

Content Management en XML: terminologie en achtergrondinformatie

In dit document zet ik kort uiteen hoe *content management systemen* werken en geef ik wat achtergronden over het werken met *XML*.

Wat is een CMS?

De term content management systeem (CMS) wordt gebruikt voor alle systemen waarbij tekst en beeld op een centrale plek wordt opgeslagen en deze op gestandaardiseerde (meestal ook op geautomatiseerde) wijze wordt gepubliceerd. Als het eindproduct uit webpagina's bestaat dan spreken we van een web content management systeem; dit is een zeer belangrijk toepassingsgebied.

Bij Quantes zal een web content management systeem niet voldoen, omdat hier het belangrijkste eindproduct drukklare PDF-bestanden zullen zijn.

In alle content management systemen is een zekere mate van workflow-ondersteuning aanwezig. Dat betekent dat van de content-bouwstenen die in het systeem zijn opgeslagen een status wordt bijgehouden. Zo'n status kan zijn: "concept", "klaar voor eindredactie", "definitief". Vaak ondersteunt het systeem ook versiebeheer van de bouwstenen.

Inhoud van een CMS

Een CMS kan in principe allerlei soorten informatie bevatten: tekst, beeld, maar ook geluid en film. Kenmerkend is dat er geen volledige kant-en-klare documenten worden opgeslagen, maar losse bouwstenen. Dit kunnen dus losse plaatjes zijn, of losse tekstgedeelten (in CMS-terminologie heten die vaak *topics*). Zo'n los tekstgedeelte omvat meestal één of meer paragrafen, maar het kan ook een hoofdstuk zijn. Om geautomatiseerde opmaak mogelijk te maken, dient ook de structuur van de tekst te worden vastgelegd. Daarvoor wordt tegenwoordig vrijwel uitsluitend *XML* gebruikt (zie hieronder).

Voordat een publicatie kan worden gegenereerd vanuit de verschillende content-bouwstenen, moet eerst worden vastgelegd welke bouwstenen in welke volgorde gebruikt moeten worden. De selectie van topics kan *handmatig* zijn (de auteur maakt dan een *assemblagelijst*), of *automatisch*: de topics en/of illustraties worden bij elkaar gezocht aan de hand van een zoekvraag.

Bouwstenen van een CMS

Het "hart" van een CMS wordt gevormd door de centrale opslagplaats. Dit is bijna altijd een database. Deze database wordt ook wel *data warehouse* genoemd (vooral als de gegevens uit verschillende bronsystemen afkomstig zijn) of *repository*. De term repository is algemener dan database en wordt gebruikt voor elke vorm van centrale opslag met het oog op hergebruik.

Om gegevens in het CMS te krijgen is er een auteurs-interface. Vaak is dit een specifieke webtoepassing waarmee via het invullen van browserschermen de content in de repository gewijzigd kan worden. Voor gebruikers van het CMS is deze auteursomgeving de belangrijkste manier om met het systeem te werken. Overigens zijn er natuurlijk nog andere manieren om gegevens in het CMS te krijgen, bijvoorbeeld door het uploaden van documenten. Voor plaatjes wordt bijna altijd alleen een upload-mogelijkheid voorzien, die bewerk je dus niet rechtstreeks in het CMS.

De "achterkant" van het CMS wordt gevormd door een *publicatie engine*. Voor *CMS-to-print* oplossingen is dit bijna altijd een losstaand deel van het systeem dat vanuit het centrale gedeelte in batch wordt aangestuurd. Vooraf heeft dan de assemblage en export van alle benodigde content-bouwstenen plaatsgevonden. Deze *compound* (export-set) wordt opgepikt

door de publicatie engine, die vervolgens PDF aanmaakt. Bekende applicaties die worden ingezet als publicatie engine zijn Adobe FrameMaker en Adobe InDesign. (Maar er zijn meer mogelijkheden: TeX, XSL-FO.) Meestal is het mogelijk om verschillende publicatie engines in te zetten bij één CMS.

