Lecture 1:
Course Introduction

Importance of Database Systems

- Banking
- Healthcare
- Airlines
- E-Commerce

Why take this course?

Curiosity

Scalability

Efficiency

Versatility

Why take this course?

- Storage Management
- Query Optimization
- Index Structures
- SIMD Instructions

Course Overview

Course Objectives

- Learn about building a database system from scratch.
- Become proficient in systems programming.
- Understand the impact of hardware trends on software design.

Course Topics

- This course focuses on the internals of a database system.
    - Relational Databases
    - Storage Management
    - Index Structures
    - Query Execution

Next Course

- Course on advanced database implementation
  - Logging and Recovery
  - Concurrency Control
  - Query Optimization
  - Potpourri of advanced topics
- This course is a pre-requisite for that course

Expected Background

- Should have taken an introductory course on computer systems.
- All programming assignments will be in C++.
  - Programming assignment #1 will help get you caught up with C++.
  - If you have not encountered C++ before, need to put in extra effort.
  - Use a large language model like ChatGPT for assistance.
  - Relevant parts of C++ will be briefly covered in this course.

Course Logistics

- Course Website (link on Canvas)
- Discussion Tool: Piazza (link on Canvas)
- Grading Tool: Gradescope (link on Canvas)
- In-Class Quiz Tool: Point Solutions (link on Canvas)

Course Rubric

- Exams (50%)
- Programming Assignments (20%)
- Exercise Sheets (15%)
- In-Class Quizzes (15%)

Course Policies

- Programming assignments & exercise sheets must be own work.
  - Not group assignments.
  - You may not copy source code from other people or the web.
  - Plagiarism will not be tolerated.
  - We will follow the late submission policy listed on Canvas.
- Academic Honesty
  - Refer to Georgia Tech Academic Honor Code.
  - If you are not sure, ask me.

Textbooks for Reference

- Silberschatz, Korth, & Sudarshan:
  - Database System Concepts. McGraw Hill, 2020.
- Hector Garcia-Molina, Jeff Ullman, and Jennifer Widom:
  - Database Systems: The Complete Book. Prentice-Hall, 2008.

Intro Sheet

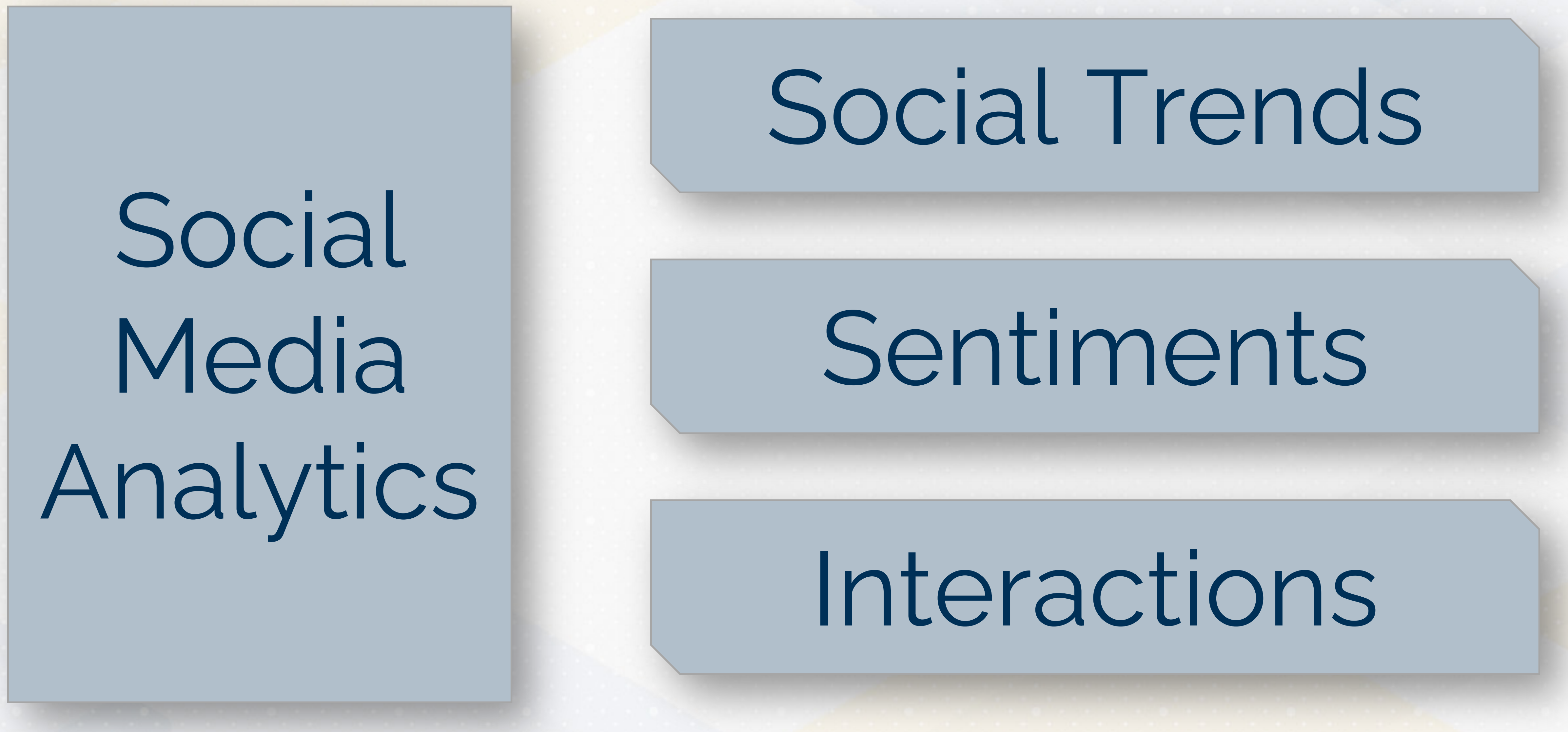
- Upload a one-page PDF with your details on Gradescope.
  - Picture (ideally 2x2 inches of face).
  - Name, interests, and other details mentioned on Gradescope.
- Purpose of this sheet
  - Help me know more about your background for tailoring the course.
  - Recognize you in class.

