

Scriptie

Onderzoek naar beschikbare IaaS oplossingen en het toepassen hiervan bij i3D.net

Abstract:

Dit document beschrijft het verloop van het afstudeerproject bij i3D.net waarbij een plan van aanpak werd opgesteld en een vergelijkend onderzoek werd uitgevoerd tussen 'Infrastructure as a Service'-pakketten. Daaruit volgde een adviesrapport op basis van het marktonderzoek en de softwareanalyse. Ter controle van de keuze werd een Proof of Concept opgezet. Ten slotte is er een adviesrapport geschreven met vervolgstappen.

Opdrachtgever:

Rick Sloot

i3D.net

Van Nelleweg 1, 3044BC

Rotterdam

Opdrachtnemer:

Chris Zonneveld

09032835

Rotterdamseweg 137, 2628AL

Haagse Hogeschool Delft

Copyright © 2015. Het copyright berust bij C.E. Zonneveld

Niets uit dit document mag worden verveelvoudigd en/of openbaar gemaakt door middel van druk, fotokopie, microfilm, geluidsband, elektronisch of op welke andere wijze dan ook en evenmin in een retrieval systeem worden opgeslagen, zonder voorafgaande schriftelijke toestemming van C.E. Zonneveld

Voorwoord

Voor u ligt de scriptie betreffende de afstudeeropdracht: 'Onderzoek naar het toepassen van Openstack, en vergelijkbare oplossingen, bij i3D.net.' Een onderzoek uitgevoerd in opdracht van i3D.net op het gebied van IaaS oplossingen en cloud virtualisatie. Dit document is geschreven voor mijn afstuderen aan de Haagse Hogeschool te Delft. De periode waarin dit project liep en deze documentatie is geschreven was van Februari 2015 tot en met Juni 2015.

Samen met de bedrijfsbegeleider Rick Sloot is de opdracht tot stand gekomen. Alhoewel de vraag in de eerste instantie vrij breed was heb ik deze met hulp van Rick wat kunnen specificeren en uit kunnen werken. Verder heb ik met hulp van Dhr. J.D. Schagen vanuit de Haagse Hogeschool genoeg begeleiding kunnen krijgen wat betreft vragen met betrekking tot het verloop van het afstuderen en het schrijven van documentatie. Dhr. Schagen heeft mijn vragen altijd spoedig beantwoord.

Ten slotte zou ik graag de volgende personen willen bedanken:

- Rick Sloot - Voor het leveren van een interessante opdracht en werkomgeving
- Jan Dirk Schagen - Voor de begeleiding vanuit de Haagse Hogeschool
- Collega's bij i3D.net - Het leveren van een fijne werksfeer en het open staan voor vragen tijdens dit project
- De leden van de Openstack community - Voor de behulpzaamheid en het beantwoorden van vragen.
- Sabine Waasdorp - Motivatie, steun en feedback
- Ed Zonneveld en Coby Zonneveld - Motivatie en steun

Met jullie support heb ik dit project succesvol kunnen voltooien.

Versiebeheer

Versie	Auteur	Datum	Wijzigingen
Scriptie 0.1	C. Zonneveld	27-2-2015	Eerste concept
Scriptie 0.2	C. Zonneveld	8-3-2015	Ruwe layout opgezet
Scriptie 0.3	C. Zonneveld	15-3-2015	Opvulling eerste hoofdstukken
Scriptie 0.4	C. Zonneveld	29-4-2015	Eerste controleversie af
Scriptie 1.0	C. Zonneveld	4-6-2015	Laatste controle

Verdeellijst scriptie

Organisatie	Naam	Occupatie
De Haagse Hogeschool - Delft	C.E. Zonneveld	Student
De Haagse Hogeschool	J.D. Schagen	Examinator
De Haagse Hogeschool	M.G.M. Maris	2 ^e Examinator
De Haagse Hogeschool	A. Andrioli	Stagecoördinator
i3D.net	R. Slood	Opdrachtgever

1.) Inleiding	4
1.1) Doel van dit document	4
1.2) Motivatie	4
1.3) Organisatie	5
1.4) Taak en positie binnen de organisatie	6
2.) Initiatiefase + Requirements - Projectbeschrijving	7
2.1) Achtergrond	7
2.2) Doelstelling opdracht	7
2.3) Opdrachtbeschrijving	7
2.4) Aanpak	8
2.5) Projectinrichting	13
2.6) Risicofactoren	15
2.7) Kosten en baten	16
3.) Design and Architecture	17
3.1) Wat is Infrastructure as a Service?	17
3.2) Software onderzoek: Welke IaaS oplossingen zijn er beschikbaar voor i3D.net?	18
3.3) Advies: Welk pakket is geschikt voor i3D.net?	26
4.) Development and Coding	27
4.1) Openstack introductie	27
4.2) Het Proof of Concept ontwerpen en opzetten	29
4.3) Is het Openstack Proof of Concept geslaagd?	37
5.) Quality Assurance and Software Testing + Afronding	38
5.1) Advies Openstack	38
6.) Terugblik resultaten	40
6.1) Plan van Aanpak	40
6.2) Bedrijfsoriëntatie	40
6.3) Softwareonderzoek	40
6.4) Proof of Concept documentatie	41
6.5) Adviesrapport	41
7.) Evaluatie producten	42
7.1) Plan van Aanpak	42
7.2) Bedrijfsoriëntatie	42
7.3) Softwareonderzoek	43
7.4) Proof of Concept	43
7.5) Adviesrapport	43
8.) Terugblik afstuderen	44
8.1) Methoden & Technieken	44
8.2) Aanpak planning	45
8.4) Aantonen competenties	46
8.5) Evaluatie persoonlijke ervaring	47
8.6) Evaluatie Stage	48
8.7) Evaluatie Opleiding	48
Bibliografie	49
Appendix	50

1.) Inleiding

Deze sectie beschrijft het doel van dit document, wat de aanleiding voor het project was, bij welke organisatie dit project is uitgevoerd en ten slotte waar de student binnen het bedrijf stond.

1.1) Doel van dit document

Het doel van dit verslag is om alle betrokkenen bij dit project inzicht te geven over het verloop van dit onderzoek.

Ten eerste wordt het probleem en het doel van het project toegelicht. De probleemstelling bespreekt wat de aanleiding voor i3D.net (Hierna: Opdrachtgever) geweest is om deze opdracht beschikbaar te stellen. Hierna worden de doelstellingen weergegeven.

1.2) Motivatie

De opdrachtgever heeft dit project opgezet als een afstudeeropdracht om zelf meer informatie te kunnen vergaren over cloudoplossingen en hun kennis te verbreden op dit gebied. Door onderzoek te blijven doen naar cloudoplossingen wil de opdrachtgever meegaan met de andere hosting aanbieders op het gebied van gebruiksgemak en snelheid van het leveren van hun diensten. De stagiair had al kennis van de VMWare Cloud oplossingen en heeft op deze manier meer kennis op dit gebied kunnen vergaren.

1.3) Organisatie

Hosting is de focus van i3D.net, het bedrijf i3D.net is gericht op het snel en gebruiksvriendelijk aanbieden van deze dienst. Het bedrijf is in 2004 begonnen in Rotterdam en is inmiddels een van de snelst groeiende managed hosting providers van de Benelux. Op het moment bedienen ze ruim 30.000 klanten vanuit 16 datacenters wereldwijd naast hun eigen serverpark van ruim 8.000 servers in Rotterdam. De klanten komen uit verschillende branches waaronder: overheid, zorg, sport, gaming, educatie, webshops, hosting en printshops.

Een korte lijst met feiten over i3D.net:

- Bezit en beheert meer dan 8,000 servers
- Huisvest meer dan colocatie-servers
- 16 locaties wereldwijd
- 24/7 technische ondersteuning
- Gecertificeerde professionals
- Opgenomen in de 'Deloitte Fast500' snelst groeiende technologiebedrijven
- Winnaar van de 'Financieele Dagblad Gouden Gazellen Award'
- CDSA gecertificeert op Content Protection & Veiligheid
- ISO/IEC 27001 audited datacenters
- Lid van de Dutch Hosting Provider Association (DHPA)
- Management heeft meer dan 10 jaar werkveldervaring

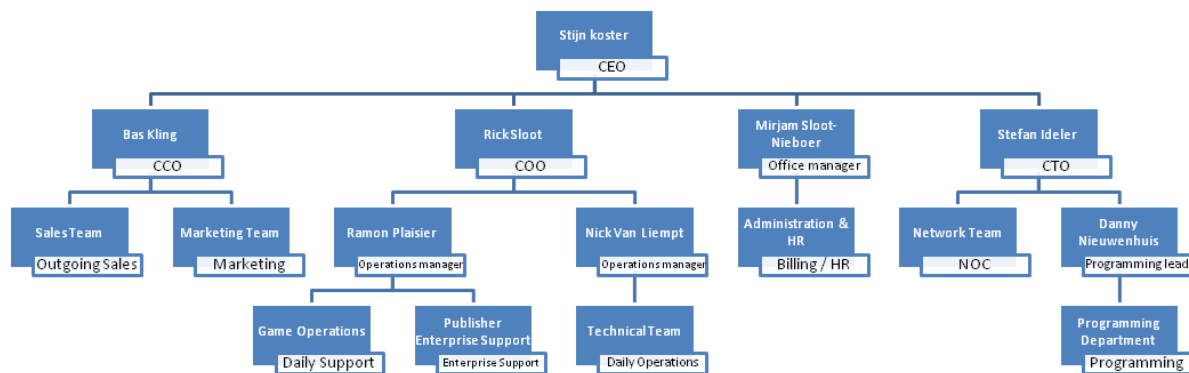
De afstudeeropdracht zal plaatsvinden op de netwerk afdeling van de opdrachtgever, die zich bevindt in het datacenter van i3D.net in de Van Nelle fabriek te Rotterdam. Op de afdeling zijn gemiddeld tussen de 2-5 personen dagelijks aanwezig, die zich bezig houden met alle ontwikkelingen en research rondom de core van het netwerk. Het resultaat van dit project zal voornamelijk van toepassing zijn voor deze afdeling, maar zal ook een positieve invloed hebben op de afdeling Operations welke ook baat heeft bij dit project.

1.4) Taak en positie binnen de organisatie

Bij i3D werk ik als afstudeerstagiair op de het NOC (Network Operations Center). Deze afdeling is rustiger vergeleken met de afdeling operations waar het rumoerig kan worden i.v.m. support calls. In deze rustige omgeving kan ik zonder te veel gestoord te worden aan mijn eigen project werken.

Daarnaast biedt deze afdeling de mogelijkheid mee te luisteren en mee te denken met problemen en projecten die op dat moment spelen. Op deze manier wordt er ook meteen een idee verkregen van hoe de dagelijkse werkzaamheden op deze afdeling er aan toe gaan.

Om mijn eigen project uit te voeren heb ik een werkplek toegewezen gekregen. Alle benodigdheden voor het project zijn door korte communicatie lijnen gemakkelijk te bemachtigen. De servers die aangevraagd zijn worden vlot opgeleverd met de benodigde configuratie zodat de stagiair hier vlot mee aan de slag kan.



2.) Initiatiefase + Requirements - Projectbeschrijving

2.1) Achtergrond

De opdracht voor de stagiair is voortgekomen uit de groei van i3D.net als bedrijf. Als hosting bedrijf is i3D continu bezig met onderzoek naar de huidige relevante technieken in de hosting wereld. Zij zijn begonnen met het hosten van gameservers, maar na het succes hiervan hebben ze kunnen uitbreiden naar managed- en sharedhosting oplossingen en colocatie. Voor colocatie is één van de grootste concerns ruimte; bij managed hosting is dat echter techniek en kennis. Om klanten meer mogelijkheden te bieden op het gebied van hosting is de logische volgende stap virtualisatie. Hiervoor is i3D zelf al bezig met een hyper-V omgeving, maar het bieden van een alternatief is waardevol voor i3D. Ze zouden naast Hyper-V hosting, goedkopere hosting kunnen bieden zoals KVM(Linux) hosting. Hiervoor is wel een manier nodig om dit op een duidelijke en gebruiksvriendelijke manier te hosten. De probleemstelling van de afstudeeropdracht komt voort uit de zoektocht naar een oplossing om IaaS op een slimme wijze te kunnen aanbieden aan klanten.

2.2) Doelstelling opdracht

Het doel is om klanten een groter aanbod te kunnen bieden van hosting diensten, zonder de werkdruk bij i3D buiten proporties te beïnvloeden. Hiervoor zal een platform nodig zijn wat hosting met support voor meerdere hypervisors kan aanbieden, op een manier welke voor de klant duidelijk en gebruiksvriendelijk is. Deze omgeving moet daarnaast door i3D gemakkelijk te schalen en beheren zijn. De Hoofdvraag welke hiermee i3D.net in samenwerking met de stagiair wil beantwoorden is "Welke IaaS oplossingen zijn geschikt voor i3D.net en hoe kan deze geïmplementeerd worden".

2.3) Opdrachtbeschrijving

Naar aanleiding van onderzoek naar verschillende IaaS oplossingen, zal er een IaaS oplossing uitgewerkt worden tot een Proof of Concept. Vervolgens zal getest worden of deze oplossing naar wens functioneert voor de opdrachtgever. Mocht dit Proof of Concept geslaagd zijn, kan deze uitgewerkt en overgedragen worden naar de opdrachtgever om verder in productie te laten nemen.

2.4) Aanpak

Voor dit project zijn eerst een aantal methoden gekozen op basis van een aantal eigenschappen die de toepasbaarheid op dit specifieke project waarborgen. Deze methoden zijn met elkaar vergeleken, samengevat te zien in onderstaande tabel, waarna de beste gekozen is. De kandidaat methoden voor dit project waren als volgt:

Waterval methode [sashimi variatie]

Dit is een methode welke door middel van duidelijke richtlijnen werkt. Er wordt in duidelijke fases gewerkt die niet afgesloten worden tot ze 100% compleet zijn. De Sashimi variatie voegt hier overloop tussen de fasen aan toe waardoor er efficiënter gebruik gemaakt wordt van de beschikbare tijd en er meer controle is over het project.^[1, 17]

RUP

Een software ontwikkelingsproces, gebaseerd op het incrementeel en iteratief ontwikkelen van software. Voornamelijk in gebruik genomen om grote software projecten te ontwikkelen om deze efficiënt te kunnen splitsen in meerdere onderdelen die simultaan ontwikkeld worden.

SCRUM

Een flexibel proces om software te ontwikkelen. Het wordt voornamelijk ingezet om binnen korte perioden werkende producten op te leveren. Kenmerkend zijn dagelijkse scrum meetings om voortgang en planning te bespreken.

PRINCE2

Een zeer flexibele methode met grote flexibiliteit. Kenmerkend is het tailoren van PRINCE2 aan het specifieke project en de voortdurende rechtvaardiging. Onderdelen die niet van belang zijn kunnen weggesneden worden uit de methode om deze aan te passen aan de huidige situatie

	Eigenschappen	Flexibiliteit	Projectgrootte	Onderwerp relevantie	Contactmomenten	Doorlooptijd
Methode						
Sashimi		+-	+	+	++	+
RUP		+-	+-	--	+-	+-
SCRUM		-	+	--	--	+-
PRINCE2		++	-	+	+-	+-

Evaluatie:

Bij het evalueren van de methoden tegenover de onderstaande eisen is er besloten met de Sashimi methode verder te werken om de volgende redenen:

-Projectgrootte

Dit is een van de belangrijkste punten bij de keuze van een methode. Ontwikkelmethodes gaan er snel vanuit dat er in groepen van 20 tot 1000 man gewerkt wordt. Bij dit soort grote groepen is duidelijke documentatie een must om het overzicht te behouden. Dit project is echter een dusdanig kleiner project dat grote hoeveelheden documentatiewerk niet alleen onnodig zijn om overzicht te houden maar ook kostbare tijd in beslag nemen. Hiervoor zijn Sashimi en PRINCE2 ideaal omdat beiden of niet al te veel documentatie vereisen of mogelijkheden hebben om de hoeveelheid documentatie in te perken.

-Flexibiliteit

Flexibiliteit is geen grote eis van het project. Alleen halverwege het project is er flexibiliteit vereist, wanneer er een software pakket gekozen moet worden die tot Proof of Concept uitgewerkt wordt. Hiervoor is Sashimi geschikt omdat deze de mogelijkheid van een overlapping tussen twee fasen biedt, waardoor zonder een hele fase opnieuw te moeten starten er wijzigingen in de vorige fase gemaakt kunnen worden. De andere methoden zijn hier naast SCRUM in principe ook geschikt voor. SCRUM heeft echter als nadeel dat er snel gewerkt wordt naar een prototype, wat hier bij niet nodig en te limiterend is.

-Onderwerp relevantie

Op dit criteria vallen een aantal methodes al af: SCRUM en RUP zijn namelijk primair methoden gebaseerd op product- en softwareontwikkeling. Aangezien dit project niet daadwerkelijk software ontwikkelt maar een pakket kiest en op basis daarvan er een Proof of Concept uitrolt, zijn deze methoden wel in zeker mate toe te passen maar niet geheel geschikt voor dit soort projecten.

Sashimi en PRINCE2 zijn beide breder toepasbaar.

-Contactmomenten

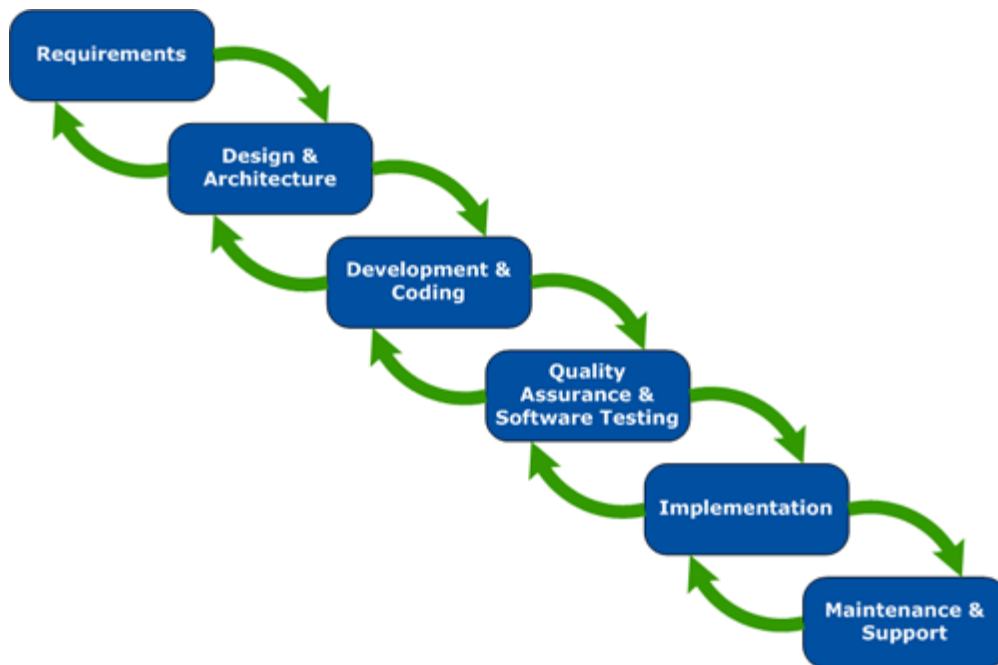
Er wordt tijdens dit project gerekend op contact momenten van een maal per één of twee weken. Verder is het wel tussendoor mogelijk om vragen te stellen, maar dit staat niet gelijk aan vooraf geplande afspraken waar beide partijen zich op kunnen voorbereiden. SCRUM is hierbij één van de grootste missers vanwege de dagelijkse meetings. De andere methoden zijn allemaal geschikt voor de relevante contactmomenten.

-Doorlooptijd

Er moet rekening gehouden worden met een projectduur van 17 weken. SCRUM werkt doorgaans met korte sprint planningen van 1-4 weken waarna opnieuw gepland wordt aan welke features er gewerkt moet worden. Dit is naar mijn mening inefficiënt voor een onderzoeksproject van 17 weken. De overige methoden kunnen allemaal in voldoende mate toegepast worden op de verwachte doorlooptijd.

Projectplanning

Voor dit project is de Sashimi variant, gedeeltelijk gewijzigd om beter aan te sluiten op het project, gekozen. Normaal gesproken bestaat de Sashimi methode uit de volgende fasen: Requirements, Design and Architecture, Development and Coding, Quality Assurance and Software Testing, Implementation, and Maintenance and Support.



Gezien er een Proof of Concept omgeving gerealiseerd wordt in tegenstelling tot een productie omgeving welke continu onderhoud vereist, is er gekozen om de Maintenance and Support fase te vervangen door een generieke afrondingsfase. De Implementation zal als afrondingsfase gebruikt worden. Tijdens de afronding wordt het implementatieadvies gegeven en ter afronding wordt aan het eind van deze fase de documentatie overgedragen.

De werkzaamheden van het project zijn onderverdeeld in vier fasen, namelijk de *Initiatiefase + Requirement fase*, de *Design and Architecture fase*, de *Development and Coding fase* en de *Quality Assurance and Software Testing + Afronding fase*.

Initiatiefase + Requirements

Gedurende deze fase zal er voornamelijk documentatie geschreven worden betreffende het project. Deze fase is bedoeld om de afstudeerder zich een duidelijk beeld te laten vormen over de organisatie waarbinnen het project uitgevoerd wordt en zich te kunnen oriënteren op de werksfeer binnen het bedrijf. In samenwerking met de begeleiders vanuit het bedrijf zal de stagiair de werkplek opzetten zodat er zelfstandig verder gewerkt kan worden.

Verder zullen er tijdens de initiatie ook afspraken gemaakt worden tussen de begeleiders van het bedrijf en de begeleiders van de Haagse Hogeschool om het verloop van het project bij te houden. Hiervoor zullen afspraken gemaakt worden zoals wekelijkse afspraken om vragen te beantwoorden, bedrijfsbezoeken en documentatie reviews.

Deze fase word ook gebruikt om alvast een idee te krijgen van wat de opdrachtgever verwacht van het project en globale eisen te formuleren.

Design and Architecture

Tijdens deze fase zal er gewerkt worden aan de theoretische achtergrond van het project. De stagiair verdiept zich verder in de theorie achter IaaS en de oplossingen welke hiervoor beschikbaar zijn. Tijdens dit onderzoek zal er documentatie geschreven worden over de beschikbare oplossingen en welke van hen het beste toepasbaar zullen zijn voor de omgeving van de opdrachtgever.

Verder wordt deze fase gebruikt om de eisen uit de Requirements fase te verfijnen. Aan de hand van de kennis welke opgedaan is tijdens het onderzoek naar verscheidene IaaS oplossingen is hier de mogelijkheid toe. Hierdoor is het mogelijk voor de stagiair om vast te stellen waar het eindproduct aan moet voldoen om als een Proof of Concept gekwalificeerd te worden.

Development and Coding

In deze fase zal het resultaat van de designfase uitgewerkt worden tot een Proof of Concept. Dit Proof of Concept zal de opdrachtgever meer inzicht moeten geven over de gekozen oplossing zodat er besloten kan worden of dit is wat verwacht wordt van het gekozen pakket.

Quality Assurance and Software Testing + Afronding

Tijdens de laatste fase zal er gekeken worden of het Proof of Concept inderdaad voldoet aan de eisen welke gesteld zijn door de opdrachtgever. Hierbij zal gekeken worden naar wat er eerder met de opdrachtgever besproken is. Op basis hiervan zal een adviesrapport geschreven worden.

Hierbij zal tevens alle documentatie welke voor de opdrachtgever bestemd is overhandigd worden. Dit zal dienen ter informatie en naslag voor de opdrachtgever mochten zij andere medewerkers verder laten werken met de uiteindelijke oplevering.

De fasering is globaal ingedeeld als volgt:

Afstudeer dossier schrijven	Alle fasen	15 dagen
Plan van Aanpak schrijven	Initiatiefase + Requirements	4 dagen
Bedrijfsoriëntatie schrijven	Initiatiefase + Requirements	4 dagen
Software onderzoek uitvoeren	Design and Architecture	4 dagen
Gesprekken voeren met medewerkers	Design and Architecture	3 dagen
Bespreken eisen medewerkers en opdrachtgever	Design and Architecture	3 dagen
Beschikbare documentatie software oplossingen uitzoeken	Design and Architecture	3 dagen
Proof of Concept maken	Development and Coding	15 dagen
Proof of Concept bespreken met opdrachtgever/werknemers	Development and Coding	3 dagen
Proof of Concept testen	Development and Coding	10 dagen
Documentatie over Proof of Concept schrijven	Development and Coding	5 dagen
Advies geven over software pakket	Quality Assurance and Software Testing + Afronding	1 dag
Implementeren	Quality Assurance and Software Testing + Afronding	5 dagen
Overdragen	Quality Assurance and Software Testing + Afronding	1 dag
Uitloop	Alle fasen	9 dagen

2.5) Projectinrichting

Om het project succesvol te later verlopen moeten er een aantal afspraken vastgelegd worden over de organisatie van het project en wie waar verantwoordelijk voor is. Ten eerste zal de organisatie besproken worden.

Organisatie

Jan Dirk Schagen & Marinus Maris (afstudeerbegeleiders)

Respectievelijk de primaire en secundaire begeleiders vanuit de Haagse Hogeschool. De stagiair zal contact opnemen met de begeleiders om afspraken te maken m.b.t. het geven van feedback en andere verplichte onderdelen van het afstuderen. Verder zal het resultaat beoordeeld worden door deze begeleiders.

Rick Slood (Opdrachtgever & Bedrijfsbegeleider)

De opdrachtgever zal aanspreekbaar zijn voor vragen met betrekking tot de werkzaamheden van de stagiair binnen het bedrijf. De begeleider zal tevens aanwezig zijn bij de eindbeoordeling van de stagiair.

Chris Zonneveld (Afstudeerder)

De afstudeerder zal de in het Plan van Aanpak beschreven taken uitvoeren en daarnaast de taken die tijdens het uitvoeren van de opdrachten mogelijk nog naar voren komen. De verantwoordelijkheden bestaan uit het managen van het project om deze op de juiste manier te laten verlopen en de begeleiders hierbij op de hoogte te houden door middel van het tijdig maken van afspraken.

Communicatie

De informatievoorziening is als volgt ingericht:

-Bedrijfscommunicatie

Om binnen het bedrijf te communiceren hanteer ik de methode die hier standaard gebruikt wordt door alle medewerkers. Voor formele communicatie wordt e-mail gebruikt, hiervoor heeft de stagiair ook een e-mail adres van i3D.net tot beschikking. Voor informele interne communicatie zijn een aantal mogelijkheden. Intern wordt voornamelijk gebruik gemaakt van Skype, voor korte vragen of gesprekken. Dit is een snelle manier om kort informatie op te vragen. Voor de problemen of vragen die meer uitleg vereisen kan er ook gebeld worden; hiervoor is een telefoon beschikbaar op de werkplek van de stagiair. Verder is er uiteraard altijd de mogelijkheid om collega's persoonlijk aan te spreken als deze op de afdeling aanwezig zijn.

-Communicatie naar de opleiding

De communicatie naar de begeleiders vanuit de Haagse Hogeschool zal voornamelijk via e-mail plaatsvinden; waar nodig zullen er gesprekken ingepland worden om elkaar direct te spreken. Indien gewenst kan er telefonisch contact opgenomen worden met de begeleiders.

Begeleiding

Voor kwaliteitsborging worden een aantal controles bijgehouden.

-Schoolbegeleiding

De begeleiders vanuit de Haagse Hogeschool zullen voornamelijk controles aanhouden op basis van de ingeleverde documentatie. Per relevant document zal er een conceptversie naar de begeleiders gestuurd worden, zodat deze op kwaliteit gecontroleerd kan worden. Verder kunnen de begeleiders aan de hand van gesprekken een beeld krijgen van de voortgang van het project. Mochten zij hierin valkuilen voorspellen of ontdekken, dan kan dit tijdens deze gesprekken aangegeven worden zodat de afstudeerder tijdig zijn werkwijze kan aanpassen

-Werkbegeleiding

Op de werkplek zullen de controles voornamelijk plaatsvinden op het gebied van documentatie en de uitwerking van het Proof of Concept. De documentatie zal net als voor de schoolbegeleiding in conceptfase besproken worden, voordat deze gefinaliseerd worden. Verder kunnen de softwarepakketten en Proof of Concepts regelmatig bekeken worden door medewerkers van i3D.net om hun mening continu mee te nemen in het ontwikkelen van het Proof of Concept.

2.6) Risicofactoren

De volgende factoren kunnen in de weg staan van het succesvol voltooien van het project:

Probleem	Gevolg	Mitigatie
Tijsduur	Doorlooptijd van 17 weken voor 1 fulltime stagiair kan mogelijk onvoldoende zijn om de opdracht geheel uit te voeren.	duidelijke planning aanhouden en op tijd ingrijpen bij problemen. Regelmatige meetings met begeleiders
Geen juiste oplossing	Geen van de beschikbare softwareoplossingen geschikt zijn voor dit project.	Op tijd een gesprek aangaan met de opdrachtgever om verder verloopt te bespreken.
Verwachtingsproblemen	Onjuiste verwachting ontstaan bij de opdrachtgever, eindresultaat niet zoals gewenst.	Regelmatig gesprekken aangaan met de opdrachtgever om voortgang bij te houden en het doel te bespreken.
Onvoldoende resources	Uitloop van project of onvoldoende onderzoek kunnen uitvoeren	Het op tijd inventariseren en aanvragen van resources. Bespreken met opdrachtgever indien nodig.

2.7) Kosten en baten

Dit project heeft de volgende kosten en baten.

Kosten

De stagiair zal gedurende 17 weken fulltime werk verrichten voor i3D.net, de bijbehorende te besteden manuren zijn als volgt.

[17 weken * 5 dagen] * 8uur = 680 uren. [minus officiële vaste vrije dagen]

Van deze 17 weken zal de stagiair vier dagen vast op het Datacenter van i3D.net aanwezig zijn, de laatste dag zal door de stagiair gebruikt worden om afspraken richting de Haagse Hogeschool te plannen en aan het afstudeer dossier te werken.

Verder zal de stagiair gebruik maken van de werkplek welke door de opdrachtgever beschikbaar is gesteld. Andere benodigde hardware omvat servers om het Proof of Concept op te hosten en eventuele testservers.

Na uitvoering van het project zal de opdrachtgever, indien ze de oplossing accepteren, verder de exploitatie kosten voor operatie en onderhoud moeten rekenen.

Baten

Onder de baten die i3D ondervindt van dit project vallen onder andere de volgende punten:

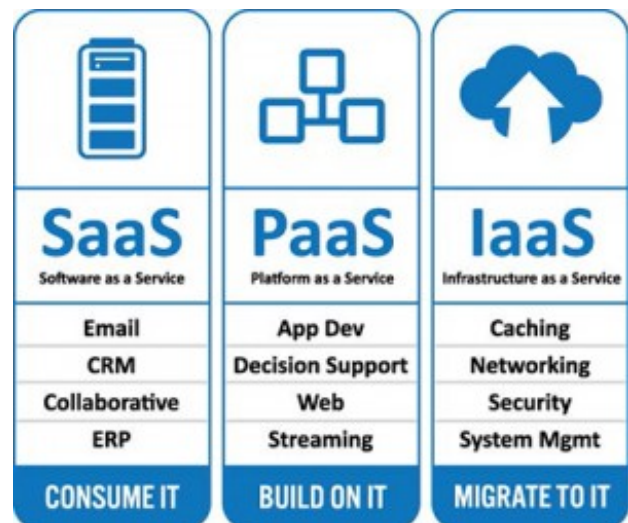
- Uitbreiden van dienstenaanbod voor klanten.
- Verhogen van efficiency waarmee deze diensten uitgevoerd worden.
- Hogere kwaliteit van bestaande diensten indien deze gekoppeld worden.
- Betere service aan klanten vanwege verhoogd gebruiksgemak.

