# Windows Presentation Foundation

Graphical User Interface Design
Using .NET and WPF

Adam Hilton
Software Architect II
TransWorks

# What is a User Interface?

- The space where interactions between humans and programs occur
  - Commands
  - Data
- Programs need input
- Humans need output

# Command Line Interfaces (CLI)

- Textual commands in a specialized vocabulary
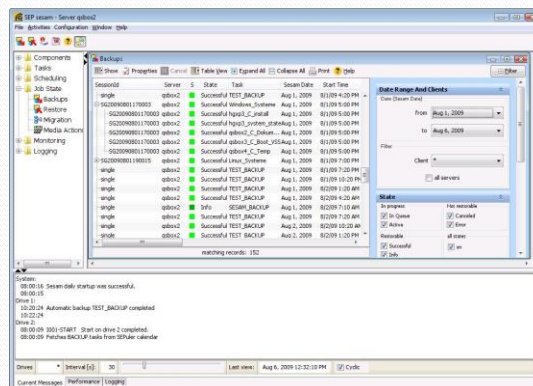- *Yes, CLI's are still in <u>wide</u> use today*
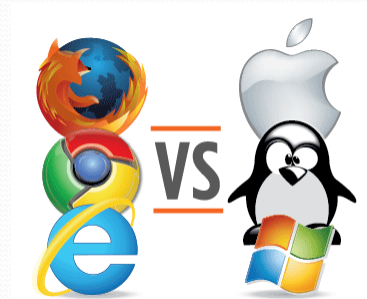


# Graphical User Interfaces (GUI)

- A system which uses a mouse, pointers, hypertext, and multiple windows
  - Desktop
    - WinForms
    - WPF
  - Web
    - HTML
  - Mobile
    - Android
    - iOS

# Desktop vs. Web Applications

- Thick/thin client
- Installation
- Latency
- Complexity
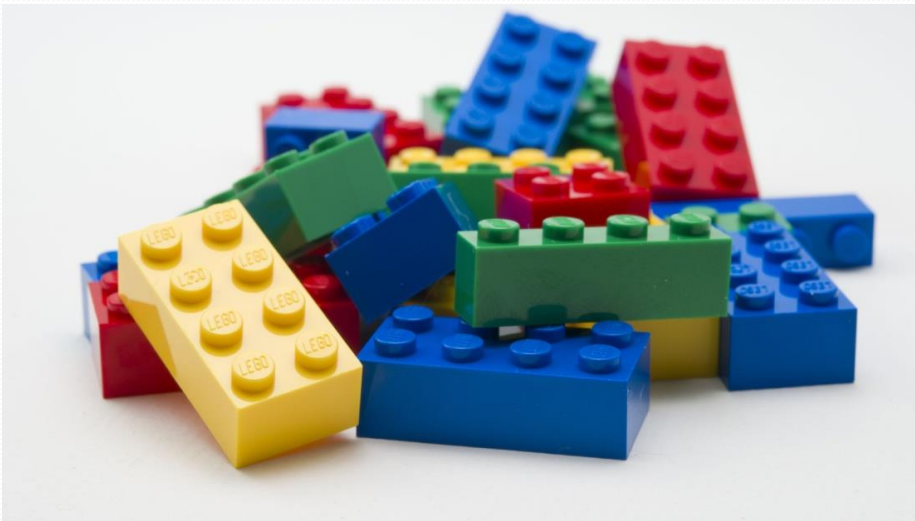- Multitasking
- User experience

# Windows Desktop Applications

- Visual Basic (6.0)
  - 1991
  - Still around, but losing support
- WinForms
  - 2001 (.NET 1.0)
  - VB upgradable
  - Most widely used for legacy applications
- WPF
  - 2006 (.NET 3.0)
  - Most widely used technology for new applications

# Why WPF over WinForms?

- Extensibility
- Data binding
- Better performance
  - 2D/3D
    - Hardware acceleration
- Cross platform (Silverlight/XBAP)
- Learning curve

# WPF vs. WinForms - Extensibility

# WinForms - Extensibility

Click Me!!!

# WPF - Extensibility

# WPF - Extensibility

- The Lego pieces are called, "User Controls"
  - XAML
  - C# or VB in the background
- Containers vs. single child

- *Example....*

# WPF - Data Binding

- Provides a simple and consistent way for applications to present and interact with data
- Modify the underlying data once
- All user controls consuming that data automatically update

- *Example....*

# WPF - Designer

- "Visual Basic" style designing
  - More control with raw XAML
- Toolbox
  - Drag and drop
  - XAML automatically written for you
- Visualizes control as they're written

- *Example....*

# WPF - Examples

- TimeBlox