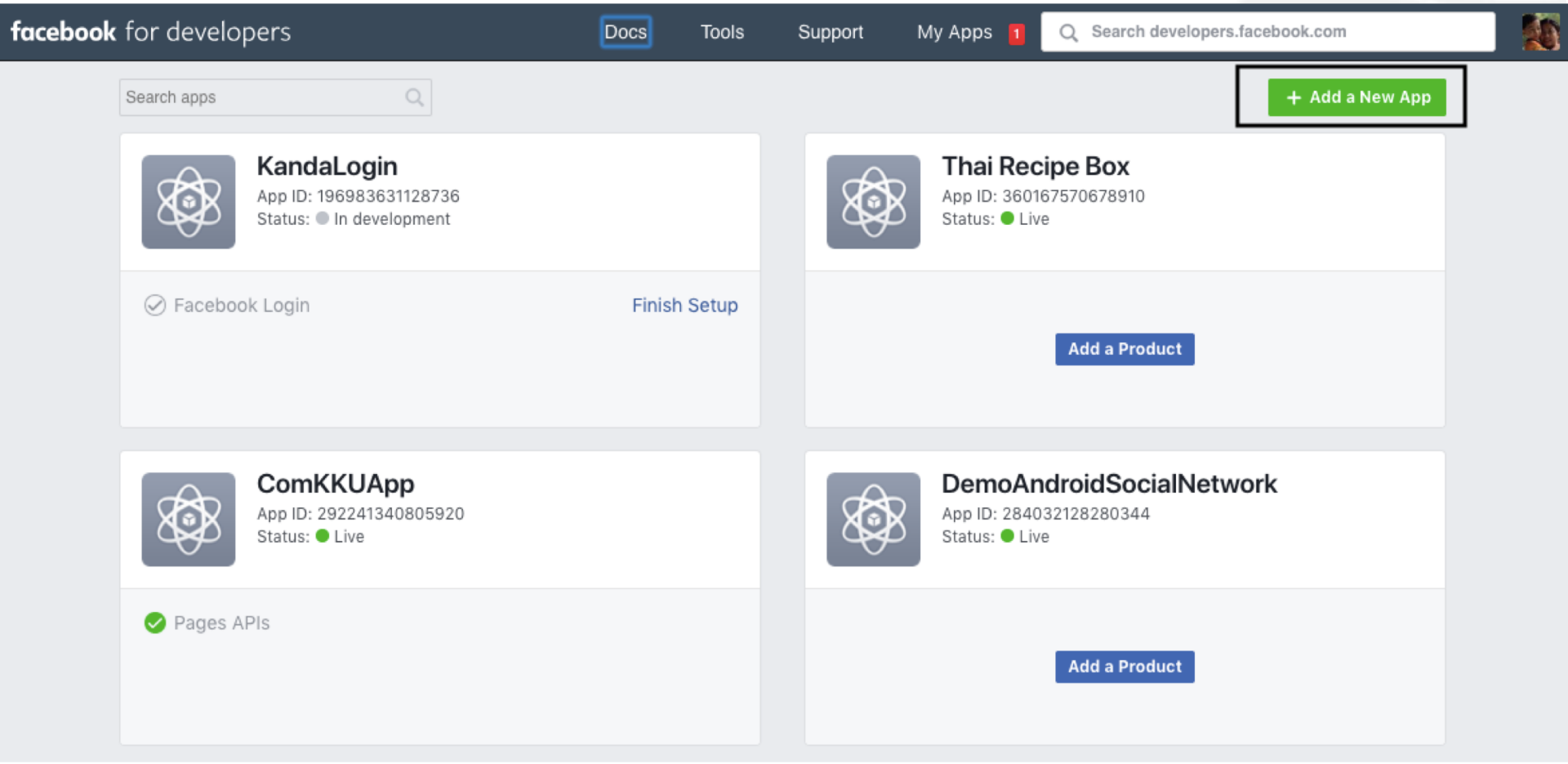
- An encoding is used in the authentication because we don't want our data to be transmitted directly over the network
- There are a numerous reasons for that. Network scanners can read your Request and retrieve the Username and Password sent without encoding
- We use **base64** particularly because it transmits the data into textual form and send it in easier form such as HTML form data

