TIM 50 - Business Information Systems

Lecture 13

Instructor: Terry Allen
UC Santa Cruz 11/7/2011

Outline

- Announcements
- Modularity and Layering (continued)
- Student Presentation
- More on Layering
- Student Presentation
- Components, Suppliers

Announcements

- Assignment 3 will be posted this week
  - Due November 14
- Database Assignment out this week on site
  - Due: November 21
  - Cut-off date: November 21 (30% penalty)
  - Deliverables:
    - ***Hardcopy*** of survey results & report
    - Report & MSAccess files via email
- Forum
  - You can participate in discussion topics
  - This will earn you extra participation points

Announcements

Forthcoming presentations

- 11/12
  - Nataljia Robinett (Sun case study)
- 11/9
  - Are you kidding?! That's when we have our midterm!
- Reading for Monday
  - Messerchmitt Ch.7.2, 7.4.2 (pp.204-226)
  - Messerchmitt Ch. 15
  - MySQL Database Case

Quiz

1. What is a component?
2. Name the 3-tiers in the 3-tier architecture
3. Give 2 layers in a computer infrastructure

Another Interface Example: Automatic teller machine (ATM)

What is the interface between this machine and the customer?

Steps

1. Identify interface building blocks

2. Define available actions

3. Define, for each higher level function, a **protocol**
   - Single action or a finite sequence of actions

1.Interface building blocks

Message on screen or printed
- Menu of actions or returns from an action
- Touch selection of action

Keypad
- Input parameters to an action

Card reader
- Authentication, input parameters

Money output slot
- Returns money

2.ATM actions

A)Authentication

B)Account specification

C)Amount specification

A)Action: authentication

Parameters
- Identity (card in slot)
- Institution (card in slot)
- PIN (typed on keypad)

Internally, it contacts institution and matches against its database, institution noted for all subsequent actions (functionality)

Returns
- Screen message
  - "Invalid PIN", or
  - Menu of available actions

B)Action: specify_account

Parameters
- Account (touch screen from menu of choices)

Internally, choice noted for all subsequent actions (functionality)

Returns
- None

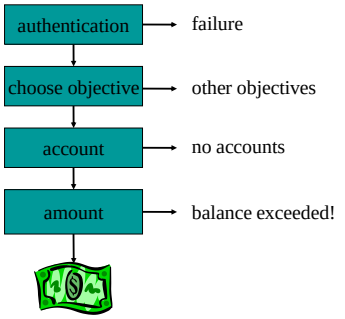
C)Action: amount

Parameters
- Dollars_and_cents (typed on keypad)

Internally, amount noted

Returns
- Success or failure (state dependent, for example for a withdraw failure when dollars_and_cents exceeds balance)

