

SNEL EN VEILIG DE CLOUD IN MET PIXELZEBRA

11 01 1111

DEN HAAG

Auteur:

Paddy van der Kaay
10075496

Opdrachtgever:

Robert van der Kleij

Examinatoren:

Peter Ritman
Hamza Soekhai

03-06-2016



*Het ontwerpen en ontwikkelen van een dynamisch CRM
migratieproces van on-premise naar online.*

Versie Beheer

Versie	Datum	Eigenaar	Beschrijving
0.1	10-02-2016	P. van der Kaay	- Opzetten van de structuur van het document.
0.2	11-02-2016	P. van der Kaay	- Sprint 0 toegevoegd; - Opmaak document verbeterd.
0.3	23-03-2016	P. van der Kaay	- H7 gestart; - H8 gestart; - H9 gestart.
0.4	22-04-2016	P. van der Kaay	- Bijlagen bijgewerkt; - H7.3 bijgewerkt.
0.5	25-04-2016	P. van der Kaay	- H7.3 afgerond; - H7.4 bijgewerkt.
0.6	28-04-2016	P. van der Kaay	- H7.4 afgerond.
0.7	29-04-2016	P. van der Kaay	- H8.1 afgerond; - H8.2 bijgewerkt.
0.8	03-05-2016	P. van der Kaay	- H8 afgerond; - 3 t/m 7 toegevoegd.
0.9	09-05-2016	P. van der Kaay	- H2 toegevoegd; - H3 toegevoegd; - H5 bijgewerkt.
0.10	10-05-2016	P. van der Kaay	- H4 afgerond; - H5 afgerond; - bijlagen bijgewerkt.
1.0	13-05-2016	P. van der Kaay	- H6 t/m H11 afgerond; - verwerking commentaar TTA.
1.1	16-05-2016	P. van der Kaay	- Spellingscontrole.
2.0	20-05-2016	R. van der Kleij	- Review + commentaar.
2.1	21-05-2016	P. van der Kaay	- verwerking commentaar Robert van der Kleij.
2.2	24-05-2016	P. van der Kaay	- Bijlagen bijgewerkt.
3.0	30-05-2016	J. Groeneveld	- Review + commentaar.
3.1	31-05-2016	P. van der Kaay	- verwerking commentaar J. Groeneveld.
4.0	01-06-2016	J. Ottevanger	- Spellingscontrole.
4.1	01-06-2016	C. Schrijver	- Spellingscontrole.
4.2	02-06-2016	P. van der Kaay	- verwerken commentaar J. Ottevanger, C.schrijver.
4.3	02-06-2016	P. van der Kaay	- afronding verslag.

Referaat

Dit verslag is geschreven naar aanleiding van een afstudeeropdracht bij de organisatie pixelzebra vanuit de opleiding Business IT & Management, Faculteit IT & Design van de Haagse Hogeschool te Zoetermeer. De werkzaamheden zijn uitgevoerd in de periode van 8 februari 2016 tot en met 3 juni 2016 met een totale duur van 17 weken.

De opdracht betreft het ontwerpen van een CRM-migratieproces waarbij het streven een zo kort mogelijk doorlooptijd met een hoge kwaliteit is. Over drie migratiefasen: voorbereiding, uitvoering en nazorg wordt er per fase een beschrijving gegeven van de werkzaamheden die zijn uitgevoerd om het migratieproces op te zetten. Hierbij komen punten al de orde als: data integriteit, automatisering van de dataoverdracht en aandachtspunten tijdens de nazorg van een CRM-migratie. Het migratieproces is uitsluitend bedoeld voor migraties naar Dynamics CRM Online toe vanuit andere CRM omgevingen zoals Dynamics CRM on premise, Dynamics CRM4 of PerfectView.

Descriptoren:

- ❖ Cloud
- ❖ CRM
- ❖ CRM-migratie
- ❖ Data integriteit
- ❖ Data mapping
- ❖ Data migratie
- ❖ Datatransformatie
- ❖ Dynamics CRM Online
- ❖ Entiteiten
- ❖ ETL
- ❖ Microsoft
- ❖ Office 365
- ❖ Scribe
- ❖ Scribe Online
- ❖ XML

Voorwoord

Pixelzebra zag in mij wat ik in hen zag: enthousiasme. Enthousiasme voor alles wat nieuw en spannend is, enthousiasme voor de chaos die onherroepelijk op ons pad ligt en enthousiasme om hier samen voorover in te duiken. Ik wil Vanessa en Robert heel erg bedanken voor alle steun en kansen die zij de afgelopen twee jaar gegeven hebben. Zonder hen had ik niet gestaan in het leven waar ik nu sta. Verder wil ik graag Jan Groeneveld bedanken voor de gesprekken tijdens het afstuderen. Die hebben mij echt geholpen om het hele traject door te komen. De overige crew van pixelzebra bedank ik ook voor alle gezelligheid en hulp. Ik had mij geen leukere plek kunnen bedenken om af te studeren.

*Even as a bird, 'mid the beloved leaves,
Quiet upon the nest of her sweet brood
Throughout the night, that hideth all things from us,*

*Who, that she may behold their longed-for looks
And find the food wherewith to nourish them,
In which, to her, grave labors grateful are,*

*Anticipates the time on open spray
And with an ardent longing waits the sun,
Gazing intent as soon as breaks the dawn.*

Voor Claudia en de kinderen. Voor wie ik dit alles doe. Bedankt voor al geduld, steun en goede zorg, voor zowel Dante, Mason en mijzelf. De betere tijden komen er nu aan.

Inhoudsopgave

H1 Inleiding	8
H2 Organisatie.....	10
2.1 Bedrijfsstructuur	10
2.2 Bedrijfscultuur.....	11
H3 Stage-opdracht	12
3.1 Probleemstelling	12
3.2 Doelstelling	12
3.2 Gewenste Resultaat	13
H4 Project Framework	14
4.1 Project Development Framework.....	14
4.2 Modelleertechniek.....	15
4.3 Requirements techniek.....	17
4.4 Risico analyse	20
4.5 Planning.....	21
4.6 Communicatie.....	21
H5 Vooronderzoek	23
5.1 Literatuuronderzoek	23
5.2 Interviews.....	29
H6 Het voorbereiden op de migratie	30
6.1 Pre-voorbereiding	31
6.2 Scoping.....	33
6.3 Requirements.....	36
6.4 Inrichting	42
H7 het uitvoeren van de dataoverdracht	48
7.1 Dynamics CRM Online.....	49
7.2 Scribe Online	54
7.3 Scribe CRM Migration Assistant.....	57
H8 Nazorg van de migratie.....	64
8.1 Testen.....	65
8.2 Transformatie na migratie	68
8.3 Nazorg	71

H9 Product evaluatie.....	73
9.1 Literatuuronderzoek	73
9.2 Scribe Online	73
9.3 Scribe CRM Migration Assistant.....	74
9.4 Draaiboek	75
H10 Proces evaluatie	76
H11 Competenties	80
11.1 Business IT & Management	80
11.2 Informatica.....	86
Glossary.....	89
Literatuurlijst.....	92

Bijlage 1: Projectenplan

Bijlage 2: Onderzoeksplan

Bijlage 3: Onderzoeksrapport

Bijlage 4: Migratie draaiboek

Bijlage 5: Template interviewboekje

Bijlage 6: Interviewresultaten

Bijlage 7: Entiteitenmodel

Bijlage 8: Model uitvoeringsstappen

Bijlage 9: Scribe CRM Migration Assistant Views

Bijlage 10: Scribe CRM Migration Assistant Klassendiagram

Bijlage 11: Procesmodellen

Bijlage 12: Voorbeeld backlogitem

Bijlage 13: Uittreksel Productbacklog

Bijlage 14: Voorbeeld Scribe Online Mapping

Bijlage 15: Voorbeeld ETL-schema

Bijlage 16: Mindmap

Bijlage 17: het interveniëren op de bedrijfsgroei van pixelzebra

Bijlage 18: Scrum heeft meer toepassingen dan uitsluitend als software ontwikkel framework

H1 Inleiding

Dit verslag is geschreven ten behoeve van de Haagse Hogeschool. Het verslag geeft een beschrijving van mijn uitgevoerde werkzaamheden gedurende mijn afstudeerperiode van 8 februari 2016 tot en met 3 juni 2016. Het doel van dit verslag is het leveren van bewijs waarmee ik aantoon dat ik, door de Haagse Hogeschool aangeleverde, competenties op HBO niveau heb uitgevoerd.

Ik heb mijn afstudeeropdracht uitgevoerd bij **pixelzebra**, een kleine organisatie dat zich uitsluitend richt op het implementeren van Dynamics CRM Online. Gedurende de opdracht ben ik ingedeeld onder de afdeling consultancy waarbij de Robert van der Kleij (consultant en Managing partner) de rol van opdrachtgever/productowner op zich nam. Mijn rol binnen de organisatie is het beste te omschrijven als CRM consultant/developer. In hoofdstuk twee van dit verslag ga ik verder in op de structuur van **pixelzebra**, haar visie en de bedrijfscultuur.

Pixelzebra wil naast standaard implementaties van Dynamics CRM Online haar aanbod aan de klant verbeteren door CRM-migratietrajecten aan te bieden. Het gaat hierbij om migraties vanuit een vrijwel willekeurige CRM-omgeving naar Dynamics CRM Online toe. De organisatie heeft al ervaring met dergelijke migraties maar, deze zijn tot nu toe altijd ad hoc uitgevoerd zonder gestructureerde werkwijze.

Pixelzebra is ontevreden over de wijze waarop haar migratieproces (of het gebrek hieraan) momenteel verloopt. Met name de doorlooptijd van haar huidige proces is te hoog volgens de productowner van het project. Om **pixelzebra** een betere concurrentiepositie te geven als het gaat om CRM-migraties heeft zij bij mij de vraag neergelegd of ik wil helpen om de doorlooptijd van het migratieproces zo veel mogelijk te verbeteren zonder in te boeten op de kwaliteit van het proces. De volledige opdrachtomschrijving is te vinden in hoofdstuk drie van dit verslag.

Om de kwaliteit van mijn afstudeeropdracht hoog te houden heb ik van tevoren een aantal technieken en methoden aangewezen om te gebruiken tijdens de uitvoering van mijn project. Zo heb ik Scrum gekozen als project framework omdat ik mij daarmee flexibel richting het doel kan begeven. Een ander voorbeeld is mijn keuze voor het vastleggen van requirements en het beheren ervan. Ik heb hierbij gekozen voor de Scrum wijze van requirements vastleggen middels backlogitems met user story opbouw en beheersbaarheidstechnieken zoals een Definition of Ready en Definition of Done. In hoofdstuk vier licht alle keuzes toe die ik heb gemaakt heb als het gaat om het technieken en methoden die bijdragen aan de kwaliteit van mijn project.

Ik ben het project inhoudelijk gestart met een, in hoofdstuk vijf, beschreven onderzoek naar datamigratie. De leidende vraag voor dit onderzoek was: *“Wanneer is een migratie succesvol voor pixelzebra?”*. Middels dit literatuuronderzoek heb ik in kaart gebracht wat best practises zijn rondom datamigratie, welke valkuilen er zijn en hoe dit alles zich verhoudt tot **pixelzebra**. Voor de uitvoering van het onderzoek heb ik gebruik gemaakt van technieken zoals de mindmap (uitgevoerd met medewerkers van **pixelzebra** voor het verkrijgen van zoektermen), prioriteringstechnieken (voor het prioriteren van zoektermen) en controletechnieken voor het bepalen van de kwaliteit van de bronnen.

Na het uitvoeren van een literatuuronderzoek heb ik de definitieve overkoepelende structuur voor mijn datamigratieproces geselecteerd. De eerste fase van dit proces is de voorbereidingsfase. Dit is de fase waarin aspecten zoals Scoping, Requirement analyse, inrichting van de doelomgeving van een migratie en het offerteprocess afgehandeld worden. In hoofdstuk zes geef ik een beschrijving van mijn visie op deze

fase. Daarbij geef ik uitleg over de methodieken die ik pixelzebra aanraad te gebruiken. Een voorbeeld hiervan is het uitvoeren van een data cleansing. Data cleansing is het herstellen van de data integriteit van een systeem door het vinden en corrigeren van corrupte gegevens. Zonder een data cleansing uit te voeren zou de bronomgeving mogelijk geen integere data bevatten. Dit kan op zijn beurt een migratie doen falen. Zo zijn er nog een aantal technieken die ik toelicht in dit hoofdstuk die allemaal zijn opgenomen in een draaiboek dat ik opgeleverd heb aan pixelzebra. Dit draaiboek dient als ondersteunend middel voor het uitvoeren van mijn migratieproces en is te vinden in bijlage 4.

De tweede fase van mijn migratieproces is de uitvoeringsfase. Binnen deze fase heb ik alle handelingen vastgelegd die direct te maken hebben met de inhoudelijke dataoverdracht van een migratie (dus niet de metadata). In het achterhoofd houden dat mijn primaire doel is om de doorlooptijd van migraties voor pixelzebra te verbeteren heb ik voor dit onderdeel voornamelijk gekozen om tijdwinst te behalen met behulp van automatisering. Dit heb ik gerealiseerd middels het gebruik van een standaard Cloud-based migratietool met de naam Scribe Online en een door mij ontwikkeld stuk software met de naam Scribe CRM Migration Assistant. Het maatwerk is ontwikkeld om bepaalde aspecten van Scribe Online te verbeteren zoals heb mappen van entiteiten en het rapporteren na afloop van de migratie. In hoofdstuk zeven licht ik beide softwareproducten uitgebreid toe. Tot slot geef ik meer inzicht in de werking van Dynamics CRM Online om zo een beeld te schetsen van de omgevingen en de gevolgen daarvan op mijn migratieproces.

In het achtste hoofdstuk van mijn verslag beschrijf ik mijn inrichting van de laatste stap van mijn migratieproces: de nazorgfase. Kenmerkend voor deze fase zijn aspecten als het testen van de migratieresultaten middels een user acceptance test en vergelijkingstest op het aantal gemigreerde records, het omgaan met een trickle na de migratie en het verkrijgen van decharge. Naast inhoudelijke aspecten rondom de migratie bevat dit hoofdstuk nog de resultaten van een pragmatische testmigratie van standaard entiteiten. Deze resultaten zijn bedoeld om een indicatie te geven van de snelheid waarmee een dataoverdracht wordt uitgevoerd middels mijn migratieproces.

De hoofdstukken negen, tien en elf bevatten evaluaties met betrekking tot mijn afstuderen. Het gaat hierbij om een productevaluatie waarin de mening van de productowner en ikzelf vertegenwoordigd zijn, een proces evaluatie waarin ik middels de STARR methode reflecteer op mijn uitgevoerde werk en een lijst van de beroepstaken die ik behaald heb tijdens mijn afstuderen inclusief motivatie.



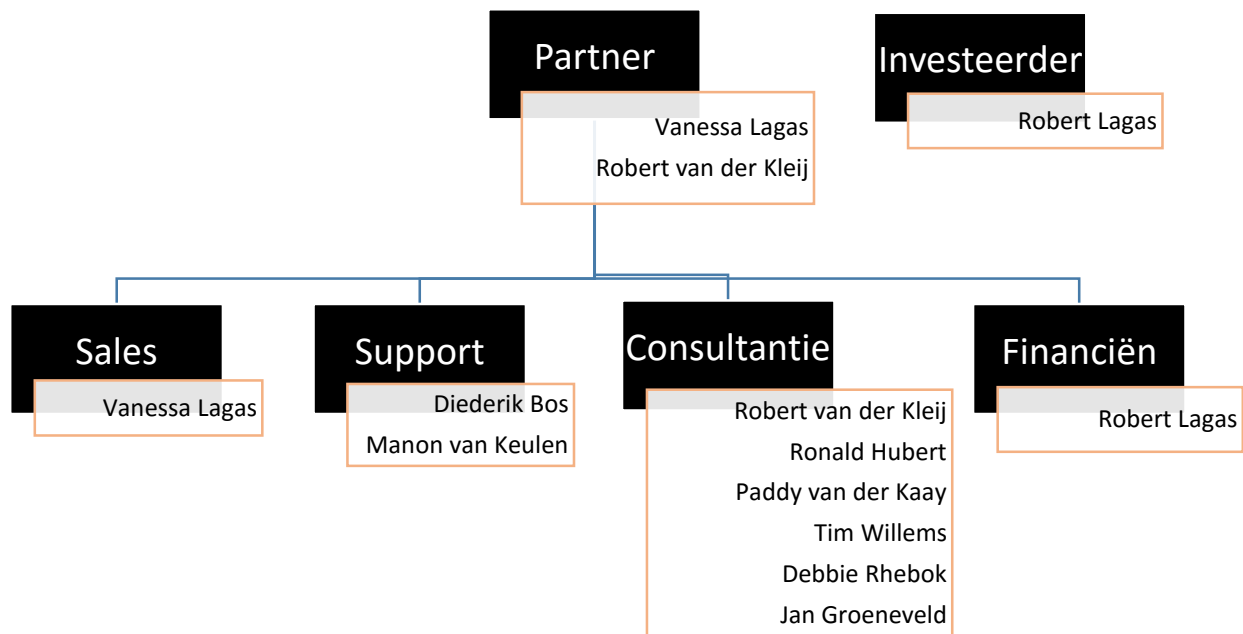
H2 Organisatie

Pixelzebra is een Microsoft partner en leverancier van Microsoft Dynamics CRM. Het bijna vijf jaar oude bedrijf telt op het moment van schrijven 10 medewerkers en dit aantal groeit gestaag. De steile groei die het bedrijf doormaakt is reflectief voor het succes van de organisatie. Met klanten zoals Sator, Residentieorkest en stichting 4&5 mei is het bedrijf druk bezig om zichzelf op de kaart te zetten.

Gestationeerd in de Caballerofabriek te Den Haag is pixelzebra een innovatieve organisatie die zich kenmerkt door haar constant evoluerende karakter (Kaay, 2015). Pixelzebra is een organisatie die leeft op onderlinge samenwerking in tegenstelling tot onderlinge concurrentie. Dit gaat gepaard met een drang om constant te leren van zowel haar omgeving, branche-innovaties als haar eigen fouten.

2.1 Bedrijfsstructuur

Ondanks de kleinschaligheid van de organisatie is zij druk bezig om zichzelf qua structuur te professionaliseren. Dit heeft als gevolg dat pixelzebra een duidelijk ingedeelde bedrijfsstructuur heeft met typerende afdelingen (figuur 1). De medewerkers van pixelzebra hebben naast hun standaard taken vaak nog een aantal neventaken, zo kan het voorkomen dat dezelfde medewerker daardoor onder meerdere afdelingen van de organisatie valt.



Figuur 1: De organisatiestructuur van pixelzebra

Henry Mintzberg heeft een typologie ontwikkeld om organisatiestructuren te kunnen indelen. Binnen deze typologie heeft hij vijf basisconfiguraties gedefinieerd waaronder bedrijven in te delen zijn. Deze configuraties beschrijven bepaalde manieren van omgaan met coördinatie, besluitvorming en waar de basis van het bedrijf ligt. Pixelzebra



Figuur 2: Mintzberg 's 'Simple structure'

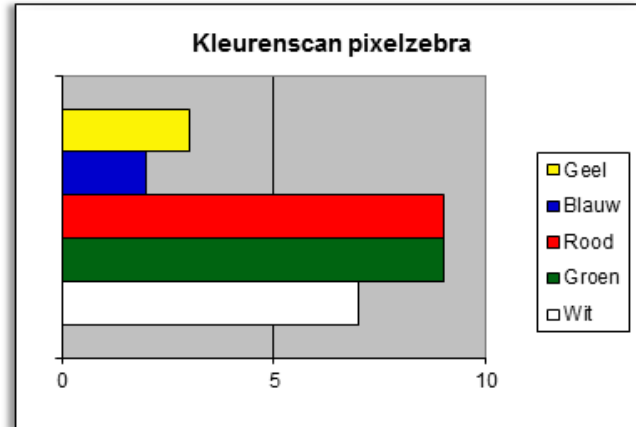


is goed in te delen in deze typologie en hanteert zonder twijfel de 'Simple Structure' (Mintzberg, 2013). Zo heeft zij een strategische apex; deze bestaat meestal uit één persoon. In het geval van Pixelzebra zijn dit de twee partners. Daarnaast heeft zij een duidelijke verticale en horizontale centralisatie, ondanks inbreng van alle medewerkers liggen de eindbeslissingen altijd bij het management. Het gevolg van de Simple Structure is dat de medewerkers, met name de consultants, veel eigen verantwoordelijkheid hebben. In de praktijk betekent dit dat consultants veelal eigen beslissingen maken omtrent keuzes in een project. Het management zal meestal pas ingrijpen bij 'gevoelige' projecten of als projecten dreigen te ontsporen.

2.2 Bedrijfscultuur

In een eerder onderzoek binnen pixelzebra heb ik de Caluwé Kleurentest gebruikt om de cultuur binnen de organisatie te schetsen. De test categoriseert een organisatie over vijf kleuren die representatief zijn voor een bepaald gedrag. Het doel van de resultaten is om te controleren of een verandertraject geaccepteerd zal worden of niet. Met behulp van de kleurentest kan ik mijn project sturen richting een oplossing die aansluit op de organisatie.

De afgenomen kleurentest toont aan dat pixelzebra voornamelijk een rood/groene organisatie is dat eveneens sterk leunt richting wit. In mijn eerder uitgevoerde onderzoek heb ik daaruit de volgende conclusie getrokken: pixelzebra is een bedrijf waarin ieder individu een sterke intrinsieke motivatie bezit. Het primaire doel van de medewerkers is om pixelzebra een succes te maken; niemand zit er voor eigen gewin. Het gevolg is dat er een sterke familiesfeer hangt binnen de organisatie.



Figuur 3: Resultaten Caluwé kleurentest voor pixelzebra

Alle medewerkers binnen de organisatie genieten een hoge mate van eigen verantwoordelijkheid. Dit betekent eveneens dat pixelzebra verwacht van haar medewerkers dat zij proactief aan zelfstudie doen om de juiste keuzes te kunnen maken. De organisatie stimuleert haar medewerkers daarnaast ook actief om te blijven leren door middel van het financieren van studies en door het bieden van ruimte aan haar medewerkers om een studie uit te voeren.

De laatste heersende karakteristiek binnen pixelzebra is die van dynamiek, complexiteit en verandering. De karakteristieken die hierboven zijn beschreven hebben als gevolg dat pixelzebra zichzelf continu verbetert en verandert. Dit maakt de organisatie ietwat onvoorspelbaar, waardoor de onderlinge communicatie niet altijd sterk en het inplannen van afspraken bemoeilijkt wordt.

H3 Stage-opdracht

Pixelzebra's primaire diensten bestaan uit het implementeren en onderhouden (support) van Microsoft Dynamics CRM omgevingen. Dit doet zij exclusief met de Cloud versie van Dynamics CRM en altijd de nieuwste versie; op het moment van schrijven Dynamics CRM 2016. Het gros van pixelzebra's potentiële klanten zijn bedrijven die op zoek zijn naar een CRM-systeem. In de basis is dit voor marketing/sales doeleinden maar, dit kan variëren. De grote gelijktrekker is ieder geval dat zij nog geen CRM-systeem bezitten.

Pixelzebra is recentelijk op zoek naar 'veilige' manieren om haar marktaanbod te verbreden. Enerzijds wil zij dit doen door het dienstenaanbod te vergroten middels Power BI. Anderzijds wil zij dit doen door een variatie op de standaard CRM-implementatie aan te bieden: een CRM-migratie.

Veel organisaties werken met verouderde of slecht-ingerichte CRM-systemen. Deze organisaties zijn actief op zoek naar een nieuw CRM-systeem dat kan voorzien in de behoeften die hun huidige systeem niet (meer) kan leveren. Voor pixelzebra zijn dit open kansen voor het uitvoeren van een CRM-migratie.

3.1 Probleemstelling

Pixelzebra heeft een ruime ervaring met CRM-migraties, de uitvoering hiervan was echter ad hoc en ongestructureerd. Hierdoor is vrijwel geen sprake van een interne benchmark waarmee de kwaliteit van migraties vergeleken kan worden. Naar eigen zeggen vindt de productowner de migraties voornamelijk te lang duren. *"Het migreren van sommige omgevingen kan soms weken duren."* aldus: de productowner. Een migratie met een lange duur is onwenselijk voor beide partijen van de migratie. Daarbij komt nog eens dat lang niet alle migraties het doel halen dat de klant voor ogen had. Naast het realiseren van haar doelen wil de klant voornamelijk zo min mogelijk hinder ondervinden aan het migratieproces. Daarentegen wil pixelzebra zo min mogelijk uren gebruiken voor het migreren om zo de klantpropositie meer waarde te geven en de winst te maximaliseren.

Pixelzebra biedt CRM-migratieprojecten voornamelijk fixed-price aan. Hoe langer de migratie duurt, hoe meer geld dit kost voor de organisatie. Om CRM-migraties rendabel te maken voor de organisatie moet de doorlooptijd ervan omlaag, anders weegt de investering vanuit pixelzebra niet op tegen de baten. Volgens de productowner is het probleem van pixelzebra dan ook eenvoudig. Pixelzebra heeft de kennis om een migratie goed uit te voeren. Zij mist alleen een proces en de middelen om dit snel en goed te doen.

3.2 Doelstelling

Het doel van dit project is om de gemiddelde doorlooptijd van pixelzebra's CRM-migraties zoveel mogelijk te verminderen. Het uiteindelijke droomdoel hierbij is een mogelijkheid voor pixelzebra om een geheel migratietraject af te leggen in slechts zeven dagen. Het gaat hierbij om het gehele traject vanaf de aftrap van het project tot en met het verkrijgen van decharge. De productowner zal echter tevreden zijn als de verwachte doorlooptijd significant verminderd is ten opzichte van pixelzebra's huidige ad hoc werkwijze. Een randvoorwaarde hiervoor is wel dat de kwaliteit van de geleverde diensten niet mag dalen en, bij voorkeur zelfs stijgt.

3.2 Gewenste Resultaat

Tijdens het intakegesprek met de productowner Robert van der Kleij, kwam naar voren dat hij een aantal verwachtingen heeft van het project. Zo wil hij dat de dataoverdracht geautomatiseerd wordt met behulp van een datamigratiepakket. Zijn verwachting hierbij is dat dit niet zal gaan zonder de hulp van een miniem stuk maatwerk. Buiten deze verwachtingen om heeft de productowner mij de vrije hand gegeven om het proces goed mogelijk in te richten met zo'n laag mogelijke doorlooptijd. Op basis van het gesprek met de productowner ben ik gekomen tot de volgende lijst met resultaten die dit project als resultaat moet hebben om de productowner tevreden te houden en om te voldoen aan de randvoorwaarden van mijn projectplan:

- Een gestructureerd migratieproces met een lagere doorlooptijd dan de huidige ongestructureerde werkwijze van pixelzebra;
- Een stuk maatwerk dat de dataoverdracht versneld;
- Een selectie en inrichting van een standaard migratietool om de dataoverdracht te versnellen;
- Een eenvoudige beschrijving van het migratieproces;
- De kwaliteit van de CRM-migratie moet minstens kwalitatief zo goed zijn als pixelzebra's huidige werkwijze.

H4 Project Framework

De productowner wenst dat ik gebruik maak van bewezen methoden voor het uitvoeren van het project. Om aan zijn eisen te voldoen, en om de kwaliteit van mijn project hoog te houden, heb ik voor een aantal onderdelen van het project technieken of methoden geselecteerd die mij zullen helpen om de kwaliteit van mijn project te waarborgen. Per gekozen techniek of methodiek lever ik in dit hoofdstuk een korte beschrijving, mijn beweegredenen voor mijn keuze en een motivatie voor het niet selecteren van een alternatief.

4.1 Project Development Framework

In overleg met de opdrachtgever heb ik gekozen voor Scrum als Project Framework. Gebaseerd op een artikel uit de Harvard Business Review paper genaamd “The New New Product Development Game” levert Scrum een framework dat de mogelijkheid biedt om softwareprojecten middels een agile en iteratieve werkwijze uit te voeren.

Scrum onderscheidt zich van projectmanagementmethodieken zoals Prince2 of software-ontwikkelprocessen zoals RUP door het gebrek aan inhoudelijke artefacten (Solingen & Rustenburg, 2010). Dit hangt samen met de filosofie van agile die werkende producten boven documentatie stelt (Sutherland, 2001). Mijn project bevat meer dan alleen softwareontwikkeling en ondanks het originele gedachtegoed achter Scrum is het Framework goed toepasbaar op mijn project (Kaay, 2014).

Als het aankomt op het faciliteren van controle binnen een project laat Scrum zijn gebruikers een beetje in het diepe achter. Het framework verwacht dat een Scrumteam dit zelf oplost middels de aangeboden communicatiemomenten en/of aanwezige methoden zoals de Definition of Done (Solingen & Rustenburg, 2010). Ik heb gekozen voor de aanpak zoals deze wordt gehanteerd binnen pixelzebra. Zij maakt gebruik van een Definition of Ready en Definition of Done. Uitgebreid toegelicht in hoofdstuk 4.3, zijn de Definition een krachtig middel om controle over mijn uit te voeren werk te behouden zonder te veel overhead te creëren binnen mijn project.

Beweegredenen

Zoals beschreven in de bovenstaande alinea is de keuze voor Scrum in overleg met de opdrachtgever gemaakt. Het doel van het project is het behalen van tijdwinst. De specifieke wijze waarop dit het effectiefst gedaan kan worden is voorafgaand aan het project echter nog niet duidelijk. Een agile methodiek zoals Scrum helpt om snel te reageren op het moment dat er een onderdeel gevonden is dat meer aandacht behoeft.

Ik vind het lastig actief te communiceren. Ik heb geen problemen met de communicatie zelf maar, ik neig veelal te zelfstandig te willen werken. Scrum helpt mij om dit probleem op te lossen doordat het structureel communiceren forceert. Het enige communicatiemoment dat ik achterwege heb gelaten is de daily standup. Deze wordt normaliter uitgevoerd met het Scrumteam. Aangezien ik de enige persoon ben in mijn Scrumteam heeft dit geen toegevoegde waarde. Mogelijk had ik de opdrachtgever(product owner) kunnen betrekken in de daily standups, gezien zijn drukke agenda zag ik dit echter niet als realistische optie. Ik heb dit afgevangen door regelmatig problemen voor te leggen aan consultants die op dat moment beschikbaar waren. De belangrijkste communicatie zal lopen via de sprint reviews en sprint retrospectives. Deze momenten zullen fungeren als primaire momenten voor overleg en reflectie.

Alternatieven

Hoewel de opdrachtgever de voorkeur geeft aan het gebruik van Scrum (dit is de standaard binnen pixelzebra) heb ik een vergelijking gemaakt met een aantal veelgebruikte alternatieven die ik gedurende mijn opleiding eerder toegepast heb. Dit heb ik gedaan om te controleren of Scrum daadwerkelijk de beste keuze is voor dit project. De onderstaande tabel toont een overzicht van de Scrum, waterval en RUP methodieken met daarbij de kernwaarden en de zwakten per methodiek.

	Type	Kernwaarden	Zwakten
Scrum	Agile	Zelfstanding, klein, zelf organiserend, structurele communicatie.	Verleidelijk om geen documentatie vast te leggen. Biedt eveneens inhoudelijk beperkte middelen om dit te doen.
Waterval	Lineair	Hoge mate van structuur, vooraf gepland, eenvoudig, nadruk op documentatie.	Geen flexibiliteit, projectkwaliteit is sterk afhankelijk van de eerste projectfasen.
RUP	Iteratief	Verzameling best practises uit de praktijk, gestructureerd, onderdelen kunnen los van elkaar toegepast worden, documentatie templates.	Veel overhead, complex.

Figuur 4: Vergelijking van projectmethodieken

Kijkend naar de zwakten van de waterval en RUP methoden wordt vrijwel meteen duidelijk waarom ik hier niet voor gekozen heb. De watervalmethode biedt weinig flexibiliteit, iets wat ik nodig heb omdat ik de precieze oplossingsrichting nog niet ken (Royce, 1970). De RUP methodiek heeft veel overhead en is complex. Hoewel dit de kwaliteit van het project goed waarborgt denk ik dat ik met RUP te veel tijd kwijt ben aan overhead, waardoor ik te weinig tijd overhoud voor inhoudelijke werkzaamheden (Rational, 2001). Scrum sluit vrijwel naadloos aan op de eisen voor mijn project, het is toepasbaar op een klein project met veel zelfstandigheid en het forceert structurele maar eenvoudige communicatie (Solingen & Rustenburg, 2010). Zelfs na vergelijken met andere methodieken heb ik daarom gekozen voor Scrum voor mijn afstudeerproject.

4.2 Modelleertechniek

Vanuit mijn opleiding ben ik met name bekend de Unified Modelling Language (UML). UML is een modelleringstaal bedoeld voor het ontwerpen en analyseren van informatiesystemen. Een kenmerkende eigenschap van UML is het brede scala aan modellen die het bevat om vrijwel ieder aspect van een informatiesysteem in kaart te brengen. UML wordt beheerd door de Object Management Group. Deze organisatie zorgt dat UML altijd up-to-date blijft met de laatste technieken. Een belangrijke kanttekening is wel dat ik UML niet tot op de letter gevolgd heb. In een aantal gevallen heb ik technieken vereenvoudigd om mijn modellen te verduidelijken.

Het UML dogma schrijft voor dat er een volgordelijkheid zit in de gebruikte modellen. Dit heeft een significant voordeel in standaard softwareontwikkelpromen omdat het een mate van traceerbaarheid aanbrengt binnen het project. Hoewel mijn project een stukje softwareontwikkeling bevat zal ik in mijn project verscheidene onderdelen van verschillende tools modelleren. Deze onderdelen hebben geen directe aansluiting waardoor volgordelijkheid geen toegevoegde waarde heeft, ik heb deze dan ook achterwege gelaten.

Beweegredenen

Mijn eigen persoonlijke voorkeur is niet de enige reden waarom ik voor UML als modelleertechniek heb gekozen. Binnen mijn project zal ik in ieder geval vier onderdelen van Dynamics CRM moeten modelleren. Dit zijn:

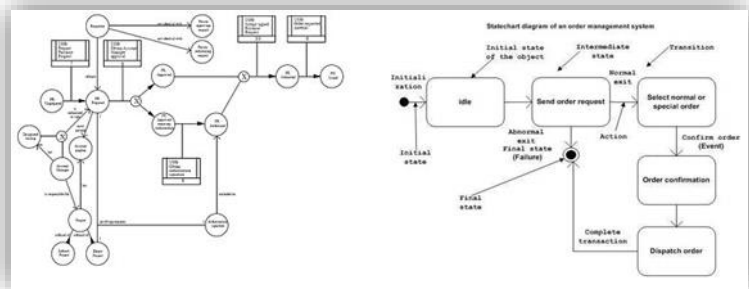
- De databasestructuur;
- De klassenstructuur van maatwerk;
- De migratie datamappings,
- De processen binnen Dynamics CRM en Office 365.

