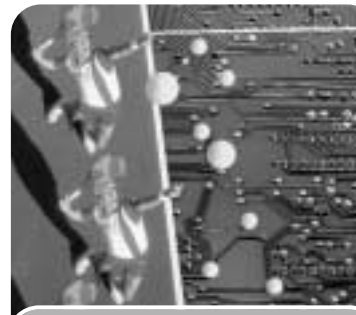


De afgelopen periode stond het internet in het teken van standaardisatie. Echter, tegelijkertijd proberen verschillende partijen zich toch op allerlei manieren ten opzichte van elkaar te onderscheiden. Zo voegen leveranciers van webtechnologie allerlei features toe, om zodoende onderscheidend te zijn. Bouwers van websites proberen vervolgens deze technologieën zo creatief mogelijk in te zetten, om een attractieve en een herkenbare gebruikersinterface te krijgen. Dit betekent dat een bezoeker van websites geen websites zal vinden, waarvan de navigatie en look-and-feel overeenkomen. Ondanks al die afwijkingen is een gemiddelde bezoeker van een willekeurige website al vrij snel in staat om er de weg te vinden.



thema

Transactiestandaarden schieten tekort

Business Transaction Protocol (BTP) veelbelovend

De doelstelling van webservices is dat niet eindgebruikers, maar applicaties, informatie en diensten over het web kunnen aanspreken. Het gaat dus om applicatiekoppelingen over het internet. Mensen hebben de intelligentie om in een niet geformaliseerde omgeving hun weg te vinden. Een voorwaarde om applicaties met elkaar te laten communiceren is dat wel alle aspecten ten aanzien van de communicatie, tot op het laagste detailniveau geformaliseerd zijn. De huidige standaarden, die onder de paraplu webservices geschoven kunnen worden (SOAP, WSDL, UDDI), vormen hiervoor een belangrijk uitgangspunt. Deze set van standaarden is bij lange na niet volledig genoeg om applicaties via het internet op een veilige en betrouwbare manier met elkaar te laten communiceren. Zo zijn aspecten als vertrouwelijkheid van gegevens en garanties over de herkomst van berichten, nog onvoldoende uitgekristalliseerd.

Een ander aspect dat de aandacht verdient, is dat er bij betrokken applicaties geen verwarring kan optreden over de wijze waarop er met foutsituaties wordt omgegaan. Gelukkig is er binnen de IT reeds lang een standaard bekend die hier een goede bijdrage aan kan leveren. De ACID-standaard is een klassieker, die zich op dit vlak al duidelijk bewezen heeft. Een van de grote diensten van de ACID-standaard is dat ontwikkelaars

geen complexe foutafhandeling procedures hoeven te ontwerpen en te implementeren om de gevolgen van opgetreden fouten op te sporen en eventueel ongedaan te maken. Het is echter de vraag of de concepten waarop de ACID-standaard gebaseerd is, in een webservices omgeving nog steeds bruikbaar zijn. Om deze vraag te kunnen beantwoorden, zullen we eerst dieper op die concepten ingaan.

DE ACID-STANDAARD ACID is een acroniem voor een verzameling van afspraken die is vastgelegd onder ISO/IEC 10026-1. Het acroniem staat voor Atomicity, Consistency, Isolation en Durability. De ACID-standaard biedt de mogelijkheid om programma-code, waarin meerdere acties worden uitgevoerd, als een eenheid van werk te markeren (een zogenaamde transactie). Een dergelijke transactie wordt vervolgens als een ondeelbaar geheel behandeld (*atomicity*). Wanneer een transactie wordt aangeroepen zijn er maar twee mogelijke uitkomsten, namelijk de transactie is succesvol uitgevoerd en goedgekeurd (*committed*), of er is wat misgegaan (*aborted*). Naast de ondeelbaarheid van deze samengestelde verzameling aan acties, moet tevens aan een aantal garanties worden voldaan. Wanneer de eenheid van werk succesvol is goedgekeurd, kan men er zeker van zijn dat alle activiteiten zijn uitgevoerd en alle bijbehorende wijzingen onuitwisbaar zijn doorge-

voerd. Dit aspect wordt met het begrip *durability* aangegeven. In het geval dat er een probleem optreedt voordat er goedkeuring heeft kunnen plaatsvinden, zullen alle voorafgaande wijzigingen ongedaan worden gemaakt, ofwel het begrip *consistency*. Het consistency begrip voorkomt dat een database als gevolg van tussentijdse fouten vervuild raakt. Om de voorgaande