In-Person Office Hours

- Sign up for a ten-minute slot in the sign-up sheet (link on Canvas)
- Teaching assistants will guide you with assignments & sheets.

Motivating Application

Social Media Analytics Application



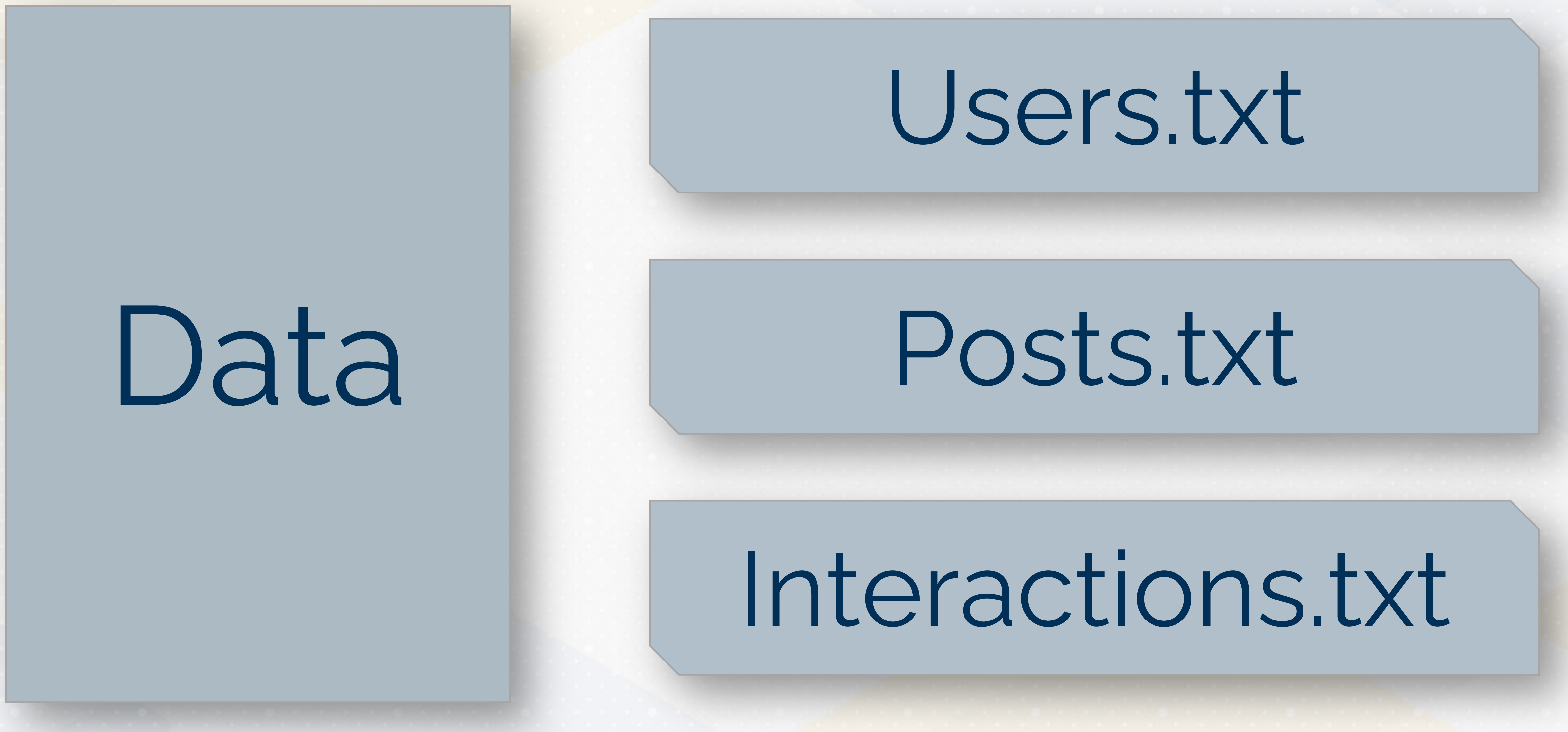
Social Media Analytics

Social Trends

Sentiments

Interactions

Flat-File Database System



Data

Users.txt

Posts.txt

Interactions.txt

Users Text File

Users.txt

UserName,	Location
Timothée Chalamet,	Paris
Lana Condor,	Los Angeles
Liu Yifei,	Beijing
Burna Boy,	Lagos
Kriti Sanon,	Mumbai

Posts Text File

Posts.txt

PostID,	UserName,	Location
1001,	Timothée Chalamet,	Excited to start filming my new movie!
1002,	Lana Condor,	Had a great time at the beach today! 🌊☀️
1003,	Liu Yifei,	Enjoying the scenery in Beijing! 🏞️
1004,	Burna Boy,	Live performance tonight in Lagos! 🎤🎶
1005,	Kriti Sanon,	Loving the vibrant energy of Mumbai! 🌆

Interactions Text File

Interactions.txt

PostID,	UserName,	Reaction Type,	Comment
1001,	Lana Condor,	Comment,	Love it!
1002,	Liu Yifei,	Like,	-
1003,	Burna Boy	Like,	-
1004,	Kriti Sanon	Comment,	Wish I could be there!

Limitations of Flat-File Database

Limitation #1: Data Redundancy

PostID,	UserName,	Location
1001,	Timothée Chalamet,	Excited to start filming my new movie!
1006,	Timothée Chalamet,	Exploring the streets of Paris! 🇫🇷
1007,	Timothée Lamet,	Just wrapped up a day of filming 🎬
1008,	Timothée Chalamet,	Any book recommendations?

Limitation #2: Slow Operations

UserName	Location
Timothée Chalamet,	Paris
~~Lana Condor,~~	~~Los Angeles~~
Liu Yifei,	Beijing
Burna Boy,	Lagos
Kriti Sanon,	Mumbai

Limitation #3: Slow Queries

UserName	Location
Timothée Chalamet,	Paris
Lana Condor,	Los Angeles
Liu Yifei,	Beijing
Burna Boy,	Lagos
Kriti Sanon,	Mumbai

