

De cloud is bepaald geen rocketscience. Als je met de Microsoft-technologie overweg kunt, zul je ook weinig moeite hebben met de technieken die voor Azure worden gebruikt. Maar het is even wennen. Bij Macaw Business Solutions deden ze dat onlangs in de vorm van een RAD-race. De conclusie: leuker, beter en zinvoller dan een taaie training buiten de deur.

‘The cloud is just another database’

Spelen met Azure in belang van de klant

Natúrlijk is het kantoor veel te klein. Het is behelpen met de beschikbare ruimte, maar de sfeer is er niet minder om. Zoals dat hoort voor een System Integrator, CRM'er en BI'er werken de meeste medewerkers bij de klanten. Maar nu zijn ze allemaal binnen bij de vestiging van Macaw in Barendrecht, teruggeroepen om deel te nemen aan de 'MBS Bootcamp 2011'. Het doel? In een competitieve sfeer de mogelijkheden van de cloudstack van Microsoft (beter) leren kennen en daardoor straks de klant beter kunnen adviseren. Het gaat nadrukkelijk niet om koste-wat-kost cloudoplossingen te verkopen, maar om weloverwogen oplossingen te kunnen bieden.

Om dit te kunnen bereiken moeten de medewerkers goed op de hoogte zijn van het aanbod in de markt en de mogelijkheden die dit biedt. Macaw heeft vanuit de disciplines– Software Integration (SI), Customer Relationship Management (CRM) en Business Intelligence (BI) een opdracht samengesteld, die de medewerkers in groepen van vijf/zes in één dag moeten vervullen.

De opdracht

De opdracht luidt: Zet een administratie op voor de interne beheer afdeling, waarbij per medewerker wordt bijgehouden wie welke auto, laptop(s) en telefoon(s) heeft, inclusief aanvullende informatie zoals soort gebruiker, basisvergoeding, serienummers, softwareversies, etc.

De RAD-race begint met een algemene instructie over SI, CRM en BI. Vervolgens wordt uiteengezet wat de 'klant' verwacht en kan het modelleren programmeerwerk beginnen. Tussendoor zijn er momenten, waarop de teams de 'klant', c.q. de

jury, nog vragen kunnen stellen over de gewenste uitvoering van de oplossing. "Op deze manier doet de hele groep kennis op over het werken met Cloud Services. Ze leren zo van elkaar, de CRM'ers van de BI'ers, de BI'ers van de SI'ers, enzovoort", vertelt Jack Klaassen, directeur Macaw Business Solutions (MBS) en één van de organisatoren van het Bootcamp.

Jack en zijn collega's Mark de Haan (principal CRM consultant), Dennis Rosenbrand (principal BI consultant) en Michel Heijman (SI architect) bereidden tevoren ieder vanuit hun eigen vakgebied de opdracht voor en stelden de eisen op, waaraan de oplossingen van de teams moeten voldoen.

De checklist voor de jurering ligt er niet om: Hoe zijn de teams omgegaan met projectmanagement? Zijn er momenten op de dag gepland om de voortgang door te nemen? Is er afgestemd met de business owner? Is er een rolverdeling gemaakt? Deze algemene aspecten worden aan het eind van de dag meegenomen in de beoordeling van de jury.

Daarnaast worden de verschillende technieken beoordeeld. Voor BI wordt gekeken naar aspecten als: Is gebruik gemaakt van een snapshot oplossing? Is de basisoplossing zelf gemaakt of gekocht? Zijn niet-BI'ers betrokken geweest bij het ontwikkelen van de oplossing? Zijn de gevraagde dashboards opgeleverd? Is er een alternatieve oplossing?

Op het vlak van System Integration wordt gekeken naar: Is gebruik gemaakt van Azure? Is de basisoplossing zelf gemaakt of gekocht? Is er documentatie van de oplossing? En voor CRM: Is er naast de gevraagde entiteiten verfraaiing aangebracht? Is de basisoplossing zelf gemaakt of gekocht? Is er voor de entiteit contactpersonen een apart formu-



Robert de Ruiter
is hoofdredacteur
van Release.

lier voor assetmanagement gemaakt? Hoe ziet de oplossing er grafisch uit?