3.) Design and Architecture

Voordat de stagiair begonnen is met het praktische aspect van de opdracht is er onderzoek gedaan naar de theorie achter cloud-hosting. De hoofdvraag was “Wat is Infrastructure as a Service?.” Daarna is onderzocht welke oplossingen er beschikbaar zijn om Infrastructure as a Service te kunnen leveren en wat er geschikt is voor de opdrachtgever. Het onderzoek is voornamelijk gebaseerd op de vergelijking van white-papers met gebruikerservaringen waarnaast er verscheidene onafhankelijke artikelen bekeken zijn over de verschillende marktleiders. Om alvast een beeld te krijgen van de communities en de documentatie is er ook gekeken naar de officiële wiki's ^[11](en vergelijkbare kennisbanken) van de producten.

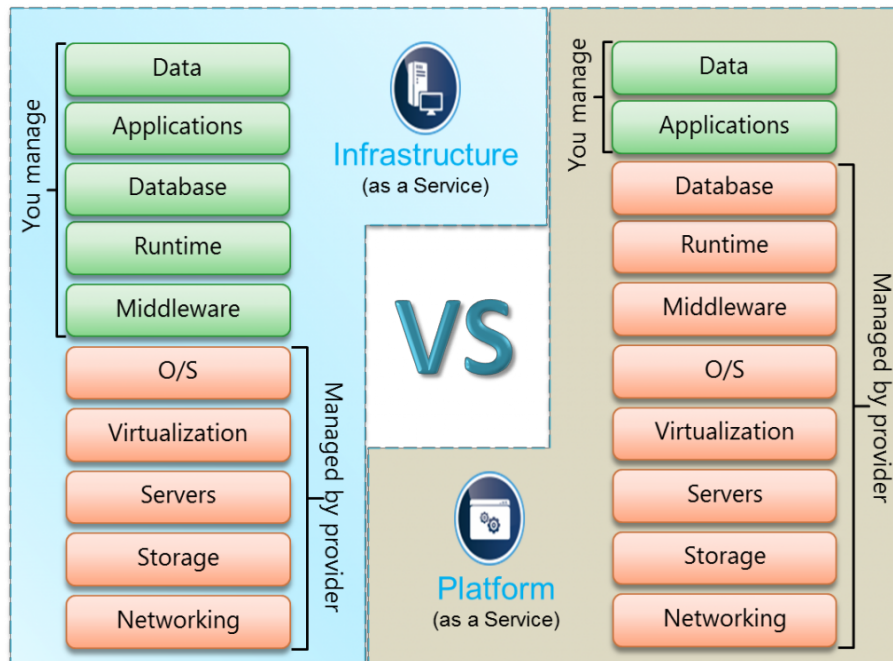
3.1) Wat is Infrastructure as a Service?

Voordat er verder onderzoek uitgevoerd kan worden naar IaaS software oplossingen moet er eerst onderzocht worden wat IaaS precies is. Ten eerste staat IaaS voor ‘Infrastructure as a Service.’ IaaS is één van de drie fundamentele methoden waarop cloud computing diensten worden aangeboden. De andere twee fundamentele methoden zijn respectievelijk PaaS (Platform as a Service) en SaaS (Software as a Service). Wat de opdrachtgever met een IaaS oplossing aan hun klanten aanbiedt is een gevirtualiseerde computing omgeving welke door middel van het internet te bereiken is.



Bij deze computing omgeving moet gedacht worden aan gevirtualiseerde machines waarbij de CPU's, RAM en andere fysieke specificaties virtueel gealloceerd worden. Verder worden ondersteunende diensten zoals externe storage, netwerkverbindingen, bandbreedte, IP adressen en loadbalancers ook gezien als diensten onder IaaS. Het hosten van deze omgeving gebeurt op een fysieke backend, welke uit meerdere servers en netwerken bestaat, verspreid over meerdere datacenters. Dit in tegenstelling tot SaaS, waarbij een groot deel van het managen van de software backend ook gedaan wordt door de provider.

Met een IaaS opstelling wordt een efficiënt en makkelijk schaalbaar netwerk opgezet waar de complexe netwerk werkzaamheden en de kosten van het onderhouden hiervan bij de cloud provider ligt.



Dit zorgt voor een oplossing waar klanten hun omgeving kunnen opschalen indien dit nodig is, zonder dat ze de eenmalige aanschafkosten hoeven te betalen voor permanente extra capaciteit. Er kan simpelweg voor een aantal uren/dagen/weken extra capaciteit bij gekocht worden waarna dit weer gemakkelijk opgezegd kan worden.

De voordelen die klanten ondervinden van IaaS zijn dus als volgt:

- **Schaalbaarheid**

Er is capaciteit beschikbaar wanneer de klant dit nodig heeft, zonder vertraging van orders.

- **Geen hardware investering**

Klanten hoeven niet op grote hardware kosten te rekenen, deze vallen voor de hosting provider.

- **Pay-as-you-go**

Je betaalt voor hoelang je gebruik maakt van je diensten, dus niet als deze inactief zijn.

- **Overal bereikbaar**

Zolang de klant over een internetverbinding beschikt, is zijn omgeving altijd beschikbaar

- **Fysieke beveiliging van datacenters**

Het datacenter zorgt ervoor dat alle fysieke beveiliging geregeld is.

- **Redundantie**

Switchen en servers worden door de provider redundant uitgevoerd.

3.2) Software onderzoek: Welke IaaS oplossingen zijn er beschikbaar voor i3D.net?

Om te starten met het onderzoek is er gekeken welke IaaS oplossing er op het moment beschikbaar zijn. Dit is gedaan door te kijken naar huidige artikelen over Cloud software^[4,5] en onderzoek te doen naar welke namen er onder de techneuten regelmatig vallen.^[6] Gezien de subjectiviteit van deze bronnen word de shortlist van namen hierna nog individueel bekeken om te kijken of ze bij de omgeving van i3D.net passen



Een open-source pakket om cloud hosting te faciliteren met een sterke focus op integratie op AWS public cloud.^[7]



Gestart als VMOPs en later overgenomen door Citrix waarna het project opgenomen is aan het Apache incubator programma.^[8]



Het Openstack project is een open-source pakket wat geheel open is en voor iedereen toegankelijk is om aan deel te nemen. Het project wordt beheerd door de Openstack Foundation welke afhankelijk is van de contributies van organisaties welke zich aansluiten bij de Foundation.^[9,10]

Alle genoemde pakketten zijn functioneel geschikt om een public/private cloud op te richten. Dit was echter niet de enige eis van de opdrachtgever. Na gesprekken met de opdrachtgever is gebleken dat het vooral gaat om een pakket dat open-source is met een gedreven community. De redenen hiertoe zijn als volgt: Het gaat om een budget alternatief voor de Azure cloud (High Performance) waarin klanten tegen lagere prijzen toch IaaS diensten kunnen afnemen. Vanwege de budget prijs van deze cloud is er daarom gekozen om de cloud zo voordelig mogelijk op te zetten. Dit betekent dat het laten opbouwen en managen van de cloud met hulp van een derde partij direct afgevallen is. Het gevolg van deze richting is dat de cloud zelf gebaseerd wordt op een best-effort model bij het leveren van de diensten. De cloud kan op deze manier goedkoper aangeboden worden tegenover de klanten ten koste van de mogelijkheid om problemen met de cloud met SLA garanties op te lossen.

Verdere eisen haken in op de werkwijze van de huidige i3D.net diensten. Deze werkt met een geheel custom administratie panel waarin vrijwel alle acties met betrekking tot de geleverde diensten uitgevoerd worden. Denk hierbij aan het reageren op tickets van klanten, tot het aansturen van servers in het datacenter. Idealiter wil de opdrachtgever deze cloud kunnen koppelen aan het panel via API's zodat deze ook vanuit het eigen admin panel beheerd kan worden. Dit betekent dat er dus ruime API support beschikbaar moet zijn in het gekozen pakket.

Uiteindelijk waren er nog een aantal kleinere functionele eisen. Deze zijn in het Proof of Concept document besproken en gaan dieper op de techniek achter Openstack in.

Om een pakket te kiezen uit de shortlist van drie pakketten zijn er eerst twee analyses uitgevoerd, een community- en een functieanalyse.

3.2.1)Community-analyse

Open-source projecten hebben de eigenschap dat ze community-driven zijn. Er worden projecten opgezet waarbij gehoopt wordt dat de community het project interessant genoeg vindt om hier zelf aan bij te dragen. Het idee hierbij is dat hoe meer mensen er naar een open platform kunnen kijken en bijdragen, hoe sneller problemen hierin gevonden en opgelost kunnen worden. Een probleem dat hiermee gemeden wordt is als volgt: Een closed-source software provider heeft software draaien bij meerdere bedrijven, een kwaadwillende derde partij vindt hier een kwetsbaarheid in de beveiliging. Het lek komt aan het licht en de gegevens van de bedrijven zijn niet meer veilig. Voor de bedrijven zit er niets anders op dan te wachten tot de software provider een patch heeft uitgebracht om het lek te dichten. Wanneer een dergelijk lek bij open-source software was voorgekomen, dan was de kans groter dat het lek al gevonden kon worden voordat er mogelijk misbruik van gemaakt zou worden. Mocht er alsnog een lek ontdekt worden, dan kunnen gebruikers zelf de code inzien van de software en dit waar nodig zelf patchen en oplossen. Hierdoor is de tijd tot het probleem opgelost potentieel veel lager.

Naast het oplossen van security-issues is de insteek van gebruikers erg waardevol voor een open-source project. Mochten meerdere mensen problemen hebben met bepaalde functionaliteit, dan kunnen zij samenwerken om deze functionaliteit te verbeteren of alternatieve functionaliteit in het product in te bouwen. De community is dus de drijvende kracht achter veel open-source pakketten; vandaar dat het eerste punt waar de drie pakketten op beoordeeld zijn de community is.

Bij het verzamelen van informatie is het onderzoek “Open Source IaaS Community Analysis - Openstack vs OpenNebula vs Eucalyptus vs Cloudstack” van Qingye Jiang^[13] geraadpleegd. Dit is een objectief onafhankelijk onderzoek naar de communities achter de respectievelijke open source producten. Dit onderzoek geeft een duidelijk inzicht op de huidige staat en groei van de communities aangezien dit onderzoek sinds 2011 per kwartaal uitgevoerd wordt. Er is duidelijk te zien op wat voor manier de gekozen pakketten zich ontwikkeld hebben over de afgelopen vierjaar. Deze informatie is waardevol om te voorspellen of het project de komende tijd levendig blijft of dat het project simpelweg niet van de grond af komt.

De informatie uit het onderzoek van Qingye Jiang is bovendien goed te combineren met een evaluatie van een succesvolle life cycle van Open-source projecten. Verschillende onderzoeken beschrijven de life cycle van Open-source project globaal als volgt:

● **Phase 1: Concept**

Er bestaat een idee waar een aantal mensen basisfunctionaliteit voor geschreven hebben en ideeën over beschrijven op een project website (Google Code, Codeplex etc.). Minimale documentatie is beschikbaar, vooral concepten.

● **Phase 2: Bootstrap**

Een paar mensen zullen serieus aan het product werken en hiermee experimenteren. Een kleine hoeveelheid documentatie kan beschikbaar zijn. De functionaliteit verschilt per omgeving.

● **Phase 3: Early Development**

Er worden door meerdere mensen kleine contributies gedaan naast de hoofd-contributors. Er wordt meer feedback verkregen door de groeiende groep geïnteresseerde en het project functioneert in meer omgevingen.

● **Phase 4: Early Adoption**

Vanwege een groeiend aantal deelnemers groeien ook de contributies. Het project beschikt inmiddels over een groot deel van de basisfunctionaliteit en is bruikbaar. Er wordt meer over gesproken op blogs en de kennisbasis wordt uitgebreid. Er wordt meer gewerkt aan de toegankelijkheid door het documenteren van API's, FAQ's en Getting Started guides.

● Phase 5: Development

Meer deelnemers zijn serieus bezig met het project waardoor de core features sneller ontwikkeld worden. Er worden automatische builds opgezet waarnaast regelmatig binaries beschikbaar zijn, wat gepaard gaat met groeiende bekendheid. De wiki is op dit punt uitgebreid en bruikbaar en begint verschillende scenario's te beschrijven.

● Phase 6: Adoption

Meer mensen dragen bij in de vorm van patches en andere kleine toevoegingen, de user base groeit naar een paar honderd mensen en er wordt een regelmatig release schedule opgezet. Er worden gedetailleerde artikelen geschreven over de werking van functies en de wiki ziet regelmatig bijdragen.

● Phase 7: Maturity

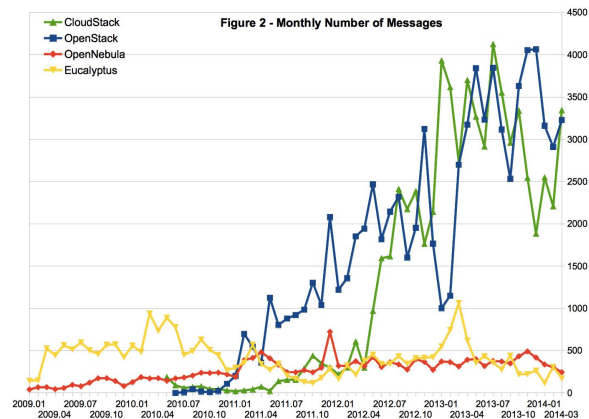
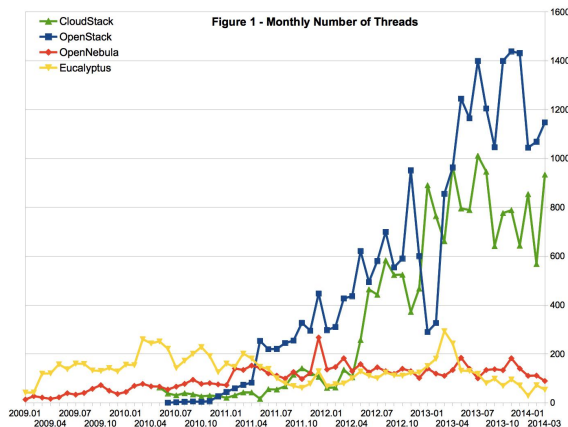
Vanwege de hoeveelheid patches en contributors wordt er een systeem opgezet om bijdragen en patches te evalueren en goed te keuren. Er worden releases opgezet voor verschillende omgevingen en bijdragen projecten worden samengevoegd.

● Phase 8: Mainstream

Tegen deze tijd zijn de originele contributors verder gegaan naar andere projecten of minder betrokken. Het project zelf groeit enorm in verschillende richtingen en consultancy bedrijven pakken het product op om er commerciële support op te bieden. Het project staat stevig en zorgt voor duidelijke toegevoegde waarde. De documentatie rondom het project groeit consistent met artikelen, handleidingen, blogs en evaluaties.

Door de pakketten te vergelijken met de fasen welke hierboven beschreven worden is een idee te krijgen van op welk punt in zijn levenscyclus het pakket is en wat te verwachten is van de toekomst. Met deze informatie is er gekeken naar de drie te evalueren pakketten. De belangrijkste punten uit het onderzoek van Qingye Jiang worden hier besproken.

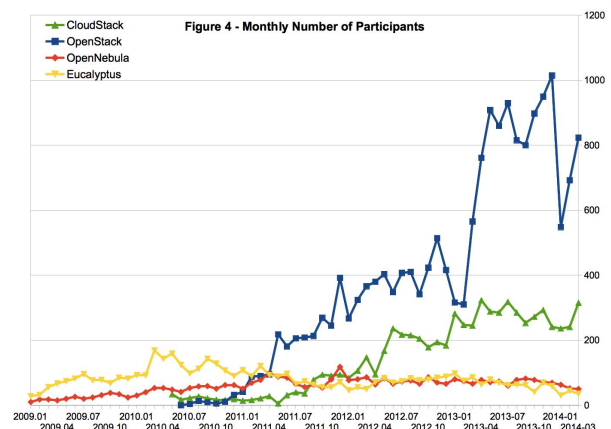
Threads en Messages



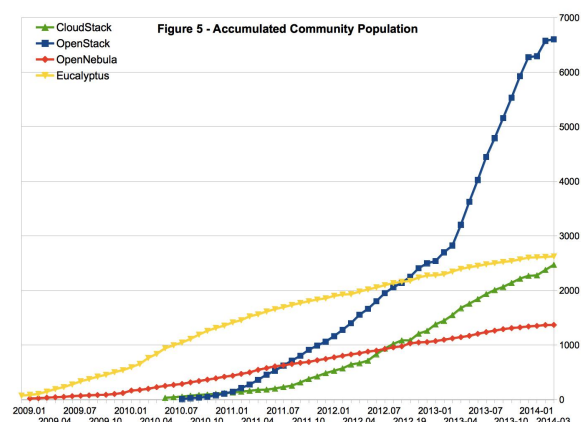
De bovenstaande grafieken (figuur 1 en 2) geven weer hoe actief de communities zijn wat betreft het posten van threads en messages. Er is hier duidelijk te zien dat de communities voor Cloudstack en Openstack sterk gegroeid zijn de afgelopen vier jaar. Alhoewel de Eucalyptus al langer bestaat dan Cloud- en Openstack is de activiteit van de community vrij constant gebleven tegenover de sterk toenemende activiteit van de concurrenten.

Participants en population

Deze gegevens worden interessanter als er gekeken wordt naar het aantal daadwerkelijke deelnemers aan de community welke in figuur 4 weergegeven wordt. In deze grafiek is te zien dat Openstack een groter aantal participants heeft ten opzichte van Cloudstack, ongeveer 200% meer, en Eucalyptus, ruim 800%. In 2013 heeft met name Openstack een grote groei gekend Cloudstack heeft daarentegen al een aantal jaren een consistente kleine groei. Ten slotte heeft Eucalyptus gedurende de afgelopen vier jaar een consistent ratio participants gehad, welke bij de opkomst van Openstack lijkt af te nemen.

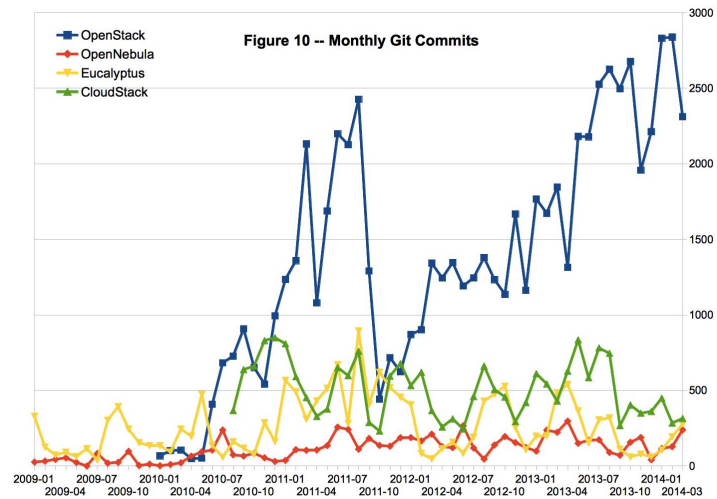


In Figuur 5 is tevens de groeiende interesse in Openstack te zien: deze grafiek geeft de totale community grootte weer. Er is duidelijk te zien dat de interesse in Openstack sinds 2013 sterk toegenomen is, waarbij Cloudstack op ongeveer de helft van deze groei zit boven de stagnerende groei van Eucalyptus.



Git contributie

Ten slotte zijn de maandelijkse git commits interessant. Uit de hoeveelheid Git commits is namelijk af te leiden hoeveel suggesties en toevoegingen er worden bijgedragen aan de code van een project door ontwikkelaars en leden van de community. In figuur 10 is te zien dat Openstack wederom actiever is dan de andere kandidaten. Eucalyptus en Cloudstack hebben beide een redelijk constante stroom commits. Als er echter vergeleken wordt met de hoeveelheid commits welke op Openstack uitgevoerd wordt, ligt deze significant lager. Naast een groter volume commits is er ook een duidelijk stijgende lijn te zien in het aantal commits voor Openstack: dit is conform met de stijgende hoeveelheid contributors welke in de eerdere grafieken weergegeven werd.



Conclusie community analyse

Als we kijken naar de gegevens in de bovenstaande grafieken, deze combineren met de informatie over Open-source lifecycles en de huidige staat van pakketten, dan kunnen er een aantal conclusies getrokken worden.

Ten eerste is het duidelijk dat als er naar volume word gekeken Openstack ver boven de competitie uitgegroeid is. Met een userbase welke groter is dan de andere pakketten gecombineerd, een groter aantal contributors en een constant groeiende community is Openstack wat dit betreft duidelijk de beste keuze.

Als we kijken naar op welk punt Openstack van zijn lifecycle is, kan er veilig gezegd worden dat Openstack zich tussen Maturity en Mainstream bevindt. Grote namen als Walmart, Paypal en NASA^[12,13] hebben zich aangesloten bij Openstack, er zijn communities dedicated aan het ontwikkelen van de verschillende sub-projecten van Openstack, consultancy en hosting bedrijven beginnen zich steeds meer te verdiepen in het aanbieden van Openstack en er zijn verschillende vertakkingen van Openstack ontstaan (Packstack, devstack, Helion, Mirantis). In deze fase kan er aangenomen worden dat het project zich zonder grote risico's verder zal ontwikkelen en dus een solide keuze zal zijn.

3.2.2) Functieanalyse

Naast de activiteit van de project is het belangrijk dat het project geheel voldoet aan de functionele eisen van i3D.net. Hiervoor zijn de specificaties per pakket naast elkaar gezet om dit te bepalen. Verdere functionaliteit is lastig in te schatten op theoretische basis en zal tijdens het Proof of Concept behandeld worden.

In deze sectie wordt er ingegaan op de theoretisch ondersteunde functionaliteit van de pakketten. Indien de volledige functionaliteit van een pakket getest wordt, moet deze voornamelijk tegenover de eisen van i3D.net getest worden in een praktische omgeving. Om dit te realiseren moet er een complete testomgeving opgezet worden welke vervolgens getest wordt op de gestelde eisen. In dit project zou dit teveel tijd en resources kosten waardoor gekozen is voor één project gebaseerd op de globale eisen een testomgeving op te bouwen.

De punten die de opdrachtgever als belangrijk heeft aangegeven, zijn:

- Hyper-V compatibiliteit
- AWS compatibiliteit
- API's zodat functionaliteit ingebouwd kan worden in het admin panel van i3D.net

Verder zijn alle pakketten vooraf gekozen op de algemeen criteria dat deze in staat zijn om private en public clouds te realiseren.

Door kritisch te kijken naar de beschikbare white papers en feature sheets is de volgende tabel opgesteld

	Functionaliteit	Hyper-V	AWS compatibiliteit	API functionaliteit
Pakket				
Eucalyptus		--	++	+
Cloudstack		+	++	++
Openstack		++	++	++

Aan de hand van de bovenstaande eisen is dus te zien dat Eucalyptus vrijwel direct afvalt door de vaste requirement van Hyper-V support. Verder zijn over het algemeen alle pakketten redelijk uitgebreid op het gebied van functionaliteit. Het doorslaggevende voor i3D is echter of de functionaliteit aan te passen is naar wens. In alle gevallen is dat in zekere mate mogelijk. Openstack heeft echter een toegewijde community die constant streeft naar het uitbreiden van de userbase en het betrekken van de community bij de development.

Conclusie Functie Analyse

Aan de hand van de globaal verwachte functionaliteit is Openstack de meest geschikte keuze. Alhoewel Cloudstack bijna dezelfde functionaliteit biedt als Openstack, was er veel onduidelijkheid over de beloofde functionaliteit en in hoeverre deze beschikbaar was. Omdat transparantie erg belangrijk is en er geen tijd is om voor onverwachte problemen tijdens het project te staan, is er daarom gekozen voor Openstack. Openstack biedt een duidelijk beeld van wat er te verwachten valt indien Openstack in gebruik genomen wordt en geeft aan wat er in ontwikkeling is met een planning voor verwachte implementatie.

3.3) Advies: Welk pakket is geschikt voor i3D.net?

Na het bekijken van de functionaliteit en de community's van alle project kan geconcludeerd worden dat voor i3D.net het Openstack pakket het beste past. Dit pakket heeft een grote en actieve community welke ondersteund wordt door grote toonaangevende bedrijven. De community heeft inmiddels een vaste development cycle opgezet en heeft in principe een vaste groep developers welke aan hun eigen sub-projecten werken. De actieve community en snelle development is voordelig voor i3D.net in verband met de snelle patches voor problemen in Openstack en fixes voor kritieke issues.

De openheid van Openstack zorgt er tevens voor dat indien i3D genoeg in het project ziet, zij hier hun eigen development team aan kunnen toewijden om zelf bij te dragen aan het project. Om deze redenen is er dus besloten om met Openstack verder te werken aan het Proof of Concept.

4.) Development and Coding

In het komende hoofdstuk wordt uitleg geboden over het gekozen software pakket en de werking hiervan. Er zal eerst beschreven worden hoe het pakket in elkaar steekt. Hierna zal het opzetten van het Proof of Concept beschreven worden met uitleg over hoe dit gerepliceerd kan worden. Met dit document zal er kennis verkregen kunnen worden over de werking van Openstack en hoe hiermee een Proof of Concept omgeving opgebouwd en uitgebreid kan worden tot productie.

4.1) Openstack introductie

Openstack is een geheel gratis en open-source cloud operating systeem. Hiermee kunnen pools van computing kracht, storage en networking capaciteit, verspreid door datacentra, beheerd worden door middel van een centraal dashboard. Via dit centrale dashboard kunnen administrators hun gehele Openstack cloud beheren en hebben de gebruikers de mogelijkheid om hun eigen capaciteit te provisionen door een web interface. Naast de web interface is er de mogelijkheid om te managen via command-line tools of de RESTful API; voornamelijk de laatste twee zijn interessant voor bedrijven zoals Openstack.

Openstack is gestart in 2010 als een coöperatief project tussen Rackspace Hosting en NASA. Het doel was om een Open Source cloud computing platform op te zetten waarmee public en private clouds beheerd kunnen worden, ongeacht hoe groot deze zijn. Het doel is simpel in gebruik te zijn en gemakkelijk schaalbaar te zijn. Na de eerste release in 2010 is de community gegroeid tot een community van meer dan 9000 contributors in 90 landen met support van meer dan 150 bedrijven. Een aantal van de grote bedrijven achter Openstack zijn Red Hat, Canonical, IBM, AT&T, Cisco, Intel, Paypal en Comcast. Deze community werkt volgens een 6-month time-based release cycle waarbij tijdens de planningsfase van elke cycle de community bijeenkomt om de plannen voor de komende release te bespreken.

In de vijf jaar dat Openstack inmiddels in development is zijn er een aantal features bij gekomen. Een korte release history is als volgt:

Release Name	Release Date	Included Components / Projects
Austin	Oct 21, 2010	Nova, Swift
Bexar	Feb 3, 2011	Nova, Glance, Swift
Cactus	Apr 15, 2011	Nova, Glance, Swift
Diablo	Sep 22, 2011	Nova, Glance, Swift
Essex	Apr 5, 2012	Nova, Glance, Swift, Horizon, Keystone
Folsom	Sep 27, 2012	Nova, Glance, Swift, Horizon, Keystone, Quantum , Cinder
Grizzly	Apr 4, 2013	Nova, Glance, Swift, Horizon, Keystone, Quantum , Cinder
Havana	Oct 17, 2013	Nova, Glance, Swift, Horizon, Keystone, Neutron , Cinder
Icehouse	Apr 2014	Nova, Glance, Swift, Horizon, Keystone, Neutron , Cinder, More

Een van de interessante punten aan Openstack is dat het modulair opgezet is. De huidige Openstack release (Juno) bestaat uit elf componenten welke individueel ontwikkelt worden en waar nog vijf nieuwe componenten van in development zijn. De huidige lijst bestaat uit de volgende componenten:

- OpenStack Compute (code-name Nova)
- OpenStack Networking (code-name Neutron)
- OpenStack Object Storage (code-name Swift)
- OpenStack Block Storage (code-name Cinder)
- OpenStack Identity (code-name Keystone)
- OpenStack Image Service (code-name Glance)
- OpenStack Dashboard (code-name Horizon)
- OpenStack Telemetry (code-name Ceilometer)
- OpenStack Orchestration (code-name Heat)
- OpenStack Database (code-name Trove)
- OpenStack Data Processing (code-name Sahara)

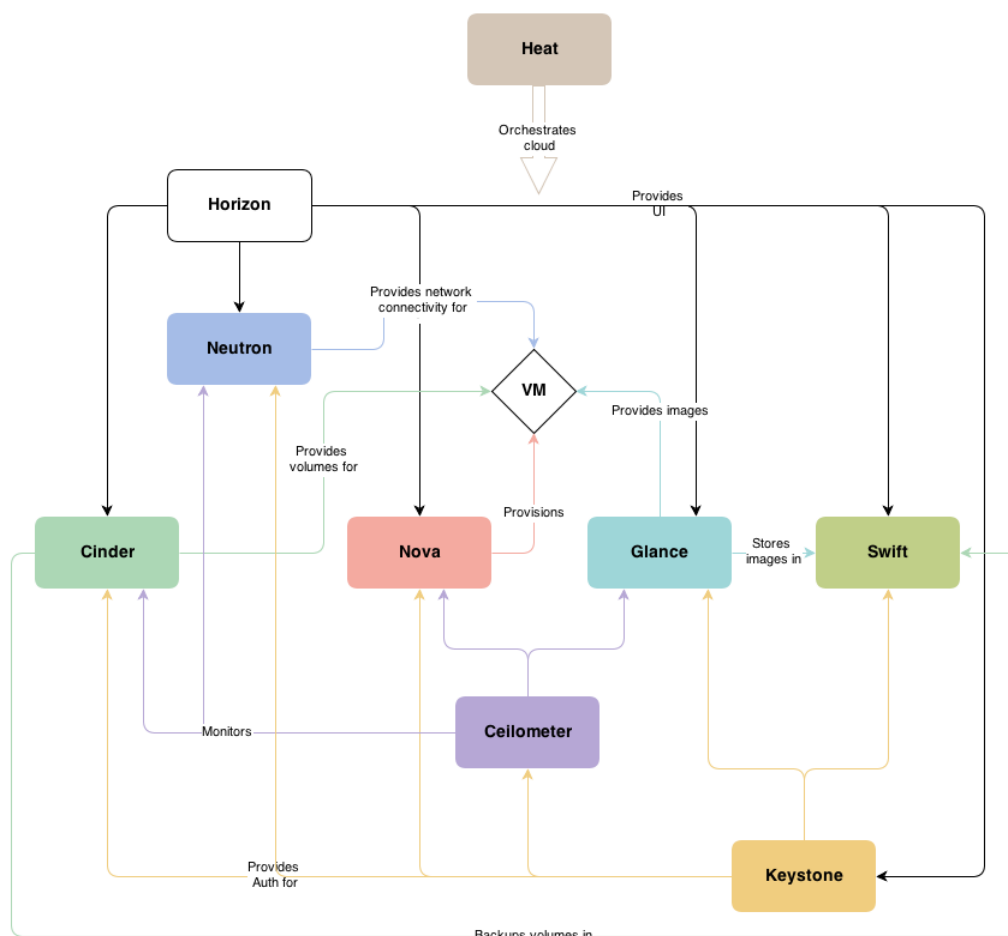
4.2) Het Proof of Concept ontwerpen en opzetten

Alle componenten hebben een specifieke taak binnen het Openstack geheel. Alhoewel alle componenten hun eigen functionaliteit bieden aan Openstack hoeven niet alle componenten verplicht geïmplementeerd te worden. Zo begon het Openstack project origineel als puur Nova en Swift, waarin alleen van computing en storage gebruikt gemaakt werd. Om te begrijpen welke onderdelen bij i3D van toepassing zijn is er gekozen om alle componenten eerst individueel te bekijken en te beoordelen.

