

PRISM

(Composite Application Guidance)

John McLoughlin

Batfish Solutions Ltd

NxtGen Southampton Founder

Agenda

- Who am I?
- Aim of this session
- What is a composite application?
- Regions
- Commands
- Events
- Services
- Summary
- Questions



Who am I?

John McLoughlin

- Freelance C# Developer/Technical Architect
- Over 10 years in the software industry
- Started off in Real-Time Embedded world
- Made the move to .Net in 2003
- Founded NxtGenUG – Southampton in Oct 2007

John@BatfishSolutions.com

blog.BatfishSolutions.com



Aim of this Session

To introduce you to the concepts in the
Composite Application Guidance (PRISM)



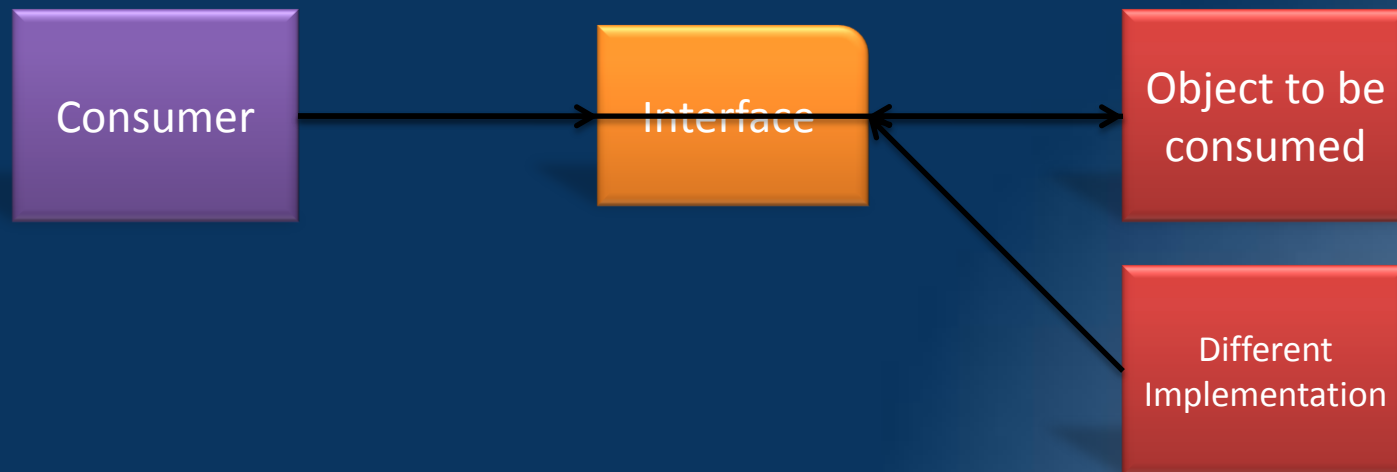
What is a composite application?

Modular
Loosely coupled
Extensible



What is Dependency Injection?

Also know as “Inversion of Control” (IoC)
or “Dependency Inversion Principle”
or Code to Interfaces



Types of Dependency Injection

```
/// <summary>
/// Dependency Injection via a Property
/// </summary>
public IPropertyInjection PropertyInjection
{
    set
    {
        // Implementation
    }
}
```