Enthousiast

De deelnemers zijn zonder uitzondering enthousiast over het Bootcamp. Zowel vakinhoudelijk als sociaal vinden ze het een prima initiatief. Ofwel: “Je leert van elkaar en je ziet al je collega’s weer eens”. “Grappig is ook dat ze ons een andere discipline hebben gegeven dan waar we in het dagelijkse leven mee bezig zijn. Normaliter doe ik SI, maar nu moet ik me met CRM bezighouden. Ik zit hier geloof ik wat te stuntelen, maar met zijn hulp (wijzend op een collega) kom ik er wel uit”.

De hele sfeer in het kantoor is er één van enthousiasme en gezelligheid. Van broodjes, fris en fruit. Van secretaresses en pr-medewerksters die hun collega’s verwennen. Van hard werken aan de cloudoplossing voor de klant. En van spanning, want aan het einde van de dag valt er ook nog een ‘mooie prijs’ te winnen, zo wordt in het vooruitzicht gesteld.

Dan is het moment van de eerste jury-ronde daar. Van de zes teams moeten er in dit stadium drie afvallen. Omwille van de tijd is de jury - die bestaat uit de eerder genoemde consultants, aangevuld met directeur operations Roel Bergsma, peoplemanager Karen Welter, marketeer Rachelle Tunk en de auteur van dit artikel – in drie groepen opgesplitst. Die beoordelen ieder twee teams, waarvan er één doorgaat en één afvalt. In de eindronde zullen de drie beste teams, nadat ze nog een half uur de tijd hebben gekregen om hun oplossing te verfinaaien, deze aan de voltallige jury presenteren.

Van de twee teams, die hun oplossing voorleggen aan de jury waarin de auteur zitting heeft, heeft er één geen kans gezien binnen de gestelde tijd een kant-en-klare proof of concept (POC) op te leveren. Maar veel onderdelen zijn gereed, waardoor van een goede start kan worden gesproken. Het data-model ziet er goed uit. Het team heeft niets gekocht om tot een snellere oplossing te komen en de presentatie is fris.

Het andere team heeft een werkende oplossing, maar zou iets meer tijd willen hebben om het voor de klant mooier te maken. Op verzoek van de klant staan de basisdata on-premise en is een schaalbare applicatie op Azure gebouwd. De analyse vindt on-premise plaats met data vanaf Azure. Het team vond met name de software integration ‘heel heftig’.

Weinig verschil

De jury beslist dat de teams 1, 4 en 5 doorgaan voor de laatste ronde. Zij krijgen nog een half uur om de details af te werken of door te bouwen. Daarbij mogen ze ook mensen van andere groepen inzetten, wat nauwelijks lijkt te gebeuren. In de ‘elevator pitch’ krijgen ze vijf minuten om de oplossing te ‘verkopen’ aan de jury en alle mededingers in

de bootcamp.

De jury is het snel eens, unaniem zelfs. Team 1 van Aarjan Meirink, Antoon Tuijl, Dave Ruijter, Erik Proost en Floor Verheul gaan met de eer strijken, mede op grond van de overtuigende presentatie van Floor en Dave.

Bij de prijsuitreiking stelt Roel Bergsma dat hij “ongelooflijk onder de indruk” is van wat de teams in vijf uur hebben kunnen doen. Enkele teams hebben naast de basisopdracht ook extra functionaliteit kunnen inbouwen, zoals een overzicht van de brandstofkosten en het gemiddelde verbruik per werknemer. Met name in het laatste half uur hebben de drie teams kans gezien razendsnel functionaliteit te kunnen toevoegen, zowel on-premise als op Windows Azure.

De oplossing

De oplossing bestaat uit de volgende onderdelen, opgesplitst per discipline:

- System Integration o Koppeling van on-premise data met Microsoft Dynamics CRM Online o Koppeling van Microsoft Dynamics CRM Online data met ODS (Operational Data Store)
- Customer Relationship Management o Microsoft Dynamics CRM Online omgeving, met custom entiteiten voor Laptops, Telefoons, Auto’s en Brandstofverbruik (dit laatste is een toevoeging van het team) o Importeren van Excel documenten met gegevens over laptops, telefoons, auto’s en het brandstofverbruik o Ook door ons team toegevoegd: Operational BI Dashboard in CRM
- Business Intelligence o Operational Data Store in Microsoft SQL Azure Database o Analyse-omgeving in PowerPivot for Excel o Rapportage in Microsoft SQL Azure Reporting

System Integration

De System Integration oplossing bestaat uit twee onderdelen, het uitlezen van het on-premise systeem - waar de gegevens van de medewerkers zijn opgeslagen - naar CRM, en het vullen van het ODS vanuit CRM.

