

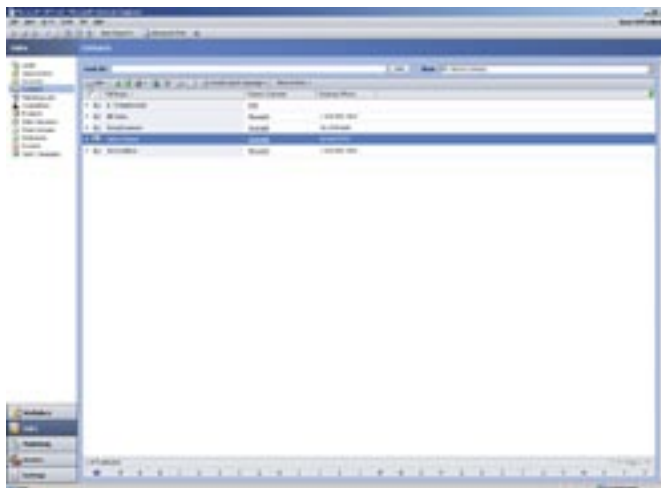
Microsoft Dynamics CRM 3.0

NIEUWE MOGELIJKHEDEN CRM UITGELICHT

Microsoft Dynamics Customer Relationship Management is gepositioneerd in het midmarket-segment naast CRM-pakketten als Super Office en SalesLogix. In de praktijk komt Microsoft Dynamics CRM ook regelmatig voor in het Enterprise-segment, waar Siebel en SAP CRM de scepter zwaaien. Begin december vorig jaar heeft Microsoft de nieuwste versie van zijn CRM-pakket op de markt gelanceerd: Microsoft Dynamics CRM 3.0. In dit artikel gaan de auteurs in op de mogelijkheden van de applicatie. Ook zullen de nieuwe mogelijkheden van versie 3.0 ten opzichte van 1.2 uitgelicht worden aan de hand van de uitbreidingsmogelijkheden van de applicatie.

Microsoft Dynamics CRM is een applicatie waarmee het CRM-proces wordt ondersteund. Dit proces is gericht op het bijhouden van gegevens over de klant. Naast adresgegevens worden ook de contactmomenten, wensen van de klant en verkoopinformatie bijgehouden. Door het bijhouden van deze gegevens kan een organisatie adequaat en correct reageren bij een volgend contactmoment en op deze manier de klant beter te woord staan. Hierdoor ontstaat een betere band tussen de organisatie en de klant. Microsoft's CRM-applicatie gaat nog een stap verder. Deze applicatie omvat drie modules: Sales, Service en Marketing. De sales-module neemt het beheer van klantgegevens op zich, evenals gegevens over concurrenten, producten en verkoopgegevens; zie afbeelding 1 voor de sales-module. De service-module is ontwikkeld voor servicedesks. Deze module herbergt de gebruikelijke taken van een servicedesk. Hieronder valt het registreren van incidenten, onderhouden van een knowledge base en het inrichten van een servicekalender. De marketingmodule is een nieuwe module die in de laatste versie is toegevoegd. Hiermee kan de gebruiker campagnes opstellen en uitvoeren. Deze kunnen uiteenlopen van het versturen van een eenvoudige informatiebrief tot een volwaardige marketingcampagne, rekeninghoudend met de responses van de geadresseerden.

Deze applicatie is een goed voorbeeld van een software factory, zoals is beschreven in het .NET-magazine van december 2004 door Edwin

