Open Source Software & Business Continuity

Afstudeerscriptie postdoctorale IT-Audit opleiding Vrije Universiteit

Door: Drs. P. Nauta

Titelblad

Titel:	Open Source Software & Business Continuity
Scriptienummer:	708
Auteur:	Drs. Pier Nauta
Opleiding:	Postdoctorale IT-Audit opleiding, Vrije Universiteit
Eerste scriptiebegeleider:	Wim Krol RE
Tweede scriptiebegeleider:	Drs. Rob Christiaanse RA
Studiejaar:	2004-2007

“The arguments for and against open source software get trivialized. It’s not a technology issue; it’s a business issue.”

- CTO Andy Mulholland, Cap Gemini Ernst & Young

Inhoudsopgave

Hoofdstuk 1. Reflectie

1.1. Inleiding

Deze scriptie geeft antwoord op de volgende twee vragen:

1. *Welke risico's zijn voor de IB-Groep te identificeren tijdens een mogelijke overstap op open source software, vanuit het perspectief van business continuity?*
2. *Welke maatregelen moeten minimaal geïmplementeerd worden om deze risico's te minimaliseren?*

1.2. Reflectie

Tijdens het onderzoek is naar voren gekomen dat de kostenbesparing en de wil om niet langer afhankelijk te zijn van één leverancier de redenen zijn om een overstap te overwegen of deze stap te nemen. Vanuit de literatuur blijkt dat business continuity als proces en als doel in de belangstelling staan. Dit blijkt eveneens uit de gehouden interviews. De risico's met betrekking tot een succesvolle implementatie van open source software zijn hieronder in hoofdlijnen weergegeven.

1. Onderschatting of foutieve inschatting van de kosten en baten;
2. Onvoldoende kennis en kunde met betrekking tot de specifiek geldende technische functionaliteiten van de software en het gebruik daarvan;
3. Het hebben van een beheerorganisatie die in onvoldoende mate is ingericht voor de nieuwe situatie;
4. Het ontbreken of niet naleven van een projectimplementatiemethodiek.

De beheersmethodiek moet worden aangepast op de specifieke risico's die open source software met zich meebrengt. De specifieke aandachtspunten hebben betrekking op:

- De openbaarheid van de broncode;
- De gewenste functionaliteiten van de software;
- De juridische consequenties van een nieuw soort licentie.

De IT-infrastructuur van de IB-Groep is complex, met veel onderlinge afhankelijkheden. De complexe situatie is tot stand gekomen door beslissingen in het verleden. Dit heeft als resultaat dat de IB-Groep beschikt over vier platformen: mainframe, iserie, webservers en een Microsoft netwerk voor de kantoorautomatisering. Uitsluitend de laatste twee komen in aanmerking voor een transitie. De IB-Groep heeft drie mogelijke scenario's indien de keuze voor open source gemaakt wordt. Dit zijn:

- De nul optie – dit houdt in dat de huidige situatie niet wordt veranderd;
- Gedeeltelijk implementeren – dit houdt in dat de huidige situatie op bepaalde onderdelen wordt aangepast en dat andere onderdelen niet gewijzigd worden;
- Volledig implementeren – dit houdt in dat de IB-Groep voor alle relevante platformen over gaat op open source software.

Vanuit de techniek geredeneerd bestaat geen bezwaar tegen de introductie van open source software. Open source software moet in eerste instantie gezien worden als een business issue en pas daarna als een technology issue. Indien bij de IB-Groep de wens bestaat om over te gaan tot open source software dient zij maatregelen te treffen die de continuïteit van de organisatie garanderen. Met andere woorden: zij dient maatregelen te treffen die de genoemde vier risico's tot een minimum beperken. Van belang is dat de IB-Groep zich realiseert dat de genoemde risico's een onderlinge relatie hebben. De genoemde risico's zijn mede afhankelijk van de specifieke situatie binnen de IB-Groep. Gerelateerd aan de vier genoemde hoofdrisico's kom ik tot de volgende conclusies:

Vanuit een kosten baten perspectief is duidelijk dat open source software een achterstand heeft. Als er een functionerende IT-organisatie aanwezig is dan is een overstap een financieel risico. Om een overstap te bewerkstelligen moet de organisatie alle kosten en baten objectief in kaart hebben. Dit dient te gebeuren op basis van de Gartner methodiek: Total Cost of Ownership (TCO). Deze methodiek stelt een organisatie in staat om op gestructureerde wijze alle kosten en baten in kaart te brengen. Op basis hiervan kan een onderbouwde beslissing worden genomen.

De huidige organisatie beschikt over voldoende kennis en kunde met betrekking tot de huidige infrastructuur. De kennis en kunde met betrekking tot een nieuwe situatie moeten volledig worden opgebouwd. Zowel binnen de IT-afdeling als in de gebruikersorganisatie heeft dit als risico dat de nieuwe software niet goed wordt geïmplementeerd of gebruikt.

De IB-Groep heeft ITIL als beheersmethodiek geadopteerd en hanteert de Code voor informatiebeveiliging als relevant referentiekader bij de inrichting van de IT-organisatie. De gekozen methodieken zijn onafhankelijk van het type software dat wordt gebruikt. Hierbij moet worden opgemerkt dat de huidige inrichting op specifieke punten moet worden aangescherpt. De reden hiervoor is dat de inrichting aangepast moet worden voor de specifieke risicogebieden die open source software met zich meebrengt.

De IB-Groep heeft een werkende projectimplementatiemethodiek, PRINCE2. Indien de IB-Groep op basis van objectieve en harde criteria een besluit neemt, vormt het juist en volledig naleven van de vanuit PRINCE2 geldende richtlijnen geen risico. Voorwaarde is dat de methodiek nauwgezet wordt nageleefd en dat daarop wordt toegezien. De methode zelf vormt geen garantie dat het project succesvol wordt afgerond. Het niet naleven van een methodiek biedt een hogere mate van zekerheid dat het project mislukt. Dit risico is groter in een complexe omgeving zoals de IB-Groep. De impact is dusdanig dat de continuïteit van de organisatie in gevaar kan komen.

Gezien de huidige complexe situatie van de IB-Groep is mijn advies om niet over te gaan op een volledige implementatie van open source. Eveneens raad ik af om in de toekomst niets te doen met open source software. Ik raad aan om bij de aanschaf van nieuwe software het open source software alternatief mee te nemen in de pakketselectie.

1.3. De rol van de IT-Auditor

Open source software moet in eerste instantie gezien worden als een business issue en pas daarna als een technology issue. Dit houdt in dat de IT-Auditor zich daarvan bewust moet zijn, tijdens het vervullen van zowel zijn toetsende als adviserende rol. Vanuit de beroepsuitoefening is de inhoudelijke kant van belang: de deskundigheid van de IT-Auditor inzake open source software. Ten aanzien van de houding en gedrag is geen onderscheid te herkennen. De geldende regels met betrekking tot de gedragingen van de IT-Auditor blijven staan.

De vier geïdentificeerde risico's zijn primair organisatiekundige risico's met hier en daar een relatie met technologie. Dit heeft tot gevolg dat de IT-Auditor bekend moet zijn met de organisatorische consequenties van technologische keuzes.

De tegenstanders van open source software noemen de openbaarheid van de broncode als potentieel risicogebied. Zwakheden in de software zouden het makkelijker maken voor kwaadwillenden om er misbruik van te maken. Dit mogelijke risico stelt de IT-Auditor in staat om eventuele zwakheden sneller te herkennen en acties te laten ondernemen.

Hoofdstuk 2. Inleiding

De IB-Groep heeft als missie: een excellente dienstverlener in het onderwijsveld zijn. In deze filosofie past een continu streven naar verbetering van haar bedrijfsprocessen en het scherper stellen van haar doelstellingen. In die filosofie moet deze scriptie gezien worden.

Open source software is een onderwerp dat de gemoederen bezig houd in ICT-land. Ook vanuit audit is het een onderwerp dat de nodige aandacht verdient. Het concept open source software is niet nieuw, maar heeft de laatste jaren een vlucht genomen. Met de toegenomen populariteit van open source software is ook voor de IT-auditor het tijdstip aangebroken om serieus aandacht te besteden aan dit fenomeen. Steeds meer organisaties gaan over of overwegen om over te gaan op open source software.

Opvallend is dat het afgelopen jaar ook steeds meer overheden over zijn gegaan op open source software en/of open standaarden. Dit zijn onder andere de gemeenten Almere, Assen, Eindhoven, Enschede, Groningen, Haarlem, Leeuwarden en Nijmegen. Deze groep gemeenten heeft een convenant gesloten waarin zij het gebruik van open standaarden propageert, vanuit een continuïteitsperspectief en vanuit een kostenperspectief. Het laatste om te voorkomen dat de organisaties van één leverancier afhankelijk worden. Daarnaast zijn onder andere verschillende Bundesländer en een deel van de Franse centrale overheid overgegaan. Met een dergelijke ontwikkeling en mede door het aannemen van de motie Vendrik (Groen Links 2002) is binnen de IB-Groep behoefte ontstaan om meer inzicht te krijgen in de voor- en nadelen. Binnen de stafafdeling Audit wordt eveneens met interesse naar dergelijke ontwikkelingen gekeken en afgevraagd welke consequenties dit voor de IB-Groep kan hebben vanuit een audit perspectief.

Ondanks de toegenomen populariteit van open source software is op dit moment weinig tot geen betrouwbare wetenschappelijke literatuur te vinden over de werkelijke voor- en nadelen. Dit heeft mede tot gevolg dat het onduidelijk is wat voor consequenties de introductie kan hebben voor de continuïteit van de organisatie. In hoofdlijnen zijn er uitgesproken voorstanders en tegenstanders. Dit maakte het lastig om een genuanceerde mening of objectieve kritiek te vinden. Het onderzoeken van beide argumentaties vanuit een IT-Audit perspectief was leerzaam en zeer interessant.

Tot slot wil ik iedereen bedanken die mij heeft geholpen bij de totstandkoming van deze scriptie.

Hoofdstuk 3. Probleemstelling en methodologische onderbouwing

3.1. Inleiding

Dit hoofdstuk bevat de aanleiding en de doelstelling van de scriptie. Daarnaast wordt vastgesteld wat de te beantwoorden vragen zijn. Beschreven wordt wat de te beantwoorden deelvragen en geldende randvoorwaarden zijn. Tevens wordt beschreven wat de methodiek is op basis waarvan de onderzoeksvragen worden beantwoord en wat de geldende randvoorwaarden zijn.

3.2. Aanleiding

De motie Vendrik (Groen Links 2002) en het overgaan van verschillende overheden naar open source software heeft intern de discussie doen oplaaien of open source software ook geschikt is voor de IB-Groep. Hierbij zijn de volgende gremia betrokken geweest: de interne auditafdeling, ICT/Beveiliging en het platform informatiebeveiliging. In haar streven om een excellente dienstverlener in het onderwijsveld te zijn, is de IB-Groep continu op zoek naar nieuwe en betere mogelijkheden om haar doelstellingen te realiseren.

ICT en de beveiliging van de informatie zijn voor de IB-Groep van dusdanig belang dat indien de integriteit, vertrouwelijkheid en beschikbaarheid niet gegarandeerd is, de continuïteit van de IB-Groep als organisatie niet langer gegarandeerd is.

3.3. Doelstelling

Tijdens het intakegesprek voor de vaststelling van het onderwerp van de scriptie heeft de opdrachtgever annex begeleider de onderstaande twee vragen geformuleerd. Met beantwoording van deze vragen dient de IB-Groep een beeld te krijgen en een inschatting te kunnen maken over de haalbaarheid van een migratietraject naar open source software.

1. *Welke risico's zijn voor de IB-Groep te identificeren tijdens een mogelijke overstap op open source software, vanuit het perspectief van business continuity?*
2. *Welke maatregelen moeten minimaal geïmplementeerd worden om deze risico's te minimaliseren?*

3.4. Deelvragen

Om antwoord te geven op de twee onderzoeksvragen zijn per vraag een aantal deelvragen gesteld. Deze zijn hieronder weergegeven:

1. *Welke risico's zijn voor de IB-Groep te identificeren tijdens een mogelijke overstap op open source software, vanuit het perspectief van business continuity?*
    a. Wat zijn erkende risico's bij migratietrajecten naar nieuwe software?
    b. Wat zijn erkende risico's van open source software?
    c. Wat zijn de relevante risico aspecten van business continuity vanuit een software perspectief?
    d. Welke bijzonderheden, die direct te koppelen zijn aan open source software, zijn relevant voor een eventuele implementatie?
    e. Welke risico's zijn hieraan verbonden?
    f. Herbergt het implementatietraject van open source software een business continuity risico?
    g. In hoeverre is bovenstaande relevant voor de IB-groep?
    h. Wat zijn de relevante aandachtspunten voor de IT-auditor ten aanzien van open source software, in de uitvoering van diens werkzaamheden?

2. *Welke maatregelen moeten minimaal geïmplementeerd worden om deze risico's te minimaliseren?*
    a. Welke impact/gevolgen hebben de geïdentificeerde risico's op de situatie van de IB-Groep?
    b. Welke mogelijke oplossingsrichtingen zijn aanwezig?
    c. Welke randvoorwaarden voor implementatie van open source software vloeien hieruit voort?
    d. Voldoet de IB-Groep aan de gestelde randvoorwaarden?
    e. Welke stappen moet de IB-Groep nemen om aan de gestelde randvoorwaarden te voldoen?
    f. Wat zijn de relevante aandachtspunten voor de IT-auditor ten aanzien van geïdentificeerde risico's, in de uitvoering van diens werkzaamheden?

