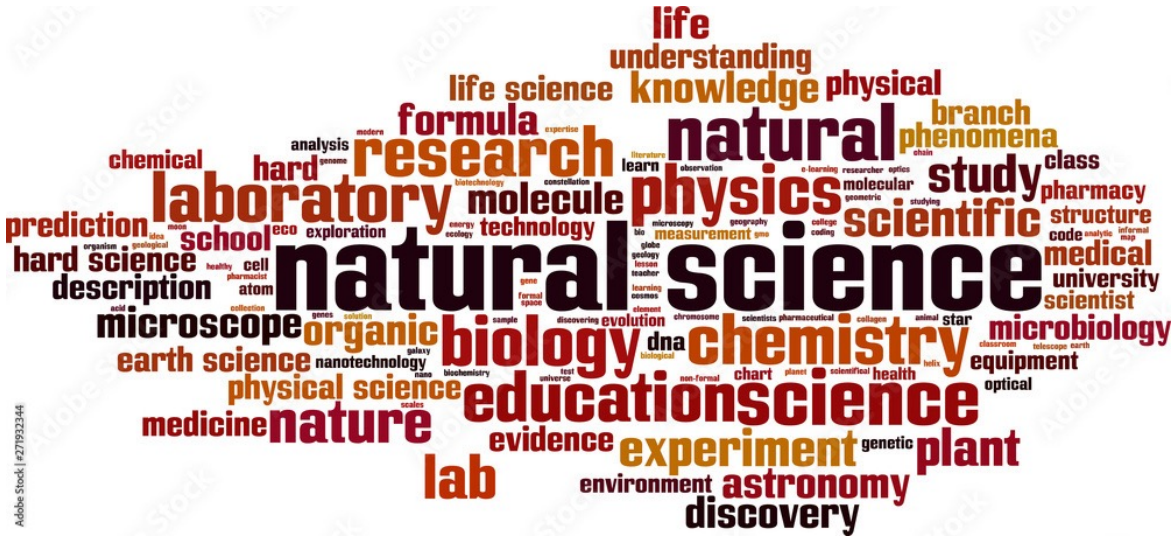
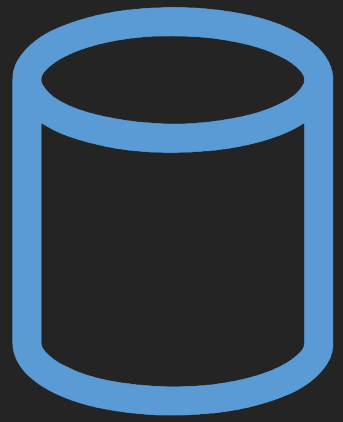
Data Management for Scientific Application Using Semi structured Data



Subhasis Dasgupta, Ph.D.

Assistant Scientist, University of California San Diego

sudasgupta@ucsd.edu



Outline of the presentation.

- An Overview of Data Management and Changes in Data Requirements
- Structured Data versus Semi-structured Data
- Application and Example
- Understand data management using systems and code.
- Discussion and open session