Het uitlezen van de gegevens uit het on-premise systeem gebeurt via een gateway, deze bevat methodes om in te kunnen loggen en gegevens op te halen. Vervolgens moeten deze gegevens naar CRM geschreven worden, ook dit gebeurt via een gateway, maar in dit geval een CRM gateway.

Om gegevens naar CRM te kunnen exporteren, moet er een vertaling worden gemaakt tussen de entiteiten die in de SI oplossing gehanteerd worden en de entiteiten die in het CRM systeem zijn aangemaakt. We hebben entiteiten aangemaakt voor o.a. medewerkers, telefoons, auto’s, brandstofverbruik en laptops. Vervolgens moeten er in de CRM gateway functies worden aangemaakt die de aangemaakte entiteiten vertalen naar de entiteiten zoals deze in CRM gedefinieerd zijn. Hierbij wordt een “Employ-

‘Ongelooflijk onder de indruk van wat teams in vijf uur kunnen doen’.

**In een half
uur hebben
drie teams
razendsnel
functies
toegevoegd.**

ee" entiteit welke uit het on-premise systeem komt, gemapt naar een "Contact" entiteit in CRM. Als voorbeeld kunnen we het uitlezen van werknemers gebruiken, we lezen eerst de gegevens uit het on-premise systeem en vullen een lijst met Employees met deze gegevens, zie onderstaand codefragment:

```
public List<Employee> GetEmployees()
{
    var retVal = new List<Employee>();
    var thisConnection = CreateConnection();
    SqlDataReader thisReader =
    ExecuteRead(thisConnection, "SELECT
    [WorkgroupName], [FirstName], [LastName],
    [MiddleName], [UserDefinedResourceId] FROM
    [Changepoint].[dbo].[DS_Resource]");
    while (thisReader.Read())
    {
        retVal.Add(new Employee
        {
            //new_medewerkernummer
            EmployeeID =
            thisReader["UserDefinedResourceId"].ToString(),
            FirstName =
            thisReader["FirstName"].ToString(),
            LastName =
            thisReader["LastName"].ToString(),
            SolutionCenter =
            thisReader["WorkgroupName"].ToString()
        });
    }
    thisReader.Close();
    thisConnection.Close();
    return retVal;
}
```

Vervolgens wordt deze lijst doorlopen en per Employee wordt er een vertaling gemaakt naar de Contact entiteit in CRM en het contact wordt opgeslagen in CRM, zie onderstaand codefragment:

```
public Guid CreateNewContact(string firstname,
string lastname, int employeenumber, string
solutionCenter)
{
    var contact = new Entity(ContactColumns.
    EntityName);
    contact[ContactColumns.Firstname] =
    firstname;
    contact[ContactColumns.Lastname] =
    lastname;
    contact[ContactColumns.Medewerkernummer] =
    employeenumber;
    contact[ContactColumns.Solutioncenter] =
    solutionCenter;
    return Service.Create(contact);
}
```

Op dit moment hebben we de medewerkers uit het on-premise systeem in het CRM systeem staan en zijn de gegevens daarmee in de cloud.

De volgende System Integration taak voor het team was het vullen van de Operational Data Store met de medewerkergegevens uit CRM, incl. laptops, telefoons, auto's en het brandstofverbruik.