Webservices zijn bedoeld voor applicaties, niet voor mensen

aspecten te kunnen garanderen, is het noodzakelijk dat er tussentijds geen andere processen de gegevens van niet-goedgekeurde transacties kunnen wijzigingen (het begrip *isolation*). De meest toegepaste methode om isolation te kunnen bieden is locking.

Moderne databases zijn individueel in staat om aan de ACID-standaard te voldoen. Ook in een configuratie waarbij de database vanuit een applicatieserver wordt aangesproken, kan men er vanuit gaan dat het ACID-transactie principe wordt ondersteund. In het geval van webservices hebben we echter te maken met een gedistribueerde omgeving waarbij services zich op verschillende locaties bevinden. In een dergelijke gedistribueerde omgeving is het niet meer voldoende om af te spreken dat aan de ACID-standaard wordt voldaan. In dat geval zijn er aanvullende afspraken noodzakelijk. Eén van de meest gebruikte standaarden voor deze probleemstelling is de X/Open DTP-standaard (met als meest bekende onderdeel: de XA-interface). De standaard is een implementatie van het two phase commit algoritme.

TWO PHASE COMMIT Het two phase commit (2PC) protocol heeft als doel er voor te zorgen dat een transactie kan worden uitgevoerd waaraan meerdere gescheiden databronnen (resource managers) deelnemen. In het protocol is een transactievoordinator opgenomen, die in de plaats van de resource manager zal beslissen of een transactie als geheel goed- of afgekeurd gaat worden. Voordat de transactievoordinator deze beslissing kan nemen zullen de betrokken resource managers eerst hun lokale transacties uitvoeren, de zogenaamde prepare-fase. Wanneer de prepare-fase zonder problemen verloopt, zal een resource niet direct een commit doorvoeren, maar zal dan eerst een prepared-bericht naar de transactievoordinator versturen. Wanneer de transactievoordinator van alle betrokken resource managers het positieve prepared-bericht ontvangen heeft, zal aan alle

resource managers een confirm-bericht worden teruggestuurd om de transactie definitief te maken. Echter, wanneer slechts één van de resource managers via het cancel-bericht heeft aangegeven dat er problemen zijn opgetreden, zal de transactievoordinator aan de overige resources een cancel-bericht versturen. Deze resource managers zullen vervolgens hun lokale transactie terugdraaien, waardoor er indirect voor wordt gezorgd dat de gehele transactie ongedaan wordt gemaakt.

GESCHIKTHEID VOOR WEBSERVICES

Een transactie die met behulp van het 2PC protocol gecoördineerd wordt, vereist behoorlijk wat (netwerk) communicatie tussen de verschillende participanten. Om die reden is het een relatief traag protocol, dat tevens gevoelig is voor netwerkproblemen. In een omgeving waarbij de resource managers uit geïntegreerde databases bestaan, bestaat al vrij snel het risico dat er voor langere tijd locks blijven staan, waardoor andere transacties geblokkeerd kunnen worden. Als gevolg van deze eigenschappen wordt het 2PC protocol in de praktijk slechts selectief en met mate toegepast. Heeft een dergelijk (netwerkgevoelig) protocol dan toegevoegde waarde in een omgeving waar gebruik wordt gemaakt van webservices? Webservices bieden namelijk een mechanisme om functionaliteit over het internet te kunnen aanroepen.

Wanneer we de marketingverhalen ten aanzien van webservices mogen geloven, dan is het een technologie, die er in hoofdzaak op gericht is om applicaties van verschillende bedrijven aan elkaar te koppelen. Een veel aannemelijker scenario is, dat webservices met name gebruikt gaan worden voor de realisatie van service georiënteerde applicatie architecturen, waarbij softwarecomponenten van elkaars services gebruik kunnen maken. Men doet er uiteraard verstandig aan dergelijke softwarecomponenten zo veel mogelijk te centraliseren op een centrale executie omgeving. Wanneer de softwarecomponenten toch over meerdere locaties worden verspreid, heeft men te maken met het interne netwerk. In het algemeen heeft men een goed inzicht in en controle over het netwerk en de softwarecomponenten die via het netwerk benaderd kunnen worden. In een dergelijke omgeving kunnen webservices met 2PC een beheersbare combinatie vormen.

