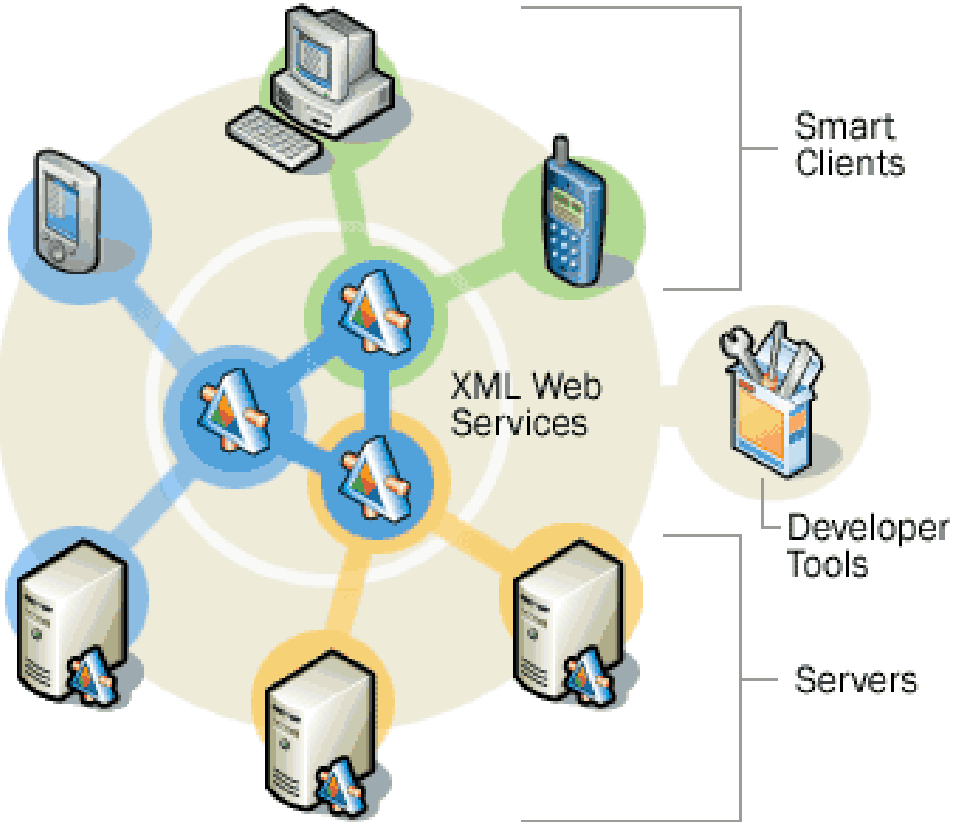
Introduction to Microsoft .NET



Web services are small, reusable applications written in XML, a universal language for data exchange.

Client-to-client
Client-to-server
Server-to-server
Service-to-service

The .NET Strategy

- Interoperability: The .NET is independent from a specific language platform
- Portability: The .NET architecture can exist on multiple hardware platforms
- Web Services are applications that can be used over the Internet as programming blocks
- Software reuse to the Internet: XML and SOAP
- Universal Data Access

The .NET Framework

- The .NET Framework is a development and execution environment that allows different programming languages and libraries to work together seamlessly to create Windows-based applications that are easier to build, manage, deploy, and integrate with other networked systems.

- The .NET Framework consists of:

  The Common Language Runtime (CLR)

  – A language-neutral development and execution environment that provides services to help manage application execution

  The Framework Class Libraries (FCL)

  – A consistent, object-oriented library of prepackaged functionality

The .NET Framework

CLR provides portability between operating systems, interoperability between languages, and execution management, e.g. memory management and security.

A huge Framework Class Library can be used by any .NET language.

High-level Language Interoperability

- Program is first compiled into MS Intermediate Language (MSIL)
- Code converted into MSIL from other languages can be woven together by the CLR
- A compiler in CLR translates the MSIL into machine code, creating a single application
- MSIL makes the .NET Framework language-independent
- Any language that can be compiled into MSIL is called a .NET-compliant language, e.g. Visual Basic .NET, Visual C++ .NET, C#, COBOL, Fortran, J#, Pascal, Perl, etc.