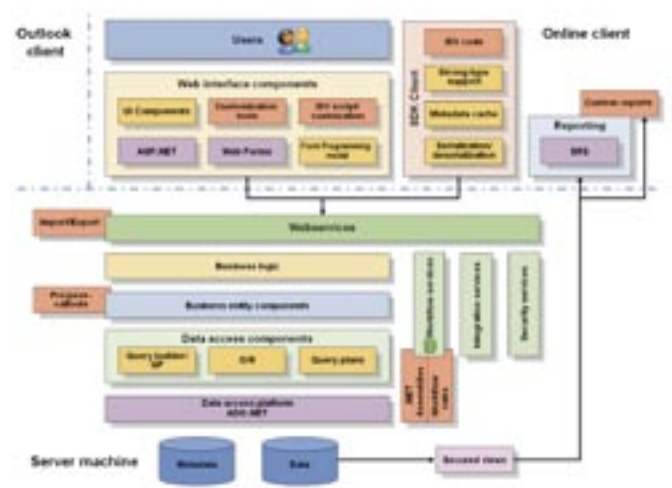


Afbeelding 1. Intro afbeelding

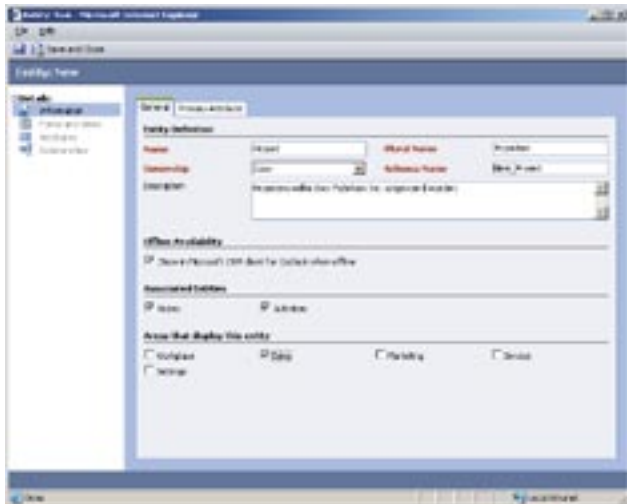
Jongsma. Dit houdt in dat de applicatie voor ongeveer 80% out-of-the-box te gebruiken is en 20% aan maatwerk bijgebouwd dient te worden. De meeste aanpassingen zijn zonder programmeerkennis door te voeren door gebruik te maken van de uitgebreide configuratiemogelijkheden van Microsoft Dynamics CRM 3.0. Indien men echter bedrijfsspecifieke eisen stelt die niet standaard in CRM zijn opgenomen, dan is het mogelijk om als programmeur maatwerk toe te voegen op diverse plaatsen in de applicatie. In dit artikel gaan wij in op de mogelijkheden van de applicatie voor zowel ontwikkelaars en lieden zonder kennis van programmeren.

Customization-tools

De CRM-applicatie kan direct na de installatie gebruikt worden. Het systeem voorziet de klant van een voorgedefinieerde set aan entiteiten en attributen. Een entiteit is een object, in de breedte zin van het woord, dat in CRM kan worden gebruikt. Denk hierbij aan contactpersonen, accounts, producten en zakelijke kansen. Deze meegeleverde entiteiten zijn gebaseerd op standaard werkprocessen en komen niet per definitie overeen met de werkprocessen, zoals deze gewenst zijn in de applicatie. De applicatie biedt de ontwikkelaar hiervoor een set met tools waarmee het systeem aangepast kan worden. Met deze tools is het mogelijk om zonder kennis van programmeren het systeem aan te passen aan de bedrijfsprocessen. Je kunt nieuwe attributen toevoegen, bestaande attributen



Afbeelding 2. Applicatie-architectuur Microsoft Dynamics CRM 3.0



Afbeelding 3. Toevoegen van een nieuwe entiteit.

wijzigen en met de nieuwe release ook attributen en zelfs nieuwe entiteiten toevoegen of verwijderen.

Bestaande entiteiten wijzigen

Waar in de vorige release de ontwikkelaar alleen attributen kon toevoegen en wijzigen, is de huidige set met veel nieuwe mogelijkheden uitgebreid. Zoals aangegeven kunnen nieuwe entiteiten worden aangemaakt, maar ook kunnen bestaande entiteiten hernoemd worden naar, voor de business, logische namen. Wordt in de organisatie niet over accounts maar van bedrijven gesproken, dan kan dit zonder problemen aangepast worden. Een nieuwe mogelijkheid, die zowel functioneel als technisch erg nuttig is, is het verwijderen van attributen en entiteiten. Alle metadata worden - uiteraard - opgeslagen in een Microsoft SQL Server-database. Dit leidt automatisch tot de beperking dat er maximaal 8000 karakters per regel kunnen worden opgeslagen. Dit leidde voorheen geregeld tot problemen, omdat CRM v1.2 alle attributen als Unicode in één tabel opslaat. Hierdoor wordt het maximum van 8000 karakters snel bereikt en kan er geen wijziging meer doorgevoerd worden. Dit is nu verholpen door een onderscheid in tabellen te maken voor de systeemeigen en de nieuw toegevoegde attributen. Daarnaast heeft de ontwikkelaar de mogelijkheid om velden te verwijderen.

