

# Neues in .NET

.NET 4.6, .NET Core 1.0 &  
Visual Studio 2015



Raphael Schwarz

codeworx gmbh

Web <http://www.codeworx.at>  
Mail [raphael@codeworx.at](mailto:raphael@codeworx.at)  
Twitter @CwRaphael

The logo for www.IT-Visions.at features the text "www.IT-Visions.at" in a bold, black, sans-serif font. A blue swoosh underline starts under the "w" and ends under the "t". A registered trademark symbol (®) is located to the upper right of the "s".

# Agenda

- ▶ The new Microsoft
- ▶ C# 6 – Roslyn
- ▶ Visual Studio 2015 – Update 1
- ▶ .NET 4.6, .NET Core 5.0, .NET Native

Things are changing

# New Thinking!



Linux

Demo

# C# 6.0

Compiler nicht CLR abhängig (Roslyn)

```
// default values
public class Customer
{
    public string First { get; set; } = "Jane";
    public string Last { get; set; } = "Doe";
}
```

```
// getter only
public class Customer
{
    public string First { get; } = "Jane";
    public string Last { get; } = "Doe";
}
```

```
// read only backing fields
public class Customer
{
    public string Name { get; }
    public Customer(string first, string last)
    {
        Name = first + " " + last;
    }
}
```

# Auto Properties

- Default values
- Getter only
- Read only

```
// method  
public void Print() => Console.WriteLine(First + " " + Last);
```

```
// property  
public string Name => First + " " + Last; public Customer
```

```
// indexer  
this[long id] => store.LookupCustomer(id);
```

## Expression bodies

- Methods
- Properties
- Indexers



# Using static

```
using static System.Console;
using static System.Math;
using static System.DayOfWeek;
class Program
{
    static void Main()
    {
        WriteLine(Sqrt(3*3 + 4*4));
        WriteLine(Friday - Monday);
    }
}
```

```
// properties
int? length = customers?.Length; // null if customers is null

// indexers
Customer first = customers?[0]; // null if customers is null

// null conditional - possible Null reference on .Count()
int? first = customers?[0].Orders.Count();
// inline
int? first = (customers != null) ? customers[0].Orders.Count() : null;
// better
int? first = customers?[0].Orders?.Count();

// void
PropertyChanged?.Invoke(this, args);
```

# Null-condition

The Elvis operator

# String interpolation

The new String.Format

```
// old
var s = String.Format("{0} is {1} year{{s}} old", p.Name, p.Age);
// new
var s = $"{p.Name} is {p.Age} year{{s}} old";

// format info
var s = $"{p.Name,20} is {p.Age:D3} year{{s}} old";

// expressions
var s = $"{p.Name} is {p.Age} year{(p.Age == 1 ? "" : "s")} old";
```

# nameof

```
if (x == null) throw new ArgumentNullException(nameof(x));
```

```
// prints "ZipCode"  
WriteLine(nameof(person.Address.ZipCode));
```

# Index initializer

|

```
var numbers = new Dictionary<int, string> {  
    [7] = "seven",  
    [9] = "nine",  
    [13] = "thirteen"  
};
```

# Exceptions

```
// Exception filter
try { ... }
catch (MyException e) when (myfilter(e))
{
    ...
}

// async - await
Resource res = null;
try
{
    res = await Resource.OpenAsync(...);
}
catch(ResourceException e)
{
    await Resource.LogAsync(res, e);}
finally
{
    if (res != null) await res.CloseAsync();
}
```

# Demo

# Visual Studio 2015



# Editions

Visual Studio Professional

Aufgewertet: Code Lens und Code Reviews

Visual Studio Enterprise

Visual Studio Community

Visual Studio Express (jetzt doch!)

Visual Studio Express 2015 for Web

Visual Studio Express 2015 for Windows

Visual Studio Express 2015 for Windows Desktop

Visual Studio Code

# Allgemeines

Weiterhin nur 32-Bit

- 64-Bit-Version nicht geplant

- Einige Teile von VS sind (wie bisher) 64-Bit

Weiterhin nur für Windows

Alte VS-Add-Ins nicht mehr Unterstützung, nur noch VS-Extensions (VSIX)

High-Resolution Icons für (wenn > 100% DPI Scaling)

Menüs nun im Standard wieder in Groß-Klein-Schrift

Window / Save Window Layout

Window / Apply Window Layout

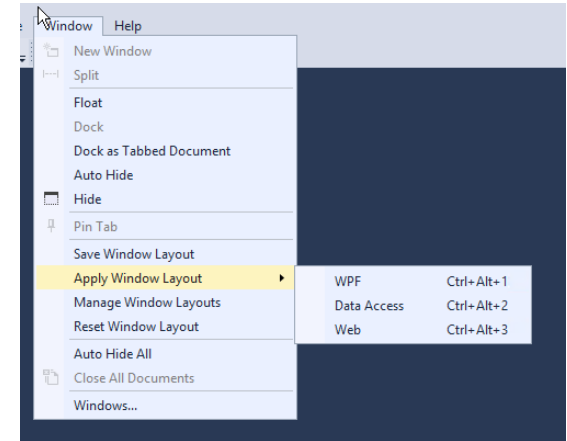
Window / Manage Window Layout: Sortieren,  
Löschen

Keyboard Shortcuts für die ersten neun:  
Ctrl+Alt+1 bis Ctrl+Alt+9.

