

Webcontentmanagement in Microsoft Office SharePoint Server 2007

EEN INTRODUCTIE VAN HET NIEUWE CONTENTMANAGEMENTSYSTEEM

Afgelopen mei kwam Microsoft met de bètaversie 2 van Office SharePoint Server 2007. Voor veel ontwikkelaars die zich bezighouden met Content Management Server 2002 is dit de langverwachte release, omdat hierin functionaliteit voor webcontentmanagement is geïmplementeerd. Met dit artikel wil ik een eerste indruk geven hoe je websites kunt ontwikkelen met Office SharePoint Server 2007, zodat je hopelijk zelf gaat experimenteren met webcontentmanagement.

Microsoft heeft besloten om de webcontentmanagement- en portal & collaboration-technologie samen te voegen tot één product: Microsoft Office SharePoint Server 2007. Het is dus niet alleen de opvolger van SharePoint Portal Server 2003 (SPS), maar ook van Content Management Server 2002 (CMS). Daarnaast zijn er talloze verbeteringen opgenomen en biedt het veel nieuwe functionaliteiten, te veel om in dit artikel te beschrijven. Een paar van de belangrijkste verbeterpunten voor webcontentmanagement (wcm) zullen in dit artikel aan bod komen: workflow, versioning, site variations en search. SharePoint is nu, met documentmanagement, recordsmanagement, formsmanagement en dus ook webcontentmanagementfunctionaliteit, nog meer een information life-cycle managementproduct geworden. Voor CMS-ontwikkelaars lijkt er in eerste instantie veel te veranderen. Toch zie je veel concepten van Content Management Server terugkomen in webcontentmanagement. Zo creëert een contentmanager nog steeds webpagina's, maar heten deze nu geen postings maar pages en een template is een page lay-out. Tabel 1 geeft een overzicht van de vergelijking van de terminologie en functionaliteiten binnen Content Management Server 2002 en webcontentmanagement. Voordat je een website gaat maken met Office SharePoint Server 2007 is het goed je te realiseren dat een website een collectie van sub-websites is. De hiërarchie die hiermee ontstaat, wordt gebruikt voor de navigatie en security (afbeelding 1). Elke website heeft een documentlibrary om de 'pages' in op te slaan. Dit zijn jouw webpagina's binnen jouw (sub)site (ofwel in CMS-analogie; jouw postings binnen een channel). Een nieuwe webcontentmanagement-site

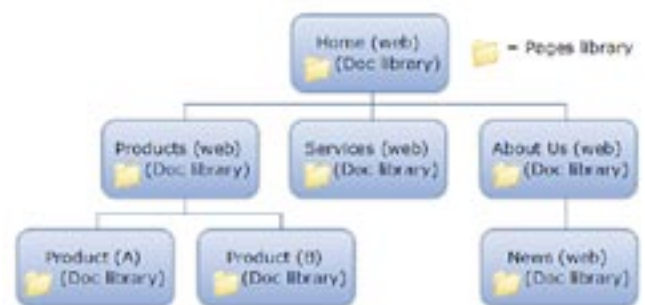
Microsoft Content Management Server 2002	Microsoft Office SharePoint Server 2007
Template (aspx page)	Page Lay-out
Template Definition	Content Type
Placeholder	Field Control
Placeholder Definition	Column Templates
Posting	Page
Channel	Site
Top Level Channel	Site Collection
Users, Roles, Rights Groups	Users, Roles, Groups, Permissions
Resources	Items in Documents Libraries
Custom placeholder data	Gemigreerd als tekst velden