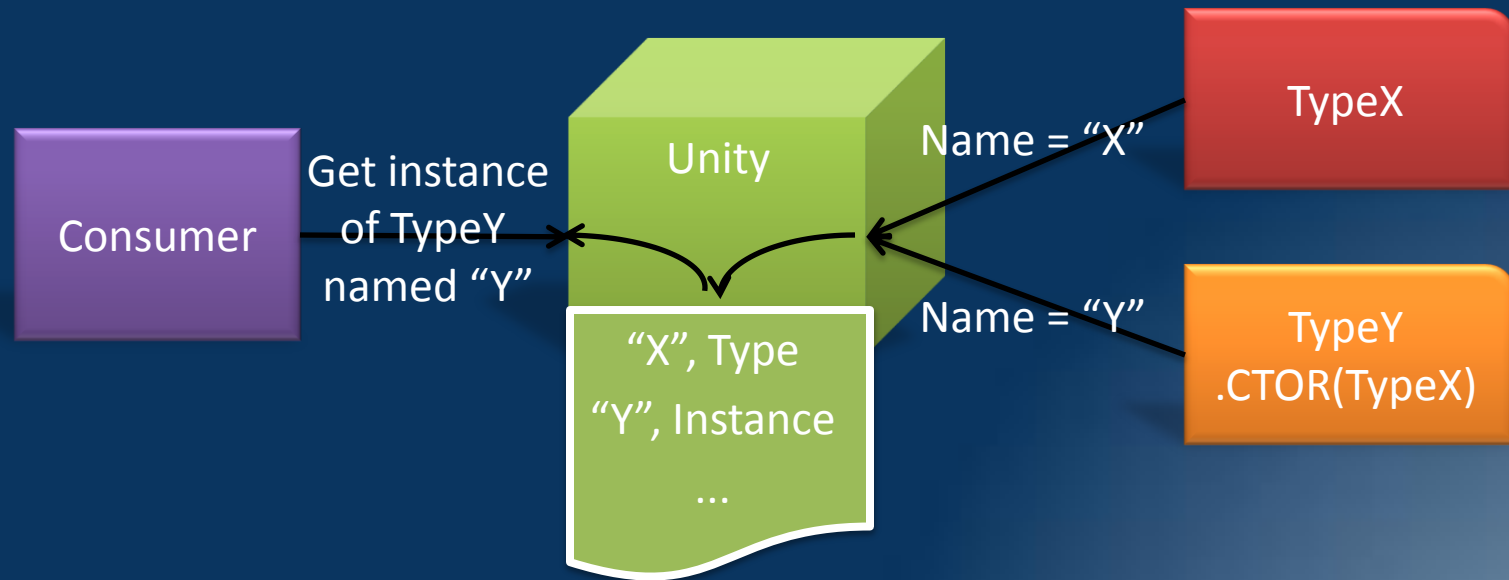
```
/// <param name="injection">The type being injected</param>
public void MethodInjection(IMethodInjection injection)
{
    // Implementation
}
```

```
/// <summary>
/// Dependency Injection via a Constructor
/// </summary>
/// <param name="injection">The type being injected</param>
public ExampleClass(ICtorInjection injection)
{
    // Implementation
}
```



The Dependency Injection Container

PRISM by default uses the Unity container



Dependency Injection with CAL

Composite App

The screenshot shows a web application interface for a stock trader. The main content area is titled 'CFI STOCKTRADER' and is divided into several sections:

- POSITION / WATCH LIST:** A table listing stock positions.
- Buy STOCK:** A form to execute a trade.
- STOCK HISTORICAL DATA:** A line chart showing price fluctuations.
- PIE CHART:** A pie chart representing the distribution of stock holdings.
- NEWS ARTICLES:** A list of recent news items.

Symbol	Shares	Last	Cost Basis	Market Value	Gain/Loss %	Actions
STOCK0	10	\$30.71	\$280.98	\$307.11	9.3%	[+/-] [x]
STOCK2	100	\$15.79	\$1,900.22	\$1,578.79	-16.9%	[+/-] [x]
STOCK3	100	\$13.93	\$1,900.22	\$1,392.88	-26.7%	[+/-] [x]
STOCK6	50	\$117.92	\$523.43	\$5,895.88	1026.4%	[+/-] [x]

Buy STOCK [Submit] [Cancel]

Buy Sell Shares: Price Limit:

Order Type: Term:

[Submit All] [Cancel All]

