WINDOWS FORMS FRAMEWORK USING C#.NET

CSCI – 4448 / 6448

October 23, 2008

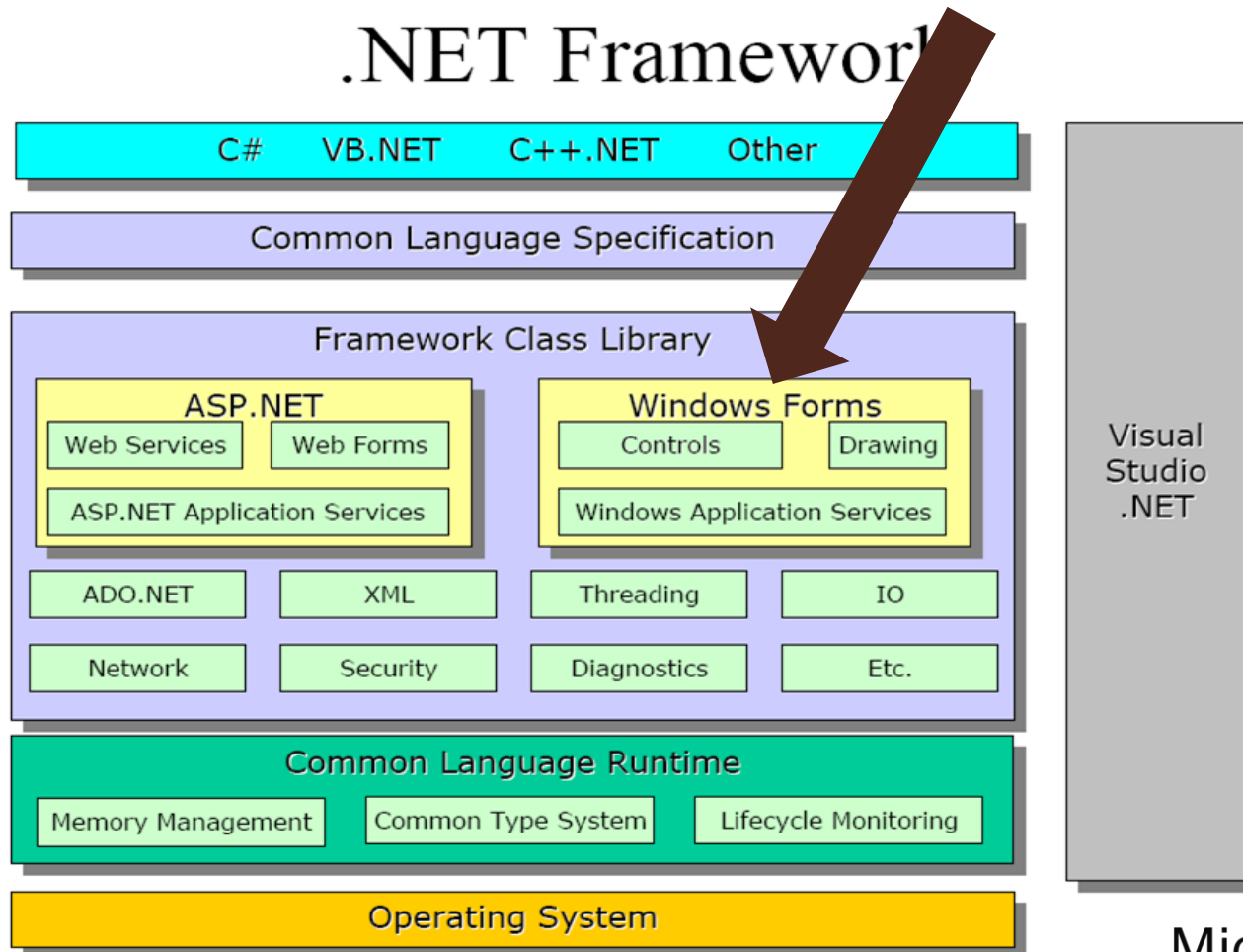Neha Shanbhag
Ajinkya Kulkarni
Mark Lewis Prazen
Kaustubh Sahasrabudhe

