3-Tier Architecture

Subject: Web engineering

University of Education Okara Campus

Inam Ul Haq

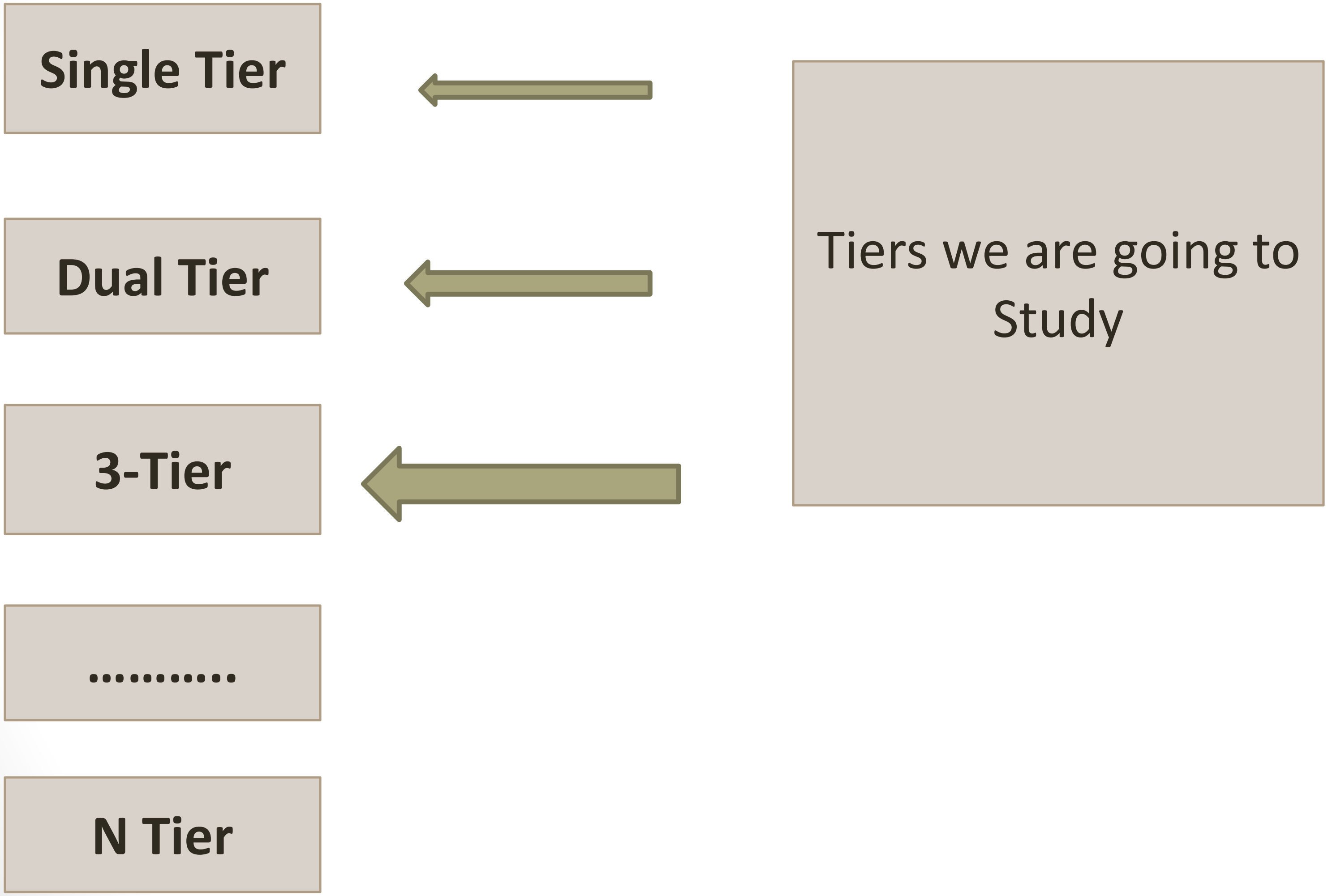
Inam.bth@gmail.com

3-Tier Architecture

3 tier architecture is a three way interaction in Client-Server environment

- User interface stored in **Client**
- **Business Application Logic** is used as one more server
- The Data is stored in **Database server**