Microsoft Visual Studio .NET

- Developers can use a variety of programming environments to create Web services. Microsoft Visual Studio .NET represents the development environment for .NET-connected software and services.

- Visual Studio .NET advances the high-productivity programming languages: Microsoft Visual Basic, Microsoft Visual C++, and C#.

- Visual Studio .NET gives developers the ease of importing Web services or using Web services hosted remotely and programming against them.

Microsoft Visual Studio .NET

- The .NET Framework's common language runtime (CLR) allows developing Web services using any modern programming language.

- Visual Studio .NET provides a single, unified development environment. Built on the .NET Framework, it provides support for working with Web services created in all modern programming languages.

- Applications and Web services created in one language can be programmed against and debugged in any other language supported by Visual Studio .NET.

Object-Oriented Programming

- What is an Object?
  - Real-world objects share two characteristics: state and behavior.

- An object is a software bundle of variables and related methods.
  - A software object maintains its state in one or more variables. A variable is an item of data named by an identifier.
  - A software object implements its behavior with methods. A method is a function (subroutine) associated with an object.

- Software objects interact and communicate with each other using messages

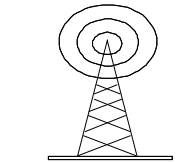
Object-Oriented Programming

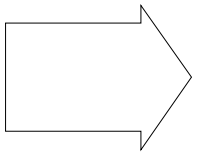
- Object-oriented programming is defined in its purest sense as programming implemented by sending messages to objects. With this definition, problem solutions that use object-oriented programming principles consist of identifying the objects, messages, and object-message sequences to effect a solution.

- Computer languages are object-oriented if they support the four specific object properties called *abstraction*, *encapsulation*, *inheritance*, and *polymorphism*.

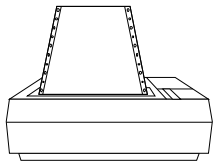
Abstraction

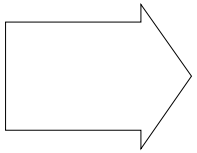
Abstraction is the process of modeling what's important (to a programmer) about a real-world system in a computer representation



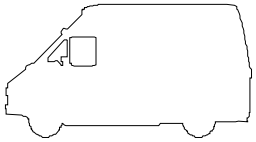


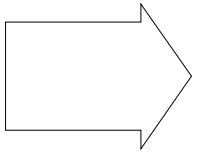
Radio Tower				
Id No	Height	Owner	Location	Power





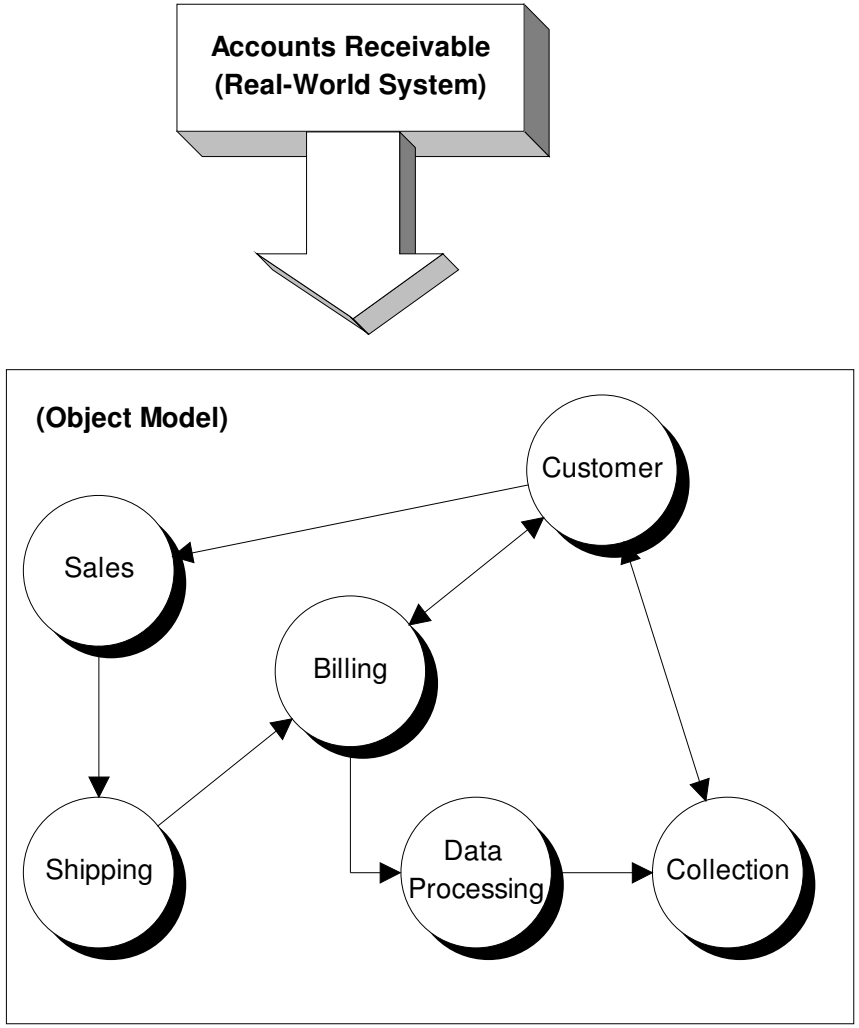
Dot Matrix Printer				
Id No	Make	Model	Location	Paper Size





