

Rulebase met beweringen in natuurlijke taal helpt

# AGILE ONTWIKKELEN MET BUSINESS RULES

Bedrijfsregels formuleren, centraal beheren en gebruiken wordt vaak geassocieerd met grote organisaties in de financiële wereld, of met grootscheepse toepassingen in de juridische sector. Alsof business rules pas wat opleveren bij een 'zwaardere' bedrijfslogica dan je met een procesmatige analyse aankunt.

Door Rob van Haarst

**O**ok de meeste standaarden en voorschriften voor business rules gaan uit van een ambitieuze aanpak, waarbij een corporatie zoals een multinationale bank haar bedrijfsregels organisatiebreed in kaart brengt en beheert, los van procesimplementaties en IT-projecten.

Wat gebeurt er als je business rules toepast in een middelgroot *agile* project voor ontwikkeling van een nieuw bedrijfsproces met ondersteunende maatwerksoftware? Hebben business rules dan nog meerwaarde? Houden standaarden en voorschriften voor business rules voldoende rekening met de specifieke omstandigheden van zo'n project? Deze vragen beantwoorden we aan de hand van een praktijkvoorbeeld.

## Planmatig onderhoud

Woningcorporaties in Nederland hanteren aanzienlijke budgetten voor het planmatig onderhoud van woningcomplexen. Denk aan het periodiek schilderen van houtwerk, vervangen van dakdelen, installeren van liften, ketels en dubbelglas, en het repareren van goten en houtrot. Er is veel geld te verdienen door precies op het juiste moment op de juiste plekken in te grijpen. Dan is offertes vergelijken niet genoeg. Je moet zelf gedetailleerde kennis opbouwen over je woningbezit. Recentelijk bouwde USoft in nauwe samenwerking met woningcorporatie Vestia Groep een applicatie om de kosten van planmatig onderhoud aan woningcomplexen te plannen, te onderhouden en te beheren. Daarvoor werd een nieuwe, gefaseerde werkwijze bedacht. In het proces vindt eerst een uitvoerige bouwkundige inventarisatie van het woningcomplex plaats. Hierbij

worden alle details in kaart gebracht, zoals aantallen ramen, afmetingen en gebruikte materialen, maar er wordt nog niet gekeken naar de staat van onderhoud. De inventarisatie biedt databasegegevens die in het hele verdere proces centraal staan, zie afbeelding 1.

In de volgende fase worden complexen met een bepaalde regelmaat door inspecteurs bezocht. Zij vullen op een door de software gestandaardiseerde manier inspectielijsten in, waarbij expliciete ISO kwaliteitsnormen worden toegepast. De lijsten zijn gebaseerd op de inventarisatie. In de laatste fase neemt de woningcorporatie op basis van de nu zeer gedetailleerde gegevens over de staat van onderhoud beslissingen over onderhoudsbudgetten. De software kan snel grote groepen geïnspecteerde bouwdelen uit fase 2 koppelen aan onderhoudsactiviteiten en prijzenboeken, waardoor *what-if* scenario's ontstaan waarin de totale kosten per jaar direct duidelijk worden bij een bepaalde aanpak. De corporatie kiest de beste aanpak en is in staat om deze aan te besteden voor een expliciet onderbouwd budget, dat is gebaseerd op een gedegen kennis van het eigen bezit.

De methodiek is nu drie jaar oud en in gebruik bij twee grote woningcorporaties. In dit artikel geven we een schets van de voors en tegens van business rules in zo'n project.

## Business rules aanpak

In bedrijfssoftware staan in zekere zin bedrijfsprocessen altijd centraal. Toch werd voor dit project het ontwikkelplatform USoft ingezet, dat bedrijfsregels en niet bedrijfsprocessen als uitgangspunt neemt. Het project vertoonde namelijk een aantal kritische indicatoren, die het uitgangspunt van bedrijfs-

regels aantrekkelijker maakten dan dat van processen:

- Het proces is data-intensief. De meeste handelingen bestaan uit interactie met dataverzamelingen. Opvolgende projectstappen zijn meestal verbonden door dataverzamelingen, die in de vorige stap worden samengesteld en bij de volgende stap handmatig worden geraadpleegd (bijvoorbeeld resultaten van inspecties worden gebruikt bij budgetbeslissingen);
- Inpassing in bestaande procesmodellen of service bus-modellen is relatief eenvoudig. Het gaat om een voorcalculatiemodule. Alleen het uiteindelijke resultaat wordt als richtprijs aan het aanbestedingsproces doorgegeven;
- De meeste processtappen worden uitgevoerd door getrainde eindgebruikers die een interne of semi-interne rol (vaste onderaannemer) hebben.

