Applied Data Analysis (CS401)

Lecture 3
V for Variety
2018/10/04

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Robert West

dlab

Announcements

- Everyone registered in a **team?** Does your team have **3 members?**
  - If not, talk to **Tuğrulcan** (@teajay) **in break or after lecture**
- Mattermost: help each other by answering questions!
- Tomorrow's lab session:
  - Questionnaire to get to know you better for upcoming hands-on activities
  - Talk by Claudiu Musat, Swisscom

swisscom

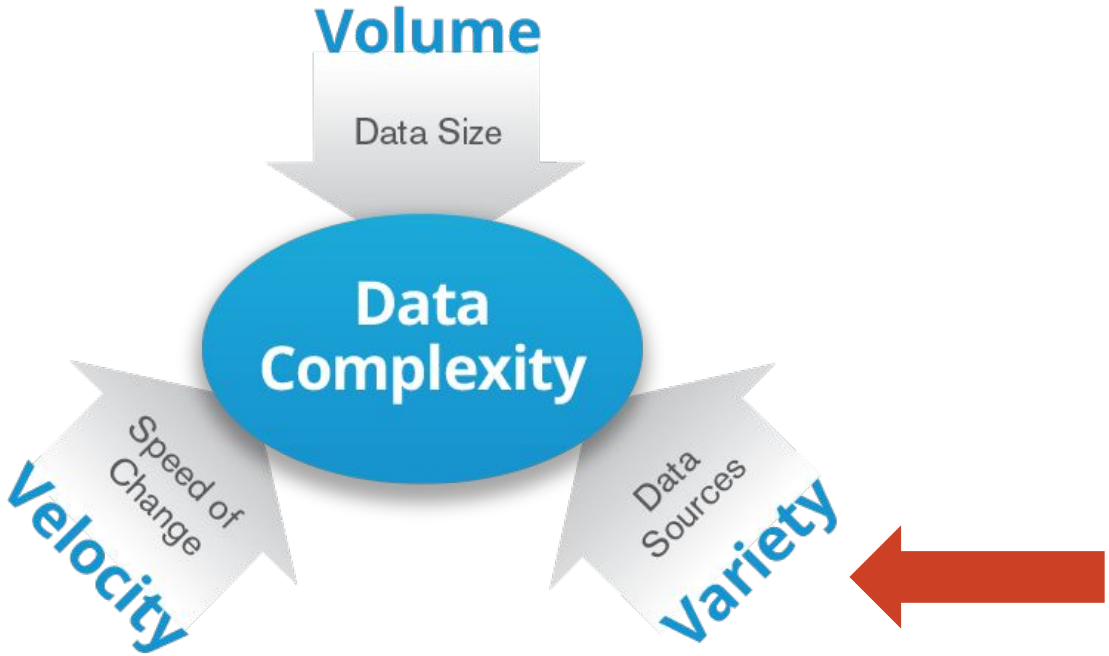
Feedback

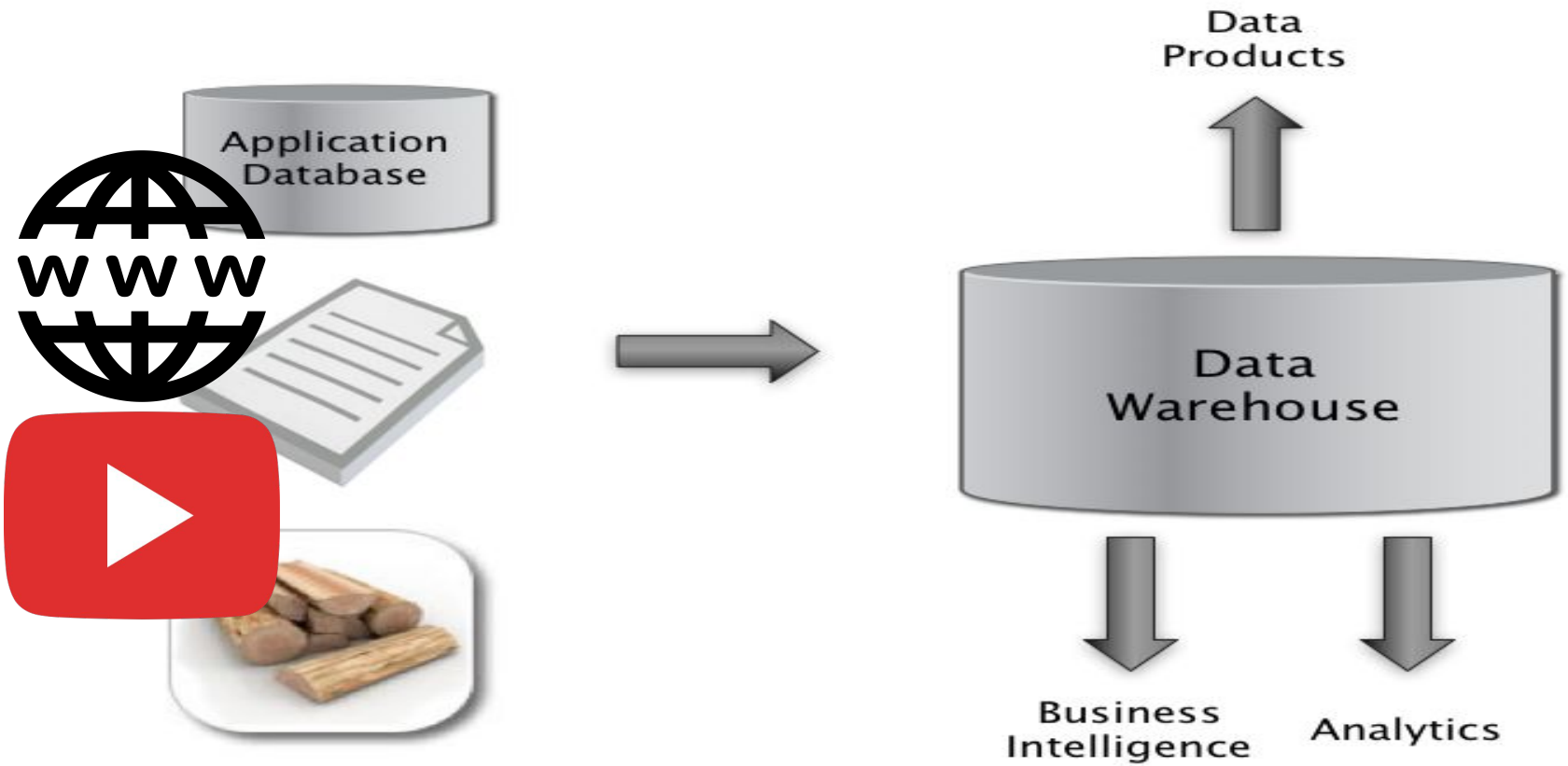
Give us feedback on this lecture here:
https://go.epfl.ch/ada2018-lec3-feedback

