

Welcome to The ABCs of APIs

Presented by:
Lori Guillory & Allison Ellington



Overview

This presentation will:

- Explain what APIs are and their benefits
- Show how APIs are growing and how that growth could benefit you
- List the information that good API documentation should provide
- Show API examples



What is an API?

- An **Application Programming Interface** (API) is a protocol for allowing one software program to communicate with another.
- APIs let the programs share information and influence each others' behavior.
- This presentation addresses web service APIs, which involve programs that communicate across the internet.



APIs = Access to data

Data is at the center of APIs. APIs provide access to data, both public and private.

90% of the data in the world today has been created in the last two years alone.

IBM, 2012

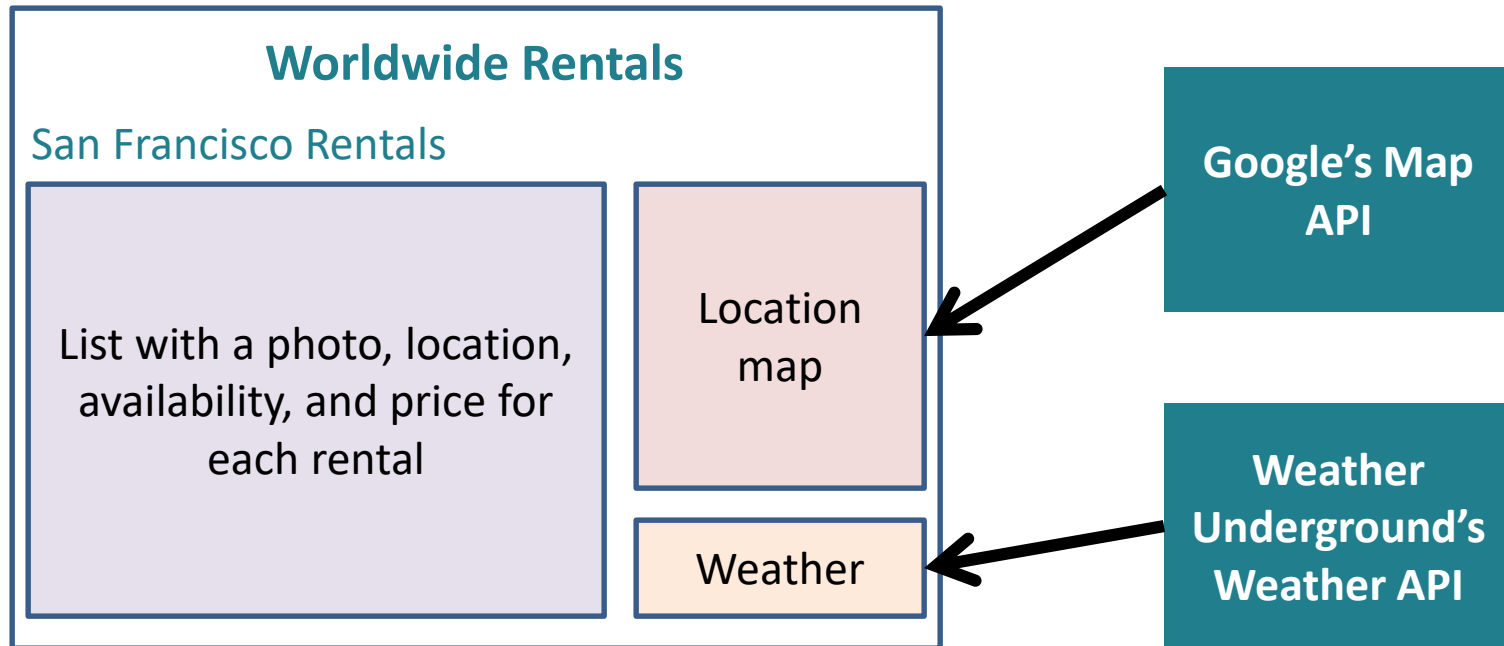
Companies create APIs to give others the opportunity to interface with their data and to broaden their audience.

Companies that develop APIs are not worried about how the data is presented; instead, they allow other developers to create the front-end that uses their data.



Why use an API?

