
Intreerede

Uitgesproken op 12 december 2003
aan de Technische Universiteit Eindhoven

onzichtbare architecturen

tussen chaos en structuur
in e-business

prof.dr.ir. P.W.P.J. Grefen

Inleiding

Voor Ria

Mijnheer de Rector Magnificus, collega's, familie en vrienden, dames en heren,

De rol van informatie wordt steeds belangrijker in de hedendaagse maatschappij. Waar klassieke economen van drie basisfactoren voor productie spreken – grondstoffen, arbeid en kapitaal – voegen moderne economen informatie als vierde factor toe. Diensten en producten worden steeds complexer, zowel in de consumentenmarkt als in de zakelijke markt – denkt u maar aan fysieke producten, zoals auto's en vliegtuigen, aan logistieke en bancaire diensten en hybride producten in de telecomsector. De organisaties die deze diensten en producten leveren, worden ook steeds groter en complexer. Steeds vaker bestaan deze uit samenwerkingsverbanden tussen autonome deelorganisaties. De hoeveelheden informatie die binnen een organisatie moeten worden verwerkt, worden dan ook steeds groter en de complexiteit van de informatie neemt toe. Om deze reden worden geautomatiseerde informatiesystemen steeds belangrijker om informatieverwerking beheersbaar te houden. Maar tevens worden deze systemen zelf ook steeds groter en complexer en dus moeilijker beheersbaar.

Deze ontwikkeling is het meest duidelijk in de moderne e-business – het voeren van bedrijven op een wijze die zonder geautomatiseerde informatiesystemen onmogelijk is, ook wel aangeduid als 'IT-enabled business'. In deze bedrijven is informatie vaak de onderscheidende productiefactor – met de term 'productie' in de brede zin des woords opgevat – en is het klassieke trio productiefactoren meer naar de achtergrond geschoven. Soms zijn processen in welke grondstoffen en arbeid een belangrijke rol spelen zelfs geheel aan toeleveranciers uitbesteed, zodat een e-business bedrijf alleen de coördinatie van bedrijfsprocessen via de informatieverwerking uitvoert. Zo kan een bedrijf een productassortiment aanbieden zonder zelf voorraden te hebben, zelf producten te vervoeren of zelfs maar betalingen af te handelen.

In e-business is geautomatiseerde informatieverwerking vaak 'end-to-end' georganiseerd – dat wil zeggen dat het gehele primaire bedrijfsproces op geïntegreerde wijze wordt ondersteund door



Het ontstaan van chaos

informatiesystemen. Dit vereist dat vele informatiesystemen met elkaar gekoppeld worden, zowel front-end systemen, die de interactie met externe partijen faciliteren, als back-end systemen, die de kern van de interne bedrijfsvoering ondersteunen. De vereiste niveaus van efficiëntie, aantallen transacties en vereiste 24x7 openstelling vergen informatieverwerking met zo weinig mogelijk menselijke tussenkomst. Denkt u in uw persoonlijke sfeer bijvoorbeeld maar aan het moderne internetbankieren, waarbij u als klant alleen nog maar met informatiesystemen interacteert.

Informatiesystemen worden in op e-business geënte bedrijven dan ook een cruciaal productiemiddel. Eén dat bepaalt of een bedrijf succesvol is of niet. In andere woorden, informatiesystemen zijn onderdeel van de basisinfrastructuur voor management van operationele bedrijfsprocessen. Het spreekt voor zich dat een zo belangrijk productiemiddel van goede kwaliteit moet zijn – bijvoorbeeld snel, betrouwbaar en onderhoudbaar. Goede kwaliteit begint bij goede structuur – en structuur komt tot uitdrukking in architectuur.

In deze rede bespreek ik de rol van architecturen in de realisatie van complexe informatiesystemen, meer in het bijzonder e-business systemen in het business-to-business oftewel B2B segment. Ik begin met het bespreken van het gevaar van het ontstaan van chaos – duidelijk een ongewenste situatie. Ik geef daarna aan hoe chaos vermeden kan worden door het aanbrengen van structuur. Het centrale begrip in deze structuur is architectuur – ik bespreek hiervan het wezen.

Vervolgens bespreek ik de gevaren van een overvloed aan structuren: 'Back to Babylon'. Ik zal tenslotte uiteenzetten hoe dit alles toegepast kan worden in een gebied waarin ik de afgelopen jaren onderzoek heb gedaan: dynamic service outsourcing. Ik sluit af met enkele conclusies. Hierin zal ik onder andere de rol van architectuur in het onderwijs van informatiesystemen aansnijden.

Voordat we ons aan het ontstaan van chaos gaan wijden, eerst een kleine maar belangrijke voetnoot. Zoals gezegd richt ik mij in deze rede met name op e-business systemen. Dit neemt niet weg dat veel van het besprokene ook van toepassing is op complexe informatiesystemen in een niet-commerciële context, bijvoorbeeld bij de overheid of in de gezondheidszorg – tegenwoordig vaak aangeduid als e-government respectievelijk e-health. De focus op e-business zorgt echter voor een prettige stroomlijn in mijn betoog.

In moderne grote bedrijven zien we in het algemeen een uitbreiding van de toepassing van geautomatiseerde informatiesystemen van secundaire, ondersteunende bedrijfsprocessen naar primaire, bedrijfskritische processen. Dit geldt in klassieke productiebedrijven, maar zeer zeker in gegevensintensieve bedrijven, zoals banken en verzekeringsbedrijven. Waar vroeger informatiesystemen vooral de boekhouding en personeelsadministratie ondersteunden, zien we tegenwoordig een geautomatiseerde ondersteuning van de belangrijkste kernprocessen van bedrijven, zoals het afhandelen van klantorders, het bewaken van productieprocessen of het aansturen van logistiek. Dit legt een steeds grotere druk op de kwaliteit van de informatiesystemen: het gedurende enkele uren uitvallen van een personeelsadministratiesysteem is hinderlijk, het uitvallen van een klantorderafhandelingssysteem kan desastreus zijn voor een bedrijf dat in een snelle, competitieve markt opereert.

