**Dẫn nhập**

Một phần mềm tổng thể được xây dựng nên gồm những phần nào. Và kiến trúc cụ thể của nó ra sao. Để có thể xây dựng thành công một cái nhà thì cần nắm được kiến trúc của nó trong đầu. Phần mềm cũng không ngoại lệ. Chúng ta cùng tìm hiểu về **kiến trúc của phần mềm** nhé!

**Nội dung**

Để đọc hiểu bài này tốt nhất các bạn nên có kiến thức cơ bản về các phần:

* Đã từng sử dụng qua vài phần mềm
* Đã từng suy nghĩ đến việc cấu thành của một phần mềm ra sao
* Biết sử dụng máy tính cũng như các công cụ thành thạo.
* Đã đọc hiểu rõ bài [GIỚI THIỆU VỀ PHÂN TÍCH THIẾT KẾ PHẦN MỀM](http://www.howkteam.com/Course/Phan-tich-thiet-ke-phan-mem/Gioi-thieu-ve-Phan-tich-thiet-ke-phan-mem-1414)
* Đã học qua bài [CHIẾN LƯỢC PHÂN TÍCH ER DIAGRAM](http://www.howkteam.com/Course/Phan-tich-thiet-ke-phan-mem/Chien-luoc-phan-tich-ER-Diagram-trong-PTTKPM-1418)

Trong bài học này, chúng ta sẽ cùng tìm hiểu các vấn đề:

* Giới thiệu kiến trúc Client/Server
* Các thành phần trong kiến trúc Client/Server
* Các loại kiến trúc Client/Server thông dụng

**Giới thiệu kiến trúc Client/Server**

**Kiến trúc client server là**

Kiến trúc tính toán trong đó các xử lý và dữ liệu được phân bố trên các **máy chủ** (server) và **máy khách / trạm** (client) một cách hợp lý nhằm khai thác tối đa khả năng của hệ thống và tăng tính bảo mật của dữ liệu trên server.

Kiến trúc mạng làm việc mà trong đó mỗi máy tính hoặc xử lý trên mạng đóng vai trò như là một máy **client**hoặc **server**.

Mỗi client hoặc server kết nối vào mạng làm việc có thể được gọi là một node

**Server là**

**Máy chủ** hay là máy phục vụ.

Các máy tính hoặc xử lý mạnh phục vụ cho việc quản lý các tài nguyên: ổ đĩa (file server), máy in (printer server), quản lý đường truyền mạng (network server) hoặc thực hiện các tính toán back-end khác.

Lưu trữ và bảo vệ dữ liệu.

Xử lý những yêu cầu được gửi đến từ máy khách.

**Đặc điểm**

Bị động (slave)

Đợi các yêu cầu từ client gửi đến

Đáp ứng các yêu cầu hoặc gửi những thông điệp phản hồi tùy theo yêu cầu của client.

**Ví dụ:**

**Database server:** là một hệ thống máy tính xử lý các yêu cầu truy vấn về cơ sở dữ liệu.

**Mail server:** là một hệ thống máy tính thực hiện các yêu cầu gửi / nhận và lưu trữ email của các người dùng.

**Printer server:** là một hệ thống máy tính xử lý các yêu cầu in ấn từ các người dùng (client)

**Client là**

Là những ứng dụng chạy trên các máy tính sử dụng các dịch vụ hoặc tài nguyên mà server có cung cấp.

Phụ thuộc vào server về mặt tài nguyên: tập tin, thiết bị và các xử lý.

**Đặc điểm:**

* Tạo những yêu cầu.
* Định dạng dữ liệu
* Chủ động (master)
* Gửi yêu cầu đến server
* Đợi cho đến khi nhận được trả lời từ server

Trong kiến trúc client – server, client biết rõ server nhưng server không biết client là ai.

Ứng dụng được mô hình hóa như là một tập hợp các dịch vụ được cung cấp bởi server và một tập hợp các client sử dụng dịch vụ đó.

