

Hertentamen Capita Selecta Bedrijfsinformatica

02/07/2004, 13:30-16:30

Naam:
Studentnummer:
Studie:



Dit tentamen bevat twee reviews van artikelen, plus twee vragen over deze artikelen. Voor elke review kun je maximaal 30 punten scoren. Voor elke andere vraag kun je maximaal 20 punten scoren. In totaal kun je maximaal 100 punten scoren. Je cijfer voor dit tentamen is het totaal aantal punten dat je scoort, gedeeld door 10. Dit tentamen bepaalt 40% van je cijfer voor het vak CSBI.

Schrijf je naam en studentnummer bovenop elke pagina van dit tentamen.

Succes!

Maximale lengte voor antwoorden op vragen 1 en 2: één pagina per antwoord. Denk goed na voordat je begint met schrijven.

1. Review de bijgevoegde papers aan de hand van de bijgevoegde review sjabloon. Mocht er niet voldoende ruimte zijn voor je antwoorden, dan kun je op de achterzijde van de pagina's doorgaan. De artikelen die je moet reviewen heten:
 - a. Functie en Evolutie van Intelligent Agents in de B2B markt
 - b. Case-based architectuur ter verbetering van het softwareontwikkelingsproces
2. Leg uit wat het zwakste punt in de paper "*Case-based architectuur ter verbetering van het softwareontwikkelingsproces*" is en waarom.
3. Leg uit wat het zwakste punt in de paper "*Functie en Evolutie van Intelligent Agents in de B2B markt*" is en waarom.

Case-based architectuur ter verbetering van het softwareontwikkelingsproces

Zijn cases alleen genoeg?

Hans Borst (1202561)
Robert Gilaard (1353411)
Bas van der Raadt (1339745)

Vrije Universiteit,
Amsterdam,
7 mei 2003

Samenvatting

Case-based architectuur is een vorm van softwarearchitectuur waarbij kennis en ervaringen in het maken van softwareoplossingen worden hergebruikt, hetgeen resulteert in een verkort ontwikkelproces. De behoefte van het management om het ontwikkelproces gestructureerd en voorspelbaar te laten verlopen, zodat kosten en planning goed kunnen worden ingeschat, wordt door case-based architectuur echter niet vervuld. Rule-based architectuur vervult deze behoefte wel en dient in combinatie met case-based architectuur ingezet te worden. Hierdoor kunnen de meeste problemen die zich voordoen in het softwareontwikkelingsproces, worden aangepakt.

1 Introductie

Het hergebruik van kennis en ervaring van en het realiseren van een leerproces over architectuur, wordt mogelijk gemaakt door case-based architectuur. De duur van het softwareontwikkelingsproces wordt hierdoor verkort. Deze architectuurvorm richt zich echter niet op het structureren van dit proces, waaraan het management wel behoefte heeft. Daarom dient in combinatie met case-based architectuur een andere architectuurvorm gebruikt te worden, om deze structuur wel te realiseren, namelijk rule-based architectuur.

Dit artikel bespreekt het combineren van deze twee vormen en is als volgt opgebouwd. In sectie 2 zullen de voor- en nadelen van softwarearchitectuur worden besproken. Vervolgens bespreekt sectie 3 twee vormen van softwarearchitectuur, samen met hun specifieke voor- en nadelen. In sectie 4, wordt het gebruik van deze twee vormen in praktijk besproken. Sectie 5 geeft een advies over het gebruik van softwarearchitectuur. Ten slotte geeft sectie 6 de conclusie.

2 Architectuur

Softwarearchitectuur is ontstaan om het ontwerpen van software op een gestructureerde wijze te laten plaatsvinden. Voordat men begint een systeem te programmeren, ontwerpt men eerst een model van het te realiseren systeem. Een model, is een abstracte weergave van de werkelijkheid. Door steeds een bepaald aspect van deze werkelijkheid uit te lichten, wordt de complexiteit van het complete systeem beheersbaar gemaakt.

2.1 Voordelen

Het is voor een bedrijf echter belangrijker te kijken naar de voordelen van softwarearchitectuur, dan wat het precies inhoudt. Het gebruik van softwarearchitectuur tijdens het ontwikkelingsproces resulteert in:

1. systemen die beter aansluiten op de wensen van alle stakeholders,
2. reductie in ontwikkelkosten,
3. kwalitatief hoogwaardige systemen met minder kosten aan onderhoud en
4. overzichtelijke en goed geïntegreerde informatiesystemen.