We zien tevens een groeiende complexiteit van informatiesystemen als gevolg van een aantal factoren. Op de eerste plaats staat de toenemende functionaliteit: steeds meer bedrijfsfuncties moeten individueel ondersteund worden en steeds meer bedrijfsfuncties moeten aan elkaar gekoppeld worden. Zoals gezegd, geldt dit vooral binnen de e-business, waarin menselijke tussenkomst in bedrijfsprocessen zo veel mogelijk vermeden wordt. Op de tweede plaats zien we toenemende eisen met betrekking tot beschikbaarheid – zowel geografisch als in de tijd. Geografische beschikbaarheid betekent dat informatiesystemen op vele plaatsen en via vele kanalen toegankelijk moeten zijn, tegenwoordig zelfs via mobiele terminals. Dit betekent dat systemen voorzien moeten zijn van geavanceerde mechanismen voor distributie van functionaliteit en gegevens, soms via interne netwerken, soms via publieke netwerken als het Internet. Beschikbaarheid in de tijd betekent dat systemen in veel situaties meer dan 99% van de tijd 'in de lucht moeten zijn' – soms zelfs veel meer dan 99% (hoe tegenstrijdig dit ook klinkt). Dit wil zeggen dat er nauwelijks ruimte is voor reparaties en onderhoud. Dit heeft grote invloed op de structuur van systemen: zo kan replicatie van systeemdelen noodzakelijk zijn. Op de derde plaats is er een hoog niveau

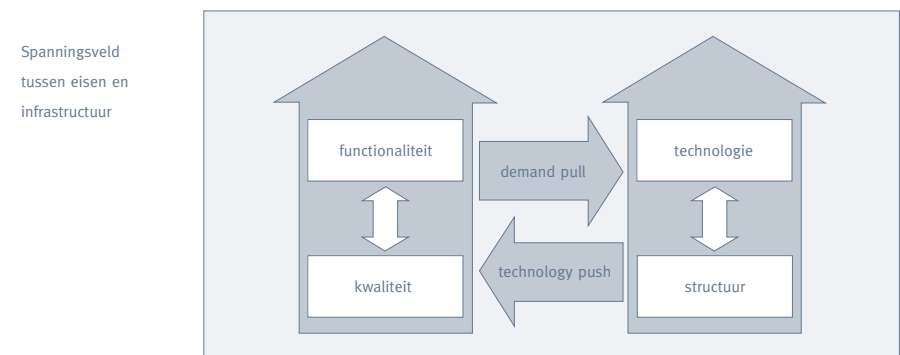
van gewenste interoperabiliteit tussen systemen als gevolg van de end-to-end automatisering die ik al noemde. Deze interoperabiliteit speelt zelfs over organisatiegrenzen heen, bijvoorbeeld in supply chains en service markets. Zoals ik later zal bespreken, moeten in deze laatste context systemen zelfs dynamisch aan elkaar gekoppeld worden, zelfs wanneer ze van heterogene aard zijn. Het koppelen van heterogene systemen van autonome organisaties is echter geen sinecure. Probeert u maar eens uw koelkast te koppelen aan de televisie van de buurman om bier koud te hebben voor een voetbalwedstrijd. Op de vierde plaats worden steeds hogere eisen gesteld aan de flexibiliteit en uitbreidbaarheid – zeg maar dynamiek – van een architectuur en de onderliggende infrastructuur om markt- en technologiebewegingen te kunnen volgen. Markt- en technologiebewegingen veroorzaken veranderingen in productassortimenten, bedrijfsprocessen en gebruikte toegangskanalen. Technologiebewegingen bieden nieuwe vormen van ondersteuning van processen, nieuwe toegangskanalen, beter gebruik van media, enzovoorts. In e-business is verandering de enige constante.

Met de groeiende complexiteit worden informatiesystemen vaak aangepast aan nieuwe eisen door er stukken aan te bouwen: nieuwe modules met additionele functionaliteit, nieuwe koppelingen tussen bestaande modules of nieuwe interfaces naar andere systemen. Deze aanbouwaanpak leidt maar al te vaak tot gedrochten. Hierbij moet ik denken aan een bezoek aan de Catalaanse stad Manresa – ooit voor het Europese WIDE project – waar men woningen aan-, uit- en opbouwt wanneer de familie groeit. In het positieve geval leidt dit tot een minder fraai stadsgezicht, in het negatieve geval tot gevaar van instorting. In Nederland kan men dezelfde ervaring krijgen van de ‘achterkant’ van een binnenstad wanneer men met de trein een grote stad binnenrijdt: men is getuige van een ‘rijke’ schakering van aan- en opbouwsele die de aanblik en structuur van de bebouwing niet bepaald verbeteren. Op het gebied van informatiesystemen is het vaak helaas niet veel anders: ook hier komen we de veranda’s, aanbouwkeukens, dakkappellen en duiventillen tegen. En ook hier houdt het aan- en opbouwen ooit op wanneer de zaak dreigt in te storten. Herbouw van systemen die niet meer onderhoudbaar zijn, is vaak echter een probleem – zeker in de e-business waar uitval van systemen meestal niet mogelijk is.

Door deze ontwikkelingen is er een spanningsveld ontstaan tussen enerzijds gewenste functionaliteit en vereiste kwaliteit van informatie-systemen (de zijde van de functionele en non-functionele eisen) en

anderzijds beschikbare informatie- en communicatietechnologie en de structuur waarin deze is toegepast (de zijde van de infrastructuur). Dit spanningsveld is continu in beweging omdat zowel de eisen-zijde zich ontwikkelt (de zogenaamde ‘demand pull’ oftewel ‘requirements pull’) als de technologie-zijde (de zogenaamde ‘technology push’), zoals geïllustreerd in Figuur 1. Verandering is de enige constante, chaos het gevaar. Architectuur is de integrerende factor om met dit spanningsveld om te gaan.

