APAN PS5430
Applied Text & Natural Language Analytics
Week 2: Data Crawling & Corpus Building

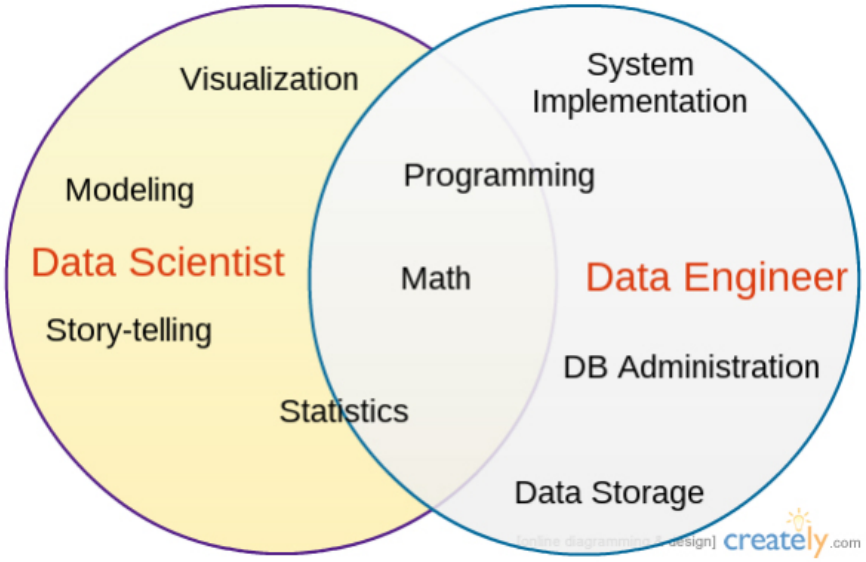
Javid Huseynov, Ph.D.
Thursday, January 30, 2020

Week 2 Agenda

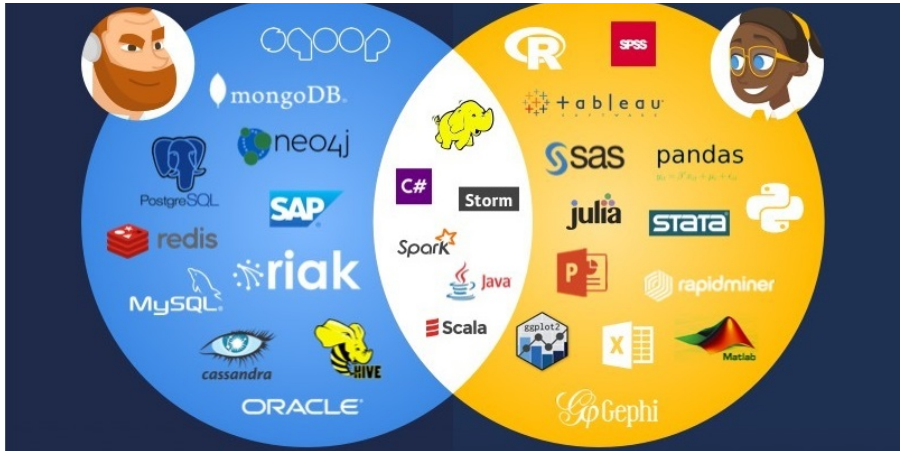
- Data Science vs Data Engineering
- Understanding the ETL Process
- Unstructured Data Sources
- NLTK Text Corpora
- File Formats
- Data Warehouses
- REST API
- Python libraries
- Python Flask
- Class Exercise: Webhose.io Data Acquisition

Data Science vs Data Engineering

- **Data Science**: given a data source design models, algorithms, methods to extract insights or meaning
- **Data Scientists** can come from any scientific background with a solid expertise in math and statistics

