

ICT-beheerdomeinen laten samenwerken (deel 5)

Samenwerken bij wijzigingen op applicaties – vervolg

In het vierde deel van de artikelenreeks over de samenwerking tussen de ICT-beheerdomeinen zijn Machteld Meijer en Frances van Haagen ingegaan op bepaalde onderdelen van het doorvoeren van wijzigingen in de informatievoorziening. Het ging met name om het belang van samenwerking bij het opstellen en verwerken van de requirements en bij het bepalen van de consequenties van een wijzigingsverzoek, de impactanalyse. In dit vijfde deel komen enkele andere aspecten van wijzigingenbeheer aan de orde, zoals de samenwerking bij de planning van een release, het ontwerpen en het testen.

Machteld Meijer en Frances van Haagen

Dit artikel heeft betrekking op wijzigingen in maatwerkapplicaties. Wijzigingen in standaardpakketten zijn uiteraard ook relevant, maar komen hier niet aan de orde.

Planning en besturing

Vooraf bij grote releases of vernieuwingslagen is het belangrijk dat verantwoordelijkheden goed worden belegd. In de praktijk zien we vaak dat niet duidelijk is wie de regie heeft over een wijziging op een applicatie en wie verantwoordelijk is voor deelprocessen, zoals de impactanalyse en het testproces. Meestal werkt het niet goed om 'samen' ergens verantwoordelijk voor te zijn; beter is het om één functie aan te wijzen die verantwoordelijk is. Dit vereist dan wel dat deze functie de noodzakelijke bevoegdheden heeft, bijvoorbeeld voor het aanvragen van de benodigde resources voor de impactanalyse.

Bij dergelijke grote wijzigingen zijn vaak veel partijen betrokken. Aan de opdrachtgeverskant kunnen dat functioneel beheerders, informatieanalisten, kerngebruikers (*super users*) of proceseigenaren zijn. Aan de ICT-kant zijn dat applicatie-, systeem-, netwerk- en werkplekbeheerders, elk met hun di-

verse afdelingen en specialismen. Vaak zijn er meerdere leveranciers bij betrokken, terwijl het aanpassingstraject veel momenten kent waarop de activiteiten van de diverse partijen op elkaar moeten aansluiten. Pas als alle requirements voor de wijziging helder zijn, kan applicatiebeheer (AB) ze uiteindelijk doorvoeren. Tussentijdse verandering van eisen kan veel meerwerk met zich meebrengen, zeker als de planning al in een vroeg stadium is vastgelegd. Dan is het risico van tijdsoverschrijding groot. Om toch op tijd op te leveren, neemt AB dan bijvoorbeeld niet voldoende tijd om te testen, waarna blijkt dat de gebruikersorganisatie onvoldoende testers heeft of de acceptatietestomgeving niet was klaargezet.

Iemand moet dus in de gaten houden dat de planning van op elkaar aansluitende werkzaamheden goed wordt gevolgd en dat de noodzakelijke wijzigingen in de planning goed worden doorgevoerd. In de praktijk neemt de projectleider van AB bij aanpassingen in maatwerkapplicaties de rol op zich om ook de collega's van technisch beheer (TB) aan te sturen. Maar als functioneel beheer (FB) bij de opdrachtgever zelf zijn plannen niet kan waarmaken, heeft de projectleider

Reeks samenwerkende beheerdomeinen

Organisaties hebben er belang bij om hun informatievoorziening op peil te houden. Daartoe moeten de ICT-diensten goed aansluiten op de behoeften. Vandaar dat er steeds meer aandacht is voor het verbeteren van de ICT-beheerprocessen, aan zowel de vraagkant als de aanbodkant van ICT. Hiervoor wordt steeds vaker gebruikgemaakt van drie op elkaar aansluitende procesmodellen: ITIL (voor het inrichten van servicemanagementprocessen, met het accent op het beheer van technische infrastructuren), ASL (voor applicatiebeheer) en BiSL (voor functioneel beheer). In deze modellen worden processen en activiteiten genoemd die plaats zouden moeten vinden binnen de drie genoemde beheerdomeinen. Maar uiteraard is samenwerking tussen de drie domeinen onontbeerlijk. Bovendien hoeven deze beheerdomeinen niet altijd synchroon te lopen met de beheerorganisaties; bepaalde applicatiebeheeractiviteiten worden ook wel eens door het rekencentrum of door de afdeling Functioneel Beheer uitgevoerd.

