

SitePublisher 2.0 Highlights

URL-rewriting
Caching
Content filtering

1. Introductie

SitePublisher maakt onderdeel uit van de SiteManager CMS productrange. SitePublisher biedt de ontwikkelaar van websites een verzameling bouwblokken.

Door gebruik te maken van SitePublisher kan het realisatieproces sterk worden versneld, omdat veel voorkomende functies niet meer hoeven te worden geprogrammeerd. Daarnaast zijn deze bouwblokken grondig getest, waardoor het eenvoudig is om snel een hoogwaardig eindproduct te realiseren.

SitePublisher versie 2.0 is volledig herschreven, om optimaal gebruik te kunnen maken van nieuwe mogelijkheden van het Microsoft .NET raamwerk. Hierbij zijn diverse zeer krachtige nieuwe functies gerealiseerd.

Dit document beschrijft de belangrijkste daarvan.

Het document is geschreven voor SiteManager CMS gebruikers die geïnteresseerd zijn in de achter SiteManager CMS liggende technieken.

2. URL-rewriting

URL-rewriting is gericht op het wijzigen van “technische” URL’s in webadressen die informatief zijn voor zowel menselijke bezoekers als de robots van Google en andere zoekmachines. SitePublisher 2.0 biedt URL rewriting in verschillende smaken.

2.1. Basic URL rewriting

De meest basic vorm van URL-rewriting is gebaseerd op objecten in de database (nieuwsartikelen, maar ook pagina’s en categorieën), en verandert dit webadres:

`www.integrace.nl/News.aspx?id=2329`

in dit webadres:

`www.integrace.nl/News/2329/De-titel-van-dat-nieuwsbericht`

Het voordeel hiervan is simpeler URL’s, en betere indexering door Google (de naam van het bericht staat in de URL). Tevens is er nog een veiligheidsaspect: je kunt nu niet meer aan de URL zien dat de site met ASP.NET is gebouwd.

2.2. URL-rewriting op basis van structuur

SiteManager CMS biedt de mogelijkheid om content-items onder te brengen in een classificatie-boom.

Stel dat wij handelen in paarden, en deze willen tonen op een website. Hierbij willen we een onderscheid maken tussen 2-jarigen en 3-jarigen. In de sitenavigatie zien we dan (met tussen haakjes de CMS bron):

- Home (Pagina)
- Paarden (Pagina)
 - 2-jarigen (Paardencategorie – custom type)
 - AllStar Z (Paard – module)
 - Casper (Paard – module)
 - 3-jarigen (Paardencategorie – custom type)
 - Castello (Paard – module)
 - Charissa Z (Paard – module)
- Nieuws (Pagina)

SitePublisher kan URL’s opbouwen op basis van deze hiërarchische informatie. In het voorbeeld zou de URL van Castello bijvoorbeeld normaal gesproken zijn:

`www.paarden.com/Horse.aspx?id=23048`

Of met de eerstgenoemde vorm van URL-rewriting:

`www.paarden.com/Horse/23048/Castello`

Maar met rewriting op basis van structuur wordt het:

`www.paarden.com/Paarden/2-jarigen/Castello`

Het voordeel is dat er veel meer contextuele informatie in de URL zit, waar Google wel degelijk naar kijkt, en voor de bezoeker is het ook erg duidelijk.

2.3. Redirects/mappings

Dit is een custom type in de database waarbij je gewoon kunt aangeven dat je wilt dat een bepaalde URL op een bepaalde locatie uitkomt.

Zo kun je dus zeggen dat:

`www.integrace.nl/onzeoudesite/contact_formulier.html`

uitkomt op:

`www.integrace.nl/Over-Integrace/Contact`

Erg handig voor migraties dus. Maar ook voor landingspagina's is het handig, want je kunt dus aangeven dat:

`www.muziekcentrum.nl/Racoon`

uitkomt op:

`www.muziekcentrum.nl/Concerten/2010/Pop/Nederlands/Racoon`

Ook kun je aangeven of de oorspronkelijke URL moet blijven staan ('mapping') of dat de bezoeker moet worden doorgestuurd ('redirect').

Natuurlijk is het onhandig als je dit voor pagina's en producten moet doen waarbij je continu moet opzoeken wat de huidige URL ook alweer is, dus daar is een handige feature voor:

2.4. URL-override

Bij URL-override krijg je simpel gezegd in een module een extra veld erbij waarin je kunt aangeven wat de uiteindelijke URL moet zijn. Zo heb je een pagina over SiteManager:

`www.integrace.nl/Producten/ContentManagement/SiteManager/Over-SiteManager`

Daar kun je dan een URL-override voor ingeven zodat het wordt:

`www.integrace.nl/SiteManager`

Deze korte URL's zijn handig voor bezoekers, en tevens weer voor Google cs. We hebben hier tenslotte een combi van bedrijfsnaam en zoekwoord, zonder allerhande overbodige informatie, dus met een hoog zoekwoordgewicht.