Nieuwe entiteiten toevoegen

Behalve het wijzigen van bestaande entiteiten kan de programmeur nu ook nieuwe entiteiten toevoegen. De nieuw toegevoegde entiteiten worden volledig in het systeem geïntegreerd. Zo is het mogelijk om op nieuwe entiteiten te zoeken, de security in te stellen en deze te gebruiken in rapporten. De mogelijkheid om nieuwe entiteiten toe te voegen, zorgt ervoor dat de CRM-applicatie een breed platform is geworden. Ze treedt hiermee buiten de grenzen van een typische CRM-applicatie. De eenvoud van het toevoegen van een entiteit wordt weergegeven in afbeelding 3.

Relaties tussen entiteiten aanleggen

Door relaties te leggen tussen nieuwe en/of bestaande entiteiten, worden nieuwe mogelijkheden gecreëerd. In het CRM-systeem kunnen alleen één-op-veel relaties worden toegevoegd.

Activiteiten wijzigen

Naast de in versie 1.2 al aanwezige activiteitstypen 'task', 'e-mail', 'fax', 'letter', 'phone call' en 'appointment' zijn de typen 'service activity' en 'campaign response' toegevoegd. Al deze typen zijn in deze versie, zoals iedere andere entiteit, te wijzigen.

ISV script / Form customization

In CRM 1.2 was het al enigszins mogelijk om het een en ander te customizen. Er waren echter beperkingen in wat mogelijk was

```
// Haal het veld op welke het event afgevuurd heeft:
var currentField = event.srcElement;
// Valideer de content van het veld.
if(typeof(currentField) != "undefined" && currentField != null)
{
// Verwijder alle niet numerieke karakters
var tempCurrentField = currentField.DataValue.replace(/[^0-9]/g, "");
// Als het nummer een geldige lengte heeft, voer handelingen op uit
switch(tempCurrentField.length)
{
//Opmaak code wordt hierin geplaatst
}
}
}
```

Codevoorbeeld 1.

Telefoonnummercontrole

en ook daadwerkelijk werd toegestaan. In CRM 3.0 is op het vlak van ISV-scripting en form customization een hoop veranderd. De mogelijkheden met de ISV.config file (navigatieopbouw in CRM) en het customizen van de formulieren zijn uitgebreid.

Navigatiemodel

CRM 3.0 heeft een nieuw navigatiemodel. Om dit nieuwe model te kunnen ondersteunen heeft de organisation-entiteit een nieuwe sitemap gekregen die de navigatiestructuur definieert voor de applicatie. Hiermee kan je het systeem aanpassen, wanneer entiteiten of pagina's worden toegevoegd of gewijzigd in het systeem. Ook stelt het de eindgebruikers in staat hun 'Workplace'-navigatiestructuur aan te passen door een profiel te kiezen - in principe groepen van applicatiepagina's - dat gedefinieerd wordt in de sitemap. Met de sitemap kan je de gehele navigatiestructuur en de precieze weergave en ordering van pagina's beheren. Het is mogelijk om aangepaste entiteiten in de navigatiestructuur onder te brengen door gebruik te maken van de 'Custom Entities Area' in de webapplicatie.

De user-interface voor CRM 3.0 wordt opgebouwd uit de volgende elementen:

- Het sitemap xml-attribuut in de organization entity.
- Navigatietoevoegingen, die in de isv.config-file staan.
- Custom entiteitinformatie, die in de metadata-database van CRM staat.

De sitemap definieert de gebieden die in de webclient's navigatie-pane worden gepresenteerd. Hij beschrijft ook de Workplace-profielen, die in feite groepen van sub-areas zijn, die de gebruikers kunnen toevoegen aan of verwijderen van de workplace. Hiervoor maak je gebruik van de user settingpagina van het workplace-menu. De sitemap geeft geen configuratieondersteuning voor menu-links van formulieren in CRM. Ook maakt hij de ISV.config-file niet overbodig. Je kunt de ISV.config gewoon blijven gebruiken om bijvoorbeeld menu's en knoppen toe te voegen aan de applicatie. Menu's en knoppen kunnen in dit geval zelfs verwijzen naar externe webapplicaties die op deze wijze vanuit de applicatie kunnen worden benaderd. Een laatste handigheid is dat de sitemap kan worden geëxporteerd en geïmporteerd. Dit is erg handig bij het overzetten van aanpassingen tussen ontwikkel- en productiesystemen.

