

End to end Beheer  
End to end service level management van applicaties

1

## 8.3 End-to-end service level management van applicaties

Moderne applicaties maken gebruik van veel netwerkcomponenten, en zijn daarom voor hun beschikbaarheid vaak afhankelijk van veel beheerders. Welke aanpakken blijken succesvol te zijn in de bewaking van de beschikbaarheid van deze applicaties? In dit artikel wordt besproken op welke manieren gewerkt kan worden aan de beschikbaarheid van applicaties, en wat daarvan de voor- en nadelen zijn.

Applicatiebeschikbaarheid aan de gebruikerskant is het belangrijkste. Op het koppelvlak van gebruiker en applicatie, de user interface, telt vooral beschikbaarheid: doet hij het, en is het snel genoeg? Als deze beschikbaarheid onvoldoende is, klaagt een gebruiker meestal bij de verkeerde mensen, namelijk degenen die er niets aan kunnen doen, maar worden hierdoor wel afgeschrikt zelf de applicatie te gebruiken.

*Auteur: Peter van Eijk is adviseur bij Deloitte & Touche Bakkenist in Diemen.*

*E-mail: [pvaneijk@deloitte.nl](mailto:pvaneijk@deloitte.nl).*

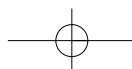
### INLEIDING

Organisaties en individuen maken steeds meer gebruik van complexe netwerktoepassingen. Het belang van deze toepassingen wordt steeds groter, de schaal van de toepassingen wordt groter, hun ingebruikname moet steeds sneller, en het aantal partijen dat ermee verbonden wordt neemt toe. Tezelfdertijd worden deze toepassingen steeds complexer in hun techniek, en kunnen meer partijen dan ooit de kwaliteit en beschikbaarheid van deze applicaties in positieve of negatieve zin beïnvloeden. De waargenomen kwaliteit van een e-commerce-site is bijvoorbeeld

afhankelijk van de kwaliteit van de applicatie, de hosting, de nameservers op het internet, de internetprovider van de gebruiker, de computer van de gebruiker, de afhandeling van de bestelling in de back-office etcetera. In een situatie waarin een aantal partijen een onderdeel van een dergelijke oplossing leveren is het een zeer bekend verschijnsel dat men bij een probleem 'van het kastje naar de muur' wordt gestuurd. Elke partij houdt stug vol dat het probleem niet bij hen zit, want alles wat bij hen hoort te werken, doet het, volgens hen.

Elke van deze ontwikkelingen op zichzelf is al

8



2

voldoende om beheerders slapeloze nachten te bezorgen. Tezamen genomen zorgen ze voor een bijna onontwarbare kluwen.

Wat zijn de implicaties voor het beheer? Welke aanpakken blijken nu wel en welke blijken niet succesvol te zijn in de bewaking van de beschikbaarheid van applicaties, waar vele partijen betrokken zijn bij het beheer? Wat we waarnemen is dat traditionele manieren voor het inrichten van beheer tekortschieten, en dat modernere vormen van samenwerking, soms ondersteund met technologie, nodig zijn.

De eenvoudigste vorm van afspraken maken is géén afspraken maken. Ook al is het duidelijk dat dat niet wenselijk is, in de praktijk komt het daar toch vaak op neer. Het blijkt namelijk erg moeilijk om sluitende, gedetailleerde technische afspraken te maken waar je een tijd mee vooruit kunt. Een verouderde of onduidelijke afspraak is weinig méér waard dan geen afspraak.

### HOE METEN WE?

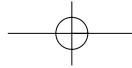
Wat voor beschikbaarheid is nu werkelijk van belang? Degene die verantwoordelijk is voor de beschikbaarheid van computertoepassingen kan zijn aandacht maar al te makkelijk laten afleiden door bespiegelingen over technische parameters zoals MHz en bits per seconde, de robuustheid en redundantie van verschillende deelsystemen, of de aanwezigheid van een escalatieladder.

