

.NET 4.5, Visual Studio 2012 und Windows 8

Neuheiten aus der Microsoft-Welt

W3L AG
info@W3L.de

2012



Agenda

■ Motivation

- Betriebssysteme

■ Windows 8

- Modern-UI
- Windows Runtime
- Windows-Store

■ Visual Studio 2012

- Erneuerungen

■ .NET 4.5

- Update-Merkmale
- Neue Funktionalitäten

■ Fazit



Betriebssysteme

■ Was ist ein Betriebssystem?

- Ein Betriebssystem ist eine Software
- Es stellt...
 - einen Software-Stack für die Abstraktion von Hardwaredetails bereit
 - eine Benutzerschnittstelle bereit (UI)
 - eine Schnittstelle zur Anwendungsentwicklung bereit (API)

■ Problem: Änderbarkeit

- Umso mehr Schnittstellen eine Software bereitstellt, umso schwieriger ist es, Änderungen vorzunehmen
- Benutzerschnittstelle: Bereits erlerntes Wissen sollte erhalten bleiben
- Programmierschnittstelle: Bisherige Anwendungen sollten weiterhin funktionieren

■ Ohne Veränderungen sind keine Innovationen möglich!

- Welche neuen Wege und Kompromisse ist Microsoft mit Windows 8 eingegangen?

WINDOWS 8

Windows 8

■ Neuste Version des Betriebssystems aus dem Hause Microsoft

- Verbesserung von Stabilität und Geschwindigkeit
- Größte Veränderung seit 17 Jahren
- Grundlegende Erneuerungen
 - Neue Benutzer- und Bedienoberfläche
 - Lauffähig auf unterschiedlichen Rechnerarchitekturen
 - Neue sandbox-basierte Anwendungsart
 - Windows Runtime
 - Windows Store

Windows 8

■ „Bring-Your-Own-Device“-Trend

- Zusammen mit Windows 8 veröffentlicht Microsoft eine eigene Hardware namens „Surface“
- Surface ist ein 10,6-Zoll großer Tablet-Computer
 - ARM-Version und x86-Version



Quelle: <https://www.microsoft.com/surface>

■ Benutzeroberfläche

- Mit Windows 8 entfällt das bisherige Startmenü
- Stattdessen wird das neue „Modern UI“ eingeführt (*Vorher: Metro*)
- Übernahme des Kacheldesigns von Windows Phone 7
- Die „Modern UI“ verbindet die Tablet- und Smartphone-Welt mit dem klassischen Desktop-PC
 - Eine durchgehende Bedienoberfläche für das Smartphone, den Tablet, den Desktop-PC und die Spielekonsole

Windows 8 - „Modern UI“



Windows 8

■ Verschiedene Rechnerarchitekturen

- Der Windows-8-Kernel unterstützt neben dem x86-basierten Befehlssatz, den reduzierten Befehlssatz von ARM-Architekturen
- Die Windows-8-Variante für ARM-Architekturen trägt den Namen Windows RT
 - Vorinstalliertes Office 2013 RT (Word, Excel, PowerPoint und OneNote)
 - Einschränkung
 - Keine klassische Desktop-Oberfläche
 - Ausschließlich signierte „Modern-UI-Apps“ (Windows Store Apps)
- Ziel: Ein Betriebssystem für verschiedene Hardware-Plattformen
 - Desktop: Windows 8 (NT 6.2)
 - Server: Windows Server 2012 (NT 6.2)
 - Smartphone: Windows Phone 8 (NT 6.2)
 - Tablet: Windows RT (NT 6.2)
 - Spielekonsole: voraussichtlich Xbox 720 (~2013)

Windows 8

■ Anwendungsarten

■ Traditionelle Windows-Anwendungen

- Win32-Anwendungen oder .NET-Anwendungen
- Besitzen die vollen Berechtigungen des ausführenden Benutzers (Full-Trust)

