

**DataModel-View-
ViewModel in WPF**

Filling the Gap Between the
Presentation and View



A Little Background...

- John Blumenauer
- Started with .NET 1.0
- Works at Metrico Wireless
- Enjoys talking and learning about .NET



In the Remaining 28 Minutes...

- Goals
 - Understand DataModel-View-ViewModel
 - Learn why WPF and DMVVM work well together.
 - See it in action!

A vertical column of ten white squares of varying sizes is positioned on the left side of the slide. The squares are arranged in a roughly descending order of size from top to bottom, with some squares appearing in pairs or small groups.

So Where Did it Come From?

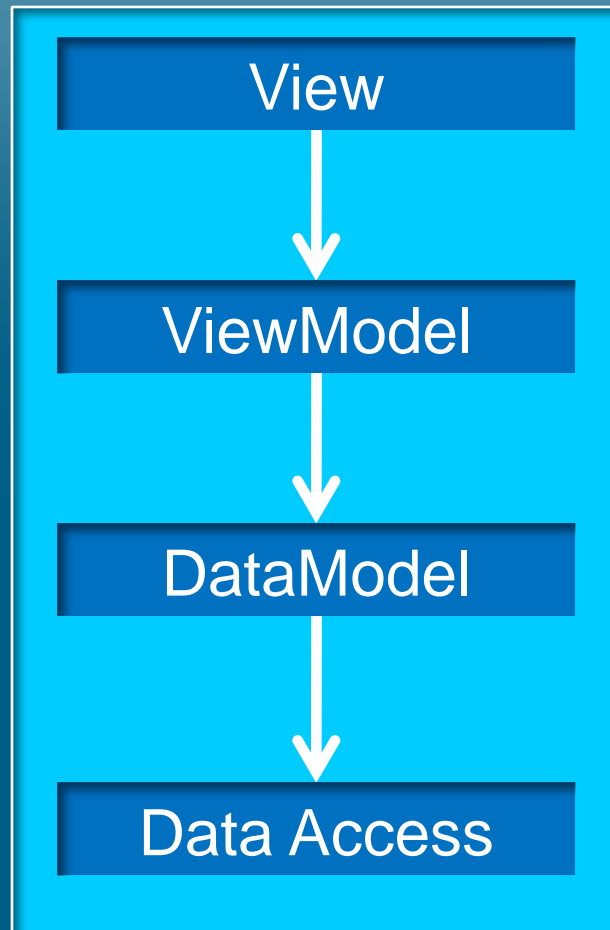
- DataModel-View-ViewModel
- Sometimes called Model-View-ViewModel
- Martin Fowler's "Presentation Model" is similar.
- John Gossman of MSFT Published it 1st.
- Dan Crevier of MSFT also contributed with a multi-part series.

A vertical column of ten white squares of varying sizes is positioned on the left side of the slide. The squares are arranged in a roughly descending order of size from top to bottom, with some smaller squares interspersed among the larger ones.

So What Does it Do?

- **DataModel**
 - *Objects representing problem domain.*
- **View**
 - *UI controls that display “things”*
- **ViewModel**
 - *UI-friendly objects implement UI-specific interfaces*
 - *Stores the UI data and state*
 - *Application code works with VM objects, not UI*

The DMVVM Pattern



A vertical column of ten white squares of varying sizes is positioned on the left side of the slide. The squares are arranged in a roughly descending order of size from top to bottom, with some smaller squares interspersed among the larger ones.

DMVVM has Classic Roots

- MVC and MVP serve as the DMVVM basis
- ViewModel is a morph of the classic Controller/Presenter
- Model and View is joined by the ViewModel.
- ViewModel is an abstraction of a View.



So Why Does WPF and DMVVM Work Together so Well?

- In a word...Databinding
- View and ViewModel are joined by binding.
- View initiates Commands and requests actions from a bound ViewModel

A vertical column of ten white squares of varying sizes is positioned on the left side of the slide. The squares are arranged in a roughly descending order of size from top to bottom, with some squares appearing in pairs or small groups.

So What's the Cost/Benefit?

- Cost
 - Learning Curve
 - Development Time
- Benefits
 - Great Separation
 - Simple Unit Testing
 - UI Design and Workflow
 - UI can be Easily Re-Skinned



Demo/Sample Code





Summary

- DMVVM = DataModel-View-ViewModel
- Databinding is Key.
- View is not attached to logic, only ViewModel
- ViewModel is an Abstraction of a View.



Related Links

- John Gossman - Tales from the Smart Client - <http://blogs.msdn.com/johngossman>
- Dan Crevier's Blog - <http://blogs.msdn.com/dancre/archive/2006/07/23/676272.aspx>
- Josh Smith - Inside XamSalesManager - <http://community.infragistics.com/articles/application-design/inside-xamsalesmanager-3.aspx#mod>
- The Orbifold - <http://www.orbifold.net/default/?p=550>