Voor de gebruikers telt echter maar één ding: de transacties die zij willen uitvoeren met het systeem, moeten uitgevoerd worden wanneer zij dat willen, en met een redelijke doorlooptijd. Bij transacties denken we hier bijvoorbeeld aan orderinvoer, versturen van e-mail, inloggen, raadplegen van webpagina's of het besturen van een computerspeltje. Veelal zijn die transacties weer opgebouwd uit deeltransacties, die uiteindelijk zijn opgebouwd uit user interface-elementen zoals toetsaanslagen of muisclicks.

Een voorbeeld: stel dat men een bestelling of aanvraag wil doen via een website. Een redelijke eis lijkt om dat in enkele minuten te willen afronden. Na het vinden van de betreffende website wil men dan bijvoorbeeld binnen enkele tientallen seconden de juiste pagina gevonden hebben. Die moet dan binnen enkele seconden verschijnen. Een uitgebreide of complexe zoekoperatie mag dan ook wel 10 seconden of zo duren, maar het intoetsen van letters heeft toch liefst wel binnen 200 milliseconden een waarneembaar resultaat op het scherm. Kortom: de acceptabele doorlooptijd hangt voor de gebruiker nogal af van de precieze transactie. Om de beschikbaarheid en de bruikbaarheid van een applicatie te beoordelen is het dus in eerste instantie van belang om van de belangrijkste transacties de gewenste maximale doorlooptijd vast te stellen. Iets dergelijks geldt voor de faalkans: die wil men zo beperkt mogelijk hebben. In deze analyse is het in eerste instantie niet belangrijk door welke componenten deze getallen worden beïnvloed. Zo zal de netwerkbeheerder waarschijnlijk weinig invloed hebben op de snelheid van de pc die de echotijd van toetsaanslagen bepaalt. Voor de gebruiker is dat niet van belang; die maakt alleen onderscheid tussen zichzelf en alle ICT bij elkaar.

De transacties dienen uiteindelijk een bedrijfs- of persoonlijk doel. Men doet er zijn werk mee, en communiceert ermee naar andere mensen. Als deze transacties soepel verlopen voelen mensen controle over hun werk, contact met elkaar, dan wel hebben plezier. Als transacties niet soepel verlopen kan men zich machteloos voelen, afgesneden, en simpelweg boos. Denk maar eens in: je moet een formulier op het scherm invullen, maar het gaat zeer langzaam of helemaal niet. Je kan niet meer bij je e-mail, terwijl je een belangrijke boodschap verwacht. In een multi-user game bevriest het beeld af en toe, waardoor je een eenvoudig slachtoffer bent voor je tegenstanders.

Het is mogelijk, maar niet altijd eenvoudig, om de verwachting te kwantificeren. Een



## End to end Beheer

End to end service level management van applicaties

voorbeeld hiervan is als volgt: een webpagina zou 99% van de pogingen binnen 5 seconden geladen moeten zijn (het is ook mogelijk criteria te stellen aan foutcondities: als een zoekactie geen resultaat oplevert wil je dat binnen bijv. 10 seconden weten). Er is een vrij uitgebreide ontwerpdiscipline (user interface- en interactieontwerp) waarmee aannames over wat voor een computergebruiker wel of niet acceptabel is kunnen worden uitgewerkt en getoetst. Met deze technieken kunnen ook aannames over indicatoren naar prestatie en beschikbaarheid worden getoetst.