Evolution of 3-Tier Architecture



Single Tier

Dual Tier

3-Tier

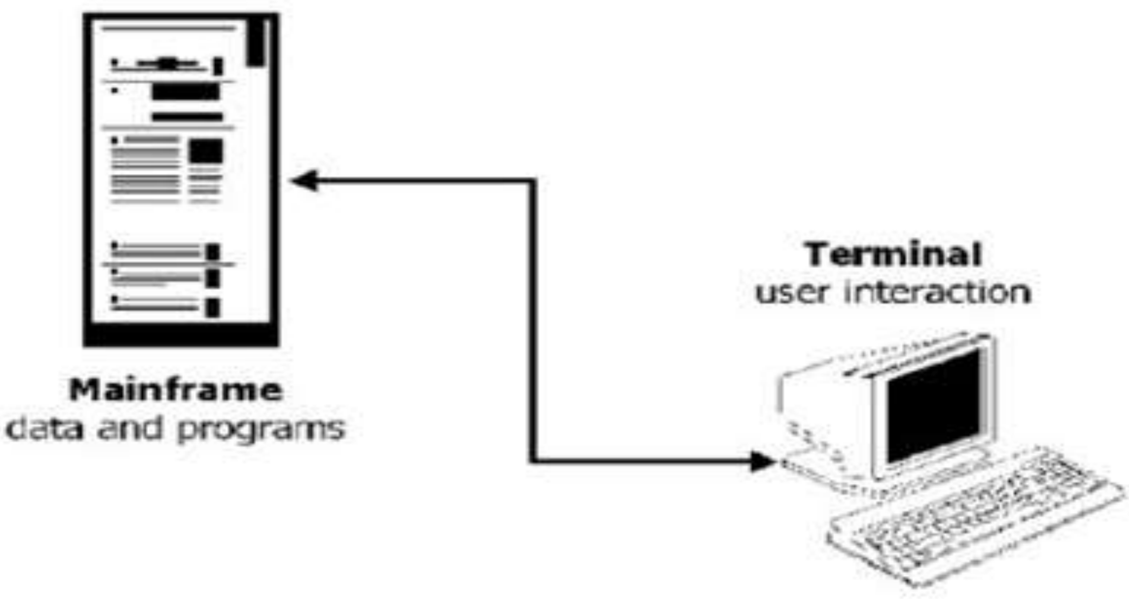
………..

N Tier

Tiers we are going to Study

Single Tier Architecture

- Time of Huge Main-Frame computers
- Single Computer manage all processes
- All Resources Attached to the same computer
- Access via dumb Terminals