Data/Knowledge Management Goals

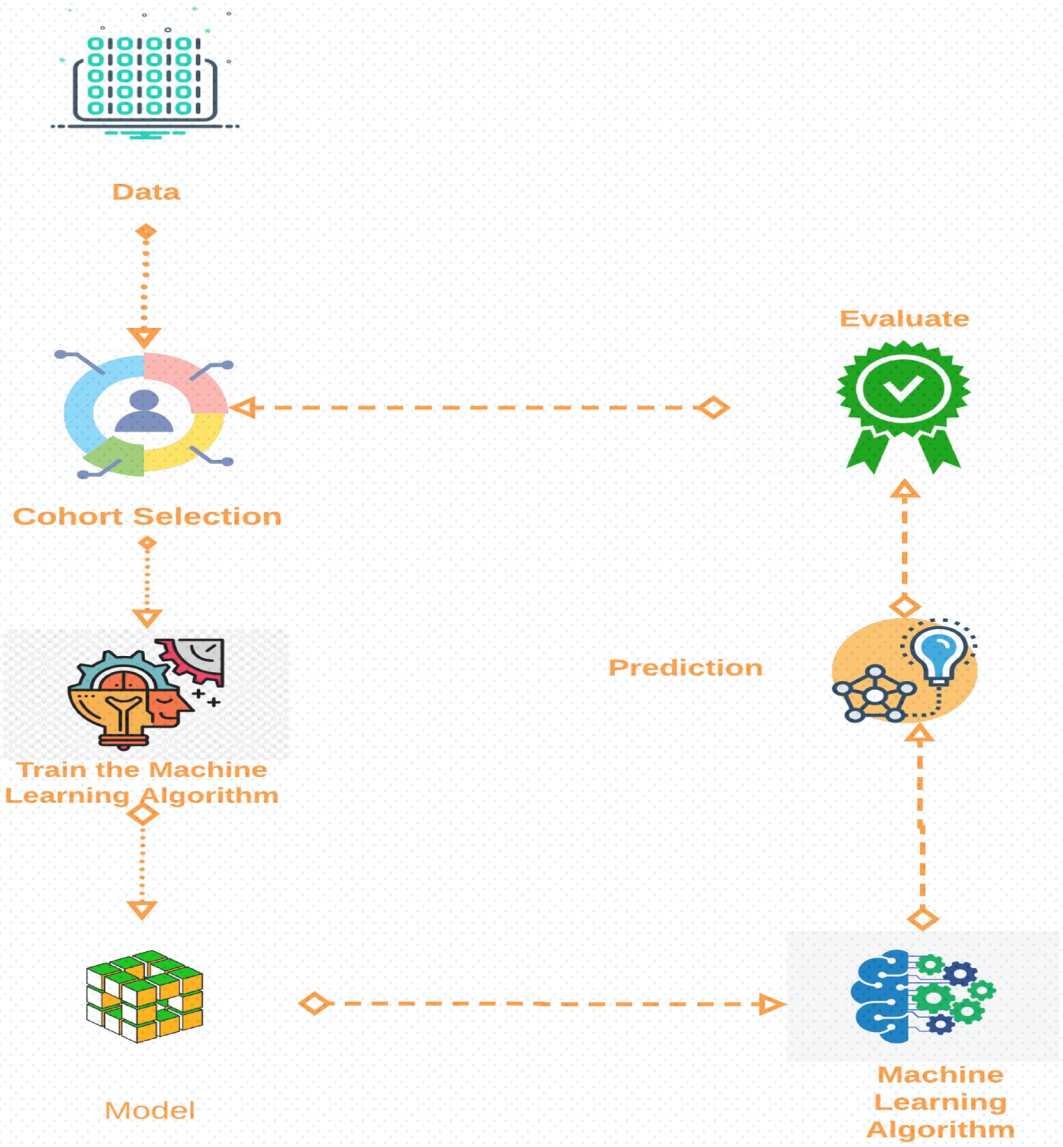
- There is a vast amount of data available from diverse sources, including various cohorts, demographics, and features.
- In a perfect world, the data is expertly ingested, meticulously modeled, seamlessly indexed, and efficiently processed to guarantee effortless searching.
- Efficient exploration is an absolute necessity for achieving faster innovation.
- Developing effective knowledge management systems for your research group is essential and should be prioritized to prevent any potential delays or setbacks.
- Choosing the appropriate management stack is necessary.

A typical ML workload

Working with machine learning data and selecting cohorts can be complex when using a file-based system.

In-memory DataFrame technologies such as Pandas or Dask can support cohort selection, but for large volumes of data, they may be inefficient regarding resource utilization.

While using a database for data retrieval can be effective, traditional databases may not offer the level of flexibility that some situations require.

