

BUILD Windows 2011

Keynote 1

Die Highlights und das Fazit

Keynote 2

Die Highlights und das Fazit

Wir waren auf der BUILD-Konferenz in Anaheim - der Microsoft-Konferenz des Jahres 2011. Die wichtigsten Neuerungen und unsere Eindrücke haben wir hier für Sie zusammen getragen und ganz metro-like aufbereitet. Viel Spaß beim Lesen.

Unser Fazit

Bereit für die Zukunft. Bereit für Gemeinsames.

Windows 8

Die Besonderheiten des neuen berührbaren Windows

Designing Metro-Style

Die wichtigsten Schritte zur erfolgreichen App

Metro-Style Prinzipien

Das Design in nachvollziehbaren Prinzipien

Windows 8 Marktplatz

Endlich ein Marktplatz für Windows Apps

HTML-Apps mit Blend

Blend als WYSIWYG-Editor für HTML-Metro-Apps

Blend in Visual Studio

Das Beste von Blend für Entwickler

Der Tablet-PC

Unsere Beurteilung des Samsung Tablets



Die erste Keynote

Mit einem Gefühl von kindlicher Weihnachtsvorfreude sitzen wir in der letzten Reihe der großen Arena in Anaheim / Kalifornien. Microsoft hat zur BUILD-Konferenz geladen, um ihren Entwicklern die Zukunft von Windows zu präsentieren. Die Erwartungen sind groß bei uns wie mit Sicherheit auch bei den schätzungsweise 4.000 Teilnehmern in dieser Halle. Die Frage, die uns alle eint ist die Frage nach den großen Neuerungen, die uns hier erwarten werden.

Was macht Windows 8 aus?

Steven Sinofsky

Nach einem kurzen Rückblick auf die Erfolge der letzten Jahre gibt es den direkten Blick in die Zukunft. Steven Sinofsky verspricht uns eine Komplett-Änderung der User Experience in der Computer-Nutzung. Die Basis von Windows 8 ist es ein besseres Windows 7. Damit ist garantiert, dass alle Software, die unter Windows 7 läuft, auch unter 8 laufen wird. Mit der Windows 8 wurde Windows generell neu erfunden. Das Wesen der UX von Windows 8 ist stark mobileorientiert, touchbasiert, metrostyled und cloudangebunden.

Wie wird Windows 8 aussehen?

Julie Larson-Green

Der Look der neuen Windows 8 Oberfläche ist Metro "at its best". Der Login erfolgt per Gestensteuerung. Auf der Startseite begrüßen den Nutzer direkte interaktive Live-Tiles, hoch personalisierbar und anpassbar an die Nutzer-Wünsche und Benutzungsszenarien. Alles ist änderbar in der Verteilung und Priorisierung mit Hilfe entsprechender Gestensteuerung. Die Gestensteuerung ist in jedem Detail manifestiert: Swiping, Pinching, Dragging. Der Vorrang liegt auf dem Inhalt und der schnellst möglichen einfachen Interaktion damit. Auch der künftige Internet Explorer wird gezeigt. Man sieht allerdings nicht mehr viel bis gar nichts vom Browser, da der Rahmen fast komplett wegreduziert wurde zu Gunsten der Website-Inhalte. So wird der Besuch von Internetseiten ein Teil der gesamten Interaktion und des Systems. Die Applikationsanzeige einzelner Applikationen hat den Charme von Windows Phone 7. Die Denke scheint von wenigen schweren Software-Lösungen in Richtung vieler kleiner leistungsstarker Apps zu gehen, die parallel und multitaskingtauglich laufen, ohne sich zu stören, sondern miteinander interagieren. Alles wurde auf einem netten Tablet präsentiert, ist aber genauso auf Desktop PC vorstellbar.



Die erste Keynote

Womit werden diese neuen Apps gemacht?

Antoine Leblond

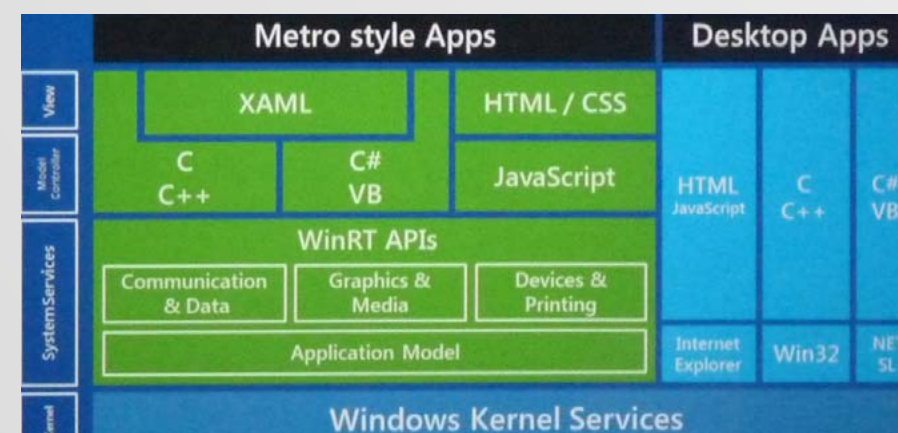
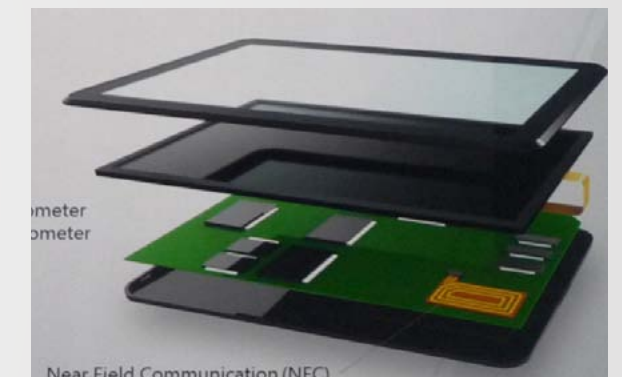
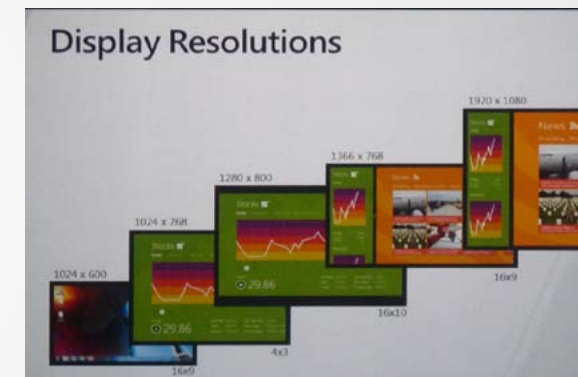
Ja. Hurra. Es ist XAML. OK, es ist wahlweise auch HTML ... Man kann sich selber die Sprache aussuchen, die man nutzen will. Das gilt sowohl für die Oberflächensprachen mit HTML und XAML sowie für den Code behind mit wahlweise Javascript, C#, C++ oder Visual Basic. Am Beispiel einer kleinen einfachen HTML-basierende App direkt in Visual Studio 11 umgesetzt, kann man sehen, dass es nicht sehr schwer ist: ein bisschen HTML5 mit Canvas Elementen und paar kleine Javascripte. Die touch-basierenden Befehle sind darüber eingebunden, so dass man z. B. mit einem Stylus etwas malen kann. Um das UI der XAML- oder HTML-App zu gestalten, benutzt man sinnvollerweise den WYSIWYG Editor Expression Blend. Das meint, dass in Blend nicht mehr nur noch XAML erstellt wird, sondern auch HTML5. Es war so logisch, weil sich HTML5 und XAML so ähnlich sind, aber wir hätten es nie zu träumen gewagt, dass es umgesetzt wird.

Es wird also genauso XAML-basierende Apps geben – je nach Wunsch des Designers bzw. Developers. Nicht, dass es damit schon genug wäre. Eine gute App gehört veröffentlicht. In Visual Studio ist auch schon eine Publikationsfunktion für den App-Store integriert, so dass ich meine kleine feine App sofort in den Store portieren und der Welt zur Verfügung stellen kann. Der Store wird uns somit auch präsentiert.

Was macht die Hardware aus?