By using data that other companies already have and maintain, your company's development resources can focus on your company's offerings.



How does an API work?

APIs use a “request and response” method of conversation. APIs function much like a conversation between people but with defined rules about the type of communication allowed.

1. Greetings
2. Ask for information
3. Information is provided



An API conversation

1. Greetings
2. Ask for information
3. Information provided

What's the weather

in San Francisco?

Partly cloudy, 66.3F (19.1C), 65% humidity.

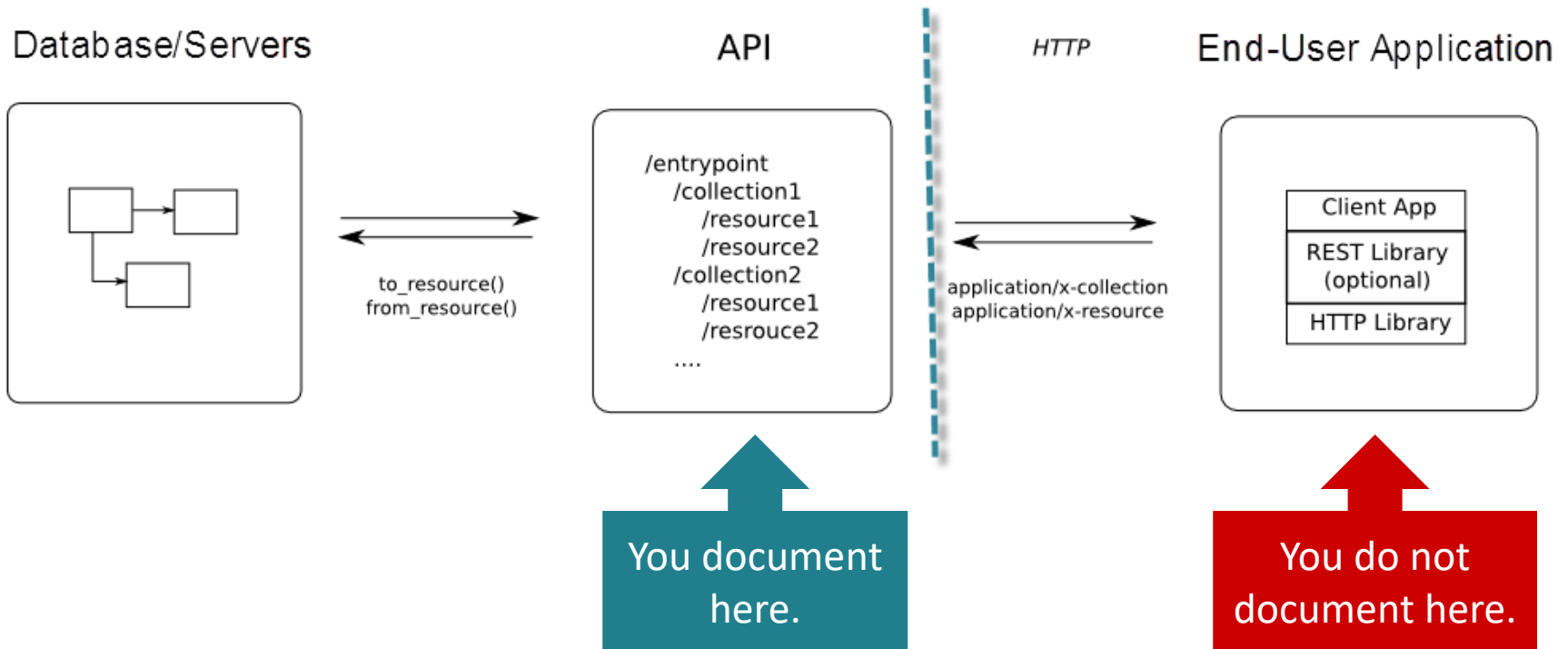


http://api.wunderground.com/api/Your_Key/conditions/q/CA/San_Francisco.json

```
{
  "response": {
    "version": "0.1",
    "display_location": {
      "city": "San Francisco",
      "state": "CA",
      "weather": "Partly Cloudy",
    },
  }
}
```



API informational flow



API web services

SOAP (Simple Object Access Protocol)

SOAP is an XML-based (Extensible Markup Language) communication protocol for accessing web services that often uses HTTP (Hypertext Transfer Protocol).