Tabel 1. Vergelijking van de terminologie en functionaliteiten

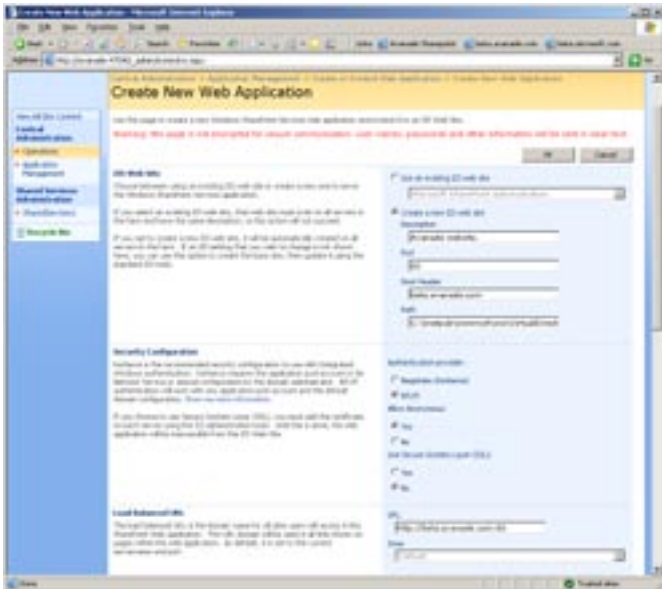
starten. Een nieuwe website maak je aan in SharePoint 3.0 Central Administration. De tab 'Application Management' toont de settings om (web)applicaties en componenten op de server aan te maken en te configureren. Onder het kopje 'SharePoint Web Application Management' klikken we op 'Create or extend Web Application' om een nieuwe virtuele website in Internet Information Services (IIS) aan te maken. Vanaf Windows SharePoint Services 3.0 (WSS) noemt Microsoft een virtuele IIS-website een webapplicatie. We geven hier aan dat we een nieuwe IIS-website ('Create a new IIS web site') willen maken: 'beta.avanade.com' en dat deze communiceert over poort 80. Zoals afbeelding 2 toont zijn er daarnaast nog verschillende andere settings te configureren, zoals security settings (anonieme toegang en beveiligde verbinding), application-pool settings en database-settings. Na op 'Ok' te klikken wordt een nieuwe SharePoint Services-webapplicatie in IIS aangemaakt. De Central Administrator vraagt, nadat de webapplicatie is aangemaakt, of we een site-collection hiervoor willen aanmaken.

Site Collection

Een site-collection is onze eigenlijke website. We gaan een nieuwe top-level site-collection aanmaken. Zorg dat je eerst IIS herstart, omdat anders de nieuwe IIS-site nog niet beschikbaar is. We creëren vervolgens deze (root) site-collectie voor onze zojuist aangemaakte webapplicatie (http://beta.avanade.com). Na het instellen van de site-administrator kiezen we als laatste stap de template, aan de hand waarvan onze site-collection wordt aangemaakt. Omdat we een website gaan maken, kiezen we voor de 'Internet Presence Web Site'-template (afbeelding 3). Deze template is ingericht om een website voor het internet te maken. Dit betekent



Afbeelding 1. Een typische website-structuur.



Afbeelding 2. Aanmaken van een nieuwe SharePoint Services webapplicatie



Afbeelding 4. Een webpaginastructuur in Office SharePoint Server 2007

onder andere dat er al een initiële site-structuur is opgezet en dat we een set van page lay-outs tot onze beschikking hebben die specifiek zijn ontwikkeld voor het maken van webpagina's.

Webpagina's

Content Management Server 2002 draait op IIS en slaat zijn data op in een SQL Server-database. Via de Internet Server Application Programming Interface (ISAPI) worden de requests voor Content Management Server eruit gefilterd, zodat deze kunnen worden afgehandeld. Content Management Server 2002 gebruikt vervolgens zijn Publishing API om de data op te halen uit de database. SharePoint Server 2007 is als een 'native' ASP.NET 2.0- applicatie gebouwd op SharePoint Services versie 3 (WSS). De content wordt ook hier opgeslagen in SQL Server, maar nu wordt gebruik gemaakt van de SharePoint lists-infrastructuur om webpagina's, resources en templates op te slaan. Omdat SharePoint Server als een ASP.NET 2.0 applicatie draait, is het niet langer een extensie bovenop IIS. Daarom handelt ASP.NET de request voor een webcontentmanagement-webpagina af. De rendering van webpagina is hierdoor anders.