Deze artikelenreeks beschrijft voor vijf belangrijke samenwerkingsgebieden welke koppelvlakken (interfaces) er zijn tussen de modellen, welke processen op elkaar aan moeten sluiten en hoe deze activiteiten verdeeld zouden kunnen worden over verschillende afdelingen of organisaties.

AB onvoldoende bevoegdheden om ook daar te kunnen ingrijpen. Daarom is het aan te raden om altijd een eindverantwoordelijke (algemeen projectleider) aan te wijzen, die optreedt namens de opdrachtgeversorganisatie, en hem of haar voldoende bevoegdheden mee te geven. Het is bij voorkeur iemand van de opdrachtgeversorganisatie zelf, maar het kan ook een onafhankelijke projectleider zijn vanuit een ICT-organisatie of een adviesbureau. In figuur 1 is zo'n structuur afgebeeld.

Ontwerp

Tijdens de eerste stappen van het wijzigingsproces wordt bepaald welke wijzigingsverzoeken doorgevoerd worden en worden de eisen aan de nieuwe informatievoorziening (de requirements) opgesteld¹.

Een deel van de requirements zal leiden tot aanpassingen in de niet-geautomatiseerde informatievoorziening, zoals wijzigingen in formulieren of werkinstructies. Deze aanpassingen worden aangebracht door FB. De requirements voor de benodigde wijzigingen in de applicaties worden door AB vertaald in 'bouwbare' functionele en technische ontwerpen. Afhankelijk van de gekozen ontwikkelmethode kunnen de ontwerpen verschillende vormen aannemen, zoals use cases (RUP), functioneel-ontwerprapporten (SDM) en datamodellen. De scheiding en relatie tussen de geautomatiseerde en de niet-geautomatiseerde oplossingen moet uiteraard duidelijk worden vastgesteld.

Tijdens de ontwikkeling van het ontwerp door AB is het nodig om ook TB te be-

trekken om de technische haalbaarheid van de gekozen oplossing te blijven valideren. Dit kan bijvoorbeeld in de reviewrondes.

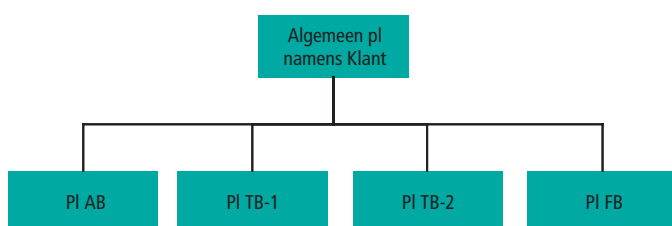
Het functioneel ontwerp biedt de eerste gelegenheid om te toetsen of business en ICT een gemeenschappelijk beeld hebben van wat de wijziging precies inhoudt. In de praktijk wordt er verschillend over gedacht wie het functioneel ontwerp moet schrijven. ASL en BiSL gaan ervan uit dat een functioneel ontwerp een vertaalslag is van *wat* een applicatie zou moeten doen naar *hoe* het werkt in functionele zin, en dat het maken ervan thuishoort in AB. Als definitie² wordt gegeven "Een functioneel ontwerp is een zodanig gedetailleerde nadere uitwerking van de (gebruikers)specificaties van het informatiesysteem of de wijzigingen hierop, dat deze op eenduidige wijze kunnen worden gerealiseerd en getest". Het ontwerp dient twee doelen: de gebruikersorganisatie moet zich ervan kunnen overtuigen dat gebouwd wordt wat bedoeld is, en de ontwikkelaars moeten voldoende houvast hebben om het effectief te kunnen bouwen. Hiervoor zijn kennis en vaardigheden nodig die niet vereist zijn voor de kernactiviteiten van een gebrui-

kersorganisatie. Het is dus verstandig om deze 'vertaalslag' door ICT (veelal AB) te laten maken en het resultaat samen te bekijken. Mogelijk leidt dit overleg weer tot wijzigingen in de requirements. De ontwikkelmethode moet dit iteratieve proces ondersteunen.