2.5. Multi-site URL's

2.5.1. Enkelvoudige multi-site URL's

Hierbij kun je bij elk item uit een module aangeven op welke site dit item moet verschijnen. Als een website een Nederlandse en een Engelse site heeft, en bij een nieuwsitem is aangegeven dat dit op de Nederlandse site staat, dan zal de bezoeker die het item benadert op de Engelse site automatisch worden doorgestuurd naar de Nederlandse site.

2.5.2. Meervoudige multi-site URL's

Hier kun je bij een item meerdere (nul of meer) sites aangeven waar dit item moet komen te staan, en je moet een primaire site aangeven. Dit heeft te maken met canonical URL's (zie verderop).

Als een bezoeker een item benadert op een site die niet aangevinkt is (door zelf een URL samen te stellen), wordt deze bezoeker doorgestuurd naar dat item op de primaire site.

2.5.3. Site-URL's

Het doel van de nieuwe sites is ook dat het zogeheten 'REST'-principe wordt **aangehouden, wat staat voor *Representational State Transfer*, i.e., elke pagina (die niet een post-back is) moet z'n eigen URL hebben. Dat geldt dus ook voor sites.** SitePublisher 2.0 ondersteunt 3 manieren om een site een unieke URL te geven:

- Domeinnaam (`www.integrace.nl`, `www.sitemanager-cms.nl`);
- 'Directory' (`www.integrace.nl/nl`, `www.integrace.nl/en`);
- Gecombineerd (`www.integrace.nl/nl`, `www.sitemanager-cms.nl/nl`)

2.5.4. Canonical URL's

De canonical URL is een meta-tag die op de pagina staat en die voor Google aangeeft, simpel gezegd: 'maakt niet uit wat de URL van deze pagina nu is, de enige officiële, belangrijke URL is: ...'.

Door gebruik te maken van de bovenstaande multi-site features kan SitePublisher eenvoudig bepalen wat de canonical URL van een pagina (of object of formulier of...) is en die in de tag zetten, wat de indexeerbaarheid van een site weer verhoogt.

2.6. Meta-data overrides

Elk object heeft een aantal meta-data eigenschappen, zoals de paginatitel, de keywords, de beschrijving etcetera. Voor bijvoorbeeld de paginatitel van een nieuwsitem geldt dat deze standaard gelijk is aan de titel van het bericht.

Maar voor Google wil je soms een andere titel opgeven, en in SitePublisher 2.0 kan dat dus. Dit geldt voor de volgende eigenschappen:

- Paginatitel (standaard wordt de titel/naam van een item gebruikt);
- URL (dito, en met 'URL' wordt bedoeld de naam van het item die in de URL verschijnt);
- Tooltip (wat je ziet als je met je muis op een linkje blijft hangen)

3. Caching

Snelheid is een belangrijke kwaliteitsindicator, en zowel belangrijk voor bezoekers als voor Google. SitePublisher 2.0 biedt een aantal faciliteiten voor het verhogen van de snelheid waarmee pagina's worden getoond. De belangrijkste daarvan is een geheel vernieuwd caching mechanisme.

Caching is simpelweg het bewaren en hergebruiken van de uitvoer van een complexe of zware functie (zoals het genereren van een pagina of het opvragen van data uit een database), zodat de functie niet telkens opnieuw hoeft te worden uitgevoerd. SitePublisher 2.0 gebruikt hier twee varianten van.

3.1. Pagina-caching

Bij pagina-caching wordt de output van een pagina opgeslagen en opnieuw gebruikt als de pagina niet veranderd is, wat de snelheid van het opvragen sterk verlaagt.

Door de 'conditional GET' feature van HTTP te gebruiken wordt deze snelheid nog weer fors hoger. In een conditional GET ziet de browser elke keer als een pagina wordt opgevraagd twee dingen:

- De datum en tijd van de laatste wijziging van die pagina;
- Een zogeheten 'E-Tag', een hash van de URL, die laatste wijzigingsdatum en eventuele andere informatie die de staat van die pagina aangeeft.

De volgende keer dat de browser diezelfde pagina opvraagt, geeft hij headers mee waarmee wordt gezegd 'geef mij een nieuwe versie van die en die pagina, **mits** de pagina nieuwer is dan dit, of als de E-Tag anders is dan dit'. Als de pagina gewijzigd is of als de E-Tag anders is, dan wordt de pagina opnieuw aan de browser gegeven met de nieuwe wijzigingsdatum en E-Tag. Is dat echter niet zo, dan is de pagina ongewijzigd, en krijgt de browser alleen maar het antwoord '304 Not Modified'.