Dit hebben we gedaan door een gateway te maken voor de Operational Data Store (ODS), deze gateway bevat functies om entiteiten in het ODS toe te voegen. Daarbij moet rekening gehouden worden dat er ook logica bij moet zitten om te controleren of de entiteit reeds aanwezig is, om duplicaten te voorko-

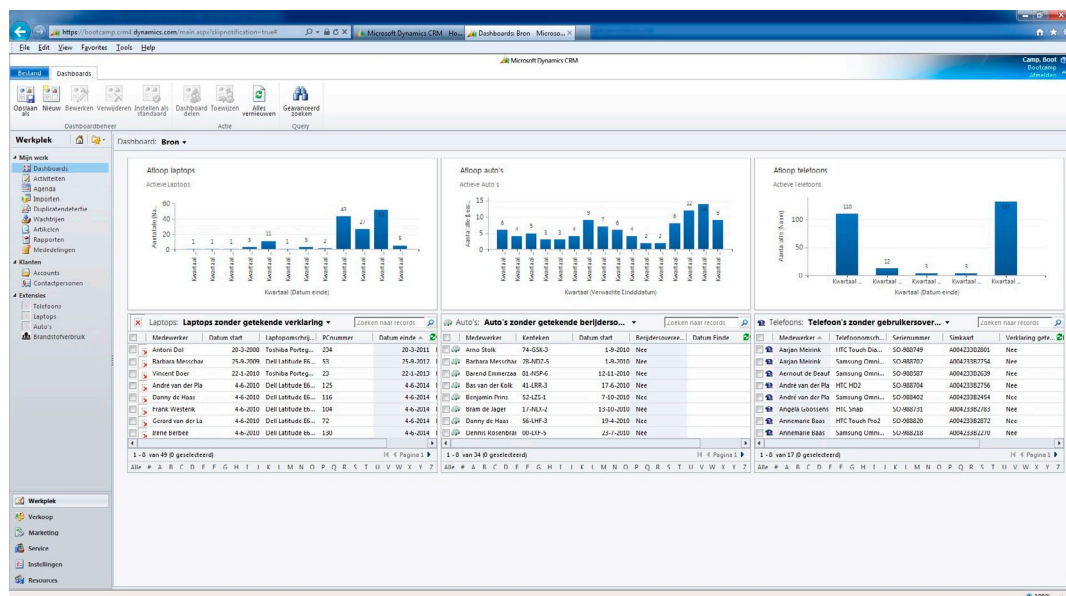
men. Aangezien het ODS een SQL Azure database betreft, hebben we gebruik gemaakt van de functionaliteit uit de System.Data.SqlClient namespace in het .NET Framework.

Als voorbeeld kunnen we het inlezen van medewerkers naar het ODS gebruiken. In de eerste plaats moeten de medewerkers uit het CRM systeem gelezen worden, dit doen we weer met de CRM gateway, deze bevat een functie om de Contact entiteiten uit te lezen, zie onderstaand codefragment:

```
public List<Contact> GetContacts()
{
    RetrieveMultipleRequest request =
    RequestMessageCreator.CreateContactRequest();
    var response =
    (RetrieveMultipleResponse)Service.Execute(request);
    return
    response.EntityCollection.Entities.Select(
    entity =>
    new Contact
    {
        Employeenumber =
        EntityAttributes.GetIntFromEntity(entity,
        ContactColumns.Medewerkernummer),
        Firstname =
        EntityAttributes.GetStringValueFromEntity
        (entity, ContactColumns.Firstname),
        Lastname =
        EntityAttributes.GetStringValueFromEntity(entity,
        ContactColumns.Lastname),
        Id = EntityAttributes.
        GetGuidValueFromEntity(entity, ContactColumns.Id),
        SolutionCenter =
        EntityAttributes.GetStringValueFromEntity
        (entity, ContactColumns.Solutioncenter)
    }).ToList();
}
```

Nu we de CRM Contacten hebben, kunnen we per contact deze naar het ODS exporteren, wat we doen via een Insert functie in de ODS gateway, deze insert functie gaat eerst na of de medewerker bestaat in het ODS. De functie LookupEmployee gaat na of er reeds een medewerker bestaat met het gegeven Employeenumber, deze krijgt de GUID van de bestaande medewerker terug indien de medewerker bestaat in het ODS, zie onderstaand code fragment:

```
public void InsertEmployee(Contact contact)
{
    SqlConnection connection = null;
    try
    {
        // Controleer voor bestaande Medewerker in ODS
        Guid medewerkerId = LookupEmployee(contact.
        Employeenumber);
        if (medewerkerId == Guid.Empty)
        {
            // Medewerker bestaat nog niet.
            connection = CreateConnection();
            ExecuteInsert(connection, "INSERT INTO [Team1].[dbo].
            [Medewerker] ([Naam],[Medewerkernummer]
            ,[SolutionCenter]) VALUES ('" + contact.Firstname + " "
            + contact.Lastname + "', '" + contact.Employeenumber +
            "', '" + contact.SolutionCenter + "')");
        }
    }
    catch (Exception ex)
    {
        Trace.WriteLine(ex);
    }
    finally
    {
        if (connection != null) connection.
        Close();
    }
}
```