Form customization

Ondersteuning van IFrames. Het is nu mogelijk om gebruik te maken van IFrames en deze op te nemen in een CRM-formulier, waarbij je gebruikmaakt van de form editor in de webclient. IFrames kunnen worden gebruikt voor verschillende customizations, zoals:

- Het opnemen van een pagina van een Sharepoint-portal.
- Het opnemen van html-content zoals buttons, checkboxes, textboxes, enzovoort. Denk bijvoorbeeld aan mogelijkheden met custom forms in CRM.

- Het tonen van interessante websites in je CRM-applicatie zoals bedrijfswebsites, intranet, extranet, portals, enzovoort.
- De mogelijkheid een custom webapplication op te nemen in CRM.

Bij het gebruik van bovenstaande mogelijkheden kan rekening worden gehouden met de security, zoals het verbieden van cross frame scripting. Ook kan je eigenschappen van een IFrame zoals formatting en scrolling aanpassen.

Ondersteuning van read-only-velden. CRM 3.0 geeft bij het aanmaken of het wijzigen van een formulier de mogelijkheid om read-only fields te implementeren. Denk hierbij aan tekstvelden (textboxes), waarbij de gebruiker de getoonde informatie slechts kan lezen.

Form- en Field-events. Een zeer interessante functionaliteit is de aanwezigheid van events op formulier- en field-niveau.

- Form-events. Er zijn twee events aanwezig op formulierniveau, namelijk de OnLoad() en OnSave(). Het 'OnLoad()'-event vindt plaats op het moment dat het formulier is geladen. Dit event is een interessante toevoeging, omdat je hier logica kunt inbouwen over hoe het formulier getoond moet worden. Het kan ook gebruikt worden om bepaalde velden op read-only te zetten of zelfs te verbergen. Je kunt in de editor logica inbouwen in het 'OnLoad()'-event door middel van javascript. Het 'OnSave()' wordt afgevuurd zodra een gebruiker op de 'Save'- of 'Save and Close'-knop het formulier klikt, ongeacht of de data op het formulier wel of niet zijn gewijzigd. Het event is veelal geschikt voor het verifiëren van data door het implementeren van custom javascript-code, voordat ze naar de server worden gestuurd.
- Field-events. Op veldniveau is in de applicatie het 'OnChange()'-event toegevoegd. Het event wordt afgevuurd zodra de content van een veld is gewijzigd en de focus naar een ander veld gaat. Hiermee kan je de input in een veld valideren door middel van javascript-code en actie ondernemen indien noodzakelijk. Voor een toepassing met telefoonnummer controle, zie codevoorbeeld 1.

Door gebruik te maken van callouts, elders beschreven in dit artikel, kunnen ook codes worden uitgevoerd. Het verschil tussen de field-events en de callouts schuilen in het verschil tussen serverside en clientside codes. Bij field-events wordt Javascript-code direct uitgevoerd na het verliezen van de focus. Callouts zijn ASP.NET-codeden die pas worden aangeroepen op het moment dat de wijzigingen worden opgeslagen.

```
// Opzetten CRM Service
CrmService service = new CrmService();
service.Credentials = System.Net.CredentialCache.DefaultCredentials;

// Gebruikersgegevens opvragen
WhoAmIRequest userRequest = new WhoAmIRequest();
WhoAmIResponse user = (WhoAmIResponse) service.Execute(userRequest);

// Account aanmaken
account account = new account();

// Accountgegevens aanmaken
account.name = "Avanade";
account.ownerid = new Owner();
account.ownerid.Value = user.UserId;
account.ownerid.type = EntityName.systemuser.ToString();

// Account aanmaken in CRM
service.Create(account);
```

Codevoorbeeld 2.

Aanmaken account via SDK.

Verbeterde editor voor formulieren. In combinatie met de hiervoor genoemde functionaliteit is dit een van de laatste wijzigingen in de interface om alle bovenstaande items mogelijk te maken. Het editor-panel voor het aanmaken/wijzigen van formulieren is robuuster geworden.

