

BASTA!
.NET, WINDOWS, VISUAL STUDIO

Matthias Fischer

MF IT Consult | www.it-visions.de

Windows Phone 8 - Apps mit XAML und C# entwickeln

- Consultant und Trainer
 - .NET-Entwicklung seit 2001
 - ASP.NET, MVC4, SQL Server 2012
 - WPF, MVVM, Phone 8, Windows 8
- Autor
 - Carl Hanser Verlag (Auswahl)
 - Addison-Wesley, Wrox,
 - windows.developer
- Projekte (Auswahl)
 - netrixcomponent.net
 - augmentedbooks.com
 - textoor.de
 - APPS: OSMLogger, BatchUploader, BASTA!2013
- Expertennetzwerke (Auswahl)


www.IT-Visions.de
Dr. Holger Schwichtenberg


bestxperts

- Mehr Informationen: www.dotnetautor.de
- Kontakt : matthias@dotnetautor.de

NOKIA Developer
Certified Trainer



Premium Developer Program



- developer.nokia.com/windowsphone
- developer.nokia.com/entwicklergeraet
- entwickler@nokia.com

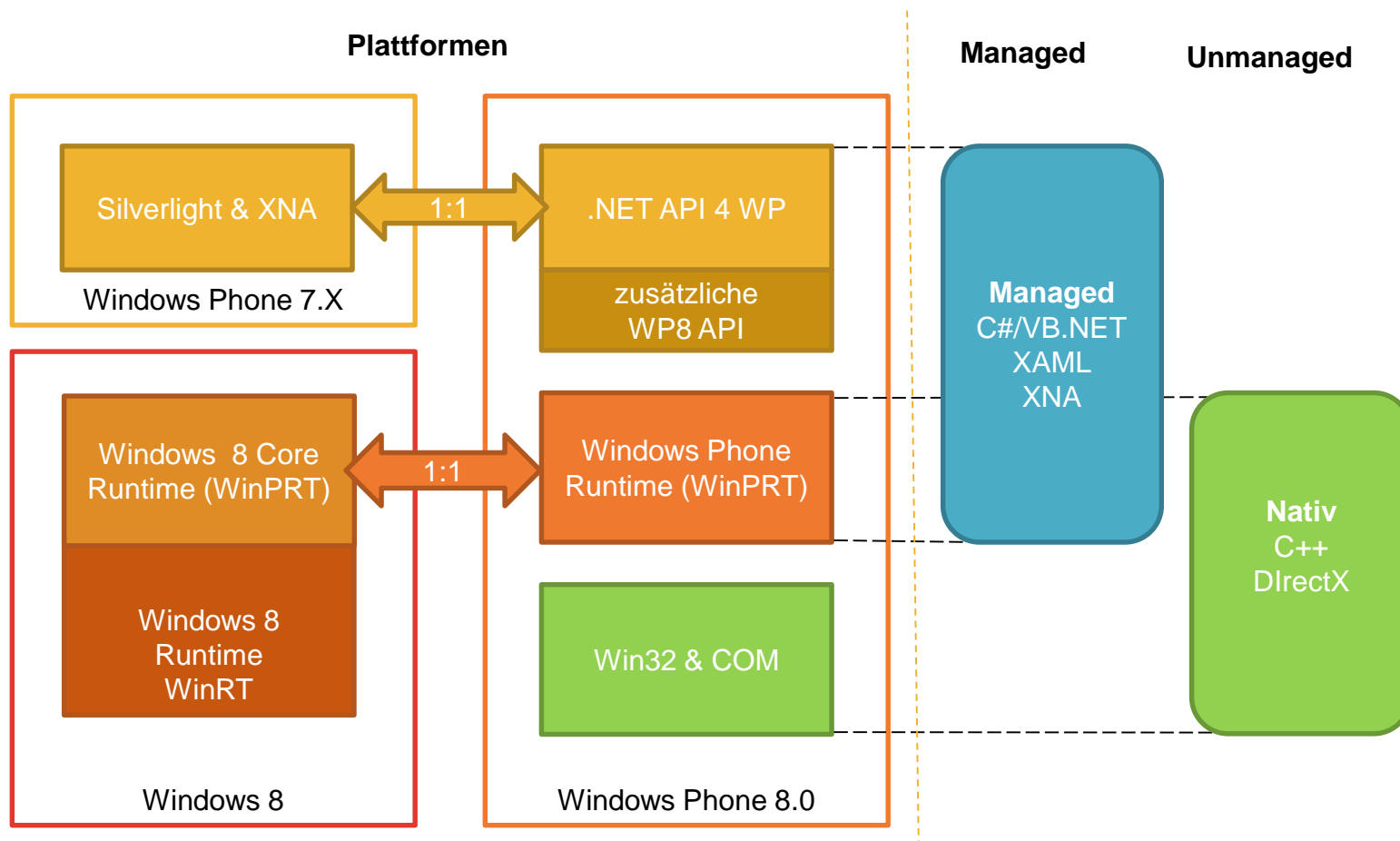
DIE WINDOWS 8 PLATTFORM

- Windows Phone und Shared Code
- Kurzer Überblick Twitter API
- Entwickeln eines WP8 Twitter Clients
- Portierung auf Windows 8
- Portierung auf Windows Phone 7.x
- **Hinweis:**
Do. 15:45 Uhr – 17:00 Uhr -- Raum: ULM
Überblick über die Windows 8 Entwicklung

Shared WinRT Core



Plattformen aus Entwicklersicht



- Ctrl-C, Ctrl-V
 - Einfach und schnell, jedoch (fast) unmöglich zu warten
 - Einzige Möglichkeit für XAML Design-Code-Share
- Linked Files
 - Visual Studio 2012 bietet die Möglichkeit Quellcode durch Drag'n'Drop + Alt über Projekte zu verlinken
- Linked Files mit #if Blöcken
 - Unter Umständen werden geringe Anpassungen benötigt, diese können mit Hilfe von #if ... #else ... #endif Blöcken angepasst werden
- Portable Class Library's (PCLs)
 - Eingeführt mit Visual Studio 2012, elegante Möglichkeit für mehrere Plattformen gleichzeitig zu entwickeln, ohne die rohen Quelldateien zu sharen
 - Nachteil: kleinste gemeinsame SubSet von Funktionen
- Plattform Adapter Abstraktion
 - Erzeugen einer abstrakten Klasse
 - Implementieren auf der jeweiligen Plattform