Limitation #4: Concurrent Updates

USER 1

Xavier Laurent,
Paris

USER 2

Xavier Laurent,
New York

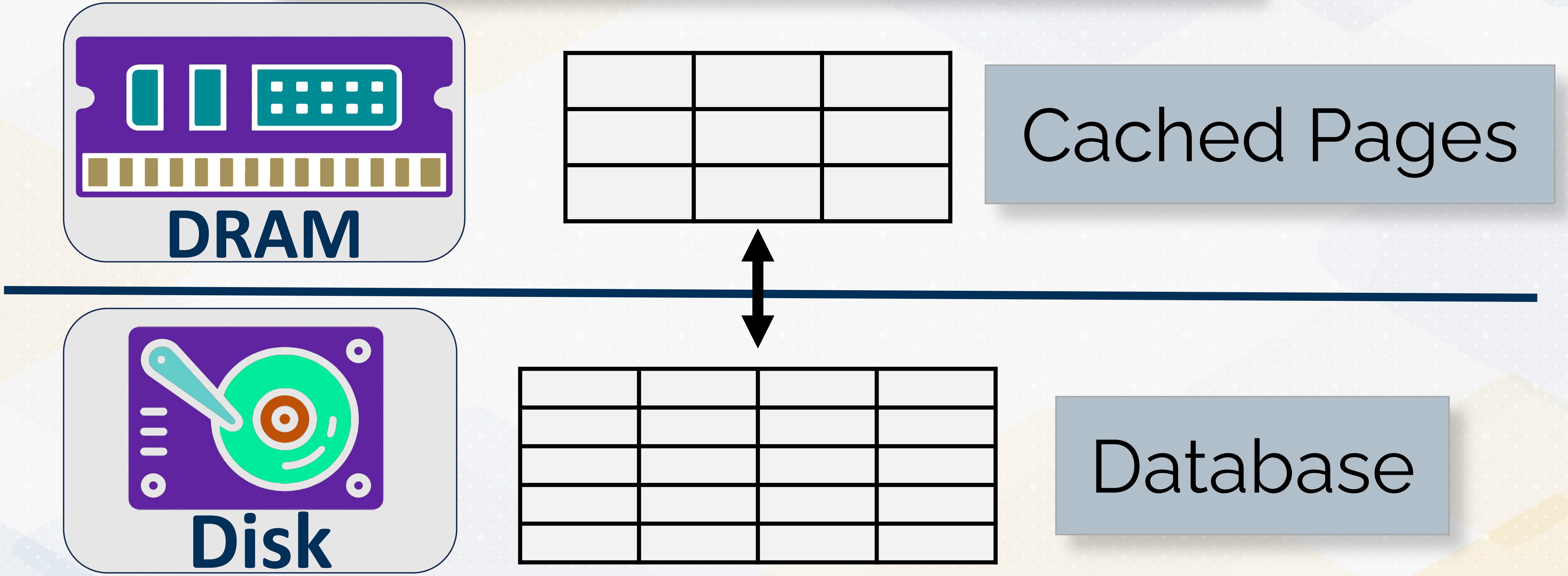
Limitation #5: Handling Disk Failure

Users.txt

UserName,	Location,	Country
Timothée Chalamet,	Paris,	France
Lana Condor,	Los Angeles,	USA
Liu Yifei,	Beijing,	China
Burna Boy,	Lagos	
Kriti Sanon,	Mumbai	

Limitation #6: Memory Management

Faster access - not durable



DRAM

Cached Pages

Disk

Database

Slower access - but durable

Limitation #7: Usability

Custom Code

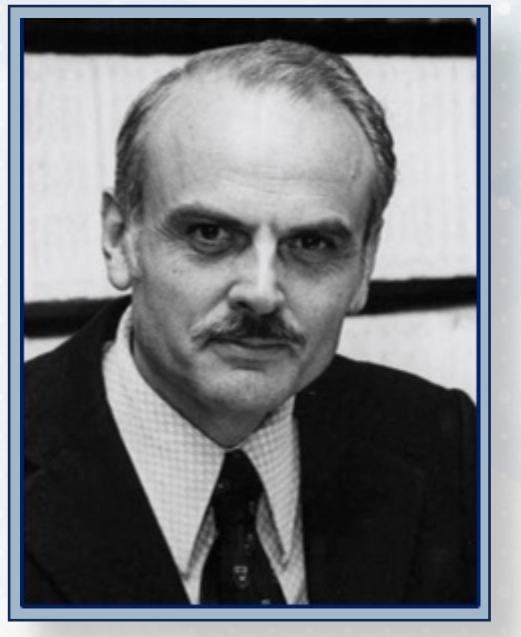
Comments Query Code

```
def get_comments_by_user(file_path, user_name):
    comments = []
    with open(file_path, 'r') as file:
        for line in file:
            post_id, user, reaction_type, comment_text =
line.strip().split(', ')
            if user == user_name and reaction_type == "Comment":
                comments.append((post_id, comment_text))
    return comments
```

Relational Database

Relational Database

Ted Codd (1970)



- Scientist at IBM
- Simplify Data Management
- Organize Data as Tables

Relational Database

Column / Attribute

Row/ Tuple

UserName,	Location
Timothée Chalamet,	Paris
Liu Yifei,	Beijing
Burna Boy,	Lagos
Kriti Sanon,	Mumbai

Relational Database

UserName,	Location
Louis Garrel,	Paris
Lana Condor,	Los Angeles
Liu Yifei,	Beijing
Burna Boy,	Lagos
Kriti Sanon,	Mumbai

PostID,	UserName,	Location
1001,	Louis Garrel,	Excited to start filming my new movie!
1002,	Lana Condor,	Had a great time at the beach today! 🌊☀
1003,	Liu Yifei,	Enjoying the scenery in Beijing! 🏞
1004,	Burna Boy,	Live performance tonight in Lagos! 🎤🎶
1005,	Kriti Sanon,	Loving the vibrant energy of Mumbai! 🌃

PostID	UserName	Reaction Type	Comment
1001,	Lana Condor,	Comment,	Love it!
1002,	Liu Yifei,	Like,	-
1003,	Burna Boy	Like,	-
1004,	Kriti Sanon	Comment,	Wish I could be there!