**Ví dụ:**

**Mail client:** là một ứng dụng cho phép thực hiện yêu cầu gửi mail, đọc mail của người dùng

**Web browser:** là một ứng dụng cho phép người dùng thực hiện những yêu cầu hiển thị một tài nguyên **word wide web**đến một server có cung cấp tài nguyên đó.

**Các thành phần trong kiến trúc Client/Server**

**Tầng giao diện (Presentation layer)**

* Thể hiện các kết quả tính toán cho người dùng
* Tạo báo cáo (report)
* Thu thập dữ liệu từ người dùng
* Hộp thoại người - máy

**Tầng xử lý nghiệp vụ (Application processing layer)**

* Thực hiện các xử lý chức năng của hệ thống

**Tầng truy cập dữ liệu (Data management layer)**

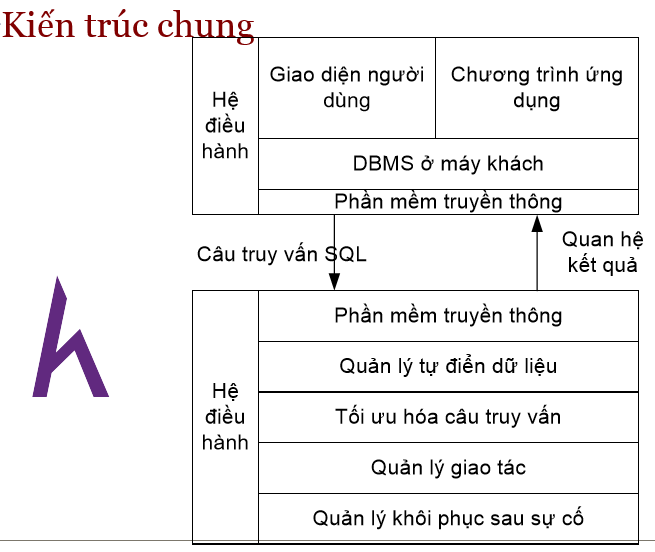
**Các loại kiến trúc Client/Server thông dụng**

**Kiến trúc 2 lớp (Two-tier)**

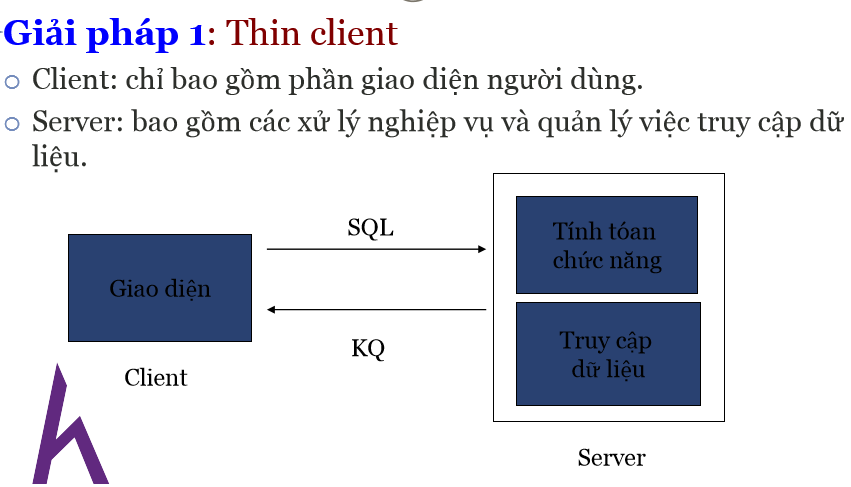
* Kiến trúc 3 tầng của ứng dụng được ánh xạ lên 2 hệ thống máy tính – client và server
* Client có thể là
  + Thin client
  + Fat client