A bit of History & Background

- v1.0 released in 2002, v2.0 in 2005, now on v3.5
- A collection of technologies to:
  - Unite applications
  - Simplify development and deployment
- Three main elements:
  - Framework (CLR, FCL)
  - Products (Windows, Visual Studio, Office)
  - Services (WindowsForms, ADO.NET, ASP.NET)
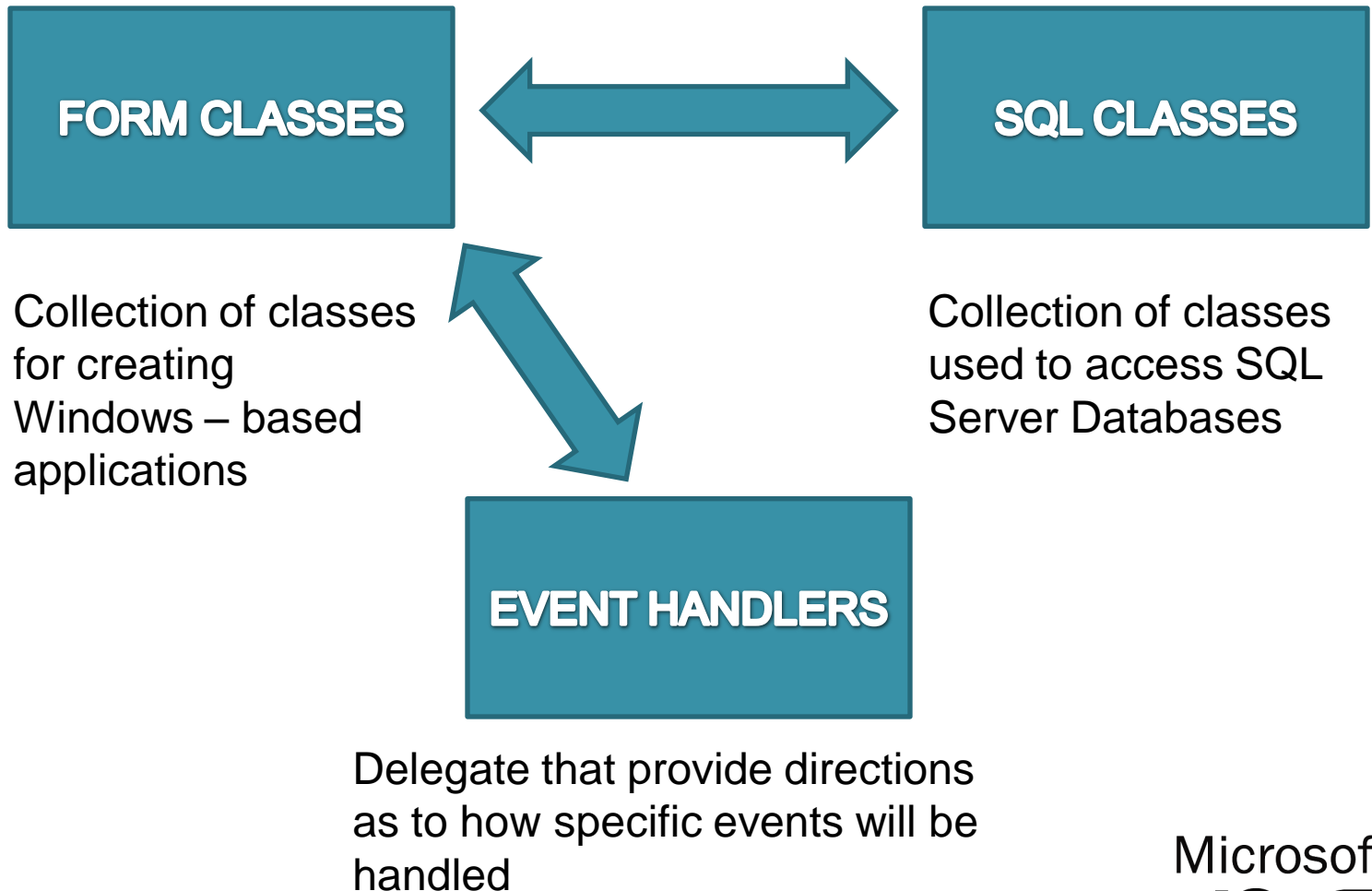
.NET Architecture Overview

Introduction

- ## What is it?
  - A GUI API included as a part of Microsoft's .NET framework
  - Provides access to Microsoft windows interface elements
  - Used for creating event-driven applications

- ## Why?
  - An entrenched application in wide use with a substantial development community
  - Provides a graphical canvas for understanding how OO development works