Relational Database

Mathematic Set Theory

Data Sets Relationship

Efficient Data Set Links

Students	Grades
Alice	B
Bob	A
Charlie	C

$R=\{(Alice,B),(Bob,A),(Charlie,C)\}$

Relational Database

Physical

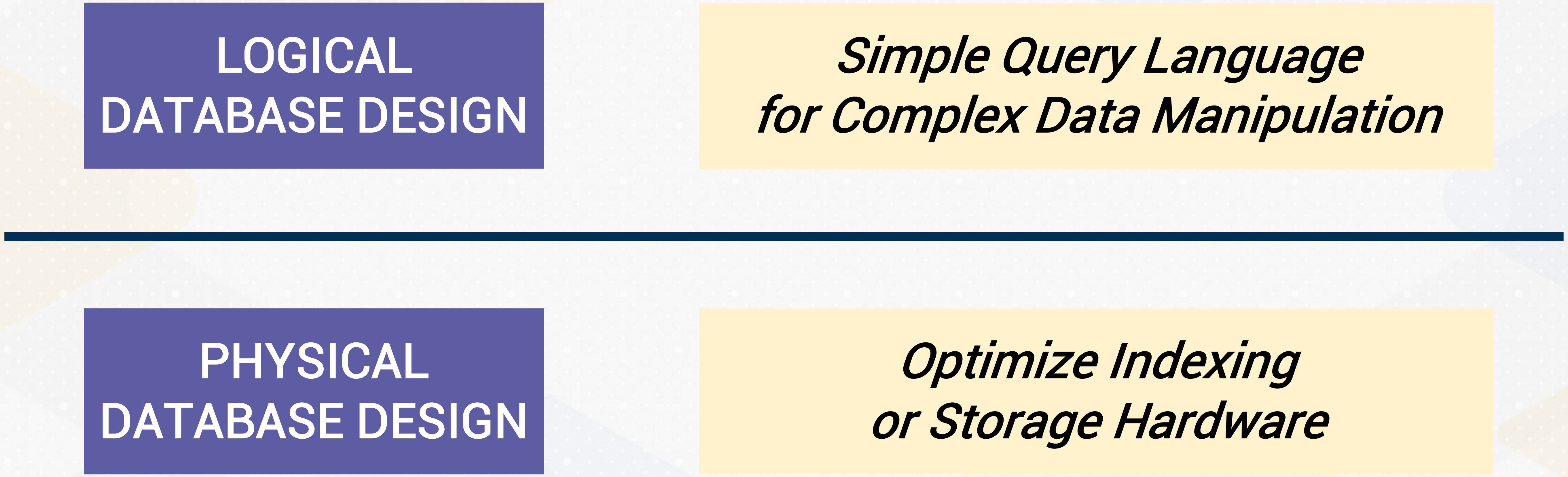
Storage Formats

Indexing Data Structures

List of Tables

Logical

Relational Database



LOGICAL DATABASE DESIGN

Simple Query Language for Complex Data Manipulation

PHYSICAL DATABASE DESIGN

Optimize Indexing or Storage Hardware

Logical Database Design: Primary Key

PostID,	UserName,	Location
1001,	Louis Garrel,	Excited to start filming my new movie!
1002,	Lana Condor,	Had a great time at the beach today! 🌊☀️
1003,	Liu Yifei,	Enjoying the scenery in Beijing! 🏞️
1004,	Burna Boy,	Live performance tonight in Lagos! 🎤🎶
1005,	Kriti Sanon,	Loving the vibrant energy of Mumbai! 🌃

Logical Database Design: Foreign Key

PostID,	UserName,	Location
1005,	Kriti Sanon,	Loving the vibrant energy of Mumbai! 🌃

UserName,	Location
Louis Garrel,	Paris
Lana Condor,	Los Angeles
Liu Yifei,	Beijing
Burna Boy,	Lagos
Kriti Sanon,	Mumbai

Referential Data Integrity

Benefits of Relational Database

#1: No Data Redundancy

UserName,	Location
Louis Garrel,	Paris

PostID	UserID	Post
1001	1	Excited to start filming my new movie!
1006	1	Exploring the streets of Paris! FR
1007	1	Just wrapped up a day of filming 🎬
1008	1	Any book recommendations?

