

Verwerking business rules kan geautomatiseerd worden

ETL-Generator (3)

Jorik Kool, Ferdy Betgen en Vincent Wylenzek

In het derde en laatste deel van de reeks over de ETL-generator staat het ETL-proces van EDW naar de data mart centraal. De auteurs stellen een model voor om de verwerking van de business rules (BR's) te automatiseren in het ETL-proces. Deze automatisering biedt voordelen op het gebied van eigenaarschap, centralisatie, kwaliteit, snelheid en flexibiliteit. Daarnaast gaan ze in op het automatiseren van autorisatie.

In de vorige twee delen is het ETL-proces van bron tot en met EDW besproken. Bij het ETL-proces van EDW naar de data mart staan de mogelijkheden om de verwerking van de business rules in de ETL te automatiseren centraal. Een kanttekening die we hierbij moeten maken is dat het data mart-deel nog in concept/ontwerp is. Het model is op basis van best practices opgezet, toch moet de uitwerking nog nader getoetst worden aan de praktijk. Zie afbeelding 1 in het eerste deel in DB/M 5.

De datamart

De data mart kan op meerdere manieren worden geïmplementeerd. De ETL-generator gaat uit van een dimensionale modellering met fysieke opslag (tabellen). Het ETL-proces van de data mart bestaat uit dezelfde acht basisstappen als het ETL-proces van staging naar het EDW (zie afbeelding 2 in het tweede deel in DB/M 7), alleen de invulling van stap drie wijkt af. Stap drie (bewerken) wordt nu gebaseerd op BR's. In de derde stap in het ETL-proces van staging naar het EDW waren dit zeer eenvoudige bewerkingen, in de stap naar de data mart omvat dit eenvoudige tot zeer complexe BR's.

Business rules

Wat is een business rule? Voordat we dieper ingaan op de (on)mogelijkheden van genereren, lichten wij toe wat wij onder BR verstaan aan de hand van een eenvoudig voorbeeld:

Een klant met meer dan vijf aankopen in de maand krijgt het label 'Goud'.

Dit artikel is het derde en laatste deel van een drieluik over het genereren van ETL. Deel 1 en 2 verschenen in DB/M 5 en DB/M 7.

Dit voorbeeld beschrijft welke afleiding of bewerking plaats dient te vinden. De definitie heeft een bepaalde toegevoegde waarde voor een stakeholder, het helpt de business om tot inzicht of een beslissing te kunnen komen. Deze informatie wordt samengesteld op basis van de beschikbare, toegankelijke data. Samenvattend resulteert dit in: *Een business rule is een afleiding of bewerking van data om te komen tot informatie vanuit het perspectief van de business.*

BR's dienen te worden opgesteld door een kennishouder van het specifieke (bedrijfs)proces. Een kenmerk van BR's is dat deze aan verandering onderhevig zijn. Om kostenoptimaal met deze veranderingen om te kunnen gaan, moeten wijzigingen met minimale inspanning kunnen worden doorgevoerd. Wanneer BR's geautomatiseerd door de ETL-generator in de ETL verwerkt kunnen worden, kan er aan deze eis voldaan worden.

Classificatie van business rules.

Er is in de bestaande literatuur over BR's nog geen breed gedragen standaard opgesteld voor categorisatie van BR's. Inmon (2008) beschrijft het business metadataconcept vrij uitgebreid. Ref. [1]. Vanuit onze ervaring hebben we een eigen classificatiematrix (zie tabel in afbeelding 1) gemaakt, met vier categorieën BR's. Deze onderverdeling draagt bij aan de keuze voor wat betreft het automatisch verwerken van BR's in het ETL-proces.

Toelichting matrix.

De matrix verdeelt de BR's langs de assen complexiteit en reikwijdte. Langs de horizontale as wordt een indeling gemaakt in enkelvoudige of meervoudige bewerkingen (afhankelijk van één of meerdere tussenresultaten). Langs de verticale as wordt bepaald of de BR impact heeft op één regel of op een dataset. De kleur van de hoofdletter G in de kwadranten geeft aan of de bewerking in aanmerking komt voor automatisering. Bij een groene kleur (kwadrant I en II) verwachten wij geen problemen, bij oranje (kwadrant III) kunnen complicaties optreden en bij rood (kwadrant IV) verwachten wij geen positieve business case. Het genereren kost dan meer tijd dan een realisatie met maatwerk.