Componenten

Aangezien Openstack uit een aantal verschillende componenten bestaat met elk unieke functionaliteit zal ik deze per component nader verklaren om een duidelijker beeld te geven bij het gehele Openstack product. In dit hoofdstuk zal alleen de globale functionaliteit beschreven worden. De in-depth installatie en functies zullen later besproken worden.

Hieronder is een conceptuele architectuur te zien van Openstack: dit laat zien hoe alle componenten in relatie tot elkaar staan. Alle componenten vormen samen een ideale Openstack situatie; echter zijn niet alle componenten nodig voor een functionele installatie. Zo kan bijvoorbeeld Ceilometer weggelaten worden indien chargeback geen vereiste is (bijvoorbeeld in private clouds).



De componenten hebben samengevat de volgende functionaliteit:

Hoofd componenten

- Keystone
 - Identity Management voor Openstack services
- Nova
 - Hypervisor management
- Glance
 - Images discoveren, registeren en retrieven
- Neutron
 - Gevirtualiseerde networking en connectiviteit
- Swift
 - Object storage voor Virtuele Machines
- Cinder
 - Block storage voor Virtuele Machines
- Heat
 - Geautomatiseerd Up- en Downscalen van infrastructuur
- Trove
 - Database as a Service
- Ironic
 - Bare-metal machines provisioning
- Sahara
 - Big data processing (Hadoop & Spark)
- Zaqr
 - Multi-tenant cloud messaging service
- Barbican
 - Key storage, management en provisioning
- Designate
 - DNS as a Service
- Manila
 - Shared persistent filestorage

Supportende componenten

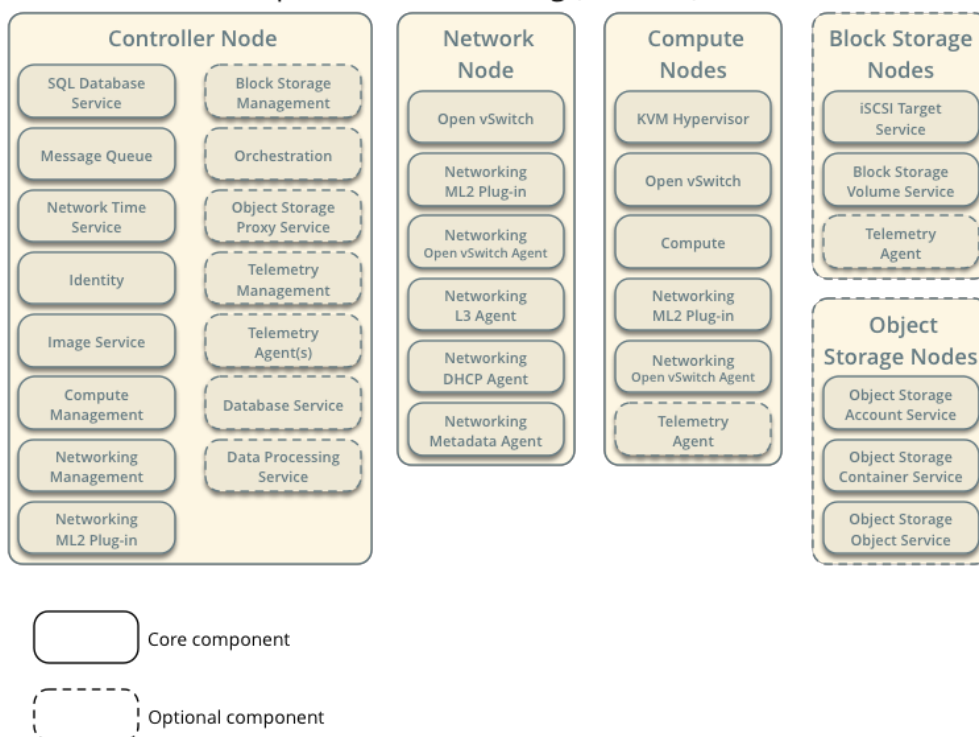
- Horizon
 - Online cloud management interface
- Ceilometer
 - Chargeback functionaliteit voor Openstack

Architectuur

Om een Proof of Concept op te zetten is er gekeken naar de eisen van i3D.net en wat de documentatie van Openstack zelf aanraadt.^[14] Met dit Proof of Concept kan gecontroleerd worden of Openstack voldoet aan de gestelde verwachtingen van dit project.

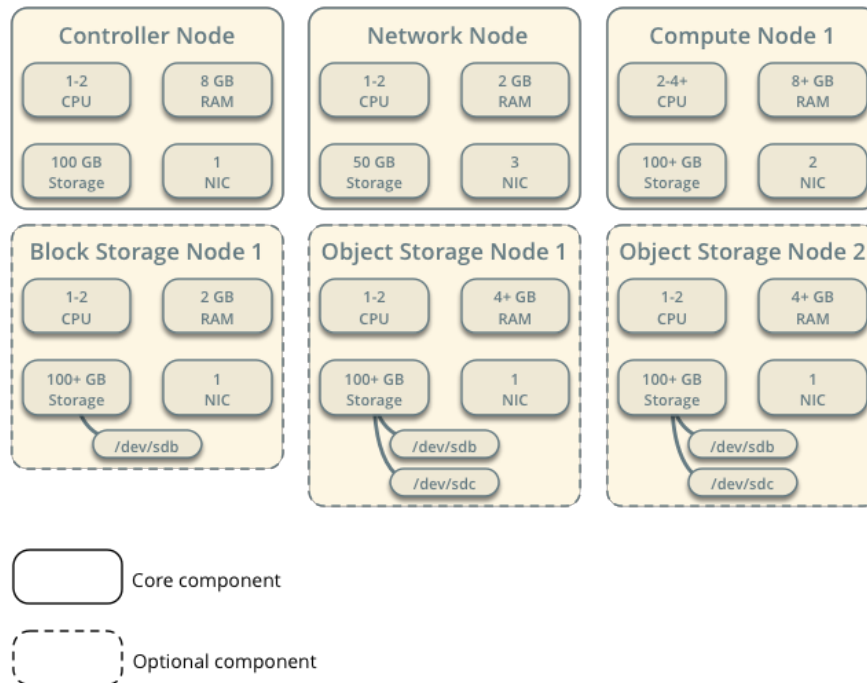
De services van Openstack hoeven niet allemaal hun eigen node te krijgen in een Openstack omgeving; vooral in een Proof of Concept omgeving zou dit erg veel hardware kosten om een testopstelling te bouwen. Om deze reden is er gekeken naar welke services aangeraden worden om te bundelen op één node. Op deze manier kan er een Proof of Concept op gezet worden met minimale hardware om toch een duidelijk beeld van Openstack geeft.

Minimal Architecture Example - Service Layout OpenStack Networking (neutron)



Zoals hierboven te zien is wordt er aangeraden een Proof of Concept te verdelen over drie nodes, met eventueel extra nodes voor de storage. Er is gekozen om voor het Proof of Concept alle core services op te zetten en mocht de tijd dit toestaan deze uit te breiden met optionele componenten.

Minimal Architecture Example - Hardware Requirements OpenStack Networking (neutron)



In de bovenstaande afbeelding is te zien wat er minimaal verwacht wordt om Openstack te draaien. Er wordt fysiek minimaal één Controller node, één Network node en één Compute node aangeraden. De servers welke door i3D.net beschikbaar zijn gesteld zijn drie HP ProLiant DL120's en drie HP ProLiant DL160's. De gemiddelde specificaties van deze servers zijn als volgt:

HP ProLiant DL120 G6

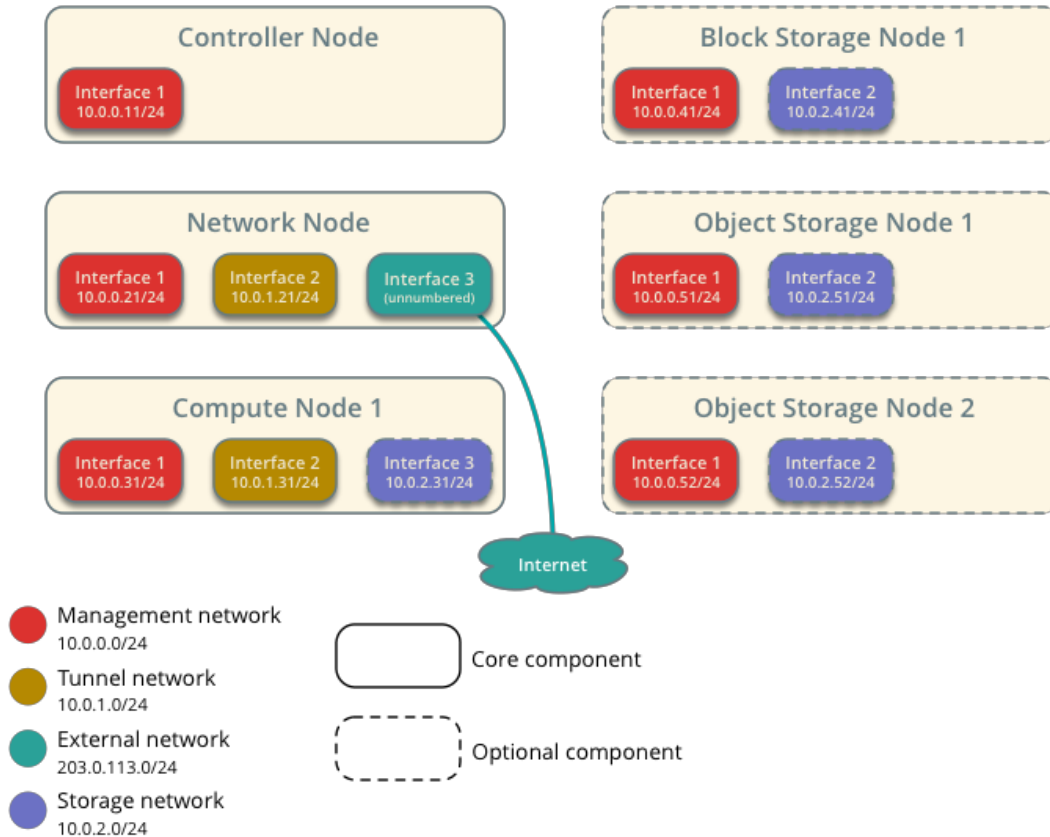
Processor: Intel(R) Xeon(R) CPU X3430 @ 2.40GHz
Memory: 4GB
Storage: 1x 250GB

HP ProLiant DL160 G6

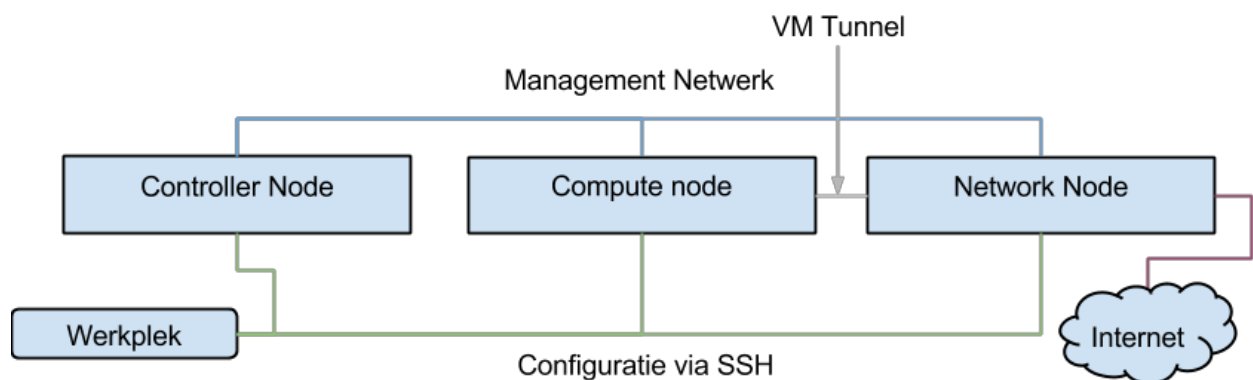
Processor: 2x Intel(R) Xeon(R) CPU L5630 @ 2.13GHz
Memory: 16GB
Storage: 1TB

De servers zijn verbonden met 1Gbit links aan het i3D.net netwerk. De nodes zijn op basis van specificaties ingedeeld op de beschikbare servers. De controller en compute nodes zijn op DL160's geplanned, de Network node is op een DL120 geplanned.

Minimal Architecture Example - Network Layout OpenStack Networking (neutron)



Om de connectiviteit tussen de nodes te realiseren wordt bovenstaande configuratie aangeraden. In het netwerk van i3D.net is deze opstelling gerealiseerd door middel van gebruik te maken van VLANs op het netwerk. Op deze manier is het netwerkverkeer gescheiden van elkaar conform de specificaties van Openstack. De configuratie ziet er uiteindelijk als volgt uit:



Configuratie

De configuratie van de services zal in dit document in het kort besproken worden. De configuratie van één van de opgezette services zal beschreven worden om een beeld van het proces te geven. Omdat de opvolgende services redelijk gelijk geconfigureerd kunnen worden zullen deze in dit document niet verder besproken worden.

Om te beginnen is er gekozen voor het Ubuntu Operating System als basis van Openstack. De reden hiervoor is dat deze versie van Openstack de grootste community heeft en er een grote hoeveelheid kennis en ervaring is over Ubuntu bij de stagiair. De keuze van OS heeft buiten voorkeur van de beheerder geen grote impact op Openstack zelf. Hierom is er dus ook geen onderzoek gestart naar welke versie de beste keuze zou zijn gezien het bedrijf bekend is met Ubuntu en de stagiair hier zelf ook ervaring mee heeft.

Algemene configuratie:

Wachtwoorden

Voordat er begonnen wordt met het configureren van verscheidene services zijn er eerst wachtwoorden gegenereerd welke vervolgens gebruikt kunnen worden voor verscheidene config files. Om veilige wachtwoorden te generen is er op een Ubuntu server gebruikt gemaakt van het commando "*openssl rand -hex 10*". Op deze manier worden er tien hexadecimale wachtwoorden gegenereerd. Deze zijn opgeslagen in een Keeppass database om veilig te kunnen beheren.

Tijd synchronisatie

Om de tijd consistent te houden tussen de nodes is er gebruik gemaakt van het NTP protocol. De Control node synchroniseert intern met de NTP server van i3D.net waarna de andere nodes zich kunnen synchroniseren met de Control node .

Repositories

Verder wordt op alle nodes de Openstack repository toegevoegd en up to date gehouden door de volgende commando's uit te voeren per node:

```
# apt-get install ubuntu-cloud-keyring
# echo "deb http://ubuntu-cloud.archive.canonical.com/ubuntu" \ "trusty-updates/juno main"
> /etc/apt/sources.list.d/cloudarchive-juno.list
# apt-get update && apt-get dist-upgrade
```

Database

Vrijwel alle services van Openstack maken gebruik van een Database. Hiervoor wordt MariaDB gebruikt. Deze database is op de Controller node geplaatst zonder gebruik te maken van uitzonderlijke configuratie opties omwille van Openstack.

Messaging server

Openstack gebruikt een message broker om onder de services de operaties en informatie te coördineren; deze broker draait meestal op de controller node. Omdat de meeste Openstack services getest zijn op RabbitMQ ondersteuning zal deze gebruikt worden.

```
# apt-get install rabbitmq-server
```

Toevoegen van Nodes

Om een node aan Openstack toe te voegen worden een aantal stappen doorlopen die voor alle nodes vrijwel hetzelfde zijn. Hierom zal er dus één node besproken worden in dit document. Voor details over de overige nodes kan de Proof of Concept documentatie geraadpleegd worden.

Er zal begonnen worden met de Identity service: dit is de service welke ervoor zorgt dat alle andere diensten zich kunnen authenticeren binnen Openstack.

Ten eerste wordt er een database opgezet. Dit gebeurt door naar de database te verbinden:

```
"mysql -u root -p"
```

Waarna er een database voor de service wordt opgezet :

```
"CREATE DATABASE keystone;"
```

Ten slotte wordt ervoor gezorgd dat de dienst zijn eigen database kan bereiken door de volgende commando's uit te voeren, KEYSTONE_DBPASS wordt vervangen door één van de eerder gegenereerde wachtwoorden.

```
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'localhost' \
  IDENTIFIED BY 'KEYSTONE_DBPASS';
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'%' \
  IDENTIFIED BY 'KEYSTONE_DBPASS';
```

Het installeren van de client zelf gebeurt door middel van het commando:

```
apt-get install keystone python-keystoneclient
```

De client moet vervolgens geconfigureerd worden voor het huidige project. De details hiervan worden in het Proof of Concept document verder besproken.

Ten slotte wordt de database van de service klaargemaakt door het commando.

```
su -s /bin/sh -c "keystone-manage db_sync" keystone
```

En wordt de service herstart om de config veranderingen in te laten gaan.

```
service keystone restart
```

Na het installeren en het voltooien van de basisconfiguratie worden de service specifieke opties geconfigureerd. Deze worden verder besproken in de documentatie over het Proof of Concept zelf. Gezien het opzetten van de opvolgende services vrijwel hetzelfde gebeurt als Keystone zullen deze ook in dit document niet besproken worden.

Na het opzetten van het Proof of Concept is er gekeken wat er wel en niet mogelijk is met Openstack. In de basisopstelling welke voor dit project gebouwd is is gebleken dat alle verwachte basisfunctionaliteit inderdaad werkte zoals verwacht. Zonder problemen zijn standaard functies als netwerken aanmaken, Virtuele machines aanmaken en machines onderling later communiceren getest.

Wat echter wel geleerd is van het opzetten van het Proof of Concept is dat een automation oplossing een vereiste is om Openstack in productie te nemen. Voor een Proof of Concept opstelling is het geen probleem geweest alle nodes met de hand te configureren, zodra de productie opgezet word is het echter sterk aan te raden om dit proces te automatiseren. Dit is te realiseren met automation applicaties als Chef, Puppet en Salt. Het automatiseren van het uitrollen van nodes spaart na een aantal keer uitgevoerd te worden de tijd uit om een automatiserings oplossing te realiseren, waarbij het automation platform hiernaast ook ingezet kan worden voor de andere diensten binnen i3D.net

4.3) Is het Openstack Proof of Concept geslaagd?

Het opzetten van een Openstack Proof of Concept is uiteindelijk succesvol voltooid. Na wat uitgebreider dan verwacht onderzoek moeten doen naar de documentatie van Openstack is het Proof of Concept opgezet en functioneel bevonden.

In de Proof of Concept opstelling is er de mogelijkheid op basis functionaliteit te testen welke verwacht wordt van Openstack. Deze opstelling is vervolgens als testomgeving te gebruiken om extra functionaliteit uit te testen. De services die getest zijn, zijn de core functies die nodig zijn om een “functionele” cloud op te zetten. De services waren als volgt: Keystone, Nova, Glance en Quantum. Met deze services is er kunnen testen of basisfunctionaliteit werkte zoals verwacht en of Openstack naar verwachting zal voldoen aan de eisen van i3D.net.

Aan de hand van het Proof of Concept en de verworven kennis op het gebied van Openstack zijn er een aantal extra vragen van de opdrachtgever beantwoord. Deze vragen zijn meegenomen in het adviesrapport om een duidelijk advies op te kunnen stellen. Het Proof of Concept is op deze manier erg waardevol geweest om ervaring op te doen met de Openstack omgeving. De keuze om Openstack tot Proof of Concept uit te werken is een juiste geweest.

5.) Quality Assurance and Software Testing + Afronding

Ten slotte word er in de laatste fase gekeken naar de afronding van het project. Ter afronding is er een adviesrapport geschreven met informatie voor de opdrachtgever over de verloop en uitkomst van het project. Dit rapport kan gebruikt worden om zelf verder te beslissen wat er met het aanbevolen IaaS product gebeurt.

5.1) Advies Openstack

Na het onderzoeken van verschillende pakketten en het opzetten van een Openstack Proof of Concept kunnen er een aantal conclusies getrokken worden over de vraag “Wat voor IaaS oplossing past het beste bij i3D.net en hoe beïnvloed dit de diensten van i3D.net”.

Ten eerste kan is er het advies om verder te gaan met Openstack. In alle uitgevoerde testen is Openstack op het moment duidelijk het beste naar voren gekomen. Het is een sterk groeiend product wat op het moment al sterke core functionaliteit heeft en snel groeit door middel van support van grote spelers als NASA, Paypal, Walmart en HP. Als er gekeken wordt naar hoe actief de community op het moment is en hoe het staat met de groei van Openstack tegenover de gemiddelde life cycle van Open-Source projecten is er tevens geen vermoeden dat het Openstack project binnenkort verlaten wordt of buiten development raakt.

Als er gekeken wordt naar het in gebruik nemen van Openstack zijn er een aantal punten sterk aan te raden. Ten eerste is er helaas geen tijd geweest om een opstelling uit te werken welke voor productie geschikt zou zijn. Vanwege de grootte van het Openstack project wordt er daarom ook aangeraden om een dedicated team van 2-3 medewerkers in te zetten om het Openstack project uiteindelijk te realiseren voor i3D.net. Op deze manier kan de workload voor alle onderdelen van Openstack verspreid worden en kan er per medewerker meer in-depth kennis opgedaan worden van de configuratie van elk sub-project van Openstack.

Verder is er tijdens gesprekken met medewerkers gebleken dat veel automation op het moment via scripting in het admin panel in combinatie met PXE verloopt. Vanwege de architectuur van Openstack wordt er sterk aangeraden om een automation platform op te zetten welke gebruikt kan worden voor het uitrollen en configureren van nieuwe Openstack nodes. Dit voorkomt een hoop repetitief handmatig werk wat gemakkelijker door een automation applicatie als Puppet, Chef of Salt uitgevoerd zou kunnen worden.

Op basis van het Proof of Concept kan er onder andere geconcludeerd worden dat de functionele eisen welke i3D.net wil behalen met Openstack grotendeels te bereiken zijn. Alle basisfunctionaliteit welke van een cloud verwacht zou worden is te implementeren via Openstack, echter zijn er een aantal later besproken optionele punten waar Openstack nog niet aan voldoet. Ten eerste was er de vraag of Shared Storage een mogelijkheid was voor Openstack. Tijdens het uitvoeren van dit project is Shared Storage nog geen ondersteunde functie van Openstack, echter is het sub-project Manila inmiddels opgezet om deze functionaliteit toe te voegen aan Openstack.

De verwachting is dat dit binnen de komende 3 releases officieel toegevoegd wordt aan Openstack.

De volgende vraag betrof het draaien van virtuele machines geïsoleerd op compute nodes, deze vraag gaat gepaard met de vraag of local storage te gebruiken is met Openstack. De kracht van Openstack ligt hem in het samenvoegen van gedistribueerde hardware naar 1 cluster. Daarom is er na het onderzoeken van deze vragen ook geconcludeerd dat VM's geheel gecentraliseerd draaien wel mogelijk is, maar sterk af te raden is. Openstack heeft een tweetal projecten gebaseerd op storage; Cinder en Swift, deze gaan er beide van uit dat er gebruik gemaakt wordt van centrale storage pools waar virtuele machines ruimte uit toegewezen krijgen. Vanwege de techniek hoe deze projecten zijn ingericht is er dus ook niet rekening gehouden met individuele pools per node. Dit is mogelijk uit te voeren, maar kost erg veel administratieve handelingen in Openstack om dit werkend te maken en is daarom ook af te raden. Openstack heeft de voorkeur om hardware in grotere pools in te delen, zoals een high performance pool en een low performance pool. Ook is er interesse in het delen van CPU kracht tussen nodes. Dit is een feature welke voor Openstack nog niet beschikbaar is en waar ook nog geen duidelijke plannen voor zijn in de toekomst.

Ten slotte is er nog de vraag geweest hoe Openstack omgaat met high availability. Hierbij moet gekeken worden naar de kracht van cloud computing. Cloud computing biedt namelijk de mogelijkheid om vele generieke "workers" op te zetten welke individueel geen impact op de functionaliteit van een service hebben mocht deze uitvallen, er is hooguit een tijdelijk disruptie in het managen van de workers. Hierom ligt de focus bij Openstack dus ook niet op het high available houden van VM's. Wat wel high available opgezet kan worden is de backend van Openstack, namelijk de databases services, storage en API services. Deze kunnen door middel van een generieke Pacemaker stack welke zorgt voor failovers waar nodig.

Concluderend kan er dus gezegd worden dat Openstack de juiste keuze is voor i3D.net en dat met een geschikt team en een duidelijke doelstelling voor de Openstack cloud dit een zeer sterke additie in het aanbod van i3D.net oplevert.

6.) Terugblik resultaten

In dit hoofdstuk wordt er teruggekeken op de opgeleverde producten en wat voor waarde deze aan het project in het geheel hebben toegedragen. De volgende producten worden besproken:

- Plan van Aanpak
- Bedrijfsoriëntatie
- Softwareonderzoek
- Proof of Concept
- Adviesrapport

6.1) Plan van Aanpak

De reden dat het Plan van Aanpak wordt geschreven is om een duidelijk beeld te krijgen over hoe het project aangepakt wordt en zal verlopen. Tijdens het project is dit zeer waardevol geweest om te schakelen tussen de stagiair en de opdrachtgever. Door het vast te leggen van alle documentatie kan er op een duidelijke manier vastgesteld worden of beide partijen het eens zijn over de doelstellingen, werkzaamheden, producten en andere regels tijdens het project.

Naast het communiceren tussen de opdrachtgever en de stagiair wordt dit document gebruikt door de stagiair zelf om een kritische blik te werpen op de beste aanpak van het project. Door het kiezen van ontwikkelmethoden en het plannen van de agenda moet er meteen gekeken worden naar de scope van het project en op wat voor manier deze het beste aangepakt kan worden.

Door het indelen van een planning kan er gekeken worden hoeveel tijd er per fase besteed kan worden. Dit geeft daarna een leidraad om niet teveel of te weinig tijd per fase te besteden.

6.2) Bedrijfsoriëntatie

De bedrijfsoriëntatie dient om de stagiair een goed beeld te geven van hoe de organisatie in elkaar steekt. Het zorgt ervoor dat er kritisch gekeken wordt naar hoe de organisatie ingedeeld is en op welke afdelingen de stageopdracht invloed heeft. Verder geeft het inzicht over hoe de organisatie in elkaar steekt en of dit voldoet aan de verwachting van de stagiair.

6.3) Softwareonderzoek

Het Softwareonderzoek is geschreven om de verschillende beschikbare softwarepakketten met elkaar te vergelijken. Er wordt gekeken naar wat de eisen zijn vanuit de opdrachtgever en of deze overeenkomen met wat de pakketten aanbieden. Omdat de community een grote rol speelt bij Open-Source projecten wordt er eerst naar de community's achter de producten gekeken. Na het beoordelen van de community's wordt de geadverteerde functionaliteit bekeken. Dit is om een globale controle uit te voeren of het pakket voldoet aan de eisen en de kans op een succesvol Proof of Concept te vergroten.

6.4) Proof of Concept documentatie

In de documentatie betreffende het Proof of Concept wordt er voornamelijk gekeken naar de techniek achter Openstack en hoe deze geïnstalleerd en geconfigureerd dient te worden. Er wordt beschreven op welke wijze er gekomen is tot een Proof of Concept en welke bijzonderheden zich hierbij voor hebben gedaan. Naar mijn mening is het opstellen van dit document nuttig geweest om niet alleen voor de stagiair zelf bij te houden wat er mis kan gaan en waar op gelet moet worden, maar ook om in de toekomst andere mensen te kunnen introduceren aan Openstack en de omgeving waarin deze draait.

Het document is heeft iets langer geduurd dan verwacht gezien het opzetten van Openstack zelf meer tijd kostte dan aanvankelijk verwacht; dit heeft echter geen vertraging opgeleverd.

6.5) Adviesrapport

Ten slotte wordt het advies voor de opdrachtgever gepresenteerd in rapportformaat waarin voor alle betrokkene makkelijk te zien is wat de opdracht van de stagiair inhoudt, en hoe deze is uitgevoerd. Naar mijn mening geeft dit rapport duidelijk en beknopt weer waarom de keuze voor Openstack is gemaakt en hoe het vervolg van dit project er uit kan zien.

7.) Evaluatie producten

In dit hoofdstuk wordt er teruggekeken op de opgeleverde producten en hoe deze tot stand zijn gekomen. De volgende producten worden besproken:

- Plan van Aanpak
- Bedrijfsoriëntatie
- Softwareonderzoek
- Proof of Concept
- Adviesrapport

7.1) Plan van Aanpak

Het Plan van Aanpak is vlot geschreven. In de eerste week is het plan grotendeels afgemaakt met behulp van het afstudeerplan welke van te voren opgezet was. In het plan worden een aantal project management methodieken beschreven; dit was een van de lastigere onderwerpen om uit te werken. De gelimiteerde ervaring met project management in bedrijfscontext zorgt ervoor dat er snel teruggevallen wordt op bekende methoden. Om een methode te selecteren is er gekeken naar welke methodieken al bekend waren en of er eventueel afgeleide methoden geschikt zouden kunnen zijn.

Verder zijn de werkzaamheden gespecificeerd en door te spreken met de opdrachtgever inline gehouden met wat er verwacht werd. De planning is besproken met de begeleiding vanuit de Haagse Hogeschool zodat deze tijdens de voortgangsgesprekken besproken konden worden. De planning zelf is vrij ruim ingepland om mogelijkheid tot uitloop te geven. Daarnaast zijn de fases ruim gerekend en er is met een buffer van ongeveer twee weken rekening gehouden om feedback te verwerken in de laatste weken van het project.

7.2) Bedrijfsoriëntatie

De bedrijfsoriëntatie is in de eerste twee weken geschreven van het project, de benodigde informatie is zonder problemen verkregen en dit document is dus ook vlot tot stand gekomen. Op een derde van het project is er nog teruggekeken op het document om te controleren of wat er in de eerste instantie beschreven was klopt met wat er na een aantal weken bevonden is.

7.3) Softwareonderzoek

Tijdens het softwareonderzoek is er gezocht naar mogelijke kandidaten voor de IaaS oplossing voor i3D.net. Het uitzoeken van de grotere projecten en projecten met grootste kans op slagen heeft relatief veel tijd in beslag genomen vanwege de omvang van de software pakketten. Aanvankelijk was de verwachting van de stagiair dat de beschikbare pakketten over duidelijke feature sheets of white papers zouden beschikken waaruit een groot gedeelte van de informatie al te halen zou zijn. In de praktijk bleek dit echter niet zo te zijn: er worden in de realiteit veel generieke en/of subjectieve termen gebruikt waardoor het lastig is om in te schatten wat er exact wel en niet ondersteund wordt.

Door kritisch te kijken naar de overige eisen van de opdrachtgever en een aantal objectievere bronnen te raadplegen zoals community peilingen is er toch tot een conclusie gekomen welke van de pakketten er verder getest zou worden.

7.4) Proof of Concept

Het Proof of Concept is gebouwd op basis van de resultaten van het softwareonderzoek. Door het uitvoeren van extra onderzoek naar de functionaliteit en documentatie hierover is er kennis opgedaan welke nodig was om het Proof of Concept op te bouwen. Zoals verwacht was er ruime documentatie beschikbaar tot het opzetten van een basis opstelling, maar is er uiteindelijk toch vrij veel extra onderzoek nodig gebleken om kleine bugs, configuraties voor unieke omgevingen en foutmeldingen duidelijk te kunnen begrijpen en oplossen.

Om het Proof of Concept te realiseren is er gebruikt gemaakt van een drietal HP Proliant DL120 en een drietal DL160's. Deze zijn ruim op tijd klaargezet door collega's en zijn zonder problemen in gebruik genomen voor het Concept.

Wat tijdens het onderzoeken van Openstack wel is gebleken is dat Openstack een stuk dieper gaat dan aanvankelijk verwacht: de verschillende projecten binnen Openstack hebben allemaal hun eigen community en contributors waardoor in de eerste instantie de informatie wat verspreid te vinden is. Er is hierom gekozen om eerst een Proof of concept op te zetten welke puur op de basis van Openstack is gericht. Met dit Concept kan de opdrachtgever een idee krijgen of er akkoord gegaan wordt met het opbouwen van een daadwerkelijke Openstack cloud. Dit om met vertrouwen een complete omgeving op te zetten welke op load getest kan worden.

