Project Overview: The Incubator

CS411 Database Systems

Kevin C. Chang

Project Options

- **Project Option 1 (Application Oriented):**

  Building a Database-Driven Web/Mobile

- **Project Option 2 (System Oriented):**

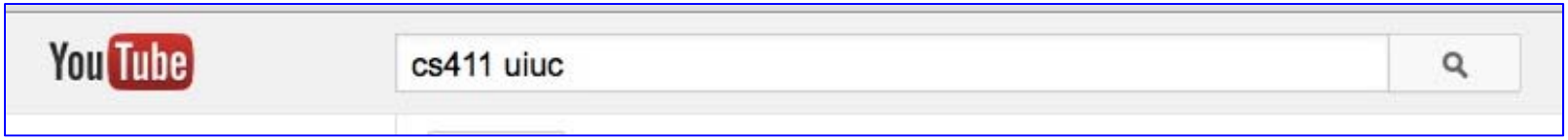
  Enhancing an Open Source DBMS

- More details, visit CS411 Web

Project Gallery

- CS411 Web, Project Page
  - https://wiki.engr.illinois.edu/display/cs411fa13/Projects--+The+411+Incubator

- Or:



Project Gallery

- What are some cool projects in the past?
- Illini Crime
  - http://www.illinicrime.com/
- Pokedex 2.0
  - http://www.youtube.com/watch?v=YTYQuESBvbU
- Ultimate Dining
  - http://www.youtube.com/watch?v=L_Wp_Y5yKoM

Track 1: Building a Database-driven Web/Mobile App

Overview

- Project expectations
- Example system:
  - Apartment Search

Project

- Database-Driven Web Information System
  - Web-based application that needs a database
  - Must be useful and realistic

How to Choose a Good Project Topic?

Two Essential Criteria:

- Useful
  - Is your system useful? Why should people use your system?
- Innovative
  - Is this anything new? A new concept? A new way of doing some thing?

Two Plus Criteria:

- Realness
  - Real data: Does your system have enough data to make it practical/interesting?
- Fun
  - Are you having fun?

Functionality Criteria

- Basic Functions (must meet all)
  - Search database and list records
  - Insert, update, delete records into the database
- Advanced Functions (must have at least 2)
  - Examples:
    - “Cool” user feature that goes beyond the basic functionalities
    - Support multi-users at a time
    - Batch loads from other datasets
    - Additional administrative interface or role-based interface

Basic Functions

- Search
- Insert
- Update
- Delete

Some Advanced Functions

Criteria

- Data-centric
- Relevant, useful to your application
- Specific to your application
- Technically challenging
- Significant work to implement