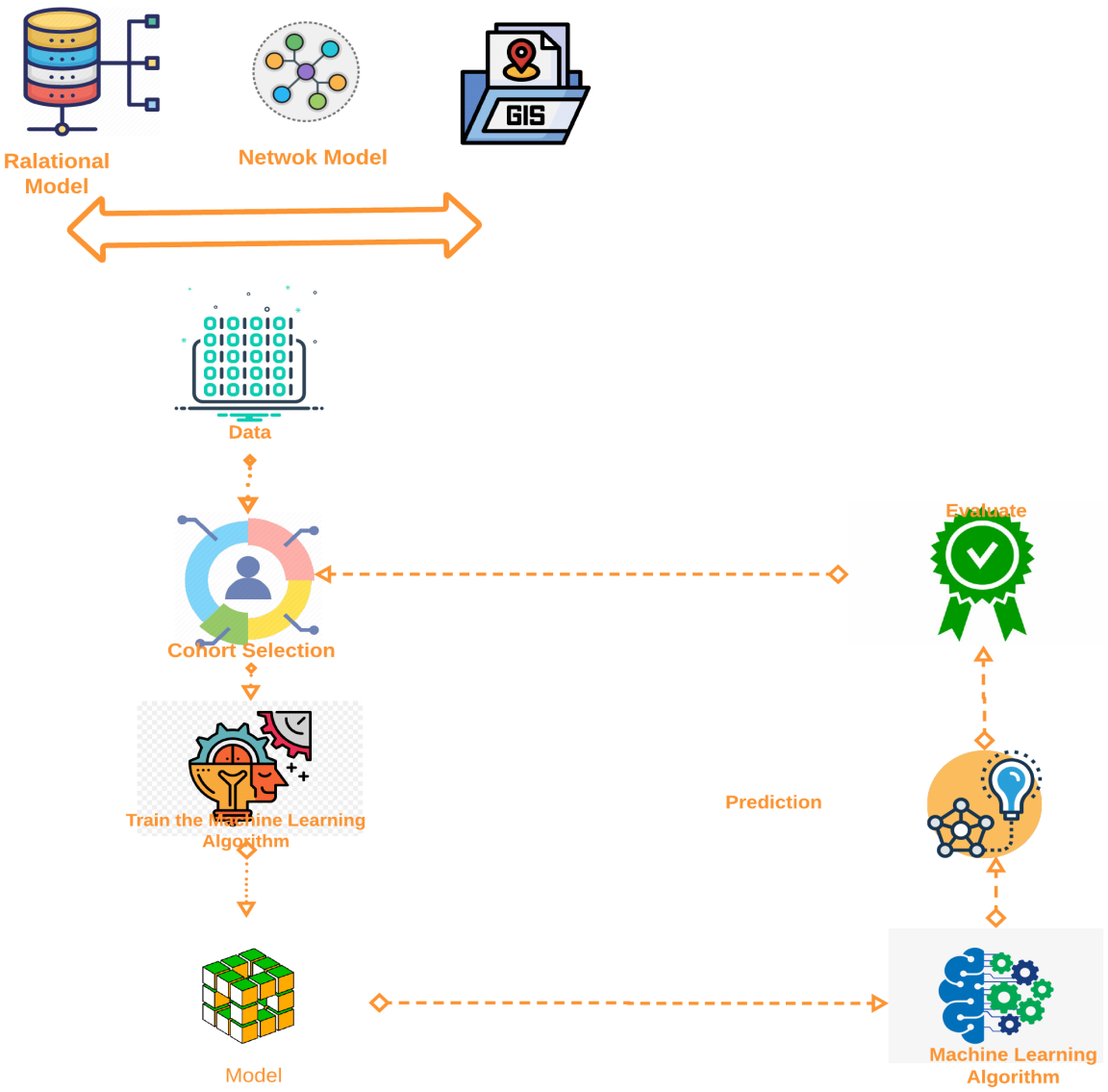


A typical ML workload

Data can come from different sources with varying structures, such as relational tables, temporal data, or network data. Creating a unified model can be very expensive or impossible.

It is essential to capture both direct and indirect relationships between entities. To validate data, researchers frequently rely on analytical and logical operations.

Establishing relationships and resolving entities require the use of semantic and schematic mapping. To ensure efficient and swift retrieval, it is crucial to implement appropriate indexing.