Een afdeling FB binnen een gebruikersorganisatie zou juist veel moeten weten van de businessprocessen en -producten, opdat de vraagstelling aan ICT zo goed mogelijk, en zo vaak mogelijk in één keer goed, kan worden geformuleerd. Bij een aantal organisaties hebben we al gezien dat de aandacht voor kennis van de bedrijfsprocessen vermindert wanneer FB zijn taken uitbreidt met ICT-achtige activiteiten zoals het opstellen van use cases of functioneel-ontwerprapporten. De afstand tot de businessunits wordt daardoor vergroot en er ontstaan binnen die businessunits weer nieuwe, vaak niet-geformaliseerde, FB-groepjes om de gebruikers te kunnen ondersteunen en de nieuwe wensen te kunnen formuleren. Hetzelfde lijkt te gebeuren als de centrale FB-organisatie wordt ondergebracht in één afdeling of sector met de ICT-organisatie. Hiermee wordt een extra afstemmingsnoodzaak gecreëerd in het proces dat moet leiden tot de goede informatievoorziening.

Testen

Wanneer een informatiesysteem is aangepast, moet het uiteraard worden getest. AB test onder meer of is gebouwd wat de bedoeling was volgens het functioneel ontwerp en of het resultaat doet wat het moet doen. TB test hoe het aan-



Figuur 1 Structuur voor de projectleiders (pl)

beheerdomeinen

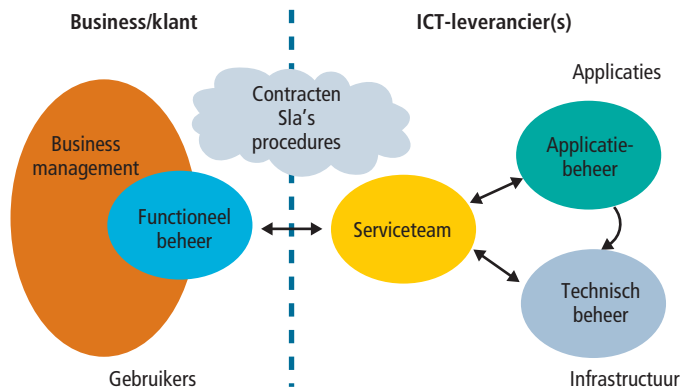
gepaste systeem zich zal gaan gedragen in een productieomgeving; FB test of het voldoet aan de opgestelde requirements en de verwachtingen, en of het past binnen de - mogelijk aangepaste - werkwijze van de gebruikers van het systeem.

Allereerst is vroegtijdig overleg vanuit AB met TB en FB nodig, opdat FB en TB zich kunnen voorbereiden op de tests (als je niet weet wat waar verandert, kun je ook geen goede testgevallen opstellen). Daarnaast komt het geregeld voor dat de functionele systeemtest binnen AB en de acceptatietest binnen FB toetsen tegen dezelfde criteria. Dit gebeurt met name als de requirements ongewijzigd zijn overgenomen in het functioneel ontwerp, zonder nadere verdieping of vertaling. In dat geval worden bepaalde testsituaties dus dubbel voorbereid en dubbel uitgevoerd, terwijl mogelijk andere belangrijke zaken niet worden getest. Maak dus goede afspraken over wie wat test. Het gezamenlijk opstellen van een testplan helpt vaak om een systematisch overzicht te krijgen van de verschillende testsoorten en hun specifieke doelstellingen.

Overige organisatorische aspecten

Om het totale wijzigingsproces goed te laten verlopen, is het vaak verstandig om voor grote wijzigingen of releases in een zo vroeg mogelijk stadium een project in te richten en vooral te zorgen dat de impactanalyse deel uitmaakt van het project.

Het lijkt op het eerste gezicht dat het beheren van de informatievoorziening hooguit drie partijen betreft: FB aan de vraagkant en AB en TB aan de aanbodkant. Zoals we al eerder hebben geconstateerd, wordt in de praktijk FB voor een applicatie soms verdeeld over meerdere organisatieonderdelen. Binnen de AB- en TB-domeinen is echter vaak ook sprake van verschillende partijen, bijvoorbeeld voor netwerkbeheer, werkplekbeheer en systeembeheer. En soms wordt binnen AB het wat grotere



Figuur 2 De plaats van het serviceteam

onderhoud uitgevoerd door een afdeling Ontwikkeling en het correctieve onderhoud door een afdeling Beheer. Dat laatste roept overigens wel een aantal mogelijke bezwaren op: wie is verantwoordelijk voor het versiebeheer en de daarvoor ingezette tooling? Is het wel efficiënt als kennis over de applicatie op meerdere plekken aanwezig moet zijn? Wie is het aanspreekpunt voor de klant?

