

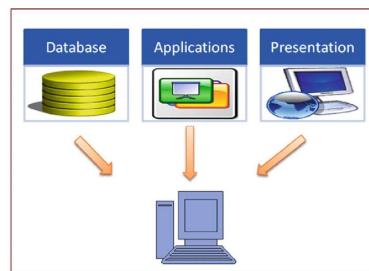
Online Learning Platform

+88 01922 999217



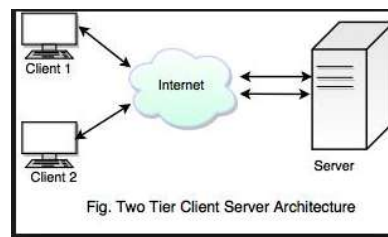
DBMS > Introduction > What is single-tier, Two-tier and Three-tier Architecture of Software?

One-tier architecture involves putting all of the required components for a software application or technology including the interface, middleware and back-end data on a single server or platform. One-tier architecture is also known as single-tier architecture.

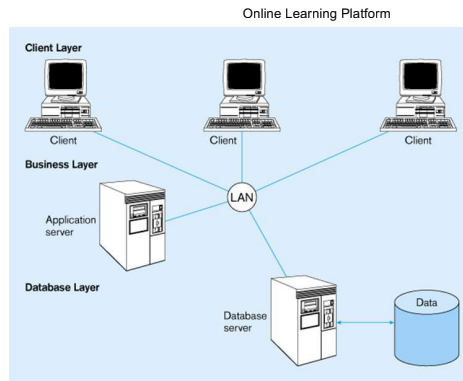


Two-tier architecture is a software architecture in which presentation layer or user interface runs on client machine, and data layer or database stored on a server.

- Each member of the workgroup has a computer,
- Computers are linked by LAN.
- Each computer has a copy of a specialized application (client) which provides the user interface as well as the business logic through which the data is manipulated.
- The database itself and the DBMS are stored on a central device called the "database server,".
- Each member of the workgroup has access to the shared data.
- Different types of group members (e.g. developer or project manager) may have different user views of this shared database.



In **Three-Tier architecture**, intermediate servers handle the client request, which coordinates the execution of the client request with subordinate servers. It adds middle ware(middle tier), which provides a way for clients of one DBMS to access data from another DBMS.



The **objective** of the three-level architecture is to separate the users' view(s) of the database from the way that it is physically represented. This is desirable since:

- **It allows independent customized user views:** Each user should be able to access the same data, but have a different customized view of the data.
- **These should be independent:** changes to one view should not affect others.
- **It hides the physical storage details from users:** Users should not have to deal with physical database storage details. They should be allowed to work with the data itself, without concern for how it is physically stored.

Advantages of three-tier architectures

- **Scalability** – Middle tier can be used to reduce the load on a database sever by using a transaction processing (TP) monitor to reduce the number of connections to a server, and additional application servers can be added to distribute application processing .
- **Technological flexibility** – Easier to change DBMS engines – middle tier can be moved to a different platform. Simplified presentation interfaces make it easier to implement new interfaces.
- **Long-term cost reduction** – Use of off-the-shelf components or services in the middle tier can reduce costs, as can substitution of modules within an application rather than a whole application.
- **Better match of systems to business needs** – New modules can be built to support specific business needs rather than building more general, complete applications
- **Improved customer service** – Multiple interfaces on different clients can access the same business process
Competitive advantage – Ability to react to business changes quickly by changing small modules of code rather than entire applications

Challenges of three-tier architectures

- **High short-term costs** – presentation component must be split from process component – this requires more programming.
- **Tools, training and experience**– currently lack of development tools and training programs, and people experienced in the technology .
- **Incompatible standards** – Few standards yet proposed.
- **Lack of compatible end-user tools** – Many end-user tools such as spreadsheets and report generators do not yet work through middle-tier services.

Prev

Database Administrators and Database Users

Next

What do you mean by Data Model?

Feedback

Type Your Name

Type Your Email

Submit

ABOUT

Statlearner

Statlearner STUDY

Statlearner

©2022 Copyright: [Online Learning Platform](#)