

Using the MVVM Design Pattern with the Microsoft Visual Studio 2010 XAML Designer

Microsoft®
tech·ed
Africa | 2010

OCTOBER 17-20, 2010 | DURBAN, SA



Developer Tools, Languages, and Frameworks



Level 300



Rudi Grobler

Barone, Budge & Dominick



Blog

<http://www.rudigrobler.net>



Twitter

@rudigrobler



DTL323 - Using the MVVM Design Pattern with the Microsoft Visual Studio 2010 XAML Designer
Mon, 18 Oct 2010 (10:45 - 11:45) | Breakout Session | Sessions Room C3 | Level: 300 - Advanced

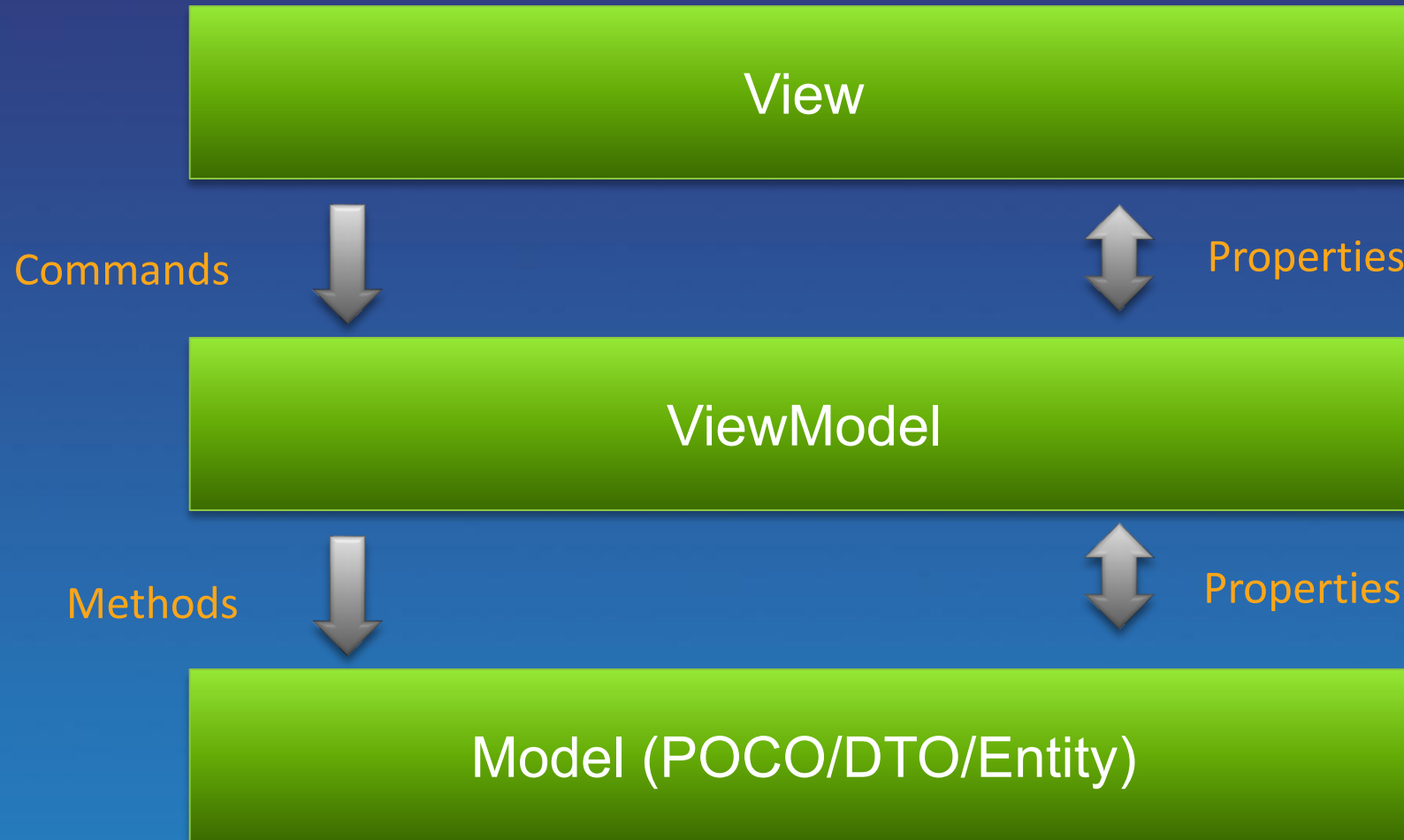
WTB312 - Powering Rich Internet Applications: Windows Server AppFabric, Web Services, and Microsoft Silverlight
Tue, 19 Oct 2010 (14:30 - 15:30) | White Board | Session Room D2 | Level: 300 - Advanced

