

ICT-ondersteuning van een SOA kan niet zonder EA.

Samenvatting

In het kader van Enterprise Application Integration (EAI) bij Meavita West, één van de werkmaatschappijen van Meavita Nederland, werd een koppeling ontwikkeld tussen twee applicaties. Hierbij werd gebruik gemaakt van een Tibco Enterprise Service Bus.

De koppeling werd ontworpen vanuit een gegevens(uitwisselings)perspectief, maar kon (tot nu toe) niet goed gerealiseerd worden.

Analyse leverde ons de volgende leerervaringen op over EAI, SOA, EA en de Business-IT-alignment.

Uitgaande van Meavita's behoefte om haar pakketten te integreren:

- SOA is een organisatie-inrichtingsparadigma dat zegt dat een organisatie zich kan inrichten door in- en externe diensten uit te wisselen.
- Het bijbehorende ICT-inrichtingsparadigma zegt dat ICT die diensten kan ondersteunen middels services.
- Het project moest wel mislukken omdat
 - a) de business niet SOA is ingericht en er geen duidelijkheid is hoe de organisatie wel werkt. En
 - b) omdat getracht is een ICT-oplossing te maken vanuit een SOA-paradigma
- Businessarchitectuur betreft de business-elementen (strategie, beleid, organisatie-indeling, werkprocessen) en ICT-architectuur de ICT-elementen (applicaties en infrastructuur).

Enterprise Architectuur is dan het geheel van beide architecturen, inclusief de onderlinge relaties, afhankelijkheden en de gehanteerde principes.

Inleiding

In het kader van Enterprise Application Integration (EAI) bij Meavita West, één van de werkmaatschappijen van Meavita Nederland, werd een koppeling ontwikkeld tussen twee applicaties. Hierbij werd gebruik gemaakt van een Tibco Enterprise Service Bus.

De koppeling werd ontworpen vanuit een gegevens(uitwisselings)perspectief, maar kon (tot nu toe) niet goed gerealiseerd worden.

In dit artikel beschrijven we hoe de koppeling tot stand kwam, de analyse van wat er fout ging en welke consequenties wij daar uit trekken voor de toekomstige werkwijze bij onze organisatie.

Tenslotte komen we tot enkele meer algemene leerervaringen over EAI, SOA, de business-it-alignment en de noodzaak van een Enterprise Architectuur.

Kennismaking met Meavita.

Meavita Nederland is een fusie-organisatie van 4 grote organisaties in de zorgsector. Op hoofdlijnen levert elk van de constituerende organisaties thuiszorg, verzorgingshuiszorg, verpleeghuiszorg, thuishulp (wmo-zorg) en worden, namens enkele gemeenten, consultatiebureaus gerund. De meeste zorg wordt uitgevoerd in het kader van de AWBZ en WMO, maar er wordt ook particuliere zorg geleverd.

De omzet van Meavita Nederland bedraagt € 500 miljoen.

Meavita opereert in Noord-, Oost-, Midden- en West-Nederland met zo'n 20.000 medewerkers.

Enkele ICT-kenmerken:

Er zijn ongeveer 3000 werkplekken, verdeeld over 5 infrastructuren en er wordt gebruik gemaakt van zo'n 300 applicaties. Bij een aankomende consolidatie moet dit aantal idealiter teruggebracht worden naar 30.

Van de infrastructuren zijn er enkele die zelf beheerd worden, terwijl andere geoutsourced zijn.

Het ICT budget bedraagt € 14 miljoen.

Uiteraard is consolidatie, in ieder geval op ICT-gebied, een belangrijk item voor deze jonge organisatie.

Meavita en SOA.

De raad van bestuur van Meavita West heeft twee jaar geleden besloten tot aanschaf van een Tibco Enterprise Service Bus (ESB) als middleware. Pas later is men gaan bedenken wat SOA is. Aanvankelijk werd SOA gezien als een ICT-architectuur/EAI-paradigma, pas later als een organisatie-inrichtingsparadigma.

Langzamerhand begrijpen we hoe dit paradigma ondersteund kan worden door het ICT-architectuurparadigma.

Hierbij moet vermeld worden dat Meavita West werkt met gekochte pakketten. Services vinden we tussen die pakketten en niet er binnen. (Als je geen COTS gebruikt kun je ook op fijner granulair niveau diensten onderscheiden en dus met ICT realiseren)

De afdeling ICT-verandermanagement is op dit moment het organisatieonderdeel waar het meest fundamenteel nagedacht wordt over SOA.