Customizen van online helpfunctionaliteit

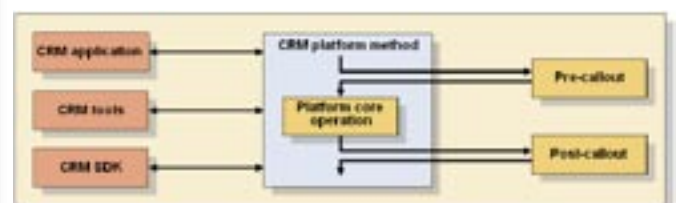
Nu vele items in de applicatie zijn te customizen, betekent dit ook dat een formulier of een pagina kunnen afwijken van de bestaande opbouw. Het is dan ook belangrijk dat dit soort wijzigingen gedocumenteerd kunnen worden. Het is in CRM 3.0 mogelijk een topic aan de helpfile toe te voegen, conform een eigen CRM-systeem. Door middel van een HTML-editor (Notepad, Word, Visual Studio, enzovoort) is het mogelijk topics aan te maken. Om de content te kunnen tonen in de CRM-help wordt gebruikgemaakt van een XML-file.

ISV-code

De applicatie heeft een bepaalde set aan functionaliteiten. Met deze mogelijkheden kan de basis van CRM in een bedrijf worden geïmplementeerd. Door middel van ISV-customization is de applicatie grotendeels aan de bedrijfsprocessen en regels aan te passen. Voor datgene wat daarna nog moet gebeuren, kan je zelf toevoegingen in de vorm van .NET-applicaties ontwikkelen. Deze assemblies kunnen bijvoorbeeld worden aangesproken door een menu-item of -knop die zijn toegevoegd in de ISV-customization of tijdens het uitvoeren van workflow-rules.

Vanuit de .NET-applicaties kan je communiceren met CRM via een tweetal webservices. Dit zijn de 'CrmService' en de 'MetadataService'. De mogelijkheden die de programmeur geboden wordt met deze webservices zijn eindeloos. De webservices maken gebruik van dynamisch gegenereerde WSDL-bestanden. De CRM-webservice verzorgt strongly typed toegang tot alle entiteiten, inclusief de custom, entiteiten en attributen. Het gegeven dat de CRM-objecten nu strongly typed zijn (voorheen werkte alles via XML), is een grote vooruitgang. Hierdoor kan je gebruikmaken van de functionaliteit die Visual Studio biedt, waaronder design- en compile time-errors en code-completion. Een van de voordelen van het gebruik XML in de vorige versie is dat er dynamisch door alle attributen heen kon worden gelopen. Door het gebruik van strongly typed objecten is deze manier van werken komen te vervallen. Dit is opgevangen door het nieuwe object 'DynamicEntity' toe te voegen. DynamicEntity is een class waarmee je tegen een instantie van een entiteit kunt programmeren, zonder dat de volledige definitie van de entiteit beschikbaar is in de WSDL.

De API-calls voor alle entiteiten maken gebruik van een enkele webservice, de CrmService. Deze is zo opgezet dat de oplossingen die op basis hiervan worden gemaakt goed schaalbaar zijn. Om een indruk te geven van het gemak in het gebruik van de webservice is codevoorbeeld 2 toegevoegd, waarin een account wordt aangemaakt. Bij de SDK van CRM 3.0 worden tal van volledige voorbeeldprojecten meegeleverd. De tweede webservice is de Metadata-webservice. Deze bevat methoden waarmee je de entiteiten in het systeem tot je beschikking krijgt. Ook kunnen deze worden gebruikt om een clientside cache op te bouwen. Deze data zijn read-only. De customization-tools in de applicatie zijn nodig voor het toevoegen van bijvoorbeeld entiteiten, attributen en formulieren.



