CS 217 Data Management and Information Processing

Final Review

This is the Last Lecture!

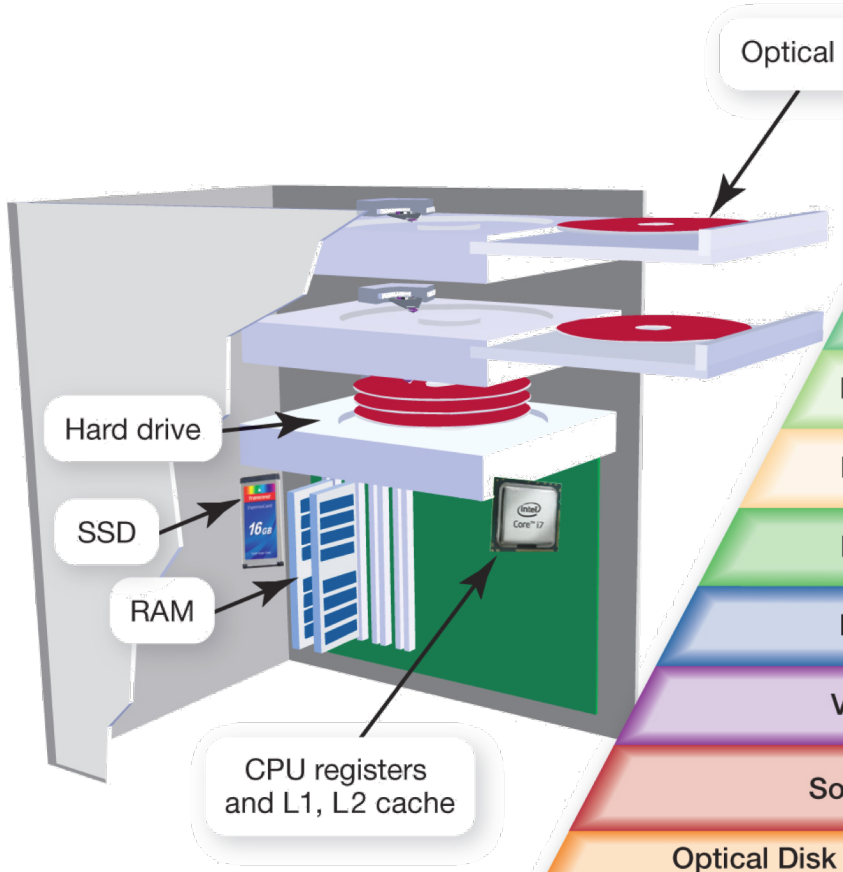
- Deadlines:
  - HW5-Data Models, due on June 4th
  - Project part 2, due on June 9th
  - Project presentation, due on June 12th
  - Q7, due at 9:30 am, June 4th
  - Q8, due at 9:30 am, June 11th

Goal of This Course

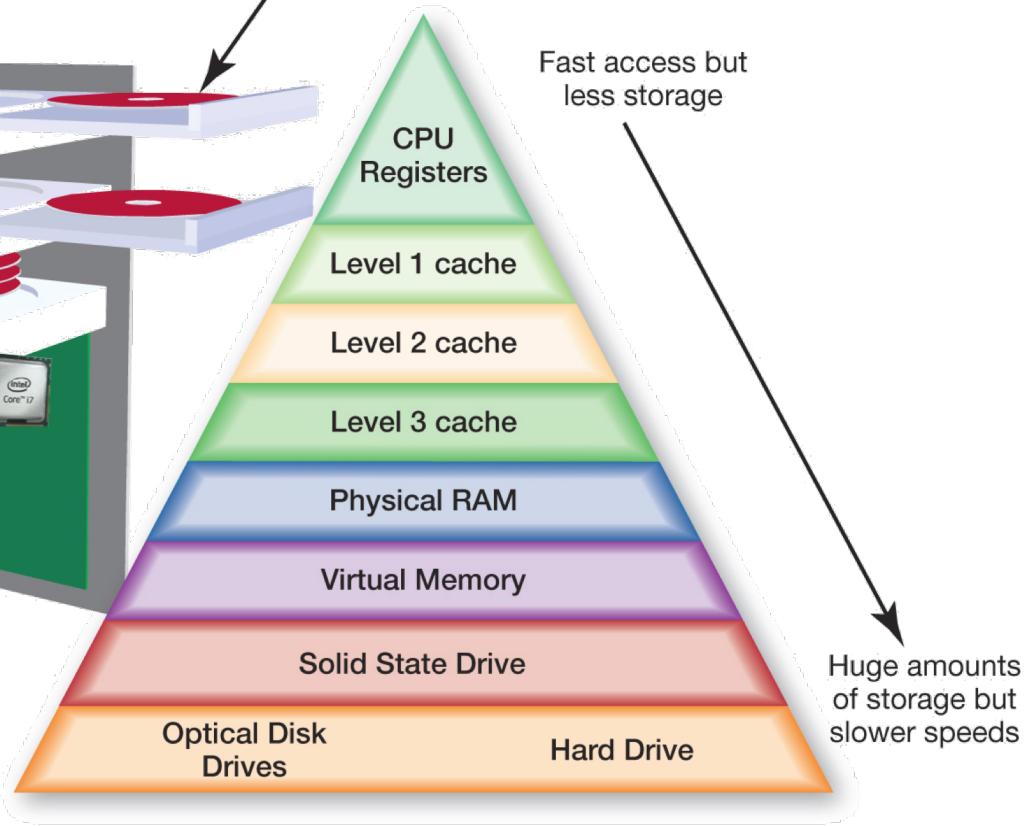
- How data is stored and organized
- How to collect data from multiple sources
- How to process data in different format
- How to pick the right tool