Het opstellen van architectuurmodellen en -beschrijvingen voordat een systeem gebouwd wordt, zorgt ervoor dat de verschillende wensen van de stakeholders al in een vroeg stadium worden besproken. Deze modellen en beschrijvingen dienen als een soort contract tussen stakeholders, hetgeen zorgt voor duidelijkheid tijdens het ontwikkelingsproces.

Architectuur maakt het leren en hergebruiken van kennis en ervaring uit eerdere projecten mogelijk, waardoor projecten sneller kunnen worden uitgevoerd. Hierdoor worden ontwikkelkosten gereduceerd.

Architectuur richt zich op 'quality by design' en niet op 'quality by maintenance', waarbij de kwaliteit van een systeem moet stijgen tijdens het onderhoud. Men vergeet hierbij echter hierbij ook nieuwe problemen kunnen ontstaan. Bij 'quality by design' wordt op kwaliteit gericht voordat het systeem gebouwd wordt; de onderhoudskosten worden hierdoor gedrukt.

Een goed overzicht van de informatiesystemen binnen een bedrijf, biedt de mogelijkheid om niet alleen systemen onderling te integreren, maar ook de informatiesystemen van verschillende organisaties op elkaar af te stemmen. Hierdoor kunnen bedrijven met samenwerkingsverbanden hun waardeketen beter op elkaar afstemmen en intensiever met elkaar samenwerken, zodat zij sneller producten en diensten kunnen ontwikkelen en aanbieden dan hun concurrentie.

2.2 Nadelen

Architectuur kan een doel op zich worden, terwijl het een middel zou moeten zijn. Men verliest het grote geheel uit het oog, waardoor men zich alleen richt op de eigenschappen van architectuurmodellen en de keuze van de beste methoden en technieken.

Werken onder architectuur brengt een extra activiteit in het ontwikkelen van software. Het lijkt daarom langer te duren voordat men uiteindelijk een werkend systeem heeft, en resulteert daarom ogenschijnlijk in hogere kosten.

Architectuur is een omvangrijk onderwerp, dat op lange termijn ingezet dient te worden. Onderdelen met goede resultaten op korte termijn, kunnen de aandacht op de voordelen op lange termijn weghalen, waardoor architectuur zijn doel mist.

3 Vormen van architectuur

Softwarearchitectuur kent twee vormen: (1) case-based en (2) rule-based. De eerste vorm richt zich op het hergebruik van architectuurn kennis en -ervaring door deze op te slaan, terwijl de tweede vorm de nadruk legt op het structureren van het ontwikkelproces door het opstellen van regels en richtlijnen.

3.1 Case-based

Bij cases wordt informatie over het ontwikkelen van een oplossing, opgeslagen in een database met cases. Een case bestaat uit een situatieschets waarmee een bepaald probleem kan worden geïdentificeerd. Deze schets bestaat uit de eisen aan, en de context van een ontwikkeld systeem, en is gekoppeld aan de beschrijving van de bijbehorende oplossing in de vorm van methodes, design patterns, modellen of hele uitwerkingen in programmeercode.

Voor elke nieuwe situatie die een bedrijf meemaakt, wordt een nieuwe case gemaakt. Het kan zijn dat een case in de ene situatie wel werkt, maar voor een andere situatie niet goed uitpakt. Dan zal de case moeten worden aangepast, waardoor weer een nieuwe case ontstaat.

Case-based architectuur kent drie grote voordelen; het zorgt voor:

1. een leerproces op het gebied van architectuur,
2. hergebruik van architectuurn kennis en -ervaring en
3. het vasthouden van architectuurn kennis binnen een bedrijf.

Het leerproces in een bedrijf zorgt ervoor dat gemaakte fouten zich niet kunnen herhalen. Kennis over verschillende situaties wordt opgeslagen en kan door iedereen binnen het bedrijf worden hergebruikt. Hierdoor hoeft het wiel niet elke keer opnieuw worden uitgevonden, waarmee veel tijd en geld wordt bespaard. Kennis kan steeds verder worden uitgebreid en als een medewerker het bedrijf verlaat, blijft zijn kennis binnen het bedrijf bewaard.

Een nadeel is echter de onduidelijkheid aan het begin van een project, over de manier waarop deze uitgevoerd dient te worden. Deze vorm schenkt weinig aandacht aan structuur in het ontwikkelproces, waardoor het schatten van de kosten en het maken van een planning van het project moeilijk is.

3.2 Rule-based

Bij rules wordt volgens vaste regels en richtlijnen gewerkt, die de uitvoering van een project in goede banen dienen te leiden. Een regel is vast en bindend. Zo moet bijvoorbeeld altijd een bepaald programma gebruikt worden of met een bepaald formaat worden gewerkt. De richtlijnen zijn minder strak. Verschillende

ontwikkelfases moeten bijvoorbeeld in een vaste volgorde worden uitgevoerd; wel kan besloten worden een fase terug te gaan of over te doen.