figuur 1





Het concept 'architectuur' heeft in het algemeen verschillende betekenissen. De Van Dale [Gee84] onderscheidt vier betekenissen die ik kort kan aangeven als bouwkunst, bouwstijl, constructie en bouwsel. Het gebruik van de term 'architectuur' in een traditionele context kan dus tot verwarring leiden. Binnen de veel jongere en minder ervaren wereld van de informatiesystemen bestaat er nog meer verwarring over deze term. Vatten we de term 'architectuur' op als 'constructie' of 'bouwsel', dan zien we op de eerste plaats vaak het onderscheid tussen hardwarearchitectuur en softwarearchitectuur. Het eerste gebruik van de term is al veel langer ingeburgerd, bijvoorbeeld als 'de architectuur van een processor'. Waarschijnlijk is dit terug te voeren op de fysieke structuur van hardware – zoals gebouwen bij de traditionele architectuur. Maar wij zijn nu vooral geïnteresseerd in onzichtbare architecturen, dus we richten ons op softwarearchitecturen. Hier zien we het onderscheid tussen enerzijds architectuur van software in het domein van de software engineering en anderzijds architectuur in het domein van het ontwerpen van informatiesystemen – met verschillende aggregatie- en abstractieniveaus. We kunnen een informatiesysteemarchitectuur afbeelden op een softwarearchitectuur en daarbij bijvoorbeeld het gebruik van middleware in kaart brengen. We kunnen een informatiesysteemarchitectuur ook afbeelden op een hoog-niveau hardwarearchitectuur, waarbij we bijvoorbeeld de allocatie bepalen van informatiesysteemmodules voor de verkrijging van redundantie ten behoeve van de betrouwbaarheid van systemen of de verkrijging van geografische spreiding in functionaliteit. Vatten we de term 'architectuur' op als 'bouwkunst' of 'bouwstijl', dan zien we een relatief korte geschiedenis in het gebied van de informatiesystemen. Bouwstijlen worden in de traditionele bouwkunst al vele eeuwen gebruikt. In de klassieke oudheid werden enkele millennia geleden al diverse bouwstijlen onderscheiden. Daarentegen worden bouwstijlen in de ontwikkeling van informatiesystemen pas sinds enkele jaren onderkend. Hetzelfde geldt voor de bouwkunst, die binnen de informatiesysteemwereld in een nogal onvolwassen stadium verkeert.

Architectuur wordt in de informatica vaak opgevat als enkelvoudige structuur, dus als enkelvoudige 'constructie' of 'bouwsel'. Zo stelt het bekende boek van Shaw en Garlan over softwarearchitectuur [Sha96]: "De architectuur van een softwaresysteem definieert dat systeem in termen van computationele componenten en interacties tussen deze componenten". Kort door de bocht zegt dit, dat een architectuur bestaat uit een enkel plaatje bestaande uit blokjes en pijltjes tussen deze blokjes. In de wereld van de informatiesystemen is de benadering van een architectuur bestaande uit een stelsel van aspectarchitecturen completer, waarbij we het dus hebben over meervoudige 'constructie' en 'bouwstijl'. Een bruikbaar stelsel van aspectarchitecturen bestaat bijvoorbeeld uit vijf elementen: de systeemarchitectuur die de structuur van de te ontwerpen software beschrijft, de configuratiearchitectuur die de structuur van het onderliggende platform beschrijft, de gegevensarchitectuur die de gegevensmodellen bevat, de communicatiearchitectuur die de netwerkinfrastructuur beschrijft en tenslotte de organisatiearchitectuur die vastlegt wat de structuur is van de organisatie voor beheer en onderhoud van informatiesystemen [Tru90]. In wezen is dit vergelijkbaar met verschillende aspecten die we in traditionele architectuur kunnen zien: een gebouw heeft bijvoorbeeld een externe architectuur met verschillende aanzichten, een binnenhuisarchitectuur en een infrastructuurarchitectuur die bepaalt hoe leidingen en voorzieningen geplaatst zijn. Zoals in de traditionele architectuur kunnen aspectarchitecturen van informatiesystemen niet los van elkaar gezien worden. De onderlinge afhankelijkheden maken het ontwerp van een complete architectuur vaak complex.

Naast de aandacht voor aspecten van architecturen behoort er aandacht te zijn voor betekenis en bedoeling van architecturen. Dat wil zeggen, dat naast syntax ook semantiek en pragmatiek van een architectuur van belang zijn. Dit hangt samen met het gebruik van architectuurstijlen, die specifieke structureringsprincipes in zich bergen. In een vak over architectuur van informatiesystemen vul ik de definitie van Shaw en Garlan dan ook aan met: "... vanuit het gezichtspunt van specifieke aspecten van dat systeem en gebaseerd op specifieke structureringsprincipes" [Gre02]. Dit is niet anders dan in de traditionele architectuur, waarin een architect ook een bedoeling heeft bij de keuze van vormen. Zoals inmiddels duidelijk moge zijn, kunnen we inzichten in architecturen van informatiesystemen koppelen aan inzichten uit andere architectuurvelden, waar men veel meer ervaring heeft met

Het aanbrengen van structuur

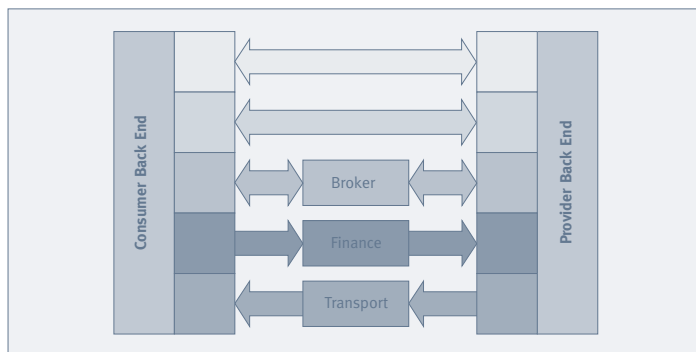
het aanbrengen van structuur in chaos. Daarbij moeten we wel de verschillen tussen architecturen in verschillende gebieden voldoende voor ogen houden. Herinner u dat we geïnteresseerd zijn in onzichtbare architecturen. Architectuur met een esthetisch oogmerk is dus voor ons in eerste instantie niet zo relevant. De Amerikaanse architect Johnson stelde ooit: “Architecture is the art of how to waste space” [Joh64]. Dit gaat voor ICT niet op: software neemt geen plaats in en voor hardware zien we miniaturisering (zie bijvoorbeeld mijn elektronische agenda) – dus hierover hoeven we ons blijkbaar geen zorgen te maken. In de traditionele architectuur zien we wel veel aandacht voor esthetiek – een fraaie architectuur voldoet beter aan een compleet eisenpakket. Het bedrijven van architectuur wordt dan ook vaak gezien als een combinatie van kunst en kunde. In de architectuur van informatiesystemen kunnen we tot op zekere hoogte een parallel trekken. Zoals gezegd, hebben we het hier over onzichtbare architecturen – er is dus geen esthetiek van deze architecturen in de wereld van de gebruikers van de systemen. Maar we moeten hier niet té reductionistisch zijn. Men kan wel van esthetiek spreken in de wereld van de ontwerpers van de architecturen. Mijns inziens geldt hier zeker dat een fraai ogende architectuur vaak van een betere kwaliteit is dan een visueel gedrocht. Dus in tweede instantie is schoonheid wel relevant. Daarnaast is de kunde van de architectuur van informatiesystemen (en wellicht e-business systemen in het bijzonder) nog zo weinig ontwikkeld, dat enige kunst (soms gecombineerd met enig vliegwerk) vaak aan de orde is bij het ontwerpen van architecturen. Ik ga het nu over de kunde in de architectuur hebben – de kunst is wat moeilijker concreet bespreekbaar.