Een aantal grafieken en tabellen die onder andere inzicht geven in het aantal laptops en telefoons dat de komende periode zal aflopen.

}

Nadat de gegevens uit CRM naar het ODS zijn geschreven, zijn we klaar om de gegevens te gebruiken in PowerPivot for Excel en de SQL Azure rapportages.

CRM Online

Een belangrijk onderdeel van de gehele oplossing is de Microsoft Dynamics CRM Online omgeving. Deze bevat standaard al vele entiteiten, waarvan we de Contacts vanuit een on-premise systeem vulden. Daarnaast moest er een aantal custom entities gemaakt worden, om de laptops, telefoons, auto's en brandstofverbruik te kunnen beheren.

Na een korte introductie les door Mark de Haan, waarin een vliegende start werd gemaakt, waren al vrij snel de benodigde entiteiten voor de basisopdracht gereed. Het duurde even voordat de medewerkergegevens uit het on-premise systeem geladen konden worden, maar daarna was het mogelijk om de aangeleverde Excel documenten met laptop-, telefoon- en autogegevens te importeren. Deze hadden uiteraard allemaal een relatie met de Medewerker entiteit.

Vervolgens kon de custom entiteit voor het brandstofverbruik aangemaakt worden, waarna hiervan de gegevens vanuit een CSV bestand geïmporteerd konden worden. Na een paar aanpassingen op de data vanuit het bijbehorende formulier was ook dit gereed.

Nadat alle entiteiten correct gevuld waren zijn we gestart met het maken van een Operational BI Dashboard, binnen CRM Online. Dit dashboard gaf de klant een idee van de dashboard en rapportage mogelijkheden voor de gebruiker van de CRM omgeving. We kozen ervoor om een aantal grafie-

ken en tabellen te tonen die o.a. inzicht gaf in het aantal laptops en telefoons dat zou aflopen in de komende periode.

Business Intelligence

Als toevoeging op de casus hebben we het brandstofverbruik van iedere medewerker in de oplossing opgenomen. Dit paste mooi bij de entiteit Auto en je kunt hier uitstekende BI analyses op loslaten.

Voor een aantal medewerkers werd een CSV bestand met verbruiksgegevens gemaakt en in CRM geïmporteerd. We hebben aan de klant voorgesteld dat iedere medewerker een dergelijk bestand zelf zou kunnen downloaden van de website van de leasemaatschappij (of zelf zou kunnen aanmaken indien het geen leaseauto betreft), waarna de medewerker deze op zijn of haar MySite in SharePoint plaatst. Vervolgens worden de bestanden automatisch geïmporteerd, waardoor er geen tussenkomst van een stafmedewerker nodig is.

De verbruiksgegevens in CRM Online worden vervolgens doorgelinkt naar een Operational Data Store, samen met de overige CRM entiteiten, waardoor het - naast het beschikbaar stellen van standaardrapportages - ook mogelijk werd om (ad-hoc) BI analyses op de gegevens uit te voeren.

Het ODS werd gebouwd in een SQL Azure database, het lukte ons niet om binnen de geboden tijd een multidimensionaal model (star- of snowflakeschema) te maken - waarmee BI analyse nog gemakkelijker zouden zijn - dus daarom hebben we voor een relationeel model gekozen, in 3e normaalvorm. Het ODS werd gebruikt als bron voor een SQL Azure Reporting rapport en in PowerPivot for Excel.

Het verversen van de gegevens is niet noodzakelijk, dus je zou bij wijze van spreken het Excel document in het vliegtuig mee kunnen nemen en doorgaan met je analyse!

Belangrijk onderdeel van de oplossing is Microsoft Dynamics CRM Online.