Advantages and Disadvantages

Advantage

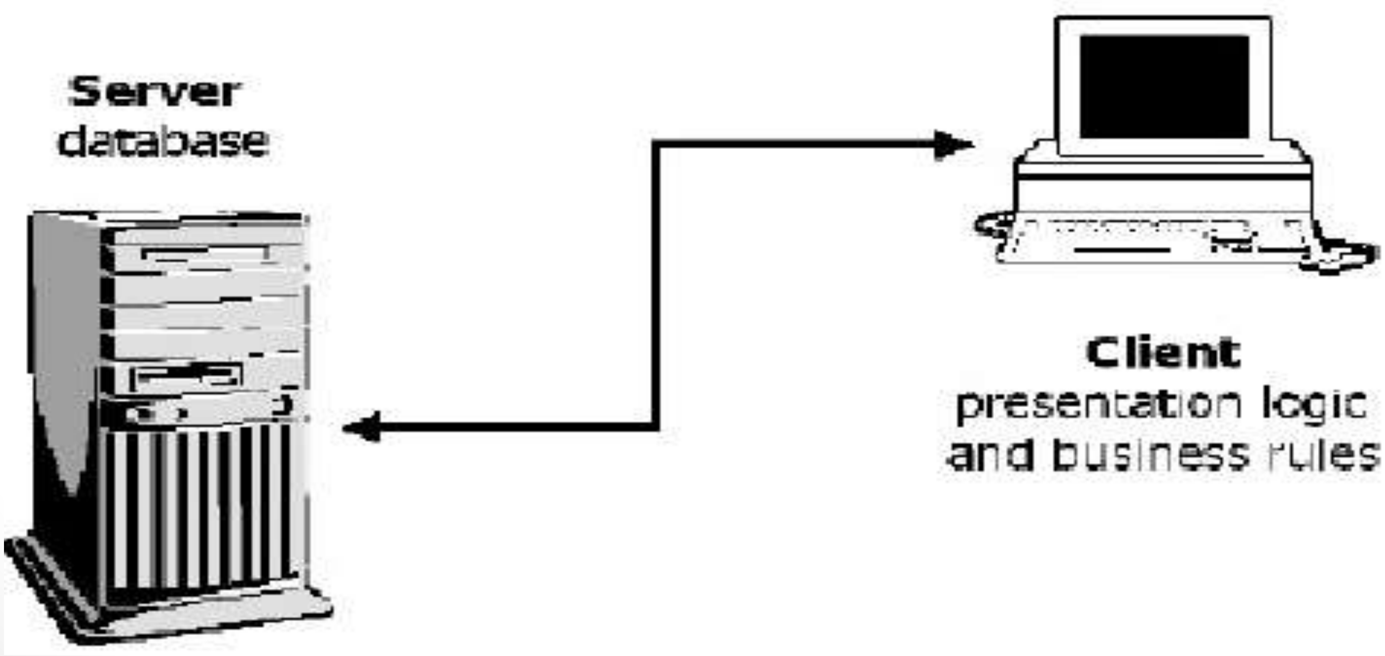
- Simple
- Efficient
- Uncomplicated

Disadvantages

- Very Expensive

Dual Tier Architecture

- Time of Personal computers
- The Client Server model was Born!
- Logical System Component most of which are on the client