WUX310 - Securing Microsoft Silverlight
Tue, 19 Oct 2010 (17:15 - 18:15) | Breakout Session | Session Room A3 | Level: 300 - Advanced

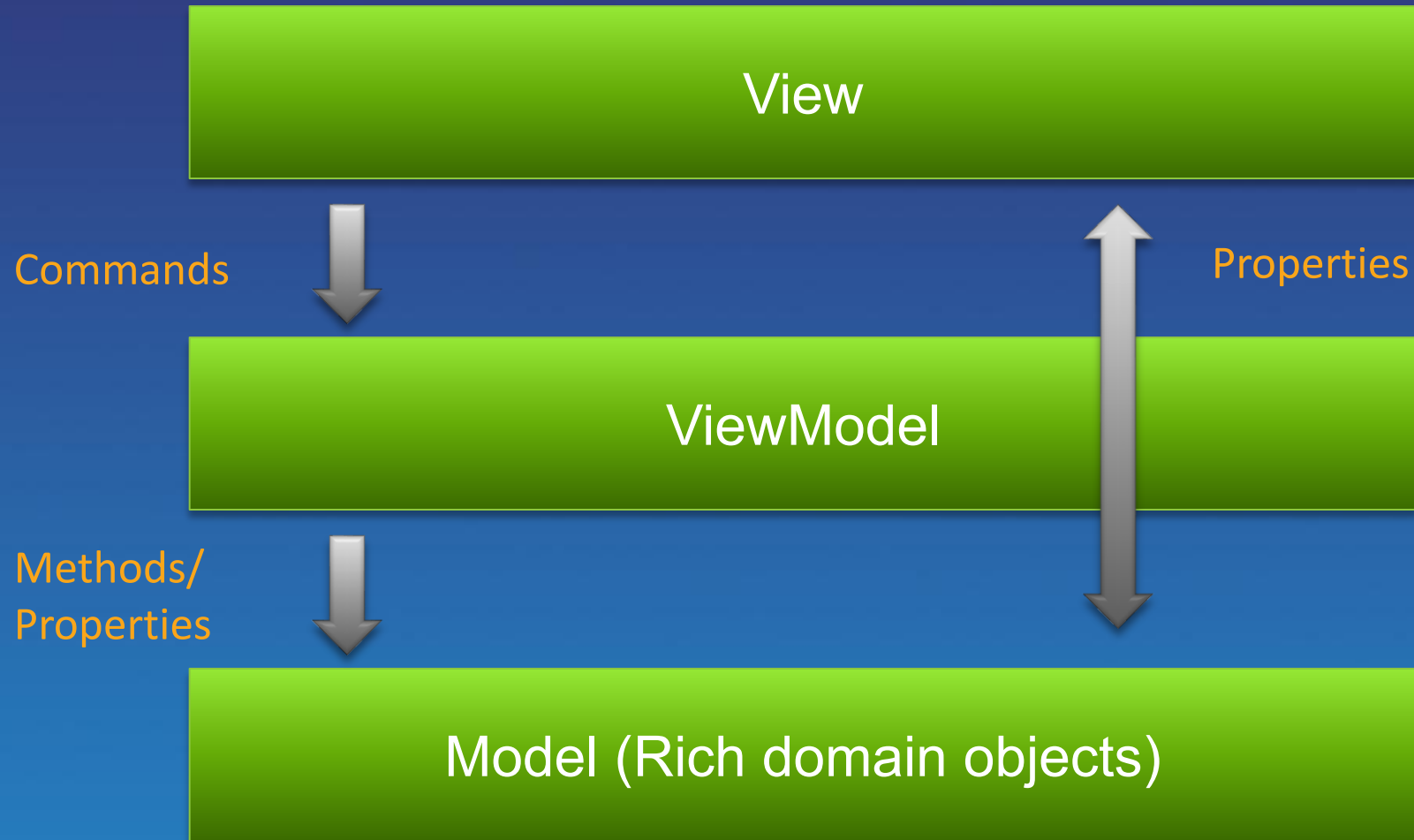
WUX407 - Best Practices: Building a Real-World Microsoft Silverlight Line-of-Business Application
Wed, 20 Oct 2010 (08:30 - 09:30) | Breakout Session | Session Room D4 | Level: 400 - Expert