WP 7.X

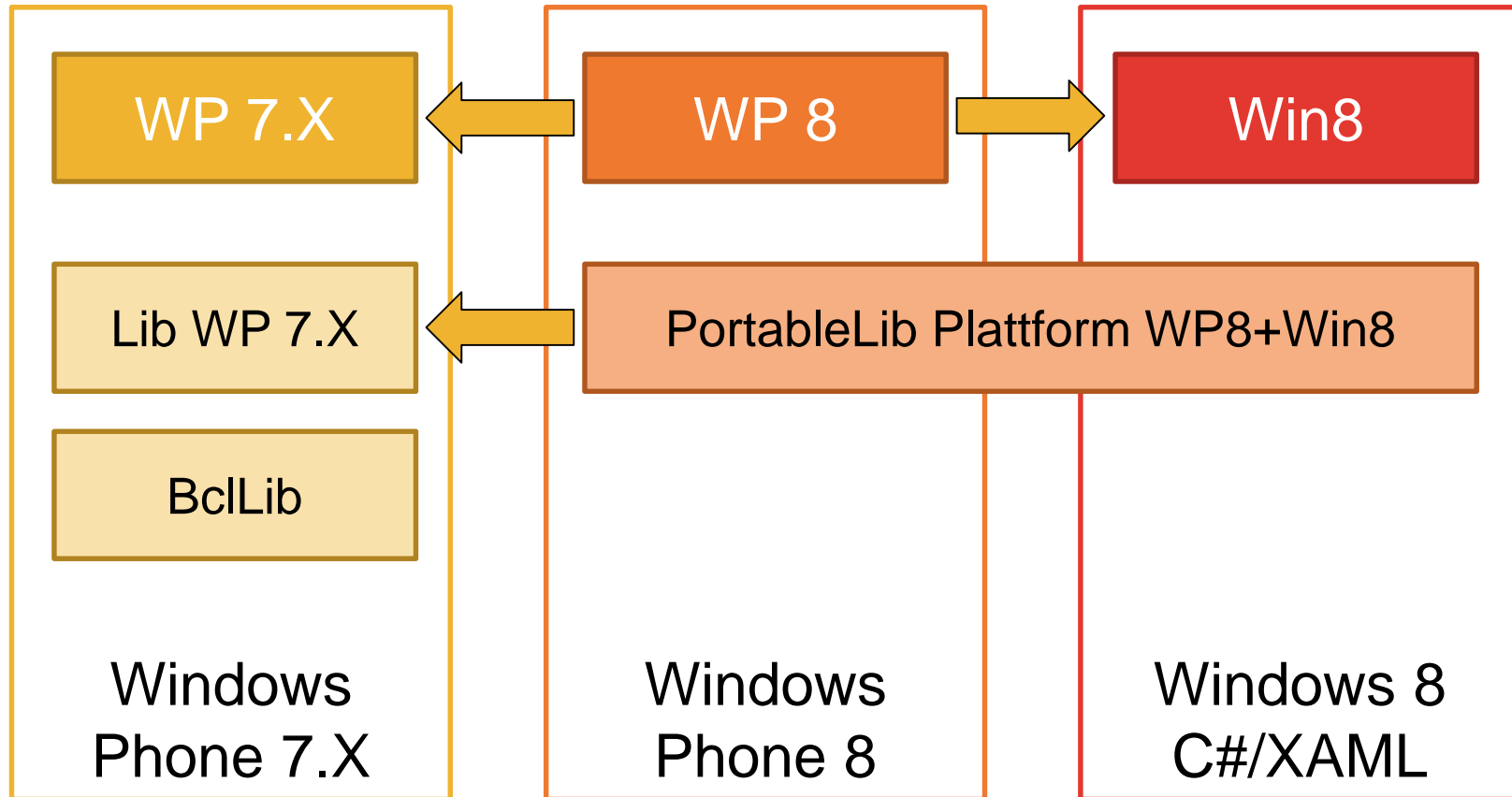
WP 8

Win8

- .NET for Windows Phone
- Async (Microsoft.Bcl.Async)

- WinRT bzw. WinPRT
- Async

- Gemeinsames SubSet von .NET 4 Phone und WinPRT
- KEIN Async verfügbar



SIMPLE TWITTER CLIENT

Twitter im Browser



The screenshot shows a browser window displaying the Twitter profile of BASTA! (@BASTAcon). The browser's address bar shows the URL <https://twitter.com/bastacon>. The page features a yellow background with a grid pattern. On the left, there is a sidebar with the BASTA! logo, event dates (25. Februar - 1. März 2013), location (Maritim Rhein-Main Hotel Darmstadt), and the website www.basta.net. Below this is a "Folge BASTA!" registration form with fields for a full name, a dotnetbird handle, and a password, followed by a "Registrieren" button. A "Globales Trends" section lists various hashtags and names. The main content area shows the profile header with the BASTA! logo, name, and handle, along with statistics: 1.163 tweets, 205 followers, and 373 following. Below the header is a "Tweets" section with several tweets from BASTA! and Rich Siegel. The right sidebar contains social media links for #bastacon, bastacon, basta, bastaconference, and bastacon.