Architectuur heeft te maken met het aanbrengen van structuur – dit ter voorkoming van chaos. De vraag is dan h^oe structuur moet worden aangebracht. Dit is in mijn gebied nog zeker niet evident – mede gezien de eerdere constatering dat bouwkunst en bouwstijl nog jonge begrippen zijn in de wereld van de bedrijfsinformatiesystemen. Ik zal enkele belangrijke principes in het aanbrengen van structuur de revue laten passeren. Deze principes hebben deels hun nut al bewezen in de praktijk van het ontwerpen van e-business architecturen en het realiseren van complexe e-business systemen en verkeren deels nog in het onderzoeksstadium.

Ik geloof sterk in de afstemming tussen functionaliteit en structuur. Zoals in een reclameleus voor mijn elektronische agenda: ‘form is function’ – maar dan anders, want we hebben het immers over onzichtbare architecturen. Ik bepleit een één-op-één relatie tussen applicatiefuncties en componenten van informatiesystemen, in andere woorden modularisering op basis van applicatiefunctionaliteit. Dit moet zowel statisch als dynamisch gebeuren. Statisch betekent, dat een verzameling modules vast bepaald is, dynamisch dat de verzameling modules verandert tijdens het gebruik van een informatiesysteem. Dynamisch gedrag kan worden verkregen via instantiatie van componenten die de levenscyclus van bedrijfsobjecten of -concepten volgen. In de architecturen die we gebruiken in het e-commerce onderwijs zien we hoog-niveau architecturen die deze filosofie volgen [Gro3b]: onderscheid tussen back/front office, afstemming van componenten en kanalen voor specifieke bedrijfsfuncties, en afstemming tussen componenten en specifieke toegangskanalen (aangeduid als ‘multi-channelling’) – zie Figuur 2. Op gedetailleerder niveau en met een meer dynamisch karakter is deze filosofie bijvoorbeeld gebruikt in het WIDE project, waarin functies en componenten voor transactiemangement sterk aan elkaar zijn gerelateerd [Gr98b]. Dit principe is gerelateerd aan de al genoemde afstemming tussen syntax, semantiek en pragmatiek van een architectuur: de vorm, de betekenis en het gebruik van software moeten op elkaar zijn afgestemd.

figuur 2

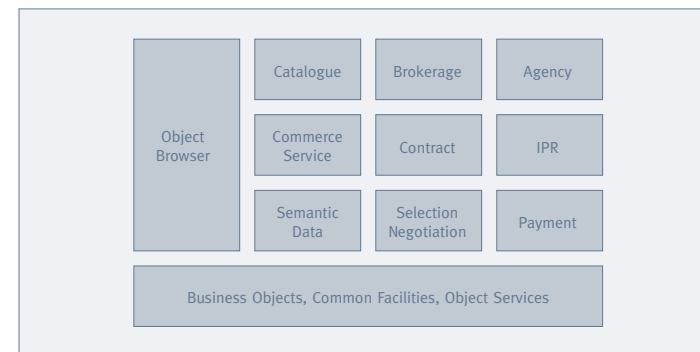
Hoog-niveau
architectuur uit
e-commerce
onderwijs



Een duidelijke gelaagdheid van architecturen brengt ordening aan in verschillende abstractie- of aggregatieniveaus van functionaliteit. In de netwerkwereld kennen we het zevenlaags ISO-OSI model [Sta87], dat communicatie tussen systemen op verschillende abstractieniveaus mogelijk maakt. In databasewereld is er het ANSI-SPARC model [Tsi78], dat drie niveaus onderscheidt voor databasemanipulatie. In de wereld van de architectuur van informatiesystemen zijn algemeen geaccepteerde gelaagde modellen niet voorhanden. In de e-business wereld zien we bijvoorbeeld gelaagdheid in het ECDF referentiemodel [McC97] van de Object Management Group, maar dit raamwerk is nooit tot volledige ontwikkeling gekomen – deze wereld is zo dynamisch dat de task force van de OMG die deze standaard ontwierp inmiddels niet meer bestaat. In het kader van de besproken afstemming van functies en structuur kunnen we de gelaagdheid van informatiesystemen ‘naar boven’ uitbreiden met bedrijfsarchitectuurlagen. Hierdoor wordt architectuur een direct hulpmiddel voor de afstemming tussen bedrijfsfuncties en informatiesysteemstructuren – de zogenaamde business-IT alignment [Wie03].

figuur 3

ECDF architectuur



Referentiearchitecturen spelen een belangrijke rol in het komen tot standaardstructuren voor informatiesystemen. Het gebruik van referentiearchitecturen is in de wereld van de informatiesystemen helaas pas zeer gedeeltelijk ingeburgerd. Er bestaan referentiearchitecturen voor specifieke technologiedomeinen, zoals object-georiënteerde e-commerce systemen [McC97] en workflow management systemen [WfM94, Gr98a], maar deze voldoen niet voor het ontwerpen van architecturen van complete e-business systemen. Het ontwerpen en gebruiken van referentiearchitecturen behoeft dan ook nog het nodige onderzoek.

Zoals al eerder opgemerkt, is in e-business verandering de enige constante. Ontwikkelingen volgen elkaar in een moordend tempo op. De hieruit volgende onbekendheid van de toekomst vereist werken met abstracte functionaliteitsbouwstenen die men kan samenstellen, configureren en parameteriseren om concrete functionaliteit te verkrijgen. In plaats van proberen te anticiperen op functionaliteit van morgen kan men beter abstraheren van de functionaliteit van vandaag en morgen (en hopelijk ook overmorgen) en voldoende flexibiliteit in functionaliteit voorzien.

Een laatste structureringsprincipe dat ik wil noemen is specialisatie. Specialisatie van componenten is een wijze van omgaan met de vele verschijningsvormen waarin bepaalde bedrijfsfunctionaliteit gewenst is in verschillende omgevingen of op verschillende punten in de tijd. In de context van e-business is specialisatie echter mogelijk langs



verscheidene dimensies, bijvoorbeeld de dimensies bedrijfsfunctie, partnerconfiguratie en communicatiekanaal. Om deze dimensies te combineren kan gewerkt worden met meervoudige overerving van eigenschappen van componenten. Toepassing van deze aanpak vergt nog onderzoek.