MVVM



MVVM (rich model)



Why MVVM?

Testing of UI code

UI code structure/discipline

Separation of concerns

Design-Developer workflow



N-Layer Architecture



The View

Binding is important

- Data binding
- Element binding
- Commanding (or similar)

RAD designers are important

- Visual Studio 2010
- Blend

“No code behind”

- Not an MVVM goal
- Enable better designer <-> developer interaction



The ViewModel

Adds commands/verbs to the model

- Triggered/invoked by the view

Extends the model with extra properties

- Possibly even reshapes the model

Enhances the model with data binding support

- If the model doesn't already support data binding



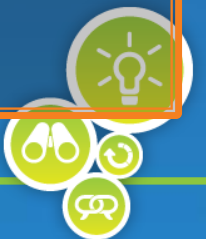
The Model

MVVM is primarily a UI pattern

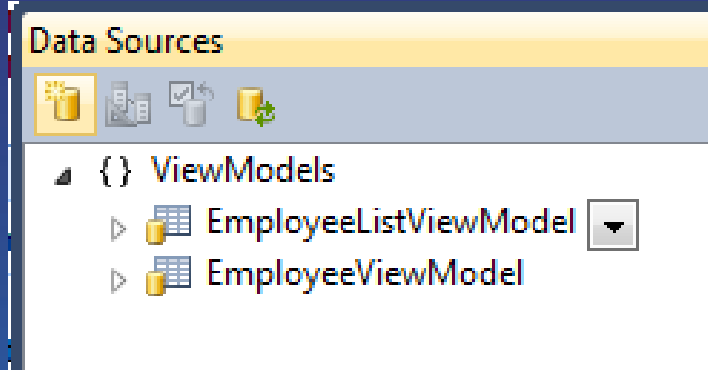
- Focus is on the View and ViewModel
- The Model is assumed to already exist

Types of Model

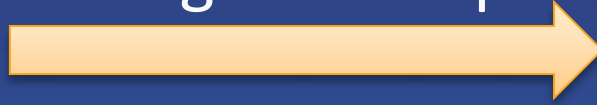
- Anemic
 - Objects don't support data binding
 - Objects encapsulate data, rarely behavior
- Rich
 - Objects support data binding
 - Objects encapsulate behavior as well as data



Connecting the Parts



Drag and Drop



View

ViewModel

Model



Resource Generated in View

```
<UserControl.Resources>  
  <CollectionViewSource x:Key="employeeViewModelViewSource"  
    d:DesignSource=  
      "{d:DesignInstance my:EmployeeViewModel, CreateList=True}" />  
</UserControl.Resources>
```



Code-behind????

```
private void UserControl_Loaded(object sender, RoutedEventArgs e)
{
    // Do not load your data at design time
    // if (!System.ComponentModel.DesignMode.DesignMode(this))
    // {
    //     //Load your data here and set the result to the ICollectionViewSource.
    //     System.Windows.Data.CollectionViewSource myCollectionViewSource =
    //         (System.Windows.Data.CollectionViewSource)this.Resources["Resource Key"];
    //     myCollectionViewSource.Source = yourData;
    // }
}
```



Basic XAML Framework (BXF)