Advantages and Disadvantages

Advantages

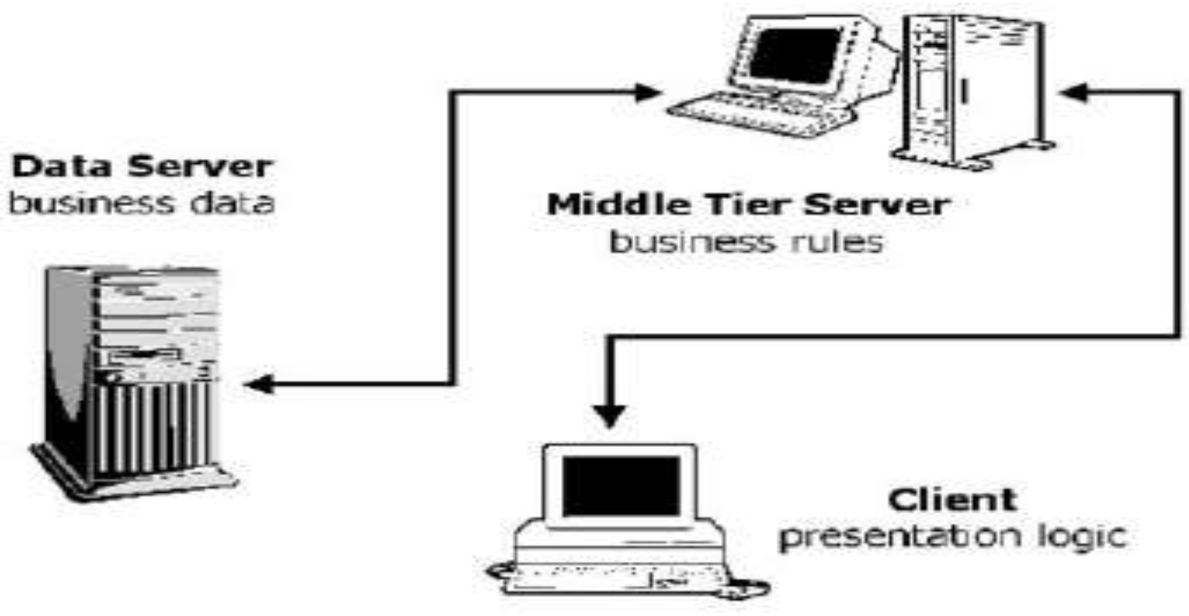
- Less expensive then Main-Frame

Disadvantages

- The connection to the database server are very expensive
- One user can connect to the user of server at a time
- Server need more time to respond existing connections
- Cost-ineffective: many users can use their connection 2-3% of time

3-Tier Architectures

A three-tier architecture is a **client-server architectur**e in which the functional process, data access, computer data storage and user interface are developed and maintained as independent modules on separate platforms



3-Tier Architectures(Cont…)

Presentation tier

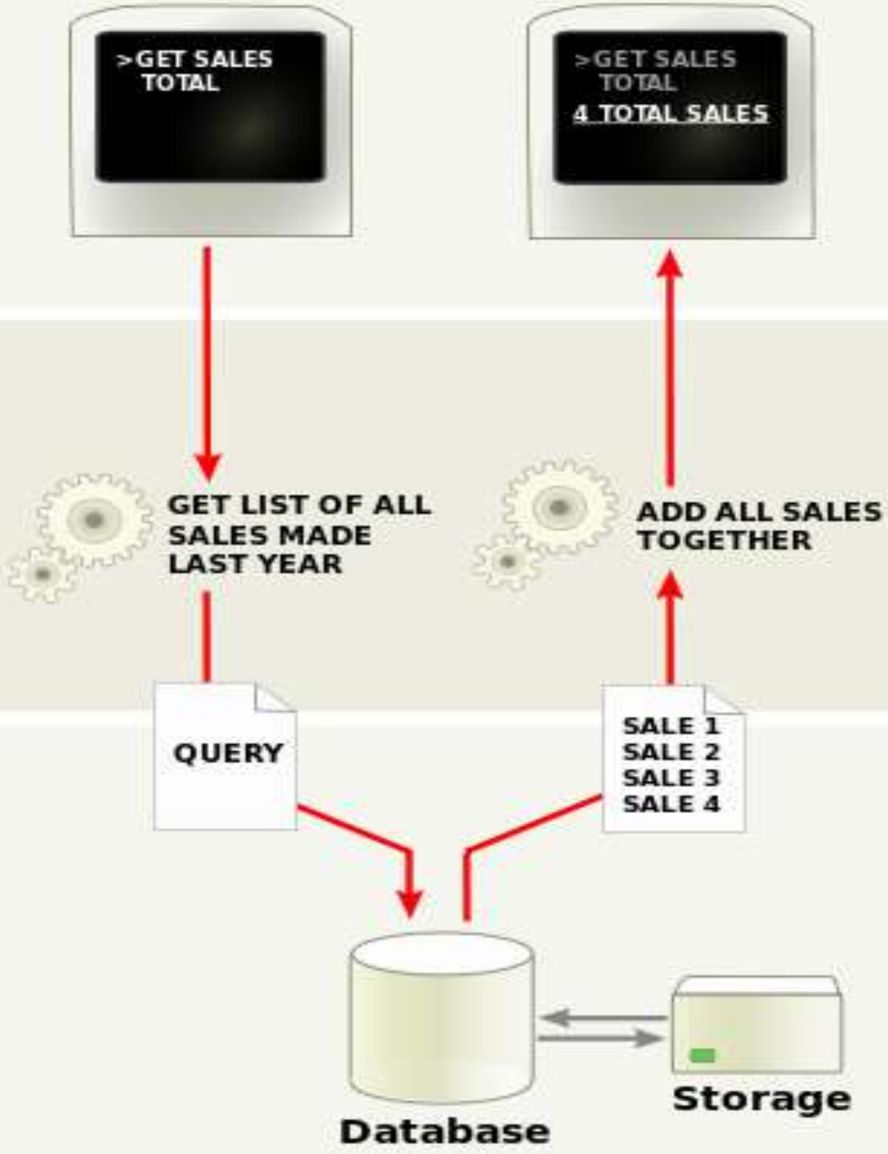
The top-most level of the application is the user interface. The main function of the interface is to translate tasks and results to something the user can understand.