BASTA!
.NET, WINDOWS, VISUAL STUDIO

25. Februar - 1. März 2013
Maritim Rhein-Main Hotel
Darmstadt
Expo: 26. - 28. Februar 2013

www.basta.net

BASTA!
@BASTAcon

BASTA! - die Konferenz für .NET, Windows, Visual Studio & More
Darmstadt · <http://www.basta.net>

1.163 TWEETS 205 FOLGT 373 FOLLOWER

Tweets

BASTA! @BASTAcon 15. Feb.
THIS is insanely funny > The Reality of Developer's Life ow.ly/hKleA
Öffnen

Rich Siegel @siegel 14. Feb.
OH: "If you can put an axe through it, it's hardware; if you *want* to put an axe through it, it's software."
Retweetet von BASTA!
Öffnen

BASTA! @BASTAcon 15. Feb.
Friday Fun: Findet den Fehler :-)
pic.twitter.com/KuGtZEEM
Foto anzeigen

BASTA! @BASTAcon 15. Feb.
#bastacon: Windows Azure ist mittlerweile die Überall-für-alle-Cloud - erleben Sie die Mächtigkeit der Plattform live!
ow.ly/hJXEM
Öffnen

BASTA! @BASTAcon 15. Feb.
Wie sieht das neue App-Modell für SharePoint aus!
@MarcAndreZhou auf der #bastacon!
ow.ly/hJO4j
Öffnen

BASTA! @BASTAcon 14. Feb.
Neues Windows Azure SDK vor Node.js via GitHub: bit.ly/vOZZKR
Öffnen