Bij dit alles dienen we rekening te houden met de omhoogschuivende grens tussen systeemarchitectuur en infrastructuur: ooit was infrastructuur alleen hardware, toen werd het besturingssysteem meegerekend, daarna ook het gegevensbeheer, nu vaak tevens het procesbeheer en de basisapplicatiefunctionaliteit. Op het niveau van infrastructuur zijn structuren merendeels gegeven en moeten we ons beperken tot het parameteriseren van deze structuren. Dit laatste zien we duidelijk binnen de ERP-systemen, waar applicatie-ontwerp voor een niet onbelangrijk deel neerkomt op het instellen van een groot aantal parameters. Ontwerpen van architectuur doen we boven het niveau van infrastructuur – deze observatie hangt samen met het al genoemde onderscheid tussen systeem- en configuratiearchitectuur. Het kiezen van de juiste hoeveelheid structuur is essentieel om goed

met architectuur om te gaan. In de wereld van de B2B e-business zien we verschillende aanpakken. De ebXML aanpak [ebXo3] legt a priori veel structuur op, maar invoering blijkt problematisch. De Web services aanpak [Aloo3] legt minder structuur op, maar de standaardfunctionaliteit is beperkt. Wanneer we te weinig structuur op architectuurniveau aanbrenge, is de ontwikkeling van e-business systemen tot mislukking gedoemd. Wanneer we echter teveel structuren proberen te combineren, krijgt B2B opeens een andere betekenis – deze bespreek ik zo dadelijk.

Het gevaar: Back to Babylon (B2B)

We zien in de e-business een groot aantal standaarden ontstaan die ieder een stuk 'structuur' beschrijven dat we in het ontwerp van architecturen kunnen gebruiken. We kunnen deze standaarden koppelen aan structureringsprincipes zoals ik die besproken heb.

Populair is het stapelen van standaarden, die ieder een 'laag' van de totale e-business architectuur bepalen – een toepassing van het principe van gelaagde architecturen. Een goed voorbeeld is de zogenaamde 'Web services stack' – een stapel van standaarden die bepalen hoe elektronische diensten via het Web uitgevoerd kunnen worden [Aloo3]. Enigszins gesimplificeerd ziet deze stapel er als volgt uit. Onder aan de stapel vinden we HTTP en HTML als basisprotocol en -taal, daarboven SOAP als communicatiestandaard, daarboven WSDL als beschrijvingstaal voor diensten, daarboven weer BPEL als beschrijvingstaal voor processen waarin diensten zijn opgenomen, daarboven weer WS-Coordination en WS-Transaction als uitbreidingen voor coördinatie tussen diensten en transacties waarin diensten participeren. En dan hebben we het slechts over domeinafhankelijke standaarden gehad, die weer de basis kunnen vormen voor domeinafhankelijke standaarden. Al deze standaarden voor communicatie komen overeen met lagen in een e-business systeemarchitectuur, waarvan ze een deel van de structuur bepalen. De standaarden samen bepalen zo een deel van de architectuur van een e-business systeem.

Hierbij kan zich de vraag opdringen of we niet een digitale Toren van Babel aan het bouwen zijn – een torenhoge architectuur die moeilijk echt realiseerbaar is omdat de architectuur problematisch is. Zo is in de praktijk van de netwerkwereld in het verleden gebleken dat de ISO-OSI zeven-lagen architectuur [Sta87] voor diverse toepassingen te veel structuur bevat en dus leidt tot te complexe systemen met te veel interfaces. In de Internetarchitectuur is het aantal lagen dan ook teruggebracht van zeven naar vier. In een multi-lagen e-business architectuur kunnen we hetzelfde tegenkomen: te veel spreiding van functionaliteit, te veel interfaces en dus mogelijk te veel onduidelijkheid en te weinig efficiëntie.

Het groeiende aantal talen en protocollen voor B2B e-business kan daarnaast tot een digitale Babylonische spraakverwarring leiden. Enerzijds kan deze spraakverwarring ontstaan bij het spreken over de talen – er bestaat een ruime sortering talen met verwante doelstellingen. Zo heb ik BPEL genoemd als taal voor het beschrijven van samengestelde e-business processen, maar deze taal is weer gebaseerd op het door IBM ontwikkelde WSFL en het door Microsoft ontwikkelde XLANG en gaat ook door het leven onder de naam BPEL4WS. Binnen andere contexten kennen we weer andere talen met een min of meer vergelijkbaar doel, zoals BPML van het Business Process Management Initiative [BPM03] en het aan de TU/e ontwikkelde XRL [Vero2]. Voor de coördinatie van samenwerkende e-business diensten zijn er onder andere WSCI [Arko3] van BEA, Intalio, SAP en Sun (als voorstel voor een standaard ingediend bij het World Wide Web Consortium), RosettaNet's [Rosoz] PIPs, de CPPs en CPAs in ebXML [ebXoz] van UN/CEFACT en OASIS, en xCBL [xCBo3] van CommerceOne, dat voorheen onder de naam CBL door het leven ging. Anderzijds kan spraakverwarring ontstaan bij het gebruik van de talen: de veelheid van talen en de daaruit volgende noodzakelijke vertalingen op de 'global market place' en het gebrek aan semantiek bij de meeste van deze talen kan in de praktijk snel tot onduidelijkheden leiden. Kortom, bij het aanbrengen van structuur moeten we oppassen niet in de valkuil van het oude Babylon te lopen. In andere woorden, we moeten zorgen dat de afkorting B2B niet uitgelegd gaat worden als 'Back to Babylon'. Als e-business architect hebben we wat dat betreft een nadeel ten opzichte van de architect uit de bouwwereld. Zoals de Amerikaanse architect Wright ooit stelde [Wri53]: "The physician can bury his mistakes, but the architect can only advise his clients to plant vines." In de ICT helpt het planten van druivenranken echter helaas niet – behalve misschien voor het op termijn wegdrinken van verdriet. En voor het begraven van een vergissing is meestal geen tijd in de snelle e-business wereld. Een foute architectuur betekent een negatieve erfenis voor de toekomst – in vaktermen vaak eufemistisch aangeduid als 'a small legacy problem'.