Op dat moment hoeft de browser alleen maar de gecachte versie van de pagina van de harde schijf halen (wat een stuk sneller is dan het opvragen over Internet) en de bezoeker ziet een pagina.

Dit principe werkt ook voor afbeeldingen, CSS en JavaScript, alleen voor pagina's is het een iets ingewikkelder en minder vanzelfsprekend verhaal.

3.2. Data-caching

SitePublisher 2.0 cacht net als versie 1.0 de resultaten van query's aan de database, alleen doet de nieuwe versie dat iets beter, en vooral slimmer.

Omdat de nieuwe SitePublisher LINQ gebruikt (een manier om verzamelingen informatie te benaderen in C#) kan SitePublisher automatisch detecteren welke tabellen je in een query gebruikt, en dus op basis daarvan de juiste zogenaamde 'dependencies' instellen. Dat is een feature uit ASP.NET die, als je die goed aanstuurt, er voor zorgt dat de resultaten van een query gecached blijven totdat de

gegevens in één van de tabellen, die in de query worden gebruikt, worden aangepast.

De nieuwe SitePublisher doet daar nog een schepje bovenop, want daar wordt de data gecached totdat één van de tabellen wordt aangepast, óf tot de vroegste archiveerdatum van het resultaat is verlopen, óf tot de eerstvolgende publicatiedatum van een item uit één van de tabellen is verlopen (waardoor je dus een item op 'Published' kunt zetten met een datum in de toekomst).

4. Content filters

SitePublisher 2.0 beschikt over een geheel nieuwe filterarchitectuur. Alle content die zichtbaar is op een site is een set van filters gepasseerd. In deze filters wordt content gecontroleerd en waar nodig aangepast. Twee belangrijke filters zijn het “email scrambling” filter en het “afbeelding optimalisatie” filter.

4.1. E-mail-scrambling

E-mail scrambling zorgt ervoor dat e-mailadressen en linkjes naar e-mailadressen worden versleuteld naar iets waar eigenlijk alleen een browser wat mee kan: JavaScript.

Neem het mailadres `corporate@integrace.nl`. Het scrambling filter zal dit automatisch omzetten naar:

```
<script type="text/javascript"
language="javascript">document.write(String.fromCharCode((5 + 55), (3 + 94), (6 + 26),
(24 + 80), (12 + 102), (30 + 71), (13 + 89), (1 + 60), (7 + 27), (30 + 79), (26 + 71),
(12 + 93), (35 + 73), (27 + 89), (8 + 103), (4 + 54), (12 + 87), (6 + 105), (8 + 106),
(14 + 98), (17 + 94), (37 + 77), (31 + 66), (20 + 96), (13 + 88), (7 + 57), (10 + 95),
(34 + 76), (25 + 91), (8 + 93), (29 + 74), (9 + 105), (23 + 74), (27 + 72), (18 + 83),
(4 + 42), (5 + 105), (12 + 96), (9 + 25), (14 + 48), (6 + 154), (4 + 95), (3 + 108),
(26 + 88), (27 + 85), (20 + 91), (15 + 99), (17 + 80), (13 + 103), (29 + 72), (14 +
50), (19 + 86), (1 + 109), (23 + 93), (3 + 98), (8 + 95), (5 + 109), (4 + 93), (11 +
88), (32 + 69), (9 + 37), (15 + 95), (13 + 95), (8 + 52), (14 + 33), (19 + 78), (12 +
50)));</script>
```

In feite is zijn het de karaktercodes van alle tekens van een stuk tekst (in dit geval een link naar een e-mailadres), verdeeld in een simpel sommetje (dus zeer snel). Google zal dit niet kunnen indexeren, en een spambot ook niet snel, maar een bezoeker kan het gewoon lezen en gebruiken.

Omdat alle content die een gebruiker invoert wordt nagelopen door het filter vóórdat het zichtbaar is op een site, is het 100% zeker dat alle emailadressen zijn versleuteld. Dit hoeft dus niet meer handmatig te gebeuren door in de HTML het adres te vervangen door een versleutelde versie.

4.2. Afbeeldings-optimalisatie

In het afbeeldings-optimalisatie filter kan worden ingesteld hoe breed en hoe hoog een afbeelding maximaal mag zijn, en de URL van elke afbeelding in de content wordt vervangen door een URL die de afbeelding resized naar maximaal dat formaat. Ook wordt er gekeken of er `width` en `height` voor die afbeelding zijn gespecificeerd. Als dat niet zo is, worden die berekend en toegevoegd, wat in de browser een veel rustiger beeld oplevert.