## Rulebase

Voor specificatie gebruikten USoft en Vestia bijna uitsluitend natuurlijke taal (Nederlands), slechts ondersteund door enkele procesdiagrammen en datamodeldiagrammen.

Er werd een centrale lijst (een *rulebase*) met elementaire natuurlijke taal beweringen aangelegd die één voor één op correctheid werden gecheckt door de opdrachtgever, en vervolgens één voor één door USoft werden gebouwd en getest, zie afbeelding 2.

De beweringen in het USoft project waren 100 procent in natuurlijke taal. Die was meteen begrijpelijk voor alle soorten betrokkenen: bestuurders, proces- en kwaliteitsdeskundigen, domeinexperts zoals filiaalmanagers, toekomstige softwaregebruikers, consultants, bouwers en testers, maar ook externe aannemers zoals bouwkundige adviesbureaus. USoft gebruikte naast taal ook wel datamodellen, BPMN-diagrammen en beslistabellen, maar alleen als *aanvulling*. De verschillende stappen in een proces of een beslisboom werden altijd *ook* in taal beschreven. Dat is een vrij radicale stap. Sommige dingen kunnen op zichzelf veel gemakkelijker in een bepaald soort diagram of tabel worden uitgedrukt. Je moet dan het geloof hebben dat het op den duur toch vruchten afwerpt om 'alles in business rules te hebben'.

## De hoge lat: business rules standaarden

Wat zijn goede business rules? Er bestaat een aantal uitstekende standaarden of voorschrijvende theorieën over wat goede bedrijfsregels zijn of goed gebruik van bedrijfsregels, zoals FCO-IM (1), het werk van Ronald Ross en BR Solutions (2), en OMG's SBVR (3). Een uitstekend begin zijn de praktische aanbevelingen van RuleSpeak waarvan enkele maanden geleden de Nederlandse versie is gepresenteerd (4).

Het academisch zuivere conceptuele model van FCO-IM biedt een onmisbaar referentiekader voor goed modelleren, maar is ook duur. In agile projecten moet je soms tijd winnen door te vertrouwen op de jarenlange ervaring van een senior informatie-analist die niet elk detail formeel aan voorbeeldgegevens hoeft te toetsen. Met SBVR schetst OMG een groots landschap

waarin de opdrachtgever lange-termijn budgetten klaar heeft liggen voor een centraal en onafhankelijk register met geldende business rules (business rules repository), dat via een serie vervolgstappen wordt vertaald in praktische maatregelen zoals werkende software. Een boeiend perspectief voor grote corporaties, maar in veel projecten werk je met een veel beperkter budget.

In het USoft project bleek dat business rules standaarden een kwaliteitsborging bieden, maar dat het ook te ambitieus kan zijn ze in detail na te leven.

## Het ambacht: business rules authoring

Wel moeten consultants vlot en trefzeker business rules in natuurlijke taal kunnen verwoorden zonder steeds te twijfelen over de juiste vorm. Gelukkig geeft met name Ross (5) praktische richtlijnen die gemakkelijk toe te passen en te onthouden zijn. Er is in 2009 nog een grote behoefte aan gereedschap die *authoring* (het opstellen of verwoorden van bedrijfsregels) direct ondersteunt. Veel ontwikkelteams schrijven specificaties wel in natuurlijke taal, maar gewoon in Word documenten. Het is daarom goed om eens in detail te bekijken welke conclusies USoft heeft getrokken uit deze meer gestructureerde toepassing van bedrijfsregels in een *agile* omgeving.

## Elementaire beweringen zijn een investering

Nagestreefd werden *elementaire* beweringen: beweringen die ondeelbaar zijn en die onafhankelijk van context kunnen worden begrepen. Een ondeelbare bewering beweert maar één ding tegelijk. Dus niet:

Opzichters boven de 60 jaar of die langer dan 15 jaar bij het bedrijf werkzaam zijn krijgen een seniorenverlof van 2 vakantiedagen.