### KOSTEN VAN SLECHTE BESCHIKBAARHEID

Wat is het effect van onvoldoende beschikbaarheid? Om te beginnen is er in een zakelijke situatie natuurlijk productieverlies omdat medewerkers moeten wachten. Vervolgens worden er kansen gemist omdat klanten te lang moeten wachten: wie ontevreden is over de respons gaat naar een ander. Deze zaken worden instandgehouden en verergeren doordat mensen doorgaans hun waardering uiten bij de leverancier, en hun klachten bij de andere gebruikers. Voor de leverancier zou het beter zijn als dat andersom was. Dit is niets nieuws; het is in de marketing een bekend verschijnsel. De leverancier zou dus veel baat hebben bij een manier om onafhankelijk van de gebruikers te weten wat de waargenomen prestatie en beschikbaarheid bij die gebruiker is. Elk gegeven levert een bijdrage aan dat beeld.

Door voortdurend bij alle gebruikers een prestatie meting te doen kan een diagram worden getekend waarin de ene as de tijd en de andere as de lokatie aangeeft. Als in dat diagram slechte prestaties worden uitgezet, wordt bijvoorbeeld zichtbaar of een prestatieprobleem locatie- of tijdgebonden is. In het ene geval zou dat kunnen duiden op een langzame verbinding naar bepaalde gebruikersgroepen toe, in het andere geval zou het kunnen duiden op een spitsuur, dat overbe-

lasting geeft op centrale machines. Voor kantoorapplicaties is tussen tien en elf in de ochtend een bekende piek, gevolgd door een lunchdip rond half een. Voor privé-gebruik van websites ligt de piek veelal tussen negen en tien 's avonds.

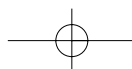
Als we trachten de consequenties van slechte beschikbaarheid in geld uit te drukken, zien we vaak schokkende getallen. Als een belangrijke applicatie in een organisatie met 1000 medewerkers van 98% beschikbaarheid naar 99% beschikbaarheid gaat (beide getallen zijn al vrij aardig), komen er in principe 10 medewerkers vrij. De kosten daarvan geven een ordegrrootte aan van de ruimte die er is voor investeringen in beschikbaarheid.

### INVESTERINGEN IN BETERE BESCHIKBAARHEID

Wat de situatie instandhoudt is natuurlijk dat het latente productieverlies vrij onzichtbaar is, terwijl de kosten voor verbetering zichtbaar en out-of-pocket zijn. Er zijn weinig organisaties waar het effect van downtime op gebruikersprocessen systematisch wordt geregistreerd. Dat is dan dus ook niet zichtbaar. Wel zichtbaar zijn de kosten van een voorgestelde uitbreiding in capaciteit. De financiële afweging tussen investeringen en gemiste kansen is daarom vrij moeilijk. Ofwel er zijn teveel spullen (servers en lijnen) wat dus te duur is, ofwel er zijn te weinig spullen, wat leidt tot onvoldoende beschikbaarheid. Het maken van schattingen van de benodigde capaciteit is op voorhand erg moeilijk, en bovendien kan het gebruikspatroon van de ene op de andere dag wijzigen. De technische infrastructuur die nodig is voor moderne netwerktoepassingen is dermate complex dat voorspellingen over prestaties vrijwel onmogelijk zijn, ook als de (onrealistische) aanname wordt gemaakt dat alle softwarecomponenten zonder fouten zijn. Gebruikers zijn verder snelkiezende wezens. Als iets niet werkt, wordt het snel verlaten. Als iets een succes is, wordt het snel doorverteld. Ironisch genoeg maken nieuwe netwerktechno-

3

8



4

logieën dit steeds makkelijker. Als het netwerk er eenmaal is, kan elke applicatie zeer eenvoudig een enorm publiek bereiken zonder dat zeker is gesteld dat de tussenliggende netwerkcomponenten daarop gedimensioneerd zijn.

### TRADITIONELE SLAS

Traditioneel worden afspraken tussen gebruikers en aanbieders vormgegeven in SLA's (Service Level Agreements, of dienstniveauovereenkomsten). In dergelijke overeenkomsten worden kwaliteitsparameters van te leveren diensten vastgesteld. De meest primaire voorbeelden van deze parameters zijn openingstijden en reactietijden op storingsmeldingen. Ook meer technische parameters kunnen in een SLA worden opgenomen. Zo is er een telecomoperator die garandeert dat internetpakketjes binnen 120 milliseconden de weg tussen twee van hun routers aan weerszijden van de Atlantische Oceaan kunnen afleggen.

