

# Development voor Windows Phone 7

FOCUS LIGT DUIDELIJK OP DEVELOPERSCOMMUNITY

**Mark Monster**

“Our vision is to bring Silverlight to every screen.” In 2007 kwam Silverlight voor het eerst op de computerschermen. 2010 is het jaar van Silverlight op de telefoon. Tijdens de MIX conferentie is Windows Phone 7 aangekondigd, een ommekeer van het mobiele platform van Microsoft. In ieder geval ligt de focus duidelijk op het opbouwen van een developerscommunity. Lang voordat de eerste toestellen in de winkels liggen, zijn er previewversies van de ontwikkelomgeving beschikbaar. Deze ontwikkelomgeving is gebaseerd op Silverlight en XNA.

Wie ooit gewerkt heeft met de voorgangers van Windows Phone 7 weet dat het erg lastig is om een applicatie te bouwen die overal werkt. Nu heeft Microsoft nooit de intentie gehad om zelf telefoons te verkopen, want dat ging via de diverse hardwarefabrikanten. Deze hardwarefabrikanten waren nogal creatief in de diversiteit van ontwerpen. Geen touchscreen of toch wel? Of misschien wel multitouch? Allemaal vragen waar je rekening mee moest houden als ontwikkelaar voor Windows Mobile 6.x applicaties. Nu heb ik het alleen nog maar over de mogelijkheid een scherm wel of niet aan te kunnen raken, dan hebben we nog verschillende resoluties en ook nog eens in verschillende verhoudingen. Als software-ontwikkelaar vind ik dat er eigenlijk te veel flexibiliteit voor de fabrikanten van hardware was. Deze flexibiliteit heeft ons als ontwikkelaars niet echt geholpen, want in plaats van effort te stoppen in interessante functionaliteit waren we altijd veel tijd kwijt om een applicatie overal te laten werken.

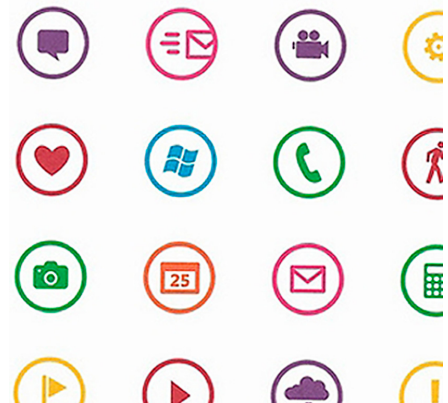
Tijdens zijn veranderd, Windows Phone 7 is aangekondigd. De regels voor de hardwarefabrikanten zijn heel strikt. Sterker nog, de drivers worden door Microsoft zelf geschreven. De minimale hardware-eisen helpen de ontwikkelaars enorm. Om er maar een paar te noemen: GPS, Accelerometer, vierpunsmultitouch. Maar om te

beginnen is bij de lancering van Windows Phone 7 de resolutie 480 bij 800. Later zal ook 320 bij 480 beschikbaar komen. Door de beperkte variatie aan hardware kunnen de ontwikkelaars zich focussen op functionaliteit en experience.

Andere platformen bieden een erg groot scala aan applicaties en games – soms wel 100.000 – die mede hebben bijgedragen aan het succes van de bijbehorende telefoons. Microsoft wil dit natuurlijk graag evenaren, met ons als ontwikkelaars. Ruim voordat de hardware beschikbaar is, heeft Microsoft al gratis previewversies van de ontwikkelomgeving beschikbaar gesteld. Sterker nog, ook na de finale release zal deze ontwikkelomgeving geheel gratis zijn. Verder is er een idee-uitwisselingsprogramma gestart met de naam Mobile App Match. Hier kunnen mensen ideeën voor applicaties plaatsen, ontwikkelaars kunnen deze dan weer oppakken. Een mooi initiatief dat helpt om een enorme marktplaats aan applicaties te creëren.

Op dit moment zijn er vijf fabrikanten bekend die hardware leveren met Windows Phone 7: Asus, Dell, HTC, LG en Samsung. De verschillen die we zien tussen de handsets zitten hem niet in de software, maar in de looks van de hardware. Sommige handsets zijn voorzien van een fysiek toetsenbord, maar ook het softwarematige toetsenbord werkt prima.

Metro: Content, Not Chrome.



