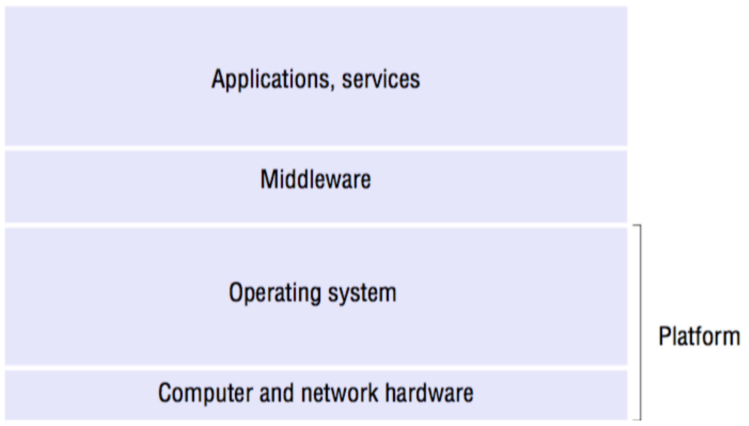
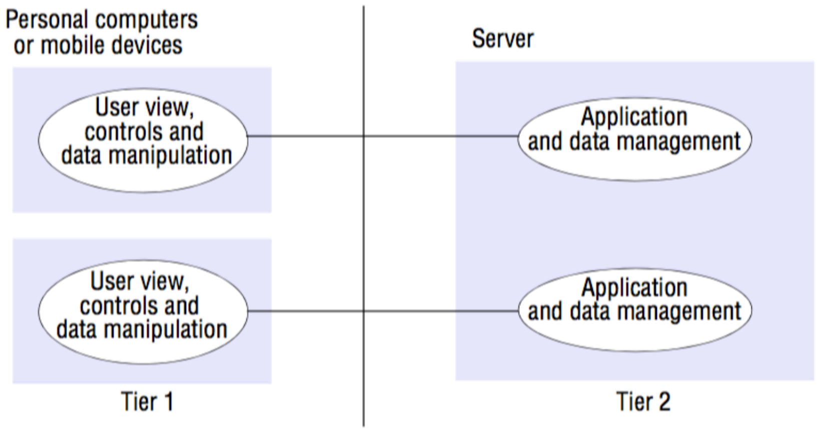
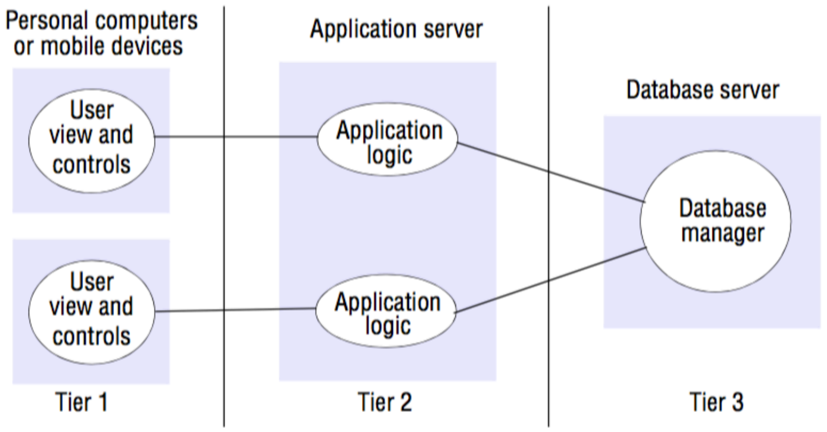
Chapter 1 Introduction

1. Characterization of distributed systems.
    1. components are located on networked computers and execute concurrently
    2. components communicate and coordinate only by passing messages
    3. time differs on each system
    4. many challenges
2. Four styles of application integration.
3. Synchronous
    1. the client calls, blocks and wait for response
4. Asynchronous communication.
    1. call and continues with other business
5. Communication Paradigms:
    1. Interprocess communications
        1. 进程间通信 low level. often use to build high level abstractions. coupled in time. (TCP/UDP/Multicast Socket)
    2. Remote invocation
        1. 远程调用 RPC(remote procedure call), RMI(remote method invocation), HTTP, DCOM, COBRA. Higher level abstractions(two way exchnage with a remote operation)
        2. coupled in time(both exist)
        3. coupled in space(know who)
    3. Indirect communication
        1. 间接通信 communicating to a group be sending a message to a group identifier
        2. publish-subscribe 发布订阅 one to many style
        3. message queues 消息队列 point to point
        4. tuple spaces 元组空间 allow for the placement and withdrawal of structured sequences of data读写不需要同时存在
6. time coupling
    1. senders and receivers need to exist at the same time
7. space coupling
    1. senders need to know who they are sending to
8. Layered architecture(分层体系结构)
    1. vertical organization of service into layers of abstraction

2. 

9. Tiered architecture(层次化体系结构)
   1. complimentary to layering
   2. applied to applications and services layer
   3. main driver: to promote separation of concerns
   4. presenation logic, business, logic and data logic


Personal computers or mobile devices
Server
User view, controls and data manipulation
Application and data management
User view, controls and data manipulation
Application and data management
Tier 1
Tier 2



   5.
   6. two tier: the business logic and user

interface may reside on the client and the data logic layer may be placed on the server. This is the classic client server architecture.

7. In a three-tier solution, the logical description may correspond directly to the physical machines and processes.

10. Software engineering principle: Separation of concerns
11. Architectural patterns
    1. Proxy pattern,
        1. the client makes call on a local objec that has the same interface as a remote object
    2. brokerage pattern
        1. consists of a trio of service provider, service requestor and service broker
12. Challenges in constructing distributed systems
    1. heterogeneity of components may hinder interoperability(存在多样性和差别)
    2. security
    3. scalability
    4. failure handling
    5. concurrency
    6. openness