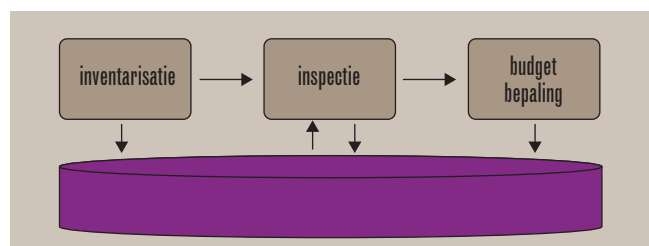
maar, opgesplitst:

Opzichters boven de 60 jaar krijgen een seniorenverlof.

Opzichters die langer dan 15 jaar bij het bedrijf werkzaam zijn krijgen een seniorenverlof.

Het seniorenverlof bestaat uit 2 extra vakantiedagen per jaar.

Als je natuurlijke taal wilt schrijven, kun je in plaats van elementaire beweringen ook juist kiezen voor een verhalende stijl. Dat is een totaal andere benadering die leidt tot samen-



Afbeelding 1: Gefaseerde aanpak en hergebruik van dataverzamelingen.

Nr	Omschrijving	Gevalideerd	Prioriteit	Vs.	Typering	Module
1744	Gebreken zijn mogelijk bij standaard onderdelen. Vb.: Houtrot is mogelijk bij vurenhouten kozijnen behandeld met dekvrf.	✓	Must Have	1.0	Feit	Inspectie
898	Een <i>blokadresbepaling</i> is het feit dat het een blokadres van toepassing is op een blok. Vb.: St. Aldegondestraat 18 t/m 48 even is het (of een) adres van blok DE403-E.	✓	--	1.0	Term	Complexen
1430	Kozijnafmetingen voor Liggende en Overige stijlen moeten ofwel allebei, ofwel geen van beide worden opgegeven. Alleen een afmeting voor Liggend, of alleen een afmeting voor Overig is niet toegestaan.	✓	Could Have	1.0	Regel	Inventarisatie
2746	Externe inspecteurs mogen alleen reeds aangemaakte inspectielijsten downloaden.		Should Have	1.1	Regel	Inspectie
2371	Planners kunnen voor elk item afwijken van de berekende richtprijs door een handmatige prijs te bepalen.	✓	Must Have	1.1	Optie	Planning
2801	Gebruikers moeten inspectielijsten kunnen downloaden met MS Excel 2003 en hoger.	✓	Must Have	1.1	Systeemeis	Inspectie

Afbeelding 2: Voorbeelden van bedrijfsregels met enkele attributen voor agile procesbeheersing.

hangende documenten met een kop en een staart: inleiding, hoofdstukken, samenvattingen, een inhoudsopgave en (hopelijk) een register.

## Waarom elementair?

Elementaire beweringen zijn veel beter te formaliseren dan verhalende documenten. Dat biedt een evident technisch voordeel wanneer u over een rules engine zoals USoft beschikt. Die engine kan elementaire bedrijfsregels automatisch op het juiste moment en in de juiste sequentie executeren, waardoor natuurlijke taal beweringen en hun implementaties parallel en één-op-één kunnen worden beheerd (gewijzigd, verwijderd ...).

Maar elementaire beweringen hebben ook algemene praktische voordelen. Een lezer kan de betekenis van een elementaire bewering begrijpen zonder context. De beweringen kunnen dus in elk gewenst verband worden beheerd en gepresenteerd. Je kunt losse beweringen indelen: naar datum aangemaakt, gevalideerd, gebouwd, getest, verwijderd; naar bewerker; naar onderwerp; naar type (definitie, feit, regel, instructie ...); naar prioriteit van de implementatie (Must Have, Should Have, enzovoort).

Daar staan wel gevoelige praktische nadelen tegenover:

- Tijdens het *opschrijven* van business rules moeten uitleggende of betogende 'stukjes' vermeden worden. De verleiding is groot om alle regels rondom het seniorenverlof bij elkaar te plaatsen; zoveel zijn het er niet. De toegevoegde waarde van splitsen is niet zo duidelijk op het moment van schrijven, die blijkt pas in een later stadium;
- Bij het *teruglezen* blijken de meeste mensen de structuur van een document te missen. Het is wennen om ineens geconfronteerd te worden met een 'bak' regels die in principe ongeordend zijn, zonder vaste plaats in een document.