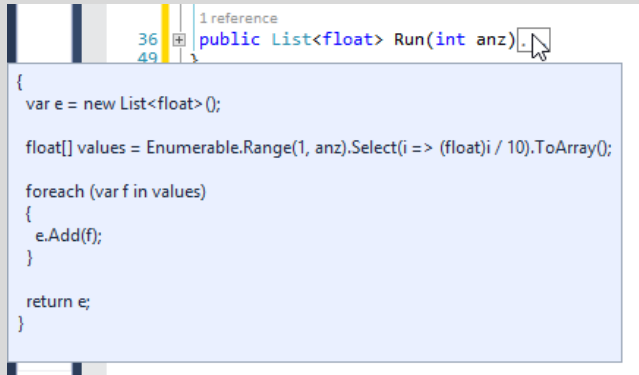
Roaming

Max 10 Layouts

# Window Layouts



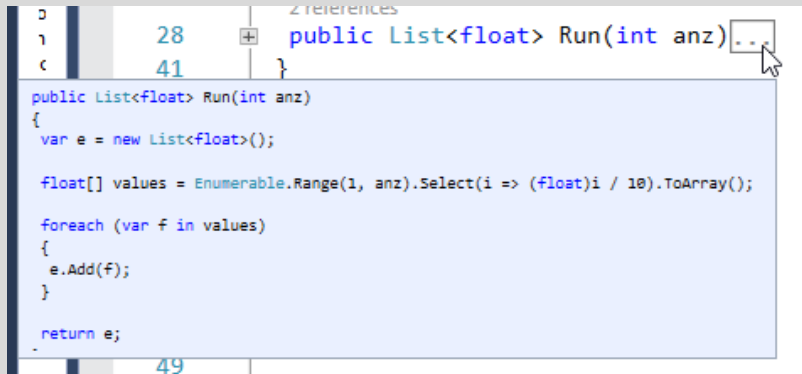
VS 2013:



A screenshot of Visual Studio 2013 showing a tooltip for a method call. The tooltip is light blue and contains the following code:

```
1 reference  
36 public List<float> Run(int anz)  
49 {  
    var e = new List<float>();  
  
    float[] values = Enumerable.Range(1, anz).Select(i => (float)i / 10).ToArray();  
  
    foreach (var f in values)  
    {  
        e.Add(f);  
    }  
  
    return e;  
}
```

VS2015



A screenshot of Visual Studio 2015 showing a tooltip for a method call. The tooltip is light blue and contains the following code:

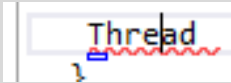
```
2 references  
28 public List<float> Run(int anz)  
41 {  
    public List<float> Run(int anz)  
    {  
        var e = new List<float>();  
  
        float[] values = Enumerable.Range(1, anz).Select(i => (float)i / 10).ToArray();  
  
        foreach (var f in values)  
        {  
            e.Add(f);  
        }  
  
        return e;  
    }  
}
```

# Tooltips

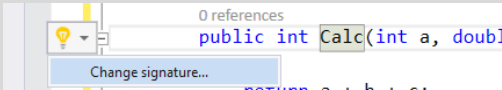
Syntax coloring

# Quick Actions

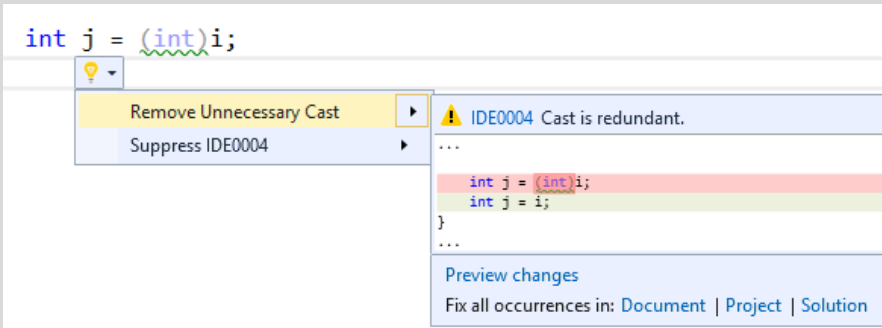
VS 2013:



VS 2015:



## Änderungsvorschau

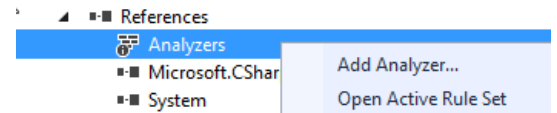


Strg+.

Code Refactoring

Code Fixes

Erweiterbar (VSIX, NuGet)



# Refactoring

Refactoring nun auch für VB und C++

Refactoring für Vb und C# basiert nun auf .NET  
Compiler Platform "Roslyn"

Neu:

- Inline Temporary Variable

- Introduce Constant

- Refactoring für Visual Basic .NET

"Rename"-Refactoring überarbeitet

# Analyzers

C# Essentials

[Extension VS Gallery](#)

Use NameOf

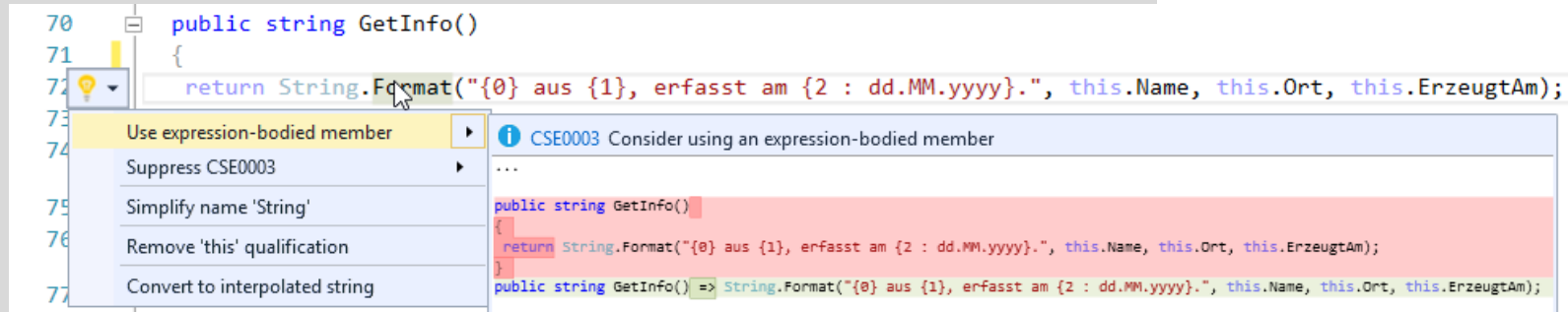
Convert to Interpolated String

Use Expression-Bodied Member

Use Getter-Only Auto-Property

Expand Expression-Bodied Member

...