Wanneer de transacties en de bijbehorende communicatie over het minder betrouwbare externe internet moeten gaan plaatsvinden, zijn de klassieke standaarden ACID en 2PC een minder geschikte keuze. Dit hoeft overigens niet alleen aan het internet zelf te liggen. Ook over de beschikbaarheid van de webservices zelf zijn vaak geen harde garanties te verkrijgen. Bekeken vanuit

het perspectief van een aanbieder van een webservice, zal de aanbieder willen voorkomen dat de aanvrager van een service voor andere aanvragers gaat blokkeren. Dit is een reëel risico wanneer een combinatie van 2PC en de ACID-standaard wordt toegepast. Dezelfde conclusies trokken ook een aantal bedrijven die het initiatief namen om een alternatief te bedenken: het Business Transaction Protocol.

RELATIE BTP-ACID In het Business Transaction Protocol (BTP) heeft men een groot aantal principes uit de ACID-standaard en 2PC overgenomen en verder aangepast en uitgebreid om het geschikt te maken ten behoeve van Webservices. Met name ten aanzien van de ACID-standaard zijn de meeste principes ongewijzigd overgenomen, maar zijn de regels op sommige punten wat versoepeld.

Eén van de belangrijkste begrippen binnen het BTP-protocol is het begrip ATOM. Een ATOM komt sterk overeen met een ACID-transactie, echter ten aanzien van isolatie worden geen eisen gesteld. Dit betekent automatisch dat er geen locking noodzakelijk is, waardoor problemen als gevolg van langdurige locks voorkomen kunnen worden. In een dergelijke omgeving kan het gebruik van compenserende transacties een alternatief zijn voor de impliciete rollback die standaard aan ACID-transacties gekoppeld is. Een compenserende transactie is een op zich staande transactie, waarmee gevolgen van een eerder uitgevoerde transactie ongedaan kunnen worden gemaakt. Een complicerende factor hierbij is dat er geen controle is over wat er zich in de tussenliggende periode allemaal heeft afgespeeld. Bij het gebruik van compenserende transacties zal men zelf wel de nodige maatregelen moeten nemen om er voor te zorgen deze transactie gegarandeerd tot een consistente eindsituatie leidt. Overigens is het niet verboden om een ATOM wel aan alle ACID principes te laten voldoen, waardoor een makkelijke integratie mogelijk is met omgevingen die wel volgens de ACID-standaard zijn ingericht. Elke partij die een webservice aanbiedt die aan het ATOM-begrip voldoet, is echter vrij om te bepalen hoe hier intern invulling aan wordt gegeven. Het gaat er uiteindelijk om dat aan de vastgestelde communicatieafspraken wordt voldaan. Het ATOM-begrip van BTP is dus nauw verwant aan het Atomicity-begrip van de ACID-standaard.

ATOMS GROEPEREN BINNEN ATOMS ATOMS vormen de bouwstenen van de overige constructies die binnen de BTP-standaard worden aangeboden. Zo bestaat de mogelijkheid om ATOMS te groeperen in een overkoepelend ATOM.

Wanneer bijvoorbeeld meerdere resources in een ondeelbare transactie moeten deelnemen, wordt dit gerealiseerd door meerdere ATOMS in een groeperende ATOM onder te brengen. Om deze verzameling van ATOMS tot een geheel te smeden, wordt onder water gebruik gemaakt van het 2PC-protocol. In dit scenario zal een coördinator aanwezig moeten zijn, die binnen de BTP-standaard de ATOM coördinator wordt genoemd. Naast de coördinatie van een ATOM zal deze ook het identificatienummer bepalen waarmee de ATOM geïdentificeerd wordt.

