



# ***Kennis en documentatie in het software ontwikkelingsproces***

***Gevalstudie: Het kennisprobleem in het software ontwikkelingsproces van een grootbank***

**Lieven Volckaert**

promotor :  
Prof. Jeanne SCHREURS

## **Woord vooraf**

Deze masterproef werd geschreven in het kader van mijn studies Handelsingenieur in de Beleidsinformatica aan de Universiteit Hasselt.

Ik zou deze plaats willen gebruiken voor enkele mensen te bedanken. Hierbij denk ik allereerst aan mijn promotor Prof. Dr. J. Schreurs voor de begeleiding, de vele raadgevingen en suggesties die ze gedaan heeft. Daarnaast wil ik ook dhr. Grosemans bedanken voor de begeleiding en altijd klaar te staan voor te antwoorden op de vragen die ik had.

Daarnaast wil ik ook mijn familie, vrienden en medestudenten bedanken voor het nalezen en de vele steun en tips die ze mij gegeven hebben.

## **Samenvatting**

Kennis is één van de manieren waarop bedrijven heden ten dage een competitief voordeel kunnen verkrijgen. Dit maakt dat kennismanagement een actueel onderwerp is waar vele bedrijven in willen investeren om tot een ideale oplossing te komen. Het vergaren en delen van kennis is een actueel probleem. Door allerlei sociale dilemma's bestaat er een inherente weerstand bij de werknemers om hun kennis te delen. Daarom moeten bedrijven voor een aantal voorwaarden zorgen voordat de werknemers effectief en efficiënt hun kennis zullen delen. Vertrouwen ligt aan de basis van kennisdeling en ook beloningssystemen kunnen helpen bij het aanmoedigen van kennisdeling. Al-Alawi (2007) bepaalt in zijn onderzoek de kritische succesfactoren van kennisdeling en Kharabsheh (2005) heeft een model vooropgesteld met de voorwaarden voor kennisdeling. 'Communities of practice' kunnen eveneens het delen van kennis in een organisatie bevorderen.

Daarnaast maken IT bedrijven gebruik van een software ontwikkelingsmethodologie en de projectmanagement methode om tot duurzame ontwikkeling te komen. Het waterval model voor software ontwikkeling is niet actueel meer en de nieuwe trend is 'agile software development'. 'Agile' is gebaseerd op een groep van iteratieve software ontwikkelingsmethodologieën, waarbij de systeemvereisten en de oplossingen samen evolueren door de inzet van crossfunctionele teams. Voor het managen van projecten wordt er tegenwoordig veel gebruik gemaakt van de PRINCE2 methode. Het is een zeer gestructureerde methode die de verschillende rollen van de gebruikers definieert en de aanmaak van een aantal documenten vereist. Deze documenten omvatten dus projectdocumentatie maar ook allerlei systeemdokumentatie. Er is een 'agile' methode om deze documentatie te creëren en onderhouden, die ook wel 'barely sufficient' documentatie genoemd wordt. Aanvragers van documenten moeten de 'Total Cost of Ownership' van die documenten goed begrijpen en documenten moeten pas geactualiseerd worden als het pijn doet.

Er heeft een interview plaatsgevonden met dhr. Grosemans uit een grootbank die met projecten in de software ontwikkeling werkt. Via een interview werd de situatie van het bedrijf in kaart gebracht. Deze situatie werd vergeleken met de theorie en geanalyseerd. Ten slotte werden er enkele conclusies geformuleerd voor het verbeteren van het kennis- en documentatiemanagement bij de grootbank.

## Inhoudsopgave

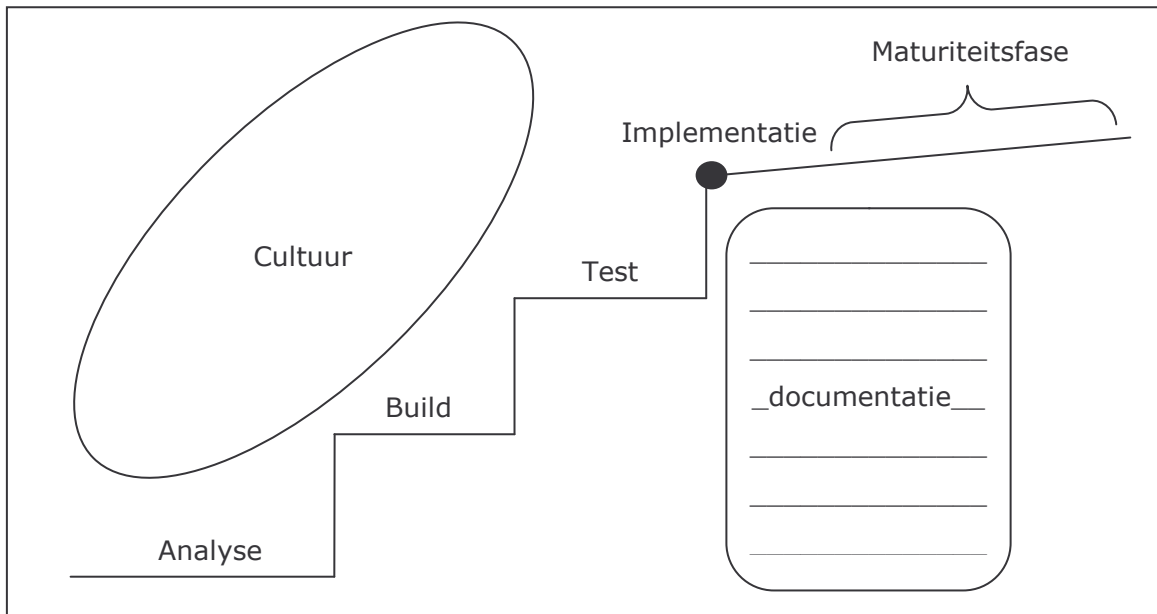
Hoofdstuk 1: Inleiding .....	1
1.1. Waaron deze thesis.....	1
1.2. Onderzoeksvraag.....	2
1.3. Methodologie .....	2
1.4. Overzicht van de thesis in hoofdstukken .....	2
Hoofdstuk 2: Organisatiecultuur van kennisdeling.....	4
2.1. Wat is kennis .....	4
2.1.1. Data, informatie, kennis .....	4
2.1.2. Epistemologische achtergronden van kennis .....	5
2.1.3. Soorten kennis.....	7
2.1.4. Kennis in organisaties .....	8
2.2. Kennisdeling .....	10
2.2.1. Inleiding tot kennisdeling.....	10
2.2.2. Het probleem van kennisdeling.....	11
2.3. Organisatiecultuur die leidt tot kennisdeling .....	14
2.3.1. Inleiding.....	14
2.3.2. Vertrouwen als voorwaarde tot aanmoediging van kennisdeling.....	14
2.3.3. Beloningssystemen als voorwaarde voor de aanmoediging van kennisdeling .....	17
2.3.4. Kritische succesfactoren van kennisdeling volgens Al-Alawi.....	17
2.3.5. Model van kennisdeling volgens Kharabsheh .....	19
2.3.6. Kennisdeling in een project.....	21
2.3.7. Communities of Practice .....	24
2.3.8. Leaving Expert Debriefing.....	25
Hoofdstuk 3: Het duurzame ontwikkelingsproces.....	27
3.1. Duurzaamheid.....	27

3.2.	Software ontwikkelingsmethodes .....	30
3.2.1.	Waterval model .....	30
3.2.2.	Agile software ontwikkelingsmethode .....	31
3.3.	PRINCE2 projectmanagement methode .....	34
3.4.	Documentatie in een duurzaam en agile ontwikkelingsproces .....	40
3.4.1.	Documentatie van het project .....	40
3.4.2.	Documentatie van het systeem .....	46
Hoofdstuk 4:	Gevalstudie: Duurzame ontwikkeling bij een grootbank.....	52
4.1.	Inleiding van de gevalstudie.....	52
4.2.	Situatie bij de grootbank .....	54
4.3.	Analyse van de gevalstudie.....	62
4.4.	Aanbevelingen.....	65
Hoofdstuk 5:	Conclusies .....	67
Hoofdstuk 6:	Lijst van geraadpleegde werken .....	69

## Hoofdstuk 1: Inleiding

### 1.1. Waarom deze thesis

Het onderwerp van deze thesis is voorgesteld door dhr. Grosemans. Hij heeft het probleem verduidelijkt met het volgende schema.



Dit schema geeft het globale verloop van de bouw en het onderhoud van een systeem weer, deze starten met de verschillende stappen van het software ontwikkelingsproces gevolgd door de implementatie en een maturiteitsfase waar het onderhoud van het systeem gebeurt. Tijdens dit volledige proces beïnvloedt de bedrijfscultuur de mate van kennisdeling en worden er documenten aangemaakt en bewaard die allerlei kennis bevatten. Het is de uitdaging voor het bedrijfsmanagement om de optimale bedrijfscultuur te creëren die het delen van kennis maximaal bevordert. Daarnaast zullen de juiste documenten aangemaakt en overzichtelijke bewaard moeten worden.

Het kennisprobleem speelt zich vooral af tijdens de implementatiefase. Tijdens deze fase is het meestal zeer druk en wordt de documentatie snel verwaarloosd. Daar bovenop volgt er vlak na deze fase ook een teamwissel. De externe medewerkers van het team worden overgeplaatst naar een ander project of hun contract wordt beëindigd. Ook veel van de interne medewerkers worden overgeplaatst naar nieuwe projecten of andere functies binnen het bedrijf. Op die manier kan er veel kennis

verloren gaan waardoor de duurzaamheid van het ontwikkelde product in het gedrang komt.

## **1.2. Onderzoeksvraag**

Deze probleemstelling geeft de volgende centrale onderzoeksvraag:

“Hoe kan de duurzaamheid van software ontwikkeling via kennis- en documentatiemanagement gewaarborgd worden?”

Deze vraag wordt verder uitgediept in de volgende deelvragen:

1. Wat is kennis en kennisdeling?
2. Hoe kan kennisdeling aangemoedigd worden?
3. Hoe garanderen we een duurzaam ontwikkelingsproces?
4. Welke project- en systeemdocumentatie is belangrijk?

## **1.3. Methodologie**

Via een literatuurstudie over zowel kennis en kennisdeling als duurzaamheid in het ontwikkelingsproces zoek ik vooreerst een theoretisch antwoord op deze onderzoeksvraag en de deelvragen. Om een antwoord te vinden op de vierde deelvraag zullen ook een aantal project en software ontwikkelingsmethodes bestudeerd.

Voor de praktijkstudie zal ik een interview afnemen van dhr. Grosemans. Dit zal mij in staat stellen om een duidelijk beeld te vormen van de huidige situatie qua kennismanagement op zijn afdeling. Na het interview zal deze situatie verder geanalyseerd en vergeleken worden met de theorie van de literatuurstudie. Tot slot wil ik met de resultaten van de analyse enkele aanbevelingen formuleren voor dhr. Grosemans die een positieve impact kunnen hebben op de kennisdeling binnen zijn afdeling.

## **1.4. Overzicht van de thesis in hoofdstukken**

In hoofdstuk twee worden vooreerst de termen kennis en kennisdeling nader gedefinieerd. De bestaande problemen met kennisdeling worden grondig beschreven. Vervolgens wordt er nagegaan hoe een organisatie een cultuur kan creëren die leidt tot kennisdeling. Hiervoor worden twee modellen uit de literatuur beschreven. De



impact van vertrouwen en beloningssystemen bij kennisdeling komen eveneens aan bod. Als laatste worden 'communities of practice' en het 'leaving expert' probleem bestudeerd.

In hoofdstuk drie start met een beschrijving van het duurzame en 'agile' ontwikkelingsproces. Eerst wordt er onderzocht wat duurzaamheid juist inhoudt en wordt er een schema weergegeven met de factoren die leiden tot een duurzaam ontwikkelingsproces. Daarna wordt de 'agile' software ontwikkelingsmethode verder besproken en vergeleken met het traditionele waterval model. Vervolgens wordt de projectmanagement methode PRINCE2 beschreven. Tot slot gaat hoofdstuk drie in op de documentatie die gegenereerd wordt tijdens het ontwikkelingsproces. Hierbij wordt zowel gekeken naar de projectdocumentatie, met een uitgebreide kijk naar het 'lessons learned' document, als naar de systeemdokumentatie.

Hoofdstuk vier bevat het praktische onderzoek van deze thesis. Eerst wordt de gevalstudie ingeleid en wordt er beschreven hoe het interview tot stand is gekomen. Daarna wordt de situatie op de afdeling van dhr. Grosemans geschetst, geanalyseerd en vergeleken met de theorie uit hoofdstukken twee en drie. Ten slotte worden enkele aanbevelingen geformuleerd die mogelijks kunnen leiden tot een verbetering van de kennisdeling op zijn afdeling.

Als laatste hoofdstuk bevat hoofdstuk vijf de conclusies van deze thesis.

## **Hoofdstuk 2: Organisatiecultuur van kennisdeling**

### **2.1. Wat is kennis**

#### **2.1.1. Data, informatie, kennis**

Kennis is geen data en ook geen informatie maar het staat wel in verband met beide. Daarom zal ik eerst de termen 'data' en 'informatie' verduidelijk vooraleer een definitie van kennis te geven.

'Data' zijn discrete, objectieve feiten over gebeurtenissen. In de organisatorische context wordt de term data omschreven als gestructureerde records van transacties. De verschillende departementen van een organisatie creëren data en al deze data worden bijgehouden op de servers. Elke organisatie heeft nood aan data en voor sommige organisaties zijn data hun core business. Vele organisaties houden zoveel mogelijk data bij maar het is veel beter dat er een beleid gevolgd wordt omtrent het bijhouden van data; te veel data maakt het moeilijk om de nuttige data te vinden. Peter Drucker zei ooit eens "information is data endowed with relevance and purpose." Data zijn dus de grondstoffen die een organisatie nodig heeft voor de creatie van informatie.

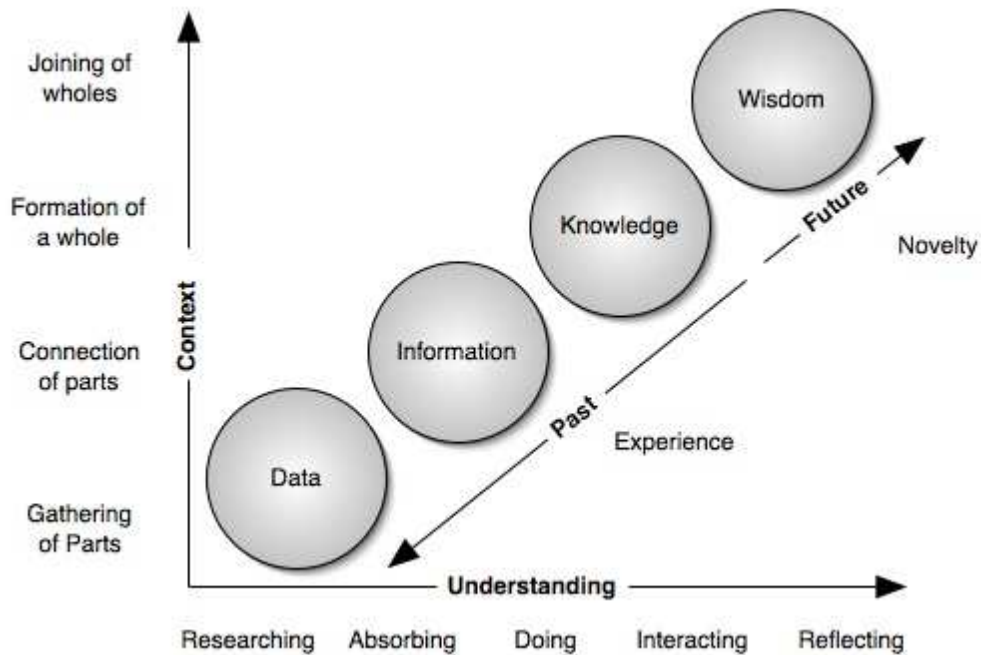
'Informatie' is een boodschap, meestal in de vorm van een document of een visuele of audio communicatie, die tot doel heeft een impact te hebben op het oordeel en het gedrag van de ontvanger. Daarnaast heeft informatie ook een betekenis, iets dat data niet hebben. Data kunnen informatie worden door de data te conceptualiseren, te categoriseren, te berekenen, te corrigeren en te condenseren.

Informatie kan overgedragen worden door harde of zachte netwerken. Harde netwerken zijn de zichtbare en formele netwerken van de organisatie, zoals e-mail berichten, brieven en vergaderingen. Zachte netwerken zijn meer informeel en ze ontstaan ad hoc, zoals bijvoorbeeld gesprekken tijdens de koffiepauze of een nota van een collega op je bureau.

Kennis is dieper, breder en rijker dan data of informatie. Kennis is een vloeiende mix van ervaringen, waarden, contextuele informatie en inzicht dat een kader vormt voor het evalueren en opnemen van nieuwe ervaringen en informatie. Kennis ontstaat en

wordt toegepast in de hoofden van de kenners. In organisaties ligt kennis niet enkel vast in de documenten en in de archieven maar ook in de routines, de processen, de praktijken en de normen van de organisatie. (Davenport en Prusak, 2000)

Er wordt een belangrijk onderscheid gemaakt tussen data, informatie, kennis en wijsheid. De verschillende begrippen hebben een lineair verband met elkaar dat duidelijk aangeduid wordt in de volgende grafiek.



Bron Hey (2004)

Volgens Hayes en Walsham (2003) is het niet mogelijk om kennis te bezitten, te codificeren, te organiseren en te delen zoals we doen met data en informatie.

### **2.1.2. Epistemologische achtergronden van kennis**

De term kennis heeft verschillende epistemologische achtergronden en verschillende auteurs hebben verschillende indelingen. Wij zullen er hier twee bespreken. Volgens Joshi, Sarker en Sarker (2006) zijn er drie epistemologische perspectieven voor kennis. Deze zijn het cognitivistische, het connectionistische en het autopoietische perspectief. Volgens Hayes en Walsham (2003) zijn er twee verschillende epistemologische achtergronden voor kennis management in de academische

literatuur, namelijk het 'content' en het 'relationeel' perspectief. We beginnen met het bespreken van de drie perspectieven van Joshi, Sarker en Sarker.

Vanuit het cognitivistisch epistemologisch perspectief is kennis vast en universeel overdraagbaar. Kennis is opgeslagen in computers, databases, archieven en documentatie. Vanuit dit perspectief bestaat er geen verschil tussen kennis en data omdat probleemloos kan uitgewisseld worden tussen twee partijen. Kennis is universeel en de karakteristieken van de verzender, de ontvanger of de kennis op zich hebben geen invloed op de overdracht of het delen van de kennis. Ook voorafgaande kennis heeft geen invloed op het aanleren van de nieuwe kennis.

Het connectionistisch epistemologisch perspectief geeft kennis niet dit universele karakter. Kennis wordt gezien als contextueel en er bestaan lokale verschillen tussen de verschillende bronnen van kennis. Connectionistische theoretici geloven dat kennis niet gemakkelijk overdraagbaar is tussen de zender en de ontvanger, door het contextuele karakter van kennis en door de overige factoren die een rol spelen bij de overdracht. Deze factoren zijn onder andere de noodzaak dat de zender en ontvanger elkaar goed verstaan en de invloed van sociale interacties, relaties en netwerken op de overdracht.

Het autopoietisch epistemologisch perspectief is het derde en laatste perspectief van Joshi et al. Hierbij staan de concepten van autonomie, eenheid en co-evolutie centraal. Kennis kan enkel autonoom opgebouwd worden, kennis is geen abstract gegeven en kan daardoor niet gedeeld worden. Onderzoekers die dit perspectief hanteren spreken dan ook over het omzetten van kennis in plaats van kennisdeling. Daarnaast geloven ze ook dat, omdat kennis nooit overgedragen kan worden, alle kennis wordt gecreëerd. In de literatuur betreffende dit perspectief spreken de onderzoekers over socialisatie, internalisering, externalisering en combinatie als mogelijke methoden voor de omzetting van kennis van de ene persoon naar de andere.