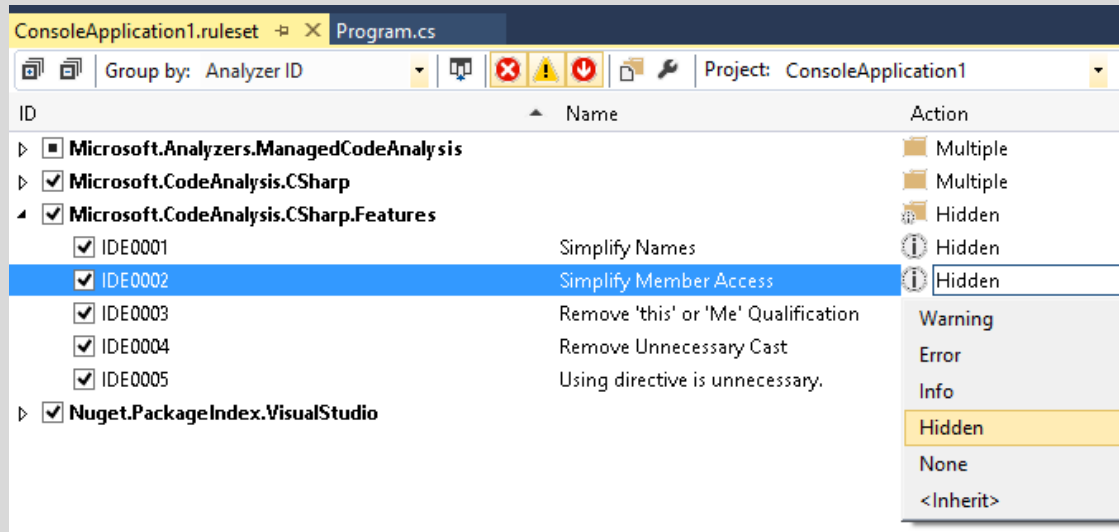


```
70 public string GetInfo()  
71 {  
72     return String.Format("{0} aus {1}, erfasst am {2 : dd.MM.yyyy}.", this.Name, this.Ort, this.ErzeugtAm);  
73 }  
74  
75  
76  
77
```

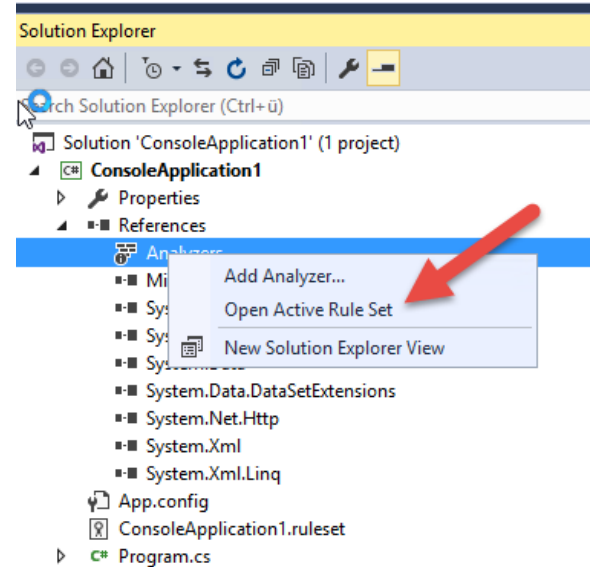
**CSE0003** Consider using an expression-bodied member

```
public string GetInfo()  
{  
    return String.Format("{0} aus {1}, erfasst am {2 : dd.MM.yyyy}.", this.Name, this.Ort, this.ErzeugtAm);  
}  
public string GetInfo() => String.Format("{0} aus {1}, erfasst am {2 : dd.MM.yyyy}.", this.Name, this.Ort, this.ErzeugtAm);
```

# Ruleset



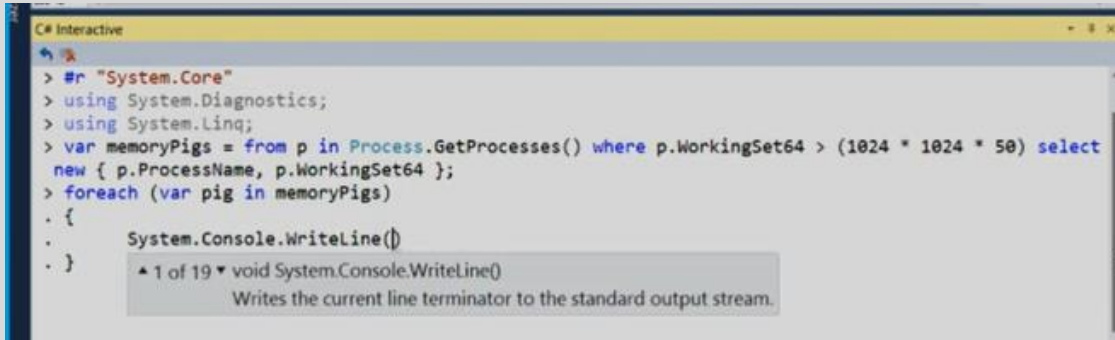
ID	Name	Action
▶ <input type="checkbox"/> Microsoft.Analyzers.ManagedCodeAnalysis		Multiple
▶ <input checked="" type="checkbox"/> Microsoft.CodeAnalysis.CSharp		Multiple
▶ <input checked="" type="checkbox"/> Microsoft.CodeAnalysis.CSharp.Features		Hidden
<input checked="" type="checkbox"/> IDE0001	Simplify Names	Hidden
<input checked="" type="checkbox"/> IDE0002	Simplify Member Access	Hidden
<input checked="" type="checkbox"/> IDE0003	Remove 'this' or 'Me' Qualification	Warning
<input checked="" type="checkbox"/> IDE0004	Remove Unnecessary Cast	Error
<input checked="" type="checkbox"/> IDE0005	Using directive is unnecessary.	Info
▶ <input checked="" type="checkbox"/> Nuget.PackageIndex.VisualStudio		Hidden