Rules hebben voor het management duidelijke voordelen. Ze geven structuur aan het ontwikkelingsproces, waardoor makkelijker kan worden bepaald:

1. wat de kosten zijn die een proces met zich meebrengt en
2. hoe de planning van een project eruit ziet.

Door het gebruik van rule-based architectuur kan vrij nauwkeurig worden bepaald waar medewerkers mee bezig zijn, waardoor het plannen van een project verbeterd wordt. Bij het opstellen van de offerte voor een project kunnen de manuren exacter geschat worden, waardoor de kosten beter vast te stellen zijn en het project meer kans maakt om binnen de geschatte tijd te worden afgerond.

4 Praktijk

Rule-based architectuur wordt voornamelijk vanuit het management geïnitieerd, door behoefte aan structuur in het ontwikkelproces. Het wordt met name gebruikt in bedrijven met een omvangrijk gebruik van ICT. Dit zijn bedrijven die voor een groot deel afhankelijk zijn van hun informatiesystemen, zoals banken en verzekeringsmaatschappijen.

Case-based architectuur ontstaat meestal in afdelingen waar systeemontwikkeling wordt uitgevoerd, door behoefte aan hergebruik van kennis en oplossingen. Met name ICT-dienstverleners en softwarebedrijven maken gebruik van case-based architectuur. Het is, vanwege de onderlinge concurrentie, voor deze bedrijven belangrijk de beste oplossing voor een probleem voor de laagste prijs aan te kunnen bieden. Dit kan bereikt worden door het ontwikkelproces te verkorten door middel van hergebruik van kennis.

Beide vormen hebben specifieke voordelen, maar geven geen volledige oplossing voor alle problemen die ontstaan gedurende de uitvoering van een softwareontwikkelingsproces. Rule-based architectuur is niet gericht op hergebruik, waardoor veel werk dubbel wordt gedaan. Case-based architectuur geeft geen structuur aan het ontwikkelproces, waardoor er geen overzicht is.

5 Advies: Combineer beide vormen

Door rules en cases te combineren en niet als rivalen van elkaar te zien, kunnen de voordelen van beide architectuurvormen elkaar aanvullen. De structuur die rule-based architectuur in het ontwikkelproces aanbrengt, wordt gecombineerd met het hergebruiken van architectuurkennis- en ervaring door het inzetten van case-based architectuur. Hierdoor ontstaat een ontwikkelproces dat sneller uitgevoerd kan worden en beter te managen is in termen van planning en kosten.

De volgorde waarin de twee vormen worden geïntroduceerd, ligt aan de specifieke situatie van een bedrijf. De vorm die al aanwezig is, hetzij rules hetzij cases, dient als basis te worden genomen, van waaruit de andere vorm geleidelijk wordt ingevoerd. Mocht de uitgangssituatie nog geen enkele vorm van architectuur bevatten, dan zal het afhangen van de oorsprong van architectuur binnen een organisatie, namelijk vanuit het management of vanuit de systeemontwikkeling.

Bij een ‘top-down’ ontwikkeling vanuit het management, worden meestal eerst rules opgesteld om het ontwikkelproces te structureren. Daarna kan worden besloten architectuurn kennis op te slaan in de vorm van cases, waardoor een leerproces ontstaat waarbij kennis en ervaring kunnen worden hergebruikt.

Vanuit systeemontwikkeling worden ter hergebruik van kennis en oplossingen, cases opgeslagen om het ontwikkelproces sneller te laten verlopen. Dit is een ‘bottom-up’ ontwikkeling waarbij tijdens het opbouwen van deze cases een bepaald aspect zo vaak in herhaling kan vallen, dat wordt besloten dit een als regel of richtlijn vast te leggen. Hierdoor ontstaat gaandeweg een rule-based architectuur.

6 Conclusie

Het gebruik van softwarearchitectuur heeft, door het in betere banen leiden van het ontwikkelproces, duidelijke voordelen voor een bedrijf. Het geeft meer structuur en voorspelbaarheid en de kosten worden gedrukt, omdat de duur van het ontwikkelproces wordt verkort.

De twee verschillende vormen van softwarearchitectuur, rule-based en case-based architectuur, hebben ieder hun eigen doelen en voordelen. In praktijk is het resultaat van softwarearchitectuur niet altijd naar wens, omdat beide vormen nauwelijks in combinatie worden gebruikt en beide geen volledige oplossing zijn voor de problemen die zich kunnen voordoen in het proces van softwareontwikkeling. Wij adviseren daarom de beide vormen te combineren, waardoor deze problemen voor een groot deel aangepakt kunnen worden.