REST (Representational State Transfer)

REST is an architectural style that often relies on a URL to make a request. REST does not rely on XML responses.



API terminology

XSD (XML Schema Definition) describes what an XML document can contain, the structure of the XML document, and the rules for data content. Also referred to as *schema*.

Elements are the main building blocks of all XML documents, containing the data and determining the structure of the instance document. Elements are defined within the XSD.

Attributes provide extra information within an element. Attributes have name and type properties and are defined within an XSD. Attributes are also defined within the XSD.

WSDL (Web Service Definition Language) is an XML-based interface definition language that is used for describing the functionality offered by a web service.

Liquid Technologies (<http://www.liquid-technologies.com>) has an XML Schema Tutorial for learning about how schemas work.



SOAP and REST examples

SOAP

```
<?xml version="1.0"?>
```

```
<soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-envelope" soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
```

```
<soap:body weather="http://www.acme.com/weather">
```

```
<weather:GetLocalTemp>
```

```
<weather:ZipCode>12345</weather:ZipCode>
```

```
</weather:GetLocalTemp>
```

```
</soap:Body>
```

```
</soap:Envelope>
```

REST

```
GET http://www.acme.com/weather/LocalTemp/12345
```



REST example

This REST example uses the Photobucket API to retrieve the URL of an album for a particular user (AllisonLori).

Request

The GET operation tells the web service to retrieve data.

The format=xml section defines that the response should be returned in XML format.

GET

http://api.photobucket.com/album/AllisonLori/url?format=xml&oauth_consumer_key=000000000&oauth_nonce=b7d12041c21a050fee2e59d46ae4f5552&oauth_signature=bRkAAwTWOQArYMA9%2BRKwzLubW%2B&oauth_signature_method=HMAC-SHA1&oauth_timestamp=1236628744&oauth_version=1.0

Authorization information is required to ensure that only an authorized user is accessing the data.

Photobucket GUI

The screenshot displays the Photobucket website's user interface. At the top, a navigation bar includes the Photobucket logo, links for 'Browse', 'Editor', 'Upload', and 'Print Shop', a search bar, and buttons for 'Log in' and 'Sign up'. Below this, a secondary navigation bar lists categories: 'All Products', 'Home Decor', 'Photo Gifts', and 'Phone & Tablets'. The main content area is divided into two sections. On the left, a 'Welcome to Photobucket' message is followed by the tagline 'Print. Store. Edit. Share. All your memories in one place.' Below this is a sign-up form with fields for 'Email' and 'Password', a 'Sign up' button, and a link to 'Sign up with Facebook'. On the right, a large promotional banner features a close-up of a young girl's face next to a white Easter bunny. The banner text reads: 'Share your boldest, brightest most colorful photo for a chance to win a \$500 prize package!' with a 'LEARN MORE' button. A dark overlay is positioned in the center-right, containing social media login options for Facebook and Twitter, and a standard login form with fields for 'Email or Username' and 'Password', a 'Remember me' checkbox, and a 'Log in' button. The browser's address bar shows 's95.photobucket.com'.

s95.photobucket.com

photobucket Browse Editor Upload Print Shop Search Log in Sign up

All Products Home Decor Photo Gifts Phone & Tablets

Welcome to Photobucket

Print. Store. Edit. Share.
All your memories in one place.