ASL en BiSL propageren het serviceteam als aanspreekpunt voor functioneel beheer waar afspraken worden gemaakt over de te leveren en al geleverde dienstverlening door AB en TB. Dit serviceteam is het gremium waar veel van de afspraken worden gemaakt over de wijze van samenwerken en monitoren (zie figuur 2).

Één Change Managementproces?

Regelmatig krijgen we de vraag of het niet goed zou zijn om één Change Managementproces te definiëren over alle beheerpartijen heen, met één eindverantwoordelijke, ondersteund door één tool. Dat klinkt als de ultieme manier van samenwerking. In de praktijk kleven daar echter wel enkele grote bezwaren aan.

In de eerste plaats is er geen een-op-een-verhouding tussen de 'changes' zoals ze door FB, AB en TB worden bestuurd. Een reeks van wijzigingsverzoeken (rfc's), die vanuit zowel FB, AB als TB kan komen, leidt tot één release, die als totaal wordt bestuurd door AB. Tijdens het uitvoeren van de release wordt regelmatig TB ingeschakeld, bijvoorbeeld voor het klaarzetten van de ontwikkel-, test- en

acceptatietestomgeving. Hieruit blijkt dat er een $n : 1 : m$ -verhouding is tussen de 'changes' van FB, AB en TB, die alle te maken hebben met het maken van een nieuwe versie van een applicatie. Het is heel belangrijk om de relaties tussen de diverse changes vast te leggen, maar het is niet nodig en niet handig om er één gezamenlijk tool voor te gebruiken. Dat heeft een paar redenen. In de eerste plaats is er weinig tot geen tooling die deze complexe relaties kan ondersteunen. Nog belangrijker is dat een ICT-leverancier mogelijk voor al zijn klanten een andere servicemanagementtool zou moeten gebruiken; dat is niet effectief. Of er zou op de hele wereld nog maar één tool kunnen worden gebruikt; ook dat lijkt geen gewenste situatie. Ook hier is het eerderbeschreven CNIP-principe³ ('cooperation when necessary, independent operation when possible') van toepassing.

Praktijkvoorbeeld

Als gevolg van een wetswijziging is een nieuwe release nodig van een groot en bedrijfskritisch financieel rapportagesysteem. Er worden requirements opgesteld door de klant, er wordt een functioneel en technisch ontwerp gemaakt en de wijziging wordt gebouwd. Bij de acceptatietest blijkt, dat de responstijden van het systeem dramatisch zijn verslechterd. Dit heeft er onder andere mee te maken dat bij TB wijzigingen in de technische infrastructuur hebben plaatsgevonden die niet zijn meegenomen in de impactanalyse. De reactie van ICT is dat de business in de requirements niet heeft aangegeven dat de performance mini-

Afspraken over 'grijze gebieden'

In de praktijk zijn er vaak enkele 'grijze gebieden' waarover geen (duidelijke) afspraken zijn gemaakt tussen de verschillende partijen. De kwaliteit en voortgang van een wijzigingstraject kan hierdoor behoorlijk worden gehinderd.

Voorbeelden hiervan in het wijzigingsproces zijn:

- Wie stuurt TB aan in projecten/releases?
- Wie stuurt FB aan in projecten/releases?
- Wie mag wijzigingsvoorstellen indienen bij wie?
- Welke inputdocumentatie is nodig voor de besluitvorming rond wijzigingen?
- Hoe is de concrete taakverdeling en overlegstructuur tussen de Change Manager van FB en de Change Managers van de ICT-leverancier? Wie beslist formeel over inhoud en planning van wijzigingen en releases?
- Wie is verantwoordelijk voor het initiëren en regisseren van een impactanalyse en voor goed gebruik van de resultaten?
- Hoe wordt de globale oplossingsrichting in de impactanalyse bepaald, en met welke mate van detail?
- Wie bepaalt, in welk stadium, of er een pakket wordt ingezet voor het realiseren van de wijziging? Hoe worden hierbij kaders vanuit leveranciersmanagement (strategisch) en contractmanagement (sturend) toegepast?
- Wie is eindverantwoordelijk voor de kwaliteit en volledigheid van tussenproducten/documentatie, zoals requirements, functioneel en technisch-ontwerp-rapporten, use cases, test cases,
- Wie beheert en onderhoudt de documentatie?
- Wie beslist over de verwerking van de testbevindingen?