7.5) Adviesrapport

Ten slotte is op basis van het Proof of Concept en de software analyse het adviesrapport geschreven. Deze is volgens de planning op tijd afgerond na het afronden van de voorgaande documentatie.

8.) Terugblik afstuderen

Ten slotte wordt het afstuderen in het algemeen geëvalueerd. Er zal gekeken worden naar de methoden en technieken, de aanpak van het project, opgeleverde producten, het aantonen van de competenties, mijn persoonlijke ervaring, de stage en de opleiding zelf.

8.1) Methoden & Technieken

Tijdens dit project is er gebruik gemaakt van de Sashimi-waterval methode. Ik heb verscheidene methoden vergeleken en uiteindelijk deze gekozen. De reden hiervoor was dat ik de Sashimi variant een goede toevoeging vond aan de methode. De waterval methode is over het algemeen gezien als een methode met een aantal fundamentele problemen. Vooral het feit dat er geen mogelijkheid is om terug te vallen op de vorige fases om problemen op te lossen en de sterk lineaire aanpak van projecten wordt bekritiseerd. Er is ondanks de kritiek op de waterval methodiek toch gekozen voor de Sashimi variant van deze methode, Hier zijn een aantal redenen voor.

Waarom was Sashimi waardevol?

De Sashimi methode verschilt tegenover de standaard waterval methode in de korte overlap tussen de fasen. Dit zorgt ervoor dat er overlap ontstaat tussen opvolgende fasen. Het voordeel hiervan is dat er bij een probleem welke pas tijdens de Development and Coding fase voorkomt, teruggevallen kan worden naar de Design and Architecture fase om te controleren of dit probleem in het ontwerp vermeden kan worden alvorens de Development and Coding fase hervat wordt. In het traditionele waterval model zou dit een gefaald project als gevolg hebben en zal het hele project opnieuw gestart moeten worden vanaf de eerste fase.

In projecten waar één fout in een ontwerp een onvermijdelijk probleem veroorzaakt is dit logisch; echter in dit afstudeerproject is één probleem geen reden tot het stoppen van het project of het herstarten van het gehele project. Het project bouwt namelijk voort op de kennis welke in de eerdere fasen opgedaan is. De grootste baat bij deze methode had ik verwacht tussen de mijlpalen waar één van de projecten gekozen wordt en het uitwerken van het Proof of Concept hiervan. Vanwege de omvang van de software pakketten verwachtte ik zeker de kans dat er problemen zich pas zouden kunnen voordoen op het moment dat een Proof of concept opgezet werd. De Sashimi methode geeft hierbij de mogelijkheid om deze ervaring terug mee te nemen naar de analyse en mogelijk toch een ander pakket te kiezen indien deze verder ook zouden voldoen aan de eisen.

Ten slotte wordt er in andere bekende methoden erg veel focus gelegd bij gedefinieerde rollen en verantwoordelijkheden, denk hierbij aan PRINCE2. Uit ervaring van eerdere stages heb ik de ervaring gekregen dat in projecten waar binnen een bedrijf één persoon verantwoordelijk is tegenover één opdrachtgever dit soort management methoden voor veel administratief werk zorgen. De tijd die hier aan besteed wordt kan waardevoller besteed worden aan het onderzoek zolang de rollen en verantwoordelijkheden duidelijk in het Plan van Aanpak beschreven zijn.

8.2) Aanpak planning

De planning heeft uiteindelijk goed gediend als richtlijn. Een groot deel van de verwachtingen op hoeveel tijd er besteed zou worden aan verscheidene onderdelen zijn correct gebleken. Er is in de eerste instantie ruim geschat hoeveel tijd er per fase benodigd zou zijn, dit heeft goed uitgedaan in dit project. De reden hiertoe is dat er aanvankelijk weinig bekend was van de verschillende IaaS pakketten, de combinatie van ruime fase planning en de Sashimi methode heeft ervoor gezorgd dat er genoeg tijd was om documenten uit vorige fasen aan te vullen en binnen de verwachte tijd te blijven.

De enige verwachting die niet geheel is uitgekomen is de mogelijkheid tot realisatie van het Proof of Concept. Vanwege de bevindingen tijdens het project is er gebleken dat het niet haalbaar zou zijn om binnen 17 weken een Proof of Concept op te bouwen en deze hierna ook nog als productie product in te zetten. De omvang van het opzetten van een IaaS omgeving bleek uiteindelijk te groot om op te pakken binnen de stage periode, hierom is gekozen om het bij een advies rapport te houden.

Verder ben ik tevreden met de inschattingen die zijn gedaan bij de aanvang van het project, ik heb ruim rekening genomen met tijd om problemen op te vangen en heb deze zodoende ook goed kunnen gebruiken.

8.4) Aantonen competenties

Van de onderstaande belangrijke competenties zal beschreven worden op wat voor wijze deze voldaan zijn op een wijze welke van HBO Technische Informatica verwacht wordt.

A3: Achterhalen van behoeften van belanghebbenden

Door middel van gesprekken met de opdrachtgever zijn de behoeften achterhaald welke door dit project gerealiseerd kunnen worden. De eisen zijn vervolgens verwerkt in het softwareonderzoek om tot een logische conclusie te kunnen komen.

B6: Selecteren van een bestaande hardware en/of softwarecomponent

Het uitzoeken van het juiste pakket is uitgevoerd op basis van de wensen van het bedrijf en een softwareonderzoek om te kijken welke pakketten er beschikbaar zijn en wat zij bieden. Dit product is vervolgens uitgewerkt tot een Proof of Concept om te controleren of deze keuze de juiste was.

D17: Testen van software systemen

Het Proof of Concept is gebruikt om te controleren of de software voldeed aan de vragen en eisen van de opdrachtgever, de resultaten hiervan staan verwerkt in het Proof of Concept rapport.

G1: Praktische aspecten hanteren in projecten

Er zijn een aantal praktische aspecten aan dit project waar rekening mee gehouden moest worden. Zo zijn er een aantal vergadermomenten geweest om problemen aan te kaarten en vragen te stellen. Verder is het oplossen van problemen met de werkplek en testomgeving overgelaten aan de stagiair. Op het moment dat er problemen optraden zijn deze direct bij de juiste collega's besproken.

J11: Je plek vinden in een organisatie

Naast het werk zelf is de sfeer binnen een bedrijf erg belangrijk: zonder goed contact met collega's kan er geen fijne werksfeer ontstaan. Door actief bezig te zijn met de afdeling en interesse te tonen in andere werknemers en projecten is er gezorgd dat de stagiair niet naar de achtergrond verdwijnt, maar zich mengt met de rest van het bedrijf.

8.5) Evaluatie persoonlijke ervaring

Zelf is er veel geleerd van de stage, een aantal vlakken waar meer kennis over opgedaan is zijn:

Inhoudelijk:

- Cloudsoftware oplossingen
 - Openstack
 - Cloudstack
 - Eucalyptus
- Open-source community's
- Software Defined Networking
- Networking
 - VLANs
 - GRE tunnels
 - Openvswitch

Projectmatig:

- Verwantwoorden van en werken met de Sashimi-Waterval methode
- Continu feedback verwerken van begeleiding
- Documentatie verwerken tijdens het praktisch werken
- Op tijd afspraken m.b.t. vergaderingen en feedback requests

8.6) Evaluatie Stage

Tijdens het project is er ruimte geboden om zelf te beslissingen te maken over mijn werkwijze. Dit heeft mij de mogelijkheid gegeven om competenties aan te tonen zoals het kiezen van een geschikte project management methode en het plannen van het project. Alhoewel dit in menig organisatie niet mogelijk zal zijn i.v.m. vastgestelde project management methodes is dit toch erg waardevol om zelf te kunnen uitwerken om in een relatief veilige situatie de benodigde ervaring op te doen.

Verder is er tijdens de opdracht altijd genoeg tijd geweest om de stagiair te begeleiden of feedback te geven waar nodig. Zolang er op tijd afspraken gemaakt worden is er binnen een week tijd geweest om een contactmoment te plannen. Het op tijd plannen van afspraken binnen een bedrijf is tijdens vorige stages geleerd en elke keer bleek dit nodig te zijn. Daarom wordt er bij iedere nieuwe werkplek scherp in de gaten gehouden wanneer mogelijke contactmomenten op zullen treden. Door dit te hanteren is er ook bij i3D.net geen sprake geweest van vertraging door te drukke werknemers of begeleiders.

Over het algemeen ben ik dus zeer tevreden over de stage bij i3D.net. Het is een zeer interessant omgeving om in te werk te gaan met meer dan genoeg afdelingen waar meer kennis opgedaan kan worden. De werknemers nemen graag de tijd om te helpen met problemen of lastige onderwerpen te bespreken, de begeleiding is daarnaast ook altijd bereid geweest om feedback te verzorgen indien hierom gevraagd werd.

8.7) Evaluatie Opleiding

Ik ben van mening dat de stof welke op de opleiding behandeld is mij over het algemeen goed geholpen heeft om een theoretische basis op te bouwen om in de praktijk mee aan de slag te gaan. Er is gebruik gemaakt van de kennis welke op de vlakken van onder andere modelleren en analyseren, infrastructuur en beheer, communicatieve vaardigheden, projectmethodiek en linux gebaseerde operating systemen.

Alhoewel er tijdens de opleiding vooral theorie behandeld is, is dit uiteindelijk erg bruikbaar geweest om nieuwe software principes en ontwikkelmethoden te begrijpen en hier zelf mee aan de slag te kunnen gaan. Hetgeen wat ik wel graag meer had willen leren is een diverser aantal projectmanagement methoden welke niet direct op softwareontwikkeling gericht zijn. De methoden zijn tijdens de opleiding kort aan bod gekomen waarbij tijdens de projecten welke management vereiste eigenlijk altijd software management methodes gebruikt zijn en er minder kans is geweest om ervaring met andere methoden op te doen.

Bibliografie

[1]

G. V. D. W. William Fox, *A Guide to Project Management*.

[2]

"Sashimi model." [Online]. Available: <http://www.waterfall-model.com/sashimi-waterfall-model/>.

[3]

"Openstack community wiki." [Online]. Available: https://wiki.openstack.org/wiki/Main_Page.

[4]

"openstack vs cloudstack." [Online]. Available:

<http://www.getfilecloud.com/blog/2014/02/a-game-of-stacks-openstack-vs-cloudstack/#.VOljcfmGi8>.

[5]

"Openstack alternatieven." [Online]. Available:

<http://siliconangle.com/blog/2012/04/24/three-open-source-based-cloud-alternatives-to-openstack/>.

[6]

"Reddit Openstack discussion." [Online]. Available: <http://www.reddit.com/r/openstack>.

[7]

"Eucalyptus datasheet," <https://www.eucalyptus.com/sites/all/files/ds-eucalyptus-iaas.en.pdf>. .

[8]

"Cloudstack features." [Online]. Available: <http://cloudstack.apache.org/software/features.html>.

[9]

"OpenStack — more than just software."

[10]

"A guide to the openstack Kilo Release," *a-guide-to-the-openstack-kilo-release*. [Online]. Available:

<https://developer.ibm.com/opentech/2015/04/30/a-guide-to-the-openstack-kilo-release/>.

[11]

"IaaS community comparison." [Online]. Available: <http://www.qyjohn.net/?p=3522>.

[12]

"Cloud platform OpenStack owes a debt to NASA.," *Computerworld*, vol. 47, pp. 10–11, 2013.

[13]

"Nebula, NASA, and OpenStack." [Online]. Available:

<https://open.nasa.gov/blog/2012/06/04/nebula-nasa-and-openstack/>.

[14]

"Installation guide Openstack." [Online]. Available:

<http://docs.openstack.org/juno/install-guide/install/apt/content/index.html>.

Appendix

Document	Aantal pagina's
Afstudeerplan	4
Plan van Aanpak	21
Bedrijfsoriëntatie	15
Software Onderzoek	17
Proof of Concept documentatie	21
Adviesrapport	15
Verklarende woordenlijst	1

Afstudeerplan

Informatie afstudeerder en gastbedrijf

Afstudeerblok: 2015-1.1 (start uiterlijk 9 februari 2015)
Startdatum uitvoering afstudeeropdracht: 9-2-2015
Inleverdatum afstudeerdossier volgens jaarrooster: 5 juni 2015

Studentnummer: 09032835
Achternaam: dhr. Zonneveld
Voorletters: C.E.
Roepnaam: Chris
Adres: Zwanenpad 16
Postcode: 2295PC
Woonplaats: Kwintsheul
Telefoonnummer: 0174-294971
Mobiel nummer: 06-27496776
Privé emailadres: zonneveld.chris@gmail.com

Opleiding: Technische Informatica
Locatie: Delft
Variant: voltijd

Naam studieloopbaanbegeleider: Dhr. M.G. Maris
Naam begeleidend examiner: Dhr. J.D. Schagen
Naam tweede examiner: Dhr. M.G. Maris

Naam bedrijf: i3D.net
Afdeling bedrijf: Infrastructure
Bezoekadres bedrijf: Van Nelleweg 1
Postcode bezoekadres: 3044BC
Postbusnummer:
Postcode postbusnummer:
Plaats: Rotterdam
Telefoon bedrijf: +31 (0)10 890 00 70
Telefax bedrijf: +31 (0)842 241 251
Internetsite bedrijf: <http://www.i3d.net/>

Achternaam opdrachtgever: dhr Rick Sloot
Voorletters opdrachtgever: R.
Titel opdrachtgever:
Functie opdrachtgever: Director of Operations
Doorkiesnummer opdrachtgever:
Email opdrachtgever: ricksloot@i3d.nl

Achternaam bedrijfsmentor: dhr Rick Sloot
Voorletters bedrijfsmentor: R.
Titel bedrijfsmentor:
Functie bedrijfsmentor: Director of Operations
Doorkiesnummer bedrijfsmentor:
Email bedrijfsmentor: ricksloot@i3d.nl

Titel afstudeeropdracht:

Opdrachtomschrijving

1. Bedrijf

i3D.net is een managed hosting en IaaS provider uit Rotterdam. Zij bedienen ruim 30.000 klanten vanuit 16 datacenters wereldwijd. i3D.net beheert een eigen serverpark van 8.000 servers. i3D.net is opgericht in 2004 in Rotterdam en heeft inmiddels heel wat onderscheidingen in de wacht gesleept, zoals een ranking in de *Deloitte Technology Fast 500* als snelst groeiende managed hosting provider van de Benelux. i3D.net is tevens de winnaar van de *Financieele Dagblad Gouden Gazellen Award* in 2012. Met een team van technische medewerkers, bieden zij Infrastructure as a Service (IaaS) en managed hosting-oplossingen voor een breed scala aan klanten. i3D.net bedient onder meer klanten uit de volgende branches: overheid, zorg, sport, gaming, educatie, webshops, hosting, en printmedia. i3D.net bezit eigen datacenters, beheert 16 datacenters en een eigen serverpark van ruim 8.000 servers wereldwijd: zij zijn een hard groeiende en financieel gezonde organisatie.

De opdracht wordt uitgevoerd op de afdeling infrastructure. De medewerkers op deze afdeling hebben allemaal een eigen focus waar zij zich dagelijks mee bezig houden. Er zijn ~5 medewerkers werkzaam op deze afdeling. Verder is er voor de medewerkers van het datacenter in het algemeen doordeweeks bezetting van 08:00 tot 24:00 en in het weekend van 10:00 tot 18:00, waarnaast nog een storingsdienst draait om 24/7 bereikbaar te zijn voor klanten.

2. Probleemstelling

Op het moment draaien veel van de servers bij i3D.net fysiek. Aangezien Cloudbeheer vele voordelen met zich meebrengt, wil i3D.net de transitie maken naar meer virtualisatie gebaseerde machines. Zij zijn hier zelf al mee bezig door een Hyper-V omgeving op te zetten in samenwerking met Microsoft, maar willen ook alternatieven voor klanten beschikbaar stellen. Hierbij moet voornamelijk gedacht worden aan alternatieven voor de Hyper-V hosting van Microsoft, om een goedkope optie aan te kunnen bieden. Hiervoor is niet alleen een ander virtualisatie platform nodig, maar ook een nieuwe methode om beide omgevingen te beheren. Hiervoor zijn pakketten als Openstack gemaakt: deze werken als overkoepelende laag voor o.a. virtualisatie pakketten en kan deze virtualisatie pakketten vanuit een centraal management punt aansturen. Dit geeft i3D.net de mogelijkheid om een alternatief voor Hyper-V aan te bieden, zoals KVM, zonder het gebruiksgemak te verliezen van één beheer oplossing.

Door over deze problemen na te denken poogt i3D.net de efficiëntie waarmee zij hun diensten kunnen aanbieden te verhogen. Daarnaast zullen zij beter mee kunnen gaan met de ontwikkelingen op het gebied van virtualisatie van de andere hosting providers.

3. Doelstelling van de afstudeeropdracht

Het doel van deze opdracht is een verbeterd virtualisatie platform te creëren voor i3D.net. Aan het eind van de afstudeerperiode zal i3D.net gemakkelijker hun virtualisatie diensten , kunnen aanbieden aan hun klanten en zullen zij deze zelf gemakkelijker kunnen beheren.

4. Resultaat

Na het uitvoeren van de stage opdracht zal i3D.net een virtualisatieplatform hebben dat zij gemakkelijk kunnen aanbieden aan hun klanten, zodat zij deze op een snellere manier aan cloud oplossingen kunnen helpen. Met een Openstack oplossing zal i3D.net tijd en geld besparen in het uitvoeren van hun werkzaamheden, gezien het feit dat deze niet meer handmatig in verschillende hypervisor omgevingen uitgevoerd hoeven te worden.

5. **Uit te voeren werkzaamheden, inclusief een globale fasering, mijlpalen en bijbehorende activiteiten**

- Afstudeer dossier schrijven [15 dagen]
- **Plan van Aanpak schrijven [4 dagen]**
- **Bedrijfsoriëntatie schrijven [4 dagen]**
- Software onderzoek uitvoeren [4 dagen]
- Gesprekken voeren met medewerkers [3 dagen]
- **Vastleggen eisen medewerkers en opdrachtgever [3 dagen]**
- Beschikbare documentatie software oplossingen uitzoeken [3 dagen]
- Proof of Concept maken [15 dagen]
- Proof of Concept bespreken met opdrachtgever/werknemers[3 dagen]
- Proof of Concept testen [10 dagen]
- **Documentatie over Proof of Concept schrijven [5 dagen]**
- Advies geven over software pakket [1 dag]
- **Implementeren [5 dagen]**
- **Overdragen [1 dag]**
- Uitloop [9 dagen]

6. **Op te leveren (tussen)producten**

- Afstudeer Dossier
- Plan van Aanpak
- Bedrijfsoriëntatie
- Software onderzoek
- Resultaten gesprekken medewerkers
- Handleiding softwarepakket
- Adviesrapport

7. **Te demonstreren competenties en wijze waarop**

Analyse competenties:

A3: Achterhalen van behoeften van belanghebbenden

Door middel van gesprekken met werknemers van de afdeling kunnen de behoeften achterhaald worden welke door dit project gerealiseerd kunnen worden

Adviserende competenties:

B6: Selecteren van een bestaande hardware en/of softwarecomponent

Het uitzoeken van het juiste pakket zal gebaseerd worden op eerder onderzoek binnen het bedrijf, en een stuk marktonderzoek om te kijken welke pakketten er beschikbaar zijn en wat zij bieden

Realisatie competenties:

D17: Testen van software systemen

De softwarepakketten zullen getest moeten worden om te zien of ze voldoen aan de eerder gestelde eisen en of ze voldoen aan de verwachte stabiliteit.

Werken in projecten:

G1: Praktische aspecten hanteren in projecten

Er zal duidelijk gecommuniceerd moeten worden met de rest van de afdeling waar het afstuderen plaatsvindt, om de opdracht vlot en succesvol te voltooien

Functioneren in organisaties:

J11: Je plek vinden in een organisatie

D.m.v. het schrijven van een bedrijfsoriëntatie, zal de afstudeerder zich kunnen oriënteren binnen het bedrijf waar hij aan de slag gaat en een beter idee krijgen van hoe de organisatie in elkaar zit.

Plan van Aanpak

Onderzoek naar het toepassen van Openstack [en vergelijkbare oplossingen] bij i3D.net



Opdrachtgever:
Rick Slood
i3D.net
Van Nelleweg 1, 3044BC
Rotterdam



Opdrachtnemer:
Chris Zonneveld
09032835
Rotterdamseweg 137, 2628AL
Haagse Hogeschool Delft

Versiebeheer

Versie	Auteur	Datum	Wijzigingen
Plan van Aanpak 0.1	C. Zonneveld	12-2-2015	Eerste concept
Plan van Aanpak 0.2	C. Zonneveld	26-2-2015	i3D.net template ingevoegd
Plan van Aanpak 1.0	C. Zonneveld	30-3-2015	Controle

Voorwoord

Bij het naderen van het einde van mijn studie Technische Informatica aan de Haagse Hogeschool te Delft is het tijd om te beginnen met afstuderen. Dit document beschrijft het Plan van Aanpak voor mijn afstudeeropdracht bij i3D, welke ik met de kennis opgedaan in mijn opleiding zal uitwerken. Dit Plan van Aanpak is het eerste document van het afstudeerproject. Hierin zullen de richtlijnen die tijdens dit project aangehouden dienen te worden uiteengezet.

Ik ben bij i3D terecht gekomen voor mijn opdracht door mijn interesse voor netwerkbeheer en mijn vroegere interesse voor serverbeheer in online gaming. Door mijn ervaringen met online gaming was ik al bekend met i3D.net. Voorheen had ik er echter niet bij stil gestaan dat er mogelijkheden waren om bij i3D.net te werken. De vereiste afstudeeropdracht bij de opleiding Technische Informatica gaf mij de mogelijkheid om i3D.net op een professionele manier te benaderen, waaruit het onderstaande project is ontstaan.

Inhoudsopgave

[Versiebeheer](#)

[Voorwoord](#)

[Inhoudsopgave](#)

[1.\) Inleiding](#)

[2.\) Achtergrond](#)

[3\) Opdrachtsomschrijving](#)

[3.1\) Opdrachtgever en opdrachtnemer](#)

[3.2\) Probleemstelling](#)

[3.3\) Doelstelling opdracht](#)

[3.4\) Opdrachtbeschrijving](#)

[3.5\) Op te leveren producten](#)

[3.6\) Projectgrenzen](#)

[3.7\) Randvoorwaarden](#)

[4\) Aanpak](#)

[4.1\) Methoden en technieken](#)

[4.2\) Werkzaamheden](#)

[4.3\) Standaarden](#)

[4.4\) Planning](#)

[5\) Projectinrichting](#)

[5.1\) Projectorganisatie](#)

[5.2\) Informatievoorziening](#)

[5.4\) Kwaliteitsborging](#)

[6\) Risicofactoren](#)

[6.1\) Risico's](#)

[6.2\) Maatregelen](#)

[7\) Kosten en baten](#)

[7.1\) Kosten](#)

[7.2\) Baten](#)

1.) Inleiding

Dit document wordt gebruikt om de doelen, resultaten en aanpak van dit project vast te stellen. De reden voor dit document is om bij aanvang van het project duidelijk de richtlijnen van het project in zicht te hebben. Dit heeft als voordeel dat er tijdens het project zo min mogelijk gewijzigd zal worden.

De opdrachtgever heeft een duidelijk eindresultaat van de stagiair voor ogen. .

Tijdens het schrijven van dit document wordt rekening gehouden met eventuele toekomstige problemen waar adequate voorbereiding voor vereist is.

Daarnaast wordt de structurering van het project beschreven waarin onder andere het globale verloop van het project, de methodes die gebruikt worden en de belanghebbenden van dit project ter sprake komen.

2.) Achtergrond

Hosting is de focus van i3D.net, het bedrijf i3D.net is gericht op het snel en gebruiksvriendelijk aanbieden van deze dienst. Het bedrijf is in 2004 begonnen in Rotterdam en is inmiddels een van de snelst groeiende managed hosting providers van de Benelux. Op het moment bedienen ze ruim 30.000 klanten vanuit 16 datacenters wereldwijd naast hun eigen serverpark van ruim 8.000 servers in Rotterdam. De klanten komen uit verschillende branches waaronder: overheid, zorg, sport, gaming, educatie, webshops, hosting en printshops.

Een korte lijst met feiten over i3D.net:

- Bezit en beheert meer dan 8,000 servers
- Huisvest meer dan colocatie-servers
- 16 locaties wereldwijd
- 24/7 technische ondersteuning
- Gecertificeerde professionals
- Opgenomen in de 'Deloitte Fast500' snelst groeiende technologiebedrijven
- Winnaar van de 'Financieele Dagblad Gouden Gazellen Award'
- CDSA gecertificeert op Content Protection & Veiligheid
- ISO/IEC 27001 audited datacenters
- Lid van de Dutch Hosting Provider Association (DHPA)
- Management heeft meer dan 10 jaar werkveldervaring

De afstudeeropdracht zal plaatsvinden op de netwerk afdeling van i3D.net, die zich bevindt in het datacenter van i3D.net in de Van Nelle fabriek te Rotterdam. Op de afdeling zijn gemiddeld tussen de 2-5 personen dagelijks aanwezig, welke zich bezig houden met alle ontwikkelingen en research rondom de core van het netwerk. Het resultaat van dit project zal voornamelijk van toepassing zijn voor deze afdeling, maar zal ook een positieve invloed hebben op de afdeling Operations welke ook baat heeft bij dit project.

3) Opdrachtsomschrijving

In de opvolgende hoofdstukken wordt de opdracht uitgebreid beschreven en toegelicht.

3.1) Opdrachtgever en opdrachtnemer

Opdrachtgevende organisatie:

Naam bedrijf: i3D.net

Afdeling bedrijf: Infrastructure

Bezoekadres bedrijf: Van Nelleweg 1

Postcode bezoekadres: 3044BC

Plaats: Rotterdam

Telefoon bedrijf: +31 (0) 10 890 00 70

Telefax bedrijf: +31 (0) 842 241 251

Internetsite bedrijf: <http://www.i3d.net/>

Achternaam opdrachtgever: dhr. Rick Slood

Voorletters opdrachtgever: R.

Titulatuur opdrachtgever:

Functie opdrachtgever: Director of Operations

Email opdrachtgever: rickslood@i3d.nl

Opdrachtnemer:

Studentnummer: 09032835

Achternaam: dhr. Zonneveld

Voorletters: C.E.

Roepnaam: Chris

Adres: Zwanenpad 16

Postcode: 2295PC

Woonplaats: Kwintsheul

Telefoonnummer: 0174-294971

Mobiel nummer: 06-27496776

Privé emailadres: zonneveld.chris@gmail.com

Opleiding: Technische Informatica

Locatie: Haagse Hogeschool te Delft

3.2) Probleemstelling

De opdracht voor de stagiair is voortgekomen uit de groei van i3D.net als bedrijf. Als hosting bedrijf is i3D constant bezig met onderzoek naar de huidige relevante technieken in de hosting wereld. Zij zijn begonnen met het hosten van gameservers, maar na het succes hiervan hebben ze kunnen uitbreiden naar managed- en sharedhosting oplossingen en colocation. Voor colocation is één van de grootste concerns ruimte, bij managed hosting is dat echter techniek en kennis. Om klanten meer mogelijkheden te bieden op het gebied van hosting is de logische volgende stap virtualisatie. Hiervoor is i3D zelf al bezig met een hyper-V omgeving, maar het bieden van een alternatief is waardevol voor i3D. Ze zouden naast Hyper-V hosting, goedkopere hosting kunnen bieden zoals KVM hosting. Hiervoor is echter wel een wijze nodig om dit op een duidelijke en gemakkelijke manier te hosten. De probleemstelling van de afstudeeropdracht komt voort uit de zoektocht naar een oplossing om IaaS op een slimme wijze te kunnen aanbieden aan klanten.

3.3) Doelstelling opdracht

Het doel is om klanten een groter aanbod te kunnen geven van hosting diensten, zonder een grote impact op de werkdruk bij i3D. Hiervoor zal een platform nodig zijn wat hosting voor meerdere hypervisors kan aanbieden op een manier die aan de klant duidelijk en gemakkelijk aangeboden kan worden. Deze omgeving moet daarnaast door i3D gemakkelijk te schalen en beheren zijn.

3.4) Opdrachtbeschrijving

Naar aanleiding van onderzoek naar verschillende IaaS oplossingen, zal er een IaaS oplossing uitgewerkt worden tot een Proof of Concept. Vervolgens zal getest worden of deze oplossing naar wens functioneert voor i3D.net. Mocht dit Proof of Concept geslaagd zijn, kan deze uitgewerkt en overgedragen worden naar i3D.net om verder in productie te laten nemen.

3.5) Op te leveren producten

- Afstudeer Dossier;
- Plan van Aanpak;
- Bedrijfsoriëntatie;
- Software onderzoek;
- Resultaten gesprekken medewerkers;
- Handleiding softwarepakket;
- Adviesrapport

3.6) Projectgrenzen

Binnen de scope van het project valt op het moment het onderzoek doen naar IaaS oplossingen, het uitwerken van een Proof of Concept, het schrijven van de documentatie betreffende het project en eventueel het opleveren van het eindresultaat. De dagelijkse werkzaamheden van i3D.net vallen grotendeels buiten de scope van dit project, maar in overleg tussen de opdrachtgever en opdrachtnemer kunnen er werkzaamheden geaccepteerd worden om ervaring op te doen in de sector.

3.7) Randvoorwaarden

Om het project succesvol te laten verlopen worden er een aantal acties vanuit i3D.net verwacht. Er zal tijd beschikbaar moeten zijn voor de afstudeerder om met vragen te komen betreffende het project mocht de afstudeerder met vragen of opmerkingen zitten rondom het project.

Verder word het volgende van i3D.net verwacht:

- Een werkplek waar de stagiair het project uit kan voeren
- Vrijheid om de benodigde software hierop te installeren
- Een testomgeving waar een Proof of Concept gerealiseerd kan worden

4) Aanpak

In de volgende hoofdstukken zal de aanpak van het project beschreven worden. De volgende onderwerpen zullen aan bod komen: de methoden en technieken welke gebruikt zullen worden, de uitleg en uitvoering van de werkzaamheden, de standaarden welke tijdens het project aangehouden worden en de globale planning van het project.

4.1) Methoden en technieken

Voor dit project zijn een aantal methoden gekozen op basis van een aantal eigenschappen die de toepasbaarheid op dit specifieke project waarborgen. Deze methoden zullen met elkaar vergeleken worden waaruit vervolgens de beste gekozen zal worden.

-Flexibiliteit

Een eis halverwege het project, omdat er bij de aanvang van het project nog geen 100% zekerheid is over waarmee het eindresultaat behaald zal worden. Er is vraag naar een methode welke flexibel genoeg is om deze mogelijke changes te ondersteunen.

-Projectgrootte

Dit project word door 1 persoon uitgevoerd met, naast de begeleiding, waar nodig wat hulp van 1 of 2 medewerkers. Hierdoor is een methode nodig welke schaalbaar is naar kleine projecten en geen papierwerk verwacht gebaseerd op grotere projectgroepen.

-Onderwerp relevantie

De ontwikkelmethode moet toepasbaar zijn op een project gefocust op research en proof of concept opzet.

-Contactmomenten

De methode mag niet afhankelijk zijn van teveel contactmomenten tijdens het verloop van de stage, omdat dit geen optie is voor de begeleiding.

-Doorlooptijd

De methode moet geschikt zijn voor de doorlooptijd van 17 weken.

De kandidaat methoden voor dit project waren als volgt:

Waterval methode [sashimi variatie]

Dit is een rechttoe-rechtaan methode. Er wordt in duidelijke fases gewerkt die niet afgesloten worden tot ze 100% compleet zijn. De Sashimi variatie voegt hier overloop tussen de fasen aan toe waardoor er efficiënter gebruik gemaakt wordt van de beschikbare tijd en er meer controle is over het project.