# C# Interactive

Scripting



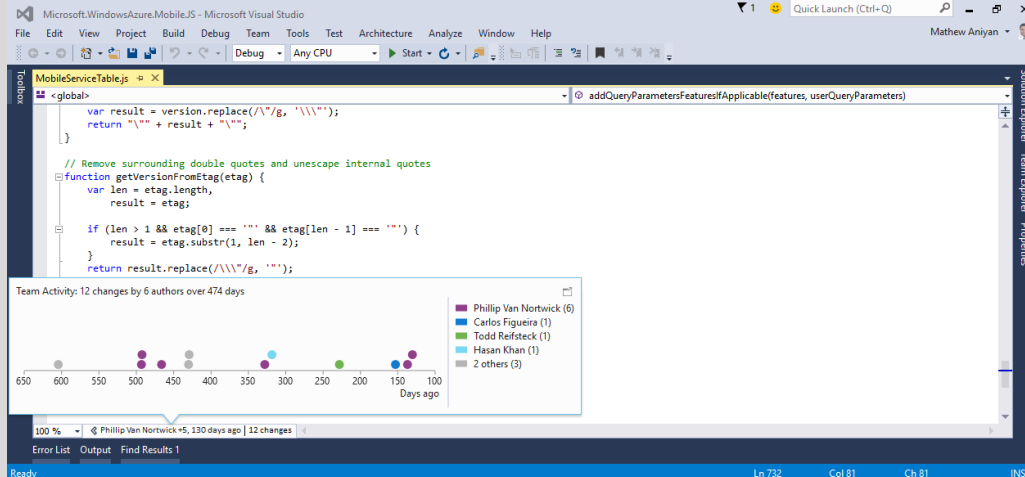
```
C# Interactive
> #r "System.Core"
> using System.Diagnostics;
> using System.Linq;
> var memoryPigs = from p in Process.GetProcesses() where p.WorkingSet64 > (1024 * 1024 * 50) select
new { p.ProcessName, p.WorkingSet64 };
> foreach (var pig in memoryPigs)
. {
.   System.Console.WriteLine()
. }
  ▲ 1 of 19 ▼ void System.Console.WriteLine()
    Writes the current line terminator to the standard output stream.
```

# CodeLens

Auch in Professional Edition

Change History (bisher nur für Git)

Codelens auch für C++, JS, SQL: nur auf Dateiebene und in Verbindung mit der Git-Versionsverwaltung.



## Visual Studio 2015-Editionen im Vergleich

	MSDN-Abon		
Alle erweitern	Visual Studio Community	Visual Studio Professional	Visual Studio Enterprise
Debuggen und Diagnose	■■■	■■■	■■■
Testtools	■■■	■■■	■■■
Integrierte Entwicklungsumgebung	■■■	■■■	■■■
CodeLens		■	■
Peek-Definition	■	■	■
Code-Klon			■
Refactoring	■	■	■
Web-Bereitstellung mit einem Klick	■	■	■

# JSON Editor

IntelliSense

Validation of JSON-Dateien mit JSON Schema

Duplicate property validation

Un-minify context menu

# Web-Tools

JSON-basierte Konfiguration

Node Package Manager (NPM)

Browser-Paketverwaltung mit Twitter Bower  
(<http://bower.io>)

Integration des Gulp Task Runner  
(<http://gulpjs.com>)

Andere wie z.B. Grunt [<http://gruntjs.com/>]  
möglich

# XAML Peek

Für

XAML (Ressourcen)

Event Handler Code

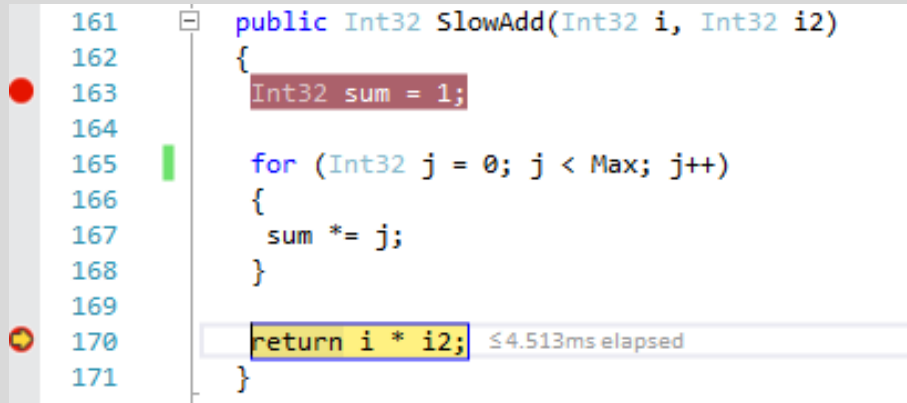
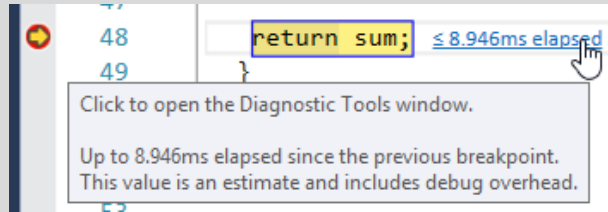
Klassen für XAML-Tags

Alt+F12

```
126 <GroupBox Grid.Row="2" Margin="5" Header="Buchung" Style="{StaticResource blau}...>
146 <GroupBox Grid.Row="3" Margin="5" Header="Meldung" Style="{StaticResource rot}">
17 <Style TargetType="GroupBox" x:Key="rot">
19 <Setter Property="Background" Value="#f2dede"></Setter>
20 <Setter Property="Foreground" Value="Red"></Setter>
21 </Style>
147 <TextBlock Margin="5" Text="{Binding Meldung}" TextWrapping="Wrap">
148 <TextBlock.Style>
```

```
10 <Button x:Name="button" Content="Hinzufügen" HorizontalAlignment="Left" Margin="33,31,0,0"
VerticalAlignment="Top" Width="75" Click="button_Click"/>
26 }
27
28 private void button_Click(object sender, RoutedEventArgs e)
29 {
30     var l = new Label();
31     l.Content = DateTime.Now.ToLongTimeString();
```