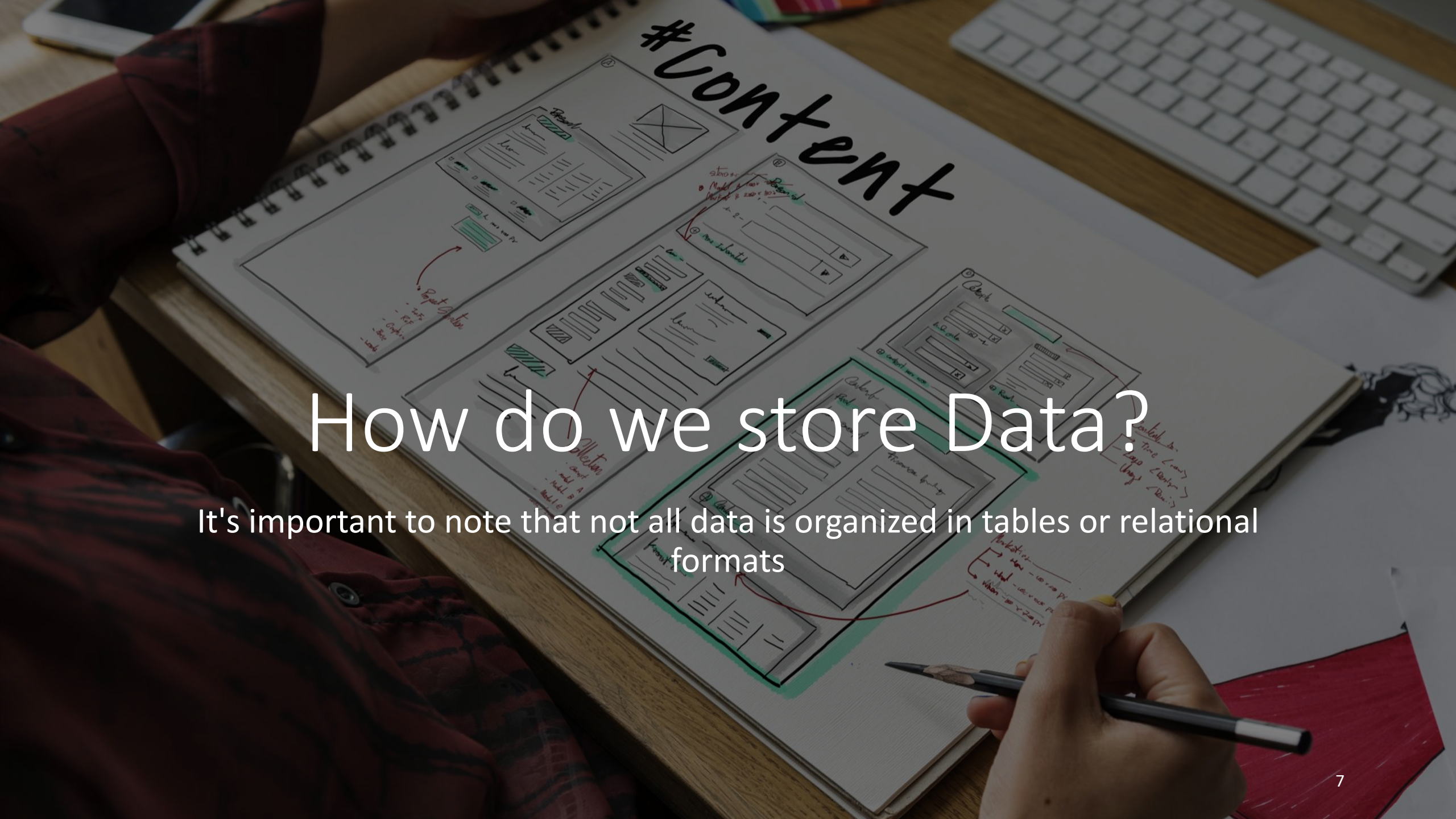
Relational Database

Each record in the table has the same number of fields, and the schema structure is strict.

Each field is closely associated with a specific data type. (for example, text, integer, float, etc.)

Each of the data types has a specific length and precision.

Modern relational databases can handle JSON or XML data, but there may be performance and capability concerns.



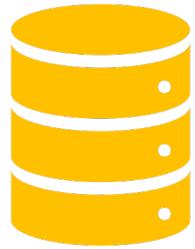
How do we store Data?

It's important to note that not all data is organized in tables or relational formats

Comma Separated Value (CSV) File

Each value is separated by a comma (except plain text)

No Specified field lengths.



The total number of CSV datasets on Kaggle is 115,936, but 21,181 JSON datasets are available.

CSV format Problem

The CSV file format lacks complete standardization, with the only standardized rule being the use of the comma.

Representing a dynamic schema can be quite challenging due to its large and complex nature.

Advantages of Semi Structure Representation

01

Structure Data: Organized according to a formal data model(i.e., relational)

02

Semi-structured Data: No formal data model, but contains symbols to separate and label data element

03

Unstructured Data: No data model no pre-defined organization

Data Example

Relational: CSV, Relational Databases

Semi-structured: XML, JSON

Unstructured: Text, Document, Image

A Few Model Specific Databases

Relational DB

Graph DB

Search DB

Semi-structured DB

Timeserise DB

Spatial DB

Why is semi-structured data important even though CSV is the most popular format?