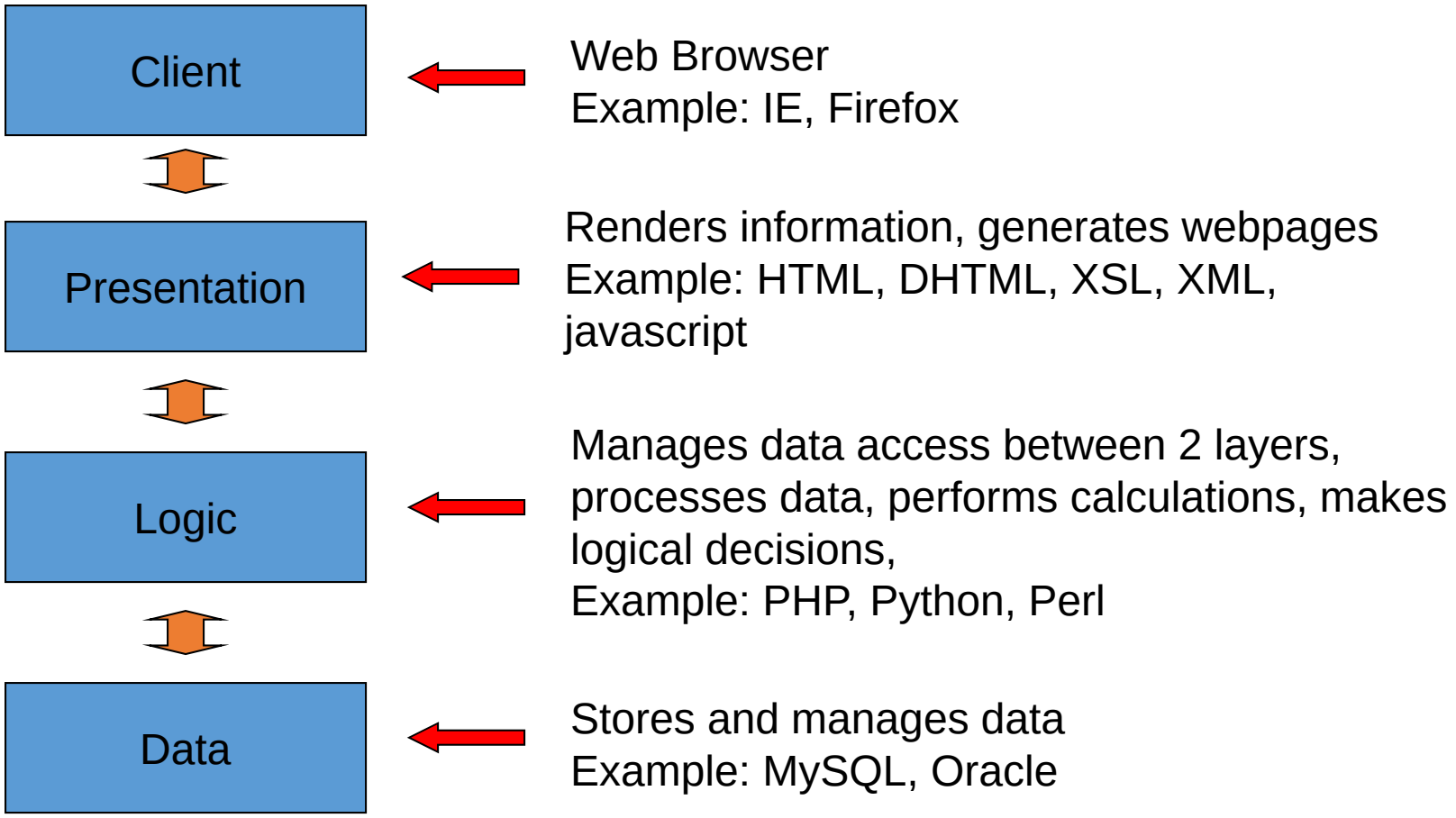
Examples:

- Creative search of data
  - Match movies by user profile?
- Creative analysis of data
  - Trend/patterns of tweets?
- Creative visualization/presentation of data
  - Overlay Pokemon world onto google map