Het 'content' perspectief volgens Hayes en Walsham zegt dat kennis een voorspelbare waarheid is. Kennis kan men codificeren en bewaren in kennis banken, die er voor zorgen dat men kennis kan delen, bewerken en bewaren onafhankelijk van de mensen van wie deze kennis komt. In de literatuur wordt er, vanuit dit perspectief, regelmatig gesproken over kenniskopers en kennisverkopers in de organisatie of over personen die kennis bezitten, aanbieden en delen met anderen.

Het 'relationeel' perspectief staat hier loodrecht tegenover. Auteurs vanuit dit perspectief zien kennis niet als een entiteit die gemakkelijk gedeeld kan worden. Kennis is relatief en context gebonden. Daarom focussen deze auteurs zich eerder op het proces van kennen in plaats van kennis op zich. Bovendien voelt voor hen kennisdeling niet aan als een economische markt, maar is kennisdeling afhankelijk van de context waarin deze kennis verworven is. Ze benadrukken ook dat kennis een waarheid is voor een bepaald individu of een bepaalde groep, die echter misverstaan kan worden of uit zijn specifieke context gehaald kan worden door andere individuen of groepen.

In deze thesis ga ik uit van het connectionistische perspectief omdat dit het meest aansluit bij mijn onderzoek over IT ontwikkelingsprojecten. Deze worden tegenwoordig gerealiseerd door een team van werknemers uit verschillende afdelingen. Dit team communiceert en onderhandelt met alle 'stakeholders' via met als doel een goed systeem te creëren dat voldoet aan de eisen van de klant.

### **2.1.3. Soorten kennis**

Verschillende prominente onderzoekers in de 'Information System' discipline hebben erop gewezen dat het niet enkel belangrijk is om de verschillende soorten kennis te onderscheiden maar eveneens om te erkennen dat kennis van nature veelzijdig is. Volgens Basselier, Reich en Benbasat (2001) bestaat IT kennis uit 2 delen, 'explicit' IT kennis en 'tacit' IT kennis. 'Explicit' IT kennis is de formele kennis die overgedragen kan worden door middel van een systematische taal of, met andere woorden, het is de kennis die gelezen, geleerd en uitgelegd kan worden. Het is gespecialiseerde kennis, een groep van feiten en concepten die relevant zijn voor een specifieke taak. Hieronder verstaan ze zowel kennis van technologie, applicaties en systeem ontwikkeling, alsook kennis over het managen van IT-projecten en weten wie over welke kennis beschikt.

Volgens de onderzoekers wordt 'tacit' IT kennis opgebouwd uit twee elementen: ervaring en cognitie. Ervaring wordt omschreven als het vermogen om goed te presteren of het bezitten van de 'knowhow' om taken succesvol uit te voeren. Het is niet helemaal juist om te zeggen dat ervaring een directe invloed heeft op 'tacit' kennis omdat iedereen zijn ervaringen anders interpreteert en er verschillende kennis uithaalt. Toch gaan de onderzoekers ervan uit dat mensen met meer ervaring gemiddeld over meer 'tacit' kennis beschikken dan mensen met minder ervaring.

Volgens Nonaka (1994) zijn er twee variabelen die de kwaliteit van de ervaring beïnvloeden. Een eerste variabele is de variëteit van de ervaringen. De hoeveelheid kennis van een monotone, repetitieve taak die men vergaart, zal verminderen in de tijd. Routine klussen werken innovatieve ideeën en creatief denken tegen. Enkel de variëteit verhogen biedt geen perspectief op meer creatief denken als de persoon geen relatie ziet tussen de verschillende ervaringen. De tweede variabele van kwaliteitsvolle ervaringen is de intensiteit waarmee de persoon de ervaring beleefd heeft. Als de werknemer bijvoorbeeld door veel verantwoordelijkheid krijgt met betrekking tot het resultaat zal dit de ervaring ten goede komen. Hoe intenser de ervaring, hoe meer kennis opgenomen wordt.

Met het onderscheiden van het cognitieve element erkennen de onderzoekers dat 'tacit' kennis meer is dan enkel ervaringen opdoen. 'Tacit' kennis bevat ook de kennis over hoe de wereld in elkaar zit. Voor IT managers zijn er twee belangrijke cognitieve elementen. Ten eerste, inzicht hebben in de organisatie van de business activiteiten. Managers moeten voorbij de taken en de hiërarchie kijken en een duidelijk overzicht hebben van de processen aanwezig in de organisatie. Ten tweede dienen IT-managers een visie te ontwikkelen over hoe IT gebruikt kan worden in de organisatie, met andere woorden, weten waartoe IT in staat is en IT zo inzetten dat het een meerwaarde biedt voor de organisatie.

#### **2.1.4. Kennis in organisaties**

Argote en Ingram (2000) hebben een theoretisch kader ontwikkeld voor het indelen van kennis van organisaties. Ze gebruiken het woord reservoir om een opslagplaats van kennis aan te duiden. Dit woord duidt aan dat kennis opgeslagen wordt om later opnieuw gebruikt te worden. Volgens dit kader is alle organisatorische kennis opgeslagen in drie basis elementen: leden, instrumenten en taken. Leden zijn de menselijke componenten van een organisatie. Instrumenten zijn zowel de hardware als software die gebruikt worden en zijn dus de technologische componenten. De taken worden afgeleid uit het strategisch plan van de organisatie.

Deze basis elementen van het theoretische kader kunnen ook gecombineerd worden in verscheidene subnetwerken. Het lid-lid netwerk is het sociale netwerk van de organisatie en bevat alle interacties die werknemers met elkaar hebben. Het taak-taak netwerk bevat de volgorde van de taken en de routines die een organisatie uitvoert. Het instrument-instrument netwerk bevat alle combinaties van technologieën die in

het bedrijf gangbaar zijn. Het lid-taak netwerk duidt aan wie welke taak uitvoert of er verantwoordelijk voor is. Het lid-instrument netwerk wijst leden toe aan instrumenten. Het taak-instrument netwerk specificeert welke instrumenten gebruikt worden voor welke taken. Tenslotte, het lid-taak-instrument netwerk specificeert welke leden welke taken toebedeeld krijgen met welke instrumenten.

Volgens dit kader kunnen organisaties op twee manieren hun prestaties verbeteren. Ten eerste kunnen ze zorgen voor een betere interne compatibiliteit van de netwerken. Bijvoorbeeld, de prestaties zullen verbeteren als het lid-taak netwerk de taken toewijst aan de mensen die er het meest voor geschikt zijn. Ten tweede kunnen organisaties ook de externe compatibiliteit met andere netwerken verbeteren. Bijvoorbeeld als de leden de geschikte instrumenten hebben om hun taken te volbrengen is er een goede compatibiliteit tussen het lid-instrument netwerk en het lid-taak netwerk.

Een zeer groot deel van de kennis die een organisatie vergaart zit in de hoofden van de werknemers van de organisatie, de 'tacit' kennis. Het is mogelijk om deze kennis op te slaan in de taken en de technologieën van een organisatie. Zo kan een nieuwe ontdekking bijvoorbeeld geprogrammeerd worden in de software van het bedrijf zodat de kennis onmiddellijk door iedereen gebruikt wordt. Het is ook mogelijk om die nieuwe ontdekking op te nemen in de taken die de organisatie uitvoert. Als iedereen de taak op de nieuwe manier uitvoert zal de nieuwe kennis vervat zitten in de taak. Ten slotte is het ook mogelijk om kennis op te slaan in de verschillende netwerken die ontstaan door combinaties van leden, instrumenten en taken. Bijvoorbeeld de kennis van wie expert is in een bepaalde taak wordt opgeslagen in het lid-taak netwerk.

## **2.2. Kennisdeling**

### **2.2.1. Inleiding tot kennisdeling**

Kennisdeling vindt plaats in een organisatie wanneer werknemers de kennis van andere werknemers in de organisatie gebruiken om hun problemen op te lossen. Kennisdeling wordt dan ook gedefinieerd als het delen tussen de werknemers van die informatie, ideeën, suggesties en expertise, die relevant is voor de organisatie. Deze definitie duidt aan dat deze kennis zowel 'tacit' als 'explicit' kan zijn. De term kennisdeling geeft ook aan dat er een wisselwerking bestaat tussen de verzender en de ontvanger. Dit is een groot verschil met de term informatiedeling waarbij de deling volledig van één kant kan komen, zoals gebeurd wanneer een manager een nota naar zijn werknemers verstuurt. (Kharabsheh, 2005)

Zoals vermeld in het vorig hoofdstuk is kennis van essentieel belang voor een organisatie om competitief te zijn in de huidige kennis economie. Maar enkel deze kennis bezitten is niet genoeg. Door de groeiende globalisering, het groeiend aantal 'joint-ventures' en het aangaan van strategische allianties wordt het steeds belangrijker om deze kennis op een efficiënte en effectieve manier te delen. Bedrijven die in staat zijn de verworven kennis over te dragen tussen de verschillende eenheden zullen productiever zijn en beter kunnen overleven dan bedrijven die minder bedreven zijn in het delen van kennis. (Argote, 2000)(Chakravarthy, 2003)

Kennisdeling in organisaties is het proces waarbij één eenheid (bijvoorbeeld een individu, een team, een departement of een divisie) beïnvloed wordt door de ervaringen van een andere eenheid. Deze definitie duidt dus aan dat kennisoverdracht zich niet enkel afspeelt op individueel vlak. Bedrijven moeten een strategische visie hebben over kennisoverdracht en deze toepassen op alle eenheden en niet enkel op individuele werknemers.

Kennisoverdracht zorgt ervoor dat organisaties, departementen, groepen en individuen niet enkel leren uit hun eigen ervaringen maar ook kunnen profiteren van de ervaringen van anderen. Deze kennisoverdracht kan expliciet gebeuren bijvoorbeeld als één eenheid een andere eenheid informeert over een manier om de productiviteit te verbeteren. Kennisoverdracht kan ook impliciet gebeuren, bijvoorbeeld als een individu een nieuw instrument gebruikt dat ontworpen is om zijn productiviteit te verhogen, kan hij genieten van de voordelen van deze verhoogde



productiviteit zonder dat hij noodzakelijkerwijs het instrument verstaat of dat hij kan uitleggen waarom het instrument zijn productiviteit verhoogt.

Kennis reservoirs (zie vorig deel) zoals individuen en departementen spelen een duale rol in kennisoverdracht bij organisaties. Bij kennisoverdracht zullen deze kennis reservoirs veranderen omdat er nieuwe kennis wordt toegevoegd en oude gedachten en ideeën eruit verdwijnen. De situatie waarin een opslagplaats zich bevindt, bepaalt het resultaat van de kennisoverdracht. Net zoals de bereidheid en de vroegere ervaringen van een individu de kennisoverdracht beïnvloeden, beïnvloedt de huidige kennis van een organisatie zijn vermogen en zijn bereidheid om nieuwe kennis op te nemen.

Volgens Bartol en Srivastava (2002) zijn er vier mechanismen die een individu kan gebruiken om kennis te delen binnen een organisatie:

1. Het toevoegen van kennis aan de databank of de 'repository' van de organisatie.
2. Het delen van kennis tijdens formele gesprekken en tijdens vergaderingen binnen het team of met meerdere teams samen.
3. Het delen van kennis tijdens informele gesprekken tussen individuen.
4. Het delen van kennis binnen 'communities of practice.'

Volgens Lee en Ahn (2006) bestaan er twee soorten kennismanagement strategieën. De eerste strategie, de codeer strategie, probeert zoveel mogelijk kennis te coderen en te stokkeren in kennisbanken, waar ze gemakkelijk opgevraagd kan worden. Deze strategie probeert het probleem van 'tacit' kennis op te lossen door ze zoveel mogelijk 'explicit' te maken. De tweede strategie is de personalisatie strategie. In deze strategie probeert het management de kennis te verspreiden door interacties aan te moedigen tussen de werknemers. Ze focussen hierbij op zowel formele als informele contacten en ontmoetingen. 'Communities of practice' kunnen hierbij een grote rol spelen.

### **2.2.2. Het probleem van kennisdeling**

Ook al kennen de meeste bedrijven de voordelen en het belang van kennisoverdracht toch blijft het moeilijk om een succesvolle overdracht van kennis te realiseren. Sterker nog, uit onderzoek (Kharabsheh, 2005) blijkt dat er een zekere vijandigheid bestaat op de werkvloer tegenover kennisdeling. Dit kan verschillende redenen hebben.

Wanneer werknemers die het nut niet inzien van kennisoverdracht processen zullen ze deze ook niet toepassen. Daarnaast zijn individuen soms ook gewoon niet bereid hun kennis met iedereen te delen. Verder kunnen ook sterke sociale identiteiten en competitie tussen departementen obstakels zijn voor een succesvolle kennisoverdracht. (Argote, 2000)

Eén van de grote obstakels die een goede kennisdeling in de weg staat, is dat kennis vaak als een individueel carrière instrument gebruikt wordt. Voor vele mensen en organisaties geldt het spreekwoord "kennis is macht" en deze werknemers zullen de kennis die ze vergaard hebben tijdens hun loopbaan voor hun eigen belangen gebruiken. Een individuele werknemer kan er belang bij hebben om zijn kennis juist niet over te dragen. Zijn specifieke kennis maakt hem uniek of onmisbaar. Het is zeer belangrijk dat het management dit gevaar inziet en de juiste maatregelen neemt om de inwendige inertie of zelfs de aversie van werknemers tegenover kennisdelen weg te werken.

Concurrentie komt op alle niveaus voor. Net zoals bedrijven hun unieke kennis, zoals 'knowhow' of patenten, gebruiken om een betere positie om de markt te bekomen, zo zullen individuele werknemers hun unieke kennis gebruiken om hun sociale status te verbeteren of om promotie na te streven.

Stel twee jonge, carrièregerichte werknemers met dezelfde kwaliteiten, maar die niet dezelfde kennis bezitten. Ze dingen beiden naar dezelfde job en moeten hun overste bewijzen dat ze hiervoor het meest geschikt zijn. Het management heeft zijn werknemers verteld dat kennisdeling belangrijk is. De twee werknemers zitten nu in een 'Prisoner's Dilemma': "zullen ze hun unieke kennis delen of houden ze deze voor zichzelf?" Ze zullen er beide voor kiezen om hun kennis niet te delen en als ze al bereid zijn mee te werken aan de vraag van het management zullen ze enkel gemeenschappelijke of generische kennis delen. Ze zijn namelijk beiden beter af als ze hun kennis niet delen. Als de één zijn kennis deelt en de andere niet zal diegene die niet meegewerkt heeft, meer kennis bezitten, betere prestaties kunnen neerzetten en dus meer macht verwerven. (Cockburn, 2001)

Ook een ander sociaal dilemma speelt hierbij nog een rol, zijnde het 'free rider' probleem. Als een team aangemoedigd wordt om zijn kennis te delen zal het voor het individu interessanter zijn om hier niet op in te gaan. Als iedereen zijn kennis deelt, heeft ieder teamlid een voordeel, maar als er één iemand zijn kennis niet deelt dan

heeft dit teamlid meer voordeel dan de andere teamleden. De werknemers die hun kennis delen kunnen genieten van alle kennis die door iedereen gedeeld wordt, maar de persoon die bereid is tot kennisdeling kan hier ook van genieten en heeft daar bovenop nog steeds zijn unieke kennis die alleen hij bezit. Dit dilemma kan ervoor zorgen dat niemand bereid is mee te werken aan een kennisdeling project, waardoor het voorbestemd is tot mislukkingen, zelfs indien de beste technologische hulpmiddelen ter beschikking staan. (Cabrera & Cabrera, 2002)

Als laatste sociaal dilemma herken ik ook de principaal-agentproblematiek. Stel, een bedrijf neemt werknemers in dienst om de bedrijfsresultaten te verbeteren. Deze werknemers hebben hun eigen doelstellingen, die niet noodzakelijke overeenstemmen met de belangen van het bedrijf. Het bedrijf hecht veel waarde aan een goede kennisdeling is omdat dit leidt tot betere prestaties van de werknemers en het kan, zoals hierboven beschreven, een competitief voordeel opleveren voor het bedrijf. Of een werknemer nu carrière gericht is of integendeel zo weinig mogelijk werk wilt verrichten voor zijn loon, in beide gevallen zal het delen van kennis voor hem niet interessant zijn. In het eerste geval omdat het delen van kennis een afbouw is van zijn unieke kennis, zijn macht en dus ook zijn carrièremogelijkheden en in het tweede geval is het delen van kennis een extra taak en belasting die de werknemer het liefst zal vermijden.

Het is de taak van het management om deze dilemma's te doorbreken waardoor het achterhouden van kennis onaantrekkelijk wordt. Welke stappen ze kunnen ondernemen zullen we onderzoeken in de volgende paragraaf.

## **2.3. Organisatiecultuur die leidt tot kennisdeling**

### **2.3.1. Inleiding**

Het delen van kennis en het gebruiken van de geïmplementeerde 'knowledge management' systemen gebeurt niet vanzelf. Voor vele mensen geldt het spreekwoord *kennis is macht* en het delen van deze kennis staat voor hen gelijk aan het afbouwen van hun eigen macht. Verschillende aspecten binnen een organisatie moeten goed functioneren voordat werknemers overgaan tot het delen van informatie.

Bedrijven mogen er niet vanuit gaan dat hun werknemers spontaan overgaan tot het delen van hun kennis. Er moet een drijvende kracht aanwezig zijn die hen stuurt naar het delen van kennis en het management mag niet zomaar veronderstellen dat deze motivatie inherent aanwezig is op de werkvloer. Hier zijn drie belangrijke redenen voor. Allereerst is veel van de organisatorische kennis enkel aanwezig in de hoofden van de mensen. Het is persoonlijke kennis, die enkel gemeenschappelijke kennis kan worden als mensen bereid zijn ze mee te delen. Ten tweede bestaat er een conflict tussen de organisatie, die de kennis wil omzetten in een economisch voordeel, en de werknemers, die hun kennis als hun persoonlijk eigendom zien en deze toepassen en delen waar en wanneer ze willen. Ten derde is er het conflict binnen de groepen. Werknemers zijn terughoudend om kennis te delen met collega's omdat ze bang zijn dat hun collega's de vrijgegeven kennis gebruiken zonder zelf hun kennis te delen. Iedereen wil zijn machtspositie behouden. (Hislop, 2005)

### **2.3.2. Vertrouwen als voorwaarde tot aanmoediging van kennisdeling**

Onderzoekers van het 'IBM Institute for Knowledge-Based Organizations' (Levin et. Al., 2002) hebben de rol van vertrouwen bestudeerd in kennisdeling. In de literatuur wordt vaak gesuggereerd dat de situatie waarin werknemers gedurende een lange periode samenwerken, automatisch leidt tot een succesvolle kennisdeling. Maar er bestaat weinig bewijsmateriaal waarom dit zo zou zijn. Volgens dit onderzoek is de onbrekende schakel tussen normale interacties bij werknemers, ook wel de 'strong ties' genoemd, en een succesvolle kennisdeling, het onderlinge vertrouwen tussen de werknemers.

Er zijn twee specifieke types van vertrouwen: het 'benevolence-based' vertrouwen en het 'competence-based' vertrouwen. Als mensen denken aan vertrouwen denken ze meestal aan het 'benevolence-based' vertrouwen, het vertrouwen in iemand dat hij geen opzettelijk kwaad zal verrichten wanneer hem de mogelijkheid gegeven wordt. Het 'competence-based' vertrouwen daarentegen, is een relatie tussen personen waarbij één individu ervan overtuigd is dat de andere persoon voldoende kennis bezit over een bepaald onderwerp. Het is mogelijk dat deze twee vormen van vertrouwen onafhankelijk voorkomen bij één en dezelfde persoon. Bijvoorbeeld een werknemer vertrouwt erop dat een collega de informatie kent die hij nodig heeft ('competence') maar hij vertrouwt het misschien niet om deze informatie met hem te delen als hij erom vraagt ('benevolence'). Anderzijds is het ook mogelijk dat een werknemer er zeker van is dat een collega hem zal helpen als hij erom vraagt ('benevolence') maar betwijfelt hij of deze collega hiervoor wel de benodigde kennis bezit ('competence'). De resultaten van het onderzoek tonen aan dat kennis uitwisseling het meest effectief is wanneer er zowel 'benevolence-' als 'competence-based' vertrouwen aanwezig was.