- What did you (not) like about this lecture?
- What was (not) well explained?
- On what would you like more (fewer) details?
- What is your credit card number?
- …

The 3 (or more…) Vs of Big Data

For Volume and Velocity: EPFL course "Database Systems"




Application Database
WWW
Data Products
Data Warehouse
Business Intelligence
Analytics

Data preparation overview

ETL:

- We need to **extract** data from the **source(s)**
- We need to **transform** data at into manageable format
- We need to **load** data into the **sink**

- Sources: file, database, event log, web site, XML…
- Sinks: Python, R, SQL DB, NoSQL store, files, HDFS…

Data sources at Web companies

Examples from Facebook

- Application databases
- Web server logs
- Event logs
- API server logs
- Ad server logs
- Search server logs
- Advertisement landing page content
- Wikipedia
- Images and video

Structured data (with clear schema)

Semi-structured data ("self-describing" structure)

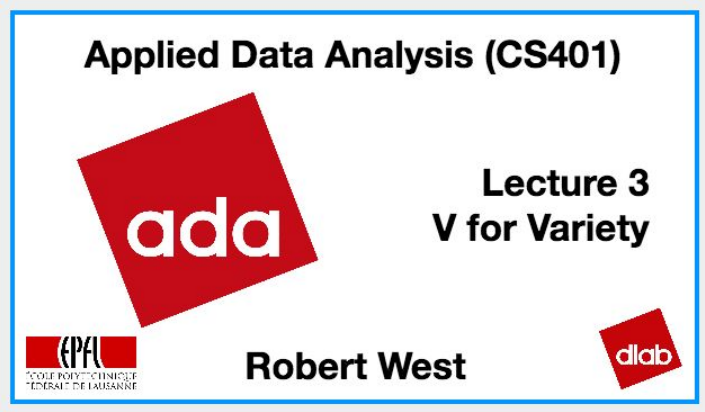
Unstructured data

opendata.swiss

opendata.swiss is the portal for Swiss open government data (OGD). Here you can download Swiss government data free of charge.

Check the **variety** of formats!

A whirlwind tour of data formats



Schemas, XML, and JSON

The (changing) role of schemas

A **schema** specifies the **structure** and **types** of a data repository, e.g., the types of each column in a table.

They may also specify constraints **within** or **between** data fields.

Traditional databases are **schema-on-write**. You cannot load data into a table without a schema.

Newer (NoSQL) data stores are **schema-on-read** or (shamelessly) **schemaless**: you can defer applying a schema until you read the data, or avoid schema altogether.

Schema-on-write

SQL:

CREATE TABLE professors (id INT, name STRING, iq INT)

INSERT INTO professors (id, name, iq)

VALUES (5543, ”Bob West”, 45)

Schema-on-read

XML: Generalizes HTML and specifies data **structure**. XML schema can be applied later to interpret XML data and specifies **data types**. Here is some XML-encoded **data**:

```
<location>
  <latitude>37.78333</latitude>
  <longitude>122.4167</longitude>
</location>
```

When stored without a schema, the numerical data is stored as **strings**.

XML schema

```
<location>
  <latitude>37.78333</latitude>
  <longitude>122.4167</longitude>
</location>
```

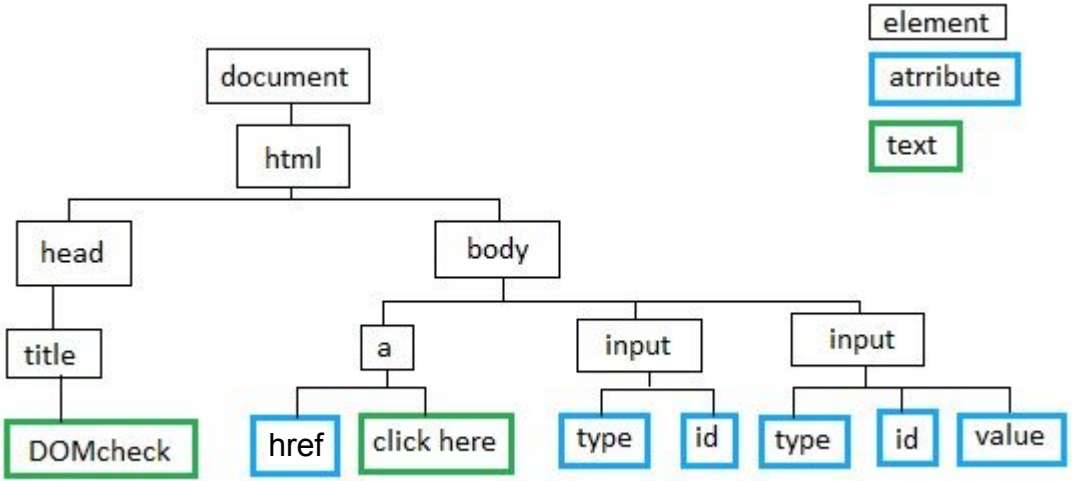
An XML schema for this element:

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
elementFormDefault="unqualified">
<xsd:complexType name="location">
  <xsd:sequence>
    <xsd:element name="latitude" type="xsd:decimal"/>
    <xsd:element name="longitude" type="xsd:decimal"/>
  </xsd:sequence>
</xsd:complexType name="location">
```

XML and DOM

XML is a text format that encodes **DOM** (Document-Object Model), a widely used data structure, e.g., for Web pages.

The DOM is tree-structured:




```
<html>
  ead><title>DOMcheck</title></head>
  ody>
  <a href="http://...">click here</a>
  ...
  body>
  html>
```

XML queries

XML schema allows a DB to interpret the data when running queries, e.g., to do **arithmetic** or **range queries** on numerical values.

XQuery is a standard for querying XML data with or without schema:

```
<map>
  <city>
    <name>Ouagadougou</name>
    <location>
      <latitude>37.78333</latitude>
      <longitude>122.4167</longitude>
    </location>
  </city>
  ...
</map>
```