Elementaire beweringen hebben dus grote meerwaarde, maar niet op de korte termijn. Ze vormen een investering op de langere termijn. Ze hebben grotere waarde naarmate ze worden gecombineerd met een Business Rules Management (BRM) methodiek en daarbij horende gereedschappen.

## Controleerbare beweringen

De beweringen in het USoft project waren alle controleerbaar. In termen van SBVR waren het falsifieerbare beweringen (*propositions*). Je kunt van elk van de beweringen zeggen of deze wel of niet klopt.

Dat is erg belangrijk omdat ze door de opdrachtgever moeten worden gevalideerd. De beweringen zijn gedurende de hele bouwperiode, en ook daarna, de inhoudelijke leidraad. Zo zijn ze het uitgangspunt bij het testen van de software later in het proces.

Elementaire beweringen zijn erg goed controleerbaar te maken omdat ze geen betekenis ontlenden aan de context (aan andere beweringen, of aan de opbouw en indeling van een document).

## Vocabulaire

SBVR staat voor *Semantics of Business Vocabulary and (business) Rules*. Een twee-eenheid dus: een *vocabulaire* of conceptuele structuur, en bedrijfsregels die 'gebouwd zijn op' die structuur. Net als bedrijfsregels kan een vocabulaire heel goed in elementaire beweringen worden gevat:

- Er zijn bouwdelen.
- Er zijn materialen.
- Er zijn behandelingen.
- Bouwdelen zijn vervaardigd van materialen.
- Een behandeling is van toepassing op een bouwdeel van een bepaald materiaal.

Zo wordt een structuur opgebouwd van zogenaamde concepten of begrippen (bouwdeel, materiaal) en zogenaamde feittypen (bouwdelen zijn vervaardigd van materialen). Overigens kunnen ook feittypen en bedrijfsregels als concept worden gezien. Bovenstaande beweringen zijn declaraties: ze introduceren elk een concept.

Kenmerkend voor SBVR is dat alle begrippen moeten worden gedefinieerd. Hier zijn twee voorbeelden van definities:

Een behandeling is een manier waarop bepaalde bouwdelen worden behandeld of afgewerkt.

Vb.: AK dekverf.

Vb.: onbehandeld.

Een behandelbaarheid is het feit dat een bepaald materiaal bouwtechnisch op een bepaalde manier kan worden behandeld.

Vb.: Een metalen bouwdeel kan met AK dekverf behandeld zijn.

Vb.: Een aluminium bouwdeel kan een coating hebben.

De concrete voorbeelden (hier steeds geplaatst na 'Vb.:') maken het veel gemakkelijker voor lezers om de definitie snel te doorgronden. Bedrijfsregels zijn gebouwd op, of refereren aan, het vocabulaire:

In standaardbegrotingen moet voor elke behandeling de eenheidsprijs worden toegepast uit het jaar van de standaardbegroting, of als die niet bekend is de meest recente eenheidsprijs die voor die behandeling wel bekend is.

Goede ondersteunende software voor *business rules authoring* maakt een onderscheid tussen declaraties, definities en bedrijfsregels, maar laat ze wel alle toe en helpt de onderlinge samenhang te doorzien en te bewaken.

Je kunt er ook voor kiezen om het vocabulaire geheel gescheiden van de bedrijfsregels te definiëren en onderhouden. Bijvoorbeeld kan het vocabulaire in een datadiagram worden weergegeven en alleen de bedrijfsregels zelf in natuurlijke taal. Daarmee wordt wel een aantal paden afgesneden. Bestaande documenten in de organisatie van de opdrachtgever bevatten vaak een mix van declaraties, definities en bedrijfsregels. Als je in staat bent die alle te behandelen in natuurlijke taal vorm, kun je ze veel gemakkelijker bundelen, herschikken, (nu of later) geautomatiseerd verwerken, en verbanden leggen over het geheel van vocabulaire en regels.

## Typering

Er zijn allerlei soorten beweringen. Naast declaraties, definities en bedrijfsregels komen we in een zakelijke omgeving ook adviezen en systeemeisen tegen. Als u een rulebase met beweringen in natuurlijke taal koppelt aan implementaties (hetzij half-geautomatiseerd of misschien zelfs volautomatisch), dan is typering essentieel op technisch niveau. Software moet iets anders doen met definities dan met bedrijfsregels.