RUP

Een software ontwikkelingsproces, gebaseerd op het incrementeel en iteratief ontwikkelen van software. Voornamelijk in gebruik genomen om grote software projecten te ontwikkelen om deze efficiënt te kunnen splitsen in meerdere onderdelen die simultaan ontwikkeld worden.

SCRUM

Een flexibel proces om software te ontwikkelen. Het wordt voornamelijk ingezet om binnen korte perioden werkende producten op te leveren. Kenmerkend zijn dagelijkse scrum meetings om voortgang en planning te bespreken.

PRINCE2

Een zeer flexibele methode met grote flexibiliteit. Kenmerkend is het tailoren van PRINCE2 aan het specifieke project en de voortdurende rechtvaardiging. Onderdelen die niet van belang zijn kunnen weggesneden worden uit de methode om deze aan te passen aan de huidige situatie

	Eigenschappen	Flexibiliteit	Projectgrootte	Onderwerp relevantie	Contactmomenten	Doorlooptijd
Methode						
Sashimi		+-	+	+	++	+
RUP		+-	+-	--	+-	+-
SCRUM		-	+	--	--	+-
PRINCE2		++	-	+	+-	+-

Na het vergelijken van de methodes is er besloten volgens de Sashimi methode te werken om de volgende redenen:

-Flexibiliteit

Flexibiliteit is geen enorme eis aan het project, behalve halverwege waar een methode gekozen wordt welke tot Proof of Concept uitgewerkt wordt. Hiervoor is Sashimi geschikt omdat deze de mogelijkheid geeft een overlapping te hebben tussen 2 fasen, waardoor zonder een hele fase opnieuw te moeten starten er wijzigingen in de vorige fase gemaakt kunnen worden. De andere methoden zijn hier naast SCRUM in principe ook geschikt voor, SCRUM heeft echter als nadeel dat er snel gewerkt wordt naar een prototype, wat hier niet bij nodig is en te limiterend is.

-Projectgrootte

Dit is een van de grootste valkuilen bij het kiezen van een methode, ontwikkelmethodes gaan er snel van uit dat er in groepen van 20 tot 1000 man gewerkt wordt. Bij dit soort grote groepen is duidelijke documentatie een must om het overzicht te behouden. Mijn project is echter een stuk kleiner dan dat er hierdoor is overbodig documentatiewerk niet alleen onnodig om overzicht te houden, maar neemt het ook kostbare tijd in beslag. Hiervoor zijn Sashimi en PRINCE2 ideaal om beide of niet al te veel documentatie vereisen of mogelijkheden hebben om documenten te schrappen.

-Onderwerp relevantie

Op dit onderwerp vallen eigenlijk een aantal methodes al af, SCRUM en RUP zijn namelijk primair methoden gebaseerd op software-ontwikkeling. Aangezien dit project niet daadwerkelijk software ontwikkelt maar een pakket kiest en op basis daarvan een Proof of Concept uitrolt, zijn deze methoden wel in zeker mate toe te passen, maar niet geheel geschikt voor dit soort projecten. Sashimi en PRINCE2 zijn beide erg breed toe te passen hier tegenover.

-Contactmomenten

Er wordt tijdens dit project gerekend op contact momenten van één maal per één of twee weken. Verder is het wel tussendoor mogelijk om tussendoor vragen te stellen, maar dit staat niet gelijk aan vooraf geplande afspraken waar beide partijen zich op kunnen voorbereiden. SCRUM is hier 1 van de grootste missers bij, vanwege de dagelijkse meetings. De andere methoden zijn allemaal redelijk geschikt voor de relevante contact momenten.

-Doorlooptijd

Er moet rekening gehouden worden met de doorlooptijd van 17 weken. SCRUM rekent bijvoorbeeld op doorlooptijden van 1-4 weken wat simpelweg te kort is. De overige methoden kunnen allemaal in voldoende mate aangepast worden aan de verwachte doorlooptijd.

De Sashimi variant is voor dit project gekozen en licht aangepast om beter aan te sluiten op het huidige project. Normaal gesproken bestaat de Sashimi methode uit de volgende fasen: Requirements, Design and Architecture, Development and Coding, Quality Assurance and Software Testing, Implementation, and Maintenance and Support. Omdat dit erg software development gebaseerd is heb ik besloten deze fase aan te passen en als volgt in te delen: initiatiefase, onderzoeksfase, uitvoeringsfase, afronding.

De initiatiefase word gebruikt om het project op te starten. Deze fase word niet officieel aangehouden door het sashimi model omdat deze aan neemt dat het project door een project manager binnen een bedrijf start in plaats van een stagiair die eerste een voorbereidingsperiode nodig heeft binnen het bedrijf.

Vervolgens de onderzoeksfase, dit heeft eigenschappen van de Requirements fase samen met de Design and Architecture fase. De combinatie hiervan is gekozen omdat er geen reden is voor een specifieke Design and Architecture fase. Er is tijdens het project alleen een keuze tussen verschillende oplossingen zonder dat hier zelf een design of architectuur voor ontworpen hoeft te worden. De fase bestaat dus uit het achterhalen van de eisen van de werknemers en opdrachtgever, om deze verder te vergelijken met de beschikbare oplossingen.

Hierna start de uitvoeringsfase, deze bestaat uit de Sashimi fasen Development and Coding en Quality Assurance and Software Testing. Dit zijn de fasen die het meest gericht zijn op software development, er is gekozen om deze simpelweg te hernoemen naar de uitvoeringsfase. In deze fase word Development and Coding vervangen door het maken van het Proof of Concept en word Quality Assurance and Software Testing vervangen door het testen van het Proof of Concept. De reden dat hiervoor gekozen is is omdat er in dit project geen directe development en coding uitgevoerd word, er word een Proof of Concept opgezet van een bestaande oplossing welke wel aangepast zal moeten worden naar de eisen van de opdrachtgever. De manier waarop hier “getest” wordt is door middel van het controleren van het Proof of Concept voldoet aan de eisen van de opdrachtgever en of deze de verwachte functionaliteit zonder problemen uitvoert.

Ten slotte is er nog de afronding, deze bestaat uit de Implementation fase met een klein stuk Maintenance and Support van Sashimi. De implementatie van het Proof of Concept staat gelijk aan de Implementation fase, maar de fase Maintenance and Support is iets ingekort. Na het opleveren van het Proof of Concept en eventuele implementatie, zal er documentatie aangeleverd worden om het systeem te onderhouden, maar er zal door de stagiair geen verantwoordelijkheid aangenomen worden om het systeem verder te supporten en onderhouden. Dit zal de verantwoordelijkheid van i3D.net worden.

4.2) Werkzaamheden

De werkzaamheden van het project zijn onderverdeeld worden in 4 fasen, namelijk de initiatiefase, de onderzoeksfase, de uitvoeringsfase en de afronding.

De initiatiefase

Gedurende deze fase zal er voornamelijk documentatie geschreven worden betreffende het project. Deze fase is bedoeld om voor de afstudeerder zich een duidelijk beeld te vormen over de organisatie waarbinnen het project uitgevoerd wordt en zich te kunnen oriënteren op de werksfeer binnen het bedrijf. In samenwerking met de begeleiders vanuit het bedrijf zal de stagiair de werkplek opzetten zodat er zelfstandig verder gewerkt kan worden.

Verder zullen er tijdens de initiatie ook afspraken gemaakt worden tussen de begeleiders van het bedrijf en de begeleiders van de Haagse Hogeschool om het verloop van het project duidelijk bij te houden. Hiervoor zullen afspraken gemaakt worden zoals wekelijkse afspraken om vragen te beantwoorden, bedrijfsbezoeken en documentatie reviews.

De onderzoeksfase

Tijdens deze fase zal er gewerkt worden aan de theoretische achtergrond van het project. De stagiair verdiept zich verder in de theorie achter IaaS en de oplossing welke hiervoor beschikbaar zijn. Tijdens dit onderzoek zal er documentatie geschreven worden over de beschikbare oplossingen en welke het beste toepasbaar zal zijn voor de omgeving van i3D.net.

De uitvoeringsfase

In de uitvoeringsfase zal het resultaat van de onderzoeksfase uitgewerkt worden tot een Proof of Concept. Dit Proof of Concept zal i3D.net meer inzicht moeten geven over de gekozen oplossing zodat er besloten kan worden of dit is wat i3D.net wenst. Mocht dit Proof of Concept geaccepteerd worden kan deze uitgewerkt worden tot een definitieve opzet van de service.

De afronding

Bij de afronding zal alle documentatie welke voor i3D.net bestemd is overhandigt worden, dit zal dienen ter informatie en naslag voor i3D.net mochten zij andere medewerkers verder laten werken met de uiteindelijke oplevering.

4.3) Standaarden

Voor dit project zijn standaarden gedefinieerd met betrekking tot de documentatie en software.

Software:

- Putty
- VMWare Player
- Google Docs

De documentatie van het project zal voornamelijk geschreven worden vanuit Google Docs, de reden dat hiervoor gekozen is, is vanwege de flexibiliteit van de Docs omgeving.

Alle documentatie word meteen online opgeslagen en is voor de stagiair overal te bereiken.

Twee maal per week word van de documentatie in Google Drive een back-up gemaakt naar Dropbox, om ervoor te zorgen dat alle documentatie redundant opgeslagen word.

Verder voldoet de documentatie aan een aantal inhoudelijke standaarden:

1. Alle tekst in Algemeen Beschaafd Nederlands
2. Elk document bevat een duidelijke voorpagina met versienummering
3. Geschreven in gestandaardiseerde koppen en tekst formattering (Variërende tekstgrootte en kopstijl in Calibri)

4.4) Planning

De globale planning is als volgt

Afstudeer dossier schrijven		Alle fasen	15 dagen
Plan van Aanpak schrijven	100%	Initiatiefase + Requirements	4 dagen
Bedrijfsoriëntatie schrijven	100%	Initiatiefase + Requirements	4 dagen
Software onderzoek uitvoeren	100%	Design and Architecture	4 dagen
Gesprekken voeren met medewerkers	75%	Design and Architecture	3 dagen
Vastleggen eisen medewerkers en opdrachtgever	75%	Design and Architecture	3 dagen
Beschikbare documentatie software oplossingen uitzoeken	75%	Design and Architecture	3 dagen
Proof of Concept maken	100%	Development and Coding	15 dagen
Proof of Concept bespreken met opdrachtgever/werknemers	50%	Development and Coding	3 dagen
Proof of Concept testen	30%	Development and Coding	10 dagen
Documentatie over Proof of Concept schrijven	75%	Development and Coding	5 dagen
Advies geven over software pakket	35%	Quality Assurance and Software Testing + Afronding	1 dag
Implementeren	0%	Quality Assurance and Software Testing + Afronding	5 dagen
Overdragen	0%	Quality Assurance and Software Testing + Afronding	1 dag
Uitloop		Alle fasen	9 dagen

De uitgebreide uitwerking van deze planning is terug te vinden in het bijgevoegde planningsdocument.

5) Projectinrichting

In de komende hoofdstukken zal beschreven worden hoe het project is georganiseerd en wie welke verantwoordelijkheden draagt.

5.1) Projectorganisatie

Jan Dirk Schagen & Marinus Maris (afstudeerbegeleiders)

Respectievelijk de primaire en secundaire begeleiders vanuit school. De stagiair zal contact opnemen met de begeleiders om afspraken te maken m.b.t. het geven van feedback en andere verplichte onderdelen van het afstuderen. Verder zal het resultaat beoordeeld worden door deze begeleiders.

Rick Sloot (Opdrachtgever & Bedrijfsbegeleider)

De opdrachtgever zal aanspreekbaar zijn voor vragen met betrekking tot de werkzaamheden van de stagiair binnen het bedrijf. De begeleider zal tevens aanwezig zijn bij de eindbeoordeling van de stagiair.

Chris Zonneveld (Afstudeerder)

De afstudeerder zal de in het Plan van Aanpak beschreven taken uitvoeren en de taken welke tijdens het uitvoeren van de opdrachten mogelijk nog naar voren komen. De verantwoordelijkheden bestaan uit het initiatief houden om het project op juiste manier te laten verlopen, en de begeleiders hierbij op de hoogte te houden door middel van het tijdig maken van afspraken.

5.2) Informatievoorziening

De informatievoorziening is als volgt ingericht.

-Bedrijfscommunicatie

Om binnen het bedrijf te communiceren hanteer ik de methode die hier standaard gebruikt wordt door alle medewerkers. Voor formele communicatie wordt e-mail gebruikt, hiervoor heeft de stagiair ook een adres ter beschikking van i3D.net. Voor informele interne communicatie zijn een aantal mogelijkheden. Intern wordt voornamelijk gebruik gemaakt van Skype, voor korte vragen of gesprekken. Dit is een snelle manier om kort informatie op te vragen. Voor de problemen of vragen

die meer uitleg vereisen kan er ook gebeld worden, hiervoor is een telefoon beschikbaar op de werkplek van de stagiair.
Verder is er uiteraard altijd de mogelijkheid om collega's persoonlijk aan te spreken als deze op de afdeling aanwezig zijn.

-Communicatie naar de opleiding

De communicatie naar de begeleiders vanuit school zal voornamelijk via e-mail plaatsvinden, waar nodig zullen er gesprekken ingepland worden om elkaar direct te spreken. Mocht het nodig zijn kan er tevens telefonisch contact opgenomen worden met de begeleiders.

5.4) Kwaliteitsborging

Voor kwaliteitsborging worden een aantal controles bijgehouden.

-Schoolbegeleiding

De begeleiders vanuit school zullen voornamelijk controles aanhouden op basis van de documentatie welke aangeleverd word. Per relevant document zal er een conceptversie richting de begeleiders gestuurd worden, zodat deze op kwaliteit gecontroleerd kan worden.

Verder kunnen de begeleiders aan de hand van gesprekken een beeld krijgen van de voortgang van het project, mochten zij hier valkuilen in ontdekken of voorspellen kunnen zij dit tijdens deze gesprekken aangeven zodat de afstudeerder tijdig zijn werkwijze kan aanpassen

-Werkbegeleiding

Op de werkplek zullen de controles voornamelijk plaatsvinden op het gebied van documentatie en de uitwerking van het Proof of Concept. De documentatie zal net als voor de schoolbegeleiding in conceptfase besproken worden, voordat deze gefinaliseerd worden.

Verder kunnen de softwarepakketten en Proof of concepts regelmatig bekeken worden door medewerkers van i3D.net om hun mening continu mee te nemen in het ontwikkelen van de het Proof of Concept.

6) Risicofactoren

De volgende factoren kunnen in de weg staan van het succesvol voltooien van het project:

6.1) Risico's

-De opdracht

- Tijdsduur, de doorlooptijd van 17 weken voor 1 fulltime stagiair kan mogelijk onvoldoende zijn om de opdracht geheel uit te voeren.
- Geen juiste oplossing, er is altijd een mogelijkheid dat geen van de beschikbare softwareoplossingen geschikt zijn voor dit project.

-De aanpak

- Verwachtingsproblemen, het kan voorkomen dat om wat voor reden dan ook er geen juiste communicatie rondom de opdracht uitgevoerd is, wat een onjuiste verwachting kan creëren bij de opdrachtgever.

-De projectinrichting

- Onvoldoende resources, mochten de benodigde resources niet aanwezig zijn/aan te vragen zijn, kan dit zorgen voor vertraging van de uitvoering van het project.

6.2) Maatregelen

-De opdracht

- Tijdsduur, duidelijke planning aanhouden en op tijd ingrijpen bij problemen.
- Geen juiste oplossing, bij mogelijkheid dat dit zich voordoet zal er op tijd een gesprek moeten plaatsvinden met de opdrachtgever om verder verloopt te bespreken.

-De aanpak

- Verwachtingsproblemen, regelmatig een gesprek houden met de opdrachtgever om voortgang bij te houden en het doel te bespreken moet dit voorkomen.

-De projectinrichting

-Onvoldoende resources, Het op tijd inventariseren en aanvragen van resources moet dit voorkomen, mocht dit toch voorkomen zal er rekening gehouden moeten worden met uitloop.

7) Kosten en baten

De volgende kosten en baten komen kijken bij dit project.

7.1) Kosten

De stagiair zal gedurende 17 weken fulltime werk verrichten voor i3D.net, de bijbehorende te besteden manuren zijn als volgt.

[17 weken * 5 dagen] * 8uur = 680 uren. [minus officiële vaste vrije dagen]

Van deze 17 weken zal de stagiair 4 dagen vast op het Datacenter van i3D.net aanwezig zijn, de laatste dag zal door de stagiair gebruikt worden om afspraken richting school te plannen en aan het afstudeer dossier te werken.

Verder zal de stagiair gebruik maken van de werkplek welke door i3D.net beschikbaar is gesteld. Andere benodigde hardware omvat servers om het Proof of Concept op te hosten en eventuele testservers.

Na uitvoering van het project zal i3D.net, indien ze de oplossing accepteren, verder de exploitatie kosten voor operatie en onderhoud moeten rekenen.

7.2) Baten

Onder de baten die i3D ondervind van dit project vallen onder andere de volgende punten:

- Uitbreiden van dienstenaanbod voor klanten.
- Verhogen van efficiency waarmee deze diensten uitgevoerd worden.
- Hogere kwaliteit van bestaande diensten indien deze gekoppeld worden.
- Betere service aan klanten vanwege verhoogd gebruiksgemak.

Bedrijfsoriëntatie i3D.net

Afstuderen Technische Informatica

Auteur: Chris Zonneveld

Studentennr: 09032835

Plaats: Rotterdam

Datum: 9-2-2015

Versiebeheer

[illegible]

Inhoud

[Inleiding](#)

[De omgeving](#)

[Stabiliteit](#)

[Markt](#)

[Agressiviteit](#)

[De organisatiedoelen](#)

[Missie](#)

[Bedrijfsdoelen](#)

[De Taken \(werkprocessen\)](#)

[Mate van voorspelbaarheid](#)

[Complexiteit](#)

[De technologie](#)

[Aspecten van technologie](#)

[De mensen in de organisatie](#)

[Sociaal- vs Technisch systeem](#)

[Organisatieontwerp](#)

[De organisatiecultuur](#)

[Rolmodellen](#)

[Statussymbolen](#)

[Gedrag patronen](#)

[Normen en Waarden](#)

[De organisatiestructuur](#)

[Afsluiting: Mijn rol als TI-er](#)

Inleiding

Tijdens het stagelopen word er van een student Technische Informatica verwacht dat deze zich kan plaatsen binnen een bedrijf en een duidelijk beeld kan vormen bij wat voor organisatie hij/zij aan de slag gaat. Dit is erg belangrijk in het bedrijfsleven omdat je zelfstandig je plaats binnen een organisatie zal moeten kunnen vinden. Verder is dit ook toepasbaar op het beklimmen van de bedrijfs ladder, mocht je naar een hogere functie willen stijgen, is het voordelig om een duidelijk beeld te hebben bij het doel en de organisatie van een bedrijf.

In dit document word stilgestaan bij de volgende onderdelen:

- Omgeving
- Organisatiedoelen
- Taken
- Technologie
- Mensen in de organisatie
- Organisatiecultuur
- Organisatiestructuur

Uiteindelijk zal ik nog stilstaan bij mijn eigen rol binnen het bedrijf.

De omgeving

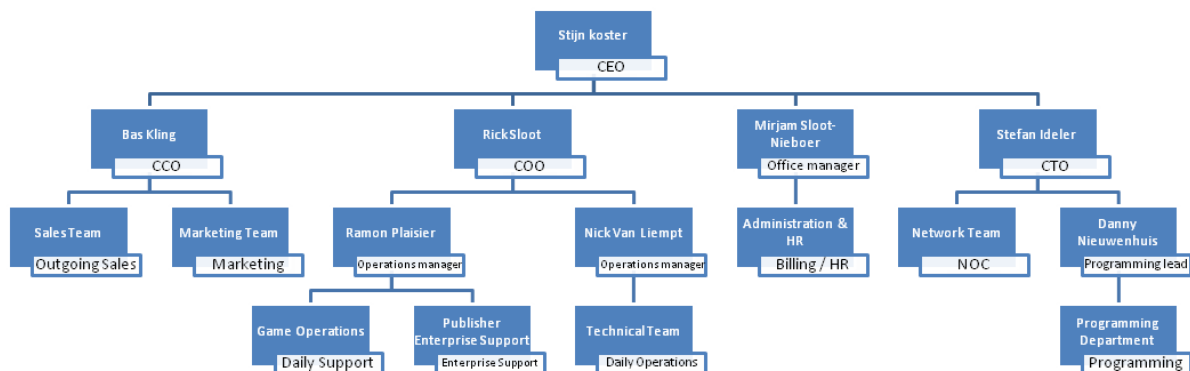
Tussen i3D en hun omgeving bestaat een uitwisselingsrelatie. i3D betreft haar technische producten van een leverancier, neemt personeel in dienst, verkoopt haar diensten, houdt zich aan de wetgeving, maakt reclame enzovoort.



De omgeving speelt een cruciale rol bij het wel en wee van i3D. i3D dient dus goed op de omgeving in te spelen en waar mogelijk zelfs te trachten deze naar haar hand te zetten. Dit vraagt om inzicht in en begrip voor de kenmerken van haar omgeving.

Aan de hand van een drietal kenmerken moet je de omgeving van je stage-organisatie kenmerken: Stabiliteit, markt en agressiviteit.

Organogram



Stabiliteit

De omgeving waarin i3D werkt zou ik op het moment stabiel noemen. De reden hiervoor is de nature van de werkzaamheden die i3D verricht. i3D verzorgt een divers scala aan diensten voor particulieren en bedrijven. Hun voornaamste dienst is het hosten van game servers, dit is over het algemeen ook de stabielste dienst. Het afnemen van deze dienst is voor i3D niet veel werk en is voor de kanten gemakkelijk aan te vragen. Er zijn voorbeelden te noemen zoals bij de release van een nieuwe multiplayer game, waarbij de werkdruk wat hoger komt te liggen i.v.m. onverwachte wijzigingen en opschaling naar aanleiding van de populariteit. Verder is het op het gebied van colocatie, de 2e grote dienst van i3D wat dynamischer. Er moet hard gewerkt worden om colocatie klanten binnen te halen, maar eenmaal binnen blijven de meeste klanten voor een langere periode. Ten slotte biedt i3D nog web hosting, dit is één van de minder grote diensten, de klanten basis hiervan is vrij constant en vereist geen enorme inspanning.

De core van i3D is dus zeer stabiel, naast deze diensten blijft i3D een hosting provider en zijn ze dus ook constant nieuwe ontwikkelingen aan het onderzoeken, zoals cloud hosting. Het toevoegen van dit soort diensten aan hun huidige aanbod zorgt voor enige dynamiek in de omgeving waarin i3D opereert vanwege de constante werkdruk om ook onderzoek te doen naar nieuwe technieken.

Markt

i3D biedt een groot scala aan services en solutions aan hun klanten, per tak beiden zij het volgende.

Dedicated Infrastructures

- Dedicated servers
waaronder Single Socket, Dual Socket en Quad Socket servers
- Managed servers
Omvat monitoring, OS beheer, netwerkbeheer, debugging en patching
- Load balancing
Verdelen van workload over meerdere virtuele of dedicated servers.
- Firewall & VPN
Firewalling van betrouwbare merken als Fortinet en Cisco

Colocatie

- **Rackspace**
Er kan onder andere voor Single Servers, Quarter Racks, Half Racks en Full Racks rackspace afgenomen worden.
- **Cages**
Indien extra beveiliging gewenst is kan er ook een gehele cage afgenomen worden.
- **Suites**
Mocht beveiliging een hoge eis hebben samen met het hoogste controle niveau, is het mogelijk een gehele suite op maat te laten maken.

Gaming

- **Games voor PC**
Er kunnen voor honderden games servers afgenomen worden. Deze worden gehost op High-end Dell Intell Xeon servers met 1,000mbit uplinks met real-time server statistieken. Verder is dit allemaal te managen vanuit het i3D control panel.
- **Voice for PC**
Vaak worden er naast de gameservers ook meteen servers afgenomen voor voice applicaties, i3D heeft de mogelijkheid om voor Mumble, Ventrilo en Teamspeak 3 servers af te leveren.
- **Xbox & Playstation**
i3D stelt ook het XLSP platform beschikbaar om xbox servers te hosten, maar beschikt ook over platformen voor o.a. Playstation 3, Playstation 4 en Nintendo

Web Hosting

- **Domeinnamen**
Er kunnen gemakkelijk domeinnamen gekocht worden via i3d voor een tientallen top level domains, mocht er ook web hosting af genomen worden zijn de domeinen gratis
- **Web Hosting**
Er zijn verscheidene webhosting pakketten beschikbaar geschikt voor kleine bedrijven/enkele websites tot Power Users/Grote forums.

Backup & Enterprise

- **FTP & Software Backup**
Mocht data een beveiligde remote backup nodig hebben, bied i3D een backup service. Hiermee kan via FTP of een software client gemakkelijk een remote backup gemaakt worden.
- **Big Data & Hadoop**
In samenwerking met KPMG bied i3D Big Data-as-a-service.
- **Managed Hosting**
Mocht er de wens zijn om het IT-Beheer binnen een bedrijf uit handen te geven, is het mogelijk i3D al het beheer te later verzorgen. Hierbij kan er gebruik gemaakt worden van de expertise van de i3D medewerkers om van 24/7/365 beschikbaarheid garant te zijn.
- **Fiber to the Office**
Mocht er een stabiele en snelle verbinding vanaf kantoor naar het datacenter nodig zijn, is het mogelijk om een glasvezelverbinding aan te laten leggen.

Agressiviteit

De hosting sector is op het moment een vrij aggressive markt. Er zijn veel hosting providers van verschillende groottes die verschillende maten van support bieden aan hun klanten. i3D heeft een redelijke niche te pakken met hun gameserver hosting waar ze groot mee geworden zijn, maar dit betekent niet dat ze hun hosting kant kunnen negeren. Om hun groei op peil te houden zullen ze hier dus ook op moeten blijven focussen om mee te gaan in de markt.

De organisatiedoelen

We beschouwen i3D als een doelgericht samenwerkingsverband. In de literatuur zijn er verschillende benaderingen en definities van het begrip 'organisatiedoel'. Deze variëren van: 'een organisatie heeft één hoofddoel, het streven naar maximalisatie van de winst', tot 'organisaties hebben geen doelen, alleen mensen hebben doelen'. Hierin wordt gekozen voor een middenweg: Een organisatie heeft doelen die afhankelijk zijn van de invloeden van de mensen in de organisatie en van de omgeving.

Iedere organisatie heeft dus een aantal doelen die worden nagestreefd. Zonder die doelen gaat een organisatie zwakken. Om de doelen te realiseren heeft de leiding een missie geformuleerd en (bewust of onbewust) een strategie uitgezet, die sturing geeft aan het functioneren en de inrichting van de organisatie.

Missie

De missie van i3D is om te groeien in de IaaS en managed hosting sector en hierbij zoveel mogelijk klanten te voorzien van kwaliteits service op het hosting gebied.

Zij proberen dit waar te maken door constant hun kennis op het gebied van verscheidene technieken te verhogen bijvoorbeeld door middel van trainingen bij leveranciers en het uitwerken van test opstellingen waar praktische ervaring nodig is.

Bedrijfsdoelen

Op het moment is i3D vooral bezig met het verder uitwerken van hun hosting diensten. Op de agenda staat op het moment vooral het uitwerken van de virtualisatie omgeving, waarbij huidig gewerkt wordt aan een Azure omgeving, welke eventueel met een IaaS oplossing geïntegreerd wordt samen met een hypervisor als KVM.

Verder zorgt i3D ervoor dat hun klanten constant de laatste ontwikkelingen op het gebied van game servers kunnen afnemen. Dit in samenwerking met de leveranciers van deze games om dit zo efficiënt mogelijk te laten verlopen.

De Taken (werkprocessen)

Mate van voorspelbaarheid

Over het algemeen is er een redelijke hoge mate van voorspelbaarheid voor de standaard werkzaamheden van i3D, aangezien dit bijna allemaal geautomatiseerd verloopt. Buiten de standaard werkzaamheden stopt het echter wel snel met de voorspelbaarheid.

Vanwege de enorme hoeveelheid aan verschillende omgevingen die klanten draaien, is het onmogelijk om de verloop van een project met 100% zekerheid te voorspellen. Gaat het bijvoorbeeld om het regelen van managed hosting voor een klant dan is van te voren maar een klein deel van de daadwerkelijke werkzaamheden in te schatten. Het grootste gedeelte is pas te zien na dat er daadwerkelijk aan de slag gegaan wordt met de omgeving. Dit zorgt ervoor dat er erg veel eigen input nodig is in projecten en dat er regelmatig eigen onderzoek gedaan moet worden naar specifieke oplossingen.

Complexiteit

De complexiteit van de werkzaamheden binnen i3D verschilt enorm. Er zijn de ene dag een hoop werkzaamheden zoals het opzetten van gameservers, wat allemaal zoals eerder genoemd redelijk geautomatiseerd is. De andere dag kan er een plan opgezet worden om in een datacenter in het buitenland extra racks bij te plaatsen, waar dus zeker weken/maanden voorbereidings werk aan verbonden zit. De verschillende soorten werkzaamheden zorgen dus ook voor een algemeen onvoorspelbare complexiteit.

De technologie

De taak geeft aan wat er gedaan moet worden, de technologie geeft aan hoe de taak moet worden uitgevoerd en welke kennis daarvoor vereist is. Technologie heeft dus betrekking op twee zaken:

1. De toegepaste technieken om producten en diensten te leveren
2. De kennis van de mensen omtrent het toepassingsgebied en de gebruikte technieken.

Aspecten van technologie

We kunnen technologie op een spectrum van eenvoudig tot complex indelen. Hierbij is het wel zaak om met beide aspecten, technologie en kennisniveau, rekening te houden. Het kennisniveau van een advocaat is hoog, maar hij gebruikt eenvoudige technologie, terwijl zeer geavanceerde technieken in een productielijn van auto's worden toegepast met productiepersoneel van een tamelijk laag opleidingsniveau. Het ontwerpen van een nieuw model tuinbouw robot en het daarbij behorende productieproces vraagt echter hoog opgeleide ingenieurs.

Een organisatie als 'hightech' bestempelen slaat zowel op de complexe technieken als op een hoog kennisniveau.

Aspecten van Technologie		Kennisniveau	
		Laag	Hoog
Technologie	Complex	Productielijn van auto's	(hightech) ruimtevaart onderneming
	Eenvoudig	Schoonmaakbedrijf	Advocaten

Als ik het bovenstaande schema aanhoud, zou i3D als een hightech organisatie beschreven kunnen worden.

De mensen in de organisatie

Zie hieronder 2 organisatie modellen

	Machinemodel (Taylor)	Sociotechnisch model (HRM)
Sociaal- vs Technisch systeem	Het 'technisch' systeem is primair. Produceren en dienstverlening wordt gezien als een technisch vraagstuk. De mens is middel bij het produceren en de dienstverlening.	Mensen worden gezien als bouwstenen van de organisatie. Het streven is naar gezamenlijke optimalisering van het sociale en technische systeem.
	Nadruk op externe coördinatie en controle door leidinggevenden en stafafdelingen.	Nadruk op interne coördinatie en controle door leden van een team.

Organisatie- ontwerp	Organisatieontwerp is gebaseerd op totale specificatie.	Organisatieontwerp laat ruimte voor uitvoerende teams. (professionele ruimte)
	Organisatieontwerp richt zich op maximale arbeidsdeling en specialistische vakkundigheid van individuele werknemers.	Organisatieontwerp streeft naar minimale arbeidsdeling en brede vakkundigheid en inzetbaarheid van teamleden.
	Organisatieontwerp beoogt maximale productieresultaten (trots op prestatie, bonus)	Organisatieontwerp beoogt maximale betrokkenheid bij het werk. (trots op het werk)

Sociaal- vs Technisch systeem

Dit is voor i3D een interessante vraag. Aan de ene kant is hun organisatie 100% afhankelijk van hun technisch diensten. Aan de andere kant kan het bedrijf niet zonder de medewerkers die deze diensten verzorgen. Ik zou in het geval van i3D toch zeggen dat het een Socio-technisch model is, omdat de nadruk hier ligt op het ontwikkelen van de benodigde kennis van de medewerkers en niet zo zeer op het functionerend houden van een productie omgeving.