Antoine Leblond

Es gilt für den Desktop-PC, das Laptop und den Tablet: die neue Hardware ist performant, grafikstark, multitouch, daueronline, mit riesen Power und Datenspeichern. Es ist sind waschechte Super-Computer. Leistungsstarke Grafik für Touch, Widescreen in perfekter Darstellung. Ach ja, und dieses nette kleine Samsung-Tablet mit Windows 8 drauf und mit der Power eines Developer-PCs drunter ist das Geschenk an alle BUILD-Teilnehmer. Danke Microsoft!!!



Unser Fazit

Der neue Samsung-Tablet-PC ist prima - auch nach dem Praxistest. Windows 8 erscheint uns in Bezug auf Touch und Mobile als die dringend notwendige Reaktion auf Marktanforderungen. Die Chance mit diesem Konzept sich nicht nur im Desktop PC, sondern auch im Bereich Tablets als echte mobilen PC zu positionieren ist damit gut genutzt. Das Metro-App-Konzept ersetzt nicht Windows, sondern es erweitert es. Und das macht es für uns schlüssiger und intelligenter als die Tablets, die nur geprägt sind von oft unnützen Apps. Dass Design in die Microsoft-Welt Einzug hält, hat seinen Beginn schon mit Einführung von WPF im Jahre 2006. Mit dem Metrostyle-Konzept, das im Zuge der Windows Phones 7 eingeführt wurde, ist User-Interface-Design erstmalig auf herausragende Weise umgesetzt worden. Das Warten der letzten Monate auf eine schlüssige Strategie und ein rundes Bild seitens Microsoft hat sich in unseren Augen also gelohnt.

Die zweite Keynote

Was macht Backend und Server aus?

Satya Nadalla

Große Designs und Applikationen bedürfen entsprechender Backends und Server, damit sie volle Leistungsfähigkeit haben. Mit Windows Server 8 und Windows Azure werden die entsprechenden Strukturen geboten durch eine elastische Infostruktur, komponierbare Service, Data as a Namespace, Identity & Access und kontinuierliche Verfügbarkeit.

Wie wird die neue Power umgesetzt?

Jason Zander

Am Beispiel eines Multiuser-Shooting-Games wird die neue Leistungsfähigkeit gezeigt und direkt am lebenden Objekt weiterentwickelt. Die neuen Azure-Services sind mit Tag heute verfügbar. Die Oberfläche des Games ist in XAML realisiert, die Game-Engine in C# und die Services sind Azure. Die 3D-Elemente können direkt in Visual Studio teils WYSIWYG editiert werden und sogar innerhalb von DirectX bis in die tiefsten Detailebenen hinein, was ich persönlich sehr beeindruckend fand. Diese Features sind ein Teil des neuen Visual Studio 2011.

Was macht die neue Plattform aus?

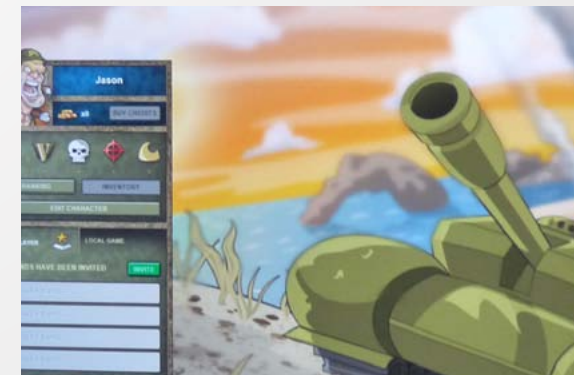
Scott Guthrie

Vorgestellt wird das neue .NET Framework 4.5. Im künftigen Visual Studio gibt es einen fantastischen Preview-Modus. Die Besonderheit ist direkte Output-Input-Markierung. Für ASP und HTML-Seiten können nun CSS-Files gemergt werden und die Größen bleiben kontrollierbar. JQuery-Mobile wird mit der nächsten ASP-Versionen verfügbar sein. Damit wird die Website auch für fast alle Smartphone-Systeme realisierbar. Durch Anbindung entsprechende Services können hochleistungsfähige Webanwendungen erstellt werden. Windows Azure macht das Handling vieler User gleichzeitig möglich. Durch das Handling des gesamten Projektes über den Team Foundation Server werden das Projekte erfolgreich über Scrum realisiert. Visual Studio 11 sowie .NET 4.5 ist ab sofort zum Download verfügbar.

Was leistet der Windows Server 8?

Brain Surace

Mit diesem Release für Windows Server kommen die gravierendsten Änderungen und Verbesserungen im Vergleich zu vorherigen Releases. Durch Windows Server 8 werden extra hohe Datenübertragungen im Gigabyte-Rahmen möglich mit geringer CPU-Belastungen. Eine Migrationsumsetzungen sind nahezu nahtlos möglich.



Die zweite Keynote

Wie nutze ich den Data Marketplace?

John Shewchuk

Content-Daten werden nicht nur gehostet, sondern können über Azure auf einzigartige Weise genutzt und konsumiert werden. Somit können im Data-Marketplace zur Verfügung stehende Daten in eigene Apps eingebunden werden. Am Beispiel einer Reise-Applikation für diverse Systeme wird die Nutzung der Inhaltsdaten aufgezeigt. Die Applikation sammelt mittels Windows Azure durch entsprechende Bindung Daten an verschiedensten Stellen und Services zusammen. Diese Beispielapplikation kann als Code-Sample eingesehen werden.

Einsatz HyperV an Azure in Autos

Dan Fernandez/Jeff Sandquist

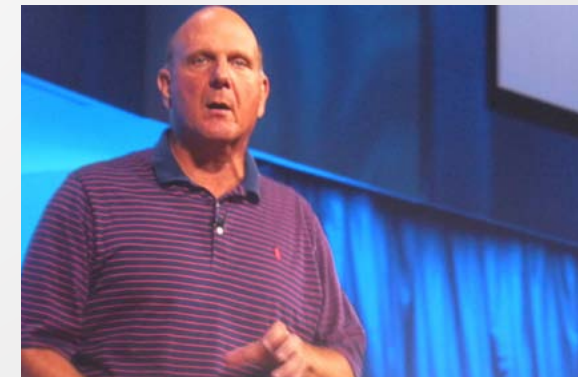
Die Demo zeigt ein kleines HyperV-Gerät, das in ein Auto eingebunden wird. Mittels über Azure übermittelten Daten kann der Fahrer von anderer Stelle überwacht werden hinsichtlich Position durch GPS-Daten, Fahrverhalten und Geschwindigkeit. Das System kann umgekehrt versorgt werden mit Kommunikation z. B. über das Windows Phone 7. Mit den neuen Technologien sind spannende neue Szenarien möglich (Gestensteuerung, Augmented Reality). West Coast Customs baut hoch individualisierte Autos und wird diese MS-Technologie nun integrieren. Um es mit dem Lieblingswort der Konferenz zu sagen: Awesome!

We want you all!

Steve Ballmer

Mit impressiver Stimme und charismatischem Auftritt berichtet uns der Microsoft CEO von seinen Eindrücken, aber auch seinem Stolz, dass die neuen Entwicklungen so gut angekommen. Sehr angenehm ist, dass er bei aller Begeisterung daran erinnert, dass wir alle erst ganz am Anfang stehen in einer Entwicklungs- und Änderungsphase. Er erinnert auch daran, dass das Kerngeschäft von Microsoft Windows hinter allem steckt. Mit Stolz verweist Steve auf die Erfolge von Windows Phone 7 und die anhaltende Positiv-Entwicklung. Die Vision schließt sich an der Stelle, wo auf neue künftige Hardware verwiesen wird: Kinect, Tablets, Phones, CloudData-Centers, neue PCs und Arms.

Die Cloud Services nehmen ihre Rolle ein zur Identifikation im E-Commerce Bereich, beim Datenaustausch, Datennutzung, Zusammenarbeit, Kommunikation, Netzwerken und Sicherheitskontrolle. Die Applikationen werden auf 7 Szenario-Plattformen ausgerichtet: Windows, Windows Phone, XBox, Windows Azure, Office, Bing, ERP/Dynamics. Kein System außer Windows erzielt Verkaufszahlen von über 350 Mio. Geräten. Entwickler im Microsoft-Umfeld haben also dieser Tage beste Chancen. Maximaler Return of Invest ist möglich, wenn man gemeinsam mit Microsoft voranschreitet. Es sind die Tage der Windows Developer!