Afbeelding 4. Callout-proces.

```
<callout.config version="2.0" xmlns="http://microsoft.com/mscrm/callout/">
  <callout entity="lead" event="PostCreate">
    <subscription assembly="FollowupCallout.dll"
      class="FollowupCallout.PostCallout">
      <postvalue>subject</postvalue>
    </subscription>
  </callout>
</callout.config>
```

Codevoorbeeld 3. XML-callout configuratiebestand

```
public class PostCallout : CrmCalloutBase
{
  public PostCallout()
  {
  }
  public override void PostCreate(CalloutUserContext userContext,
    CalloutEntityContext entityContext,
    string postImageEntityXml)
  {
    // Creëer een instantie van de webservice.
    CrmService service = new CrmService();
    service.Credentials = System.Net.CredentialCache.DefaultCredentials;

    // Creëer een follow up call.
    phonecall followupCall = new phonecall();
    followupCall.subject = "Call this lead";
    followupCall.regardingobjectid = new Lookup();
    followupCall.regardingobjectid.type = EntityName.lead.ToString();
    followupCall.regardingobjectid.Value = entityContext.InstanceId;

    // Set de owner
    followupCall.ownerid = new Owner();
    followupCall.ownerid.Value = userContext.UserId;
    followupCall.ownerid.type = EntityName.systemuser.ToString();

    // Sla de taak op in CRM
    service.Create(followupCall);
  }
}
```

Codevoorbeeld 4.

Telefoonactiviteit creatie via callout.



Afbeelding 5. Het rapport is toegevoegd aan de lijst met rapporten.

events, die door CRM afgevuurd worden. Zo zijn integraties met andere applicaties mogelijk of kunnen data van het item worden gewijzigd. Callouts zijn onderhevig geweest aan grootscheepse wijzigingen. Het gehele COM-gebaseerde model is op de schop genomen en het resultaat mag er wezen. De stappen die nodig zijn om een callout werkend te krijgen zijn:

- een XML-bestand aanmaken
- een .NET-assembly maken en
- beide bestanden in de callout-folder plaatsen.

Na deze drie stappen werkt de callout. Het oude model wordt nog wel ondersteund, maar wordt afgeraden en zal in de volgende versie verdwijnen.

Met de introductie van het nieuwe model zijn meteen een paar nieuwe mogelijkheden toegevoegd. Zo kan je tegenwoordig naast post-callouts ook gebruikmaken van pre-callouts. Hier kunnen wijzigingen in de data aangebracht worden, voordat het event door de server wordt verwerkt. De pre-callouts lopen synchroon met CRM. Dit houdt in dat het systeem wacht totdat de callout is afgerond. Dit proces wordt weergegeven in afbeelding 4. In de callout kan op basis van logica besloten worden om het event stop te zetten. Ook de set van events waarop kan worden ingehaakt is uitgebreid. Was dit voorheen alleen 'Create', 'Update' en 'Assign', nu worden de events 'SetState' en 'Merge' ondersteund.

Voorbeeld

In dit voorbeeld zal, na het aanmaken van een lead, een activiteit aangemaakt worden van het type telefoongesprek. De eerste stap is het maken van een xml-bestand met de naam 'callout.config.xml'. Zoals in codevoorbeeld 3 is te zien, is de inhoud van dit bestand eenvoudig, maar doeltreffend. Na het xml-bestand wordt de .NET-assembly gemaakt. Codevoorbeeld 4 creëert een telefoonactiviteit die is gekoppeld aan de lead. Als de DLL van de code uit het codevoorbeeld en het xml-bestand in de folder 'crm-installatie-dir\Server\bin\assembly\' wordt geplaatst, zal de callout functioneren.

Workflow-services

Work-flow (de ~ (m.))

1. de volgorde waarin taken uitgevoerd worden. Het is een achtergrondproces dat een lijst met administratieve taken verricht, gebaseerd op de criteria/regels van een gebruiker. Ook specificeert het de procedure die geassocieerd wordt met een actie.

CRM wordt geleverd met de applicatie 'Workflow Manager'. Deze applicatie regelt de workflow-services in de CRM-webapplicatie. Met dit onderdeel kan je de workflow- en sales-processen in CRM aanpassen en automatiseren.

Workflow-rules

CRM 1.2 kende de tool al, maar in CRM 3.0 is de Workflow Manager verbeterd en uitgebreid naar de wensen van de huidige CRM-gebruikers. Er zijn vier events waarop een workflow-rule toegepast kan worden, namelijk:

- Manual. Vanuit de CRM-applicatie handmatig koppelen van rule aan entiteit
- Create. Bij het aanmaken van een instantie van een entiteit

