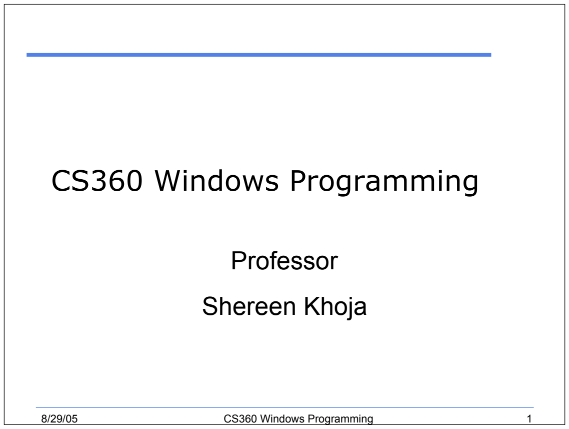
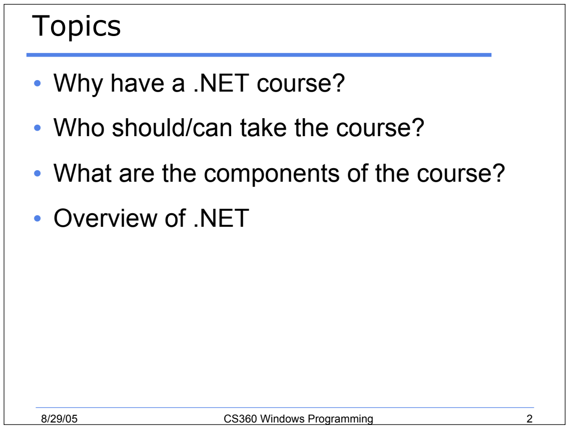


CS360 Windows Programming

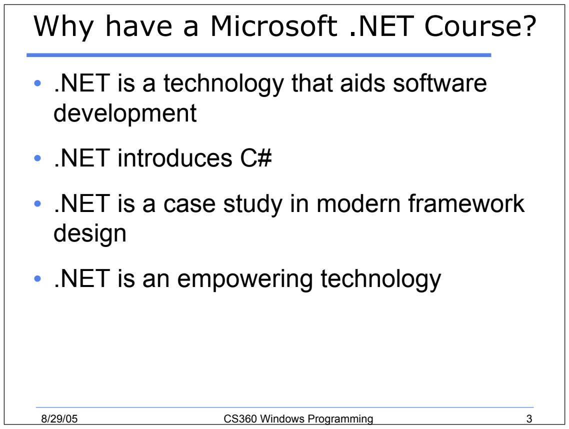
Professor

Shereen Khoja



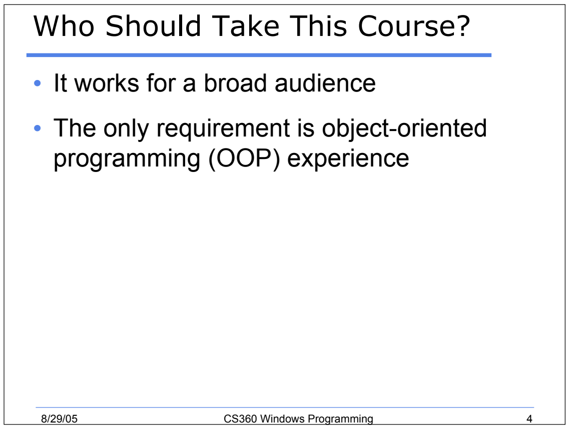
Topics

- Why have a .NET course?
- Who should/can take the course?
- What are the components of the course?
- Overview of .NET



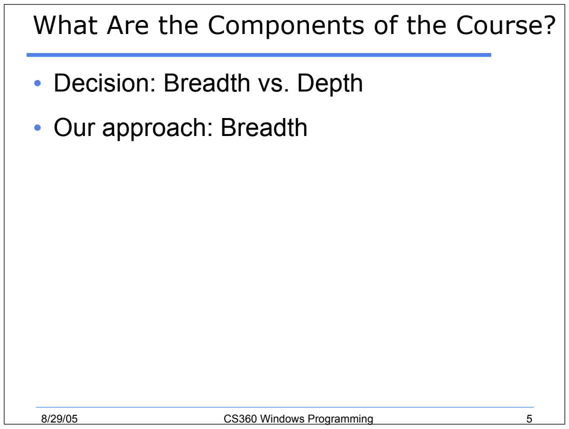
Why have a Microsoft .NET Course?