Referenties

- [1] H. de Bruin, H. van Vliet and Z. Baida, *Documenting and analyzing a context-sensitive design space*
- [2] I. Becerra-Fernandez and D.W. Aha, Case-based problem solving for knowledge management systems, *Proceedings of the 12th annual international Florida Artificial Intelligence Research Symposium*, NCARAI Technical Report AIC-99-005
- [3] A.Bahrami, D. Sadowski and S. Bahrami, Enterprise architecture for business process simulation, *Proceedings of the 1998 Winter Simulation Conference*, pp. 1409-1412

Student name:

Student number:

Date Review:

Title of Manuscript: Case-based architectuur ter verbetering van het softwareontwikkelingsproces

Which category describes this manuscript?

- ☐ Practice/Application/Case
- ☐ Study/Experience Report
- ☐ Research/Technology
- ☐ Survey/Tutorial/How-To
- ☐ Other (please specify):

Are the title, the abstract and the introduction appropriate? Please comment.

- ☐ Yes
- ☐ No

Explain:.....
.....
.....
.....

Discuss the contribution made by the manuscript:

Explain:.....
.....
.....
.....

Rate the paper in each of the following areas (1 = low ; 4 = high):

	1	2	3	4	Explain
Technical content	☐	☐	☐	☐	
					
					
					
Business content	☐	☐	☐	☐	
					
					
					

Student name:

Student number:

Originality of claim ☐ ☐ ☐ ☐

Statements are exaggerated YES / NO

Clarity and Organisation ☐ ☐ ☐ ☐

Writing style/Grammar ☐ ☐ ☐ ☐

COMMENTS ON SCIENTIFIC REASONING

Rate the paper in each of the following areas (1 = low ; 4 = high):

Claim:

	1	2	3	4	Explain
Well-formulated	☐	☐	☐	☐	
All concepts are explained	☐	☐	☐	☐	
Generic (1) vs. concrete (4)	☐	☐	☐	☐	

Reasoning:

	1	2	3	4	Explain
Well-structured	☐	☐	☐	☐	

Student name:

Student number:

Shallow (1) vs. profound ☐ ☐ ☐ ☐

Enough sound examples ☐ ☐ ☐ ☐

Generic (1) vs. concrete (4) ☐ ☐ ☐ ☐

Conclusion:

	1	2	3	4	Explain
Based on logical reasoning	☐	☐	☐	☐	

Contains new elements (1=yes, 4=no)	☐	☐	☐	☐	
-------------------------------------	--------------------------	--------------------------	--------------------------	--------------------------	-------

Does the manuscript contain sufficient and appropriate references? Please comment.

	NO			YES	Explain
Sufficient references	☐	☐	☐	☐	

References in conference style	☐	☐	☐	☐	
--------------------------------	--------------------------	--------------------------	--------------------------	--------------------------	-------

References where appropriate	☐	☐	☐	☐	
------------------------------	--------------------------	--------------------------	--------------------------	--------------------------	-------

Suggestions for improvement to the paper:

Student name:

Student number:

Final Recommendation:

- ☐ Definitely accept
- ☐ Borderline
- ☐ Accept only if the following changes are made:.....
- ☐ Reject (poor quality)
- ☐ Other (please specify)

Explain:.....

.....

.....

.....

Any further confidential comments you would like to make to the editor only

.....

.....

.....

Functie en Evolutie van Intelligent Agents in de B2B markt

Capita Selecta BedrijfsInformatica

Sybren de Kinderen, Jorg van Beek

7 mei 2003

Introductie

Door de ontwikkeling van het Internet tot een serieus communicatiemedium komen Intelligent Agents steeds meer in de belangstelling van organisaties te staan. Feit is dat de agenttechnologie in de Business-to-Consumer markt (B2C) veel verder is ontwikkeld dan in de Business-to-Business markt (B2B). Het doel is in de toekomst ook veel meer toepassingen te gaan ontwikkelen voor de B2B, vooral ook omdat 80% van alle transacties in dat segment plaats vindt. Het zal dan wel op een verantwoorde manier moeten gaan functioneren en evolueren. Vandaar ook onze stelling: "Voor een goede toepassing van Intelligent Agents in de B2B, zal men moeten uitgaan van de al bestaande B2C modellen".

Intelligent agents

Om ons paper toegankelijker te maken, zullen we in deze sectie eerst kort het concept Intelligent Agent nader toelichten.