Enkele voorbeelden van de vier kwadranten:

I – Verrijkingen zoals 'artikelen van type 1 zijn rood, vierkant en duur';

II – Categoriseren van bijvoorbeeld bonusuren. Een uur telt mee

voor de bonus indien deze aan een aantal kenmerken voldoet. De eerste indeling is in factureerbaar, opleiding of intern. Voor de categorieën opleiding of intern tellen alleen die uren mee waarvoor het management autorisatie heeft verleend;

III – Top x-lijsten (top 10-klanten = ‘voorkeursklanten’) en domeincontroles zoals orders mogen alleen binnen de landsgrenzen worden bezorgd;

IV – Berekenen van gewogen gemiddelden; een in gebreke blijvende debiteur wordt geclassificeerd als zodanig en krijgt een indicator dat hij/zij een aantal periodes niet betaald heeft. Na een langere periode van tijdige betaling, verdwijnt dit kenmerk.

Interface.

Als interface stellen we een GUI voor (zie afbeelding 2), waarmee de eigenaar of gedelegeerde de BR's kan invoeren en beheren. De grootste uitdaging in dit proces is de vertaalslag die gemaakt moet worden van functionele BR naar technische code. In het verleden zijn hier diverse initiatieven voor bedacht zoals pseudo-SQL en Structured English. Helaas bleken deze initiatieven niet voldoende in staat de complete semantiek van de business te kunnen vertalen in genereerbare code. Wij verwachten niet dat wij binnen onze scope een volledige pseudocodetaal kunnen ontwikkelen. Maar wanneer we de scope beperken tot de eerste drie kwadranten van de BR-classificatiematrix, dan moet een groot deel van de mogelijke expressies door middel van een dergelijke pseudocodetaal beschreven kunnen worden.

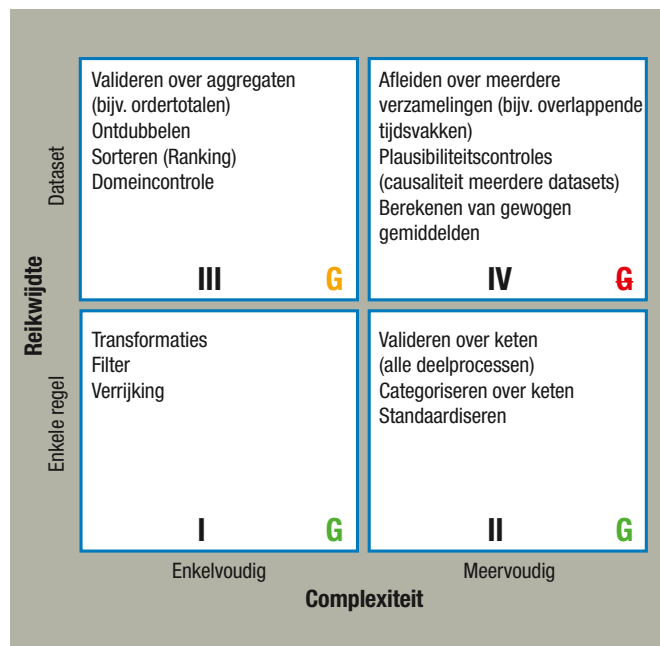
Voordelen.

Voordelen van het genereren van de business rules in de ETL en de GUI zijn:

1. Eigenaarschap; de afnemers van de informatie uit het datawarehouse die daartoe zijn geautoriseerd hebben toegang tot een GUI om de BR in te voeren;
2. Centralisatie; er is één centrale plek voor BR-opslag. Wanneer informatie over een BR benodigd is, dan is duidelijk waar deze is gedefinieerd en dus te vinden is;
3. Kwaliteit; met behulp van de GUI wordt het invoeren van BR's gestandaardiseerd. Dit komt de beheersbaarheid ten goede en verhoogt de kwaliteit;
4. Snelheid; de BR's worden opgedeeld in categorieën die voor een deel (kosteneffectief) kunnen worden gegenereerd. Het genereren reduceert de ontwikkeltijd;
5. Flexibiliteit; toevoegingen en wijzigingen aan de BR's kunnen eenvoudig worden doorgevoerd in de GUI en worden vervolgens (deels) automatisch opgepakt en verwerkt in de ETL.