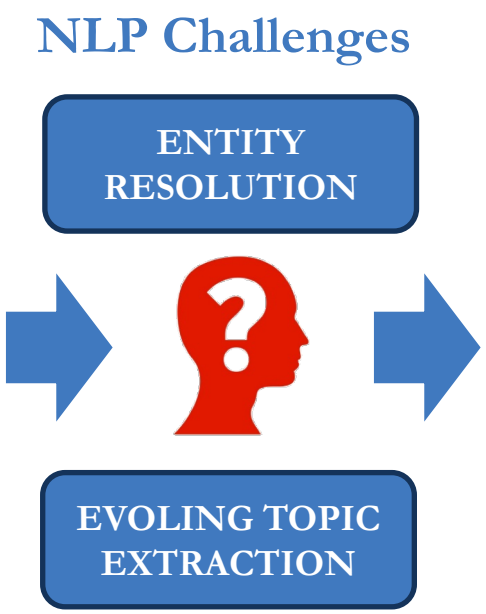


- **Data Engineering**: given a data source design systems or infrastructure to enable data science
- **Data Engineers** typically possess strong computer science background



Applied Text Analytics: Example Use Case

- Discover insights about business entities from **publicly-available unstructured data**

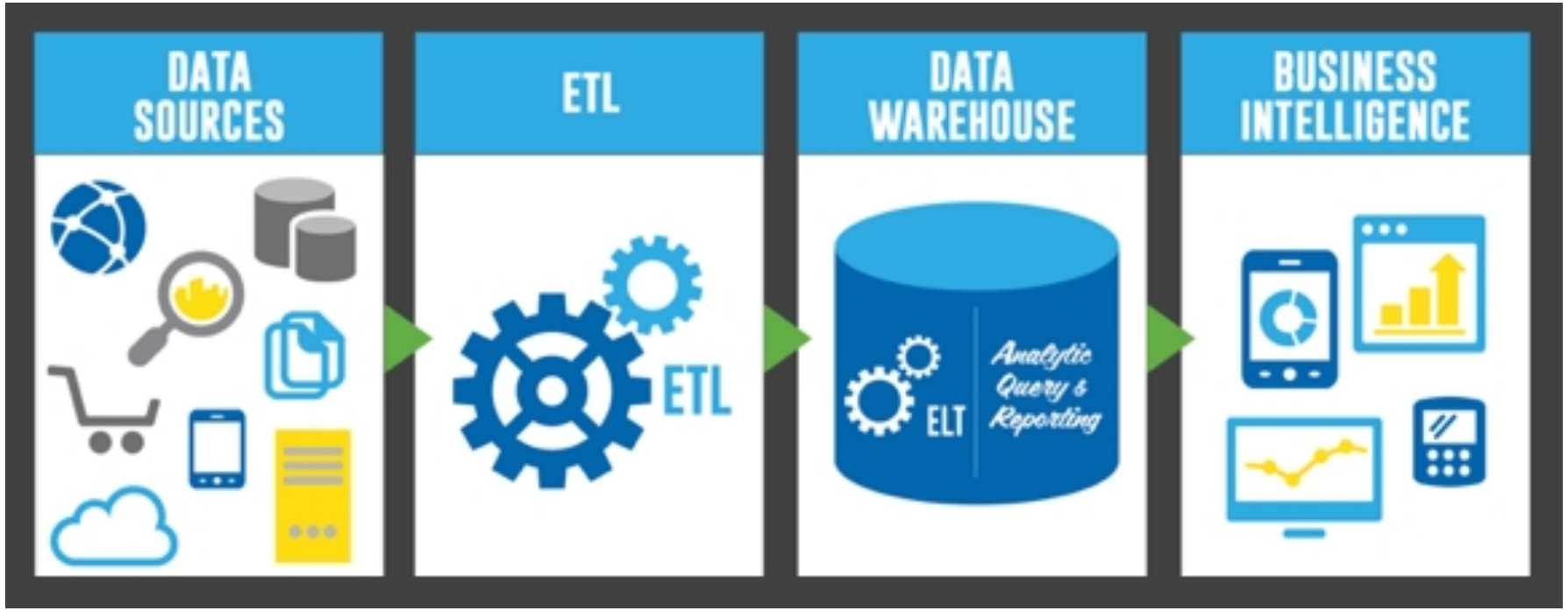


Use Cases	Benefit
Sales & Marketing	Relevant product recommendations for companies
Market Analysis	Identify trends for company or market segment growth
Trust & Compliance	Aggregate adverse information about companies
Client Credit Financing	Discover insights for the company risk assessment