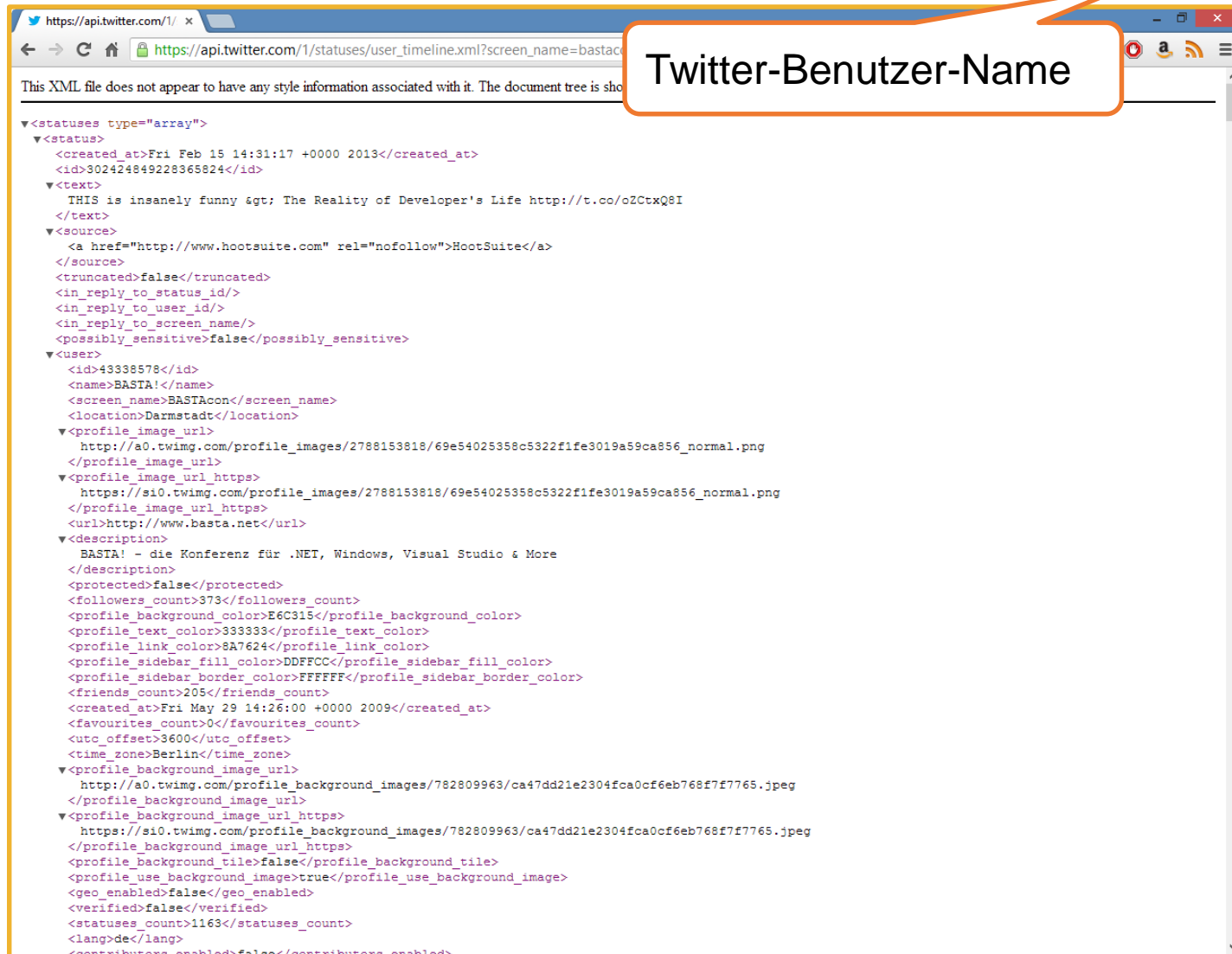
Globales Trends · Ändern

- #GritosClásicos
- #WaysToMakeMeMad
- #iWontReplyIf
- #10FatosSobreMim
- #QuéAscoSerUnMexicano
- Fred Bahnsen
- Good Luck Peche
- Cue Card
- Rusia
- Emma