3.5. Methodiek

Om de gestelde doelstellingen te halen en de vragen te beantwoorden, wordt gebruik gemaakt van de volgende methodes: literatuurstudie en interviews. De gekozen methodiek heeft in de praktijk geleid tot een interessant fenomeen: de beschikbare literatuur op internet is grotendeels onbetrouwbaar. Een zoektocht naar objectieve wetenschappelijke artikelen verliep stroef, maar heeft uiteindelijk een aantal betrouwbare bronnen opgeleverd. Naast de beschikbare wetenschappelijke bronnen heb ik eveneens gebruik gemaakt van interne IB-Groep documenten. In aanvulling op interne IB-Groep documenten zijn ook documenten van andere overheidsorganisaties onderzocht.

In tegenstelling tot de literatuur bleken de geïnterviewden veelal een genuanceerdere mening over open source software en het gebruik daarvan te hebben. De geïnterviewden zijn geselecteerd op basis van hun kennis en ervaring met het onderwerp. De lijst met geïnterviewden vindt u in bijlage 3.

3.6. Keuze voor te hanteren referentiemodel

Voor het schrijven van deze scriptie acht ik het van belang om te redeneren vanuit een referentiemodel. Ik gebruik binnen deze scriptie de best practices die door de IB-Groep worden gebruikt. De IB-Groep heeft expliciet gekozen voor:

- The British Standard 7799 (BS7799) of ISO 17799 (de Code);
- IT Infrastructure Libary (ITIL).

3.7. De code voor informatiebeveiliging

De eisen waaraan software minimaal moet voldoen is gebaseerd op de code voor informatiebeveiliging [NEN ISO 19977]. Dit document is sinds december 2003 per directiebesluit het geldende referentiemodel binnen de IB-Groep. De code [2000] schrijft in abstracto best practices voor aan de hand waarvan de organisatie haar informatie-beveiligingsbeleid kan opstellen.

De code [2000] is niet geschreven vanuit het perspectief open source of closed source. De code is op een dusdanig abstract niveau dat dit onderscheid ook niet relevant is. Op dit moment is er geen enkele reden om aan te nemen dat er specifieke aandachtspunten voorvloeien uit de code [2000] ten aanzien van software eisen.

3.8. Randvoorwaarden

De VU stelt als eis dat de scriptie een toegevoegde waarde heeft voor het vakgebied IT-Audit. Beantwoorden van de twee onderzoeksvragen dient vanuit een auditperspectief te gebeuren. Daarnaast dient de scriptie tot stand te zijn gekomen op basis van een wetenschappelijk onderbouwde methodiek. De toegevoegde waarde voor het vakgebied moet voortvloeien uit de beantwoording van vraag 2. In de conclusies worden risicoverminderende criteria en

maatregelen genoemd die kunnen fungeren als input voor een normenkader van een IT-Audit naar een dergelijk implementatietraject.

De toegevoegde waarde voor de IB-Groep vloeit voort uit de beantwoording van de vragen 1 en 2. De antwoorden en conclusies kunnen dienen als input, indien de IB-Groep voor een dergelijke keuze komt te staan. De interne audit afdeling van de IB-Groep is voor- noch tegenstander van een dergelijke implementatie. De interne audit-afdeling (CT/Audit) heeft als standpunt dat een dergelijke beslissing moet worden genomen op basis van de juiste, intern geldende, bedrijfseconomische principes, onder de voorwaarde dat de continuïteit van de IB-Groep niet in gevaar mag worden gebracht en dat de uitvoering van haar taken conform gestelde wet- en regelgeving geschiedt. CT/Audit acht het van belang dat indien de IB-Groep ooit voor een dergelijke beslissing komt te staan, zij in het vroegst mogelijke stadium de besluitvorming kunnen ondersteunen op basis van objectieve criteria.

Expliciet uitgesloten van onderzoek is een waardeoordeel over de toegevoegde waarde van welk open souce software pakket dan ook. Er wordt geen principiële uitspraak gedaan over een eventuele voorkeur voor open source software. Eveneens is uitgesloten van onderzoek een specifieke kosten/batenanalyse. De kosten en baten worden behandeld op een abstract niveau. De scriptie heeft uitsluitend als doel antwoord te geven op de gegeven vragen. De uitkomsten van deze scriptie kunnen als input dienen. Dan zal echter een vertaalslag gemaakt moeten worden naar de financiële consequenties.

Hoofdstuk 4. Open source software

4.1 Inleiding

Dit hoofdstuk geeft een beschrijving van open source software en de verschillen met traditionele closed source software. Dit hoofdstuk bevat de volgende paragrafen:

- Beschrijving van open source software;
- Kenmerken van open source software;
- Het onderscheid tussen closed en open source software.

4.2. Beschrijving van open source software

Open source software is niet een recente ontwikkeling [Weber 2000]. Sinds de begindagen van de IT bestaat open source software al. De eerste grote spelers verkochten hardware en leverden de software daarbij, inclusief de broncode. Bijzonder hieraan is dat de software nog wel onder het regime van een traditionele licentie viel. Open source software is historisch gezien onlosmakelijk verbonden met UNIX. UNIX is een operating system dat eind jaren ’60 van de twintigste eeuw ontwikkeld is door Thompson en Ritchie [Raymond 2002, Willis 1993]. De populariteit van UNIX is te danken aan haar stabiliteit en flexibiliteit. (Collegesheets VU 2005, 2006, 2007). De behoefte om dergelijke stabiliteit en flexibiliteit ook in een PC-omgeving te realiseren heeft geleidt tot de ontwikkeling van alternatieve operating systems, waarvan het GNU project (opgericht in 1984) de oudste is. Een aantal belangrijke varianten van UNIX zijn:

- Linux;
- BSD;
- Solaris;
- Mac OS.

Het is met name Linux dat de afgelopen jaren veel aandacht heeft gekregen. Dit operating system is de laatste jaren in populariteit toegenomen. De gebruikers roemen de stabiliteit en flexibiliteit van Linux. Open source software kan niet beschreven worden zonder Linus Torvalds te noemen. Hij heeft een kernel ontwikkeld die heeft bijgedragen tot de ontwikkeling van stabiele operating systems. Op basis van dit operating system zijn later “Windows-achtige” grafische omgevingen ontwikkeld. Belangrijke varianten van Linux zijn:

- Suse;
- Free BSD;
- Ubuntu;
- KDE;
- Redhat.

Veelal wordt als belangrijkste kenmerk van open source software genoemd dat de broncode meegeleverd wordt en aangepast mag worden. Dit heeft tot gevolg gehad dat er vele verschillende pakketten zijn ontwikkeld, die zich niet uitsluitend tot het operating system beperken. In de loop der jaren zijn meerdere open source applicaties ontwikkeld. Voorbeelden hiervan zijn:

- Open office, een gratis officepakket;
- Firefox, een gratis internetbrowser;
- Thunderbird, een gratis e-mail client.

Hoewel de openbaarheid van de broncode een van de belangrijkste kenmerken is, is dit niet de enige voorwaarde. In de volgende paragraaf worden de eigenschappen van open source software beschreven.

4.3. Kenmerken van open source software

Opensource.org hanteert tien criteria waaraan software moet voldoen voordat deze als open source kan worden aangemerkt. Dit zijn:

1. Free distribution;
2. Source code;
3. Derived works;
4. Integrity of the author's source code;
5. No discrimination against persons or groups;
6. No discrimination against field of endeavor;
7. Distribution of license;
8. License must not be specific to a product;
9. License must not restrict other software;
10. License must be technology-neutral.

De Vereniging voor Open Source Nederland[1] geeft op haar site een korte opsomming van wat de bovenstaande punten in de praktijk inhouden:

> "Open source betekent niet alleen toegang tot de broncode. De voorwaarden voor verspreiding van een open-source programma moeten in overeenstemming met de volgende criteria zijn:
>
> 1. **Vrije verspreiding**
>   De licentie mag geen enkele partij verhinderen om de software te verkopen of weg te geven als onderdeel van een collectie software met programma's van verschillende bronnen. De licentie mag geen royalties of andere vergoeding voor een dergelijke verkoop eisen.
> 2. **Broncode**
>   Het programma moet de broncode bevatten, en moet verspreiding toestaan, zowel als broncode als in gecompileerde vorm. Als het produkt zonder broncode verspreid wordt, moet er een goed gedocumenteerde manier zijn om de broncode, zonder kosten, te downloaden van het Internet. De broncode moet beschikbaar zijn om een programmeur in de gelegenheid te stellen het programma aan te passen. Met opzet zeer verwarrende broncode schrijven is niet toegestaan. Tussenvormen, zoals de output van een preprocessor of translator zijn niet toegestaan.
> 3. **Afgeleide programma's**
>   De licentie moet aanpassingen en van het originele programma afgeleide werken toestaan, en deze moeten onder dezelfde voorwaarden verspreid kunnen worden als het originele programma.
> 4. **Integriteit van de originele broncode**
>   De licentie mag verspreiding van aangepaste broncode alleen verbieden als de licentie wél de verspreiding van zogenaamde "patch bestanden" bij de originele broncode toestaat, met het doel het programma aan te passen tijdens het "builden" ervan. De licentie moet expliciet toestaan dat software die gebaseerd is op gewijzigde broncode verspreid wordt. De licentie mag vereisen dat afgeleide programma's een andere naam of versienummer hebben dan de originele software.
> 5. **Geen discriminatie van personen of groepen**
>   De licentie mag niet discrimineren tegen welke persoon of groep personen dan ook.
> 6. **Geen discriminatie tegen toepassingsgebieden**
>   De licentie mag niemand verhinderen om het programma te gebruiken in een bepaald toepassingsgebied. Het mag bijvoorbeeld niemand verhinderen om het programma in een bedrijf, of voor genetisch onderzoek, te gebruiken.
> 7. **Verspreiding van de licentie**
>   De rechten die bij het programma horen moeten ook van toepassing zijn op

iedereen naar wie het programma is verspreid, zonder dat voor deze partijen een extra licentie noodzakelijk is.

8. **De licentie mag niet specifiek voor één product gelden**
   De rechten die bij het programma horen mogen niet afhankelijk zijn van het feit dat het programma deel uitmaakt van een bepaalde software-distributie. Als het programma uit deze distributie wordt gehaald en gebruikt of verspreid volgens de voorwaarden van de licentie van het programma, dan hebben alle partijen naar wie het programma is herverspreid dezelfde rechten die ook van toepassing waren op de originele distributie.
9. **De licentie mag andere software niet beïnvloeden**
   De licentie mag geen beperkingen stellen aan andere software die is verspreid samen met het betreffende programma. De licentie mag bijvoorbeeld niet vereisen dat alle andere software die met het programma wordt meegeleverd ook open-source software is.
10. **Conformerende licenties en certificatie**
    Alle software die een licentie gebruikt die gecertificeerd is als zijnde conform de Open Source Definitie mag het Open Source handelsmerk gebruiken, evenals broncode die expliciet in het zogenaamde 'public domain' wordt geplaatst. Geen enkele andere licentie of software is gecertificeerd om het Open Source handelsmerk te gebruiken." [1]

Uit de beschikbare teksten blijkt dat de vereniging deze kenmerken dusdanig interpreteert dat een afwijking hierop niet mogelijk is. Zij beschouwt software uitsluitend als open source, indien deze aan alle bovengenoemde kenmerken voldoet. In geen van de onderzochte teksten is gebleken dat de wijze waarop gedistribueerd wordt relevant is. Veelal is de software verkrijgbaar via downloads. Daarnaast bestaan echter ook mogelijkheden om de software op dvd of cd-rom te verkrijgen.

Hierbij dient een kritische noot geplaatst te worden. De genoemde vereniging is voorstander van open source software, ook voor overheidsorganisaties. Alle verkregen informatie uit deze hoek dient vanuit dit perspectief te worden bezien.

Kenmerkend voor de ontwikkeling van open source software is de community, oftewel de gemeenschap. De gemeenschap ontwikkelt als een collectief onder het principe van Linus' law: '*Many eyes make errors trivial*'. Het gevolg hiervan is dat open source software in principe niet vanuit een vastomlijnd kader wordt ontwikkeld. De ontwikkeling is in hoge mate afhankelijk van zelfregulering. Frappant is dat ondanks de hoge mate van consensus over de voordelen van open source software in de gemeenschap nog geen uniforme theorie over de ontwikkeling van dergelijke software is.

Bedrijfseconomisch is weinig fundamenteel onderzoek gedaan naar open source software. Veelal blijkt dat de zogenaamde "kostenvoordelen" die behaald kunnen worden een aspect is dat de voorstanders hanteren. Microsoft [2005] daarentegen heeft onderzoeken gefinancierd waarin wordt aangetoond dat closed source software goedkoper zou zijn. Wheeler [2005] wijst erop dat veruit het meeste onderzoek naar het gebruik en nut van open source software door de traditionele softwaremaatschappijen is betaald. Hij wijst eveneens op het feit dat sommige leveranciers het afnemers niet toestaan (vanuit de licentie) om kritische opmerkingen te maken over de geleverde software. Gezien het feit dat beide partijen overduidelijk hun eigen belang hebben, stel ik dat vooralsnog geen daadwerkelijke uitspraak gedaan kan worden over het bestaan van kostenvoordelen. Dit betekent niet dat kostenvoordelen uitgesloten zijn, maar deze uiten zich primair in specifieke situaties. Met andere woorden: het is afhankelijk van de specifieke situatie in een organisatie of een keuze voor open source software voordelig is.

[1] Citaat van: http://www.vosn.nl/index.php?sectie=default&groep=open+source

Open source software dient niet verward te worden met Free software en Open standaarden. De termen zelf zijn verder niet relevant voor de rest van de scriptie, dientengevolge wordt hieraan verder geen aandacht besteedt.