```
<places>{
for $city in /map/city
  let $latlong := $city/location
  where ((xs:float($latlong/latitude) < 39) and
         (xs:float($latlong/latitude) > 38))
  return
    <place name="{$city/name}“/>
}</places>
```

JSON

JSON (JavaScript Object Notation) by contrast is a schemaless data description language (schema support was added later):

```
{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25,
  "address": {
       "streetAddress": "21 2nd Street",
       "city": "New York",
       "state": "NY",
       "postalCode": "10021-3100" },
  "phoneNumbers": [
      { "type": "home",
        "number": "212 555-1234" },
      { "type": "office",
        "number": "646 555-4567" } ],
  "children": [],
  "spouse": null
}
```

Figure 1: Shameless Jason

JSON

- Same expressivity as XML: hierarchical objects
- But JSON directly reflects how object is stored in target language (e.g., Javascript, Python), bare-bones
- More light-weight, but may be more painful, too:
- Transformations on JSON data are **procedural** in the target language (not *declarative* as in XQuery)
- (But JSON is becoming more and more shemaful…)

Figure 2: Shameful Jason

XML vs. JSON (roughly)

XML:

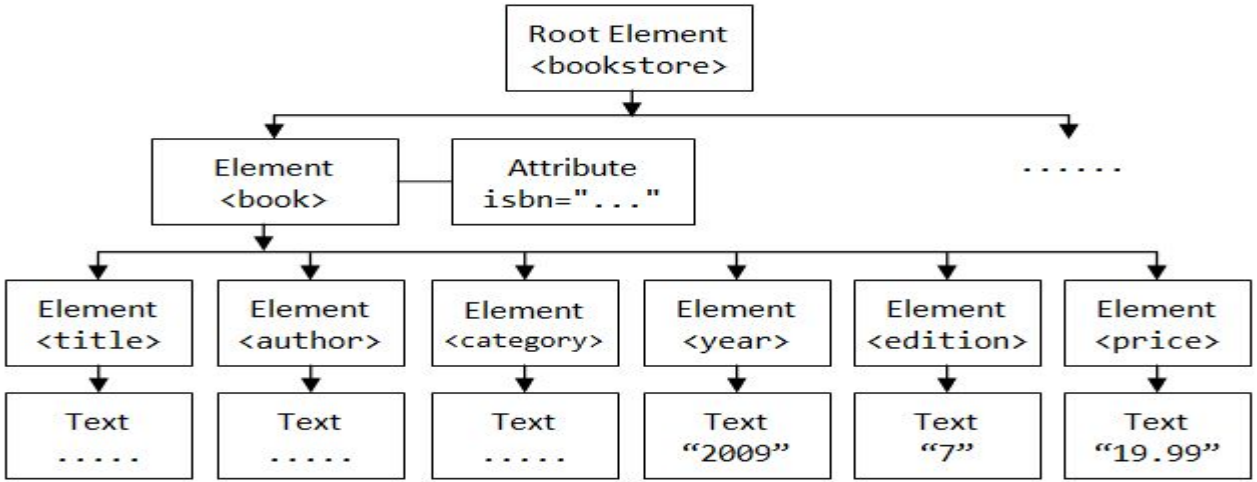
- Separation between schema and data.
- Data can be represented and stored without schema (as strings).
- More verbose (but not true after compression or in DB).
- Standard query/transformation languages: XQuery, XSLT, etc.

JSON:

- Types inferred inline. Schema rarely used but can be.
- Data without schema uses type inference (string, int, float,…).
- More succinct in ASCII form.
- Transformation/ingestion rely on code (e.g., Javascript, Python).

Processing XML and JSON

- The DOM is an easy object to work with: all the data in the object is accessible by links (edges in the DOM tree).
- The problem is that I might not care about most of the data, and I might **not be able to fit the DOM for a large object in RAM**.



Event-driven parsing: SAX

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- bookstore.xml -->
<bookstore>
  <book ISBN="0123456001">
    <title>Java For Dummies</title>
    <author>Tan Ah Teck</author>
    <category>Programming</category>
    <year>2009</year>
    <edition>7</edition>
    <price>19.99</price>
  </book>
```

→ Document Header

→ Comment

→ Start-element “bookstore”

→ Start-element “book”

→ Start-element “title”

End-element “title”

→ End-element “book”

Event-driven parsing: SAX

A SAX parser finds all the **open/close-tag events** in an XML documents, and **does callbacks to user code**.

+ User code can respond to only a subset of events corresponding to the tags it is interested in.
+ User code can correctly compute aggregates from the data rather than create a record for each tag.
+ User code can implement flexible error recovery strategies for ill-formed XML.
- User code must implement a state machine to keep track of “where it is” in the DOM tree.

What about JSON?

Most JSON parsers construct the “DOM” directly.

But there are a few SAX-style parsers:

- Jackson
- JSON-simple

Sometimes SAX-style is the only way to handle ill-formed datasets

Tabular data

Tabular Data

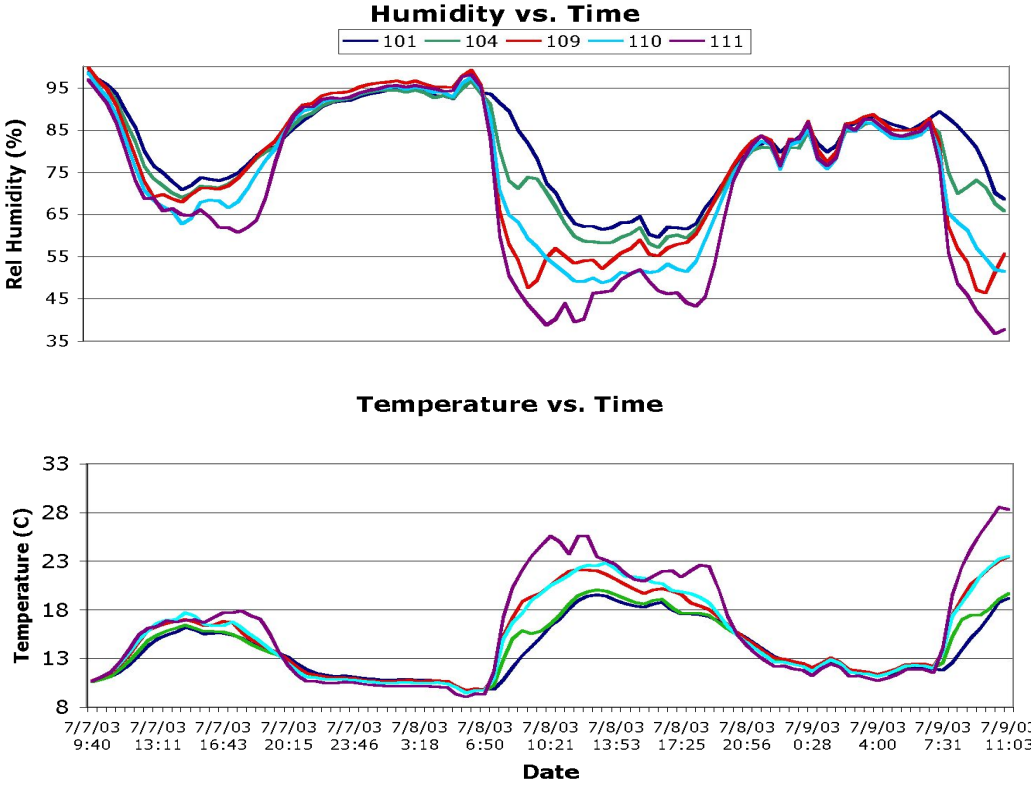
What is a table?