Het begrip intelligent agents

Er zijn veel verschillende definities te vinden van Intelligent Agents, maar één kenmerk ervan is in alle definities terug te vinden: autonomie. Een intelligent agent zou opgevat kunnen worden als een zelfstandig operend stuk software dat verantwoordelijk is voor het uitvoeren van een bepaald (deel)proces. In dat opzicht lijken de principes van Intelligent Agents veel op die van conventionele programma's. Een belangrijke eigenschap waarin agents zich echter van deze programma's onderscheiden, is het feit dat ze niet alleen reactief zijn (dus alleen taken uitvoeren wanneer dat van hen gevraagd wordt) maar ook pro-actief kunnen optreden en dus zelf actie kunnen ondernemen. Ook kan een agent binnen een netwerk samenwerken met andere autonome agents en eventueel opereren op andere systemen. Of een agent zich ook daadwerkelijk over een netwerk verplaatst en communiceert met andere agents, hangt af van de toepassing ervan, maar het is wel degelijk iets waar ze toe in staat zijn. Intelligent Agents hebben een vaak een beperkte taakomschrijving en het zijn dan ook vaak kleine stukjes software. Ze worden nu vooral in kleinschalige toepassingen gebruikt, zoals shopbots, crawlers (doorzoeken websites voor zoekmachines) en in recommender-systemen (die gebruikers producten kunnen aanbieden).

Dat ze nog niet op grotere schaal worden ingezet (bijvoorbeeld voor het afhandelen van transacties) komt omdat er een aantal problemen spelen die een verregaand gebruik van agents tegenhouden; hierop wordt in de volgende paragraaf ingegaan.

Kansen en bedreigingen voor intelligent agents

De huidige toepassing van Intelligent Agents beperkt zich dus voor het grootste deel tot de B2C-markt, waar ze worden toegepast binnen de verschillende gebieden van het CBB-model (daarover later meer). Er zit echter een grote potentie voor Intelligent Agents in B2B E-commerce, aangezien de legacy-systemen zoals EDI (electronic data interchange) binnen VAN (value-added networks) die nu gebruikt worden voor het elektronische zakendoen sterk verouderd, inflexibel en duur in het onderhoud zijn.

Veel organisaties zien agents als een technologie met veel potentie en enkele van hen (bijvoorbeeld Cisco) zijn ook al begonnen met de toepassing ervan. Dat agents zoveel voordeel op kunnen leveren voor de B2B markt, komt hoofdzakelijk door het feit dat hun eigenschappen, zoals die in voorgaande paragraaf genoemd zijn, hen in staat stellen routinematige taken goed en efficiënt uit te voeren zonder al te veel menselijke tussenkomst.

Dit levert over het algemeen een flinke kostenreductie, alsook een lagere turnaround time op binnen het bedrijfsproces waar zij worden toegepast.

Agents worden, ondanks hun voordelen, nog niet wijdverbreid toegepast. Het belangrijkste technische probleem, dat de ontwikkeling van Intelligent Agents, tegenhoudt is het gebrek aan één gestandaardiseerde taal waarmee agents met elkaar kunnen communiceren maar die ze ook kunnen gebruiken om informatie te interpreteren. Ontwikkelingen van een dergelijke standaardtaal zijn onderweg, maar het zal nog jaren duren tot deze ook daadwerkelijk op grote schaal in gebruik zal worden genomen. Door het gebrek daaraan lopen agents nu tegen beperkingen op.

Ook zit er een probleem in het feit dat de veiligheid van agents nu nog te wensen overlaat; zo kan een agent zich bijvoorbeeld anders voor doen dan ze zijn (iets wat ook wel *masquerading* genoemd wordt) en zo informatie verkrijgen die ze eigenlijk niet horen te krijgen (er zijn nog wel meer problemen met de veiligheid van agents, zie hiervoor ook Turban et al.¹). Hierdoor komen we ook meteen op het probleem van privacy, aangezien agents gevoelige gegevens kunnen vergaren die de privacy van een persoon of instelling kunnen schaden. Verder is er nog een probleem met het eigendom van informatie. Als agents namelijk op grote schaal worden toegepast en informatie van het web halen, aan elkaar doorgeven en met andere gegevens combineren, wordt het op een gegeven moment lastig (zoniet onmogelijk) om te bepalen van wie de informatie oorspronkelijk afkomstig is. Hierdoor kunnen fundamentele rechten, zoals copyright, in het geding komen.

Tenslotte is er gebrek aan een goede theoretische fundering (zoals CBB in B2C E-commerce) voor de opbouw en classificatie van Intelligent Agents. Dit is echter wel van groot belang voor de verdere ontwikkeling ervan en is dan ook iets waarop de rest van dit paper zich zal concentreren.