4.4. Het onderscheid tussen closed en open source software

De term closed source software is bedacht door de aanhangers van de open source beweging. Een minder gangbare, maar accurate naam is proprietary software. In het kader van deze scriptie wordt gerefereerd aan de meer gebruikelijk naam: closed source software. De reden hiervoor is dat deze term meer gebruikelijk is en dat zij een tegenstelling benadrukt. Closed source software is het tegenovergestelde van open source software [Wheeler 2005, Varian & Shapiro 2005, Raymond 2001, Weber 2000]. Tot aan de opkomst van de open source beweging was er feitelijk geen aandacht voor het onderscheid tussen closed en open source software. De term is bedacht door Eric Raymond in 1998. Varian & Shapiro [2005] zeggen hierover: "*We distinguish between "open" and "proprietary" interfaces. An interface that is controlled by a single group and not available for everyone to use freely is called a proprietary interface*." In deze scriptie is gekozen voor een zwart-wit tegenstelling.

Vanuit deze visie valt onder de definitie alle software die niet aan alle gestelde eisen voldoet zoals benoemd in paragraaf 4.3. In deze paragraaf zijn tien kenmerken genoemd van open source software. In het kader van deze scriptie wordt gesproken over closed source software zodra aan een van de genoemde tien eigenschappen niet is voldaan.

Hierbij moet worden opgemerkt dat een kenmerkende eigenschap van closed source software is dat de broncode als (bedrijfs)geheim wordt beschouwd. Volgens de redenatie dat de broncode de essentie is waarin de commerciële activiteit haar basis vindt, dient de broncode geheim te blijven. De redenatie is dat een bedrijf niet hetgeen vrijelijk deelt met mogelijke concurrenten, waarmee zij haar geld verdient. Daarnaast wijzen de eigenaren op een beveiligingsaspect; de openbaarheid van de broncode zou risico's met zich meebrengen.

In het vorige hoofdstuk is de Code voor informatiebeveiliging [2000] aangehaald als één van de relevante referentiemodellen. De code staat in principe neutraal tegenover closed en open source software. Hieronder wordt voor de code aangegeven wat de specifieke elementen zijn die van belang zijn voor een implementatie van open source software. De relevante hoofdstukken uit de code zijn:

- Beveiligingsbeleid;
- Beveiligingsorganisatie;
- Classificatie en beheer van bedrijfsmiddelen;
- Beveiligingseisen ten aanzien van personeel;
- Fysieke beveiliging en beveiliging van de omgeving;
- Beheer van communicatie- en bedieningsprocessen;
- Toegangsbeveiliging;
- Ontwikkeling en onderhoud van systemen;
- Continuïteitsmanagement.

De code is niet geschreven vanuit een visie op closed of open source software. De in de code genoemde maatregelen zijn op een abstract niveau. Deze maatregelen zijn organisatiespecifiek en toepasbaar voor iedere organisatie. Het gebruik van open source software is in die zin niet risicovoller dan closed source. Hierbij is het van belang dat de gebruikte software qua functionaliteiten aansluit bij de organisatiedoelstellingen. De maatregelen vanuit de code dienen hierbij geïmplementeerd te worden, indien deze van toepassing zijn voor de betreffende organisatie. Een keuze is hierbij primair afhankelijk van de degelijkheid en functionaliteit van het te kiezen pakket.

In bijlage 5 vindt u een tabel waarbij is aangegeven tussen welke hoofdstukken uit de code en de eigenschappen van open source software een relatie bestaat die specifieke aandacht behoeft vanuit een beheerskader. Deze punten worden in het volgende hoofdstuk nader uitgewerkt vanuit een ITIL perspectief. De tabel is weergegeven op hoofdlijnen. Per

intersectie is geen verklaring gegeven, alleen dat er een relatie is die aandacht verdient vanuit een business continuity perspectief. Vooruitlopend op hoofdstuk 5 wordt business continuity gedefinieerd als:

> *Een proces dat bestaat uit calamiteitenbeheersing en risicobeheersing met als doel het bereiken van bedrijfscontinuïteit.*

In hoofdstuk 5 wordt dieper ingegaan op business continuity vanuit ITIL. Vanuit business continuity wordt als primair risico genoemd de openbaarheid van de broncode. “Security by obscurity” is het devies bij closed source software. De tegenstanders van open source software hanteren als primair kritisch beveiligingspunt dat door de openbaarheid van de broncode kritische kwetsbaarheden ter beschikking worden gesteld aan eventuele kwaadwillende hackers. De conclusie is veelal dat de continuïteit van een organisatie bij gebruik van open source software gevaar loopt. Vanuit de code is geen aanwijzing gevonden dat het geheim houden van de broncode essentieel is voor de continuïteit van een organisatie. Volgens Stamp [2006] is “Security by obscurity a flawed concept.“ Deze visie wordt eveneens gehanteerd in de moderne versleutelingstechnieken (encryptie). Hierbij is het uitgangspunt dat het algoritme openbaar is, maar dat de sleutel geheim gehouden moet worden. De voorstanders van open source software stellen dat de openbaarheid van de broncode een sterk punt is vanuit beveiligingsoogpunt. Zij verwijzen naar Linus’ Law, die stelt dat: ‘Many eyes make errors trivial’. Door de broncode openbaar te maken, wordt direct de gelegenheid geschapen om zo snel mogelijk fouten te ontdekken. Het recht om de code daarna aan te passen zou het voordeel bieden dat de zwakheden snel en adequaat opgelost worden. In deze scriptie doe ik geen uitspraak over de juistheid van Linus’ Law.

Geredeneerd vanuit de relevante hoofdstukken zijn wel open source specifieke maatregelen nodig. Met andere woorden: de best practices vanuit de Code moeten vertaald worden naar de organisatiespecifieke situatie. Vanuit de Code zijn echter geen bezwaren gevonden die de tien genoemde eigenschappen van open source software uitsluiten als zinnig alternatief voor closed source software. Met name de openbaarheid van de broncode, de juridische implicaties en software specifieke functionaliteiten zijn van invloed op de inrichting van de beheerorganisatie. Deze relatie wordt in hoofdstuk 5 nader uitgewerkt.

Hoofdstuk 5. De relatie tussen open source software en ITIL

5.1. Inleiding

In dit hoofdstuk wordt een relatie gelegd tussen ITIL en open source software. Het beantwoordt de vraag aan welke eisen open source software moet voldoen vanuit het beheersmodel ITIL. Het doel van dit hoofdstuk is om aan te geven wat de relatie is tussen open source software en de relevante ITIL processen. ITIL is een set best practices voor het beheer van een IT-organisatie. In principe behoort dit onafhankelijk te zijn van het type software dat gebruikt wordt. In deze scriptie is gekozen om per relevant proces een toelichting te geven waarom het proces relevant is vanuit een open source perspectief. Dit hoofdstuk bevat de volgende paragrafen:

- Beschrijving van business continuity;
- ITIL processen en open source software;
- Conclusie.

5.2. Beschrijving van business continuity

Voor deze scriptie is gekozen voor de ITIL definitie van Business Continuity. De reden hiervoor is dat de IB-Groep zich heeft gekozen voor deze best practices. Dit is vastgelegd in het directiebesluit van juli 2004. Daarnaast hanteert de IB-Groep de Code voor Informatiebeveiliging als leidraad.

Van Bon (2004) definieert Business Continuity Management vanuit ITIL als:

> *Een proces dat bestaat uit calamiteitenbeheersing en risicobeheersing met als doel het bereiken van bedrijfscontinuïteit.*

De Code voor informatiebeveiliging [2000] benoemt als doelstelling:

> *Het reageren op verstoringen van bedrijfsactiviteiten en het beschermen van kritieke bedrijfsprocessen tegen de effecten van grootschalige storingen of calamiteiten*

Van Bon [2004] hanteert voor Calamiteit de volgende definitie:

> *Een gebeurtenis die een service of systeem zodanig verstoort dat veelal aanzienlijke maatregelen moeten worden genomen om het originele werkingsniveau te herstellen.*

Vanuit de ITIL-methodiek is tevens aandacht voor IT Service Continuity Management. Dit definieert van Bon (2004) als volgt:

> *Het proces dat nodig is om binnen de IT dienstverlening calamiteiten op te vangen en te overleven teneinde de business te kunnen continueren.*

Business continuity management en IT service continuity management zijn aan elkaar gerelateerd. Naarmate de IT een grotere rol speelt in een organisatie zijn beide processen meer gelijk aan elkaar. De IB-Groep is in hoge mate afhankelijk van haar ICT-voorzieningen, zowel voor de primaire als de ondersteunende processen. (Zie hoofdstuk 7). Gezien dit belang is gekozen om geen onderscheid te maken tussen business continuity management en IT service continuity management. Dit is in overeenstemming met algemeen geldende ITIL uitgangspunten.

Centraal in het proces is de risicobeheersing. De risicobeheersing dient voort te vloeien uit een gefundeerde risico-analyse. Na het uitvoeren van de risico-analyse wordt aan de hand van deze analyse een continuïteitsplan samengesteld.

Vanuit de code (zie bijlage 5) is gebleken dat voor open source software bijna alle relevante hoofdstukken van invloed zijn op de inrichting van de beheerorganisatie. Zij noemt geen enkel principieel bezwaar tegen het gebruik van open source software. Het gebruik van open source software vergt vanuit de beheerorganisatie specifieke aandacht.

5.3. ITIL processen en open source software

Vanuit de literatuur wordt aangegeven dat de onderstaande vijf processen het meest relevant zijn vanwege hun relatie met continuity management. [van Bon 2004].

1. *Service level management;*
2. *Availability management;*
3. *Configuration management;*
4. *Capacity management*;
5. *Change management.*

In de bovenstaande rij zijn security management, application management en release management niet opgenomen. Dit is in mijn optiek onjuist. De reden om deze drie processen toe te voegen zijn weergegeven in de onderstaande beschrijvingen.

5.3.1. Security management

Security management heeft een tweeledig doel. Ten eerste dient voldaan te worden aan vastgestelde beveiligingseisen. Ten tweede dient een bij de organisatie passend niveau van beveiliging gerealiseerd te worden. De discussie gaat tussen closed en open source software. Closed source software hanteert het principe van “security by obscurity”. Terwijl de open source software benadrukt dat openheid de beste manier is om de veiligheid van software te garanderen. Hierbij baseert de gemeenschap zich op Linus’ law. Deze zienswijze vergt een andere manier van het inrichten van het process security management. Indien het proces niet goed ingericht wordt, zie ik hier een potentieel risico. Hier staat tegenover dat in Microsoft programmatuur regelmatig zwakke punten worden ontdekt, ondanks het feit dat de broncode geheim gehouden wordt. De potentiële zwakheid die uniek is voor open source software draagt in mijn visie bij tot de noodzaak om het ITIL-proces Security management toe te voegen. Security management dient specifieke maatregelen te treffen bij het gebruik van open source software in een organisatie. Dit vindt zijn oorzaak in de openbaarheid van de broncode.

5.3.2. Application management

Application management is het beheren van applicaties als ‘corporate assets’, om zodoende met de informatiesystemen van een organisatie flexibel in te kunnen spelen op veranderingen in de markt. [van Bon, 2004]. Open source applicaties kunnen in deze visie opgevat worden als ‘corporate assets’ die om specifieke maatregelen vragen. De rol die application management speelt om de gewenste functionaliteiten te garanderen draagt in mijn visie bij om dit ITIL proces toe te voegen in de analyse.

5.3.3. Release management

Volgens van Bon [2004] richt release management *… zich op het waarborgen van de kwaliteit van de productieomgeving, door gebruik te maken van formele procedures en controles bij het implementeren van nieuwe versies.* In tegenstelling tot change management is release management uitvoeringsgericht. De onderzochte literatuur wijst erop dat release management met name bij grote projecten onderdeel dient te zijn van het projectplan. De implementatie van open source software zal vermoedelijk gebeuren door middel van een project. De specifieke aandachtspunten voor open source software zijn de openbaarheid van de broncode en de gewenste functionaliteiten van de software. Het belang van dit proces is reden om het proces mee te nemen in de analyse.

5.3.4. Service Level management

Volgens Van Bon [2004] wordt Service level management gedefinieerd als: het proces van het onderhandelen, definiëren, meten, beheersen en verbeteren van de kwaliteit van de IT-dienstverlening tegen gerechtvaardigde kosten. Dit proces bestaat uit de volgende activiteiten:

- verzorgt de integratie tussen de afzonderlijke elementen waaruit de dienstverlening bestaat;
- documenteert de dienstverlening door elementen op ordelijke wijze te rangschikken en te beschrijven in relevante documenten;
- beschrijft de aangeboden diensten in termen die de klant begrijpt;

Vanuit een dergelijk perspectief heeft open source software invloed. Het wijzigen van software heeft direct invloed op het serviceniveau indien de wijziging niet succesvol is. De rol van service level management is mijns inziens beperkt tot voor en tijdens de implementatie van open source software.

Het zijn met name de functionaliteiten van de software die bepalen in welke mate de organisatie het afgesproken Service level kan waarmaken. Dientengevolge is het van belang dat de organisatie in staat is om te bepalen welke invloed de functionaliteiten van een open source software pakket hebben op het serviceniveau van de organisatie. Dit geldt eveneens voor de overige pakketten.