Alle ATOMS die onder een ATOM coördinator vallen, worden volgens het alles-of-niets principe behandeld. Er zijn dus minimale verschillen met ACID-transacties die door een 2PC coördinator worden beheerd. Dit betekent dat bij gedistribueerde ATOMS zich vergelijkbare problemen kunnen voordoen als in een omgeving waarin de XA-standaard wordt toegepast. Binnen de BTP-standaard is daarom een tweede manier van ATOM-groepering opgenomen, het zogenaamde COHESION-begrip. De wijze waarop bedrijven in de praktijk vaak transacties tussen elkaar afwickelen heeft hierbij als voorbeeld gediend.

ATOMS GROEPEREN BINNEN COHESIONS

Een COHESION bestaat uit een groepering van ATOMS. Binnen COHESIONS wordt wat genuanceerder omgegaan met het alles-of-niets principe. In geval van COHESIONS bestaat de mogelijkheid dat wanneer niet

Binnen enkele maanden valt te beoordelen of de BTP-standaard breed geaccepteerd zal worden

alle ATOMS succesvol door de prepare-fase zijn heen gekomen, er toch een deel van de ATOMS wordt doorgevoerd. Dit betreft veelal die ATOMS die wel goed zijn gegaan. Dit betekent dat het proces dat een COHESION heeft aangevraagd (de initiator) een antwoord ontvangt waarin zowel de gelukte als mislukte ATOMS zijn opgenomen. Vervolgens zal de initiator zelf moeten bepalen welke vervolgacties na een dergelijk resultaat moeten worden uitgevoerd. De service die COHESIONS coördineert, is een COHESION composer. In tegenstelling tot een ATOM coördinator, kan binnen een Cohesion composer tevens business logica worden opgenomen die rekening houdt met de uitkomsten van de verschillende ATOMS, die direct vanuit een cohesion composer worden aangestuurd. Een groot voordeel van COHESIONS is dat ontwikkelaars kunnen vertrouwen op de

resultaten die door een cohesion composer worden teruggecommuniceerd. Voor de ATOMS, waaruit een COHESION is samengesteld, geldt namelijk nog wel steeds het alles-of-niets principe. Er kan dus geen onduidelijkheid ontstaan over de status waarin ATOMS zich bevinden.

Er bestaat enige overeenkomst tussen COHESIONS en de "Optimistic Concurrency Control"-techniek, die in database omgevingen veelvuldig wordt toegepast. Er wordt namelijk vanuit gegaan dat in de meerderheid van de gevallen er geen afwijkingen van het hoofdscenario te verwachten zijn. Alles van tevoren checken en reserveren levert dan onevenredig veel overhead op, die in de meerderheid van de gevallen overbodig blijkt te zijn. In die situaties kan het zinvol zijn er primair vanuit te gaan dat alles goed gaat, en dat ten behoeve van foutsituaties een aantal scenario's wordt vastgelegd om ongewenste uitkomsten te herstellen.

BUSINESS CASE

De hierboven beschreven strategie wordt veelvuldig in echte business transacties toegepast. Zo is het bijvoorbeeld gebruikelijk dat luchtvaartmaatschappijen meer stoelen verkopen dan er daadwerkelijk in een vliegtuig zitten. Het is ook niet ongebruikelijk dat er producten worden verkocht, waarbij achteraf de datum van levering moet worden bijgesteld. Eventuele problemen die hier uit voortvloeien tracht men achteraf te herstellen. Om dergelijke beslissingen goed te kunnen nemen, is wel diepgaande kennis over de bedrijfsprocessen vereist. In Figuur 1 is een voorbeeld van een bedrijfstransactie met behulp van de BTP-standaard uitgewerkt.