Unser Fazit

Die Entscheidung, Teil des Ganzen zu sein, hat von unserer Seite weiterhin Bestand, weil wir mit unserem User Interface und XAML-orientierten Tätigkeitsfeld perfekt dazu passen. Mit den neuen hier aufgezeigten Möglichkeiten, wird für uns das Spielfeld größer und spannender werden. Da unsere Ausrichtung XAML und UI nun als richtig bestätigt hat, werden wir in nächster Instanz den Schritt in die GmbH gehen, um uns noch weiter auf das Umfeld einzustellen.

Windows 8

Ein neuer Look für Windows

Der Look der neuen Windows 8 Oberfläche ist Metro "at its best". Der Login erfolgt per Gestensteuerung. Auf der Startseite begrüßen den Nutzer direkte interaktive Live-Tile, hoch personalisierbar und anpassbar an die User Wünsche und Szenarien. Alles ist änderbar in der Verteilung und Priorisierung mit entsprechender Gestensteuerung. Die Gestensteuerung ist in jedem Detail manifestiert: Swiping, Pinching, Dragging. Der Vorrang liegt auf dem Inhalt und der schnellst möglichen einfachen Interaktion damit. Auch der künftige Internet Explorer 10 wird gezeigt. Man sieht allerdings nicht mehr viel bis gar nichts vom Browser sieht, da der Rahmen ist fast komplett wegreduziert wurde zu Gunsten der Website. So wird der Besuch von Internetseiten ein Teil der gesamten Interaktion und des Systems. Die Applikationsanzeige einzelner Applikationen hat den Charme von Windows Phone 7. Die Denke scheint von wenigen schweren Software-Lösungen in Richtung vieler kleiner leistungsstarker Apps zu gehen, die parallel und multitaskingtauglich laufen, ohne sich zu stören, sondern mit einander interagieren. Alles wurde auf einem netten Tablet präsentiert, ist aber genauso auf Desktop PC vorstellbar.

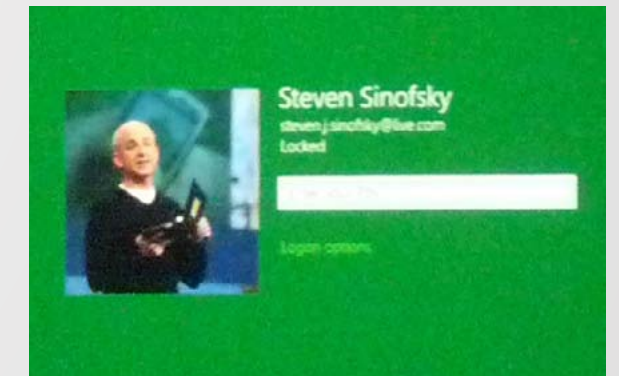


Höchst "persönlich" im Start

Mit dem neuen Startseitenkonzept und den dort enthaltenen Livetiles und Apps wird der PC wahrhaft zum Personal Computer. Die Livetiles stellen die aktuellsten Belange des Benutzers dar und sind damit kleine Assistenten. Die über die Startseite aufrufbaren Apps wiederum kann der Benutzer aus dem AppStore seinen Bedürfnissen gemäß zusammenstellen. Ein direkter Click in die App selber zeigt, dass mit der Startseite und dem App-System man einen Tablet mit einem Benutzererlebnis hat, wie man es von den entsprechenden Mitbietern kennt - allerdings geprägt vom Metrostyle statt dem Glossy Style.

Windows 7 in Windows 8

Da Windows 8 kein reines Tablet-Betriebssystem ist, sondern weiterhin auch Bestand hat auf klassischen Desktop PC, stellt innerhalb von Windows 8 das vollweiterig Windows 7 - aber mit gravierenden Verbesserungen zur Verfügung. So sind Installationsprozesse z. B. sehr vereinfacht worden. Über den Klick auf "Desktop" hat der Benutzer seinen klassischen Windows 7 Desktop mit allen entsprechenden Software-Applikationen zur Verfügung. Der Wechsel fühlt sich allerdings etwas hart an und ungeschlüssig an. Eine überzeugendere Integration des Desktops und der Windows 7 Expertensoftware fehlt noch.



Unser Fazit

Den Eindruck, dass man sich als Benutzer in eine "alte Welt" zurück bewegt, wenn man seine Software nutzen möchte, wirkt daher ein bisschen wie Zwangsbeglückung. Hier wünschen wir uns schlüssigere Lösungen, da sonst das Häkchen "Metrostyle ausschalten" zu einer Lieblingsoption der Benutzer werden könnte. Ansonsten sind wir sehr froh über die Durchsetzung eines guten Design-Konzeptes, einem sehr benutzer- und touchfreundlichen Bedienkonzept und einer Betriebssystem, dass einfacher zu bedienen kaum sein kann.

Designing Metro-Style

Das Metro-Design

Die Silhouette einer Metro-Style-App

Konsistente UX wird erreicht durch konsequente Anwendung des Metro-Styles. Es gibt ein klares Raster, das immer wieder Anwendung findet. Es gibt als Hilfestellung entsprechende Templates z. B. eine Grid-Applikation. Der Style ist durch entsprechende Anbindung vorhanden.

Content vor Rahmen.

Ein Software-Rahmen macht bei entsprechender Größe Anwendungen Sinn, besonders bei klassischen Desktopsoftware-Anwendungen wie z. B. Photoshop. Metro-Style-Apps haben Raum zum Atmen und verschwenden keinen Raum für Software-Rahmen. Vergleicht man IE 9 und 10 reduziert sich der Rahmen des Browsers gegen 0.

Charms (Buttons) als Analogie zur Appbar

Die App soll nicht mit Buttons zugesperrt werden. Interaktion mit dem Content ist meistens hinreichend.

NUI und Gestensteuerung

Schaffen Sie natürliche Benutzerinterfaces, die man anfassen und begreifen kann.

Designing für Komfort

Man hat festgestellt, dass Dinge am Rand besser wahrgenommen werden als in der Mitte.

Schritte, um Design besser zu machen:

Luft integrieren

Alles wegreduzieren, was kein echter Content ist.

Charm als Appbar integrieren am Rand.

Die Schönheit von Typografie wirken lassen.

Schnell und flüssig

Es geht um hohe Performance und flüssige Bewegung. Da wir direkt mit dem Screen interagieren, ist es wichtig, flüssige und zügige Reaktionen zu haben. Die Kunst liegt in guter Performance und entsprechender Animation. Windows 8 unterstützt entsprechende Gesten und reagiert mit entsprechenden Animationen darauf. Slide und Drag ohne Warten. Eine andere interessante Geste, ist der optische Zoom, ein inhaltliches Hineinzoomen darstellt in sich vereint (nicht vergrößern, sondern rein- und rausbewegen).

Touch-Verhalten als erstes Designen

Jeder PC der Zukunft wird touch gesteuert sein. Daher steht in der Designreihenfolge nun erst das Interaktionsdesign in Bezug auf touch und dann das Screendesign.

Daher: Nutze die Template, die Tools und die Controls. Alles, was wir brauchen ist enthalten.

Skalierung über verschiedenen Screengrößen hinweg.

Snap and Scall beautyfull.

