

Webdatabases vormen een gouden combinatie van twee verschillende ontwikkelingen. Aan de ene kant heb je het exponentieel groeiende internet dat wanhopig op zoek is naar relevante inhoud. Aan de andere kant is er de wereld van de databases, relatief rijk aan inhoud maar met een beperkt aantal gebruikers. In een webdatabase komen deze twee werelden bij elkaar, maar daarin schuilt nu juist ook de moeilijkheid.

achtergrond

Onderhoudsvrije webdatabases

Het beste van twee werelden?

De combinatie van onbeperkte toegang en onbegrensde inhoud maakt het vormgeven van webdatabases tot een moeizaam proces. Het gaat immers om verschillende IT-disciplines. In dit artikel zullen we de verschillende mogelijkheden bekijken en wel vanuit het oogpunt van onderhoud. Welke oplossing voldoet aan de gedefinieerde eisen en zorgt daarnaast voor een minimale onderhoudsinspanning?

Aan de ene kant is er de datagerichte benadering.

Daarnaast bestaat de applicatiegerichte insteek.

In grote lijnen kunnen daardoor vier verschillende wegen worden bewandeld om webdatabases te realiseren. Het eerste pad vertrekt vanuit de bestaande databaseproducten. Hieraan wordt dan webfunctionaliteit toegevoegd. Een tweede manier is om HTML editors te voorzien van database functionaliteit. Een derde mogelijkheid wordt gevormd door webdatabases op applicatieservers. En tenslotte is het mogelijk om zelf een webdatabase te programmeren.

AARD DER VERSCHILLEN Het is zaak om eerst helder te hebben waarom we een webdatabase nodig hebben. De redenen hiervoor kunnen heel divers zijn en sturen daardoor naar een andere oplossingsrichting. Het kan bijvoorbeeld om de volgende activiteiten gaan:

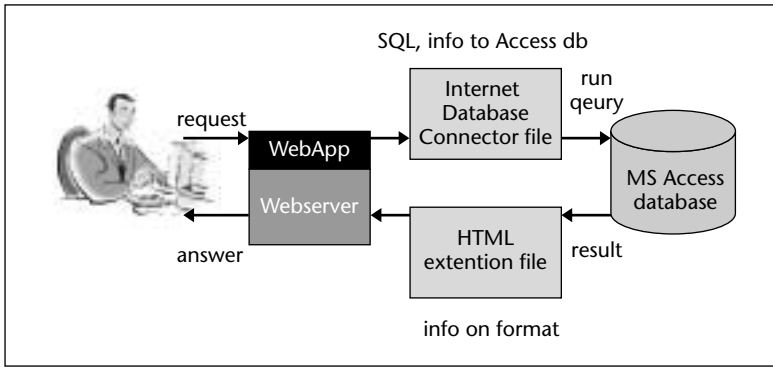
- Het dynamisch publiceren van webpagina's. De inhoud van de webpagina's wordt uit een database gehaald.
- Het afhandelen van transacties zoals bijvoorbeeld bij electronic commerce gebeurt.

- Het opslaan en ophalen van analytische en/of historische informatie.
- Combinatie van een webapplicatie en databasefunctionaliteit via een browser.

De oplossingen die door de softwareleveranciers zijn ontwikkeld kunnen dus verdeeld worden in vier verschillende categorieën. De werkwijze die hierbij hoort is

Het programmeren van een webapplicatie is vanuit het oogpunt van onderhoud de minst geschikte oplossing

heel verschillend. In het eerste geval gaat het om een databaseproduct en wordt een file geëxporteerd naar een webserver. De inhoud van de file wordt vervolgens gepubliceerd. Bij de HTML editors gaat het om pakketten die de mogelijkheid bieden om databases te benaderen en vervolgens in een expliciete publishstap webpagina's opleveren. De webdatabase applicatieservers kunnen twee vormen aannemen. Het kan hierbij gaan om een stuk middleware tussen de database en de webserver in. De aard van de connectie is hierbij dynamisch. Tweede mogelijkheid is een gecombineerde applicatieserver. Bij deze producten is rekening gehouden met de behoefte aan een database in een webomgeving. De webclient praat nu tegen de applicatieserver. Het programmeren van een webapplicatie wordt hier



FIGUUR 1. Databases met webfunctionaliteit

verder niet behandeld omdat duidelijk zal zijn dat het voor ons onderwerp (onderhoud) de minst geschikte oplossing is. Het met behulp van C++, Java of Visual Basic in elkaar sleutelen van een webapplicatie zou alleen overwogen moeten worden in het geval dat alle overige oplossingen niet voldoen.