De opslagplaats (repository), de auteursomgeving en de publicatie engine zijn de belangrijkste CMS-onderdelen. Daarnaast heeft het CMS vaak nog een administratieve interface (voor het uitdelen van gebruikersrechten, het definiëren van workflowstatussen, het aanmaken van publicaties e.d.) en een zoekinterface voor het doorzoeken van het CMS. Deze interfaces kunnen geïntegreerd zijn met de auteursomgeving.

XML

eXtensible Markup Language (XML) is een bepaalde manier om gegevens gestructureerd vast te leggen. In het geval van document-achtige structuren hebben we het dan bijvoorbeeld over: kop, tussenkop, platte tekst, tabelcel, opsomming, verwijzing naar een plaatje, etc. Deze structuuronderdelen heten *elementen* in de XML-terminologie.

De elementen zijn hiërarchisch geordend binnen een document. Deze boomstructuur moet uiteindelijk altijd één allesomvattend element hebben. Dit element noemen we het *rootelement*. Verder onderscheiden we *blokelementen* (een blokelement bevat óf één of meer subelementen, óf tekst, maar geen combinatie daarvan) en *inline-elementen* (zo'n element kan gecombineerd worden met tekst binnen het container element). Voorbeelden van inline-elementen zijn: hyperlink, verwijzing, vet en cursief.

De syntax van een XML-document is eenvoudig: elk element wordt voorafgegaan en afgesloten met de naam van het element tussen "vishaken": "<" en ">". Zo'n naam met vishaken heet een *tag*. Een openingstag ziet er dus als volgt uit: "<tag>", een sluittag heeft een extra schuine streep: "</tag>". Een XML-document waarin alle openings- en sluittags correct zijn opgenomen (zonder tikfouten) en waarin alle elementen netjes hiërarchisch (in een boomstructuur) zijn geordend heet *well-formed*. Een well-formed XML-document is eenduidig te lezen door een computerprogramma. Dit inlezen van een XML-document noemen we *parsen*.

Behalve well-formedness, kan je nog een stapje verder gaan met XML-documenten. Dan leg je de structuurdefinitie van een XML-document vast. De structuurdefinitie:

- legt vast welke namen de verschillende elementen hebben (dus de tagnamen)
- legt vast in welke volgorde deze elementen mogen voorkomen in een document

Zo kun je bijvoorbeeld definiëren dat een tabel gevormd wordt door het element <table>, dat een tabel rijen mag bevatten (element <row>) en cellen binnen zo'n rij (element <entry>). Een eenvoudige tabel zou er dan als volgt uit kunnen zien¹:

```
<table>
  <row>
    <entry>rij 1, kolom 1</entry>
    <entry>tekst in kolom 2</entry>
  </row>
  <row>
    <entry>eerste cel in kolom 2</entry>
    <entry>tekst in kolom 2</entry>
  </row>
</table>
```

Er zijn verschillende manieren om deze structuurbeschrijving vast te leggen. De meestgebruikte zijn *DTD* en *schema*. DTD staat voor *Document Type Definition*. Het is de oudste standaard van de twee, maar wordt nog veel gebruikt. Een DTD biedt wat minder mogelijkheden tot

¹ Witruimte mag vrijelijk worden toegepast binnen een XML-document; meerdere spaties en regelovergangen worden vertaald naar één spatie.

detaillering, maar de syntax is een stuk eenvoudiger dan die van een XML-schema. Let op dat “DTD” ook vaak wordt gebruikt als synoniem voor “structuurdefinitie van het document”; uiteindelijk kan deze definitie dan best zijn vastgelegd in een schema.

Verder lezen

Op het internet is natuurlijk heel veel informatie te vinden over deze onderwerpen. De Wikipedia-artikelen geven een goede start:

- <http://nl.wikipedia.org/wiki/XML>
- http://nl.wikipedia.org/wiki/Enterprise_Content_Management

Verder bestaat er een boek dat precies over dit onderwerp gaat:

Content Management, door Jan-Willem de Koning (red.), Academic Service 2006, ISBN 90 1211 623 6.