Een toepassingsgebied

Ik wil nu graag een toepassingsgebied voor e-business technologie met u bespreken: dynamic service outsourcing. Dit gebied betreft het dynamisch uitbesteden van diensten in zakelijke e-service markten. Deze keuze heeft een duale achtergrond: enerzijds combineert dit gebied vele aspecten van B2B interactie, anderzijds is dit gebied al enkele jaren mijn persoonlijke 'hobby' in onderzoek. Zoals ik zal uiteenzetten, is architectuur een belangrijk onderwerp in dit gebied. Twee specifieke en illustratieve toepassingen voor dynamic service outsourcing zijn uitgewerkt in het Europese CrossFlow project [Greoo]. Een eerste voorbeeld is een logistiek scenario, waarin een leverancier van mobiele telefoons het logistieke deel van zijn bedrijfsproces dynamisch uitbesteedt aan een logistieke dienstverlener. Concreet betekent dit, dat de uitlevering van telefoonapparatuur uitgevoerd wordt door een partner die tijdens de uitvoering van een klantorderproces gekozen wordt op basis van de karakteristieken van zowel dat klantorderproces als de markt voor logistieke diensten. Een tweede voorbeeld is een verzekeringsscenario, waarin een verzekeringsbedrijf non-core activiteiten uitbesteedt aan dienstverleners. Een zaak als call center handling wordt statisch uitbesteed, maar een zaak als schade-expertise wordt dynamisch uitbesteed. Concreet betekent dit, dat het deelproces voor schade-expertise tijdens de uitvoering van een schadeafhandelingsproces wordt uitbesteed aan een expertisebureau dat het beste past bij het onderhavige schadegeval. In beide specifieke toepassingen vindt de uitbesteding geheel automatisch en elektronisch plaats via een elektronische marktplaats. Dynamic service outsourcing is in principe een combinatie van twee gebieden: workflow management en electronic commerce. Workflow management behelst de geautomatiseerde ondersteuning van operationele bedrijfsprocessen – meestal in grote, complexe organisaties. Workflow management systemen zijn verweven met vele andere bedrijfsinformatiesystemen binnen de grenzen van een organisatie. Hun architectuur dient dan ook afgestemd te zijn op interoperabiliteit binnen een complexe enterprise architecture. Betrouwbaarheid in een complexe, maar stabiele en gecontroleerde



omgeving is een belangrijk aspect. Systemen voor electronic commerce ondersteunen de dynamische interactie tussen organisaties voor het verhandelen van producten – traditioneel fysieke producten, tegenwoordig ook steeds meer diensten. De architectuur van deze systemen is afgestemd op interoperabiliteit binnen 'vluchtige contacten' die dynamisch tot stand komen op elektronische markten. De contacten vinden plaats tussen autonome partijen, waardoor de omgeving niet geheel gecontroleerd is en afspraken expliciet moeten worden vastgelegd. Flexibiliteit in een dynamische, heterogene omgeving is dan ook een belangrijk aspect van electronic commerce systemen. In dynamic service outsourcing moeten deze ogenschijnlijk tegenstrijdige eisen met elkaar gecombineerd worden. Gezien de aard van de samenwerking tussen bedrijven moet deze plaatsvinden in een contractuele en transactionele context die voldoende betrouwbaarheid van bedrijfsverstijgende processen waarborgt. Hier dreigt dan ook chaos door complexiteit en diversiteit in eisen te ontstaan. Om dit te voorkomen, dienen we structuur aan te brengen door een duidelijke architectuur als uitgangspunt te hanteren. De structureringsmiddelen



die we eerder hebben gezien, zijn hierbij van toepassing. Ik heb één-op-één afbeelding van functionaliteit op informatiesysteem-modules hiervoor als eerste structureringsprincipe genoemd. In een toepassing als dynamic service outsourcing is dit principe van groot belang, omdat hier twee zaken een prominente rol spelen. Op de eerste plaats is er behoefte aan complexe functionaliteit voor het aanbieden, aankopen, uitvoeren en bewaken van diensten. Hierin speelt onder andere functionaliteit voor electronic market places [Hof00], electronic contracting [Ano3a, Ano3b], cross-organizational workflow management [Ver02, Huo03], web services [Gro3c] en transaction management [Gre01, Von03]. Op de tweede plaats is er behoefte aan grote flexibiliteit in de configuratie van systemen, die afhankelijk is van de samenwerkingsvorm en technische infrastructuur van zakenpartners. Deze twee zaken zijn alleen goed met elkaar te verenigen indien een duidelijke afbeelding van functionaliteit op bedrijfsniveau en softwaremodules op technisch niveau bestaat.

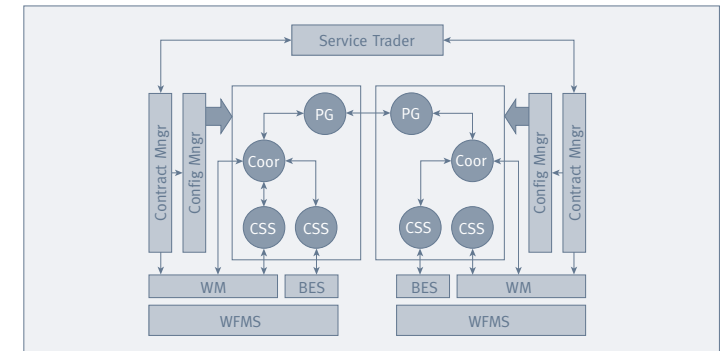
Een gelaagd ontwerp van architecturen voor dynamic service outsourcing is vereist om een goede scheiding tussen de verschillende niveaus van proces- en gegevensverwerking te bewerkstelligen. Voor dit doel hebben we in samenwerking met IBM Research een drie-lagen aanpak [Gro3a] gedefinieerd die geïnspireerd is op het eerder genoemde ANSI-SPARC model [Tsi78]. Deze drie-lagen aanpak maakt onderscheid tussen een conceptueel niveau waarop ontwerp van processen plaatsvindt, een extern niveau waarop interactie tussen organisaties plaatsvindt en een intern niveau waarop interne processen worden uitgevoerd. In dit werk hebben we een voorzichtig begin gemaakt met het definiëren van een referentiearchitectuur voor dynamic service outsourcing.