EISEN Het gaat uiteraard niet alleen om het snel en effectief ontwerpen van webpagina's. Er zijn legio sites die bestaan uit vele honderden webpagina's. Het onderhouden en beheren van dergelijke websites vormt een formidabele uitdaging. Hiermee wordt duidelijk dat er behoefte is aan een toolset om aan effectief webmanagement te kunnen doen. Hiertoe wordt gerekend het onderhoud van de webpagina's, webpagina-componenten (denk aan bijvoorbeeld grafische afbeeldingen en logo's), inhoud, hyperlinks en de structuur.

Om te kunnen bepalen hoe een bepaalde oplossing scoort moet onderhoudbaarheid verder worden benoemd. Wat bedoelen we hier eigenlijk mee? Wat zijn de specifieke eisen die we stellen aan software om een website te bouwen en te onderhouden?

Een eerste, uiteraard onvolledige, opzette zou er als volgt kunnen uitzien:

- Overzicht van alle webpagina's waaruit de website bestaat en de status hiervan. Dus niet alleen alle actieve (= gepubliceerde) pagina's, maar ook de nog niet gepubliceerde pagina's.
- Overzicht van alle hyperlinks en het automatisch kunnen controleren of ze nog werken. Als dit niet het geval is moeten ze automatisch hersteld kunnen worden.
- Overzicht van de workflow van alle personen betrokken bij de bouw c.q. het onderhoud van de website. Gebruik van de website.
- Overzicht van alle gebruikers en de rechten die zijn toegekend. Inhoudelijk beheer (documentbeheer).
- Overzicht van gebruikte componenten en deze met één actie kunnen vervangen (zoek en vervang) op elk mogelijk niveau.
- Zowel HTML als XML kunnen genereren.

DATABASE MET WEBFUNCTIONALITEIT

Voorbeelden hiervan zijn:

- FileMaker Pro 4.0 van FileMaker
- Microsoft Access 97
- Corel Paradox 8.0

Redenen om voor deze oplossing te kiezen kunnen zijn:

- Eenvoudig. Als het er alleen om gaat om vanuit de inhoud van de database webpagina's te genereren is dit de eenvoudigste methode.
- Lage aanschaffkosten.
- Lage leerkosten voor zover het gaat om databases die reeds in gebruik zijn en voorzien zijn van webfunctionaliteit.
- Als database en webcomponenten van dezelfde leverancier afkomstig zijn is het relatief eenvoudig om ze samen te laten werken.

Deze manier van werken is goedkoop en snel. Het is beperkt in de zin dat het alleen mogelijk is om de database inhoud snel te publiceren als webpagina's. Hiertoe hebben de databaseproducten de mogelijkheid gekregen om tekstfiles in een HTML-formaat te exporteren. Deze file kan vervolgens gepubliceerd worden. Nadeel is dat deze producten weinig bieden in de zin van opmaak van webpagina's. Een product als Access 97 biedt de mogelijkheid om in een aantal stappen de inhoud van een tabel beschikbaar te maken in HTML-formaat. Ook wordt de mogelijkheid geboden om een homepage mee te geven met een verwijzing naar de tabel(len). De wijze waarop homepage en tabellen worden weergegeven is een rechttoe rechtaan formaat waaraan niets gewijzigd kan worden. Alle instellingen kunnen in een file worden bewaard die gebruikt wordt bij het verdere beheer van de webpagina's.

HTML EDITORS MET DATABASE-FUNCTIONALITEIT

Voorbeelden van producten uit deze categorie zijn:

- Allaire HomeSite
- NetObjects Fusion
- Microsoft FrontPage 2002

Als voorbeeld van deze categorie van producten zal ik ingaan op Microsoft FrontPage 2000. FrontPage is één van de meest gebruikte tools om websites te bouwen.