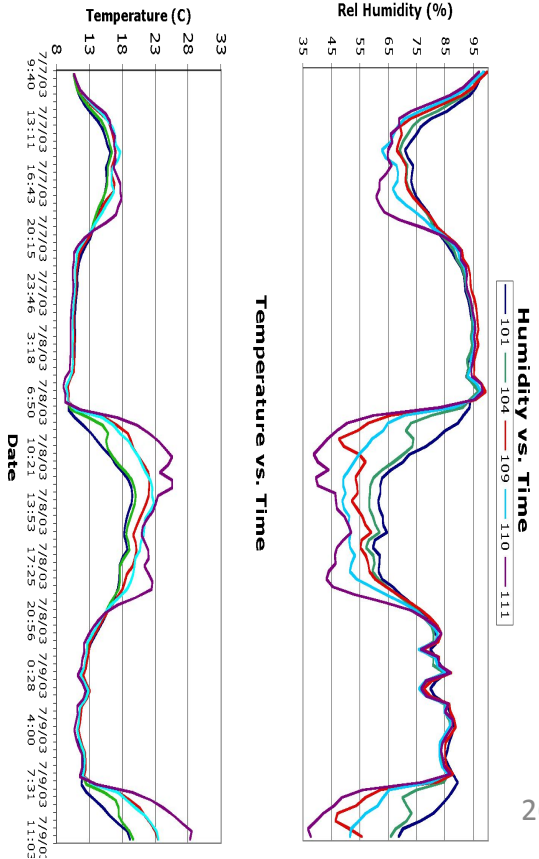
- A **table** is a collection of **rows** (or, alternatively, **columns)**
- Each row has an **index**
- Each column has a **name**
- A **cell** is specified by an (index, name) pair
- A cell may or may not have a **value**

Schema = column types (minimal), possibly also column names

Often stored as text files in CSV or TSV format.

IoT: Example measurements





Things to do with time series table

Sensors usually output data in the form of time series, collated in a tabular format. Yet, a system dealing with sensor data should:

- support both long-term (**trend**) and short-term (**real-time**) queries.
- have **low latency** but also efficient, **real-time indexing** for longer-term queries.
- support triggers (**alerts**) for a variety of conditions.

The complexity of a data format does not determine the complexity of the system required to properly handle it! (think about the stock market as well…)

Log files example: Apache Web server

Processes, usually daemons, create logs
e.g., `httpd, mysqld, syslogd`

- `66.249.65.107 - - [08/Oct/2007:04:54:20 -0400] "GET /support.html HTTP/1.1" 200 11179 "-" "Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)"`

- `111.111.111.111 - - [08/Oct/2007:11:17:55 -0400] "GET / HTTP/1.1" 200 10801 "http://www.google.com/search?q=in+love+with+ada+lovelace+what+to+do&ie=utf-8&oe=utf-8 &aq=t&rls=org.mozilla:en-US:official&client=firefox-a" "Mozilla/5.0 (Windows; U; Windows NT 5.2; en-US; rv:1.8.1.7) Gecko/20070914 Firefox/2.0.0.7"`

Syslog – A Standard for System Messages

- Developed by Eric Allman (at Berkeley) as part of the Sendmail project
- Standardized by the IETF in RFC 3164 and RFC 5424
- Listens on port 514 using UDP
- Puts data in /var/log/messages by default
- An evolution of it is most likely running on your machine

Syslog

dhcp-47-129:DataScienceF15> syslog -w 10

Feb 3 15:18:11 dhcp-47-129 **Evernote**[1140] <Warning>: -[EDAMAccounting read:]: unexpected field ID 23 with type 8. Skipping.
Feb 3 15:18:11 dhcp-47-129 Evernote[1140] <Warning>: -[EDAMUser read:]: unexpected field ID 17 with type 12. Skipping.
Feb 3 15:18:11 dhcp-47-129 Evernote[1140] <Warning>: -[EDAMAuthenticationResult read:]: unexpected field ID 6 with type 11. Skipping.
Feb 3 15:18:11 dhcp-47-129 Evernote[1140] <Warning>: -[EDAMAuthenticationResult read:]: unexpected field ID 7 with type 11. Skipping.
Feb 3 15:18:11 dhcp-47-129 Evernote[1140] <Warning>: -[EDAMAccounting read:]: unexpected field ID 19 with type 8. Skipping.
Feb 3 15:18:11 dhcp-47-129 Evernote[1140] <Warning>: -[EDAMAccounting read:]: unexpected field ID 23 with type 8. Skipping.
Feb 3 15:18:11 dhcp-47-129 Evernote[1140] <Warning>: -[EDAMUser read:]: unexpected field ID 17 with type 12. Skipping.
Feb 3 15:18:11 dhcp-47-129 Evernote[1140] <Warning>: -[EDAMSyncState read:]: unexpected field ID 5 with type 10. Skipping.
Feb 3 15:18:49 dhcp-47-129 **com.apple.mtmd**[47] <Notice>: low priority thinning needed for volume Macintosh HD (/)

splunk> : “Spelunking” for bugs

Grab data from many machines

Index it

Check for unusual events:

- Disk problems
- Network congestion
- Security attacks

Monitor resources:

- Network
- Memory usage
- Disk use, latency
- Threads

Dashboard for cloud administration

Splunk built a successful business model around these features, leading to an IPO and multi-billion $ valuation!