Using Type: Basic Auth



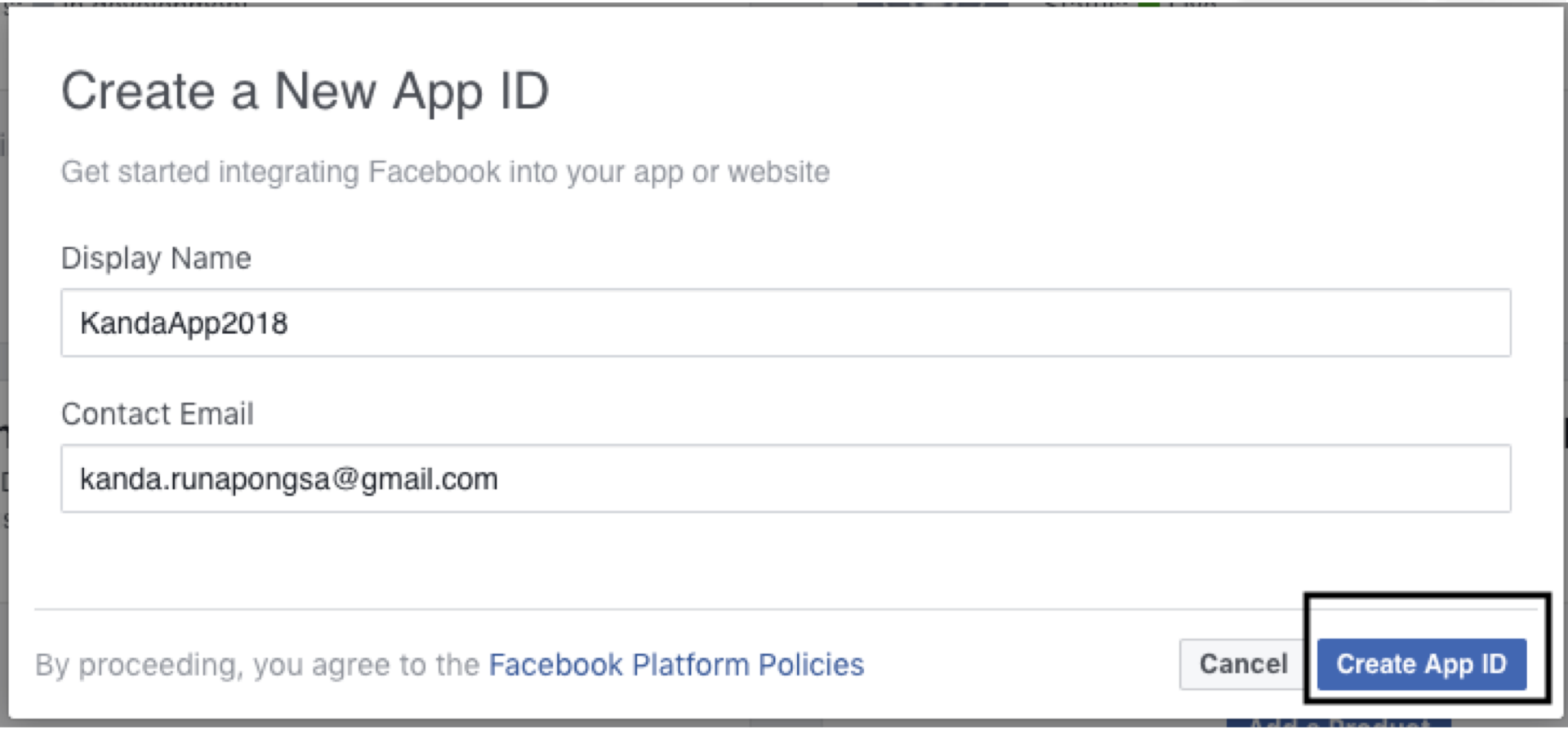
How to Generate Facebook Token (1/4)