SLA's zijn zowel van toepassing op uitbestedingsrelaties als op interne klant-leverancierrelaties. De business/gebruikersorganisatie laat vaak de behoefte voor informatieopslag, bewerking en verspreiding vervullen door de interne ICT-afdeling. Op veel aspecten is deze interne klant-leverancierrelatie gelijkwaardig aan die van externe relaties. Uiteraard zijn er verschillen, onder andere op het gebied van aansturingsmogelijkheden, prijzen en leverancierselectie.

De inhoud van deze SLA's is vaak een lastig punt. In veel gevallen worden deze opgesteld vanuit het perspectief van de dienstverlener – veel te vaak een te technisch verhaal, zie bijvoorbeeld bovengenoemde telecomoperator. De afnemers hebben veel moeite om te begrijpen wat deze afspraken betekenen. De vertaling ontbreekt naar de mogelijkheden, aansprakelijkheden en garanties die echt van belang zijn voor de uiteindelijke gebruiker.

Een leverancier is, bijna per definitie, niet geneigd om garanties af te geven over datgene wat buiten diens invloed ligt. Gevolg is dat datgene wat voor de afnemer van werkelijk belang is, niet wordt gespecificeerd, of met heel veel 'kleine lettertjes' wordt omgeven. Extra druk uitoefenen op de leverancier resulteert dan alleen nog maar in het meer zekerstellen van de eigen belangen en verdere vertraging in de opstelling van de overeenkomst.

Bovendien zijn er vaak geen instrumenten om de naleving van de afspraken te controleren. Als deze er al zijn, worden ze vaak door de leverancier zelf beheerd, niet geheel onafhankelijk dus!

### Onafhankelijke kwaliteitsmonitoring van ISP

De tevredenheid van de gebruiker is niet altijd eenvoudig te vertalen naar service levels van toeleveranciers. Het is dan handiger om die tevredenheid centraal te stellen en een permanent verbeterproces af te spreken.

Een ISP huurt dial-up (inbel) en ip-uplink infrastructuur (internetverbindingen) ten behoeve van haar klanten/gebruikers. Voor de ISP is de kwaliteit van de geleverde diensten lastig te beoordelen, want de feedback van gebruikers is zeker niet altijd even betrouwbaar. De uitbesteding is uiterst relevant voor de bedrijfsvoering: van de totale omzet gaat 80% naar services van toeleveranciers. Als gebruikers weglopen gaan de kosten gewoon door. Deze getallen illustreren het belang van kwaliteit voor het voortbestaan van de ISP. Actieve monitoring van de geleverde kwaliteit, vanuit het perspectief van de gebruikers, is daarmee voor de ISP noodzakelijk.

Indien een ISP rechtstreeks een monitoringsservice installeert of huurt, en met de resultaten aanklopt bij de leveranciers van de connectiviteit, is het erg makkelijk (en gebruikelijk)

voor de leverancier om de testmethodiek te verwerpen, en hiertegenover zijn eigen interne resultaten te zetten. Een onafhankelijke derde partij in de vorm van een ervaren adviseur wordt dan noodzakelijk. Vooraf definieert deze met zijn klant (ISP) wat de kritische parameters zijn die bepalen of de leverancier aan zijn verplichtingen voldoet. In overleg met de leverancier worden de te leveren service levels bepaald, en kunnen daar tevens financiële afspraken aan gekoppeld worden in het geval de leverancier het overeengekomen niveau van dienstverlening over een afgesloten periode niet haalt. Als laatste wordt een testplan opgesteld samen met een monitoringbureau. Vervolgens is er een zelfregulerend systeem ontstaan waarbij

- a iedereen weet dat de performance continu gemonitord wordt;
- b geen discussie over de feiten meer kan ontstaan;
- c er een situatie wordt gecreëerd waarbij gezamenlijk gezocht kan worden naar nieuwe kwaliteit in dienstverlening.