A word (or two) on binary formats

- They are often the key to performance, **avoiding expensive parsing** (“deserialization”)
- Modern binary formats support nested structures, various levels of schema enforcement, **compression**, etc.
- Protocol Buffers (Google), Avro (supports schema evolution), Parquet (column-oriented, first-class citizen in Spark), etc.

Consider converting to one of those formats at the beginning of your processing pipeline (especially on a project with “big data”!)

More formats
(and useful datasets…)

Wikipedia

I don’t think it needs any introduction… :)

200+ languages, millions of entities

San Francisco

From Wikipedia, the free encyclopedia
(Redirected from San Francisco, California)

Coordinates: 37°47′N 122°25′W

This article is about the city and county in California. For other uses, see San Francisco (disambiguation).

San Francisco (initials **SF**[17]) (/ˌsæn frənˈsɪskoʊ/, Spanish for Saint Francis; Spanish: [san franˈsisko]), officially the **City and County of San Francisco**, is the cultural, commercial, and financial center of Northern California. The consolidated city-county covers an area of about 47.9 square miles (124 km²)[18] at the north end of the San Francisco Peninsula in the San Francisco Bay Area. It is the fourth-most populous city in California, and the 13th-most populous in the United States, with a 2016 census-estimated population of 870,887.[13] The population is projected to reach 1 million by 2033.[19]

San Francisco was founded on June 29, 1776, when colonists from Spain established Presidio of San Francisco at the Golden Gate and Mission San Francisco de Asís a few miles away, all named for St. Francis of Assisi.[1] The California Gold Rush of 1849 brought rapid growth, making it the largest city on the West Coast at the time. San Francisco became a consolidated city-county in 1856.[20] After three-quarters of the city was destroyed by the 1906 earthquake and fire,[21] San Francisco was quickly rebuilt, hosting the Panama-Pacific International Exposition nine years later. In World War II, San Francisco was a major port of embarkation for service members shipping out to the Pacific Theater.[22] It then became the birthplace of the United Nations in 1945.[23][24][25] After the war, the confluence of returning servicemen, massive immigration, liberalizing attitudes, along with the rise of the "hippie" counterculture, the Sexual Revolution, the Peace Movement growing from opposition to United States involvement in the Vietnam War, and other factors led to the Su… center of liberal activism in the United States. Politi…

A popular tourist destination,[26] San Francisco is k… and landmarks, including the Golden Gate Bridge, … Chinatown district. San Francisco is also the headc… Levi Strauss & Co., Gap Inc., Fitbit, Salesforce.com… Electric Company, Yelp, Pinterest, Twitter, Uber, Ly… home to number of educational and cultural instituti… Museum, the San Francisco Museum of Modern Ar…

San Francisco has several nicknames, including "T… City", and as well as older ones like "The City that … City".[17] As of 2017, San Francisco is ranked high …

Contents [hide]

San Francisco, California
Consolidated city-county
City and County of San Francisco

San Francisco and the Golden Gate Bridge from …

link

Learning resources from Wikiversity

Places adjacent to San Francisco [show]
v • t • e City and County of San Francisco [show]
Articles relating to the City and County of San Francisco [show]

Authority control WorldCat Identities · VIAF: 143700861 · LCCN: n79018452 · ISNI: 0000 0004 0461 8991 · GND: 4051520-5 · SUDOC: 040776433 · NDL: 00628542

Categories: San Francisco | 1850 establishments in California | California counties | Cities in the San Francisco Bay Area | Consolidated city-counties in the United States | Counties in the San Francisco Bay Area | County seats in California | Hudson's Bay Company trading posts | Incorporated cities and towns in California | Populated coastal places in California | Populated places established in 1776 | Port cities and towns of the West Coast of the United States | Spanish mission settlements in North America

Location of San Francisco in California
Coordinates: 37°47′N 122°25′W

Country	United States
State	California

Wikipedia

How to work with it?

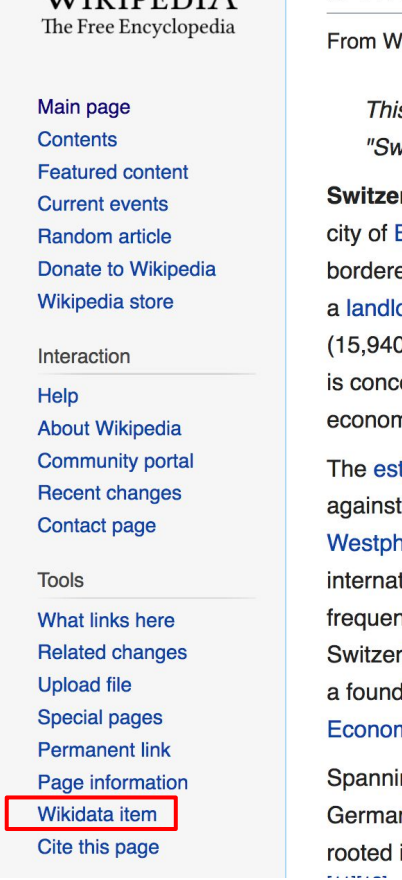
- XML dumps with wiki markup, SQL DB dumps
- Unicode, **size**, etc.