■ Modern-UI-Apps / Windows Store Apps

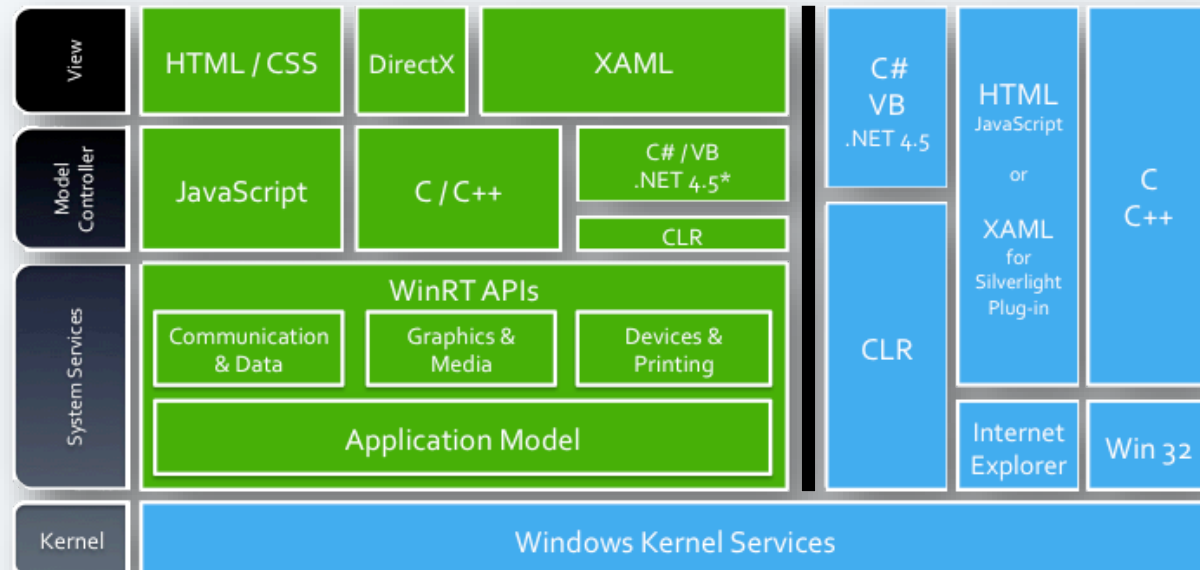
- Ausführung erfolgt in der Sandbox
 - Zugriffe auf Hardware-Komponenten oder Systemresources müssen vom Benutzer bestätigt werden
 - Der App-Lebenszyklus wird vom System verwaltet
 - Apps werden angehalten, wenn der Benutzer die App verlässt
 - Windows kann angehaltene Apps jederzeit beenden um Speicher freizugeben
 - Es existieren Callback-Methoden um den Anwendungszustand zu verwalten.
- Modern-UI-Apps müssen signiert sein → Schutz gegen nachträglicher Manipulation
- Entwickler benötigen eine Entwicklerlizenz
 - Kosten entstehen erst, wenn die App in dem Store veröffentlicht wird

Windows 8

Anwendungsarten

Modern-UI-Apps / Windows Store Apps

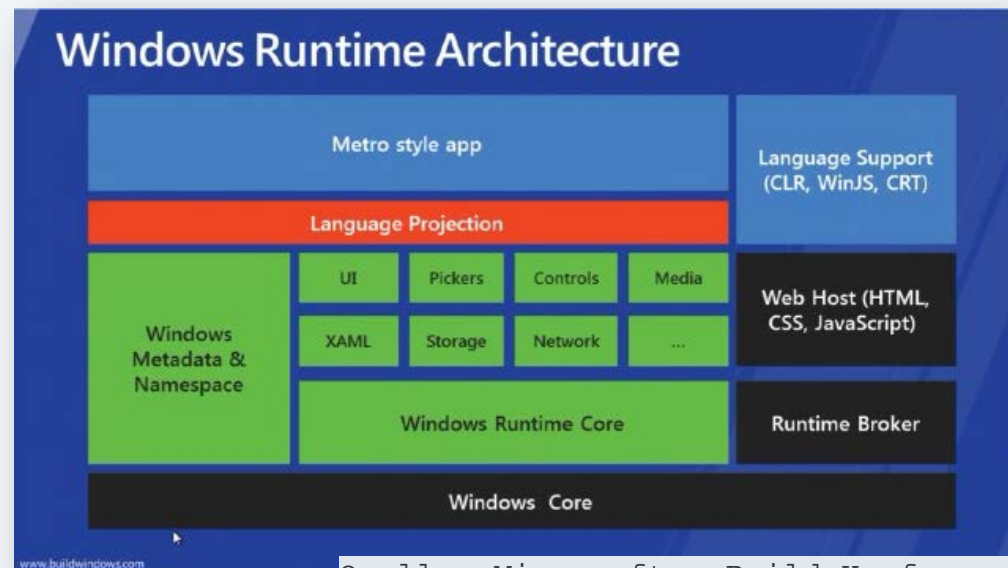
- Entwicklung erfolgt gegen die Windows Runtime (WinRT API)
- Unterstützte Programmiersprachen
 - C# und VB mit XAML
 - JavaScript mit HTML5 (Windows Library für JavaScript)
 - Visual C++ mit XAML oder Direct3D



