WPF CONTROLS

Jim Fawcett

CSE775 – Distributed Objects

Spring 2011

REFERENCES

- Programming WPF, 2nd Edition, Sells and Griffith, O'Reilly, 2007
- Silverlight 4 Unleashed, Bugnion, SAMS, 2011
- MSDN: Data Binding Overview

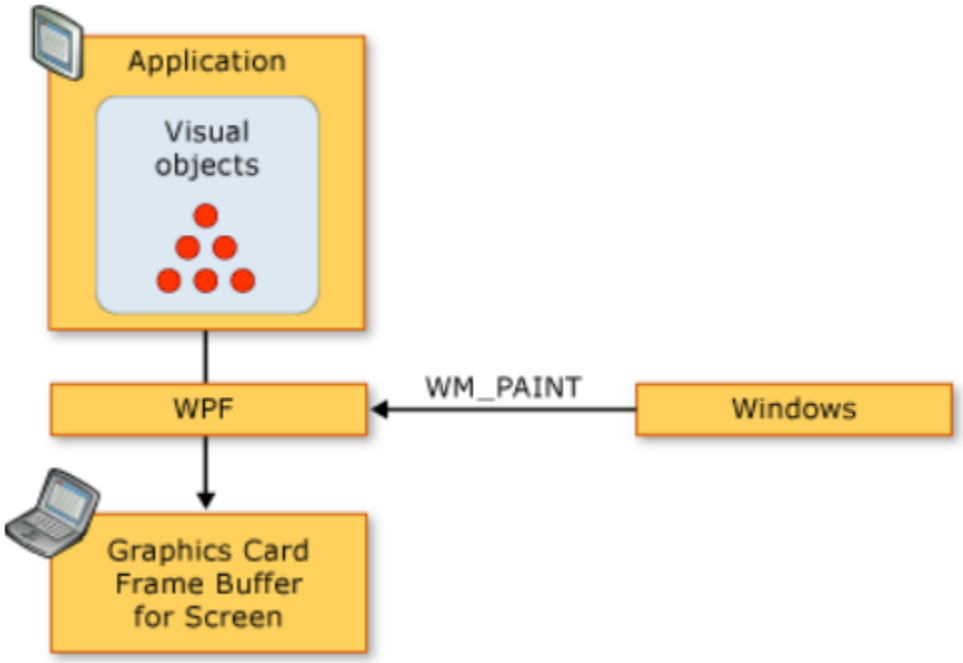
WPF - THE PARTS

- Rendering Model
- Dispatcher
- INotifyChange
- DependencyObject
  - Change notification
  - Attached Properties
- ObservableCollection
- Data Binding
  - Markup Extensions
- Control Template
- Data Template
- Resource Dictionaries
  - Markup Extensions

RENDERING MODEL

- WPF Graphics Rendering
- Visual class is base for all FrameworkElement classes.
  - Holds the Hwnd of the underlying Win32 Application Window.
  - Supports:
    - Rendering serialized, persisted drawing content
    - Transformations and clipping
    - Hit testing
    - Bounded Boxes
- Uses Retained Graphics Mode
  - Drawing information is persisted in a serialized state by application, but WPF is responsible for rendering – you never call invalidate().
  - Uses Painter's Algorithm

Diagram of WPF rendering sequence



DISPATCHER

- WPF_DispatcherDemo
- MSDN: WPF Threading Model
- WPF uses a message loop, similar to Win32 programs
  - Adds priority queuing
- All WPF applications have at least two threads
  - Message pump thread, managed by Dispatcher
  - Rendering thread
- Windows have thread affinity, e.g., the STA model.
- Worker threads return results to the UI thread for display
  - They use UIElement.DispatcherObject.Dispatcher.Invoke(…)
  - DispatcherObject, base for UIElements, holds the UI Dispatcher as a property and provides CheckAccess() and VerifyAccess() methods to see if the operation needs to be dispatched through Dispatcher.Invoke(…)

CHANGE NOTIFICATION

- WPF_ChangeNotification
- Implement Change Notification
- All WPF property binding works through change notification:
  - Every UIElement derives from DependencyObject
  - DependencyObject implements INotifyPropertyChanged
    - One element, the PropertyChanged event
  - Binding:
    - Subscriber uses UIElement.DataContext.Source as Publisher
    - Subscribes to its PropertyChanged event
    - Compiled Xaml provides a handler for the subscription that handles the property change.

DEPENDENCY OBJECT

- All UIElements derive indirectly from DependencyObject
- DependencyObject provides:
  - Facilities for one-way and two-way property change notification, based on the INotifyPropertyChanged interface
  - A property dictionary used to hold values of properties, including values for container properties, called attached properties.
    - A UIElement registers its dependency properties using Register method.
    - UIElement registers its attached properties using RegisterAttached.
  - Values returned from the property system are evaluated in a multistep process:
    - Coercion, animation, local value, parent template properties, implicit style, style triggers, template triggers, style setters, default style, inheritance, default value defined in dependency property metadata

OBSERVABLE COLLECTION

- Observable Collections
- A dynamic data collection that provides notifications when items are:
  - Added
  - Removed
  - List is refreshed
- Implemented by:
  - ObservableCollection<T>
  - List<T>
  - Collection<T>
  - BindingList<T>

DATA BINDING

- WPF_DemoPanels
- MSDN Data Binding Overview
- The markup extension:

    targetProp = “{Binding Path=PropName}”

  attempts to subscribe to the PropertyChange event of the object listed as source in the closed DataContext.

CONTROL TEMPLATE

- WPF_ControlTemplateDemo
- MSDN: ControlTemplate Class
- Awaiting more slide content

DATA TEMPLATE

- WPF_DataTemplateDemo
- MSDN: DataTemplate Class
- Awaiting more slide content

RESOURCE DICTIONARIES

- WPF_DemoPanels
- MSDN: WPF Resources
- Slide awaiting more content