How Data is Stored and Organized

Physical Storage



Hierarchy of Storage



Data Encoding

- All data are stored as bits in the end
  - Bytes, KB, …
- Integers:
  - 2’s complement, usually in either 32 or 64 bits
- Real numbers:
  - Floating point representation.
- String:
  - ASCII and UTF-8 with different character set

Data Organization

- Different ways to organize data, depending on the complexity of the data relationship

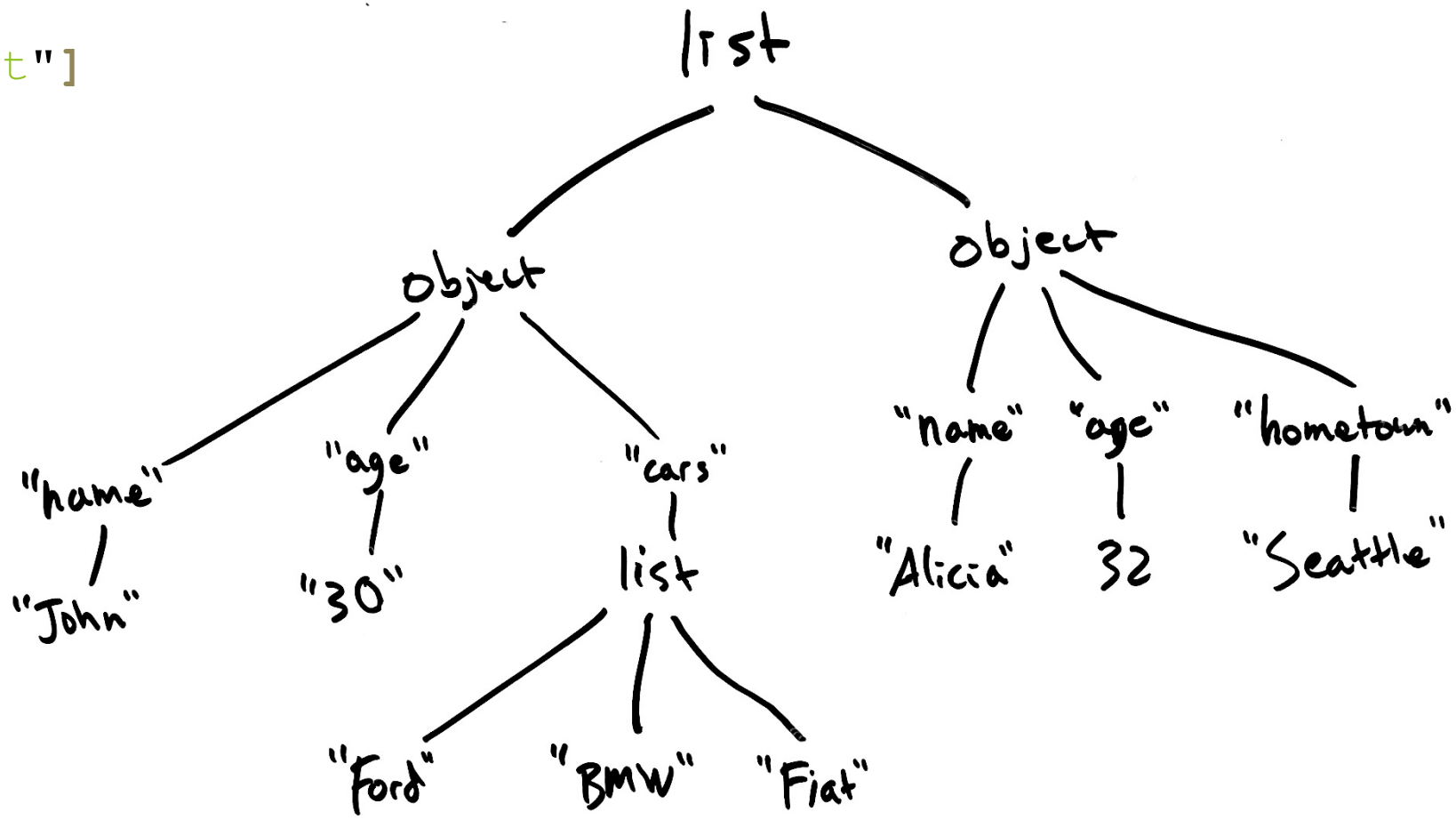
- Semi-structured data
  - Something that can be described in just one table
  - Usually stored in a CSV
- Structured data
  - Complicated inter-relationship between tables.
  - Consistency and integrity requirements