#2: Fast Operations

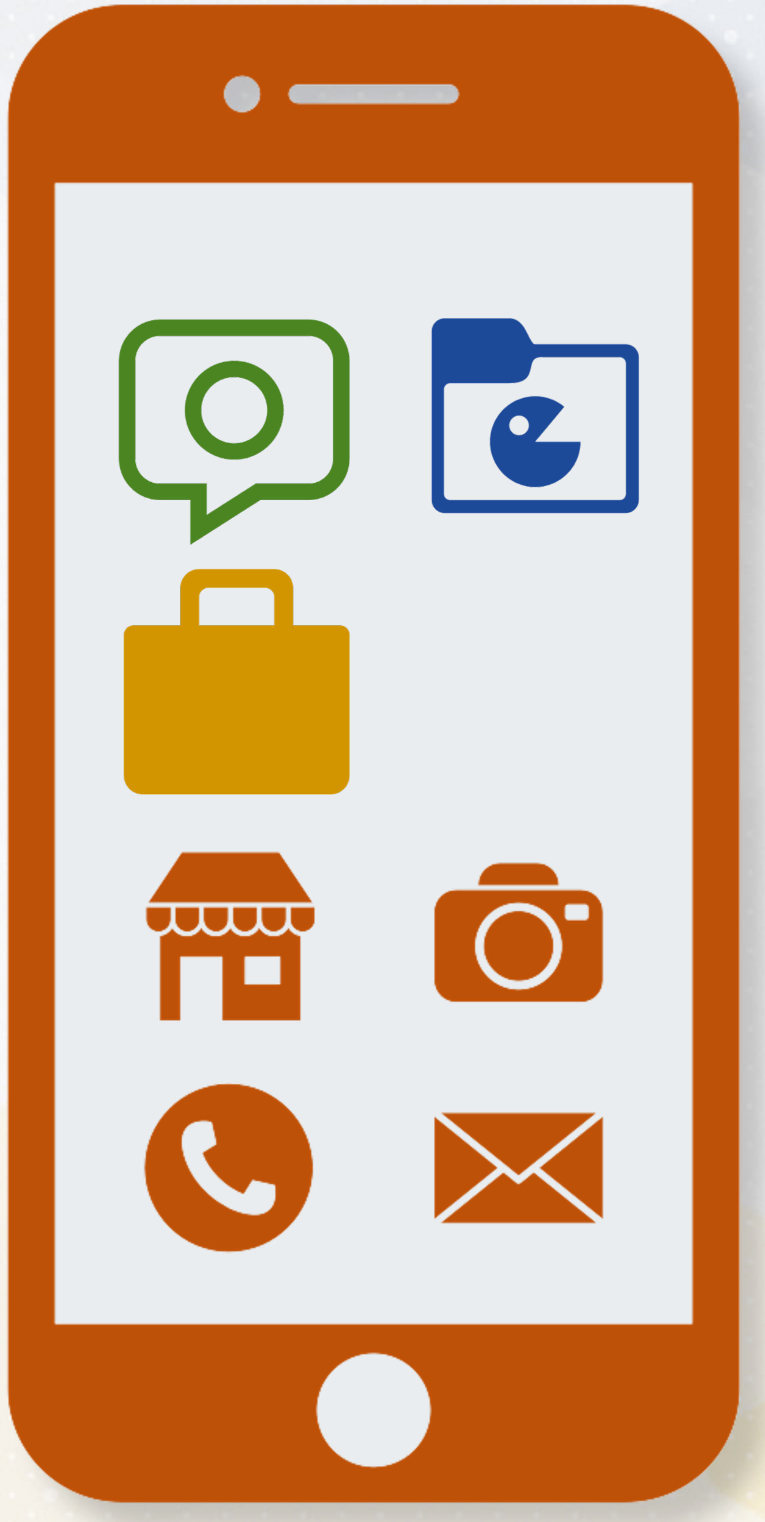
- Efficient Data Deletion
- User (Tuple) Removal
- Fast Deletion