SOA begrijpen wij op dit moment als volgt:

SOA is een organisatie-inrichtingsparadigma, dat ondersteund wordt door een ICT-architectuur-paradigma.

Het organisatie-inrichtingsparadigma is:

De organisatie dient zich dienstgeoriënteerd in te richten. Dat betekent dat de organisatie zich opdeelt in eenheden die met elkaar interageren op basis van (organisatie)diensten.

Het ICT- architectuurparadigma is:

Applicaties (pakketten) communiceren met elkaar op basis van (technische) diensten.

Het ICT-architectuurparadigma ondersteunt het organisatie-inrichtingsparadigma als volgt:

Een organisatie-eenheid wordt ondersteund door een applicatie. Applicaties communiceren met elkaar op basis van (technische) diensten, services genoemd, die (organisatie) diensten ondersteunen. Er is een 1 : 1 relatie tussen diensten en services.

Zo ontstaat dus de business-it-alignment-strategie, die als volgt in elkaar steekt:

- Als de business zich dienstgeoriënteerd organiseert
- kan ICT die (organisatie) diensten ondersteunen met technische diensten (services).

De Praktijkcase.

De case is gesitueerd bij Meavita Klantenservice (MKS), een onderdeel van Meavita West. MKS werft en beheert klanten die zorg (gaan) afnemen, een 'verkoopfunctie'. Ter ondersteuning van haar werkprocessen gebruikt zij "Miracle", gebaseerd op de telesales-module van de Oracle e-businesssuite.

Thuiszorg Den Haag (TZ) is één van de zorgleverende bedrijfsonderdelen. Haar werkprocessen worden ondersteund met het Thuiszorg Informatie Systeem (TIS), ontwikkeld door Centric.

Schets van de opdracht.

Er moest een koppeling gerealiseerd worden voor gegevensuitwisseling tussen de applicaties Miracle en TIS.

De afdeling MKS maakt afspraken met prospects en klanten over levering van zorg. De klantgegevens worden ingevoerd in Miracle.

TZ is de organisatie die deze zorg uitvoert/levert.

De gegevens die door MKS in Miracle worden ingevoerd, moeten daarom ook in TIS beschikbaar zijn. ICT heeft de koppeling gemaakt. De Tibco ESB is gebruikt als middleware.

Wanneer betrokken:

Als senior informatieanalist van de afdeling ICT-verandermanagement, werd ik bij het koppelingsprobleem betrokken toen er verondersteld werd dat er al reeds een werkende koppeling gerealiseerd was. Dat ik er zo laat bij betrokken werd, is een interessant onderwerp vanuit IT-governanceperspectief.

Al tijdens een allereerste testsessie, waar allerlei disciplines en ontwikkelaars bij betrokken waren, bleek dat de koppeling niet naar verwachting werkte.

Als gevolg hiervan kregen wij de opdracht om een functioneel ontwerp te maken voor de koppeling zoals die wel zou moeten gaan werken.

Na de schets van de gevonden problemen komen we terug op de analyse waar we het functioneel ontwerp op gebaseerd hebben.

Schets van de problemen.

Belangrijke bevindingen waren:

1. Het bleek dat MKS aan meerdere organisatieonderdelen (dus niet alleen aan TZ) vroeg om zorg te gaan verlenen en dus aan meerdere organisatieonderdelen zorgtoewijzingsgegevens gaf.
2. TZ trad steeds op als gesprekspartner van MKS. Geen van de andere betrokken organisatieonderdelen is bij de ontwikkeling van de koppeling betrokken geweest. Wie was nu eigenlijk de dienstverlenende partij?
3. Het was niet duidelijk welke werkprocessen van TZ en anderen door de koppeling ondersteund werden. Hierdoor was het ook niet duidelijk welke gegevens men nodig had voor de uitvoering van die processen.
4. Bij TZ cs was niet duidelijk wie verantwoordelijk was voor de transactie en ook niet wie verantwoordelijk was voor de controle op correctheid van de ontvangen gegevens. Uit

niets is gebleken dat TZ cs geverifieerd heeft of MKS überhaupt wel de gegevens kon aanleveren op basis waarvan zorg verleend kan worden.