Daarnaast is het van belang dat de organisatie inzicht heeft in de mogelijke juridische consequenties van een open source software pakket. In hoofdstuk 4 is beschreven dat één van de eigenschappen van open source software is dat de licentie niet toestaat dat overige software (closed source software) nadelig beïnvloed mag worden door het gebruik. Een dergelijk aspect is niet te allen tijde gegarandeerd door de leveranciers van closed source software. De meeste pakketten voor organisaties zijn geschreven voor een Microsoft omgeving. Voor een implementatie zal duidelijk moeten zijn welke pakketten (configuration management) er zijn en welke juridische en functionele eisen gesteld worden.

5.3.5. Availability management

Het doel van availability management is te zorgen voor een kosteneffectief en vastgesteld niveau van beschikbaarheid van de IT-dienstverlening waarmee de business in staat wordt gesteld om haar doelstellingen te realiseren. [Van Bon 2004]. Dit houdt in dat de verantwoordelijke de activiteiten moet doen die noodzakelijk zijn om het afgesproken niveau te garanderen. Zonder de juiste mate van beschikbaarheid is een transitie niet bespreekbaar. Het proces heeft een hoge mate van relevantie ten aanzien van welke vorm van software dan ook. Bij een transitie van closed naar open source software dient de beschikbaarheid te zijn gegarandeerd. Dit heeft niet zo zeer te maken met het betreffende pakket als wel met de onderliggende processen, afspraken en benodigde functionaliteiten. Vanuit een open source perspectief is dit een belangrijk proces. Afgezien van “beschikbaarheidsclaims” van voorstanders van open source software heb ik geen aanwijzingen gevonden dat open source per definitie een beter resultaat biedt. Met name vanuit de gewenste functionaliteiten moet hiermee rekening gehouden worden in dit proces.

5.3.6. Configuration management

Volgens Van Bon [2004] heeft Configuration management het volgende doel: het helpen bewaken van de economische waarde van de IT-dienstverlening – een samenspel van klantwensen, kwaliteit en kosten – door een logisch model van IT-infrastructuur en IT-diensten te onderhouden en daarover informatie te verschaffen aan andere bedrijfsprocessen.

Uit het doel van dit ITIL proces blijkt dat dit een sleutelrol vervult in een eventuele transitie naar open source software. Naarmate de complexiteit van de configuratie toeneemt, nemen de afhankelijkheden ook toe. Binnen de IB-Groep worden meer dan driehonderd softwarepakketten gebruikt. In het kader van een transitie dient inzichtelijk te zijn wat de consequenties van open source software zijn. Configuration management dient hiervoor alle relevante informatie ter beschikking te stellen aan de betrokkenen. Zonder deze informatie is een succesvolle implementatie zeer onwaarschijnlijk. Daarnaast dient Configuration management informatie te verschaffen over onderlinge afhankelijkheden van gebruikte pakketten. Een groot deel van de pakketten is geschreven voor een Microsoft omgeving. Het moet vooraf inzichtelijk zijn welke pakketten dit zijn en hoe daarop geacteerd dient te worden.

5.3.7. Capacity management

Volgens Van Bon [2004] heeft capacity management tot doel om voortdurend en tijdig de juiste capaciteit aan IT-middelen beschikbaar te stellen tegen te verantwoorden kosten en passend bij de huidige en toekomstige behoeften van de klant. Tijdens de implementatie moet voldoende capaciteit aanwezig zijn om een transitie te ondersteunen. Het moet duidelijk zijn in welke mate de gewenste functionaliteiten van invloed zijn op de capaciteit.

5.3.8. Change management

Volgens Van Bon [2004] heeft change management tot doel om zeker te stellen dat gestandaardiseerde methoden en procedures gebruikt worden, zodat changes conform afspraken kunnen worden afgehandeld. Tijdens de transitie van closed naar open source software is het van groot belang dat de gemaakte afspraken nagekomen worden. Met name de afspraken met betrekking tot de projectimplementatiemethodiek zijn essentieel. De reden hiervoor is dat het ontbreken van een goed functionerende projectimplementatiemethodiek een primaire oorzaak is van het falen van IT-projecten. In hoofdstuk 6 wordt een nadere relatie gelegd tussen business continuity en falende projectimplementaties.

Na de implementatie behoudt change management haar belangrijke rol. De reden hiervoor ligt in de specifieke eigenschappen van open source software. De openbaarheid van de code stelt de organisatie in staat om haar eigen aanpassingen te doen op de software. Juridisch is dit geen probleem, de licentie van open source software staat dit toe. Eventueel gewenste, maar ontbrekende functionaliteiten kunnen zelf worden gemaakt en geïntroduceerd. Zonder het afdwingen van de juiste procedure rondom wijzigingen is het gevaar van wildgroei niet ondenkbaar. Interessant hieraan is dat het beperken en beheersen van wijzigingen inherent vreemd is aan het concept open source software.

5.4. Conclusie

ITIL is als set best practises onafhankelijk van het type software dat gebruikt wordt. Echter, een transitie van closed naar open source vergt vanuit business continuity een goed geordende beheerorganisatie. Een dergelijke transitie is een grote ingreep in iedere organisatie. Naarmate de afhankelijkheden groter worden is van groter wordend belang dat de beheerorganisatie op orde is. Voor, tijdens en na de implementatie moeten de processen dusdanig beheerst worden dat de continuïteit gegarandeerd is. Met name de openbaarheid van de broncode, de gewenste functionaliteiten en de juridische aspecten zijn van belang bij het hanteren en een herinrichting van de ITIL processen. Uit de onderstaande tabel blijkt dat de gewenste functionaliteiten de primaire bron van aanpassing zijn.

Vanuit een business continuity perspectief zijn de genoemde acht ITIL processen relevant. Hierin is het onderscheid tussen closed- en open source software niet relevant. De inrichting is organisatiespecifiek en zal anders ingericht moeten worden. Het belangrijkste geïdentificeerde risico is dat de wijzigingen als gevolg van een implementatie niet goed worden doorgevoerd. Een eventueel besluit om over te gaan op open source software dient vanuit de genoemde ITIL-processen ruime aandacht te krijgen, wil zij succesvol kunnen zijn. In bijlage 6 vindt u een tabel die de relatie weergeeft voor elk van de drie benoemde aspecten

per ITIL deelproces. De genoemde impact heeft betrekking op het proces voor en tijdens de implementatie. Na voltooiing is er geen principieel onderscheid tussen closed en open source software vanuit de ITIL processen. De verwachte impact die open source software heeft op de ITIL processen is afhankelijk van relevante raakvlakken met de drie genoemde aspecten:

- Openbaarheid van de broncode;
- De gewenste functionaliteiten;
- De juridische aspecten.

Proces	**Impact**
Security management	Zeer Hoog
Application management	Matig
Release management	Hoog
Service level management	Hoog
Availability management	Matig
Configuration management	Zeer hoog
Capacity management	Matig
Change management	Hoog

Hoofdstuk 6. Open source software in het kader van business continuity

6.1. Inleiding

In dit hoofdstuk wordt antwoord gegeven op de vraag: *Welke risico's zijn te identificeren tijdens een mogelijke overstap op open source software, vanuit het perspectief van business continuity?* In dit hoofdstuk worden de volgende paragrafen behandeld:

- De risico's van open source software
  - Kosten baten analyse
  - Kennis en kunde
  - Beheerorganisatie
  - Projectimplementatie
- Conclusie

In de genoemde paragrafen beschrijf ik kwalitatief waar de primaire risico's liggen tijdens een overstap naar open source software. Het is van belang dat de vier genoemde punten niet als losstaande elementen worden beschouwd. De onderlinge samenhang wordt duidelijk gemaakt aan het einde van dit hoofdstuk en wordt belicht in de conclusie.

6.2. De risico's van open source software

6.2.1. Kosten baten analyse

Kostenreductie is een primaire drijfveer voor een gewenste implementatie [Gemeente Amsterdam 2006, Koenig 2005, Gemeente Groningen 2006, Rijkswaterstaat 2006, Varian & Shapiro 2006, Silver 2005]. Dit is veelal gebaseerd op de foute veronderstelling dat open source software gratis zou zijn. Deze veronderstelling blijkt onder andere uit de documentatie van de Gemeente Amsterdam [2006], Koenig [2005], Gemeente Groningen [2006], Rijkswaterstaat [2006], Varian & Shapiro [2006], Silver [2005].

Het wijdverbreide misverstand dat open source software gratis is, is aan te wijzen als continuïteitsrisico. Open source software en Free software zijn nauw aan elkaar gerelateerd. Het misverstand is ontstaan door het Engelse woord 'free', hetgeen vertaald kan worden met 'gratis'. 'Free' heeft echter betrekking op de licentie. Deze constructie wordt zo genoemd om dat de geldende juridische restricties beperkt zijn. De vrijheid komt voort uit het feit dat de eigenaar gestimuleerd wordt om de software te verspreiden en aan te passen waar de houder dat nodig vindt. De software wordt grotendeels gratis gedistribueerd. Dit neemt niet weg dat er wel degelijk kosten verbonden zijn aan de implementatie en het onderhoud van de nieuwe software. Het correct inschatten van de werkelijke kosten en baten is noodzakelijk om te bepalen of open source software voor de specifieke situatie van de organisatie goedkoper of duurder is.

Bedrijfseconomisch leverde de onderzochte literatuur een interessant beeld op. De kosten worden onderschat en de opbrengsten overschat. Hoewel ik persoonlijk ideeën heb hoe dit foute beeld kan ontstaan, heb ik hiervoor geen cijfermatige onderbouwing gevonden. Ik veronderstel dat de belangen die gepaard gaan met een implementatie van een project, zowel intern als extern, zorgen dat de benodigde feiten voor een juiste afweging van alle mitsen-en-maren niet worden gepresenteerd. Dit draagt bij aan een foutief beeld bij het beslissende orgaan in de desbetreffende organisatie.

Een foutief besluit in een dergelijke organisatie kan continuïteitsproblemen veroorzaken. Langendijk [2006] en van Leeuwen [2006] laten weten dat 70% van de projecten mislukt en dat 14% van de organisaties continuïteitsproblemen krijgt als gevolg van foute implementatie. Het repareren van deze inschattingsfout kost extra middelen. Dit brengt kosten met zich mee. Afhankelijk van de mate van foutief inschatten van de kosten kunnen de reparatiekosten dusdanig toenemen dat de continuïteit van de organisatie in gevaar kan komen.

Varian en Shapiro [2005] noemen het aspect van gebrek aan inzicht in Total cost of ownership (TCO) als primaire oorzaak voor de onderschatting van de werkelijke implementatiekosten. Dit wordt onderschreven door Gartner [2005]. De gemeenten die het convenant [2006] hebben getekend, hanteren het TCO principe.

De TCO methodiek [Gartner 2005] houdt in dat alle kosten (direct en indirect) van de ICT-dienstverlening in kaart moeten worden gebracht. Voor licenties en hardware zijn de inschattingen goed te maken. Deze zogenaamde directe kosten zijn vooraf met redelijke mate van zekerheid te kwantificeren. De indirecte kosten daarentegen zijn lastiger. De personele kosten (zowel intern als extern) en de benodigde communicatie zijn veelal afhankelijk van doorberekeningen van andere afdelingen en het werkelijk gemaakte aantal uren. De grootste problematiek ligt echter in de zogenoemde “hidden costs”. Dit zijn kosten die niet in eerste instantie zijn te kwantificeren, omdat deze afhankelijk zijn van minder voorspelbare elementen, dat wil zeggen de menselijke factor in de vergelijking. Genoemd worden onder andere weerstand tegen verandering, angst over de nieuwe situatie en de oude situatie beschouwen als zijnde ideaal.

Dit houdt in dat niet alleen de kostenbesparing moet worden meegewogen in het besluit maar tevens de implementatiekosten. In de TCO-visie moeten alle kosten en baten afgewogen worden in de keuze. Uit het onderzoek [collegesheets 2005, 2006, 2007, Varian & Shapiro 2005] blijkt dat de werkelijke kosten van implementatie onderschat worden. Daarnaast ontbreekt een methodiek om de opbrengsten goed in kaart te brengen. Dit maakt het doen van een goed gefundeerde kosten baten analyse lastig. Afhankelijk van een mogelijk implementatietraject, zullen de kosten geschat moeten worden qua grootte, strategie en afhankelijkheden. Naar mate een project groter wordt, nemen de hidden costs toe [Langendijk, 2006]. Er bestaat een positieve correlatie tussen de complexiteit van een project en de onzekerheid over de werkelijke in te schatten kosten van hetzelfde project.

De kostenreductie van de open source software wordt beschouwd als de baten. Met name de licentiekosten van closed source software worden als een last beschouwd. [Wheeler 2005, Varian & Shapio [2006], interviews]. Uit de beoordeelde beleidsdocumenten blijkt dat de “hoge” licentiekosten een motivatie vormen voor beleidsmakers om alternatieven te zoeken.

Van de geïnterviewden is één persoon geweest die een daling van de beheerslasten als baat noemde. In de literatuur wordt dit fenomeen slechts zijdelings genoemd. De betrokken geïnterviewde roemde een voormalige werkgever die een Linux distributie hanteerde voor haar interne netwerk. Volgens hem was de organisatie goed in staat om een flexibele en stabiele IT-organisatie te beheren. De betreffende organisatie bleek in staat om met zes beheerders een organisatie met in totaal 2800 werkplekken te beheren.