Abstracte componenten spelen een belangrijke rol in dynamic service outsourcing om structuren te bieden die parameterisering van functionaliteit mogelijk maken, om zodoende in te spelen op dynamisch gevormde eisen met betrekking tot de functionaliteit van systemen. Dit principe is op een aantal plekken in ons werk zichtbaar. In het CrossFlow project [Gre00, Hof01] hebben we parameteriseerbare Cooperative Service Support modules in de architectuur voorzien, die het mogelijk maken zowel toekomstige functionaliteit in te bouwen als deze functionaliteit van het gewenste gedrag te voorzien. In het nieuwe XTraConServe NWO-project [Pap03] voorzien we abstracte transactiebouwstenen die geparameteriseerd transactiegedrag in een service-georiënteerde context moeten ondersteunen.

Het gebruik van specialisatie van functionaliteitsbouwstenen voor dynamic service outsourcing is nog onderwerp van onderzoek, maar lijkt interessante perspectieven te bieden.

figuur 4

CrossFlow
architectuur



Tot slot

In deze rede heb ik gepoogd het belang van onzichtbare architecturen in de e-business voor u wat zichtbaarder te maken. Architecturen van e-business systemen mogen voor de 'buitenwereld' onzichtbaar zijn, het effect van deze architecturen zal duidelijk waarneembaar zijn, bijvoorbeeld in het succes van bedrijven in e-business markten. In deze rede heb ik slechts een enkel toepassingsgebied behandeld – vanzelfsprekend zijn er vele meer. In de e-commerce is weliswaar een zeebel gesprongen, maar dieper liggende trends en structuren in de e-business blijven achter [Pieo2]. Architecturen van e-business systemen zijn onderdeel van – of horen onderdeel te zijn van – deze structuren.

De architectuur van systemen voor e-business dient te zijn gefundeerd op academische inzichten, maar dient tevens aan te sluiten bij inzichten in de praktijkwereld van de e-business. E-business is een dynamisch gebied, waarin onderzoekinstellingen en ontwikkelingsafdelingen van het bedrijfsleven gezamenlijk de toon zetten. Om deze reden is een samenwerking tussen universiteiten, onderzoekinstellingen, en bedrijven een conditio sine qua non voor het doen van relevant onderzoek in dit gebied. Deze samenwerking moet kruisbestuiving tussen specifieke invalshoeken mogelijk maken, zodat voorkomen wordt dat enerzijds onderzoek van universiteiten plaatsvindt in ivoren torens en anderzijds ontwikkelingen in bedrijven leven bij de waan van de dag. Daarbij moeten we de samenwerking zowel op nationaal als internationaal niveau zoeken. Het nationaal niveau is van belang om met 'nabije' partners aan de positie van Nederland als kennisland op dit gebied te werken. Het internationaal niveau is van belang omdat in de wereld van de e-business de belangrijke ontwikkelingen op dit niveau plaatsvinden – veel initiatieven hebben een Amerikaanse oorsprong. Bij dit alles is er een noodzaak tot financiële ondersteuning van strategisch onderzoek op dit gebied voor instandhouding van kennis voor het bedrijfsleven in Nederland. Deze ondersteuning moet een zodanig karakter hebben dat deze niet leidt tot een bureaucratisch circus voor de onderzoeker – iets wat tegenwoordig helaas te vaak voorkomt. Dan wil ik nog even stilstaan bij het onderwerp architectuur in



het academisch onderwijs. Zoals ik in het voorgaande geprobeerd heb duidelijk te maken, is e-business reddeloos verloren zonder goede structuur en is architectuur de basis van een goede structuur. Architectuur van informatiesystemen is dan ook een centraal begrip voor studenten die zich specialiseren in het gebied BedrijfsInformatie-Systemen of BedrijfsInformatieTechnologie (de precieze term is afhankelijk van de toevallige universiteit). Om dit te bewerkstelligen is er in het onderwijs vanuit mijn leerstoel dan ook ruimschoots aandacht voor het architectuurbegrip – er staan drie vakken op het programma waarin architectuur een centrale rol vervult. De ervaring leert dat dit echter een moeilijk onderwerp is voor zowel docenten als studenten. U mag dan ook nog de nodige ontwikkelingen op dit terrein verwachten.

Dankwoord



Dan resten mij nog enkele woorden van dank. Ik bedank de Faculteit Technologie Management voor haar vertrouwen in mij blijkens mijn aanstelling aan deze universiteit. Ik bedank de nieuwe collega's binnen de capaciteitsgroep Informatie en Technologie voor de goede context binnen welke ik mijn leerstoel vorm kan geven. Ik bedank de oude collega's aan de Universiteit Twente en in de vanuit de UT uitgevoerde projecten voor de samenwerking en de vele leerzame ervaringen – voor het merendeel goede. Dan gaat mijn dank naar vrienden en kennissen, tussen welke Ria Marinussen – beter bekend als Ria2 – aparte aandacht verdient. Ik bedank mijn ouders, zonder wie ik hier niet gestaan had. Last but certainly not least bedank ik 'mijn' Ria voor het engelengeduld in het jarenlang luisteren naar vakgebrabbel, de vele goede adviezen van een 'well-informed outsider' en de steun in de moeilijker tijden. Tenslotte bied ik het publiek mijn excuses aan, zeker aan de niet-vakgenoten, voor de vele Engelstalige vaktermen die u de afgelopen drie kwartier hebt moeten aanhoren – een zekere professionele deformatie in het taalgebruik lijkt in mijn vakgebied helaas nauwelijks te vermijden. Ik dank u voor uw aandacht.