Windows 8

Windows Runtime

- Vereint Win32, COM und .NET
 - Ähnlichkeit mit WPF/Silverlight und der .NET-Klassenbibliothek
 - XAML wird nativ von der Windows Runtime gerendert
 - C#- und VB-Apps werden weiterhin von der CLR ausgeführt
- Visual C++ component extensions (C++/CX)
 - Windows Metadata (.winmd), Namespaces, Interfaces, Events & Delegates, Properties, Partial Classes
 - Native Codeausführung



Windows 8

■ Windows Runtime - API

■ „Keep the UI thread responsive“

- Der Benutzer muss erkennen können, ob seine Anfrage noch bearbeitet wird
 - Bei Touch-Geräten gibt es kein Alt+Strg+Entf
- Kein „Application not Responding“-Signal wie bei Android-Geräten
- Lösung: Asynchrone Programmierung und „Background Tasks“

■ Alle API-Methoden, welche länger als 50 ms andauern könnten, sind nur in asynchroner Form vorhanden

- Anzeigen von Dialogen
- Arbeiten mit dem Dateisystem
- Senden oder Empfangen von Daten über Netzwerkverbindungen

Windows 8

■ Windows Store

■ Integriertes Verkaufsportal für Windows Anwendungen

- Die Anwendung muss mit dem „Windows App Certification Kit“ geprüft werden
- Nur signierte Anwendungen werden verkauft
- Über den Sideload-Mechanismus können Modern-UI-Apps auch ohne Einsatz des Windows Stores installiert werden (Szenario: Business-Applikation)
- Desktop-Anwendungen können nur mit einem „*company account*“ verkauft werden

■ Verkaufsmodell

- Jährliche Registrierungsgebühr
 - Natürliche Person: \$49 (37€)
 - Unternehmen: \$99 (75€)
- Umsatzbeteiligung
 - Gesamtumsatz mit der App unter \$25.000 → 70% an den Entwickler
 - Gesamtumsatz über \$25.000 → 80% an den Entwickler

■ Monitoring-Dashboard für den Verkäufer

- Absatzzahlen; Markt-Trends; Kundenbewertungen und Problemlberichte

VISUAL STUDIO 2012

Visual Studio 2012

■ Erneuerungen

- Verbesserung von Stabilität und Geschwindigkeit
 - Visual Studio 2012 reagiert schneller ; Setup ist um 20% schneller
 - XAML-Editor ist stabiler und performanter
- Neues „Look and Feel“ mit dem Metro-Style
- Unterstützung von .NET 4.5 und Windows Store Apps
- Windows XP und Windows Server 2003 werden nicht mehr unterstützt
 - Zielplattform: MFC und .NET ab 2.0 werden weiterhin unterstützt
- Erweiterung des „Application Lifecycle Managements“
 - Verbesserte „Team Foundation Server“-Anbindung
 - Storyboarding-Plugin für PowerPoint
 - Stakeholder werden über ein Feedbacksystem besser angebunden
- Bessere Werkzeugunterstützung für die Qualitätssicherung
 - Code Review Tools; Andere Unit-Test-Tools können direkt integriert werden

Visual Studio 2012

■ Erneuerungen

- LightSwitch-Integration ab Visual Studio 2012 Professional
 - Kann nun eine HTML5/JavaScript-GUI erzeugen
 - Neues Layout im Modern-UI-Style
- IntelliTrace von System im Produktivsystem (VS 2012 - Ultimate)
 - Problem: Debugging eines Produktivsystems ist in der Regel nicht möglich
 - Im Fehlerfall kann nun auf dem Produktivsystem ein Dump erzeugt werden
 - Dieser Dump kann in der Entwicklungsumgebung ausgewertet werden
 - Ähneln sehr stark einer echten Debug-Session
- Task „Suspend and Resume“ (TFS2012)
 - Die komplette Visual Studio Sitzung kann zu einem „Working Item“ im TFS gespeichert werden
 - Auf diese Weise kann die Arbeit bei Unterbrechungen wieder schneller fortgesetzt werden
- IntelliSense für JavaScript wurde verbessert

.NET FRAMEWORK 4.5

.NET 4.5

■ .NET 4.5

- Stellt ein „in-place“ Update von .NET 4.0 dar
 - Ersetzt die 4.0 Runtime, aber die Versionsnummer bleibt gleich (*v4.0.30319*)
 - Gleiches Vorgehen wie bei .NET 3 & 3.5 (CLR 2.0)
- Bestandteil von Windows 8 und Windows Server 8
- Update für Windows 7 und Windows Server 2008 vorhanden
- Kein Update für Windows XP und Vista!