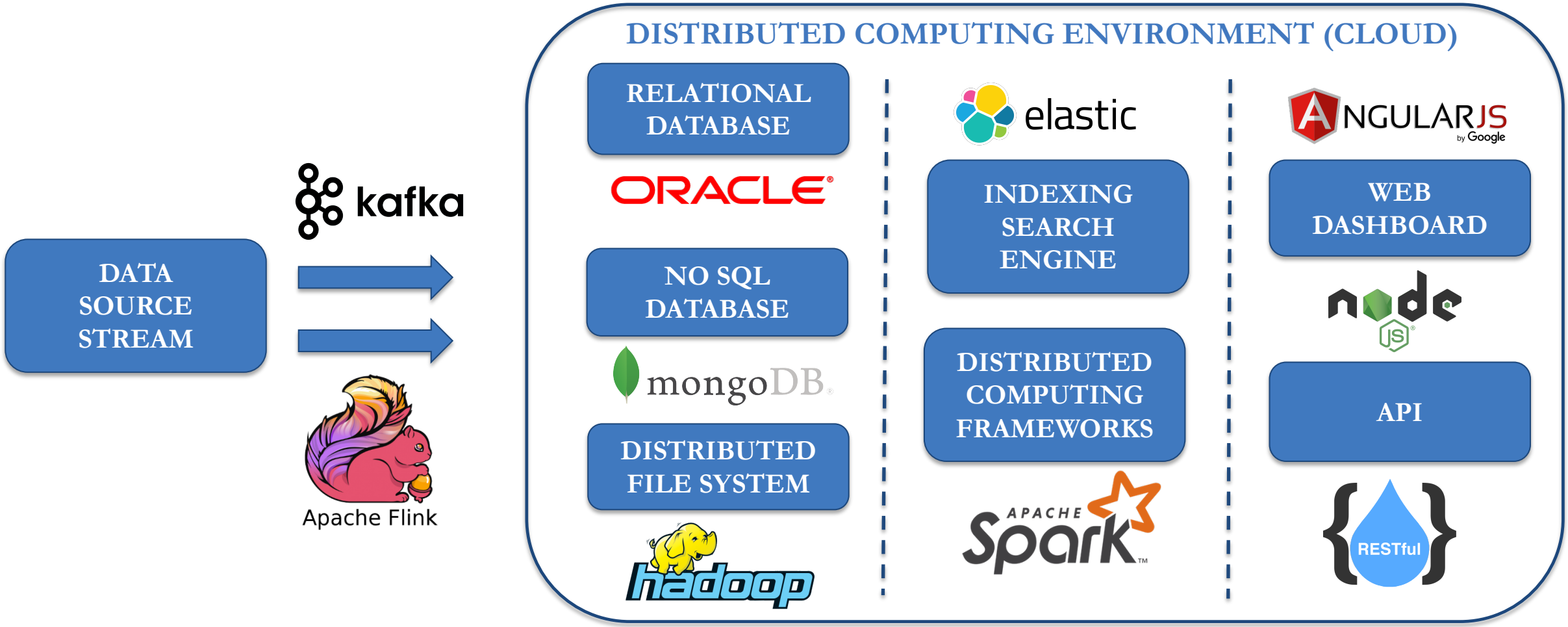
- **Challenge: How do we obtain the data?**

Extract-Transform-Load (ETL) Process

- **Extract** data from homogenous and heterogeneous data sources
- **Transform** data for the purpose of querying and analysis
- **Load** data into a target data store, data mart, or data warehouse



Sample ETL Technology Stack



Available Unstructured Data Sources for Analytics

News & Web Crawl Data

- Webhose.io
- NewsAPI
- NewsRiver API
- Scrapy
- New York Times APIs

Government Data

- US Government Open Data
- City of Seattle Open Data
- New York State Open Data
- Legiscan: US Congress & States

Financial Data

- Securities & Exchange Commission (SEC) Edgar API
- Quandl Financial Data APIs
- IEX Trading Data APIs

Firmographic Data

- Crunchbase Open Data Map
- OpenCorporates API

Healthcare Data

- HealthData.gov

NLTK Corpora and CorpusReaders

- Corpus - a large body of text comprising of multiple domain-specific documents, such as legal, medical, fiction, literature, etc.
- Come with a variety of built-in functions such as stats, counting,
- NLTK provides 66+ corpus readers for various types of domain corpora:
  - PlaintextCorpusReader
  - TaggedCorpusReader
  - ChunkedCorpusReader
  - TwitterCorpusReader
  - XMLCorpusReader

- More in NLTK Chapter 2: http://www.nltk.org/book/ch02.html

How can we obtain and store text data?