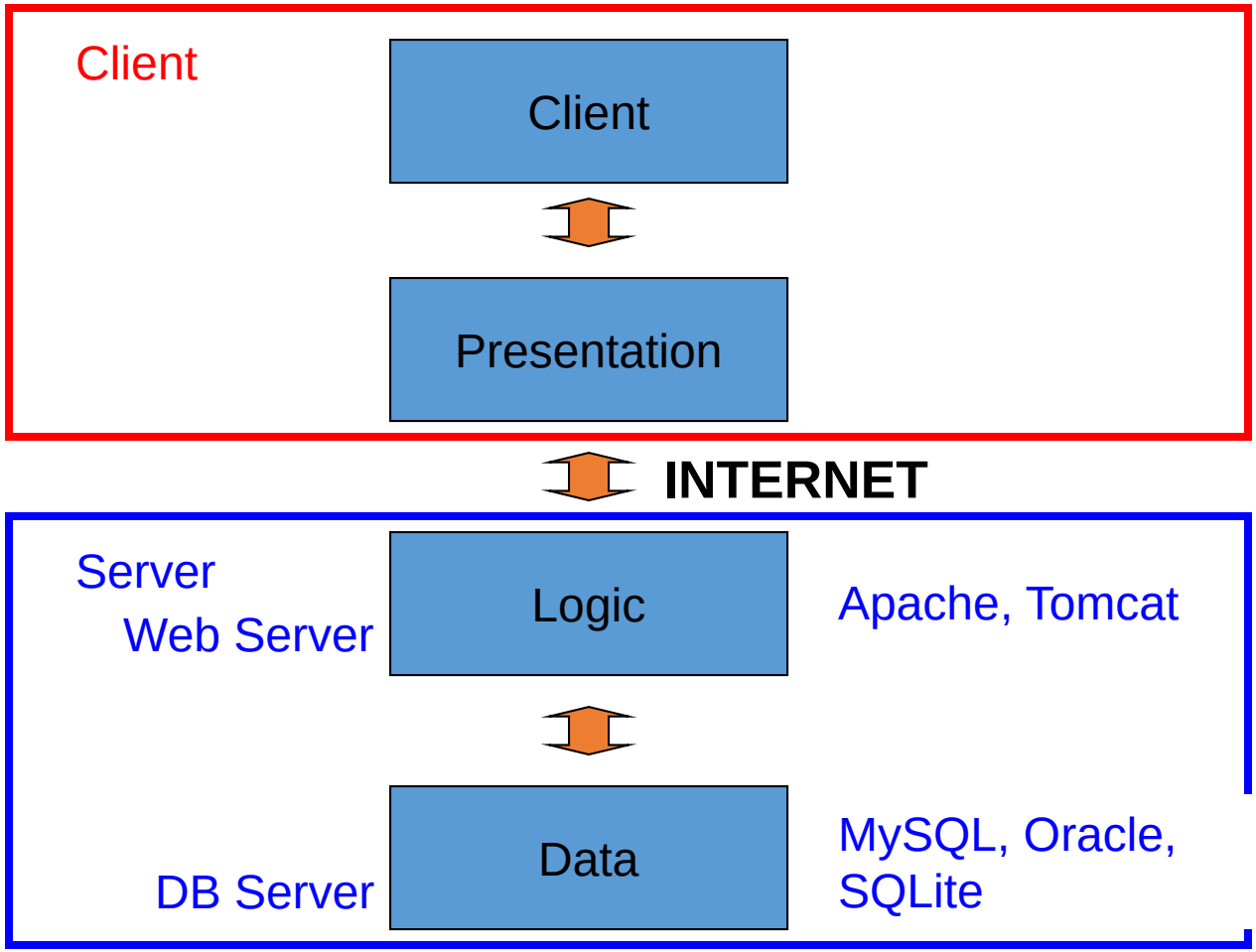
Common Tools

- Front-end: Browser rendering languages
  - HTML, JavaScript, CSS, DHTML
- Backend: LAMP stuff
  - Operating system
    - Linux
  - Web server
    - Apache HTTP Server, Apache Tomcat
  - Database
    - MySQL, Oracle, SQLite
  - Scripting/programming language
    - PHP, Python, Perl, Java, Ruby, C++