1. Visit https://developers.facebook.com/apps



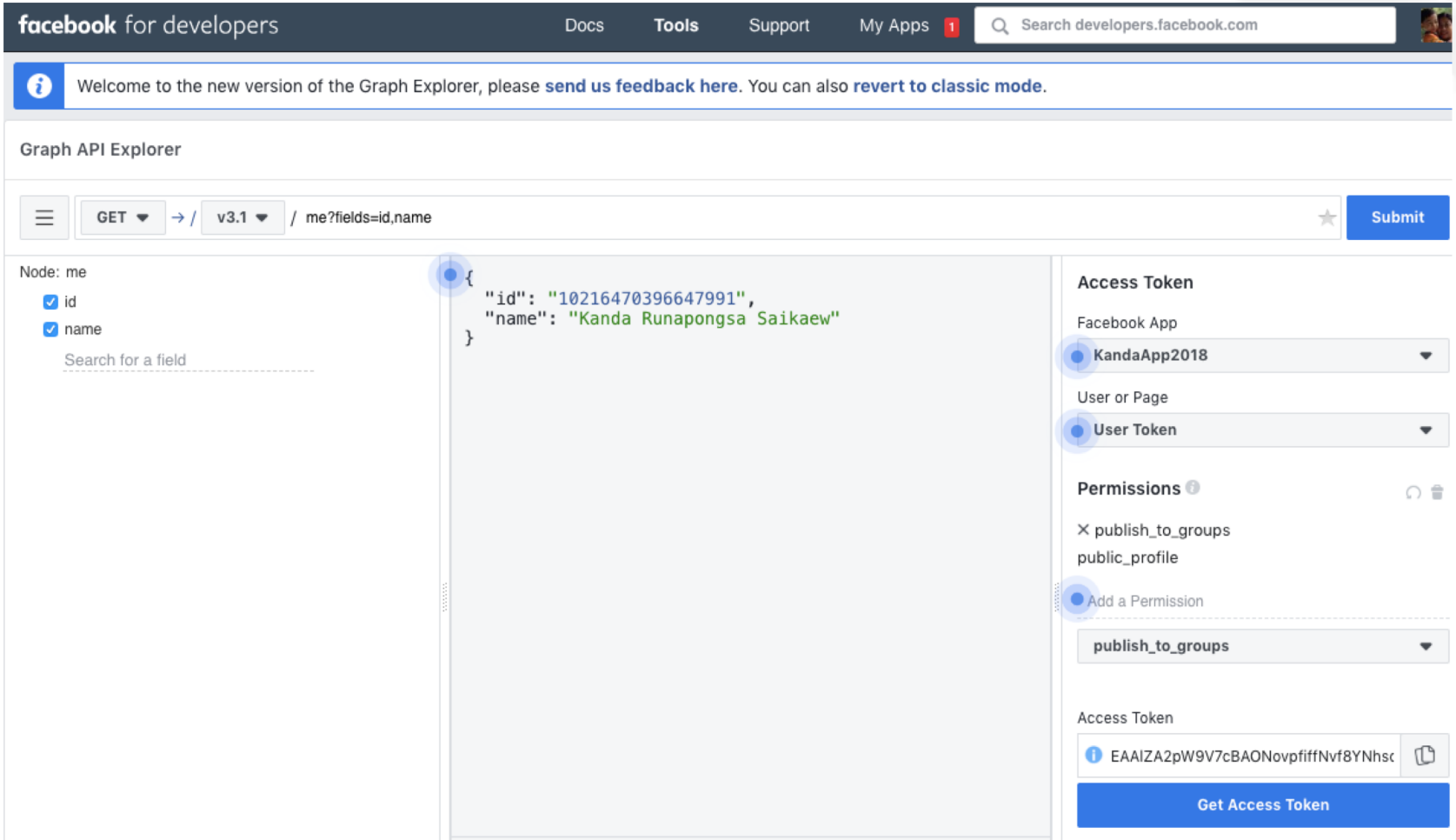
How to Generate Facebook Token (2/4)