Import/export

Sinds de introductie van de nieuwe CRM-versie kan de import, export en het publiceren van het customizations-schema via de grafische interface verlopen. Voor deze taken is de Deployment Manager niet meer nodig. Een groot voordeel is dat geen herstart van de webserver meer vereist is na het publiceren van de customizations. Het schema wordt ook naar de webservices van de SDK gepubliceerd, zodat de programmeurs na het verversen van de WSDL-file, direct gebruik kunnen maken van de nieuwe toegevoegde waarden. De mogelijkheid om custom-entiteiten toe te voegen aan het schema levert geen probleem op voor de export- en importmogelijkheden van de applicatie. Deze gaat hier zonder problemen mee om. Ook worden gedeeltelijke in- en exports ondersteund. Zo hoeft niet de gehele set aan customizations geëxporteerd te worden, maar kan ook gekozen worden voor een enkele systeem- of custom-entiteit. De geëxporteerde schemabestanden kan je nu ook transporteren naar anderstalige systemen. Dit stelt grote organisaties in staat verschillende CRM-implementaties gebruik te laten maken van hetzelfde schema bij meer vestigingen van het bedrijf. Aangezien er op één CRM-database meer webfront-ends gekoppeld kunnen worden, kan op deze wijze een meertalige implementatie worden gerealiseerd.

Pre/post-callouts

Een callout is logica die toegevoegd kan worden aan verschillende

- Change-status. Bij het wijzigen van de status van een instantie van een entiteit en
- Assign. Het toewijzen van iets aan een entiteit.

Hieronder volgt een opsomming van mogelijkheden met de Workflow Manager.

Workflow Manager User Interface. De nieuwe Workflow Manager user-interface bevat een lijst met de huidige workflow-rules met relevante informatie zoals de eigenaar en de businessunit voor elke rule. Ook heeft de menubalk een 'Actions'-onderdeel dat gebruikt kan worden om een specifieke taak te specificeren. Denk hierbij aan het activeren van een geselecteerde rule.

Workflow-rules en activiteiten in CRM. In CRM 3.0 kan de gebruiker rules koppelen aan activiteiten. Hierdoor is het mogelijk een rule toe te voegen aan een e-mailactiviteit die in CRM is gespecificeerd, zodat de e-mail naar een bepaalde groep van klanten wordt gestuurd.

Dynamische data in velden, activiteiten, notities en e-mails. Het is mogelijk om dynamische datavelden in het onderwerp en de beschrijving van een e-mail, activiteit of een notitie te plaatsen. Voorbeeld: als CRM-gebruiker heb je een nieuw contactpersoon toegevoegd aan het systeem. Automatisch wordt een notitie voor je aangemaakt, waarin vermeld staat dat je nog even contact moet opnemen met de contactpersoon. Door het gebruik van dynamische velden worden de naam en telefoonnummer van de contactpersoon in de notitie vermeld.

Toepassen van Workflow-rules aan custom-entities en custom-relationships. (Voor omschrijving entiteiten, zie Customization Tools). Met de nieuwe mogelijkheid om custom-entiteiten aan te maken in de applicatie, is het vanzelfsprekend dat ook de workflow-rules toegepast kunnen worden. De custom-entiteiten zijn zichtbaar in de workflow manager en je kunt hierop workflow-rules toepassen. De werking en mogelijkheden zijn dezelfde als voor een normale entiteit. Zo is het mogelijk om relaties te leggen tussen systeementiteiten en custom-entiteiten. Ook deze relaties worden weergegeven in de workflow manager.

Gebruik maken van Assembly Date en Time Acties in een rule. Er is een standaard assembly aanwezig in CRM die de mogelijkheid geeft om gebruik te maken van de datum/tijd- functionaliteit via een externe assembly.

.Net-assemblies

Een interessante nieuwe functionaliteit is de mogelijkheid externe assemblies te programmeren en deze te gebruiken in CRM 3.0. Dit kan door een functie uit een externe assembly aan te roepen in een workflow-rule en deze iets te laten uitvoeren tijdens de workflow. Vanuit deze assembly kan eenvoudig ISV-code worden toegevoegd. De mogelijkheden zijn in principe onbeperkt.