SITE MANAGEMENT MET FP 2002 Het onderhoud van websites kan snel complex worden naarmate het aantal webpagina's toeneemt. Het is mogelijk om de website te verdelen in folders. In een folder staan alle bestanden en data die betrekking hebben op een bepaald onderwerp. Vervolgens kun je gebruik maken van verschillende soorten views. De beschikbare views zijn: page view, folders view, reports view, navigation

view, hyperlinks view en tasks view. FP 2002 beschikt over een find and replace functie waarmee redelijk geavanceerd termen kunnen worden gezocht en vervangen. Dit kan gebeuren op het niveau van een individuele webpagina, maar het kan ook gaan om de hele website of om een serie webpagina's.

Het onderhoud van hyperlinks is ook een niet te onderschatten taak. Een link is een verwijzing naar een ander locatie waar de gezochte inhoud aanwezig is. Door het veranderen van locaties (intern of extern) wordt de link onbruikbaar. De hyperlinks view laat zien hoe elke pagina is gelinkt en waar naar de link verwijst. Begrijpen hoe de webpagina's naar elkaar verwijzen (hoe de flow verloopt) en het beheren hiervan vormt een belangrijk onderdeel van effectief management van de site. FP beschikt over de Recalculating Hyperlinks mogelijkheid. Hiermee worden alle webpagina's van een FP-site gecheckt. Eventuele problemen kunnen automatisch worden opgelost. Het regelmatig gebruiken van deze optie kan dus een heleboel problemen opsporen en verhelpen.

EISEN AAN DE OMGANG Naarmate de omvang en de complexiteit van de site toenemen zijn ook meerdere personen bij de ontwikkeling ervan betrokken. Dit stelt weer nieuwe eisen aan het omgaan met de site. Denk hierbij bijvoorbeeld aan de mogelijkheid om de workflow te regelen. Binnen FP is het mogelijk om pagina's aan gebruikers toe te kennen (assign). Ook kan de review status worden vastgelegd. De mogelijkheid bestaat om daarnaast additionele statussen te benoemen in het workflow proces. Door gebruik te maken van de optie categorieën kunnen webpagina's verder worden geordend.

Behalve dat je te maken hebt met meerdere personen is het ook vaak zo dat meerdere personen bezig zijn om te werken aan dezelfde pagina. Hier gaat het om onderwerpen als het eigenaarschap van documenten, multi-user authoring en het vastleggen van gebruikersrechten.

De check in / check out functionaliteit garandeert dat iemand die een pagina opvraagt om er iets aan te veranderen hiermee elke andere persoon uitsluit om hetzelfde te doen. Slechts nadat de initiële gebruiker de pagina heeft ingecheckt is de pagina weer beschikbaar voor wijziging.

HTML-editor	Webdatabase technologie
<ul style="list-style-type: none"> • MS Frontpage • NetObjects Fusion • Allaire HomeSite 	<ul style="list-style-type: none"> • Active Server Pages (ASP) • Internal Cold Fusion, ASP, LiveWire • Cold Fusion, ASP

FIGUUR 2. HTML-editors met databasefunctionaliteit

LOKAAL MAKEN De Usage Analysis optie verzamelt informatie over de bezoekers aan de site en analyseert deze. Dit betekent dat het niet meer nodig is om logfiles te downloaden en die vervolgens met een aparte analysetool te bewerken. Je komt er nu sneller achter waar bezoekers wel/niet naar kijken en kan sneller inspelen op hun behoeften.

Je kunt een FP website lokaal maken of op je netwerk. Het is niet meer nodig om de site direct op de webserver te ontwikkelen. Je maakt gewoon een folder waarin alle benodigde files staan en zet die vervolgens over naar de website.

Met behulp van de Database Interface Wizard is het mogelijk om de inhoud van een database naar het web te publiceren. Hiermee worden formulieren en webpagina's gegenereerd. Gebruikers kunnen gerechtigd worden om records te verwijderen of te wijzigen.