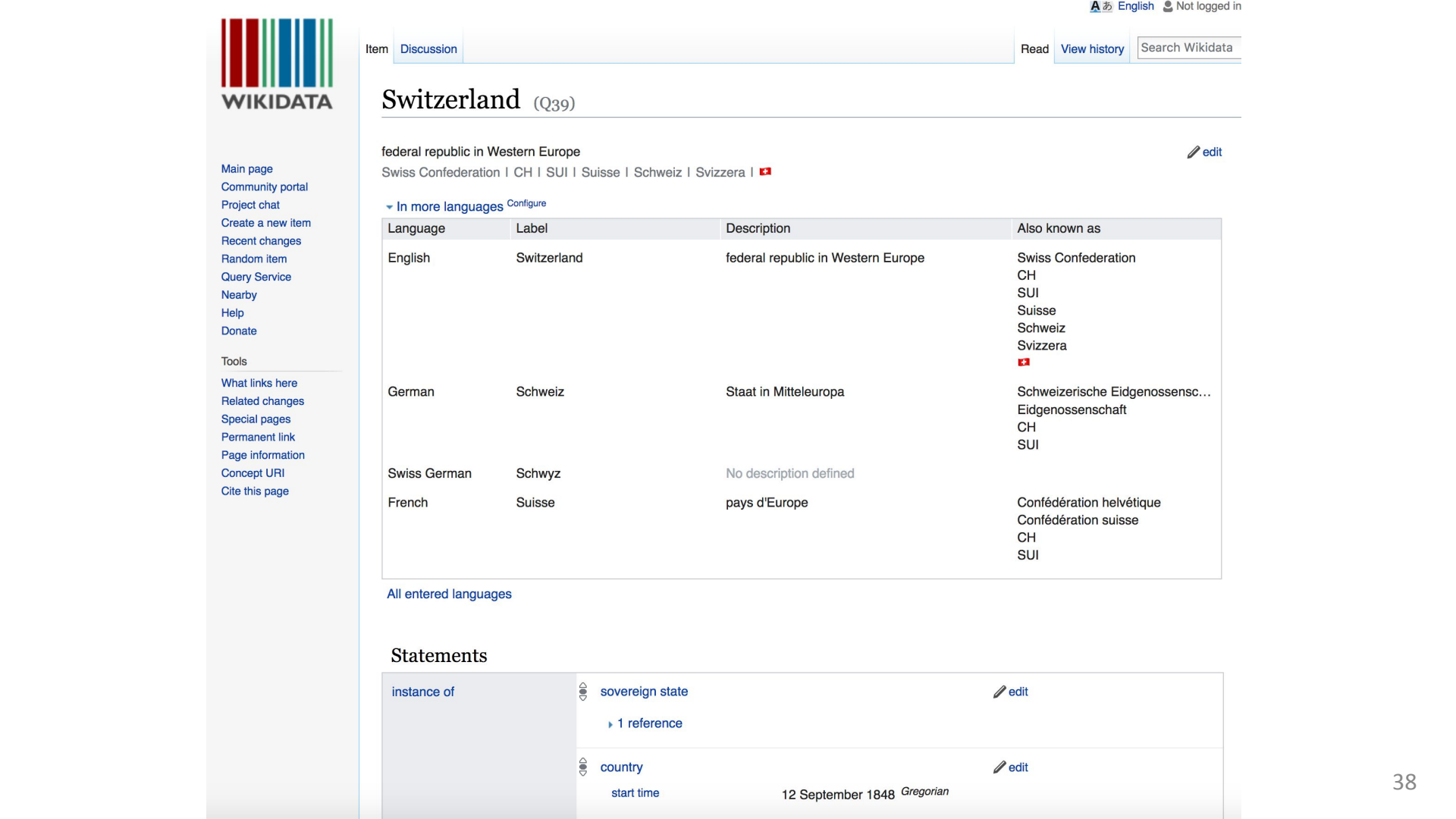
(1) Find projects on GitHub to help you

(2) Use more structured versions (p.t.o.)

Wikidata

- “DB version” of Wikipedia
- {fr:Suisse, de:Schweiz, it:Svizzera, en:Switzerland, …} ⟶ Q39
- Both API access and full database dumps (JSON, RDF)




English
Not logged in
WIKIDATA
Item
Discussion
Read
View history
Search Wikidata
Switzerland (Q39)
federal republic in Western Europe
edit
Swiss Confederation | CH | SUI | Suisse | Schweiz | Svizzera |
Main page
Community portal
Project chat
Create a new item
Recent changes
Random item
Query Service
Nearby
Help
Donate
Tools
What links here
Related changes
Special pages
Permanent link
Page information
Concept URI
Cite this page
In more languages Configure
Language
Label
Description
Also known as
English
Switzerland
federal republic in Western Europe
Swiss Confederation
CH
SUI
Suisse
Schweiz
Svizzera
German
Schweiz
Staat in Mitteleuropa
Schweizerische Eidgenossensc…
Eidgenossenschaft
CH
SUI
Swiss German
Schwyz
No description defined
French
Suisse
pays d'Europe
Confédération helvétique
Confédération suisse
CH
SUI
All entered languages
Statements
instance of
sovereign state
edit
1 reference
country
edit
start time
12 September 1848 Gregorian

Similar to Wikidata: Freebase

- Public version of Google's Knowledge Graph
- Information extracted from Wikipedia and Web
- ~~Accessible through API~~ R.I.P.
- Old dump will be available on ADA cluster

From Freebase to Wikidata: The Great Migration

Thomas Pellissier Tanon*
Google, San Francisco, USA
thomas@pellissier-tanon.fr

Denny Vrandečić
Google, San Francisco, USA
vrandecic@google.com

Sebastian Schaffert
Google, Zürich, Switzerland
schaffert@google.com

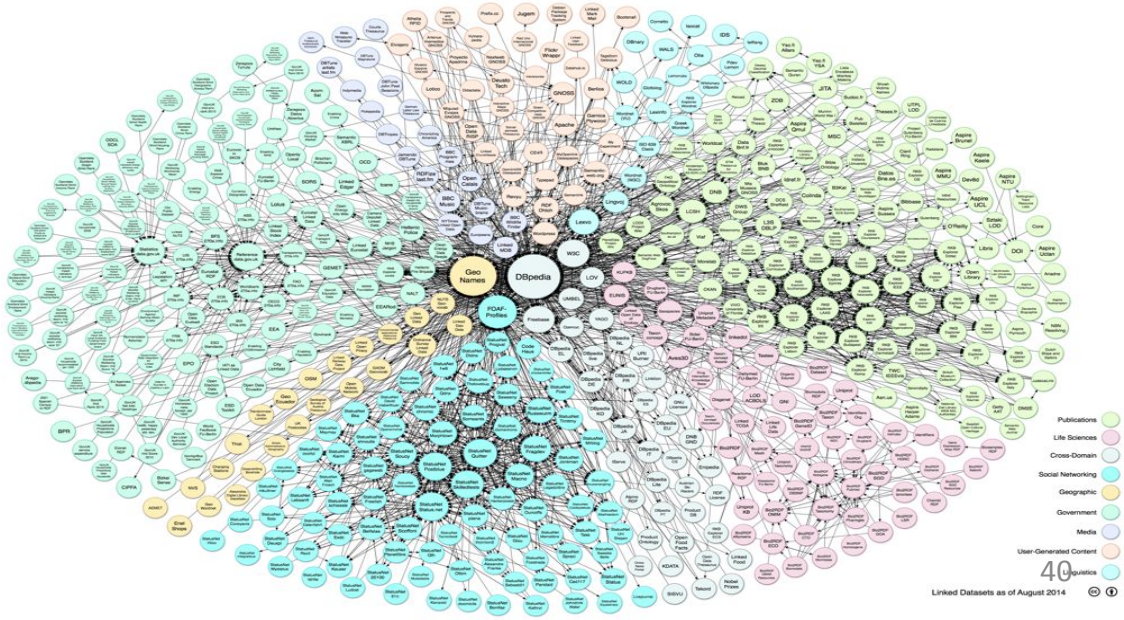
Thomas Steiner
Google, Hamburg, Germany
tomac@google.com