De informatielaag

De informatielaag is opgebouwd uit informatieproducten voor de eindgebruikers. Deze producten zijn veelal klantspecifiek en daarom ligt het niet voor de hand deze te genereren. Om die reden hebben we gekozen deze laag buiten de scope van de generator te houden. Uitzondering op dit geheel is het autorisatiemechanisme, dat is wel geautomatiseerd.



Afbeelding 1: BR-classificatiematrix.

Autorisatie.

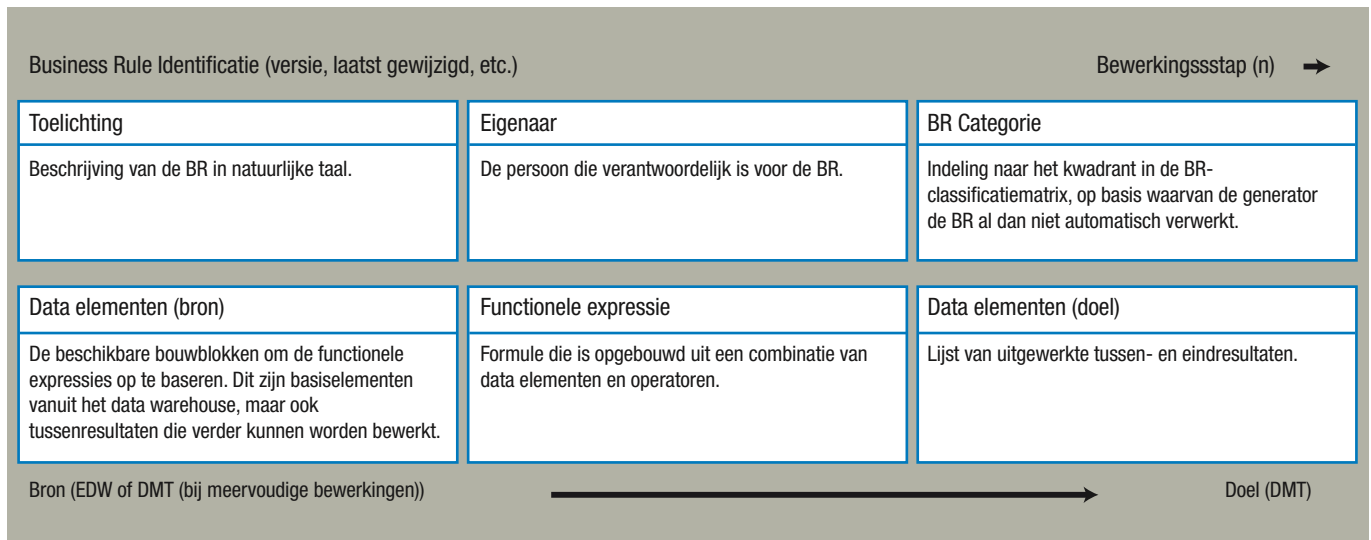
Voor vrijwel elk informatieproduct geldt dat er bepaald moet worden welke informatie een eindgebruiker in mag zien. Door middel van authenticatie wordt gecontroleerd of een gebruiker inderdaad degene is wie hij beweert te zijn en door middel van autorisatie wordt bepaald welke informatie deze geauthenticeerde gebruiker in mag zien. Hiermee is autorisatie een essentieel onderdeel van ieder BI-project. Autorisatie is onderverdeeld in drie categorieën, namelijk de product-, de portal- en de informatiegerelateerde autorisatie:

1. De productautorisatie bepaalt welke informatieproducten een gebruiker mag inzien, zoals standaardrapporten of cubes;
2. De portalgerelateerde autorisatie betreft de autorisatie van bepaalde aandachtsgebieden op de portalomgeving, bijvoorbeeld het forum, de handleidingen, de versiebeheeromgeving en de rapportage- en analyseportaal;
3. Informatieautorisatie betreft de inhoudelijke data van de informatieproducten, bijvoorbeeld welke kostenplaatsen een bepaalde gebruiker mag inzien.

Eerst wordt het proces geschetst om van autorisatie requirements naar geïmplementeerde autorisatie te komen. Vervolgens worden de benodigde stuurtabellen besproken en daarna worden portal- en productautorisatie kort toegelicht. De informatieautorisatie wordt uitgebreid toegelicht met een codevoorbeeld voor SSAS.

Het proces.