Het is ook interessant om te vermelden dat de studie bijkomend aangetoond heeft dat de aanwezigheid van sterke relationele banden tussen collega's geen voorwaarde is voor het bestaan van een vertrouwensrelatie tussen die collega's. Het is mogelijk dat werknemers, die niet frequent met elkaar in contact komen ('weak ties'), een wederzijds vertrouwen ontwikkelen en er een goede kennisdeling tussen hen plaatsvindt. Verder blijkt ook uit de studie dat, bij een gelijk niveau van vertrouwen, de respondenten de kennis die ze ontvangen uit 'weak ties' nuttiger vinden dan de kennis verkregen uit 'strong ties'. Dit komt doordat medewerkers die 'strong ties' hebben, meestal dezelfde kennis hebben en dezelfde mensen, ideeën en concepten kennen. Mensen met 'weak ties' daarentegen behoren meestal tot verschillende sociale netwerken en komen in contact met verschillende types van kennis en ideeën. Hierdoor zijn deze 'weak ties' potentieel nuttiger voor het vinden van de juiste kennis.

Uit de resultaten blijkt ook dat 'competence-based' vertrouwen een grotere rol speelt bij de kennisoverdracht van 'tacit' kennis dan van 'explicit' kennis. 'Benevolence-based' vertrouwen is echter belangrijk in beide gevallen. De belangrijke vereisten bij het opbouwen 'competence-based' vertrouwen zijn een gemeenschappelijk taal, een gemeenschappelijke visie en discretie. Voor het opbouwen van het 'benevolence-based' vertrouwen zijn deze vereisten eveneens belangrijk, maar ook

ontvankelijkheid, de mate waarin de kennisbron een goede luisteraar is en 'strong ties' zijn belangrijke voorwaarden.

Uit dit onderzoek kunnen we concluderen dat vertrouwen een grote impact kan hebben op het delen van kennis in organisaties. Het management moet voor een omgeving zorgen waar de werknemers hun 'competence-' en 'benevolence-based' vertrouwen kunnen ontwikkelen als ze willen komen tot een goede kennisdeling.

Volgens de onderzoekers kunnen de managers het opbouwen van een vertrouwensomgeving bevorderen door:

- Het creëren van een gemeenschappelijk taal en gemeenschappelijke objectieven zal de werknemers toelaten de visie en de missie van het bedrijf te begrijpen en te onderschrijven. Dit zal bijdragen tot de opbouw van zowel het 'competence-' als het 'benevolence-based' vertrouwen.
- Het demonstreren van vertrouwen opbouwend gedrag. Managers kunnen het gedrag van hun werknemers positief beïnvloeden. Indien managers het goede voorbeeld geven door zelf goed te luisteren naar werknemers, ze aan te moedigen om hun opinie te delen en discreet om te gaan met vertrouwelijke informatie zal dit zorgen voor een groeiend vertrouwen tussen werknemers en het management.
- Het samenbrengen van mensen. Managers moeten overwegen hoe ze fysieke en virtuele ontmoetingsplaatsen kunnen creëren waar mensen gemakkelijk met elkaar kunnen communiceren. Enkel regelmatige interacties tussen werknemers zijn niet voldoende voor het creëren van vertrouwensrelaties maar ze hebben zeker geen negatieve impact. Vooral voor teamleden die op verschillende sites werken is het een goed idee om het team af en toe samen te brengen, zeker aan het begin van een project.

De grootste conclusie uit dit onderzoek is dat het bevorderen van kennisdeling meer is dan mensen samenbrengen in een vergaderruimte. Het management moet een omgeving creëren waar mensen kunnen inschatten welke kennis hun collega's bezitten en of deze collega's bereid zijn om deze kennis te delen.

### **2.3.3. Beloningssystemen als voorwaarde voor de aanmoediging van kennisdeling**

Bij beloningen bestaat er een onderscheid tussen extrinsieke beloningen, zoals monetaire bonussen, cadeaubonnen, lof en publieke erkenning, en intrinsieke beloningen zoals het plezier om een taak goed te doen en de erkenning van collega's. Beloningen mogen niet zomaar willekeurig uitgedeeld worden en er moet bij de werknemers een gepercipieerde gerechtvaardigheid voor de beloning bestaan. Zonder deze gerechtvaardigheid zullen de werknemers de beloningen niet in verband brengen met hun prestaties en zullen de beloningen dus hun doel missen. Daarom moet het ook mogelijk zijn voor de beloninggever om de geleverde inspanningen te kunnen observeren en hun waarde in te schatten. (Bartol en Srivastava, 2002)

Door het bestaan van barrières voor kennisdeling, zoals werknemers die hun unieke kennis als macht zien om hun status te behouden in de organisatie, zeker in tijden van crisis, en de kost die verbonden is met het daadwerkelijke delen van kennis, zijn onderzoekers het erover eens dat het noodzakelijk is om werknemers te belonen als ze aan kennisdeling doen. Er bestaat echter onenigheid over de vorm die deze beloning moet aannemen. Bartol en Srivastava (2002) en ook Lee en Ahn (2006) bepleiten het gebruik van monetaire beloningen als ideaal middel voor het aanzetten tot kennisdeling. Andere onderzoekers, zoals Kharabsheh (2005) en Gammelgaard (2007) zijn eerder voorstander van niet monetaire beloningen en het bevorderen van de intrinsieke motivatie voor het delen van kennis.

### **2.3.4. Kritische succesfactoren van kennisdeling volgens Al-Alawi**

Er zijn reeds verschillende studies uitgevoerd naar de factoren die een invloed hebben op het delen van kennis. Al-Alawi et. al. (2007) heeft aan de hand van een vragenlijst de kritische succesfactoren van kennisdeling proberen te achterhalen. Uit hun onderzoek blijkt dat de volgende vijf factoren positief in verband gebracht konden worden met kennisdeling:

- Vertrouwen. Interpersoonlijk vertrouwen, het vertrouwen in uzelf, en vertrouwen in collega's zijn zeer essentiële eigenschappen van de

organisatiecultuur, die een sterke invloed hebben op kennisdeling. Vertrouwen is de verwachting van het individu of van de groep in de betrouwbaarheid van de beloftes of acties van andere individuen of groepen.

- Communicatie tussen de werknemers. Communicatie verwijst hier naar de menselijke interacties en het gebruik van lichaamstaal tijdens gesproken conversaties. De beschikbaarheid van sociale netwerken op de werkvloer zal de menselijke interactie in belangrijke mate positief beïnvloeden. Deze vorm van communicatie is essentieel voor het aanmoedigen van kennisdeling.
- Informatiesystemen. De term informatiesystemen verwijst naar het geheel van mensen, data en processen die samenwerken om de werknemers te ondersteunen in hun dagelijkse taken, bij het oplossen van problemen en bij het nemen van beslissingen. Organisaties gebruiken verschillende informatiesystemen die kennisdeling vergemakkelijken. Dit kan door het opzetten van kennisbanken waar werknemers hun expertise elektronisch kunnen delen en waar ze toegang hebben tot de gedeelde ervaringen van andere werknemers.
- Beloningssysteem. Werknemers hebben een sterke motivator nodig voordat ze overgaan tot het delen van kennis. Het is onrealistisch om er vanuit te gaan dat alle werknemers bereid zijn hun kennis zomaar te delen, zonder dat ze berekenen wat ze kunnen winnen of verliezen met dit extra werk. Managers moeten de belangrijkheid van samenwerking en het delen van 'best practices' overwegen bij de ontwikkeling van het beloningssysteem. Het is immers belangrijk dat ze het juiste gedrag, zoals het delen van informatie en horizontale communicatie, aanmoedigen en belonen. Het is eveneens van belang dat deze beloningen toegekend worden op basis van de prestaties van de groep en niet op die van het individu. Uit dit onderzoek blijkt dat 90% van de respondenten akkoord gaan met de stelling dat een beloningssysteem voor kennisdeling, de mate van kennisdeling beïnvloedt. Het is dus een zeer belangrijke factor voor kennisdeling.
- Structuur van de organisatie. Het traditionele model van organisaties bestond vroeger dikwijls uit ingewikkelde lagen van verantwoordelijkheid, waarbij het verloop van informatie en kennisoverdracht dikwijls maar één pad kon volgen. Tegenwoordig realiseren de meeste managers zich dat een dergelijke bureaucratische structuur een sterk beperkende factor is voor een vlotte informatiestroom. Daarnaast vergde het met deze bureaucratische procedures

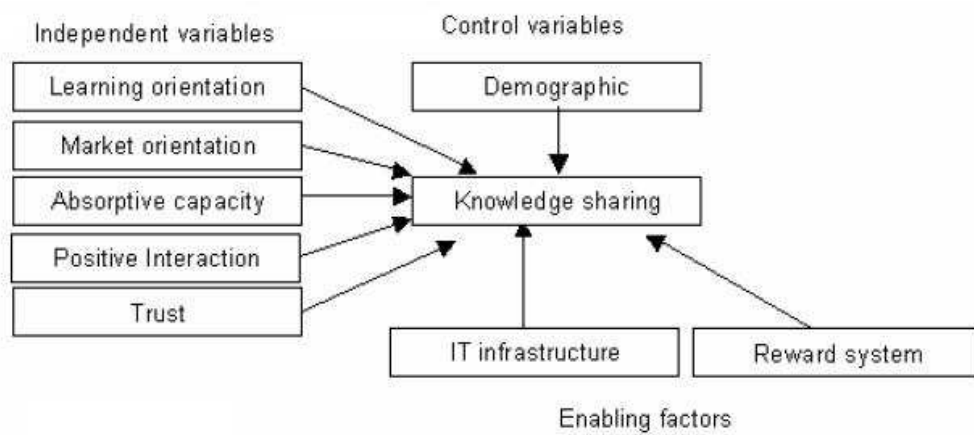


veel tijd vooraleer de kennis overgedragen was naar elk niveau. Een vlakke structuur zorgt dus zeker voor een betere en snellere kennisdeling.

Zoals eerder aangehaald blijkt uit dit onderzoek dat elk van deze factoren een positieve invloed had op de mate van kennisdeling.

### 2.3.5. Model van kennisdeling volgens Kharabsheh

Andere onderzoekers hebben het volgende model voor 'knowledge sharing' vooropgesteld. (Kharabsheh, 2005)



Bron Kharabsheh (2005)

Dit conceptueel model is een weergave van de factoren die een invloed hebben op kennisdeling volgens Kharabsheh. In dit model zijn 'learning orientation', 'market orientation' en 'absorptive capacity' de onafhankelijke variabelen. 'Positive interaction', 'trust', 'IT infrastructure' en 'reward system' zijn de sleutelfactoren.

- 'Learning orientation.' Deze term verwijst naar drie organisationele waarden die geassocieerd worden met de maturiteit van een organisatie op het vlak van leren. Deze drie waarden zijn: het enthousiasme voor het leren, een gedeelde visie op het leren en een open geest hebben voor nieuwe ideeën. De 'learning orientation' beïnvloedt de informatie die een organisatie bestudeert, interpreteert, evalueert en uiteindelijk accepteert of negeert. Organisaties die belang hechten aan het leren zullen dit ook stimuleren. Als werknemers en management een gemeenschappelijke visie hebben over hoe kennisdeling hen kan helpen, is er meer kans dat er daadwerkelijk kennisdeling tot stand komt.

Ten slotte zorgt een open geest ervoor dat verouderde werkwijzen verlaten kunnen worden en dat nieuwe ideeën gemakkelijker aanvaard en geïmplementeerd kunnen worden.

- 'Market orientation.' Dit betreft de beslissingen die op marketing gebied worden genomen om beter aan de wensen van de klant te voldoen dan de concurrerende bedrijven. Hoe hoger deze marktoriëntatie hoe meer belang de organisatie erbij heeft om de noodzakelijke kennis te delen.
- 'Absorptive capacity.' Dit is de mogelijkheid om de waarde van nieuwe externe informatie te herkennen, te assimileren en deze toe te passen voor commerciële doelstellingen. Het is een cumulatief werkend proces waarbij de 'absorptive capacity' volgens de onderzoekers geproduceerd wordt als bijproduct van de dagdagelijkse werking van de organisatie. 'Absorptive capacity' bestaat uit vier dimensies: acquisitie, assimilatie, transformatie, en exploitatie. De eerste twee dimensies definiëren de potentiële 'absorptive capacity' en de laatste twee dimensies de gerealiseerde 'absorptive capacity.'
- 'Positive interaction.' De meeste kennis wordt gedeeld via conversaties of telefoongesprekken. In organisaties waar een positieve sociale cultuur heerst, zullen management en werknemers regelmatig met elkaar communiceren zonder dat de functie van de gesprekspartners hierbij een barrière vormt. Organisaties die gericht zijn op het creëren van situaties waarbij werknemers met verschillende functies en posities, zowel formeel als informeel, met elkaar in contact kunnen treden, zullen een betere kennisdeling bekomen.
- 'Trust'. Er bestaat een wisselwerking tussen vertrouwen en samenwerking. Vertrouwen bevordert de samenwerking en het samenwerken helpt bij het opbouwen van een vertrouwensrelatie. Ze zijn dus sterk met elkaar verbonden en afhankelijk van elkaar. Hoe meer de werknemers elkaar vertrouwen, hoe beter de samenwerking en hoe beter hun kennisdeling dan ook wordt.
- 'IT infrastructure.' Er is een sterk verband vastgesteld tussen de aanwezigheid van technologie voor het bevorderen van kennisdeling en de kennisdeling op zich. Het is wel zeer belangrijk om te benadrukken dat technologie een factor is die kennisdeling bevordert maar de aanwezigheid van technologie is nog geen garantie dat die ook daadwerkelijk hiervoor gebruikt zal worden.
- 'Reward System.' Er bestaat een duidelijk positief verband tussen het bestaan van een beloningssysteem voor kennisdeling en de hoeveelheid kennisdeling. Maar net zoals bij technologie is ook een beloningssysteem geen garantie voor

kennisdeling. Er bestaat ook een belangrijk onderscheid tussen de soorten beloningen. Monetaire beloningen leveren wel een directe stijging van de hoeveelheid kennisdeling op maar de kwaliteit van de gedeelde kennis gaat naar beneden. Bovendien bevestigt een dergelijk beloningssysteem dat kennis een individueel goed is dat afgekocht moet worden en niet zomaar gedeeld kan worden. Het verdient de voorkeur om niet-monetaire beloningen te gebruiken zoals erkenning en persoonlijke ontwikkeling.

- 'Demographic.' Er is nog maar weinig onderzoek gedaan naar de demografische aspecten die een invloed hebben op kennisdeling, maar een aantal persoonlijke karakteristieken zoals introversie en geslacht hebben zeker een invloed op de mate van kennisdeling.

Al deze factoren hebben een invloed op kennisdeling en het is duidelijk dat kennisdeling een complex begrip is dat door vele verschillende factoren beïnvloed wordt.

### **2.3.6. Kennisdeling in een project**

Volgens Grillitsch (2007) zijn er in elk project drie cruciale fasen om tot het succesvol delen van projectkennis te komen. Deze zijn initialisatie, implementatie en institutionalisering. Pas als deze drie fasen doorlopen zijn, kan er sprake zijn van een succesvolle deling van de projectkennis.

Tijdens de opstartfase van het project moet ervoor gezorgd worden dat het delen van projectkennis mogelijk gemaakt en aangemoedigd wordt. Door het management dient een cultuur gecreëerd te worden die het delen van kennis tussen de werknemers bevordert. Of om het met de woorden van Grillitsch te zeggen: We hebben geen nood aan 'Management of Knowledge' maar aan 'Management for Knowledge'. De volgende factoren zijn volgens Grillitsch noodzakelijk om tot een succesvol projectkennisdeling te komen:

- Strategische relevantie van kennis voor de organisatie.
- Toewijding van het topmanagement.
- Zorgen voor de aanvaarding door de werknemers.
- De ontwikkeling van beloningssystemen.
- Een op kennis gerichte bedrijfscultuur.
- Continue ontwikkeling van de vaardigheden van het management en de werknemers.

- Een gebruiksvriendelijke IT-infrastructuur.
- Het promoten van interactie, communicatie en participatie van de werknemers.
- Doorstroming van kennis van buiten de organisatie naar binnen de organisatie.
- Het documenteren van 'lessons learned' en 'best practices'.
- Het evalueren van de effecten en de resultaten van de kennismanagement strategie.

Het principe van 'Management for Knowledge' bestaat erin dat, indien er een kennisdelende cultuur in een bedrijf gecreëerd kan worden, die bovendien sterk verankerd wordt in de structuur van de organisatie, deze cultuur continue het gebruik en de creatie van kennis zal beïnvloeden. Een kennisdelende bedrijfscultuur zal er automatisch toe leiden dat de relevante data omgezet worden in informatie en dat bepaalde kennis gegenereerd, geïntegreerd en gebruikt of juist geweerd zal worden. Het implementeren van een dergelijke cultuurverandering is geen eenvoudig proces.

De tweede stap in het succesvol verloop van de deling van projectkennis is de implementatie. Kennis management projecten zijn pas succesvol als ze een positieve invloed hebben op de dagdagelijkse werkzaamheden van de werknemers en beantwoorden aan de strategische doelen van de organisatie. Daarvoor is er eerst een duidelijk overzicht nodig van de huidige kennisactiva van het bedrijf. Het is uitermate belangrijk dat de veranderingen die het bedrijf wil doorvoeren gestructureerd verlopen en duidelijk gecommuniceerd worden met de werknemers. Deze transparantie zorgt ook voor minder onzekerheid en weerstand bij de werknemers.

Grillitsch pleit hier voor de invoering van wat zij een "knowledge-oriented project supervision" noemt. Dit is een gestructureerde werkwijze voor de creatie van project kennis. In de beginfase van het project wordt er een lijst bijgehouden van de problemen die voorkomen. Er wordt naar oorzaken gezocht voor deze problemen en de mogelijk oplossingen worden besproken. Daarna zal er een actieplan opgesteld en uitgevoerd worden. Vervolgens wordt er een evaluatie gemaakt van de gebruikte methode. Tot slot worden dan de ervaringen en de nieuw opgedane kennis besproken en gedocumenteerd. Deze kan dan later opnieuw gebruikt worden in andere, gelijkaardige projecten. Bijkomend is het ook belangrijk dat de kennis uit projecten waar werknemers niet aan deelnamen ook gedeeld wordt. Hiervoor stellen de auteurs de invoering van regelmatige vergaderingen voor, waarin de werknemers de ervaringen met hun projecten kunnen delen. De leider van zo een vergadering moet

een facilitator zijn, die de deelnemers aanmoedigt hun projectkennis maximaal te delen.

Een gevaar met kennisgebaseerde systemen is dat er te veel niet relevante informatie gestockeerd wordt of anderzijds dat werknemers vergeten essentiële aspecten op te slaan. Daarom vindt Grillitsch het noodzakelijk dat het management een checklist opstelt die werknemers gebruiken bij het documenteren van hun kennis. Hierdoor kunnen ze zich concentreren op het delen van de noodzakelijke kennis en zal het zoeken naar relevante kennis gemakkelijker worden.

Als derde en laatste stap van een succesvolle projectkennisdeling is er de institutionalisering. Zoals eerder vermeld moet er omtrent het implementeren van kennismanagement een duidelijke strategische visie aanwezig zijn die nauw samenhangt met de algemene strategische visie van het bedrijf. Het succes van de investering in kennis management zal mede bepaald worden door het bewustzijn en het vermogen van de organisatie om stap voor stap te veranderen overeenkomstig de huidige en toekomstige behoeften. Het uiteindelijke doel is de creatie van intelligente, zelf reflecterende en collaborerende werknemers die als 'change agents' kunnen functioneren in hun eigen werkomgeving. Ervaringen en potentiële oplossingen voor problemen moeten systematisch gedocumenteerd en gecommuniceerd worden. Om van alle lange termijn voordelen van een geïnstitutionaliseerde kennisdeling te genieten dient een organisatie aandacht te hebben voor de volgende factoren:

- Een continue ondersteuning van het top management.
- Het bestaan van een gestructureerde procedure voor vergaderingen.
- Het maken van evaluaties van de vergaderingen.
- Afhankelijk van de noden van de gebruikers en de organisatie de procedures voor vergaderingen en documentatie kunnen aanpassen.
- Het bestaan van duidelijke rollen en verantwoordelijkheden voor de vergaderingen en de documentatie.
- De training van interne moderators.
- Het verzamelen en documenteren van 'best practices' en 'lessons learned'.