Bij het opstellen van een beheerovereenkomst of onderliggende afspraken of procedures zal dus over al deze onderwerpen moeten worden nagedacht en waar nodig zullen afspraken erover moeten worden vastgelegd.

maal hetzelfde moest blijven als in de oude situatie.

Enkele leerpunten:

- Vanaf het begin moeten AB, TB en FB betrokken zijn bij het opstellen van de requirements, ook bij de niet-functionele requirements; hierbij kan een checklist helpen¹.
- TB moet in de impactanalyse ook zijn voorgenomen wijzigingsprojecten inbrengen.
- Bij het maken van de ontwerpen moeten AB en TB ernaar streven om alle kenmerken van de oplossing die voor de klant belangrijk zijn inzichtelijk te maken.

Aan de hand van dit praktijkvoorbeeld zijn de volgende aandachtspunten voor de samenwerking in het wijzigingsproces te noemen:

Vooraf wel doen

- Als FB de regie voeren over het hele wijzigingsproces.
- Onderhoudbaarheid, exploitatiebaarheid en andere niet-functionele aspecten expliciet en proactief meenemen bij het ontwikkelen van de requirements, ook als de gebruiker daar niet zelf mee komt.
- Structureel releaseoverleg voeren tussen FB, AB en TB.
- Alle wijzigingsverzoeken, requirements en ontwerpen documenteren en een beheerder voor de documentatie aanwijzen.
- Zorgen dat versiebeheer en configuratiemanagement goed op orde zijn en hierin afstemmen tussen AB en TB.
- De gebruikers alle mogelijkheden geven om duidelijk te maken wat ze willen hebben, ook als ze bepaalde pakketten voorstellen.
- Goede afspraken maken over de kwaliteit en het tijdstip van oplevering aan elkaar.
- Indien mogelijk afspraken maken over de benodigde kwaliteit van de opleveringen in de vorm van acceptatiecriteria.

Vooraf niet doen

- Te snel besluiten dat een wijziging wel even 'tussendoor' kan.
- Partijen 'vrijblijvend' laten 'intekenen' op een impactanalyse in plaats van erop te sturen dat alle relevante partijen erbij worden betrokken.
- Als ICT zeggen dat de business verantwoordelijk is voor de requirements en zich er verder niet mee bemoeien. Formeel is dat wel juist, maar zo werkt het in de praktijk niet. Het is bijvoorbeeld maar de vraag of de business altijd verantwoordelijk kan worden gehouden voor de goedkeuring van een ontwerp. Blijf altijd in gesprek met elkaar!
- TB pas betrekken bij een release als inproductienamen naderbij.
- Gebruikers pas inlichten als de inproductienamen al een feit is.
- Te snel besluiten dat een pakket tegemoet komt aan de requirements – soms is veel maatwerk nodig, waarop een pakketleverancier vaak niet is ingesteld.
- Als FB inhoudelijk willen sturen op de manier waarop AB de applicaties aanpast of TB de infrastructuur voorbereidt.

Conclusie

Ook bij, of liever gezegd, zeker bij het doorvoeren van wijzigingen op applicaties is het van belang dat heldere afspraken zijn gemaakt tussen alle betrokkenen over de werkwijze en de onderlinge taakverdeling. Die afspraken gelden de kwaliteit en de opleverdata van de uit te wisselen producten als wijzigingsverzoeken, impactanalyseresultaten, requirements, ontwerpen, te testen programmatuur, testverslagen en andere documentatie. Het aanwijzen van een eindverantwoordelijke voor de totale wijziging die met name de planning bestuurt, kan veel problemen voorkomen.

Drs. Frances van Haagen (Ordina) en dr. Machteld Meijer zijn senior management consultant en lid van enkele werkgroepen van de Stichting ASL BiSL Foundation. Opmerkingen, suggesties en best practices met betrekking tot dit onderwerp zijn van harte welkom op Machteld.meijer@tiscali.nl of Frances.van.haagen@ordina.nl.

Noten

- ¹ Zie het vorige artikel in deze reeks, nummer 3/2007, blz. 38.
- ² Wordenlijst ASL, te vinden op www.asl.foundation.org, onder 'ASL Begrippenlijst'.
- ³ Meijer, Machteld en Jolanda Meijers, 'Effectief IT-beheer: samenwerken waar nodig, zelfstandig opereren waar mogelijk', in: *IT Beheer Jaarboek 2002*, ten Hagen & Stam Uitgevers, 2002.