# Anzeige der Ausführungsdauer beim Debugging



## PerfTips

Einzelschritte: Pro Zeile

Zwischen zwei Breakpoints

# Diagnostic Tools

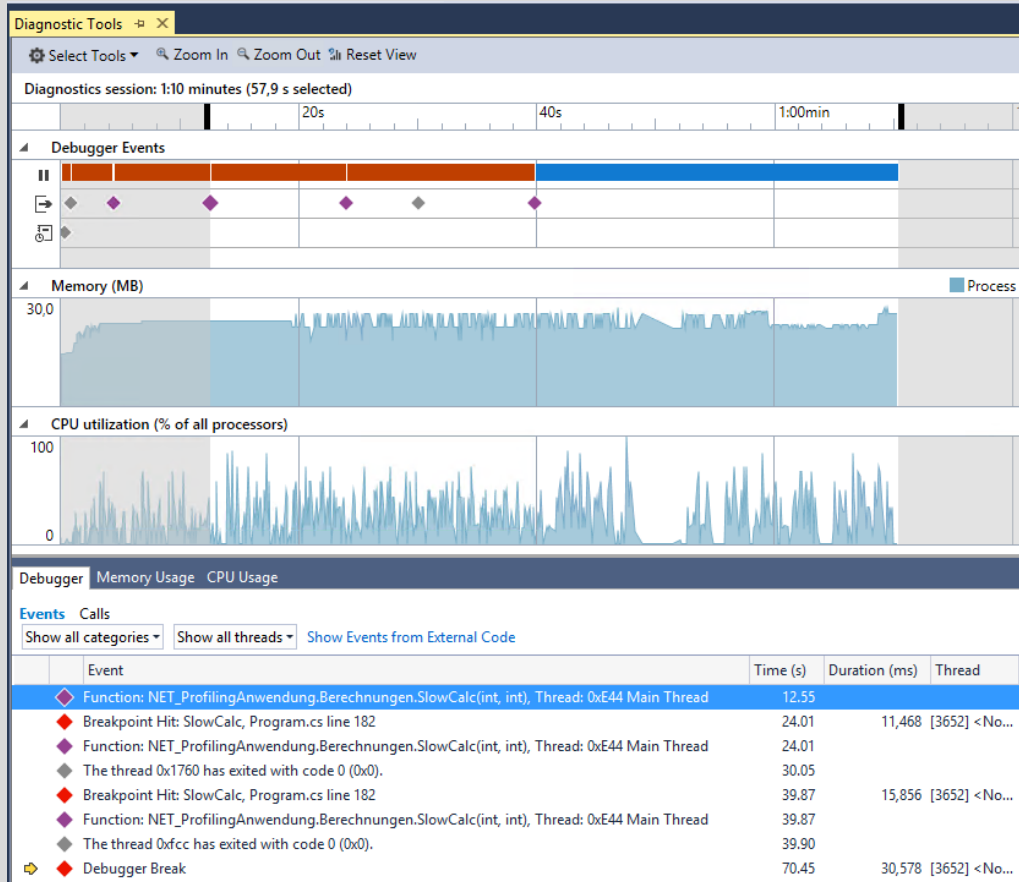
Diagnose-Informationen  
mit Timeline

IntelliTrace-Ereignisse  
während des Debuggens

Speichernutzung

CPU-Nutzung

Snapshots



# Network Diagnostic

Nur für Windows Store Apps

Nur HTTP (WinRT HttpClient API)

Report20150504-0924.diagsession - Microsoft Visual Studio

File Edit View Project Debug Team Tools Architecture Test Analyze Window Help

Report20150504-0924.diagsession

Stop

Name / Path	Protocol	Method	Result / Description	Content type	Received	Time	0ms
AppAdSettings-Money-Test?canvas=app&market=... http://cms-eastus-ppe-content-amp.cloudapp.net/non...	HTTP	GET	304 Not Modified	application/json	0 B	309 ms	
NewsPhonejs?adtype=mobile&stop=1&heroCoun... http://en-us.appex-rt.msn.com/cg/v7/EN-US/Finance/	HTTP	GET	200 OK	application/json	4.07 KB	71 ms	
readpdpdata https://personalization.services.msn.com/en-us/Finance/...	HTTPS	GET	200 OK	text/plain		383 ms	
NewsPhonejs?adtype=mobile&stop=1&heroCoun... http://en-us.appex-rt.msn.com/cg/v7/EN-US/Finance/	HTTP	GET	200 OK	application/json	4.06 KB	29 ms	
QC3CombinedDataV2?cntryMkt=EN-US&clientTyp... http://finance.services.appex.bing.com/marketv2.svc/	HTTP	GET	200 OK	application/json	5.1 KB	64 ms	
NewsPhonejs?adtype=mobile&stop=30&heroCoun... http://en-us.appex-rt.msn.com/cg/v7/EN-US/Finance/	HTTP	GET	200 OK	application/json	12.61 KB	87 ms	
M-LocalizedStringList?localizeFor=EN-US&Client-A... http://phone.finance.services.appex.bing.com/marketv2.svc/	HTTP	GET	200 OK	application/json	3.76 KB	96 ms	
WatchListV2?es=126.1TRUP.NYS%2C126.1IEMG A... http://finance.services.appex.bing.com/Market.svc/	HTTP	GET	200 OK	application/json	1.43 KB	259 ms	
TileImageChartUri?consumerType=0&index=30.101... http://finance.services.appex.bing.com/Market.svc/	HTTP	GET	200 OK	application/json	207 B	29 ms	
profilephoto https://cid-2f688fd91caac78ebusers.storage.live.com/use...	HTTPS	GET	304 Not Modified		0 B	787 ms	
AA7mX8P_m7_w2819_ifalse_uTrue_fjpg_q60.jpg http://img.stbs.msn.com/usappex/tenant/amp/entitv/d/	HTTP	GET	200 OK	image/jpeg	286.79 KB	81 ms	
AA39AID_m7_h76_w76_ifalse_fjpg_q60.jpg http://ima.stbs.msn.com/usappex/tenant/amo/entitv/d/	HTTP	GET	200 OK	image/jpeg	1.94 KB	6 ms	

0 errors 52 requests 968.07 KB transferred 3.8 s taken

Ready

Find (Ctrl+F)

Headers Body Parameters Cookies Timings

Request URL: http://cms-eastus-ppe-content-amp.cloudapp.net/non...