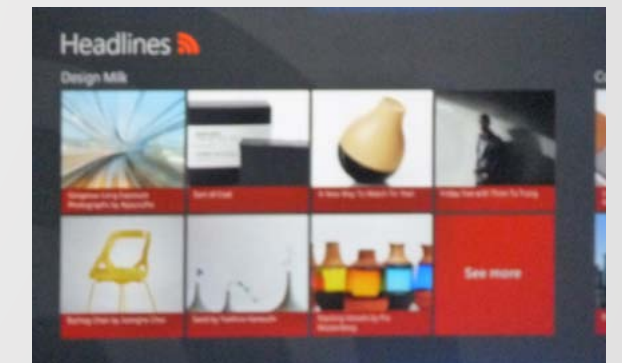
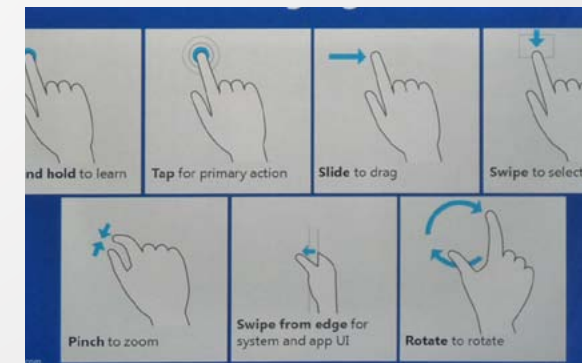
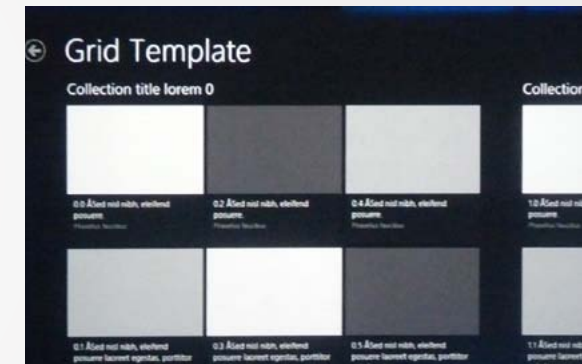
Wenn man sich an die Gesetze: Vektorgrafik, XAML, SVG und CSS hält.

Basis 100% = 1024*768

Widescreen 1366*768

SnapView = Je App muss den Snap-Status implementieren. Ein kleiner reduzierter Status, der am Rand ist.

PortraitView assistiert beim Drehen des Tablets.



Designing Metro-Style

Herstellen der Verbindungen

Das bedeutet Interaktion mit anderen Applikationen. Die eine Applikation ruft eine andere App auf und nutzt deren Funktionen. Z. B. Malen und Versenden. Zwei Apps arbeiten zusammen. Gleichermaßen werden Dinge geteilt: Inhalte, Kontakte, Daten. Die Suche z. B. übergreifend integrierbar.

Investition in große Interaktionsflächen

Ein Tile ist ähnlich zu Icon-Idee. Icons sind gestrige Ideen von App-Repräsentaten. Tiles haben ein eigenleben. Sie werden mit einem kleinen Bild repräsentiert, sie haben aktuellste Infos, sie leben vom beinhalteten Content. Man kann Typo, Bilder, Pfad-Elemente integrieren. Jeder Teil einer App kann als Tile „gepinnt“ werden auf die Startseite. Damit erhält die Startseite höchste Aktualität.

Verbunden und lebendig sein

Live-Tiles verbindet Menschen mit ihrem persönlichen Computer. Sie können klein und groß sein. Es sind für den Nutzer und ihre Bedürfnisse gedacht. Nicht für Werbung!!!
Notifications: Es geht um interessante Meldungen, die für den

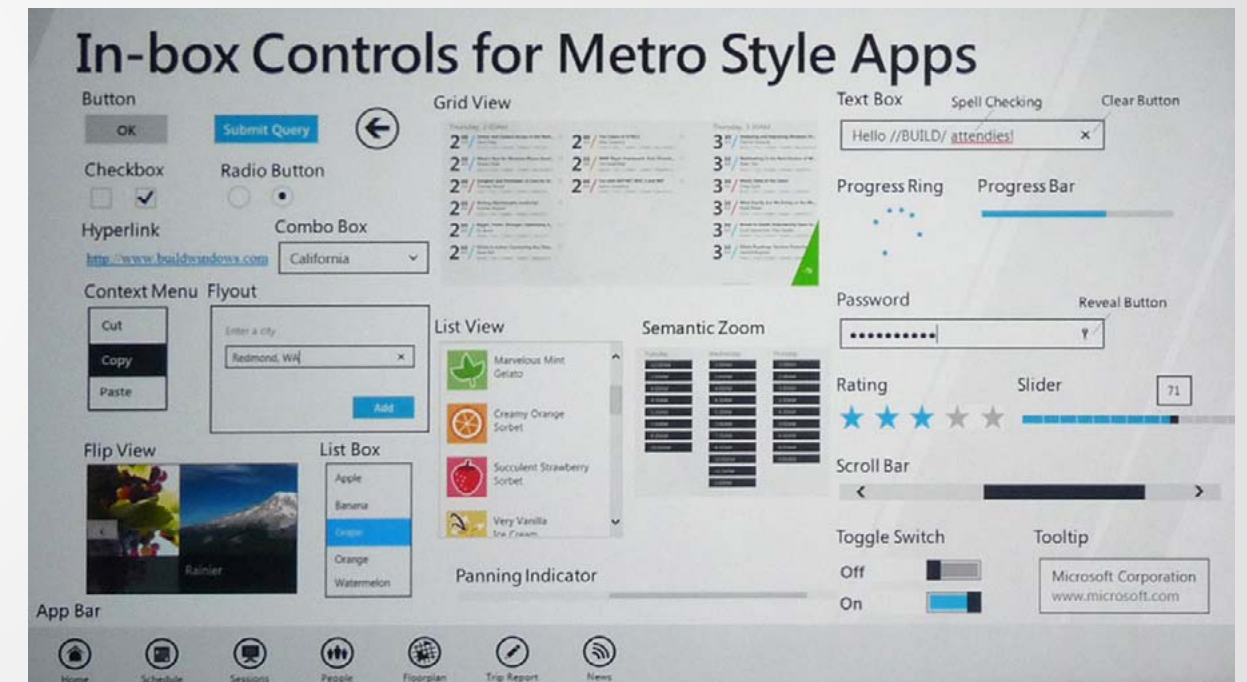
Nutzer da sind. Es geht nicht um Junk-und-Spam. Sie kommen momentbezogen und können schnell wieder entfernt werden. Notifikations sind nur sparsam zu nutzen. Das Update sollte ganz ziel genutzt werden.

Speicherung in der Cloud

Mit der Cloud wird es möglich, dass ich PC und Device-unabhängig werde. Die Stati und Settings werden in der Cloud festgehalten. Beispiel Spielstände, die nun gemerkt werden können und damit auch von anderen Rechnern angezeigt werden können.

- Roaming
- Explizites Sichern wird nicht mehr nötig, da sich die App die Stati merken.
- Windows merkt sich Settings und Stati

Embrasse Metro Style!!!



Unser Fazit

Design wird oft als Geschmackssache gewertet. Weil hier aber klare Prinzipien von guten Design angewandt wurden, die ähnlich gegeben sind wie die Prinzipien von guten Coding, gelten sinnvoller Weise viele Danke als gegeben. Wir dürfen nicht vergessen, dass wir hier auf einer Developer-Konferenz sind. Microsoft hat somit ihren Entwicklern, die ja nun mal keine Designer sind, eine perfekte Hilfestellung an die Hand gegeben für besseres Design, dass direkt integriert ist. An den vielen von Entwicklern erstellten Windows Phone 7 Apps konnte man schon sehen, dass vorintegriertes Design es zwar besser macht, aber dennoch reichlich schlechte App hervorgebracht hat. Gleiches konnte man sehen, als Designer meinten WP7 Apps ohne Entwickler zu Designen. Der Erfolgsweg liegt also in Anwendung des MetroStyle unter Zusammenarbeit von Designern und Entwicklern.

Metro-Style Prinzipien

Bauhaus-Stil.

Bauhaus ist eine deutsche Design-Bewegung und Design-Schule aus den 20ern. Bauhaus war eine Gegenbewegung auf Überfluss und Überdekoratation. Die industrielle Revolution hatte die Welt geändert, aber die Künstler haben sich in gestrigen Designideen bewegt. Ebenso wie heute. Die neue Anforderung ist also wieder Reduktion und Form follows Funktion.

Swiss Design

Die Swiss-Typografie ist geprägt von Lesbarkeit und Klarheit. Man sieht diese Designsprache besonders auf Flughäfen und Verkehrsstrukturen. Die Typografie-Ausdruck ist auch international. Weil wir in schneller Bewegung sind, brauchen wir schnell erfassbare klare Informationografie. Die Mittel des Swiss Design sind Typografie, Raster, flache Farbschemata.

Cinematografisches Motion Design.

Motion/Bewegung erzeugt Emotion. Bewegung erweckt Interface zum Leben. In Film-Intros wird Typografie in Bewegung gebracht.