#3: Fast Queries

Index Database

Apps in labeled folders

Location-based index

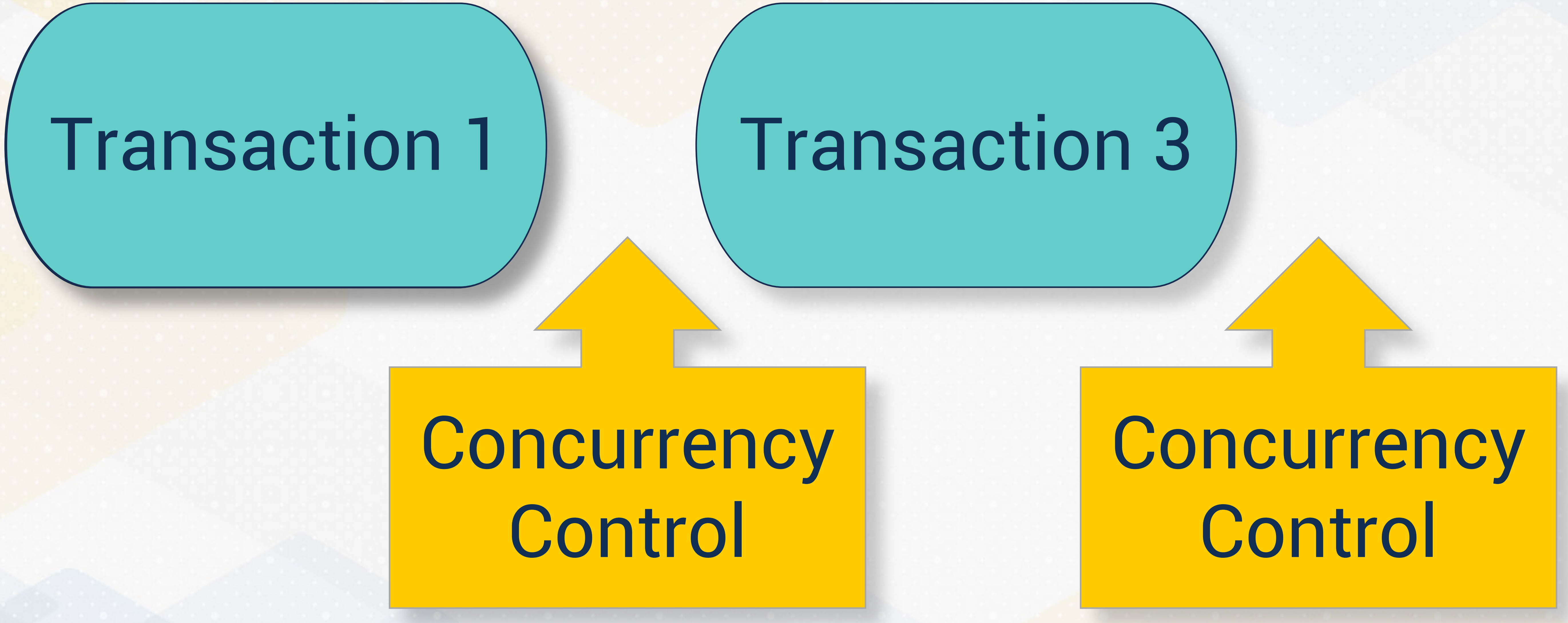


#3: Fast Queries

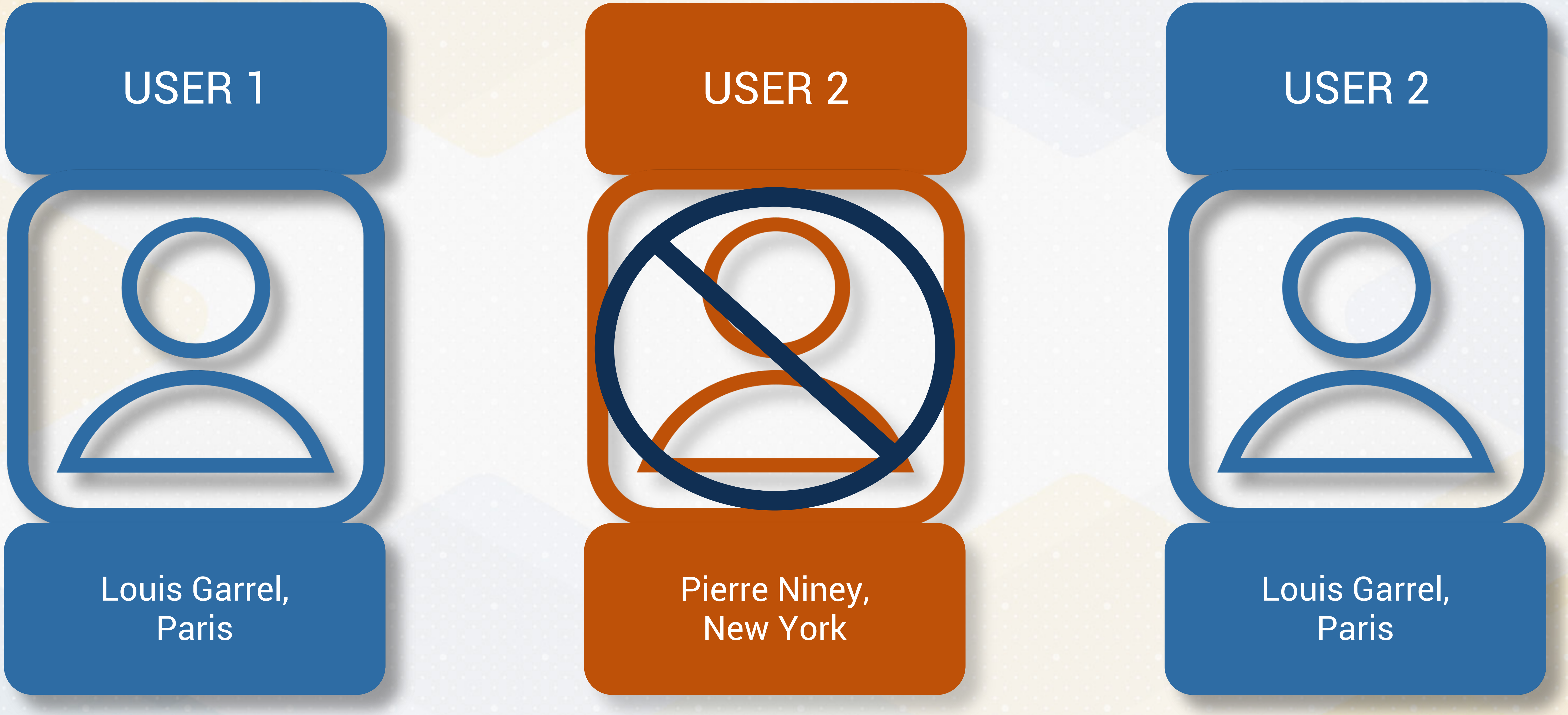
```sql
SELECT *
FROM Users
WHERE LOCATION = 'Mumbai';
```

UserName,	Location
Timothée Chalamet,	Paris
Lana Condor,	Los Angeles
Liu Yifei,	Beijing
Burna Boy,	Lagos
Kriti Sanon,	Mumbai

#4: Concurrent Updates



#4: Concurrent Updates



USER 1

Louis Garrel,
Paris

USER 2

Pierre Niney,
New York

USER 2

Louis Garrel,
Paris

#5: Handling Disk Failure

Atomic Transactions → "All or Nothing"