Voor ieder van deze behoeften heeft UML een model die daarop aansluit en voor vrijwel iedere soortgelijke behoefte die ik nog kan hebben tijdens mijn afstudeerproject (Warmer & Kleppe, 2011). Een laatste niet onbelangrijk argument waarom ik heb gekozen voor UML is het feit dat dit de standaard is binnen pixelzebra. Het is daarom een effectieve methode om te communiceren met de opdrachtgever en overige medewerkers van pixelzebra.

Alternatieven

Er zijn vrijwel geen gelijkwaardige alternatieven die op gelijke hoogte staan met het modelleren aanbod van UML. De enige soortgelijke verzamelingen van methoden die enigszins bruikbaar zijn voor mijn project zijn: Integration Definition (IDEF) en Object-Process Methodology (OPM). Een derde alternatief zou het gebruik van verschillende losse modeleringstechnieken zijn.

IDEF is voor mij meteen afgefallen. Het aanbod qua modellen is breed en bevat genoeg modellen om in alle behoeften te voorzien waarin UML kan voorzien. De notatie van IDEF wijkt echter sterk af van UML. Ik zie dat ook geen voordelen ten opzichte van UML. Tot slot is IDEF versplinterd over vele adaptaties waardoor het moeilijk is om binnen één versie te blijven (Noran, 20114). Ik ben bang dat dit verwarring voor zowel mijzelf als pixelzebra oplevert. Figuur 5 toont een vergelijking tussen de notaties voor een State Diagram. Hoewel zij hetzelfde kunnen qua modellering toont het figuur dat de notatie sterk verschilt.



Figuur 5: De verschillen in notatie tussen IDEF (links) & UML (rechts)

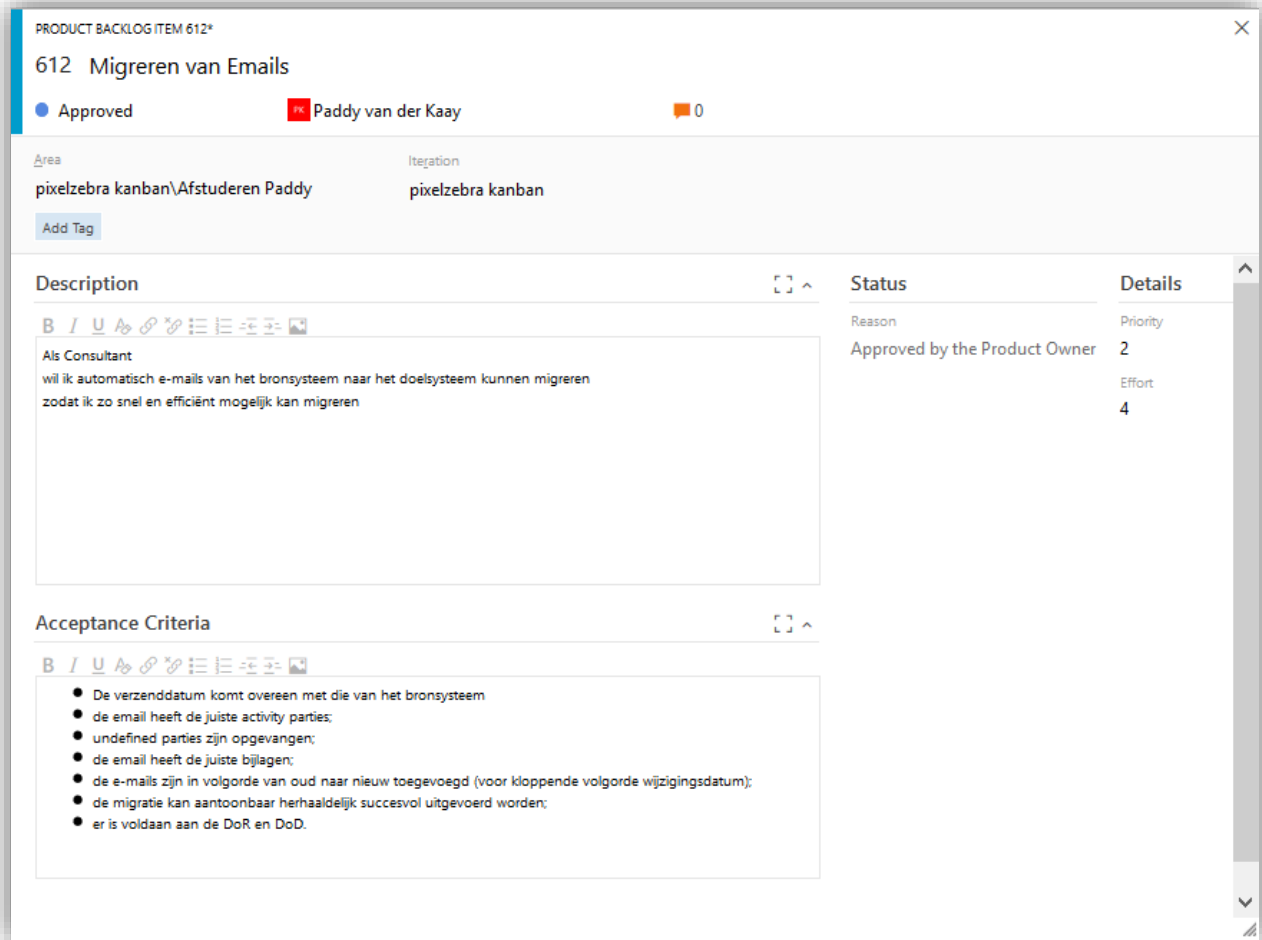
Volgens een studie naar de verschillen tussen OPM en UML is de eerstgenoemde superieur ten opzichte van UML als het gaat om het modelleren van dynamische onderdelen in webapplicaties (Reinhartz-Berger & Dori, 2005). Als het gaat om het modelleren van statische onderdelen zoals databasestructuren is er geen significant verschil tussen beide. Gezien de leercurve die nodig is om OPM aan te leren zie ik UML, dat al bekend is binnen pixelzebra, als betere optie.

Tot slot heb ik de optie voor losse modeleringstechnieken laten vallen omdat deze optie de nadelen van zowel IDEF als OPM samenbrengt. Verschillende technieken door elkaar heen gebruiken kan verwarrend werken en dit zorgt ervoor dat de modellen hun doel voorbij schieten. Ze zijn immers bedoeld ter verduidelijking van complexiteit. Daarnaast is er een leercurve voor zowel mijzelf als medewerkers van pixelzebra om de modellen te begrijpen.



4.3 Requirements techniek

Voor het vastleggen van requirements heb ik mij gehouden aan de standaard van pixelzebra. Zij maken gebruik van een combinatie van user stories met acceptatie criteria, een Definition of Ready en een Definition of Done. Ik heb dit beheerd in het Kanban bord van Pixelzebra dat gehost wordt in de Visual Studio Online omgeving van de organisatie. Figuur 6 toont een voorbeeld van een user story zoals ik deze heb gebruikt tijdens mijn project.



Figuur 6: Voorbeeld van een backlogitem, gemaakt in Visual Studio Online

User stories zijn de standaard wijze waarop Scrum requirements vastlegt. Om deze stories van nature (te) veel vrijheid geven aan de developer zijn de acceptatiecriteria toegevoegd aan het Framework. De acceptatie criteria zijn niets meer dan een lijst met specifieke vereisten waaraan een user story moet voldoen voordat deze als 'Done' beschouwd mag worden. In het geval van pixelzebra moet een story zowel voldoen aan de acceptatie criteria als de Definition of Done.

Daar waar de acceptatie criteria specifiek gelden voor één user story geldt de Definition of Done voor alle stories. De Definition of Done is daarom niets meer dan een lijst met globale acceptatie criteria die gelden voor iedere story. De Definition of Ready is het minder bekende broertje van de Definition of Done. Daar waar de Definition of Done een lijst met criteria bevat waaraan een user story moet voldoen om als afgerond beschouwd te mogen worden, bevat de Definition of Ready juist een lijst met criteria waaraan voldaan moet zijn voordat een user story opgepakt kan worden. Figuur 7 bevat de Definition of Done en

Definition of Ready zoals ik deze heb gebruikt binnen mijn project. Voor zowel de Definition of Ready als de Definition of Done geldt dat ik de template van pixelzebra heb aangepast aan mijn eigen behoeften. Zo heb ik de acceptatiecriteria en voorwaardelijke documentatie aangepast om te voldoen aan mijn eigen behoeften, bijvoorbeeld door het opnemen van resultaten in mijn afstudeerverslag op te nemen als criterium in de Definition of Done.

	Definition of Ready	Definition of Done										
Acceptatie criteria	- Het backlogitem is opgesteld volgens de onderstaande descriptie; - De descriptie is SMART opgesteld; - Het backlogitem is opgenomen in de Visual studio team server; - Het backlogitem is geprioriteerd en heeft een geschatte effort; - Het backlogitem heeft een unieke naam en nummer; - Het backlogitem heeft een datum; - Er is aan alle voorwaardelijke documentatie voldaan; - Het backlogitem is vrijgegeven door de productowner; - Alle acceptatiecriteria zijn voor het backlogitem vastgelegd.	- Er is voldaan aan de Definition of Ready; - De resultaten van het backlogitem zijn gereviewed en goedgekeurd door de opdrachtgever; - De handelingen en resultaten van het backlogitem zijn verwerkt in het afstudeerverslag; - Er is voldaan aan de acceptatie criteria voor het backlogitem; - De overgebleven effort is op nul gezet; - De resultaten van de backlogitem zijn in de Cloud opgeslagen.										
Descriptie	Vorm: “Als <type gebruiker> wil ik dat <actie> zodat <achterliggende reden> Voorbeeld: Als Consultant wil ik automatisch e-mails van het bronsysteem naar het doelsysteem kunnen migreren zodat ik zo snel en efficiënt mogelijk kan migreren	Volgens Definition of Ready										
Voorwaardelijke documentatie	<table><tr><td>Interface: beschrijving;</td><td>Views +</td></tr><tr><td>Software:</td><td>Datamodel, Klassendiagram;</td></tr><tr><td>Mapping:</td><td>ETL-schema; Databasemodel;</td></tr><tr><td>Onderzoek:</td><td>Onderzoeksplan;</td></tr><tr><td>Automatisering:</td><td>Activity diagram.</td></tr></table>	Interface: beschrijving;	Views +	Software:	Datamodel, Klassendiagram;	Mapping:	ETL-schema; Databasemodel;	Onderzoek:	Onderzoeksplan;	Automatisering:	Activity diagram.	Volgens Definition of Ready
Interface: beschrijving;	Views +											
Software:	Datamodel, Klassendiagram;											
Mapping:	ETL-schema; Databasemodel;											
Onderzoek:	Onderzoeksplan;											
Automatisering:	Activity diagram.											

Figuur 7: De Definition of Ready & Definition of Done

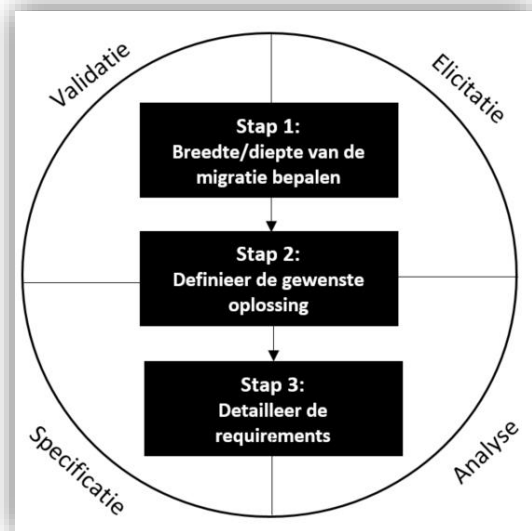
Beweegredenen

Zoals ik in de bovenstaande alinea heb beschreven is het gebruik van user stories, acceptatie criteria, Definition of Done en Definition of Ready de standaard binnen pixelzebra. Alle projecten binnen de organisatie worden tevens gemanaged binnen het Kanban bord in Visual Studio Online. Mijn voornaamste reden voor dit gebruik is dan ook geweest om gestroomlijnd mee te draaien met de rest van de organisatie. Het gebruik van het Kanban bord heeft tevens alle medewerkers binnen de organisatie inzicht gegeven in mijn werkzaamheden, hetgeen de openheid van mijn project heeft bevorderd.

De tweede reden voor het selecteren van deze werkwijze is de simpliciteit ervan. Ik heb moeite om mijn te houden aan sterk gefaseerde processen waarbij ik veel overhead heb. Het gebruik van de Definition of Ready en Definition of Done zorgen ervoor dat ik gestructureerd werk zonder zware statische eisen te stellen. Er zijn geen voorwaarden voor de acceptatie criteria voor de Definitions. Hierdoor kan ik zelf de mate van controle instellen en bijstellen als het mijn werkzaamheden te veel beperkt .

Alternatieven

Naast de werkwijze van pixelzebra heb ik slechts één alternatief overwogen en dat is het vastleggen van requirements volgens de Swart (2010). De Swart beschrijft een requirementsproces waarin requirements vastgelegd worden volgens de categorisering: Business requirements, Gebruikersrequirements en systeem requirements. Daarnaast beschrijft zij een uitgebreid proces voor het verkrijgen en verwerken van requirements. Het vastleggen van requirements zoals de Swart dat beschrijft is ontwikkeld door Abran & Moore (2004). Het proces bevat drie stappen die daarnaast verdeeld zijn over vier vlakken. Het uitvoeren van dit proces heeft veel overhead. Ik zou te veel tijd kwijt zijn aan het vastleggen van requirements ten opzichte van het uitvoeren. Pixelzebra's methode van Scrum in combinatie met de Definition of Done en Definition of Ready levert een eenvoudigere methode de beter aansluit op mijn behoeften.



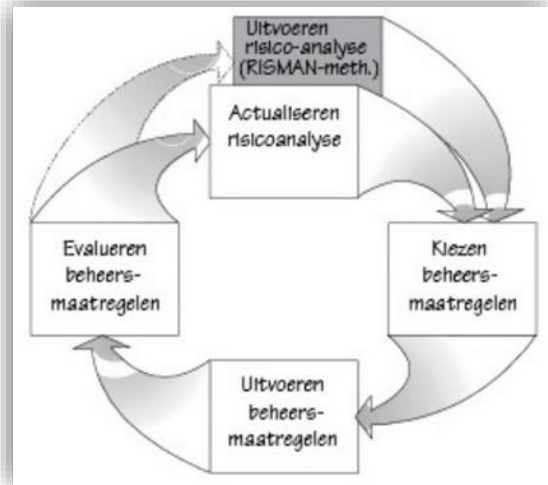
Figuur 8: het requirementsproces van Abran & Moore (2004)



4.4 Risico analyse

Voor het analyseren van de risico's binnen mijn project heb ik voor een ietwat alternatieve techniek gekozen, namelijk de cyclische risicomanagementmethode volgens Well-Stam (2013). De cyclische risicomanagementmethode legt de focus op een lerende risicoanalyse die de nadruk legt op beheersmaatregelen in tegenstelling tot het risico zelf. Daarnaast raadt de methode aan om niet alleen negatieve maar ook positieve risico's op te nemen. De weging, die op basis van de beheersmaatregelen opgesteld wordt in plaats van het risico zelf, wordt aan de hand van de volgende criteria vastgelegd:

- Tijd,
- Geld,
- Kwaliteit,
- Informatie,
- Organisatie.



Figuur 9: De risicomanagementmethode volgens Well-Stam (2013)

Beweegredenen

De cyclische risicomanagementmethode sluit naar mijn mening uitstekend aan op een Scrum project. Het lerende karakter van de methode komt goed tot zijn recht binnen het gebruik van sprints. Zo kan ik iedere sprint een cyclus van de risicomanagementmethode doorlopen.

Een tweede punt dat mij aansprak voor het gebruik van de methode is de focus op tegenmaatregelen in plaats van het risico zelf. Door naast normale risico's ook positieve risico's op te nemen, helpt de methode bij het inschatten van effort. Ik kan namelijk de doorlooptijd van een tegenmaatregel opnemen tijdens het plannen van een sprint. Dit bevordert de juistheid van de ingeschatte efforts voor mijn user stories.

Alternatieven

De standaard methode die ik voorheen altijd heb gebruikt is de methode voor het analyseren van risico's zoals beschreven in het boek: Kosten, Baten en Risico's van ICT-investeringen (Noordam & Zalm, 2012). Dit is een risicoanalyse die kans, impact en tijd weegt per risico en beschrijft daarvoor een tegenmiddel. Omdat ik mijn project agile uitvoer vind ik met name het tijd criterium niet goed aansluiten. Tijdens het project kan ik sterk wisselen in de werkzaamheden die ik uitvoer. Hierdoor zou ik constant de weging sterk moeten bijstellen. Deze risicoanalyse methode zou naar mijn mening beter tot zijn recht komen in een lineair project.

4.5 Planning

De planning voor het project heb ik uitgezet in een aantal fasen die gelijk zijn aan een standaard datamigratieproces. Binnen deze fasen heb ik een aantal taken beschreven die ik wil uitvoeren met daarbij een schatting van de tijd die ik hier aan kwijt ben. Wat mij betreft ligt de volgorde van de fasen en taken noch de tijdsschatting vast. Ik wil graag de focus leggen op de onderdelen van het migratieproces waar de meeste tijdswinst te behalen valt. Dit weet ik van te voren nog niet, om deze reden wil ik een flexibele planning die aansluit op het gebruik van Scrum.

Beweegredenen

Hoewel Agile het nut van een planning beaamd, is het doel niet gelijk aan dat van een lineair project (Greer & Conradi, 2008). De planning van een lineair project gebruikt een planning om de duur van een project, inclusief handelingen, te bepalen en is daarmee een controlerend middel. Bij een Agile project dient de planning voornamelijk om het project te scopen door het definiëren van aandachtspunten (zoals de fasen van een migratieproces). Omdat ik alleen een streefdoel heb (een zo efficiënt mogelijk migratieproces voor pixelzebra) zonder dat daar vaste werkzaamheden binnen zijn gedefinieerd is de noodzaak voor een volledig uitgezette planning afwezig. Een dergelijke planning heeft ook geen plaats binnen een Scrum project. Binnen een dergelijk project als het mijne kan een uitgebreide, in de tijd uitgezette, planning alleen maar zorgen voor een project waarvan de resultaten mogelijk niet aansluiten op het opgestelde streefdoel.

*“Stel je wilt op vakantie naar Rome, bedenkt wat je wilt doen en zien en belandt vervolgens in Berlijn. Dan komt dat doordat je blijkbaar niet naar Rome wilde. Met een klassieke planning zou je in Rome zijn uitgekomen om je vervolgens suf te zoeken naar een gezellige **Bierstube**.”*

Figuur 10: Beschrijving van de tekortkomingen van volledig geplande projecten.

Alternatieven

De enige andere wijze om te plannen die ik heb kunnen identificeren is de klassieke manier van planning middels een schema of Gantt chart. Zoals ik in de bovenstaande alinea heb beschreven sluit deze wijze van planning eenvoudigweg niet aan op mijn project. Ik besef mij dat het gebrek aan een planning kan leiden tot gebrekkige structuur wat betreft mijn project. Mijn argument hiervoor is dat de sprint reviews, sprint retrospectives, Definition of Done en Definition of Ready mij voldoende handvatten geven om het project richting mijn doelstelling te sturen zonder de scope van mijn project te buiten te gaan.

4.6 Communicatie

Voor de communicatie gedurende het verloop van mijn project heb ik gekozen voor de standaard momenten die Scrum aanbiedt. Dit zijn: De sprint planning sessie, de sprint review en de sprint retrospective.

De sprint planning sessie wordt gebruikt om een nieuwe sprint in te planning. Tijdens deze sessie worden backlogitems geselecteerd, geprioriteerd en wordt er een effort aan toegedeeld. Dit is tevens het moment waarop het doel (bij voorkeur een ‘tastbaar’ product) wordt bepaald.

De sprint review is bedoeld om de sprint te inspecteren. Op basis van de review worden er mogelijk nieuwe items toegevoegd aan de product backlog van het Scrum-project. Tijdens deze sessie kan er

eveneens besloten worden om de tijdsduur van een effort punt bij te stellen of om de prioritering van backlogitems aan te passen. Dit helpt mij vervolgens om volgende sprints nog acuter in te plannen.

De sprint retrospective wordt gebruikt om terug te kijken op de werkwijze en gebeurtenissen die tijdens de sprint zijn voorgekomen. De retrospective is bedoeld om gemaakte fouten of positieve punten te identificeren. Vervolgens zullen er manieren bedacht worden om de fouten te voorkomen en om positieve punten over te dragen naar een nieuwe sprint.

Beweegredenen

Ik heb gekozen voor een Scrum aanpak. Omdat ik mij wilde houden aan de werkwijze zoals deze gedicteerd wordt door Scrum, heb ik gekozen voor de wijze van communicatie zoals deze voorgeschreven wordt door Scrum. De drie momenten die ik heb uitgekozen voldoen aan alle behoeften die ik heb qua standaard communicatie; indien ik behoefte heb aan extra momenten zijn deze altijd te vinden bij één van de consultants.

De sprint planning en sprint review heb ik altijd uitgevoerd met de productowner. Hij heeft binnen het project gefungeerd als primaire bron van requirements en was daarom de beste persoon om deze, inhoudelijke, sessies mee uit te voeren.

Voor de sprint retrospectives heb ik gebruik gemaakt van de diensten van Jan Groeneveld. Jan is een projectmanager met 20+ jaar projectmanagement ervaring. Ik heb zijn hulp ingeschakeld om mij te assisteren met het bewaken van mijn projectvoortgang. Tijdens de wekelijkse gesprekken adviseert Jan mij over stappen die ik kan/moet nemen om mijn project op de rails te houden. Jan heeft verder geen directe invloed op de richting van het project, zijn rol is puur adviserend.

Alternatieven

Ik heb één communicatiemoment, dat wordt voorgeschreven door Scrum, weggelaten: de daily standup. Dit is een dagelijks communicatiemoment van maximaal 15 minuten met als doel het vluchtig aankaarten van de werkzaamheden en problemen van de aankomende dag. Hoewel ik dit graag had gewild is het eenvoudigweg niet mogelijk om de opdrachtgever op dagelijkse basis te spreken. Ik had dit mogelijk telefonisch kunnen doen maar, gezien de agenda van de opdrachtgever voor een gemiddelde week is het moeilijk om dit zelfs op die wijze structureel te doen.

H5 Vooronderzoek

Om mijzelf te oriënteren op de datamigratie heb ik deskresearch uitgevoerd naar het principe van datamigratie in relatie tot pixelzebra. Hierbij heb ik literatuuronderzoek uitgevoerd naar de vraag: *'Wanneer is een migratie succesvol voor pixelzebra?'*. Daarnaast heb ik interviews gehouden met een paar consultants van pixelzebra die ervaring hebben met datamigraties.

In dit hoofdstuk ga ik dieper in op de wijze waarop ik mijn literatuuronderzoek heb uitgevoerd. Dit doe ik middels een beschrijving van de vragen die ik gebruikt heb voor het onderzoek, de methoden die ik gebruikt heb om informatie te verzamelen en te verwerken en tot slot door een beschrijving van de resultaten van het literatuuronderzoek. Voor de interviews geef ik een beschrijving van de wijze waarop ik de interviews heb afgenomen en een opsomming van de belangrijkste informatie die ik uit de interviews heb kunnen halen.

5.1 Literatuuronderzoek

Het doel van mijn uitgevoerde literatuuronderzoek is: het schetsen van een beeld voor de perfecte migratie voor pixelzebra. De hoofdvraag is vrij breed opgezet. Dit heb ik bewust gedaan om geen aspecten van de migratie te kunnen vergeten. Om het onderzoek toch meer sturing te geven heb ik het onderzoek specifiek gemaakt door de hoofdvraag op te delen in een aantal deelvragen die ieder een deel van een migratietraject op zich nemen. Figuur 11 bevat een overzicht van alle opgestelde deelvragen plus motivatie.

Deelvraag	Verklaring
Welke fasering is het meest geschikt voor een datamigratie?	Verschillende organisaties hebben verschillende stappen voor het integreren of migreren van gegevens en voor het managen van dergelijke projecten. Om het migratieproces zo efficiënt mogelijk te maken wil ik graag weten welk van deze processen het meest geschikt is voor een datamigratie zoals pixelzebra deze uitvoert.
Welke tools zijn het meest geschikt om data naar CRM Online te migreren?	Met gebruik van de juiste tool denk ik dat het migratieproces voor pixelzebra zowel efficiënter als kwalitatief beter zal worden (Gartner, 2015).
Welke best practises zijn er op het gebied van datamigratie en hoe sluiten deze aan op de afstudeeropdracht?	Het is onnodig om het wiel opnieuw uit te vinden. Bedrijven die veel doen met data integratie hebben veelal bepaalde handelingen voor verschillende onderdelen van het proces. Door de effectiefste handelingen in te zetten voor pixelzebra hoop ik eveneens het migratieproces te verbeteren (IBM, 2007).
Welke valkuilen/issues komen frequent voor tijdens datamigraties en welke zijn er specifiek voor datamigratie naar CRM Online toe?	Om goed te kunnen anticiperen op problemen wil ik helder hebben welke valkuilen frequent voorkomen tijdens datamigraties. Door hier op in te spelen kan er tijdsverlies voorkomen worden (Morris, 2012).
Welke rol spelen security, performance en pricing bij het kiezen van doelomgeving?	Dynamics CRM is een cloud product. Veel bedrijven hebben angsten als het gaat om de Cloud en zien daarom af van een migratie naar Dynamics CRM Online. Omdat een migratie die nooit van de grond komt van te voren al is gefaald, heb ik dit onderdeel meegenomen in het onderzoek om tegen

	argumenten te vinden voor het niet willen verhuizen naar de Cloud (Oldenzeel, 2016).
Wat wordt er verstaan onder data?	Om de hoofd- en deelvragen beter te expliciteren wil ik mogelijk meerduidige termen expliciteren.
Wat is een migratie?	
Wat valt er onder tooling?	
Wat is een (bron/doel)omgeving?	

Figuur 11: Deelvragen voor het literatuuronderzoek met motivatie

Zoektermen verzamelen

Om zoektermen te verzamelen voor het literatuuronderzoek heb ik gekozen voor het maken van een mindmap. Ik heb gekozen voor een mindmap omdat dit een collaboratief middel is met een eenvoudige uitvoer (Veenings, 2007). Omdat pixelzebra een kleine hectische organisatie is (Kaay, 2015), is het nodig om een eenvoudig middel te gebruiken. Ik kan de medewerkers gewoonweg niet voor langere tijd samen krijgen.

Een andere optie voor het verzamelen van zoektermen waren de interviews geweest. Gezien de tijdsperiode waarin de interviews afgenomen zouden worden zou dit betekenen dat mijn literatuuronderzoek vertraagd zou worden met enkele weken. Dit zou mijn project teveel vertraagd hebben. Daarnaast is de overlap tussen de medewerkers die ik wil interviewen en de medewerkers die ik wil voor het opstellen van de mindmap niet 100%.

De bovenstaande opties gaan beide uit van de input van pixelzebra's medewerkers. Uiteraard had ik ook zelf op zoek kunnen gaan naar relevante zoektermen. Hiervoor zou ik echter het element 'pixelzebra' missen. Door de medewerkers te betrekken krijg ik zoektermen die beter slaat op mijn specifieke casus. Daarnaast levert een mindmap zoektermen op die ik niet in mijn eentje bedacht zou hebben.

Het resultaat van de mindmapsessie is een lijst met 41 zoektermen waarvan er 25 uniek zijn. Voor de weging ben ik uitgegaan van een eigen prioritering. Een zoektocht naar een geschikte prioriteringstechniek leverde namelijk niet de resultaten waar ik op had gehoopt. Uiteindelijk heb ik de zoektermen gewogen op drie factoren die direct aansluiten op de mindmap en naar mijn mening een goede weging geven aan de zoektermen. Deze factoren zijn als volgt:

- Relevantie (hoe direct gerelateerd is de zoekterm aan 'datamigratie');
- Frequentie (hoe vaak komt de zoekterm voor op de mindmap);
- Correlatie (aan hoeveel deelvragen is de zoekterm te relateren).

Bronnen verzamelen

Voor het vinden van informatie heb ik gebruik gemaakt van zes type informatiebronnen die mij tijdens eerdere onderzoeken voldoende informatie hebben opgeleverd. Naast deze type bronnen heb ik het privilege gehad om voor mijn afstuderen een live lezing van Microsoft bij te mogen wonen. Deze heb ik als zevende databron opgenomen. De volledige lijst met bronnen is als volgt:

- Google Scholar;
- Google;
- HBO Kennisbank;
- Sneeuwbalmethode binnen gevonden bronnen;
- Blogs;
- Boeken;
- Live lezingen.

Om het zoeken van informatie te scopen heb ik de doelstelling gezet op 100 bronnen. Mijn ervaring leert mij dat dit meer dan genoeg relevante bronnen voor onderzoek van dit formaat oplevert. Voor het verzamelen van bronnen heb ik maximaal acht uur (één volle werkdag) gereserveerd. Langer zoeken zal te veel tijd wegnemen van de overige projectactiviteiten en de kwaliteit van de bronnen zal er niet beter op worden.

Kwaliteit bepalen

De laatste stap die ik heb uitgevoerd voor het verzamelen van informatie is het bepalen van de bruikbaarheid van de gevonden informatie. Zonder de kwaliteit te bepalen zou ik mogelijk een proces inrichten dat slecht van kwaliteit is of niet aansluit op pixelzebra. Voor het bepalen van de kwaliteit heb gebruik gemaakt van een checklist die gebaseerd is op de criteria die de University of California voorstelt voor het bepalen van de kwaliteit van bronnen (UCSC, 2016). In tegenstelling tot andere checklists die ik ben tegengekomen bestaat deze checklist uit een set lijsten die specifiek zijn opgesteld voor verschillende type databronnen. Ik vind het daarom een effectief middel voor het bepalen van bronkwaliteit in een onderzoek als mijn literatuuronderzoek waar de type databronnen divers zijn.

De bovengenoemde checklist legt voornamelijk de focus op de metadata van een bron. Dit is een goede graadmeter voor het bepalen van de algemene bronkwaliteit (Fischer & Hulsing, 2014). Ik heb de checklists daarom gebruikt om een filtering te maken tussen bruikbare en onbruikbare bronnen. Deze filtering heeft 56 van de 105 gevonden bronnen af laten vallen. Voor mijn gevoel had ik daarna nog steeds te veel bronnen over om binnen mijn geschatte tijd het onderzoek uit te voeren.

De tweede stap die ik heb ondernomen voor het kwalificeren van de bronnen is het lezen van de inleidingen/abstracts van de bronnen. Dit heeft mijn relatief weinig tijd gekost terwijl het mij veel inzicht heeft gegeven over de informatie die in de bronnen te vinden is. Door middel van deze kwalificering heb ik weer 20 bronnen kunnen filteren.

Als laatste filtering, om echt de primaire informatiebronnen voor mijn onderzoek te kunnen identificeren, heb ik de overgebleven 29 bronnen globaal door gelezen. De bronnen die over zijn gebleven waren al van goede kwaliteit maar, door deze van te voren door te lezen heb een aantal bronnen kunnen aanstippen die als basis voor mijn onderzoek zullen dienen. Figuur 12 toont een overzicht van de bronverdeling nadat ik deze gefilterd had, figuur 13 toont de controlevragen waarop ik de filtering uitgevoerd heb.

Waardering	Aantal bronnen
Onbruikbaar	56
Bruikbaar	20
Goed	17
Onmisbaar	12

Figuur 13: Resultaten bronfiltering

Controle	Ja	Nee
Voldoet de metadata aan de richtlijnen van de University of California?	Bruikbaar	Onbruikbaar
Sluit de abstract aan op het onderzoek?	Goed	Bruikbaar
Sluit de inhoud van de bron aan goed aan op het onderzoek?	Onmisbaar	Goed

Figuur 12: Gebruikte vragen voor het filteren van bronnen

Resultaten

Middels de bronnen die ik verzameld heb gedurende het literatuuronderzoek heb ik getracht om antwoord te geven om de (deel)vragen van het onderzoek. Hieronder geef ik per vraag een beknopte beschrijving van de antwoorden die ik gevonden heb op de vragen. De volledige lijst met resultaten is te vinden in bijlage 3.

Welke fasering is het meest geschikt voor een datamigratie?

Om dit te vast te stellen heb ik een aantal White papers gebruikt van grote Tech giganten zoals IBM en Oracle. Mijn redenering was dat de ervaring van deze bedrijven een goede basis vormt voor een eigen proces. De informatie uit de White papers heb ik aangevuld met een aantal artikelen waarin best practices random datamigraties beschreven stonden. Ondanks de verschillen in de details (benaming van fasen, generieke handelingen binnen een fase, wijze van iteratie) kwamen er continu vijf fasen naar voren:

- Start,
- Voorbereiden (ontwerpen, modelleren, informatie verzamelen),
- Uitvoeren (ontwikkelen / ETL),
- Nazorg (testen, decharge),
- Einde.

De start en eind fasen bevatten geen handelingen en zijn alleen ter formaliteit te gedefinieerd door de grotere organisaties. In sommige gevallen worden deze gebruikt voor de kick-off en decharge van een project. Ik heb echter gekozen om deze handelingen op te nemen in de overgebleven drie fasen. Wat overblijft zijn de voorbereidingsfase, uitvoeringsfase en nazorgfase. De fasen zijn herkenbaar en zijn bewezen van toegevoegde waarde voor een datamigratie. Daarnaast sluiten de stappen goed aan op de algemene werkwijze die pixelzebra momenteel hanteert voor overige projecten.

Welke tools zijn het meest geschikt om data naar CRM Online te migreren?

Tijdens mijn onderzoek ben ik gestuit op een vergelijkingswebsite voor datamigratietools. Deze vergelijkingsstool heb ik gebruikt als basis voor de uiteindelijke drie tools die ik in mijn onderzoek heb aangeraden. De website vergelijkt de tools op een aantal criteria zoals gebruiksgemak, eenvoud van toepasbaarheid en onderhoudbaarheid, etc.

De eerste tool die ik op basis van de vergelijkingswebsite heb geselecteerd is Informatica integration. Dit is de hoogst scorende migratietool op de website. Op vrijwel alle criteria scoort de tool hoog en verder onderzoek naar de tool wijst uit dat Gartner Informatica integration benoemd heeft tot best integratietool van 2015. Deze argumenten zijn naar mijn mening ruimvoldende om de tool als optie voor pixelzebra op te nemen.

De tweede optie die ik geselecteerd heb als potentiële migratietool is Scribe Online. Op de vergelijkingswebsite scoort deze tool net zoals Informatica integration hoog op vrijwel alle criteria, al is dit niet zo hoog als de voorgenoemde tool. Verder onderzoek wijst uit dat Scribe Online echter een aantal specifieke voordelen voor Dynamics CRM heeft ten opzichte van Informatica integration. Scribe Online bevat namelijk een standaard connector naar Dynamics CRM toe, iets wat Informatica niet heeft. Deze connector waarborgt de data integriteit tijdens het ETL-proces. Deze eigenschap brengt de tool naar mijn mening op gelijk niveau met Informatica integration.