Voordat ik inga op hoe een webpagina in SharePoint Server 2007 wordt gerenderd, is het goed om het concept van een webpagina in SharePoint Server te begrijpen. Afbeelding 4 toont een typische webpagina binnen webcontentmanagement. De masterpage is oranje gemarkeerd. Daarbinnen laden we onze page lay-out met daarop de field controls waarin we onze dynamisch content (tekst, plaatjes, enzovoort) plaatsen. Een webpagina is gebaseerd op een page lay-out. Deze opzet resulteert in de volgende afhandeling van een webpagina:

1. Een browser vraagt een webpagina van IIS.
2. IIS geeft deze request door aan ASP.NET 2.0
3. ASP.NET 2.0 haalt de webpagina via een WSS-fileprovider op.
4. De page lay-out wordt geladen.
5. De masterpage wordt geladen.
6. De field controls op de pagina renderen de content in de pagina.
7. IIS stuurt de pagina terug naar de browser.



Afbeelding 3. Aanmaken van een nieuwe site-collection aan de hand van het Internet Presence Web Site-template.

Masterpages en page layout

Masterpages worden, net als in ASP.NET 2.0, ingezet voor de 'branding' van de site en voor site-navigatie. Op een masterpage vind je meestal de volgende items:

- Navigatie
- Logo's
- Zoeken
- Login-control
- Editing controls
- CSS-referenties

De masterpages zijn te vinden in de 'Master Page en Page Lay-out gallery' van de root site-collection. Met SharePoint Designer (opvolger van Frontpage) zijn deze aan te passen en kunnen er nieuwe masterpages toegevoegd worden. Een page lay-out dient als mal voor een webpagina. Vergelijk dit met de ASPX-pagina's die je aan de template-definitions koppelde in Content Management Server 2002. In een page lay-out refereer je aan een masterpage en daarnaast plaatsen we hierop de controls die in het contentmanagementproces beheerd kunnen worden. Denk hierbij aan de field controls, waarin we content (tekst en illustraties) neerzetten, webparts en webpartzones. Omdat SharePoint Server de dynamische content nu opslaat in de SharePoint lists-infrastructuur hoeven we hier niet meer te werken met template definitions. Net als masterpages, kunnen we ook de page lay-outs aanpassen in SharePoint Designer. Om de page lay-outs aan te passen of om nieuwe page lay-outs aan te maken, openen we de website in SharePoint Designer. In de '_catalogs/masterpage'-folder van de site-collection zijn de masterpages en de page lay-outs van de geopende site te vinden. Codevoorbeeld 1 toont een eenvoudige opzet van een nieuwe page lay-out. In dit voorbeeld staat een ContentPlaceholder (PlaceHolderMain) met daarin een webcontrol van het type PublishingWebControls: RichHtmlField en een webpartzone. De eerste is een field control waarmee we content kunnen toevoegen aan de pagina. Vergelijk dit met de HTML Placeholder-control in Content Management Server 2002. SharePoint Server 2007 regelt zelf hoe deze content opgeslagen wordt in de SharePoint lists-infrastructuur. Met een webpartzone stellen we de contentmanager in staat om webparts toe te voegen aan de pagina.

Web-authoring

Als een contentmanager inlogt, worden de opties om de site te bewerken getoond. Via 'Site Actions' kan het 'Web Author'-menu geopend worden (afbeelding 5). Dit menu toont de status van de pagina. Daar