Logic tier

This layer coordinates the application, processes commands, makes logical decisions and evaluations, and performs calculations. It also moves and processes data between the two surrounding layers.

Data tier

Here information is stored and retrieved from a database or file system. The information is then passed back to the logic tier for processing, and then eventually back to the user.



Advantages

❑ **Scalability**

- The application server can be deployed on many machines
- Database no longer require connection from every client

❑ **Data Integrity**

- Middle tier can ensure that only valid data is allowed to be updated in the database

❑ **Improved security**

- Because client does not have direct access to the database therefore data is more secure

Advantages (Cont...)

❑ **Reduced distribution**

- Updating is only need for application server and no need for distributed on client

❑ **Maintainability**

- Changes to the components in one layer have no effect on any others layers

❑ **Reliability**

- A 3-Tier architecture, if deployed on multiple servers, makes it easier to increase reliability of a system by implementing multiple levels of redundancy.

Disadvantages

❑ **Increased complexity / efforts**

- 3-tier architecture is more complex to build compared to 2-tier architecture

Why 3-Tier Architecture ?

Needs of new world’s applications(Web Apps)

- Business will increasingly compete being the first to market with new electronic goods and **services**
- Companies will **create *virtual corporations*** through alliances with a shifting set of partners
- **Roles and relationships btw. enterprises** will shift frequently

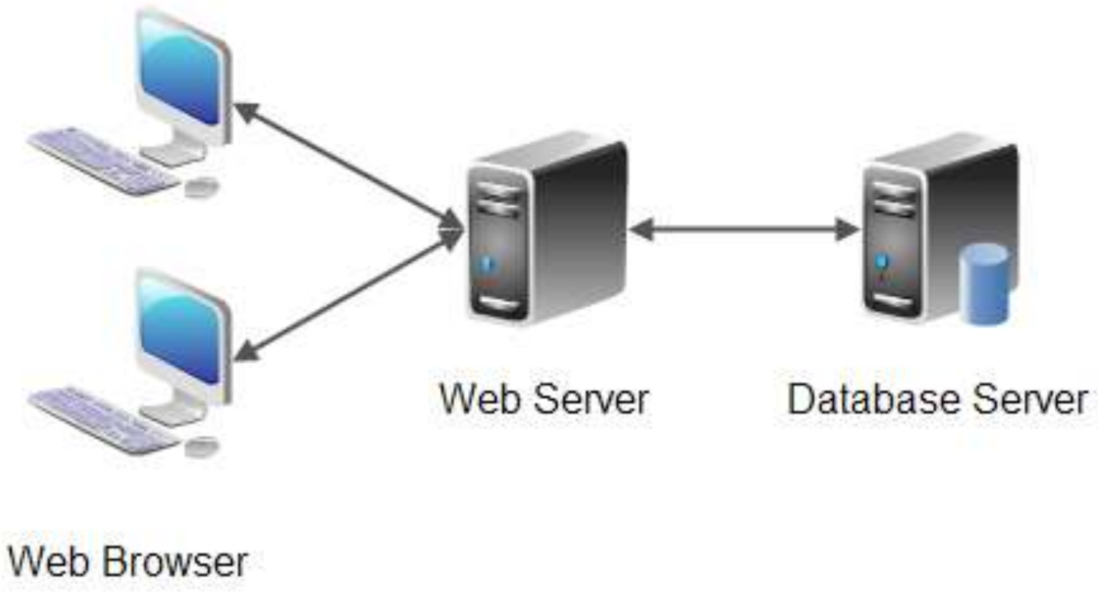
Use of artificial intelligence

Why 3-Tier Architecture ? (Cont..)