5. Ook bij MKS bleek niet duidelijk te zijn welke werkprocessen zij precies uitvoert. Er bleek sprake van meerdere processen. Ook konden klantgegevens vanuit meerdere werkprocessen gemuteerd worden. Wie welke verantwoordelijkheden had was niet duidelijk. Een organisatieschema en procedurebeschrijving ontbrak.
6. Het was niet geheel duidelijk wat de 'trigger' was die de koppeling moest activeren en op welke moment deze activering moest plaatshebben. Wel bleek, dat de 'trigger' die geprogrammeerd was, niet de juiste was. Een van de gevolgen was, dat er in TIS geen nieuwe klanten konden worden aangemaakt.
7. Niet bekend was welke gegevens in TIS verplicht zijn (bijvoorbeeld bij het aanmaken van nieuwe klanten). Er was dus ook geen rekening mee gehouden bij het schrijven van de koppeling en de configuratie van Miracle.

Samengevat:

Van de vragende partij:

- niet duidelijk waar MKS in de organisatie 'hangt';
- niet duidelijk om welke 'dienst' zij verzoeken;
- niet duidelijk wie welke verantwoordelijkheid draagt en waarover;

Van de leverende partij:

- wie is precies die partij
- welke gegevens moeten ontvangen worden en onder welke condities
- niet duidelijk wie welke verantwoordelijkheid draagt en waarover

De analyse

Nadat de problemen geconstateerd waren, kregen we de opdracht een (nieuw) functioneel ontwerp te schrijven. Naar onze opvatting kon dat alleen door een analyse te maken van de bedrijfsprocessen en de afspraken die tussen MKS en TZ gemaakt waren (of zouden moeten zijn). Die afspraken vormen de kern van de koppeling en de gegevensuitwisseling zien wij als het resultaat daarvan.

Bij deze analyse hebben we gebruik gemaakt van DEMO, vooral het transactieprocesmodel. Omdat we ons realiseren dat DEMO nog niet algemeen bekend is, verwijzen we u naar de volgende website: www.demo.nl

Diensten, services en transacties. (of: hoe komen we van DEMO naar SOA?)

Er wordt op meerdere manieren nagedacht over services. Vaak wordt gezegd dat het gaat om gegevensuitwisseling. Maar wij hebben daar een iets andere visie op.

Wij hebben ons hierbij laten inspireren door DEMO van Jan Dietz. De essentie van een organisatie is dat ze bestaat uit mensen die met bevoegdheid en verantwoordelijkheid handelen en onderhandelen. Alleen mensen kunnen afspraken maken, besluiten nemen, oordelen vellen en overeenkomsten aangaan. Die mensen gaan dus overeenkomsten aan, in DEMO-terminologie “transacties”.

In DEMO wordt onder informatisering verstaan: “de overname en de ondersteuning van de uitvoering van taken door computers”.

Services zijn in onze visie dan ook geïnformatiseerde transacties/diensten.

Een transactie kan op de volgende wijze omschreven worden:

Twee businessunits spreken met elkaar af dat de ene partij iets gaat leveren aan de andere partij. Die vraag en levering vormen de kern. Uiteindelijk zal het meestal gaan om gegevens, maar de gegevensuitwisseling is het resultaat van de afgesproken transactie. De transactie is de dienst.

De dienst zal (meestal) uitgevoerd worden met behulp van een technische voorziening. Die technische voorziening is in het geval van een SOA dan de service.

Theoretische benadering van de case:

Er zouden twee conversaties moeten plaatsvinden tussen MKS en TZ. De eerste is een zogenaamde *actagene* conversatie, omdat deze begint met een verzoek en beantwoord wordt met een belofte.

De tweede is een *factagene* conversatie omdat deze begint met een verklaring en beantwoord wordt met een aanvaarding.

Conversatie 1 (notatie in natuurlijke taal)

MKS: TZ, wilt u voor mij zorg leveren aan klant X volgens de bijgevoegde zorgtoewijzing?

TZ: natuurlijk MKS, daar ga ik voor zorgen.

Conversatie 2 (notatie in natuurlijke taal)

TZ: MKS, ik heb zorg geleverd aan klant X volgens de bijgevoegde zorgtoewijzing.

MKS: Ik zie dat u het gedaan heeft, bedankt.

Conversatie 1 (in OER-notatie)

MKS: verzoekt: TZ: er is zorg geleverd aan klant X volgens bijgevoegde zorgtoewijzing.

TZ: belooft: MKS: er is zorg geleverd aan klant X volgens bijgevoegde zorgtoewijzing.

Conversatie 2 (in OER-notatie)

TZ : verklaart: MKS: er is zorg geleverd aan klant X volgens bijgevoegde zorgtoewijzing.