Organisatieontwerp

Per afdeling binnen i3D is veel gespecialiseerde kennis aanwezig, hiervoor is gekozen vanwege de hoeveelheid vlakken van expertise die beschikbaar zijn binnen de ICT-wereld. Dit gaat dus in tegen het socio-technische model, maar in het geval van i3D vind ik dit een uitzondering vanwege de aard van het werk. De afdelingen werken verder onderling veel samen aan projecten waar gecombineerde kennis nodig is.

De organisatiecultuur

De organisatiecultuur is de gemeenschappelijke verzameling (gedrags)uitingen, opvattingen en overtuigingen over:

- Wie de inspirerende rolmodellen, de 'helden' van de organisatie zijn
- Wat de status-symbolen, de logo's e.d. zijn die tot uitdrukking brengen 'waar de organisatie voor -staat'
- Wat de meest typerende en kenmerkende gedragpatronen zijn van de medewerkers
- De normen en waarden waarmee de medewerkers hun gedrag vormgeven.

Rolmodellen

Er zijn geen rolmodellen binnen i3D, het is een klein-middelgroot bedrijf en alle medewerkers zijn in hun eigen respect erg belangrijk. Een vorm van een rolmodel wordt uiteraard wel door het management team en de directeur aangenomen. Echter zorgt het management team wel voor directe communicatie met de rest van de medewerkers, er is geen afscheiding van verticale communicatie.

Statussymbolen

i3D heeft een aantal status symbolen. Ten eerste hebben ze uiteraard hun eigen logo welke op vrijwel alle documentatie vertoont wordt. Er wordt van de werknemers verwacht dat deze representatief en verzorgd gekleed zijn. i3D zorgt hierbij voor werkkleding, waaronder polo's, sweaters en petten, verder is de kleding vrij voor de medewerkers. i3D maakt buiten het bedrijf reclame door op bijeenkomsten en evenementen gerelateerd aan gaming en hosting hun organisatie en diensten te promoten.

Gedragspatronen

Voor de werknemers van i3D is een personeelshandboek opgesteld, wat door de werknemers doorgenomen dient te worden bij het starten bij het bedrijf. Hierin staan algemene richtlijnen over hoe de werknemer dient om te gaan met collega's en wat te doen bij problemen en conflicten op de werkvloer. Dit is niet verschillend van wat ik zou verwachten bij een hosting bedrijf.

Normen en Waarden

Dimensies van normen & waarden (vrij naar Hofstede)						
4.1. Procesgericht	1	2	3	4	5	Resultaatgericht
Mijdt risico's Zo weinig mogelijk inspannen Elke dag beetje hetzelfde (grote administratieve eenheid)						Op gemak in onbekende situaties Uiterste best doen Elke dag nieuwe uitdaging (onderzoek en ontwikkeling)
4.2. Mensgericht	1	2	3	4	5	Taakgericht
Houdt rekening met persoonlijke problemen. Neemt verantwoordelijkheid voor welzijn medewerkers Groepen nemen belangrijke beslissingen (organisaties waarin voornamelijk interne normen gelden)						Sterke druk om werk af te krijgen. Vooral interesse in het werk dat wordt afgeleverd, minder in het welzijn van de medewerkers. Individueel nemen belangrijke beslissingen. (organisaties waarin voornamelijk externe normen gelden)
4.3. Organisatie-gebonden	1	2	3	4	5	Professioneel-gebonden
Werknemers identificeren zich met hun organisatie. Organisatie neemt mensen in dienst op basis van sociale klasse en/of schoolachtergrond. Normen en waarden van het werk gelden ook thuis.						Werknemers identificeren zich met hun beroep. Organisatie neemt mensen in dienst op basis van geschiktheid voor het werk. Privé-leven is ieders eigen zaak. Denkt jaren vooruit. (Hightech-bedrijf met hoogopgeleiden)
4.4. Open	1	2	3	4	5	Gesloten
Openheid tegenover nieuwkomers Nieuwe medewerkers voelen zich snel thuis						Geslotenheid en geheimzinnigheid zelfs voor eigen medewerkers. Nieuwe medewerkers voelen zich niet snel thuis. (veiligheidsdiensten)
4.5. Losse controle	1	2	3	4	5	Strakke controle
Niet kosten bewust Houdt zich bij benadering aan vergadertijden Vaak grappen over werk en bedrijf						Kostenbewust Houdt zich sterk aan vergadertijden Praat serieus over bedrijf en werk (leverancier risicovolle producten-diensten)
4.6. Klantoriëntatie	1	2	3	4	5	Normatieve oriëntatie
Komt tegemoet aan klantwensen Een tevreden klant is belangrijker dan je strikt aan procedures houden. Pragmatische houding tegenover ethiek. (service afdeling, organisatie in sterk concurrerende markt)						Past procedures correct toe Aan procedures houden is belangrijker dan klanttevredenheid. Hoge normen op het gebied van ethiek (justitie, politie, hoger onderwijs)

i3D is een resultaat gericht bedrijf, elke dag krijgen de medewerkers te maken met nieuwe problemen en oplossingen met hun hosting klanten. Vanwege de gespecialiseerde kennis van de medewerkers is i3D erg mensgericht. Mochten werknemers problemen hebben word het aangemoedigd om hier hulp bij te vragen bij collega's waar dit nodig is, en willen medewerkers zichzelf door ontwikkelen (Denk aan opleidingen/trainingen) word dit ook zeker aangemoedigd.

Als bedrijf is i3D erg professioneel-gebonden, werknemers worden in dienst genomen op basis van hun kennis en expertise van hun technisch vakgebied. Met blik op de toekomst zijn de werknemers ook vrij om, zoals eerder aangegeven, zichzelf te ontwikkelen op hun vakgebied.

Tegen nieuwe medewerkers is de organisatie in de eerste instantie wat gesloten, wat vanuit een veiligheids perspectief ook niet meer dan normaal naar mijn mening, er word gewerkt met gevoelige informatie van klanten en omgevingen die 24/7 in productie zijn, hierbij is voorzichtige omgang met deze systemen een verplichting. Dit sluit ook aan op het feit dat klantoriëntatie belangrijk is voor i3D, klanten hebben tenslotte bij i3D contracten met SLA's afgesloten. Komen zij deze zelf niet na verliezen ze niet alleen de klant, maar is er ook een grote kans op slechte reclame voor i3D en hun services.

Mate waarin onzekerheid wordt vermeden	Hoog	'geoliede machine' (ziekenhuis)	'piramide' (bank)
	Laag	'dorpemarkt' (reclamebureau)	'familie' (orkest)
Culturele Typologie van organisaties (naar Hofstede)		Klein	Groot
		Machtsafstand	

i3D vermijdt zoveel mogelijk risico's richting hun klanten vanwege de aard van hun diensten. De reden dat klanten hun diensten en omgeving bij i3D onderbrengen is dat ze zich geen zorgen willen maken over de beschikbaarheid van hun diensten. Als i3D hier zelf niet mee om kan gaan heeft dat een slechte invloed hebben op hun service.

Om risico's te vermijden is goede communicatie een pre, dit is in i3D gerealiseerd door korte communicatie lijnen te houden op een vrij informele werkvloer. Alle communicatie richting klanten word via een ticket systeem met strikte regels uitgevoerd, waarnaast de belangrijke interne communicatie per mail uitgevoerd word. Alle andere interne communicatie word gevoerd door middel van Skype/telefonisch of er kan even naar desbetreffende afdeling gelopen worden om collega's in persoon aan te spreken.

Afsluiting: Mijn rol als TI-er

Bij i3D werk ik als afstudeer stagiair op de netwerk afdeling, de afdeling is wat rustiger vergeleken met de afdeling operations waar het wat rumoerig kan worden i.v.m. support calls. Hier heb ik dus rustig de tijd en omgeving om aan mijn eigen project te werken zonder te veel gestoord te worden. Op de afdeling heb ik wel de kans om mee te luisteren en meedenken met problemen en projecten die op dat moment besproken worden.

Om mijn eigen project uit te voeren heb ik een werkplek toegewezen gekregen van uit waar er prima zelfstandig gewerkt kan worden. Alle benodigdheden voor het project zijn door korte communicatie lijnen gemakkelijk te bemachtigen. De servers die aangevraagd zijn worden vlot opgeleverd met de benodigde configuratie zodat ik hier mee aan de slag kan, en toegang tot desbetreffende servers is goed geregeld d.m.v. een jumpbox afscheiding van de rest van het netwerk.

Software Onderzoek

Onderzoek naar IaaS oplossingen voor i3D.net



Opdrachtgever:
Rick Slood
i3D.net
Van Nelleweg 1, 3044BC
Rotterdam



Opdrachtnemer:
Chris Zonneveld
09032835
Rotterdamseweg 137, 2628AL
Haagse Hogeschool Delft

Versiebeheer

Versie	Auteur	Datum	Wijzigingen
Software Onderzoek 0.1	C. Zonneveld	16-2-2015	Eerste concept
Software Onderzoek 0.2	C.Zonneveld	26-2-2015	i3D.net template ingevoegd
Software Onderzoek 0.3	C.Zonneveld	2-4-2015	Content toevoegingen
Software Onderzoek 1.0	C.Zonneveld	29-4-2015	Laatste controle

Inhoudsopgave

[Versiebeheer](#)

[Inhoudsopgave](#)

[1.\) Inleiding](#)

[2.\) Wat is IaaS](#)

[3.\) Bestaande IaaS oplossingen](#)

[4.\) Community-analyse](#)

[4.1\) Posts & Threads](#)

[4.2\) Git contributie](#)

[4.3\) Conclusie Community-analyse](#)

[5.\) Functie analyse](#)

[6.\) Conclusie Functie Analyse](#)

[6.\) Conclusie Software Onderzoek](#)

1.) Inleiding

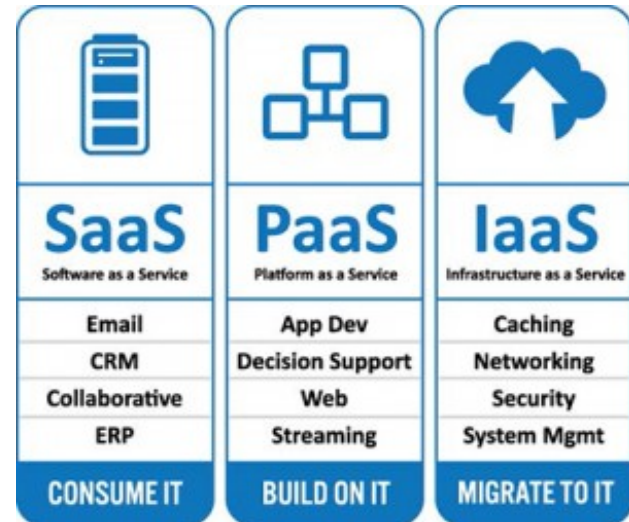
Het volgende document beschrijft de verscheidene beschikbare oplossingen op IaaS gebied en vergelijkt deze op mate van geschiktheid voor i3D.net.

Er zal eerst gekeken worden naar welke producten er op het moment relevant op de markt aanwezig zijn. Nadat er meer informatie is over welke producten mogelijk relevant zijn voor i3D.net zal ik gaan kijken welke hiervan het beste geschikt is voor i3D.net en welke er verder getest zal worden. Ten eerste zal markt relevantie onderzocht worden. De reden hiertoe is dat open-source projecten vaak komen en gaan, een pakket waar over een jaar niet meer actief aan gewerkt wordt door de community zal op lange termijn meer kosten en werktijd met zich mee brengen dan het in de eerste instantie bespaart.

Nadat de markt bekeken is, wordt er gekeken naar de individuele functionaliteit van de pakketten om te zien welke van de pakketten het beste geschikt is voor de functionaliteit welke i3D.net verwacht.

2.) Wat is IaaS

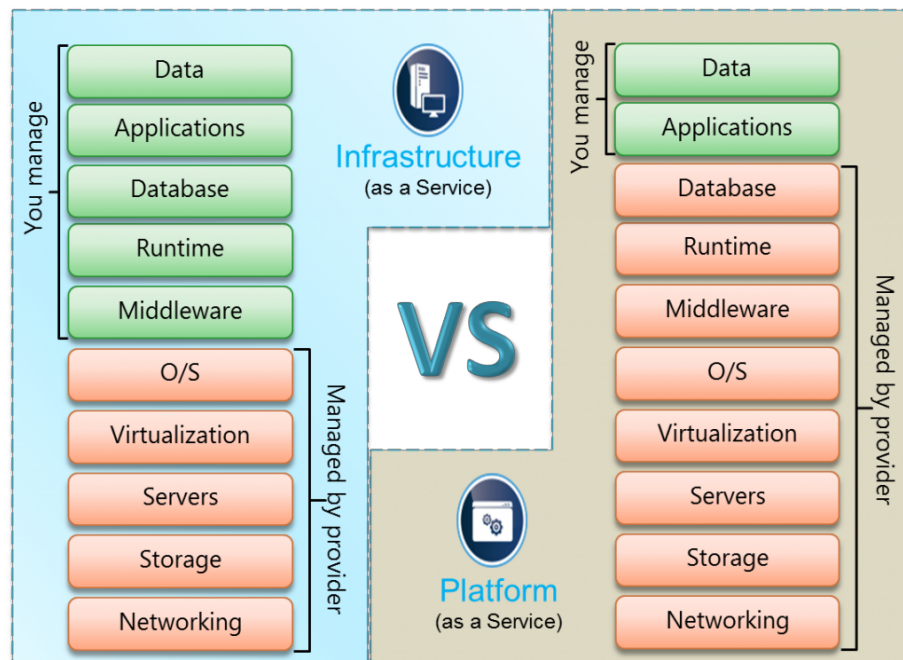
Voordat er verder onderzoek uitgevoerd kan worden naar IaaS software oplossingen moet er eerst onderzocht worden wat IaaS precies is. Ten eerste staat IaaS voor Infrastructure as a Service. IaaS is één van de drie fundamentele methoden waarop cloud computing diensten worden aangeboden. De andere 2 fundamentele methoden zijn PaaS (Platform as a Service) en SaaS (Software as a Service). Wat i3D.net hiermee aan hun klanten aanbied is een gevirtualiseerde computing omgeving welke over het internet te bereiken is.



Bij deze computing omgeving moet gedacht worden aan gevirtualiseerde machines waarbij de CPU's, RAM en andere fysieke specificaties virtueel gealloceerd worden. Verder worden ondersteunende diensten zoals externe storage, netwerkverbindingen, bandbreedte, IP adressen en loadbalancers ook gezien als diensten onder het mom IaaS. Het hosten van deze omgeving gebeurt op een fysieke backend, welke uit meerdere servers en netwerken bestaat, verspreid over meerdere datacenters. Dit in tegenstelling tot SaaS

waarbij een groot deel van het managen van de software back-end ook gedaan word door de provider

Met een IaaS opstelling word dus een kosten efficiënt en makkelijk schaalbaar netwerk opgezet waar de complexe netwerk werkzaamheden en de kosten van het onderhouden hiervan bij de cloud provider liggen.



Dit zorgt voor een oplossing waar klanten hun omgeving kunnen opschalen indien dit nodig is, zonder dat ze de hoofdprijs betalen voor permanente extra capaciteit. Er kan simpelweg voor een aantal uren/dagen/weken extra capaciteit bij gekocht worden waarna dit weer gemakkelijk opgezegd kan worden.

De voordelen die klanten ondervinden van IaaS zijn dus als volgt:

- **Schaalbaarheid**
Er is capaciteit beschikbaar wanneer de klant dit nodig heeft, zonder vertraging van orders.
- **Geen hardware investering**
Klanten hoeven geen grote hardware kosten te rekenen, dit valt voor de hosting provider.
- **Pay-as-you-go**
Je betaalt voor hoelang je gebruik maakt van je diensten, dus niet als deze inactief zijn.
- **Overal bereikbaar**
Zolang de klant over een internetverbinding beschikt, is zijn omgeving altijd beschikbaar
- **Fysieke beveiliging van datacenters**
Het datacenter zorgt ervoor dat alle fysieke beveiliging geregeld is.
- **Redundantie**
Switchen en servers worden door de provider redundant uitgevoerd.

3.) Bestaande IaaS oplossingen

Om te starten met het onderzoek is er gekeken welke IaaS oplossing er op het moment beschikbaar zijn.

Eucalyptus



Een open-source pakket om cloud hosting te faciliteren met een sterke focus op integratie op AWS public cloud.

Cloudstack



Een open-source pakket wat startte als VMOps en later overgenomen is door Citrix waarna het project opgenomen is aan het Apache incubator programma.

Openstack



Het Openstack project is een open-source pakket wat geheel open is en voor iedereen toegankelijk is om aan deel te nemen. Het project wordt beheerd door de Openstack Foundation welke afhankelijk is van de contributies van organisaties welke zich aansluiten bij de Foundation.

4.) Community-analyse

Open-source projecten hebben de eigenschap dat ze erg community-driven zijn. Er worden projecten opgezet waarbij gehoopt wordt dat de community het project interessant genoeg vindt om hier zelf aan bij te dragen. Het idee hierbij is dat hoe meer mensen er naar een open platform kunnen kijken en bijdragen, hoe sneller problemen hierin gevonden en opgelost worden. Een probleem dat hiermee gemeden wordt is als volgt: Een closed-source software provider heeft software draaien bij meerdere bedrijven, een kwaadwillende derde partij vindt hier een beveiligings kwetsbaarheid in. Het lek komt aan het licht en de gegevens van de bedrijven zijn niet meer veilig. Voor de bedrijven zit er niets anders op dan wachten tot de software provider een patch heeft uitgebracht om het lek te dichten. Had dit bij open-source software voorgekomen was er ten eerste een grote kans dat het lek al gevonden kon worden voor er misbruik van gemaakt kan worden. Mocht er alsnog een lek aan het ontdekt worden kunnen gebruikers zelf de code inzien van de software en dit waar nodig zelf patchen en oplossen waardoor de tijd tot het probleem opgelost is potentieel veel lager ligt.

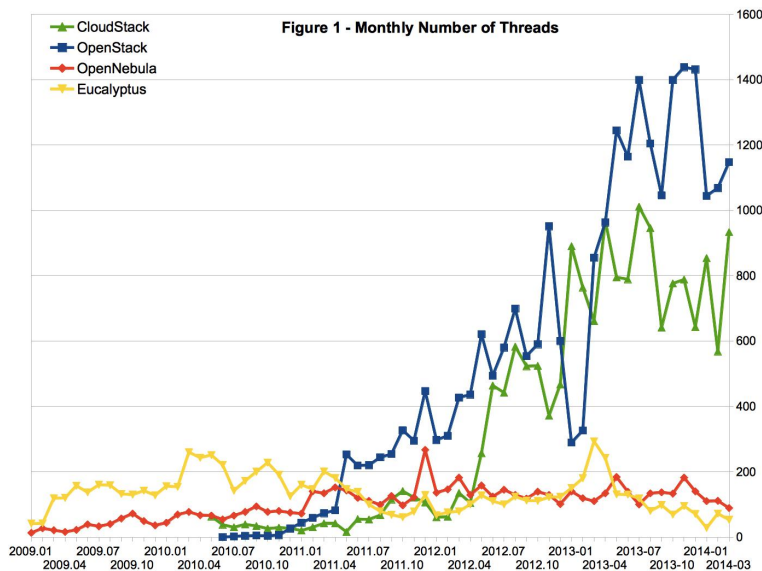
Naast het oplossen van security issues is de insteek van gebruikers erg waardevol voor een open-source project. Mochten meerdere mensen problemen hebben met bepaalde functionaliteit, kunnen zij samenwerken om deze functionaliteit te verbeteren of alternatieve functionaliteit in het product in te bouwen. De community is dus de drijvende kracht achter veel open-source pakketten, vandaar dat het eerste punt waar de drie pakketten op beoordeeld worden de community is.

Bij het zoeken naar informatie ben ik het onderzoek “Open Source IaaS Community Analysis - Openstack vs OpenNebula vs Eucalyptus vs Cloudstack” van Qingye Jiang geraadpleegd. Dit is een objectief onafhankelijk onderzoek naar de community's achter de respectievelijke open source producten. Dit onderzoek geeft een duidelijk inzicht op de huidige staat van de community's gezien dit onderzoek sinds 2011 per kwartaal uitgevoerd wordt. Er is dus duidelijk te zien op wat voor manier de gekozen pakketten zich ontwikkeld hebben over de afgelopen 4 jaar. Deze informatie is waardevol om te voorspellen of het project de komende tijd levendig blijft of dat het project simpelweg niet van de grond af komt.

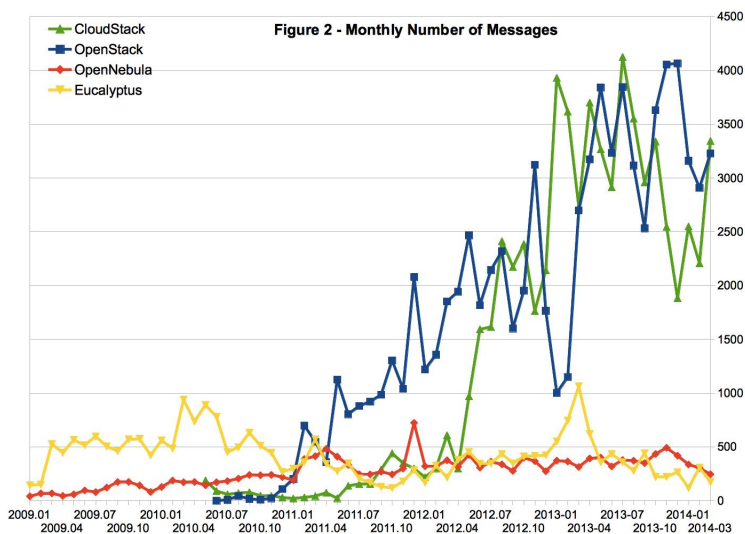
In het onderzoek worden een viertal pakketten onderzocht, drie van de pakketten hiervan worden in dit project onderzocht. Het vierde pakket, OpenNebula wordt niet onderzocht omdat deze bij de eerste globale selectie van pakketten buiten beschouwing is gesteld (De reden hiervoor is o.a. te zien in de volgende diagrammen, maar zal niet verder toegelicht worden).

4.1) Posts & Threads

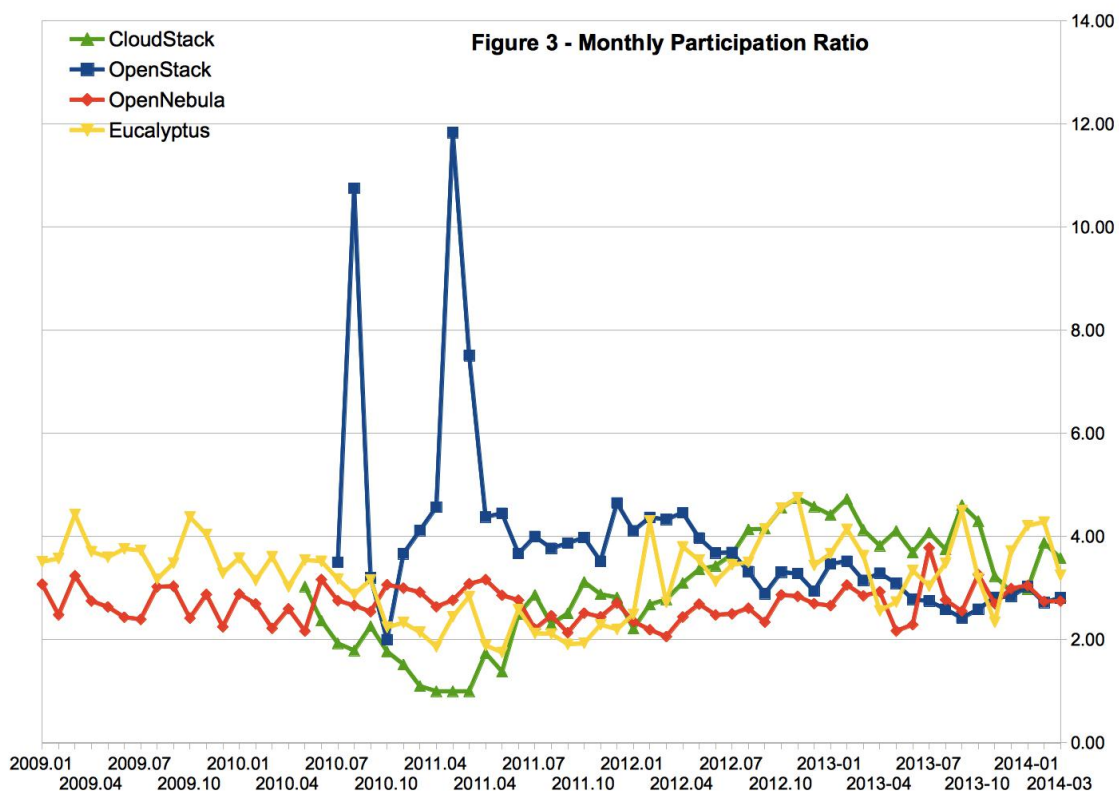
De volgende 2 grafieken geven aan wat hoeveel berichten en threads er geplaatst zijn in de fora van de respectievelijke producten. In figuur 1 is er te zien hoeveel threads er geplaatst zijn, figuur 2 geeft weer hoeveel berichten er geplaatst zijn.

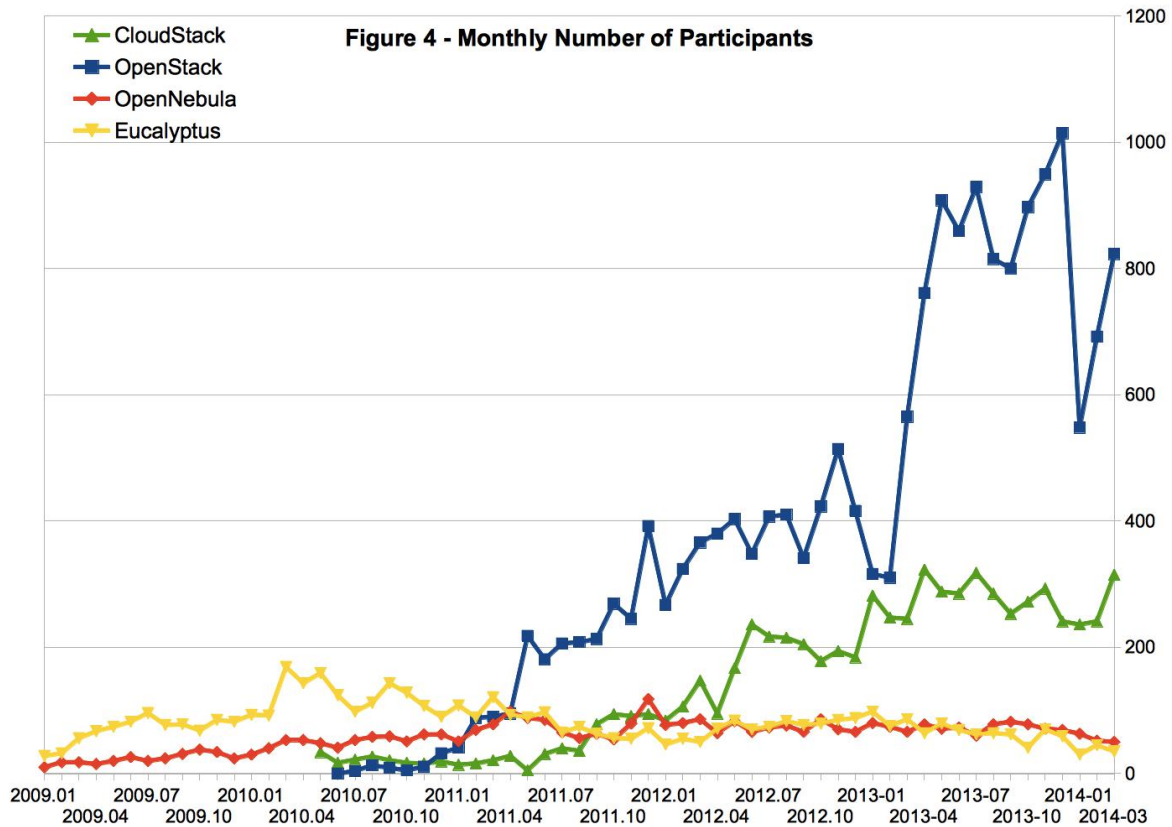


Hieruit kunnen 2 conclusies getrokken worden, namelijk dat in de afgelopen 3 jaar Openstack en Cloudstack beide een stijgende lijn hebben als het gaat om de discussies in de community. Beiden pakketten tonen een constantie groei in het aantal berichten. Als tweede punt is er te zien dat alhoewel Eucalyptus al langer in ontwikkeling is dan Cloud en Openstack, de community er niet op gegroeid is. Het heeft een vrij consistente lijn als het gaat om het aantal posts en threads.

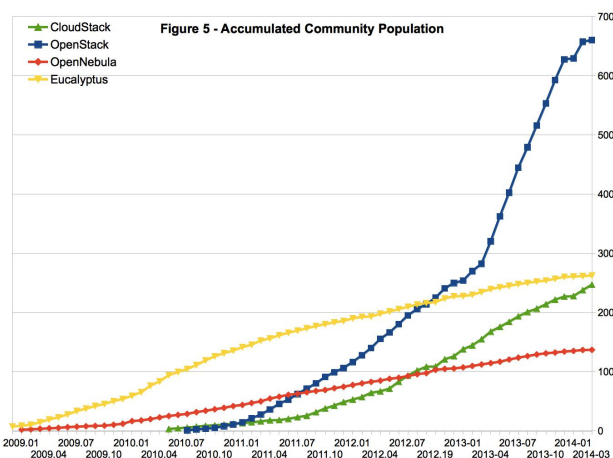


Aan de hand van deze 2 grafieken kan een participatie ratio gehaald worden, deze word in figuur 3 weergegeven. Als het aantal topics hoger is dan de hoeveelheid reacties is er een lage participation rate; er word weinig gereageerd op threads. Helaas is deze grafiek niet geheel waardevol als statistiek, een laag aantal replies op een thread kan namelijk een indicatie zijn van twee dingen, ten eerste kan het zijn dat problemen simpelweg snel verholpen worden. Ten tweede kan het aangeven dat er weinig community response is voor vragen. Helaas verschaffen de cijfers geen duidelijk beeld over de daadwerkelijke impact van de participatie ratio en liggen de statistieken erg dicht op elkaar vanwaar er besloten is om figuur 3 niet zwaar op te nemen in de beoordeling.