De laatste optie die ik heb geselecteerd komt niet voort uit de vergelijkingswebsite. Meerdere bronnen die ik heb gelezen bevestigen dat maatwerk veelal een snelheidsvoordeel heeft ten opzichte van standaard

migratietools. Voor mijn project, waarin tijdswinst voorop staat, is dit een uitstekend criterium om de optie aan te raden voor het gebruik als standaardtool voor pixelzebra. De organisatie heeft genoeg kennis in huis om een dergelijke tool te ontwikkelen. Het nadeel van maatwerk, ten opzichte van standaardpakketten, ligt hem in de ontwikkeltijd en de onontkoombare aanwezigheid van bugs. De initiële kosten zullen daardoor hoger liggen dan standaard tooling.

Ik heb de bovenstaande argumenten voorgelegd aan de productowner. Op basis van de argumenten heeft hij een keuze gemaakt voor Scribe Online. Hiervoor was de standaard connector van Scribe Online, tezamen met de laagste kosten van de drie opties, doorslaggevend.

Welke best practises zijn er op het gebied van datamigratie en hoe sluiten deze aan op de afstudeeropdracht?

Voor het antwoord op deze vraag heb ik een verzameling gemaakt van alle best practises die ik ben tegengekomen tijdens mijn literatuuronderzoek die aansluiten op de casus van pixelzebra. De best practises waarvoor ik heb gekozen zijn algemene werkwijzen die een bijdrage kunnen leveren aan het behalen van tijdswinst binnen mijn migratie project. In figuur 14 is een opsomming weergegeven van de best practises die ik heb opgenomen in de resultaten van mijn literatuuronderzoek. Een uitgebreide toelichting op de best practises is te vinden in bijlage 3.

Best practise	motivatie
Big bang versus Trickle	Dit zijn de twee standaard methoden voor datamigratie. De keuze qua methode bepaald voor een groot deel de kwaliteit en snelheid van het migratieproces.
Automatisering	Door delen van het migratieproces te automatiseren kunnen ongekend grote tijdswinsten behaald worden.
Stakeholders	Pixelzebra's migratieprojecten vallen of staan met het correct omgaan van stakeholders.
Top-down migreren	Door de juiste volgordelijkheid aan te brengen in de migratie kan er veel tijdswinst behaald worden en is bevorderlijk voor de datakwaliteit.
Kosten/baten	Pixelzebra moet zich realiseren dat goed en snel migreren geld kost. Deze best practise kan helpen om de gevolgen van de eisen van de opdrachtgever helder te maken.
Testen	Zonder een testproces is het niet mogelijk om de datakwaliteit na afloop van de migratie te garanderen. Pixelzebra wil haar klanten kunnen garanderen dat de data integriteit overeind blijft tijdens een migratie, een testproces is daarom onmisbaar.

Figuur 14: Lijst met best practises voor de migratie

Welke valkuilen/issues komen frequent voor tijdens datamigraties en welke zijn er specifiek voor datamigratie naar CRM Online toe?

Naast de best practises voor het migreren heb ik ook gekeken naar handelingen die absoluut niet uitgevoerd moeten worden. Mijn redenatie hiervoor was dat dit tijd en geld zal kosten voor pixelzebra. Zoals bij de best practises heb ik voor deze vraag alle valkuilen opgenomen die ik tijdens mijn onderzoek ben tegengekomen die aansluiten op de casus van pixelzebra. Figuur 15 bevat een opsomming van de gevonden valkuilen, in bijlage 3 is een toelichting op de valkuilen te vinden.

Valkuil/issue	Motivatie
---------------	-----------



Boom, klap, klaar	De keuze voor pixelzebra om migraties uit te voeren volgens het 'Bing Bang' principe. De tijdswinst is hierbij namelijk het grootst. Big Bang migraties zijn echter niet zonder ernstige risico's die grote vertraging kunnen opleveren.
Eenrichtingsverkeer	De opzet van de opdrachtgever is dat gegevens slechts één kant uitgestuurd zullen worden. Niet vreemd met een migratie. Mijn literatuur wijst echter uit dat dit in veel gevallen niet waar blijkt te zijn. Veel klanten willen systemen tijdelijk parallel laten lopen waardoor het niet rekening houden met tweerichtingsverkeer funest kan zijn voor de nazorg van een migratie.
Geen commitment vanuit organisatie	Pixelzebra werkt graag intensief samen met de klant. Dit betekent dat zij veel vertrouwen legt in de verantwoordelijkheden van de klanten en de uitvoering ervan. In de realiteit wil het nog wel eens gebeuren dat stakeholders binnen een organisatie niet willen meewerken met een project. Pixelzebra's vertrouwen kan hierdoor een project in gevaar brengen.
Data cleaning	Data integriteit bepalen na afloop van een migratie is belangrijk. Minstens even belangrijk is het bepalen van de datakwaliteit voorafgaand aan een migratie. Dit bijna vanzelfsprekende feit wordt bijzonder vaak vergeten of genegeerd in datamigratietrajecten.

Figuur 15: Lijst met valkuilen van migratie

Welke rol spelen security, performance en pricing bij het kiezen van doelomgeving?

Tijdens de lezing die ik heb bijgewoond bij Microsoft werd er uitgebreid gesproken over angsten die potentiële overstappers naar Dynamics CRM Online hebben. Hierbij kwamen drie punten aan bod die door vrijwel iedere klanten worden aangehaald. Het gaat hierbij om de licentiekosten, de performance van een Cloud product en veiligheid van de opgeslagen data. Omdat dit allemaal angsten zijn waardoor een klant kan kiezen om niet te migreren naar Dynamics CRM Online heb ik deze vraag opgenomen. Een migratie faalt namelijk bij voorbaat als deze nooit wordt uitgevoerd.

Samengevat heb ik tijdens mijn onderzoek gezocht naar argumenten die de angsten van een potentiële klant om te migreren naar Dynamics CRM Online moeten wegnemen. Microsoft is zich bewust van deze angsten en heeft zelf documenten opgesteld om consultants bij te staan tijdens discussies waar deze angsten aan bod komen. De lijst met argumenten voor het migreren naar de Dynamics CRM Online zijn te vinden in bijlage 3.

Wanneer is een migratie succesvol voor pixelzebra?

Op basis van mijn onderzoek heb ik geconcludeerd dat deze vraag niet letterlijk beantwoord moet worden. Het eenvoudige antwoord zou anders zijn dat het verkrijgen van decharge het succesvolle einde van de migratie inluidt. Het antwoord op de vraag is daarom niet alleen het criterium voor het afsluiten van een migratieproject maar, eveneens de stappen die nodig zijn om dat succes te bereiken.

Door de antwoorden van de deelvragen samen te voegen ontstaat een korte beschrijving die naar mijn mening goed antwoord geeft op de hoofdvraag. Hieronder geef ik een beknopte samenvatting van het antwoorde op de hoofdvraag. Een migratie voor pixelzebra is succesvol als zij werkt via een gestructureerde proces verdeeld over de fasen: voorbereiding, uitvoering en nazorg. De migratie kan het best uitgevoerd worden met ondersteuning van een migratietool. Tijdens het doorlopen van het migratieproces moet er rekening gehouden worden met een reeks best practices en valkuilen. Wat pixelzebra niet moet vergeten is dat klanten veelal een angst hebben om naar de Cloud te gaan. Dit moet voorafgaand aan het project goed gemanaged worden. Dit zijn de minimale criteria voor pixelzebra om succesvol te migreren.

5.2 Interviews

De interviews hebben een gelijke insteek als het literatuuronderzoek uit hoofdstuk 5.1. De variatie zit hem in de scope van het onderzoek. Daar waar het literatuuronderzoek de focus legt op feitelijke informatie vanuit de hele wereld leggen de interviews de nadruk op de ervaring en meningen van pixelzebra's medewerkers, de interviews zijn daardoor potentieel emotioneler van aard. In bijlage 2 is een template te vinden van het interviewboekje dat ik gebruik heb ter ondersteuning van het onderzoek.

Ik heb gekozen voor een interviewboekje omdat ik dit een prettige manier van werken vind. Het helpt bij het opzetten van een ongestructureerd interview en dient eveneens als rode draad en geheugensteun tijdens het interview (Cohen & Crabtree, 2006). Daarnaast heb ik goede ervaringen bij pixelzebra met het gebruik van een interviewboekje tijdens een eerder onderzoek.

Interviewstructuur

Middels het interviewboekje heb ik het interview zo opgesteld dat de geïnterviewde niet alleen veel aan het woord is maar, ook met zijn of haar handen aan de slag gaat. Ik ben de interviews gestart met een algemene vraag naar ervaring rondom datamigraties. Door de vraag algemeen te stellen geef ik de geïnterviewde zelf de kans om de richting van het interview te bepalen, ik hoef alleen in te grijpen als ik het gevoel heb dat het gesprek te veel gaat afwijken.

Nadat de geïnterviewden hun verhaal heeft kunnen doen heb ik hem een aantal dingen laten tekenen. Ik heb hem het migratieproces voorgelegd zoals ik uit het literatuuronderzoek geconcludeerd had. Ik heb de geïnterviewden gevraagd hoe zij hier tegenover staan. Vervolgens heb ik hen het procesmodel laten uitbreiden zoals zij denken dat ideaal is. Ik heb hen vervolgens het proces laten tekenen voor de voorbereidingsfase. Dit is de kritiekste fase van mijn migratieproces en ik wilde graag weten hoe de medewerkers dit nu zouden aanpakken. Omdat pixelzebra geen standaard ingericht migratieproces heeft, hebben de tekenen zijn een goed beeld gegeven van hoe pixelzebra op individuele basis een migratie zou aanpakken.

De laatste vragen van het interview gaan over het identificeren van stakeholders, het herkennen van valkuilen en overige tip en tricks. Deze antwoorden heb ik vervolgens naast de resultaten van het literatuuronderzoek kunnen leggen om zo de kwaliteit van het draaiboek voor pixelzebra te verbeteren.

De resultaten van de interviews zijn te vinden in bijlage 3.



H6 Het voorbereiden op de migratie

Tijdens het uitvoeren van mijn literatuuronderzoek heb ik gekeken naar een geschikt migratieproces dat aansluit op de behoeften van pixelzebra aan een kwalitatief goed migratieproces met een zo kort mogelijke doorlooptijd. Mijn keuze is uiteindelijk gevallen op een eenvoudig driestappenproces met de stappen: voorbereiding, uitvoering en nazorg. In dit hoofdstuk ga ik de eerste stap van dit migratieproces behandelen.

Zoals met ieder softwareproject worden fouten bij voorkeur aan de start van een project opgemerkt. Problemen die klein zijn aan de start kunnen in een latere fase catastrofale vormen aannemen (Steyn, 2002), hetgeen op zijn beurt de doorlooptijd van een datamigratie zal vertragen. Om dit tegen te gaan heb ik voor mijn migratieproces gekozen om extra aandacht te besteden aan de voorbereidingsfase. De consultants van pixelzebra zijn altijd druk en af en toe eigenwijs (Kaay, 2015). Hierdoor willen zij wel eens iets te pragmatisch omgaan met projectaspecten zoals scoping en het vastleggen van requirements. Door hen te 'forceren' om meer tijd te besteden aan de voorbereiding van de migratie hoop ik dat de algemene valkuilen, benoemd in hoofdstuk zes, hen bespaard blijven.

'Voorbereiding' is tamelijk algemeen. Het geeft naar mijn mening geen duidelijk beeld over welke handelingen verwacht worden van de consultant in deze fase, wanneer de voorbereiding (en daarmee de migratie zelf) start en wanneer de voorbereiding is afgerond. Om de pixelzebra de benodigde handvatten te geven voor het correct uitvoeren van de voorbereidingsfase heb ik de fase opgedeeld in sub-fasen zodat er correct kan geïdentificeerd kan worden in welk onderdeel van de migratie van toepassing is en welke handelingen daarbij gewenst zijn. Het gaat hierbij om de sub-fasen:

- Pre-voorbereiding,
- Scoping,
- Requirements,
- Inrichting.

Pre-voorbereiding

Scoping

Requirements

Inrichting



- ⇒ Sales stakeholder-
- ⇒ analyse
- ⇒ Offerte

6.1 Pre-voorbereiding

Om een migratie uit te kunnen voeren moet het project eerst binnengehaald worden. Dit is officieel geen onderdeel van het migratieproject maar, een project dat niet binnengehaald wordt is een project dat nooit uitgevoerd zal worden. Ik vind het daarom belangrijk om een stukje van het salesproces, dat vooraf gaat aan het migratieproces, mee te nemen. Een tweede reden voor het benoemen van de pre-migratie is het feit dat pixelzebra's consultants veel invloed op het salesproces (Kaay, 2015). Tijdens de interviews die ik met leden van pixelzebra heb gehouden, gecombineerd met mijn literatuuronderzoek, heb ik een drietal handelingen geïdentificeerd binnen het salesproces waarin de consultant een kritieke rol speelt. Deze handelingen genomen heb ik benoemd tot de 'pre-voorbereiding', oftewel de handelingen die de consultant uit dient te voeren voorafgaand aan het migratieproject.

Sales

Pixelzebra werft veel van haar klanten middels campagnes, zowel zelf opgestart als uitgevoerd via Microsoft. Vanessa Lagas, hoofd van de Sales afdeling, heeft hierin de leidende rol. Na initieel contact met een potentiële klant wordt er door pixelzebra een gratis CRM-scan aangeboden. Tijdens deze scan, uitgevoerd op locatie bij de klant of via een Webinar, worden de algemene wensen en eisen van de klanten geïnterviewd en wordt hiervoor een advies uitgebracht. Dit is tevens het moment binnen het salestraject waarin de consultant een rol gaat spelen, het advies komt namelijk altijd uit de hand van een consultant bij pixelzebra. Het is belangrijk dat de consultant rekening houdt met de angsten van klanten rondom het overgaan naar Dynamics CRM Online. Als deze niet goed gemanaged worden kan het gehele project afgeblazen worden.

Tijdens mijn afstuderen heb ik een workshop op locatie bij Microsoft bijgewoond van de heer Steven van Vlaanderen (2016), een goeroe op het gebied van Dynamics CRM. Hij heeft gedurende zijn carrière drie vragen geïdentificeerd die tijdens vrijwel iedere overgang naar Dynamics CRM in de Cloud naar voren komen. De wijze waarop de consultant antwoord geeft op deze drie vragen is kritiek voor het slagen van een salesproces, met name voor migraties vanuit een on premise omgeving. De vragen hebben betrekking op:

- Privacy (hoe veilig staat mijn data);
- Performance (hoe goed performed de Online versie van Dynamics CRM ten opzichte van de on premise versie);
- Kosten (zijn de maandelijkse kosten niet veel hoger dan de eenmalige kosten voor hardware en implementatie voor on premise).

In mij draaiboek bied ik pixelzebra een aantal argumenten om deze angsten weg te nemen bij de klant. Het draaiboek met de antwoorden is te vinden in bijlage 4.

Stakeholderanalyse

De tweede taak van een consultant tijdens het sales proces is het identificeren van de juiste stakeholders tijdens een CRM-scan. De verantwoordelijke vanuit Sales zal meestal een afspraak voor een scan hebben opgezet. De salesmedewerker zal gezorgd hebben dat de scan de nodige stakeholders bevat die nodig zijn voor het goedkeuren van een project of die op zijn minst invloed uit kunnen oefenen op zij die dat wel kunnen. Ik heb dit onderdeel meegenomen in mijn draaiboek omdat ik vind dat de consultant hier wederom een grote rol speelt. Wat pixelzebra's salesmedewerkers niet kunnen tijdens een CRM-scan is bepalen welke stakeholders inhoudelijk een rol in het project moeten gaan spelen. De consultant moet op dit punt al bepalen welke stakeholders kritiek zullen zijn tijdens het migratieproject. Als de consultant dit correct doet kan hij/zij bij voorbaat een band opbouwen met de stakeholders om de mate van kennis en commitment te identificeren en te beïnvloeden. Deze manier van werken sluit aan op pixelzebra's filosofie over het opbouwen langdurige relaties met de klant (Kaay, 2015).

Offerte

De laatste stap van pixelzebra's salesproces is het leveren van een offerte aan de klant. Dit onderdeel kwam tijdens mijn interview met Debbie Rhebok, senior consultant bij pixelzebra's nadrukkelijk aan bod. Ik heb dit onderdeel van het salesproces geselecteerd om twee redenen: de consultant speelt een grote rol in dit onderdeel en de verdiensten van pixelzebra op het project worden tijdens dit onderdeel bepaald. Met name het tweede onderdeel maakt het zo belangrijk om het te benadrukken in mijn draaiboek, uiteindelijk is het verdienen van geld pixelzebra's uiteindelijke doel voor een CRM-migratie. Ondanks het belang noem ik geen standaard werkwijze voor het opstellen van een offerte. Ik benoem het wel om de consultants te herinneren dat er een duidelijke relatie aanwezig is tussen de verdiensten van pixelzebra en de scope van een CRM-migratie. De offerte dan ook het document waarin deze twee geconsolideerd worden.

Mijn migratieproces eist een vrijwel 'perfecte' scope (zie H6.2 voor toelichting). Dit vereist op zijn beurt een grote tijdsinvestering van zowel pixelzebra als de klant. Pixelzebra's standaard werkwijze is om de scope van een project voorafgaand aan de start in de offerte vast te leggen. Dit gebeurt op kosten van pixelzebra zelf omdat de CRM-scan en opstellen van de offerte volgens pixelzebra's beleid een investering is op de relatie met de klant. In mijn proces moet pixelzebra goed nadenken over de wijze waarop de scope wordt beschreven in de offerte. Er moet ruimte blijven om scope bij te stellen indien dit nodig blijkt te zijn. De scope heeft namelijk directe invloed op de doorlooptijd van het migratieproces en daarmee op het succes ervan.



Pre-voorbereiding

Scoping

Requirements

Inrichting



- ⇒ Kennis
- ⇒ Commitment
- ⇒ Flexibiliteit
- ⇒ Prioriteit
- ⇒ Decharge

6.2 Scoping

Uit eigen ervaring (ik ben inmiddels bijna twee jaar actief binnen pixelzebra) weet ik dat pixelzebra's consultants niet zelden buiten de scope van hun project gaan. Omdat mijn migratieproces gaat over het behalen van de hoogst mogelijke tijdwinst is het vastleggen van de scope en het niet afwijken ervan kritiek (Nielsen, 2010). Om de consultants beter in het gareel te houden tijdens een migratieproject heb ik daarom het scopen van de migratie opgenomen als stap binnen de voorbereidingsfase. Tijdens mijn literatuuronderzoek heb ik vervolgens gezocht naar parameters voor de 'perfecte' scope. Uiteindelijk heb ik vijf parameters kunnen identificeren die aansluiten op pixelzebra's werkwijze. In dit hoofdstuk leg ik per parameter uit welke toegevoegde waarde de parameter heeft voor mijn proces en wat de gevolgen zijn als pixelzebra geen rekening houdt met de parameter.

Kennis

Pixelzebra is een organisatie die graag klantgericht werkt (Kaay, 2015). Zij vertrouwt op de kennis van kunde van haar klant om zo samen een project uit te voeren. Om deze reden heb ik kennis geselecteerd als één van de parameters voor het scopen tijdens mijn migratieproces. Scoping gaat namelijk niet alleen over het identificeren van het uit te voeren werk, het gaat net zo zeer over het identificeren van de stakeholders (kennis) die nodig zijn om dit werk uit te voeren (Moss, 2000). De samenwerkende kracht van pixelzebra is daarmee echter net zo zeer een zwakte als een kracht tijdens mijn migratieproces.

Mijn draaiboek, dat de rode draad is voor het uitvoeren van mijn migratieproces, heeft voor pixelzebra meer een bewakende dan assiterende functie als het gaat om het benaderen van stakeholders. Omdat de consultants vaak zoveel aandacht besteden aan het opbouwen van relaties met verschillende stakeholders willen zij hier nog wel eens in doorschieten. Door in mijn draaiboek te benoemen welke kennis minimaal nodig is voor een succesvolle migratie is het de bedoeling dat pixelzebra aandacht gaat besteden aan de stakeholders die corresponderen met de rollen die ik heb benoemd in mijn draaiboek. Zo blijven zij binnen de scope van het project en vermindert de doorlooptijd van het migratieproces. Verzuim om dit te doen, door te veel tijd te besteden aan het opbouwen van banden met stakeholders, zal onherroepelijk leiden tot tijd- en kwaliteitsverlies (Oracle, 2011).

Commitment

Commitment is een parameter die onlosmakelijk samenhangt met kennis. Daar waar de kennis parameter bedoeld is om stakeholders te identificeren die nodig zijn voor het migratieproject, is de commitment parameter nodig om te bepalen of de geïdentificeerde stakeholders hun kennis en tijd ook daadwerkelijk willen delen (Russom, 2006). Zoals bij kennis sluit dit aan op pixelzebra's samenwerkingsfilosofie. Indien pixelzebra pas in een later stadium ondervindt dat bepaalde stakeholders niet mee willen werken met de migratie kan het gebeuren dat het project vertraging oploopt. In extremere gevallen kan gebrek aan

commitment zelfs leiden tot gebrekkige data integriteit of zelfs het falen van de migratie. Een voorbeeld hiervan is key user die slechte informatie geeft over de interfaces van de doelomgeving. Dit zal leiden tot klachten van gebruikers waardoor veel werk opnieuw uitgevoerd moet worden om hen tevreden te stellen.

Flexibiliteit

Pixelzebra prefereert haar projecten uit te voeren middels Scrum, een Agile aanpak (Solingen & Rustenburg, 2010). De wijze waarop ik mijn migratieproces heb ingericht heeft echter een lineair karakter. Om te zorgen dat pixelzebra geen problemen ondervindt van dit feit heb ik flexibiliteit opgenomen als parameter voor het bepalen van de scope.

Met flexibiliteit doel ik om de mate waarin pixelzebra kan reageren op problemen tijdens de migratie. Het migratieproces voelt zich het meeste thuis in een lineaire aanpak en heeft binnen deze werkwijze dan ook een aantal voordelen die ik bewust geselecteerd heb. Zo biedt een lineaire aanpak een hoge mate van structuur waardoor ik eenvoudiger een volgordeelijkheid in mijn proces kan aanbrengen. Daarentegen levert het gebruik van de Agile aanpak een aantal voordelen die ik van te voren niet voorzien had. Door de keuze bij de consultant neer te leggen geef ik hen de keuze om het proces 100% volgens mijn bedoeling uit te voeren of om het proces uit te voeren volgens hun eigen voorkeurswijze. In de onderstaande tabel is een overzicht te vinden van de keuzecriteria die de consultant kan gebruiken voor het selecteren van een projectaanpak.

Criterium	Agile	Lineair
Uitvoering	Het project wordt opgedeeld in sprints Dit kan door het project op te delen in mini-lineaire projecten of door per migratieprojectfase een aantal sprints in te plannen.	Het project wordt uitgevoerd als standaard lineair project, opgedeeld in de drie fasen van mijn migratieproject.
Contactmomenten	Vinden plaats aan de start en aan het einde van iedere sprint conform de Scrumregels. Dit is vrij tijdsintensief (Sutherland, 2001), mogelijk negatief voor de scope van het project (Adelman & Bischoff, 2002)	Vinden uitsluitend plaats tijdens de voorbereidingsfase en deels tijdens de nazorgfase (decharge). Minder tijdsintensief dan Agile (Royce, 1970).
Foutopsporing	Afhankelijk van de wijze waarop de sprints zijn ingericht kunnen problemen potentieel later of eerder gevonden worden dan in een lineair project.	Fouten zullen voornamelijk aan de start van het project opgespoord worden. Dit is de fase waarin er het actiefst gecommuniceerd wordt met de klant (Royce, 1970).
Fout impact	De impact van fouten zullen met name in latere stadia van het migratieproject minder zwaar wegen dan in een lineaire aanpak. De keerzijde is echter een zwaardere tijdsinvestering in de communicatie (Solingen & Rustenburg, 2010).	Fouten die na de eerste fase van het migratieproject gevonden worden hebben een zware impact. Hoe later de fout gevonden wordt, hoe exponentieel hoger de impact is (Moss, 2000). De tijdsinvestering is daarentegen lager dan die van de Agile aanpak.

Figuur 16: Agile vs. Lineair tijdens het migratieproces

Prioriteit

Pixelzebra is de vreemde eend in de bijt binnen de CRM-branche. Zij garandeert klanten dat alles mogelijk is, tegen een schappelijke prijs en binnen een strak ingeplande doorlooptijd. In de realiteit heeft pixelzebra met name op het laatste punt nog wel eens moeite om zich aan haar beloften te houden.

Dat het bedrijf niet alle aspecten op het hoogste niveau kan behalen is niet vreemd. De 'ijzeren driehoek' van projectmanagement zegt dat slechts twee van de drie aspecten: scope, kosten en planning, die samen de kwaliteit van een project bepalen, behaald kunnen worden (Lewis, 2005). In pixelzebra's geval wordt met name de doorlooptijd (planning) onderschat. Dit is onacceptabel binnen mijn migratieproces waarin tijdswinst de grootste factor voor succes is. Om de pixelzebra hier bewust van te maken heb ik daarom gekozen om prioriteit als parameter voor het scopen van het migratieproject op te nemen.

Om mijn beoogde tijdswinst te behalen is een goed afgestelde scope kritiek. De standaard ijzeren driehoek legt echter de nadruk op de kwaliteit van het project. Ik ben daarom op zoek gegaan naar soortgelijk model dat beter aansluit aan mijn behoefte om het belang van een goede scope aan te tonen. Ik ben hierbij gestuit op het SRT-model, een variatie op de ijzeren driehoek waarbij de criteria: snelheid, kwaliteit en prijs samen de scope van een project bepalen. het SRT-model geeft pixelzebra een handvat om aan te tonen wat de gevolgen zijn van het verkeerd plannen van een scope.

Een vluchtige blik op afbeelding 17 toont de twee belangrijkste criteria voor mijn project: kwaliteit (good) en snelheid (fast). Dit betekent dat bij zowel pixelzebra als haar klanten de illusie van een goedkoop project weggenomen moet worden, willen zij een migratieproject realiseren dat aan die twee criteria voldoet. Als één van beide partijen toch besluit om hier van af te wijken zal of de snelheid of de kwaliteit van het migratieproject dalen. Beiden zijn bewezen redenen waarop veel CRM-migraties falen (Missi, Alshawi, & Fitzgerald, 2005) (Lewis, 2005).

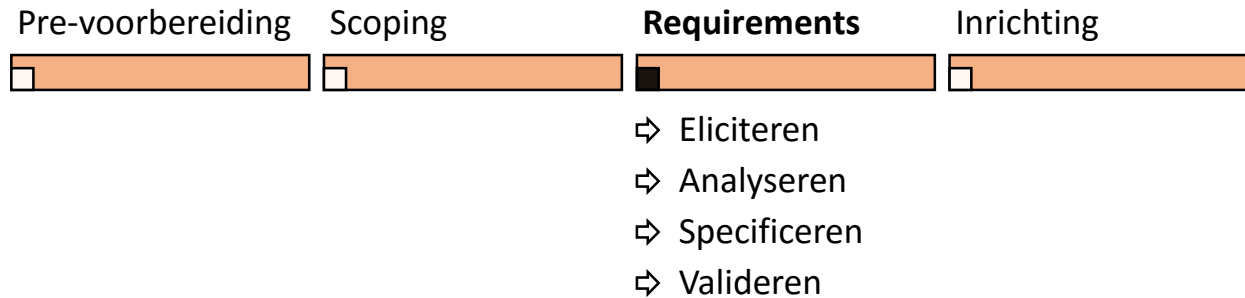


Figuur 17: De ijzeren driehoek van projectmanagement op scoping

Decharge

Ik heb uit eerste hand mogen ervaren hoe moeilijk het kan zijn om een decharge te verkrijgen aan het einde van een CRM-project. Zonder een scope waarin expliciet de voorwaarden voor het afsluiten van een project zijn opgenomen, kan het voor pixelzebra lang duren voordat een klant officieel decharge verleend. De klant lijkt op dit moment in een (migratie)project nog even het onderste uit te kan te willen halen. Dit gedrag is funest voor mijn migratieproces en daarom heb ik decharge opgenomen als laatste parameter voor een perfecte scope.

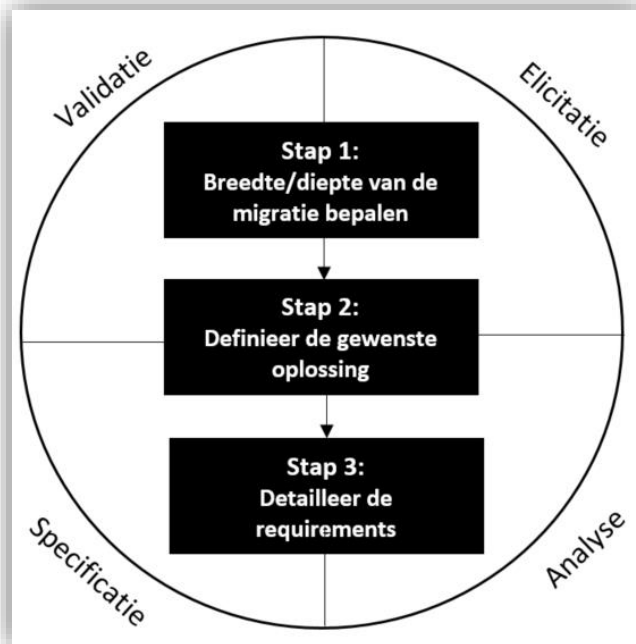
Zoals met de voorgaande parameters heb ik decharge voornamelijk geselecteerd om pixelzebra bewust te maken van het belang van een goede scope. In het draaiboek heb ik daarom opgenomen dat zij aan de start van het project de voorwaarden voor het verkrijgen van decharge zo expliciet mogelijk moeten vastleggen en door de klant moeten laten ondertekenen. Dit voorkomt een reeks aan problemen die de kwaliteit en doorlooptijd van een migratieproject kunnen beïnvloeden (Nielsen, 2010).



6.3 Requirements

Pixelzebra's 'karakter' leunt op samenwerking met de klant (Kaay, 2015). Dit geeft de organisatie een krachtig voordeel tijdens het vastleggen van requirements doordat de samenwerking een positieve uitwerking heeft op de communicatie tussen pixelzebra en haar klanten. Een voorbeeld de sterke samenwerking is de CRM-scan die pixelzebra aanbiedt aan haar klanten. De scan wordt uitgevoerd door

een Consultant en een Salesmedewerker op locatie bij de klant. Via deze scan worden de behoeften van een klant in kaart gebracht en verkoopt pixelzebra tegelijkertijd haar product. Het samenspel van de twee medewerkers en de nauwe betrekking van klant in deze sessie heeft tot op heden uiterst positieve resultaten opgeleverd voor pixelzebra als het gaat om het achterhalen van eisen en wensen.



Figuur 18: Het requirementsproces van Abran & Moore (2004)

Om tijdswinst te behalen op het vastleggen van de requirements moet pixelzebra een stap verder durven gaan dan alleen een CRM-scan. Om mijn migratieproces succesvol af te ronden is het noodzakelijk om de requirements tot in het kleinste detail vast te leggen. Dit betekent dat er een meer gestructureerd proces nodig is om de benodigde requirementskwaliteit te behalen.

De wijze waarop pixelzebra werkt heeft als gevolg dat projectelementen zoals het vastleggen van requirements nogal eens een ietwat losbandig proces is dat met name intern betere communicatie behoeft (Kaay, 2015). Omdat pixelzebra snel in de rol van CRM-expert kruipt kan zij eenvoudig requirements opleggen aan de klant. De organisatie doet dit ook gretig en klanten waarderen dit in veel gevallen ook. Voor mijn migratieproces is deze aanpak niet voldoende, requirements omtrent het behouden van data integriteit kunnen bijvoorbeeld niet zomaar opgelegd worden zonder de integriteit direct te beïnvloeden. Om pixelzebra bewust om te laten gaan met haar eigen gedrag gaat mijn voorkeur uit naar een requirementsproces dat direct tegen haar normale werkwijze indruist.

Voor het selecteren van een requirementsproces heb ik beroep gedaan op mijn eigen ervaring rondom het vastleggen van requirements. Ik heb zelf goede ervaringen met het requirement development proces

van Abran en Moore (2004) en ik geloof volhartig dat dit 100% tegen pixelzebra's manier van werken indruist. Mits correct gevolgd denk ik wel dat dit proces de kwaliteit van de requirements ten goede zal komen. Daarnaast is een dergelijke diepgang noodzakelijk voor mijn migratieproces.

Ik verwacht van pixelzebra dat zij de migratiebehoeften van de klant tot in de kleinste details vastlegt. Het gaat hierbij niet alleen om specifieke wensen en eisen van velden binnen een entiteit of de plaatsing van een veldje binnen formulier, maar ook bijkomende zaken zoals het selecteren van een transformatiestap en het vastleggen van testcasussen voor de gebruikersacceptatietests.

In dit hoofdstuk ga ik dieper in op de vier vlakken: eliciteratie, analyse, specificatie en validatie, van het requirement development proces van Abran en Moore. Hierin licht ik toe hoe ik verwacht dat de pixelzebra gaan werken en wat dit voor gevolgen heeft voor de organisatie.

Eliciteratie

Eliciteratie, ook wel de ontdekkingstocht voor het expliciteren van de (on)bewuste wensen en eisen genoemd. Een deel van dit subgebied wordt opgelost in de scoping fase door het identificeren van de stakeholders en hun commitment richting het migratieproject. De volgende stap is om per processtap de requirements van deze stakeholders te identificeren middels het toepassen van eliciteratietechnieken (Swart, 2010).

In het draaiboek voor de migratie draag ik twee technieken aan om de requirements te verzamelen. Eén is het houden van 1 op 1 interviews met de stakeholders. Een tweede optie is het houden van een co-creatie workshop. Zo bied ik pixelzebra twee effectieve eliciteratie methoden die aansluiten op haar manier van werken.

Om pixelzebra een handvat te bieden voor het inwinnen van requirements heb ik het interviewboekje aangedragen. Het interviewboekje is een goede aanvulling op pixelzebra's huidige werkwijze omdat het de consultants assisteert in het structureren van een requirementsproces (Cohen & Crabtree, 2006) dat door complexiteit moeilijk te structureren valt (Oracle, 2011). Daarnaast forceert het opstellen van een interviewboekje pixelzebra om van te voren na te denken over de vragen die er gesteld moeten worden en de wijze waarop deze gesteld worden. Met name de wijze waarop is belangrijk, het kan een interview namelijk positief of negatief beïnvloeden, bijvoorbeeld door het projecteren van de eigen mening op de klant (Williamson, Campion, Malos, Roehling, & Campion, 1997). Voor mijn migratieproces, dat geen fouten permitteert, verwacht ik daarom dat pixelzebra hier bewust mee om gaat.



Analyse

Ik wil graag dat pixelzebra zich bewust is dat de kwaliteit van de requirements directe invloed heeft op de kwaliteit van het migratieproject (de Swart, 2010). Intern documenteert pixelzebra sporadisch en bestaande en de documentatie is veelal ad hoc van aard; zowel in de selectie als uitvoering. Om de organisatie te assisteren bij het waarborgen van de requirementskwaliteit heb ik een aantal technieken geselecteerd die bij zullen dragen aan kwalitatief goede requirements. Dit verhoogd de kans op een succesvol verloop van mijn migratieproces.