Lydia Pintscher
Wikimedia, Berlin, Germany
lydia@pintscher.de

Tying it all together: Linked open data

Repository of open data and knowledge bases and tools

https://www.w3.org/wiki/DataSetRDFDumps



Linked open data examples

DataSetRDFDumps

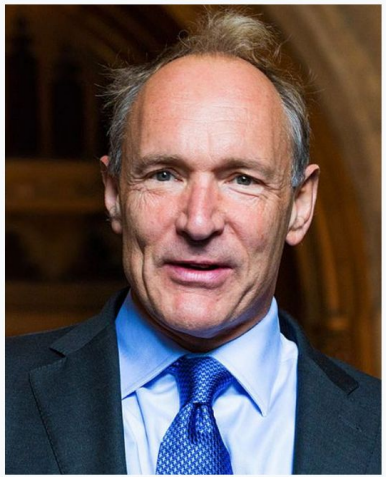
Linked Data Sets (i.e., with Dereferenceable URIs) available as RDF Dumps

- Please provide the URL for the directory containing the RDF dump files.
- Please try to have one directory or tarball per dump set -- such that we can retrieve and load the entire URL contents, to have a restored snapshot.
- Please include a Publisher/Maintainer URI, for use in constructing attribution triples.

Project	Data Exposed	Size of Dump and Data Set	Archive URL	Publisher / Maintainer URI
Addgene	Addgene catalog (tab delimited file)	1.1 MB	tab-delimited file	
Allen Brain Atlas	Science Commons extract from ABA Web site, on or shortly before 26 Feb 2007	51 MB	dump file	
Airport Data	SPARQL	754,585		Rob Styles
BAMS	BAMS	5.6 MB	bams-from-swanson-98-4-23-07.owl	
BBC John Peel sessions from DBtune.org	holding data released during Hackday, 2007	277,000 triples	peel.tar.gz	[URI?]
BBOP	All OBO ontologies	36 MB	obo-all.tgz	[URI]
BBOP	selected OBO ontologies, downloaded ~21 April 2007, augmented with inferred relations	2.6 MB	obo-in-owl.tgz	[URI]
Billion Triples Challenge Dataset 2008	various dumps	1 billion triples	download page	
Billion Triples Challenge Dataset 2009	crawled Web data	1.14 billion triples	download page	
Billion Triples Challenge Dataset 2010	crawled Web data	3.2 billion triples	download page	
Bio2RDF	various bio- and gene- related datasets	10+ billion triples	download page	
Bitzi	collaborative file describing service	330,026 discrete files, 270MB uncompressed	dump directory	
British Geological Survey (BGS) OpenGeoscience	1:625 000 Geology of the UK (DigMapGB), BGS geochronology and chronostratigraphy, BGS Lexicon of Named Rock Units	Approx 840,000 (19 MB compressed) N-Triples	data_bgs_ac_uk_ALL.zip	British Geological Survey (BGS)
	Chef Moz	290344 restaurants - 104856 reviews - 59243 links to reviews - 2402 editors	size?	URL?
Data-gov Wiki	Datasets containing RDF data converted from datasets published at http://data.gov (and other sources). The datasets are clustered by dc:subject, e.g. government budget, environmental statistics, housing and population statistics, medical cost, energy consumption, public library statistics, and labor statistics.	5+ billion triples	dump directory	Tetherless World Constellation
DBpedia	Data set containing extracted data from Wikipedia. About 2.6 million concepts described by 247 million triples, including abstracts in 14 different languages	247 million triples	dump directory	

Sir Tim about Linked Open Data

1. All kinds of conceptual things, they have names now that start with HTTP (a.k.a. **URI = Uniform Resource Identifier** ~ URL).
2. If I take one of these HTTP names and I look it up, I will get back some **data in a standard format** which is kind of useful data that somebody might like to know about that thing, about that event.
3. When I get back that information it's not just got somebody's height and weight and when they were born, it's got **relationships.** And when it has relationships, whenever it expresses a relationship then the other thing that it's related to is given one of those names that starts with HTTP.



Schema.org: microformats for Web pages

Collaborative, community activity to create, maintain, and promote schemas for structured data embedded within Web pages

- Sponsoring companies: **Google, Microsoft, Yahoo!, and Yandex**

Schema.org: microformats for Web pages

```
<div itemscope itemtype="http://schema.org/Movie">
  <h1 itemprop="name">Avatar</h1>
  <div itemprop="director" itemscope itemtype="http://schema.org/Person">
  Director: <span itemprop="name">James Cameron</span>
(born <time itemprop="birthDate" datetime="1954-08-16">August 16, 1954</time>)
  </div>
  <span itemprop="genre">Science fiction</span>
  <a href="../movies/avatar-theatrical-trailer.html" itemprop="trailer">Trailer</a>
</div>
```

- Two type hierarchies: `itemtype`, `itemprop`
- Core vocabulary currently: 642 types, 992 properties
- Used by other knowledge bases, e.g., Wikidata, Freebase

This brings us to…
HTML and (the) REST

HTML is everywhere

- Common Crawl dataset, about 1.82 billion web pages -- huge!
- (… but less than 0.1% of Google's Web crawl, as of 2015)
- 145 TB, hosted on Amazon S3, also available for download
- Includes link data, PageRank, and various preprocessed formats
- In ARC (Internet Archive) File format

So there's plenty of ready-made data

… but if you're hungry for more: many crawlers for getting your own: Apache Nutch, Storm, Heritrix 3, Scrapy, etc.

Useful HTML tools for this course

Beautiful Soup http://www.crummy.com/software/BeautifulSoup/
a Python API for handling real HTML. DOM or SAX interfaces.

Requests http://docs.python-requests.org/en/master/
an elegant and simple HTTP library for Python, built for human beings.

Scrapy https://scrapy.org/
an open-source framework to build Web crawlers.