Met behulp van de Database Interface Wizard is het mogelijk om de inhoud van een database naar het web te publiceren. Hiermee worden formulieren en webpagina's gegenereerd. Gebruikers kunnen gerechtigd worden om records te verwijderen of te wijzigen.

WEBDATABASE APPLICATIESERVERS

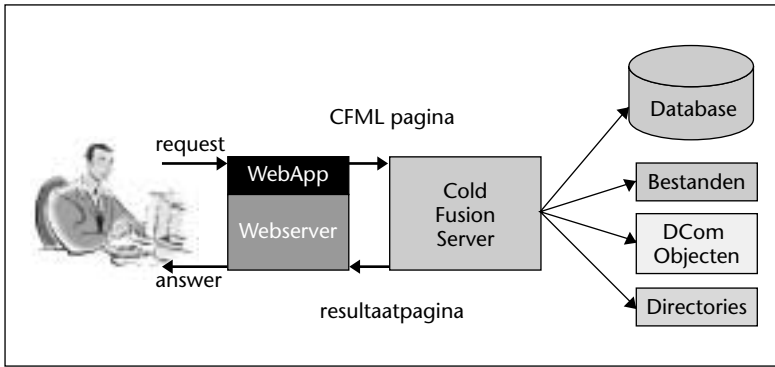
Tot de producten in deze categorie behoren:

- Tango
- Active Server Pages
- Macromedia ColdFusion server 5

MACROMEDIA Het verschil met de voorgaande categorie is dat een applicatieserver onderdeel uitmaakt van de oplossing. In het geval van Macromedia gaat het om de ColdFusion server. De ontwikkelomgeving bestaat uit Dreamweaver Ultradev en/of ColdFusion Studio.

ColdFusion is een platform voor webapplicaties. Het is een applicatieserver die browser, server, database en ontwikkelomgeving in zich verenigt. De ColdFusion Application Server werkt met webserver die onder Windows NT en Sun Solaris draaien. Webverzoeken komen binnen op de webserver. In de webpagina's staan tags die geschreven zijn in de CF Markup language (CFML). Dit is een tag-based programmeertaal. De betreffende pagina's worden doorgestuurd van de webserver naar de CF server. De CF server handelt op basis van de CFML activiteiten af. In de betreffende webpagina kunnen naast CFML ook andere instructies staan die bijvoorbeeld geschreven zijn in HTML, CGI, Javascript of Java. Alleen de CFML instructies worden door de CF Application Server behandeld. Instructies die niet in

Zelf een webomgeving 'from scratch' bouwen is alleen in laatste instantie het overwegen waard



FIGUUR 3. Webdatabases applicatieservers

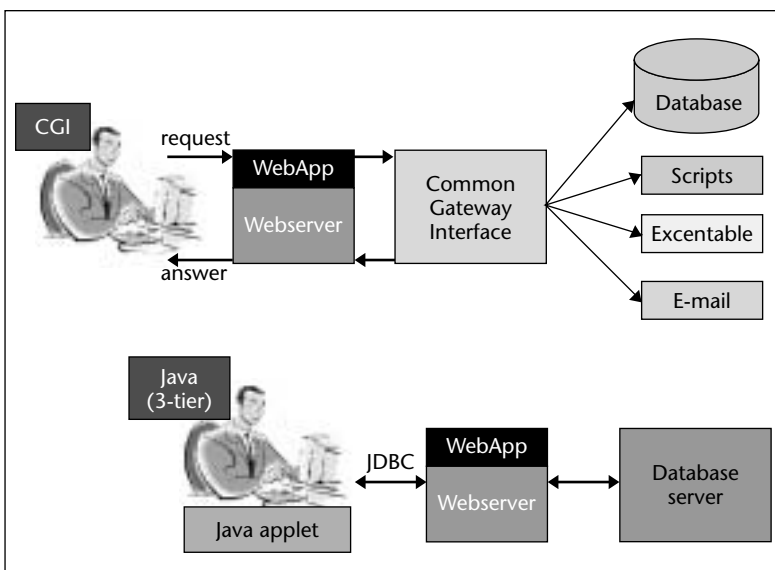
CFML staan worden gewoon doorgegeven. De CF applicatieserver beschouwt een database, DCOM objecten, directories, files, e-mail en webservers als externe functies. De resulterende pagina wordt door de CF server weer teruggegeven aan de webserver en vervolgens aan de gebruiker.