Domeinmodel

De eerste techniek die ik aandraag is het maken van een domeinmodel of een Entity Relationship model. Door een domeinmodel op te stellen kan pixelzebra toetsen of er bepaalde entiteiten missen, of er entiteiten overbodig zijn en of de relaties tussen de entiteiten correct gelegd zijn (Warmer & Kleppe, 2011). Op deze manier wordt er een levend artefact gecreëerd dat bij ieder overleg gebruikt kan worden om te controleren of wijzigingen in de requirements de kwaliteit van de migratie ondermijnen (DeMarco, 1979).

Prioriteringstechnieken

Het “Handboek Requirements” (de Swart, 2010) biedt een voorgeselecteerde lijst van vijf prioriteringstechnieken die geschikt zijn voor het prioriteren van requirements. Dit zijn de volgende technieken:

- Onderverdeling: Hoog, Middel, Laag;
- De MoSCoW-prioritering;
- Relatieve prioritering;
- De tevredenheids- en ontevredenheidsratio's;
- De urgentie versus belangrijkeheidsmatrix.

Ik heb gekozen voor de laatste prioriteringstechniek omdat deze aansluit op mijn TELT-proces (zie H6.4 voor toelichting) (Wiegers, 2003). De mogelijkheid om de urgentie te bepalen geeft pixelzebra een handvat om te bepalen of requirements in de eerste of tweede transformatiefase uitgevoerd moeten worden. Dit staat los van hoe belangrijk de stakeholder het vindt, het gaat bij de urgentie puur om het vaststellen van requirements die noodzakelijk zijn om de afgesproken datakwaliteit te waarborgen. Denk hierbij aan heterogeniteitsproblemen op schema, datatype en value niveau (Eltabakh, 2012).

	Belangrijk	Minder belangrijk
Urgent	Transform 1	Transform 1
Niet urgent	Transform 2	Nieuw project

Figuur 19: Prioriteren van transformatie in het migratieproces

Naast de urgentie is er de belangrijkheid. Dit onderdeel vergt een stukje verwachtingsmanagement van pixelzebra, de verkeerde vragen zullen namelijk leiden tot een prioritering waarin alles belangrijk is (Wiegers, 2000). Het bepalen van de belangrijkheid is in mijn migratieproces direct van invloed op de kosten van het project. Belangrijke maar niet urgente requirements zullen door pixelzebra aangedragen worden voor uitvoering in de tweede transformatiefase na de dataoverdracht. Dit betekent dat het migratieproject duurder zal uitvallen en meer tijd zal kosten. Deze kosten kunnen cumulatief zijn indien er gekozen is voor een “trickle” na afloop van de initiële dataoverdracht (Boehm, 1981). Alle nieuwe data zal immers steeds weer getransformeerd moeten worden, hetgeen steeds opnieuw werkzaamheden van pixelzebra vergt. Minder belangrijke requirements die niet urgent zijn zullen door pixelzebra buiten de scope van het project gehouden worden. Er zal aandrigen worden om deze in een losstaand project op

te nemen na afronding van de migratie. Zo kan het bedrijf kosten besparen of uitstellen en krijgt pixelzebra sneller een decharge op het migratieproject. De scope kan tevens beter bewaakt worden doordat één groot project opgedeeld wordt in kleinere projecten (Nielsen, 2010).

Horizontaal prototype

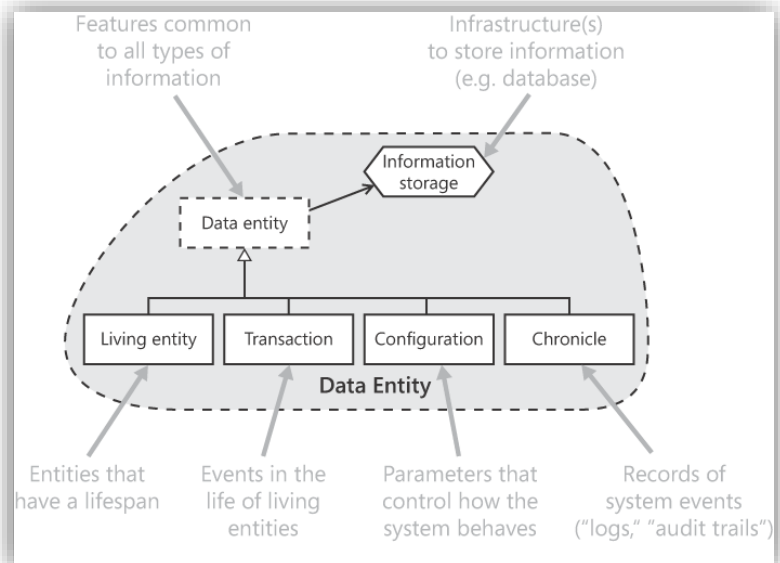
Een goed en eenvoudig middel om de vastgelegde requirements levend te maken voor de klant is een prototype (Wiegers, 2003). Er zijn grofweg twee manieren om een prototype aan te dragen tijdens het requirementsproces: een horizontaal prototype of een verticaal prototype (Pohl & Rupp, 2011). Het primaire doel van een verticaal prototype is om een 'proof of concept' te leveren en daarmee de haalbaarheid van een project aan te tonen. Gezien de wensen van pixelzebra om te een aantal standaard migraties uit te voeren zal dit prototype veelal overbodig zijn. Alleen in extremere gevallen waarin er grove financiële en/of juridische obstakels zijn zal ik het gebruik van dit type prototype aanraden, anders wegen de kosten waarschijnlijk niet op tegen de baten (Swart, 2010) (Boehm, 1981).

In plaats van het verticale prototype heb ik gekozen om het horizontale prototype te gebruiken als laatste analysemiddel. Een horizontaal prototype is in principe niets meer dan een Mock-up van een entiteitformulier in Dynamics CRM. De consultant heeft hierin de vrijheid hoe hij/zij deze wil tonen. Dit kan met PowerPoint, afbeeldingen of zelfs in Dynamics CRM zelf. Door Mock-ups te maken van de formulieren kan gecontroleerd worden of alle benodigde data aanwezig is. Zo is het prototype een middel om nieuwe requirements te eliciteren, om de requirements te toetsen en om overbodige requirements te verwijderen (Swart, 2010). Tot slot kan het prototype in de nazorgfase herbruikt worden voor het opstellen van user acceptance tests, wat weer tijd bespaard.

Specificatie

Het specificeren van requirements beslaat het daadwerkelijk vastleggen ervan. Er zijn talloze checklists te vinden die 'perfecte' requirements te kunnen vastleggen (Firesmith, 2003). Deze checklists hebben echter veel overhead zijn veelal algemeen opgesteld. Naast het gebruik van een checklist is er een optie om het specificeren van requirements uit te voeren met behulp van een requirementspatroon (Withall, 2007). Requirementspatronen bieden structuren die aansluiten op projecten met bepaalde karakteristieken zoals het werken met entiteiten.

Om te zorgen dat de requirements zo perfect mogelijk vastgelegd zijn heb ik een requirementsproces opgenomen in mijn draaiboek dat specifiek bedoeld is voor omgevingen die werken met levende entiteiten zoals Dynamics CRM Online. Het patroon dat ik heb gekozen heeft de naam 'Data Entity Pattern' en biedt een framework voor het vastleggen, categoriseren en specificeren van requirements. Figuur 20 illustreert de wijze waarop



Figuur 20: Het Data Entity Pattern voor het vastleggen van requirements

het patroon is opgebouwd. Binnen de categorisatie zijn kenmerkende eigenschappen te herkennen die terugkomen in Dynamics CRM. Zo is de basis van Dynamics CRM opgebouwd uit levende entiteiten, bevat de omgeving eveneens een configuratie op basis van entiteiten en heeft Dynamics CRM audit logs. Gezien de vrijwel identieke overeenkomst tussen het Data Entity Pattern en Dynamics CRM Online zie ik het patroon als uiterst geschikte techniek om requirements zo specifiek en expliciet mogelijk vast te leggen binnen mijn migratieproces. Bijlage 4 bevat templates voor het vastleggen van de requirements.

Validatie

Het valideren van requirements is de laatste stap van het Requirement development proces van Abran & Moore. Het is in mijn migratieproces tevens de laatste stap in de voorbereidingsfase waarin er actief gecommuniceerd hoeft te worden met de klant. Alle stappen tot aan het toetsen van de resultaten kunnen in theorie uitgevoerd worden zonder dat hier nog inmenging van de klant nodig is.

Hoewel het valideren van requirements een tijdrovend proces is zal de verloren tijd ruimschoots teruggewonnen worden in de latere fasen van het migratieproces (Wiegers, 2003). Er is een ruime keuze qua validatietechnieken die allemaal goed toepasbaar zijn binnen requirements engineering. Ik heb een keuze gemaakt voor twee technieken die, als de consultant deze goed toepast, genoeg raakvlak hebben met alle belangrijke aspecten van het migratieproces om zo kwalitatief goede requirements te creëren (de Swart, 2010). De twee technieken zijn: reviewen, bedoeld voor het valideren van requirements op inhoud en checklists, bedoeld voor het valideren van requirements op basis van metagegevens.

Reviewen

Het houden van een reviewsessie is één van de meest gebruikte technieken om requirements te valideren. Dit kan zowel formeel als informeel gedaan worden. Inhoudelijk betekent het reviewen van de requirements niets anders dan het kritisch doornemen ervan. Er is niets mis met deze techniek behalve dat het een zeer tijdrovend proces is; met name als dit middels individuele sessies met individuele stakeholders gedaan moet worden. Om de snelheid te bevorderen zonder in te boeten aan de kwaliteit heb ik een samensmelting van drie technieken aangedragen die samen de kwaliteit van een formele review evenaren (Pohl & Rupp, 2011).

De basis van de techniek ligt in het houden van een informele review, een walkthrough genoemd. De walkthrough kenmerkt zich door het hebben van minder strikte regels dan een review. Het aantal rollen dan minimaal vervuld moet worden is bijvoorbeeld een stuk kleiner en de sessie zelf is vergelijken met een review informeler. Door de sessie uit te voeren in groepsverband wordt er tevens een platform gecreëerd waarin discussies over de requirements ondersteund worden. De consultants van pixelzebra hebben ervaring met dergelijke sessies en zullen goed het tempo erin kunnen houden.

Om de reviewers meer perspectief te bieden adviseer ik om de stakeholders elkaars stakeholderrollen toe te delen tijdens de walkthrough. De zogenaamde 'Perspective-Based Reading' techniek levert bewezen verbeterde resultaten als het gaat om requirement validatie (Basili, et al., 1996). Verschillende perspectieven zullen daarnaast verschillende manieren van validatie leveren (Shull, Rus, & Basili, 2000). De consultants kunnen deze techniek gebruiken om hun eigen enthousiasme over te brengen naar de klant omdat de techniek helpt bij het 'leuk' houden van de reviewsessie. Dit helpt om de stakeholders scherp te houden gedurende de sessie.

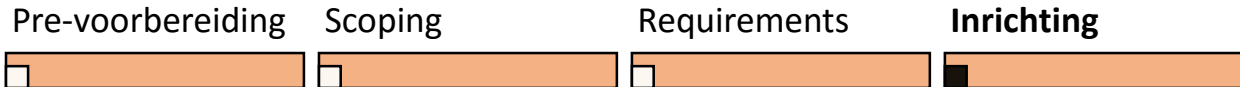
De laatste techniek die ik adviseer te gebruiken tijdens de reviewsessie is het horizontale prototype (te vinden onder de kop 'Analyse' van dit hoofdstuk). Door het reviewen te combineren met Mock-ups krijgen

de reviewers meer gevoel met de requirements. Dit is de effectiefste manier om fouten in de requirements tijdig te ontdekken (Jones, 1998). Dit sluit eveneens goed aan op pixelzebra's huidige werkwijze die veelal gebruik maakt van demo's en prototypen.

Checklist

De reviewsessie levert waardevolle input voor het valideren van requirements op basis van inhoud. Om de volledigheid van een requirement te garanderen is er meer nodig dan alleen een validatie op de inhoud. Er is ook een validatie nodig die ingaat op de volledigheid, begrijpelijkheid en overeenstemming van de requirement (Pohl & Rupp, 2011). Het middel waarvoor ik heb gekozen om de 'overhead' van individuele requirements te valideren is de checklist.

Voor het opstellen van een lijst met punten voor de checklist heb ik gebruik gemaakt van de IEEE 830 standaard voor het vaststellen van kwaliteitscriteria voor requirements (IEEE std, 1998). Ten opzichte van andere checklists met gelijke doeleinden is deze het meest compleet. Een bijkomend voordeel is dat niet alleen de requirements maar ook de handelingen van de consultants getoetst worden. Dit dwingt hen tot zelfreflectie met als gevolg dat zij fouten in de eigen werkwijze nog kunnen bijsturen voordat het te laat is. Met behulp van de checklist is het mogelijk om fouten in de requirements op te sporen die tijdens een reviewsessie gemist zouden worden (Boehm, 1984).



- ⇒ Pre-migratie
- ⇒ Data cleansing
- ⇒ Transformatie

6.4 Inrichting

Uit mijn literatuuronderzoek en interviews heb ik geconcludeerd dat datamigratie uit meer bestaat dan alleen de dataoverdracht. Zo moet de data opgeschoond en/of getransformeerd, om de data integriteit te bewaken, en moet de metadata voorafgaand aan de dataoverdracht gemigreerd worden om te zorgen dat de datamappings gemaakt kunnen worden. Ik heb deze drie stappen: Pre-migratie (van de metadata), data cleansing en transformatie samengevoegd onder de sub-fase “inrichting”.

Pre-migratie

Onder de pre-migratie heb ik alle handelingen samengevoegd die niet gaan over inhoudelijke data (Tonnu, et al., 2014). Het gaat hierbij zowel om onderdelen van de CRM-omgevingen als onderdelen van Scribe Online. De onderdelen die ik heb geïdentificeerd voor de pre-migratie zijn als volgt:

- De migratie van alle CRM-metadata (Entiteitenmodel, workflows, etc.);
- Het mappen van de entiteiten (valt deels onder de voorbereiding en deels onder de uitvoeringsfase);
- Het inrichten van Scribe Online.

Metadata migratie

In mijn draaiboek heb ik niet lang stilgestaan bij het migreren van de metadata. Dit is een handeling die de consultants van pixelzebra vrijwel dagelijks uitvoeren middels implementaties. De enige crux in dit proces is zijn importbeperkingen tussen verschillende versies van Dynamics CRM en verschillende CRM-omgevingen over het algemeen. Normaal gesproken kan customization vanuit de ene Dynamics CRM omgeving rechtstreeks gemigreerd worden naar een andere Dynamics CRM omgeving met gebruik van zogenaamde oplossingen (Microsoft, 2016). Een oplossing is een verzameling metadata binnen Dynamics CRM. Ter ondersteuning heb ik een overzicht toegevoegd in het draaiboek om aan te geven welke versies van Dynamics CRM compatibel zijn. Deze is echter hoofdzakelijk bedoeld als hulpmiddel voor de consultants bij het inschatten van de tijd die nodig is voor de migratie.

	Import to Version (target)						
	2011 (v5.0)	2013 (v6.0)	2013 SP1 (v6.1)	2013 SP1 (v6.1) With Enhanced Service Capabilities	2015 (v7.0)	2015 (v7.1)	2016 (v8.0)
Microsoft Dynamics CRM Version							
2011 (v5.0)	Yes	Yes	Yes	Yes	No	No	No
2013 (v6.0)	No	Yes	Yes	Yes	Yes	Yes	Yes
2013 SP1 (v6.1)	No	No	Yes	Yes	Yes	Yes	Yes
2013 SP1 (v6.1) Export As v6.0	No	Yes	Yes	Yes	Yes	Yes	Yes
2013 SP1 (v6.1) With Enhanced Service Capabilities	No	No	No	Yes	Yes	Yes	Yes
2013 SP1 (v6.1) With Enhanced Service Capabilities Export As v6.0	No	Yes	Yes	Yes	Yes	Yes	Yes
2015 (v7.0)	No	No	No	No	Yes	Yes	Yes
2015 (v7.1)	No	No	No	No	No	Yes	Yes
2016 (v8.0)	No	No	No	No	No	No	Yes

Figuur 21: Overzicht van Dynamics CRM-omgevingen die compatibel zijn voor solution import

Entity mapping

Eén van de grootste bottlenecks van een datamigratie is het mappen van de bron- en doelgegevens (Hovy, Ambite, & Philpot, 2003). Voor pixelzebra is het mappen manueel en daarom een tijdrovend. In overleg met de productowner is dit dan ook één van de grootste aandachtspunten van mijn project geweest. Om een handmatig proces te versnellen zijn er twee mogelijke technieken die toegepast kunnen worden: Het gebruiken van een template of automatisering. Zo helpt de template om middels structurering sneller handmatig te kunnen mappen en helpt automatisering door handmatige taken automatisch te laten verlopen. Ik heb beide methoden gebruikt om tijdswinst te behalen op het mappen van entiteiten.

Dynamics CRM en soortgelijke CRM-systemen maken gebruik van een standaardpakket. Een standaardpakket is niets meer dan een verzameling van entiteiten die altijd aanwezig zullen zijn in een omgeving. Ik had met gebruik van een tool of maatwerk mogelijk automatisch een mapping voor deze entiteiten kunnen generen. Dit is echter niet de effectiefste methode voor entiteiten die altijd aanwezig zijn binnen CRM. De betere optie is het eenmalig handmatig mappen van alle standaard entiteiten binnen Dynamics CRM Online. Omdat de entiteiten standaard aanwezig zijn binnen de omgeving zijn de mappings onbeperkt herbruikbaar. Omdat het handmatig mappen een tijdrovende klus is, heeft dit eenmalig mappen meer tijd gekost van mijn project dan ik had verwacht. Relatief aan de tijdswinst die het oplevert is dit waarschijnlijk echter wel één van de succesvolste manieren waarop ik het migratieproces versneld heb.

Voor custom entiteiten is het mappen lastiger. Deze hebben geen vaste namen en velden, hierdoor kan ik geen statische templates maken. Wel zijn de velden binnen een custom entiteit beperkt tot een select aantal datatypes zoals string, integer of decimaal. Ik heb dit principe gebruikt om het mappen van custom entiteiten (deels) te automatiseren. Dit geldt echter alleen voor entiteiten en velden die in beide migratieomgevingen dezelfde naam hebben. Het is aan pixelzebra om tijdens de migratie te zorgen dat dit het geval is. Ondanks de beperkingen is dit een grote stap vooruit ten opzichte van de handmatige werkwijze die pixelzebra voorheen hanteerde. In hoofdstuk 7.3 ga ik dieper in op de werking van het automatisch mappen.

Scribe inrichten

Zoals de bron- en doelomgeving ingericht moeten worden, moet ook Scribe Online ingericht worden om de migratie uit te kunnen voeren. Om Scribe Online voor te bereiden moeten de volgende stappen uitgevoerd worden om de tool 'migratie-klaar' te maken:

- De connectie informatie van de bron- en doelomgeving moet ingevuld worden;
- De mappings moeten gecorrigeerd worden om de juiste connectie informatie te bevatten;
- De mappings moeten geïmporteerd en gesorteerd worden.

Iedere mapping handmatig corrigeren, importeren en sorteren zou bijna evenveel tijd kosten als het handmatig mappen van entiteiten. Om de pixelzebra werk uit handen te nemen heb ik er daarom wederom voor gekozen om deze handelingen te versnellen. Een template was in dit geval geen optie. Het aanpassen van tientallen tot honderden verschillende XML bestanden (Scribe Mappings zijn opgebouwd met XML) is niet praktisch op te lossen met een template. Het automatiseren van dit proces leek mij daarom de beste optie.

Tijdens het automatiseren van Scribe ben ik tegen het probleem aangelopen dat ik niet direct de connecties in de mappings kan aanpassen. De XML bestanden bevatten alleen een referentie naar een connectie, niet de connectie informatie zelf. Uiteindelijk heb ik dit opgelost door middel van een template

en automatisering. Door één mapping met de correcte connectie informatie als template te nemen kan ik de connectie informatie voor alle andere mappings corrigeren. Dit betekent dat pixelzebra slechts eenmalig een mapping hoeft te corrigeren in tegenstelling tot de 300+ maal die anders nodig zou zijn. In hoofdstuk 7.3 ga ik dieper in op de technische details van mijn oplossing.

Data cleansing

Voorafgaand aan mijn afstuderen heb ik eenmaal eerder aan een data-integratie mogen werken voor pixelzebra. Wat mij hier vooral van is bijgebleven is de gebrekkige controle die toen is uitgevoerd op de brondata, oftewel het 'cleansen' van de data. Er passeerde soms weken voordat eenvoudige problemen m.b.t. de datakwaliteit opgepakt dan wel opgelost werden. Dit is een groot gebrek in pixelzebra's werkwijze die haast slordig genoemd mag worden. Daarnaast zou een dergelijke werkwijze mijn migratieproces volledig teniet doen omdat het goede data integriteit van de bronomgeving vereist.

Pixelzebra is niet alleen in het negeren van data cleansing; ondanks het belang van data en analyse in de hedendaagse tijd blijft datakwaliteit een doorn in het oog van menig organisatie. De grondoorzaak van datakwaliteitsproblemen zijn grofweg te classificeren als één van de volgende vier oorzaken:

- Foutieve input van gebruikers;
- Foutieve input data;
- Consolidatiefouten;
- Beleids- of procesveranderingen (Hellerstein, 2008).

Cleansing vooraf

Mijn migratieproces helpt bij het data cleansen door hier een mate van structuur in aan te brengen. De volgende stappen komen hierbij aan bod:

- Bij voorkennis van gebreken van de datakwaliteit:
 - o Noodzaak bepalen;
 - o Scopen van de transformatie;
 - o Asynchroon uitvoeren van de transformatie.
- Bij gebrek aan voorkennis:
 - o Op basis van business requirements exploratief Data Minen om gebreken in de data te achterhalen (Dasu & Johnson, 2003);
 - o Noodzaak bepalen;
 - o Scopen van de transformatie;
 - o Asynchroon uitvoeren van de transformatie.

Cleansing achteraf

Dit behoeft weinig uitleg. Als mijn migratieproces gevolgd wordt, zou in ieder geval de kritieke data in de doelomgeving integer moeten zijn. Verdere cleansing kan uitgevoerd worden middels de stappen zoals beschreven bij cleansing vooraf. Het enige verschil is dat de cleansing uitgevoerd wordt binnen de doelomgeving.

Transformatie: ETL wordt TELT

Extract-Transform-Load(ETL) processen zijn de standaard als het gaat om datamigraties en data integraties (Albrecht & Naumann, 2008). Standaard ETL-processen hevelen periodiek (trickle) of eenmalig (big bang) informatie over van één of meerdere systemen naar één geconsolideerd systeem. In veel gevallen is het ETL-proces inefficiënt naar mate de complexiteit van de transformatie stijgt. Met name voor bulkoperaties geldt dat waar de complexiteit stijgt de snelheid daalt (Ankorion, 2005). De kern van het ETL-proces in de loop der jaren onveranderd gebleven, desalniettemin zijn er verschillende oplossingen uitgedacht om het ETL-proces efficiënter te maken.

Een voorbeeld van een toevoeging op het standaard ETL-proces is het Change Data Capture principe. Dit principe maakt het ETL-proces event-driven en verlicht zo de werklast door overbodige ETL-operaties te voorkomen. Helaas is dit principe alleen toe te passen in trickle migraties en de migraties van pixelzebra zijn eerder van het type big bang (IBM, 2007) (Oracle, 2011). Naast het Change Data Capture proces zijn er nog een aantal oplossingen die niet goed aansluiten op de werking van Dynamics CRM of niet passen binnen het beeld dat ik heb van een efficiënte migratie voor pixelzebra; zelfs Scribe heeft zijn tekortkomingen op dit gebied (Fagundes, 2013). Met name de voorwaarden omtrent het uitvoeren van bulkoperaties luistert nauw in zowel Scribe Online en Scribe Insight. Zo dwingt Scribe Online af dat de mappings geen Lookup operaties mogen bevatten als je gebruik wilt maken van de bulk optie. Het wegnemen van een dergelijke operatie levert op zichzelf weer problemen op.

De oplossing

Mijn oplossing om een aantal efficiëntie problemen in pixelzebra's migratieproces op te lossen is om de transformatiestap(T) uit het ETL-proces te trekken en deze voor en na de Extract en Loads stappen te zetten. Het resultaat is een Transform-Extract-Load-Transform(TELT) proces. Dit betekent dat er geen noemenswaardige transformatiestap meer is tussen de bron- en doelomgeving waardoor zoveel mogelijk transformaties nu binnen het domein van of de bron- of het doelomgeving uitgevoerd zullen worden. Uiteraard kunnen transformaties tussen twee systemen niet compleet weggenomen worden in gevallen waarin een omgeving bepaalde datatypen of datastructuren niet ondersteunt. Binnen de scope van dit project, waarin beide omgevingen van het type Dynamics CRM zijn, is dit echter goed mogelijk. Figuur 22 toont een voorbeeld van de werking van TELT ten opzichte van ETL.

Casus: De bronomgeving bevat een veld van het Type string. Het veld wordt gebruikt om de status van het Account aan te geven. In de praktijk bevat dit veld altijd de waarde: klant, leverancier of partner. Het probleem is dat soms verkeerd getypt wordt, soms met hoofdletter en soms zonder. Op dit recht te trekken wil de klant graag dat dit veld in de bronomgeving een optieset wordt met de drie bovengenoemde waarden. Hieronder is aangegeven hoe standaard ETL dit zou transformeren en hoe mijn TELT-proces dit zijn doen, met als aanname dat de transformatie uitgevoerd dient te worden in de eerste transformatiefase.		
	ETL	TELT
Transformatiewijze	Met behulp van een migratietool worden de stringwaarden in de bronomgeving gemapped aan één van de drie waarden in de doelomgeving. De transformatie vindt plaats bij het uitvoering van de dataoverdracht.	Met behulp van een asynchrone workflow worden alle stringwaarden omgezet naar een numerieke waarde die overeenkomt met de gelijknamige waarde van de optie set in de doelomgeving. Tijdens de dataoverdracht wordt de waarde zonder transformatie gemigreerd.
Verwachte doorlooptijd Op basis van 1000 records	+/- 7 minuten	+/- 3 minuten
Opmerking	Indien er afwijkende waarden zijn in de bronomgeving zullen deze falen tijdens de migratie. Dit tast de data integriteit van de doelomgeving aan.	Indien er afwijkende waarden zijn in de bronomgeving komen deze aan het licht na afloop van de workflow. Deze kunnen voorafgaand aan de dataoverdracht gecorrigeerd worden om zodoende de data integriteit in de doelomgeving te waarborgen.

Figuur 22: Een vergelijking tussen ETL en TELT

Transformatie één

Transformeren in de bronomgeving is geen nieuw idee. Het is al eens voorgesteld door Fivien Savitri (2011) met als doel het ETL-proces efficiënter te maken. Savitri's idee is in de basis gelijk aan mijn idee al specificeer ik het voor CRM-omgevingen. Door te transformeren in de bronomgeving wordt de 'workload' tijdens de datawerkelijke datamigratie verlicht. De datamapping voor de migratie wordt minder complex, hetgeen op zijn beurt de doorlooptijd voor het overzetten van de data exponentieel kan verkleinen (CRM, 2014). Met name het gebruik van Scribe Online noodzaakt deze aanpak omdat Scribe Online bulkoperaties

alleen toestaat op eenvoudige datamappings. Complexiteit in de vorm van query's, of een hoge mate van operaties binnen één mapping maakt de bulkoptie van Scribe inefficiënt en in veel gevallen zelfs onmogelijk (Fagundes, 2013).

Dat het transformeren binnen het doelsysteem de dataoverdracht eenvoudiger maakt is evident. Dit betekent echter niet dat het migratieproces als geheel efficiënter wordt. Het transformeren binnen de bronomgeving gaat niet vanzelf. Dit kost eveneens tijd en geld. Vergeleken met een standaard ETL-proces durf ik echter te beweren dat het transformeren binnen de bronomgeving dermate veel voordelen heeft voor **pixelzebra** dat dit zonder twijfel de betere optie is voor haar standaard migratieproces. In de volgende alinea's zal ik toelichten waarom ik dit vind.

ETL-processen noodzaken veelal het platleggen van zowel de bron- als de doelomgeving. Om deze reden worden migraties meestal in de nacht uitgevoerd wanneer er vrijwel geen gebruikers actief zijn. Dit gaat prima voor trickle-migraties maar wordt moeilijker voor grootschalige eenmalige migraties, daar is één nacht eenvoudigweg te weinig voor (Ankorion, 2005). Indien dit onrealistische scenario toch doorgevoerd wordt heeft dat negatieve gevolgen voor de kwaliteit van de migratie (Morris, 2012); iets wat ik koste wat het kost wil voorkomen. In eerste instantie lijkt het transformeren binnen één omgeving een sofisme. De transformatie zal veel processorkracht vergen en zal de omgeving eveneens functioneel platleggen. Ik heb hier echter de oplossing voor gevonden het behulp van de asynchronisatie functionaliteit van Dynamics CRM.

Zowel workflows als plug-ins kunnen asynchroon 'afgevuurd' worden in Dynamics CRM. Dit betekent dat Dynamics CRM de actie pas uitvoert zodra de omgeving middelen vrij heeft om dit te doen. Het gevolg is het systeem geen of minder hinder ondervindt van de acties (Microsoft, 2016). Door de transformaties in te plannen, middels een workflow of een plug-in, kunnen transformaties 24/7 lopen. Dit levert een enorme tijdsbesparing op om verschillende redenen:

- Een deel van de migratie kan nu in de voorbereidingsfase uitgevoerd worden, parallel aan andere taken;
- De dataoverdracht verloopt sneller door eenvoudigere mappings (CRM, 2014);
- Transformeren binnen één systeem forceert data cleaning (Savitri, 2011);
- Indien klant zelf enige kennis over CRM bezit kan deze zelf workflows aanmaken voor het transformeren, hetgeen de organisatie geld en **pixelzebra** tijd bespaard.

Transformatie twee

De tweede transformatiestap zal uitgevoerd worden in de doelomgeving. In de voorbereidingsfase zullen afspraken gemaakt worden welke transformaties tijdens welke fase uitgevoerd zullen worden. In hoofdstuk 6.3 heb ik toegelicht welke criteria hier voor gelden. Daarnaast moeten er afspraken gemaakt worden over de uitvoering van de tweede transformatie; wordt deze meegenomen in de migratie of wordt deze voor een apart project bewaard. In hoofdstuk 8.3 licht ik de gevolgen van deze keuze verder toe.



H7 het uitvoeren van de dataoverdracht

In dit hoofdstuk beschrijf ik de stappen die ik heb gezet om de doorlooptijd van de uitvoeringsfase van mijn project zo snel mogelijk te maken. De uitvoeringsfase beslaat alle handelingen die nodig zijn om de data daadwerkelijk over te zetten van de bron- naar de doelomgeving. Onder de uitvoeringsfase vallen twee handelingen die gekenmerkt kunnen worden als de grootste bottlenecks van een migratietraject: het mappen van de metadata en het uitvoeren van de daadwerkelijk dataoverdracht. Ik heb bij het verbeteren van de doorlooptijd voornamelijk aandacht besteed aan deze twee bottlenecks. Mijn aanname hierbij was dat het verbeteren van de bottlenecks het meeste tijdswinst zou opleveren (Albrecht & Naumann, 2008).

De kern van mijn verbetering bestaat uit het behalen van tijdswinst middels automatisering van de uitvoeringsfase. Automatisering is een bewezen effectieve methode voor het verbeteren en versnellen van de dataoverdracht (Halevy, Rajaraman, & Ordille, 2006). Voor het automatiseren heb ik gebruik gemaakt van twee tools:

- Scribe Online (Een standaardpakket voor datamigratie);
- Scribe CRM Migration Assistant (Een stuk zelfgebouwde software).

Scribe Online dient als basistool voor het migreren van de data. De tool bevat een ingebouwde dataconnector voor Dynamics CRM die assisteert bij het bewaken van de datakwaliteit. Daarnaast bevat de tool een bulkoptie om de dataoverdracht te versnellen. De Scribe CRM Migration Assistant wordt gebruikt als aanvulling op Scribe Online. Er zijn tijdswinsten mogelijk die Scribe Online niet zelfstandig kan behalen. Met behulp van Scribe CRM Migration Assistant worden deze handelingen mogelijk gemaakt. Een voorbeeld hiervan is een rapportage overzicht van de migratie.

De standaardwijze waarop Scribe migreert is efficiënt ten opzichte van een handmatige migratie, maar er zijn stappen te ondernemen om dit verder te verbeteren (CRM, 2014). Kort door de bocht heb ik dit voornamelijk gedaan door wegnemen van veiligheidschecks. Ieder record dat middels Scribe Online verwerkt wordt, wordt onderworpen aan een aantal 'checks' om te zorgen dat het overzetten van een record niet faalt. Deze checks vertragen een dataoverdracht echter enorm. Eén van de punten die ik in dit hoofdstuk uitgebreid zal behandelen is de wijze waarop ik deze checks heb weggelaten uit het proces, wat de voordelen hiervan zijn en welke risico's het wegnemen met zich meebrengt.



Dynamics CRM Online

Scribe Online

Scribe CRM Migration Assistant



⇒ Structuur

⇒ Configuratie

⇒ Office 365

7.1 Dynamics CRM Online

Voordat ik uitleg geef of over de wijze waarop ik het uitvoeringsproces heb geautomatiseerd vind ik het nodig om meer inzicht te geven de werking van Dynamics CRM Online; dit zal immers altijd de doelomgeving van de CRM-migratie zijn en de karakteristieken van de Dynamics CRM hebben dan ook een grote invloed op mijn migratieproces. In dit hoofdstuk zal ik een beschrijving geven van drie onderdelen van Dynamics CRM Online die rol spelen in de uitvoeringsfase. Deze zijn als volgt:

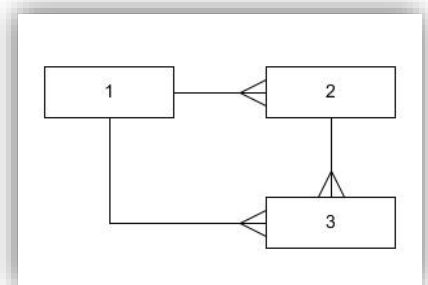
- De structuur (De gevolgen van de databasestructuur op de migratie);
- De configuratie (De beperkingen die inherente eigenaardigheden van CRM hebben op de migratie);
- De relatie met Office 365 (De rol van Office 365 in de migratie naar Dynamics CRM Online toe).

Structuur

Onder de motorkap draait Dynamics CRM Online op een SQL database. Deze database is echter nooit rechtstreeks aan te spreken in de Online versie, daar zit de Xml-structuur van Dynamics CRM voor. Het gevolg hiervan is aan de ene kant beperktere query mogelijkheden. FetchXml, de querytaal in Dynamics CRM Online heeft beduidend minder mogelijkheden dan een SQL-query. Aan de andere zijn er relaties mogelijk zijn met objecten die zich buiten de CRM-database bevinden zoals SharePoint documenten of werkstromen.