Plain ol' /regular/expres*ion/s…

How to scrape EPFL profiles with BeautifulSoup

```
def profile_extractor(query):
  profile = {}

  r = requests.get('http://search.epfl.ch/psearch.action?request_locale=en&q=%s' % query)
  soup = BSoup(r.text.encode('utf-8'), 'lxml')

  profile['Name'] = soup.find('div', class_='presentation').h4.string.replace(' ', ' ')
  profile['Email'] = soup.find('div', class_='presentation').script.string.split('msgto(\'')[1].split('\'')[0] + '@epfl.ch'
  profile['SCIPER'] = soup.find('div', class_='button vcard').a['href'].split('id=')[1].split('&')[0]
  return profile


def main():
  query = ' '.join(sys.argv[1:])
  for k,v in profile_extractor(query).items():
    print('%s' % v)
```

Web services

Most large web sites today actively discourage screen-scraping to get their content, and provide Web service APIs instead.

This is the **right** way to get data from online sources

Web services

W3C definition:

a "Web service" is "a software system designed to support interoperable machine-to-machine interaction over a network".

Two kinds:

- ~~XML-based RPC-style messages: SOAP~~
- REST-style stateless interactions, URLs encode state

Can run over different transports, but usually HTTP

Examples

Twitter: REST API and streaming API with JSON content. Provides sampling, searching and filtering capabilities.

Amazon: has a “product advertising API” in XML with a WSDL spec. Includes product search, reviews etc.

Netflix: Javascript, Atom and REST interfaces.

Ebay: Many APIs for searching, buying and posting. WSDL descriptions, client code in Java and .NET

Flickr: Comprehensive API set, free for non-commercial use. REST, XML-RPC, SOAP, with client code in many languages.

vBulletin: REST interface, most actions supported

REST

REpresentation State Transfer

Stateless client/server protocol. Principles:

1. Each message in the protocol contains all the information needed by the receiver to understand and/or process it. This constraint attempts to "*keep things simple*" and avoids needless complexity

2. Set of uniquely addressable resources
   - "*Everything is a resource*" in a RESTful system (recall Sir Tim)
   - Requires universal syntax for resource identification (e.g. URI)

REST

3. Set of well-defined operations that can be applied to all resources
   - The primary methods are **POST, GET, PUT, DELETE**
   - These are similar to the database notion of CRUD (Create, Read, Update, Delete)

4. The use of hypermedia both for application information (query results) and state transitions
   - Resources are typically stored in a structured data format that supports hypermedia links, such as XHTML or JSON

REST example

```
{
  "user": {
    "name": "Jane",
    "gender": "female",
    "location": {
      "href":
"http://www.example.org/us
/ny/new_york",
      "#text": "New York"
    }
  }
}
```

This resource is a description of a user named Jane

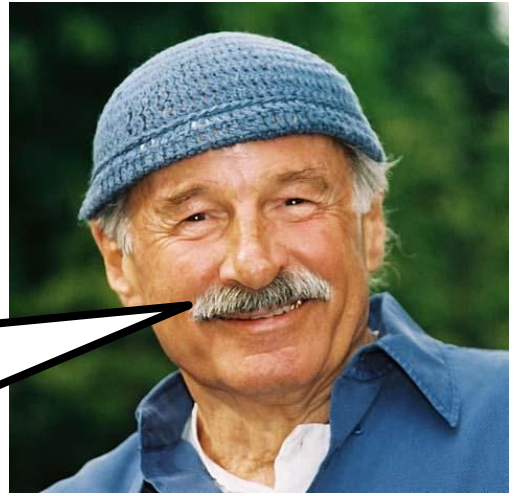
It might live at http://www.example.org/users/jane/

- If a user needs information about Jane, they **GET** this resource
- If they need to modify it, they **GET** it, modify it, and **PUT** it back
- The href to the location resource allows savvy clients to gain access to its information with another simple **GET** request

Implication: Clients cannot be too “thin”; need to understand resource formats!

Joe Zawinul

I said, “What’s that?” and he said, “That’s **jazz**.” “How do you write that?” And he spelt it out. Somehow I saw my name in there, and I liked this word.



Robert West

I said, “What’s that?” and he said, “That’s **REST**.” “How do you write that?” And he spelt it out. Somehow I saw my name in there, and I liked this word.

Combining data sources: From cookies to cooks

Q: What d

- Find Amazon add-to-cart events heuristically in logs:

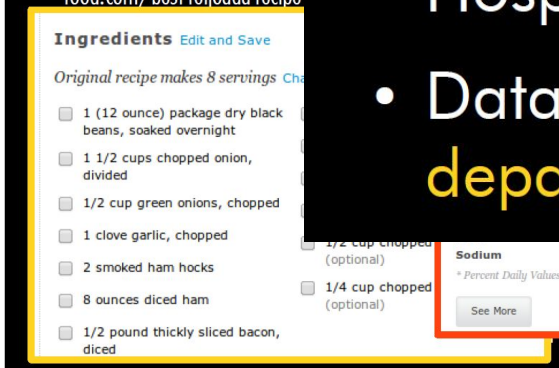
  Referrer: http://www.amazon.com/Forks-Over-Knives-Plant-Based-Health/dp/1615190457

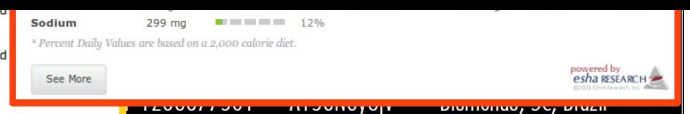
  URL: http://www.amazon.com/gp/cart/view-upsell.html?...
- Get product info for product id using Amazon API
- Consider all add-to-cart events for category "Diets & Weight Loss"

Our method: I

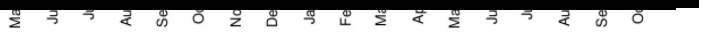
Consenting IE users, 18 mo

URL	Referrer	Tim
yahoo.com?q=the+onion	yahoo.com	128
theonion.com	yahoo.com?q=the+onion	128
theonion.com/Area-Man-Sad		
bing.com		
bing.com?q=feijoada+recipe		
allrecipes.com/tasty-feijoada		
food.com/best-feijoada-recipe		





- Collaborated with clinician at Washington Hospital Center, Washington, D.C.
- Data: All CHF admissions to emergency department for time period of our browsing logs



[Paper with results and plots]

Feedback

Give us feedback on this lecture here:
https://go.epfl.ch/ada2018-lec3-feedback

- What did you (not) like about this lecture?
- What was (not) well explained?
- On what would you like more (fewer) details?
- What is your credit card number?
- …