Request Method: GET

Status Code: 304 / Not Modified

Request Headers

Accept-encoding: gzip

Connection: Keep-Alive

Host: cms-eastus-ppe-content-amp.cloudapp.net

If-None-Match: "DZFuG5+5ksRbPHzco+TNQZWNi9E="

User-Agent: X-Client/AppexWin8Microsoft.BingFinance X-Client-App...

Response Headers

AppEx-Activity-Id: 1e8a1bde-0527-4b06-ba43-2e7001945653

Cache-Control: public, max-age=300

Content-Length: 0

Content-Type: application/json; charset=utf-8

Date: Mon, 04 May 2015 16:26:43 GMT

ETag: "DZFuG5+5ksRbPHzco+TNQZWNi9E="

Server: Microsoft-IIS/8.5

X-AspNet-Version: 4.0.30319

X-CMS-ExecutionTimeInMilliseconds: 1




# Lambda Expressions

Im Debugger

## Immediate Window

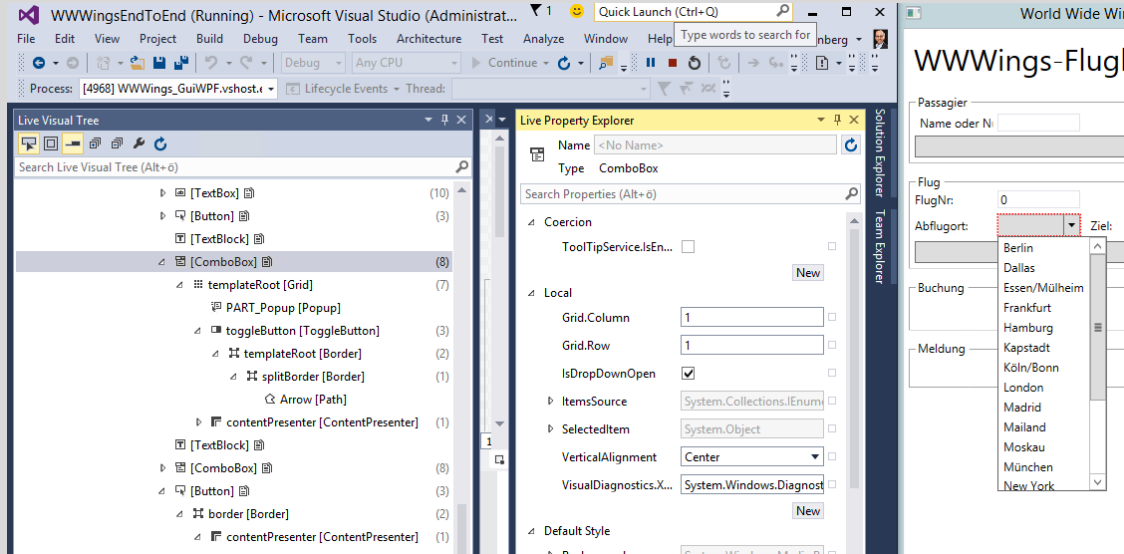
```
?values.Where(x => x % 2 == 0).ToList()  
Count = 5  
[0]: 2  
[1]: 4  
[2]: 6  
[3]: 8  
[4]: 10
```

## Watch 1

Name	Value	Type
values.Where(x => x % 2 == 0).ToList()	Count = 5 	System.Collections.Generic.List<float>
[0]	2	float
[1]	4	float
[2]	6	float
[3]	8	float
[4]	10	float
Raw View		