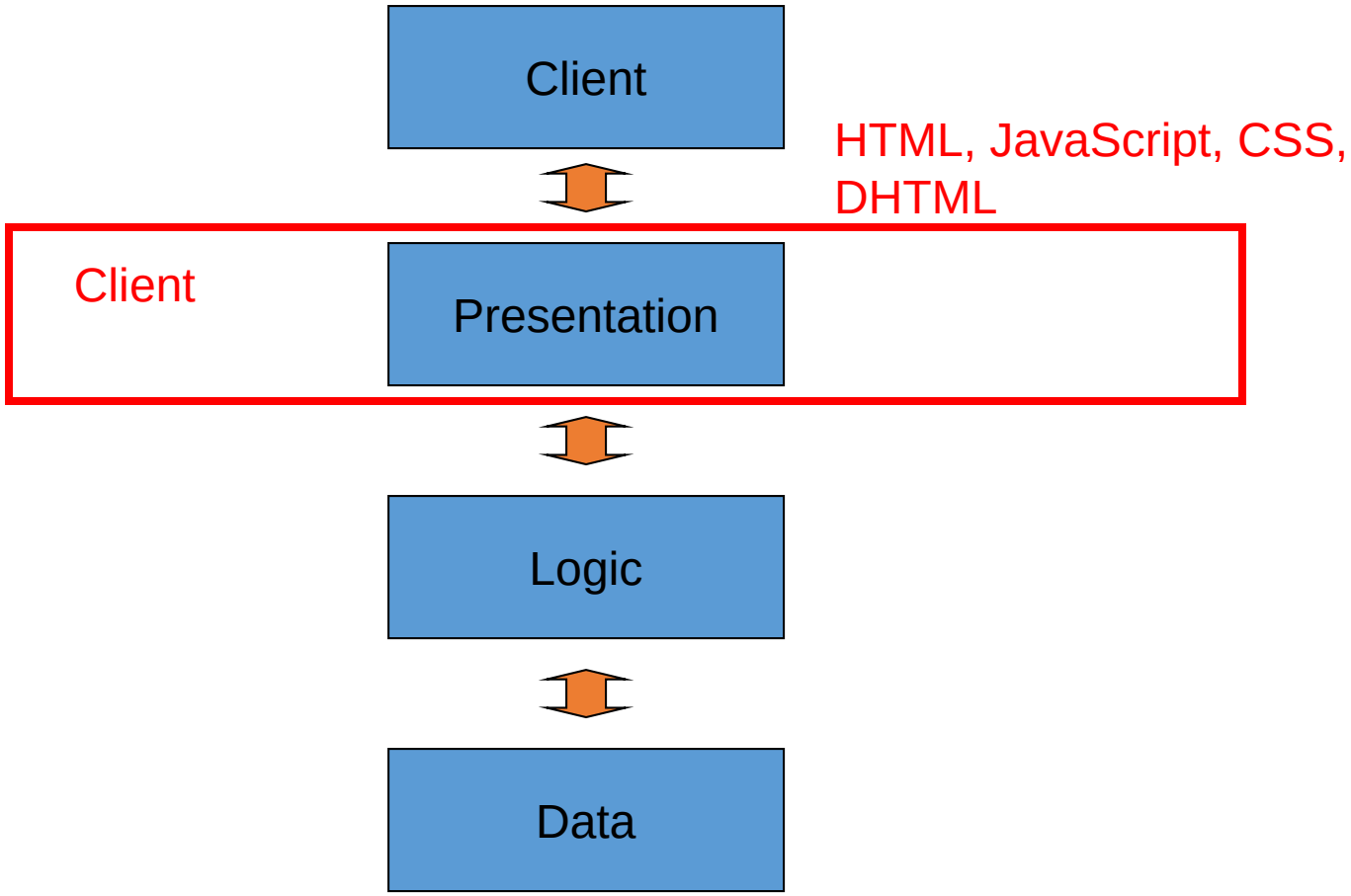
Web Apps
Share One Common Architecture



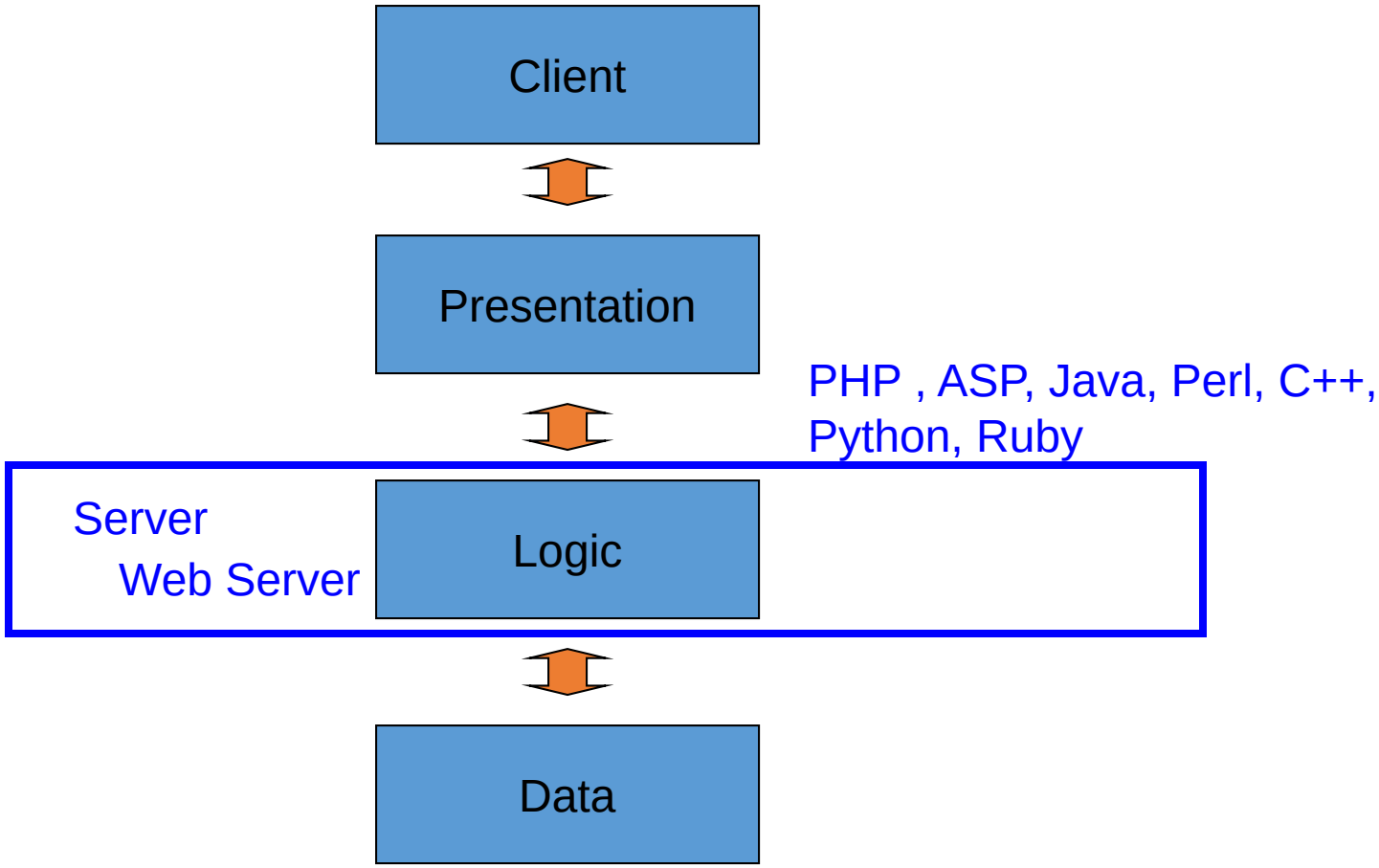
Client vs Server