Metro Style Design Principles

Prinzip 1: Pride in craftsmanship

Metro ist ein sehr modernes Design, das findet Anlehnung an den traditionelle Designprinzipien.

Sweat the details: Das Große und Ganze muss für den Benutzer stimmig sein. Daher muss viel Arbeit ins Detail gesteckt werden.

Make it safe and reliable: Design müssen Benutzerführung, Performancen, Informationsdarstellung fördern.

Balance, symmetry, hierarchy: Ausgewogenheit, Symetrien und Hierarchien führen das Auge.

Align to the grid: Raster helfen, Orientierung, Struktur und Raumnutzung aufzubauen. In einem Raster findet man die Grundlage für Ordnung und Freiräume.

Prinzip 2: Be fast and fluid

- Life is mobile
- Delight with motion
- Design for touch
- Intuitive interaction
- Be responsive and ready
- Be immersive

Prinzip 3: Authentically digital

- Cloud connected
- Dynamic and alive
- Beautiful use of typografie
- Bold virant colors
- Motion

Prinzip 4: Do more with less

- Be great at something
- Focused and direct
- Content before chrome: Es geht um den Inhalt und nicht den Rahmen – bei Browsern, Software und Apps.
- Inspire confidence

Prinzip 5: Win as one

- Fit into the UI model
- Reduce redundancy
- Work together to complete scenarios
- Tools and templates are design to scale: Es gibt viele Hilfes-



Unser Fazit

Design wird oft als Geschmackssache gewertet. Weil hier aber klare Prinzipien von guten Design angewandt wurden, die ähnlich gegeben sind wie die Prinzipien von guten Coding, gelten sinnvoller Weise viele Danke als gegeben. Wir dürfen nicht vergessen, dass wir hier auf einer Developer-Konferenz sind. Microsoft hat somit ihren Entwicklern, die ja nun mal keine Designer sind, eine perfekte Hilfestellung an die Hand gegeben für besseres Design, dass direkt integriert ist. An den vielen von Entwicklern erstellten Windows Phone 7 Apps konnte man schon sehen, dass vorintegriertes Design es zwar besser macht, aber dennoch reichlich schlechte App hervorgebracht hat. Gleiches konnte man sehen, als Designer meinten WP7 Apps ohne Entwickler zu Designen. Der Erfolgsweg liegt also in Anwendung des MetroStyle unter Zusammenarbeit von Designern und Entwicklern.

Der Windows Store

Es wird einen Store ähnlich zum WP7-Store geben. Die genauen Business-Spezifikationen werden kommuniziert, wenn die Partnerschaften geschlossen sind für den Bau die entsprechenden Applikation. Auch der Faktor Partnerschaft ist neu erdacht worden. In mehreren Stufen wird man unterstützt von den Tools, über Marketing, Monetarisierung, Feedback. Microsoft möchte hier helfen, Geschäfte mit den Kunden entwickeln. Die App wird nicht über den Store alleine gehen müssen. Die Entwickler entscheiden es selber, ob der Store, der eigene Shop oder die internen Distributionswege genutzt werden sollen.

Curation and protection:

Der Microsoft-Store wird der einzige Platz sein, wo Metro-style-Apps herausgeben werden. Microsoft checkt in einem 3 Stufen-Programm jede App. Technische Zertifizierung, Security und Inhaltliche Korrektheit.

Consistency and predictability:

Die Qualitätsanforderungen von Microsoft sollen erfüllt und sichergestellt werden, um damit künftig gute Geschäft zu

Commitment:

Es geht um gemeinsam Verträge und Commitments. Es geht um Vertrauen und ein Miteinander.

450 Millionen Windows 7 Lizenzen sind verkauft worden in 2 Jahren.

Es werden also große System versorgt mit der Software, aber mit über 500 Millionen Windows 8 Kandidaten ist man entsprechend großes Geschäftsfeld.

Windows 8: Reimagine user experience, new application model, form factor diversity

Store: Integration, Discovery, Telemetry

User Experience:



Das Benutzer-Erlebnis soll ganz stark von Metro-Style geprägt sein. Nur so kommt man in den Store.

Das ist die Story:

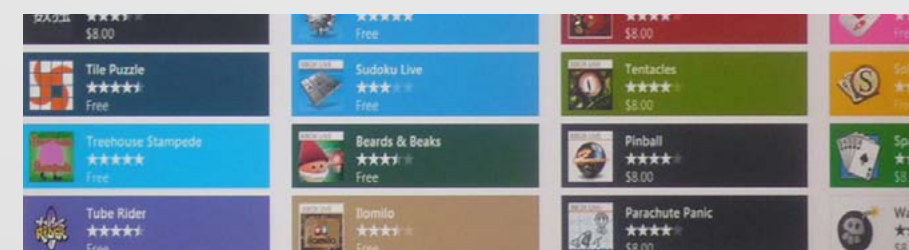
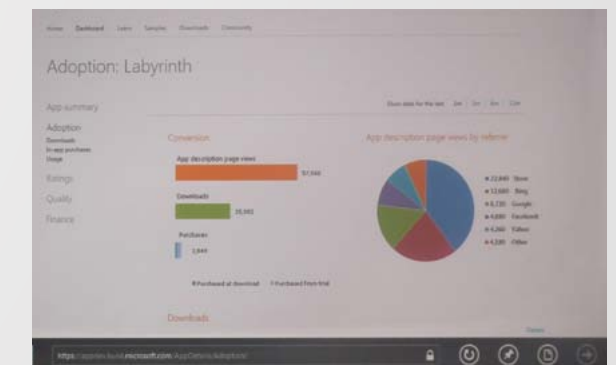
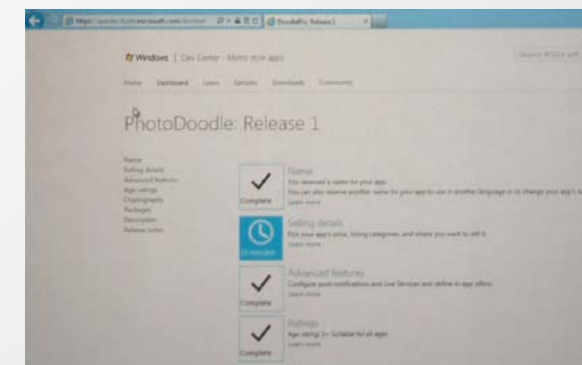
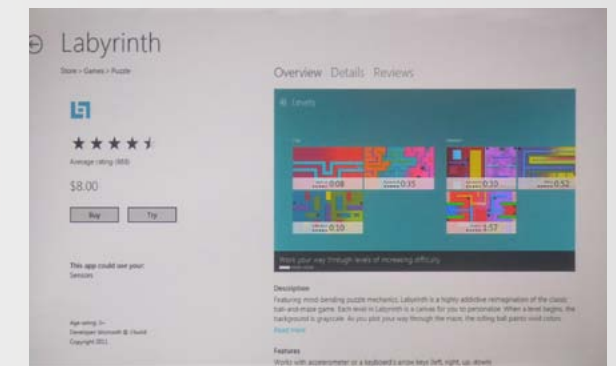
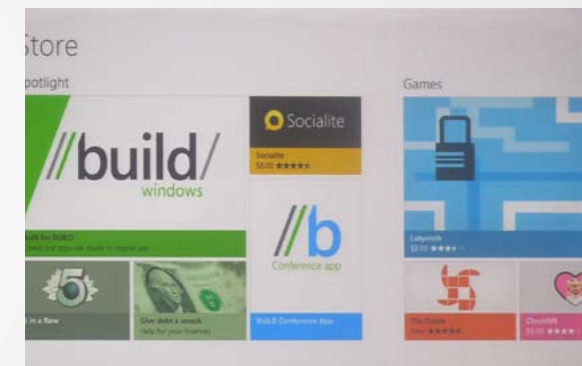
Der Benutzer ruft von seiner Startseite die Storeseite auf. Es wird entsprechende Kategorieebene geben, wo sich der Benutzer Apps aussuchen kann. Der Durchschnittspreis für eine App wird 8 \$ sein. Nur Apps, die dem Nutzen einen Mehrwert geben, sind Apps mit Chancen. Jede App hat eine kleine Landingpage. Es wird ein App-Lizenzierungs-Modell geben. An dieser Stelle wird zurecht in Erinnerung geführt, dass es sich hier um moderne Software handelt, die hier vertrieben und publiziert wird.