# Live Visual Tree

## Snoop Alternative

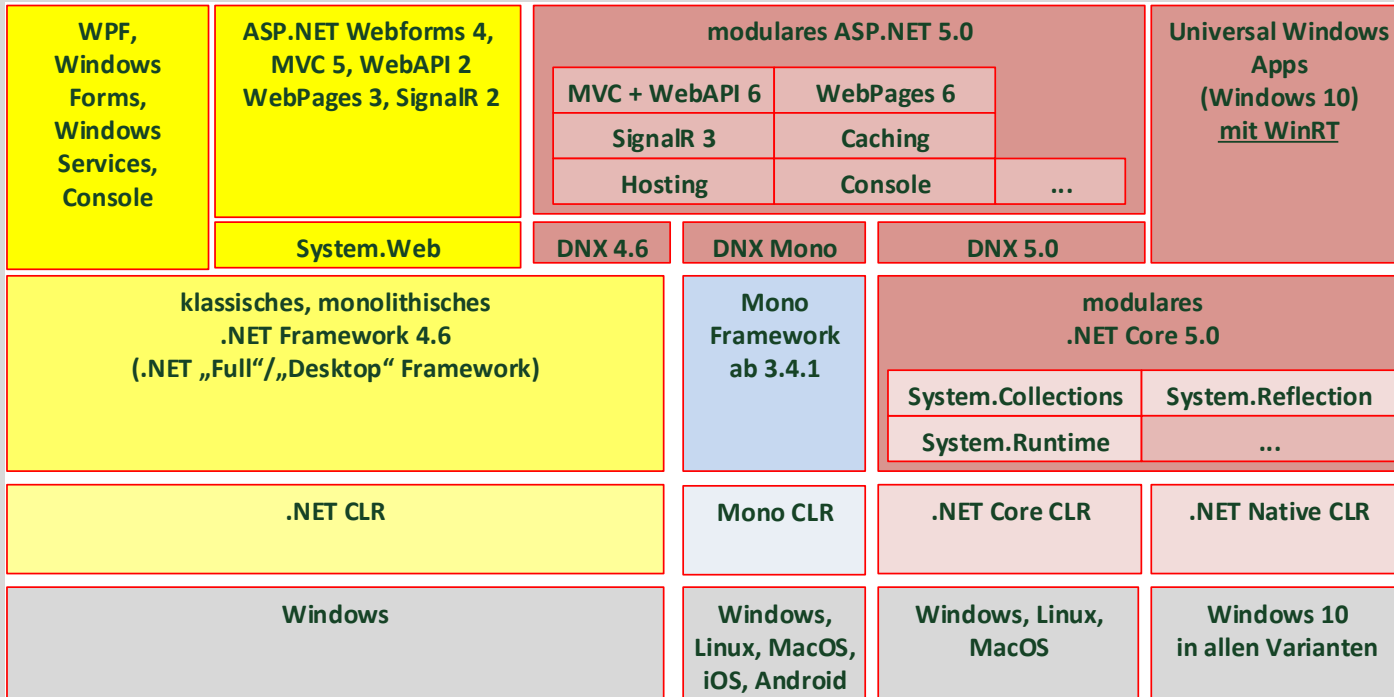


# Cross Platform

# Cross Platform in VS 2015

- ▶ .NET Core: Deployment und Remote Debugging für Linux/Mac
- ▶ C++ für Android und iOS
- ▶ Open Graphics Library for Embedded System (OpenGL ES)
- ▶ Visual Studio Tools for Apache Cordova  
(früher: Multi Device Hybrid Apps – "MDA")  
Android 2.3.3 and later (4.4 provides the best developer experience)  
iOS 6, 7, and 8  
Windows 8 and 8.1  
Windows Phone 8 and 8.1  
Android Emulator  
Remote Debugging für iOS
- ▶ Xamarin Starter Edition enthalten

.NET 4.6, .NET Core



# .NET 2015

# Open Source

.NET wird Stück für Stück „Open Source“ mit MIT-Lizenz

.NET Foundation als offizieller Inhaber

.NET "Full" Framework 4.x

Mono u.a. dürfen sich dort "bedienen" → macht Mono bereits  
Aber ohne Benutzerbeteiligung (wäre hier kaum möglich)  
<https://github.com/Microsoft/referencesource>

.NET Core Framework 5.x „Develop in the Open“

Microsoft arbeitet auf Github  
Issues / Work Items einsehbar  
Design Meeting Notes  
Pull-Requests gestattet  
Microsoft übernimmt QS + Support  
ca. 10% von .NET Core von Externen

Große Chance für Mono/Xamarin

# Wo läuft .NET Framework 4.6?

Windows 10: vorinstalliert

Add-On: Windows Vista Service Pack 2, Windows 7 Service Pack 1, Windows 8, Windows 8.1, Windows Server 2008 R2 SP1; Windows Server 2008 Service Pack 2, Windows Server 2012 und Windows Server 2012 R2.

In Place update: Überschreitet 4.0, 4.5, 4.5.1, 4.5.2

Microsoft .NET Framework 4.6 ist eine hochkompatible Direktaktualisierung von Microsoft .NET Framework 4, Microsoft .NET Framework 4.5, Microsoft .NET Framework 4.5.1 und Microsoft .NET Framework 4.5.2. Das Offlinepaket kann in Situationen verwendet werden, in denen die Webinstallation aufgrund fehlender Internetverbindung nicht möglich ist.

**Zitat Microsoft**



Microsoft .NET Framework 4.0	31.03.2010	Review-Hinweis	12.01.2016	Der Support für .NET Framework 4, 4.5 und 4.5.1 endet am 12. Januar 2016. Microsoft empfiehlt Kunden ein Upgrade auf .NET Framework 4.5.2, um weiterhin technischen Support und Sicherheitsupdates zu erhalten. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Support Lifecycle-Richtlinie für .NET Framework unter <a href="http://support.microsoft.com/gp/Framework_FAQ">http://support.microsoft.com/gp/Framework_FAQ</a> .
Microsoft .NET Framework 4.5	09.10.2012	Review-Hinweis	12.01.2016	Der Support für .NET Framework 4, 4.5 und 4.5.1 endet am 12. Januar 2016. Microsoft empfiehlt Kunden ein Upgrade auf .NET Framework 4.5.2, um weiterhin technischen Support und Sicherheitsupdates zu erhalten. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Support Lifecycle-Richtlinie für .NET Framework unter <a href="http://support.microsoft.com/gp/Framework_FAQ">http://support.microsoft.com/gp/Framework_FAQ</a> .
Microsoft .NET Framework 4.5.1	15.01.2014	Review-Hinweis	12.01.2016	Der Support für .NET Framework 4, 4.5 und 4.5.1 endet am 12. Januar 2016. Microsoft empfiehlt Kunden ein Upgrade auf .NET Framework 4.5.2, um weiterhin technischen Support und Sicherheitsupdates zu erhalten. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Support Lifecycle-Richtlinie für .NET Framework unter <a href="http://support.microsoft.com/gp/Framework_FAQ">http://support.microsoft.com/gp/Framework_FAQ</a> .

Der Support für .NET Framework 4, 4.5 und 4.5.1 endet am 12. Januar 2016. Microsoft empfiehlt Kunden ein Upgrade auf .NET Framework 4.5.2, um weiterhin technischen Support und Sicherheitsupdates zu erhalten. Weitere Informationen finden Sie in den häufig gestellten Fragen zur Support Lifecycle-Richtlinie für .NET Framework unter [http://support.microsoft.com/gp/Framework\\_FAQ](http://support.microsoft.com/gp/Framework_FAQ).