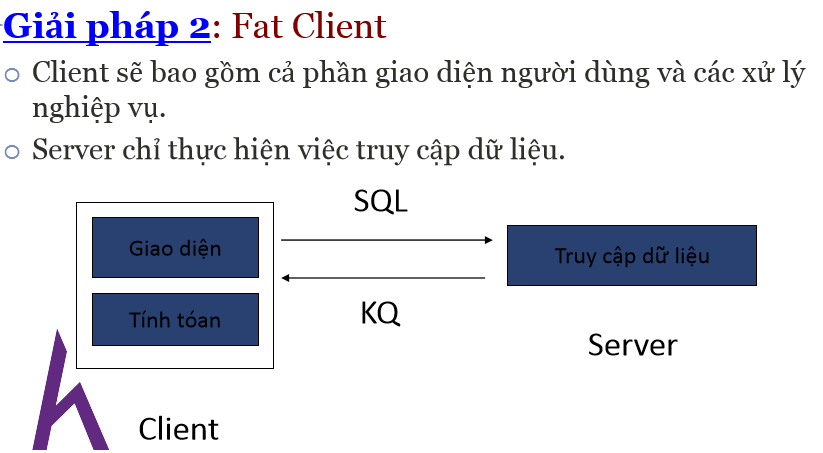
**Kiến trúc 3 lớp (Three-tier: multi client/server architecture)**

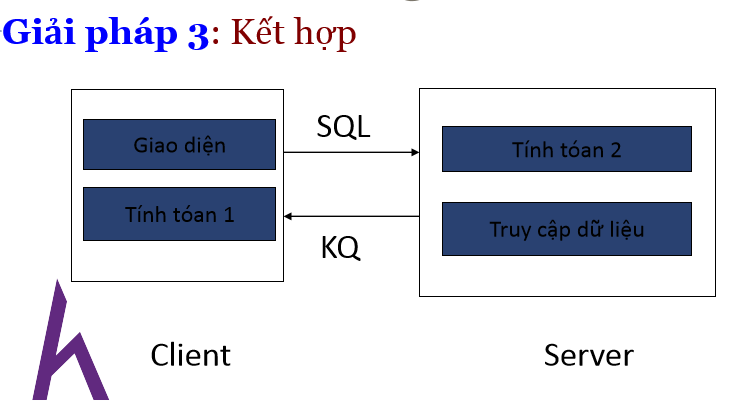
Ba tầng của ứng dụng được ánh xạ lên 3 hệ thống máy tính riêng biệt.