- Meets the requirements of large-scale internet and **intranet client/server applications**
- Is easier to **manage and deploy on the network**
- Provides **better security**

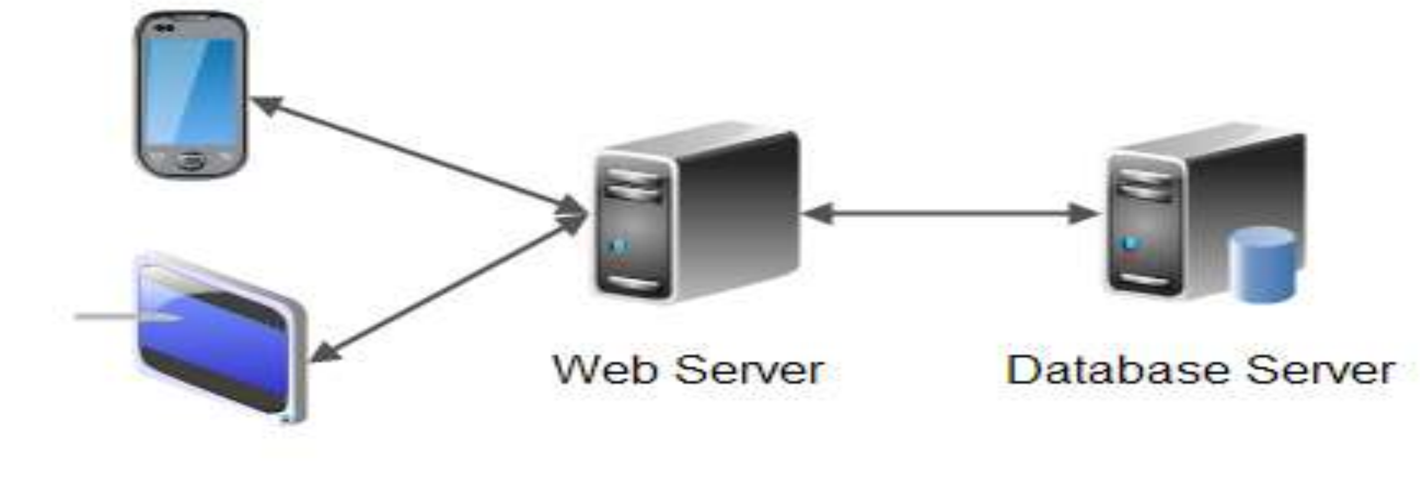
Web Applications

Web applications are a very common example of 3 tier applications. The **presentation tier** consists of HTML, CSS and JavaScript, the **application logic tier** runs on a web server in form of Java Servlets, JSP, ASP.NET, PHP etc., and the **data tier** consists of a database of some kind (mysql, postgresql, a noSQL database etc.). Here is a diagram of a typical 3 tier web application



Mobile Applications

- Actually, it is the same principle with mobile applications that are not standalone applications. A mobile application that connects to a server typically connect to a web server and send and receive data. Here is a diagram of a typical 3 tier mobile application



Another 3-Tier Model

- You could also think of this as a 4-tier architecture:
  - The database management system (**DBMS**)
  - A **persistence manager (** primary interface for application components.)
  - The main **application software**
  - A **GUI**

Another 3-Tier Model (cont..)

- The purpose of N tier architecture is to insulate the different layers of the application from each other. The GUI client doesn't know how the server is working internally, and the server doesn't know how the database server works internally etc. They just communicate via standard interfaces.