Windows, is Windows niet meer. Althans, Windows Mobile 6.x is op zichzelf niet erg fingertouch vriendelijk. Waar de scrollbar echt alleen met een pennetje te bedienen is en het startmenu wel erg veel lijkt op het startmenu van Windows 95 is het dus niet echt geschikt voor kleine schermen. Sinds we gezien hebben hoeveel gebruiksgemak er in andere platformen zit moest het drastisch anders. Het antwoord is: METRO.

Helemaal nieuw is Metro niet, het is namelijk afgeleid van de Zune. De Zune is overigens hier in Europa minder bekend, omdat deze alleen in de Verenigde Staten is uitgekomen. Niet getreurd, de software van de Zune zit ook in Windows Phone 7. Metro is de ontwerptaal voor Windows

Phone7. Waar we in het verleden graag mooie gradiënts zagen, zien we nu simpele duidelijke iconen, vergelijkbaar met wat we zien op stations en vliegvelden. Door tekst uit het scherm te laten lopen, wordt de indruk gewekt dat het scherm groter is. Dit is in mijn optiek sterker dan een horizontale scrollbar. Uiteindelijk gaat het veel meer over de inhoud. Afleidende scherm-elementen die geen inhoud zijn moeten dan ook worden vermeden.

## Silverlight

De vorige generatie ondersteunde al .NET, al was het niet het volledige Framework. Het Compact Framework mocht er best zijn, maar grafische hoogstandjes waar we tegenwoordig zo aan gewend zijn waren moeilijk te creëren. Sterker nog, sommige hardware was niet krachtig genoeg voor hoogstandjes. Op het web hebben we steeds vaker te maken met prachtige Silverlight-applicaties, een erg mooi platform en dat heeft Microsoft na veel aankondigingen en uitstellingen naar het mobiele platform gebracht.

Op het web hebben we te maken met Silverlight 4. Op de Windows Phone is dit echter een groot deel van Silverlight 3 met eigen toevoegingen. Vooral belangrijk is de mogelijkheid om met XAML de User Interface op te bouwen. De meeste .NET-developers zullen wel eens de kracht van XAML hebben gezien, het gaat te ver voor dit artikel om hierop in detail in te gaan, maar de meest gebruikte onderdelen van XAML in Silverlight 3 kunnen ook op de Windows Phone worden gebruikt. Natuurlijk is de styling van de schermonderdelen aangepast aan de afmetingen van het apparaat en de principes van Metro. Wat voor WPF en het 'gewone' Silverlight geldt, is dat er kan worden ontwikkeld binnen Visual Studio, maar de echte grafische hoogstandjes worden gemaakt in Expression Blend. Dit gaat ook op voor Windows Phone 7 applicatie-ontwikkeling, de vertrouwde ontwikkelomgeving met daaraan toegevoegd een emulator om de applicaties binnen een virtuele Windows Phone 7 te testen. Ook zijn er twee belangrijke schermelementen aangekondigd. Tot op de dag van vandaag zijn die nog niet officieel beschikbaar, maar wel als open source alternatief. Ik heb het dan over de Panorama en de Pivot. Deze controls worden door Microsoft beschikbaar gesteld in de uiteindelijke versies van de ontwikkeltools. De Pivot kun je het beste zien als een soort tabcontrol. Met name de navigatie is wat

aangepast aan de eisen van een mobiele telefoon. Dit type control kom je regelmatig tegen in applicaties die standaard op de Windows Phone geïnstalleerd staan, zoals de calendar en de schermen voor de verschillende telefooninstellingen.



AFBEELDING 1: PIVOT VOORBEELD WAARBIJ SEATTLE, PARIS EN LONDON DE TABS VOORSTELLEN.

De Panorama is een schuifbaar passe-partout dat over een breed beeld (niet te verwarren met een breedbeeldscherm) kan worden geschoven. Dit concept kom je veel tegen binnen Windows Phone 7, aangezien alle Hubs hierop zijn gebaseerd. Hierbij gaat het om People Hub, Pictures Hub, Office Hub, Games Hub, Music + Video Hub en Marketplace Hub.



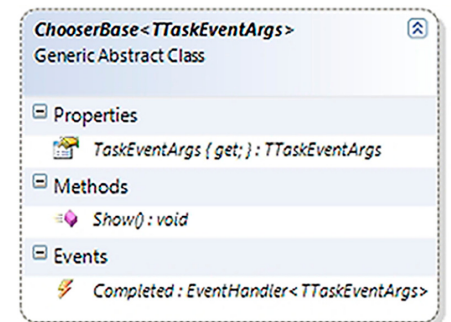
AFBEELDING 2: PANORAMA WAAR JE NAAR WENS VAN LINKS NAAR RECHTS KUNT SCHUIVEN OM ANDERE CONTENT TE ZIEN.