#bastacon
bastacon
basta
bastaconference
bastacon

Twitter Status API

https://api.twitter.com/1/statuses/user_timeline.xml?screen_name=bastacon



```
<statuses type="array">
  <status>
    <created_at>Fri Feb 15 14:31:17 +0000 2013</created_at>
    <id>302424849228365824</id>
    <text>
      THIS is insanely funny &gt; The Reality of Developer's Life http://t.co/o2CtxQ8I
    </text>
    <source>
      <a href="http://www.hootsuite.com" rel="nofollow">HootSuite</a>
    </source>
    <truncated>false</truncated>
    <in_reply_to_status_id>
    <in_reply_to_user_id>
    <in_reply_to_screen_name>
    <possibly_sensitive>false</possibly_sensitive>
    <user>
      <id>43338578</id>
      <name>BASTA!</name>
      <screen_name>BASTAcon</screen_name>
      <location>Darmstadt</location>
      <profile_image_url>
        http://a0.twimg.com/profile_images/2788153818/69e54025358c5322f1fe3019a59ca856_normal.png
      </profile_image_url>
      <profile_image_url_https>
        https://si0.twimg.com/profile_images/2788153818/69e54025358c5322f1fe3019a59ca856_normal.png
      </profile_image_url_https>
      <url>http://www.basta.net</url>
      <description>
        BASTA! - die Konferenz für .NET, Windows, Visual Studio & More
      </description>
      <protected>false</protected>
      <followers_count>373</followers_count>
      <profile_background_color>E6C315</profile_background_color>
      <profile_text_color>333333</profile_text_color>
      <profile_link_color>8A7624</profile_link_color>
      <profile_sidebar_fill_color>DDFFCC</profile_sidebar_fill_color>
      <profile_sidebar_border_color>FFFFFF</profile_sidebar_border_color>
      <friends_count>205</friends_count>
      <created_at>Fri May 29 14:26:00 +0000 2009</created_at>
      <favourites_count>0</favourites_count>
      <utc_offset>3600</utc_offset>
      <time_zone>Berlin</time_zone>
      <profile_background_image_url>
        http://a0.twimg.com/profile_background_images/782809963/ca47dd21e2304fca0cf6eb768f7f7765.jpeg
      </profile_background_image_url>
      <profile_background_image_url_https>
        https://si0.twimg.com/profile_background_images/782809963/ca47dd21e2304fca0cf6eb768f7f7765.jpeg
      </profile_background_image_url_https>
      <profile_background_tile>false</profile_background_tile>
      <profile_use_background_image>true</profile_use_background_image>
      <geo_enabled>false</geo_enabled>
      <verified>false</verified>
      <statuses_count>1163</statuses_count>
      <lang>de</lang>
      <contributors_enabled>false</contributors_enabled>
```

Twitter-Benutzer-Name

- Verwendung der WebRequest-Klasse
- Zusammensetzen der URL
- Asynchrones herunterladen der Antwort

```
string result = "";
var request = WebRequest.CreateHttp(
    "http://api.twitter.com/1/s...ne.xml?screen_name=" + username);
request.Accept = "application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8";
//request.Headers[HttpRequestHeader.AcceptEncoding] = "gzip";

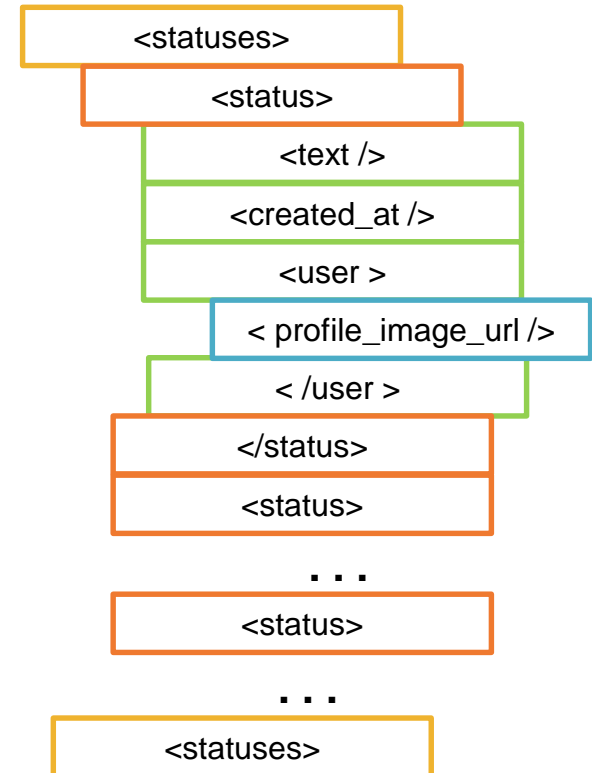
var response = await request.GetResponseAsync();

using (var stream = response.GetResponseStream()) {
    using (var reader = new StreamReader(stream)) {
        result = await reader.ReadToEndAsync();
    }
}
```

Informationen extrahieren

- Eine Klasse, um die Tweets zu speichern

```
public class TwitterPost
{
    public string PostText { get; set; }
    public string DatePosted { get; set; }
    public string UserImage { get; set; }
}
```