naast kunnen we met dit menu contentmanagementtaken uitvoeren op de huidige pagina. Zo kunnen we de pagina bewerken, verwijderen, (scheduling-) eigenschappen zetten of een workflow starten. Naast webauthoring bestaat ook de mogelijkheid om via XML-bestanden, InfoPath-formulieren of Word-documentencontent te beheren (dit is in een *pluggable* architectuur opgezet, zodat meer converters beschikbaar komen). Dit wordt Smart Client Authoring (SCA) genoemd. Een in Word geschreven persbericht wordt in een SharePoint Server document-library opgeslagen. Via de optie 'Convert Document > From Word Document to Web Page' wordt er van het persbericht een webpagina aangemaakt. Hierbij geven we aan welke page lay-out we hiervoor willen gebruiken, waar de webpagina aangemaakt moet worden en wat de naam ervan is. In dit voorbeeld gaan we echter een nieuwe webpagina via de Web Author maken. We selecteren via 'Site Actions' de optie 'Create Page'. In de 'New Page'-dialog geven we de titel, omschrijving (optioneel) en de naam van de pagina. Daarnaast kiezen we een page lay-out aan de hand waarvan we de pagina willen aanmaken. In dit geval kiezen we de page lay-out die we zojuist hebben aangemaakt met één content control field en één webpartzone. Vervolgens wordt de pagina (als ASPX-pagina) aangemaakt en kunnen we de pagina van content voorzien. Naast het handmatig invoeren van content, biedt SharePoint Server 2007 een nieuwe feature: reusable content. Reusable content is een lijst met voorgedefinieerde teksten of HTML-fragmenten. Tijdens het aanmaken van een nieuwe pagina kun je zo eenvoudig veelgebruikte content toevoegen, bijvoorbeeld een copyright- of disclaimer-tekst. Daarnaast kunnen we, net als in Content Management Server 2002, de webpagina scheduling-features zetten. We kunnen aangeven wanneer deze pagina gepubliceerd moet worden en wanneer deze weer offline moet. Als dit alles naar volle tevredenheid is gedaan, dien je de pagina in ter goedkeuring. De webpagina wordt dan opgeslagen in de pages document-library van de site. Hierdoor wordt de approval workflow gestart. Nadat de pagina goedgekeurd is voor publicatie krijgt deze versie nummer 1.0. Alle wijzigingen op deze pagina worden bijgehouden via major of minor versies. Via dit versiebeheer kan een webpagina nu ook teruggezet worden naar een eerdere versie. Deze wordt dan als nieuwe (minor) versie gepubliceerd. Daarnaast bestaat nog steeds de mogelijkheid om de (tekstuele) verschillen tussen de verschillende versies te bekijken. Via een gekleurde markering wordt aangegeven wat toegevoegd, wat gewijzigd en/of wat verwijderd is.

Pages document-library

De webpagina's worden als SharePoint-listitems opgeslagen in de pages document-library. Door middel van de page-contenttype wordt een aantal eigenschappen hiervan bepaald. Zo heeft de page content type-standaard een aantal, dat de metadata van een webpagina beschrijft (start- en einddatum publicatie, contactpersoon, page lay-out, enzovoort). Daarnaast zijn hier natuurlijk meer kolommen aan toe te voegen. Hier kan bijvoorbeeld de informatie, die in Content Management Server 2002 in de 'custom properties' van een posting stond, opgeslagen worden.

Workflow

Daarnaast kunnen we in deze document-library de workflow aanpassen. Ga hiervoor naar de 'View All Site Content' (via het 'Site Actions' menu) en kies de 'Pages' Document Library. In 'Settings > Document Library Settings' vinden we onder 'Permissions and Policies' het menu: 'Workflow settings'. Out-of-the-box is de Serial Approval-workflow aan deze document library gekoppeld. Voor alle webpagina's die hierin worden aangemaakt, wordt deze workflow gestart. De achilleshiel van Content Management Server 2002 was



Afbeelding 5. Het web author-menu in SharePoint Server 2007.

dat de workflowmogelijkheden erg beperkt waren en dat deze lastig aan te passen/uit te breiden waren. Met SharePoint Server 2007 kunnen we hier de workflow eenvoudig aanpassen of nieuwe workflows toevoegen. SharePoint Server 2007 maakt gebruik van de Windows Workflow Foundation (WinWF) engine. We kunnen met SharePoint Designer of met Visual Studio 2005 zelf workflows bouwen. Nadat we deze hebben opgeslagen in SharePoint Server 2007, zijn deze te koppelen aan de pages document-library. Hierbij kun je aangeven of deze handmatig gestart moet worden, of dat deze automatisch start als er een webpagina aangemaakt of gewijzigd is.