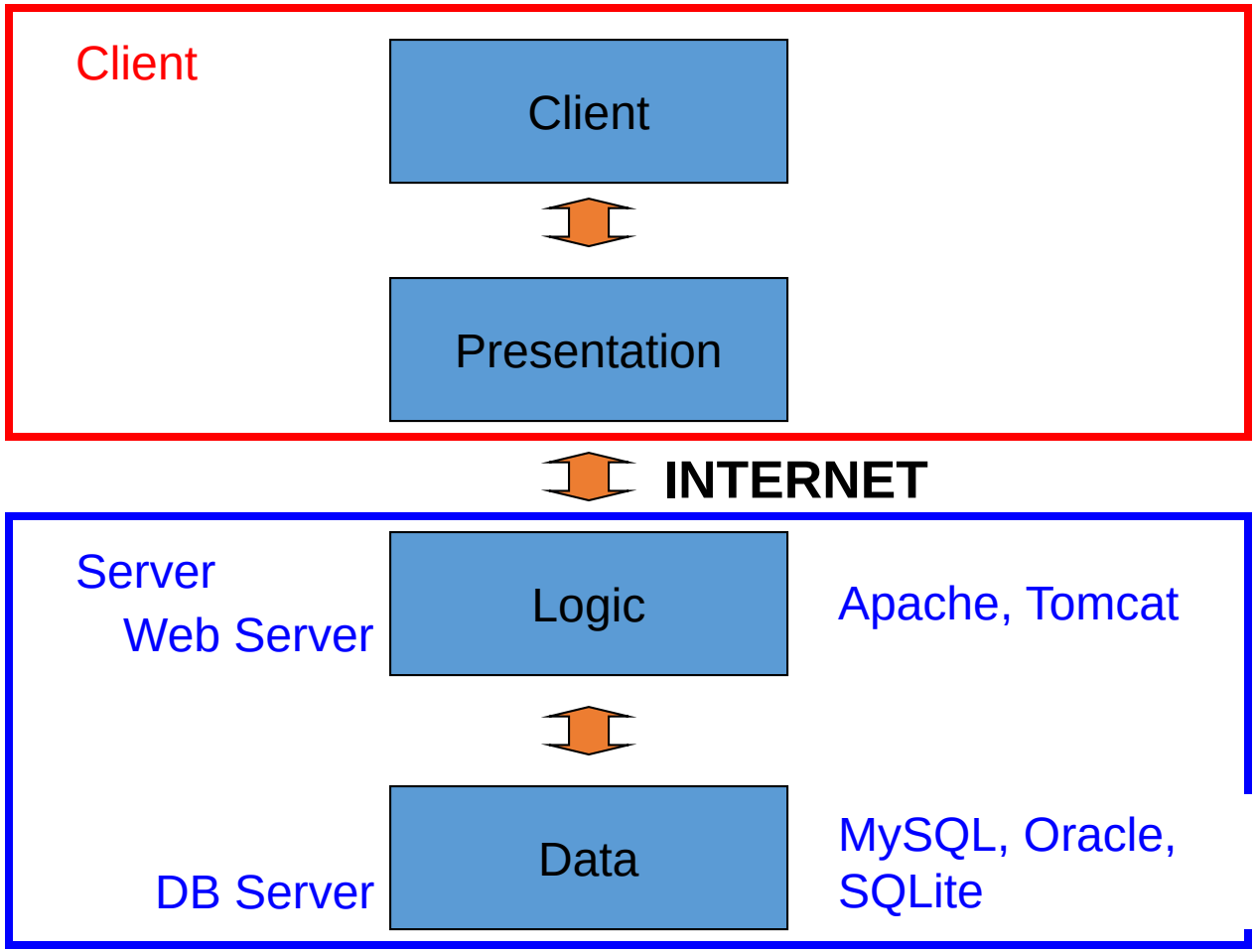
Client Side Scripting



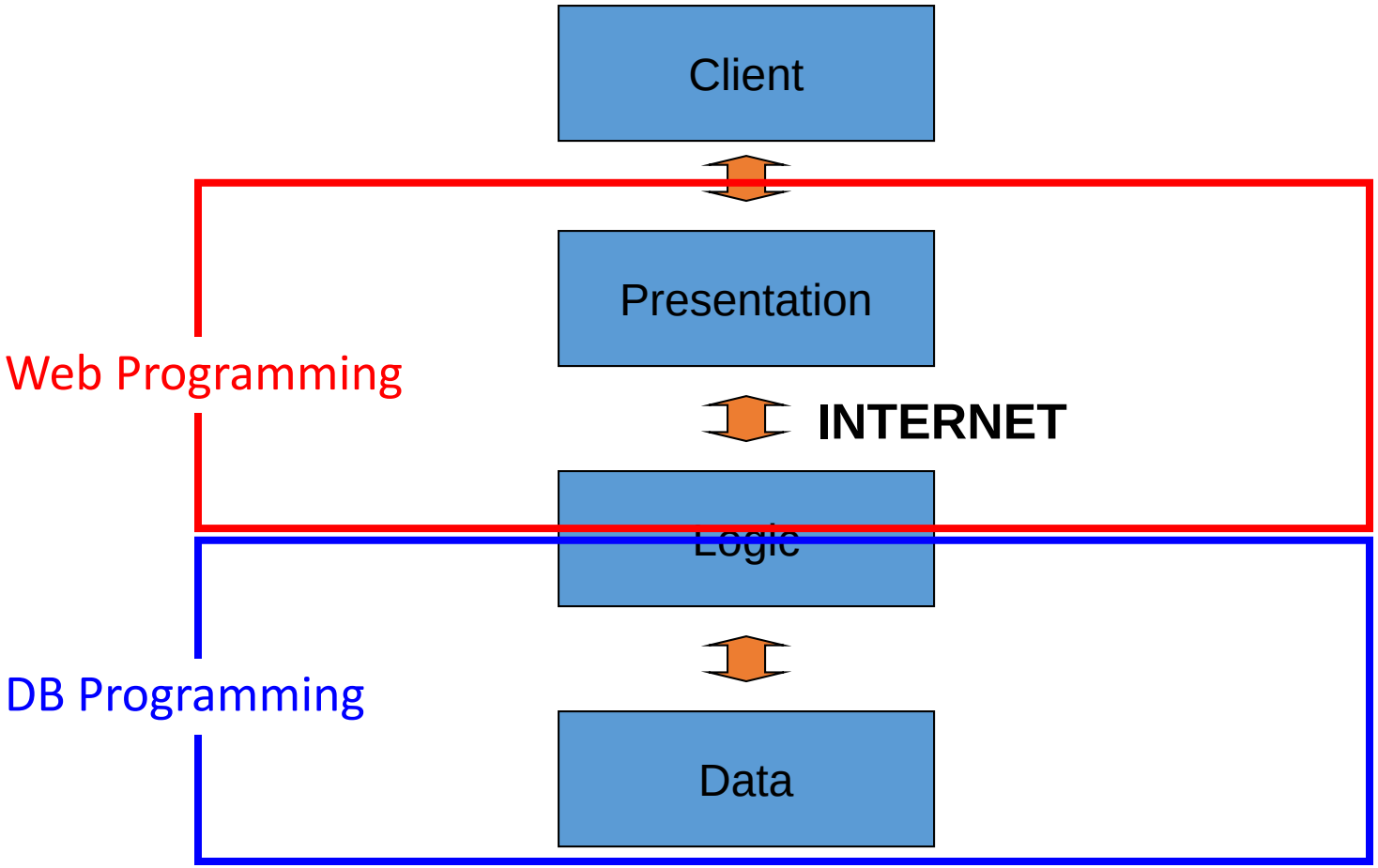
Server Side Scripting



Client vs Server



Web vs. DB


Client
Presentation
Web Programming
INTERNET
Logic
DB Programming
Data

Project Stages

- Stage 0: Group formation
- Stage 1: Functional description & ER design
- Stage 2: Development plan
- Stage 3: Setup development environment
- Stage 4: Mid-term checkpoint
- Stage 5: Final demo