Semi-Structured Data

- Data model is simple, and thus a simple tool is sufficient
  - JSON files -- use json package to process
  - CSV files – use pandas or similar package to process
- Large data-science oriented tasks usually fall in this category

JSON

```
[
  {
    "name": "John",
    "age": 30,
    "cars":
       ["Ford", "BMW", "Fiat"]
  },
  {
    "name": "Alicia",
    "age": 32,
    "hometown": "Seattle"
  }
]
```



Pandas

Can essentially perform all SQL queries on a table

- A **Series** is a named Python list (one-entry dict with list as value):

```
{ 'grades': [50,90,100,45] }
```

- A **DataFrame** is a collection of Series (dict-like container for series):

```
{'names': ['bob', 'ken', 'art', 'joe'],
  'grades': [50,90,100,45]
 }
```

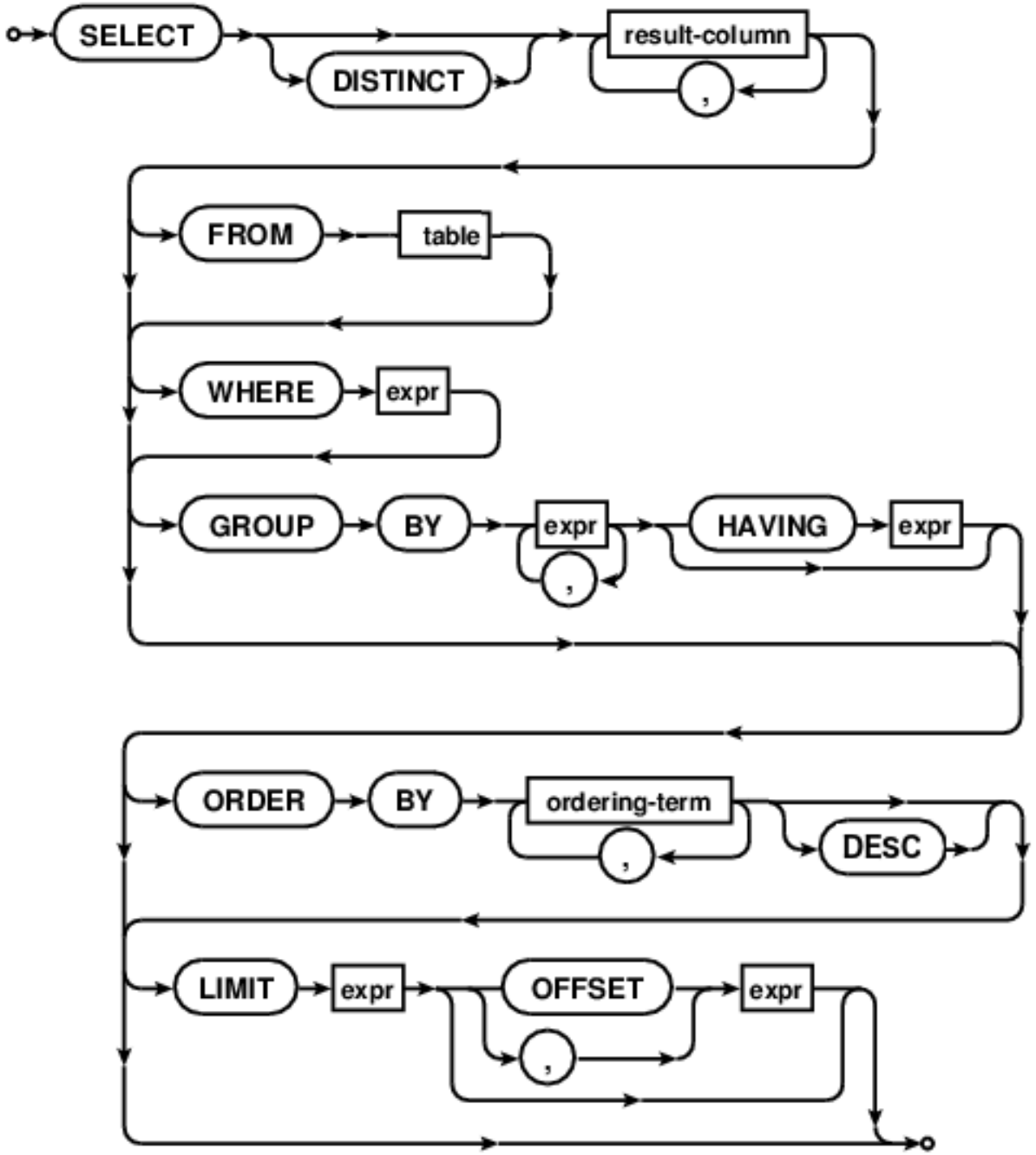
Structured Data

- Data relationship is rigorously defined.
  - Schema
- Access to the data is restricted to certain ways
  - Structured Query Language
- Updates to the database must comply the constraints in the schema