Op deze wijze is het vluchtige product 'connectiviteit' zichtbaar en managable geworden.

SLA's met alle betrokken (beheer)partijen maken garandeert dus weinig en kost veel te veel werk. De gebruikelijke manier van werken is dat een beheerorganisatie pas garanties afgeeft op het moment dat zij die meent waar te kunnen maken. In het algemeen is dat ver na de ingebruikname, en dat wordt alleen maar erger omdat de oplossingen feitelijk steeds complexer worden. Bovendien weet niemand uitputtend te benoemen welke componenten werkelijk de prestatie beïnvloeden en hoe ze dat doen.

## DE WERELD IS TE COMPLEX

De ontwikkelingen op het gebied van netwerktoepassingen maken dat er meestal meerdere beheerders een essentiële bijdrage leveren aan de prestaties van deze toepassingen. Een van de eerste oorzaken daarvan is gelegen in het feit dat netwerken worden gebruikt om autonome organisaties te koppelen. Elk van deze organisaties heeft al eigen beheerorganisaties, en koppelingen voegen daar waarschijnlijk weer nieuwe aan toe. Een andere oorzaak is in de functionele taakverdeling die er noodzakelijkerwijs tussen beheerders moet zijn. Dat zijn dan bijvoorbeeld regionale beheerders, en gespecialiseerde beheerders voor communicatie, internet en hostingdiensten. Elk van deze beheerders heeft een belang om de eigen verantwoordelijkheid goed af te timmeren, en alleen datgene te garanderen waarvan men overtuigd is dat men het kan waarmaken. Dat is jammer, want men kan meestal meer. Die extra vaardigheden komen echter pas tot ontwikkeling na een leerproces. In het algemeen is die leerervaring er pas enige tijd na de ingebruikname. Dat wordt alleen maar erger omdat de oplossingen feitelijk steeds complexer worden, terwijl de afspraken juist in de eerste fases sturend zouden moeten zijn. Zoals gezegd, niemand weet uitputtend te benoemen welke componenten werkelijk de prestatie beïnvloeden en hoe ze dat doen. Welke Domain Name Server-componenten zijn nu werkelijk van belang? Als er een vliegtuig neerstort naast een netwerkknooppunt, wanneer stopt dat dan met functioneren? Welke bugs zitten er nog verstopt in de software? Men ziet dus eenvoudig iets over het hoofd. Dit leidt er vervolgens bij problemen toe dat de gebruikersorganisatie van het kastje van de ene dienstverlener naar het muurtje van de andere wordt gestuurd.

Door een onvolledig zicht op essentiële componenten kan de beschikbaarheid dus slechter zijn dan we dachten. Het omgekeerde komt ook echter ook voor. Moderne netwerkprotocollen en redundantie maken dat het systeem als geheel een hogere beschikbaar-

6

heid kan hebben dan elk van de delen. De wereldwijde DNS-servers (die bijvoorbeeld webadressen helpen vertalen naar machinenummers) hebben soms tussen de 80 en 90% uptime. Dat niemand daarvan last heeft komt doordat er van elke DNS-server meerdere zogeheten 'secondaries' zijn, die permanent worden gesynchroniseerd. Binnen het internet is een mechanisme aan het werk dat verkeer automatisch omleidt als er verbindingen wegvallen (de zogeheten routing-protocollen).

Het punt is hier dat een gedegen analyse van de invloed van de beschikbaarheid van componenten op de totale beschikbaarheid erg moeilijk kan zijn.