Werknemers hebben tijd nodig om te kunnen reflecteren op hun ervaringen, te communiceren met hun collega's en om hun kennis te documenteren. Doordat werknemers moeten oefenen met de nieuwe systemen en gewenning noodzakelijk is met de werkmethode voordat routines opgebouwd worden, vergt de invoering van een

projectkennis management systeem een hoge initiële investering. Bij een goede implementatie zullen de baten echter vele malen groter zijn dan de kosten.

### **2.3.7. Communities of Practice**

In de literatuur wordt er ook vaak verwezen naar de 'communities of practice' benadering. Volgens deze benadering is kennis sociaal ingeburgerd in 'communities' en is daardoor onafscheidelijk verbonden met de 'practice' of de toepassing ervan. Deze verbondenheid wordt ook wel 'knowledge stickiness' genoemd.

Een 'community of practice' bestaat uit personen, die, door het uitoefenen van hun beroep, een eigen taal, vooroordelen, werkgewoonten en opvattingen over andere 'communities' en hun 'practices' hebben. Een 'community' of gemeenschap omvat divisies, functionele gebieden, productlijnen en project teams en deze verschillende gemeenschappen zijn verweven doorheen de verschillende niveaus van een organisatie. Een individu kan dus tegelijkertijd lid zijn van verschillende gemeenschappen.

Organisaties kunnen de affiniteit die werknemers voelen met hun gemeenschap gebruiken om kennisdeling te bevorderen. Door het opzetten van online forums per gemeenschap of het aanmoedigen van informele contacten tussen de verschillende leden van een gemeenschap wordt er aan de werknemers de mogelijkheid gegeven om kennis met elkaar te delen. Deze contactpunten vormen voor de werknemers nieuwe bronnen voor het vinden van informatie en moedigen hen aan om ervaringen met elkaar te delen.

Zo wordt de 'tacit' kennis die aanwezig is bij de werknemers beter toegankelijk voor andere werknemers die nood hebben aan deze kennis doordat de werknemers een gemeenschappelijke plaats hebben om kennis op te vragen en het duidelijker wordt wie welke kennis bezit. (Hayes & Walsham, 2003)

### **2.3.8. Leaving Expert Debriefing**

Het vertrek van een expert uit het team of uit het bedrijf kan grote gevolgen hebben op de kennis aanwezig in het team of de organisatie, omdat er mogelijk kennis verloren gaat. Een expert is meestal in bezit van 'tacit' kennis die enkel hij bezit. In teamverband is de expert soms in bezit van bepaalde sleutelkennis die moeilijk te vervangen is en de expert kan de enige zijn die weet waar bepaalde 'explicit' kennis opgeslagen is. Een bedrijf moet zich ook afvragen of het vervangen van de expert noodzakelijk is. Ze moeten nagaan of de kennis overgedragen moet worden naar een nieuwe medewerker of dat ze een andere manier moet vinden om competitief te blijven.

Josef Hofer-Alfeis (2008) geeft twee oplossingen voor het probleem van de 'leaving expert'. De introductie van een 'Leaving Expert Debriefing' helpt om de kennis van een expert op een gestructureerde manier achter te laten in het bedrijf. Een 'Leaving Expert Debriefing' bestaat uit verschillende stappen. Vooreerst moet het management de situatie en de motivatie voor het vertrek van de expert begrijpen en een kennis portfolio samenstellen met de verschillende domeinen waar de expert kennis over bezit.

De eerste stap van de 'Leaving Expert Debriefing' is het identificeren en beschrijven van de verschillende domeinen waar de expert bekwaam in is. De bedrijfskritische domeinen en de domeinen waar de expert exclusieve kennis over heeft, zijn uiteraard de belangrijkste. Daarnaast wordt er ook een kennisportfolio gemaakt van alle unieke kennis die de expert bezit en wordt er een kennisoverdracht actieplan opgesteld samen met expert en de opvolger.

De tweede stap is het maken van een relatiemap. Hier worden alle bedrijfskritische relaties van de expert op weergegeven. Extra informatie zoals de functie van de contactpersoon, de relatie met die persoon en hoe de communicatie zal moeten verlopen wanneer de expert vertrekt, geeft de opvolger een goed beeld van hoe het netwerk van de expert eruit ziet. Dit zal helpen bij het oplossen van de verschillende communicatieproblemen die zich kunnen voordoen wanneer een expert vertrekt.

De derde stap is het identificeren en beschrijven van bedrijfskritische 'explicit' kennisdocumenten die de expert gebruikte en die belangrijk zijn na zijn vertrek. De opvolger moet toegang hebben tot die documenten en alle documenten die mogelijk interessant zijn in de toekomst moeten worden opgeslagen.

De vierde stap is het documenteren van de 'lessons learned' en de andere adviezen van de expert.

De vijfde stap is de laatste stap en bestaat uit het samenstellen van de verschillende actiepunten uit de vorige stappen en ze een prioriteit toe te kennen. De kennis kan dan op een gestructureerde manier overgedragen worden. Na de overdracht moet geverifieerd worden dat alle kennis overgedragen is en dat de opvolger in staat is de taken uit te voeren.

Het kan interessant zijn om de 'Leaving Expert Debriefing' te institutionaliseren in het bedrijf. Enkele 'triggers' starten een 'Debriefing' op en opgeleide moderatoren en 'templates' zorgen voor het vlotte verloop van het volledige proces.

Een tweede oplossing is het beperken van de exclusieve kennis van een expert door een goed kennismanagement beleid. Hierdoor wordt bedrijfskritische informatie meer zichtbaar en beschikbaar voor iedereen die het nodig heeft. De hoeveelheid exclusieve kennis die de expert bezit wordt hierdoor beperkt en zorgt ervoor dat het bedrijf kan verder werken wanneer een expert plots weg valt.

Naast de kennisdeling zoals beschreven in de vorige onderdelen van dit hoofdstuk kunnen ook de volgende punten hierbij helpen:

- Het opzetten van een tandem collaboratie tussen de expert en de opvolger.
- Het samenstellen van teams met leden uit verschillende generaties.
- Het hanteren van modellen voor job rotatie en collaboratie.
- Het bevorderen van 'communities of practice' of gelijkaardige kennis netwerken.
- Het ter beschikking stellen van handboeken, richtlijnen en trainingen.



## **Hoofdstuk 3: Het duurzame ontwikkelingsproces**

### **3.1. Duurzaamheid**

'Sustainability' of duurzaamheid in een ontwikkelingsproces is een filosofie en omvat een aantal praktijken die ervoor zorgen dat een team een optimaal ontwikkelingsproces doorloopt en voor altijd behoudt.

Bij een niet duurzame ontwikkeling wordt er door het team hoofdzakelijk gereageerd op veranderingen die zich voordoen in de omgeving, in plaats van te anticiperen op mogelijke veranderingen. Hierdoor geraken de teams in een vicieuze cirkel van constant reageren op nieuwe signalen en hebben ze meer geld en tijd nodig om het systeem draaiende te houden.

Bij een duurzame ontwikkeling anticipeert het team op de toekomstige veranderingen. Dit doen ze door hun aandacht te vestigen op het werk dat voor de klant het meest waarde toevoegt en ervoor te zorgen dat de producten die ze afleveren betrouwbaar en van goede kwaliteit zijn. Ze streven ook naar een continue productverbetering ook al neemt de complexiteit toe. Bij duurzame ontwikkeling produceert het team op regelmatige basis software van hoge kwaliteit, hierbij gebruik makend van methoden die het project op lange termijn blijvend ondersteunen. Hierdoor kan het team de klant goede producten leveren, is het in staat om op de veranderende omgeving in te spelen en blijvend kwaliteitsvolle producten af te leveren.

Het team zal mogelijke veranderingen bestuderen en hierop trachten in te spelen. Wat vandaag goed werkt, kan morgen voorbijgestreefd zijn. Het team is zich hiervan bewust en beseft dat anticiperen op toekomstige evoluties zeer nuttig is voor het ontwikkelen van producten die langdurig onderhouden moeten worden in een snel veranderende omgeving.

Er bestaat geen set van gouden regels die zorgen voor een duurzame ontwikkeling. Elk bedrijf moet zelf bekijken hoe ze een duurzame ontwikkeling implementeren. De principes achter een duurzame ontwikkeling zijn belangrijker dan de praktijken die de werknemers moeten volgen. Enkel en alleen regels opstellen voor duurzame ontwikkelingen geeft de werknemers en het management de valse indruk dat die op zich zorgen voor duurzame producten. De regels kunnen nooit alle situaties en

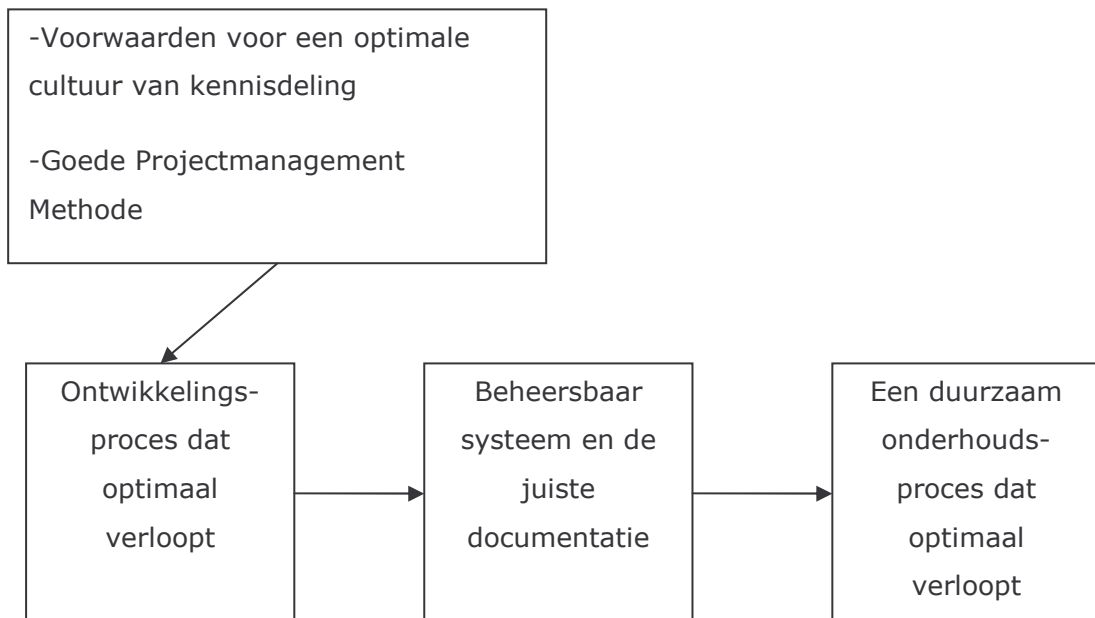
uitzonderingen bevatten en dus zijn de principes belangrijker. De filosofie voor duurzame ontwikkeling moet aanwezig zijn in de cultuur van de organisatie. Goede teams weten dat het resultaat belangrijker is dan de wijze waarop dit resultaat tot stand is gekomen. Dit wil niet zeggen dat het opstellen van goede werkmethodes niet kan helpen. Ze kunnen een handige leidraad zijn bij een duurzame ontwikkeling maar ze zijn zeker niet de enigste of de belangrijkste stap.

De principes van duurzame ontwikkeling zijn volgens Kevin Tate (2005):

- Een continue verbetering van het product en de projectmethodes nastreven.
- Er moet altijd een werkend product beschikbaar zijn.
- Het bestaan van een continue investering in ontwikkelingsprocessen.
- Foutenpreventie moet belangrijker zijn dan foutendetectie.

De cultuur omtrent duurzame ontwikkeling is dus zeer belangrijk maar een cultuurverandering is niet gemakkelijk te bekomen. Bedrijven moeten dan ook actief werken om de cultuur te veranderen en onderhouden. (Tate, 2005)

Voor de realisatie van een duurzaam ontwikkelingsproject ga ik in deze thesis uit van het volgende schema:



De beginvoorwaarde is het bestaan van een optimale bedrijfscultuur voor het aanmoedigen van kennisdeling. Hiervoor dienen alle componenten besproken in het vorige hoofdstuk aanwezig te zijn. Bij een positieve kennisdelingscultuur zal ook het software ontwikkelingsproces optimaal verlopen. Dit zorgt op zijn beurt voor beheersbare systemen die logisch opgebouwd zijn en op een eenvoudige wijze aangepast kunnen worden. Daarnaast zorgt het ook voor een correct gedocumenteerd systeem, met een goede weergave van de relevante kennis die eveneens gemakkelijk terug te vinden zal zijn. Dit alles zorgt er dan uiteindelijk voor dat het latere onderhoudsproces optimaal verloopt.

Ook het belang van een goede projectmanagement methode is weergegeven in bovenstaand schema. In de volgende paragrafen worden dan ook een aantal methodes voor software ontwikkeling en project management nader toegelicht.

## **3.2. Software ontwikkelingsmethodes**

### **3.2.1. Waterval model**

Het traditionele model voor software ontwikkeling is het waterval model. In dit model worden de verschillende stappen voor de ontwikkeling van nieuwe software sequentieel doorlopen en afgerond. De zes verschillende stappen zijn:

1. Initiatie. Dit is de start fase van een ontwikkelingsproject. Deze fase omvat het uitvoeren van haalbaarheidsstudies en het opstellen van een project plan.
2. Analyse. In deze fase wordt de omgeving waarin het product moet werken geanalyseerd en beschreven. De vereisten voor het product worden vastgelegd aan de hand van bestaande documentatie en gesprekken met de gebruikers.
3. Ontwerp. Hier wordt een antwoord gegeven op de vraag "Hoe zal het product aan de vereisten voldoen?" Er wordt een evaluatie gemaakt van de mogelijke alternatieven voor de creatie en integratie van de nieuwe systeemcomponenten.
4. Ontwikkeling. In deze stap wordt het ontwerp geïmplementeerd, dit wil zeggen dat het product wordt geprogrammeerd en geconfigureerd.
5. Implementatie. Tijdens de implementatie wordt het ontwikkelde product getest en geïnstalleerd. Indien het product ontwikkeld werd ter vervanging van een bestaand product, zal tijdens deze fase ook de conversie van het oude naar het nieuwe product gepland en uitgevoerd worden.
6. Onderhoud. Tijdens deze fase wordt het systeem onderhouden en worden nieuwe aanpassingen aangebracht aan het systeem.

Volgens Dave Chaffey en Steve Wood (2005) werden in de beginperiode van de software ontwikkeling deze stappen serieel doorlopen, zonder veel overlap tussen de verschillende fasen. Waardoor de benaming "waterval model" ontstond. Elke fase nam maanden tot jaren in beslag en werd niet geëvalueerd vooraleer de fase afgerond was en er overgegaan werd naar de volgende stap. Dit zorgde voor verschillende problemen:

1. Weinig flexibiliteit voor veranderende vereisten. Eens de vereisten vastgelegd waren in de analysefase konden ze niet meer gewijzigd worden. Nieuwe vereisten konden pas geïmplementeerd worden in de onderhoudsfase aan een

hoge kost of er diende een volledig nieuw ontwikkelproject opgestart te worden om de aanpassingen te incorporeren.

2. Scheiding tussen het ontwikkelingsteam en de gebruikers in de business. De communicatie tussen het ontwikkelingsteam en de gebruikers was beperkt tot de initiatie-, de analyse- en de implementatiefase. Tussentijds was er geen enkele vorm van communicatie en werkten de twee partijen onafhankelijk van elkaar, voor maanden of zelfs jaren, tot het moment dat het ontwikkelingsteam het afgewerkte product aan de gebruikers kon tonen. Er was meestal geen verduidelijking van de vereisten mogelijk door de gebruikers en het ontwikkelingsteam stelde ook geen mogelijke verbeteringen aan het systeem voor.
3. Lange ontwikkelingstijden. Doordat er geen communicatie was tussen de gebruikers en het ontwikkelingsteam, bestond er ook geen gevoel van urgentie bij het ontwikkelingsteam. Erger nog, wanneer het systeem geëvalueerd werd door de eindgebruikers tijdens de implementatie fase, werden er vaak grote tekortkomingen vastgesteld en diende men, voor de noodzakelijke aanpassingen aan het product, weer de verschillende stappen te doorlopen.

Al deze tekortkomingen zorgden ervoor dat het waterval model thans niet meer gebruikt wordt en er overgaan werd naar nieuwere methoden voor software ontwikkeling. Interessant is ook om op te merken dat volgens Conrad Weisert (2003) het waterval model nooit bestaan heeft en het enkel in de literatuur beschreven is als voorbeeld van hoe het niet moet. Geen enkel artikel heeft ooit het waterval model verdedigd of het voorgesteld als het ideale model. Het wordt enkel gebruikt om het nut van nieuwe methoden te verdedigen door het in contrast te stellen met dit fictieve model. De term "waterval" verwijst ook naar een inflexibele structuur en als iemand het al uitgevonden zou hebben, zou hij er waarschijnlijk ook wel een andere naam aan gegeven hebben.

### **3.2.2. Agile software ontwikkelingsmethode**

Tegenwoordig wordt er veel gesproken over 'Agile software development'. 'Agile' is gebaseerd op een groep van iteratieve software ontwikkelingsmethoden, waarbij de systeemvereisten en oplossingen samen evolueren door de inzet van crossfunctionele teams. De term 'Agile software development' werd voor het eerst gebruikt in het 'Agile Manifesto'. In dit manifest vatten de auteurs van verschillende nieuwe software

ontwikkelingsmethoden de principes samen die aan de grond liggen van 'Agile software development'.

'Agile' wil meer waarde hechten aan:

- Interacties tussen individuen dan aan processen en tools;
- Werkende software dan uitgebreide documentatie;
- Collaboratie met klanten dan contract negotiatie;
- Het reageren op verandering dan het volgen van een plan.

De auteurs van het 'Agile Manifesto' (Beck, et. al., 2001) erkennen dat de opgesomde elementen aan de rechterkant waardevol kunnen zijn, maar ze hechten beduidend meer waarde aan de elementen aan de linkerkant. Onderliggend aan dit manifest zijn de volgende principes van belang:

1. De hoogste prioriteit bestaat erin te voldoen aan de behoeften van de klant door het snel en continue leveren van waardevolle software.
2. Openstaan voor veranderende vereisten van de klant, zelfs laat in het ontwikkelingsproces. 'Agile' processen kunnen omgaan met veranderingen waardoor gebruikers een competitief voordeel hebben.
3. Lever regelmatig werkende software af. Hierbij worden termijnen van een paar weken tot een paar maanden vooropgesteld, met een voorkeur voor kortere termijnen.
4. Eindgebruikers en de software ontwikkelaars werken dagelijks samen gedurende het volledige project.
5. Bouw projecten rond gemotiveerde individuen. Geef hen de omgeving en de ondersteuning die ze nodig hebben en vertrouw hen het klaren van de klus toe.
6. De meest efficiënte en effectieve manier om informatie door te geven aan en binnen het ontwikkelingsteam is door persoonlijke conversaties, en niet via telefoon of e-mail.
7. Werkende software is de belangrijkste maatstaf van vooruitgang.
8. 'Agile' processen bevorderen 'sustainable development.' De sponsors, ontwikkelaars en eindgebruikers kunnen hierdoor samen een constant tempo van software ontwikkeling aanhouden.
9. Een continue aandacht voor technische uitmuntendheid en goede ontwerpen verhoogt de 'agility' van de ontwikkeling.

10. Eenvoud is essentieel. Eenvoud is "the art of maximizing the amount of work not done."
11. De beste architecturen, systeemvereisten en ontwerpen ontstaan uit 'self-organising' teams.
12. Op vaste tijdstippen moet het team evalueren hoe het meer effectief kan worden en past het zijn gedrag daarna aan.