B2C en intelligent agents; het CBB model

Zoals in het vorige hoofdstuk al kort is aangestipt, worden agents nu vooral toegepast in de B2C markt, waardoor ze als een stuk volwassener kunnen worden beschouwd dan in de B2B. Dit heeft als oorzaak dat de B2C minder hoge eisen stelt aan het netwerk waarover agents zich verplaatsen, maar het komt ook doordat er binnen de B2C al een goede theoretische fundering voor de classificering van agents bestaat. Op een van de belangrijkste modellen voor classificering, genaamd het Customer Buying Behavior (CBB) wordt in dit hoofdstuk dieper ingegaan.

CBB in zijn oorspronkelijke context

Het CBB is oorspronkelijk een marketing tool die gebruikt wordt om de verschillende behoeften van een consument in kaart te brengen. Met de opkomst van agent technologie in de B2C bleek het echter ook uitstekend bruikbaar om agents aan de hand van de functies die zij vervullen te classificeren. Hierop wordt in de volgende paragraaf dieper ingegaan.

Het oorspronkelijke CBB model bestaat uit 6 opeenvolgende fasen, die het aankoopgedrag en daarmee ook de behoeften van een consument aangeven. Dit zijn:

¹ TURBAN et al., Are Intelligent E-Commerce Agents Partners or Predators?, Comm. of the ACM, Mei '02

- **Needs Identification:** De consument krijgt, eventueel door middel van informatie uit een advertentie, behoefte aan een product of dienst; het is nog niet concreet duidelijk welke dienst/welk product dit is
 - **Product Brokering:** Dit is de fase waarin de consument bepaald wat hij/zij precies aan wil schaffen; de behoefte uit de needs identification-fase wordt hier dus vertaald naar een specifiek product/dienst
 - **Merchant Brokering:** Hier wordt bepaald van welke afnemer het product uit de product brokering fase wordt gekocht
 - **Negotiation:** Deze fase, die overigens niet altijd doorlopen wordt in het aankoopproces, wordt gebruikt voor de onderhandeling tussen consument en leverancier over de voorwaarden van de transactie. Hierbij kan bijvoorbeeld gedacht worden aan de veiling van een product onder een aantal geïnteresseerde consumenten
 - **Purchase and Delivery:** Hier kan de klant de voorwaarden van de levering van het product/de dienst specificeren
 - **Product Service and Evaluation:** Hier kan de consument zijn/haar tevredenheid uiten.
-

CBB en intelligent agents

Zoals gezegd kan het CBB model toegepast worden voor de classificatie agents binnen de B2C markt. In deze paragraaf wordt bekeken hoe het CBB hiervoor gebruikt kan worden.

De fase needs identification kan vertaald worden naar agents die de behoefte aan een product of dienst bij de consument aanwakkeren. Hierbij kan gedacht worden aan de eerder genoemde toepassing van agents binnen recommender systemen.

De daarop volgende drie fasen kunnen worden vertaald naar agents die als mediator tussen consument en leverancier dienst doen; dit wil zeggen dat de agent dan als 'tussenpersoon' dient bij de onderhandeling tussen consument en leverancier waarbij deze een groot deel van de taken binnen deze fasen op zich neemt. Een voorbeeld van een huidige toepassing van mediation-agents zijn shopbots.

Binnen de laatste twee fasen uit het model zijn er nog niet echt veel toepassingen van intelligent agents te vinden, maar er wordt wel onderzoek gedaan hoe ook hier agents kunnen worden toegepast.

B2B en intelligent agents; functionaliteit en evolutie

Nu we hebben aangegeven op welke gebieden agents actief zijn in de B2C, gaan we dat op twee manieren gebruiken om de stelling te bewijzen, en dus aan te tonen dat B2C de basis moet zijn voor agents in de B2B. De eerste manier is kijkend naar de overeenkomst van de functionaliteit van de agents in de beide business gebieden. De andere invalshoek is dat we de agents in de B2C als basis zien voor de ontwikkeling van agents in de B2B.

Het Agent-Mediated B2B Process Model.

In de B2B is er ook een bepaalde manier hoe een aankoopproces verloopt. Dit model is in theorie haast identiek aan het CBB model. Blake² heeft aan de hand van het CBB model het Agent-Mediated B2B Process Model gedefinieerd. Bij elke fase in dat model past ook een fase uit het CBB model (tabel 1).