Protocol: cash_withdrawal



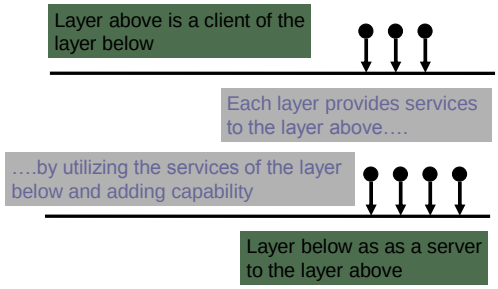
Student Presentations

Morgan Marie Hunt Bus Proj: Netflix

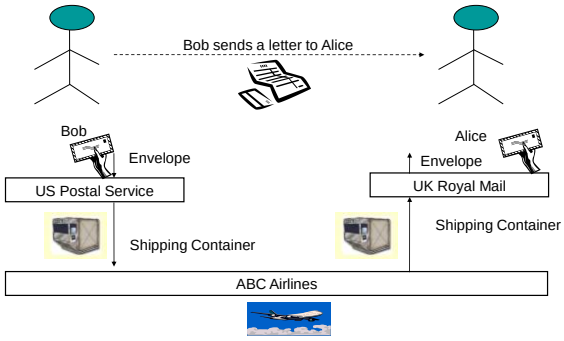
More on layering

by
David G. Messerschmitt

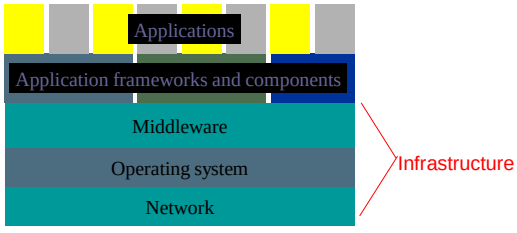
Interaction of layers



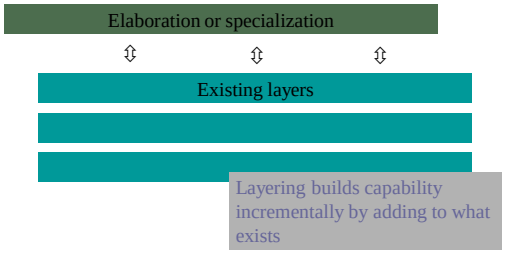
Example 1



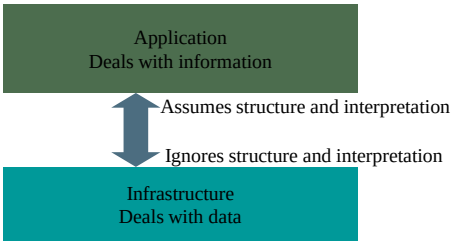
Major layers



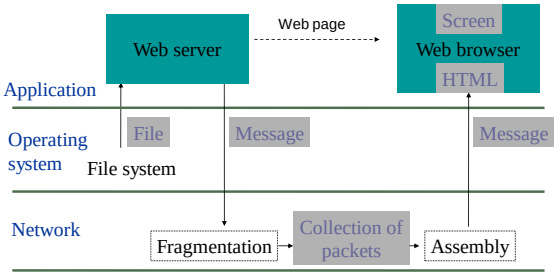
Layering



Data and information



Example 2



Package = file or message

Infrastructure deals with a package of data (non-standard terminology)

- collection of bits
- specified number and ordering

Infrastructure stores and communicates packages while maintaining **data integrity**

→ File for storage

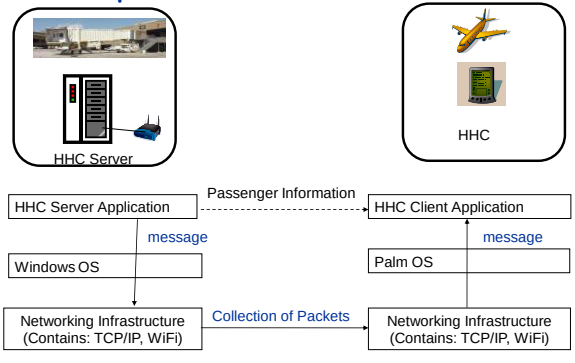
→ Message for communication

Data integrity

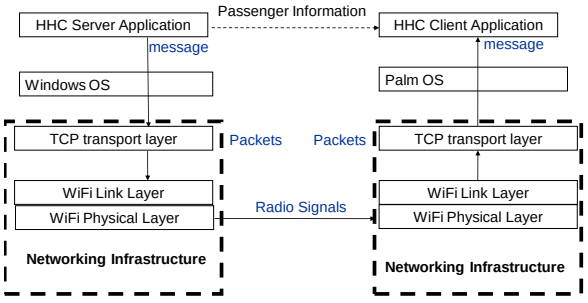
- Nothing is lost/changed in the representation/recovery of information
- Retain the
  - values
  - order
  - number

  of bits in a package
- Also applies to more complicated forms of representation and data processing
  - E.g. Data Integrity in Databases

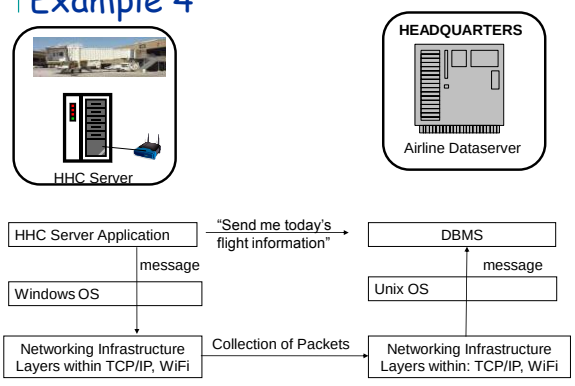
Example 3



Example 3: Network Infrastructure Expanded



Example 4



Data and information in layers

- The infrastructure should deal with data,
  - or at most minimal structure and interpretation