Email

Password

Sign up

OR

Sign up with Facebook

Log in

OR

Email or Username

Password

Remember me

Log in

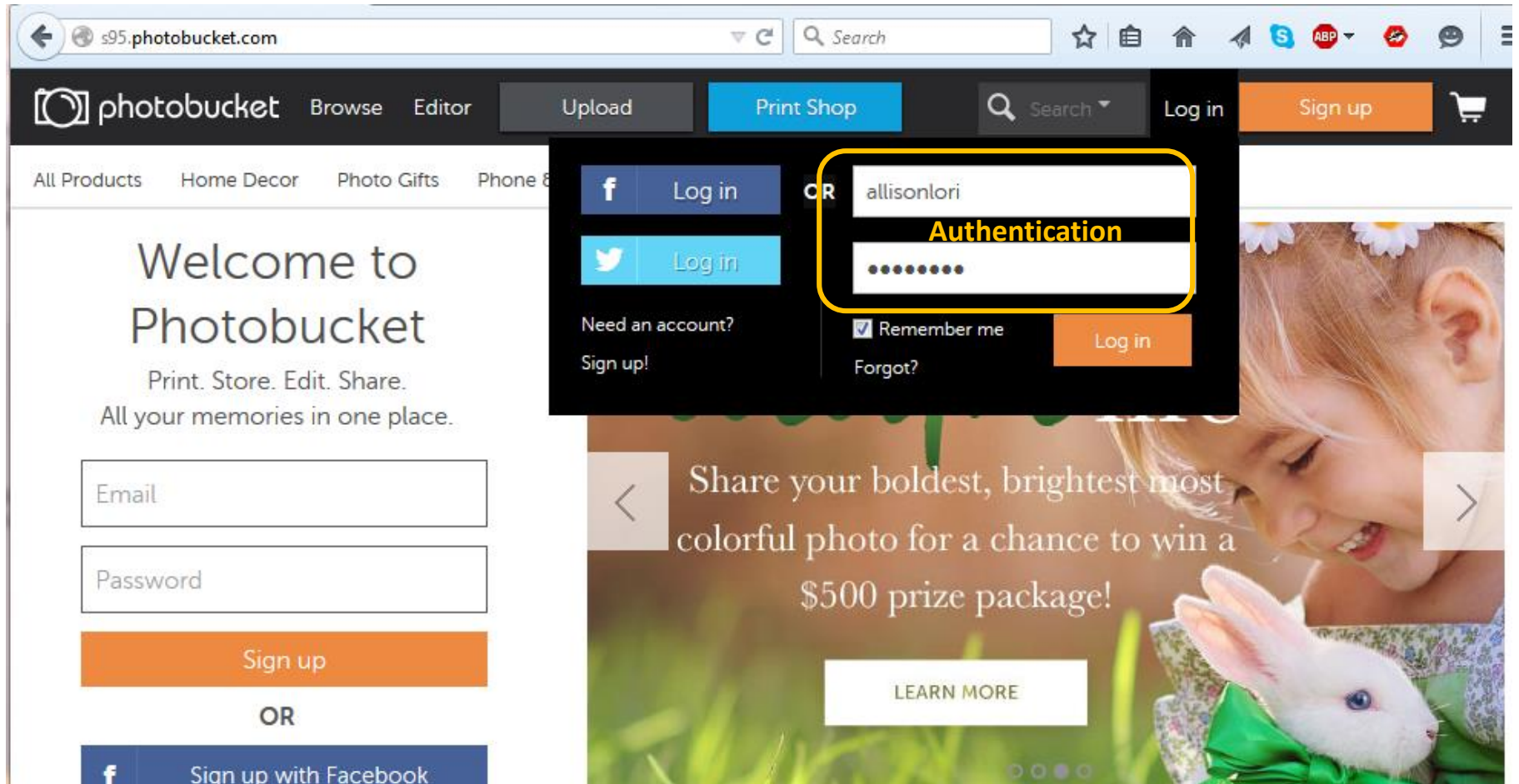
Need an account?
Sign up!

Forgot?

Share your boldest, brightest most colorful photo for a chance to win a \$500 prize package!

LEARN MORE

Photobucket GUI



Photobucket GUI

The screenshot displays the Photobucket web interface for a user named AllisonLori. The browser's address bar is highlighted with a yellow box, showing the URL `s95.photobucket.com/user/AllisonLori/library/?sort=3&page=1`. The navigation bar includes links for Library, Browse, Editor, Upload, and Print Shop, along with a search bar and a shopping cart icon. Below the navigation bar, there are links for All Products, Home Decor, Photo Gifts, Phone & Tablet Cases, Prints, and Calendars. The main content area is titled "Your Bucket" and features a sidebar with links to "Your Bucket (6)", "Recent Uploads", "Mobile Uploads", "Facebook", and "Albums". The main area shows a grid of photos, including a blue hydrangea, a red flower, a desert landscape, two penguins, and a koala. A dashed box with a plus icon and the text "Drag photos here to upload" is visible. The "Order by" dropdown is set to "Newest First", and the "View as" option is set to "Grid". On the right, the "SHARE LINKS" section provides various sharing options: Email & IM, HTML Embed, Flash Embed, Slideshow, Story, and Story Embed, each with a corresponding code or link. The "ACTIONS" section at the bottom right includes links for "Create album" and "Order prints".