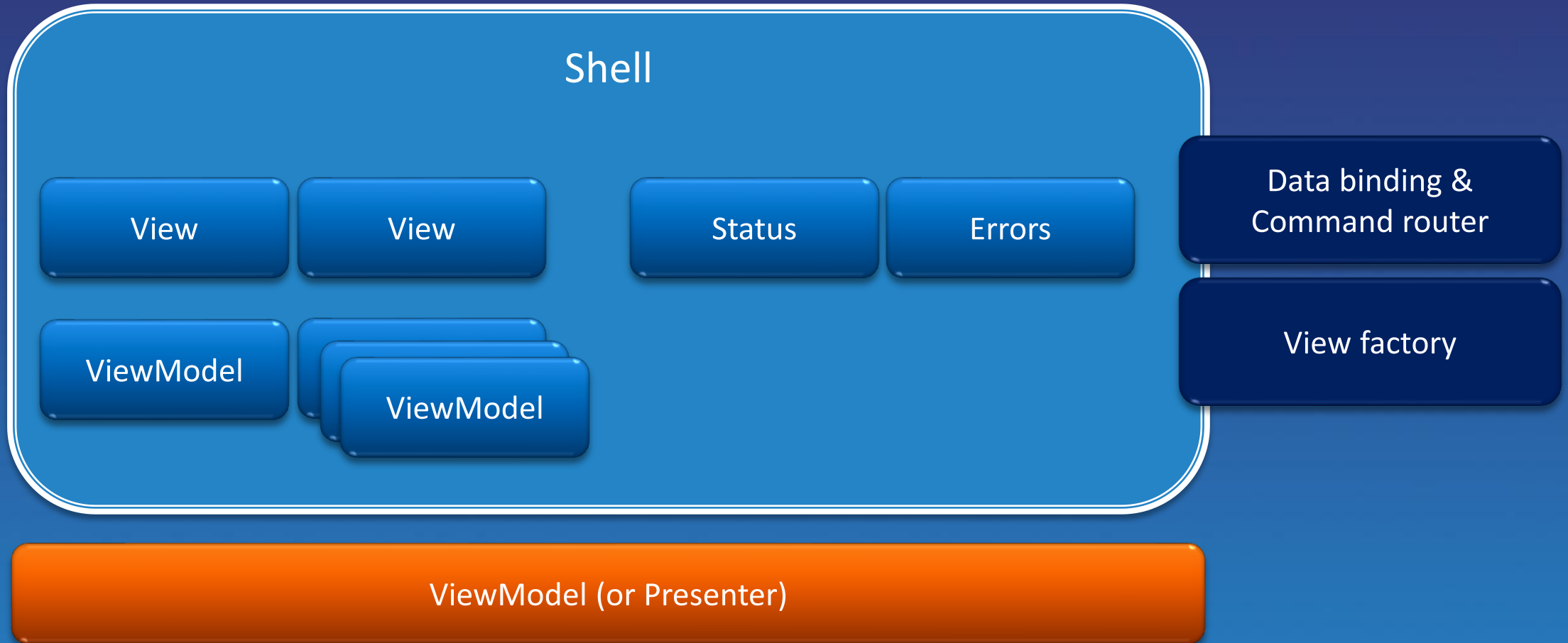
Basic Xaml Framework (**Bxf**) is a simple, streamlined set of UI components designed to demonstrate the **minimum** framework functionality required to make MVVM work well while leveraging the Visual Studio 2010 XAML designer ("**Cider**"). Bxf works with Silverlight, WPF and (soon) WP7.

<http://bxf.codeplex.com/>

Rockford Lhotka



Making MVVM Work





Building an App with MVVM

DEMO



Summary

- MVVM is a good pattern
- Avoid code-behind
- MVVM requires a little “plumbing”
 - UI shell abstraction
 - Route View events to ViewModel verbs
 - *View factory* concept



Questions?