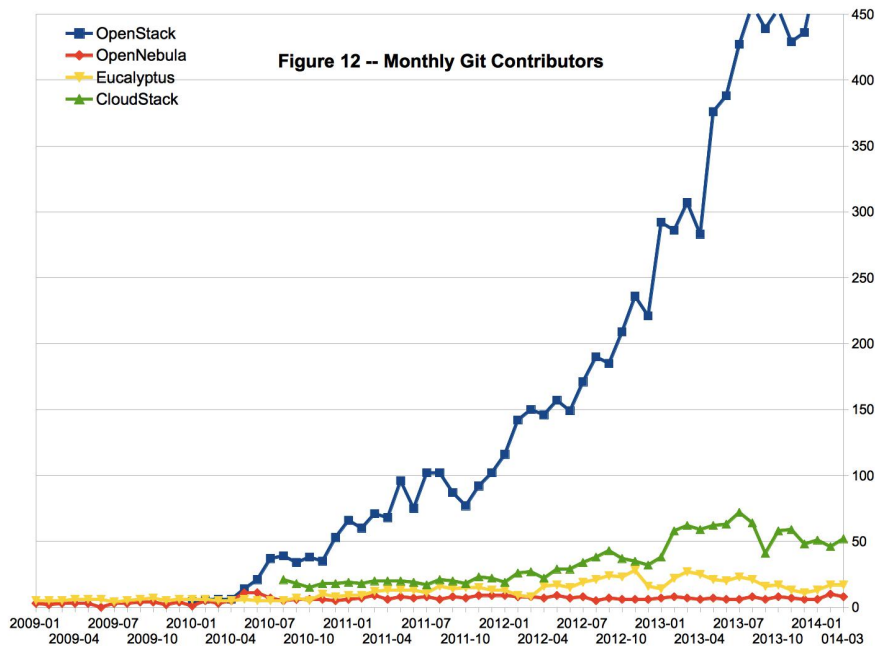


In figuur 4 is te zien hoeveel personen er maandelijks discussiëren. Er is duidelijk te zien dat het aantal deelnemers voor Openstack een stuk hoger ligt dan die van de andere projecten, Cloudstack ligt echter ook weer een stuk hoger dan Eucalyptus. Er is duidelijk te zien dat over de afgelopen paar jaar de Openstack community nog stevig aan het groeien is, Cloudstack groeit ongeveer 50% zo snel. Wat interessant is is wel dat Cloudstack een kleinere community heeft dan Openstack, maar ongeveer evenveel discussies hebben.



Verder is het het nog waard om te noemen dat de Openstack community zelf een flinke groei heeft ondervonden zoals te zien is figuur 5, er zijn ruim 2x zoveel members dan de andere projecten. Dit gaat echter wel om de gehele populatie, dus ook mensen die van pakket gewisseld zijn, echter is hier wel uit op te maken dat er in de afgelopen 2 jaar sterke interesse is naar Openstack gezien hoe hard de trend gestegen is.

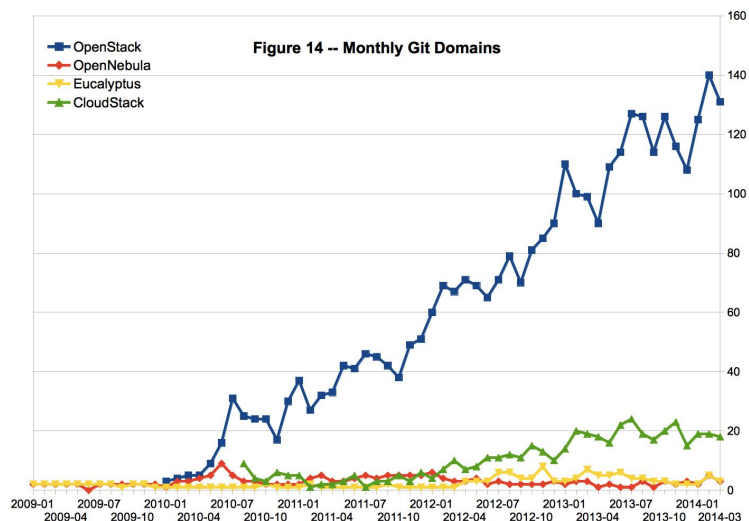
4.2) Git contributie

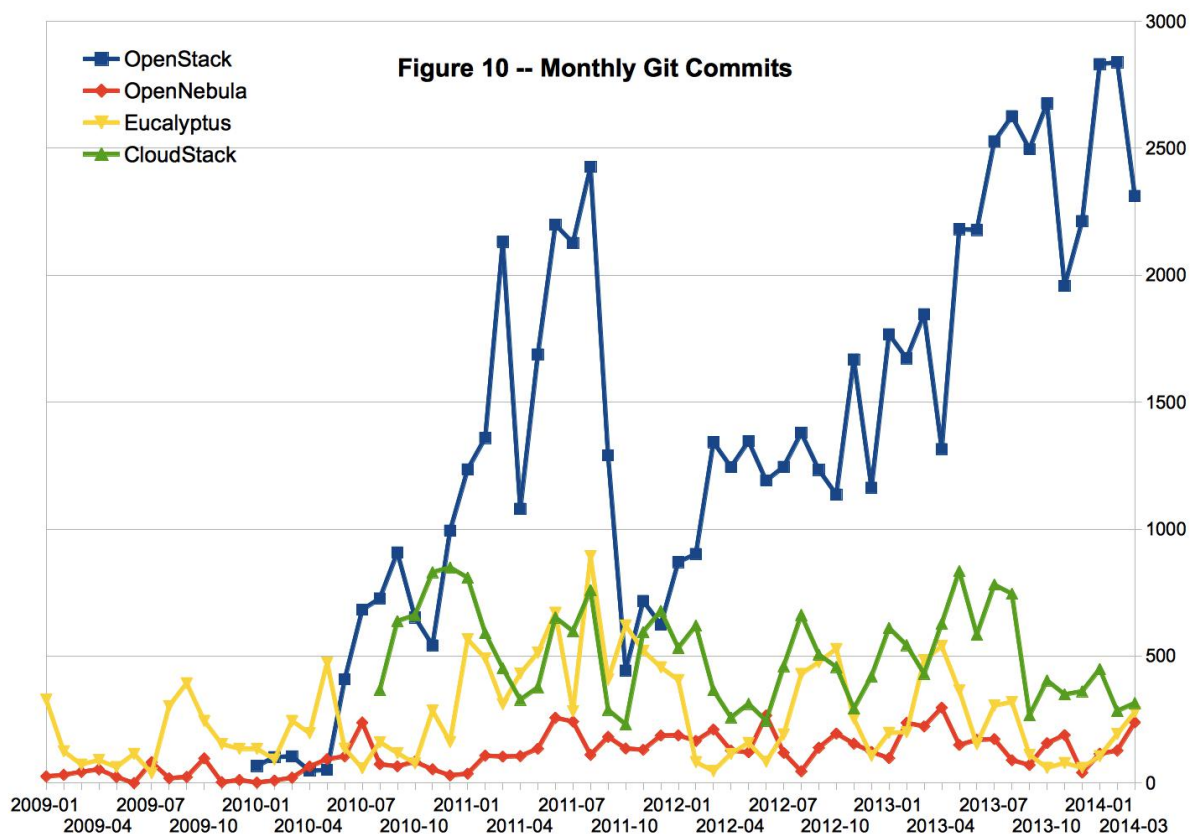


In Figuur 12 is te zien hoeveel maandelijkse git contributors elke project heeft. Over het algemeen heeft Openstack een flink hogere populatie aan contributors en groeit deze nog steeds snel. Cloudstack geeft ook wat groei weer maar significant minder dan Openstack. Tenslotte is te zien dat Eucalyptus een vrij kleine populatie heeft welke eigenlijk ook niet significant groeit.

Een groot aantal contributors kan aangeven dat een project de support heeft gekregen van een grotere organisatie, echter geeft figuur 14 weer dat dit bij Cloudstack en Openstack niet het geval is.

Het is een diverse user base wat aangeeft dat deze projecten breed inzetbaar zijn en niet alleen in één specifieke bedrijfstak succesvol geïmplementeerd kan worden maar ook geschikt is voor diverse bedrijven.





Ten slotte zien is er in figuur 10 nog de maandelijkse hoeveelheid git commits te zien. Er is overduidelijk te zien dat Openstack een flink hoger aantal commits heeft dan Eucalyptus en Cloudstack, maar hiernaast is te zien dat er bij Openstack constant een stijgende lijn is in het aantal commits. Dit betekent dat Openstack steeds meer developers aantrekt en daarmee een gezonde community aan het opbouwen is.

4.3) Conclusie Community-analyse

Aan de hand van de hiervoor besproken figuren kan een vrij duidelijke conclusie over de community getrokken worden. Ten eerst is Eucalyptus over het algemeen het kleinste project, de groei van de totale community is gestagneerd net boven Cloudstack. De hoeveelheid discussie over Eucalyptus is net als de hoeveelheid Git commits maar een fractie van die van Openstack.

Ten tweede is Cloudstack het middel grote project. Er word bijna evenveel over gediscussieerd als Openstack, maar met een userbase welke de helft zo groot is als Openstack en een commit activiteit 1/6de van Openstack loopt Cloudstack toch achter.

Ten slotte Openstack, eigenlijk is uit alle grafieken op te maken dat sinds het Openstack project gestart is de community constant aan het groeiend is en er steeds meer organisaties zich aansluiten bij het Openstack project. Zo is recentelijk bijvoorbeeld Wal-Mart in zee gegaan met Openstack. De support van grotere organisaties betekent enorm veel voor de groei van het Openstack project in zijn geheel vanwege de naambekendheid en extra developer support.

5.) Functie analyse

Naast de activiteit van de project is het belangrijk dat de project geheel voldoen aan de verwachtingen van i3D hiervoor zijn de specificaties per pakket naast elkaar gezet om te bepalen. Een aantal punten welke de opdrachtgever aan heeft gegeven te belangrijk zijn, zijn o.a. Hyper-V compatibiliteit, AWS compatibiliteit en APIs waarmee de functionaliteit ingebouwd kan worden in het admin panel van i3D.net zelf.

In dit document word er kort ingegaan op de ondersteunde functionaliteit van de pakketten. De reden hiertoe is namelijk dat om de volledige functionaliteit te testen van een pakket deze tegenover de eisen van i3D.net getest moet worden. Om dit te realiseren moet er een complete testomgeving opgezet worden welke vervolgens getest word op de gestelde eisen. In dit project zou dit teveel tijd en resources kosten, waardoor gekozen is voor één project gebaseerd op de globale eisen een testomgeving op te bouwen.

Eucalyptus

-Hyper-V compatibiliteit

Dit is het grootste minpunt voor Eucalyptus. Er is geen support voor een Hyper-V hypervisors. Dit is echter een van de Must's voor i3D.net.

-AWS compatibiliteit

Eucalyptus heeft volledige support voor de AWS omgeving. Er is tussen Eucalyptus en Amazon een agreement afgesloten om ervoor te zorgen dat transfer tussen de omgevingen gemakkelijk en vloeiend verloopt.

-API functionaliteit

Er is voor zowel de AWS omgeving als de Eucalyptus cloud zelf een set tools genaamd Euca2ools waarmee controle over vrijwel de hele omgeving mogelijk is.

Cloudstack

-Hyper-V compatibiliteit

Cloudstack biedt sinds Maart 2013 support voor Hyper-V.

- AWS compatibiliteit

Cloudstack heeft volledige support voor de AWS omgeving.

-API functionaliteit

Ruime functionaliteit om te integreren in het admin panel.

Openstack

-Hyper-V compatibiliteit

Openstack biedt Hyper-V support sinds de Grizzly release. Inmiddels is er vrijwel complete support voor deze hypervisor.

-AWS compatibiliteit

Openstack biedt volle AWS ondersteuning

-API functionaliteit

Naast het ondersteunen van verscheidene methoden om de Openstack API aan te spreken is er ook support voor onder andere API's als Euca2tools, Hybridfox, Boto, fog en php opencloud.

In de onderstaande tabel word er bekeken of en in welke mate de pakketten voldoen aan de eerste globale eisen van de opdrachtgever. De informatie is verkregen door white papers en feature sheets in te zien van de verschillende pakketten. Dit geeft een beeld van waar de gekozen pakketten tot in staat zijn en of er direct reden is om een pakket niet te kiezen.

	Functionaliteit	Hyper-V	AWS compatibiliteit	API functionaliteit
Pakket				
Eucalyptus		--	++	+
Cloudstack		+	++	++
Openstack		++	++	++

Aan de hand van de bovenstaande eisen is dus te zien dat Eucalyptus vrijwel direct afvalt als voor de andere pakketten, vanwege de vaste requirement van Hyper-V support. Verder zijn over het algemeen alle pakketten redelijk uitgebreid op het gebied van functionaliteit. Het doorslaggevende voor i3D is echter of de functionaliteit aan te passen is naar wens. In alle gevallen is dat in zekere mate mogelijk, echter heeft Openstack een dedicated community die constant streeft naar het uitbreiden van de userbase en het betrekken van de community bij de development.

6.) Conclusie Functie Analyse

Aan de hand van de globaal verwachte functionaliteit is Openstack duidelijk de meest geschikte keuze. Alhoewel Cloudstack bijna dezelfde functionaliteit biedt als Openstack, was er veel onduidelijkheid over de beloofde functionaliteit en in hoeverre deze beschikbaar was. Omdat transparantie erg belangrijk is en er geen tijd is om voor onverwachte problemen tijdens het project is er daarom gekozen voor Openstack. Openstack biedt een duidelijk beeld van wat er te verwachten valt indien Openstack in gebruik genomen wordt en geeft aan wat er in ontwikkeling is met een planning voor verwachte implementatie.

7.) Conclusie Software Onderzoek

Na het bekijken van de functionaliteit en de community's van alle project kan geconcludeerd worden dat voor i3D.net het Openstack pakket het beste past. Dit pakket heeft een grote en actieve community welke ondersteund word door grote toonaangevende bedrijven. De community heeft inmiddels een vaste development cycle opgezet en heeft in principe een vaste groep developers welke aan hun eigen sub-projecten werken. De actieve community en snelle development is voordelig voor i3D.net in verband met de snelle patches voor problemen in Openstack en fixes voor kritieke issues.

De openheid van Openstack zorgt er tevens voor dat indien i3D genoeg in het project ziet zij hier hun eigen development team aan kunnen toewijden om zelf bij te dragen aan het project.

Om deze redenen is er dus besloten om met Openstack verder te werken aan het Proof of Concept.

Proof of Concept documentatie

Uitleg over Openstack en het Proof of Concept



Opdrachtgever:
Rick Slood
i3D.net
Van Nelleweg 1, 3044BC
Rotterdam



Opdrachtnemer:
Chris Zonneveld
09032835
Rotterdamseweg 137, 2628AL
Haagse Hogeschool Delft

Versiebeheer

Versie	Auteur	Datum	Wijzigingen
Proof of Concept 0.1	C. Zonneveld	17-2-2015	Eerste concept
Proof of Concept 0.2	C.Zonneveld	26-2-2015	i3D.net template ingevoegd
Proof of Concept 1.0	C.Zonneveld	29-5-2015	Laatste content ingevoerd

[Versiebeheer](#)

[1.\) Inleiding](#)

[2.\) Openstack introductie](#)

[3.\) Componenten](#)

[3.1\) Keystone](#)

[3.2\) Nova](#)

[3.3\) Glance](#)

[3.4\) Neutron](#)

[3.5\) Swift](#)

[3.6\) Cinder](#)

[3.7\) Heat](#)

[3.8\) Trove](#)

[3.9\) Ironic](#)

[3.10\) Sahara](#)

[3.11\) Zaar](#)

[3.12\) Barbican](#)

[3.13\) Designate](#)

[3.14\) Manilla](#)

[3.15\) Horizon](#)

[3.16\) Ceilometer](#)

[4.1\) Technisch Ontwerp Proof of Concept](#)

[4.2\) Beschikbare hardware](#)

[4.3\) Installatie proces](#)

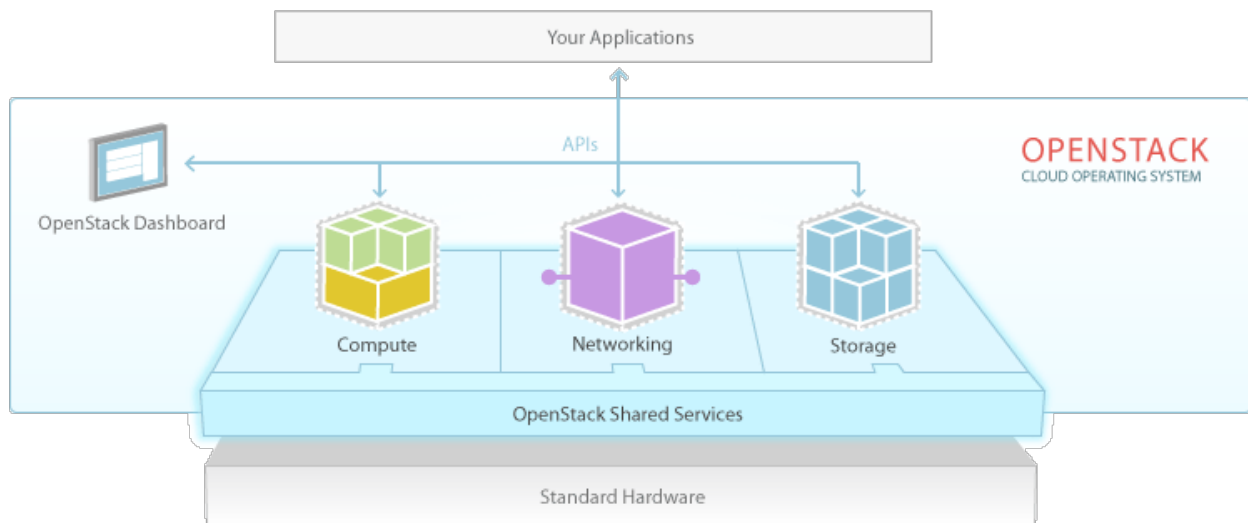
[4.3.1\) Algemene configuratie](#)

[4.3.2\) Keystone, De Identity Service](#)

1.) Inleiding

Dit document word gebruikt om uitleg te bieden over het gekozen software pakket en de werking hiervan. Er zal eerst beschreven worden hoe het gekozen pakket in elkaar steekt. Hierna zal het opzetten van het Proof of Concept beschreven worden en wat voor bijzonderheden zich hierbij hebben voorgedaan. Met dit document zal er kennis verkregen kunnen worden over de werking van Openstack en hoe hiermee een Proof of Concept omgeving opgebouwd kan worden en uitgebreid kan worden tot productie.

2.)Openstack introductie



Om een korte introductie te geven is Openstack is een geheel gratis en open-source cloud operating systeem. Hiermee kunnen pools van computing kracht, storage en networking capaciteit verspreid door datacentra beheerd worden door middel van een centraal dashboard. Via dit centrale dashboard kunnen administrators hun gehele Openstack cloud beheren en hebben de gebruikers de mogelijkheid om hun eigen capaciteit te provisionen door een web interface. Naast de web interface is er de mogelijkheid om te managen via command-line tools of de RESTful API, voornamelijk de laatste twee zijn interessant voor bedrijven zoals Openstack.

Openstack is gestart in 2010 als een coöperatief project tussen Rackspace Hosting en NASA, het doel was om een Open Source cloud computing platform op te zetten waarmee public en private clouds beheerd kunnen worden, ongeacht hoe groot deze zijn, door simpel in gebruik te zijn en schaalbaar te zijn. Na de eerste release in 2010 is de community flink gegroeid tot een community van meer dan 9000 contributors in 90 landen met support van meer dan 150 bedrijven. Een aantal van de grote bedrijven achter Openstack zijn Red Hat, Canonical, IBM, AT&T, Cisco, Intel, Paypal en Comcast. Deze community werkt volgens een 6-month time-based release cycle waarbij tijdens de planningsfase van elke cycle de community bijeenkomt om de plannen voor de komende release te bespreken.

In de 5 jaar dat Openstack inmiddels in development zijn er een flink aantal features bij gekomen. Een korte release history is als volgt:

Release Name	Release Date	Included Components / Projects
Austin	Oct 21, 2010	Nova, Swift
Bexar	Feb 3, 2011	Nova, Glance, Swift
Cactus	Apr 15, 2011	Nova, Glance, Swift
Diablo	Sep 22, 2011	Nova, Glance, Swift
Essex	Apr 5, 2012	Nova, Glance, Swift, Horizon, Keystone
Folsom	Sep 27, 2012	Nova, Glance, Swift, Horizon, Keystone, Quantum , Cinder
Grizzly	Apr 4, 2013	Nova, Glance, Swift, Horizon, Keystone, Quantum , Cinder
Havana	Oct 17, 2013	Nova, Glance, Swift, Horizon, Keystone, Neutron , Cinder
Icehouse	Apr 2014	Nova, Glance, Swift, Horizon, Keystone, Neutron , Cinder, More

Een van de interessante punten aan Openstack is dat het modulair opgezet is. De huidige Openstack release (Juno) bestaat uit 11 componenten welke individueel ontwikkelt worden waar nog 5 nieuwe componenten van in development zijn. De huidige lijst bestaat uit de volgende componenten:

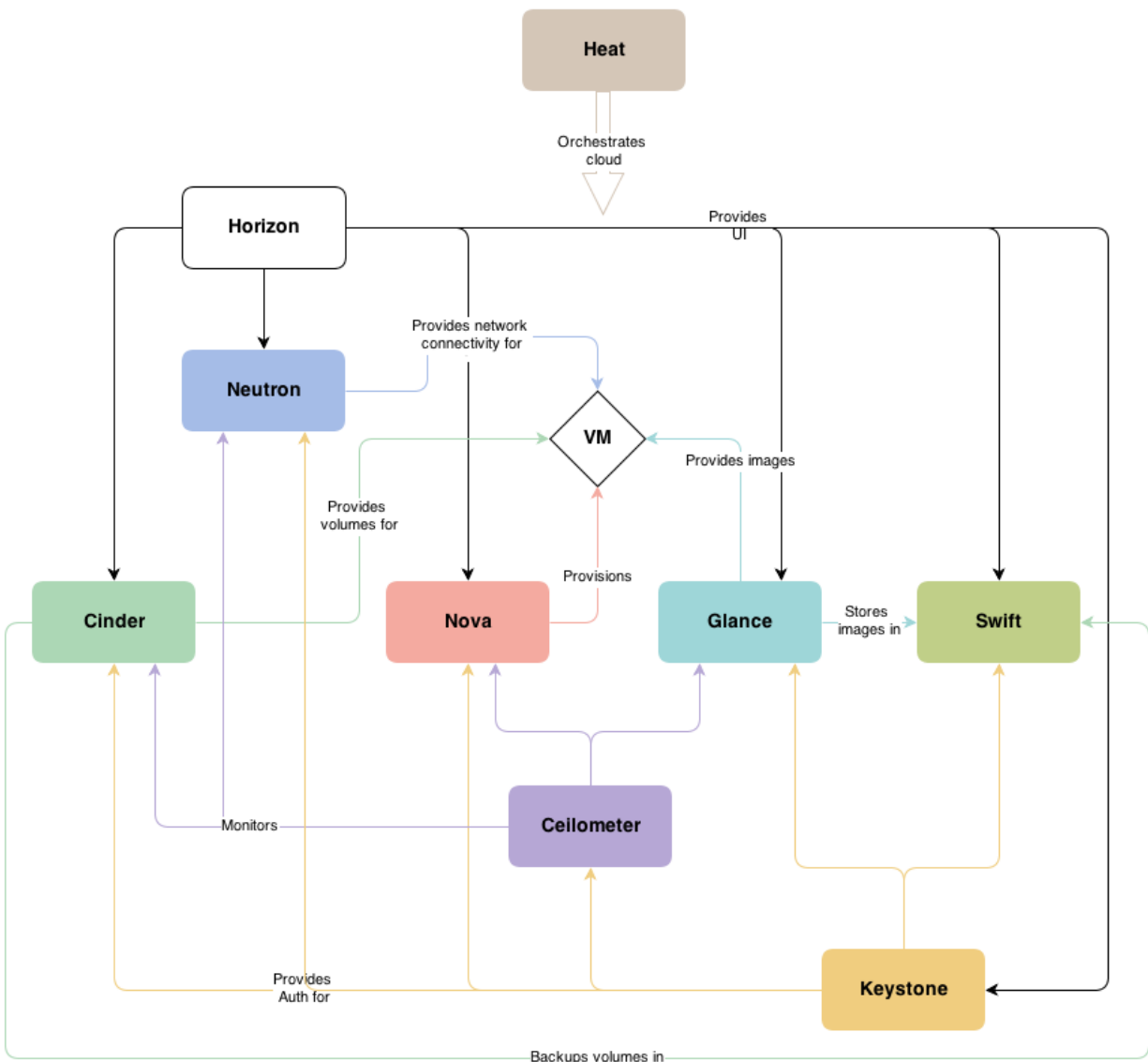
- OpenStack Compute (code-name Nova) - integrated project since Austin release
- OpenStack Networking (code-name Neutron) - integrated project since Folsom release
- OpenStack Object Storage (code-name Swift) - integrated project since Austin release
- OpenStack Block Storage (code-name Cinder) - integrated project since Folsom release
- OpenStack Identity (code-name Keystone) - integrated project since Essex release
- OpenStack Image Service (code-name Glance) - integrated project since Bexar release
- OpenStack Dashboard (code-name Horizon) - integrated project since Essex release
- OpenStack Telemetry (code-name Ceilometer) - integrated project since the Havana release
- OpenStack Orchestration (code-name Heat) - integrated project since the Havana release
- OpenStack Database (code-name Trove) - integrated project since the Icehouse release
- OpenStack Data Processing (code-name Sahara) - integrated project since the Juno release

Al deze componenten hebben een specifieke taak binnen het Openstack geheel. Alhoewel alle componenten hun eigen functionaliteit bieden aan Openstack zijn niet alle componenten verplicht te implementeren. Zo begon het Openstack als puur Nova en Swift, waarin dus alleen computing en storage door gebruikt werd. Om te begrijpen welke onderdelen bij i3D van toepassing zijn is er gekozen om alle componenten eerst individueel te bekijken en te beoordelen op nut voor i3D.

3.) Componenten

Aangezien Openstack uit een aantal verschillende componenten bestaat met elk unieke functionaliteit zal ik deze per component nader verklaren om een duidelijker beeld te geven bij het gehele Openstack plaatje. In dit hoofdstuk zal alleen de globale functionaliteit beschreven worden. De in-depth installatie en functies zullen later besproken worden.

Hieronder is een conceptuele architectuur te zien van Openstack, dit laat zien hoe alle componenten in relatie tot elkaar staan. Alle componenten vormen samen een ideale Openstack situatie echter zijn niet alle componenten nodig voor een functionele installatie. Zo kan bijvoorbeeld Ceilometer weggelaten worden indien chargeback geen vereiste is (Bijvoorbeeld in private clouds).



De componenten welke besproken worden zijn als volgt:

Hoofd componenten

- Identity (Keystone)
- Compute (Nova)
- Image service (Glance)
- Networking (Neutron)
- Object Storage (Swift)
- Block Storage (Cinder)
- Orchestration (Heat)
- Database Service (Trove)
- Bare Metal (Ironic)
- Data processing (Sahara)
- Message service (Zaqar)
- Key management (Barbican)
- DNS (Designate)
- Shared Filesystems (Manila)

Supportende componenten

- Dashboard (Horizon)
- Telemetry (Ceilometer)

3.1) Keystone

Het Keystone project verzorgt alle Identity, token, Catalog en Policy services welke door de verscheidene andere componenten binnen Openstack gebruikt worden. De Keystone service support directory services zoals LDAP en meerdere authenticatie mechanismen zoals username/password, token based en AWS-style logins.

3.2) Nova

Nova, ook wel bekend als Openstack Compute, is de software welke de het IaaS platform aanstuurt. Zelf beschikt Nova niet over virtualisatie software. Wat nova in plaats daarvan doet is het communiceert met het onderliggende virtualisatie systeem; denk aan KVM, Hyper-V en ESXi; en stelt deze functionaliteit via een Web API beschikbaar.

3.3) Glance

Glance is de imaging service van Openstack. Het bied een service waar gebruikers assets kunnen uploaden en verwerven om met andere services te gebruiken. Op het moment is er ondersteuning voor images en metadata definities. De functionaliteit hiermee omvat op op het moment het ontdekken, registreren en retrieven van images en metadata door middel van een RESTful API.

3.4) Neutron

Het Neutron project heeft als doel Network-as-a-Service aan te kunnen bieden tussen interface devices welke door andere Openstack services gemanaged worden. Het biedt support voor Layer 2 networking en IP address management, naast een Layer 3 router construct om externe connectiviteit te faciliteren. Hiernaast heeft Neutron een groeiende lijst van plugins om interoperabiliteit te supporten voor verscheidene routers, switches, virtuele switches en SDN controllers.

3.5) Swift

Swift is een highly available en gedistribueerd object/blob storage systeem. Met Swift kunnen grote hoeveelheden data efficiënt, veilig en goedkoop opgeslagen worden. Het is gebouwd voor schaalbaarheid en geoptimaliseerd voor duurzaamheid, beschikbaarheid en concurrency over de gehele data set.

3.6) Cinder

Cinder is de block storage voor Openstack. Cinder virtualiseert pools van block storage devices en presenteert eindgebruikers met een API om deze resources te gebruiken zonder kennis nodig te hebben van waar of hoe deze data opgeslagen is.

3.7) Heat

Heat is het hoofdproject in het Openstack orchestration programma. Het gebruikt een orchestration systeem om de infrastructuur voor een cloud applicatie op te zetten gebaseerd op templates. Deze templates beschrijven de infrastructuur in tekst welke te lezen en schrijven zijn voor mensen. Heat kan projecten geautomatiseerd schalen door samen te werken met Ceilometer.

3.8) Trove

Trove is het Database as a Service component van Openstack. Het doel van Trove is om gebruikers snel en gemakkelijk de functionaliteit van relationele en non-relationale databases te bieden zonder het complexe administratieve werk. Gebruikers kunnen gemakkelijk meerdere database instanties opzetten en beheren.

3.9) Ironic

Ironic is een Openstack project welke bare metal machines (in tegenstelling tot de gebruikelijke Virtual Machines) provisioned door gebruik te maken van gebruikelijke technieken als PXE boot en IPMI om een groot schaal aan hardware te supporten. Om deze functionaliteit uit te breiden is er support voor pluggable drivers om vendor-specifieke functionaliteit toe te voegen. Op deze manier zijn fysieke servers even makkelijk te provisionen als cloud VMs.

3.10) Sahara

Sahara is ontworpen om gebruikers gemakkelijk een Hadoop cluster op te laten zetten welke gemakkelijk te starten, stoppen en schalen is binnen Openstack.

3.11) Zaqr

Zaqr is een multi-tenant cloud messaging service voor web developers. Het beschikt over een RESTful API waarmee developers tussen verschillende componenten van hun SaaS en mobiele applicaties berichten kunnen sturen. Het systeem hierachter is gebouwd op basis van schaalbaarheid en beveiliging.

3.12) Barbican

Barbican is een REST API ontworpen voor het veilig opslaan, provisionen en managen van secrets. Barbican kan onder andere omgaan met secrets als: Symmetric Keys, Asymmetric Keys en Raw Secrets.

3.13) Designate

Designate is de DNSaaS service voor Openstack. Het biedt een REST API voor domain en record management, het is multi-tenant, geïntegreerd met Keystone voor authenticatie en heeft out of the box support voor PowerDNS en Bind9.

3.14) Manilla

Manilla is ontworpen om een shared file system te ontwerpen voor Openstack. Het coordineert de toegang tot shared of distributed file systems welke door Openstack Compute instances gerequest kunnen worden.

3.15) Horizon

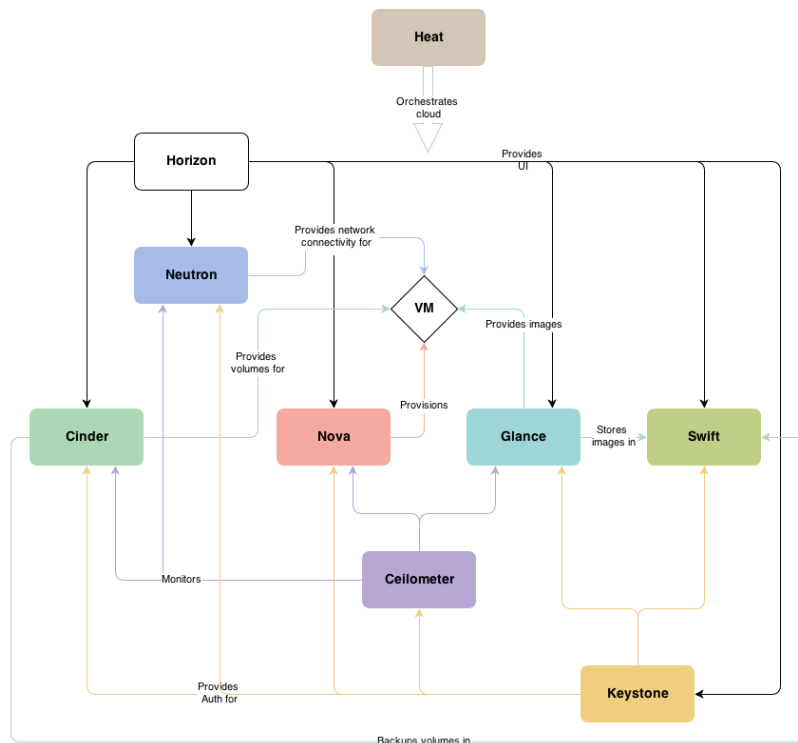
Horizon is de implementatie van het Openstack Dashboard. Dit dashboard geeft gebruikers een web-based interface tot de services als Nova, Swift, Keystone e.d.