De autorisatie requirements worden vastgelegd in een autorisatiematrix en beheerd met een GUI. De GUI slaat de matrix op in stuurtabellen in een onderliggende database. Op basis van de stuurtabellen wordt de autorisatie over alle drie de autorisatie-categorieën geautomatiseerd doorgevoerd. In afbeelding 3 wordt



Afbeelding 2: BR GUI.

het proces van requirement tot implementatie schematisch weergegeven.

Requirements.

De procesbeschrijving om tot een autorisatiematrix te komen valt buiten de scope van dit artikel, zie hiervoor Ref. [2].

Stuurtabellen.

De autorisatiematrix is opgeslagen in enkele stuurtabellen. De gebruikers- en gebruikersgroepentabellen worden gebruikt voor alle drie de autorisatietypen. Voor de productautorisatie worden de informatieproducttypen en informatieproducten in gelijknamige tabellen vastgelegd. Op beide niveaus kan autorisatie verleend worden. De mogelijke autorisaties zijn hier ‘geen rechten’, ‘openen’ of ‘wijzigen’.

Voor de portalautorisatie wordt de portalfunctie vastgelegd in functiecategorieën en functies. Hier zijn de rechten afhankelijk van het functietype. Voor functionaliteit met betrekking tot documentbeheer kan dit bijvoorbeeld lezen, aanpassen en/of verwijderen zijn.

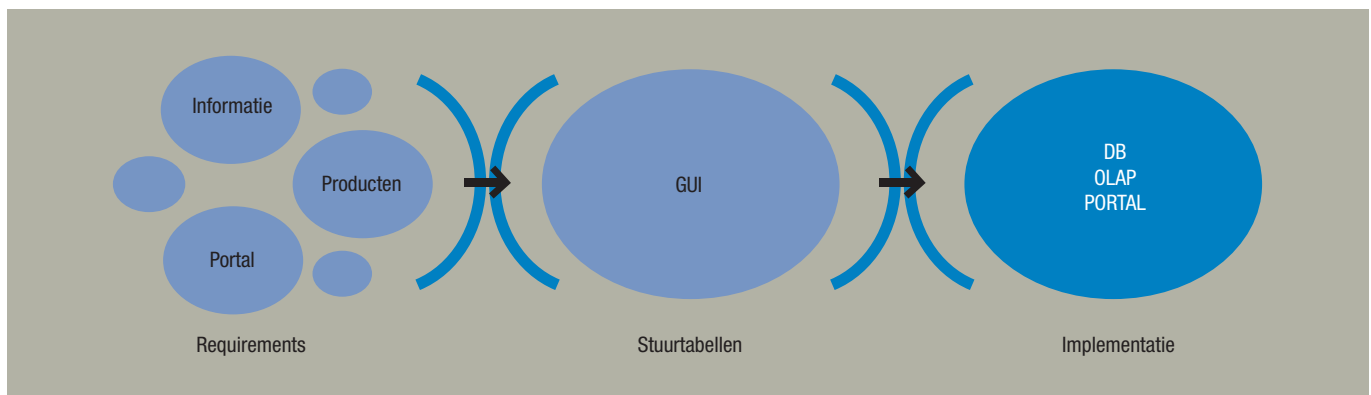
Voor de informatieautorisatie wordt voor elke te autoriseren

dimensie een additionele tabel opgenomen. In deze tabel wordt de gebruikers- of groeps-ID gerelateerd aan de te autoriseren data.

Implementatie.

Algemeen. Voor alle categorieën zijn gebruikers en gebruikersgroepen benodigd. Idealiter lopen de gebruikers en de gebruikersgroepen synchroon met de Active Directory (AD)/LDAP. Wij hanteren het uitgangspunt dat geautoriseerd wordt op Active Directory-gebruikers. Indien hiervan afgeweken wordt, dan adviseren wij gebruik te maken van een koppelingstabel, waarin interne gebruikers gekoppeld worden aan AD/LDAP-gebruikers of -groepen.

Productautorisatie. Productautorisatie wordt vanuit de stuurtabellen – informatieproducttypen en informatieproducten – vertaald naar de diverse tools en is afhankelijk van de functie/rol van een medewerker. Helaas is het niet mogelijk één generiek script te ontwikkelen waarbij productautorisatie voor elke tool kan worden doorgevoerd. Elke tool heeft zijn eigen specifieke kenmerken.