■ Kompatibilität

- .NET 4.0-Anwendung sind Binary, Source und Serialization kompatibel mit .NET 4.5
- Einschränkungen:
 - Der SQL Server 1997 wird nicht mehr unterstützt
 - WCF – HTTP Status: 413 (Request Entity Too Large) statt 400 (Bad Request)
 - Ein MEF-Katalog kann nicht mehr serialisiert werden
 - Weitere: <http://msdn.microsoft.com/en-us/library/hh367887%28v=VS.110%29.aspx>

.NET 4.5

■ Performance-Verbesserungen an der CLR

■ Multicore Just-in-Time (JIT) Kompilierung

- Parallele Übersetzung von Programmteilen zur Laufzeit
- Im Idealfall ist die JIT-Übersetzung im Hintergrund-Thread schneller als im eigentlichen Haupt-Thread, sodass es zu keiner Verzögerung kommt
- Beschleunigt insbesondere den Anwendungsstart

■ Large Object Heap (LOH)

- Alle Objekte größer als 85kb werden auf dem Large Object Heap verwaltet
- Der LOH wird nicht vom Garbage Collector verdichtet. Dies bedeutet das eine Fragmentierung des LOH entstehen kann → OutOfMemory-Exception



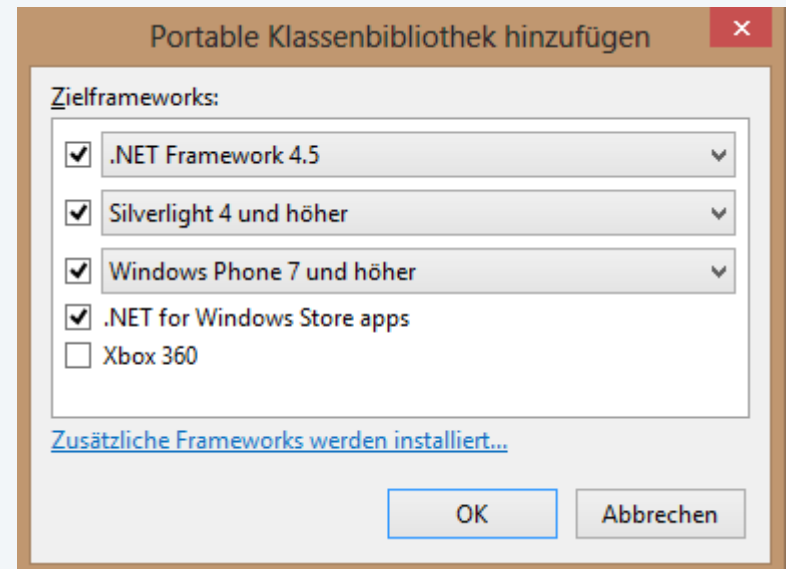
- In .NET 4.5 werden die freien Bereiche effizienter ausgenutzt → Fragmente sollen möglichst klein bleiben.
- Nach wie vor keine Defragmentierung des Large Object Heaps möglich

.NET 4.5

■ Neue Funktionalitäten

■ Portable Class Library

- Es existieren unterschiedene Library-Formen in .NET (CLR-Profile)
 - Standard .NET, Silverlight, .NET for Windows Store apps und Xbox 360
- Eine Portable Class Library kann in unterschiedlichen Projekttypen als Bibliothek eingebunden werden
- Keine Quellcode-Verlinkung mehr!



.NET 4.5

■ Neue Funktionalitäten

■ Asynchrone Programmierung mit Async und Await

- Goldene Regel: Performance-Engpässe vermeiden und die Reaktionsfähigkeit der Anwendung aufrecht halten
 - Beispiel: Wenn der UI-Thread synchron Inhalte aus dem Internet herunterlädt, kann das Fenster nicht verschoben oder die Größe geändert werden.

```
public async void Start()
{
    //Asynchroner Task wird ausgeführt
    Task<string> getWebPageTask = GetWebPageAsync("http://www.w3l.de");

    Console.WriteLine("IndependentWork...");

    string webText = await getWebPageTask;
    //Callback-Teil
    Console.WriteLine(" [Webseite geladen - Size:" + webText.Length + "] ");
}

private async Task<string> GetWebPageAsync(string url)
{
    //Task ausführen
    Task<string> getStringTask = (new HttpClient()).GetStringAsync(url);
    string webText = await getStringTask;
    return webText;
}
```