## XNA

.NET-developers die ook nog eens gameliefhebbers zijn, hebben thuis een Xbox staan en op hun laptop de XNA Game Studio. Nu is het natuurlijk niet eenvoudig om op de Xbox met een game te komen die erg succesvol is, maar op een telefoon zijn de eisen toch anders. Want een game die je 15 minuten lang vermaakt op een telefoon is al leuk. Hoewel het natuurlijk ook goed mogelijk is om een game te maken in Silverlight, zullen bepaalde typen games meer tot hun recht komen in XNA. Ook hier zijn we geholpen met de mogelijkheid om Windows Phone 7 games te maken in XNA.

## Phone Integratie

Integratie met hardware levert nogal eens problemen op, maar ook integratie met specifieke onderdelen van het besturingssysteem. In het verleden was het gebruik van unmanaged code onvermijdelijk. Het mooie is, dat kan niet meer. Althans, het is slechts voor een beperkt aantal ISVs weggelegd gebruik te maken van unmanaged code. Nu zal het voor ontwikkelaars die het open Windows Mobile platform kennen even wennen zijn. Sommige integraties zullen niet mogelijk zijn, maar hebben een alternatieve aanpak gekregen: Tasks en Sensor API. Bijna alle tasks kun je overerven van ChooserBase en bevatten daardoor eigenlijk dezelfde structuur. Om de taken te starten gebruik je de Show methode. Verder is er het Completed event dat voor iedere taak beschikbaar is, echter de EventArgs hebben verschillende overervingen. Zo is er de PhotoResult waar een referentie naar een foto in zit verwerkt. Deze wordt door de CameraCaptureTask en de PhotoChooserTask gebruikt.



AFBEELDING 3: REPRESENTATIE VAN DE ABSTRACTE CHOOSEBASE CLASS.

Ieder Windows Phone 7 apparaat zit vol met sensoren. GPS en Accelerometer zijn standaard, evenals het eerder genoemde vierpuntstouchscherm. Verder zitten er een lichtsensoren, 'benaderingssensoren' en kompas in. Voor deze laatstgenoemde sensoren is geen API beschikbaar, althans nog niet. Voorlopig hebben we wel met een aantal beperkingen te maken, omdat we onze ontwikkelomgeving nog niet kunnen completeren met een echte Windows Phone 7. Voorlopig moeten we het doen met de emulator die wordt geïnstalleerd met de Windows Phone 7 Developer. Deze emulator ondersteunt prima multitouchschermen. Als je al een multitouch scherm hebt zullen de aanraakpunten worden doorgegeven aan de emulator. De GPS en Accelerometer moeten worden vervangen door een nepimplementatie zolang er geen apparaat is om te testen. Voor de GPS is

Taak	Omschrijving	Completed Event Type
CameraCaptureTask	Deze taak kan gebruikt worden om de gebruiker een foto te laten maken.	PhotoResult
EmailAddressChooserTask	Lanceert de contactpersonenapplicatie en retourneert het e-mailadres van de geselecteerde contactpersoon.	EmailResult
EmailComposeTask	Lanceert de e-mailapplicatie met een nieuw e-mailbericht.	Geen
MarketplaceDetailTask	Lanceert de Marketplace applicatie en toont de detailinformatie pagina voor het opgegeven product	Geen
MarketplaceHubTask	Lanceert de Marketplace applicatie	Geen
MarketplaceReviewTask	Lanceert de Marketplace applicatie en toont de review pagina voor het opgegeven product.	Geen
MarketplaceSearchTask	Lanceert de Marketplace applicatie en toont de zoekresultaten voor de opgegeven zoekwoorden.	Geen
MediaPlayerLauncher	Lanceert de media player.	Geen
PhoneCallTask	Lanceert de telefoniefunctie voor het maken van een telefoonverbinding.	Geen
PhoneNumberChooserTask	Lanceert de contactpersonenapplicatie en retourneert het telefoonnummer van de geselecteerde contactpersoon.	PhoneNumberResult
PhotoChooserTask	Deze taak kan gebruikt worden om de gebruiker een foto te laten selecteren.	PhotoResult
SaveEmailAddressTask	Slaat een e-mailadres op, de contactpersonen applicatie wordt gestart en een nieuwe of bestaande contactpersoon kan worden geselecteerd.	TaskEventArgs
SavePhoneNumberTask	Slaat een telefoonnummer op, de contactpersonenapplicatie wordt gestart en een nieuwe of bestaande contactpersoon kan worden geselecteerd.	TaskEventArgs
SearchTask	Start een zoekopdracht op het web.	Geen
SmsComposeTask	Lanceert de sms-applicatie met een nieuw sms-bericht.	Geen
WebBrowserTask	Lanceert de webbrowser.	Geen