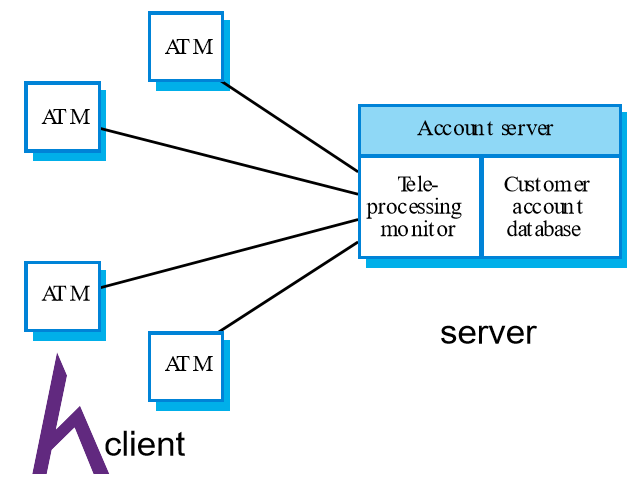


**Kiến trúc 2 lớp**





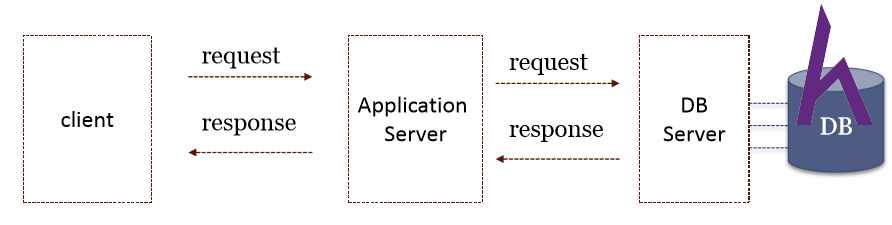


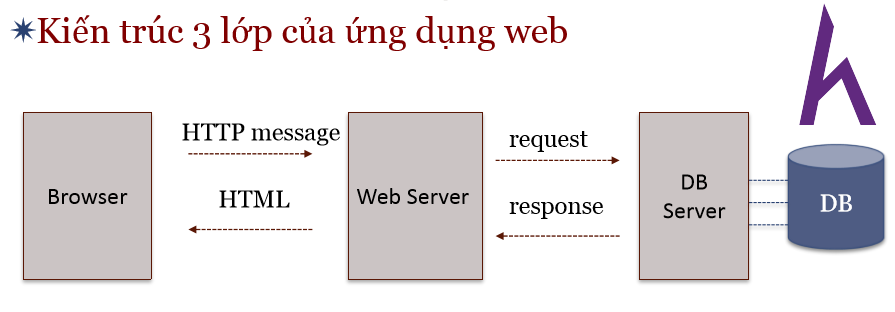


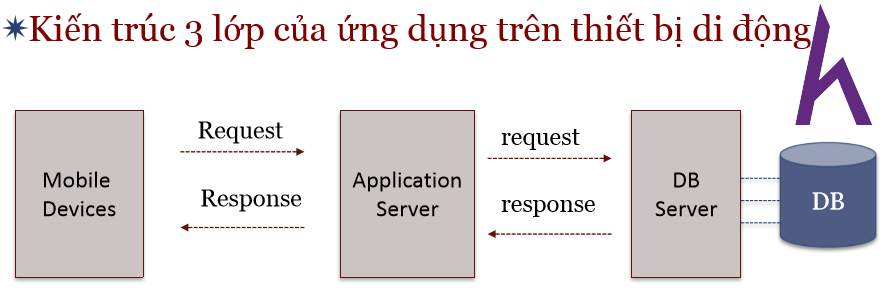
**Kiến trúc 3 lớp**

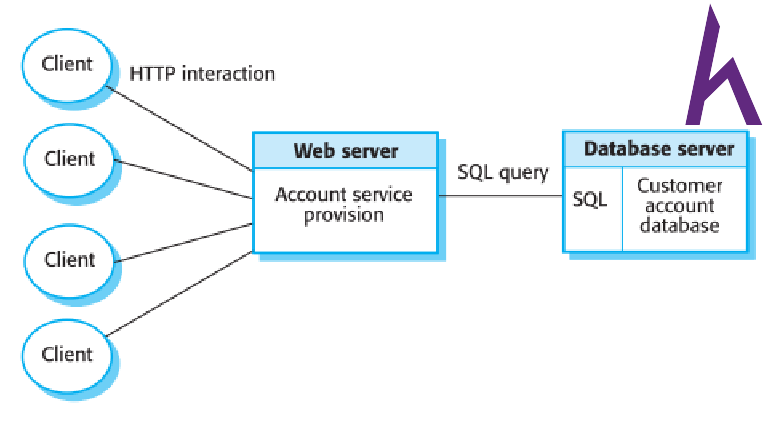
**Các thành phần trong kiến trúc 3 lớp:**

* **Client**: Gửi yêu cầu, dữ liệu và nhận kết quả từ lớp ứng dụng.
* **Server ứng dụng**: thực hiện các xử lý dữ liệu được gửi từ các client.
* **Server cơ sở dữ liệu**: lưu trữ dữ liệu cho lớp ứng dụng, cung cấp các dịch vụ cho các máy tính và chương trình trong lớp ứng dụng. Thường sử dụng các hệ quản trị cơ sở dữ liệu để thực hiện các dịch vụ.