Truck				
Id No	Make	Model	License	GVW

Abstraction



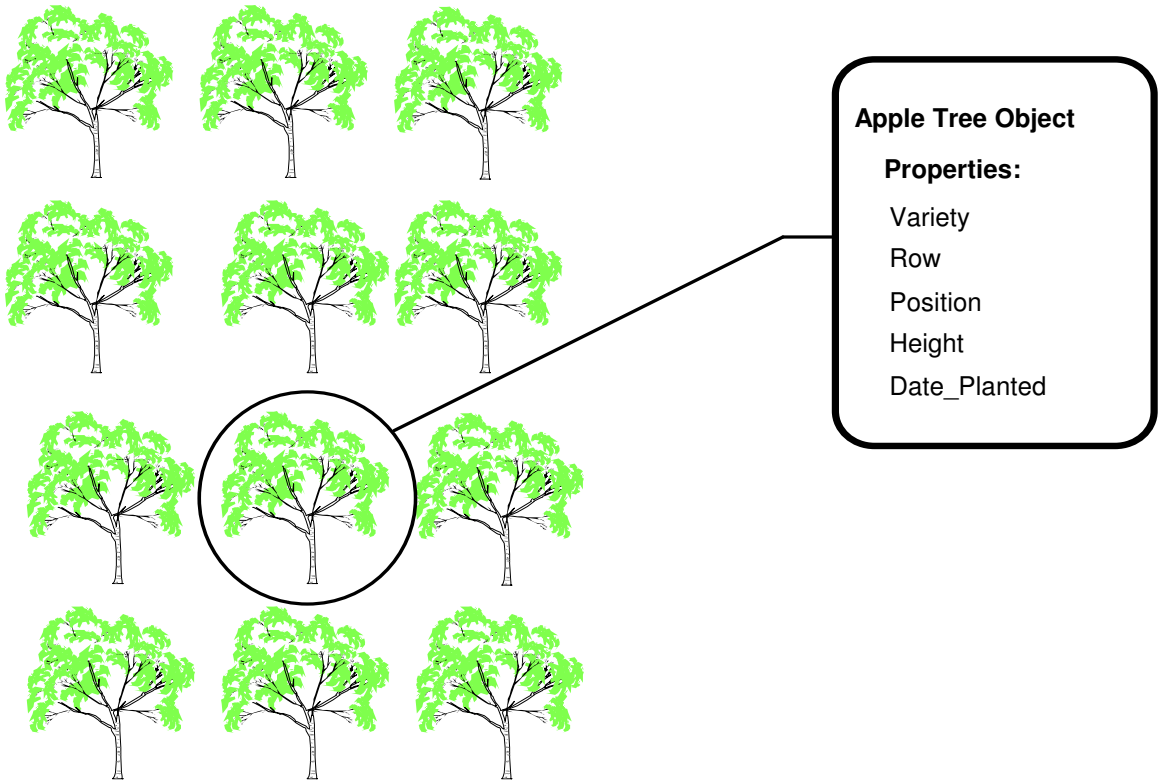
A real-world Accounts Receivable system could be mapped to an OOP model.