Volgens Kevin Tate (2005) berust 'Agile software development' op drie peilers. Deze zijn project management, samenwerking en technische uitmuntendheid. Elk van deze peilers is een stimulans voor de 'agility' van het ontwikkelingsproces. Als één van deze drie peilers niet aanwezig is dan zal het volledige 'agile' software ontwikkelingsproces afgeremd worden. Als een team 'agile' project management methoden toepast maar geen 'agile' technische uitmuntendheid methoden nastreeft of als er weinig of geen samenwerking binnen het projectteam bestaat, zal dat de mogelijkheid van het team om 'agile' te zijn, sterk beperken.

Ook al is project management een van de drie peilers van 'agile software development', toch wordt er bij vele van de 'agile' methoden nauwelijks gesproken over hoe deze ingericht moet worden. (Abrahamsson et al., 2003) De Agile Alliance heeft zelf een 'white paper' geschreven waarin ze het gebruik van PRINCE2 in samenwerking met 'agile software development' aanraadt. (DSDM consortium 2009) Deze project management methodologie wordt in de volgende paragraaf beschreven.

### **3.3. PRINCE2 projectmanagement methode**

PRINCE (PROjects IN Controlled Environments) is een gestructureerde methode voor effectief project management. De methode werd voor het eerst ontwikkeld door het Central Computer and Telecommunication Agency in 1989. PRINCE2 werd in 1996 gelanceerd en hield rekening met de input van vele project teams en projectleiders op PRINCE. Tegenwoordig is PRINCE2 een standaard geworden die in vele organisaties gebruikt wordt.

PRINCE2 definieert een project als:

*'a management environment that is created for the purpose of delivering one or more business products according to a specified Business Case'*

Daarnaast veronderstelt PRINCE2 dat diegenen die verantwoordelijk zijn voor de realisatie van het project nog niet eerder hebben samengewerkt, dat deze samenwerking dus goed georganiseerd en gecoördineerd moet worden en dat de verantwoordelijkheden gedeeld worden door de teamleden, de projectmanagers en de sponsers van het project.

Een PRINCE2 project heeft de volgende karakteristieken:

- Een eindigende levensloop.
- Een aantal welbepaalde en meetbare eindproducten.
- Een set activiteiten om deze eindproducten te realiseren.
- Een welbepaalde groep middelen om deze eindproducten te realiseren.
- Een organisatie structuur, met duidelijke verantwoordelijkheden, om het project te managen.

Elk project valt binnen een specifieke business context. Een project kan op zichzelf bestaan, het kan een onderdeel zijn van een sequentie van aan elkaar gerelateerde projecten of het kan een onderdeel zijn van een programma of een bedrijfsstrategie.

Een project is een tijdelijke structuur die opgezet wordt om een specifiek business voordeel of een objectief te behalen. Wanneer het werkt voltooid is, wordt het project afgesloten. Een project heeft een levenscyclus, een levensloop en een opeenvolging van verschillende activiteiten om het finale product te verkrijgen.



PRINCE2 bestaat uit acht processen die alle activiteiten van een project omvatten. Elk project dat via PRINCE2 gemanaged wordt, zal deze processen in één of andere vorm aanwenden. De sleutel tot succes bestaat erin om deze processen op de juiste manier aan te passen aan de noden van het project. De acht processen zijn de volgende:

- Start van een Project, het eerste proces in PRINCE2. Het is een proces dat voor de eigenlijke start van het project plaatsvindt en ervoor zorgt dat alle voorwaarden om het project te kunnen starten vervuld zijn. Dit proces zou zeer snel moeten verlopen en bestaat uit het aanstellen van het projectteam, evenals het creëren van enkele documenten, zoals, een korte beschrijving van het project, de vooropgestelde aanpak, de kwaliteitsvoorwaarden zoals meegedeeld door de klant, het initiële risico logboek en een plan voor de initiatiefase.
- Leiden van een project. Dit proces loopt continue door vanaf het proces 'start van het project' tot het project beëindigd wordt via het proces 'afsluiten van een project'. Dit proces is bedoeld voor de project board, een groep mensen die de management beslissingen van het project op zich nemen. Ze volgen het project op via de tussentijdse verslagen en controleren de voortgang van het project op enkele vooraf vastgelegde beslissingsmomenten.
- Initialisatie van een project. Tijdens dit proces start de uitwerking van het project. Dit proces heeft als doel het opleveren van een gedocumenteerde en geaccepteerde business case, het opstellen van het projectplan met alle benodigde kosten, het nagaan of de investeringen van tijd en geld gerechtvaardigd zijn rekening houdend met de risico's van het project, het aanmoedigen de project board om zijn verantwoordelijkheid voor het project op te nemen en een consensus bereiken aangaande de middelen die nodig zullen zijn voor de volgende stap. Een belangrijk eindproduct van dit proces is het 'Project Initiation Document' dat de wat, waarom, wie, wanneer en hoe van het project beschrijft.
- Managen van projectfase overgangen. In dit proces worden alle overgangen van de ene naar de volgende projectfase afgebakend. Bij een fase overgang wordt het eindverslag van de fase overgemaakt aan de Project Board, die hiermee beslissingen kan nemen om het verdere verloop van het project te plannen of bij te sturen. Er wordt indien nodig een update gemaakt van het projectplan en het 'lessons learned' document wordt bijgewerkt.

- Controle van een fase. Dit proces beschrijft de verantwoordelijkheden van de projectmanager met betrekking tot het toezicht en de controle op een projectfase. De basistaak van de projectmanager bestaat er immers in dat het project op koers blijft, dat het project tijdig en volgens de afgesproken vereisten wordt opgeleverd binnen de vooropgestelde kost en dat er gepast gereageerd wordt op onvoorziene omstandigheden.
- Managen van de oplevering van het product. Een project resulteert in de oplevering van een aantal producten. Voor de realisatie van deze producten wordt het project één of meerdere 'work packages', een werkeenheid die een aantal producten oplevert. De uitwerking van een 'work package' wordt meestal toevertrouwd aan een teammanager. Dit proces beschrijft op welke wijze de teammanager de details en de voortgang van het 'work package' op regelmatige basis met de project manager dient te overlopen, met speciale aandacht voor de kwaliteitseisen van het afgeleverde werk. Het doel van dit proces is een goedkeuring te krijgen voor de afgeleverde producten.
- Afsluiten van een project. Het gecontroleerd afsluiten van een project is zeer belangrijk voor PRINCE2. Tijdens dit proces zal de Project manager alle nodige input verzamelen en de documenten voorbereiden die de Project Board nodig heeft om een gefundeerde beslissing over het al dan niet afsluiten van het project te nemen. Dit proces gaat na in welke mate de initiële doelstellingen behaald zijn, of de klant akkoord gaat met het opgeleverde product en of de middelen voor de verdere werking en het verdere onderhoud van het product beschikbaar zijn. Verder geeft dit proces aanbevelingen voor andere projecten, finaliseert het het "lessons learned" document, levert het het eindrapport van het project af, archiveert het de projectbestanden en meldt het aan de organisatie dat het project beëindigd is en dat de middelen terug beschikbaar zijn.
- Plannen. Dit proces wordt iteratief uitgevoerd parallel met de andere processen. Naast een projectplan zal dit proces initieel ook een lijst van op te leveren producten en een risico register opleveren, documenten die eventueel geupdated dienen te worden tijdens het verdere verloop van het project.

Dit is één manier om naar PRINCE2 te kijken. Henry Portman (2009) benadert PRINCE2 via de documenten die opgemaakt moeten worden, wat resulteert in een meer abstracte kijk op het verloop van een project met PRINCE2.

Zo zorgt de opeenvolging van processen voor een aaneensluitend geheel van documentatie. Processen die in een later stadium in het project plaatsvinden, maken gebruik van documentatie die opgeleverd werd tijdens voorafgaande processen in het project. Documenten zijn het resultaat van en de input voor de PRINCE2 processen.

PRINCE2 beschrijft de volgende documenten:

- D1: Mandate. Dit is de input voor het proces 'start van een project.' Het mandaat geeft een korte beschrijving van de redenen en de objectieven van het project en bevat een lijst van de producten die afgeleverd moeten worden.
- D2: Business Case. Dit document beschrijft de redenen voor de start van het project en de verwachte impact op de bedrijfsresultaten na de realisatie van het project. Het presenteert dus de verwachte kosten en baten. Tijdens het project wordt dit document regelmatig geëvalueerd en aangepast wanneer nodig.
- D3: Project Brief. Dit is een beschrijving van het voorgestelde project in termen van de business case, de initiële risico's en de in te zetten middelen. De goedkeuring van dit document geldt als startpunt voor het proces 'initialisatie van het project'.
- D4: Project Initiation Document. Dit document beschrijft alle relevante informatie voor de belanghebbenden van het project en geeft een beschrijving van de op te leveren producten, de levertermijnen, de nodige middelen en materialen en de in te zetten personen. Daarnaast geeft het ook een overzicht van het benodigde budget en beschrijft het de kwaliteitseisen waaraan de op te leveren producten dienen te voldoen. In geval van veranderingen tijdens het verloop van het project, zal dit document de basis vormen voor de beslissing of het project al dan niet verder uitgevoerd wordt.
- D5: Work package. Dit document bevat alle nodige informatie voor de ontwikkeling van één of meerdere producten van het project, inclusief de bijhorende product beschrijvingen.
- D6: Status Report. Dit document beschrijft hoe het project evolueert in vergelijking met het projectplan en het budget en is bedoeld voor de Project Board.
- D7: Project End Report. Dit document wordt opgemaakt op het einde van het project. Het bevat een verslag van de opgeleverde producten. Daarnaast worden het originele plan, het budget en de kwaliteitseisen, zoals beschreven

in D4, vergeleken met de werkelijkheid en wordt er een aangepaste business case opgesteld. Ten slotte bevat dit document ook aanbevelingen over hoe men moet omgaan met onopgeloste factoren.

- D8: Exceptions Report. Dit document beschrijft alle uitzonderlijke maatregelen die tijdens het project werden genomen en moet het in de toekomst mogelijk maken om beter te reageren op veranderende omstandigheden.
- D9: 'Lessons Learned' Report. Dit document bevat alle lessen die men geleerd heeft tijdens de uitvoering van dit project. Het beschrijft wat er goed ging en wat er minder goed ging en wordt gebruikt als input bij de start van toekomstige projecten om niet in dezelfde fouten te vervallen.
- D10: Issue Register. Dit is een lijst met alle problemen die men vastgesteld heeft tijdens het project. Elk probleem wordt beschreven en er wordt vermeld welke beslissingen het team reeds genomen heeft om het probleem op te lossen en wat de huidige staat is van het probleem.
- D11: Risk Register. Dit document wordt continu uitgebreid tijdens het project en omvat een opsomming van alle mogelijke risico's voor het project met voor elk risico een beschrijving van zijn mogelijke impact, de waarschijnlijkheid van voorkomen, de verantwoordelijke voor het beheer van het risico en de maatregelen die moeten genomen worden als het risico zich effectief zou voordoen.
- D12: 'Lessons Learned' Register. Dit is een continu evoluerend document waarin alle 'lessons learned' opgenomen worden tijdens het uitwerken van het project.

PRINCE2 stelt zelf geen 'templates' voor voor deze documenten, maar er zijn veel websites die dergelijke 'templates' aanbieden. Henry Portman is echter geen fan van deze aanpak, omdat volgens hem 'templates' zorgen voor documenten die op zichzelf staand zijn, die niet beknopt zijn, die ruimte laten voor interpretatie, die arbeidsintensief zijn om te vervolledigen, die lastig zijn om te lezen en die eerder gericht zijn op formele verantwoording of rapportage. Als het mogelijk is om deze karakteristieken te vermijden zal het voor betere projecten zorgen. Het moet de bedoeling zijn dat de projectmanager zich niet verliest in het documenteren van het project, zodat hij zijn prioriteit kan maken van het managen van het project.

De auteur verkiest het concept van 'building blocks' voor het opstellen van projectdocumentatie omdat volgens hem vele delen of bouwstenen van documenten

herhaald worden. Elke bouwsteen is een onafhankelijk subdivisie van een document en kan in een of meerdere projectdocumenten gebruikt worden. Deze opdeling van documenten in bouwstenen zorgt voor een kleinere kans op miscommunicatie of verlies van informatie, wat uiteindelijk ook resulteert in een betere motivatie van de teamleden.

De voorwaarden voor kwaliteitsvolle PRINCE2 documenten kunnen dan als volgt samengevat worden:

- **Standaardisatie.** Zorg ervoor dat vergelijkbare documenten voor projecten op eenzelfde manier opgebouwd worden.
- **Essentie.** Schrijf geen uitgebreide passages tekst maar alleen wat nodig is voor een goed verloop van het project.
- **Visualisatie.** Maak zoveel mogelijk gebruik van foto's, diagrammen en grafieken.

Het resultaat van deze aanpak is dat de 'project board' alle informatie op een herkenbare en overzichtelijke manier gepresenteerd krijgt. De 'project board' leden zullen hierdoor minder werk moeten besteden aan het ontrafelen van de informatie en meer tijd vrij krijgen voor het bestuderen van het project en het nemen van beslissingen. Daarnaast zal ook de projectmanager tijd besparen omdat hij alles maar één keer moet beschrijven, zodat ook hij zijn tijd efficiënter en effectiever kan gebruiken.

Deze aanpak heeft Henry Portman getest bij enkele ondernemingen. Zo werden bijvoorbeeld bij een 'financial service provider' in totaal 25 bouwstenen gedefinieerd voor het samenstellen van al hun documenten. Elke bouwsteen had zijn eigen template, maar niet elke bouwsteen was noodzakelijk voor elk project. Als een bouwsteen niet relevant was voor een bepaald project werd deze genegeerd en niet ingevuld. Hun bevinding was dat standaardisatie, door het gebruik van bouwstenen, een grote tijdswinst opleverde. Het management kon zeer snel zien hoe het gesteld was met een bepaald project en kon snel beslissingen nemen. Ook andere bedrijven ondervinden de voordelen van deze werkwijze. (Portman, 2009)

## **3.4. Documentatie in een duurzaam en agile ontwikkelingsproces**

### **3.4.1. Documentatie van het project**

Volgens Simon Wallace (2007) is de productie en controle van documentatie een deel van elk project. Er zijn vele soorten documentatie met verschillende doeleinden en levenscycli. Uiteraard is de meeste documentatie tegenwoordig niet meer in papieren vorm en documentatie kan zelfs helemaal niet meer op een conventioneel document lijken. De huidige technologie zorgt ervoor dat documentatie op een heel snelle en efficiënte manier geraadpleegd en gedeeld kan worden. Project documentatie management is het waarborgen dat alle gewenste documentatie eenvoudig opgeslagen kan worden door de projectmedewerkers en dat deze documentatie daarna eenvoudig opgevraagd en eventueel geupdated kan worden door de belanghebbenden van het project.

Er bestaan vier verschillende scenario's voor projectdocumentatie in termen van levensloop en gebruik. Ten eerste kan de documentatie permanent of tijdelijk bedoeld zijn en ten tweede kan documentatie voor externe klanten of voor intern gebruik opgesteld worden. Documentatie voor externe klanten moet uiteraard aan hoge kwaliteitseisen voldoen, terwijl interne documentatie eerder informeel is en niet compleet hoeft te zijn. Permanente documentatie moet aangepast kunnen worden als er zich wijzigingen voordoen, terwijl tijdelijke informatie onveranderd mag blijven.

Bij een verdere verbetering van de technische oplossingen, wordt het steeds interessanter om de documentatie door de programma's zelf te laten aanmaken. Zodoende dient er geen apart document meer aangemaakt, opgeslagen, opgevraagd en gelezen te worden. Al de nodige informatie is al aanwezig in de component zelf. Deze manier van werken is nuttig voor:

- Analyse, ontwerp en ontwikkeling tools: deze moeten zelf documenterend zijn.
- Ontwikkelingsstandaarden voor broncode: deze moeten gemakkelijk verstaanbaar zijn.
- Gebruikers procedures: deze kunnen gevisualiseerd worden aan de hand van 'workflow diagrams.'
- Gebruikers handleidingen en hulp informatie: deze kan geïncorporeerd en als contextgevoelige, online informatie beschikbaar gesteld worden.

- Trainingsmaterialen: deze kunnen ontwikkeld worden voor elektronische zelfstudie.

Traditioneel bestond IT documentatie maar uit één dimensie. Per grote stap in de realisatie van een project werd er een document geproduceerd waarin alles beschreven stond. Tegenwoordig wordt er voor het opstellen van de project documentatie gebruik gemaakt van een matrix benadering, waarbij er minstens twee dimensies zijn. Deze matrix toont de verschillende types kennis, tijdens de verschillende stadia van een project. Zo zullen er business vereisten opgesteld worden per business proces dat tijdens het project aangepakt wordt, met telkens ook de aanduiding van de verantwoordelijke personen.

De meeste gebruikers hebben enkel interesse in een bepaald deel van het project en willen daarom niet doorheen de volledige documentatie worstelen. Op een bepaald moment zijn ze waarschijnlijk maar geïnteresseerd in een paar cellen van de matrix. Als de documentatie opgebouwd is in een gestructureerde manier met een goede index en opzoekmogelijkheden zal de gebruiker de informatie die hij nodig heeft gemakkelijk kunnen vinden.

Nog een belangrijk voordeel van de matrix aanpak is dat informatie en documentatie in delen vrijgegeven kunnen worden voor herziening, goedkeuring of een andere actie zonder dat er gewacht dient te worden op andere documentatie van het project. Het resultaat van deze matrix aanpak is dat:

- Eindgebruikers en projectteamleden enkel met de documentatie werken die relevant is voor hen.
- Ze deze documentatie voorgeschoteld krijgen in beheersbare delen.
- Ze niet moeten wachten op ongerelateerde inhoud om de documenten te kunnen raadplegen.
- Door de documentatie enkel aan te bieden aan de bevoegde personen of de personen die daadwerkelijk in het project geïnteresseerd zijn, zorgt ervoor dat de complexiteit van het gebruik van het documentatiesysteem sterk verminderd.
- Er minder pieken en dalen in de werklust voorkomen omdat de verschillende documentatiedelen op verschillende momenten toekomen.

Bij de start van een project moet er een duidelijk beeld geschetst worden van het eindproduct. Maar het is dus niet nodig om in dit eerste document reeds alle projectdetails te willen opsommen. Wat belangrijk is, is dat er vastgelegd wordt hoe de documentatie beheerd en gecontroleerd zal worden tijdens het verloop van het project. In de meeste gevallen zal er een systeem opgezet worden dat het beheer van de op te leveren documenten kan vergemakkelijkt. Dit kan gaan van een simpele spreadsheet voor kleinere projecten tot een volwaardig document management systeem voor grote projecten.

Na de gedetailleerde planning van een projectfase zijn de nodige data beschikbaar om het documentatie management van die fase uit te werken. Hierbij wordt voor de verschillende documenten bepaald welke belangrijke details aanwezig moeten zijn, in welk formaat het document moet opgeleverd worden, wie de verantwoordelijken zijn, hoe het document dient te circuleren en wat het niveau van kwaliteitsbeoordeling zal zijn. Waar mogelijk zal de project manager 'templates' beschikbaar stellen die de teamleden helpen om op een correcte manier de documentatie te genereren.

Tijdens de uitvoering van het project moet de werking van het document management systeem zo eenvoudig en efficiënt mogelijk verlopen, zonder uit het oog te verliezen dat controle en verificatie van documentatie belangrijk blijft. Individuele gebruikers moeten in staat zijn documenten uit te lenen, te wijzigen en terug te plaatsen in het systeem, rekening houdend met de regels voor verantwoordelijkheden, machtigingen en controles. Een goed werkend document management systeem zal de project manager in staat stellen om de staat van de documentatie nauwlettend in het oog te houden.