- *Heterogeneous data integration is achieved through input and output sources that can be of various types.*
- *The most accepted format for web services or machine learning libraries is used.*

XML VS JSON

Here is a comparison between the XML and JSON data outputted by the GridLab-D Simulator.

```
▼<property>
   <object>trip_meter13</object>
   <name>measured_real_energy</name>
   <value>+169411 Wh</value>
   <type>double</type>
 </property>
```

```
{       "object" : "trip_meter13",
        "name" : "measured_real_energy",
        "type" : "double",
        "value" : "+996682 Wh"
}
```

XML (Extensible Markup Language)

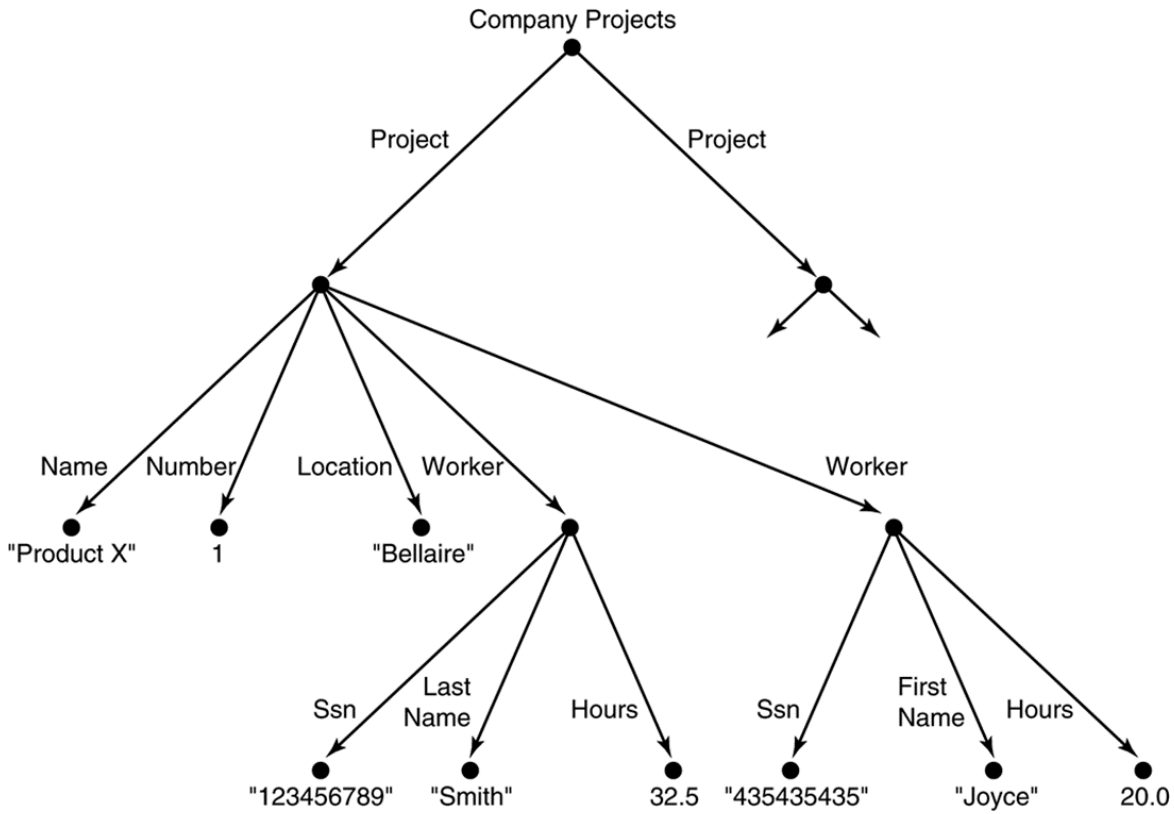
- Plain Text
- User text for values between tags for labels
- Value can be any length
- Commas and Quotes are valid
- Field can be skipped or create a hierarchy

```
▼<property>
   <object>trip_meter13</object>
   <name>measured_real_energy</name>
   <value>+169411 Wh</value>
   <type>double</type>
 </property>
```

JavaScript Object Notation(JSON)

- Plain Text
- Organized as objects within braces {}
- Uses key-value pairs
  - Keys are field names, and values are data strings, numbers, Boolean
- You have the option to skip the "Field" section.
- Is it possible to use a datatype such as datetime or a specific type of index like r-index?