Voor het modelleren van de databasestructuur heb ik gekozen voor een database diagram (Warmer & Kleppe, 2011). Ondanks mijn voornemen om alleen gebruik te maken van UML diagrammen heb ik toch gekozen voor de notitie van het Entity Relationship Diagram (ERD). De reden hiervoor is dat ik het Entiteitenmodel door de hoeveelheid aan entiteiten en relaties al zeer complex vind. Kardinaliteiten toevoegen hieraan zou slechts toevoegen aan deze complexiteit. Daarnaast is de kardinaliteit tussen entiteiten niet van belang voor mijn migratie. Referentiele integriteit regel ik aan de kant van de bronomgeving in de voorbereidingsfase en aan de kant van de doelomgeving dwingt Dynamics CRM dit zelf af.

Een element dat wel belangrijk is voor mijn migratie is een overzicht van de sequentie tussen de onderlinge entiteiten (Morris, 2012). Hiermee doel ik de volgorde waarin de entiteiten gemigreerd kunnen worden op basis van hun onderlinge relaties zonder de referentiele integriteit in gevaar te brengen. Figuur 23 toont een eenvoudig voorbeeld van sequentie op basis van onderlinge relaties. De cijfers geven de volgordelijkheid aan van de migratie. Deze is gebaseerd op de onderlinge relaties.



Figuur 23: Voorbeeld van sequentie

Ter verduidelijking van de sequentie heb ik gebruik gemaakt van Peter Coad's (1999) 'Modeling with color in UML' principe. Ik heb hiervoor gekozen omdat dit een eenvoudige en overzichtelijke manier is om een extra dimensie toe te voegen aan een UML (in dit geval ERD) diagram. Figuur 24 toont het Entiteitenmodel inclusief kleurcodering voor het aangeven van de sequentie. Opvallend daarin zijn de witte entiteiten. Deze representeren multi-entitey relaties, oftewel een relatie naar een entiteit waarvan op voorhand niet bekend is welke entiteit dit is. Zo heeft een activiteit een relatie die kan wijzen naar iedere mogelijke entiteit in de Dynamics CRM Online omgeving. Het model inclusief een voorbeeld van een entiteitenmodel zoals deze wordt geleverd door Microsoft zijn te vinden in bijlage 7.



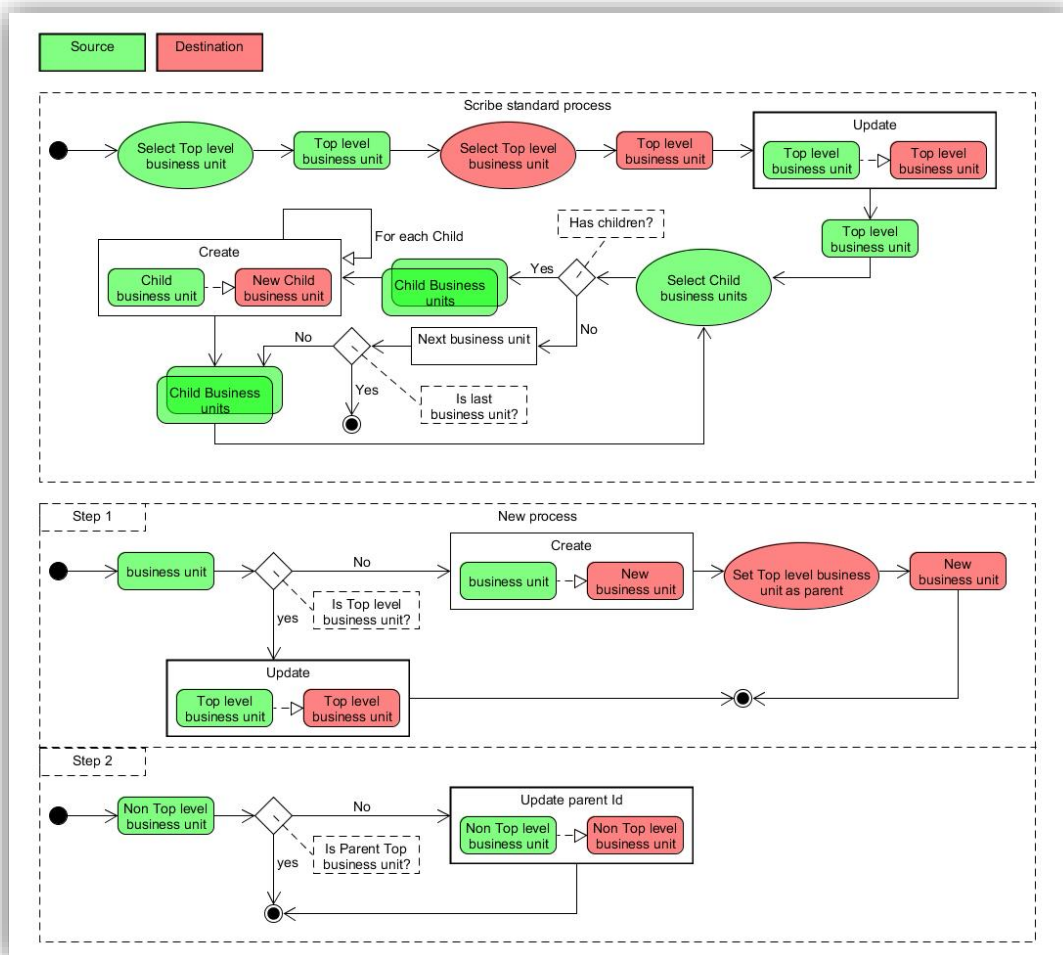
Figuur 24: Entiteitenmodel van Dynamics CRM Online met kleurcodering voor de sequentie

Configuratie

Ik heb een korte beschrijving van de configuratie toegevoegd aan mijn verslag om duidelijkheid te verschaffen over de hordes die ik ben tegen gekomen tijdens mijn project als het gaat om gebruik van Dynamics CRM Online. Een aanzienlijk deel van mijn project is besteed aan het maken van standaard mappings voor de migratie, waarbij ik meerdere malen gestuit ben op zaken die het migreren naar Dynamics CRM Online bemoeilijken.

De eerste 'beïnvloeder' van de tijdwinst zijn de zogenaamde multi-entity relaties. Dit zijn één op veel relaties waarbij de één kant geen vaste entiteit is. Deze relaties kunnen pas aan het einde van de dataoverdracht gemigreerd worden omdat alle records eerst moeten bestaan. Gebeurt dit niet, dan is er een kans dat er een relatie aangemaakt wordt naar een record dat nog niet bestaat. Dit zal de migratiestap doen falen.

Zelf-referentiele relaties en hiërarchie leveren een soortgelijk probleem als multi-entity relaties. Ook hier kan ik niet bepalen of een record al bestaat, ik heb immers geen invloed op de volgorde van migratie binnen één entiteit. Het gevolg hiervan is eveneens dat ik in een tweede migratiestap pas de relatie tussen twee records van dezelfde entiteit kan aanleggen. Figuur 25 toont de wijze waarop Scribe Online standaard de business unit entiteit migreert ten opzichte van hoe ik dit oplos. De business unit heeft een zelf referentiele relatie die hiërarchisch van aard is.



Figuur 25:
Migratie van
business units.

Boven toont de
standaardwijze
van Scribe
Online.

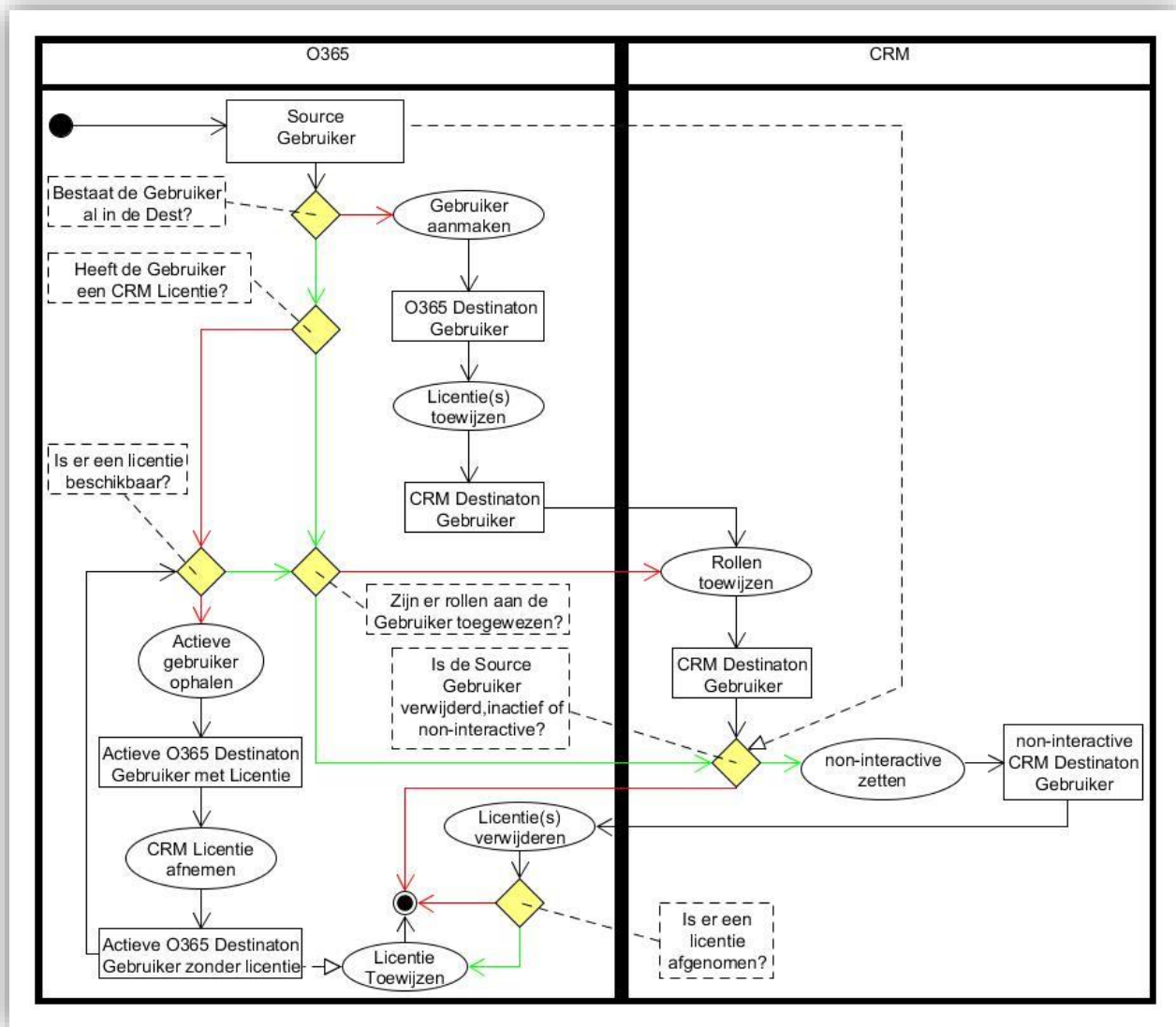
Onder toont de
wijze waarop
business units
gemigreerd
worden in het
migratieproces

De laatste eigenschap van Dynamics CRM Online die mijn migratieproces beïnvloed is het automatisch creëren van records. Om relaties aan te kunnen leggen tussen twee records moet ik de Id's van de records weten. Dit is normaal gesproken geen probleem omdat ik de Id's mee migreer. In een aantal specifieke gevallen maakt CRM eigen records aan. Dit zijn ook nog eens records waarnaar veel gerefereerd wordt. Om de relaties toch te kunnen leggen heb ik creatieve manieren moeten bedenken om de Id's van deze automatisch aangemaakte records te achterhalen.

Een eenvoudig voorbeeld van automatische record generatie is de Valuta entiteit. Bij het aanmaken van een CRM-omgeving wordt er standaard een valuta record aangemaakt. Ik weet niet welke valuta dit is of wat de Id van het record is. Wel weet ik dat als ik probeer een valuta record van het type Euro te migreren naar een omgeving waar Euro de standaard valuta is, dit record en alle records die ernaar verwijzen zullen falen. Ik moet dus een manier hebben om de standaard valuta te vinden en het Id ervan te verkrijgen. In het geval van de valuta heb ik mijn soelaas gevonden in het veld "ISO Currency Code". Iedere valuta heeft deze code. De codes zijn tevens gelijk in iedere omgeving omdat zij moeten voldoen aan de ISO-standaard. Door gebruik van de ISO-code heb ik een veldwaarde die net zoals het Id uniek is voor ieder record in een omgeving. Ik kan deze dus gebruiken om de Id van de standaard valuta in de doelomgeving te achterhalen en zodoende te migreren.

Ik heb nu gekozen om entiteiten met automatisch gegenereerde records te mappen middels andere velden die bruikbaar zijn als identificatiemiddel. Een andere oplossing had kunnen zijn om zelf een veld aan te maken in de bronomgeving en deze als Id te gebruiken. Hoewel dit eenvoudiger zou zijn tijdens het mappen zou dit voor pixelzebra meer werk opleveren. Zij zouden een extra oplossing in moeten laden en deze onderhouden. Op de lange termijn is het mappen met bestaande velden daarom naar mijn mening de betere keuze.

Het migreren van gebruikers is één van de voornaamste redenen waarom ik gekozen heb voor een stuk maatwerk, middels een zelf gebouwd stuk software is het wel mogelijk om het migreren van gebruikers te automatiseren. Zelfs voor organisaties met minder dan 10 gebruikers is het nog een tijdrovende klus om deze aan te maken. Om inzicht te geven in het proces voor het aanmaken van een gebruiker heb ik gekozen voor een UML-flowchart. Dit is een geschikt model binnen de UML bibliotheek voor het modelleren van een algoritme (Warmer & Kleppe, 2011). Figuur 26 toont het proces voor het aanmaken van een gebruiker in Office 365 en Dynamics CRM Online.



Figuur 26: De stappen voor het migreren van gebruikers naar Dynamics CRM Online



Dynamics CRM Online



Scribe Online



Scribe CRM Migration Assistant



⇒ Automatisering

⇒ Voordelen

⇒ Risico's

7.2 Scribe Online

In overleg met de productowner heb ik gekozen voor Scribe Online als primaire tool voor het migreren van data tijdens mijn migratieproces. Bij het ontwerpen van het proces voor de dataoverdracht heb ik daarom een aantal specifieke keuzes gemaakt die alleen mogelijk zijn met Scribe Online. Deze keuzes vervienvoudigen de tijdswinst bovenop de tijdswinst die al behaald wordt met gebruik van standaard Scribe Online. Deze keuzes zijn echter niet zonder gevolgen. In dit hoofdstuk beschrijf ik welke keuzes ik heb gemaakt, welke voordelen ik hiermee behaal en welke risico's daar aan hangen.

Automatisering

Als het gaat om de doorlooptijd is automatisering veelal superior ten opzichte van een manueel proces; een handmatige migratie van duizenden records zou onbegonnen werk zijn en maanden duren. Bij het automatiseren van de dataoverdracht ben ik niet gestopt bij het gebruik van alleen Scribe Online. De mappings die Scribe aanbiedt zijn 'veilig' maar daardoor ook traag. Om de dataoverdracht te versnellen heb ik alle 'checks' verwijderd uit de mappings. Denk hierbij aan lookups om te controleren of een record als bestaat en een IF-statement om te bepalen wat er moet gebeuren als dit scenario zich voordoet. Als al deze checks weggehaald worden blijven er slechts drie stappen die samen de meest basale mapping vormen die mogelijk is in Scribe Online. Dit zijn:

- Een Query voor het ophalen van brondata;
- Een 'Foreach' loop voor het itereren van de brondata;
- Een Create/Update stap voor het toevoegen van de records aan de doelomgeving (Fagundes, 2013).

Mijn keuze voor het vereenvoudigen van de mappings is niet zonder gevolgen, de veiligheidschecks waren immers niet voor niets toegevoegd. Het primaire doel van de checks was het controleren van het bestaan van records bij het toewijzen van een relatie of het aanmaken van een record. Omdat ik de checks heb verwijderd moet ik een andere manier vinden om de 'veiligheid' te garanderen. De eenvoudigste manier die ik kon bedenken om dit te doen, zonder in te leveren op de tijdswinst, was door de entiteiten op basis van hun onderlinge relaties te migreren in volgorde van afhankelijkheid.

Het eenvoudigweg migreren van de entiteiten in een bepaalde sequentie lijkt het probleem, dat het wegnemen van de veiligheidschecks achterlaat, op te lossen. De in 7.1 beschreven configuratie van Dynamics CRM Online gooit echter roet in het eten. Sommige entiteiten kunnen niet in één keer toegevoegd worden. De eigenschappen van bepaalde relaties maakt dit onmogelijk. Om dit op te lossen had ik twee keuzes: of ik ga terug naar een versie van de mappings waarin ik een aantal veiligheidschecks weer toevoeg, of ik deel de migratie op in stappen om de sequentie uit te breiden. Voor vijf specifieke entiteiten heb ik een aantal veiligheidschecks terug moeten plaatsen. De configuratie stond niet toe dat



ik dit niet deed (het ging hierbij met name om automatisch gegenereerde records). Voor het overgrote deel heb ik de dataoverdracht opgedeeld in een aantal stappen. Ik heb gekozen voor het aanbrengen van sequentie om ik hiermee niets inhoudelijks aan de mappings hoeft toe te voegen. Het resultaat, tevens de sequentie voor de gehele dataoverdracht, is te zien in figuur 27.

Proces stap	Operator	Omgeving	Omschrijving
1	Create/Update	Office 365 / Dynamics CRM	Overzetten van gebruikers naar Office 365 en Dynamics CRM. Zie figuur 25 voor toelichting.
2	Import/Export	SharePoint	Exporteren en/of importeren van SharePoint bestanden die gekoppeld zijn of moeten worden aan de Destination omgeving.
3	Create	Scribe	Overzetten standaard entiteiten exclusief afwijkende Lookup/Status velden.
4	Create	Scribe	Overzetten van alle custom entiteiten exclusief Lookup/Status velden.
5	Update	Scribe	Update standaard entiteiten / standaard Lookup velden.
6	Update	Scribe	Update van alle Lookup velden van custom entiteiten en alle Custom velden op standaard entiteiten.
7	Update	Scribe	Update van alle status velden van zowel standaard als custom entiteiten.

Figuur 27: Eenvoudig overzicht van de dataoverdracht stappen

Voordelen

Ik heb uiteraard niet voor niets gekozen om de mappings te vereenvoudigen. Ik hoop hiermee veel tijdswinst te behalen. Om dit te doen heb ik gekeken naar de mogelijkheden die Scribe Online en Dynamics CRM Online mij bieden en welke voordelen/gevaren dit oplevert. Hieronder geef ik een beschrijven van de voordelen het vereenvoudigen van de mappings heeft ten opzichte van de standaard Scribe Online mappings.

Dynamics CRM Online kan een maximum van 2000 Create/Update records in één keer verwerken. Scribe Online kan gebruik maken van deze eigenschap middels een vinkje met de naam 'Batch processing'. Hierdoor wordt de bulkoptie voor de migratie ingeschakeld. De bulkoptie heeft een aantal vereisten om gebruikt te worden, de voornaamste is een extreme beperking in het aantal aanroepen dat naar de bronomgeving gedaan wordt. Dit mag er namelijk één zijn. Dit toont aan waarom ik de mappings heb moeten vereenvoudigen tot één aanroep (Create/Update). In de meest optimale omstandigheden zorgt het gebruik van deze bulkoptie voor een verviervoudiging van de dataoverdracht snelheid (CRM, 2014). Dit levert een enorme verbetering op de doorlooptijd van mijn migratieproces op.

Naast de tijdswinst heeft het gebruik van Scribe Online nog een voordeel ten opzichte van handmatige migratie en zelfs ten opzichte van andere migratietools: de CRM-Connector. Deze ingebouwde connector assisteert de gebruiker bij het bewaken van de data integriteit. Dit doet het door tijdens het mappingproces restricties op te leggen aan wat wel en niet gemapped mag worden. Zo worden heterogeniteitsproblemen op basis van datatypen en waarden voorkomen (Eltabakh, 2012). Had ik gekozen om het proces manueel te houden zouden er onherroepelijk fouten gemaakt worden door foutieve mappings.

Risico's

De wijze waarop ik het uitvoeringsproces voor de dataoverdracht heb ingericht is niet zonder risico's. In principe worden alle records over de 'schutting' gegooid. Bij het leggen van relaties controleer ik niet of het record waarnaar de relatie gelegd moet worden überhaupt bestaat. Platgeslagen komt het er op neer dat voorafgaand aan de dataoverdracht er precies duidelijk moet zijn welke data er overgezet moet worden, welke mogelijke problemen zich kunnen voordoen en hoe dit opgelost is. Dit vraagt veel van de consultant want de scope, requirements en verdere executie van de voorbereidingsfase moeten nagenoeg perfect zijn om middels mijn proces te kunnen migreren. Daarnaast is de sequentie van de dataoverdracht cruciaal om de referentiele integriteit van de dataoverdracht te waarborgen. Omdat Dynamics CRM Online referentiele integriteit afdwingt zal het records die daar niet aan voldoen weigeren. Het gevolg is een cascade aan falende records wat als worst-case scenario een rollback van de gehele dataoverdracht kan betekenen.

Ondanks grote impact van de risico's die het vervangen van complexiteit door sequentie heeft, blijf ik bij mijn keuze. De tijdwinst die behaald wordt is daarvoor te groot om te negeren. Wel leg ik een deel van de complexiteit nu neer bij de voorbereidingsfase door het vereisen van een perfecte scope en requirements. Desondanks geloof ik dat de voordelen opwegen tegen de risico's van deze manier van werken.



Dynamics CRM Online



Scribe Online



Scribe CRM Migration Assistant



☐ Modulair

⇒ Gebruikersmigratie

⇒ Automatisch Mappen

⇒ Rapportage

7.3 Scribe CRM Migration Assistant

De Scribe CRM Migration Assistant (SCMA) is een stuk maatwerk dat ik tijdens mijn project zelf gebouwd heb. Het is een Windows Form applicatie geschreven in de programmeertaal C#. De tool is gebouwd om tijdswinst te behalen via automatisering op plekken waar Scribe Online tekort schiet. Ik heb drie vlakken geïdentificeerd waar de tool de grootste bijdrage, in de vorm van tijdswinst, aan het migratieproces zal leveren. Dit zijn:

- Het migreren van gebruikers (niet mogelijk via Scribe Online);
- Het automatiseren van datamappings (kan Scribe Online wel maar, dit is beperkt);
- Rapportage (Scribe Online heeft geen rapportage mogelijkheden).

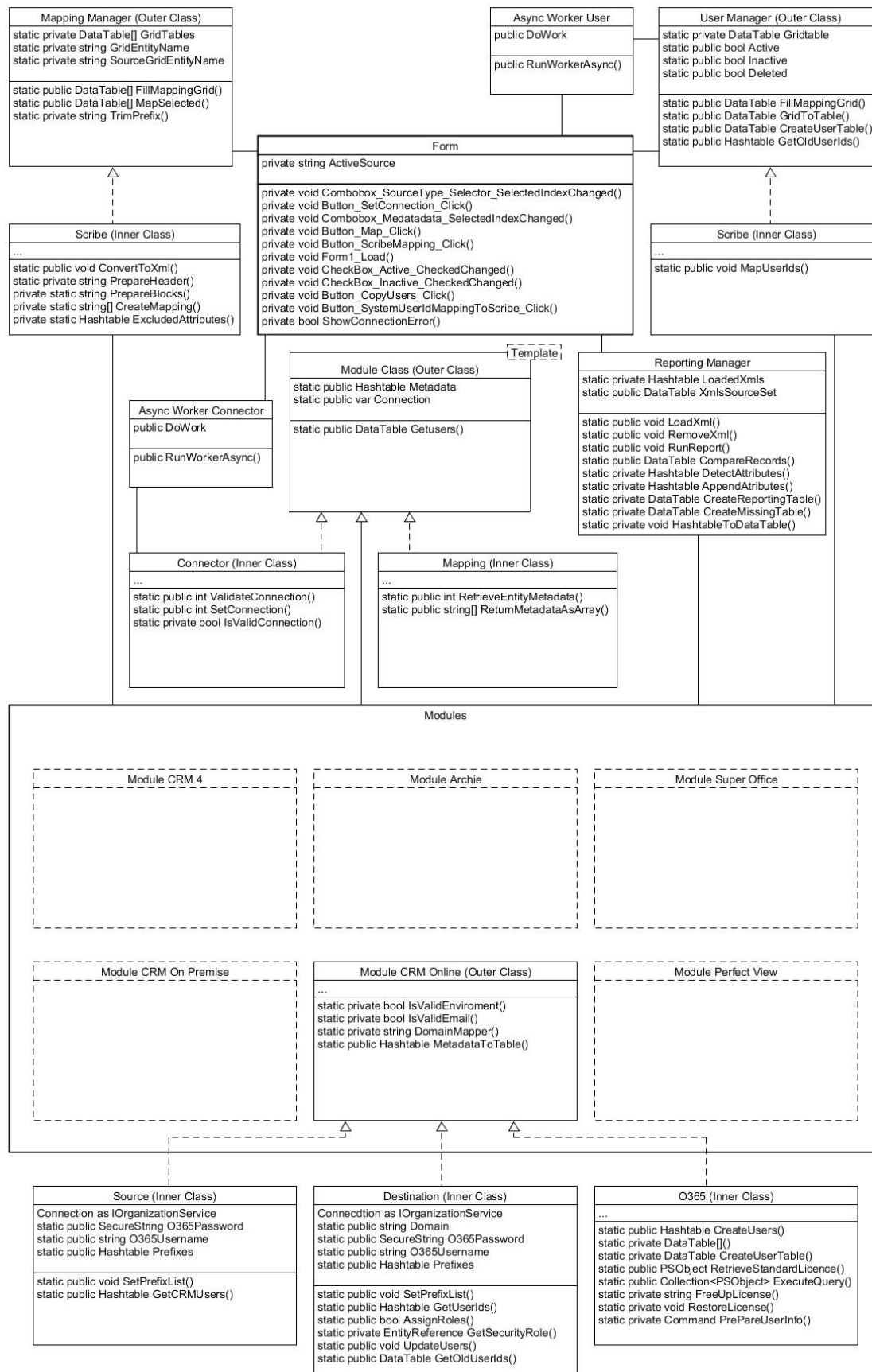
Ik heb gekozen voor een Windows Form applicatie omdat ik veel ervaring heb met het schrijven van Windows applicaties, hierdoor beperk ik de leercurve zodat ik meer tijd heb voor overige zaken. In een later stadium (na afronding van dit project) zou ik graag de tool, indien mogelijk, overzetten naar ASP.Net. Zo is de tool altijd beschikbaar in de Cloud, hetgeen aansluit op de bedrijfsvoering van pixelzebra. Tot die tijd ben ik voor de gulden middenweg gegaan door de applicatie portable te maken, zo kunnen pixelzebra's consultants de tool op de SharePoint omgeving vinden. Ik heb het stuk maatwerk in C# geschreven omdat deze taal de voorkeur heeft bij Microsoft en haar producten. Gezien het feit dat pixelzebra een Microsoft partner is, lijkt deze keuze mij evident.

In dit hoofdstuk geef ik uitleg over de werking van de tool op de bovengenoemde drie vlakken. Hierbij geef ik aan waarom ik specifiek voor dat vlak gekozen heb, hoe het werkt en welke verbeteringen er nog mogelijk zijn.

Modulair

Eén van de voorwaarden voor alle producten die ik ontwerp, ontwikkel en schrijf is dat ik rekening houd met een aantal verschillende bronomgevingen. In het geval van mijn maatwerktool heb ik dit opgelost door de klassenstructuur modulair te maken. Dit betekent dat de kleinste functionele component niet de applicatie zelf maar, een deel van de applicatie. Ik heb voor dit principe gekozen omdat deel een aantal voordelen heeft ten opzichte van traditioneel ontwikkelen. Zo kan ik tijdens mijn project een volledig functionele tool opleveren. De componenten voor andere bronomgevingen kunnen later ontwikkeld en toegevoegd worden. Het voordeel voor pixelzebra is dat zij de tool straks tegen verschillende prijzen aan de klant kan leveren door de tool aan te bieden met alleen de componenten die de klant nodig heeft.

Figuur 28 toont het Klassendiagram die als plattegrond voor de ontwikkeling van de tool gediend heeft.



Figuur 28: Klassendiagram Scribe CRM Migration Assistant

Gebruikersmigratie

Zoals ik heb beschreven in hoofdstuk 7.1 is het toevoegen van gebruikers aan Dynamics CRM Online een tijdrovende klus. Er zijn veel stappen nodig om de gebruiker correct te migreren en zelfs met een migratie waarbij het aantal gebruikers beperkt is kost het handmatig aanmaken en activeren van gebruikers veel tijd. In overleg met de productowner heb ik daarom besloten om het migreren van gebruikers zo veel mogelijk te automatiseren. Helaas is dit niet zomaar mogelijk door de tussenkomst van de Windows PowerShell: de interface voor communicatie met Office 365. Standaard migratiepakketten zoals Scribe Online en Informatica kunnen hier niet mee communiceren wat mij de keuze liet: of gebruikersmigratie handmatig opnemen in mijn migratieproces of een stuk maatwerk ontwikkelen om de gebruikersmigratie te automatiseren. Ik heb voor het laatste gekozen omdat dit aansluit op het doel van mijn project: tijdswinst.

Figuur 29 toont een view van het gebruikersmigratie onderdeel van de Scribe CRM Migration Assistant. De tool heeft de optie om actieve en/of inactieve gebruikers aan te vinken. Dit onderscheid heb ik bewust gemaakt. Het proces dat doorlopen wordt verschilt in beide gevallen en is afhankelijk van het aantal beschikbare licenties voor Dynamics CRM in de doelomgeving. het kan tijdswinst opleveren door eerst inactieve gebruikers te migreren en daarna de actieven. Andersom zou het aantal beschikbare licenties verminderen waardoor mijn tool minder inactieve gebruikers tegelijkertijd kan aanmaken.

The screenshot shows the 'User Migration' tab of the Scribe CRM Migration Assistant. It features a table with columns: DisplayName, FirstName, LastName, UserPrincipalName, Department, Country, State, and Status. The table lists various users, including Diederik Bos, Jan Groeneveld, Vacerra Lagas, Robert Lagas, Michel Schonis, Tim Willems, service account, Jeroen van den ..., Marco Waagmee..., Ivo de Witte, Ronald Hubert, Pixelzebra helpt, Admin PIXELZEB..., Debbie Rhebok, Seafra Moleveld, TenantRelocatio..., and Emek van Delden. Red annotations highlight specific UI elements: a box around the 'Active' and 'Inactive' checkboxes (1), a box around the 'Save As Scribe Lookup Table' button (2), a box around the 'Copy Users' button (3), and a box around a row of user data (4).

DisplayName	FirstName	LastName	UserPrincipalName	Department	Country	State	Status
Diederik Bos	Diederik	Bos	diederik.bos@pix...				Active
Jan Groeneveld	Jan	Groeneveld	Jan.Groeneveld...				Active
Vacerra Lagas	Vacerra	Lagas	vacerra.lagas@...				Active
Robert Lagas	Robert	Lagas	robert.lagas@pix...				Active
Michel Schonis			michel.schonis_fi...				Active(O365 Onl)
Tim Willems	Tim	Willems	Tim.Willems@pix...				Active
service account	service	account	service.account...				Active
Jeroen van den ...			jeroen.vandenber...				Active(O365 Onl)
Marco Waagmee...			waagmeester.mar...				Active(O365 Onl)
Ivo de Witte			witte78_hotmail.c...				Active(O365 Onl)
Ronald Hubert	Ronald	Hubert	ronald.hubert@pi...				Active
Pixelzebra helpt			help@pixelzebra.nl	Support			Active(O365 Onl)
Admin PIXELZEB...	Admin	PIXELZEBRA	admin@PixelZebr...				Active
Debbie Rhebok	Debbie	Rhebok	Debbie.Rhebok...				Active
Seafra Moleveld	Seafra	Moleveld	seafra.moleveld...				Active
TenantRelocatio...			TenantRelocatio...				Active(O365 Onl)
Emek van Delden	Emek	van Delden	emek.van.delden...				Active(O365 Onl)

Figuur 29: View van de Scribe CRM Migration Assistant Gebruikersmigratie

Voor het migreren van de gebruikers heb ik gekozen voor een intuïtieve methode waarbij er regels aangeklikt kunnen worden op dezelfde wijze zoals dit gebeurt in Microsoft Word. Mijn idee was dat ik op deze manier niet hoeft te knoeien met vinkjes of andere selectieonderdelen. Dit houdt de interface schoon.

Na het selecteren van gebruikers zijn er twee opties: de geselecteerde gebruikers opslaan als Lookup Table of de gebruikers kopiëren. De laatste optie activeert de migratie van de geselecteerde gebruikers. Ik heb dit proces geheel geautomatiseerd en door gebruik te maken van een asynchrone functie heb ik er voor gezorgd dat er gewoon verder gewerkt kan worden met de tool. Dit heb ik gedaan om, hoe weinig het ook is, nog wat extra tijd te winnen. Bij een synchrone functie zou er gewacht moeten worden tot de migratie klaar is.

De eerste optie levert een CSV bestand van de geselecteerde gebruikers waarin de Id's van de gebruikers vanuit beide omgevingen worden vastgelegd. Ik heb deze optie toegevoegd omdat dit CSV bestand ingelezen kan worden in Scribe Online. Mappings die een relatie hebben naar de Gebruiker entiteit (vrijwel alle mappings) kunnen vervolgens refereren naar het ingelezen CSV bestand. Dit zorgt ervoor dat er per gemigreerd record één call minder naar de doelomgeving gedaan kan worden, hetgeen een tijdswinst oplevert voor de dataoverdracht.

Hoewel de tool momenteel in standaard scenario's goed functioneert zijn er een aantal scenario's waarin er conflicten kunnen ontstaan. Gezien de beschikbare tijd en het gebrek aan toepasselijke testomgevingen heb ik deze scenario's momenteel achterwege gelaten. Zonder te diep in te gaan op de scenario's komt het er op neer dat er conflicten kunnen ontstaan met de UserPrincipalName. Dit is een verplicht unieke waarde waarmee gebruikers in de front end van Dynamics CRM geïdentificeerd worden. Er zijn Dynamics CRM omgevingen met meerdere domeinen. De Scribe CRM Migration Assistant gebruikt altijd het eerste domein dat het tegenkomt in een omgeving als default. In het geval dat er gebruikers zijn met een UserPrincipalName in twee domeinen zal dit een conflict opleveren.