Daarnaast hebben de juridische aspecten gevolgen voor het kosten baten verhaal. Om de continuïteit van een organisatie te garanderen wordt voor bedrijfskritische software veelal een Escrow overeenkomst gesloten. Escrow is een juridische overeenkomst waarbij een goed, met specifieke waarde (in dit geval de broncode), is ondergebracht bij een derde onafhankelijke partij. Deze zogenoemde Escrow agent wordt toevertrouwd met het waardegoed voor het geval dat een vooraf bepaald fenomeen zich voordoet, bijvoorbeeld faillissement van de leverancier, met als gevolg dat de ondersteuning niet langer gegarandeerd is. De Escrow agent draagt zorg dat het goed na het voordoen van het fenomeen wordt toegekend aan de rechthebbende. Dit houdt in het kader van business continuity bijvoorbeeld in dat een organisatie, na faillissement van een softwareleverancier, de beschikking krijgt over de broncode om zodoende de continuïteit van de organisatie te kunnen garanderen. Met de openbaarheid van de broncode is een Escrow niet langer van toepassing. Dit kan worden gezien als een kostenbesparing. De keerzijde is echter dat de organisatie zich moet realiseren dat het ontbreken van een Escrow andere kosten met zich meebrengt. De kosten hiervan moeten worden gezocht in een aanpassing van de beheerorganisatie en de aanpassing van het niveau van kennis en kunde, met betrekking tot de aanwezigheid van de open source software.

Open source software vergt op organisatorische vlak kosten. Uit hoofdstuk 5 is gebleken dat de beheerorganisatie aangepast moet worden aan de nieuwe situatie. De kosten die gepaard gaan met een reorganisatie beperken zich niet uitsluitend tot procedurele aspecten. Een reorganisatie heeft ook hidden costs. De medewerkers moeten ingelicht worden en overtuigd worden dat een reorganisatie noodzakelijk is.

De laatste kostenpost is waarschijnlijk de belangrijkste. Een overstap vergt kosten vanuit een opleidingstraject. Niet alleen de IT-afdeling zal geschoold moeten worden, ook de eindgebruikers moeten onderwezen worden. Uit de interviews blijkt dat eindgebruikers veelal zeer gehecht zijn aan de systemen die ze kennen. De overstap naar andere applicaties zal begeleid moeten worden. Veranderingen worden gezien als bedreigingen.

6.2.2. Kennis en kunde

Een overstap van closed naar open source software vraagt kennis en kunde, niet in de laatste plaats gedegen technische kennis van de nieuw te implementeren technologie. Open source software wordt gekenmerkt door de noodzaak van een hoog kennisniveau, alvorens een overstap gedaan kan worden. De technologie zal in de meeste gevallen nieuw zijn voor de organisatie, waardoor relevante ervaring ontbreekt. Dit houdt in dat de gehele organisatie opgeleid moet worden. Niet alleen de IT-afdeling, die de implementatie en het onderhoud moet doen, maar ook de eindgebruikers zullen opgeleid moeten worden om succesvol met de nieuwe software om te gaan.

De implementatie moet goed verlopen. Technische fouten tijdens de implementatie komen voort uit gebrekkige kennis en kunde. Deze kunnen nadelige gevolgen hebben voor de continuïteit van de organisatie. Dit houdt in dat de betrokken projectmedewerkers voldoende kennis van het technische product en de projectmethode moeten hebben, wil de implementatie succesvol zijn.

Open source software is in principe afhankelijk van de inzet van zogenoemde communities, oftewel gemeenschappen. Deze gemeenschappen zetten zich min of meer op vrijwillige basis in voor de verdere ontwikkeling van de gekozen software. Dit is voor overheidsorganisaties een problematische situatie. De gemeenschap met betrekking tot overheidssystemen zal zich beperken tot professionals. Daarnaast kenmerkt open source software zich door het feit dat er geen centrale regie is op de ontwikkeling van de software [Varian & Shapiro 2005]. Dit houdt een risico in ten aanzien van business continuity. De ontwikkeling kan abrupt stoppen indien de betrokken gemeenschap dit wenst. Mocht een overheidsorganisatie na het beëindigen van de gemeenschappelijke ontwikkeling doorgaan met de gekozen software, dan zal zij kennis en kunde in huis moeten hebben voor onderhoud en verdere ontwikkeling. De openbaarheid van de broncode biedt hiervoor de mogelijkheid.

Uit de gehouden interviews en de literatuur [Oud, 2006] wordt het ontbreken van de benodigde documentatie als zwak punt genoemd voor open source software. Het ontwikkelmodel van open source software is dusdanig dat het proces niet of nauwelijks gedocumenteerd wordt. Langendijk [2006] noemt het niet juist en volledig vastleggen van alle beslissingen, eigenschappen en bijzonderheden als een oorzaak voor het falen van projecten. Dat is geen specifieke eigenschap voor open source software. Bij standaardpakketten is de documentatie veelal aanwezig, of is een licentie aanwezig waarop beroep gedaan kan worden. Bij maatwerk is het actueel, juist en volledig houden van de documentatie eveneens problematisch. Dit houdt in dat kennis rondom de gekozen software niet altijd is vastgelegd. Wil de organisatie haar kennisniveau op peil houden, dan is het noodzakelijk dat de juiste procedures afgedwongen worden.

Als laatste aspect rondom de kennis en kunde wil ik de juridische aspecten noemen, die voortvloeien uit een eventuele implementatie. De organisatie moet de onduidelijkheden over de juridische consequenties van een eventuele overstap in kaart brengen. Verreweg de meeste van de gebruikte applicaties in een kantoorautomatiseringsomgeving zijn ontwikkeld voor een Microsoft omgeving. Het is onduidelijk in hoeverre de leveranciers nog ondersteuning bieden voor de software indien een overstap gedaan wordt.

6.2.3. Beheerorganisatie

Uit de interviews, de literatuur en de gedane analyses (zie hoofdstuk 5) blijkt dat het inrichten van een goed functionerende beheerorganisatie noodzakelijk is. In hoofdstuk 5 is aan de orde gekomen dat de openbaarheid van de broncode, de functionaliteiten van de software en de juridische gevolgen van belang zijn bij de inrichting.

De grootste kracht van open source software is tegelijk haar zwakste punt, zodra deze in een organisatie wordt geplaatst. Open source software wordt niet ontwikkeld vanuit een vastomlijnd kader. Met name voor professionele organisaties kan dit problemen opleveren. Het is niet helder in welke richting software zich zal ontwikkelen en welke impact dat heeft op de beheersing van de IT-organisatie. Dit geldt ook voor closed source software, zij het in mindere mate. Closed source software wordt ontwikkeld vanuit een commerciële gedachte. De ontwikkelmethode houdt voor open source software feitelijk in dat de gebruikersorganisatie de functionaliteit bepaalt van de te gebruiken software. Het niet geheel denkbeeldige gevaar hiervan is dat functionaliteiten ontwikkeld worden die haaks staan op de wensen van de organisatie. Vanuit een organisatiekundig perspectief bestaat inherent de behoefte om ‘in control’ te zijn.

Het tweede erkende risico is dat met de openbaarheid van de broncode de functiescheiding intern aanwezig moet zijn. In tegenstelling tot closed source software is het relatief eenvoudig om aanpassingen te doen in open source software. In tegenstelling tot closed source software kunnen aanpassingen dan gedaan worden op een fundamenteel punt van de software, namelijk de broncode. De Code [2000] schrijft voor dat de wijzigingen op software beheerst moeten verlopen. Dit heeft tot gevolg dat procedures moeten worden ingericht en afgedwongen. De organisatie moet voorkomen dat niet-geautoriseerden in staat zijn om aspecten van de software aan te passen.

De beschreven change-procedure uit ITIL komt neer op de implementatie van functiescheiding. Interessant is dat deze beheersingsvorm feitelijk strijdig is met het basisconcept van de open source beweging. In een organisatie gaat de filosofie van vrijheid blijheid niet op. Zonder de relevante functiescheiding is de kans op wildgroei aanwezig. Deze wildgroei heeft een aantal gevolgen:

- De organisatie verliest overzicht op de gedane wijzigingen;
- De wijzigingen kunnen gevolgen hebben voor de stabiliteit van de IT-infrastructuur;
- De wijzigingen kunnen gedaan worden met kwaadwillende intentie;
- Er kunnen mogelijk nadelige juridische gevolgen zijn.

Het risico hiervan is dat de organisatie niet zeker weet welke gevolgen een wijziging in de infrastructuur heeft voor de continuïteit van de organisatie. Daar komt bij dat niet alle medewerkers de benodigde kennis en kunde hebben om aanpassingen correct door te voeren. Daarnaast is er een risico dat moedwillig wijzigingen doorgevoerd kunnen worden die nadelige effecten hebben op de organisatie. Hierbij zou gedacht kunnen worden aan wraakacties van ontevreden (ex-)werknemers.

6.2.4. Projectimplementatie

Volgens Langendijk [2006] mislukt 70% van alle IT-projecten en in 14% van de mislukte gevallen zijn de gevolgen dusdanig groot dat de organisatie niet langer kan voortbestaan. De door Langendijk gepresenteerde cijfers komen overeen met eerder onderzoek van Ernst en Young [2005] waaruit blijkt dat 28% van de IT-projecten succesvol wordt afgerond.

Om de overstap naar open source software te maken, dient een project in het leven te worden geroepen. Vanuit de gepresenteerde onderzoeken blijkt dat de slagingskans klein is, ongeveer 30%. Als een organisatie in hogere mate afhankelijk is van haar IT-voorzieningen, zijn de gepresenteerde cijfers reden om maatregelen te nemen. Mijns inziens tonen deze

cijfers aan dat het gevaar toeneemt vanuit een business continuity perspectief. Om dit risico te beperken bestaat de noodzaak om een methodiek voor projectimplementatie goed toe te passen. Een goed functionerende methodologie is geen garantie voor het succesvol afronden van een project. Maar het ontbreken van de methodologie is een verhoogd risico. Het risico neemt toe naarmate de complexiteit van de omgeving en de projectgrootte toenemen.

Zowel de literatuur als de geïnterviewden noemen als risico: de wijze van implementatie. Dit kan op hoofdlijnen op twee manieren. Ten eerst kan het via een Big Bang, waarbij de gehele organisatie in één keer overgaat. De tweede mogelijk is een geleidelijke transitie, waarbij delen van de organisatie afzonderlijk overgaan. Voor beide wijzen van implementatie geldt dat het project gemanaged moet worden vanuit strak omlijnde richtlijnen en heldere doelstellingen.

6.3. Conclusie

Vanuit het perspectief van business continuity zijn vier gebieden aan te wijzen met een verhoogd risico voor de implementatie van open source software. Dit zijn:

- Gebrekkige kosten baten analyse
- Gebrekkige kennis en kunde;
- Niet goed ingerichte beheerorganisatie;
- Niet goed ingerichte methodiek voor de projectimplementatie.

Indien de bovenstaande vier elementen afzonderlijk niet goed zijn ingeschat bestaat een risico voor de continuïteit van de organisatie. Daarnaast moet duidelijk worden gesteld dat de vier elementen niet los van elkaar te zien zijn.

Hoofdstuk 7. De IB-Groep en open source software

7.1. Inleiding

In dit hoofdstuk wordt antwoord gegeven op de onderzoeksvragen:

- Welke risico's zijn voor de IB-Groep te identificeren tijdens een mogelijke overstap op open source software, vanuit het perspectief van business continuity?
- Welke maatregelen moeten minimaal geïmplementeerd worden om deze risico's te minimaliseren?

Dit hoofdstuk bestaat uit de volgende paragrafen:

- Huidige situatie;
- Risico's en impact voor de IB-Groep;
- Mogelijke oplossingsrichtingen;
- Conclusie.

7.2. Huidige situatie

De IB-Groep beschikt over een complexe IT-infrastructuur en een eveneens complexe software omgeving. Zij is in hoge mate afhankelijk van haar ICT-voorzieningen voor de uitvoering van haar taken. De complexiteit van de voorzieningen zijn hieronder weergegeven op hoofdlijnen. De IB-Groep heeft vier verschillende platformen:

- Mainframe;
- Mid-range;
- Webservers;
- PC-netwerk.

Het beheer van de Mainframe is uitbesteed aan Getronics PinkRoccade. Hierop draait het grootste systeem dat de IB-Groep gebruikt: WSF 18+. Dit systeem wordt gebruikt ten behoeve van de Wet op Studiefinanciering. Daarnaast beschikt de IB-Groep over een eigen rekencentrum, dat gebaseerd is op de iSeries van IBM. Op deze machines draaien overige primaire processen zoals ILS, Innen Langlopende Schuld. Voor deze twee platformen is specifieke software ontwikkeld.

De overige twee platformen, de webservers en het PC-netwerk, draaien op Microsoft-software. Op de webservers draait onder andere www.ib-groep.nl en de mijnib-groep applicatie. Met deze laatste applicatie kunnen de klanten van de IB-Groep via internet mutaties doorgeven en overzichten opvragen. Op het PC-netwerk draait de kantoorautomatiseringomgeving (KA-omgeving). Dit is een Microsoft XP omgeving. De KA-omgeving bevat applicaties als Microsoft office suite en meer dan 350 andere applicaties die de uitvoering van de taken van de IB-Groep moeten ondersteunen. De gebruikte pakketten zijn allemaal closed source software, ontwikkeld voor een Microsoft omgeving. De genoemde applicaties zijn primair niet afkomstig van Microsoft, maar van andere leveranciers van closed source software. De licenties zijn hierop afgestemd.

Van gebruik van open source software is op dit moment feitelijk nog geen sprake. De enige uitzondering hierop is de afdeling IT-Beveiliging. Deze afdeling werkt regelmatig met Linux en de daarvoor beschikbare hacktools. Zij heeft hiervoor toestemming van de directie verkregen omdat met behulp van de beschikbare tools zij haar werkzaamheden beter kan uitvoeren.