Schema

- Defines the tables, including:
  - **Columns** in each table (both the name and *type*)
  - **Primary key** for each table
  - **Foreign keys** that link tables
  - **Unique keys**
  - **Data type** for each column
  - …
- Tables can represent **objects**, **events**, or **relationships**

SQL


SELECT
DISTINCT
result-column
,
FROM
table
WHERE
expr
GROUP
BY
expr
,
HAVING
expr
ORDER
BY
ordering-term
,
DESC
LIMIT
expr
OFFSET
,
expr

Predicates

- Algebraic and logical expressions
- Aggregations like `SUM, MIN, MAX, AVG`
- String-Like
- Case condition
- Regular expression

Regular Expression

- Regular expressions are used to match text, both in SQL and in many other data management tools.
- A match anywhere in the text returns *true*.
- `^` anchors to the beginning
- `$` anchors to the end
- `.` matches any character
- `[…]` specifies a set of possible characters
- `[a-z]` hyphen specifies a range
- `[^abc]` carrot within brackets negates the match

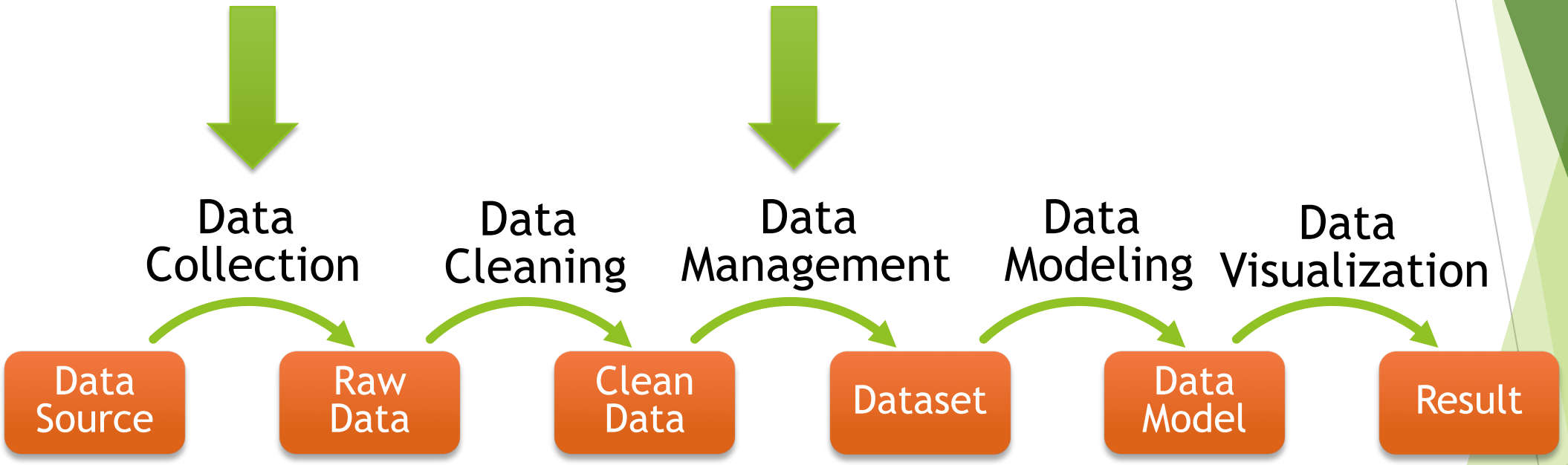
- Repetitions are supported:
  - `*` any number
  - `+` one or more
  - `?` zero or one
  - `{n,m}` n to m repetitions
- `|` pipe character gives OR
- `(…)` can be used for grouping

Very powerful and useful in many other applications too

Updates

- `CREATE TABLE …`
- `INSERT INTO …`
- `DELETE FROM …`
  - Three foreign key options for “ON DELETE”:
    - **Restrict** (don’t allow delete)
    - **Cascade** (delete children)
    - **Set NULL** (make orphan)
- `UPDATE …`
- `ALTER TABLE …`

Where do We Stand in the Data Science Pipeline



For comprehensive overview, take CS396 – Introduction to the data science pipeline!