AMBPM	Omschrijving	CBB
Requisitioning	agent bepaald welke producten of diensten nodig zijn	Needs Identification
Request for Quote	agent bepaald welke leverancier het gevraagde leveren	Product Brokering
Vendor Selection	agent bepaald welke producten – leverancier combinatie de juiste is	Merchant Brokering
Purchase Order	agent bestelt het gevraagde	Negotiation
Delivery	agent bepaalt voorwaarden bij de aankoop	Purchase and Delivery
Payment Processing	agent voert betaling uit	Purchase and Delivery
Evaluation and Evolution	agent evalueert het proces	Product Service & Evaluation

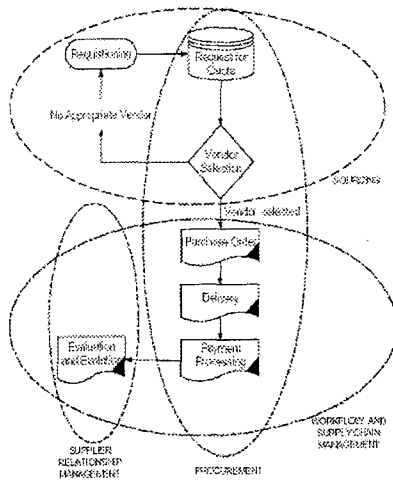
Tabel 1. Agent Mediated Process Model en het CBB

We hebben nu, met het CBB uit de B2C, een model gedefinieerd dat aangeeft hoe een agent in de fases van de B2B een rol kan spelen. Volgens Blake³ is kan men dit model ruwweg samenvatten tot 4 hoofdgroepen waarin agents actief zijn in de B2B (figuur 1).

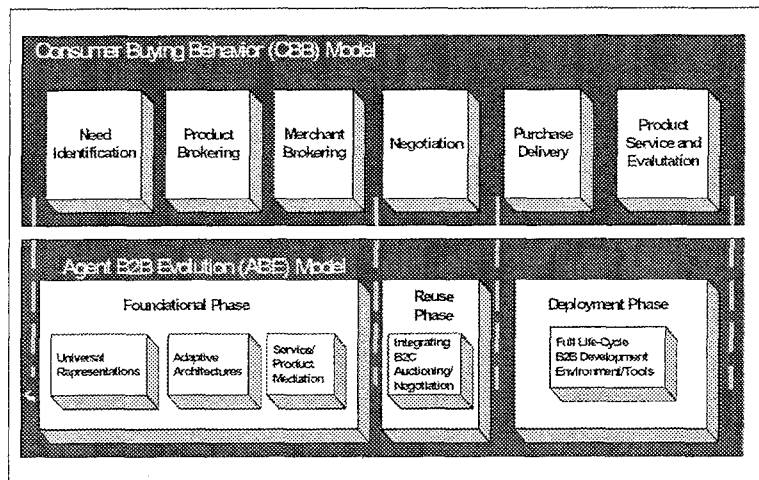
De eerste groep is de Agent Support voor Strategic Sourcing, hier gaat het om het vaststellen wat nodig is en welke leveranciers voorhanden zijn om het te leveren (Data Analyse). Hierop volgt de Electronic Procurement fase, waarin de agents vaststellen welke aanbieder in het gegeven geval de juiste is, aangenomen dat de juiste service of product al bekend is. De derde groep beslaat alle na-aankoop beslissingen, zoals garantie, afleveren etc. (Agent Mediated Supply-chain and Workflow automation). De laatste groep is de Agent Based supplier Relationship Management, die vergelijkbaar is met het Customer Relationship Management in de B2C, maar nu vooral gaat om onderhouden van relaties met leveranciers.

² BLAKE, Innovations in Software Agent-Baed B2B Technologies, 2001

³ BLAKE, B2B Electronic Commerce: Where Do Agents Fit In?, AAAI-2002 Workshop



figuur 1 De 4 B2B agent gebieden



figuur 2 Het Agent B2B Evolution model

De evolutie van agents in de B2B

Aan de hand van het CBB-model heeft Blake [1] ook een model (figuur 2) gemaakt van de evolutie die B2B agenttechnologie zal moeten doorlopen, om een breed geaccepteerde en toegepaste techniek te worden. Dit model heet het Agent B2B Evolution model (ABE) en is een chronologische set fasen die moeten worden doorlopen. Het heeft overeenkomst met het CBB omdat de eerste fase zich afspeelt in de eerste fasen van het CBB en de tweede en derde fasen in latere fasen van het CBB. Dat is ook voor te stellen, omdat zaken als het herkennen van een behoefte door een agent, het zoeken naar leveranciers etc, nog relatief weinig complex is, maar dingen als agents die onderhandelen, betalingen uitvoeren en dergelijke al een stuk complexer is.

Zoals in figuur 2 te zien is zijn er in het evolutiemodel 3 hoofdfasen die wij kort zullen bespreken.