Aan het einde van een projectfase moeten alle documenten in de repository gecontroleerd worden door de project manager om na te gaan of alle documenten aan de kwaliteitsverwachtingen voldoen. Deze goedgekeurde documenten zijn dan beschikbaar als input voor de volgende projectfase. Aan het einde van het project worden alle documenten in de repository overlopen en, afhankelijk van hun aard, worden er bepaalde acties ondernomen. De tijdelijke documenten worden waarschijnlijk gearhiveerd, 'just in case'. Definitieve, interne documentatie wordt bijgehouden voor raadpleging en eventuele updates tijdens het onderhoud van het product. Hiervoor zal de project manager, voordat het project volledig is afgerond, een verantwoordelijke voor het onderhoud van elk document aanduiden. De externe



documentatie voor de klanten moet verdeeld en onderhouden worden. (Simon Wallace, 2007)

Volgens Luckey en Phillips (2006) zien vele project managers er tegenop om de documentatie van hun project op te stellen, nochtans is documentatie zeer belangrijk. Het documenteren van de projectplannen is belangrijk, maar nog belangrijker is het bijhouden van wat er juist gebeurd is tijdens het project. Deze informatie wordt bijgehouden in een 'lessons learned' rapport en kan nuttige zijn voor toekomstige projecten. De dingen die nu logisch lijken, zijn dat binnen enkele jaren misschien niet meer, wanneer er nieuwe producten ontwikkeld zullen worden of updates aan bestaande producten moeten gebeuren. Immers, tijdens die periode kan er heel wat kennis verloren gegaan zijn doordat teamleden het bedrijf verlaten hebben, op pensioen gegaan zijn of overgeplaatst werden naar andere departementen.

Omdat een project het werk is van vele verschillende mensen is het noodzakelijk om iedereen te betrekken bij het opstellen van de documentatie. Specifieke taken en situaties vereisen een specifieke aanpak en daarom kunnen de teamleden best ieder voor zich hun eigen 'lessons learned' rapport opschrijven. Project managers documenteren het beheer van het project en systeemanalisten beschrijven hun ervaringen tijdens de analyses. Dit zal er ook voor zorgen dat werknemers die deze documenten in de toekomst raadplegen een persoonlijkere en daarom interessantere visie op het project te lezen krijgen.

Er zijn twee belangrijke manieren om tot betere 'lessons learned' documenten te komen. Ten eerste is het belangrijk om vergaderingen te organiseren betreffende dit document zodat ieder lid van het projectteam bekend is met de inhoud ervan en ook om iedereen aan te moedigen om zowel positieve als negatieve ervaringen, alsook om mogelijke problemen te bespreken en te documenteren. Ook na het vinden van oplossingen van problemen kan de project manager best eens vragen wat iedereen hieruit geleerd heeft. Dit kritisch denken zal de hoeveelheid en kwaliteit van de 'lessons learned' documentatie verbeteren. Ten tweede is het ook belangrijk om mijlpalen te zetten voor de oplevering van de documentatie. De projectmanager moet ervoor zorgen dat zijn teamleden voldoende gemotiveerd zijn zorgen voor het opstellen van de vereiste documentatie en dient het daarom als een belangrijk onderdeel van het project behandelen.

Het is mogelijk dat teamleden weerstand bieden voor het opstellen van een 'lessons learned' document omdat ze hun kennis niet willen delen uit angst om overbodig gevonden te worden binnen het projectteam. De auteur suggereert om in die situaties de individuen te verzekeren dat de informatie die ze vrijgeven enkel toont hoeveel ze weten en hoe onmisbaar ze zijn, niet andersom.

De input voor elk nieuw project zouden de 'lessons learned' documenten van vorige gelijkaardige projecten moeten zijn. De output van elk project is een 'lessons learned' document voor toekomstige projecten. Luckey en Phillips (2006) beschouwen dit document dan ook als het belangrijkste document van een project. Het is het geschreven bewijs van het project en het vertelt het verhaal over hoe het project tot stand is gekomen en wat men geleerd heeft door het project te realiseren. Of het project op lange termijn belangrijk is of niet, zonder dit document bestaat er een reële kans dat het project vergeten wordt. Het is de kans voor de projectmanager om het bedrijfsmanagement te tonen wat hij verwezenlijkt heeft en het spreekt voor zich dat het de organisatie in de toekomst veel tijd en geld kan besparen. Want elk project, zelfs al wordt het geannuleerd in een vroeg stadium, kan iets bijleren aan de organisatie.

Dit document moet ook aan het begin van het project geïnitieerd worden zodat alle projectervaringen direct bijgehouden kunnen worden. Hierdoor kan je ook aan het team tonen dat het acceptabel is om vergissingen te maken zolang je maar leert uit de fouten en de nieuwe kennis gebruikt om betere beslissingen te nemen in de toekomst. Dit laat werknemers toe om berekende risico's te nemen als ze beseffen dat fouten ook gelegenheden zijn om bij te leren.

Een 'lessons learned' document moet goed vermelden wie aan het project heeft meegewerkt en wie dit document heeft samengesteld. Dan zijn toekomstige teamleden in staat om bijkomende informatie te vragen als deze personen nog op het bedrijf werken.

Een opsomming van de problemen en mislukkingen en hun oplossingen zijn een groot deel van het 'lessons learned' document. Het laat toe om toekomstige teamleden van andere projecten om niet dezelfde fouten te maken. Daarom is het ook belangrijk om de opeenvolging van gebeurtenissen te beschrijven die leidde tot de fout zodat er patronen ontdekt kunnen worden en fouten minder vaak gemaakt zullen worden. Maar naast de opsomming van de problemen is het ook zeer belangrijk om de

succesverhalen te vertellen. Het projectteam mag er trots op zijn als ze iets innovatief verwezenlijkt hebben en dit moet dan ook even gedetailleerd opgeschreven worden als de mislukkingen, zonder ze te kleineren of toe te schrijven aan geluk. De meeste werknemers beschouwen succes als iets voor de hand liggend maar toch kan je evenveel leren uit de successen als uit de mislukkingen. Succesverhalen, zoals een nieuwe tool die werd geïntroduceerd, een nieuwe aanpak voor teamvergaderingen, een bepaalde manier van communiceren met de gebruikers, enz. kunnen ook een belangrijke leerfactor zijn voor de gehele organisatie.

Tot slot is het aanbevolen om een betere projectaanpak voor te stellen. Hierbij geven de projectmanager en de teamleden aan hoe ze het project zouden aanpakken als ze het helemaal opnieuw zouden doen. Dit is vaak zeer nuttige informatie voor de toekomstige projecten. (Luckey en Phillips, 2006)

### **3.4.2. Documentatie van het systeem**

De 'agile software development' methode is niet echt geïnteresseerd in absoluut perfecte documentatie. Het Agile Manifesto hecht immers meer waarde aan werkende software dan aan uitgebreide documentatie. Daarnaast toont Scott W. Ambler (2009) ook aan dat er geen relatie bestaat tussen de geproduceerde documentatie en het succes van het project, erger nog, hoe meer documentatie dat men schrijft hoe groter de kans dat het project faalt. Dit wil echter niet zeggen dat er helemaal geen documentatie geschreven dient te worden. Documentatie blijft belangrijk maar er is een 'agile' manier om deze documentatie te creëren. Volgens Scott W. Ambler is het belangrijk dat de projectleiders de noodzaak van goede documentatie niet uit het oog verliezen. Volgens hem zijn er vier belangrijke observaties die gemaakt moeten worden:

1. Documentatie is een even groot deel van het systeem als de broncode. Naast werkende software moet het software ontwikkelingsteam waarschijnlijk ook handleidingen voor de eindgebruikers, support documentatie, operationele documentatie en globale systeem documentatie opleveren.
2. Het eerste doel van het projectteam moet het afleveren van werkende software zijn en het tweede doel van het team bestaat erin om het onderhoud van de software mogelijk te maken. Het is uiteraard belangrijk om werkende software van hoge kwaliteit af te leveren die voldoet aan de eisen van de belanghebbenden van het project, maar daarnaast moet het team er ook voor zorgen dat de verantwoordelijken voor de onderhoudsfase het systeem optimaal kunnen aanpassen, uitbreiden en ondersteunen.
3. Belangrijke informatie moet nog steeds goed gedocumenteerd worden en soms bestaat er ook een wetgeving die het aanleggen van bepaalde documentatie verplicht. Het team moet alle high-level informatie documenteren maar daarom niet al de kleinste details.
4. 'Agile software development' teams schrijven nog steeds documentatie. Uit een eigen onderzoek blijkt dat 'agile software development' teams evenveel leverbare documentatie, zoals eindgebruikers documentatie en operationele documentatie, maken. Dat er geen documentatie opgeleverd wordt is een groot misverstand over 'agile software development.'

Het verschil met de vroegere software ontwikkelingsmethodes is dat de 'agile' methodes weinig nadruk leggen op documentatie, zeker als die voor intern gebruik

bedoeld is. Documentatie kan verwacht of gevraagd worden voor verschillende redenen en soms zijn dat gewoon geen goede redenen, zoals bijvoorbeeld een vragende partij die in controle wil blijven en zijn fiat wil geven op alle beslissingen die het projectteam neemt of een vragende partij die denkt dat de documentatie noodzakelijk is voor het succes van het project of die gewoon de richtlijnen van het bedrijf opvolgt zonder erbij na te denken wat het nut van de documentatie is.

Binnen de 'agile' filosofie wordt documentatie meestal behandeld zoals elke andere vereiste van de gebruikers. De vraag naar een document moet dezelfde behandeling krijgen als een vraag naar een nieuw kenmerk in het programma. Elke verzoek voor documentatie moet geraamd en een prioriteit toegekend krijgen. Elke investering in documentatie is een investering die niet naar functionaliteit kan gaan en vice versa. Dus het is belangrijk dat iemand een berekende beslissing neemt of er al dan niet in bepaalde documentatie geïnvesteerd zal worden. De investering in documentatie is een business beslissing, en geen beslissing van het technische team. Een team moet geen documentatie maken omdat het proces dat vereist, het moet documentatie enkel maken als een 'stakeholder' ernaar vragen.

Een goede manier om deze filosofie kracht bij te zetten is dat iedereen die documentatie vraagt, dient te verantwoorden waarom hij die documentatie nodig heeft. De verzoeker voor documentatie moet begrijpen dat er een 'total cost of ownership' (TCO) voor elk document bestaat en het doel van elk project is om de ROI voor de 'stakeholders' te verhogen. Het voordeel dat het bestaan van het document oplevert, moet groter zijn dan de kost voor de creatie en het onderhoud van het document.

Kevin Tate (2005) beschrijft dit als het hebben van documentatie die net volstaat. Teams moeten zich concentreren op het afleveren van werkbare producten en niet op het creëren van documentatie. De 'agile' methode probeert de interne documenten te beperken tot het minimum. Het gebeurt veel te vaak dat teams vergeten waarom ze documentatie maken, namelijk om deze af te leveren bij het volledig werkend product. In nog te veel gevallen is documentatie vereist voordat een project naar de volgende projectfase kan overgaan. Dit zorgt voor een lineaire ontwikkeling die ongezond en onrealistisch is. Het meest belangrijk in een project zijn niet de documenten maar de conversaties en beslissingen die genomen worden en leiden naar het tot stand komen

van een nuttig product. Documentatie is het resultaat van het ontwikkelingsproces en mag niet de focus zijn. Daarom is het secundair aan samenwerken en leren.

Deze manier van werken heeft vooral tot doel om het team te laten focussen op het belangrijkste, namelijk het maken werkende software producten. Klanten weten meestal niet goed wat ze willen tot ze iets kunnen bekijken. Zelfs bergen documentatie helpen de klanten niet bij het visualiseren van het eindproduct. De onderzoeker geeft hierbij nog de volgende tips:

- Gebruik indexkaarten bij het plannen van de projectfasen. Deze indexkaarten bevatten de karakteristieken van het product, moedigen de teamleden aan om over de verschillende karakteristieken te discussiëren en geven de teamleden iets tastbaar om tijdens het plannen, nieuwe ideeën te visualiseren.
- Een verzameling van 'feature cards' zou moeten volstaan als documentatie van de vereisten. Het is mogelijk dat in uitzonderlijke gevallen extra documentatie nodig is maar dat is niet de norm.
- Ontwerp software samen in teamverband voor een 'whiteboard' of met behulp van een schetstool. De discussies tijdens deze sessies zullen de problemen aanduiden en mogelijke alternatieve aanpakken naar boven brengen. Na de sessie zou het voldoende moeten zijn om een foto van het 'whiteboard' te nemen als documentatie van het ontwerp.

Dit zijn maar enkele voorbeelden en geven aan waar er tijd en kosten bespaard kunnen worden bij het aanmaken van documentatie. Volgens de auteur zijn de gevaren van te veel documentatie groter dan de gevaren van te weinig documentatie.

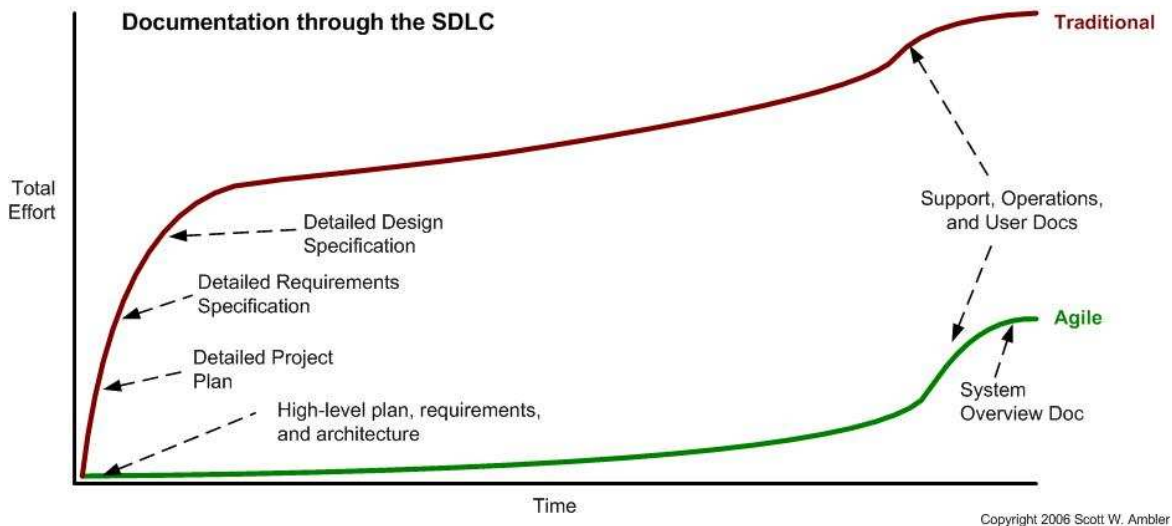
Scott W. Ambler (2009) kiest ervoor om 'agile' documenten te creëren. Een 'agile' document bestaat uit de volgende kenmerken:

1. 'Agile' documenten maximaliseren de ROI voor de 'stakeholder'. Het voordeel van het bestaan van het document moet groter zijn dan de kost voor de creatie van het document.
2. 'Stakeholders' kennen de 'total cost of ownership' van het document.
3. 'Agile' documenten zijn 'lean and sufficient'. Het document bevat net genoeg informatie voor het vervullen van zijn functie, met andere woorden het is zo eenvoudig mogelijk van opzet.

4. Elk 'agile' document wordt opgemaakt voor het vervullen van één bepaald doel. Als dat doel niet duidelijk is, is het document niet nuttig.
5. 'Agile' documenten beschrijven belangrijke dingen en bevatten kritische informatie, zoals de motivering van het ontwerp, de gebruikers procedures of de operationele procedures. Evidente informatie hoort niet thuis in 'agile' documentatie.
6. 'Agile' documenten worden opgemaakt voor een specifieke persoon en ze helpen die persoon bij zijn taken. Het is daarbij belangrijk dat de schrijfstijl van het document aangepast wordt aan die persoon en dat er met de persoon samengewerkt wordt om tot gebruiksvriendelijke documentatie te komen.

'Agile' documenten zijn net voldoende en worden dus pas aangepast of geupdated wanneer het pijn doet. Soms zijn documenten verouderd maar heeft dit geen effect op het verloop van het project. De projectmanager moet telkens afwegen of de kost die het updaten van het document het met zich meebrengt verantwoord is. Het kan frustrerend zijn om modellen en documenten te hebben die niet helemaal accuraat of consistent zijn, maar deze frustraties worden gecompenseerd door de productiviteitsverhoging van het team dat naliet de verouderde, niet kritische documentatie aan te passen.

De 'agile software development' methode zorgt ervoor dat de noodzakelijke documentatie pas op het einde het project opgeleverd wordt. De grafiek hieronder geeft het verschil van inspanningen in documentatie weer tussen de traditionele 'software development' methode en de 'agile software development' methode.



Bron Scott W. Ambler (2009)

Het bestaan van systeem documentatie is essentieel en het vermindert de afhankelijkheid van de kennis van de teamleden binnen een project. De kritische informatie is dan gedocumenteerd voor het geval er bijvoorbeeld plots iemand het team verlaat. Maar volgens de auteur is dit echter een valse zekerheid. Als een projectleider plots een teamlid verliest dan geeft die meestal een groot probleem, onafhankelijk van hoe goed de documentatie op dat moment is. Er zal nog steeds een nieuw teamlid gevonden moeten worden, dat aan het project toegewezen moet worden en het systeem zal moeten leren kennen. Dit kost allemaal tijd en het bestaan van documentatie vermindert die tijd niet zo vaak. Het is beter om ervoor te zorgen dat de kennis niet enkel aanwezig is bij één persoon en dat het project dus geen gevaar loopt als er iemand plots wegvalt. De auteur verkiest hiervoor kennisdeling methodes zoals een actieve 'shareholder' participatie tijdens het project, het collectief eigendom waarbij iedereen aan een model kan werken en het gezamenlijk modelleren. Het is ook belangrijk dat het systeem of product niet zonder meer overgedragen wordt aan het onderhoudsteam. De teamleden die het onderhoud van het project toegewezen krijgen, moeten eerst het systeem leren kennen. Het is gebruikelijk om de leden van dit onderhoudsteam al in te werken vooraleer het eigenlijke project afgelopen is.

'Templates' kunnen een voordeel zijn voor het opstellen van documentatie maar meestal zijn ze een hinderpaal omdat de opstellers van de 'template' alle mogelijke situaties en informatie een plaats willen geven. Dit zorgt ervoor dat teamleden extra

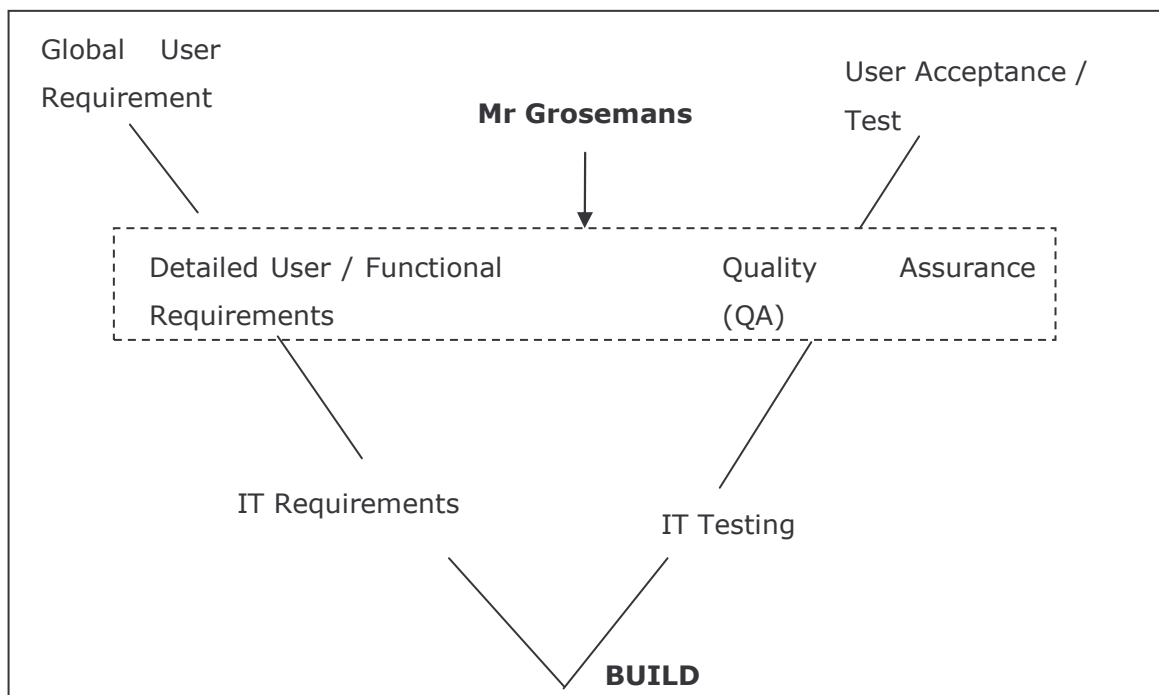


informatie en documentatie gaan creëren die niet nuttig is voor het project. Ze spenderen extra tijd in het opzoeken van informatie voor bepaalde onderdelen van de 'template' die achteraf nooit gelezen worden. Als er met 'templates' gewerkt wordt dan moeten die zich concentreren op de minimale, kritische informatie. Alle bijkomende informatie die de teamleden belangrijk achten kunnen ze dan toevoegen aan het document.