- Bulk Downloads
- REST API Services
- Unstructured Data Warehousing

- Dealing with
  - Different text formats
  - Different character encodings

Relational DBMS

- Schema and Table based
- Queries using SQL
- Support *join* operations
- Support ACID transactions
- No horizontal scaling
- Examples:
  - Oracle
  - Microsoft SQL Server
  - IBM DB2

NoSQL DBMS

- Schema-Free
- Document-Based
- No *join* or ACID
- Horizontal Scaling
- Map-Reduce Support
- Examples:
  - MongoDB
  - Cassandra
  - Neo4j

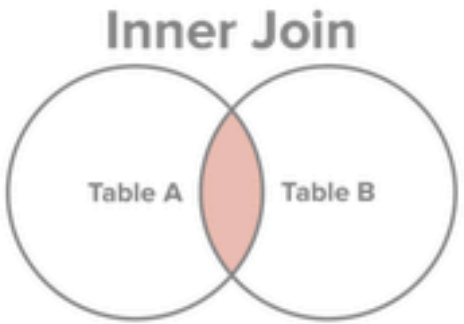
Indexing Engines

- Schema-Free
- Document-Based
- Can run on top of DB
- Fast Search
- Horizontal Scaling
- Examples:
  - Apache Lucene
  - Elasticsearch

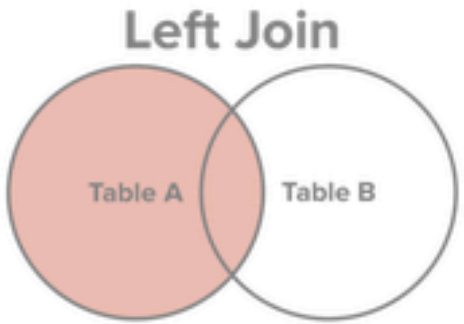
Relational Databases

- Schema/Table
- Join operations
- Not scalable

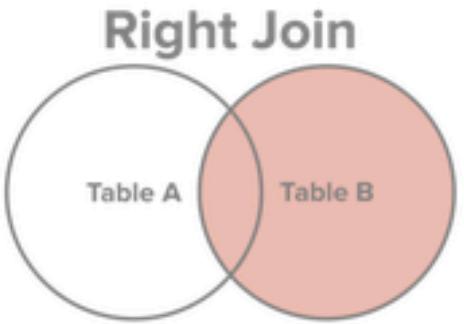
- Atomicity, Consistency, Isolation, Durability (ACID) Transactions
- Fast query on structured data



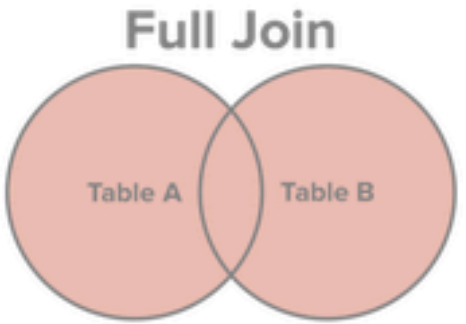
Select all records from Table A and Table B, where the join condition is met.



Select all records from Table A, along with records from Table B for which the join condition is met (if at all).



Select all records from Table B, along with records from Table A for which the join condition is met (if at all).



Select all records from Table A and Table B, regardless of whether the join condition is met or not.

NoSQL Databases

- Types:
  - Key-Value Stores
  - Document Databases (JSON | XML)
  - Wide-Column Stores
  - Graph Databases
- Horizontal Scaling (Sharding)
- Object oriented APIs
- Map-Reduce (MongoDB)