TABEL 1: OVERZICHT VAN TAKEN, HUN DOEL EN HET COMPLETED EVENT TYPE.

er de IGeoPosition Watcher-interface die kan worden geïmplementeerd. De Accelerometer is wat lastiger, omdat hier geen interface voor beschikbaar is, maar een proxy kan hier helpen. Het is overigens verstandig om voor elke hardware-component waarmee wordt gecommuniceerd een proxy te schrijven. Naast de bepaling van positie op basis van GPS kan ook een positie worden bepaald op basis van GSM-masten. De positiebepaling op basis van GSM-masten verbruikt minder stroom en is dus aan te raden ten opzichte van GPS. De positiebepaling is dan wel minder secuur, maar lang niet altijd is het belangrijk om de exacte positie in een straat te weten en voldoet de plaatsnaam.

## Marketplace

Zoals bij meerdere platformen worden applicaties voor Windows Phone 7 geïnstalleerd via de Marketplace. Het is een ideaal platform om applicaties en games te verkopen. Van de opbrengst gaat slechts 30% naar Microsoft.

Voordat een applicatie gepubliceerd wordt in de Marketplace wordt deze getoetst aan de Application Policy Guidelines. Microsoft is open over deze regels van acceptatie. Dat is heel prettig, want hierdoor kun je vooraf al een self-test doen. Het is mooi dat het voor de uitgevers zeer eenvoudig is gemaakt om van een applicatie een probeerversie uit te brengen. Door gebruik te maken van de Marketplace API kun je in


één regel code controleren of de applicatie als probeerversie draait.

In de Microsoft.Phone.Marketplace namespace vinden we de LicenseInformation class, deze bevat slechts één operatie: IsTrial. Dit betekent dat de simpele code zoals hieronder mogelijk is.

```
LicenseInformation licenseInformation =
new LicenseInformation();
if(licenseInformation.IsTrial())
{
//Probeerversie, dus reclame.
}
```

Hoe de applicatie in de probeerversie reageert, is alleen gelimiteerd aan de eigen verbeelding. Denk bijvoorbeeld aan: reclame, opslaan niet mogelijk, maximaal 10x starten. Houd er rekening mee dat in de ontwikkelomgeving deze methode altijd 'true' retourneert. Er is een oplossing mogelijk op basis van Conditional Compilation Symbols (zie referenties).

## Enthousiast?

Direct na de aankondiging dat Windows Phone 7 applicaties op het Silverlight platform komen te draaien, werd ik erg enthousiast. Nu we een release verder zijn – de beta – sta ik te trappelen om een echte Windows Phone 7 in mijn handen te krijgen. Daar moeten we nog even op wachten, maar kijk alvast op de Windows Phone 7 Developer Portal. 

## Links

- Mobile App Match: <http://www.mobileappmatch.com/>
- Windows Phone 7 Panorama & Pivot controls: <http://phone.codeplex.com/>
- Marketplace Application Policy Guidelines: <http://go.microsoft.com/?linkid=9730556>
- How to improve the Windows Phone 7 Licensing development experience: <http://mark.mymonster.nl/2010/06/11/how-to-improve-the-windows-phone-7-licensing-development-experience/>
- Windows Phone 7 Developer Portal: <http://developer.windowsphone.com/>

.....  
**Mark Monster**, is Software Engineer/Software Architect bij Rubicon en houdt zich bezig met custom development. Dit jaar heeft Mark een Silverlight MVP award gekregen voor zijn inzet in de Silverlight community, online en als vice-voorzitter in de Silverlight & Expression Insiders User Group (<http://sixin.nl/>). Hij is te bereiken via [mark@mymonster.nl](mailto:mark@mymonster.nl) of via zijn blog op <http://mark.mymonster.nl/>.