## Hoofdstuk 4: Gevalstudie: Duurzame ontwikkeling bij een grootbank

### 4.1. Inleiding van de gevalstudie

Deze thesis is tot stand gekomen in samenwerking met dhr. Grosemans, die werkzaam is bij een grootbank in België. De bank heeft meer dan tienduizend werknemers en is actief in verschillende landen. De grootbank is een van de grootste ICT-werkgevers van ons land. Dhr. Grosemans werkt op de afdeling Betalingen en staat aan het hoofd van Applicatie en Projecten van Internationale betalingen. Hij komt dus dagelijks in aanraking met de problematiek omtrent het aanmoedigen van kennisdeling en het opstellen van documentatie. Om zijn positie binnen het ontwikkelingsproces duidelijker weer te geven heeft hij de volgende grafiek getekend:



De dienst van dhr. Grosemans is dus de link tussen de business en de IT-afdeling, zorgen ervoor dat de business vereisten vertaald worden naar verstaanbare IT-vereisten en controleert ook of de producenten die de IT-afdeling aflevert voldoen aan de vooropgestelde vereisten.

Om de situatie bij de grootbank beter te begrijpen heb ik besloten om dhr. Grosemans te interviewen. De opzet van dit interview was om een duidelijk beeld te krijgen van de situatie qua kennisdeling bij de grootbank in het algemeen en op de afdeling van dhr. Grosemans in het bijzonder. Op basis van dit interview kan ik dan een analyse van de situatie maken, de uitdagingen, die in de vorige hoofdstukken aangekaart werden, toetsen aan de realiteit en aanbevelingen voor mogelijke verbeteringen in de grootbank voorstellen.

## **4.2. Situatie bij de grootbank**

Dhr. Grosemans begint het interview met een duidelijke afbakening van de term kennis zoals die relevant is voor zijn situatie in de ICT development sector. De term kennis verwijst voor zijn afdeling naar productkennis, inhoudskennis of eventuele methodieke kennis die eigen is aan bepaalde afdelingen. Er zijn enorm veel soorten kennis aanwezig in een bedrijf maar tijdens dit interview wil dhr. Grosemans het begrip kennisdeling afbakenen in termen van "wij zetten een applicatie op en nadien moeten anderen weten hoe die applicatie werkt." Of zoals hij het met een heel extreem voorbeeld verduidelijkte: "Stel indien er hier een bom ontploft, zullen de werknemers dan nog in staat zijn om de systemen verder te onderhouden?" Een ander probleem dat regelmatig opduikt is het vertrek van 'key people', belangrijke personen binnen het team. Het is dan altijd even afzien omdat er in ieder geval een deel kennis van het project verloren gaat en de overige teamleden het systeem zo goed mogelijk dienen te onderhouden trekken totdat er na enige tijd weer een nieuwe expert gevormd is die de verloren kennis terug heeft opgebouwd.

Dhr. Grosemans heeft vastgesteld dat er binnen de bank een aantal initiatieven gelanceerd zijn om aan kennisdeling te doen maar hierbij gaat het vooral om IT-tools die het beschikbaar stellen van informatie tussen de teamleden zouden moeten vereenvoudigen. Zo zijn er bijvoorbeeld IT-forums beschikbaar waar de geautoriseerde gebruikers toegang krijgen tot bepaalde documenten om deze te lezen of te wijzigen. Deze initiatieven zijn bedoeld om de kennisdeling op de departementen te bevorderen, maar het is nog steeds aan het departement zelf om te beslissen hoe ze het probleem van kennisdeling aanpakken. Meestal wordt er gewerkt met een 'shared drive', een map op het netwerk, waar de documenten bewaard worden volgens de overgekomen structuur van het departement wat de werknemers moet toelaten de informatie vlot terug te vinden.

Dhr. Grosemans ontvangt geen richtlijnen vanuit het hogere management omtrent kennisdeling, maar dat wil niet zeggen dat ze niet zouden bestaan op andere afdelingen. Het is best mogelijk dat er bijvoorbeeld binnen de IT-afdeling een betere methodologie voor kennisdeling bestaat. Hij merkt op zijn afdeling wel op dat er allerlei initiatieven zijn om aan een beter projectmanagement te doen. Dat er geen richtlijnen zijn van bovenaf wil niet zeggen dat men niet aan kennismanagement doet,

want het is het afdelingshoofd zelf dat het aangaande het kennismangement mag bepalen.

Op zijn afdeling proberen ze vooral een goede structuur aan te brengen in de documentatie. Er wordt gekeken naar de business processen en hoe de applicaties hierop gebouwd zijn. Dan wordt er een top-down beschrijving gemaakt van de architectuur, de scope en worden detailbeschrijvingen van alle componenten opgesteld. Van elke component van de applicatie wordt de documentatie onderhouden of op zijn minst trachten ze de applicatiewijzigingen te documenteren. Als er zich problemen voordoen is dat meestal omdat er één basis document bestaat met daaraan gekoppeld alle gedocumenteerde 'change requests' en 'incident fixes' die gebeurd zijn. De kennis bevindt zich dan niet meer in één document maar in verschillende documenten, wat de kennisdeling niet bevordert. Daarnaast zijn er ook geen algemene richtlijnen over hoe deze documenten eruit moeten zien, dat hebben ze op zijn afdeling zelf bepaald.

De werknemers van zijn afdeling zijn zeker bereid om hun kennis te delen. Het enige probleem dat zich soms voordoet is dat oudere werknemers het concept van kennisdeling en kennisbeheer niet altijd goed begrijpen omdat dit redelijk nieuwe termen zijn. Omdat ze al jaren in een bepaald stramien functioneren is het moeilijker om hen met nieuwe media te laten werken voor kennisdeling. Het concept van kennisdeling is een nieuwe stroming in onze maatschappij. Jonge mensen zijn hiermee beter vertrouwd en maken het zich ook sneller eigen. Het wil niet zeggen dat oudere werknemers niet bereid zijn tot kennisdeling maar hun jarenlange werkwijze kan wel een barrière vormen. Als aan de werknemers gevraagd wordt hun kennis te delen antwoorden ze altijd positief en willen ze hier zeker oprecht aan meewerken. Als het dan toch niet gebeurd heeft het vooral met het onderbewuste maken, maar zeker niet uit vrees voor het verlies van hun job. De afdeling is niet geraakt door de huidige financiële crisis en heeft momenteel zelfs meer werk dan voordien. Het probleem van kennisdeling is meestal tijd. Zeker naar het einde van het project toe, als het product opgeleverd moet worden, dreigt het belang van documentatie nogal eens naar de achtergrond te verschuiven.

Het is volgens dhr. Grosemans zeker niet zo dat werknemers kennis achterhouden in de hoop hun carrière vooruit te helpen. Volgens hem bestaat de manier om carrière te maken binnen het bedrijf erin om regelmatig van project wisselen. Als iemand wil

vertrekken is het echter noodzakelijk dat hij misbaar is in het huidige project. Hij moet ervoor zorgen dat hij een goede opvolger heeft en dat zijn unieke kennis goed gedocumenteerd en overgedragen is binnen het team. Zo niet, wordt zijn vertrek geweigerd, ook om te voorkomen dat het bedrijf in moeilijkheden zou kunnen komen.

Na de oplevering van het project vertrekt meestal een groot deel van het team naar een ander project en blijven er enkele mensen, soms te weinig, achter die de 'maintenance' van het project op zich zullen nemen. Zo vertrekken de externe medewerkers meestal allemaal naar een ander project of verlaten zelfs het bedrijf, terwijl de interne medewerkers, welke meestal ook een goede globale kennis van de bestaande applicaties bezitten, ook verhuizen naar een ander project of in meer overkoepelende functies gaan werken. Er blijven gewoonlijk een paar goede experts werkzaam in het opgeleverde project, die zeer specifieke productkennis bezitten, maar meestal wel het globale overzicht missen. Het inwerken van nieuwe teamleden in de afdeling gebeurt door middel van een minimale opleiding over de dagelijkse gang van zaken. Daarnaast worden hen ook een aantal specifieke taken toevertrouwd, die hen moeten toelaten om een globale kennis op te bouwen door middel van het raadplegen van de bestaande documentatie en het opvragen van de nodige informatie bij de ervaren teamleden.

Kort na de oplevering van een project worden er meestal nog ettelijke aanpassingen doorgevoerd zoals kleine tekortkomingen die dringend verholpen dienen te worden of functionaliteiten die toch niet volledig werken zoals verwacht. Daardoor is de documentatie soms al verouderd bij de aanvang van de onderhoudsfase. Op dat moment komen er ook een aantal mensen het onderhoudsteam versterken, die weinig of niets te maken hadden met de vorige projectfasen en die zich dus moeten inwerken en opgeleid moeten worden. Bij deze teamwissel gaat er telkens een deel kennis verloren omdat de systeemdokumentatie niet meer helemaal up-to-date is. Een teamwissel zorgt ook steeds voor een vermindering van de focus en een tempoovertraging waardoor een achterstand opgelopen kan worden die nog moeilijk in te halen is, maar voor het management is de doelstelling gehaald, namelijk de oplevering van het werkend product. Als tijdens de onderhoudsfase problemen ontstaan, worden er soms een aantal extra medewerkers ingezet om erger te voorkomen. Het is zeer zeldzaam dat een nieuw opgeleverd product vanaf de eerste dag optimaal functioneert en er is bijna altijd een stabilisatieperiode nodig om de onvolmaaktheden van het product weg te werken. Soms moeten er zelfs 'work

arounds' geconstrueerd worden als tijdelijke oplossing voor een gebrekkig functionerend product. Omdat tijdens zo een stabilisatieperiode alle aandacht gefixeerd is op het optimaal doen werken van het opgeleverde product, zijn de werknemers niet echt begaan met de correcte documentatie van alle aanpassingen, deels ook wegens tijdgebrek. Het niveau van formalisering van de documentatie wordt ook hier aan het teammanagement overgelaten, wat zeker een risico inhoudt, omdat zij in die stabilisatiefase vooral bekommerd zijn over het vermijden van business problemen en het verzekeren van de kwaliteit van het product en niet over het up-to-date houden van de documentatie.

Na deze stabilisatieperiode wordt er een industrialiseringsoefening van de 'maintenance' uitgevoerd. Hiervoor, wordt voor het opgeleverde product, nagegaan welke controles de werknemers moeten voorzien, wat er tijdens de uitvoering mis kan lopen en welke acties er dan genomen moeten worden. Dit resulteert in een aantal 'lessons learned' rapporten.

Op de afdeling van dhr. Grosemans is een algemene industrialiseringsoefening gebeurd waarbij er nagegaan is hoe een typische productoplevering eruit zou moeten zien en welke documentatie daarvoor nodig is. Zijn afdeling stelde vast dat de projecten die opgeleverd worden over het algemeen goed gedocumenteerd zijn. Het probleem is echter dat de algemene productdocumentatie meestal niet bijgewerkt wordt als er wijzigingen aangebracht worden of soms bestaat er gewoon ook geen algemene documentatie van het product. De samenwerking met een nieuwe partner, die meer belang hecht aan formalisering en documentatie, heeft er echter toe bijgedragen dat de vraag naar documentatie nu gestegen is. Er werd zelfs een budget ingepland om aan retrodocumentatie te werken, het documenteren van de belangrijkste componenten waarover geen geschreven informatie beschikbaar is. Het doel van retrodocumentatie is vooral om de globale structuur van een component goed in kaart te brengen, zonder daarbij elk detail te vermelden. Dit laat het ook toe om onafhankelijker te worden van enkele medewerkers die deze cruciale informatie in hun hoofd hebben. Het vertrek van dergelijke personen kan immers heel wat kosten met zich meebrengen, die je als bedrijf zoveel mogelijk wil vermijden.

In extreme gevallen kan er ook aan 'reverse engineering' gedaan worden, waarbij er nagegaan wordt wat er precies met het product gebeurd tijdens de productie of waarbij soms ook overgegaan wordt tot het lezen van de broncode. 'Reverse

engineering' wil je als bedrijf te allen tijde vermijden maar het kan verantwoord zijn als er iemand ziek wordt of plots vertrekt. 'Reverse engineering' levert goede resultaten op, maar het neemt veel tijd in beslag.

Volgens dhr. Grosemans worden er door het bedrijf geen stappen ondernomen om het vertrouwen van de werknemers te bevorderen. De reden hiervoor is dat er momenteel een gevoel van veiligheid bestaat en de werknemers niet bang zijn voor het verlies van hun baan. Beloningssystemen voor het delen van kennis zijn ook niet aanwezig. Volgens dhr. Grosemans is er geen enkel bedrijf waar het delen van kennis als iets abnormaals beschouwd wordt. Als iemand binnen het team niet goed meewerkt aan het delen van kennis, krijgt hij een slechte evaluatie. Daarnaast mag je medewerkers ook niet extreem dankbaar zijn voor dingen die gewoon van hen verwacht worden. Een beloningssysteem zou volgens dhr. Grosemans een goede manier kunnen zijn om kennisdeling in de hand te werken binnen een bedrijfscultuur waar het delen van kennis niet vanzelfsprekend is. Gelukkig bestaan er hieromtrent binnen zijn bedrijf geen problemen.

Van het bestaan van 'communities of practice' heeft dhr. Grosemans nog niet gehoord. Er zijn dan ook geen initiatieven vanuit de bedrijfsleiding om kennisdeling binnen zulke 'communities of practice' aan te moedigen. Er wordt momenteel ook met het idee gespeeld om projectleiders ook buiten hun departement tijdelijk projecten te laten leiden. Wat dhr. Grosemans verder ook binnen zijn team promoot is het presenteren van nieuwigheden tussen mensen met dezelfde functie. Indien er bijvoorbeeld een nieuwe tool werd geïnstalleerd, of een medewerker heeft voor een project iets vernieuwend gerealiseerd, dan wordt aan de betrokkene gevraagd om er een presentatie over te geven aan de andere experts of projectleiders. Zo kan iedereen kennis nemen van de nieuwe informatie en weet men ook aan wie men achteraf vragen kan stellen.

Op zijn afdeling werken ze dus via twee verschillende modi. Enerzijds is er de projectmodus en anderzijds de onderhoudsmodus. Tijdens de onderhoudsmodus gebeuren er enkel incrementele wijzigingen wordt het product ondersteund door middel van 'incident fixing' en gebeuren de implementaties van 'change requests.' Als er grote wijzigingen aan het product dienen te gebeuren of er moet een volledig nieuw product gebouwd worden, waarvoor er een team met specialisten uit verschillende disciplines vereist is, wordt er een nieuw project opgestart. Hiervoor wordt er een



tijdelijk, niet hiërarchisch, samenwerkingsverband opgericht tussen de verschillende werknemers en één of twee personen die het project coördineren.

Als projectmethodologie wordt er gebruik gemaakt van PRINCE2. Dhr. Grosemans wil hierbij wel nuanceren dat PRINCE2 een generische methode is die voor gelijk welk project, onafhankelijk van het eindproduct, gebruikt kan worden, maar op zich is het een goede methode. De rollen die beschreven zijn, zijn generiek maar wel relevant. Er dient wel op toegezien te worden dat de grootte van het project, de hoeveelheid project-administratie verantwoord. Voor kleinere projecten is bijvoorbeeld samenbrengen van een volledige 'project board' met de verschillende sponsors waarschijnlijk overdreven er zal ook geen nood hebben aan een project support office of een secretaresse. De graad van formalisering moet dus in overeenstemming zijn met de complexiteit van het project, zoals bijvoorbeeld het aantal partijen dat betrokken moeten worden bij de beslissingsmomenten van het project. Een tweede punt dat dhr. Grosemans aankaartte is dat PRINCE2 wel vermeldt welke documenten er moeten aangemaakt worden maar dat het geen voorbeelden of in te vullen 'templates' aanreikt. Het is aan de projectleider of het bedrijf om te kiezen hoe de documenten moeten opgesteld worden. Het belangrijkste document voor dhr. Grosemans is het PID, het 'Product Initiation Document' dat beschrijft wat de 'scope' van het project zal zijn, welke de beschikbare budgetten zijn, enzovoort. Voor de andere documenten worden de PRINCE2 voorschriften nogal eens met een korrel zout genomen. Dhr. Grosemans heeft ook eens met een externe partner samengewerkt die CMMI gebruikte als projectmethodologie. Zij hadden een hogere graad van formalisatie en documentatie voor hun projecten en pasten de voorgeschreven regels veel strikter toe, met als resultaat dat hun projecten veel meer kosten en ook veel langer duurden. Er werd er daardoor ook minder gecommuniceerd omdat de mensen veel meer op hun eigen eilandjes werkten. Zo kunnen er misverstanden ontstaan en zorgt de minste vertraging in één fase van het project voor vertragingen in het hele project. Dhr. Grosemans vergelijkt deze twee benaderingswijzen voor een projectmethodologie en hij erkent dat een bedrijf een afweging moet maken tussen een meer geformaliseerde of een meer pragmatische aanpak. Dit betekent niet dat er geen projectstructuur nodig is. Er moeten steeds duidelijke rollen toegekend worden en ook een goede administratie is essentieel, zelfs voor kleine projecten. Maar bijvoorbeeld een 'steering committee' met tien verschillende rollen is niet te verantwoorden als er maar twee partijen bij het project betrokken zijn.

Voor IT gerelateerde projecten kunnen er twee soorten documentatie onderscheiden worden. Projectdocumentatie beschrijft hoe het project gerealiseerd zal worden en maakt voor de projectmedewerkers duidelijk wat van hen verwacht wordt. Deze documentatie laat een goede opvolging van het project toe en is niet meer relevant na de afloop van het project. De systeemdokumentatie echter, die een technische en functionele beschrijving van het product omvat, zal na afloop van het project de basis vormen voor een goede 'maintenance' van het product. Dhr. Grosemans verkiest dat externe werknemers niet verantwoordelijk zijn voor het onderhoud van systeemdokumentatie, maar door tijdsgebrek op het einde van het project kan dit wel eens gebeuren.

Op zijn afdeling maken ze gebruik van zelf ontwikkelde 'templates' voor de PRINCE2 projectmethodologie. Het vervolledigen van deze 'templates' wordt niet zo formeel benaderd. De werknemers worden niet steeds op de vingers getikt als er al eens iets wordt opengelaten, hoewel de kwaliteit van de opgeleverde documenten wel geverifieerd wordt. De medewerkers worden niet te vrij gelaten in het opstellen van de documentatie en er moet een bepaald niveau van formalisering blijven bestaan. Ter vergelijking gebruikte de externe partner met de hoge formalisering voor CMMI uitgebreidere en meer gestandaardiseerde 'templates' waarop veel meer informatie ingevuld diende te worden.

De documentatie wordt bewaard op een 'shared drive,' die verder wordt onderverdeeld naargelang de architectuur van de applicatie. Dit moet iedereen toelaten om de documenten terug te vinden die hij nodig heeft. Voor sommige projecten wordt er niet via een 'shared drive' gewerkt, maar werd er een IT-forum opgericht om de documenten te bewaren en met elkaar te delen. De documenten betreffende de opvolging van de testen, de incidenten en de 'change requests' worden in een 'repository' opgeslagen, genaamd HP Quality Center, waardoor deze documenten voor iedereen toegankelijk zijn.