Der Entwickler selber ist derjenige, der das Business anschiebt.

Discoverability:

Die Suchindizierung wird durch MS den Store erstellt werden, aber auch gleichermaßen durch IE10 als Browser. Desktop-Apps laufen ebenfalls über den Store. Erkannt werden die Apps durch entsprechende Meta-IDs. Der Zertifizierungs-KIT ist jetzt verfügbar und auf den Tablets die gestern ausgehändigt wurden. Die Qualität der Apps sollte und kann offline getestet werden, aber den Schlusstest wird Microsoft machen. Eine Vorzertifizierung kann man mit dem KIT machen und bekommt entsprechende Ergebnisse.

Als weiteres Tool für Entwickler wird es ein Dashboard geben unter dev.windows.com. Dort wird der gesamte Zertifizierungsprozess abgebildet. Ebenso wird über das Dashboard auch der Erfolg und die Statistik der App abgebildet. Das Dashboard ist auch ein Debugging-Tool hinsichtlich Exceptions.



Unser Fazit
Bla

Metro-Apps in XAML

XAML Styling und Animation

Vieles, von dem, was es hier zu hören gibt, dürfte sich auf Silverlight-TV wiederfinden. Hier geht es um XAML. Sehr sympathisch finde ich die explizite Ausladung an Leute, die kein Interesse an XAML haben. Er wird uns die Metro-Style-Dinge zeigen, die out-of-the-box mitkommen, ebenso aber auch den hohen Grad an Individualisierungsmöglichkeiten.

Control Styling

Am Beispiel der Memories App wird er uns diverse Out-Of-the-Box-Beispiele zeigen. Eines der Controls ist die Grid-View-Control, die Layout-Strukturen und Animationen von Haus aus hat. Das ResourceDictionary mit den Impliziten Style ist ähnlich wie das des Windows Phone 7. Alle Controls sind als Styles optimal umgesetzt und als helle oder dunkle Variante vorgesehen. Die Source „light_generic.xaml“ muss nur integriert werden und ist damit implementiert. Sie können über den üblichen Weg drag and drop auf die Bühne gezogen werden im VisualStudio. Wenn der Hintergrund auf Hell geändert wird, ändern sich (wie auch immer) magisch die Controls in das dunkel Theme.

Er zeigt uns nun ControlStyling – also das umändern des Standards. Das beantwortet auch die Frage, ob man den Style ändern darf. Technisch ist es ein Switch von implizit zu explizit. Aus Designersicht ist dieser kleine Schritt eine große Aussage: Metro darf angepasst werden – auf Branding und auf eigene Designideen. Es darf also etwas erstellt werden, was zur eigene Sache paßt unter Beibehaltung der Prinzipien. Technisch gesehen ist das „Based on“.

Auch das beinhaltet wieder eine indirekt Aussage: Man hält sich in Windows 8 an die Strukturen von Silverlight und nicht an die Triggers von WPF. States einer Control werden über den VisualStateManager realisiert. Es gibt keinen Click-EventHandler.

Animationen

Build in Animations for free: Die Default-Controls haben Animationen integriert. Ebenso auch Template-Pages.

Es gibt mit den neuen Tools eine Animationsbibliothek, die von Hause aus integriert sind. Sie müssen nur angewandt werden. Diese Animationen haben ein ansprechendes Motion Design sowie ein gute Performance.

In- und Out-Animationen: Wenn Seiten sich wechseln.

Content-Animationen: Wenn Content sich ändert.

Repositionierung: Wenn etwas drag and drop bewegt wird.

Add-Delete: Wenn Dinge hinzugefügt werden.

Hinzugefügt werden sie durch ein paar kleine Zeilen XAML-Code und die Verfeinerung. (ähnlich wie bei Behaviours)

Und viele andere wie TapUp, PopIn, Reposition, CrossSlideBack, Drag, Im Grunde immer das gleiche Prinzip: Auf ein Event wird eine optimale Animation des betreffende Elements ausgelöst.

Animationen können selbstverständlich auch selbergemacht werden wie vormals in Silverlight. Man kann sich die Quälerei im Coding antun oder man kann es in Blend machen wie gewohnt.

1. Animations innerhalb der impliziten Controls nutzen
2. Für die Konsistenz sollte man die Library nutzen und spielerisch mit der Anwendung der Animationsverhaltens umgehen.
3. Eigene Animationen sollen zum Metro-Style passen und ein ähnliches Motion-Design haben.
4. Des Weiteren gibt es SDK-Sample, die man nutzen und aus denen man lernen kann. In einem Beispiel werden zum Beispiel über den VisualStateManager gemischte Animationen realisiert: Fadeln der gesamten Seite, der Elemente und der Einzelelemente des Grids.



Unser Fazit

Eine Session mit John Papa ist immer eine Freude. Besonders erfreulich bei der gesamten Konferenz war der deutliche Stellenwert von XAML. Das vorgestellte Prinzip des Stylings ist natürlich bekannt in jeder Ausführungsstufe. Im Grunde ist es eine Fortführung des Metro-Themes aus Windows Phone 7. Was dabei für uns interessant war, dass die Controlsstruktur in künftiger Windowssoftware der Struktur von Silverlight entspricht und nicht der von WPF, wie man eigentlich vermutet hätte. Klare Tendenzen also. Neu und erfreulich waren für uns allerdings die Build-In integrierten Animationen. Das ist sehr schick und erspart viel Arbeit.

HTML-Apps mit Blend

Styling HTML mit Blend

Mal sehen, was Blend auf HTML-Seite so kann.

Einfaches CSS-Editing mit Live-Umsetzung? Das UI kommt sehr stark rüber wie Blend für XAML: Projekt, Styles, Attributes, CSS-Properties sind die entsprechenden Panels. Das Ändern von entsprechenden CSS-Eigenschaften verhält sich wie in XAML: Anklicken auf der Stage und Ändern der Eigenschaften. OK, ein guter CSS-Editor anscheinend.

Aber man verspricht noch mehr. Wirkliches Customizen von Controls am Beispiel einer Checkbox. Sprich neue visuelle Elemente hinzufügen. Das Konzept ist hier das Integrieren von SVG-Bildern. Die Advanced Fassung ist das Einfügen von unterschiedlichen Media-Elementen abhängig vom Monitor-Modus. Auch das wird in einem CSS geregelt. Hier wird Transparenzfüllungen als Trick gespielt im Default-Modus, damit je nach Monitor-Modus das entsprechend andere Media-Element eingesetzt wird. Mit CSS können SVGs auch umgefärbt werden, weil es Pfade sind. Der Pfad eine ID erhalten und entsprechen angesprochen werden. Das stellt eine andere interessante Alternative dar.

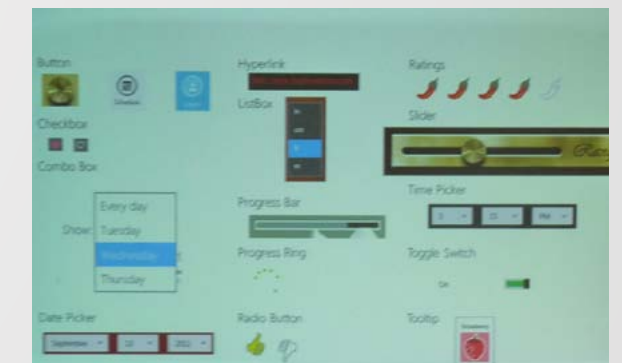
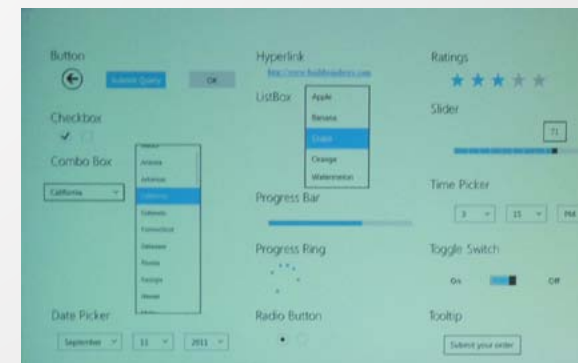
Animationen in HTML-Metro-Apps

Animationsmöglichkeiten in HTML5 über JavaScript sind vielfältig. In Metro-Style-Apps kann die WIN-JS-Library-App genutzt werden. Der Unterschied zwischen Metro-Style-Apps und HTML-Website ist insbesondere die Bewegung durch die Seite. Im App-Fall bleibt man auf einer Seite, im Websitefall bewegt man sich von Seite zu Seite.