MKS: aanvaardt: TZ: er is zorg geleverd aan klant X volgens bijgevoegde zorgtoewijzing.

De eerste conversatie geeft de opdrachtfase weer, de tweede de resultaatfase.

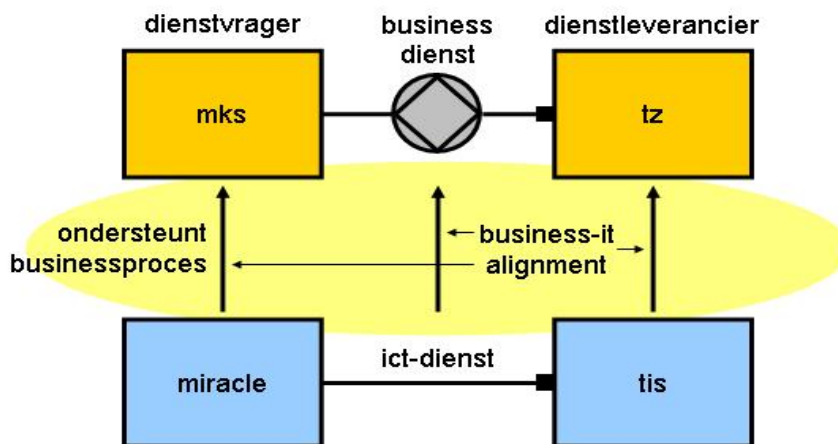
In feite staan hier de kernen van de (organisatie-)diensten die geïnformatiseerd worden in een service. Omdat tussen de opdrachtfase en de resultaatfase een lange tijd (executiefase) kan zitten (enkele maanden zorgverlening) is het ook zeer goed verdedigbaar om voor de opdracht en het resultaat een eigen dienst te beschrijven. In ieder geval moet MKS door TZ op de hoogte gesteld worden dat de gevraagde zorg is geleverd (al was het maar vanwege de control). Uiteraard leidt dit dan in theorie tot 2 services.

De praktijk (van de case).

In de praktijk is alleen het verzoek uit conversatie 1 in een koppeling vertaald. De reden daarvoor ligt in het feit dat TZ, op basis van haar beleid, altijd zorg gaat verlenen als MKS daar met een zorgtoewijzing om vraagt.

De bevestiging dat het verzoek is ontvangen en de verklaring dat de zorg is geleverd, worden als impliciet gedaan beschouwd. Als de zorg niet geleverd kan worden wordt op andere wijze contact opgenomen met MKS. Er kan door MKS echter niet gecontroleerd worden of de zorg daadwerkelijk is geleverd en of dit gebeurd is volgens de opgegeven voorwaarden (zorgtoewijzing).

Hieronder een schematische voorstelling, waarbij gebruik gemaakt is van enkele DEMO-technieken.



Legenda:

Geel vierkant: (MKS en TZ) actoren/systeemkern

Blauw vierkant: (Miracle en TIS) applicatie die actor ondersteunt

Cirkel met vierkant: transactietype

Voorlopig leidde de analyse van de bevindingen tot het volgende leerpunt:
Als de organisatie niet SOA is ingericht, kan ICT ook geen SOA ondersteunen.

Onze stelling is dat het niet mogelijk is om een goede service te realiseren als een Enterprise Architectuur ontbreekt.

Architectuur

In de gezondheidszorg zijn 'architectuur' en SOA als verbijzondering daarvan als organisatie-inrichtingsparadigma nog niet zo algemeen bekend en moeten dan ook vaker verklaard worden. Wij doen dat op de volgende wijze:

De organisatie heeft een strategie die geoperationaliseerd wordt in beleid.

Het beleid wordt uitgevoerd in businessunits, die daarbij werken volgens vastgestelde werkprocessen.

Die werkprocessen kunnen ondersteund worden door applicaties, die beschikbaar worden gesteld via een infrastructuur.

Businessarchitectuur betreft de business-elementen (strategie, beleid, organisatie-indeling, werkprocessen) en ICT-architectuur de ICT-elementen (applicaties en infrastructuur).

Enterprise Architectuur is dan het geheel van beide architecturen, inclusief de onderlinge relaties, afhankelijkheden en de gehanteerde principes.

SOA is een businessarchitectuur waarbij de organisatie en de werkprocessen dienstgeoriënteerd zijn ingericht.

SOA en Meavita West

Om de EAI goed te kunnen realiseren zou de organisatie SOA moeten zijn ingericht. Maar dit roept wel een aantal vragen op:

De **eerste vraag** voor de business is of zij zich dienstgeoriënteerd **wil** organiseren, het heeft zo z'n voor- en nadelen.