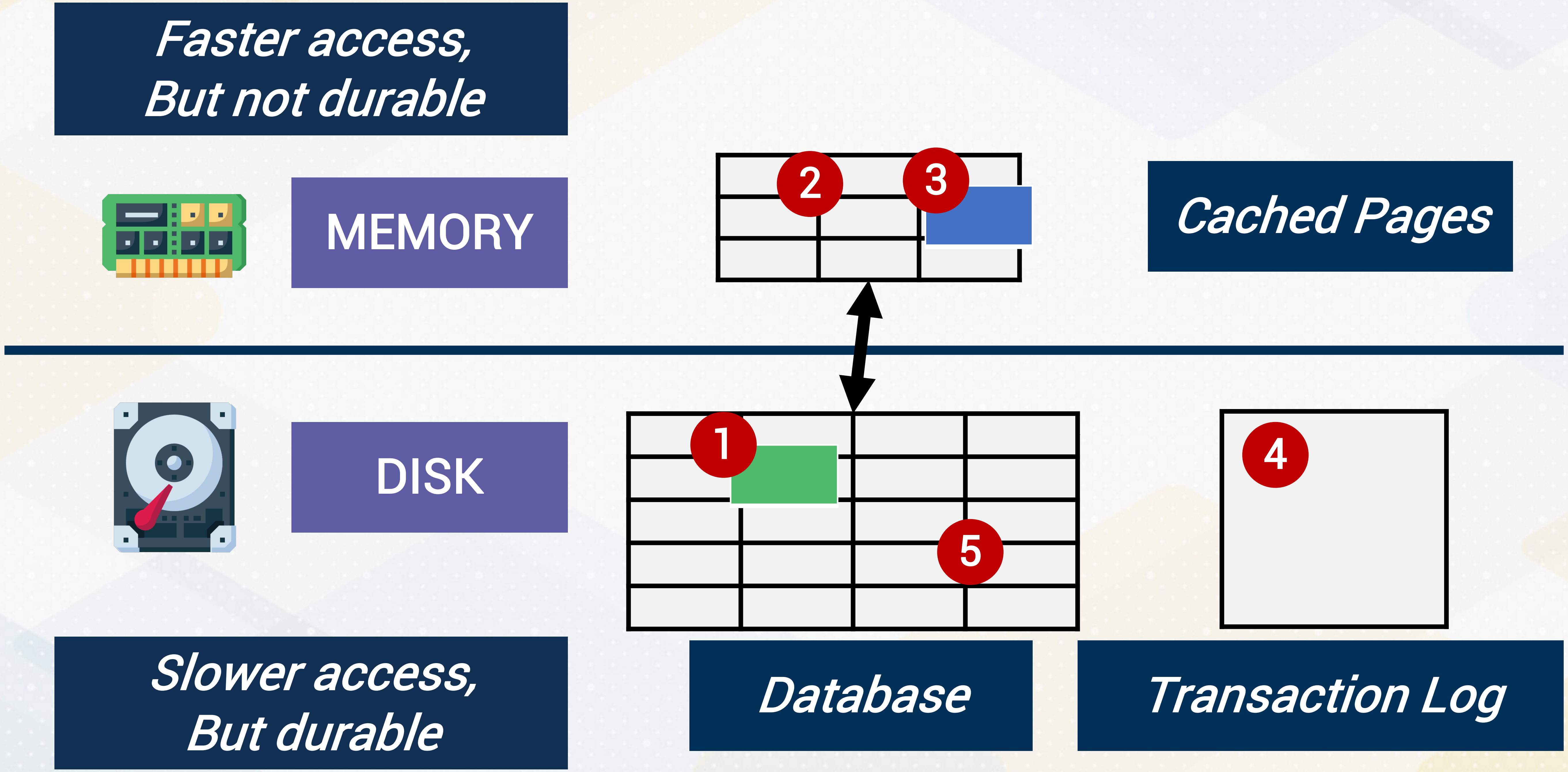
UserName	Location	Country
Louis Garrel	Paris	France
Lana Condor	Los Angeles	USA
Liu Lifei	Beijing	China
Burna Boy	Lagos	
Kriti Sanon	Mumbai	

Reversion →

#5: Handling Disk Failure

UserName	Location	Country
Timothée Chalamet	Paris	France
Lana Condor	Los Angeles	USA
Liu Lifei	Beijing	China
Burna Boy	Lagos	
Kriti Sanon	Mumbai	

#6: Memory Management



Faster access, But not durable

MEMORY

Cached Pages

DISK

Slower access, But durable

Database

Transaction Log

#7: Usability

UserName	Location
Timothée Chalamet	Paris
Lana Condor	Los Angeles
Liu Yifei	Beijing
Burna Boy	Lagos
Kriti Sanon	Mumbai

SQL = Declarative

Python, C++ = Imperative

Conclusion

- Illustrative Social Media Analytics
- Limitations of a Flat-file Database System
- Benefits of a Relational Database System