Outline For Today

- Class Announcements
- Student Presentation
- Application Architecture
- American Airlines Case
Class announcements

- Database Assignment due date extended!
  - Wednesday May 25th

- Shiv out of town May 17 - 23

- Assignment 5 Due May 18 (Wednesday).

- Read Ch 16 and XML note in reader for Wednesday

- Student Presentations Wednesday
  - Sean Martin
  - Cathy Zhu
Student Presentations

- Jonathon Dwyer
Application Architecture

- **Decomposition** - Divide the architecture into interacting modules.

- **Assembly** - Find subsystems available for purchase

- Most architecture design is a mixture of decomposition and assembly.
Decomposition Example

- Example: manage bank accounts

- Decompose into software modules for
  - transaction processing,
  - statement generation

- Further decompose transaction processing module into deposit and withdraw modules...
Assembly Example

- example - ecommerce platform
  - Acquire
    - Linux pc (application server)
    - IBM Mainframe (data server)
    - Oracle DBMS
    - Apache Web Server Software
    - Websphere application middleware
  - Assemble all pieces together.
  - Mix with custom developed application logic module.
Object-Oriented Architectures

- Object-Oriented Programming (OOP) Languages
  - C++
  - Java
  - Smalltalk

- The basic unit of modularity in OOP is an object.
Real world objects

Consider real-world objects or entities

- Example: A Cat
  - You can see that it is black
  - You observe that it bites when petted
  - You observe that it meows when given food
  - You cannot see the internals of how the cat functions.

- This entity has
  - Attributes (the cat's color)
  - Behavior (it bites when petted)
Software objects

- Also have
  - Attributes
  - Behavior

- For software objects
  - An *attribute* is a numerical value or data that is externally visible, and may be changeable.
  - Ex: The cat is black
A **method** is an action available at the object interface

- Other objects invoke method, pass parameters and get returned data or other objects.

- We can invoke the “feed” method and pass to the cat food, the cat will return a meow or other data.
Software Objects

- The set of all interfaces and methods of an object, as well as accompanying documentation, constitute its **interface**.
- A **class** is a set of objects with the same interface and functionality.

**Example:**
- *Cat* is a class of objects
- *Tom, Sylvester, and Toonces* are *instances* of that class.
Software Objects

- In OOP an object can
  - Represent a real world entity
    - Bank account
  - Be a proxy of a real world entity
    - Customer interface
  - Model a real-world entity
    - Motion of a train
Software Reuse

- Size and complexity of applications growing dramatically
- In order to contain costs, we need to be able to reuse pieces of software
- Reuse is difficult. Why?
- OOP was developed in part to promote reuse, but has had limited success in that regard.
Software components

- Software components are reusable modules that can be bought from outside vendors.
- How is a component different from an object?
  - More importance on
    - Encapsulation
    - Well defined interfaces
Component Assembly Tools

- Visual or integrated development environment (IDE)
  - MS Visual Studio
  - IBM Visual Age
  - Symantec Visual Café

- Scripting Assembly – Text based
  - TCL
  - Perl
  - JavaScript
Software Frameworks

- A preexisting architecture and library of components from a common vendor to help developers
- Enables reuse, and ensures component interoperability.

Examples:
- Sun J2EE/Java Beans
- Microsoft .Net