We zien verder dat afspraken over beheer nadjien op de werkelijkheid die beheerd moet worden. Projectmatig bekeken is een van de grote nadelen daarvan dat de behoefte aan dat beheer juist in het begin van de levenscyclus van een applicatie het grootste is. Dan zijn er nog de meeste misverstanden en configuratiefouten.

Kortom: gebruikerseisen doorvertalen naar SLA's met alle betrokken (beheer)partijen maken garandeert weinig, kost heel veel werk, en loopt achter de feiten aan

## DE BEHEERDERS WILLEN OOK LEREN

Het boeiende is natuurlijk dat deze ontwikkelingen ook door de beheerorganisaties niet als gewenst worden gezien. Uiteraard wil men zich als beheerder niet aansprakelijk stellen voor problemen waarop men geen invloed heeft, maar de kwaliteit van de dienstverlening naar de gebruikers en opdrachtgevers en de effectiviteit waarmee men die dienstverlening uitvoert zijn voor het voortbestaan van de beheerorganisatie van groot belang. Door zich terug te trekken op datgene wat men vrij zeker kan leveren ontwikkelt men geen nieuwe kennis en vaardigheden, en mist men op den duur de aansluiting met de markt. Geen leverancier kan het zich veroorloven om geheel blind te zijn voor de

wijze waarop haar dienstverlening past in een groter geheel. Op zijn minst levert dat inzicht in de ontwikkelingen aanknopingspunten op voor de inrichting van de dienstverlening en voor nieuwe diensten.

### Het kan ook zonder formele afspraken werken

Ook zonder gedetailleerde formele afspraken over beschikbaarheid is het mogelijk om netwerken te ontwikkelen die goede diensten leveren.

Het internet zelf is een conglomeraat van dienstverleners en softwareontwikkelaars. Om dit te laten werken is een veelheid aan hardware, software en netwerkcomponenten nodig. Zeker in de vroege jaren, rond 1980 en 1990, waren er weinig formele opdrachtgever/opdrachtnemerrelaties. Bandbreedte was vaak gesponsord, softwareontwikkeling was veelal public domain, en systeembeheerders deden hun werk erbij, of in ieder geval zonder dat hun baas begreep waar ze mee bezig waren. Desondanks is juist in die tijd de benodigde technologie enorm ontwikkeld, evenals de kwaliteit van de dienstverlening. Voor wie het van dichtbij heeft meegemaakt zijn er een aantal factoren aan te wijzen die een en ander tot een succes hebben gemaakt. Beheerders en ontwikkelaars hadden een sterke gezamenlijke professionele identiteit. Het hoogste goed was niet de baas tevreden te stellen, maar de gebruikers en medebeheerders bij andere organisaties. De sterren waren die personen die een belangrijk protocol of een veelgebruikt stuk software op hun naam hadden staan, of die een belangrijke netwerkdienst instandhielden. De leidende criteria waren: wat hebben de gebruikers nu en binnenkort nodig? Hoe maken we het beheer beter en eenvoudiger? Ik help een ander, want een ander helpt mij ook. Formele afspraken tussen dienstaanbieders waren er nagenoeg

niet. Technische standaardisatie werd in professionele groepen uitgewerkt, maar verder stonden daar geen formele sancties op.

### FIRST THINGS FIRST: DE GEBRUIKER

Een heel andere aanpak is om te beginnen met het vaststellen van de verwachtingen van de gebruikers en deze te vertalen naar concrete normwaarden voor prestatie-indicatoren. Vanaf het allereerste begin van 'operationeel' zijn, bijvoorbeeld in de ontwikkel- of testfase, worden deze prestatie-indicatoren gemeten en getoetst tegen de normwaarden. Deze meetresultaten worden dan gedeeld met alle partijen die een bijdrage kunnen leveren aan de end-to-end-prestaties en beschikbaarheid.