2. Create A Facebook App

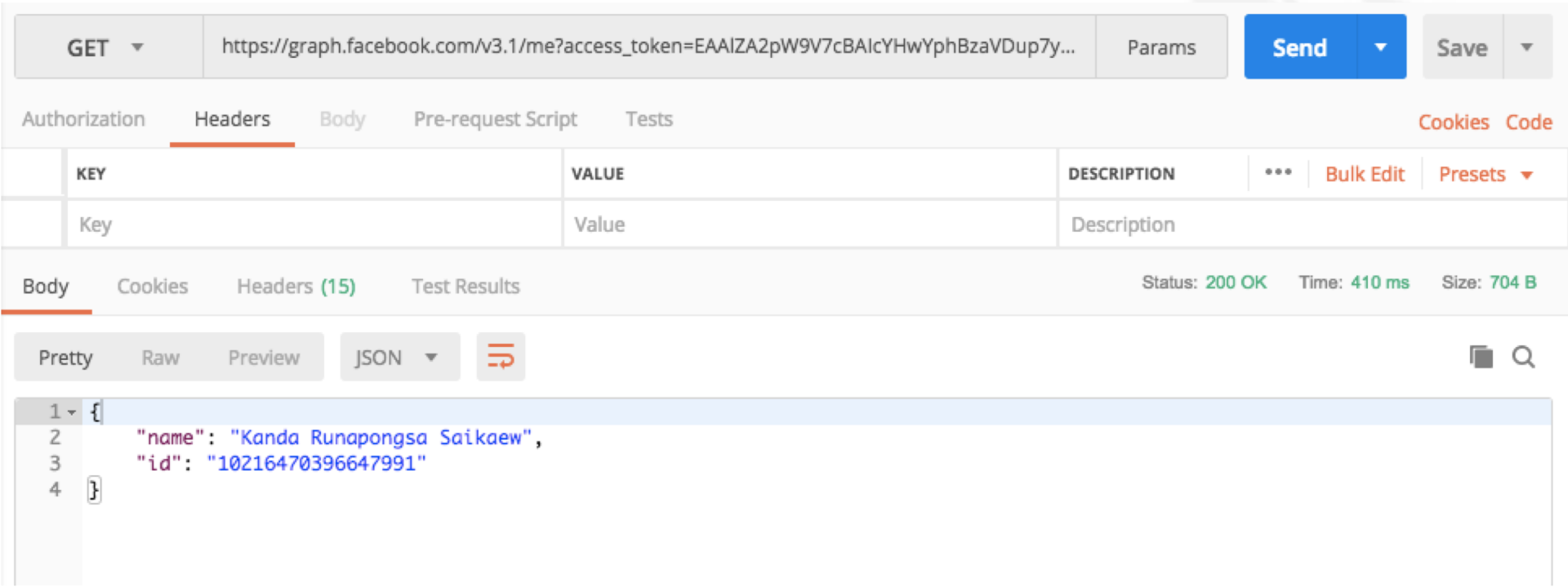


How to Generate Facebook Token (3/4)

3. Visit https://developers.facebook.com/tools/explorer



Call Graph Facebook API using Postman



HTTPie

- HTTPie (pronounced aitch-tee-tee-pie) is a command line HTTP client. Its goal is to make CLI interaction with web services as human-friendly as possible
- It provides a simple http command that allows for sending arbitrary HTTP requests using a simple and natural syntax, and displays colorized output
- HTTPie can be used for testing, debugging, and generally interacting with HTTP servers.

HTTPie and cURL

- cURL is a commonly used command-line tool for making HTTP requests
- HTTPie is a command-line tool for making HTTP requests in a human-friendly way
- HTTPie uses commands with simple and natural syntax and displays the output in a presentable manner.

Installing and Using HTTPie

- Install on Mac, using command
  - brew install httpie
- Sample httpie usage

```
macair:~ krunapon$ http https://graph.facebook.com/v3.1/me?access_token=EAAlZA2p
W9V7cBAIcYHwYphBzaVDup7ya3hEZBGLmZBx9ltkRb2LpH89CpySXT1fg7M7nK8P4u2uZCTZCAbZAzjZ
Cnivpmyxzwq6rLRAbyVoZAviROmkueVC5oNmb5oPxst4aBf1ZBr004fA9LjaT6nVcnCVe9rNwusOcHBZ
AlU3PlI7Xzu7xH7bX5oa76VqGNKeALdglvGBgzTvB3SSJ5zuN8CckAZApGjNZBQbYmNdFiqRyuAMka3a
DRJ1f
HTTP/1.1 200 OK
Access-Control-Allow-Origin: *
Cache-Control: private, no-cache, no-store, must-revalidate
Connection: keep-alive
Content-Length: 60
Content-Type: application/json; charset=UTF-8
Date: Wed, 15 Aug 2018 05:20:56 GMT
ETag: "787a15f5131da67303c575a995a1624c2ac50fb0"
Expires: Sat, 01 Jan 2000 00:00:00 GMT
Pragma: no-cache
Strict-Transport-Security: max-age=15552000; preload
X-FB-Debug: 79MXeBZ03zJ74kaMOZfQJbCV6v1GdOmyQY5VjDqEZyuTAOWu3XLAicsTtyI6Ux683Nns
PfeQCizwpVKoNWb3wg==
facebook-api-version: v3.1
x-app-usage: {"call_count":4,"total_cputime":0,"total_time":0}
x-fb-rev: 4209571
x-fb-trace-id: FasTXD7vl5q

{
    "id": "10216470396647991",
    "name": "Kanda Runapongsa Saikaew"
}
```

Summary

- Web services technology exists for making different systems seamlessly work together
  - XML and JSON are the languages of exchange data
- Web services have been developed and used extensively in many countries
- Thailand should develop and employ Web services technology more for the benefits of sharing and exchanging data as well as increasing the number of mobile apps

References (1/2)

- https://en.wikipedia.org/wiki/Cloud_computing
- https://en.wikipedia.org/wiki/Mobile_backend_as_a_service
- https://www.slideshare.net/onem2m/onem2m-how-standardization-enables-the-next-internet-evolution/3
- https://catmousavi.wordpress.com/2011/11/10/what-is-web-services-according-to-w3c-big-web-services-vs-restful-services/
- http://theopentutorials.com/post/uncategorized/types-of-web-services-big-and-restful/
- http://en.wikipedia.org/wiki/Representational_State_Transfer
- https://stormpath.com/blog/rest-vs-soap

References (2/2)

- https://www.tutorialspoint.com/webservices/why_web_services.htm
- https://www.parkmycloud.com/blog/aws-vs-azure-vs-google-cloud-market-share/
- http://toolsqa.com/postman/basic-authentication-in-postman/
- https://weblizar.com/blog/generate-facebook-access-token/