The arrows indicate how information might flow through the system. In this model, each object represents some specific entity in the real-world system.

Note that the objects could represent a person, a department, or an automated order entry system that the customer uses via telephone.

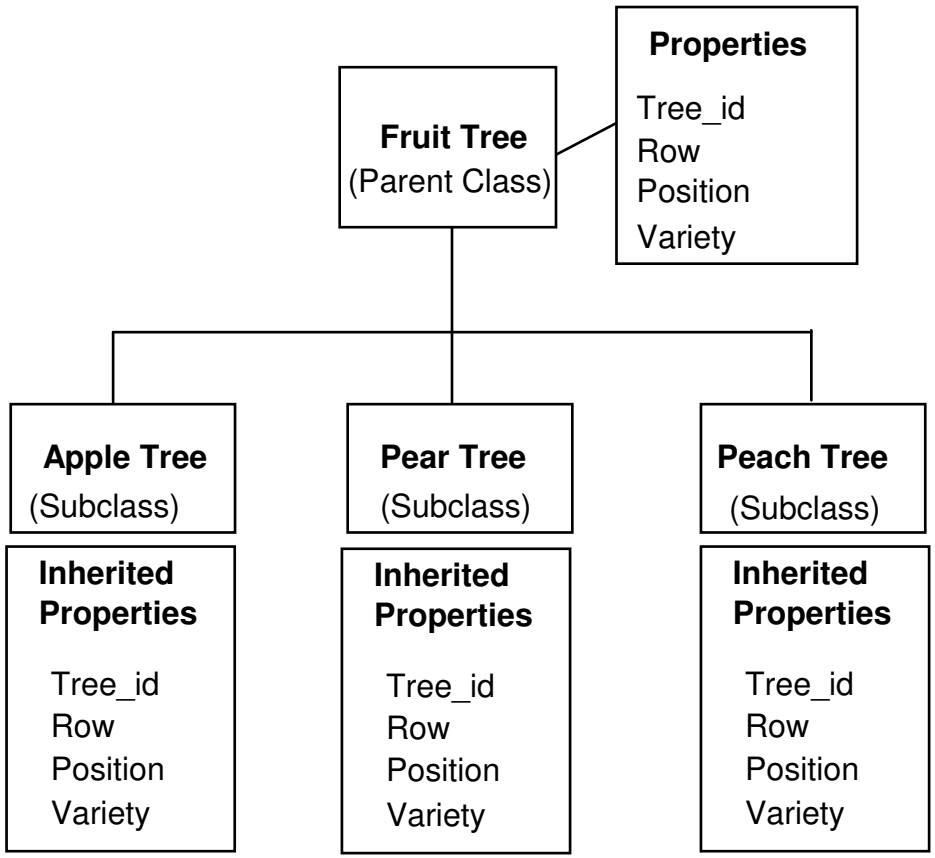
Encapsulation

Encapsulation: Object is an independent self-contained unit.
Data hiding: Hiding the data structures and implementation details of an object from other objects in the system.