Example Semi-Structured Data



A complex XML element called <projects>

```
<?xml version="1.0" standalone="yes"?>
<projects>

 <project>
  <Name>ProductX</Name>
  <Number>1</Number>
  <Location>Bellaire</Location>
  <DeptNo>5</DeptNo>
  <Worker>
   <SSN>123456789</SSN>
   <LastName>Smith</LastName>
   <hours>32.5</hours>
  </Worker>
  <Worker>
   <SSN>453453453</SSN>
   <FirstName>Joyce</FirstName>
   <hours>20.0</hours>
  </Worker>
 </project>
 </project>
  <Name>ProductY</Name>
  <Number>2</Number>
  <Location>Sugarland</Location>
  <DeptNo >5</DeptNo >
  <Worker>
   <SSN>123456789</SSN>
   <hours>7.5</hours>
  </Worker>
  <Worker>
   <SSN>453453453</SSN>
   <hours>20.0</hours>
  </Worker>
  <Worker>
   <SSN>333445555</SSN>
   <hours>10.0</hours>
  </Worker>
 </project>

...

</projects>
```

Example API Services

How to handle JSON Data?

File System with Python library? (You could have similar for R, Java, Go, rust, etc.)

Accessing the Data using dataframe: I will use Panda

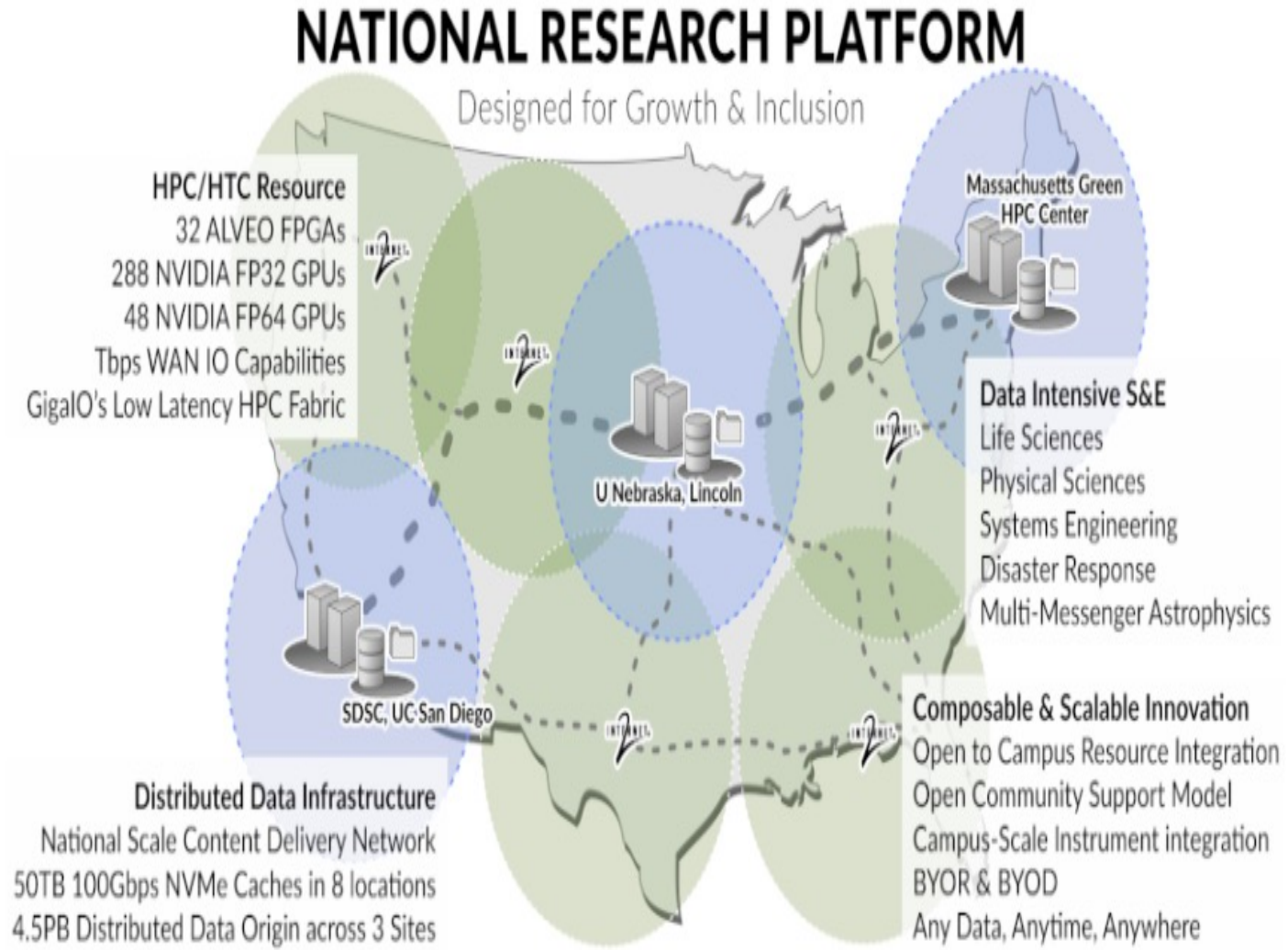
Can we use Relational Data?