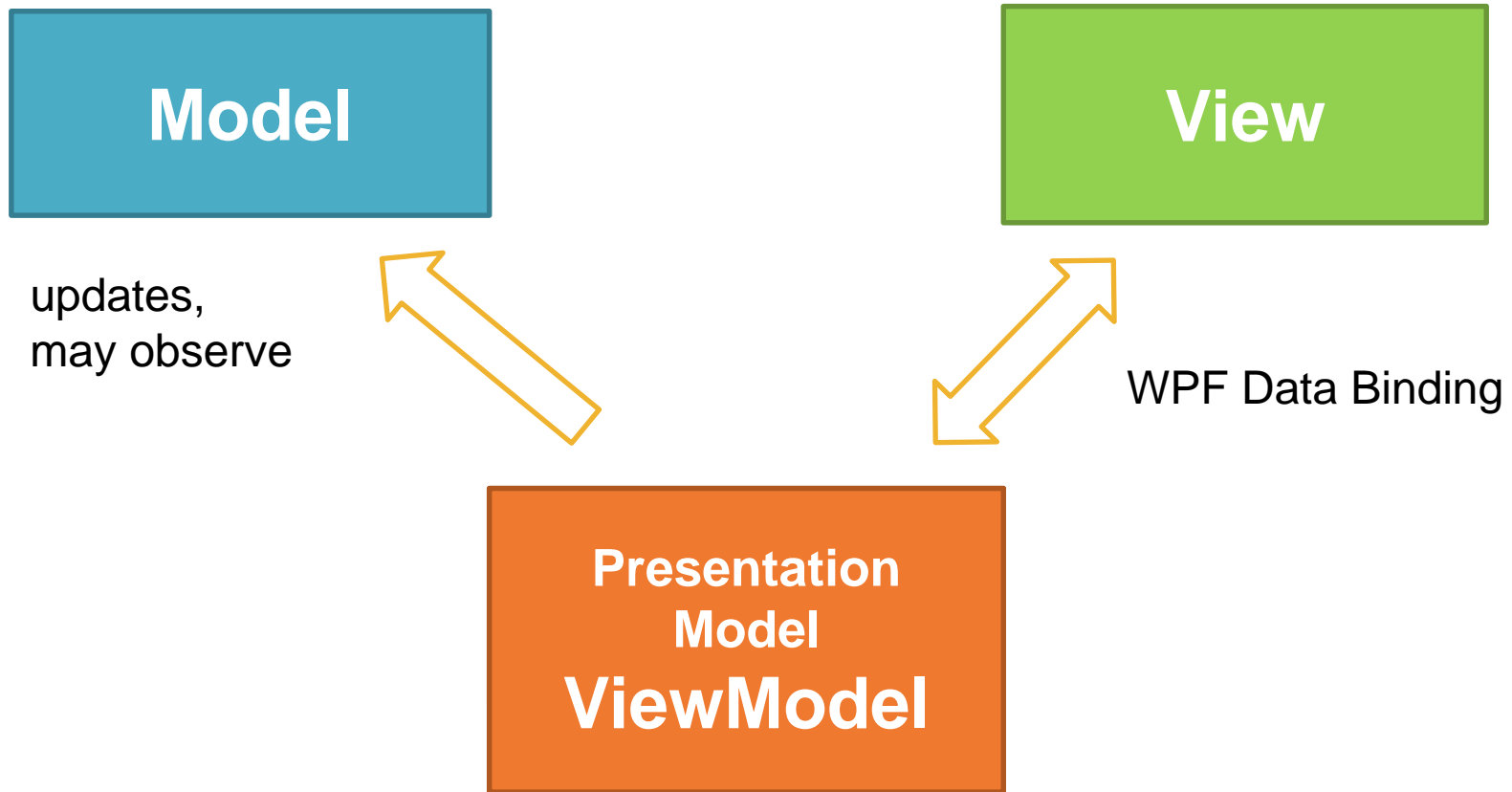


- Extrahieren der Tweets mit Hilfe von Linq2XML

```
XElement twitterElements = XElement.Parse(result);
var postList = from tweet in twitterElements.Descendants("status")
               select new TwitterPost {
                   UserImage = tweet.Element("user").Element("profile_image_url").Value,
                   PostText = tweet.Element("text").Value,
                   DatePosted = tweet.Element("created_at").Value
               };
```