STOCK HISTORICAL DATA

3/1/2008 12:00:00 AM

PIE CHART

- STOCK0
- STOCK2
- STOCK3
- STOCK6
- STOCK7

NEWS ARTICLES

- 3/31/2008 12:00:00 AM Great Lakes Goes Healthy
- 12/6/2007 12:00:00 AM Ravioli Scare Prompts Government Action
- 5/24/2006 12:00:00 AM It's Pasta La Vista for Great Lakes
- 2/14/2005 12:00:00 AM Henderson: It's Gone Well Pasta Joke

Module A

View

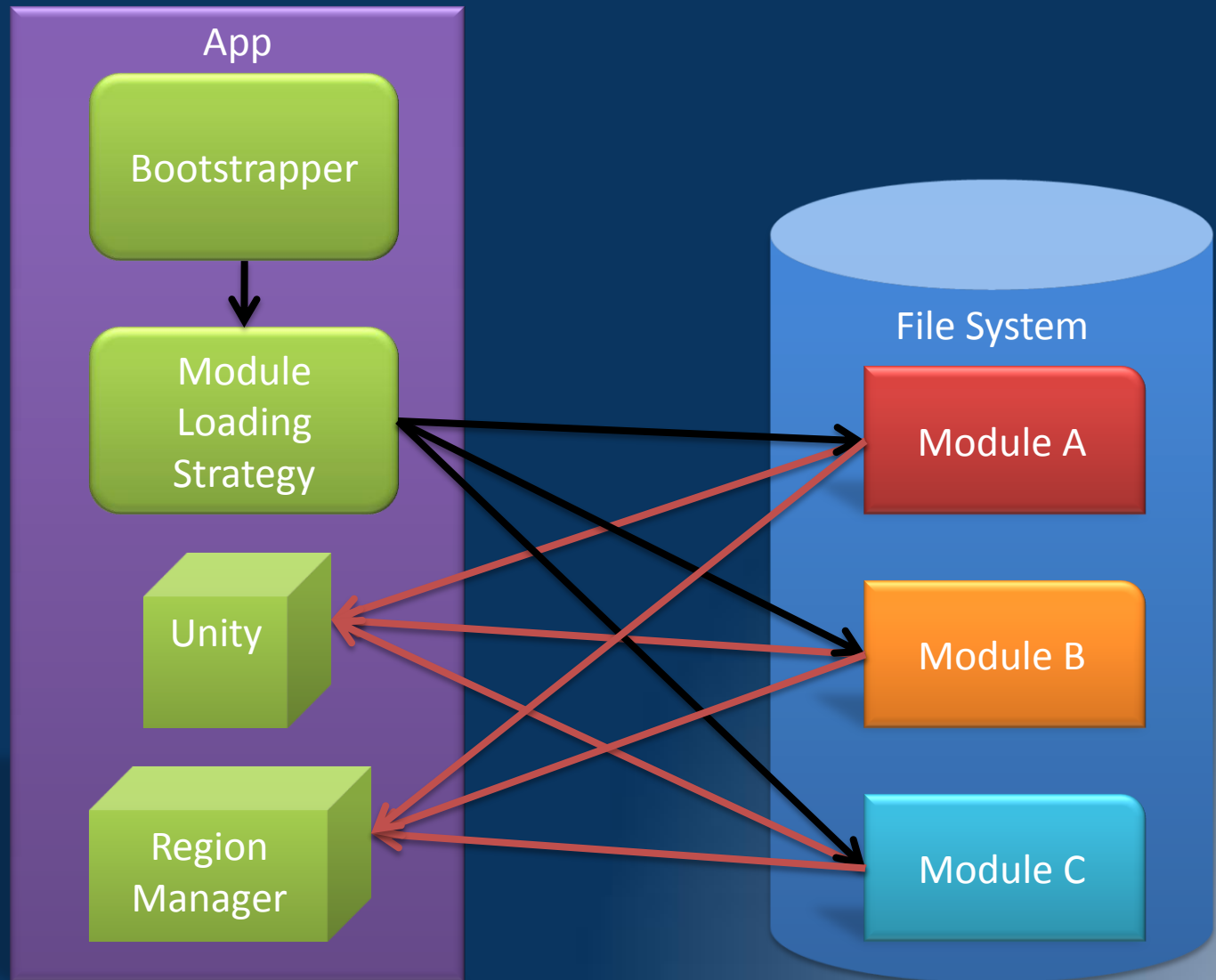
IService

Other

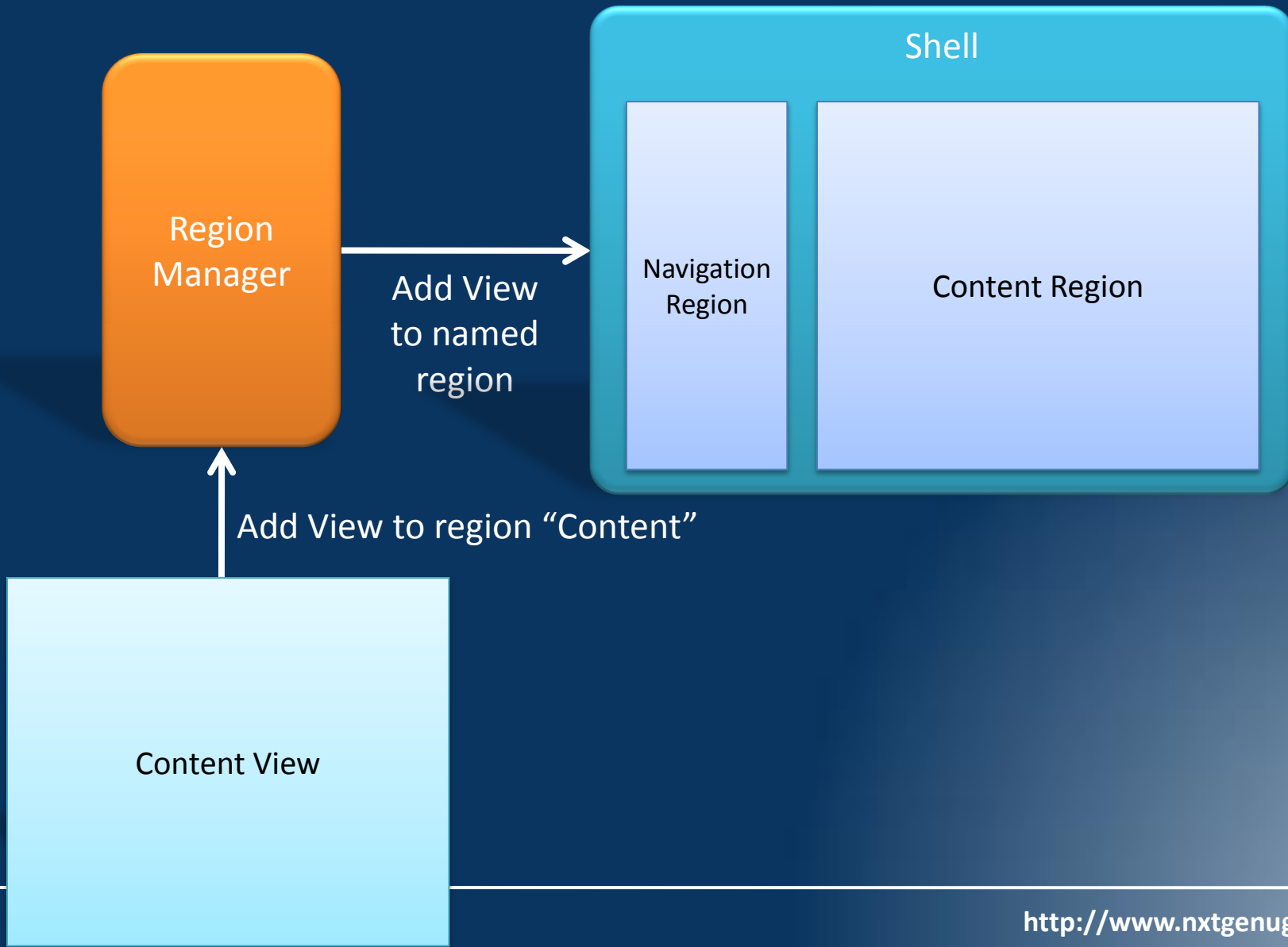
Module C



How the framework starts up



How Regions work

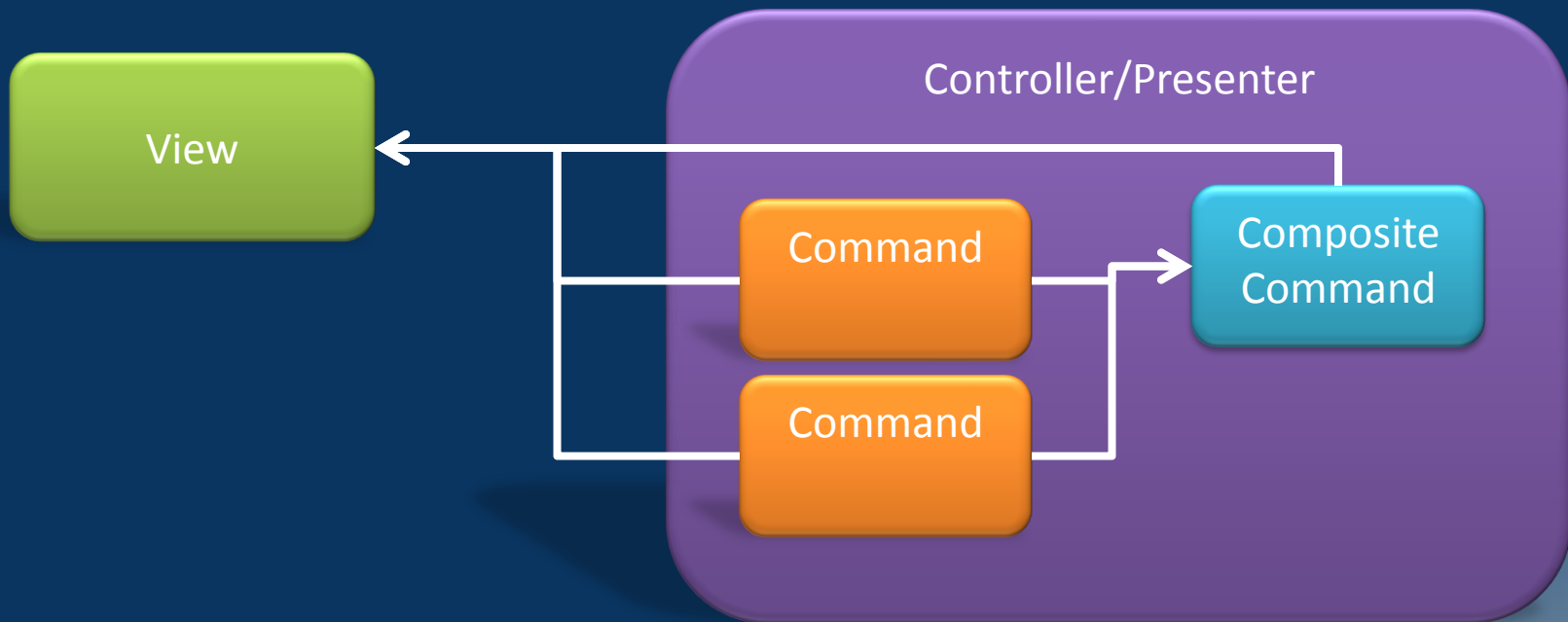


Converting an app and utilising Regions