Can we have a Semistructured Database?

Resources for Hands On

https://github.com/dsubhasis/tutorial-json.git


NRP
NATIONAL RESEARCH
PLATFORM


NATIONAL RESEARCH PLATFORM
Designed for Growth & Inclusion
HPC/HTC Resource
32 ALVEO FPGAs
288 NVIDIA FP32 GPUs
48 NVIDIA FP64 GPUs
Tbps WAN IO Capabilities
GigaIO's Low Latency HPC Fabric
Massachusetts Green HPC Center
U Nebraska, Lincoln
SDSC, UC San Diego
Data Intensive S&E
Life Sciences
Physical Sciences
Systems Engineering
Disaster Response
Multi-Messenger Astrophysics
Composable & Scalable Innovation
Open to Campus Resource Integration
Open Community Support Model
Campus-Scale Instrument integration
BYOR & BYOD
Any Data, Anytime, Anywhere
Distributed Data Infrastructure
National Scale Content Delivery Network
50TB 100Gbps NVMe Caches in 8 locations
4.5PB Distributed Data Origin across 3 Sites

Tech Publication and Patents

Dasgupta, S., K. Coakley, and A. Gupta. 2016. “Analytics-Driven Data Ingestion and Derivation in the AWESOME Polystore.” *2016 IEEE International*. https://ieeexplore.ieee.org/abstract/document/7840897/.

Dasgupta, S., C. McKay, and A. Gupta. 2017. “Generating Polystore Ingestion plans—A Demonstration with the AWESOME System.” *2017 IEEE International*. https://ieeexplore.ieee.org/abstract/document/8258297/.

Zheng, Xiuwen, Subhasis Dasgupta, Arun Kumar, and Amarnath Gupta. 2023. “An Optimized Tri-Store System for Multi-Model Data Analytics.” *arXiv [cs.DB]*. arXiv. http://arxiv.org/abs/2305.14391.

Gupta, A., and S. Dasgupta. (4th July,) 2023, Query processing in a polystore. *US Patent 11,693,856*, issued 2023. https://patents.google.com/patent/US20220083552A1/en.

Gupta, A., S. Dasgupta, and M. Roberts. 2022. Data ingestion into a polystore. *US Patent 11,288,261*, issued 2022. https://patents.google.com/patent/US11288261B2/en.

Significant Other Publications

1. Dasgupta, S., and A. Gupta. 2020. “Discovering Interesting Subgraphs in Social Media Networks.” In Social Networks Analysis and Mining …. https://ieeexplore.ieee.org/abstract/document/9381293/.

2. Mason, Ashley E., Frederick M. Hecht, Shakti K. Davis, Joseph L. Natale, Wendy Hartogensis, Natalie Damaso, Kajal T. Claypool, et al. 2022. “Author Correction: Detection of COVID-19 Using Multimodal Data from a Wearable Device: Results from the First TemPredict Study.” Scientific Reports 12 (1): 4568.

3. Purawat, Shweta, Subhasis Dasgupta, Luke Burbidge, Julia L. Zuo, Stephen D. Wilson, Amarnath Gupta, and Ilkay Altintas. 2021. “Quantum Data Hub: A Collaborative Data and Analysis Platform for Quantum Material Science.” In Computational Science – ICCS 2021, 656–70. Springer International Publishing.