Webpart ontwikkelen en toevoegen

Bij een typische implementatie van een Content Management Server 2002-website wordt er altijd maatwerk ontwikkeld, zoals navigatiecontrols, authenticatiecontrols, lijsten/overzichten (bijvoorbeeld voor FAQ's of een nieuwsoverzicht), zoeken en meestal ook nog wat functionaliteiten van applicatieve aard. SharePoint Server 2007 neemt veel maatwerk werk uit handen. Zoals al eerder aangegeven maak je in SharePoint Server 2007 gebruik van ASP.NET 2.0-providers; hierin zullen de Navigation-provider en Membership-provider ingezet kunnen worden voor de navigatiecontrols en de inlogcontrols op de website. Daarnaast zijn lijsten altijd al een feature van SharePoint geweest. Deze kunnen we nu dus ook inzetten in onze webcontentmanagement-site. Hierdoor wordt ons veel maatwerk uit handen genomen. Toch is het erg waarschijnlijk dat een website-implementatie om maatwerk vraagt. Omdat het webcontentmanagement nu onder SharePoint valt, kunnen we hier nu dus webparts voor inzetten. Omdat SharePoint Server op het ASP.NET 2.0 framework is gebouwd, ontwikkelen we de webparts nu ook op het ASP.NET 2.0 framework. Hierdoor vallen ze in de 'System.Web.UI.WebControls' namespace. Dit betekent dat 'Microsoft.SharePoint.WebPartPages.WebPart' vervangen is door 'System.Web.UI.WebControls.WebParts.WebPart'. Dit betekent ook dat dezelfde webparts gebruikt kan worden in een ASP.NET 2.0-applicatie en in een SharePoint Server 2007/ SharePoint Services 3.0-site. In codevoorbeeld 2 wordt een eenvoudige managementconsole voor webcontentmanagement gebouwd. Deze laat de metadata van de huidige website zien.

Omdat onze webpart nu van 'System.Web.UI.WebControls.WebParts.WebPart' erft, overschrijven we nu niet de 'override void RenderWebPart(HtmlTextWriter output)'-methode, maar de 'override void Render(HtmlTextWriter output)'-methode. Via 'this.Context.User.Identity.Name' wordt de naam van de gebruiker, die in SharePoint Server 2007 is ingelogd (bijvoorbeeld de contentmanager). Vervolgens gaan we de huidige SharePoint-website ophalen via 'SPControl.GetContextWeb(this.Context)'. Van deze website gaan we de volgende metadata ophalen; 'mySite.Author' (auteur van de site) en 'mySite.Created' (datum wanneer de site is aangemaakt). Vervolgens zoeken we de 'Pages' Document Library ('mysite.Folders["Pages"]') om te kijken hoeveel webpagina's er in deze website staan én om te kijken wat de welkomspagina van deze website is. Na een suc-

```
<% ... %>
<asp:Content ContentPlaceholderID="PlaceHolderMain" runat="server">
  <PublishingWebControls:RichHtmlField
    ID="Content"
    runat="server"
    FieldName="PublishingPageContent" />
  <hr />
  <WebPartPages:WebPartZone
    ID="BottomZone"
    runat="server"
    Title="<%%$Resources:cms, WebPartZoneTitle_Bottom%>" >
    <ZoneTemplate></ZoneTemplate>
  </WebPartPages:WebPartZone>
</asp:Content>
Codevoorbeeld 1.
```

```

using System.Text;
using System.Web.UI;
using Microsoft.SharePoint;
using Microsoft.SharePoint.WebControls;

namespace Avanade.NETMagazine
{
    public class WebSiteMetaData :
        System.Web.UI.WebControls.WebParts.WebPart
    {
        protected override void Render(HtmlTextWriter writer)
        {
            StringBuilder outputText = new StringBuilder();

            // Haal de identiteit van de gebruiker op.
            outputText.Append(string.Format("<b>Welcome,</b> {0}<br />",
                this.Context.User.Identity.Name));

            // Haal de huidige website op.
            SPWeb mySite = SPControl.GetContextWeb(Context);

            // Publiceer meta-data van de website.
            outputText.Append("<p>Web Content Management
                details voor deze website.</p>");
            outputText.Append("<ul>");
            outputText.Append(string.Format("<li>Auteur: {0}</li>",
                mySite.Author));
            outputText.Append(string.Format("<li>Aangemaakt: {0}</li>",
                mySite.Created));
            if (mySite.Folders["Pages"].Exists)
            {
                outputText.Append(string.Format("<li>Aantal webpagina's
                    binnen deze website: {0}</li>",
                    mySite.Folders["Pages"].Files.Count));
                outputText.Append(string.Format("<li>Welkom pagina van
                    deze website: {0}</li>",
                    mySite.Folders["Pages"].WelcomePage));
            }
            outputText.Append("</ul>");

            writer.Write(outputText.ToString());
            writer.Flush();
        }
    }
}