Automatisch Mappen

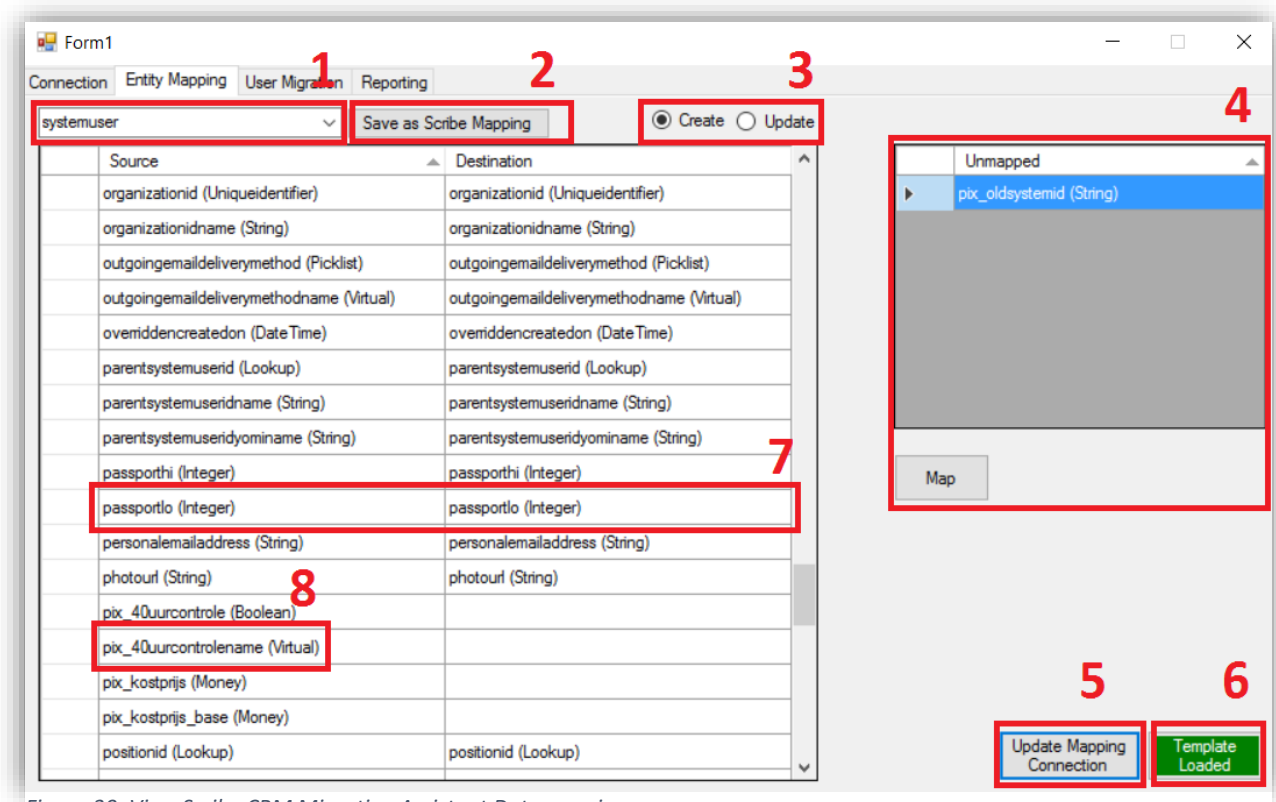
Het mappen van twee entiteiten kost veel tijd en is bovendien een monotone taak. Voor pixelzebra's migraties zullen de velden van een entiteit veelal overeenkomen in beide omgevingen. Scribe Online bezit zelf de mogelijkheid om velden met overeenkomende namen te mappen. De intelligentie van Scribe Online stopt echter bij het mappen van veldnamen die 100% overeenkomen; alles minder dan een gelijkenis van 100% wordt niet gemapped. Dynamics CRM Online heeft een eigenschap die automatisch het mappen via Scribe Online in de weg staat: de prefix.

Alles in Dynamics CRM Online wat niet in het standaardpakket aanwezig is (customization) heeft een prefix voor de naam, bijvoorbeeld "new_". De veldnaam "adres" zal dan "new_adres" heten. Het is niet ondenkbaar dat binnen een migratie entiteiten en/of velden gemapped moeten worden die verschillende prefixes hebben. Scribe Online kan hier geen onderscheid in maken wat betekent dat het mappingproces terugvalt naar een manueel proces. Dit zou betekenen dat ik flink moet inleveren in tijd binnen mijn migratieproces, iets wat ik absoluut niet wil.

Figuur 30 toont een voorbeeld van het mapping onderdeel van de Scribe CRM Migration Assistant. De tool kan net zoals Scribe Online automatisch velden mappen op basis van de naam. Wat de tool eveneens kan is velden mappen met verschillende of geen prefixes. Dit zorgt ervoor dat het mappingproces over het algemeen weer volledig geautomatiseerd is waarmee de te behalen tijdswinst binnen dit project voor dit onderdeel gemaximaliseerd is. Wat mij betreft is de keuze om dit onderdeel te ontwikkelen de inspanning dan ook meer dan waard.

De mappings die gemaakt worden met de Scribe CRM Migration Assistant kunnen geëxploreerd worden als Xml bestanden die kunnen worden ingelezen in Scribe Online. De reden dat ik er voor gekozen heb om de dataoverdracht niet via de SCMA te laten gaan (dit is iets sneller dan Scribe) is de mogelijke impact op

de data integriteit. De CRM-connector van Scribe Online is dermate goed dat ik met maatwerk niet zomaar hetzelfde niveau van data integriteit kan behalen.



Figuur 30: View Scribe CRM Migration Assistant Datamapping

Rechts onderin in figuur 30 zijn twee knoppen te zien. Deze knoppen helpen bij het updaten van connectie informatie van Scribe Online mappings. De enige manier om connecties te wijzigen in mappings is door de mappings handmatig aan te passen. Omdat er tot wel 300+ mappings per migratie zijn kan dit zo een dag tijd, zo niet meer, in beslag nemen. Ik wilde dit daarom graag automatiseren zodat ik deze tijd kon terugbrengen naar seconden in plaats van uren. De enige manier waarmee ik dit kon realiseren is door het zelf te maken. Er is niets op de markt dat een soortgelijk iets doet.

Zelfs in de XML bestanden, waaruit een Scribe Online mappings zijn opgebouwd, is het niet eenvoudig om connectie informatie te wijzigen. Er zijn veel Id's en namen die op ogenschijnlijk willekeurige plekken gewijzigd moeten worden. De Id's zijn eveneens niet eenvoudig te achterhalen. Ik heb een eenvoudige manier gedacht om dit op te lossen. Door een mapping in Scribe Online aan te maken de connectie informatie goed staat heb ik alle informatie in handen die ik nodig heb. Deze 'template' mapping en overige mappings kunnen vervolgens ingelezen worden in de Scribe CRM Migration Assistant. De tool wijzigt vervolgens de connectie informatie van alle mappings naar die van de template. Deze optie in de Scribe CRM Migration Assistant zorgt ervoor dat een dag werk nu letterlijk 5 seconden duurt. Dit één van de grootste tijdswinsten die ik behaald heb met mijn proces ten opzichte van de manuele werkwijze die pixelzebra voorheen hanteerde.

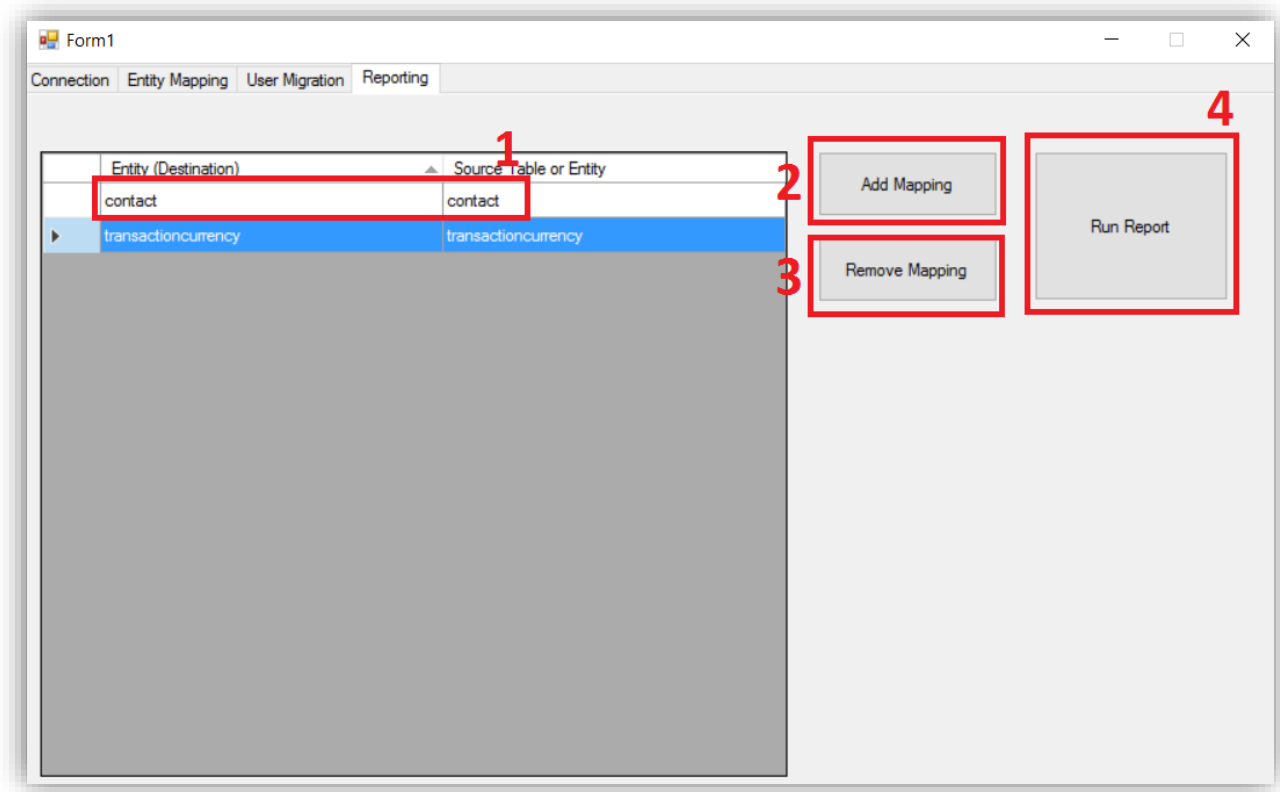
Hoewel ik enorme tijdswinsten heb behaald met de bovenstaande maatregelen is het mappingproces niet perfect. Velden met niet overkomende namen (los van prefixes) moeten nog altijd handmatig gemapped worden. Daarnaast herkent de SCMA momenteel customization op basis van de prefix. Er is echter één specifiek, maar onwaarschijnlijk, scenario waarin customization geen prefix heeft. Momenteel zal de tool

deze negeren. Voor het mappen van velden die qua naam niet gerelateerd zijn bestaan momenteel geen praktische oplossingen. Ik zie zelf mogelijkheden met gebruik van machine learning tezamen met een grote database van mappings. Dit zou echter een onderzoek op zichzelf zelf en past qua inspanning niet binnen de scope van mijn huidige project.

Rapportage

De laatste keuze die ik m.b.t. maatwerk heb gemaakt is het rapporteren van de migratieresultaten. Scribe Online heeft zelf geen mogelijkheid tot rapportage en voor het testen van de migratie in de Nazorgfase wil kunnen controleren of het aantal records in beide omgevingen overeenkomt. Hiermee kan ik valideren of alle data gemigreerd is. Binnen Dynamics CRM zelf is er ook geen optie om het aantal records specifiek te controleren. Deze telt namelijk tot 5000 en hogere aantallen worden eenvoudigweg weergegevens als 5000+. Om toch een mate van rapportering toe te kunnen voegen aan mijn migratieproces heb ik daarom gekozen om dit zelf te bouwen in de Scribe CRM Migration Assistant. Het enige alternatief voor het valideren van de overgezette data is een steekproef. In mijn migratieproces raad ik dit alsnog aan, maar alleen om inhoudelijk records op compleetheid te controleren. Ik vind de steekproef echter geen geschikte methode om te controleren of de absolute aantallen correct gemigreerd zijn.

Bij het ontwerpen van het rapportageonderdeel van de Scribe CRM Migration Assistant liep ik meteen tegen een praktisch probleem aan. De tool weet niet welke entiteiten gemigreerd zijn. Eén eerste optie die ik overwogen had om dit op te lossen was om de tool een query te laten uitvoeren op iedere entiteit in de doelomgeving. Het probleem met deze oplossing is dat de tool alleen niet weet aan welke entiteit in de bronomgeving nodig is voor de vergelijking. De oplossing waar ik uiteindelijk voor gekozen heb maakt gebruik van de mappings die toch al in het bezit zijn voorafgaand aan de migratie. De mappings bevatten



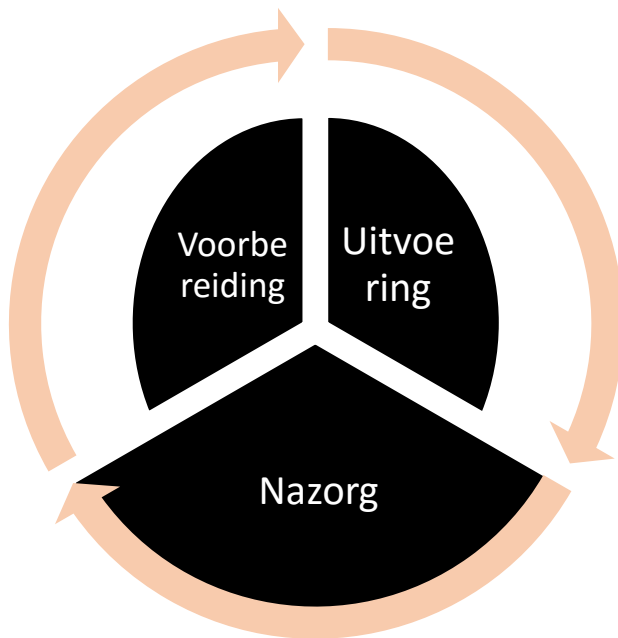
Figuur 31: View Scribe CRM Migration Assistant Reporting

de links tussen de bron- en doelenentiteiten. Dit is precies wat ik nodig heb om een overzicht van gemigreerde entiteiten te tonen.

Figuur 31 toont een voorbeeld van het rapportage onderdeel van de Scribe CRM Migration Assistant. Door op 'Add Mapping' te klikken kan er een Xml bestand ingelezen worden. Indien deze van het correcte format is checkt de tool alle mogelijke koppelingen tussen twee entiteiten in de mappings. Deze worden vervolgens in tabelvorm getoond, waarbij er tevens een optie is om ongewenste mappings weer weg te halen. Dit is de eenvoudigste wijze die ik heb kunnen vinden om alle benodigde informatie voor de rapportage te verzamelen. Door tot slot op de 'Run Report' knop te drukken geeft de SCMA een telling en vergelijking van alle gelinkte entiteiten.

Zoals met de eerdere componenten van de Scribe CRM Migration Assistant is ook het rapportage onderdeel nog niet af. Het scenario waarin er gewerkt wordt met een 'vuile' doelomgeving heeft ondanks de wijze waarop ik de rapportage opstel nog steeds gevolgen. Een telling zal met een vuile omgeving altijd vertekende resultaten geven omdat er al records aanwezig zijn. De enige manier om dit op te lossen is door gemigreerde record te merken. Dit is de enige wijze om een 100% correcte telling te maken. Dit heeft echter gevolgen voor de inrichting van de doelomgeving en de dataoverdracht. Omdat het corrigeren veel tijd kost en het tevens een onwaarschijnlijk scenario is heb ik deze verder buiten beschouwing gelaten in dit project.

Een laatste kanttekening voor het rapportage onderdeel is het gebrek aan automatisering voor de gebruikersacceptatietest. Hiermee doel ik op een steekproef die aantoont of alle benodigde data binnen een record is overgezet. Momenteel beschrijf ik dit als een manuele taak die samen met de klant uitgevoerd moet worden. In de toekomst zal dit met behulp van de SCMA geautomatiseerd worden. Om zo de tijdwinst nog verder te vergroten. Dit was door tijdgebrek niet mogelijk in dit project.



H8 Nazorg van de migratie

De nazorgfase staat in een standaard migratieproces in het teken van testen en decharge (Russom, 2006). Door mijn TELT-proces komt hier nog een potentiële transformatiefase bij. Tot slot houd ik in de fase rekening met een mogelijke nasleep van de migratie in de vorm van een trickle. In dit hoofdstuk wijd ik uit over de wijze waarop ik het testproces graag zie, welke rol mijn TELT-proces heeft binnen de nazorgfase en wat de gevolgen zijn van de onvermijdelijke decharge en trickle op mijn migratieproces. Opgesomd zal ik de volgende punten beschrijven in dit hoofdstuk:

- Het testen van de migratieresultaten;
- Mijn visie op de tweede transformatie uit mijn TELT-proces;
- De overige nazorgstappen: decharge en trickle.

Het testen gebeurt in mijn migratieproces op twee manieren. De eerste manier is een automatische kwantitatieve test van de records. Hierbij wordt gecontroleerd of de hoeveelheden records in de bron- en doelomgeving overeenkomen. De tweede manier is een gebruikersacceptatietest. Deze test is niet geautomatiseerd en dient uitgevoerd te worden met stakeholders. Om hier toch tijdswinst op te behalen heb ik voor de acceptatietest een template opgenomen in mijn migratiedraaiboek.

Na, of mogelijk voor, het testen kan het zijn dat er nog gegevens getransformeerd moeten worden. Afhankelijk van de belangrijkheid en urgentie kan deze transformatiestap uitgevoerd worden binnen het migratieproject of daarbuiten. Beide opties hebben voor- en nadelen die ik in dit hoofdstuk zal beschrijven.

De wijze waarop pixelzebra een dechargegesprek in gaat laat ik grotendeels over aan de consultant. Indien mijn migratieproces goed is uitgevoerd zouden er genoeg handvatten moeten zijn om een dergelijk gesprek door te komen. Ik zie daarom geen noodzaak om deze stap getailleerd in te richten. Wel beschrijf ik dit hoofdstuk een manier waarop pixelzebra het dechargegesprek een extra positieve wending kan geven voor haarzelf door het gesprek te gebruiken als input voor zelfreflectie.

De laatste stap van de nazorgfase is het opzetten en/of beheren van een trickle. Een trickle is niet meer dat het laten lopen van (delen van) de migratie om de doelomgeving volledig te maken.



Testen



Transformatie na migratie



Nazorg



⇒ Kwantiteit

⇒ kwaliteit

8.1 Testen

Het testen van de doelomgeving is nodig om te controleren of de gehele migratie correct is uitgevoerd. Om zoveel mogelijk tijdswinst te behalen heb ik geprobeerd om binnen de tijdspanne van mijn project zoveel mogelijk van dit proces te automatiseren. Dit heeft een tweedeling van het testproces als gevolg gehad met aan de ene kant een test op de kwantiteit van de data en aan de andere kant een test op de kwaliteit van de data. De kwalitatieve test zal uitgevoerd worden middels een gebruikersacceptatietest, de kwantitatieve test met een eenvoudige vergelijkingstest.

Kwaliteit

De kwaliteit van de gegevens zal getest worden middels een gebruikersacceptatietest. Ik heb gekozen voor een type test waarbij de gebruiker betrokken wordt omdat de eindgebruiker het beste kan controleren of alles op operationeel niveau klopt (TDWI, 2012), hij/zij is immers de persoon die altijd met de data en inrichting zal gaan werken.

Pixelzebra gebruikt in haar standaard projectproces een gebruikersacceptatietest. Deze test heb ik grotendeels overgenomen in mijn migratieproces. Wat pixelzebra doet is het opstellen van een aantal testcasussen op basis van de gemigreerde gegevens. De test moet vervolgens zonder bijzijn van pixelzebra's consultant uitgevoerd worden. Dit zorgt ervoor dat de eindgebruiker niet beïnvloed wordt door zijn of haar aanwezigheid.

Testcasus: Contactpersoon			
Primair formulier:			
Veld:	Afwezig	Aanwezig maar leeg	Aanwezig en gevuld
Naam			
Achternaam*			
Tussenvoegsel			
Postcode			
Woonplaats			
Email*			
Bedrijfsnaam+			
Gedrag:	Nee, er gebeurt niets	Nee, niet het verwachte resultaat	Ja
Als de postcode wordt gewijzigd naar "1000 HD", dan wijzigt de woonplaats automatisch naar "Amsterdam" op voorwaarde dat het formulier is opgeslagen.			
Quick Create formulier:			
Veld:	Afwezig	Aanwezig maar leeg	Aanwezig en gevuld
Achternaam*			
Email*			
bedrijfsnaam			
(* = Verplicht ingevuld in CRM, + = Gewenst ingevuld in CRM)			

Figuur 32: Voorbeeld van een opgemaakte gebruikersacceptatietest template

De opzet van de gebruikersacceptatietest is om de klant een reeks aan acceptatietesten te laten uitvoeren, zo kan steekproefsgewijs de kwaliteit van de data bewezen worden. Het is aan pixelzebra om te zorgen dat de testcasussen van te voren zijn besproken en zijn vastgelegd in de requirements en scope van het migratieproject. Zo is het verloop van het project tot en met het testen vastgelegd en transparant voor iedereen. Doet pixelzebra dit niet dan is mijn angst dat de klant probeert af te zien van het testen. Dit kan zijn om proberen kosten te besparen of omdat medewerkers er gewoonweg geen zin in hebben. Welke achterliggende reden dan ook, het gevolg is dat pixelzebra de data integriteit niet kan valideren.

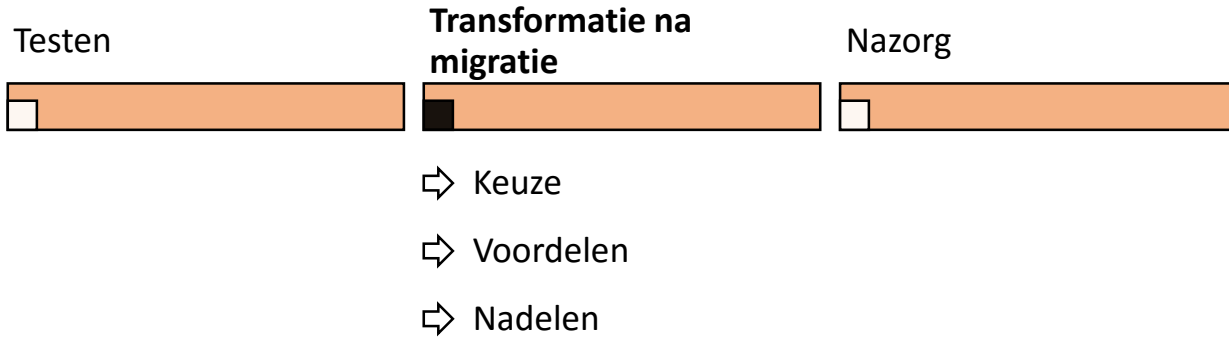
Voor het opstellen van een testcasus zal de template uit mijn draaiboek leidend zijn. Op basis van de requirements kan pixelzebra vervolgens invullen geven aan de template. Zo kan zij de velden van een formulier toevoegen ter controle, meerdere formulieren toevoegen en gedrag op het formulier toevoegen dat veroorzaakt wordt door workflows. Figuur 31 toont een eenvoudig voorbeeld van een testcasus zoals deze opgesteld zal worden binnen mijn migratieproces. De casus is hierbij de contactpersoon entiteit.

kwantiteit

De kwantiteit, oftewel volledigheid, van de data in de doelomgeving is bepalend voor de integriteit van de gehele Dynamics CRM Online omgeving. Zonder deze volledigheid kan de historie van de data niet bewaakt worden. De historie is van belang is voor alle bedrijfsprocessen die CRM gebruiken. Ik test de volledigheid van de data middels een telling van alle records van een entiteit. Vervolgens maak ik een vergelijking tussen het aantal records in de bron- en doelomgeving. Als deze overeenkomen klopt de

kwantiteit voor de entiteit. Dit is een uiterst eenvoudige maar effectieve methode om de volledigheid en succes van de migratie te valideren (Katzoff, 2009), daarnaast kan deze methode vrij eenvoudig geautomatiseerd worden, hetgeen weer tijdwinst voor mijn migratieproces oplevert.

Het testen van de kwantiteit heb ik middels de Scribe CRM Migration Assistant geautomatiseerd. Pixelzebra kan deze test uitvoeren met vrijwel één druk op de knop. Deze test behoeft tevens geen aanwezigheid van een gebruiker of andere stakeholder. In tegenstelling tot het kwalitatief testen is deze test snel uitgevoerd en er is geen ruimte voor persoonlijke voorkeuren. Bij kwalitatieve testen kan een gebruiker bijvoorbeeld aangeven toch graag velden op een andere plek in een formulier te zien. De kwantitatieve test controleert louter feitelijke data. Een gebruiker zal nooit aangeven dat hij liever 9999 van de 10000 records gemigreerd had willen zien omdat hij dat een prettiger getal vindt. De verwachte resultaten voor de kwantitatieve test staan voorafgaand aan de test vast en zijn onveranderlijk.



8.2 Transformatie na migratie

In hoofdstuk 6.4 beschrijf ik hoe ik het standaard ETL-proces wil veranderen naar mijn eigen ontworpen TELT-proces. Daar waar de eerste transformatiestap uitgevoerd wordt in de voorbereidingsfase van het migratieproces wordt de tweede transformatiestap uitgevoerd na afloop van de dataoverdracht in, of na, de nazorgfase. Dit hangt van de prioritering van de requirements en de wensen van de klant.

De eerste mogelijkheid om te transformeren is binnen het migratieproject. In dit geval wordt de transformatie meteen na de dataoverdracht uitgevoerd. Om deze manier worden alle aspecten van de migratie binnen één project uitgevoerd en is de doelomgeving na afloop van het migratieproject volledig klaar voor gebruik.

De tweede mogelijkheid is om de tweede transformatie op te nemen in een apart project dat losstaat van de het migratieproject. De tweede optie heeft als voordeel dat gebruikers de tijd krijgen om in de nieuwe omgeving te werken en additionele eisen toe te voegen. Zo kunnen bepaalde transformaties overbodig blijken of kunnen er extra transformaties toegevoegd worden. De reden waarom ik de tweede optie voor transformaties heb toegevoegd is omdat dit aansluit op de filosofie van pixelzebra. Deze stelt dat gebruikers klein moeten beginnen en dan stapsgewijs extra functionaliteit moeten toevoegen. Uit pixelzebra's ervaring blijkt dat dit de adoptie van Dynamics CRM Online drastisch verhoogd. De eerste transformatie optie is daarentegen eveneens nodig voor klanten die pers sé de inrichting en migratie binnen één project afgerond willen hebben.

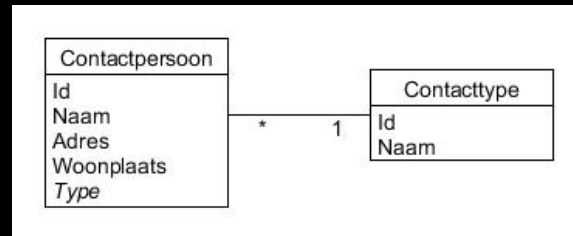
Keuze

Bij uitvoering van mijn migratieproces verwacht ik dat de klant bewust een keuze maakt voor transformeren in de nazorgfase of na de afronding van het migratieproject. In hoofdstuk 6.3 heb ik een prioriteringshulpmiddel beschreven die zal helpen om de klant inzicht te geven in wat zij graag zou willen en wat zij echt nodig hebben. De realiteit is dat veel migraties falen door teveel complexiteit (Moss, 2000) of door gebrek aan adoptie (Russom, 2006). Door te assisteren bij het identificeren wat de klant echt nodig heeft hoop ik de adoptie soepeler verloopt.



Voorbeeld Casus:

Organisatie X heeft in de bronomgeving in de database een tabel genaamd contacttype. In de praktijk bevat de contacttype tabel slechts twee waarden: klant of leverancier. De klant vindt dit een overbodige toevoeging aan het databasemodel en wil de twee typen graag rechtstreeks op de contactpersoon kunnen invullen zodat de contacttype verwijderd kan worden uit het model.



Wat blijkt nu. Drie maanden na de migratie wordt het nieuwe CRM-systeem wordt dusdanig goed gebruikt dat organisatie X Business Intelligence gegevens uit Dynamics CRM Online wil gaan halen. Om dit te doen wil zij onder andere gegevens m.b.t. de klant specifiek gaan bijhouden. Eén van de onderdelen die specifiek moet is het Contacttype. In plaats van bij te houden of de contactpersoon bij een klant of een leverancier hoort wil graag de organisatie graag bijhouden wat de specifieke functie van de contactpersoon is. Bijvoorbeeld een Salesmedewerker of een directeur. Dit betekent dat de lijst met mogelijke contacttypen groter zal worden en continu uitgebreid zal worden met nieuwe functies.

Transformatie binnen het migratieproject	Transformatie in een apart project
Stappen voor de uitvoeringsfase: De consultant heeft de databasestructuur vanuit de bronomgeving overgezet naar de doelomgeving. De consultant heeft de contacttype records gemigreerd naar de doelomgeving.	Stappen voor de uitvoeringsfase: De consultant heeft de databasestructuur vanuit de bronomgeving overgezet naar de doelomgeving. De consultant heeft de contacttype records gemigreerd naar de doelomgeving.
Stappen na de uitvoeringsfase (nazorgfase): De consultant maakt een nieuw veld aan op de contactpersoon met de naam "Contacttype" en maakt dit veld van het type "Optionset" met de waarden "Klant" en "Leverancier". De consultant verwijdert de Contacttype tabel inclusief records uit de doelomgeving. Hiermee wordt tevens het veld "Type" op de Contactpersoon verwijderd.	Stappen na de uitvoeringsfase (nazorgfase): De consultant adviseert om eerst een aantal maanden te werken volgens de inrichting van de bronomgeving om daarna gewenste wijzigingen in het systeem aan te brengen.
Stappen na afronding migratieproject: De support afdeling krijgt continu aanvragen binnen om nieuwe functies toe te voegen aan de Contacttype Optionset. De grootte van de Optionset begint inmiddels de performance van het formulier te beïnvloeden.	Stappen na afronding migratieproject: De klant geeft aan dat de transformatie voor het contacttype niet meer uitgevoerd hoeft te worden. De lijst met contacttypes wordt constante uitgebreid en binnen Dynamics CRM is de huidige werkwijze de meeste efficiënte.

Figuur 33: Een vergelijking tussen transformatie binnen de migratie en transformatie na de migratie.

Voordelen

Kijkend naar figuur 33 lijkt ik te impliceren dat transformatie in een apart project superieur is aan transformatie binnen het migratieproject. Er zijn scenario's, zoals in de voorbeeld casus, waarbij dit het geval is. Beide transformatieopties hebben echter hun voordelen. Om de juiste optie te kiezen (beide kan uiteraard ook) is het, naast het identificeren van de klantbehoeften, belangrijk om de kijken naar de impact van potentiële issues die zich in de nazorgfase kunnen voordoen (Adelman & Bischoff, 2002). In dit hoofdstuk geef ik inzicht in voordelen die transformatiefasen hebben, zowel in het algemeen als ten opzichte van elkaar.

Migreren binnen het migratieproject heeft voornamelijk voordelen voor pixelzebra zelf. Als de klant erop staat dat de transformaties worden uitgevoerd binnen het project zal pixelzebra eenvoudigweg meer verdienen aan het project. Gezien de meeste klanten van pixelzebra, is de kans eveneens groot dat de klant meteen een support contract afsluit met pixelzebra, hetgeen ook geld oplevert. Het grootste voordeel ten opzichte van transformeren na de migratie is het feit dat er sneller geld verdient wordt aan het project.

Transformeren na afloop van het migratieproject heeft voornamelijk voordelen voor de klant. De omgeving zal vertrouwder aanvoelen omdat deze zo identiek mogelijk is gemaakt aan de bronomgeving. Dit zal de adoptie van Dynamics CRM Online verhogen. Door geleidelijk kleine projecten af te nemen bij pixelzebra krijgt de klant meer tijd om de gevolgen van alle veranderingen te ondervinden. Dit helpt om te bepalen wat wel en niet werkt voor de organisatie. Pixelzebra is zeer ervaren in het geleidelijk uitbreiden van Dynamics CRM Online en gelooft dat dit leidt tot een succesvolle implementatie. Hiermee komt meteen het voordeel voor pixelzebra aan het licht. Het transformeren na de migratie sluit perfect aan op pixelzebra's visie van incrementeel implementeren en tegelijkertijd relaties opbouwen met de klant.

Nadelen

Zoals beide transformatieopties voordelen hebben, hebben beide nadelen. Deze zijn in principe het tegenovergestelde van de voordelen. Zo wordt er sneller geld verdient met het transformeren in het migratieproject zelf maar, er wordt minder aandacht aan de adoptie besteed. Voor het transformeren na de migratie geldt juist dat er veel aandacht aan de adoptie en relaties wordt besteed maar, dit hoeft niet noodzakelijkerwijze een extra project op te leveren. Om meer inzicht te geven in mijn TELT-proces zal ik in dit hoofdstuk juist dieper ingaan op de nadelen van de transformatieopties in het algemeen.

Gebruik maken van mijn TELT-proces betekent lokaal transformeren, hierdoor ontstaat er onvermijdelijke vervuiling van de doelomgeving. Dit tast de datakwaliteit van operationele gegevens niet direct aan maar, zeker met de beperkte opslagruimte van Dynamics CRM Online kan dit problemen veroorzaken voor de klant en verwarring veroorzaken voor individuele gebruikers (Missi, Alshawi, & Fitzgerald, 2005). Het kan voorkomen dat er gegevens verwijderd moeten worden uit de doelomgeving (zie casus figuur 32). Op zich geen probleem, data deleten gebeurt overal. In een CRM-omgeving is het permanent verwijderen van gegevens uit den boze. Dit komt omdat het behoud van records nodig is om datahistorie, en daarmee de kwaliteit van de data, gezond te houden (Williamson K. , 2009).



Testen



Transformatie na
migratie



Nazorg



⇒ Decharge

⇒ Trickle

8.3 Nazorg

Naast het testen en transformeren van gegevens in de doelomgeving beschrijf ik nog twee stappen waarvan ik vind dat die uitgevoerd moeten worden mijn migratieproces om deze succesvol te maken. Het gaat hier bij om de decharge van het migratieproject en de trickle die zich vrijwel onvermijdelijk voordoet na de initiële dataoverdracht.

Het verkrijgen van decharge is een kritiek moment tijdens de migratie omdat dit de laatste stap van het project en tevens het moment waarin het project officieel wordt afgerond. De dechargestap zal tevens een moment van reflectie zijn voor pixelzebra, tijdens de decharge worden positieve en negatieve aspecten van het project nogmaals kort besproken (Grit, 2015). Voor pixelzebra betekent levert dit waardevolle informatie voor het optimaliseren van het migratieproces.

De afronding van een migratieproject betekent niet noodzakelijkerwijze het einde van het gehele migratieproces. In de tijd dat de dataoverdracht is uitgevoerd kan er nog data toegevoegd zijn aan de bronomgeving. Een trickle kan een lange tijd duren. Het opzetten ervan kan daarom mogelijk nog vallen binnen het migratieproject. Het controlerende aspect zal echter nog doorgaan na afronding van de migratie. De vraag hierbij is hoe pixelzebra dit moet beheren.