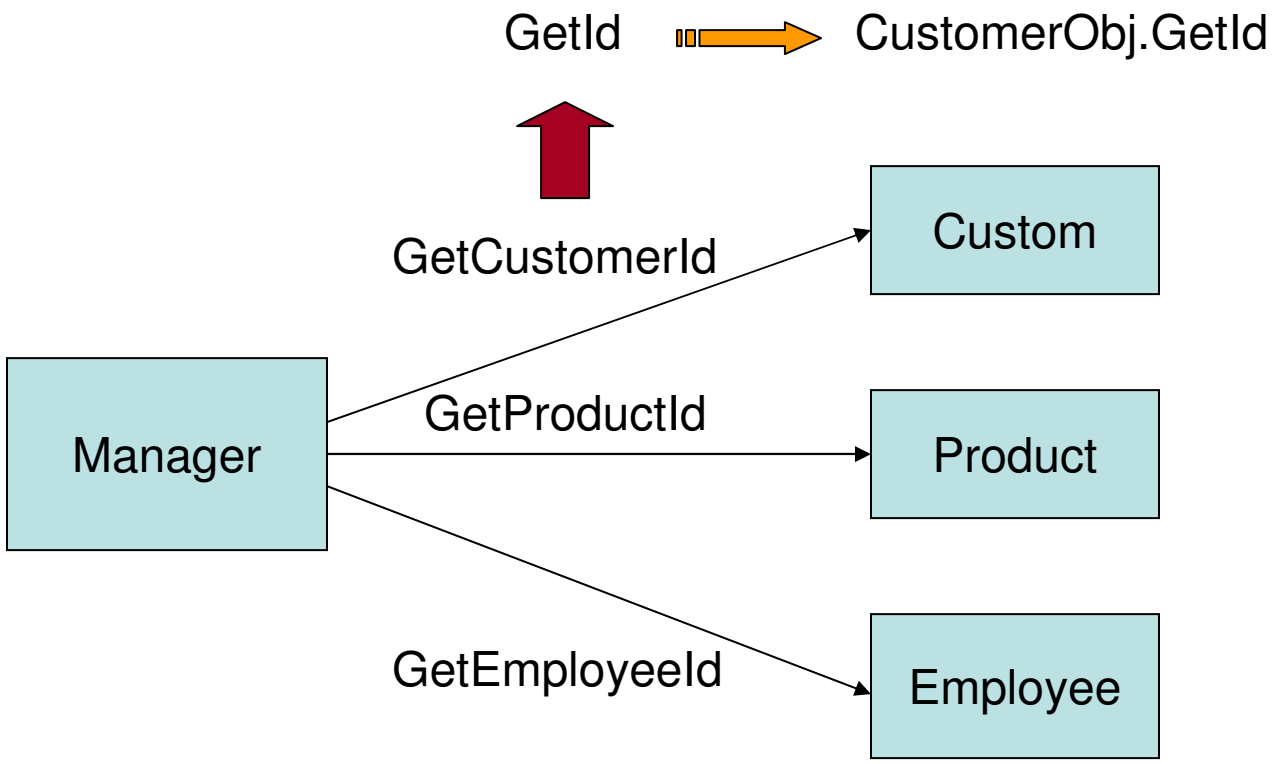
Inheritance

Inheritance builds hierarchies that express the relationship between types. Classes can inherit features from classes higher in the hierarchy.



Polymorphism

Polymorphous means having, assuming, or passing through many or various forms. In the OOP context this refers to the ability to send the same message to related objects of different classes and have each respond according to its own defined purpose.



Summary of OOP

- Object Oriented Programming is a paradigm shift from a function-centric approach to an object-centric approach to software development. That is, instead of focusing on the underlying bits of data and procedures to work with that data, we focus on who or what works with the data, and embed the details within objects.

- In the object model, data and the functionality specific to that data are bundled together (encapsulated). Once this encapsulation is complete, programmers are freed from thinking about the details of how the object works with the data, and can focus on what the object does.

- OOP allows designers and programmers to think in terms of real-world objects rather than the details of how the computer stores and manipulates data.

Classes and Objects

- In object-oriented software, it's also possible to have many objects of the same kind that share characteristics.
- A class is a blueprint, or prototype, that defines the variables and the methods common to all objects of a certain kind.
- A class is a user-defined type; it encapsulates both the data and the methods that work on that data.

Introduction to Visual Basic .NET

- In 1991, Visual Basic (VB) appeared to develop MS Windows – based application.
- Visual Basic .NET is an event – driven, visual programming language in which programs are created using IDE.
- Visual Basic is the world's most widely used rapid application development (RAD) language.
- Visual Basic .NET offers enhanced object orientation, including a powerful library of components.
- Visual Basic .NET enables enhanced language interoperability.
- Visual Basic .NET applications can interact via Internet.
  - a new programming style, in which applications are created from components called Web Services available over the Internet.