```

Codevoorbeeld 2.

cesvolle build kunnen we de assembly deployen via de GAC. In dit voorbeeld wordt onze webpart uitgerold door hem te kopiëren in de BIN-directory van onze webapplicatie (\\inetpub\\wwwroot\\wss\\VirtualDirectories\\beta.avanade.com80_app_bin\\). De tweede stap is het aanmaken van een 'Safe Control' entry in de web.config van onze website: `<SafeControl Assembly="Avanade" Namespace="Avanade.NETMagazine" TypeName="*" />`

Om deze demo eenvoudig te houden, maken we hier geen gebruik van strong names en volstaat het om alleen de assembly-naam en namespace op te geven. Met `TypeName="*">` wordt aangegeven dat alle classes binnen deze namespace als safe control gedeclareerd worden.



Afbeelding 6. De WCM-beheer Webpart opgenomen in de webpagina.

Nu kunnen we onze webpart nog niet toevoegen aan onze website. Hiervoor moeten we eerst in de site settings onze webpart toevoegen aan de Webparts Gallery. Nieuw in SharePoint Server 2007 is dat we in onze webpart eenvoudig kunnen toevoegen aan de Webpart Gallery door op 'New' te klikken. SharePoint opent dan de web.config-file en leest alle assemblies die een safecontrol-entry hebben. Vervolgens worden alle assemblies geladen en via reflection wordt gezocht naar de classes die geconfigureerd zijn als safe control en toegevoegd kunnen worden als webpart. In de gegenereerde lijst selecteren we onze webpart (WebSiteMetaData.webpart). Deze wordt, door op 'Populate Gallery' te klikken, toegevoegd aan de galerij. Als we vervolgens op edit klikken voor onze webpart, kunnen we de XML te bekijken. Dit is de webpart-file die automatisch gegenereerd werd toen we onze assembly toevoegden aan de Webpart Gallery. Uiteindelijk kunnen we onze webpart toevoegen aan onze webpagina. Het resultaat is te zien in afbeelding 6. Als we vervolgens onze code aanpassen en deze opnieuw compileren, volstaat het om de assembly opnieuw te kopiëren naar de bin-directory. SharePoint pikt deze dan automatisch op.

Variations

Variations is een nieuwe feature. Variations is ontwikkeld om eenzelfde website geschikt te maken voor verschillende platformen (bijvoorbeeld mobile devices), om een website via een andere look-and-feel (branding) te publiceren of voor het opzetten van een meertalige site. Als we een meertalige website willen opzetten, moeten we in site administration aangeven dat we variations willen gebruiken ('Enable Variations'). Vervolgens maken we een label aan voor elke taal. Hierbij is één van de labels de 'source'. Deze dient als uitgangspunt voor website. Als we bijvoorbeeld twee labels aanmaken (NL en EN), geven we aan dat NL onze source is. Nadat de labels zijn aangemaakt moet de variation-hiërarchie gebouwd worden. Nu worden er twee subsites, NL en EN genaamd, aangemaakt. Als er daarna een pagina in de source (NL) wordt aangemaakt, wordt deze automatisch naar alle andere labels aangemaakt. Hoewel SharePoint Server deze webpagina's zelf natuurlijk niet vertaalt, helpt deze functie om het overzicht te houden op jouw meertalige site. Een lijst van niet gepubliceerde webpagina's toont namelijk alle webpagina's die nog gepubliceerd (en dus vertaald) dienen te worden.