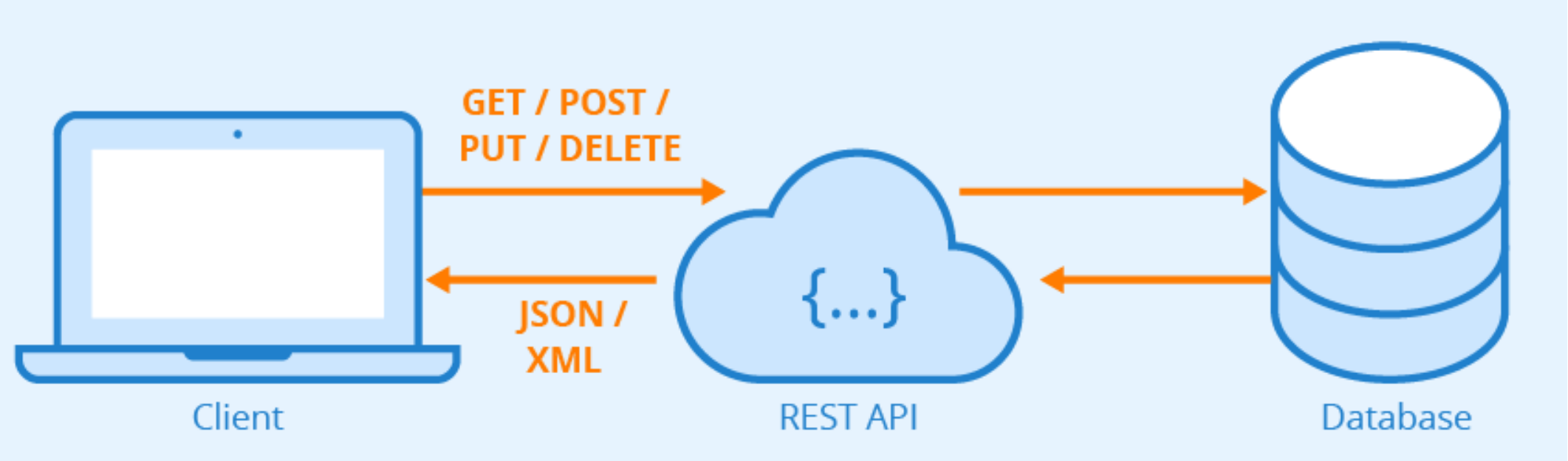
```
{
  "firstName": "John",              -- String Type
  "lastName": "Smith",              -- String Type
  "isAlive": true,                  -- Boolean Type
  "age": 25,                        -- Number Type
  "height_cm": 167.6,               -- Number Type
  "address": {                      -- Object Type
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021-3100"
  },
  "phoneNumbers": [                  // Object Array
    {                               // Object
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "office",
      "number": "646 555-4567"
    }
  ],
  "children": [],
  "spouse": null                    // Null
}
```

Popular File Formats for Textual Data

File Type	Extensions	Description
HyperText Markup Language	.html, .htm	An established standard maintained by the World Wide Web Consortium (W3C) for encoding web pages on the Internet.
Extensible Markup Language	.xml	Created by the World Wide Web Consortium (W3C) to define a syntax for encoding documents that both humans and machines could read
Portable Document Format	.pdf	Developed by Adobe to present documents, including text formatting and images, in a manner independent of the platform
Comma-Separated Values	.csv	Delimited text file that uses a comma to separate values in tabular format
Plain Text	.txt	Stores plain text with no special formatting beyond basic fonts and font styles
JavaScript Object Notation	.json	Open-standard file format for transmitting data objects consisting of attribute–value pairs and array data types
Microsoft Word	.doc, .docx	Standard for storing texts in a proprietary Microsoft Word Binary File Format
Microsoft Excel	.xls, .xlsx	Proprietary Microsoft Binary Interchange File Format (BIFF) file format for storing spreadsheets

REST API Service

- Representational State Transfer (REST) – software architectural standard for web services



Python **`requests, urllib`** and **`json`** libraries

`requests` library

- Unofficial standard for making HTTP requests in Python
- Supports REST standard requests: **`GET,POST,PUT,DELETE`**

`urllib` library

- To scrape/read web pages from Python program

`json` encoder & decoder library

- Used for conversions between text strings and JSON objects
- Supports JSON **`dumps, dump, load, loads`** commands

Python Flask for developing REST API

- Popular microframework for web applications
- Included in Anaconda or pip install
- Essential for building interactive web-based dashboards
- Can run on top of Apache or Nginx
- More at: http://flask.pocoo.org/

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
   return 'Hello World!'

if __name__ == '__main__':
   app.run()
```

http://localhost:5000