Microsoft®
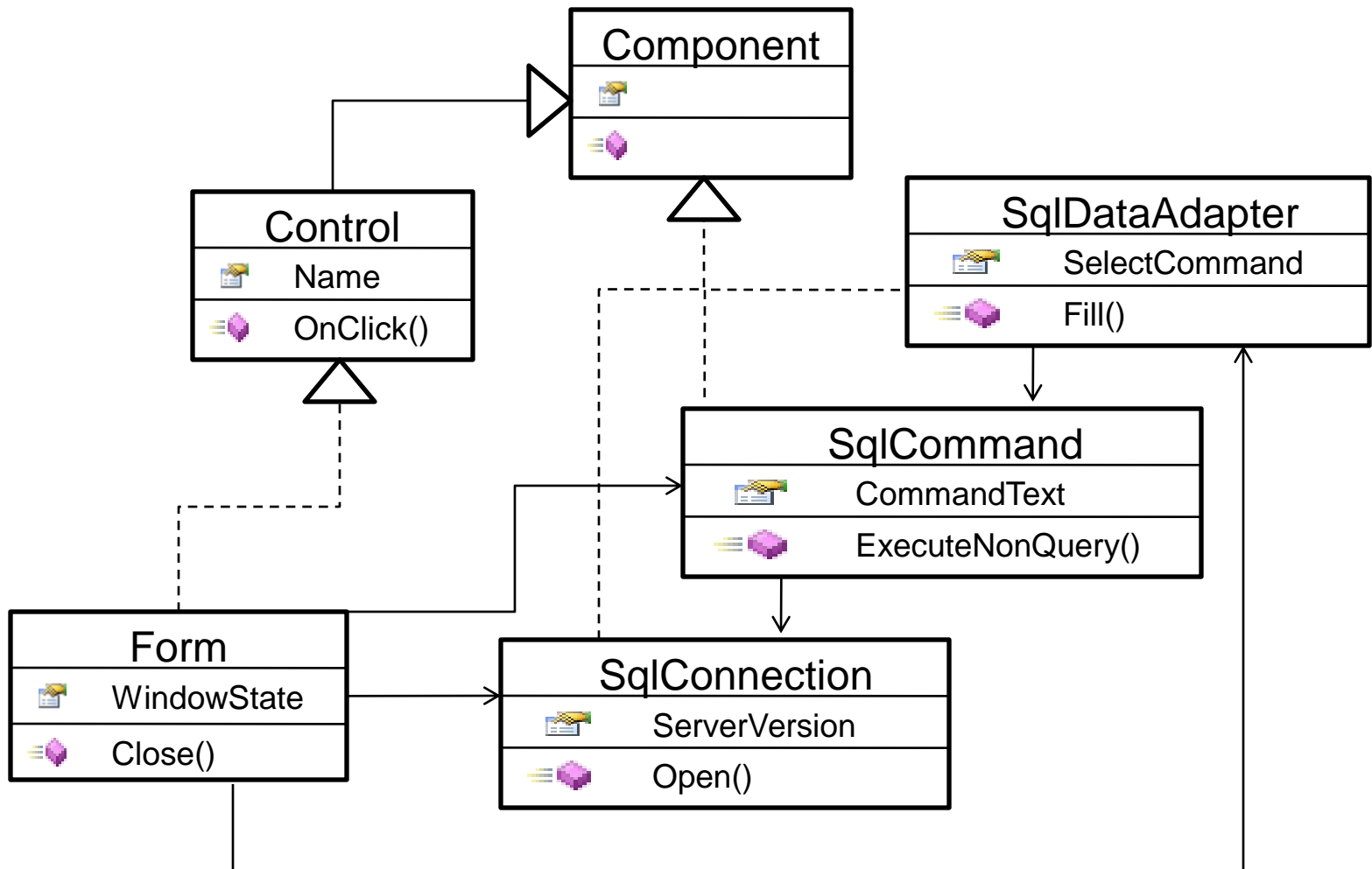.net™

The main services aka modules!

- Standard GUI components
  - System.Windows.Forms
- Supportive components and services
  - System.Drawing
  - System.EventHandler
- Other Application services like ADO.NET
  - System.Data.SqlClient
  - System.Data

Microsoft®
.net™

High Level Design

FORM CLASSES ⟷ **SQL CLASSES**

EVENT HANDLERS

Collection of classes for creating Windows – based applications

Collection of classes used to access SQL Server Databases

Delegate that provide directions as to how specific events will be handled

Microsoft®
.net™

Class Diagram
System.Windows.Forms

Control
Name
OnClick()

Form
WindowState
Close()

Button
ClickMode
OnKeyUp()

Textbox
MaxLength
AppendText()