Track 2: Enhancing an Open Source DBMS

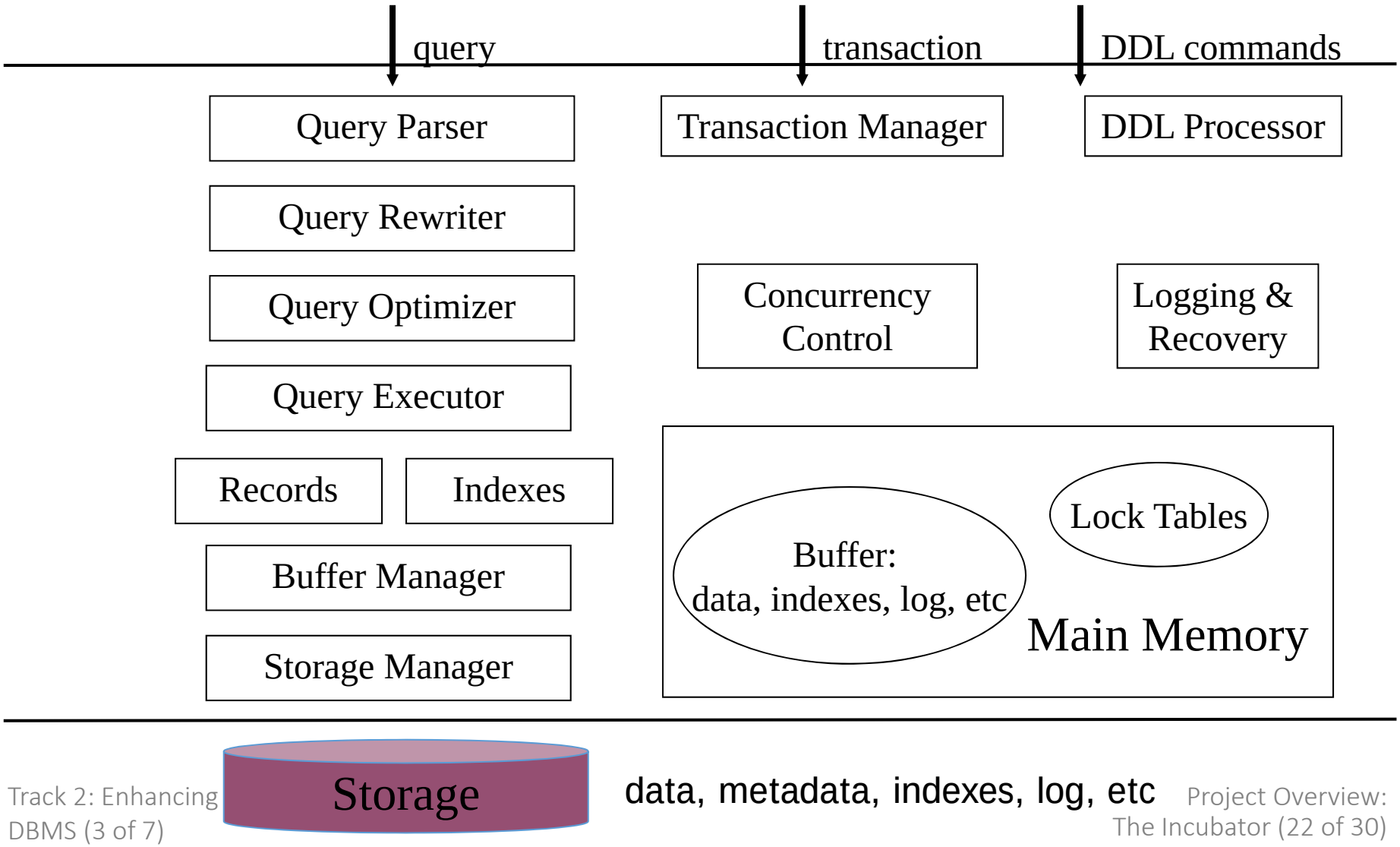
Overview

- Project expectations
- Example systems:
  - Support text relevance natively within a DBMS
  - Support “Hibernation” in a DBMS

Project

- Tweaking an Open Source DBMS System
  - Implement/Improve a system component of an open source DBMS system
  - Must be useful and novel
  - Open ended; you can propose your own ideas and discuss with the class staff

DBMS Architecture



Topic Example: DBMS with text relevance

- How to make DBMS aware of texts, and support query relevance over them
  - Why is it useful?
    - Nowadays, many data comes with attributes, as well as a text description
      - E.g., product specs with a detailed review
  - Why is it novel?
    - Many DBMS nowadays only support simple text matching
      - E.g., description contains "good quality"

Topic Example: DBMS Hibernation

- How to make DBMS Hibernate/Resume like an OS does?
  - Why is it useful?
    - Nowadays, many data processing takes a long time, and you may need to halt the processing
  - Why is it novel?
    - Most DBMS nowadays do not support such functionality. There is some recent research, though.

Open Source Systems

- DBMS Systems:
  - MySQL
  - PostgreSQL
  - SQLite
- Text Retrieval Systems:
  - Lucene

Project Stages

- Stage 0: Group formation
- Stage 1: Development plan
- Stage 2: Mid-term checkpoint
- Stage 3: Final demo & report

Conclusion

Project Tips

- Start early
  - “Unexpected issues”
  - Data collection can take some time
  - Schema may change
  - Web space, database access will take some time to get familiar with
  - Get the basics out of the way. Then focus on the challenging parts.

How to find partners for project

- Group size 3-4 is recommended
- Talk to your fellow classmates in class
- Newsgroup
  - Give a short intro about your background, and your interest on project

- Come give a pitch up here
- Put a flyer at Illini Union
- Hold a parade at Illinois Homecoming (a little too late)

- Or, talk to staff

And, the highlights

- Midterm pitch in class
  - Ideas, mockups, questions

- Final demos open to the public
  - VCs from the valley and the prairie will be in disguise