Allereerst de Foundational Phase. Deze fase kenmerkt zich doordat hierin de huidige B2C infrastructuur wordt verbeterd om B2B activiteiten aan te kunnen. Dat is nodig om de omvang van de transacties aan te kunnen en rekening te houden met privacy en veiligheid. Er zijn in deze fase drie hoofdactiviteiten, te weten het verkrijgen van een universele representatie (het ontwikkelen van een algemene agenttaal om communicatie te vergemakkelijken), het aanpassen van de architecturen zodat ze toegankelijk zijn voor agents en de mate van communicatie en coördinatie bepalen die nodig is om services te integreren. Bij het laatste gaat het om het vaststellen van de manier hoe agents zich gedragen c.q. opstellen.

De volgende twee fasen zijn de Reuse Phase en de Deployment phase. De eerste zit vooral in het negotiation gebied van het CBB. Het gaat hier om het ontwikkelen van onderhandelingsstrategieën voor agents in de B2B op basis van de al bestaande B2C strategieën. Deze fase is ontzettend complex. De laatste evolutiefase is van belang om de B2B agenttechnologie tot een doorzichtige standaard te krijgen, waarbij het relatief gemakkelijk is om agents toe te voegen. De focus is dus op het construeren van een omgeving waarin alles op de activiteiten van agents is afgestemd

Conclusie

We hebben gezien dat de B2C agenttechnologie en het daarbij behorende CBB model op twee manieren een basis is voor de B2B technologie. Ten eerste kan het op functioneel gebied een basis zijn, wat resulteert in het Agent-Mediated B2B Process Model. Maar ook is het een basis voor het voorschrijven van een verantwoorde evolutie voor agenttechnologie binnen de B2B markt, zoals te zien in het ABE model. Met deze twee modellen is een goede basis gelegd voor de verdere ontwikkeling van geavanceerde B2B toepassingen, want er moet worden uitgekeken voor een te snelle, ongecontroleerde

ontwikkeling, in verband met problemen op het gebied van privacy, veiligheid en eigendom van informatie, die vooral van groot belang zijn in het B2B segment.

Student name:

Student number:

Date Review:

Title of Manuscript: *Functie en Evolutie van Intelligent Agents in de B2B markt.*

Which category describes this manuscript?

- ☐ Practice/Application/Case
- ☐ Study/Experience Report
- ☐ Research/Technology
- ☐ Survey/Tutorial/How-To
- ☐ Other (please specify):

Are the title, the abstract and the introduction appropriate? Please comment.

- ☐ Yes
- ☐ No

Explain:.....
.....
.....
.....

Discuss the contribution made by the manuscript:

Explain:.....
.....
.....
.....

Rate the paper in each of the following areas (1 = low ; 4 = high):

	1	2	3	4	Explain
Technical content	☐	☐	☐	☐	
					
					
Business content	☐	☐	☐	☐	
					
					
					

Student name:

Student number:

Originality of claim ☐ ☐ ☐ ☐

Statements are exaggerated YES / NO

Clarity and Organisation ☐ ☐ ☐ ☐

Writing style/Grammar ☐ ☐ ☐ ☐

COMMENTS ON SCIENTIFIC REASONING

Rate the paper in each of the following areas (1 = low ; 4 = high):

Claim:

	1	2	3	4	Explain
Well-formulated	☐	☐	☐	☐	

All concepts are explained	☐	☐	☐	☐	
----------------------------	--------------------------	--------------------------	--------------------------	--------------------------	-------

Generic (1) vs. concrete (4)	☐	☐	☐	☐	
------------------------------	--------------------------	--------------------------	--------------------------	--------------------------	-------

Reasoning:

	1	2	3	4	Explain
Well-structured	☐	☐	☐	☐	

Student name:

Student number:

Shallow (1) vs. profound ☐ ☐ ☐ ☐

Enough sound examples ☐ ☐ ☐ ☐

Generic (1) vs. concrete (4) ☐ ☐ ☐ ☐

Conclusion:

Based on logical reasoning

1	2	3	4	Explain
☐	☐	☐	☐	

Contains new elements (1=yes, 4=no) ☐ ☐ ☐ ☐

Does the manuscript contain sufficient and appropriate references? Please comment.

Sufficient references

NO			YES	Explain
☐	☐	☐	☐	

References in conference style ☐ ☐ ☐ ☐

References where appropriate ☐ ☐ ☐ ☐

Suggestions for improvement to the paper:

Student name:

Student number:

Final Recommendation:

- ☐ Definitely accept
- ☐ Borderline
- ☐ Accept only if the following changes are made:.....
- ☐ Reject (poor quality)
- ☐ Other (please specify)

Explain:.....
.....
.....
.....

Any further confidential comments you would like to make to the editor only

.....
.....
.....