Decharge

Hoewel ik de decharge een uiterst belangrijke stap binnen mijn migratieproject vind, heb ik er niet bijzonder veel aandacht aan besteed. Ik geloof dat de decharge vanzelf komt als pixelzebra de eerdere stappen uit mijn migratieproces correct volgt. Op dit moment heeft pixelzebra genoeg middelen in handen om zelfs een niet-willende klant te overtuigen tot dechargeverlening. In gevallen waarin pixelzebra haar afspraken zelf niet kan nakomen, bijvoorbeeld een afgesproken opleverdatum van de doelomgeving, heb ik het volste vertrouwen de verantwoordelijken binnen de organisatie hier een gepaste oplossing voor vinden. Zij zijn hier ervaren genoeg voor.

Het enige advies dat ik pixelzebra meegeef tijdens deze stap is om een de klant aandachtig uit te horen over alle positieve en negatieve punten rondom het project. Hoewel ik zelf na afloop van dit project invloed kan blijven uit oefenen op de verbetering van mijn migratieproces, verwacht ik dat de iedereen binnen pixelzebra meewerkt om het migratieproces incrementeel te verbeteren. Het uithoren van de klant tijdens een dechargegesprek is een goed voorbeeld van. Gezien de adaptieve en leergierige aard van pixelzebra's medewerkers, gecombineerd met de drang om de organisatie continu te blijven verbeteren, is mijn verwachting dat pixelzebra dit principe snel zal adopteren.

Trickle

Er zullen migratieprojecten zijn waarbij het nodig is om na decharge van het migratieproject te blijven migreren. Binnen de tijdspanne van de migratie kan er immers nieuwe data gegenereerd zijn in de bronomgeving; te laat voor de initiële migratie. Voor het verkrijgen van de decharge zal er in deze gevallen een proces opgezet moeten worden om deze data alsnog over te zetten. Bij een keuze tussen een big bang aanpak of een trickle aanpak is in dit geval de trickle het meest geschikt (Russom, 2006).

Mijn persoonlijke reden voor het kiezen van de trickle aanpak is de geringe aandacht die na inrichting nodig is vanuit pixelzebra. Met name bij scenario's waarbij de klant besluit om de bron- en doelomgeving tijdelijk parallel te laten draaien komt dit tot zijn recht. De consultant zijn met een big bang aanpak constant handmatig moeten migreren

The screenshot shows the Scribe Online configuration interface. On the left, under 'On Demand', the 'Recurring' option is selected. A red box highlights the 'Every week on' section, which includes checkboxes for days of the week and a calendar grid. On the right, the 'Label' is 'Query contracttemplate', 'Connection' is 'Pixelzebra', 'Connector' is 'Microsoft Dynamics CRM', and 'Entity' is 'contracttemplate'. A red box highlights the 'Process only records created or updated since last run based on the value of the last modified field' option, with a dropdown menu showing 'modifiedon' and 'Mostrecentrecord'.

Figuur 34: Instellingen voor Trickle migraties in Scribe online

Scribe Online komt goed tot zijn recht tijdens de trickle. De mappings voor de migratie kunnen gelijk blijven en moeten slechts twee instellingen veranderd worden om de trickle tot in het oneindige te kunnen laten draaien. Figuur 34 toont een view van de twee instellingen die ik pixelzebra aanraad om aan te passen voor de trickle. De linkerkant toont een instelling om een migreren herhaaldelijk uit te voeren. Door dit voor iedere nacht in te stellen hebben gebruikers geen last van de dataoverdracht. Voor zeer kritische klanten is er echter een optie om de recursie terug te brengen naar seconden. Dit is uiteraard niet zonder mogelijke gevolgen voor de performance.

De rechterkant van de afbeelding toont de tweede instelling die ik aanraad. Deze instelling filtert de migratie op alleen die records die aangemaakt zijn sinds de laatste keer dat er gemigreerd is. zo wordt de werklust van de migratie exponentieel verlaagd. Zonder deze instelling zou de trickle niet mogelijk zijn omdat de migratie te lang zou duren voor één nacht.

De laatste vraag die speelt bij een trickle na de migratie is wie de verantwoording heeft. Ligt dit bij de klant, ligt dit bij consultant die de migratie heeft uitgevoerd of ligt dit ergens anders. Wat mij betreft zal dit vallen onder de support afdeling van pixelzebra. Dit geeft pixelzebra een handvat om een supportcontract te verkopen aan de klant en het onderhouden van een trickle is iets wat pixelzebra's support afdeling prima kan. Dit lijkt mij dus eenvoudigste optie die het meeste geld oplevert voor pixelzebra.

H9 Product evaluatie

De product evaluatie is het eerste onderdeel van mijn reflectie met betrekking tot mijn afstudeerproject. In dit hoofdstuk zal ik mijn mening geven over de producten die ik heb opgeleverd tijdens mijn project. Per product zal ik een oordeel geven over de kwaliteit en stel ik ter discussie wat ik denk dat gedaan kan worden in de toekomst om de kwaliteit naar een hoger niveau te tillen. Tijdens mijn afstudeerproject heb ik vier producten opgeleverd die ik in volgorde van oplevering binnen mijn project zal bespreken. Dit zijn: Een literatuuronderzoek naar de vereisten voor een goede migratie, een lijst met mappings voor Scribe Online, een stuk maatwerk ter verbetering van mijn ontworpen migratieproces en een draaiboek voor de consultants om inzicht te krijgen in mijn migratieproces.

9.1 Literatuuronderzoek

Het eerste product dat ik heb opgeleverd tijdens mijn afstudeerproject was een literatuuronderzoek waarbij ik op zoek ben gegaan naar de vereisten voor een goed migratieproces dat pixelzebra kan inzetten bij migraties voor klanten. Het uiteindelijke onderzoeksrapport dat ik heb opgeleverd was niet de eerste versie van het onderzoek. Ik heb het onderzoek in eerste instantie zeer praktisch uitgevoerd. De resultaten had ik rechtstreeks opgenomen in dit afstudeerverslag. Ik dacht dat het onderzoek geen grote invloed had op mijn project en dat ik via deze wijze veel tijd kon besparen. Het resultaat van deze eerste versie was een eenvoudige opsomming van do's en dont's voor een migratieproces. Hoewel informatie in de opsomming valide leek, had ik geen proces gevolgd om de kwaliteit van de informatie te controleren. Daarnaast had ik de resultaten niet gerelateerd aan pixelzebra. Ik heb nog voor het opleveren van deze eerste versie van het literatuuronderzoek besloten om het opnieuw te doen, de kwaliteit was eenvoudigweg te laag.

De tweede, definitieve, versie van mijn verslag is van beduidend betere kwaliteit. Ik heb het onderzoek grondig en structureel uitgevoerd. Zo heb ik een duidelijk plan opgesteld voor het verkrijgen van zoektermen, het verzamelen van bronnen en het beoordelen van de zoekresultaten. Dit is terug te zien in de kwaliteit van de bronnen en de conclusies die ik daaruit trek. Het literatuuronderzoek heeft mij geholpen bij het maken van keuzes tot en met het schrijven van dit verslag. Ik ben daarom uiterst tevreden over de kwaliteit en het resultaat van mijn literatuuronderzoek.

9.2 Scribe Online

Binnen Scribe Online heb ik een lijst van 150 mappings opgeleverd. Het maken van de mappings lijkt in eerste instantie eenvoudig maar, rekening houdend met alle eigenschappen van Dynamics CRM Online die het mappen exponentieel moeilijker maken (Zie H7 voor toelichting) vind ik dit zelf een geslaagd product. De wijze waarop de mappings zijn opgesteld haalt dusdanig veel complexiteit weg (deze verspreid ik over meerdere mappings) dat de snelheid waarmee gemigreerd wordt minimaal verviervoudigd ten opzichte van een 'normale' migratie.

De mappings die ik heb opgeleverd zijn goed maar, niet perfect. Er zijn waarschijnlijk nog een aantal 'tweaks' mogelijk om de migratie nog sneller te laten verlopen. Bijvoorbeeld door het aantal connectiemomenten met omgevingen verder te beperken door middel van interne Lookup tabellen in Scribe. Dit leermoment kwam echter te laat in mijn afstudeerproject om dit nog te kunnen veranderen. In een volgende iteratie zal ik dit gaan testen en bij succes opnemen in het migratieproces.

9.3 Scribe CRM Migration Assistant

Om een goed oordeel te geven over de maatwerk tool die ik gebouwd heb tijdens vind ik het nodig om de drie componenten: Gebruikersmigratie, Entity Mapping en Reporting apart te behandelen. Over de algemene kwaliteit van de tool ben ik aardig te spreken. Voor het bouwen van de tool heb ik anderhalf jaar aan ervaring met Dynamics CRM moeten benutten en heb ik flink wat nieuwe dingen bij moeten leren. Het feit dat ik in die korte tijd een werkende tool heb kunnen bouwen die kan wat de Scribe CRM Migration Assistant kan vind ik zelf indrukwekkend.

Gebruikersmigratie

Het onderdeel gebruikersmigratie werkt op zich goed. Ik heb alle stappen voor het migreren van gebruikers in zowel Office 365 als Dynamics CRM Online kunnen automatiseren. Alle, niet eenvoudige, problemen die ik daarbij ben tegengekomen heb ik op kunnen lossen. Zoals ik heb beschreven in hoofdstuk 8 zijn er wel nog een aantal scenario's die nog niet ondersteund worden. Verder is dit onderdeel van de tool operationeel. Ondanks dat ik meerdere tests succesvol heb uitgevoerd vind ik niet dat dit onderdeel al gebruiksklaar is voor een live migratie. Er moeten nog toevoegingen gedaan worden omtrent defensief programmeren zodat mogelijke excepties correct worden afgevangen. Daarnaast zijn er nog twee statische waarden (toevoeging aan gebruikersnaam en de licentie) die nog dynamisch gemaakt moeten worden. Zodra de bovenstaande drie aanpassingen zijn gedaan is dit onderdeel naar mening kwalitatief klaar voor inzet bij klanten. De reden dat ik deze aanpassingen niet heb gedaan binnen mijn project komt eenvoudigweg neer op tijdsgebrek. De functionele aspecten zijn daarom voor gegaan op de beveiliging.

Entity Mapping

Het mappen van entiteiten heeft met de Scribe CRM Migration Assistant is echt een merkbare verbetering ten opzichte van het auto mappen met Scribe Online. Dankzij het algoritme dat ik geschreven heb voor het 'negeren' van prefixes kan de tool mappen wat Scribe Online niet kan. Wat de tool momenteel niet kan is schijnbaar niet-gerelateerde velden mappen. Om dit (deels) mogelijk te maken is een zeer complexe klus en paste eenvoudigweg niet binnen mijn scope als het überhaupt al mogelijk is.

Reporting

Ik heb het rapportage onderdeel van de tool opgesteld met als doel om het aantal records van twee entiteiten te kunnen vergelijken. Mijn voorkeur ging uit om de tool zelf alle entiteiten in de doelomgeving af te laten gaan en deze te vergelijken met zijn tegenhanger in de bronomgeving. Dit zou mogelijk werken in een migratie tussen twee Dynamics CRM omgevingen maar, bij uitbreiding van de tool naar andere omgevingen zullen de namen waarschijnlijk niet overeenkomen waardoor de tool niet meer zou werken. Een volledige automatisering was daarom niet mogelijk. Ik ben wel tevreden over de manier waarop ik dit uiteindelijk heb opgelost. Dit vraagt slechts één handeling van de gebruiker en komt daarmee zeer dichtbij volledige optimalisatie. Wat de tool nog niet kan maar, zoals bij de andere onderdelen in de toekomst toegevoegd zal worden is test die steekproefsgewijs de kwaliteit van de records controleert. Dit vergt een meer tijd dan ik had in mijn project. Omdat dit eigenlijk een onderdeel is van de user acceptance test kan dit nooit volledig geautomatiseerd worden. Er zal altijd een proces blijven waarbij de input van de gebruiker nodig is. Dit onderdeel zou minder tijdswinst opleveren dan het vergelijken van de records, waardoor mijn keuze uitging van de vergelijking en niet de steekproef.

9.4 Draaiboek

Het draaiboek is een gulden middenweg geworden tussen het voorschrijven wat de consultant moet doen om succesvol te migreren en het bieden van ruimte om de consultant zelf te laten bepalen welke aanpak het beste werkt. Zo laat ik het dechargemoment binnen een migratieproces over aan de consultant maar schrijf ik wel voor hoe de consultant de dataoverdracht moet uitvoeren. Ik denk dat deze combinatie uiteindelijk het beste haalt uit de meeste tijdwinst oplevert voor mijn migratieproces. Het schrijft voor wat er moet gebeuren op plekken waar dit significante tijdwinsten oplevert en laat ruimte waar pixelzebra zelf ter plekke de best oplossing kan bedenken. Op basis van deze punten vind ik het draaiboek een geslaagd product. Het brengt op praktische wijze de filosofie van mijn migratieproces over.

Het enige discussiepunt voor het draaiboek is mogelijk de leesbaarheid. Ik heb het draaiboek zeer praktisch opgesteld en geschreven met als doelgroep: de consultants. Voor nieuwe medewerkers die niet veel of geen ervaring hebben met Dynamics CRM en/of Scribe Online zullen het draaiboek mogelijk moeilijk leesbaar vinden. Daarentegen geloof ik dat zij het boek prima zullen begrijpen tegen de tijd dat zij mee mogen draaien met een migratie.

H10 Proces evaluatie

Om aan te tonen dat ik HBO- en afstudeerwaardig werk heb uitgevoerd is het belangrijk om inzicht te geven in de wijze waarop mijn afstudeerproces heb doorlopen. In dit hoofdstuk zal ik middels een evaluatie van het proces en middels de STARR-methode inzicht geven in mijn gedachteproces tijdens mijn afstuderen. Om dit te doen zal ik opvallende situaties beschrijven die kenmerkend waren voor mijn stage en waarin ik nogmaals duidelijk laat zien op welke wijze ik belangrijke momenten tijdens mijn afstuderen heb aangepakt.

Integer managen

Tijdens de laatste retrospective met Jan Groeneveld heb ik hem dit verslag voorgelegd. Samen hebben wij zo terug gekeken om de manier waarop ik dit project heb uitgevoerd. Jan was trots op de manier waarop ik de Definitions of Ready en Done heb toegepast en hij zij dat hij dit terug zag in de kwaliteit van mijn werk. Hij had echter één vraag aan waardoor ik nogal schrok. Hij vroeg zich af hoe de ik de acceptatie criteria voor de items had gecontroleerd. Ik snapte de vraag niet zo en legde uit dat ik de lijst ben afgegaan en gecontroleerd heb of de item hieraan voldeed. De acceptatie criteria zijn eenvoudig te begrijpen dus dit is niet moeilijk. Op dat moment vroeg hij of ik hem de afgestreepte lijst kon laten zien. Dit kon ik niet en op dat moment realiseerde ik mij wat Jan bedoelde. Ik heb nergens de resultaten van het controleren van de acceptatie criteria bijgehouden. Ik kan dus nooit naar de product owner toe bewijzen dat ik een item naar alle volledigheid heb afgerond. De productowner heeft gedurende het project gevalideerd met het vertrouwen dat de items volledig voldeden aan de acceptatie criteria. Omdat ik het enige lid was van de projectgroep en de communicatielijn met de productowner kort was heeft dit mijn project niet ernstig beïnvloed. In grotere projecten zal een dergelijk gebrek de integriteit van project echter ernstig schaden. In toekomstige Scrum projecten zal ik dan ook bij de voorwaardelijke documentatie voor de Definition of Done een ingevuld testdocument voor de acceptatie criteria opnemen. Dit zal de integriteit van het project waarborgen.

Literatuuronderzoek

Vroeg in mijn afstudeerproject heb ik in overleg met de productowner besloten om een literatuuronderzoek uit te voeren. Tijdens de eerste backlog-grooming sessie, op het kantoor van pixelzebra, hadden wij in overleg besloten dat het verstandig was als ik mijzelf ging verdiepen in het principe van datamigratie. Eén onderdeel waar ik verplicht naar moest kijken was de tool die gebruikt zou worden voor de migratie. Ik heb toen een backlogitem opgenomen waarin ik het uit te voeren onderzoek beschrijf. Mijn verwachting voor het onderzoek was niet hoog. Ik was onder de veronderstelling dat ik de meeste aspecten van datamigratie wel kende. Buiten de voorwaarde een migratietool had de productowner ook geen verwachtingen geuit ten opzichte van het onderzoek. Ik ben vervolgens zeer praktisch bronnen gaan zoeken middels Google en Google Scholar. Deze resultaten had ik vervolgens direct opgenomen in mijn afstudeerverslag. Nadat ik de backlogitem had afgerond ging ik van start met backlogitems omtrent het inrichten van Scribe Online Dynamics CRM Online. Bij het uitvoeren deze items kwam ik erachter dat ik veel dingen niet wist. *Hoe map ik velden die leeg zijn? Wanneer moeten de mappings straks gemaakt worden? Kies ik voor een migratie in één keer of juist in kleine stappen?* Ik was constant nog extra informatie aan het bij zoeken. Dit zorgde ervoor dat de effort van mijn backlogitems niet meer klopte. Tijdens de retrospective van mijn eerste sprint heb ik vervolgens met Jan Groeneveld overlegd dat ik bij voorkeur in de volgende sprint het onderzoek opnieuw wilde uitvoeren maar dan 'goed'. Hij vond dit verstandig en was het met mij eens dat het heruitvoeren van het onderzoek de effort-

schatting beter zou maken en daarmee het project beheersbaarder. Tijdens de sprint planningssessie van sprint twee heb ik vervolgens het onderzoek opnieuw ingepland met een grotere effort. Het resultaat was een onderzoeksrapport dat tot en met schrijven van dit verslag valide is gebleven. Gezien het resultaat van het tweede verslag was het onverstandig van mij om een fundamenteel iets als een literatuuronderzoek even pragmatisch aan te willen pakken. De resultaten van mijn gehele project weerspiegelen de heruitvoer van het onderzoek. Vanaf nu zal ik dan ook zorgen dat ik altijd genoeg tijd inplan om literatuuronderzoek grondig uit te voeren.

Modulair ontwikkelen

Aan de start van mijn project heb ik een backlog grooming sessie gehouden met de product owner. Deze sessie heeft als doel het initieel vullen van de backlog. Tijdens deze sessie heb ik de eerste wensen en eisen van de productowner omgezet in backlogitems. Eén van deze items besloeg het rekening houden met verschillende bronomgevingen. Binnen de scope van het project zou ik alleen testen van Dynamics CRM Online naar Dynamics CRM Online. Ieder product dat ik oplever moest echter uiteindelijk wel 'klaar staan' voor andere omgevingen.

Grofweg halverwege mijn project had ik een review sessie met de productowner. Hierin heb ik hem de vooruitgang van de Scribe CRM Migration Assistant laten zien. Terwijl ik hem toonde wat de tool kon op dat moment kreeg ik de vraag hoe de tool straks om gaat met andere omgevingen. Ik schrok een beetje van deze vraag. Ik realiseerde mij dat ik alleen op het inlogscherf rekening had gehouden met mogelijke verschillende omgevingen. De klassenstructuur was echter compleet verworven met elkaar waardoor de toon compleet ongeschikt was voor eenvoudige code uitbreiding. Dit moment is mij goed bij gebleven omdat ik mij toen ben gaan verdiepen in het modulair programmeren. Ik heb de tool vervolgens herbouwt naar het Klassendiagram zoals deze te vinden is in de bijlagen.

Het herbouwen van de tool heeft gelukkig niet immens veel tijd gekost. Ik realiseer mij echter dat ik niet goed ben omgegaan met mijn Definition of Done en Definition of Ready. Ik heb deze opgesteld ben er mee omgegaan alsof het statische producten zijn. Dit is natuurlijk niet zo. Had ik tijdens de sprint planning sessies de Definitions erbij gehouden dan had ik backlogitems waarin dergelijke voorwaarden vermeld stonden opgenomen. Dit had zowel de beheersbaarheid van mijn project ten goede gekomen als de tijd die nodig is om onnodige fouten te corrigeren.

Tunnelvisie

In mijn projectplan heb ik opgenomen in de scope dat ik niet te lang de focus moet leggen op één aspect van datamigratie. Door alleen de focus te leggen op het versnellen van de dataoverdracht zou ik de kans missen om de migratie te versnellen middels het sneller vastleggen van requirements en vice versa. Na afloop van de derde sprint sprak ik tijdens de sprint retrospective (deze hielden wij in een aparte ruimte buiten pixelzebra's kantoor) met Jan Groeneveld over mijn angst voor de tunnelvisie. Ik gaf aan voornamelijk bezig te zijn met de uitvoeringsfase van mijn migratieproces maar, de voorbereidingsfase en nazorgfase zijn er ook nog. Jan's advies was om dit voor te leggen aan de productowner in de aankomende sprint; dit heb ik dan ook gedaan. Tot mijn frustratie stond de productowner er echter op dat ik de volgende sprint verder zou gaan met de uitvoeringsfase totdat deze af was. Daarna konden we verder kijken naar andere focuspunten. Ik ben toen akkoord gegaan maar vervelend gevoel in mijn maag. Mijn angst was dat ik dezelfde discussie de volgende sprintsessie weer moest houden en dat ik niet uit het tunnelvisieritme zou komen. Ik bleek gelijk te hebben want de volgende sprint review was qua discussie gelijk aan de voorgaande. Dit keer had ik echter besloten niet over mij heen te laten lopen, ik ben toen mijn poot stijf gehouden en heb passioneel beargumenteerd dat de voorwaarden van mijn project anders

niet waar kon maken. Uiteindelijk is de productowner overstag gegaan. Wat ik mij gerealiseerd heb op dit moment is dat niet meer zomaar een student ben. Mijn argumenten zijn in principe net zo valide als die van ieder anders en of je iets bereikt ligt geheel aan je eigen attitude. Deze realisatie heeft er voor gezorgd dat ik de rest van het project meer richting mijn eigen visie en minder richt de (tunnel)visie van de productowner heb kunnen inrichten.

BIMformatica

Ieder die dit verslag leest zal zich één ding afvragen: is dit nu een Business IT & Management of Informatica opdracht. Ik heb tijdens mijn vier jaar binnen de opleiding Business It & Management mijzelf altijd sterk gemaakt voor het feit dat ik vind dat 'BIM' studenten programmeer kennis moeten hebben. Al is het maar nihil. Ikzelf heb deze ervaring opgedaan tijdens de opleidingen MBO-applicatieontwikkeling en HBO Technische informatica. Hierdoor zie ik dat BIM steeds meer afdrijft richting de bedrijfskunde en steeds verder af komt te staan van de ICT.

Hoewel ik zelf vind dat mijn opdracht juist de aspecten van een BIM student omvat, ben ik bang dat mijn opdracht niet begrepen zal worden. Naast de bovengenoemde tunnelvisie is dit dan ook een reden waarom ik niet te diep in het programmeerwerk wilde kruipen. Terecht, aan de ene kant. De competenties van BIM bevatten nergens een stuk softwareontwikkeling maar, met Agile technieken zoals Scrum en XP is de vraag of er nog wel een dergelijke scheidslijn moet zijn.

Tijdens een sprint planning sessie met de product owner legde ik hem deze vraag voor. Hij wilde graag dat ik mijn tool bleef door ontwikkelen maar, ik was daar geen voorstander van omdat ik dat slechts deels kon verantwoorden aan de hand van BIM-competenties. Omdat ik zelf ook een sturende taak had voor het inplannen van sprints had ik daarom aangegeven dat ik andere werkzaamheden wilde uitvoeren om mijn project meer "all-round" te maken. De productowner kon zich hier in vinden maar, eiste wel dat ik de huidige items in de backlog eerst uit zou voeren.

Ik geloof dat ik er goed aan heb gedaan om niet nog dieper de techniek in te duiken. Hierdoor heb ik een mooi proces opgezet en heb ik, naar mijn mening, kunnen voldoen aan genoeg BIM-competenties. Wel zal ik mij altijd hard blijven maken voor het feit dat ik vind dat een BIM student meer is dan veredelde bedrijfskunde student met een ICT tintje. Voor mij is de BIM student een all-round ICT-er en dat stopt niet bij het afleveren van een model met Use cases bij de ontwikkelaars. Ik ben zelf dan ook tot de conclusie gekomen dat ik vind dat mijn project prima binnen deze definitie van een BIM student past.

Testmigratie

Aan het einde van mijn project had ik in overleg met de productowner besloten om een testmigratie uit te voeren middels mijn mappings en tools. Ik had dit eveneens voorgelegd aan mijn examinatoren tijdens mijn tussentijds assessment. Op dit punt was ik al druk bezig om dit verslag uit te werken maar ik dacht dat ik de migratie gemakkelijk tussendoor kon uitvoeren. Het voordeel was dat ik dit zowel thuis als op werk kon doen omdat alle benodigdheden in de Cloud staan.

Tijdens het uitvoeren van de migratie kwam ik keer op keer kleine problemen tegen. Zo kwam ik entiteiten tegen die automatisch gegeneerd worden en velden die niet gevuld hoeven te worden. De noodzaak van een test is daar meteen mee aangetoond. De resultaten kunnen direct verwerkt worden in mijn draaiboek. Ik ben de test echter te laat gestart om deze af te kunnen maken. Mijn verwachting voor het aantal problemen lag een stuk lager waardoor ik de testmigratie wat laks heb ingepland. Uiteindelijk heb ik de migratie voor +/- 40% uitgevoerd voordat ik gestopt ben. Ik realiseerde mij dat het meten van de

doorlooptijd van mijn test geen recht doet aan het migratieproces zodra de kleine bugs verholpen zijn. Daarnaast beïnvloedde de tijd die ik stak in het testen mijn andere werkzaamheden te veel. Ik heb daarom de keuze gemaakt om de testmigratieresultaten weg te laten uit mijn onderzoek. Ik vind de testmigratie niet nodig om mijn competenties aan te kunnen tonen en de gebrekkige resultaten zouden de kwaliteit van mijn verslag om laag gehaald hebben.

H11 Competenties

In dit laatste hoofdstuk besteed ik extra aandacht aan het toelichten van de competenties die ik gedurende mijn afstudeerproject heb uitgevoerd. Ik wil hiermee de nadruk leggen op het bewijs dat ik de competenties behaald heb en daarom fungeer op een niveau dat HBO waardig is. Ik ga er van uit dat middels het lezen van: het verslag, de product evaluatie en de proces evaluatie er al duidelijk is geworden dat ik een mengelmoes aan competenties behaald heb; allereerst competenties binnen Business IT & Management maar, ook competenties die normaliter vallen onder de opleiding Informatica. Ik zal hieronder per behaalde competentie kort herhalen wat de voorwaarden hiervoor zijn volgens de Haagse Hogeschool, welk bewijs ik heb geleverd voor het behalen van de competentie en een motivatie om te tonen dat twee aansluiten.

11.1 Business IT & Management

Hieronder is de toelichting met betrekking tot mijn behaalde Business IT & Management competenties te vinden. Per competentie is de beschrijving weergegeven zoals deze wordt aangeleverd door de Haagse Hogeschool, een motivatie voor het behalen van de competentie en een opsomming van het bewijs voor het behalen ervan.

Analyseren probleemdomein & opstellen probleemstelling

Analyseert en beschrijft een vraagstuk of probleem in de informatievoorziening. Voert een probleem oriëntatie uit. Analyseert de bedrijfssituatie. Stelt een probleemstelling op.

Bewijs	Locatie
Onderzoeksplan	Bijlage 2
Onderzoeksrapport	Bijlage 3
Template interviewboekje	Bijlage 5
Interviewresultaten	Bijlage 6
Migratie Draaiboek	Bijlage 4
Migratie Uitvoeringsstappen	Bijlage 8
Voorbeeld Scribe Online Mapping	Bijlage 14
Scribe CRM Migration Assistant Views	Bijlage 9
Procesmodellen	Bijlage 11

Pixelzebra heeft een eenvoudig probleem. Zij voert migraties uit maar heeft hiervoor geen structureel proces. Als gevolg hiervan is zij veel tijd kwijt is aan handelingen die veel minder tijd kunnen kosten. Zo kost het pixelzebra gemiddeld 2,5 tot 5 dagen om 500 gebruikers te migreren. Met behulp van mijn migratieproces kost dit slechts enkele minuten. Om pixelzebra een structureel proces te kunnen bieden heb ik onderzoek moeten naar wat een goede datamigratie nu eigenlijk is, welke aspecten daarin een rol spelen en welke elementen wel of niet relevant voor pixelzebra zijn.

Ik heb uiteindelijk de competentie bewijzen door het opstellen en opleveren van een aantal producten. Zo heb ik een onderzoeksplan en -rapport opgesteld om een literatuuronderzoek voor te bereiden, uit te voeren en vast te leggen. Daarnaast heb ik een paar consultants van pixelzebra bevraagd rondom het onderwerp datamigratie. Dit heb ik gedaan met behulp van een interviewboekje. De resultaten van dit onderzoek hebben bijgedragen aan vrijwel alle resultaten van mijn afstudeerproject.

Het doel van mijn project was niet om een primair of secundair bedrijfsproces te ontwerpen, implementeren of veranderen. Van een adviesrapport of interventies is daarom geen sprake. Het doel was duidelijk, creëer een proces dat pixelzebra kan gebruiken voor snelle(re) CRM-migraties. Het is dus een losstaand proces geworden dat specifiek ingericht is naar de wensen en behoeften van pixelzebra. Het is een direct inzetbaar proces dat geen implementatie vereist.

Omdat ik geen adviesrapport oplever heb ik producten opgesteld die naar mijn mening equivalent om de competentie te bewijzen. Dit is allereerst het draaiboek waarin het proces beschreven is. Daarnaast is er nog de maatwerktool “Scribe CRM Migration Assistant” die ontwikkelt is op basis van herkende behoeften. Verder heb ik nog een aantal procesmodellen opgesteld die inzicht bieden in complexe probleemprocessen die ik ben tegen gekomen tijdens mijn project. Tot slot zijn er nog een aantal producten benoemd die een direct resultaat zijn van het onderzoek.

Informatie vergaren, analyseren, beoordelen & verwerken

Vergaart informatie middels literatuuronderzoek, deskresearch, observatie, testen, enquêtes of ander veldonderzoek. Analyseert deze data of informatie en beoordeelt deze op basis van een gefundeerde set criteria. Resultaten worden op overzichtelijke wijze gevisualiseerd in tabellen, figuren, modellen en tekst.

Bewijs	Locatie
Onderzoeksplan	Bijlage 2
Onderzoeksrapport	Bijlage 3
Template interviewboekje	Bijlage 5
Interviewresultaten	Bijlage 6
Migratie Draaiboek	Bijlage 4
Mindmap	Bijlage 16

De aanleiding voor het uitvoeren van deze competentie is gelijk aan die van de competentie “Analyseren probleemdomein & opstellen probleemstelling”. Ik heb deze competentie dan ook bewezen met een deel van de producten die ik heb gebruikt om de bovengenoemde competentie te bewijzen. Voor het bewijzen van deze competentie zal ik uitwiden over de wijze waarop ik mijn literatuuronderzoek heb uitgevoerd.

Zoals de competentiebeschrijving suggereert is het nodig om mijn literatuuronderzoek gestructureerd uit te voeren middels duidelijke criteria. In mijn onderzoeksrapport heb ik dit op twee manieren bewezen. Eerst heb een duidelijke weging en prioritering aangebracht aan de zoektermen die ik heb verkregen middels een mindmapsessie. Door de termen te wegen middels: relevantie, frequentie en correlatie heb ik het zoekproces van mijn onderzoek gestructureerd. De tweede prioritering heb ik gedaan op basis van de gevonden bronnen. Door de bronnen te prioriteren op basis van kwaliteit heb ik mijzelf tijd bespaard, door minder ‘slechte’ informatie te lezen, en heb ik gezorgd dat ik de kwalitatief beste bronnen gebruikt heb voor de resultaten van mijn onderzoek. In het onderzoek heb ik tevens een tabel opgenomen waarin duidelijk opgenomen is welke bron middels welke techniek gevonden is. Dit draagt eveneens bij aan de kwaliteit van mijn literatuuronderzoek.

Zoals beargumenteerd bij de voorgaande competentie heb ik voor verder bewijs gebruikt gemaakt van het migratiedraaiboek als tegenhanger van een ‘standaard’ adviesrapport of interventieplan. De reden

hiervoor is de het feit dat mijn migratieproces als losstaande methode gebruikt dient te worden en niet geïmplementeerd moet worden als primair of secundair bedrijfsproces.

Ontwerpen Software

Gebruikt een standaard systeemontwikkelmethode en ontwikkelt een proof of concept volgens deze systeemontwikkelmethode.

Bewijs	Locatie
Scribe CRM Migration Assistant Views	Bijlage 9
Scribe CRM Migration Klassendiagram	Bijlage 10
Voorbeeld Backlogitem	Bijlage 12
Toelichting modulair ontwerpen	H7.3

Tijdens het lezen van dit afstudeerverslag zal duidelijk geworden zijn dat ik gewerkt heb middels de Scrummethodiek. Daarnaast zal duidelijk geworden zijn dat ik qua technische diepgang een stuk verder ben gegaan dan een gemiddelde Business IT & Management student. Deze twee karakteristieken van mijn afstudeerproject hebben gevolgen voor de wijze waarop ik deze competentie bewezen heb.

Scrum is een Agile techniek; Het Agile Manifesto dicteert werkende software boven documentatie. Daarnaast heb ik in plaats van alleen ontwerpen ook daadwerkelijk een stuk software ontwikkelt. Het "ontwerpen" heb ik daarom voornamelijk gedaan middels een klassendiagram en het vastleggen van requirements. Verder ontwerpen is gedaan door incrementeel ontwikkelen van het stuk software volgens het Agile principe. Hiermee doel ik op het vullen van de projectbacklog, het discussiëren over de noodzaak van backlogitems en het prioriteren en wegen ervan alsmede het achteraf beoordelen middels de sprint review.

Ontwerpen Database

Het gestructureerd ontwerpen van een database op basis van een gegevensmodel.

Bewijs	Locatie
Entiteitenmodel	Bijlage 7
Migratie Uitvoeringsstappen	Bijlage 8

De wijze waarop ik deze competentie heb bewezen is mogelijkwerwijs ietwat onorthodox. Ik heb het Entiteitenmodel (gebouwd op een SQL database) ik kaart gebracht middels een aantal bronnen zoals de Dynamics CRM Online interface, de advanced Query optie van Dynamics CRM Online, Scribe Online en met een pragmatische analyse van Dynamics CRM's metadata via de Scribe CRM Migration Assistant. Ik heb hiervoor louter de entiteiten in kaart gebracht die ik nodig had voor mijn migratie. De reden voor deze keuzen is de complexiteit van het standaard Datamodel van Dynamics CRM. Het aantal relaties, hiërarchische afhankelijkheden en entiteiten is dermate hoog dat de datamodellen geleverd door Microsoft zelf onbegrijpelijk zijn. Wat ik dus gedaan heb is het vereenvoudigen van Microsofts eigen ontwerp maar, wel vanaf nul.