# Achtung

Supportende für .NET 4.0/4.5/4.5.1

War schon am 12. Januar  
2016

Von Microsoft  
bekanntgegeben am  
7.8.2014

Update auf .NET 4.5.2  
notwendig

Support für 4.5.2: Wie  
Betriebssystem (10 Jahre)

## Globale, Assembly- und Thread-übergreifende Variablen

```
14 System.AppContext.SetSwitch("CS60.APIDemo.AppContextDemo.Verbose", true);
```

```
18 bool verbose = false;  
19 System.AppContext.TryGetSwitch("CS60.APIDemo.AppContextDemo.Verbose", out verbose);
```

## System.AppContext

### Empfohlenes Format:

Switch.namespace.switchname  
Switch.library.switchname

# CultureInfo in Tasks

Bisher:

Neue Tasks verwenden die Systemeinstellung für Kultur  
Änderbar durch `DefaultThreadCurrentCulture` and  
`DefaultThreadCurrentUICulture` (wirken pro App Domain)  
`CultureInfo.DefaultThreadCurrentCulture =`  
`CultureInfo.CurrentCulture;`

Neu:

Neue Tasks bekommen die Kultur des Aufrufers  
Sogar beim Wechseln der App Domain  
Siehe [https://msdn.microsoft.com/de-de/library/system.globalization.cultureinfo\(v=vs.110\).aspx](https://msdn.microsoft.com/de-de/library/system.globalization.cultureinfo(v=vs.110).aspx)

# Time Conversion

Mit `System.DateTimeOffset` von/nach Unix-Zeiten

```
static DateTimeOffset FromUnixTimeSeconds(long  
seconds)
```

```
static DateTimeOffset FromUnixTimeMilliseconds(long  
milliseconds)
```

```
long ToUnixTimeSeconds()
```

```
long ToUnixTimeMilliseconds()
```

# WPF

WPF lebt!

Microsoft hat eine Roadmap veröffentlicht am  
12.11.2014

[http://blogs.msdn.com/b/dotnet/archive/2014/11/  
12/the-roadmap-for-wpf.aspx](http://blogs.msdn.com/b/dotnet/archive/2014/11/12/the-roadmap-for-wpf.aspx)

Am meisten gelesener .NET-Blog-Eintrag aller  
Zeiten  
und > 300 Kommentare

# Roslyn

## Bisher:

- Black-Box-Compiler (in C++)
- Werkzeuge nutzen CodeDOM und VS Code Model

## Neu:

- Compiler für C# und VB.NET reimplementiert
  - (nicht der C++-Compiler!)
  - C#-Compiler wurde in C# neu geschrieben
  - VB.NET-Compiler wurde auch in C# neu geschrieben
- Offene Plattform mit vielen APIs
- Basis für eine besseres Entwicklerlebnis ("Developer Experience")
  - IntelliSense, Refactoring, Code-Analyse,
  - Code-Generierung/-Transformation etc.
- Open Source (<https://github.com/dotnet/Roslyn>)
  - Eigene Erweiterungen möglich
  - Beiträge zum offiziellen Compiler möglich

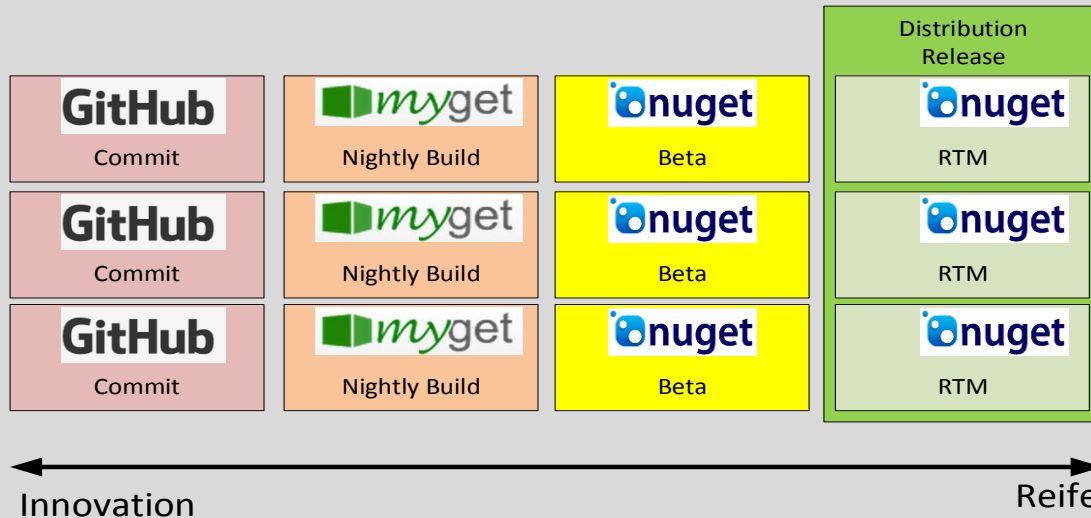
# .NET Core

	.NET "Full" 4.6	.NET Core 5.0
Anwendungsarten	Web, Desktop, Apps Console, Windows Services	Web, Win 10-Apps
Installation des Frameworks	Setup.exe	XCopy oder setup.exe
Innovationen	●●●	●●●●
Agilität	●	●●●●
Stabilität	●●●●	● bis ●●●● (je nach Releaseform)
Speicherplatzbedarf	●	●●●●
Geschwindigkeit	●●	●●●● (Startzeit, .NET Native)
Funktionsumfang	●●●●	● (wird aber steigen)
Zukunftsperspektive	●● (Support für 4.6 bis min. 2025. Es wird auch 4.x mit x>6 geben!)	●●●●

# Releasezyklen

.NET 4.x: wie bisher, ca. 1x im Jahr ein Update

.NET Core 5.x:





## Chancen

- .NET lebt und ist agiler denn je!
- Anpassung an Markttrends: .NET wird konkurrenzfähiger
- .NET auf allen Betriebssystemen
- Vereinheitlichung: „One .NET“
- Einfache Erweiterbarkeit und Austauschbarkeit

## Herausforderungen

- Pakete: Qual der Wahl, Risikobewertung, Abhängigkeitsmanagement, Update-Management, technische Herausforderungen bei Nuget
- Dokumentation leidet unter Agilität
- Umlernen & Umstiegsaufwand