- .NET is a technology that aids software development
- .NET introduces C#
- .NET is a case study in modern framework design
- .NET is an empowering technology



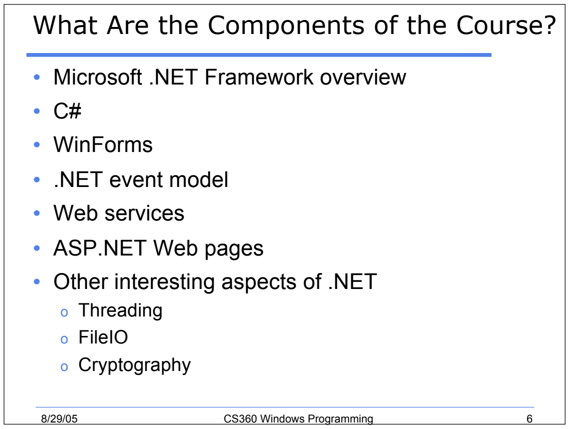
Who Should Take This Course?

- It works for a broad audience
- The only requirement is object-oriented programming (OOP) experience



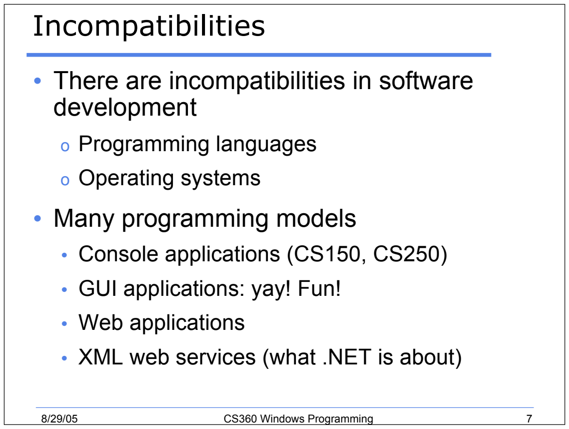
What Are the Components of the Course?

- Decision: Breadth vs. Depth
- Our approach: Breadth



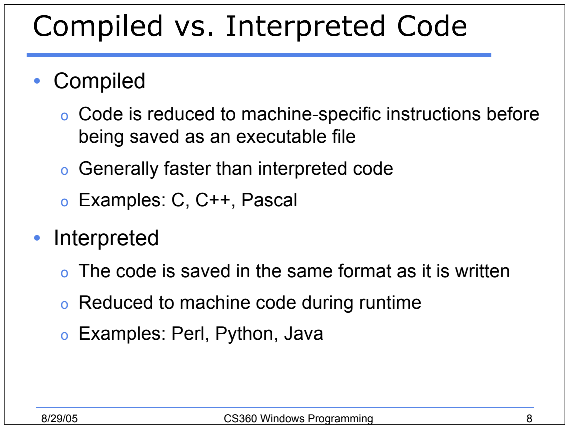
What Are the Components of the Course?

- Microsoft .NET Framework overview
- C#
- WinForms
- .NET event model
- Web services
- ASP.NET Web pages
- Other interesting aspects of .NET
  - Threading
  - FileIO
  - Cryptography



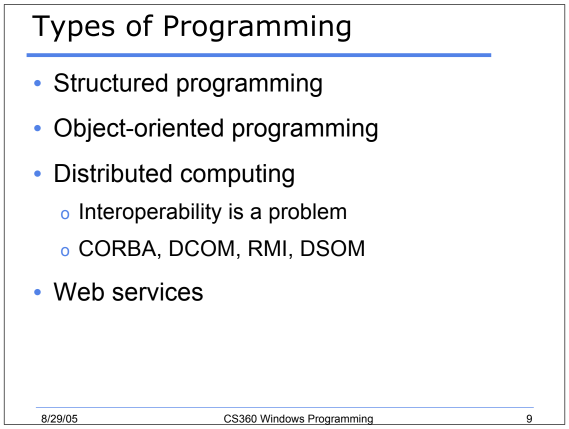
Incompatibilities

- There are incompatibilities in software development
  - o Programming languages
  - o Operating systems
- Many programming models
  - Console applications (CS150, CS250)
  - GUI applications: yay! Fun!
  - Web applications
  - XML web services (what .NET is about)



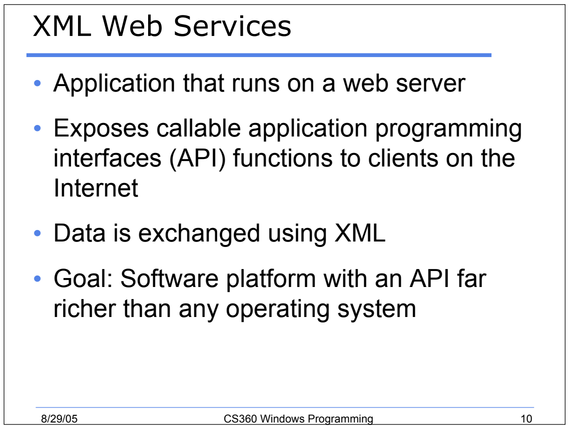
Compiled vs. Interpreted Code

- Compiled
  - o Code is reduced to machine-specific instructions before being saved as an executable file
  - o Generally faster than interpreted code
  - o Examples: C, C++, Pascal
- Interpreted
  - o The code is saved in the same format as it is written
  - o Reduced to machine code during runtime
  - o Examples: Perl, Python, Java