Voordeel: als je je organisatie dienstgeoriënteerd hebt ingericht, is daarmee je organisatie in kleinere werkeenheden verdeeld. Dat kan een bijdrage leveren naar de transformatie tot marktgerichte zorgleverancier. Uit de wereld van de profitorganisaties leren we immers dat een dienstgeoriënteerde organisatie zich sneller en efficiënter aan de veranderende markt kan aanpassen.

Nadeel: allerlei dingen die vroeger ook nog wel even konden, kunnen nu opeens niet meer of gaan anders. In die zin veroorzaakt het een (verdere) bureaucrativering van de organisatie.

Van onze ICT-infrastructuuroutsourcing hebben we ook geleerd dat we pas kunnen outsourcen als we het ICT-beheer zelf goed op orde hebben. In SOA-termen: als we het ICT-beheer intern als dienst hebben ingericht.

Wat je niet moet doen is dienstoriëntatie gebruiken om je processen nu eens op orde te krijgen; het is andersom: eerst je processen op orde (dwz dienstgeoriënteerd ingericht) en dan pas de dienstoriëntatie ondersteunen.

Om de tweede vraag te begrijpen eerst even een toelichting op de begrippen runtime- en design-time bewustzijn.

Run- versus designtime bewustzijn

Runtime bewustzijn is het bewustzijn dat ieder van ons heeft in zijn dagelijks werk: steeds als wij in een situatie komen waarin een handeling of beslissing (als dat iets anders is) van ons verwacht wordt ondernemen wij actie (ook geen actie is een actie).

Designtime bewustzijn is het bewustzijn van de planner; deze denkt na over toekomstige situaties en neemt nu beslissingen die dienen te worden uitgevoerd als de nu nog hypothetische situaties zich straks voordoen.

Dit onderscheid is belangrijk omdat automatiseren een verschuiving van run- naar designtime bewustzijn vereist en omdat designtime bewustzijn van een andere orde is dan runtime bewustzijn. Een deel van de problemen die Meavita op automatiseringsgebied, ook in dit integratieproject, meemaakt valt terug te voeren op het veronachtzamen van dit onderscheid. Kort gezegd: als je runtime bewust bent wil dat nog niet zeggen dat je ook designtime bewust bent.

De **tweede vraag** voor de business is daarom of zij zich wel dienstgeoriënteerd **kan** organiseren. Waar een wil is is een weg, maar in ons geval is die weg geen gelopen race. Onze waarneming is dat de business er moeite mee heeft om boven haar runtime bewustzijn uit te stijgen naar designtime bewustzijn. Dit is bijvoorbeeld precies wat er fout ging in de koppeling tussen Miracle en TIS (er ging meer fout maar ik concentreer me nu even op het bewustzijnsstuk). De eerstopgeleverde koppeling werkte niet omdat die ontrafeling naar een paar heldere businessdiensten niet plaats had gevonden.

Gewoon automatiseren is al niet eenvoudig maar SOA gaat dus nog een stap verder, abstracter om precies te zijn. Hoe gaan we als business voor elkaar krijgen dat we dienstoriëntatie succesvol kunnen toepassen in onze organisatie?

Tenslotte.

Tot slot willen wij nog even terugkomen op de titel van dit artikel.

Door de beschreven case leerden we beter te begrijpen wat SOA is, een organisatie-inrichtingsparadigma.

SOA is een wijze waarop de business zich organiseert. De transacties die de verantwoordelijken in de organisatie met elkaar aangaan, leiden tot diensten. ICT ondersteunt de SOA door de diensten te informatiseren naar services.

SOA is een paradigma dat in de Business Architectuur beschreven wordt. Ter ondersteuning van de business heeft ICT een architectuur van applicaties en infrastructuur: de ICT-architectuur. Wij beschouwen de Enterprise Architectuur als het geheel van beide architecturen, inclusief de onderlinge relaties, afhankelijkheden en de gehanteerde principes. Voor de business-IT-alignment is de Enterprise Architectuur dus noodzakelijk.

En zoals al uit de titel blijkt: ICT-ondersteuning van een SOA kan niet zonder EA.

Aan het eind van dit artikel wil ik mijn dank uitspreken aan Jeroen van Beele. Hij inspireerde mij om over architectuur na te denken en leverde door zijn kritische vragen een belangrijke bijdrage aan dit artikel.

Jan van der Meer