Met behulp van een wizard kunnen eenvoudige drill-down acties op een database worden gedaan. CF beschikt over templates waarmee een zoekpagina, een resultaatpagina (tabelvorm) en een detailpagina (om individuele records te bekijken) eenvoudig kunnen worden aangemaakt. Met behulp van een <CFQUERY> tag wordt een SQL-query naar de database gestuurd. Het resultaat van de database query wordt met behulp van <CFTABLE> in HTML-formaat gezet.

PROGRAMMEREN VAN WEBDATABASES

De belangrijkste redenen om zelf geen webdatabase te programmeren zijn:

- De hoeveelheid tijd die nodig is om een webdatabase van de grond af op te bouwen.



FIGUUR 4. Webdatabases programmeren

- De korte levenscyclus van webtechnologieën betekent dat de kans groot is dat een eenmaal gebouwde webdatabase in geval van een nieuwe versie niet meer werkt.
- Gezien het grote aantal componenten waar je mee te maken hebt (besturingssysteem, database, nieuwe technologieën) is het moeilijk om een 'goede' oplossing te vinden.

Bij het zelf bouwen van een webdatabase spelen CGI en Java een rol.

CGI

De Common Gateway Interface (CGI) is een gateway tussen de webserver en applicaties aanwezig op andere servers. Tot deze laatste kunnen we de databases, script files, executables etc rekenen. CGI is onderdeel van de HTTP-standaard. Het grote voordeel van CGI is dat het eenvoudig is en algemeen wordt ondersteund. CGI-programma's zijn geschreven in een programmeertaal als Perl of in een scripting taal. Het spreekt voor zich dat onderhoud en beheer zelf geregeld moeten worden. Wel zijn er modules beschikbaar die ingezet kunnen worden. Database Interface (DBI) is de naam van de api om databases te kunnen benaderen met behulp van Perl scripts. Elke database heeft een eigen Database Driver (DBD) nodig om het DBI-commando te vertalen voor de specifieke database. In het algemeen wordt mSQL vaak gebruikt als database voor een Perl/CGI web database applicatie. Voor hele grote toepassingen komen Oracle, Informix etc in aanmerking.

Java

In een two tier client/server model communiceert de Java applet door middel van JDBC met de database. De database moet hiervoor wel aanwezig zijn op de webserver (dus een gecombineerde web/databaseserver). In een three tier model moet er sprake zijn van een data access server (speciale applicatie) op de webserver. De Java applet kan nu via JDBC communiceren met de data access server. De data access server communiceert vervolgens met de database op de databaseserver. Resultaten worden ook weer via de data access server teruggegeven aan de client. Met behulp van JDBC en de JDBC-ODBC brug is het mogelijk om elke ODBC-compliant database te koppelen aan een Java frontend.

ENKELVOUDIGE ENGINE De ontwikkelingen ten aanzien van webdatabases staan niet stil. In een product als Aestiva HTML/OS vormen de database, ontwikkelomgeving en integratietools een enkelvoudige engine. Hierdoor ontstaat uiteraard een naadloze aansluiting tussen al deze onderdelen.

De voordelen van een HTML/OS database zijn volgens Aestiva:

- database en 4GL-omgeving zijn geïntegreerd;
- het gaat om een honderd procent webbased oplossing; indexering kan gebeuren door gebruik te maken van tags;
- geen aparte databaseserver nodig. Door het ontbreken van deze component kan de server ook niet down gaan en verbetert de betrouwbaarheid. Geen DBA nodig en lagere onderhoudskosten.
- database schema's zijn niet nodig. Joins tussen tabellen kunnen in HTML documenten worden gespecificeerd. Hierdoor wordt het definiëren en wijzigen van tabellen makkelijker.