**Các giải pháp phần mềm**

**Microsoft**

* Windows Server
* SQL Server
* Visual Studio

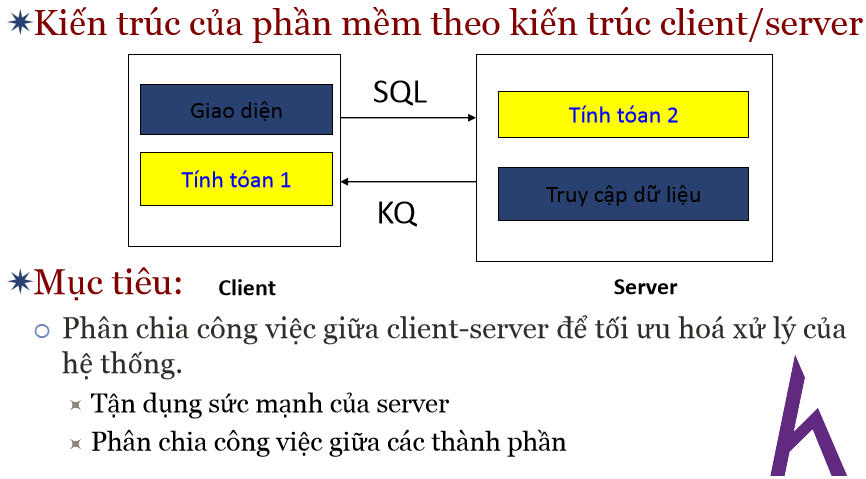
**IBM**

* OS2
* DB2
* Java

**Open source**

* Linux
* MySQL
* Java

**Thiết kế chức năng trên các tầng kiến trúc**



**Cách thực hiện**

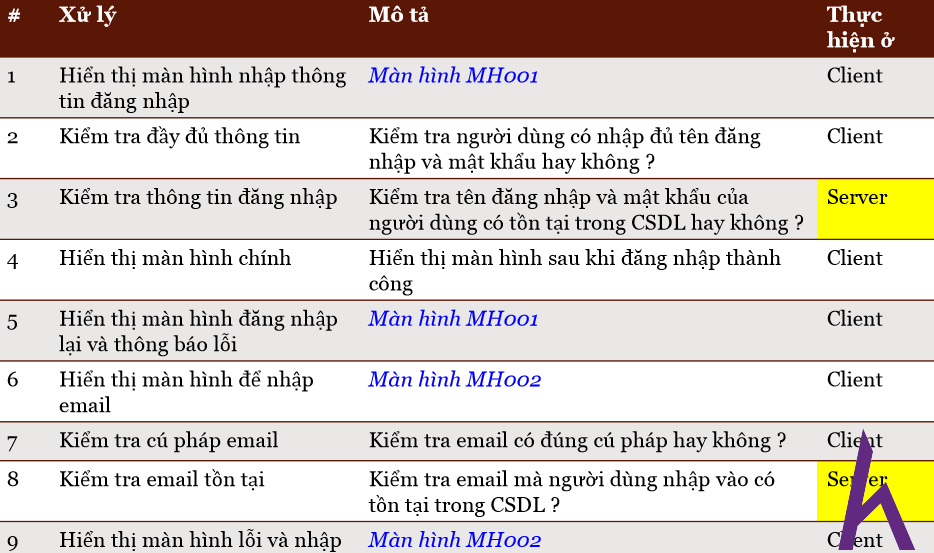
**Với mỗi chức năng của phần mềm (Use-Case):**

**Bước 1:** Tinh chỉnh activity diagram

Một xử lý của lược đồ activity diagram sẽ được thực hiện tại 1 nơi duy nhất (client / server)

**Bước 2:**Xác định các nơi xử lý sẽ được cài đặt và thực thi

**Ví dụ**



**Kết luận**

Bài viết có sử dụng nhiều hình ảnh của tài liệu PTTKHT của trường ĐH KHTN

Qua bài này các bạn đã biết về kiến trúc phần mềm.

Mong qua serial này các bạn đã có một cái nhìn cụ thể hơn về phân tích và thiết kế hệ thống. Mọi thứ đều có kiến trúc và kiến trúc tốt thì hệ thống sẽ phát triển tốt.

Cảm ơn các bạn đã theo dõi bài viết. Hãy để lại bình luận hoặc góp ý của mình để phát triển bài viết tốt hơn. Đừng quên “**Luyện tập – Thử thách – Không ngại khó**”.