DEMONSTRATION



How Commands work

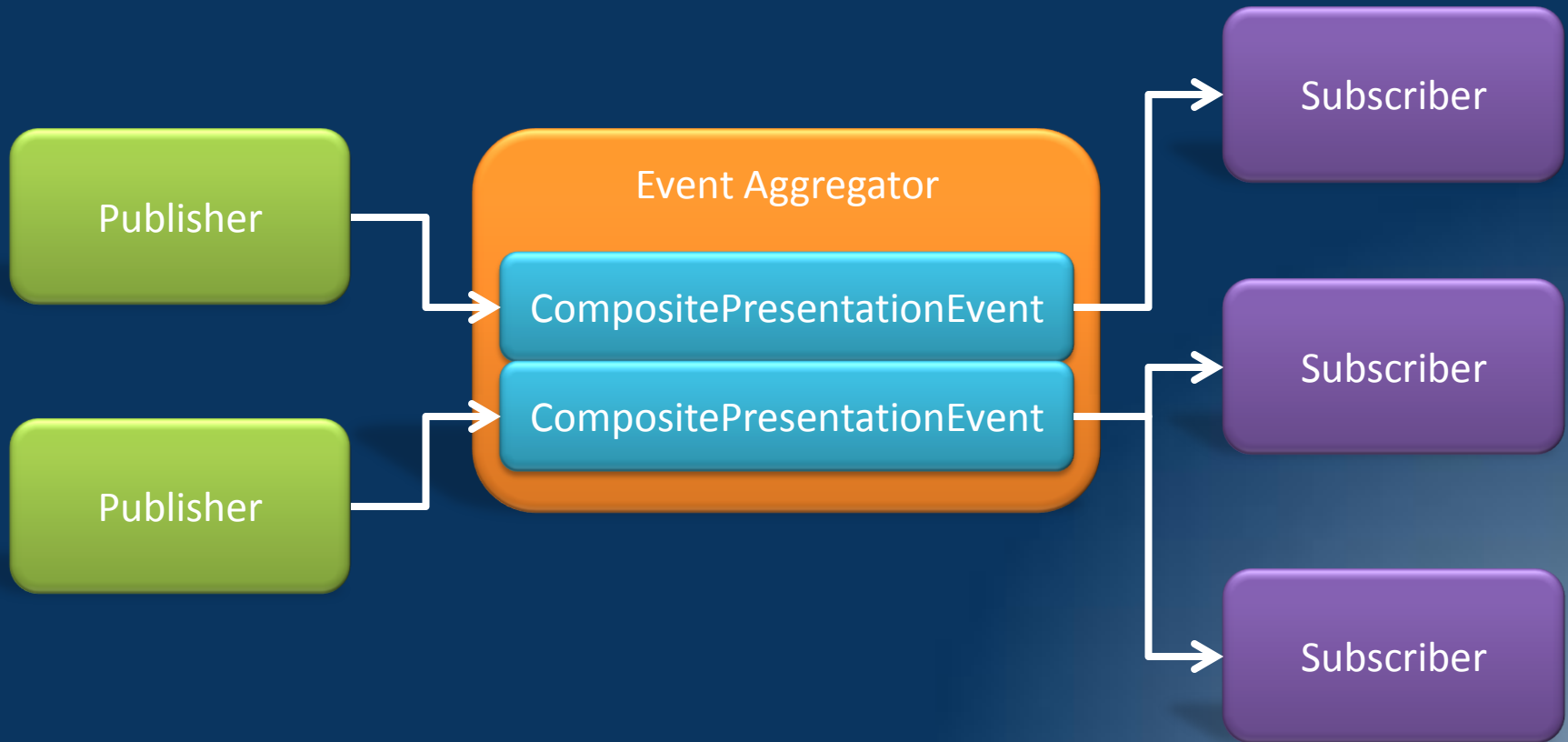


Utilising Commands

DEMONSTRATION



How Events work

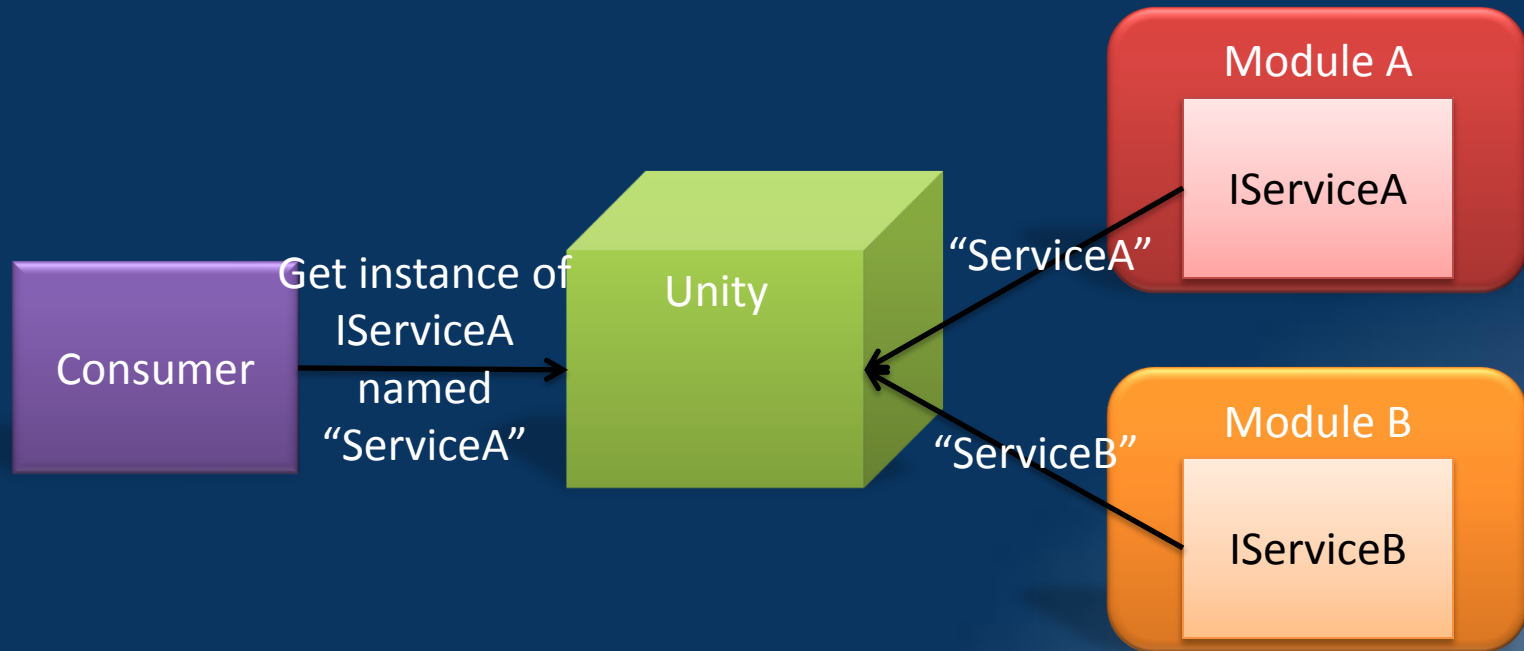


Utilising Events

DEMONSTRATION



How Services work



Utilising Services

DEMONSTRATION



What didn't I show you?

- Customising a Region Adapter
- Customising a Logger
- Using a DI Container other than Unity
 - Getting services via the ServiceLocator
- Multi targeting
- Mocking data when using Expression Blend



Summary

- Very simple to update your WPF and Silverlight app to use PRISM
- Encourages you to break you app up logically into modules
- You can use the same Views in both Silverlight and WPF apps (within reason)
- Based on proven Design Patterns



Questions?

Resources

www.codeplex.com/CompositeWPF

www.codeplex.com/Unity

www.codeplex.com/SmartClient

John@BatfishSolutions.com

blog.BatfishSolutions.com