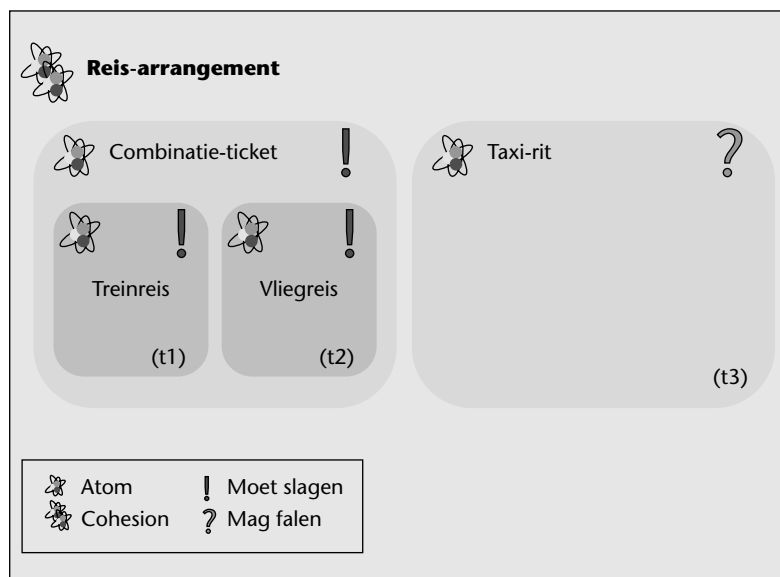
In figuur 1 wordt een samengestelde bedrijfstransactie voorgesteld. De applicatie dient een reis van punt A

naar punt B te regelen, waarbij gebruik gemaakt wordt van diverse typen van vervoer. Cruciaal zijn een treinreis naar het vliegveld (t1) en de vliegreis (t2) zelf. Na aankomst op de luchthaven dient bij voorkeur nog een laatste stuk afgelegd te worden per taxi (t3). Van tevoren zou die taxi ook geregeld moeten zijn, maar als dat niet zou slagen, bijvoorbeeld omdat het boekingsbureau niet bereikbaar is, dan is er niets aan de hand, want dan zal ter plaatse een andere vorm van vervoer worden gezocht. Op een vliegveld hoeft dat geen enkel probleem te zijn.

Als alle genoemde operaties zouden worden uitgevoerd als een enkelvoudige, klassieke transactie, dan zullen alle resources die zijn gealloceerd ten behoeve van t1 ontoegankelijk blijven voor andere applicaties totdat uiteindelijk de volledige transactie afgehandeld is. Dit terwijl de activiteiten t2 en t3 misschien wel een minder sterke afhankelijkheid met elkaar hebben.

Laten we aannemen dat in voorgenoemd voorbeeld de activiteiten t1 en t2 slagen en dat inderdaad activiteit t3 mislukt, aangezien het boekingsbureau niet bereikbaar is. In dat geval kan het boeken van het arrangement toch als gedeeltelijk geslaagd worden gezien. Eventueel als vervolg kan alsnog naar een alternatief worden gezocht, door bijvoorbeeld het laatste stuk van de reis per Openbaar Vervoer te gaan regelen.

APPLICATIEKOPPELING Gezien de bovenstaande voorbeelden, geeft het COHESION-begrip een formele invulling aan een manier van handelen, die bij alledaagse bedrijfstransacties al veelvuldig wordt toegepast. In een omgeving waarbij applicaties via webservices aan elkaar gekoppeld gaan worden, is het prettig dat deze wijze van werken in een bedrijfstakoverstijgende standaard geformaliseerd is. De standaard zal echter alleen tot haar recht komen in processen, die volgens een degelijk en afgewogen ontwerp gerealiseerd zijn. BTP zal daarom met name toepasbaar zijn, als hulpmiddel



FIGUUR 1: Boeken van een reis-arrangement

Toelichting bij figuur 1

Een aantal (on)afhankelijke kortdurende transacties (atoms), kunnen als een samenhangend geheel worden opgevat (cohesion). Binnen een cohesion bestaat de mogelijkheid om deeltransacties op te nemen, die tijdens de uitvoering van de cohesion genegeerd kunnen worden, wanneer deze niet direct tot een bevredigend resultaat leiden. Hiermee wordt de duur dat bepaalde resources voor een activiteit ontoegankelijk blijven voor anderen tot een minimum beperkt. Tevens kan binnen een cohesion worden gereageerd op de uitkomst van de verschillende activiteiten. Een groot voordeel van een cohesion is dat er geen twijfel kan ontstaan over het slagen van de atoms, waaruit de cohesion is samengesteld.

bij de realisatie van business proces management oplossingen, waar applicatiekoppeling met de buitenwereld een onderdeel kan zijn.