Animationen reichern Applikation an, lassen sie modern und emotional ansprechend werden. In Metro-Style-Apps stehen dafür die WIN-JS-Libraries zur Verfügung. Da hier Editierung der Animations-JS-Dateien gezeigt wird, werde ich neugierig, ob ich das auch in Blend kann und schau mal auf dem neuen Tablet nach, wo die BlendBeta für Metro-Apps vorinstalliert ist. Und ich bin begeistert: Alle notwendigen JS-Dateien sind implementiert und ich kann sie on top editieren. Animationen für Metro-Style-Apps bekommt man mit den Libraries for free. Beliebte Animationen in Metrostyle in HTML und XAML: Cross-Fade, Fadeln-FadeOut, enterPage, transitionPage, transitionContent, expand/collapse, Add/Delete

Metro-Style-Apps für HTML in Blend

Der deutsche Schormann ist Teil des Expression Team von MS Corp. Am Beispiel eines Games wird er uns html in Blend zeigen. Historisch betrachtet ist HTML die eine Beschreibungssprache für UIs. Da UIs visuell ist, ist eine visuelle Bearbeitung logisch und notwendig. Blend versteht sich auch als professionelles Designtool, weil man direkt visuell arbeiten kann. Ebenso ist es eine gute Schnittstelle zu den Entwicklern. Obschon Blend erst für XAML da war, ist Blend für HTML logisch und das optimale Tool. Der große Teil von Blend ist das Bearbeiten der CSS-Styles sowie des HTML-Layouts. Blend hat durch den Interaktive Mode aber auch die direkte Bearbeitung im Build-Modus möglich gemacht. Blend wird in Visual Studio eingebunden, so dass das Workflow besser sein wird. Wir sind noch im Preview-Modus. Die Blend-XAML-Version kommt später.

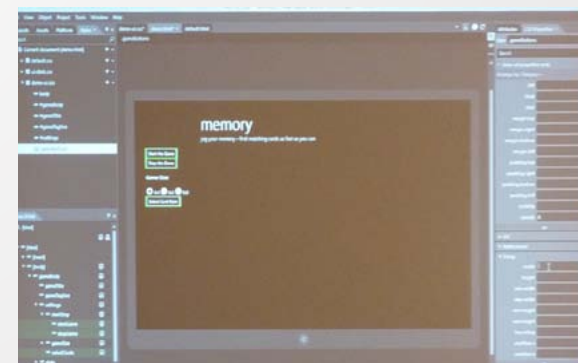
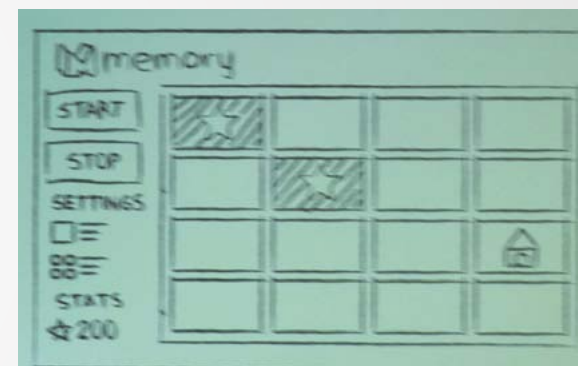


HTML 5 in Blend

Dann geht es in die Praxis. Am Beispiel des Memory-Spiels zeigt er erst mal das Output in verschiedenen Views wie Fullscreen, Snapscreen und Portrait. Das kann auch alles in Blend angezeigt werden.

Der Weg begann mit einer Skizze. Diese wurde von den Entwicklern roh aufgesetzt als JS-Funktionen und HTML-DivLayer. Die Skizze ist kommentiert worden vom Entwickler. Der Entwickler hat ein auf den ersten ein komplett ungestaltetes Layout geliefert, ohne Struktur, Design ... nur reduziert auf Inhalte. Aber man kann mit diesen Inhalten schon interagieren. Durch Anwendung des Metrostyles-CSS-Datei sind sofort alle UI-Elemente per Default umgestylt. Für das spezielle Design wird ein neues Stylesheet hinzugefügt. Es gibt also die default-Styles, die Metro-Styles und die eigenen app-spezifischen. Zum Umlayouten werden die Gameelemente in ein neues Layoutelement gepackt – in ein Layoutgrid. Diese kann ich dann genauso editieren, wie ich es aus Blend kenne. Das Layout der Elemente passiert wie bekannt mit Zuordnungen in das Grid, mit Margins, mit Farben, Schriftgrößen ... Um Elemente per Style umzudesignen lege ich ähnlich wie im XAML einen neuen Style an. Sehr viel kommt mir bekannt vor. Auch hier mache ich eine neue CSS-Klasse. Diese kann ich super easy editieren: Anklicken, sehen wo ich bin, Eigenschaften live einstellen. Feine Sache.

Die Javascript Funktionen kann ich über den Namen zugewiesen, so dass ich alles auch funktional angezeigt wird. Es gibt eine schicke neue Sache: Anzeige von VisualTreeElemente und gemeinschaftliches Editieren. Die perfekte Umsetzung wird im interaktive-Modus angezeigt. Über die Assets kann ich natürlich noch weitere Elemente hinzufügen wie z. B. Bilder. Im Abgleich mit den anderen ViewModus (Portrait) ist das Layout nicht mehr so toll. Das kann man mit entsprechenden Style-Style bzw. ContentDataTemplates abgefangen werden. Abhängig von der Media-Query, werden unterschiedliche Styles angesteuert. Die integrierten HTML-Controls verhalten sich ebenfalls ähnlich wie die von XAML. Sehr cool !



Unser Fazit

Blend als WYSIWYG-Editor für HTML5-basierende Metro-Apps ist von der User Experience vollkommen identisch zu Blend für XAML. Daher fühle ich mich gewissermaßen zu Hause. Im Vergleich zum "echten" Blend, fehlen zwar noch ein noch ein paar entscheidende Killerfeatures, aber die Hoffnung dass diese implementiert werden haben wir. Die Umsetzung von Blend als WYSIWYG-Editor ist die Umsetzung des Naheliegendsten, da HTML und XAML schließlich beide Oberflächen-Beschreibungssprachen sind. Witzigerweise ist HTML die banaler als XAML ... HTML5 werden wir vielleicht dank der Erweiterung von Expression Blend und unserer Erfahrung als "Blender" sowie als Webber der ersten Stunden in unseren Portfolio aufnehmen, wenn es sich als sinnvoll zeigen wird.

Blend in Visual Studio 2011

Als Blendkenner sehen wir schon auf dem ersten Blick: Das Beste von Blend wird in Visual Studio verfügbar sein. Der helle Look von Visual Studio bleibt allerdings erhalten.