Browser Address Bar: `s95.photobucket.com/user/AllisonLori/library/?sort=3&page=1`

Navigation Bar: Library, Browse, Editor, Upload, Print Shop, Search, Cart

Product Links: All Products, Home Decor, Photo Gifts, Phone & Tablet Cases, Prints, Calendars

Your Bucket: Organize: Off, f, p, t, g+, e, l, r

Order by: Newest First, View as: Grid

Drag photos here to upload

Albums: + Create New Album, SubAlbum

SHARE LINKS ?

Email & IM: `http://s95.photobucket.com/`

HTML Embed: `<iframe width="480" height="`

Flash Embed: `<div style="width:480px;text-i`

Slideshow: `http://s95.photobucket.com/`

Story: `http://s95.photobucket.com/`

Story Embed: `<iframe width="480" height="`

ACTIONS: + Create album, Order prints

REST example

Response

<response>

<status>OK</status> -- Indicates that the request was successful

<content>

<username>AllisonLori</username> -- The username we asked for in the GET request

<subdomain>

<album>library</album> -- The album name, *library*, is the default album for a user on Photobucket

</subdomain>

<path>s95.photobucket.com/user/AllisonLori/library/?sort=3&page=1</path> -- The path to the album

</content> <format>xml</format>

<method>GET</method>

<timestamp>1236628744</timestamp>

</response>



Travelport Universal API Demo

[SHOP](#)[RETRIEVE UNIVERSAL RECORD](#)[LOGS](#)

☒ Show Developer Tooltips ☐ Close Panel

From

TO

Departure

Return

Adults

Child (2-12)

Infant (0-2)

1

0

0

GDS

Galileo

☐ AIR AND RAIL

☒ HOTEL

Standard Hotel Shopping ⓘ

☒ GDS

☐ Travelport Rooms and More ⓘ

Location

Reference Point

KBV

i.e. "Eiffel Tower"

22/05/2015

25/05/2015

Rooms

1

Cribs

Rollaway Beds

0

0

SEARCH

Travelport Universal API Demo

The Travelport Demo was created to inspire you to develop your application and get ahead of your competitors. The site allows you to display code that's ready to consume and showcases new content and functionality, as we continuously expand our supplier offering through our Open Platform, Travelport Universal API.

Travelport Universal API offers an array of travel content: air, hotel, vehicle, and rail. Ancillaries may also be offered, depending on the type of content.

Ready to begin your development? [Request test access to Travelport Universal API now!](#) ➤

Important! When using the Demo Site on the Production environment, there is no option to create a booking.



AIR

Api Connect



CAR

Api Connect



RAIL

Api Connect



HOTEL

Api Connect



MERCHANDISING

APTCO/API



RICH CONTENT

& Branding



SOAP example

This SOAP example uses the proprietary Travelport Universal API Demo site (<https://demo.travelportuniversalapi.com/>) to create a request and response.

Request

```
<HotelSearchAvailabilityReq xmlns="http://www.travelport.com/schema/hotel_v27_0" TraceId="7ebe1c2c-61d0-4f63-a600-0f90695912c0" AuthorizedBy="Travelport" TargetBranch="TRGT_BRCH">
  <BillingPointOfSaleInfo xmlns="http://www.travelport.com/schema/common_v27_0" OriginApplication="uAPI" />
  <HotelLocation Location="KBV" />
  <HotelSearchModifiers NumberOfAdults="1" NumberOfRooms="1" AvailableHotelsOnly="true">
    <PermittedProviders xmlns="http://www.travelport.com/schema/common_v27_0">
      <Provider Code="1G" />
    </PermittedProviders>
  </HotelSearchModifiers>
  <HotelStay>
    <CheckinDate>2015-05-22</CheckinDate>
    <CheckoutDate>2015-05-25</CheckoutDate>
  </HotelStay>
</HotelSearchAvailabilityReq>
```