Literatuur

- [Aloo3] G. Alonso, F. Casati, H. Kuno, V. Machiraju; Web Services: Concepts, Architectures and Applications; Springer, 2003.
- [Ano3a] S. Angelov, P. Grefen; The 4W Framework for B2B e-Contracting; International Journal of Networking and Virtual Organizations; Vol. 2, No. 1; Inderscience Publishers, 2003; pp. 78-97.
- [Ano3b] S. Angelov, P. Grefen; An Analysis of the B2B E-Contracting Domain: Paradigms and Required Technology; Beta Working Paper 102; Technische Universiteit Eindhoven, 2003.
- [Arko3] A. Arkin e.a. (red.); Web Service Choreography Interface (WSCI) 1.0; 2003; beschikbaar via <http://www.w3.org/TR/2002/NOTE-wsci-20020808/>.
- [BPMo3] Business Process Management Initiative Web Site; <http://www.bpmi.org>.
- [ebXo3] ebXML Web Site; <http://www.ebxml.org>.
- [Gee84] G. Geerts, H. Heestermans; Van Dale Groot Woordenboek der Nederlandse Taal – Elfde, Herziene Druk; Van Dale Lexicografie, 1984.
- [Gr98a] P. Grefen, R. Remmerts de Vries; A Reference Architecture for Workflow Management Systems; Journal of Data & Knowledge Engineering, Vol. 27, No. 1; North Holland - Elsevier, 1998; pp. 31-57.
- [Gr98b] P. Grefen, R. Wieringa; Subsystem Design Guidelines for Extensible General-Purpose Software; Proceedings 3rd International Software Architecture Workshop; Orlando, USA, 1998, pp. 49-52.
- [Gre00] P. Grefen, K. Aberer, Y. Hoffner, H. Ludwig; CrossFlow: Cross-Organizational Workflow Management in Dynamic Virtual Enterprises; International Journal of Computer Systems Science & Engineering, Vol. 15, No. 5; CRL Publishing, 2000; pp. 277-290.
- [Gre01] P. Grefen, J. Vonk, P. Apers; Global Transaction Support for Workflow Management Systems: from Formal Specification to Practical Implementation; VLDB Journal; Vol. 10, No. 4; Springer, 2001; pp. 316-333.
- [Gre02] P. Grefen; Sheets College 1 Architectuur van Complexe InformatieSystemen; Universiteit Twente, 2002.
- [Gro3a] P. Grefen, H. Ludwig, S. Angelov; A Three-Level Framework for Process and Data Management of Complex E-Services; International Journal of Cooperative Information Systems, Vol. 12, No. 4; World Scientific, 2003; pp. 487-532.
- [Gro3b] P. Grefen; Sheets College 5 Electronic Commerce; Universiteit Twente, 2003.
- [Gro3c] P. Grefen, H. Ludwig, A. Dan, S. Angelov; Web Services Support for Dynamic Business Process Outsourcing; IBM Research Report RC22728; IBM Research Division, 2003.
- [Hof00] Y. Hoffner, C. Facciorusso, S. Field, A. Schade; Distribution Issues in the Design and Implementation of a Virtual Market Place; Computer Networks, Vol. 32, No. 6; Elsevier, 2000; pp. 717-730.
- [Hof01] Y. Hoffner, S. Field, P. Grefen, H. Ludwig; Contract Driven Creation and Operation of Virtual Enterprises; Computer Networks; Vol. 37, No. 2; Elsevier, 2001; pp. 111-136.
- [Huo3] J. Hu, P. Grefen; Conceptual Framework and Architecture for Service Mediating Workflow Management; Information and Software Technology Journal; Vol. 45, No. 13; Elsevier, 2003; pp. 929-939.



- [Joh64] P. Johnson; New York Times; 27 December 1964.
- [McC97] Stephen McConnell (red.); The OMG/CommerceNet Joint Electronic Commerce Whitepaper; OMG/CommerceNet, 1997.
- [Pap03] M. Papazoglou, P. Grefen; XTraConServe NWO Project Proposal; Universiteit van Tilburg en Technische Universiteit Eindhoven, 2003.
- [Pie02] R. Pieper, V. Kouwenhoven, S. Hamminga; Beyond the Hype: E-Business Strategy in Leading European Companies; Van Haren Publishing, 2002.
- [Ros03] RosettaNet Web Site; www.rosettanet.org; 2003.
- [Sha96] M. Shaw, D. Garlan; Software Architecture; Prentice Hall, 1996.
- [Sta87] W. Stallings; Handbook of Computer Communications Standards, Vol. 1; MacMillan, 1987.
- [Tsi78] D. Tsichritzis, A. Klug; The ANSI/X3/SPARC DBMS Framework; AFIPS Press, 1978.
- [Tru90] J. Truijens, A. Oosterhaven, R. Maes, H. Jägers, F. van Iersel; Informatie-Infrastructuur: een Instrument voor het Management; Kluwer Bedrijfswetenschappen, 1990.
- [Ver02] H.M.W. Verbeek, A. Hirschall, W.M.P. van der Aalst; XRL/Flower: Supporting Interorganizational Workflows using XRL/Petri-net Technology; Proceedings CAiSE 2002 International Workshop on Web Services, E-Business, and the Semantic Web; Toronto, Canada, 2002; pp. 93-108.
- [Von03] J. Vonk, P. Grefen; Cross-Organizational Transaction Support for E-Services in Virtual Enterprises; Journal of Distributed and Parallel Databases; Vol. 14, No. 2; Kluwer Academic Publishers, 2003; pp. 137-172.
- [WfM94] Glossary - A Workflow Management Coalition Specification; Workflow Management Coalition, 1994.
- [Wie03] R. Wieringa, H. Blanken, M. Fokkinga, P. Grefen; Aligning Application Architecture to the Business Context; Proceedings 15th International Conference on Advanced Information Systems Engineering; Klagenfurt/Velden, Austria, 2003; pp. 209-225.
- [Wri53] F. L. Wright; New York Times Magazine, 4 Oktober 1953.
- [xCB03] xCBL Web Site; www.xcbl.org; 2003.

Curriculum Vitae



Aan de Technische Universiteit Eindhoven is per 1 maart 2003 prof.dr.ir. P.W.P.J. Grefen benoemd tot voltijds hoogleraar ICT Architectures aan de faculteit Technologie Management.

Paul Grefen (1963) studeerde Informatica aan de Universiteit Twente (UT). In 1987 startte hij zijn onderzoekscarière als medewerker onderzoek aan de UT in het PRISMA project, waarin onder leiding van Philips Research is gewerkt aan de ontwikkeling van parallelle database-technologie. In 1992 promoveerde hij op het onderwerp 'Integrity Control in Parallel Database Systems'. Van 1992 tot en met 1997 was hij universitair docent, van 1998 tot begin 2003 universitair hoofd-docent bij de faculteit Informatica van de UT. In 1994 verbleef hij enkele maanden als gastonderzoeker aan Stanford University in de VS. Van 1995 tot 2000 leidde hij de UT-inbreng in de Europese WIDE en CrossFlow projecten. Paul Grefen geeft in zijn functie van hoogleraar ICT Architectures leiding aan de gelijknamige groep, die zich toelegt op structuren van informatiesystemen voor e-business applicaties.

Colofon

Productie:
Communicatie Service Centrum TU/e

Fotografie cover:
Rob Stork, Eindhoven

Ontwerp:
Plaza ontwerpers,
Eindhoven

Druk:
Drukkerij Lecturis,
Eindhoven

ISBN: 90-386-1173-0

Digitale versie:
www.tue.nl/bib/