Maar ook als u een rulebase inricht zonder deze meteen te koppelen aan implementaties, geeft het veel houvast om de soort bewering die je aan het schrijven bent, alleen al te *herkennen*. Dat kan door de SBVR standaard te bestuderen, want die geeft semantische (inhoudelijke) criteria voor typering.

Het zou gemakkelijker zijn om het type bewering te herkennen aan de *vorm*. Maar lukt dat ook in de praktijk? RuleSpeak geeft praktische richtlijnen voor het opstellen van bedrijfsregels, uitgaand van het gegeven dat bepaalde taalvormen het idee van verplichting uitdrukken dat kenmerkend is voor bedrijfsregels, en andere juist niet. Bedrijfsregels hebben bijvoorbeeld wel de vorm:

... moet / moeten ... (niet) ...

... mag / mogen ... alleen ...

... mag / mogen ... niet ...

maar niet de vorm:

... mag / mogen ... (dat wil zeggen zonder *alleen* en zonder *niet*)

Die laatste vorm is ook heel valide, maar drukt een advies, suggestie of opmerking uit en niet, zoals een echte bedrijfsregel, een verplichting.

De RuleSpeak taalvormen blijken een uitstekende leidraad tijdens het *formuleren* van beweringen. Maar bij het *interpreteren* van beweringen (die een klant formuleert, of die in een bestaand bedrijfsdocument worden aangetroffen) kunnen ze helaas niet als een wet van Meden en Perzen toegepast worden. Neem een bedrijfsregel die een gedeelte van een prijsberekening beregelt. Die kan inderdaad luiden:

In standaardbegrotingen moet voor elke behandeling de eenheidsprijs worden toegepast die ...

*Moet* geeft hier aan dat dit een bedrijfsregel is, iets dat opgevolgd moet worden of dat bedoeld is om richting te geven aan iemands handelen. Maar het neutralere:

In standaardbegrotingen wordt voor elke behandeling de eenheidsprijs toegepast die ...

(hetzelfde, zonder *moet*) is ook bepaald niet onnatuurlijk. De bedrijfsregel wordt hier minder als verplichting en meer als vaststaand feit gepresenteerd, maar in de context van een rulebase is het effect eigenlijk precies hetzelfde. Als je beweringen of bedrijfsdocumenten al of niet geautomatiseerd wilt overnemen en gebruiken, is het een praktische realiteit dat je heel vaak geen *moet* zult tegenkomen maar er toch sprake is van een bedrijfsregel.

## Systeemeisen toelaten

Bij discussies over business IT alignment is een welbekende angst dat een te technische IT-invalshoek een goed begrip van de business kan 'vervuilen'. RuleSpeak waarschuwt expliciet voor de gevaren van het opnemen van systeemeisen. Ook

OMG houdt SBVR op veilige afstand van IT-systeemeisen, door SBVR te positioneren als CIM (computation independent model) dat op een *ander* moment moet worden omgezet in een *ander* (getransformeerd) model dat dan een rol gaat spelen bij software-ontwikkeling.

We hebben het bij systeemeisen over beweringen als:

Een externe uitvoerder moet alle complexgegevens kunnen lezen die horen bij een project waarop hij/zij is ingezet.

Een gebruiker moet een inspectielijst kunnen openen met Excel 2003 of hoger.

Redenen genoeg voor informatie-analisten om te aarzelen om dit soort beweringen in een bedrijfsregelregister op te nemen. Kunnen die niet toch beter in een afzonderlijk functioneel ontwerp (een document)? Nee, in een agile omgeving is het beter dat niet te doen. Daarmee verlies je weer het zwaar bevochten directe contact tussen softwarebouw en de immer veranderende business.

Bovendien liet het USoft project een heel andere dagelijkse realiteit zien. Als je bedrijfsregels begint op te schrijven met het doel om op relatief korte termijn een bedrijfsproces in te richten inclusief de bijbehorende software, dan zul je merken dat bijna de helft van de beweringen systeemeisen betreft.

Mag dit? Hebben we het hier nog wel over de 'business'? Het antwoord is ja, als tenminste het gebruikte vocabulaire betekenis heeft voor de opdrachtgever. Als die hard meewerkt door in hoog tempo uitspraken te valideren, is niet uit te leggen dat iets niet over IT mag gaan, zeker niet als hij zelf een begrip als Excel 2003 heeft aangedragen en dit dus kennelijk voor hem van belang is.