Point
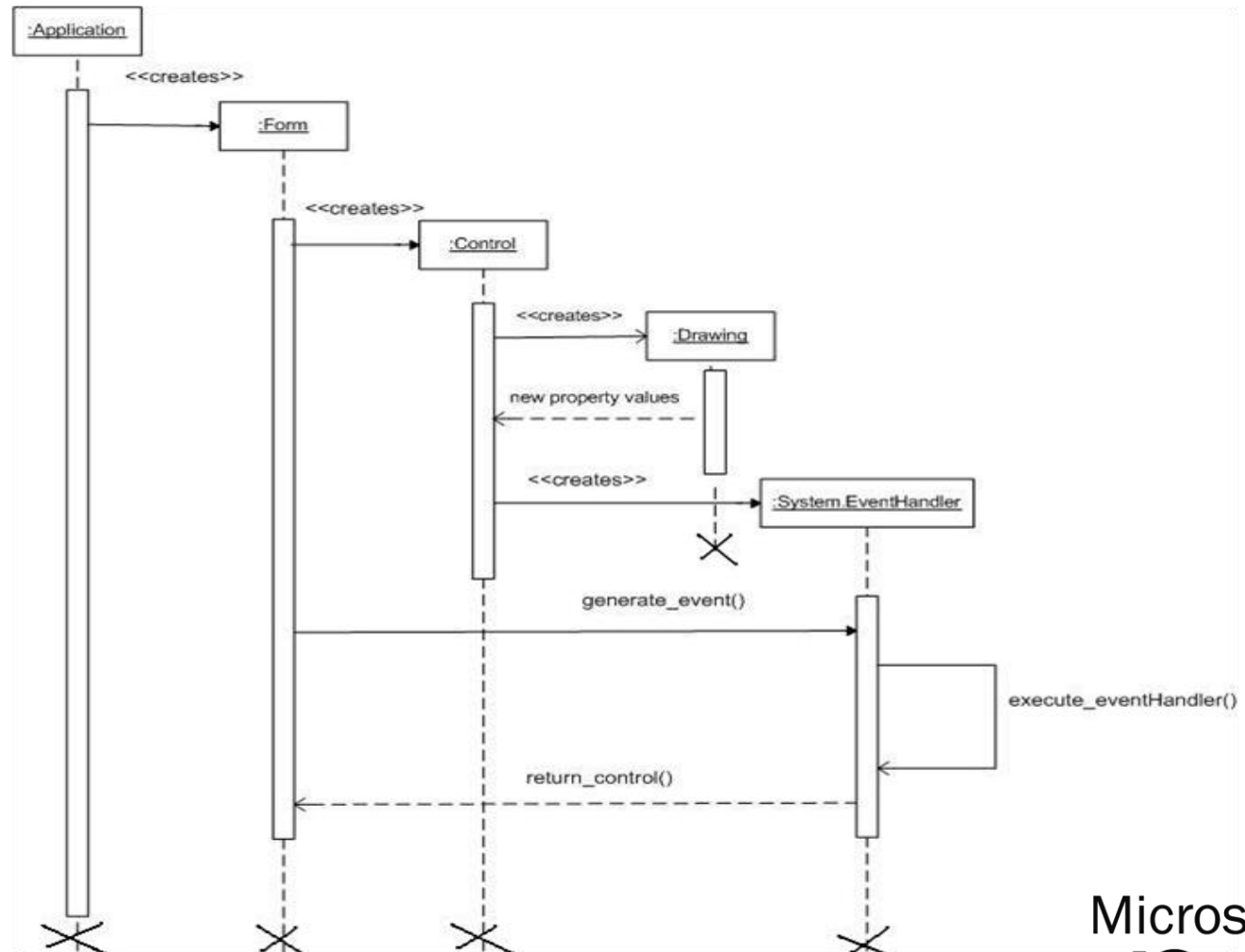
Size

Class Diagram
System.Data.SqlClient

System.EventHandler

- Delegate: The basis for events
- Similar to function pointers in C C++
- Allows programmer to encapsulate a reference to a method inside a delegate object
- Another way to achieve "Polymorphism"

Microsoft
.net

Sequence Diagram EventHandler

Demo Application

- Desktop Database for Dr. Anderson to manage the grades of the students in the OOAD class
  - Demonstrates the use of Windows Forms and Controls
  - Utilizes event handlers to trigger actions based on control events
  - Interacts with database to store and import data

Microsoft®
.net™

OO Concepts & Principles

- Polymorphism
  - Inheritance hierarchy
- Code to an interface
  - OnClick
- DRY
  - Delegation
- SRP
  - TextBox vs. TextBlock
- OCP
  - New service eg. Pie Menus
- LSP
  - EventHandlers

Microsoft
.net

Conclusion / Evaluation

- ## Advantages
  - Comprehensive form creation application
  - Easy to learn and use with IDE
  - Linkages to a broad framework provides robust tools and features

- ## Disadvantages
  - Proprietary nature hinders portability. Applications not Mac or Linux compatible.
  - MS resources increasingly more focused on web
  - Framework imposes constraints on the application that might not be encountered in a more open and sparser architecture

Microsoft®
.net™

Questions??

Microsoft®
.net™