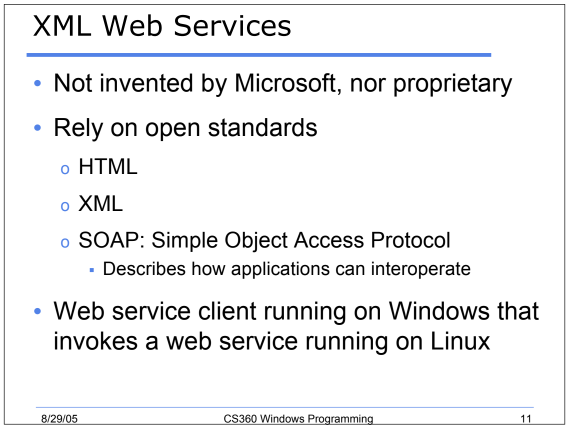
Types of Programming

- Structured programming
- Object-oriented programming
- Distributed computing
  - o Interoperability is a problem
  - o CORBA, DCOM, RMI, DSOM
- Web services



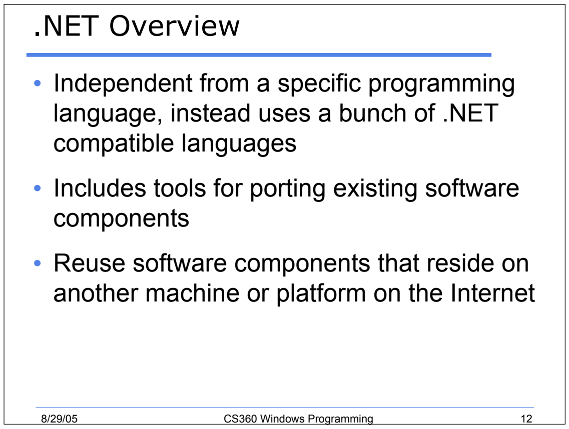
XML Web Services

- Application that runs on a web server
- Exposes callable application programming interfaces (API) functions to clients on the Internet
- Data is exchanged using XML
- Goal: Software platform with an API far richer than any operating system



XML Web Services

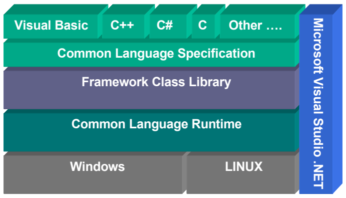
- Not invented by Microsoft, nor proprietary
- Rely on open standards
  - o HTML
  - o XML
  - o SOAP: Simple Object Access Protocol
    - Describes how applications can interoperate
- Web service client running on Windows that invokes a web service running on Linux



.NET Overview

- Independent from a specific programming language, instead uses a bunch of .NET compatible languages
- Includes tools for porting existing software components
- Reuse software components that reside on another machine or platform on the Internet

Microsoft .NET Framework Architecture



Microsoft .NET Framework Architecture

- Manages and executes applications and web services
- Common Language Specification (CLS)
  - Defines the common features of .NET languages and includes information about the storage of data types and objects

Microsoft .NET Framework Architecture

- Framework Class Library (FCL)
  - Contains reusable components that programmers can incorporate into their applications
  - The object-oriented API
- Common Language Runtime (CLR)
  - Executes programs written in any .NET programming language
  - Abstracts operating system services

Steps in Compiling .NET Programs

- Program is compiled into the Intermediate Language (IL)
- IL is then compiled into machine code during execution on a specific platform
- Increases portability and interoperability between platforms

Common Language Runtime (CLR)

- Heart and soul of .NET
- All code runs in the CLR or is given permission by the CLR to run outside of the CLR
- Hosts managed applications

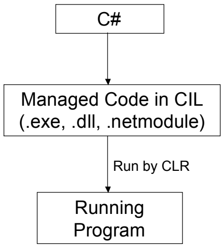
Compiling .NET Programs

- C++ code is compiled into an pseudo-machine language called Common Intermediate Language (CIL)
- This is managed code that is executed when it is run
  - CIL instructions are Just In Time (JIT) compiled into native machine code
  - Methods are compiled only once
  - Code that is never called is never JIT compiled

Why Managed Code?

- The benefits of managed code include:
  - Code is verified to ensure that it's type safe
    - No problem with stray pointers!
  - Resources are garbage collected
    - You allocate memory, the system deletes it for you
    - How did we allocate and free memory in C++ in CS250?
    - No memory leaks!

Managed Modules



C#

Managed Code in CIL (.exe, .dll, .netmodule)

Run by CLR

Running Program

Managed Modules

- Managed modules contain four elements
  - Windows Portable Executable (PE) file header
  - CLR header
  - Metadata describing everything in the module
  - CIL instructions generated from source code
- Metadata helps you figure out what classes and members are inside an executable!

In Summary

- Completed p. 1 - 9 from the book
- Covered these acronyms
  - OOP, GUI, API
  - XML, HTML, SOAP, CLS
  - FCL, CLR, IL, JIT, PE