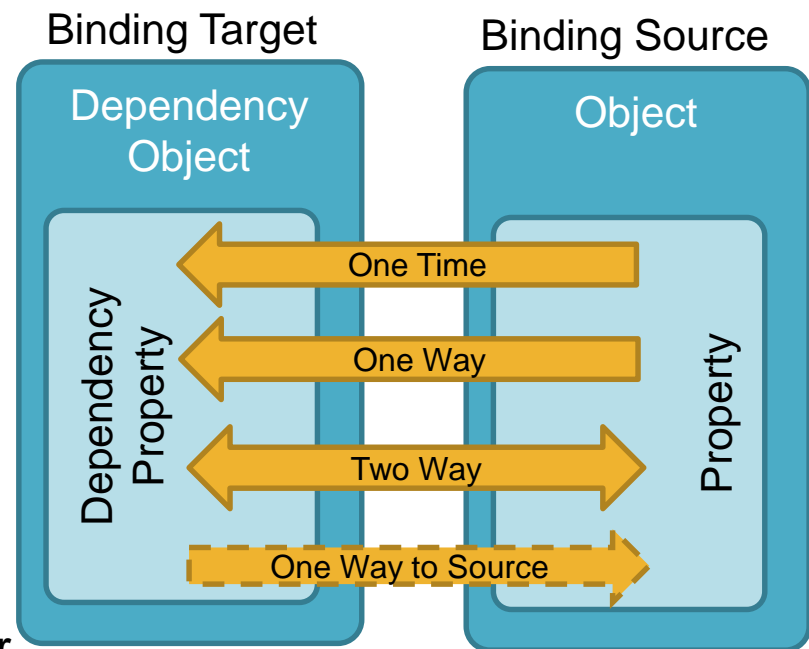
KURZÜBERBLICK MVVM

Presentation Model - MVVM

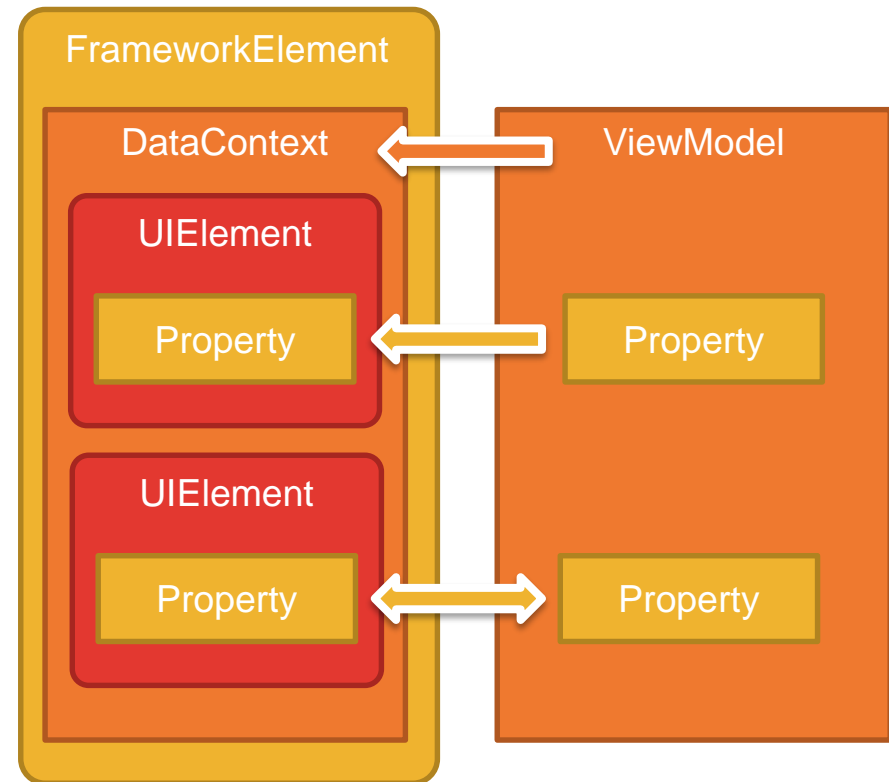


View.DataContext = ViewModel;

- Datenbindung (data binding) ist ein Mechanismus, um eine lose Kopplung zwischen der UI und der Anwendungslogik herzustellen
- Als Ziel für eine Datenbindung können nur **Dependency Objects** verwendet werden.
- Als Quelle kann jedes Objekt verwendet werden. Wenn Änderungen automatisch an das Bindungsziel übermittelt werden sollen, muss
 - entweder die Schnittstelle **INotifyPropertyChanged**
 - oder ein **Dependency Property** verwendet werden.
- Die Bindung kann einseitig, beidseitig oder einmalig erfolgen.
- Die Bindung kann in XAML oder im CodeBehind angelegt werden.