Volgens de methodiek van Gartner is de IB-Groep een type B organisatie, oftewel een "Risk avoiding follower". Binnen de IB-Groep geldt het directiebesluit dat uitsluitend proven technology gebruikt mag worden. Dit besluit is gebaseerd op het feit dat proven technology

minder risico biedt ten aanzien van de business continuity. Daarnaast is besloten door de directie dat de voorkeur gegeven wordt aan standaardpakketten ten opzichte van zelf bouwen. De IB-Groep heeft in het verleden zelf softwarepakketten gebouwd. De reden hiervoor was dat de in de markt aanwezig standaardpakketten niet voldeden aan de intern gestelde eisen met betrekking tot de gewenste functionaliteiten.

Hieruit volgt dat er binnen de IB-Groep twee platformen zijn waar open source software geïmplementeerd zou kunnen worden. Dit zijn:

1. De webservers;
2. Het PC-netwerk.

De geïnterviewde medewerkers geven allen aan dat er technisch gezien geen enkel bezwaar is tegen een implementatie. Uit de gehouden interviews met de IB-Groep medewerkers blijkt dat er geen feitelijke bezwaren zijn tegen de implementatie van open source software. Tegelijk geven de geïnterviewde interne medewerkers aan dat zij de kans op een implementatie klein achten.

Voor de webservers is open source software aanwezig die voldoet aan de door de directie gestelde criteria. Dit is de zogenoemde LAMP[2] stack. De LAMP stack heeft zich inmiddels bewezen als proven technology. Voor het PC netwerk is dat anders. Een volledige overgang van Microsoft (operating system, office applicatie en overige applicaties) naar een open source distributie strandt op de door de directie gestelde eis, ongeacht de overige technische criteria. Open source distributies als operating systems en de bijbehorende programmatuur zijn veelal nog niet aan te merken als proven technology of worden nog niet als dusdanig gezien. Objectieve onderzoeken over de betrouwbaarheid en bruikbaarheid van open source distributies in vergelijking met bijvoorbeeld een Microsoft distributie heb ik niet gevonden.

Onduidelijk is in hoeverre de overige applicaties, die niet van Microsoft zijn en niet vervangen kunnen worden door een open source variant, zullen reageren in een nieuwe omgeving. Dit is een risico dat onderzocht moet worden. Hiervoor is onder andere de input van configuration management noodzakelijk.

Uit het “Programma technologie” (2006, zie bijlagen 7 en 8), blijkt dat de IB-Groep open source software niet uitsluit. Zij geeft hierbij helder aan dat de continuïteit van de organisatie niet in gevaar mag komen.

7.3. Risico’s en impact voor de IB-Groep

De in het vorige hoofdstuk genoemde risico’s behandel ik hier afzonderlijk, uitgaande van de specifieke IB-Groep situatie. De volgende risicogebieden zijn geïdentificeerd en behandeld:

- Onderschatting van de kosten;
- Kennis en kunde;
- Beheerorganisatie;
- Projectimplementatie.

7.3.1. Onderschatting van de kosten

Aangetoond is dat kosten veelal onderschat worden. Er is geen reden om aan te nemen dat de IB-Groep daarin zal verschillen met andere organisaties. De in het verleden opgelegde bezuinigingen door de opdrachtgever hebben geresulteerd in een organisatie waar de financiering van een project te allen tijde moet worden afgewogen tegen de baten.

[2] LAMP = Linux, Apache, MySQL en PHP.

Als overheidsorganisatie dient de IB-Groep de beschikbare middelen zorgvuldig te gebruiken. Een faillissement is voor overheden niet relevant. De IB-Groep is echter een zelfstandig bestuursorgaan en daarmee een zelfstandig rechtspersoon, die afhankelijk is van haar opdrachtgever voor de financiering. De IB-Groep heeft eigen vermogen en dit betekent dat het in principe mogelijk is dat de continuïteit in gevaar komt indien de organisatie geen juiste inschatting maakt over de te maken kosten en baten.

In hoofdstuk 6 is aangetoond dat projecten veelal mislukken en dat projecten veelal uitlopen, zowel in tijd als geld. Vanuit een continuïteitsperspectief is het van belang dat vooraf helder is welke kosten en baten gemoeid zijn met een implementatie, gedurende het implementatieproject en na de implementatie.

De kosten zijn zowel continu als incidenteel. Met name de projectimplementatie is te kenmerken als een incidentele, doch grote kostenpost. Zonder juiste inschatting heeft dit als risico dat de IB-Groep de werkelijke kosten van een projectimplementatie onderschat. Daarnaast zijn de opleidingskosten aan te wijzen als zijnde incidentele kosten. Deze kosten worden niet uitsluitend voor het IT personeel gemaakt, maar ook voor de gebruikersorganisatie. De weerstand tegen verandering vergt eveneens een financiële injectie. Het aanpassen van de beheerorganisatie dient te gebeuren in projectvorm.

De baten voor de IB-Groep zijn te vinden in de reductie van de licentiekosten en reductie van de beheerorganisatie. Zowel de kosten als de baten moeten in een nader onderzoek gekwantificeerd worden.

7.3.2. Kennis en kunde

Het algemeen geïdentificeerde risico rondom kennis en kunde is ook voor de IB-Groep van toepassing. Vanuit het verleden is binnen de IB-Groep in de genoemde omgevingen kennis en kunde opgebouwd met betrekking tot de aanwezige platformen. Een overstap naar “iets” nieuws vergt inspanning en tijd. Open source software kent een hoog kennisniveau met specifieke methoden en technieken die de organisatie zich eigen moet maken. Binnen de IB-Groep is voor de twee geïdentificeerde platformen, waar open source software mogelijk is, andersoortige kennis aanwezig. In de huidige setting is kennis van Microsoft-omgevingen vereist en aanwezig. De betrokken medewerkers hebben ervaring met en zijn gecertificeerd vanuit Microsoft. Dit houdt in dat de kennis en kunde met betrekking tot open source software niet aanwezig is. Uit de interviews blijkt dat verschillende medewerkers op de IT-afdeling wel ervaringen met open source software hebben, maar dit is voornamelijk vanuit een persoonlijke voorkeur. De kennis voor implementatie en onderhoud in een organisatie is fundamenteel anders en op dit moment niet aanwezig. Daarnaast is er geen inzicht in de mogelijke gevolgen die de introductie van open source software zou hebben op de aanwezige afhankelijkheden van de bestaande applicaties.

Benodigde kennis heeft niet uitsluitend betrekking op de technische aspecten van de betreffende distributie, maar ook op de reeds aanwezige programmatuur. Voor een implementatie dient duidelijk te zijn welke programmatuur aanwezig is. Het dient bekend te zijn in hoeverre aanwezige programmatuur onder een andere distributie zal functioneren. Eveneens dient bekend te zijn hoe de ondersteuning van de leveranciers op de reeds aanwezige programmatuur zal zijn, indien gekozen wordt voor een ander operating system.

7.3.3. Beheerorganisatie

In hoofdstuk 6 is een gebrekkige beheerorganisatie als risico genoemd. De IB-Groep hanteert ITIL. De methodiek heeft zich in de praktijk bewezen. Hierbij is het succes van ITIL afhankelijk van de wijze waarop deze wordt geïmplementeerd [van Bon 2004]. De implementatie van de methodiek staat in principe los van de gebruikte software. Als beheersmethodiek heeft ITIL voldoende basis om ook open source software in organisaties te beheren. De invoering van open source software heeft directe gevolgen voor de inrichting van de beheerorganisatie van

de IB-Groep, in zake de business continuity. De in hoofdstuk 5 gepresenteerde afhankelijkheden en relaties gelden eveneens voor de IB-Groep.

In hoofdstuk 5 is een relatie gelegd tussen enerzijds de eigenschappen openbaarheid van de broncode, de gewenste functionaliteiten en de juridische aspecten en anderzijds de ITIL processen. Vastgesteld is dat business continuity als afzonderlijk proces ingericht moet zijn. Daarnaast is het van belang dat de onderliggende processen eveneens correct zijn ingericht. Van de acht genoemde ondersteunende ITIL-processen zijn service level management, security management, change management en configuration management als belangrijkste aangemerkt.

De IB-Groep heeft op dit moment haar IT service continuity management process niet voldoende op orde. Op dit moment is niet voorzien in een strategisch business continuity plan. De aanwezig continuïteitswaarborgen zijn op operationeel niveau. De elementaire aspecten zoals genoemd in de Code [2000] zijn wel aanwezig. Back-up, uitwijk en beveiliging zijn aanwezig. De overall sturing en organisatiebrede visie hierop ontbreekt vooralsnog. De IB-Groep en haar CSO zien het belang in van een dergelijke visie en de bijbehorende sturing. Dientengevolge wordt prioriteit gegeven aan het opstellen hiervan. In het kader van deze scriptie betekent dit dat de IB-Groep vooralsnog niet klaar is om een overstap te maken.

Het gewenste serviceniveau van de IB-Groep is gerelateerd aan de gebruikte software. De IB-Groep heeft voor intern gebruik een Service Level Agreement. De naleving hiervan moet garanderen dat het gewenste serviceniveau wordt bereikt. Indien gekozen wordt voor een implementatie van open source software, ongeacht de reikwijdte, moet helder zijn welke gevolgen dit heeft voor het gewenste serviceniveau.

De openbaarheid van de broncode wordt gezien als risico. Dit heeft direct invloed op de beveiliging van de gegevens van de IB-Groep. In hoofdstuk 6 is aangetoond dat de beheersprocedures functiescheiding moeten afdwingen. Dit geldt eveneens voor de IB-Groep. De huidige situatie is dusdanig dat functiescheiding aanwezig is en wordt afgedwongen. De openbaarheid van de broncode in combinatie met een nieuwe situatie heeft dezelfde risico's als genoemd in hoofdstuk 6. Om de genoemde risico's te minimaliseren dient de IB-Groep maatregelen te nemen in de beheerorganisatie.

In hoofdstuk 6 is aan bod gekomen dat het sterke punt van open source software ook haar zwakte is: het ontwikkelen vanuit niet vastomlijnde ontwikkelkaders. Het niet ontwikkelen vanuit een gestructureerde omgeving heeft als risico dat de organisatie een wildgroei aan wijzigingen ondergaat. Dit mede dankzij de afhankelijkheid van een goed functionerende beheersmethodiek voor een succesvolle implementatie van open source software. Hiervoor is het van belang dat de IB-Groep haar change management procedures op orde heeft en dat de procedures worden afgedwongen.

Naar schatting heeft de IB-Groep in de kantoorautomatiseringomgeving meer dan 350 afzonderlijke applicaties. Configuration management levert een belangrijk deel van de informatie waarop een besluit tot het overgaan naar open source software gebaseerd moet worden. Deze informatie is vooralsnog niet voorhanden. Niet helder is in welke mate de onderlinge afhankelijkheden van de gebruikte applicaties in kaart zijn gebracht. De meest applicaties zijn uitsluitend geschreven voor een Microsoft omgeving. Onduidelijk is welke applicaties ook onder een open source platform kunnen functioneren, of te vervangen zijn door een open source variant. Het op orde hebben van configuration management is een basisvoorwaarde voor een succesvolle implementatie.

Bovenstaande houdt in dat voor een succesvolle implementatie van open source software de IB-Groep haar ITIL processen moet aanpassen. De ITIL processen an sich zijn geschikt voor zowel de closed als open source software. Met een correcte inrichting is gegarandeerd dat de IB-Groep een transitie goed kan voorbereiden, implementeren en na afronding van het project kan beheren.

7.3.4. Projectimplementatie

In hoofdstuk 6 is aangetoond dat de implementatie van open source software in beginsel een groot IT-project is en dat IT projecten 70% kans hebben om te falen. Als dit gegeven wordt gecombineerd met de hoge mate van afhankelijkheid van IT binnen de IB-Groep is aangetoond dat de business continuity gebaat is bij een snelle en correcte implementatie. In een complexe omgeving is een goed werkende projectimplementatiemethodiek een vereiste.

De IB-Groep heeft een complexe IT infrastructuur. De onderlinge afhankelijkheden van de verschillende platformen en de gebruikte software is op dit moment niet helder in kaart gebracht. Dit vergroot het risico met betrekking tot een succesvolle implementatie. Vanuit configuration management is niet duidelijk gemaakt in welke mate sprake is van onderlinge afhankelijkheden. De relevante risico-analyse hanteert de term: “Spaghetti”. De geïnterviewde interne functionarissen noemen de complexiteit in de kantoorautomatiseringsomgeving als risico voor een eventuele implementatie. De aanwezigheid van de 350 applicaties en onduidelijke onderlinge afhankelijkheden geven aan dat de complexiteit een niet te onderschatten risico is.

De IB-Groep hanteert PRINCE2 als methodiek voor projectimplementaties. De overstap naar een nieuw softwarepakket is altijd een risico. Naar mate de complexiteit van het project toeneemt, is aangetoond dat de kans op falen toeneemt. Afhankelijk van de te kiezen routes voor implementatie (zie paragraaf 7.4) nemen de onderlinge afhankelijkheden per aspect in complexiteit toe.

Dit houdt voor de IB-Groep in dat zij een gedegen situatieanalyse dient te maken, alvorens een besluit te nemen. De input van configuration management is hierbij onmisbaar. Gezien de huidige onbekendheid met de onderlinge afhankelijkheden, zal vooraf aan een open source implementatie een afzonderlijk project uitgevoerd moeten worden om de afhankelijkheden in kaart te brengen. Anders is het bij de webservers van de IB-Groep. De huidige situatie is overzichtelijk en onderlinge afhankelijkheden zijn duidelijk in kaart gebracht.