Blog

<http://www.rudigrobler.net>



Twitter

@rudigrobler

And don't forget
to poken me



Resources

Microsoft®
tech·ed
Online

Sessions On-Demand & Community

www.microsoft.com/teched

Microsoft® | Learning

Microsoft Certification & Training Resources

www.microsoft.com/learning

Microsoft® TechNet

Resources for IT Professionals

<http://microsoft.com/technet>

msdn®

Resources for Developers

<http://microsoft.com/msdn>

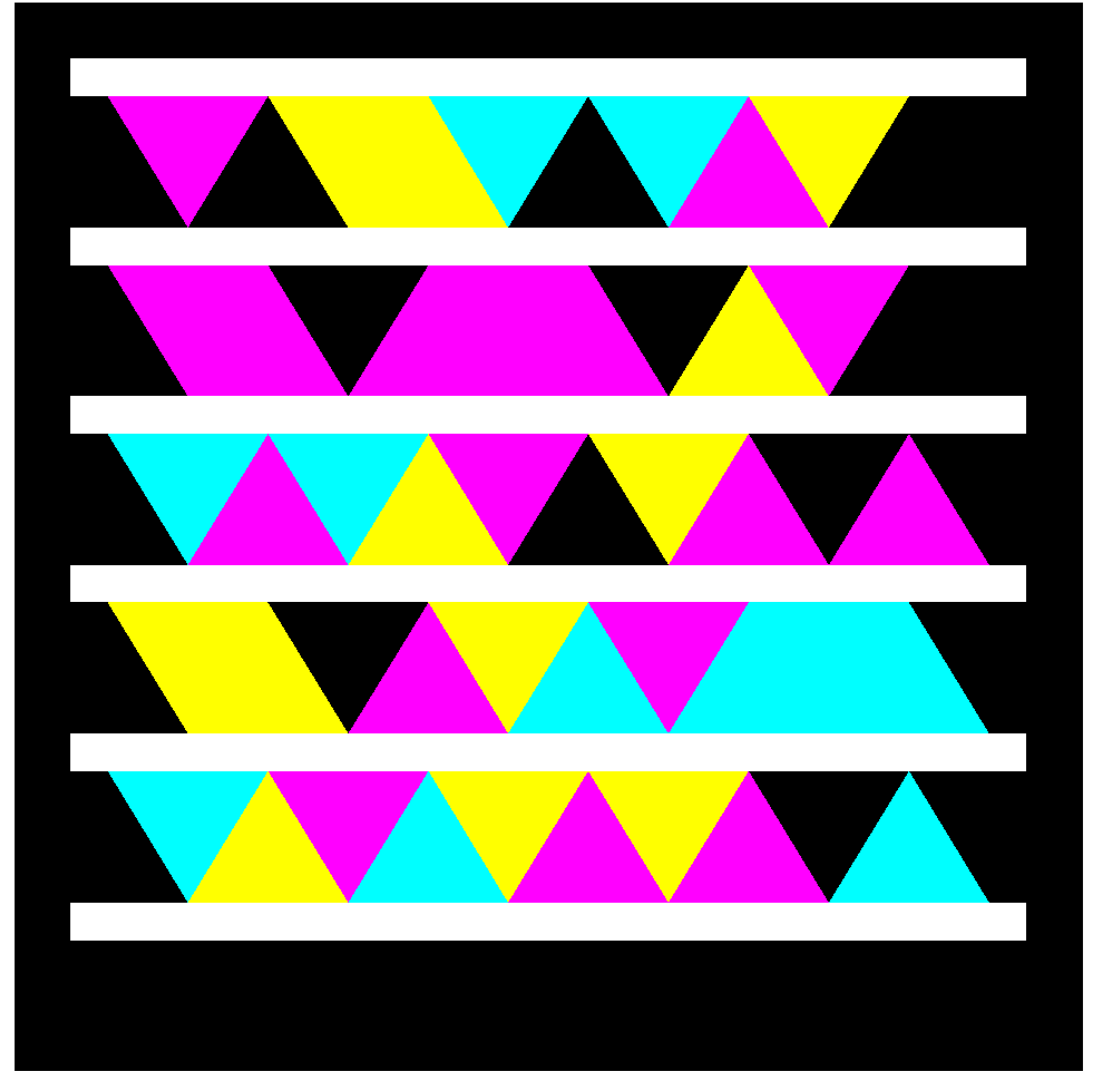
Need more Information?

SMS [Your Name] and the word "Dev" to 41491





Complete an evaluation
via CommNet and Tag
to win amazing prizes!



Microsoft®

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries. The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.