SAMENVATTING Het ontwerpen en onderhouden van websites betekent dat de databaserwereld en de wereld van webprogramming bij elkaar komen. Verschillende leveranciers bieden verschillende producten en oplossingen aan om bedrijven hierbij te ondersteunen. In dit artikel zijn vier verschillende oplossingsrichtingen genoemd. Vanuit de database geleiden worden databaseproducten aangeboden die zijn voorzien van de mogelijkheid om snel en eenvoudig database-inhoud op het web te zetten. Vanuit de hoek van de HTML-editors is databasefunctionaliteit toegevoegd. Hier ligt de nadruk op het genereren van webpagina's. De mogelijkheden hiertoe zijn dan ook groter dan bij de eerste categorie. De database capabilities zijn minder groot. De webdatabase applicatieservers vormen een derde benaderingswijze. De applicatieserver en de ontwikkelomgeving staan centraal. De positie die wordt ingenomen is die tussen de webserver en de database(server) die beide als extern worden gezien. Tenslotte is het ook mogelijk om de hele omgeving (ontwikkelomgeving en database) te programmeren.

CONCLUSIES De ontwikkelingen op dit terrein staan niet stil. Naarmate meer bedrijven langer op het web aanwezig zijn (meer informatie, meer pagina's, meer gebruikers, meer ontwikkelaars) wordt de noodzaak van goed beheer en onderhoud evident. Door de ongelijke producten die door leveranciers worden aangeboden is het vaak onduidelijk welke oplossingsrichting in een gegeven situatie de voorkeur verdient. Inhoud vanuit een database op het web zetten zonder dat daar (veel) bewerkingen op hoeven plaats te vinden betekent dat van webfunctionaliteit voorziene databases de oplossing met de minste onderhoudsinspanning zijn. Staat het proces van het creëren van webinhoud centraal dan vormen HTML-editors voorzien van database functionaliteit de te prefereren oplossing. Bij de webdatabase applicatieservers staat de dynamiek van de webomgeving zelf centraal. In een voortdurend veranderende webomgeving is dit dan ook de ideale oplossing. Toepassen van andere oplossingen levert niet alleen minder goede oplossingen, maar ook meer onderhoud

Oplossingsrichting	Wanneer gebruiken?
Databases met webfunctionaliteit	Webinhoud publiceren vanuit de database zonder dat daar (veel) aan veranderd moet worden.
HTML-editors met databasefunctionaliteit	Content creatie proces staat centraal en is complex (veel inhoud, veel gebruikers, veel bewerkingen).
Webdatabase applicatieservers	Webomgeving staat central en is dynamisch. Aanpassingsvermogen van groot belang.
Webdatabases programmeren	Situaties die niet voldoen aan bovenstaande criteria. Bijvoorbeeld: beperkt en eenvoudig content creatie proces in een stabiele webomgeving (steeds dezelfde bronnen/doelen).

op. Zelf een webomgeving *from scratch* bouwen is alleen in situaties waarin alle andere oplossingen niet voldoen het overwegen waard. De onderhoudsinspanning is echter niet te onderschatten. De compleet onderhoudsvrije webomgeving bestaat (nog) niet, alhoewel verschillende softwareleveranciers zeggen hier druk aan te werken. Voorlopig blijft het dus: producten kiezen die passen bij de aard van de ontplooidde webactiviteiten.

Literatuur

- 'Microsoft FrontPage. Effective Site Management with Microsoft FrontPage 2002', White Paper, May 2001.
- 'Microsoft FrontPage. Using Electronic Commerce with Microsoft FrontPage 2002', White Paper, April 2001.
- 'Microsoft FrontPage version 2002. Product Guide', April 2001.
- 'Choosing a Database for Your Web Site', John Paul Ashenfelter, John Wiley & Sons, 1999.
- 'Maximizing the Return on Your Web Application Server Technology Investment', Macromedia, 2001.
- 'Macromedia ColdFusion Server 5 Feature Comparison Matrix', Macromedia, 2001.
- 'Macromedia ColdFusion 5 Performance Brief', Macromedia.

Drs. P.F.H. van der Linden
(Paul.PFH.vanderLinden@AtosOrigin.com) is senior consultant
Data Warehousing/BI bij Atos Origin.