Daarnaast zijn er in hoofdlijnen twee methoden om over te gaan: een geleidelijke transitie en een Big Bang. Vanuit de literatuur en de interviews concludeer ik dat een Big Bang implementatie een risico an sich is. Bij projecten die gekenmerkt worden door een hoge complexiteit is een geleidelijke transitie aan te bevelen [Onna 2003]. Het voordeel is dat de projectorganisatie dan beter in staat is om de verandering te beheersen. De organisatie is daarbij beter in staat om eventuele hiaten in zowel de implementatie als in afhankelijkheden beter in te schatten en daarop te anticiperen. Bij een Big Bang implementatie dienen vooraf met zekerheid de gevolgen te overzien te zijn. Voor beide wijzen van implementeren dienen voorzorgsmaatregelen geïmplementeerd te zijn, zoals rollback en backups.

7.4. Mogelijke oplossingsstrategieën

Het besluit om over te gaan op open source software dient te gebeuren op basis van een situatieanalyse en kosten-baten analyse. Daarnaast moet de implementatie passen in de visie van de IB-Groep zonder de continuïteit in gevaar te brengen. Dit moet bezien worden vanuit de gewenste functionaliteiten van de benodigde software. Dit hoofdstuk geeft de mogelijkheden voor een eventuele implementatie weer. De mogelijke richtingen voor de implementatie worden hier beschouwd als op zichzelf staande fenomenen. In principe zijn drie oplossingsrichtingen mogelijk. Deze oplossingsrichtingen zijn hieronder in hoofdlijnen weergegeven en uitgewerkt.

1. De nul-optie;
2. Gedeelte implementatie;
3. Volledig implementatie.

7.4.1. De nul-optie

Dit scenario houdt in dat de beschreven situatie gehandhaafd blijft. De webservers en kantoorautomatiseringomgeving worden niet blootgesteld aan open source software. Het voordeel hieraan is dat de IB-Groep geen risico's neemt, met betrekking tot de staande organisatie. Gevolg van dit scenario is dat mogelijke voordelen van open source software niet worden gebruikt.

7.4.2. Gedeeltelijke implementatie

Deze optie bestaat uit meerdere scenario's. Uitgaande van de twee genoemde gebieden in de IB-Groep waar open source software geïmplementeerd kan worden, bestaat de mogelijkheid om deels over te gaan. De mogelijkheden zijn:

- De webserver over laten gaan naar open source en de kantoorautomatisering niet. Bijvoorbeeld door de webservers te voorzien van de zogenoemde LAMP-stack;
- De kantoorautomatisering over laten gaan en de webserver niet. Bijvoorbeeld door het huidige operating system van Microsoft te vervangen door een linux-distributie of door Microsoft office te vervangen door Open Office.

Binnen de implementatiemogelijkheid om over te gaan naar een open source kantoorautomatiseringomgeving bestaan drie afzonderlijke scenario's:

- Het operating system vervangen en de applicaties ongewijzigd laten;
- Het operating system vervangen en de applicaties wijzigen;
- Open source uitsluitend als alternatief voor nieuwe initiatieven meenemen.

Een keuze voor een van deze scenario's moet gedaan worden met inachtneming van de geïdentificeerde risicogebieden.

Nadeel van een gedeeltelijk implementatie is dat onduidelijk is in hoeverre de reeds aanwezige applicaties zullen reageren. Dit nadeel geldt niet voor de introductie van open source software voor de webservers. De situatie op het platform is dusdanig dat een introductie geen continuïteitsproblemen mag opleveren. De technologie in een dergelijke setting heeft zich bewezen. Het enige dat een succesvolle implementatie in de weg staat is de nu nog ontbrekende kennis en kunde rondom de nieuwe technologie. Het verkrijgen van de kennis en de kunde is feitelijk geen probleem. Het besluit om de webservers te wijzigen is daarmee afhankelijk van een factor: de kosten baten analyse.

Het voordeel van een gedeeltelijke implementatie is dat de IB-Groep zichzelf in staat stelt om op gedegen wijze kennis en kunde eigen te maken en het implementatierisico te verminderen. Nadeel is dat een gedeeltelijke implementatie, ongeacht het scenario, de situatie nog complexer zal maken dan deze al is.

7.4.3. Volledige implementatie

Dit scenario houdt in dat zowel de webservers als de kantoorautomatiseringomgeving overgaan en binnen de laatste zowel het OS als alle applicaties overgaan. De IB-Groep is een type B organisatie, oftewel Risk avoiding follower. Dit is in overeenstemming met het directiebesluit dat de IB-Groep uitsluitend proven technologie dient te gebruiken. Dit betekent dat alle in paragraaf 7.3 genoemde knelpunten moeten zijn opgelost om de continuïteit van de organisatie niet in gevaar te brengen. De genoemde hiaten zijn op hoofdlijnen. Voordat de IB-Groep een keuze heeft gemaakt over een implementatie zal nader onderzoek gedaan moeten worden naar deze hiaten. Eveneens dient in dit geval de Big Bang strategie vermeden te worden.

7.5. Conclusie

De genoemde risico's zijn geaggregeerd en niet op zichzelf staand. Tussen de vier genoemde hoofdrisico's bestaat een relatie. De genoemde risico's zijn allen van toepassing op de IB-Groep. De scope van dit onderzoek sluit onderzoek uit naar de wijze waarop de IB-Groep voldoet aan de gestelde eisen. Indien de IB-Groep een keuze voor open source software maakt, dient zij zorg te dragen voor:

- Een gedegen kosten baten analyse op basis van de TCO methodiek;
- Een beheerorganisatie die de transitie kan begeleiden en na de transitie garandeert dat organisatie in control blijft;
- Een goed functionerende methodiek voor de projectimplementatie en maatregelen om deze methodiek af te dwingen;
- Ruim voldoende kennis en kunde van de te implementeren software op alle relevante gremia.

De vraag die op natuurlijke wijze voortvloeit uit dit hoofdstuk is: "Doet de IB-Groep er verstandig aan om open source software te implementeren?". Mijns inziens is het nemen van een stap met dergelijke risico's onverantwoord en zal dit leiden tot een gevaar in de business continuity. Het concept vendor buy in is van toepassing voor de IB-Groep. Tenzij de gestelde randvoorwaarden veranderen, is een volledige implementatie van open source software een onnodig risico voor de continuïteit. De kennis en kunde met betrekking tot open source software, onbekendheid met de implementatie daarvan en onduidelijkheid over de gevolgen op de beheerorganisatie zijn redenen om voorzichtig te zijn. Dit neemt niet weg dat het voor de IB-Groep verstandig kan zijn om voor nieuwe applicaties in de kantoorautomatiseringomgeving het gebruik van open source software mee te nemen als optie.

De webservers zijn op dit moment voorzien van Microsoft systemen. Deze zijn stabiel en voldoen aan de gestelde functionele eisen. Open source in deze omgeving voldoet aan de proven technology eis. De keuze om over te gaan op open source software is daarmee afhankelijk van een kosten baten analyse, een aanpassing van de beheerorganisatie en de juiste implementatie van een projectrealisatiemethodiek.

Hoofdstuk 8. De rol van de IT-auditor

8.1. Inleiding

In dit hoofdstuk komt de rol van de IT-auditor aan bod. Het gebruik van open source software en de implementatie daarvan heeft voor de werkzaamheden van de IT-Auditor de nodige gevolgen.

8.2. Kwaliteitseisen

De NOREA stelt voorwaarden waaraan de IT-Auditor zich dient te conformeren. Sinds 14 juli 2006 is het reglement Gedrags- en Beroepsregels voor Register EDP-auditors (GBRE) vervangen door de Code of Ethics. Dit dient als waarborg voor de te leveren kwaliteit van de individuele IT-Auditor en de beroepsgroep. Samengevat bepalen deze artikelen dat de IT-auditor:

1. Integer en onbevooroordeeld dient te zijn;
2. Niets mag te doen wat de naam en faam van de orde en diens collegae kan schaden (collegialiteitbeginsel);
3. Deskundig moet zijn, met betrekking tot de objecten van onderzoek;
4. Duidelijkheid moet betrachten in de uitoefening van diens professie;
5. Onpartijdig moet zijn;
6. Onafhankelijk moet zijn;
7. Zorg moet dragen voor hoge kwaliteit ten aanzien van diens werkzaamheden;
8. Geheim moet houden wat hem bij de uitoefening van diens professie is toe vertrouwd.

Het auditen van (IT-)object(en) in relatie tot open source software is mijns inziens primair een business issue en niet een technology issue. Dit betekent voor de IT-Auditor dat diens houding en gedrag niet principieel zal wijzigen bij de uitvoering van diens beroep. Vertaald naar de bovenstaande acht eisen betekend dit dat eis 1, 2, 4, 5, 6 en 8 betrekking hebben op de houding en het gedrag van de IT-Auditor en staan daarmee, mijns inziens, volledig los van de technologische component. Eis 3 en 7 hebben betrekking op de inhoudelijke aspecten van de werkzaamheden van de IT-Auditor. In het specifieke geval van open source software dient de IT-Auditor deskundig te zijn. Voordat de IT-Auditor kan komen tot een oordeel of advies moet hij op de hoogte zijn van de voor- en nadelen van en voor de betreffende objecten.

8.3. De aandachtspunten voor de IT-Auditor

Uit de vorige hoofdstukken is naar voren gekomen dat open source vanuit een business continuity perspectief een viertal risicogebieden met zich meebrengt. Te weten:

- Kosten baten analyse;
- Kennis en kunde;
- Beheerorganisatie;
- Projectimplementatie.

De rol die de IT-auditor vervult blijft hetzelfde, de invulling wijzigt. De IT-auditor heeft primair een toetsende rol en secundair een adviserende rol. Dit doet hij op basis van zijn deskundigheid. Dit houdt in dat de IT-Auditor moet weten waarin closed en open source software van elkaar verschillen. Ten aanzien van de wijzigende rol springt de openbaarheid van de broncode meteen in het oog. Uit de vorige hoofdstukken is naar voren gekomen dat dit een beveiligingsrisico kan opleveren indien de beheerorganisatie niet adequaat is ingericht. Dit geeft kwaadwillende een mogelijkheid om rechtstreeks in de programmatuur ongewenste wijzigingen aan te brengen. Het niet op orde zijn van de beheerorganisatie biedt bij het gebruik van closed source software eveneens de gelegenheid om ongewenste handelingen uit te voeren. Het openbaar zijn van de broncode biedt de IT-Auditor de mogelijkheid om de broncode van de software te beoordelen. Dit onder de randvoorwaarde dat de betrokken IT-Auditor over voldoende kennis en kunde beschikt om de code te beoordelen.

Bijlagen

Bijlage 1. Literatuurlijst

- Baarsma, B et al – *Kosten en baten van open standaarden en open source software in de Nederlandse publieke sector – een analyse op meso- en macro niveau,* SEO Amsterdam 2004;
- Bon, J. van – *IT service management – een introductie op basis van ITIL,* van Haren publishing 2004;
- Braam, R, Coppen F, et al – *ICT-Infrastructuur en datacommunicatie – organisatie, beheer en techniek* – 2e druk, Academic Service 2005;
- Collegesheet postdoctorale opleiding IT-Audit, Vrije Universiteit, 2005, 2006, 2007;
- Franken, H, Kasperen H.W.K, Wild, A.H. de – *Recht en computer – recht en praktijk – vijfde druk*, Kluwer 2004;
- Gemeente Amsterdam – *Open software strategie* – Directie concern Organisatie afdeling informatiebeleid 2006;
- Gemeente Groningen – *Raadsadvies Open Source* – DIA 2005;
- Gemeente Groningen – *Voorstel aanpak OS op de werkplekken Gemeente Groningen*, DIA 2005;
- Glashouwer, B. et al – *Informatiebeveiliging voor de overheid – een praktische aanpak*, Het expertise centrum 2002;
- Hoepman, J.H. – *Veiliger software door open source*, Informatiebeveiliging December 2005;
- Hoffer, J.A. et al – *Modern Database Management (8th edition)*, Prentice hall 2006;
- IB-Groep – *PRINCE2 – De IB-Groep afspraken,* IB-Groep 2004;
- IB-Groep – *Programma Technologie 2006 – 2009: 3 Beleidsplan,* IB-Groep 2006;
- IT Governance Institute and office of government commerce – *Aligning CobiT, ITIL and ISO 17799 for business benefit*, 2005;
- Koening, J. – *Does linux make a difference?*, Enterprise open source magazine, 2006;
- Langendijk, S. – *Falend projectbeheer hindert sector* – Computable juni, VNU 2006;
- Leeuwen, S. van – *Tien redenen waarom it –projecten falen,* Indora informatisering 2006;
- Lister, A.M. & Eager, R.D. – *Inleiding besturingssystemen – 4e geheel herziene uitgave,* Academic service 1994;
- Mitnick, K. – *The art of deception – Controlling the human element of security,* John Wiley and Sons 2003;
- Nederlands Normalisatie Instituut – Code voor informatiebeveiliging – een leidraad voor beleid en implementatie, NEN-NNI 2000;
- NOREA – *Geschrift nummer 1 – IT-auditing aangeduid* – NOREA 1998;
- NOREA – *Studierapport nummer 2 – Een kwaliteitsmodel voor Register EDP-auditors – de eerste stap*, NOREA 1997;
- Onna, M. Van & Koning A. – *De kleine PRINCE2 – Gids voor projectmanagement*, tenHagen stam 2003;
- Oud, E. – *Inrichten en onderhouden van een continuïteitsvoorziening*, GetronicsPinkroccade 2005;
- Praat, J. Van & Suerink J.M. – *Inleiding EDP-Auditing*, tenHagen stam 2004;
- Raymond, E.S. – *The cathedral and the bazaar*, O'Reilly 2001;
- Rijkswaterstaat – *De realisatie van een Geodata infrastructuur bij Rijkswaterstaat*, Rijkswaterstaat 2006;
- Silberschatz, A, Galvin P.B, Gagne, G. – *Operating systems with Java,* Academic service 2004;
- Silver, M.A. – Examaning where Linux desktop and Open source products make sense – Gartner 2005;
- Stamp, M. – *Information security principles and Practice*, Wiley Interscience 2006;
- Thiadens, T. – *Beheer van ICT-voorzieningen – infrastructuren, applicaties en organisatie – vierde herziene druk*, Academic service 2002;

- Varian, H.R & Shapiro, C.– *Linux adoption in the public sector: an economic analysis*, Berkeley 2005;
- Weber, S. – The political economy of open source software, Brie working papers 2000;
- Wheeler, D.A. - *Why open source software / free software (OSS/FS, FLOSS or FOSS)? Look at the numbers,* Wheeler 2005;
- Willis, N. – *Computer architecture and communications* – McGraw Hill 1993.