- Eine Daten-Klasse, welche als Quelle für die Datenbindung verwendet wird, wird **ViewModel** genannt.
- Die Datenquelle wird festgelegt, indem der Eigenschaft **DataContext** eines UI-Containers (FrameworkElement) eine Instanz des ViewModels zugewiesen wird.
- Eigenschaften von UI Elementen innerhalb des Containers, sowie Eigenschaften des Containers können nun an Eigenschaften des **ViewModels** gebunden werden.



- ViewModel-Klasse

```
public class MyViewModel {  
    public string TextProperty { get; set; }  
}
```

- Der Eigenschaft **DataContext** wird das ViewModel zugewiesen.

```
MainContainer.DataContext = new MyViewModel {TextProperty = "Hello Word"};
```

- Die Bindung einer Eigenschaft des **ViewModels** zu einer Eigenschaft des UI-Elements erfolgt in der XAML Datei

```
<Grid Name="MainContainer" >  
    <TextBlock Name="ContentText" Text="{Binding TextPropety, Mode=OneWay}"  
                TextWrapping="Wrap" Style="{StaticResource PhoneTextTitle3Style}" />  
</Grid>
```

- Komponenten für eine MVVM Anwendung
 - DelegateCommand
 - ViewModels (von ViewModelBase abgeleitet)
 - DataBinding und CommandBinding
- Zur Vereinfachung der Erstellung der ViewModel-Klassen
 - die Schnittstelle `INotifyPropertyChanged` in der Klasse `ViewModelBase` implementieren.

```
public class ViewModelBase : INotifyPropertyChanged {  
  
    public event PropertyChangedEventHandler PropertyChanged;  
  
    protected virtual void OnPropertyChanged(  
        [CallerMemberName] string propertyName = null) {  
  
        if (PropertyChanged != null)  
            handler(this, new PropertyChangedEventArgs(propertyName));  
    }  
}
```

- Die Implementierung für Kommandos mit Hilfe der Schnittstelle [ICommand](#) ist immer gleich. Um nicht jedes Mal eine eigene Klasse für jedes Kommando ausprogrammieren zu müssen, kann die folgende Klasse [DelegateCommand](#) als allgemeine Implementierung verwendet werden.

```
public class DelegateCommand : ICommand
{
    Func<object, bool> canExecute;
    Action<object> executeAction;

    public DelegateCommand(Action<object> executeAction)
        : this(executeAction, null)
    {
    }

    public DelegateCommand(Action<object> executeAction,
        Func<object, bool> canExecute)
    {
        if (executeAction == null)
        {
            throw new
                ArgumentNullException("executeAction");
        }
        this.executeAction = executeAction;
        this.canExecute = canExecute;
    }
}
```

```
public bool CanExecute(object parameter) {
    bool result = true;
    if (canExecute != null) {
        result = canExecute(parameter);
    }
    return result;
}

public event EventHandler CanExecuteChanged;
public void RaiseCanExecuteChanged()
{
    if (CanExecuteChanged != null) {
        CanExecuteChanged (this,
            new EventArgs());
    }
}

public void Execute(object parameter) {
    this.executeAction(parameter);
}
}
```

TwitterViewModel

```
public class TwitterViewModel : ViewModelBase {
    private string _userName;
    private ObservableCollection<TwitterPost> _tweets;

    public TwitterViewModel() {
        RefreshCommand = new DelegateCommand(async (o) => {
            Tweets = new ObservableCollection<TwitterPost>(
                await DataManager.Instance.GetTweetsByUsername(UserName));
        });
    }

    public string UserName {
        get { return _userName; }
        set {
            if(_userName == value) return;
            _userName = value;
            OnPropertyChanged();
        }
    }

    public ObservableCollection<TwitterPost> Tweets {
        get { return _tweets; }
        set {
            if (_tweets == value) return;
            _tweets = value;
            OnPropertyChanged();
        }
    }

    public DelegateCommand RefreshCommand { get; private set; }
}
```

Fragen?



- Beratung bei Einführung, Migration und Betrieb
- (Vor-Ort-)Schulungen, Workshops
- Coaching (Vor-Ort | Telefon | E-Mail | Online-Meeting)
- Support (Vor-Ort | Telefon | E-Mail | Online-Meeting)
- Entwicklung von Prototypen und Lösung



Matthias Fischer IT Consult

<http://www.dotnetautor.de.de>

Telefon +49 1520 1920 708

matthias@dotnetautor.de