Het zal nodig zijn om deze verwachtingen, resultaten en inzichten proefondervindelijk door te vertalen naar belangrijke koppelvlakken en daar hetzelfde mee te doen. De belangrijkste koppelvlakken zijn die tussen de dienstaanbieders onderling. De interne koppelvlakken zijn voor de dienstaanbieder wel van belang, maar niet voor diens klanten en toeleveranciers. Zo is het van belang om te meten hoe lang het maken van een telefoonverbinding naar het internet duurt, gerekend vanaf de opdracht van de gebruiker, en wat de kans is dat die verbinding niet lukt. Een internetprovider kan alleen constateren dat zijn systemen (bijvoorbeeld het inbelmodemrek) overbelast zijn, hij kan niet vaststellen dat het inbelprobleem van de gebruiker wordt veroorzaakt door overbelasting van het telefoonnet. De inbelkans van de gebruiker wordt dus gemeten, maar ook de inbelkans op het koppelvlak tussen telecomoperator en internetprovider.

Vervolgens wordt in de aansturing van de dienstaanbieders en hun onderaannemers niet alleen naar de technische prestatie gekeken, maar ook naar de mate waarin zij rapporteren over de kwaliteit van de dienstverle-

End to end service level management van applicaties

End to end Beheer

ning en de inspanning die zij leveren om die te verbeteren. Vanzelf gaat dit niet; dienstaanbieders hebben de neiging zich te verzetten tegen het over hun hoofd uitvoeren van metingen over hun prestaties. In de beginfase zou de kwaliteit en tijdigheid van deze rapportages zelfs zwaarder moeten wegen dan kwaliteit van de dienstverlening waarop zij betrekking heeft.

Als zij echter de wederkerigheid en het nut van de manier van werken inzien, kan daarmee tot een nieuwe manier van samenwerken worden gekomen. De organisatiestijl verandert daarmee van hiërarchisch/formeel naar zelflerend en optimaliserend. Ook vertegenwoordigers van de gebruikersorganisatie draaien hierin mee. Deze zijn namelijk waarschijnlijk het beste op de hoogte van prognoses over aantallen gebruikers en de verdeling van hun transacties over de dag.

### METEN BIJ DE GEBRUIKER

Om deze gezamenlijke aanpak technisch te ondersteunen is het van belang om op vitale punten de prestaties van het geheel goed te kunnen meten. Het belangrijkste punt daarvoor is bij de gebruikers, op hun computer. Dat kan met extra software, of door de applicaties aan te passen. Deze technologie heet ook wel 'management agents'. Dat klinkt exotischer dan het is. Voor een webgebaseerde applicatie is het bijvoorbeeld redelijk eenvoudig om, met behulp van Javascript, een meting in de webpagina te doen, en die naar een centrale server te sturen. Voor applicaties die met een stuk client-software draaien kan deze client-software op een vergelijkbare manier worden aangepast. Er zijn ook oplossingen die geen ingrijpen in de software vergen. Deze bestaan uit een soort 'netwerk-sniffer', die al het verkeer van en naar de servers bekijkt, en op netwerk-niveau zowel de client- als de server-responsetijden kan waarnemen. In alle gevallen wordt er gelogd naar een database met managementinformatie. Vanuit deze database kunnen vervolgens analyses en rapportages worden gemaakt.

7

8

### Begin bij de gebruiker

Als een groot aantal beheerorganisaties invloed kan hebben op beschikbaarheid, is het vrijwel ondoenlijk om op voorhand met elk van deze organisaties sluitende afspraken te maken. Metingen dicht bij de gebruiker zijn eenvoudiger en relevanter.

Een ministerieel departement ontwikkelt een intranet voor kennismangement. Uiteindelijk zullen er 7000 gebruikers zijn, verdeeld over enkele tientallen lokaties. Bij slechte prestaties van het intranet zullen gebruikers afhaken en mist het intranet zijn doel. In totaal zijn er enkele tientallen beheerorganisaties betrokken, dat wil zeggen: in staat de beschikbaarheid te beïnvloeden. De organisatie neemt van de verschillende beheerders ICT-infrastructuurdiensten af (webserver, netwerk, werkplekken met browsers). Het is vrijwel onmogelijk om hier met iedereen een compleet SLA af te sluiten.