This section requests a hotel in Krabi (in Thailand) from May 22 to 25, 2015

SOAP example

The header looks similar to this:

POST https://americas.universal-api.pp.travelport.com/
B2BGateway/connect/uAPI/HotelService HTTP/2.0
Accept-Encoding: gzip,deflate
Content-Type: text/xml; charset=UTF-8
SOAPAction: ""
Authorization: Basic **UniversalAPI:UserName:Password**
Content-Length: **length**

The POST operation tells the web service to send data and provides the URL of where to send the data.

Authorization sends the username and password for authentication.

The Content-Type section defines that the request and response use XML format.

Travelport Universal API Demo

[SHOP](#)[RETRIEVE UNIVERSAL RECORD](#)[LOGS](#)[UPDATE SEARCH >](#)

Shop Results KBV 22/05/2015 - 25/05/2015



HOTEL

Show: ALL

Filter by Amenities

[List View](#)[Cart](#)

Your Cart is Empty

▶ 	IBIS STYLES KRABI AO NANG	866 ⁷¹ _{THB}	-	4,200 ⁰⁰ _{THB}
▶	VOGUE RESORT AO NANG	1,870 ⁰⁰ _{THB}	-	3,200 ⁰⁰ _{THB}
▶ 	HOLIDAY INN RESORT KRABI AO NAN	2,210 ⁰⁰ _{THB}	-	5,100 ⁰⁰ _{THB}
▶	SHERATON KRABI BEACH RESORT	3,040 ⁰⁰ _{THB}	-	9,820 ⁰⁰ _{THB}
▶	SOFITEL KRABI PHOKEETHRA	3,315 ⁰⁰ _{THB}	-	18,420 ⁰⁰ _{THB}
▶	AMARI VOGUE KRABI	3,570 ⁰⁰ _{THB}	-	10,200 ⁰⁰ _{THB}
▶	NAKAMANDA RESORT AND SPA	6,200 ⁰⁰ _{THB}	-	6,200 ⁰⁰ _{THB}
▶	CENTARA GRAND BEACH RSRT KRABI	7,020 ⁰⁰ _{THB}	-	9,400 ⁰⁰ _{THB}
▶	RAYAVADEE	7,705 ⁰⁰ _{THB}	-	70,000 ⁰⁰ _{THB}
▶	PIMANN BURI LUXURY POOL VILLA RES	9,880 ⁰⁰ _{THB}	-	25,500 ⁰⁰ _{THB}

1

2

»

[XML](#)