Op papier is dus heel duidelijk wat een systeemeis is, en dat het niet een business rule is (in nauwe zin). Maar wanneer een agile ontwikkelteam aan de slag gaat met bedrijfsregels, rijst al gauw de vraag of het nu wel zo praktisch is om systeemeisen uit de business rule repository te weren. (6)

## De juiste vorm

Het voorbeeld van systeemeisen laat en passant zien hoezeer business rules een praktische leidraad vormen bij het opstellen van tekst.

Goede systeemeisen hebben wel de vorm:

... moet / moeten ... kunnen ...

... moet / moeten ... in staat zijn om ...

maar niet ... kan/kunnen ...:

Gebruikers kunnen inspectielijsten openen met Excel 2003 of hoger.

Dit laatste voorbeeld beschrijft iets dat pas geldig is op het moment dat het is geïmplementeerd. Dat leidt tot misverstanden en opdrachtgevers hebben moeite met het valideren van zo'n bewering omdat je (nog) niet kunt zeggen dat hij 'klopt'.

'Echte' bedrijfsregels beschrijven iets dat gedurende langere tijd steeds weer toepasbaar is. Wat je valideert is dat die regel moet gaan gelden, of nog steeds geldt. De vormen ... *moeten kunnen* ... en ... *moeten ... in staat zijn om* ... passen goed in zo'n omgeving, want ook zij beschrijven een duurzame realiteit (die van een wens of eis), onafhankelijk van de aanwezigheid van een technische implementatie.

## Samenvatting

Bedrijfsregels bieden veel voordelen, ook als niet meteen een centraal en organisatiebreed regelregister wordt aangelegd.

Daarbij geldt:

- Elementaire beweringen hebben meerwaarde, al vragen ze op de korte termijn veel extra inspanning;
- Elke bewering moet controleerbaar worden opgeschreven;
- Het is haalbaar en aantrekkelijk om beweringen die een vocabulaire opbouwen, op dezelfde manier en op dezelfde plek te beheren als bedrijfsregels, adviezen en systeemeisen;
- Ondanks de ideologie dat bedrijfsregels over de business en niet over de IT moeten gaan, is het acceptabel en praktisch om systeemeisen en bedrijfsregels te combineren, mits dat gestructureerd gebeurt;
- Een business rules aanpak biedt veel houvast bij het vinden van de juiste vorm.

## Literatuur

1. FCO-IM: Bakema, Guido, Jan Pieter Zwart en Harm van der Lek, Volledig communicatiegeoriënteerde informatiemodellering, Academic Service 1996, en [www.fco-im.nl](http://www.fco-im.nl).
2. BR Solutions: [www.brsolutions.com](http://www.brsolutions.com).
3. SBVR: een goede inleiding is Annex A van de specificatie zelf. Versie 1.0, januari 2008, is gratis te downloaden van [www.omg.org/spec/SBVR/1.0/](http://www.omg.org/spec/SBVR/1.0/) (4MB PDF).
4. RuleSpeak: [www.rulespeak.com/nl](http://www.rulespeak.com/nl).
5. Ross, Ronald, Business Rule Concepts, Business Rule Solutions, 3e druk, 2009.
6. Coenen e.a.\* wijzen op het verschil tussen systeemeisen en business rules in nauwe zin (ze noemen systeemeisen ontwerp-regels), maar merken ook op dat de twee typen 'in tweede instantie' vergelijkbaar zijn omdat het maken van IT-systemen niet minder een bedrijvigheid is dan alle andere bedrijvigheid ('business') in de organisatie. Bij agile ontwikkeling is dat argument nog geldiger, omdat bij deze methode het maken van het IT-systeem als gezamenlijke bedrijvigheid en verantwoordelijkheid van opdrachtgever en softwareleverancier wordt beschouwd, en deze partijen (juist dankzij de business rules repository) hetzelfde vocabulaire hanteren.  
\* Coenen, Alcedo, Leo Hermans, Matthijs van Roosmalen en Silvie Spreeuwenberg, (Uw bedrijf geregeld met) Business Rule Management, Academic Service, 2008, p. 58.

Rob van Haarst ([rob.van.haarst@usoft.com](mailto:rob.van.haarst@usoft.com)) is specialist Software Development bij USoft B.V.