Of er binnen een project een medewerker full-time aangeduid wordt als verantwoordelijke voor de kwaliteit van de documentatie hangt af van de grootte van het project. Het opleveren van kwaliteitsvolle documentatie maakt deel uit van het kwaliteitsmanagement en PRINCE2 voorziet dan ook een rol voor een kwaliteitsmanager die onder andere op de kwaliteit van de documentatie moet toezien. Uit zijn eigen praktijk haalde dhr. Grosemans nog volgend voorbeeld aan. Bij

een industrialiseringsoefening die laatst op zijn afdeling plaatsvond, voor een product in 'maintenance' modus, werden er templates opgemaakt en werd er iemand aangesteld die diende te controleren of alle medewerkers wel rigoureuus de voorgeschreven regels opvolgden. Deze industrialiseringsoefening was nodig omdat er bij een bepaalde applicatie geen gestructureerde manier meer bestond voor het beheer van de documentatie. Via een end-to-end analyse van het product, de inventaris van de waargenomen fouten en de bestaande documentatie werd de huidige kennis gebundeld. Deze oefening resulteerde in een efficiënter onderhoud van de applicatie en het was een goede oefening om de documentatie op orde te stellen. Dit resulteerde ook in meer kwaliteit voor de gebruikers, minder incidenten in productie, minder operationele verliezen en minder risico's.

PRINCE2 legt op dat er een 'lessons learned' logboek bijgehouden wordt tijdens het project en op het einde van het project moet er ook een 'project closure' document opgesteld worden met daarin een opsomming van alle 'lessons learned.' Maar deze vereiste wordt niet goed opgevolgd bij de afdeling van dhr. Grosemans. Het werd een aantal keer geprobeerd maar het lukte niet echt en daarom krijgt zijn afdeling ook weinig feedback van andere projecten over hun 'lessons learned'. Meestal wordt de kennis van andere projecten overgedragen door middel van intermenselijke relaties, bijvoorbeeld door aan werknemers uit andere projecten te vragen waar de moeilijkheden liggen en wat de beste aanpak zou zijn voor bepaalde problemen. Dhr. Grosemans erkent dat dit een punt is dat voor verbetering vatbaar is, omdat het bijhouden van een 'lessons learned' logboek ook niet veel tijd in beslag neemt. Het management, zichzelf inclusief, zou het gebruik ervan meer moet aanmoedigen. Het probleem met zulke documenten is volgens dhr. Grosemans dat de nood ervan pas duidelijk wordt als er iets mis gaat, als er zaken gebeuren die vermeden hadden kunnen worden met een dergelijk document. Op de vraag of de werknemers soms niet bereid zouden zijn om hun fouten toe te geven, erkent dhr. Grosemans dat er inderdaad een persoonlijke factor kan spelen. Het is gemakkelijk om fouten toe te geven waar je maar half verantwoordelijk voor bent, maar het is nog steeds gênant om echte flaters toe te geven. Deze komen echter zelden voor en het zou geen blokkerende factor mogen zijn voor het opstellen van goede 'lessons learned' rapporten.

### **4.3. Analyse van de gevalstudie**

In dit deel zullen we de situatie in de grootbank, zoals beschreven door dhr. Grosemans, toetsen aan de theorie van de vorige twee hoofdstukken.

Het is zeer interessant dat door de bestaande project- en teamstructuur de kennisdeling niet gehinderd wordt door sociale dilemma's. Werknemers die carrière willen maken door regelmatig van project te wisselen, worden verplicht hun kennis goed te documenteren en door te geven. Ze moeten een goede opvolger hebben zodat het projectteam verder kan functioneren zonder hen. Ze krijgen daardoor een intrinsieke motivatie om hun kennis te delen zonder dat er beloningssystemen aan te pas komen. Dit is zeer interessant voor het bedrijf maar anderzijds doen er zich na de implementatie van een nieuw project soms problemen voor bij de overdracht naar het onderhoudsteam. Op dat moment vertrekken er medewerkers uit het projectteam en gaat een deel van de kennis verloren wat voor een mogelijke achterstand zorgt bij het onderhoudsteam. Het is misschien interessant om te onderzoeken of een aantal projectmedewerkers tijdelijk en deeltijds beschikbaar moeten gehouden worden voor het onderhoudsteam tot het productieproces vlot verloopt en volledig gedocumenteerd is.

Daarnaast worden de 'lessons learned' documenten niet als dusdanig opgesteld en worden aanpassingen aan het systeem soms apart gedocumenteerd en niet toegevoegd aan de bestaande documentatie. Mogelijks is het ontbreken van de motivatie bij de werknemers hiervoor de oorzaak, dit in tegenstelling tot de oplevering van goede systeemdokumentatie waarvoor de werknemers wel veel tijd vrijmaken. Misschien kan het management overwegen om een beloningssysteem te ontwikkelen voor het delen van deze kennis. Als bijvoorbeeld de kwaliteit van de documentatie en kennisdeling mee opgenomen wordt in de evaluaties en de werknemers die zeer goede documentatie opleveren daar openlijk erkenning voor krijgen, kan dit zorgen voor een betere 'lessons learned' en onderhoudsdocumentatie en kennisdeling.

De organisatie organiseert geen specifieke activiteiten om het vertrouwen tussen de werknemers aan te wakkeren, al vermoed ik wel dat de presentaties die gegeven worden door de experts aan hun collega's onrechtstreeks bijdragen tot het opbouwen van 'competence based' vertrouwen. Er zijn momenteel geen problemen met vertrouwen binnen de organisatie, waardoor er vermoedelijk aan dit aspect niet zoveel

aandacht besteed wordt. Het belang van vertrouwen bij kennisdeling wordt echter best niet uit het oog verloren.

Van de term 'communities of practice' had dhr. Grosemans nog niet gehoord en het kan interessant zijn voor het bedrijf om het gebruik hiervan verder te onderzoeken. Volgens de literatuur levert het aanmoedigen van het gebruik van 'communities of practice' toch een positieve bijdrage tot kennisdeling. Doordat kennis ook over de grenzen van departementen of projecten heen met elkaar gedeeld kan worden.

Op de afdeling van dhr. Grosemans wordt er gebruik gemaakt van PRINCE2 voor de projectmethodologie. Hij is hierover tevreden en de methode wordt aangepast naar gelang de grootte van het project. Doordat het bedrijf constant met projecten werkt is hun projectmethodologie verder verfijnd over de jaren. Ook de IT-infrastructuur die aanwezig is voor kennisdeling blijkt adequaat en vormt zeker geen hindernis in het delen van kennis. De 'shared drive' of de IT-forums laten de teamleden toe om op een snelle en efficiënte manier kennis met elkaar te delen.

Uit de industrialiseringsoefening is gebleken dat als er iemand verantwoordelijk is voor de kwaliteit van de documentatie, de vooropgestelde standaarden beter opgevolgd worden. Mogelijks wordt de kwaliteit van de projectdocumentatie momenteel niet opgenomen in de beoordelingscriteria van de projectleiders, waardoor zij op hun beurt misschien ook geen al te hoge eisen stellen aan de kwaliteit van de documentatie die de teamleden afleveren. Het zou interessant kunnen zijn om te onderzoeken of een kwaliteitsverantwoordelijke voor de projectdocumentatie een positieve bijdrage kan leveren en het is misschien ook een goed idee om voor elk document een verantwoordelijke aan te duiden die erop moet toezien dat het document juist en volledig wordt opgesteld.

Dhr. Grosemans verwijst regelmatig naar de partner waarmee ze samenwerkten die een hogere formalisering voor documentatie hanteerde. Hij vindt dat dit echter geen ideale werkwijze is en zichzelf gaan iets pragmatischer om met de methodologie en de documentatie wat dhr. Grosemans ook goed vindt want het zorgt voor meer communicatie tussen de verschillende partijen. Hij vraagt zich wel af welke documentatie best door een projectteam opgesteld en bijgehouden wordt. Het toepassen van 'agile' methoden kan hem hierop een antwoord geven. 'Agile' methoden hechten een groot belang aan communicatie tussen teamleden en 'stakeholders.' Daarnaast moet de noodzaak van elk document gefundeerd worden door de aanvrager

van dat document. De 'Total Cost of Ownership' van de documenten wordt in rekening gebracht en zal ervoor zorgen dat enkel de noodzakelijke documenten aangemaakt worden.

Tijdens het interview leerde ik ook dat sommige delen van de huidige 'templates' niet steeds ingevuld werden. Dit kan erop wijzen dat de 'templates' te uitgebreid zijn en zich niet beperken tot de minimale vereisten. De huidige 'templates' zijn dus waarschijnlijk aan herziening toe.

## 4.4. Aanbevelingen

Op basis van de analyses uit het vorige deel zou ik de volgende aanbevelingen willen formuleren.

- Overweeg de invoering van 'agile software development' methodologie en 'agile' documentatie. De 'agile' methodologie hecht veel belang aan communicatie en het leveren van werkende producten. 'Agile' documentatie is net voldoende en zorgt ervoor dat de 'Total Cost of Ownership' van de documentatie in rekening gebracht wordt.
- Zorg dat voor elk document een verantwoordelijke is aangeduid. De kwaliteit van de documenten zal verbeteren als een werknemer de verantwoordelijkheid krijgt voor een bepaald document. Dit moet echter wel samen bekeken met het volgende punt, het bestuderen van de huidige 'templates' zodat er geen tijd verspild wordt met het neerschrijven van nutteloze informatie.
- Herbekijk de 'templates' voor de documenten. 'Templates' moeten enkel de essentie voorzien en niet alle uitzonderingen. Indien er bepaalde zaken opengelaten worden, wijst dit erop dat de 'templates' te veel informatie en uitzonderingen proberen op te nemen.
- Bestudeer beloningssystemen voor het delen van bepaalde kennis aanmoedigen. Dankzij de interessante projectstructuur bestaan er weinig problemen met het delen van systeemkennis tijdens de uitvoering van projecten in de organisatie. De documentatie van de 'lessons learned' en de kennis opgedaan tijdens de 'maintenance' van het product, kan echter verbeterd worden en hierbij kan een beloningssysteem mogelijk helpen. Dit beloningssysteem moet echter goed bestudeerd worden en passen in de cultuur van de organisatie.
- Overweeg het aanmoedigen van kennisdeling binnen 'communities of practice.' 'Communities of practice' zijn een vrije nieuw begrip voor het aanmoedigen van kennisdeling. Het kan in het voordeel zijn van het bedrijf om werknemers actief aan te moedigen om kennis te delen met collega's die dezelfde functie uitoefenen in een ander departement.

- Vergeet vertrouwen niet. Vertrouwen ligt aan de basis van kennisdeling. Dus als er ooit problemen opduiken met kennisdeling, vergeet dan niet om na te gaan of er geen onderliggend probleem met vertrouwen bestaat.



## Hoofdstuk 5: Conclusies

De huidige kenniseconomie en snel evoluerende marktomgeving maken van de opslag van kennis en het aanmoedigen van kennisdeling actuele uitdagingen, die de meeste bedrijven erkennen, maar toch slagen nog maar weinig bedrijven erin om tot een optimale kennisdeling te komen. De 'agile' software ontwikkelingsmethode kan een goede methode zijn om in deze snel evoluerende omgeving duurzame producten te ontwikkelen, omdat deze methode veel waarde hecht aan individuen en hun persoonlijke communicaties, aan werkende software in een vroeg stadium van het project, aan de continue samenwerking met de klant en het adequaat reageren op veranderingen.

Communicatie is een belangrijke factor in zowel het 'agile' software ontwikkelingsproces als in het aanmoedigen van kennisdeling. Een sociale bedrijfscultuur en een vlakke organisatiestructuur, al dan niet aangevuld door 'communities of practice', verlagen de drempels voor communicatie en verhogen het vertrouwen tussen de werknemers. Deze factoren alsook beloningssystemen en informatiesystemen hebben een positief effect op het delen van kennis.

De personalisatiestrategie van het kennismanagement, waarvoor het delen van 'tacit' kennis tussen werknemers de basis vormt, is belangrijk om de kennisleemtes op te vullen die ontstaan bij het plotse vertrek van een teamlid en biedt in de meeste gevallen betere resultaten dan de codeerstrategie omdat deze laatste voor een overvloed aan documentatie kan zorgen en desondanks het kennisverlies maar deels beperkt.

Dit wil niet zeggen dat er geen documentatie nodig is. Er moet een bepaalde hoeveelheid van project- en systeemdokumentatie aanwezig zijn maar het blijft moeilijk voor bedrijven om het gepaste documentatieniveau te definiëren. Te veel documentatie kan een grote kost meebrengen voor het bedrijf maar te weinig documentatie kan ook dramatische gevolgen hebben. De 'agile' software ontwikkelingsmethode geeft aanbevelingen voor het opleveren van de juiste hoeveelheid documentatie. De kost van elk document wordt in rekening gebracht en de 'stakeholder' dient elke documentaanvraag te verantwoorden, ook voor de projectdocumentatie. Niet alle documenten die PRINCE2 voorschrijft dienen in elk

project aangemaakt te worden. Dit is afhankelijk van de grootte van het project. De 'templates' voor alle documenten beperken zich best tot de minimale vereisten en het zijn de projectmedewerkers die bepalen welke relevante kennis hieraan nog toegevoegd kan worden.

Uit de gevalstudie blijkt dat er nog verschillen bestaan tussen de theorie en de praktijk op de afdeling van dhr. Grosemans. De nieuwste theorieën worden nog niet toegepast op zijn afdeling en zouden mogelijk een positieve invloed kunnen hebben op het creëren van duurzame producten.

Om betere aanbevelingen te kunnen geven aan de grootbank is het noodzakelijk om nog meer praktijkonderzoek uit te voeren. Bijkomende gesprekken met meer werknemers in verschillende functies en het observeren van het team tijdens een implementatiefase zouden een beter beeld van de huidige situatie kunnen opleveren. Daarnaast kan het ook nuttig zijn om de documenten te bestuderen die dagdagelijks gebruikt worden. Al deze informatie samen kan tot betere aanbevelingen leiden.

Een andere interessante wijze voor verder onderzoek is het implementeren van de 'agile' software ontwikkelingsmethode voor een klein project op de grootbank. Er kan dan onderzocht worden of deze methode de duurzaamheid van de nieuwe producten ten goede komt. Het zal een hele inspanning vergen van zowel de teamleden als de klanten om op deze nieuwe manier te werken en ook het bedrijfsmanagement zal eventueel via een nieuwe bedrijfscultuur het gebruik van deze methode dienen te faciliteren.

## Hoofdstuk 6: Lijst van geraadpleegde werken

- Abrahamsson, P., J. Warsta, et al. (2003). New directions on agile methods: A comparative analysis. *International Conference on Software Engineering (ICSE25)*, Portland, Oregon.
- Al-Alawi, A. I., Al-Marzooqi, N. Y., & Mohammed, Y. F. (2007). Organizational culture and knowledge sharing: critical success factors. *Journal of Knowledge Management*, 11 (2), 22-42
- Argote, L. & Ingram, P. (2000). Knowledge Transfer: A basis for competitive advantage in Firms. *Organizational Behaviour and Human Decision Processes*, 82 (1), 150-169
- Argote, L., Ingram, P., Levine, J. M., & Moreland, R. L. (2000). Knowledge transfer in organizations: learning from the experience of others. *Organizational Behaviour and Human Decision Processes*, 82 (1), 1-8
- Bartol, K. M. & Srivastava, A. (2002) Encouraging knowledge sharing: The role of organizational reward systems. *Journal of Leadership and Organizational Studies*, 9 (1), 64-76
- Bassellier, G., Reich, B. H., Benbasat, I. (2001). Information technology competence of business managers: A definition and research model. *Journal of Management Information Systems*, 17 (4), 159-182
- Cabrera, A. & Cabrera E. F. (2002). Knowledge-sharing Dilemmas. *Organization Studies*, 23 (5), 687-710
- Chaffey, D. & Wood, S. (2005). *Business Information Management: Improving Performance Using Information Systems*. Essex: Pearson Education Limited.
- Chakravarthy, B., McEvily, S., Doz, Y., & Rau, D. (2003). Knowledge management and competitive advantage. In M. Easterby-Smith & M. A. Lyles (Eds.), *The Blackwell handbook of organizational learning and knowledge management*, 305-323. Malden, MA: Blackwell Publishing.

Cockburn, A. (2001). *Agile Software Development*. New Jersey: Addison-Wesley Professional.

Davenport, T. H. & Prusak, L. (2000). *Working Knowledge: How Organizations Manage What They Know*. Boston: Harvard Business School Press.

Gammelgaard, J. (2007). Why Not Use Incentives To Encourage Knowledge Sharing? *Journal of Knowledge Management Practice*, 8 (1)

Grillitsch, W., Müller-Stingl, A., & Neumann, R. (2007). Successful sharing of project knowledge: Initiation, Implementation and Institutionalisation. *The Electronic Journal of Knowledge Management*, 5 (1), 19-28

Hayes, N. & Walsham, G. (2003). Knowledge sharing and ICTs: A relational perspective. In M. Easterby-Smith & M. A. Lyles (Eds.), *The Blackwell handbook of organizational learning and knowledge management*, 54-77. Malden, MA: Blackwell Publishing.

Hislop, D. (2005). *Knowledge Management in Organizations*. New York: Oxford University Press.

Hofer-Alfeis, J. (2008). Knowledge management solutions for the leaving expert issue. *Journal of Knowledge Management*, 12 (2), 44-54

Joshi, K. D., Sarker, S. & Sarker, S. (2006). Knowledge transfer within information system development teams: Examining the role of knowledge source attributes. *Decision Support Systems*, 43, 322-335

Kharabsheh, R. A. (2007). A model of antecedents of knowledge sharing. *The Electronic Journal of Knowledge Management*, 5 (4), 419-426

Lee, D. & Ahn, J. (2006). Reward systems for intra-organizational knowledge sharing. *European Journal of Operational Research*, 180, 938-956

Luckey, T & Phillips, J. (2006). *Software Project Management for Dummies*. Indianapolis: Wiley Publishing, Inc.

Nonaka, I. (1994). A dynamic theory of organizational knowledge creation. *Organization Science*. 5 (1), 14-37

Portman, H. (2009). *PRINCE2 in Practice – A practical approach to creating project management documents*. Zaltbommel: Van Haren Publishing.

Tate, K. (2005). *Sustainable Software Development: An Agile Perspective*. New Jersey: Pearson Education, Inc.

## **Online bronnen**

Ambler, S. W. (2009). *Agile/Lean Documentation: Strategies for Agile Software Development*. Opgevraagd op 2 oktober 2009, van de volgende website:  
<http://www.agilemodeling.com/essays/agileDocumentation.htm>

Ambler, S. W. (2008). Newsflash: Agilists write documentation! Dr. Dobb's Journal. Opgevraagd op 13 december 2009, van de volgende website:  
<http://www.ddj.com/architect/211201940>

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., et. al. (2001). *Manifesto for Agile Software Development*. Opgevraagd op 14 november 2009, van de volgende website: <http://agilemanifesto.org/>

DSDM Consortium (2000). *Using DSDM with PRINCE2*. Opgevraagd op 14 november 2009, van de volgende website:  
<http://www.agilealliance.org/system/article/file/904/file.pdf>

Hey, J. (2004). *The Data, Information, Knowledge, Wisdom Chain: The Metaphorical link*. Opgevraagd op 5 mei 2008, van de volgende website:  
[http://best.berkeley.edu/~jhey03/files/reports/IS290\\_Finalpaper\\_HEY.pdf](http://best.berkeley.edu/~jhey03/files/reports/IS290_Finalpaper_HEY.pdf)

Levin, D. Z., Cross, R., Abrams, L. C., & Lesser, E. L. (2003). *Trust and knowledge sharing: A critical combination*. *IBM institute for knowledge-based organizations*. Opgevraagd op 20 oktober 2008, van de volgende website: <http://www-935.ibm.com/services/in/igs/pdf/g510-1693-00-cpov-trust-and-knowledge-sharing.pdf>

Wallace, S. (2007). *The ePMbook*. Opgevraagd op 12 september 2008, van de volgende website: <http://www.epmbook.com/>

Weisert, C. (2003). *There is no such thing as the Waterfall approach*. *Information Disciplines*. Opgevraagd op 11 januari 2009, van de volgende website:  
<http://www.idinews.com/waterfall.html>

# Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:

**Kennis en documentatie in het software ontwikkelingsproces.      Gevalstudie:  
Het kennisprobleem in het software ontwikkelingsproces van een grootbank**

Richting: **master in de toegepaste economische wetenschappen :  
handelsingenieur in de beleidsinformatica**

Jaar: **2010**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

**Volckaert, Lieven**

Datum: **8/01/2010**