SOAP example

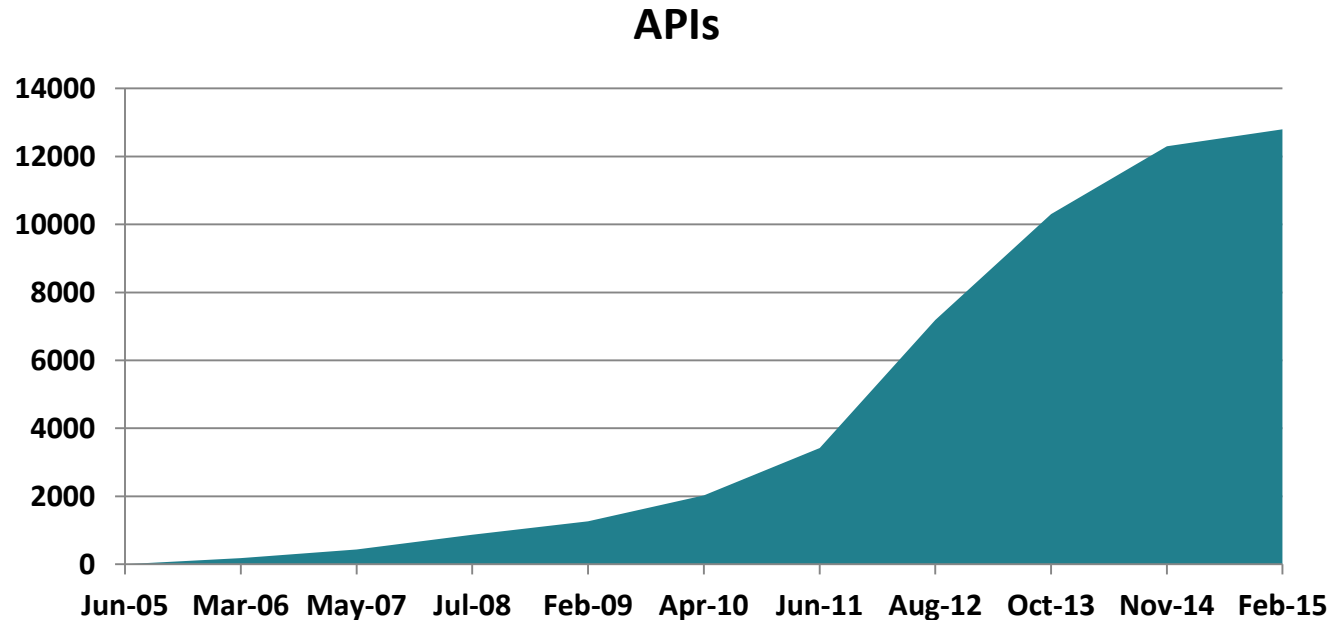
Response

```
<hotel:HotelSearchResult>
  <common_v31_0:VendorLocation ProviderCode="1G" VendorCode="SI" VendorLocationID="53742"
Key="caz3FGu8S2K00Ve64rEmrw==" />
  <hotel:HotelProperty HotelChain="SI" HotelCode="53742" HotelLocation="KBV" Name="SHERATON KRABI BEACH
RESORT" VendorLocationKey="caz3FGu8S2K00Ve64rEmrw==" HotelTransportation="Limo"
ReserveRequirement="Other" ParticipationLevel="Best Available Rate and Inside Shopper participant"
Availability="Available">
    <hotel:PropertyAddress>
      <hotel:Address>155 MOO 2 NONG THAL</hotel:Address>
    </hotel:PropertyAddress>
    <common_v31_0:Distance Units="MI" Value="2" Direction="S" />
    <hotel:Amenities>
      <hotel:Amenity Code="AICO" />
      << Over 35 amenity codes returned >>
    </hotel:Amenities>
  </hotel:HotelProperty>
  <hotel:RateInfo MinimumAmount="THB3040.00" MinAmountRateChanged="false" MaximumAmount="THB9820.00"
MaxAmountRateChanged="false" />
</hotel:HotelSearchResult>
```



Why should I care about APIs?

The number of APIs is constantly growing. APIs speed development and allow companies to use services to enhance their products. Many companies are offering APIs to reach a wider audience with their products.