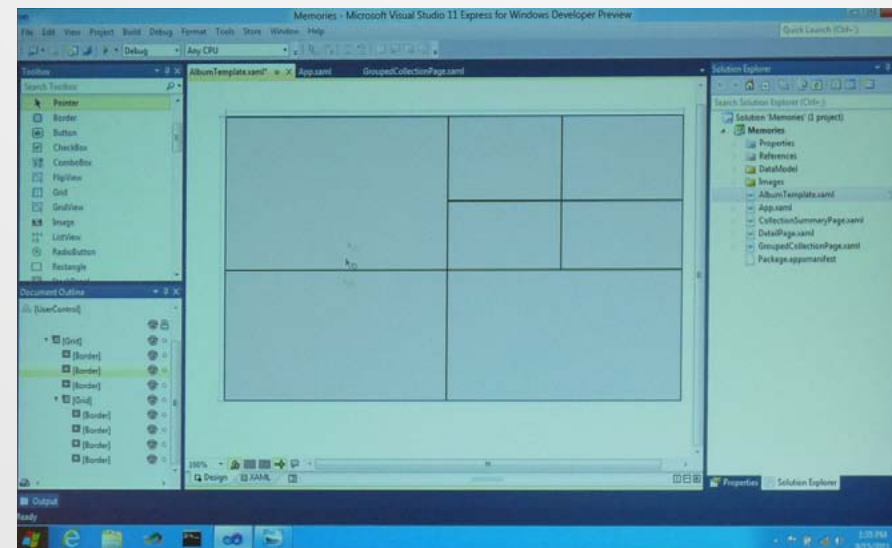
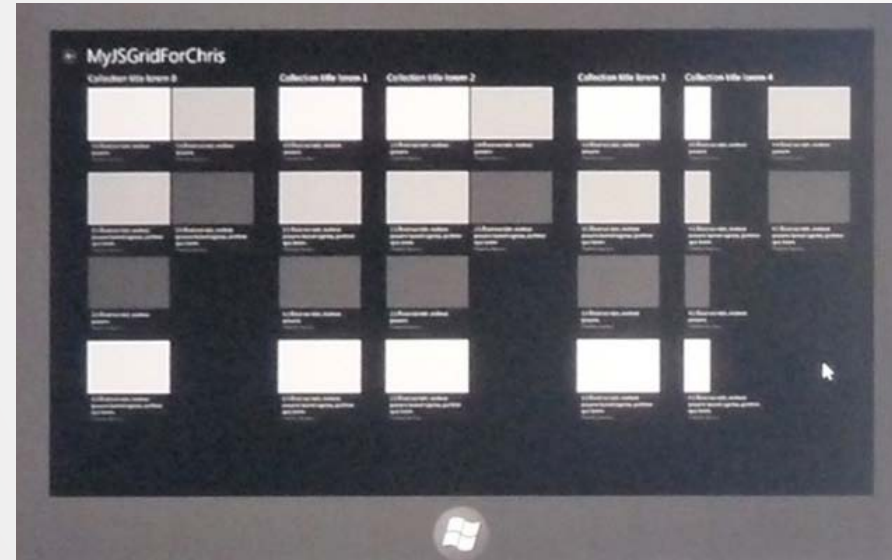
Blendangelehnte Killerfeatures:

- Split-Modus
- Zooming innerhalb der Stage
- Auswahl der Objekte auf der Stage
- Brush-Editor
- Panel-Verhalten
- Objekt-Layer als visuelle Umsetzung der XAML-Layer
- Property-Panel im Stil von Blend
- Reset-Punkt für Properties
- Visuelles Masken-Editing
- Ressourcen-Dialog
- Xaml-Tag-Assets

Feature-Erweiterung im Vergleich zu Blend:

- Template-Preview
- Grid-Einstellungen als Contextmenu.
- Drei Ansichten des Property-editor nach Source, Name, Kategorie
- Verbessertes Style-Editing über Contextmenus
- Sehr schöner Grid-Editor

Das Visual Studio 2011 PreView ist verfügbar, aber sollte zügig mit einem Update versorgt werden, weil es noch ziemlich buggy ist.



Unser Fazit

Nun ein kleines Fazit zu dem Gesehenen: In Visual Studio und Blend werden die Sachen zusammengezogen, die zusammen gehören, ohne die Programme zusammen zu führen. Das birgt den Vorteil, dass man nun nicht mehr wechseln muss, um kleine Änderungen durchzuführen als Entwickler. Designer werden sich in Blend bewegen, Entwickler im Visual Studio. Die vielen Möglichkeiten, nun visuell zu arbeiten, bergen zwei Gefahren: Designer halten sich noch mehr aus dem XAML-Umsetzungspart raus, was das Visual Studio die Tätigkeit nun Entwicklern indirekt zuweist. Entwickler fühlen sich nun als Designer berufen, weil ihnen nun die WYSIWYG-Mittel zur Verfügung stehen (die ihnen durch Blend eh zur Verfügung standen und ungenutzt bleiben). Es steht zu befürchten, dass die nun für Windows 8 sehr gut zusammenarbeiten sollten, es doch nicht tun. Da wir hier aber auf einer Entwickler-Konferenz sind, finde ich diese Änderungen an Visual Studio hervorragend sowie auch die Tools, die den Entwicklern zur Umsetzung toller Metro-Apps gegeben werden.

Bereit für die Zukunft

Bereit für Windows 8

Das auf der BUILD Gezeigte hat uns überzeugt. Wir sind bereit uns begeistert auf Windows 8, Metro Style Apps sowie die technologischen Änderungen einzulassen. Die Metro-Design-Ausrichtung, die Verstärkung der Rolle von XAML im User Interface, der App-Marktplatz sowie die Erweiterungen im Tooling passen bestens zu unseren Kernkompetenzen Design und Development von User Interfaces mit XAML. Die befürchteten strategischen Änderungen werden nicht notwendig, sondern vielmehr eine Verstärkung dessen, was wir begonnen haben mit XamlLab und XamlFab.

XAML und C# passen nun mehr denn je zu modernen Software-Applikationen - sei es für Desktop-Software mit WPF, im Web mit Silverlight, für Mobile Apps mit Windows Phone 7 oder für das künftige Windows 8. Den Metrostyle kennen und beherrschen wir schon seit Einführung mit Windows Phone 7 im Frühling 2010. Die neuen Möglichkeiten von Windows 8 bieten weiterhin gute Einsatzmöglichkeiten von XAML. Dass wir uns seit 2006 mit XAML-orientierter Software-Entwicklung zu modernen Oberflächensprachen bewegt haben, bestätigt sich heute mehr als richtig. Daher sind die nächsten Schritte die logische Fortsetzungsgeschichte.

Bereit für Expansion

Die immer notwendige Team-Arbeit zwischen Designern und Entwicklern ist bei uns schon lange erfolgreich erprobter Status Quo, da wir seit 2003 als Team zusammen arbeiten. Mit dem Rückenwind der erfolgreichen Selbständigkeit der letzten zwei Jahre sowie den Ausblicken auf die richtige strategische Orientierung durch die BUILD Windows 2011 gehen wir nun den nächsten folgerichtigen Schritt und werden in naher Zukunft die Fusion von XamlFab und XamlLab zu einer gemeinsamen GmbH vorantreiben. Neben dem formalen Rahmen werden wir uns der Anforderung entsprechend vergrößern und in Mitarbeiter sowie deren Ausbildung investieren.

Trotz Expansion und Professionalisierung halten wir an unseren Werten fest. Das bedeutet für Sie als Kunden, dass die Honorare fair bleiben, Sie das Recht an den Rohdaten behalten, weiterhin transparentes stundenbezogenes Arbeiten besteht sowie anderer Vorteile unserer kleinen flexiblen Struktur. Andererseits werden Sie die Vorteile eines wachsenden professionalisierten Unternehmens nutzen dürfen durch transparentes Erfolgstracking, Entwicklung qualitätsgesicherter Prozesse, Usability-Expertise inklusive, Spezialisierung in mehreren Bereichen, zertifiziertes Projektmanagement, mehrere Microsoft-Zertifizierungen sowie einer stetigen Sicherstellung einer gesunden und verantwortungsvollen Unternehmensentwicklung.

Bereit für Gemeinsames

Ob Sie nun eine gewachsene Software in Design und Entwicklung mittels WPF zu einer modernen Software weiterentwickeln, Ihre Applikationen mit Silverlight in Web bringen oder ob Sie sich in die mobile Welt mit einer Windows Phone 7 Applikation bewegen wollen. In allen Fällen sind Sie bei uns richtig. Mit diesen Technologien sind Sie für die Zukunft gerüstet.

Sollten Sie mit uns gemeinsam allerdings jetzt schon den Sprung in die Zukunft in Form von Windows 8 Apps wagen wollen, scheuen Sie nicht uns anzusprechen. Wir möchten mit Ihnen gemeinsam in einem fairen Modell, Ihre Windows8-App-Ideen umsetzen.

Wir werden in Windows 8 und Metro-Style-Apps investieren und freuen uns darauf, mit Ihnen gemeinsam Schritte in die Zukunft zu machen. Besonders unseren bestehenden Kunden, Partnern und Kontakten möchten wir den Vorteil unseres neuen Knowhows zum Vorzug anbieten.

Wir freuen auch in Zukunft mit Ihnen spannende Projekte zu realisieren.