BTP EN WEBSERVICES In principe is het BTP een implementatie-onafhankelijk protocol. Het zal niemand verwonderen dat er in eerste instantie een SOAP implementatie voor dit protocol ontwikkeld is. De informatie, die ten behoeve van de BTP-standaard aan berichten gekoppeld dient te worden, wordt in de header van een SOAP-enveloppe onderbracht. Verder zullen de partijen, die via de BTP-standaard communiceren, de beschikking moeten hebben over services die ATOMS en COHESIONS kunnen coördineren. Het is de verwachting dat leveranciers die bij de definitie van de BTP-standaard betrokken zijn, in ieder geval de noodzakelijke producten hiervoor zullen leveren. Dit zijn de volgende leveranciers: BEA, HP, Entrust, IONA, Choreology, Oracle, Talking Blocks, SeeBeyond. HP komt de eerste helft van dit jaar met het product HP Web Services Transactions (HP-WST) op de markt. Dit is de eerste leverancier in de markt, die de BTP-standaard in een commercieel product heeft geïmplementeerd. HP heeft tevens de specificatie voor een aanvullende API met de naam XML Transactioning API for Java (JAXTX), als voorstel aan het Java Community Process aangeboden (JSR-156: zie www.jcp.org). Op deze wijze bestaat de kans dat BTP ook in de Java-omgeving door middel van een gestandaardiseerd protocol zal worden geïmplementeerd, en daarmee dus een nog breder draagvlak kan creëren.

DE TOEKOMST VAN BTP BTP kan boven op de huidige verzameling webservices standaarden (SOAP, WSDL en UDDI), een duidelijk toegevoegde waarde bieden. De standaard kan helpen om de kwaliteit van webservices naar een hoger niveau te brengen. Het succes van een standaard is echter in grote mate afhankelijk van de ondersteuning in de markt. De BTP-standaard wordt ontwikkeld onder de paraplu van OASIS. Dit is tevens de organisatie die de ontwikkeling van de ebXML-standaard onder haar hoede heeft genomen. Hiermee wordt in ieder geval verzekerd dat de BTP-standaard goed aansluit op de standaarden, die specifiek gericht zijn op de business-to-business problematiek.

KANTTEKENINGEN Er zijn twee kanttekeningen te plaatsen ten aanzien van de status van de BTP-standaard. Ten eerste is de BTP-standaard op dit moment nog niet volledig uitgewerkt en als standaard nog niet goedgekeurd. Er zijn echter weinig signalen te bespeuren die een goedkeuring in de weg zullen staan. Alle alternatieve initiatieven die op dit gebied zijn opgestart,

zijn namelijk met het BTP-initiatief samengevloeid. Ten aanzien van de acceptatie van de standaard door leidende leveranciers in de markt is het beeld nog erg mistig. Naast de leveranciers die betrokken zijn geweest bij de specificatie van de BTP-standaard is er nog weinig initiatief te bespeuren bij andere belangrijke spelers in de Webservices markt (SUN, IBM, Microsoft). De standaard heeft in een webservices omgeving pas echt toegevoegde waarde wanneer er een brede marktacceptatie

ATOMS vormen de bouwstenen van de overige constructies binnen de BTP-standaard

aanwezig is. Applicaties kunnen namelijk alleen met elkaar communiceren, wanneer alle details tot op het laagste niveau overeenkomen. Een situatie waarbij alleen leveranciersspecifieke webservices op een juiste wijze met elkaar kunnen communiceren, is tegenstrijdig met de ideeën waarvoor webservices ooit bedacht zijn. Het is dus in het belang van de klant dat er wel voldoende draagvlak gevonden kan worden. Dit geldt overigens ook voor alle andere standaarden die een bijdrage kunnen leveren aan de kwaliteitsverhoging van webservices. Het is echter de verwachting dat binnen enkele maanden te beoordelen is of de standaard breed geaccepteerd zal worden of een niche zal blijven.

Richard Claassens en Bert Ertman.

Claassens en Ertman zijn beiden als ICT architecten werkzaam bij Info Support B.V.