Bijlage 2. Gehanteerde websites

- http://www.abraxax.com/html/disaster_recovery.html
- http://www.abraxax.com/html/informatiebeveiliging.html
- http://www.abraxax.com/html/tco.html
- http://www.amsterdam.nl/nieuws?ActItmIdt=26376
- http://www.baldrige.nist.gov/
- http://www.bkwi.nl
- http://www.bkwi.nl/fileadmin/informatiebeveiligingsplan/docs/Richtlijnen_html/11.Continuteitsmanagement.html
- http://www.gartner.com/DisplayDocument?id=482489
- http://www.opensource.org/docs/definition.php
- http://www.ososs.nl
- http://www.ososs.nl/article.jsp?article=63957
- http://www.politiek-digitaal.nl/opensource/economische_orde_en_de_geschiedenis_van_open_source
- http://www.vosn.nl/index.php?sectie=default&groep=open+source
- http://www.vosn.nl/index.php?sectie=overheid&groep=ministeries

Bijlage 3. Lijst geïnterviewden

- Cadee, Gerben – *Procesmanager ICT-infrastructuur en exploitatie* bij IB-Groep;
- Gelder, Suzanne van – *Ondernemer / Programmeur* bij Tricat Consulting;
- Kingma, Seth - *Ondernemer / Programmeur* bij Tricat Consulting;
- Meijer, Harry - *ICT-Specialist Informatie management* bij IB-Groep;
- Menduapessy, Piet – *Decentrale Auditor Dienst Informatisering en Automatisering* bij Gemeente Groningen;
- Nauta, Jan Lolle – *Ondernemer / Programmeur* bij Magenta Multimedia Tools;
- Widt, Ernst de - *Central security officer bij* IB-Groep.

Bijlage 4. Onderscheid closed en open source software vanuit de code voor informatiebeveiliging

De code voor informatiebeveiliging is een handleiding voor de beveiliging van organisaties. De code zegt hierover: "*De code is bedoeld als referentiepunt voor het vaststellen van de reeks beveiligingsmaatregelen die nodig zijn in de meeste situaties waarin informatiesystemen worden gebruikt in industrie en handel.*" De concrete invulling van de te nemen beveilingsmaatregelen is organisatiespecifiek. Hieronder vindt u per relevant hoofdstuk uit de code voor informatiebeveiliging de doelen van het gebruik van open source software.

Definitie en doel
Beveiligingsbeleid
Doel: het bieden van sturing en ondersteuning van het management ten behoeve van informatiebeveiliging.

Beveiligingsorganisatie
Doel: het managen van de informatiebeveiliging binnen de organisatie.

Classificatie en beheer van bedrijfsmiddelen
Doel: waarborgen dat informatiebedrijfsmiddelen een passend niveau van beveiliging krijgen.

Beveiligingseisen ten aanzien van personeel
Doel: het verminderen van de risico's van menselijke fouten, diefstal, fraude of misbruik van voorzieningen.

Fysieke beveiliging en beveiliging van de omgeving
Doel: het voorkomen van ongeautoriseerde toegang tot, schade aan of verstoring van de gebouwen en informatie van de organisatie.

Beheer van communicatie- en bedieningsprocessen
Doel: het garanderen van een correcte en veilige bediening van IT-voorzieningen

Toegangsbeveiliging
Doel: het beheersen van de toegang tot informatie

Ontwikkeling en onderhoud van systemen
Doel: waarborgen dat beveiliging wordt ingebouwd in informatiesystemen.

Continuïteitsmangement
Doel: het reageren op verstoringen van bedrijfsactiviteiten en het beschermen van de kritische bedrijfsprocessen tegen de effecten van grootschalige storingen of calamiteiten.

De Code voor Informatiebeveiliging is niet geschreven vanuit een closed of open source software specifieke situatie. De wijze waarop invulling wordt gegeven aan de specifieke situatie die open source software met zich meebrengt, verschilt van closed source software.

Bijlage 5. Invloed van de code op open source kenmerken

De hieronder gepresenteerde tabel geeft de relatie weer tussen de relevante hoofdstukken uit de Code voor Informatiebeveiliging en de tien kenmerkende eigenschappen van open source software, zoals gedefinieerd door de Vereniging Open Source Nederland en Opensource.org. In de tabel geeft een vink op de intersectie aan waar aanpassingen gedaan moeten worden. Binnen de scope van deze scriptie is hier alleen aangeven dat aan de intersecties specifieke aandacht besteed moet worden.

	Beveiligingsbeleid	**Beveiligingsorganisatie**	**Classificatie en beheer van bedrijfsmiddelen**	**Beveiligingseisen ten aanzien van personeel**	**Fysieke beveiliging en beveiliging van de omgeving**	**Beheer van communicatie- en bedieningsprocessen**	**Toegangsbeveiliging**	**Ontwikkeling en onderhoud van systemen**	**Continuïteitsmanagement**
Free distribution	√	√				√		√	√
Source code	√	√	√	√		√		√	√
Derived works	√	√	√				√	√	√
Integrity of the author's source code		√						√	√
No discrimination against persons or groups									
No discrimination against field of endeavor									
Distribution of license	√	√	√						√
License must not be specific to a product		√							
License must not restrict other software	√	√	√			√	√	√	√
License must be technology-neutral		√				√			√

Bijlage 6. Open source & business continuity

Van Bon (2004) definieert Business Continuity Management vanuit ITIL als:

> *Een proces dat bestaat uit calamiteitenbeheersing en risicobeheersing met als doel het bereiken van bedrijfscontinuïteit.*

Bij calamiteiten wordt veelal gedacht aan zaken als: brand, waterschade, vandalisme of terroristische aanslagen. Het verstoren van de IT-voorzieningen kan eveneens een continuïteitsrisico herbergen. Het risico neemt toe naarmate de afhankelijkheid van de organisatie van IT toeneemt. De toenemende afhankelijk van IT door bedrijven heeft tot gevolg dat de impact van verstoringen toeneemt. Uit de onderzochte literatuur blijkt in zake het ITIL proces business continuity dat de openbaarheid van de broncode, de gewenste functionaliteiten en de juridische aspecten, de gebieden zijn waar de meeste risico's zich bevinden ten aanzien van open source software. Vanuit de scope van het onderzoek is onderzocht waar de drie genoemde gebieden tevens een relatie hebben met de zes geïdentificeerde gerelateerde ITIL processen. Op basis van de gedane analyse is de onderstaande tabel tot stand gekomen.

	Openbaarheid broncode	**Gewenste functionaliteiten**	**Juridische aspecten**
Security management	√	√	√
Application management		√	
Release management	√	√	
Service level management		√	√
Availlability management		√	
Configuration management	√	√	√
Capacity management		√	
Change management	√	√	

Bijlage 7. IB-Groep technologie speerpunten

Onderstaande tekst is een citaat uit Hoofdstuk 4 van het Programma Technologie 2006 – 2009: 3 Beleidsplan.

Bij de strategische doelen zijn kernwoorden opgesomd die gebruikt zijn om de verbeteractiviteiten te clusteren in speerpunten. Deze speerpunten worden in voorzien van acties die een-op-een kunnen worden omgezet naar businesscases. Hieronder worden eerst de speerpunten uitgewerkt:

Modernisering

1. Modernisering van de Internet-faciliteit. Hieronder vallen: excellente online-faciliteit (B2C), excellente online-services (B2B), Websphere-ontwikkelstraat;
2. Modernisering van de middleware. Flexibilisering, c.q. ontvlechting informatiesystemen, scheiding front/backoffice, migratiearchitectuur, online-koppelingen met ketenpartners. Hiervoor kunnen nieuwe middleware-technieken worden ingezet, zoals *Enterprise Service Bus*. De organisatie zal moeten worden aangepast (*Integration Competence Center*, c.q. afdeling *Koppelingen*);
3. Modernisering Technische Infrastructuur
   (Up-to-date blijven voor alle onderdelen van de infrastructuur: AS400-park, Netwerk, koppelingen, Windows-servers, Windows-werkplekken, Randapparatuur, etc.);
4. Modernisering Infrastructuur voor Software Ontwikkeling
   (ontwikkeltraject, ontwikkelmethode, ontwikkeltool, omgeving en versiebeheer, productiviteitstools, etc.).

Innovatie en samenwerking

Anticiperen op toekomstige ontwikkelingen, dus meer aandacht voor technologische vernieuwing, o.a. door:

1. Het inrichten van een *ICT Competence Center;*
2. Samenwerken met ketenpartners op technologisch gebied.

Professionalisering

1. Verhogen beschikbaarheid (uitwijk op alle niveaus, hot-stand-by waar dat nodig is);
2. Centraliseren van beheer (servers, gebruikersinstellingen, werkplekken, …);
3. Monitoring (alle systemen, en later ook alle services);
4. Beveiliging (alle systemen, alle niveaus, intern/externe bedreigingen);
5. Terugdringen complexiteit (alle systemen, alle infrastructuur, één oplossing per probleem);
6. Vergroten van kennis bij ontwikkelaars en beheerders;
7. Toevoegen van nieuwe senior-ontwikkelaars en senior-beheerders tbv kwaliteitsverbetering.

Kostenbeheersing

1. Aantal ontwikkelstraten terugbrengen van vier naar twee:
   - AllFusion-Plex voor Backoffice-systemen;
   - Websphere/Java voor Frontoffice-systemen;
2. Minimalisatie aantal systemen, tools, applicaties;
3. Minimalisatie functionaliteit (80-20 regel);
4. Consolideren en virtualiseren Windows-servers;
5. Consolideren Windows-dataopslag (SAN);
6. Meer aandacht voor goed ontwerp.

Beleidsuitwerking

Het Programma Technologie zal in onderdelen worden uitgewerkt, zodat concrete richtlijnen, architecturen en inrichtingsplannen ook op operationeel niveau gebruikt kunnen worden.

Bijlage 8. Beleidsuitgangspunten IB-Groep inzake technologie

Onderstaande tekst is een citaat uit Hoofdstuk 6 van het Programma Technologie 2006 – 2009: 3 Beleidsplan.

Algemeen Beleid
Hieronder staat het beleid voor de inrichting van de ICT-infrastructuur in steekwoorden opgesomd, uitgaande van de 6 kwaliteitseisen die aan de ICT infrastructuur gesteld worden.

Kwaliteitseisen ICT-infrastructuur

1. Robuust;
2. Flexibel;
3. Kostenefficiënt;
4. Beheersbaar;
5. Betrouwbaar;
6. Innovatief.

Uitgangspunten voor aanschaf/inrichting

1. Marktconform;
2. Kopen in plaats van (laten) maken en daarbij de markt volgen;
3. Geen IB-Groep specifieke aanpassingen op standaard oplossingen/tools;
4. Zo eenvoudig mogelijk, 80/20 principe;
5. Up-to-date, maar niet voorop;
6. Herbruikbaar;
7. Kosteneffectief;
8. Wereldstandaard & bewezen technologie;
9. Platformonafhankelijk, leverancieronafhankelijk;
10. Open, gelaagd, modulair;
11. Eén oplossing per probleem (één applicatie per functie);
12. Schaalbaar, beheersbaar, beveiligbaar;
13. Open-source waar dat mogelijk en nuttig is.

Inrichtingsdoelen

1. Schaalbare internetfaciliteit voor grootschalige elektronische dienstverlening;
2. Volledig ingerichte internet ontwikkelstraat;
3. Flexibele middleware, geschikt voor: internet, legacy-ontvlechting en platform-migratie;
4. Minimalisatie aantal ontwikkelstraten;
5. Volledig eigen rekencentrum (op termijn mogelijk inclusief WSF);
6. 7x24 beschikbaarheid van de on-line (gerelateerde) systemen;
7. Volledig eigen uitwijkrekencentrum (bij nieuwbouw mogelijk elders);
8. Volledige scheiding omgevingen;
9. Gescheiden netwerkzones (dmz, internetserverzone, backoffice, extranet);
10. Centrale beheertools voor beheer, monitoring, trendanalyse van alle lagen van de ICT-infrastructuur;
11. Geconsolideerde/virtualiseerde servers en dataopslag;
12. Schaalbaar netwerk/koppelingen, minimaliseren extranet;
13. Minimalisatie van beheerslast van servers, netwerken, werkplekken en randapparatuur;
14. Hoogwaardige interne en externe ICT-beveiliging;
15. Een actief *ICT Compentence Centre;*
16. Uitfaseren niet meer ondersteunde/gebruikte systeemcomponenten;
17. Gebruik maken van open-source waar dat mogelijk en nuttig is.

Bijlage 9. Organisatiediagram IB-Groep