.NET 4.5

■ Neue Funktionalitäten

■ ADO.NET 4.5

- Unterstützung der SQL Server 2012 Datentypen: Geometrie und Geographie
- LocalDB-Funktionalität
 - Low Footprint Version des Microsoft SQL 2012 Express Servers
 - 1 User, 1 CPU, keine Replikation, max. 10 GB, kein Import und Export
 - Datenbank für den Clientrechner
 - Die Datenbank wird im gleichen Context wie die Applikation ausgeführt
- Asynchrone Funktion

■ Entity Framework 5.0 (auch für .NET 4.0)

- Performance Verbesserung bei LINQ-Abfrage durch automatisch Vorkompilierung (6x)
- Verbesserung des Modellierungswerkzeugs in Visual Studio 2012
- Neuer „Code First“-Ansatz
 - Entity-Klassen werden „händisch“ implementiert (POCO)
 - Datenbank wird automatisch angelegt
 - Die „Code First Migration“ ermöglicht ein Update von bestehenden Datenbanken

Fazit

■ Es wurden viele Erneuerungen eingeführt

- „Alte Anwendungen“ können im Desktop-Modus betrieben werden
- Erlernte Bedienkonzepte gehen teilweise verloren
- Vieles in der Windows Runtime erinnert an WPF bzw. Silverlight
 - Bisheriges Technologie-Wissen bleibt bei den Entwicklern erhalten

■ Innovationen führen zu einer Verbesserung der Performanz

- Durch die native Ausführung sind Windows Store-Apps schneller als bisherige WPF/Silverlight-Applikationen
- Asynchrone Vorgänge verbessern das Nutzungserlebnis (User Experience)

■ Verschiedene Entwicklergruppen werden abgeholt

- Web-Entwickler (HTML5/Javascript); .NET-Entwickler; C++-Entwickler

■ Einstieg von Microsoft in den boomenden Tablet-Markt [FTD]

- Microsoft ist nicht Monopolist, sondern muss sich gegen Apple und Google beweisen

Quellen und Links

■ Visual Studio 2012

- <http://msdn.microsoft.com/en-us/library/bb386063%28v=vs.110%29>
- <http://blogs.msdn.com/b/vsnewsde/archive/2012/05/31/jetzt-verf-252-gbar-visual-studio-2012-und-net-4-5-release-candidates.aspx>
- <http://blogs.msdn.com/b/visualstudio/archive/2012/05/08/visual-studio-11-user-interface-updates-coming-in-rc.aspx>
- <http://msdn.microsoft.com/en-US/library/vstudio/bb385832>

■ .NET Framework 4.5

- Application Compatibility in the .NET Framework 4.5
 - <http://blogs.msdn.com/b/dotnet/archive/2011/09/26/compatibility-of-net-framework-4-5.aspx>
 - <http://msdn.microsoft.com/en-us/library/hh367887%28v=VS.110%29.aspx>
- What's New in the .NET Framework 4.5
 - <http://msdn.microsoft.com/en-us/library/ms171868.aspx>

■ Windows Runtime

- Learn to build Windows Store apps
 - <http://msdn.microsoft.com/en-us/library/windows/apps/br229519.aspx>
 - <http://technet.microsoft.com/en-us/library/hh852635.aspx>
- Asynchronous programming in the Windows Runtime
 - <http://www.microsoft.com/en-us/download/details.aspx?id=27411>
 - <http://msdn.microsoft.com/en-us/library/windows/apps/hh464924.aspx>
- .NET neu erfunden in Native Code und COM
 - <http://www.heise.de/developer/artikel/Windows-8-Apps-benoetigen-neue-Windows-Runtime-1344071.html>

Vielen Dank!

Inhouse-Schulungen



Wir bieten Inhouse-Schulungen und Beratung durch unsere IT-Experten und -Berater.

Schulungsthemen

- Softwarearchitektur (OOD)
- Requirements Engineering (OOA)
- Nebenläufige & verteilte Programmierung

Gerne konzipieren wir auch eine individuelle Schulung zu Ihren Fragestellungen.



Sprechen Sie uns an!
Tel. 0231/61 804-0, info@W3L.de

W3L-Akademie



Flexibel online lernen und studieren!

In Zusammenarbeit mit der Fachhochschule Dortmund bieten wir

zwei Online-Studiengänge

- B.Sc. Web- und Medieninformatik
- B.Sc. Wirtschaftsinformatik

und 7 Weiterbildungen im IT-Bereich an.



Besuchen Sie unsere Akademie!
<http://Akademie.W3L.de>