In de betreffende intranetapplicatie zitten performance-tests ingebouwd. Deze tests registreren de performance van de applicatie vanuit het perspectief van de eindgebruiker: hoe lang duurt het voordat een pagina op het scherm staat, hoe lang duurt een transactie etcetera. De resultaten van deze tests worden op periodieke basis door een adviseur geanalyseerd en gepresenteerd aan zowel de klant (business) als leverancier (ICT). Deze analyse en rapportage vergelijkt de resultaten met de afgesproken SLA's, identificeert pijnpunten en bevat voorstellen voor verbeteringen in de (technische) infrastructuur en het ontwerp van de applicatie.

Het is ook mogelijk deze metingen en analyses in handen te leggen van een gespecialiseerde dienst aanbieder. Dat heeft onder meer als voordeel dat daarmee een stuk on-

afhankelijkheid wordt gewaarborgd. Verder kan deze dienst aanbieder kennis inbrengen over wat zinvolle rapportages zijn, en de analyses consolideren tot wekelijkse of maandelijks rapportages.

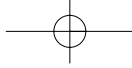
Naast metingen bij de gebruikers zijn ook metingen op de koppelvlakken tussen de aanbieders zinvol. Als eerste nut hiervan geldt natuurlijk dat daarmee vaak de bron van een gebruikersprobleem kan worden aangewezen. In het internet voorbeeld is het bijvoorbeeld van belang om te weten of de webserver traag is, of het tussenliggende netwerk.

Als op de koppelvlakken wordt gemeten, wordt ook duidelijker waar de oplossing van problemen het beste kan worden gezocht. Voor het verhelpen van een onvoldoende respons is de voor de handliggende aanpak om eerst de component met de grootste bijdrage aan die responstijd te onderzoeken. Niet-tegenstaande dat, is het niet altijd de beste aanpak om de langzaamste component aan te pakken. Soms is het kostentechnisch interessanter om voor een andere oplossing te kiezen. Als een lijnverbinding traag is kan opwaarderen bijvoorbeeld duurder zijn dan het inzetten van compressietechnologie.

### CONCLUSIE

Wat we dus zien is dat de gebruikelijke aanpak, waarbij eenmalig harde service levels worden afgesproken, niet van de grond komt. Daarmee verwordt deze aanpak tot een aanpak waarbij in feite niets wordt geregeld. Door het erkennen van de onmogelijkheid van totale beheersing wordt het, paradoxaal wellicht, pas mogelijk die beheersing in te richten. Immers, die erkenning staat toe dat de betrokken partijen hun eigen beperkte vaardigheden op constructieve wijze, met elkaar, verder ontwikkelen, en samen leren hoe deze technologie werkelijk een bijdrage kan leveren.





End to end Beheer

End to end service level management van applicaties

**PRAKTISCHE TIPS**

Ook al is het een open deur: de belangrijkste gegevens voor end-to-end-prestatiebeheer komen van een meting op de clients. Dat is in feite de meest nauwkeurige bepaling van de techniek die de gebruikerstevredenheid beïnvloedt. Het op een min of meer permanente basis bewaken hiervan is van groot belang. Het is verder aan te raden om deze bewaking

in een zo vroeg mogelijk stadium van ontwikkeling in te richten. Juist in het begin van de levenscyclus van een applicatie worden de meeste configuratiefouten gemaakt, en is er daarom de grootste kans op problemen. Als de applicatie eenmaal werkt zal de gebruikerspopulatie snel groeien. Hoe snel is een interessant gegeven op zichzelf, maar nog boeiender is de invloed op de prestaties die dit heeft.

9