Afbeelding 3: Autorisatie; van requirement tot en met implementatie.

```

private static void AddRoles(Database database, AuthorisatiesData.SSASRolesDataTable
                           requestedRolesDataTable)
{
    foreach (AuthorisatiesData.SSASRolesRow row in requestedRolesDataTable.Rows)
    {
        //Role aanmaken (indien nog niet aanwezig)
        if (!database.Roles.ContainsName(row.RoleName))
        {
            Role role = database.Roles.Add(row.RoleName);
            RoleMember member = new RoleMember(row.Login);
            role.Members.Add(member);
            role.Update();

            //Rechten op SSAS-DB inregelen
            DatabasePermission dbPermission = database.DatabasePermissions.Add(role.ID,
            "DatabasePermission_" + row.RoleName);
            database.Update();
            dbPermission.Read = ReadAccess.Allowed;
            dbPermission.Update();

            //Leesrechten op SSAS-cubes inregelen
            foreach (Cube cube in database.Cubes)
            {
                CubePermission permission = new CubePermission(row.RoleName, cube.Name, cube.ID);
                permission.Read = ReadAccess.Allowed;
                permission.ReadSourceData = ReadSourceDataAccess.None;
                permission.ID = "CubePermission_" + cube.ID + "_" + row.RoleName;
                permission.Name = permission.ID;
                cube.CubePermissions.Add(permission);
                cube.Update();
                permission.Update();
            }
        }
    }
}

```

Afbeelding 4: Codevoorbeeld.

```

private static void WriteAllowedSet(AttributePermission currentAttributePermission, string allowedSetString)
{
    //concatenate allowedSet string
    string allowedSetStringComplete = "";
    if (allowedSetString.Length > 0)
        allowedSetStringComplete = "{" + allowedSetString + "}";
    if (currentAttributePermission.AllowedSet != allowedSetStringComplete)
    {
        currentAttributePermission.AllowedSet = allowedSetStringComplete;
        currentAttributePermission.VisualTotals = "1"; //Enable visual totals
        ((DimensionPermission)currentAttributePermission.Parent).Update();
    }
}

```

Afbeelding 5: Codevoorbeeld.

Wij hebben productautorisatie voor twee Microsoft producten uitgewerkt: SSRS en SSAS. Voor SSRS maken we gebruik van de Reporting Services API (ReportService2010.asmx). Met deze API worden op basis van de stuurtabellen de juiste SSRS-Roles toegerekend aan de eindgebruikers. Voor SSAS maken we, net als bij de informatieautorisatie, gebruik van de SSAS-roles. Hiermee kan de toegang op kubus(sen) worden ingeregeld. Afhankelijk van de gedefinieerde producten in de stuurtabellen, worden rechten toegekend aan de SSAS-roles.

Portalautorisatie. Portalautorisatie wordt vanuit stuurtabellen, functie en functiecategorieën vertaald naar de portal. Net als bij de productautorisatie geldt ook hier dat er niet één generiek script te ontwikkelen is voor meerdere tools. Voor SharePoint (2007) hebben we reeds een uitwerking gerealiseerd. Op basis van data uit de stuurtabellen worden de rechten binnen SharePoint (2007) ingedeeld. Bij SharePoint moet rekening gehouden worden met de aanwezigheid van een Windows user account, een SharePoint user account en rechten op een specifiek component (bijvoorbeeld webpart).

Informatieautorisatie. Voor elke te autoriseren dimensie is een eigen stuurtabel aanwezig. Deze stuurtabel wordt gevuld met de data op de laagste granulariteit van de te autoriseren dimensie. Bijvoorbeeld bij het autoriseren op de organisatiedimensie betekent dit dat alle afdelings-ID's (laagste niveau in ons voorbeeld) naar deze stuurtabel worden geladen. Op basis van de autorisatiematrix wordt de koppeling gelegd tussen de gebruikers (groepen) en deze ID's.

Er zijn meerdere manieren om informatie uit de data mart op te vragen, bijvoorbeeld met directe query's, rapportages direct op de data mart of met een OLAP-cube.