CRM-rapporten

CRM biedt goede mogelijkheden om rapportages te genereren op basis van gegevens in het systeem. Er wordt een scala aan rapportages meegeleverd met de installatie van CRM en additionele rapporten zijn eenvoudig toe te voegen. Waar versie 1.2 nog gebruikmaakte van Crystal Reports, versie 3.0 wordt standaard geleverd met SQL Reporting Services. Een praktische zet, omdat op deze wijze naar een volledige integratie van CRM met Microsoft-producten wordt gestreefd. Hieronder volgt een overzicht van de voordelen van Reporting Services boven Crystal Reports:

- Gebruik van Microsoft's eigen Reporting Engine, waardoor CRM een volledig Microsoft-pakket is geworden.
- Reporting Services integreert naadloos met Visual Studio, waardoor Visual Studio-development mogelijk is. Je kunt vanuit Visual Studio een rapport ontwikkelen, builden en tevens deployen.
- SQL Reporting Services is gratis als een SQL-licentie aanwezig is.
- Exportmogelijkheden. Microsoft's Reporting Services biedt meer exportmogelijkheden dan Crystal Reports. De formaten

die worden ondersteund zijn: TIFF, PDF, Web Archive, Excel, XML, CSV en HTML.

- Security. Doordat de security afgehandeld wordt door Microsoft SQL Server, wordt de informatie die aan de ingelogde CRM-gebruiker wordt getoond, gepersonaliseerd.

Toepassing

Het toevoegen van een rapport aan CRM is vrij eenvoudig. Na installatie van de Reporting Services Report Designer kan vanuit Visual Studio een nieuw rapport worden opgesteld. Hierbij wordt vanuit het menu gekozen voor het Report Project van de Business Intelligence Projects. Visual Studio opent vervolgens een nieuw project. Het project bevat twee folders, namelijk Data Source en Reports. Data Source is de folder die gebruikt wordt om de connectie-informatie naar de database vast te leggen. Reports is de folder waarin de rapporten komen die de programmeur maakt. Aan de hand hiervan wordt begonnen met het toevoegen van een rapport aan het project. Het is voor CRM niet nodig om een datasource aan te maken, omdat deze al in de reportmanager van CRM gedefinieerd is. Nadat het rapport toegevoegd is aan het project, kan inhoudelijk aan het rapport worden begonnen.

De eerste stap die wordt genomen bij het ontwikkelen van een rapport is het kiezen van de data die getoond zullen worden in het rapport. Voor een CRM-rapport is het aan te raden om gebruik te maken van de 'filtered views'. Een filtered view is een view waarin diverse tabellen zijn samengevoegd om alle data vanuit één view te kunnen raadplegen. Ook wordt vanuit een filtered view bepaald welke records getoond mogen worden aan de ingelogde CRM-gebruiker. Qua security hoeft er dus niets gedaan te worden voor het filteren van data. Dit houdt in dat het mogelijk is een telefoonlijst te genereren van de contactpersonen die de ingelogde gebruiker mag zien, door middel van een query als 'SELECT firstname, lastname, telephone1 FROM FilteredContact'.

Nadat de data zijn bepaald, kan vervolgens in de lay-outtab gewerkt worden aan de vormgeving en weergave van het rapport. De velden van de dataset zijn direct beschikbaar en kunnen op het rapport worden gesleept. Zodra het rapport afgerond is, kan het Report-bestand (*.rdl) in CRM geplaatst worden. Dit wordt bewerkstelligd door in de CRM-applicatie naar de Workplace te gaan en op de link Reports te klikken. Er wordt dan een overzicht gegeven van de rapporten. Vanuit dit venster kan vervolgens een rapport toegevoegd worden aan de applicatie. Nadat het rapport is toegevoegd, wordt deze in de lijst van rapporten weergegeven; zie afbeelding 5.

Toepasbaarheid

Naar aanleiding van bovenstaande informatie kunnen we concluderen dat Microsoft Dynamics CRM met versie 3.0 een volwassen product is geworden met tal van mogelijkheden op het gebied van sales, service en marketing. Het is dan ook een interessante applicatie die gemakkelijk te plaatsen is in vrijwel iedere organisatie.

Ronald Lemmen en Sahin Ozdemir zijn beiden senior programmer bij **Avanade** (www.avanade.com), een samenwerkingsverband tussen Microsoft en Accenture. Zij zijn te bereiken op ronald@avanade.com en sahino@avanade.com.

Referenties

CDM 3.0 SDK: www.microsoft.com/dynamics/crm/using/downloads/
www.microsoft.com/dynamics/crm/
ronaldlemmen.blogspot.com/
www.avanade.com
partner.microsoft.com/US/productssolutions/business/businesscrm/