Search

Een zoekmachine is geïntegreerd in SharePoint Server 2007 en kan ingezet worden voor de zoekfunctionaliteit van onze website. Ten opzichte van de vorige versie van SharePoint is de search-engine sterk verbeterd. Het sterkste punt is dat Microsoft veel aandacht heeft geschonken aan het verbeteren van de berekening van de relevantie van de zoekresultaten. In vergelijking met de vorige versie van SharePoint wordt er nu bijvoorbeeld ook gekeken naar:

- De 'click distance', die bepaalt in hoeveel 'clicks' je vanaf de homepage bij deze content kunt komen. Hoe verder weg, hoe minder belangrijk de content waarschijnlijk is.
- Hyperlink anchor-tekst
- Metadata
- Taalafhankelijk
- Met bestandstypen kun je aangeven dat bepaalde documenttypen (bijvoorbeeld PDF) binnen een website belangrijk zijn, omdat deze website belangrijke informatie meestal via PDF-documenten publiceert.

Natuurlijk is de tekstanalyse ook sterk verbeterd. Daarnaast gebruikt SharePoint Server 2007 ook de term 'authoritative sites' waarmee aangegeven kan worden dat niet alle sites binnen de website even belangrijk zijn. Authoritative sites zijn de hoofdsites die de belangrijkste informatie bevatten. Tijdens het implementeren van de Content Management Server 2002-website werd er vaak gezocht naar third-party zoekmachines omdat een integratie met SharePoint-search niet echt een optie was. SharePoint Server 2007

levert een krachtigere zoekmachine die daarom nu goed als onderdeel van de webcontentmanagement-website, ingezet kan worden voor onze website.

Dit artikel is geschreven tijdens de bèta 2 release van SharePoint Server 2007. Bij de uiteindelijke release hiervan kunnen er wijzigingen zijn opgetreden.

Een thuisgevoel

Microsoft heeft ervoor gekozen om hun Content Management Server 2002-functionaliteit te integreren in SharePoint Server 2007. Dit is een sterk verbeterd product ten opzichte van de vorige versie van SharePoint Server en ook ten opzichte van Content Management Server 2002. Je zult je als CMS-ontwikkelaar snel thuis voelen in het webcontentmanagement van SharePoint Server. De manier van werken is wel iets veranderd. Vooral de aanpak om maatwerk voor je website te bouwen. Zo vind je in SharePoint Server 2007 veel features out-of-the-box die je in Content Management Server 2002 nog zelf moest ontwikkelen, zoals navigatie- en menu-controls. Het maatwerk dat overblijft zul je nu voornamelijk via een webpart implementeren. Ook de concepten voor de contentmanager zijn niet veel veranderd. Het aanmaken en beheren van webpagina's met templates doet vertrouwd aan.

Jules Hoppenbrouwers is werkzaam als consultant bij Avanade Nederland (www.avanade.nl). Hij is voornamelijk werkzaam in de Information Worker (IW) richting. Zijn emailadres is julesh@avanade.com.

Links

SharePoint Server 2007 for CMS 2002 Developers: <http://msdn2.microsoft.com/en-us/library/ms406043.aspx>

Microsoft Office SharePoint Server 2007 SDK: <http://msdn2.microsoft.com/en-us/library/ms550992.aspx>

WCM resources in SharePoint Server 2007: <http://mcmsfaq.com/moss/>

Enterprise Content Management (ECM) Team Blog: <http://blogs.msdn.com/ecm/>

Content Management Server Dev Center: <http://msdn.microsoft.com/office/server/mcms/>

(advertentie Microsoft Press)



Debugging Microsoft® .NET 2.0 Applications

ISBN: 0-7356-2202-7

Auteur: John Robbins (Wintellect)