Binnen onze huidige scope (Microsoft) maken wij gebruik van OLAP-cubes met SSAS. SSAS heeft namelijk in vergelijking met de andere twee methoden de meest uitgebreide mogelijkheden om de informatieautorisatie door te voeren. Dit komt ten gunste van beheer en flexibiliteit. Wij zijn er dan ook voorstander van om alle rapportages te voeden met data uit de OLAP-cube. Hierna volgt een technisch voorbeeld van hoe autorisatie in SSAS geautomatiseerd kan worden.

```
private static void SetAllowedSet(Database database, string dimensionName, string attributeID,
List<RoleAllowedElement> allowedSetRows)
{
    ...
    Dictionary<string, string> rolesWithAllowedSet = new Dictionary<string, string>();
    foreach (RoleAllowedElement row in allowedSetRows)
    {
        ...
        //Wegschrijven AllowedSet van vorige role.
        if (currentRole != null)
        {
            WriteAllowedSet(currentAttributePermission, allowedSetString);
            allowedSetString = "";
        }

        //Opzoeken huidige role.
        currentRole = database.Roles[row.RoleName];

        if (currentRole != null)
        {
            rolesWithAllowedSet.Add(currentRole.Name, currentRole.Name);
            string dimensionPermissionName = dimensionName + " permission " + currentRole.ID;
            currentDimensionPermission = permissions.FindByName(dimensionPermissionName);
            if (currentDimensionPermission == null)
                currentDimensionPermission = permissions.Add(currentRole.ID, dimensionPermissionName);
            currentAttributePermission =
                currentDimensionPermission.AttributePermissions.Find(attributeID);
            if (currentAttributePermission == null)
                currentAttributePermission =
                    currentDimensionPermission.AttributePermissions.Add(attributeID);
        }
    }
    ...
    if (currentRole != null)
        allowedSetString += row.AllowedElement

    //Wegschrijven AllowedSet van laatste role.
    if (currentRole != null)
    {
        WriteAllowedSet(currentAttributePermission, allowedSetString);
    }
    ...
}
```

Afbeelding 6: Codevoorbeeld.

De techniek – Autorisatie in SSAS.

SSAS maakt gebruik van interne autorisatie-‘roles’. Een ‘role’-definitie bestaat uit de toegang tot de onderliggende data sources, databases, cubes, meetwaarden en dimensies (cells). Met name de dimensies zijn interessant voor de informatieautorisatie, daar wordt bepaald welke records een eindgebruiker/groep mag inzien. Het mechanisme om van autorisatiematrix tot SSAS-roles te komen is volledig geautomatiseerd. Hiervoor wordt gebruik gemaakt van het Analysis Services Objectmodel (AMO), met in het bijzonder de *Microsoft.AnalysisServices* namespace.

Met behulp van de code in afbeelding 4 creëren we nieuwe roles en stellen we de rechten voor databases, cubes, meetwaarden en dimensies (cells) in. De set met roles wordt gebaseerd op een dataset bestaande uit een join tussen de gebruikers(groepen) en de dimensie stuurtabellen. Per te genereren role wordt de code doorlopen. In de eerste stap wordt de SSAS-role aangemaakt en gekoppeld aan de gebruiker/groep. In de tweede stap wordt toegang tot de Analysis Services database gegeven. Het betreft hier alleen leesrechten, dit is een minimale vereiste om objecten binnen de SSAS DB te kunnen benaderen. De derde stap geeft leesrechten op alle kubussen. De rechten die hier nu hard geprogrammeerd zijn, kunnen uiteraard ook in de stuurtabellen worden opgenomen.

In de code (afbeelding 4) was de dimensie (cell) security nog niet ingeregeld. Voor deze autorisatie gaan we uit van een negatieve benadering (standaard geen rechten), de zogenaamde ‘SSAS-allowedset’. Deze ‘allowedset’ wordt opgebouwd vanuit de stuurtabellen, met de code in afbeelding 5.

Nu het allowedset-statement gereed is, wordt dit met de code in afbeelding 6 aan de juiste role gekoppeld (let op, de procedure is niet volledig, alleen de relevante code is weergegeven).

Samenvatting

De ETL-generator biedt voordelen met betrekking tot kwaliteit en reductie van de ontwikkeltijd. Met de juiste toetsingscriteria (zie deel 1 in DB/M 5) kan bepaald worden welke ETL in aanmerking komt voor automatisering. In onze uitwerking baseren we ons op Microsoft BI, waarbij het data model Data Vault gemodelleerd wordt. Het ETL-proces van EDW naar data mart is nog in concept.

Literatuur

1. *Business Metadata*, William Inmon (2008).
2. *DB/M 5, 2009 - Dynamische front-end autorisatie* (Wylenzek).

Jorik Kool, Ferdie Betgen en Vincent Wylenzek zijn allen werkzaam bij Ordina als BI-consultant.