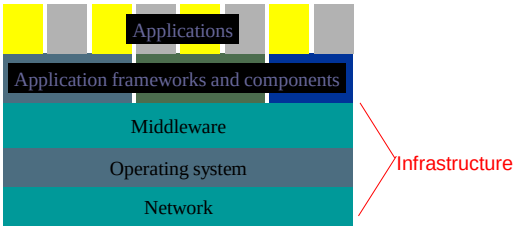
- The application adds additional structure and interpretation

- This yields a separation of concerns

Information in the infrastructure

- *Sometimes* it is appropriate for the infrastructure to assume structure and interpretation for data
  - to add capabilities widely useful to applications
  - to help applications deal with heterogeneous platforms, where representations differ
- Data types

Major layers –Review

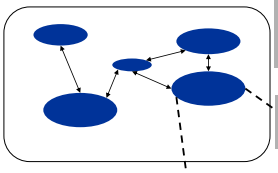


Student Presentations

- Warren Fung (Dell)
- Bryant Yang (Jawbone)

Components, Suppliers (cont'd)

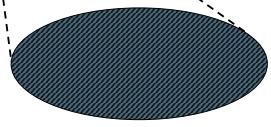
Components (Examples?)



Component: A subsystem purchased "as is" from an outside vendor

(Alternative – building your own subsystem)

A component implementation is encapsulated (although often configurable)



Supplier Types

- Three types of infrastructure/application suppliers:
  - Component Suppliers
  - Custom Subsystem Developers
  - System Integrators
- (Some suppliers are 2 or even 3 of above.)

Two ways to sell Software

Product	Service
Customer installed and operated	Functionality provided over a wide-area network
Often (but not necessarily) sold or licensed at a fixed price	Often (but not necessarily) sold by subscription

Recall: Infrastructure and Applications

Infrastructure

- Equipment and/or software used by many applications

Applications

- Provide specific capabilities and features serving individual users.

Four possibilities (examples)

	Product	Service
Application	Microsoft Office	Hotmail
Infrastructure	Microsoft Windows	Internet DNS (Domain Name System)

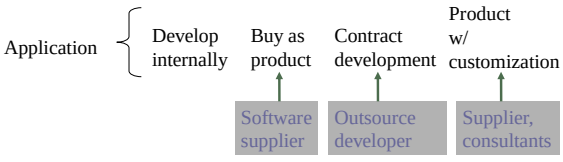
Application Service Providers (ASPs)

- Two types
  - Bundled
    - An infrastructure provider bundles applications with their infrastructure
      - Example: AOL, telephony service providers
  - Unbundled
    - A provider of an application service without providing an infrastructure service
      - Examples?

Examples of unbundled ASP model

- Web-based calendar (e.g. Yahoo, Google)
- Web-based email (e.g. Hotmail, Gmail)
- Web-based stock trading (e.g. Charles Schwab)

Application acquisition

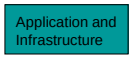


The changing industry structure

Stovepipe vs. Integrated Infrastructure

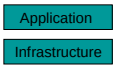
Stovepipe Architecture
---or---
Turnkey Solution

- Single supplier provides all encompassing solution
- (complete with infrastructure)

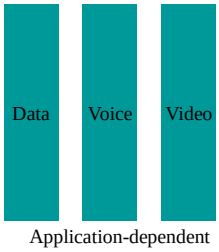


Integrated Infrastucture

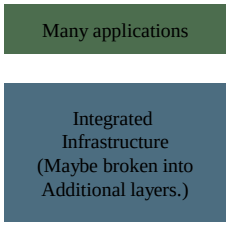
Separate infrastructure that can support many applications



From stovepipe to layering



Application-dependent infrastructure



Application-independent

Stovepipe vs. Integrated Infrastructure

- What are some examples of each?
- What are the advantages of each approach?

Vertical Integration - Diversification

- Two approaches for companies wishing to expand their product offerings
- A company is ***vertically integrated*** when it makes rather than buys the subsystems in its products.
- A ***diversified*** company produces products across different industry segments.

Less Vertical Integration - More Diversification

- Why do customers favor less vertical integration?
  - Prefer competition amongst component suppliers
  - Mix and match components
  - Reduced lock in
- Disadvantages??
  - Customer needs to integrate components from different suppliers.

Less Vertical Integration - More Diversification

- Why do customers favor diversification?
  - Reduce coordination costs by having to deal with fewer suppliers.