Ik ben nog een stap verder gegaan dan een 'eenvoudig' Entiteitenmodel. Sequentie is een bijzonder belangrijke factor voor mijn proces, het migreren gaat namelijk op basis van de entiteiten en hun relaties en om tijdwinst te behalen moet dit in een bepaalde volgorde. Ik heb daarom op basis van de onderlinge

relaties een kleurcodering aangebracht op het Entiteitenmodel die de onderlinge sequentie duidelijk maakt.

Vergaren en analyseren van requirements

Achterhaalt behoeften van stakeholders, definieert op basis van behoeften de requirements voor het informatiesysteem en evt. hieraan gerelateerde verdere systeemeisen.

Bewijs	Locatie
Voorbeeld backlogitem	Bijlage 12
Definition of Ready & Definition of Done	Bijlage 1
Uittreksel Productbacklog	Bijlage 13

Voor het vastleggen van behoeften heb ik gebruik gemaakt van Scrum tezamen met een Definition of Ready en Definition of Done. Alle behoeften omtrent mijn project heb ik opgenomen in de productbacklog van mijn project. De analyse van de requirements heb ik uitgevoerd tijdens het inplannen van de sprint. Op dit moment heb ik samen met de productowner de effort en prioriteit van de backlogitems bepaald. Het verfijnen en beheersen van de requirements heb ik gedaan middels Definition of Ready en Definition of Done. Deze bevatten criteria waarin een backlogitem moet voldoen als volwaardige user story gezien te worden. Daarnaast bepalen zijn wanneer een story gestart mag worden en wanneer deze als afgerond beschouwt mag worden. Punten als volgordelijkheid, voorwaardelijke documentatie en validatie zitten daarmee ingebouwd in de backlogitems zelf.

Projecten managen

Initieert, beheerst en sluit projecten af op basis van projectmanagement methodieken

Bewijs	Locatie
Voorbeeld backlogitem	Bijlage 12
Definition of Ready & Definition of Done	Bijlage 1
Uittreksel Projectbacklog	Bijlage 13
Projectplan	Bijlage 1

Ik heb Scrum gebruikt als Framework voor mijn project. Hoewel het initieel bedoeld was voor softwareontwikkelprojecten kan het tegenwoordig veel breder ingezet worden, waartoe mijn project getuigt. Scrum is een goed middel om structuur aan te brengen in mijn project, met name omdat het doel van het project duidelijk is maar, de weg er naar toe niet. Middels het backlogitem en sprints heb ik op gestructureerde wijze werk ingepland. Daarnaast heb ik de sprint planning, sprint review en sprint retrospective gebruikt om sturing te geven aan mijn project.

Hoewel ik Scrum een bijzonder prettige manier vind om mijn project te managen mist het inhoudelijke artefacten om de kwaliteit ervan te waarborgen. Om deze reden heb ik Scrum expliciet aangevuld met de Definition of Ready en Definition of Done. Deze methoden geven mij de extra middelen die ik nodig heb om de kwaliteit, validiteit, controleerbaarheid en traceerbaarheid van mijn project te waarborgen. Een voorbeeld hiervan is pas mogen starten van een backlogitem als deze geprioriteerd is, de effort geschat is, het een user story notatie heeft, alle voorwaardelijke backlogitems afgerond zijn en alle voorwaardelijke documentatie gemaakt of ingepland is. Dit zijn precies de punten waarom Scrum niet inhoudelijk een antwoord op heeft.

Beheersbaar laten verlopen van ontwikkelprocessen

Stuurt het systeemontwikkelpject, oa door het selecteren van de juiste methoden en technieken en het opstellen van een business case.

Bewijs	Locatie
Voorbeeld backlogitem	Bijlage 12
Definition of Ready & Definition of Done	Bijlage 1
Uittreksel Projectbacklog	Bijlage 13
Scribe CRM Migration Assistant Views	Bijlage 9
Scribe CRM Migration Assistant Klassendiagram	Bijlage 10

Mijn keuze voor het managen van mijn project is Scrum geweest. Gezien het feit dat dit van oorsprong een softwareontwikkelmethode is, is dit een eerste keuze geweest die bijdraagt aan het beheersen van het ontwikkelonderdeel van mijn project. Zoals bij de “*Project managen*” competentie dragen de Definition of Ready en Definition of Done bij om het proces inhoudelijk op orde te houden.

Zoals ik heb beschreven in H10 heeft het gebruik van Scrum mij de mogelijkheid gegeven om tijdig bij te sturen in mijn project. Daar waar het helpt met het managen want mijn softwareontwikkelonderdeel heeft het eveneens geholpen bij het tijdig stoppen ermee om zodoende niet te veel tijd eraan te besteden waardoor ik andere competenties niet had kunnen

Naast de algemene keuzes die ik heb gemaakt, die met name slaan om projectmanagement, heb ik ook een aantal inhoudelijke keuzes gemaakt die gezorgd hebben dat ik het product zo ver mogelijk heb kunnen ontwikkelen. De eerste keuze hiervoor is het ontwikkelen middels een Windows Form Application. Door een Windows Form Application te gebruiken heb ik kunnen teren om mijn ervaring met Console applicaties. Hoewel mijn voorkeur uitgegaan was om te ontwikkelen middels ASP.net, heb ik door mijn keuze sneller kunnen ontwikkelen en testen.

Mijn tweede keuze die bijgedragen heeft aan het inhoudelijk beheersen van mijn softwareontwikkeling is de keuze om te werken met C#. Hoewel C# de geprefereerde ontwikkeltaal is binnen Microsoft was het prima mogelijk geweest om te ontwikkelen in Java of VB.net. Alle medewerker van pixelzebra hebben echter alleen ervaring met C#. Om de onderhoudbaarheid te waarborgen en de ontwikkelsnelheid te bevorderen heb ik daarom voor C# gekozen. Deze keuze toont eveneens aan dat ik inzicht heb getoond m.b.t. het beheersen van mijn softwareontwikkeltraject.

Mijn laatste bijdrage aan de beheersbaarheid van mijn softwareontwikkeltraject is de keuze om modulair te programmeren. Het stuk maatwerk dat ik ontwikkeld heb kan momenteel alleen communiceren met Microsoft Dynamics omgevingen. De bedoeling is echter dat dit in de toekomst uitgebreid wordt naar een variëteit van andere CRM-omgevingen. Door modulair te programmeren creëer ik kleine stukjes die individueel te testen en onderhouden zijn. Dit maakt het geheel beheersbaarder zowel tijdens als na mijn afstudeerproject.

11.2 Informatica

Mijn persoonlijke mening is dat de competentielijst van Business IT & Management relatief beperkt is. Een afstudeerder met een project als dit dat wat meer richting de IT gaat dan een gemiddeld 'BIM' project zal merken dat er enorm veel raakvlak is met de opleiding Informatica. Om aan te tonen dat ik niet alleen op HBO niveau fungeer, maar zelfs een dusdanige diepgang weet te bereiken dat ik direct raakvlak heb met de zusterstudie Informatica, heb ik een aantal competenties van de opleiding meegenomen in mijn lijst met behaalde competenties. Mijn doel is om hiermee aan te tonen dat ik binnen de opleiding Business IT & Management zowel in de breedte als in de diepte van de opleiding op HBO niveau heb gefungeerd.

Selecteren van methoden, technieken en tools

Selecteren en adviseren over een projectmanagementmethode. Selecteren en adviseren over methoden en technieken om een systeem te ontwikkelen, beheren en te testen. Selecteren en adviseren over ontwikkeltools die een onderdeel (gaan) vormen van de ontwikkel-, beheer-, test- en productieomgeving. Opzetten van de ontwikkel-, beheer-, test- en productieomgeving. Kunnen werken met de geselecteerde methoden binnen de gerealiseerde omgeving.

Bewijs	Locatie
Onderzoeksplan	Bijlage 2
Onderzoeksrapport	Bijlage 3
Projectplan	Bijlage 1
Scribe CRM Migration Assistant Views	Bijlage 9

Omdat het hier gaat om een Informatica competentie tracht ik de competentie te bewijzen middels de activiteiten die ik heb uitgevoerd rondom het ontwikkelen van software en het selecteren en gebruik van softwarepakketten. In het geval van deze competentie heb ik een selectie gemaakt voor methoden, technieken en tools voor zowel mijzelf als de productowner.

Hoewel ik Scrum heb gekozen als Framework voor het uitvoeren van mijn project en deze goed te beargumenteren valt binnen de context van deze competentie vind ik dat ik de keuze voor Scrum genoeg heb toegelicht in voorgaande BIM-competenties. Verder toelichting zou louter redundant zijn.

Voor het ontwikkelen van mijn maatwerk en het bijhouden van mijn voortgang heb ik gekozen voor Visual Studio Online. Visual Studio is de mainstream keuze als het gaat om ontwikkelen middels het .net framework. Daarnaast is het de standaard binnen pixelzebra. Een bijkomend voordeel van Visual Studio is de mogelijkheid om mijn werk in te checken. Zo staat mijn code veilig in de Cloud en kan iedere belangrijke stakeholder van mijn project desgewenst de code reviewen en eveneens mijn voortgang controleren.

Eén van de onderdelen van mijn literatuuronderzoek was het selecteren van een standaardtool voor het migreren tussen CRM-omgevingen. De keuze die ik hierin heb gemaakt (Scribe Online) toont aan dat ik verder kan kijken dan alleen criteria zoals kosten, gebruiksgemak en onderhoudbaarheid. Ik heb bewust voor een tool gekozen die potentie heeft voor het verbeteren middels maatwerk. Dit laat zien dat ik met een 'Informatica bril' kan kijken naar de potentie van een softwarepakket.

Ik heb een bewuste keuze gemaakt voor UML. In hoofdstuk vier beargumenteer ik waarom ik UML prefereer voor mijn project (sluit aan op de kennis binnen pixelzebra) en waarom ik geen gebruik maak

van de volgorde van de UML predikt. De reden hiervoor is dat ik continu modelleer voor een andere tool of omgeving. Dit toont aan dat ik genoeg inzicht heb vanuit een Informatica perspectief om een modelleerverzameling of techniek te selecteren en te adapteren binnen de context van het softwarelandschap van mijn project.

Selecteren van standaardsoftware

Adviseren over en/of selecteren van software, zoals een applicatie (commerciële software of open source), framework (al dan niet gebaseerd op open standaarden), database managementsysteem of besturingssysteem.

Bewijs	Locatie
Onderzoeksplan	Bijlage 1
Onderzoeksrapport	Bijlage 2

Het best voorbeeld dat ik hiervoor heb ik de keuze voor Scribe Online. Tijdens mijn onderzoek heb ik een long list van datamigratietools gebruikt die beschikbaar is op g2crowd.net. Dit is een op reviews gebaseerde vergelijkingswebsite. Uit deze lijst heb ik drie tools geselecteerd die op basis van verschillende criteria goede kanshebbers waren om binnen pixelzebra als standaardtool gebruikt te worden. Dit waren:

- Informatica: marktleider en geselecteerd door IBM als beste algemene migratietool;
- Scribe Online: uiterste gebruiksvriendelijke tool met speciale CRM-connectoren;
- Maatwerk: kan migratiesnelheden behalen die niet door standaardpakketten bereikt kunnen worden.

Dit onderdeel van mijn onderzoeksrapport is duidelijk bewijs dat ik bewuste selecties heb gemaakt als het gaat om standaard softwarepakketten. In de 'Selecteren van methoden, technieken en tools' competentie staat eveneens argumentatie rondom het selecteren van tools die van toepassing zijn op deze competentie zoals Visual Studio.

Bouwen applicatie

Verfijnen en/of transformeren van het (detail)ontwerp van de systeemdelen (subsystemen, componenten, modules) van een applicatie. Bouwen en documenteren van de systeemdelen. Systeemdelen samenstellen tot een werkende applicatie.

Bewijs	Locatie
Voorbeeld backlogitem	Bijlage 12
Definition of Ready & Definition of Done	Bijlage 1
Uittreksel Projectbacklog	Bijlage 13
Scribe CRM Migration Assistant	Bijlage 9
Scribe CRM Migration Assistant Klassendiagram	Bijlage 10
Procesmodellen	Bijlage 11

Tijdens mijn afstudeerproject heb ik een stuk maatwerk ontwikkeld met de naam Scribe CRM Migration Assistant. het bouwen van deze applicatie heb ik gemanaged middels Scrum en de Definition of Ready/Done. Het Klassendiagram toont aan dat ik de tool bewust modulair heb opgebouwd. Deze keuze heeft drie dingen opgeleverd die naar mijn mening aantonen dat ik bewust ben omgegaan met het bouwen van de tool:

- Het maakte het mogelijk om de tool werkend op te leveren;

- De tool kan straks verkocht worden terwijl modules voor andere CRM-omgevingen apart verkocht kunnen worden. Zo kan pixelzebra een flexibel product aan haar klanten leveren;
- De tool kan doorontwikkeld worden tot en met een oneindige hoeveelheid CRM of zelfs andere omgevingen.

Het procesmodel m.b.t. gebruikersmigratie maakt duidelijk dat ik een tool heb kunnen ontwikkelen die gelijktijdig met verschillende omgevingen kan communiceren. Het toevoegen van gebruikers in Office 365 en Dynamics CRM is een complexe reeks aan handelingen die voor grote hoeveelheden gebruikers dagen kan duren. De tool kan grote hoeveelheden gebruikers in theorie in paar uur migreren. Dit proces toont aan dat ik kennis genoeg heb om tegen complexe omgevingen aan te programmeren, zelfs binnen één proces.

Het ontwikkelen van de Scribe CRM Migration Assistant toont aan dat ik de competenties bevat een Windows Form Applicatie te bouwen binnen korting tijd die van grote toegevoegde waarde is voor pixelzebra en haar klanten. De tool toont aan dat ik mij bewust ben van keuze gevolgen zoals het statisch maken van classes om de performance te beteren en dat ik mij bewust ben van het platform waarop ik programmeer door defensief te programmeren middels try/catch methoden. Ik kan dus als Business IT & Management niet alleen een tool ontwerpen maar deze ook ontwikkelen, laten communiceren met complexe omgevingen, defensief programmeren en modulair opzetten.

Glossary

Term	Uitleg
A	
Acceptatie Criteria	Lijst met voorwaarden waaraan voldoen moet worden. Wordt meestal gebruikt binnen de context van een Scrum backlogitem.
Asynchroon	Een plug-in of workflow of methode die in een aparte thread wordt uitgevoerd en zodoende de werkzaamheden van de gebruiker niet in de weg staat.
B	
Big Bang	Een datamigratie die in één keer wordt uitgevoerd.
BIM	Zie Business It & Management.
Business Intelligence	Staat voor het verzamelen van gegevens binnen de eigen handelsactiviteit. Het kan omschreven worden als het proces van gegevens omzetten in informatie, dat vervolgens zou moeten leiden tot kennis en aanzetten tot adequate actie.
Business IT & Management	It opleiding binnen de academie voor ICT & Media aan de Haagse Hogeschool.
Business Rule	Client Side Proces dat velden en waarden op een formulier kan aanpassen.
C	
C#	Uitgesproken als "C sharp". Een programmeertaal ontwikkeld door Microsoft als deel van het .NET-initiatief.
CRM	Customer relationship management. Zie Microsoft Dynamics CRM.
Customization	Maatwerk in een migratieomgeving.
D	
Datamapping	Zie mapping.
Decharge	Officieel akkoord tussen de klant en pixelzebra omtrent het afsluiten van een project.
E	
Entiteit	Een object binnen Ms Dynamics CRM. Representeert een object in de werkelijkheid waarin het systeem een interesse heeft.
ETL	Extraction, Transformation and Load. Een groep technologieën die veelal gebruikt worden bij de koppeling tussen systemen, waarbij er gestreefd wordt naar een minimale technische koppeling. Het is een (batch)proces dat regelmatig gebruikt wordt bij datamigraties.
F	
For Each loop	Een iteratieve functie voor het doorlopen van ieder element in een verzameling die als parameter is meegegeven.
Form/formulier	Een formulier voor het aanmaken van een entiteit binnen CRM.
G	
Gebruiker	Records van het type Systemuser in Dynamics CRM Online. Representeert een eindgebruiker.
H	

N.v.t.	
I	
IDEF	Een verzameling modellen voor het modelleren en analyseren binnen softwaretrajecten.
J	
Javascript	Een veelgebruikte scripttaal om webpagina's interactief te maken en web applicaties te ontwikkelen.
K	
N.v.t.	N.v.t.
L	
N.v.t.	N.v.t.
M	
Mapping	XML/Scribe Online bestand voor het toewijzen van gegevens naar de juiste entiteit/veld tijdens een import.
MoSCoW	Prioriteringstechniek volgens de opties: Must have, Should have, Could have en Wont have
Microsoft Dynamics CRM	Tool van Microsoft voor het bijhouden van klantgegevens. Wordt veelal gebruikt door Sales- en marketing afdelingen binnen een organisatie.
Migratie/Migration	Het proces van het verplaatsen van een operationele omgeving naar een andere werkomgeving.
N	
.Net	Uitspraak: dotNET. Een applicatieframework voor de naadloze samenwerking van applicaties en bibliotheken geschreven in verschillende programmeertalen.
O	
Offerte	Financieel en functioneel aanbod voor het uitvoeren van een project.
Office 365	Cloud oplossing van Microsoft voor het office pakket. Dynamics CRM Online wordt tevens beheerd via de Office 365 portal.
Oplossing	Zie solution.
P	
Pixelzebra	Microsoft partner gespecialiseerd in CRM oplossingen met gebruik van Ms Dynamics CRM. Tevens productowner binnen dit project.
Plug-in	Een stuk 'custom' .Net code voor Ms Dynamics CRM.
Prefix	Een voorzetsel voor de naam van een custom object in Dynamics CRM Online.
Product owner	Term binnen Scrum, Verantwoordelijke vanuit de organisatie met mandaat om te spreken voor deze organisatie.
Q	
Query	Het bevragen en het aanpassen van gegevens in een relationele database.
R	
Record	Gegevensrij van een entiteit binnen de CRM database.
S	
Scribe	Een standaardpakket voor het migreren van records

Scribe CRM Migration Assistant	Maatwerktool ter ondersteuning van Scribe Online. Wordt gebruikt voor gebruikersmigratie, mappen van customization en migratierapportage.
Scrum	Een agile software ontwikkel framework.
Sequentie	Volgorderlijkheid in het migreren van entiteiten tijdens een datamigratie.
SharePoint	Microsoft Cloud oplossing voor bedrijven.
Solution	Een verzameling aangepaste entiteiten, workflows en andere CRM onderdelen. Een solution is exporteerbaar en importeerbaar.
Sprint	Iteratie binnen Scrum.
SQL	Structured Query Language. Een ANSI/ISO-standaardtaal voor een relationeel databasemanagementsysteem.
Sub-grid	Veld binnen CRM voor het verwijzen naar een child entiteit.
Synchroon	Een workflow of plug-in die tijdens het opslaan van een wijziging afgaat.
T	
Timeboxen	Het specificeren van een aantal uur en het werk dat daarin verzet moet worden. Als de tijd voorbij is maar, het werk is nog niet af wordt dit meegenomen in een nieuwe timebox.
Transformatie	Het wijzigen van (meta)data tijdens een datamigratie.
Trickle	Een datamigratie die middels kleine stappen (trickles) wordt uitgevoerd. Meestal van nacht op nacht.
Tweak	Aanpassing
U	
UML	Een verzameling modellen voor het modelleren en analyseren binnen softwaretrajecten. Bevat een volgorderlijkheid in de uitvoering.
V	
VBA	Visual Basic for Applications. Een programmeertaal die bestemd is om applicatieprogramma's te automatiseren of uit te breiden.
Veld	Gegevensveld in een CRM formulier gekoppeld aan een attribuut van de entiteit.
View	Beschrijving binnen CRM die aantoont welke gegevens worden getoond in een overzicht lijst van een entiteit.
Visual studio	Software ontwikkeltool van Microsoft.
W	
Workflow	Een proces binnen Ms Dynamics CRM. Wordt veel gebruikt voor het 'CRUD' van records naar aanleiding van een gebeurtenis.
X	
XML	Extensible Markup Language. Een standaard van het World Wide Web Consortium voor de syntaxis van formele opmaaktalen waarmee men gestructureerde gegevens kan weergeven in de vorm van platte tekst.
Y	
N.v.t	N.v.t
Z	
N.v.t	N.v.t

Literatuurlijst

- Abran, A., & Moore, J. W. (2004). *Guide to the Software Engineering Body of Knowledge*. Los Alamitos: IEEE Computer Society Press.
- Adelman, S., & Bischoff, J. (2002). *Impossible Data Warehouse Situations: Solutions from the Experts*. Boston: Addison-Wesley Longman Publishing Co.
- Albrecht, A., & Naumann, F. (2008). *Managing ETL Processes*. Potsdam: Hasso-Plattner-Institute.
- Ankorion, I. (2005, 01 1). *Change Data Capture – Efficient ETL for Real-Time BI*. Opgehaald van information-management.com: <http://www.information-management.com/issues/20050101/1016326-1.html>
- Aoyama, M. (1998). *New Age of Software Development: How Component-Based Software Engineering Changes the Way of Software Development ?* Fujihashi: Department of Information and Electronics Engineering Niigata Institute of Technology.
- Basili, V., Green, S., Laitenberger, O., Lanubile, F., Shull, F., Sörumsgard, S., & Zelkowitz, M. (1996). Empirical Investigation of Perspective-Based Reading. *Empirical Software Engineering*, Vol. 1, No. 12, Springer-Verlag, 133–144.
- Berardi, D., Calvanese, D., & Giacomo, G. D. (2005). reasoning on UML class diagrams. *Artificial intelligence*, Volume 168, Issue 1-2, 70-118.
- Boehm, B. (1981). *Software Engineering Economics*. Englewood Cliffs: Prentice Hall.
- Boehm, B. (1984). Verifying and Validating Software Requirements and Design Specifications. *IEEE Software*, Vol. 1, No. 1, 75–88.
- Coad, P., Luca, J. d., & Lefebvre, E. (1999). *Java Modeling Color with UML*. Upper Saddle River: Prentice Hall.
- Cohen, D., & Crabtree, B. (2006, 07). *Qualitative Research Guidelines Project*. Opgehaald van qualres.org: <http://www.qualres.org/HomeUnst-3630.html>
- Cohn, M. (2013, 02 26). *Sprint Zero: A Good Idea or Not?* Opgehaald van mountaingoatsoftware.com: <http://www.mountaingoatsoftware.com/blog/sprint-zero-a-good-idea-or-not>
- CRM, J. (2014, 06 18). *CRM Data Integration Performance Testing in Dynamics CRM*. Opgehaald van Powerobjects.com: <http://www.powerobjects.com/2014/06/18/crm-data-integration-performance-testing-dynamics-crm/>
- CRM, J. (2016). *The CRM Book*. Minneapolis: Power Objects.
- Dasu, T., & Johnson, T. (2003). *Exploratory Data Mining and Data Cleaning*. Hoboken: Wiley.
- de Swart, N. (2010). *Handboek Requirements*. Delft: Eburon Business.
- DeMarco, T. (1979). *Structured Analysis and System Specification*. Englewood Cliffs: Prentice-Hall.

- Eltabakh, M. (2012). *Data Integration*. Worcester: Worcester Polytechnic Institute.
- Fagundes, A. (2013, 08 20). *Performing Bulk Operations Using Scribe Insight Adapter for Dynamics CRM 2011*. Opgehaald van community.dynamics.com:
<https://community.dynamics.com/crm/b/crmpowerobjects/archive/2013/08/20/performing-bulk-operations-using-scribe-insight-adapter-for-dynamics-crm-2011>
- Firesmith, D. (2003). Specifying good requirements. *Journal of Object Technology*, 77-87.
- Fischer, T., & Hulsing, M. (2014). *Onderzoek doen!: kwantitatief en kwalitatief onderzoek*. Groningen: Noordhoff.
- g2crowd. (2016). *Compare: Informatica, Talend, Azure BizTalk, Informatica, webMethods, Integration Bus Advanced*. Opgehaald van g2crowd.com: <https://www.g2crowd.com/compare/informatica-integration-vs-talend-data-integration-vs-microsoft-azure-biztalk-services-vs-informatica-cloud-vs-webmethods-vs-ibm-integration-bus-advanced>
- G2Crowd. (2016). *Scribe Online Alternatives and Competitors*. Opgehaald van g2crowd.com:
<https://www.g2crowd.com/products/scribe-online/competitors/alternatives>
- Gartner. (2015). *2015 Gartner Magic Quadrant for Data Integration Tools*. Stamford: Gartner.
- Greer, D., & Conradi, R. (2008). *Software project initiation and planning – an empirical study*. Opgehaald van CiteSeer: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.163.9127>
- Grit, R. (2015). *Projectmanagement*. Groningen: Noordhoff.
- Groen, M. (2011). *Reflecteren de basis*. Houten: Noordhoff Uitgevers bv Groningen.
- Halevy, A., Rajaraman, A., & Ordille, J. (2006). *Data Integration: The Teenage Years*. Washington: BibTex.
- Hellerstein, J. M. (2008). *Quantitative Data Cleaning for Large Databases*. Berkeley: EECS Computer Science Division.
- Hovy, E., Ambite, J., & Philpot, A. (2003). Addressing a Bottleneck in Data Integration using Automated Learning Techniques. *Proceedings of the U.S. EPA Emission Inventory Conference*. San Diego: Information Sciences Institute of the University of Southern California.
- Huang, K.-T., Lee, Y. W., & Wang, a. R. (1999). *Quality Information and Knowledge*. Upper Saddle River: Prentice Hall.
- IBM. (2007). *Best practises for data migration*. Armonk: IBM Global Technology Services.
- IBM. (2013). *Business Case for Enterprise Data Integration Strategy*. Santa Cruz: IBM International Technology Grou.
- IEEE std, 8.-1. (1998). *Institute of Electrical and Electronics Engineers: IEEE Recommended Practice for Software Requirements Specifications*. New York: IEEE Computer Society.
- Johnson, T. (2004, 02 12). *Data Quality and Data Cleaning: An Overview*. Bedminster, New Jersey, Verenigde Staten.

- Jones, T. C. (1998). *Estimating Software Costs*. New York: McGraw-Hill.
- Kaay, P. v. (2014). *Scrum heeft meer toepassingen dan uitsluitend als software ontwikkel framework*. Zoetermeer: Haagse Hogeschool.
- Kaay, P. v. (2015). *Het interveniëren op de bedrijfsgroei van pixelzebra*. Den Haag: Haagse Hogeschool.
- Katzoff, D. (2009, 11 30). *How to Implement an Effective Data Migration Testing Strategy*. Opgehaald van datamigrationpro.com: <http://www.datamigrationpro.com/data-migration-articles/2009/11/30/how-to-implement-an-effective-data-migration-testing-strateg.html>
- Leach, P., Mealling, M., & Salz, R. (2005, 07). *A Universally Unique IDentifier (UUID) URN Namespace*. Opgehaald van <http://tools.ietf.org>: <http://tools.ietf.org/html/rfc4122>
- Lewis, J. P. (2005). *Project Planning, Scheduling & Control*. McGraw Hill: Columbus.
- Microsoft. (2014). *Protecting Data and Privacy in the Cloud*. San Francisco: Microsoft.
- Microsoft. (2015). *Best practices for developing with Microsoft Dynamics CRM*. Opgehaald van <http://msdn.microsoft.com>: <http://msdn.microsoft.com/en-us/library/gg509027.aspx>
- Microsoft. (2016). *Asynchrone service in Microsoft Dynamics CRM*. Opgehaald van msdn.microsoft.com: <https://msdn.microsoft.com/nl-nl/library/gg328021.aspx>
- Microsoft. (2016). *Gebruikers maken en Microsoft Dynamics CRM Online-beveiligingsrollen aan gebruikers toewijzen*. Redmond: Microsoft.
- Microsoft. (2016). *Introduction to entities in Microsoft Dynamics CRM*. Opgehaald van msdn.microsoft.com: <https://msdn.microsoft.com/en-us/library/gg309396.aspx>
- Microsoft. (2016). *Introduction to solutions*. Opgehaald van msdn.microsoft.com: https://msdn.microsoft.com/en-us/library/gg334576.aspx#BKMK_VersionCompat
- Microsoft. (2016). *Multiple Threads*. Opgehaald van msdn.microsoft.com: <https://msdn.microsoft.com/en-us/library/windows/desktop/ms684254%28v=vs.85%29.aspx>
- Mintzberg, H. (2013). *Organisatiestructuren*. Amsterdam: Pearson Benelux.
- Missi, F., Alshawi, S., & Fitzgerald, G. (2005). Why CRM efforts fail? A study of the impact of data quality and data integration. *Proceedings of the 38th Hawaii International Conference on System Sciences*. Uxbridge: Brunel University.
- Morris, J. (2012). *Practical Data Migration*. BCS learning & Development Ltd.: Swindon.
- Moss, L. T. (2000). *Data Warehouse Project Management*. New York: Pearson Education (US).
- Nielsen, D. (2010). *Controlling the scope*. Opgehaald van projectsmart.co.uk: <http://cdn.projectsmart.co.uk/pdf/controlling-project-scope.pdf>
- Noordam, P., & Zalm, M. v. (2012). *Kosten, Baten en Risico's van ICT-investeringen*. Alphen a/d Rijn: Vakmedianet Management.

- Noran, O. (20114). *UML vs IDEF: An Ontology-oriented Comparative Study in View of Business Modelling*. Porto: Conference: 6th International Conference on Enterprise Information Systems (ICEIS 2004) Volume 3.
- Oldenzeel, S. V. (2016, 02 09). CRM performance Workshop. *Managen Performance verwachtingen*. Schiphol, Noord-Holland, Nederland: Microsoft.
- Oracle. (2011). *Succesful Data Migration*. Phoenix: Oracle.
- Pohl, K., & Rupp, C. (2011). *Requirements Engineering Fundamentals*. Santa Barbara: Rocky Nook.
- Prakash, A. (2013, 9 4). *What Is Sprint Zero?* Opgehaald van scrumalliance.org:
<https://www.scrumalliance.org/community/articles/2013/september/what-is-sprint-zero>
- Rahm, E., & Do, H. H. (2000). Data Cleaning: Problems and Current Approaches. *Bulletin of the technical committee on Data engineering*, 3 - 14. Opgehaald van <http://betterevaluation.org>.
- Rational. (2001, 11). *Rational unified Process*. Opgehaald van ibm.com:
https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf
- Reinhartz-Berger, I., & Dori, D. (2005). OPM vs. UML--Experimenting with Comprehension and Construction of Web Application Models. *Empirical Software Engineering*, 57-80.
- Royce, W. W. (1970). *Managing the Development of Large Software Systems*. proceedings of IEEE WESCON.
- Rozsa, R. (2014). *Data Mapping and Conversion Made Easy With the Right Tools*. Opgehaald van toptal.com: <https://www.toptal.com/data-science/metadapper-data-conversion-and-mapping-made-easy>
- Russom, P. (2006). *Best Practices in Data Migration*. Redwood City: The data warehousing institute.
- Savitri, F. N. (2011). Study of localized data cleansing process for ETL performance improvement in independent datamart. *Electrical Engineering and Informatics (ICEEI), 2011 International Conference* (pp. 1-6). Bandung: IEEE.
- Seligman, L., Rosenthal, A., Lehner, P., & Smith, A. (2002). Data Integration: Where Does the Time Go? *Bulletin of the technical committee on Data Engineering* (pp. 3 - 46). Washington D.C.: IEEE Computer Society.
- Sherman, R. (2012, 08). Decision time: Automated data integration tools versus manual coding. (C. Stedman, Interviewer)
- Shull, F., Rus, I., & Basili, V. (2000). How Perspective-Based Reading Can Improve Requirements Inspections. *IEEE Computer*, Vol. 33, No. 7, 73–79.
- Solingen, R. v., & Rustenburg, E. (2010). *De kracht van Scrum*. Amsterdam: Pearson Education Benelux.
- Steyn, H. (2002). Project management applications of the theory of constraints beyond critical chain scheduling. *International Journal of Project Management*, Volume 20, Issue 1, 75-80.

- Sutherland, J. (2001). *Agile manifesto*. Opgehaald van [agilemanifesto.org](http://www.agilemanifesto.org): <http://www.agilemanifesto.org/>
- Swart, N. d. (2010). *Handboek Requirements*. Delft: Eburon business.
- SYSQA. (2013). *Checklist softwaremigraties*. Almere: SYSQA B.V.
- TDWI. (2012). *Data Integration Testing*. Renton: The Data warehousing institute.
- Tonnu, M., Nair, A., Mohammed, A., Choppadandi, N., Varma, A., Hansen, E., . . . Jochim, L. C. (2014). *Microsoft Dynamics CRM (on-premises) to Microsoft Dynamics Online Migration Guide*. Redmond: Microsoft.
- UCSC. (2016). *Evaluate the quality and credibility of your sources*. Santa Cruz: University of California.
- Veenings, P. (2007). *State of the art of "visualearning" in the Netherlands*. Maastricht: The Reflexion Foundation.
- Wake, B. (2003, 08 17). *INVEST in Good Stories, and SMART Tasks*. Opgehaald van <http://xp123.com>: <http://xp123.com/articles/invest-in-good-stories-and-smart-tasks/>
- Warmer, J., & Kleppe, A. (2011). *Praktisch UML*. Amsterdam: Pearson Benelux B.V.
- Well-Stam, D. v., Kinderen, S. v., & Brunt, B. v. (2013). *Risicomanagement voor projecten*. Houten: Het Spectrum.
- Wiegiers, K. (2000). *Software Testing & Quality Engineering. Software Quality Engineering*. Opgehaald van <http://processimpact.com>.
- Wiegiers, K. (2003). *Software requirements, second edition*. Washington: Microsoft Press.
- Williamson, K. (2009, 04 20). *Don't Delete. Deactivate*. Opgehaald van blogs.msdn.com: <https://blogs.msdn.microsoft.com/dynamicscrmonline/2009/04/20/dont-delete-deactivate/>
- Williamson, L. G., Campion, J. E., Malos, S. B., Roehling, M. V., & Campion, M. (1997). Employment interview on trial: Linking interview structure with litigation outcomes. *Journal of Applied Psychology issue 82*, 900-912.
- Withall, S. (2007). *Software Requirement Patterns*. Redmond: Microsoft Press.
- Wlodarz, D. (2014). *Comparing cloud vs on-premise? Six hidden costs people always forget about*. Opgehaald van <http://betanews.com>: <http://betanews.com/2013/11/04/comparing-cloud-vs-on-premise-six-hidden-costs-people-always-forget-about/>
- Zeoli, T. (2009, 11 21). *The Best Option For Integrating and Migrating Data for Microsoft Dynamics CRM*. Opgehaald van [crmsoftwareblog.com](http://www.crmsoftwareblog.com): <http://www.crmsoftwareblog.com/2009/11/the-best-option-for-integrating-and-migrating-data-for-microsoft-dynamics-crm/>
- Zhang, X., & Qiao, D. (2010). *Quality, Reliability, Security and Robustness in Heterogeneous Networks*. Houston: Springer.