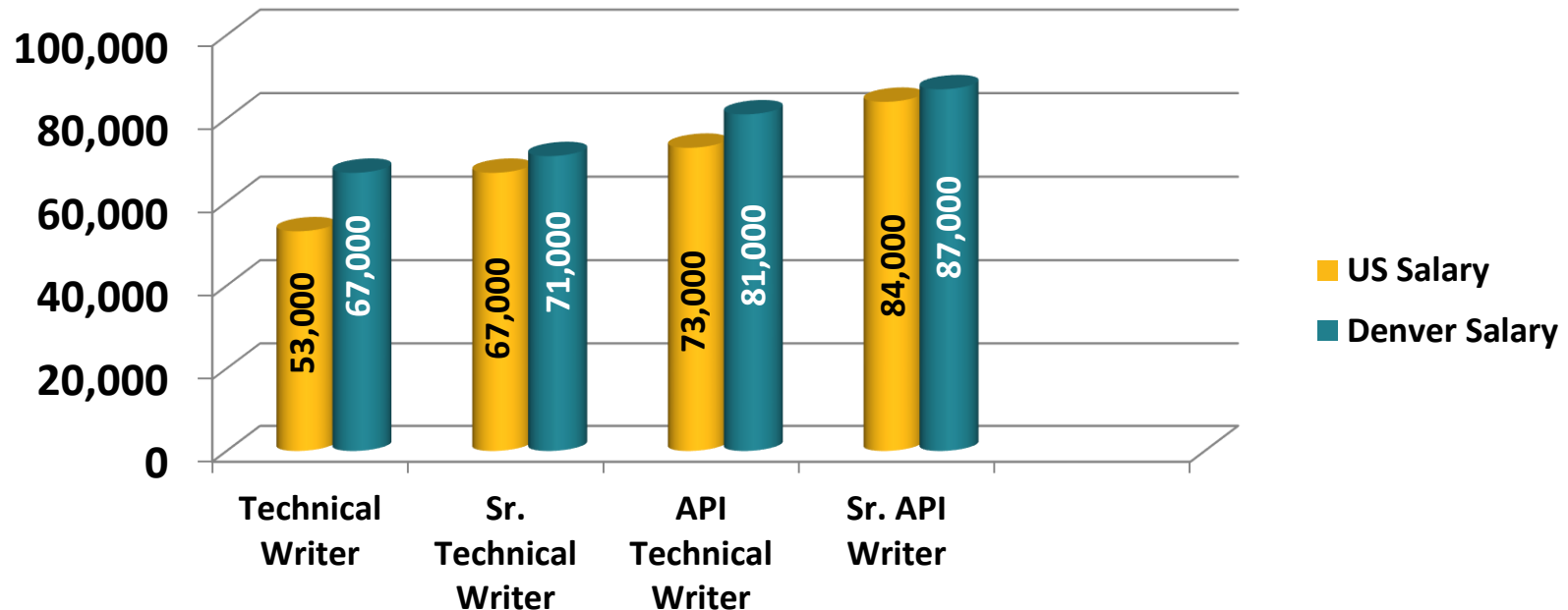


On February 1, 2015, ProgrammableWeb had 12,794 APIs listed on its site.



What can APIs do for me?

- Because APIs have no GUI, documentation is key.
- API documentation skills could lead to more money.



Data from SimplyHired.com as of February 2015



Documenting your first API

What you don't need to know:

- How to program

What you do need to know:

Most of what you need to know is already in your technical writer toolbox. Do what you normally do when documenting a GUI:

- Gather information.
- Filter the information you receive down to what a user needs to know.
- Lay out information such that it can be easily found and searched.

What you need to learn:

- API terminology
- Minimal familiarity with code



What information do I provide?

- Getting started information
 - Credentials
 - Connectivity
 - Security
 - Quick start task
- Industry information
- Why use this API



What information do I provide?

- How to use the API
 - Workflows for complex functionality
 - Interactions between the transactions
 - For each transaction, describe the required and optional information, acceptable values, return values, errors/exceptions
 - Tutorials and message examples are highly desirable
- Reference documentation
 - WSDLs and schema
 - Sample code
- Versioning information



What's the big deal with versions?

New API versions can break existing functionality.

- Define the version support timeline.
- Ensure customers are notified in advance of breaking changes, so they can assign resources to upgrade to the new version.
- Decide how you will support multiple versions with the same source documentation. Or, will you create a help version for each API release?
- Provide a detailed list of changes between versions and clearly identify breaking changes.



Where do I find this information?

Like information about a GUI product, API product information comes from a variety of sources, but you may have to dig a little more to find it.

- Interview subject matter experts
- Attend functionality walkthroughs and developer meetings
- Read product documents, such as functional designs, Wiki pages and use cases, or JIRA stories

Pare down the information to what is needed by the customer.



I have to read code?!

- Poke around code (schema files and sample code)
- Ask questions
- Use tools to make it easier
 - Altova XMLSpy
 - Eclipse (free)
 - XSD Diagram (free)
 - <oXygen/> XML Editor
 - Notepad ++

API documentation examples

Good documentation examples:

- [Flickr API](#)
- [Stripe](#)

How do they compare to our documentation?

- [Travelport Universal API](#)



Wrap-up and questions

- APIs are growing, as is the need for those who can provide user documentation.
- API documentation skills can make you a more desirable job candidate and could yield higher pay.
- The skills you have as a technical writer are easily adaptable to documenting APIs.



Every accomplishment starts with the decision to try.

Author unknown

Allison Ellington

Allison.Ellington@Travelport.com

Lori Guillory

Lori.Guillory@Travelport.com