3.16) Ceilometer

Ceilometer verzamelt statistieken over het gebruik van fysieke en virtuele resources van uitgerolde cloud omgevingen. Deze data word opgeslagen om later opgehaald en geanalyseerd te worden, later kan deze data gebruikt worden om acties te triggeren als bepaalde criteria behaald worden

4.1) Technisch Ontwerp Proof of Concept

Zoals eerder besproken in de sectie componenten bestaat Openstack uit een groot aantal verschillende componenten. Een aantal van deze componenten zijn core en zeker nodig voor een basis cloud, andere zijn optioneel en kunnen in principe weggelaten worden zonder functionele impact.

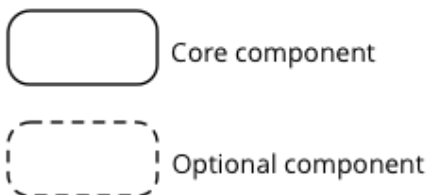
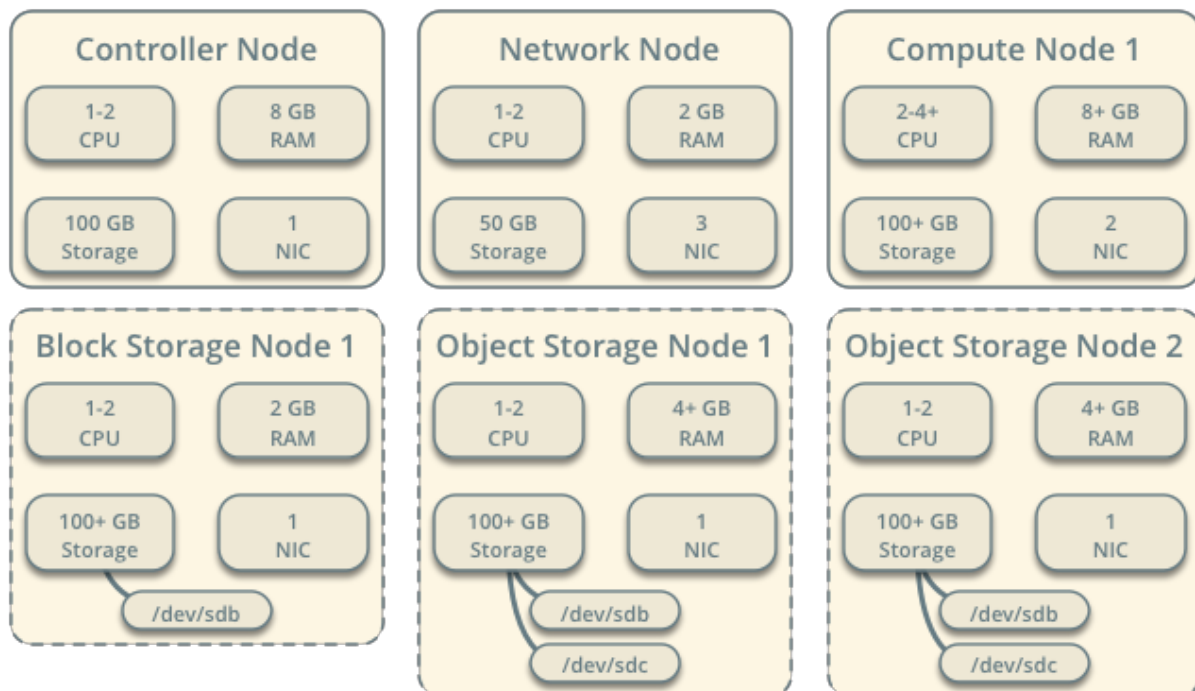


Omdat Openstack dusdanig uitgebreid is is er kozen om het Proof of Concept stap voor stap op te bouwen. Dit betekent dat de core componenten eerst één voor één toegevoegd zullen worden aan het PoC zodat er bekendheid geworven kan worden met de Openstack omgeving in de praktijk. Mocht het opzetten van een core gebaseerd Proof of Concept volgens plan verlopen kan er gekeken worden welke optionele componenten er meegenomen kunnen worden in het Proof of Concept.

In de documentatie van Openstack is een voorbeeld architectuur beschreven welke een minimale architectuur voor Openstack beschrijft. Voor het Proof of Concept is deze architectuur een goed startpunt.

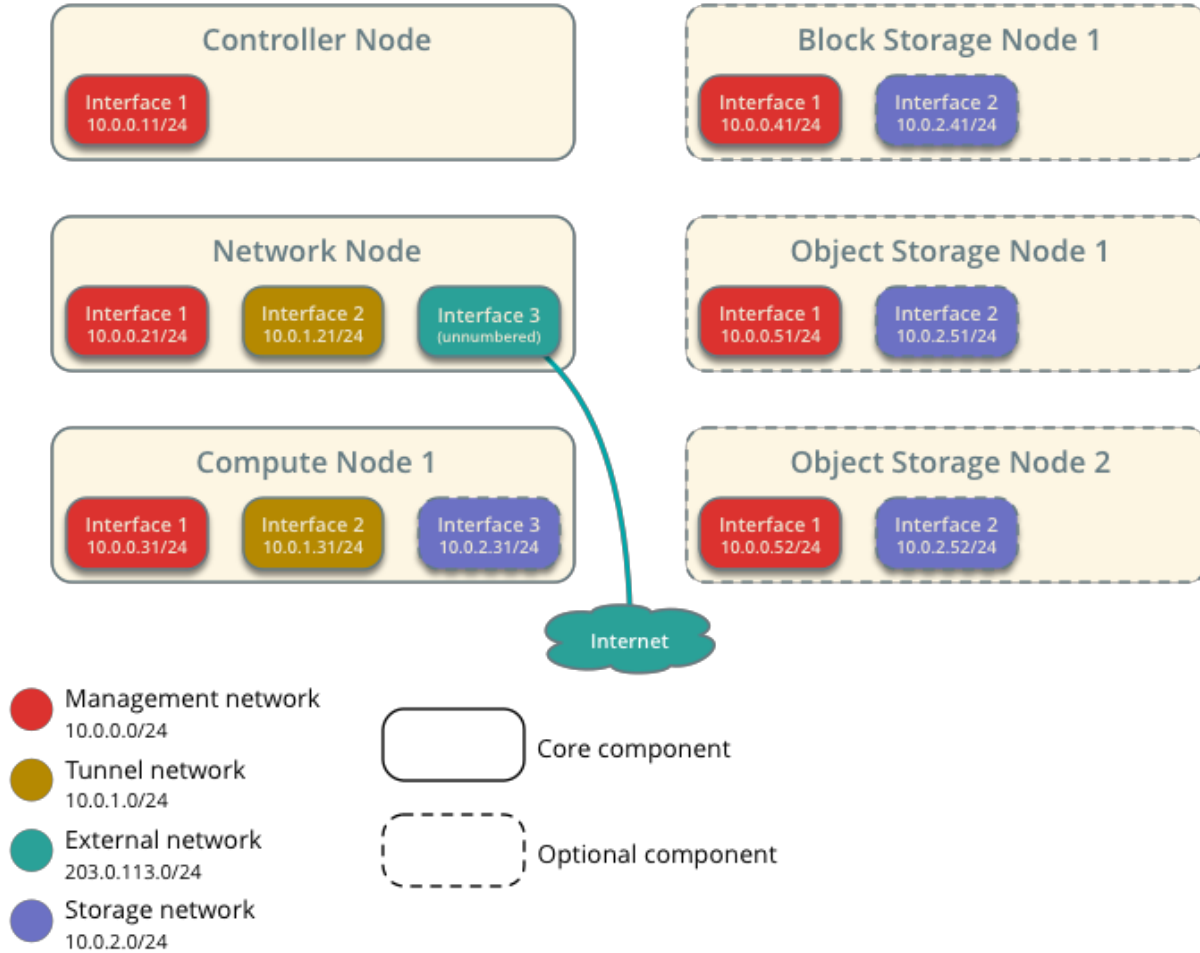
De minimale hardware eisen worden beschreven als volgt:

Minimal Architecture Example - Hardware Requirements OpenStack Networking (neutron)



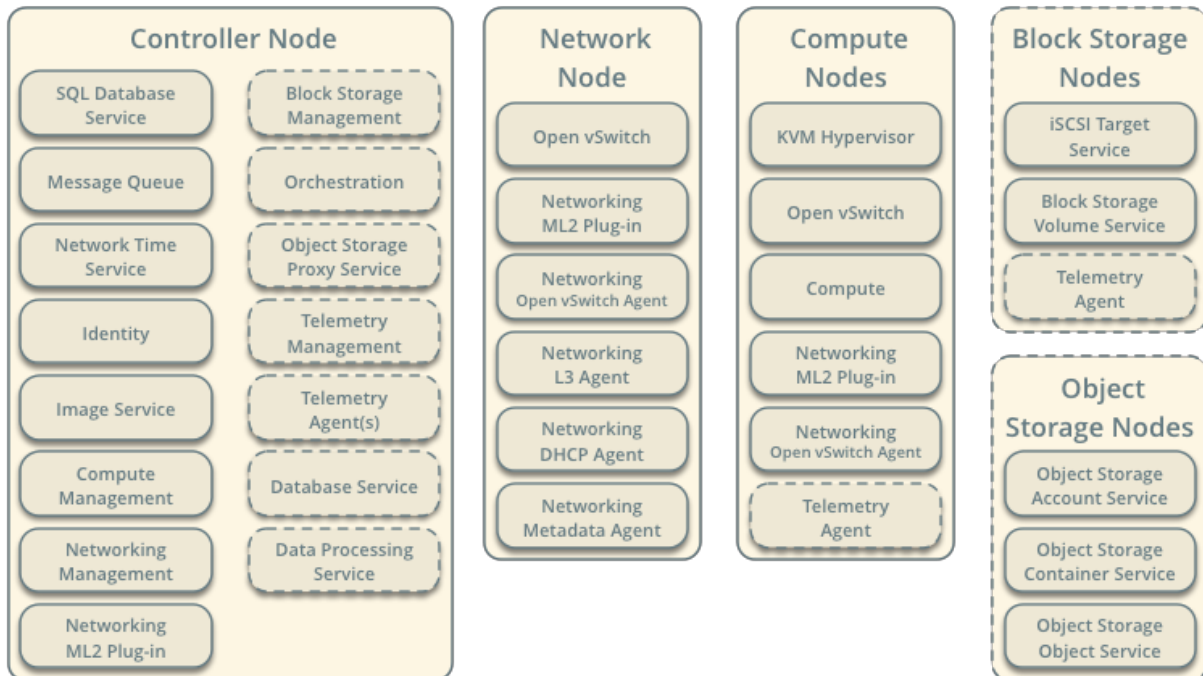
Met de door i3D beschikbaar gestelde servers is er genoeg hardware om deze basis architectuur te realiseren.

Minimal Architecture Example - Network Layout OpenStack Networking (neutron)



Hierboven word de minimale netwerk layout beschreven.

Minimal Architecture Example - Service Layout OpenStack Networking (neutron)



 Core component

 Optional component

Een minimale service configuratie word als volgt beschreven. Er zijn veel optionele componenten welke in de eerste instantie niet opgenomen zullen worden in het Proof of Concept gezien deze niet direct vereist worden als functionaliteit.

4.2) Beschikbare hardware

Om te beginnen met het Proof of Concept heeft i3d.net een zestal servers ter beschikking gesteld. De server specificaties zijn als volgt:

1x	HP ProLiant DL120 G6	Server [1073]
Processor:	Intel(R) Xeon(R) CPU X3430 @ 2.40GHz	
Memory:	4GB	
Storage:	2x 1TB 1x 250GB	
2x	HP ProLiant DL120 G6	Servers [1074, 1075]
Processor:	Intel(R) Xeon(R) CPU X3430 @ 2.40GHz	
Memory:	4GB	
Storage:	1x 250GB	
2x	HP ProLiant DL160 G6	Servers [1842, 1844]
Processor:	2x Intel(R) Xeon(R) CPU L5630 @ 2.13GHz	
Memory:	16GB	
Storage:	1TB	
1x	HP ProLiant DL160 G6	Server [1843]
Processor:	2x Intel(R) Xeon(R) CPU E5620 @ 2.40GHz	
Memory:	16GB	
Storage:	1TB	

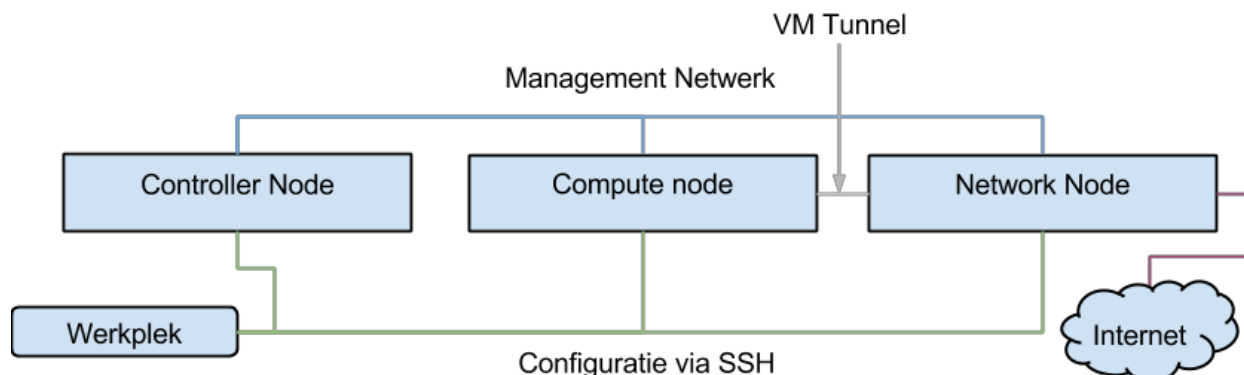
Deze servers zijn allemaal voorbereid met een installatie van Ubuntu 14.04 en aangesloten op het i3D netwerk met 1Gbit links.

Dit is de manier waarop de servers aangeleverd zijn door i3D.net, mochten de specificaties van de servers niet voldoen kunnen deze gemakkelijk aangepast worden door deze in overleg met de opdrachtgever aan te passen.

Om het Proof of Concept te realiseren zijn de servers als volgt ingedeeld:

Controller Node:	Server 1842
Network Node:	Server 1074
Compute Node:	Server 1843
Storage Node:	Server 1073

De servers hierboven zijn netwerktechnisch op de volgende manier aangesloten:



Alle servers zijn volgens de recommended networking configuratie van de Openstack documentatie met elkaar verbonden.

Het management netwerk zorgt ervoor dat de controller node voor de andere nodes het management in beheer kan nemen. Hieronder vallen services zoals Keystone, Databases voor ander andere diensten en messaging services. Op alle nodes is deze geconfigureerd op de primaire NIC welke via een private VLAN onderling connectiviteit biedt.

De VM Tunnel is een tunnel netwerk welke connectiviteit voor de VMs op de compute node naar de netwerk node verzorgt. Deze tunnel is nodig omdat de compute node geen directe verbinding naar het externe netwerk biedt en van zichzelf geen virtualisatie van netwerken en routing biedt, dit verloopt via de Network node.

De connectie naar het internet loopt via de Network node. Deze node krijgt de data van de virtuele machines binnen via de VM tunnel en zorgt ervoor dat deze correct gerouteerd wordt indien deze een verbinding naar het internet of naar andere subnetten moet maken. De network node zorgt dus voor dat alle data correct gerouteerd wordt.

Verder is de configuratie van alle nodes via SSH op het interne netwerk uitgevoerd om gemakkelijk vanaf de werkplek de werkzaamheden uit te voeren.

Het storage netwerk zit nog niet verwerkt in het schema aangezien er op het moment gebruik gemaakt wordt van local storage. Indien de Storage node toegevoegd wordt zal er een extra Storage netwerk tussen de Compute node en de Storage node lopen en zal er vanaf de Controller node een verbinding naar de Storage node lopen om deze aan het management netwerk te koppelen.

4.3) Installatie proces

In de volgende hoofdstukken word het installatieproces per node beschreven. Ten eerste zullen de basis componenten Keystone, Glance en Nova opgezet worden. Hierna word gekeken welke componenten benodigd zijn om het Proof of Concept af te ronden voor de opdrachtgever. Er zal begonnen worden met het opzetten van wat algemene configuratie op de servers.

4.3.1) Algemene configuratie

De volgende configuratie onderdelen zijn vereist op elke sever welke Openstack componenten draait en zullen dus ook op alle servers uitgevoerd worden tenzij anders aangegeven. Per installatie onderdeel worden afwijkende of benoemenswaardige parameters aangegeven en besproken. Mocht er niets aangegeven worden kan er van default configuratie uitgegaan worden,.

NTP Server:

Ten eerste word de NTP geconfigureerd, hiervoor word de NTP server van i3D gebruikt. Op elke servers wordt NTP geïnstalleerd en de configuratie aangepast.

Commando:

```
apt-get install ntp
```

Configuratie:

```
# /etc/ntp.conf, configuration for ntpd; see ntp.conf(5) for help
```

```
driftfile /var/lib/ntp/ntp.drift
```

```
statistics loopstats peerstats clockstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable
```

```
server time.i3d.net iburst
```

```
# Use Ubuntu's ntp server as a fallback.
server ntp.ubuntu.com
```

```
restrict -4 default kod notrap nomodify
restrict -6 default kod notrap nomodify
```

Openstack repository:

Om de Openstack repository aan Ubuntu toe te voegen dient het volgende uitgevoerd te worden:

```
# apt-get install ubuntu-cloud-keyring
# echo "deb http://ubuntu-cloud.archive.canonical.com/ubuntu" \ "trusty-updates/juno main" >
/etc/apt/sources.list.d/cloudarchive-juno.list
# apt-get update && apt-get dist-upgrade
```

Database:

Bijna alle Openstack services gebruiken een SQL database om informatie op te slaan. In dit geval draait de database op de Controller node.

```
# apt-get install mariadb-server python-mysqldb
```

Tijdens de installatie word een wachtwoord gevraagd voor de root mariaDB user, hiervoor worden hex passwords van 10 tekens lang gegenereerd. Na de installatie van MariaDB word er in `/etc/mysql/my.cnf` het volgende aangepast:

```
bind-address = [het management network IP van de controller node]

default-storage-engine = innodb
innodb_file_per_table
collation-server = utf8_general_ci
init-connect = 'SET NAMES utf8'
character-set-server = utf8
```

Hiermee word het management IP toegankelijk voor de andere nodes in het management netwerk en worden er een aantal handige opties op aanraden van Openstack geactiveerd.

Vervolgende word de database gerestart en secure gemaakt

```
# service mysql restart
# mysql_secure_installation
```

Messaging server:

Openstack gebruikt een message broker om onder de services de operaties en informatie te coördineren, deze broker draait meestal op de controller node. Omdat de meeste Openstack services RabbitMQ ondersteunen zal deze gebruikt worden.

```
# apt-get install rabbitmq-server
```

4.3.2) Keystone, De Identity Service

In dit hoofdstuk word beschreven hoe de Openstack Identity service geïnstalleerd en geconfigureerd word. Dit is om een idee te geven van op wat voor manier de configuratie doorlopen word. Bij de volgende services zullen alleen de bijzonderheden genoteerd worden.

Vorbereiding:

Er word eerst een database aangemaakt, dit word als volgt uitgevoerd.

Verbinden naar de database:

```
mysql -u root -p
```

Keystone database aanmaken:

```
CREATE DATABASE keystone;
```

Juiste rechten toekennen:

```
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'localhost' \
IDENTIFIED BY 'KEYSTONE_DBPASS';
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'%' \
IDENTIFIED BY 'KEYSTONE_DBPASS';
```

Hierbij moet voor KEYSTONE_DBPASS een wachtwoord gegenereerd worden, bijvoorbeeld door de functie *openssl rand -hex 10* te gebruiken om een hexadecimaal wachtwoord te genereren.

Keystone installatie:

Nu de database voorbereid is kan Keystone geïnstalleerd worden, dit word als volgt gedaan:

```
apt-get install keystone python-keystoneclient
```

Als keystone geïnstalleerd is kan de configuratie uitgevoerd worden, de config file bevind zich in */etc/keystone/keystone.conf*. De waardes welke hierbij van belang zijn zijn de volgende:

```
admin_token = ADMIN_TOKEN
connection = mysql://keystone:KEYSTONE_DBPASS@controller/keystone
```

De ADMIN_TOKEN is een wachtwoord welke met behulp van het *openssl rand* commando gegenereerd kan worden. In de connection settings moet er op gelet worden dat het juiste adres van de controller mee word gegeven en dat hetzelfde wachtwoord gebruikt word voor KEYSTONE_DBPASS welke gegenereerd is tijdens het opzetten van de database.

Als de configuratie voltooid is kan de database gepopuleerd en herstart worden met het commando:

```
su -s /bin/sh -c "keystone-manage db_sync" keystone
service keystone restart
```

Rollen aanmaken:

Nu Keystone geconfigureerd is kunnen er tenants users en rollen aangemaakt worden. het aanmaken hiervan is simpel en vlot te voltooien.

Administrator tenant aanmaken:

```
keystone tenant-create --name admin --description "Admin Tenant"
```

Administrator user aanmaken: (vervang ADMIN_PASS door een openssl rand waarde)

```
keystone user-create --name admin --pass ADMIN_PASS
```

Administrator role aanmaken:

```
keystone role-create --name admin
```

Ten slotte word de admin rol toegevoegd aan de administrator user en tenant. Op dezelfde manier kunnen er extra gebruikers, rollen en tenants aangemaakt worden voor bijvoorbeeld demo gebruikers. Naast het bijhouden van de gebruikers rollen en tenants binnen Openstack houdt Keystone ook bij welke services er draaien binnen Openstack en waar deze te bereiken zijn. Hiervoor zal eerst Keystone zichzelf toevoegen aan de catalogus. Dit gebeurt voor iedere service op dezelfde manier:

De service word aangemaakt:

```
keystone service-create --name keystone --type identity --description "OpenStack Identity"
```

De locatie word beschreven:

```
keystone endpoint-create --service-id $(keystone service-list | awk '/ identity / {print $2}') \  
--publicurl http://controller:5000/v2.0 --internalurl http://controller:5000/v2.0 \  
--adminurl http://controller:35357/v2.0 --region regionOne
```

Testen van de service

Ten slotte kan de service getest worden door de volgende handelingen uit te voeren.

Het aanvragen van een authenticatie token:

```
keystone --os-tenant-name admin --os-username admin --os-password ADMIN_PASS \  
--os-auth-url http://controller:35357/v2.0 token-get
```

Het Opvragen van een lijst tenants:

```
keystone --os-tenant-name admin --os-username admin --os-password ADMIN_PASS \  
--os-auth-url http://controller:35357/v2.0 tenant-list
```

Als deze functies beide werken is keystone correct geconfigureerd en kan er verder gewerkt worden aan het opzetten van de volgende services. Aangezien het opzetten van de opvolgende services vrijwel hetzelfde is aan Keystone zullen alleen de complicates in het volgende hoofdstuk besproken worden

4.4) Complicaties

Tijdens het opzetten van het Proof of Concept is niet alles geheel zonder problemen verlopen. Een aantal van de interessante of noemenswaardige punten zullen besproken worden. Er zal gekeken worden naar wat het probleem veroorzaakt en wat er gedaan is om het probleem te mitigeren of op te lossen.

4.4.1) Compute nodes communiceren niet

Het eerste probleem waar tegenaan gelopen werd was een probleem met de communicatie tussen de compute node en de control node. Tijdens het opzetten van de compute node leek alles netjes te zijn verlopen en waren beide nodes onderling bereikbaar via het netwerk. Echter op het moment dat de control node functies aanriep van de compute node werd deze niet herkend. Gezien de netwerk connectiviteit welk functioneerde (ping connectiviteit is correct functionerend geconstateerd) is er gekeken naar de messaging service welke de connectiviteit tussen de services regeld. Er bleek uiteindelijk uit de logs van de controller node dat er geen reactie kwam op de berichten welke naar de compute node gestuurd werden. Om verder te zoeken naar de oorzaak is er gekeken in de logs van de compute node, hieruit bleek dat deze wel commando's ontvang, maar geen reactie terug kon sturen naar de controller. Dit bleek een foutief adres te zijn in de configuratie van de controller node. Dit probleem is verholpen door de configuratie aan te passen op de compute node en deze te herstarten. Na deze controle verliep de communicatie weer zoals verwacht.

4.4.2) Network Interfaces

Bij het volgen van de voorgestelde Openstack Deployment word er van uit gegaan dat elke over drie netwerk interfaces beschikt. De servers welke voor dit project gebruikt worden beschikken over 2 netwerk interfaces waardoor een exacte replica niet mogelijk is.

In de eerste instantie is er gekozen om de 2e netwerk interface te laten werken via 2 VLANs en de primaire (welke via SSH benaderd word) te gebruiken voor externe communicatie. Tijdens het maken van dit ontwerp werd er tegen het probleem aangelopen dat de externe interface voor de Network node lastig te configureren is via SSH (verbinding word verbroken als eth0 aangepast word). Hiervoor is besloten om toegang tot de server via KVM aan te vragen en vanaf KVM de interfaces te configureren zodat netwerk configuraties geen problemen veroorzaken voor de SSH connectiviteit. Uiteindelijk is zelf het werken via KVM onnodig geworden. Vanwege het gemak van een dedicated extern netwerk is er uiteindelijk toch gekozen om een extra NIC in de Network node te plaatsen zodat deze zonder gedeelde verbindingen naar buiten kan verbinden. Hierdoor heeft het configureren van deze NIC geen invloed op de SSH sessie welke via de andere NIC verloopt.

Naast de twee hierboven genoemde complicaties zijn er verder geen problemen ondervonden bij het opzetten van het Proof of Concept.

Adviesrapport

“Welke IaaS oplossing is geschikt voor i3D.net en hoe kan deze geïmplementeerd worden”

Versiebeheer

Versie	Auteur	Datum	Wijzigingen
Adviesrapport 0.1	C. Zonneveld	9 - 4 - 2015	Eerste concept geschreven
Adviesrapport 0.5	C. Zonneveld	27 - 4 - 2015	Inhoud afgemaakt
Adviesrapport 1.0	C. Zonneveld	1 - 6 - 2015	Laatste controle

Samenvatting

In dit rapport wordt er advies gegeven op de vraag “Wat voor baat heeft i3D.net bij een IaaS oplossing en hoe beïnvloedt dit de diensten van i3D.net”. Door middel van onderzoek zijn er een aantal aspecten van IaaS oplossingen bekeken en is er naast de individuele functionaliteit van het pakket ook gekeken naar hoe deze in de organisatie toegepast kan worden. Bij het introduceren van een IaaS pakket binnen i3D.net zijn vanaf de opdrachtgever ook nog een aantal vragen naar voren gekomen welke tevens behandeld worden in het rapport.

De reden dat i3D.net deze vraag stelt is vanwege de huidige ontwikkelingen op het gebied van cloud computing. Hedendaags is dit een groot sales punt binnen hosting bedrijven en i3D.net liep hier nog in achter, zij focusten nog voornamelijk op dedicated server hosting en co-locatie. Recent is i3D.net in zee gegaan met Microsoft om een high-performance Azure cloud op te zetten waarmee ze toch in de cloud industrie kunnen meegaan. Echter wil i3D.net een budget cloud kunnen aanbieden naast de Azure cloud om cloud-hosting voor de kleinere partijen mogelijk te maken.

In het rapport is bevonden dat Openstack op het moment de beste keuze is om een budget alternatief voor Azure te bieden. De reden hiertoe was onder andere een grote community achter het project, welke nog steeds groeit, en een brede toepasbaarheid van de verscheidene projecten binnen Openstack.

Door het implementeren van Openstack zal er voor budgetklanten een extra optie openen om hun diensten in de cloud te hosten en vanwege het feit dat Openstack open-source is kan i3D.net makkelijker hun diensten customiseren naar de wensen van hun klanten.

Deze omgeving is te realiseren door uitgaand van het Proof of Concept en de bijbehorende documentatie een productie omgeving op te zetten. Deze omgeving zal hierna onderhouden moeten worden door een klein team engineers welke zich focust op Openstack.

- [1.\) Inleiding](#)
- [2.\) Het Onderzoeksproces](#)
- [3.\) De Keuze voor Openstack](#)
 - [3.1\)Community analyse](#)
 - [3.2\)Functieanalyse](#)
 - [3.3\) Advies: Welk pakket is geschikt voor i3D.net?](#)
- [4.\)Advies voor i3D.net](#)

1.) Inleiding

In het volgende document wordt besproken wat het advies van de stagiair is op het gebied van IaaS oplossingen voor i3D.net. Eerst zal er besproken worden hoe de stagiair tot de conclusies in dit document is gekomen, hier zal een korte samenvatting van het proces voor worden beschreven. Hierna volgt een uitleg waarom Openstack gekozen is als hoofdoplossing, wat vervolgd wordt met antwoorden op de deelvragen welke vanuit de opdrachtgever naar voren zijn gekomen. Uiteindelijk word dit document afgesloten met een samenvatting van het geheel en een eindadvies.

2.) Het Onderzoeksproces

De onderwerpen en adviezen welke in dit document besproken worden zijn bereikt volgens de volgens de methoden en technieken welke in het Plan van Aanpak opgesteld zijn, dit proces is globaal te beschrijven als volgt.

Het project is onderverdeeld in een viertal fasen, namelijk de initiatiefase, de onderzoeksfase, de uitvoeringsfase en de afronding.

Tijdens de initiatiefase heeft de stagiair zich voornamelijk bezig gehouden met het opzetten van de werkplaats en het regelen van de benodigde hard- en software om het project te starten. Na het voltooien hiervan is er een Bedrijfsoriëntatie en een uitgebreid Plan van Aanpak geschreven. Het doel van de oriëntatie is om de bedrijfsinrichting voor de stagiair duidelijk te maken zodat deze duidelijk zijn plaats in het bedrijf weet. Verder is het Plan van Aanpak geschreven om richtlijnen voor de verloop van het project op te stellen en verwarring tussen de stagiair, bedrijfsbegeleider en stagebegeleider te voorkomen door duidelijk op papier te hebben wat er uitgevoerd wordt.

Eenmaal in de onderzoeksfase heeft de stagiair zich verdiept in de verschillende IaaS oplossingen, uiteindelijk zijn er hier drie van uitgezocht welke nader bekeken werden. De drie oplossingen zijn gekozen op basis van een combinatie van functionaliteit, marktbekendheid, community support en zichtbare community activiteit. De uitgebreide motivatie voor het kiezen van Openstack is te vinden in het softwarekeuze document.

Na het kiezen van één van de oplossingen is er op basis van deze keuze een Proof of Concept opstelling gemaakt om ervaring op te doen met de omgeving en demo's te kunnen geven. Met deze testomgeving kunnen eventuele vragen en vraagstukken over specifieke situaties beantwoord worden.

Na het opdoen van ervaring door het opzetten en configureren van het Proof of Concept is er met de opdrachtgever gesproken om uit te zoeken wat voor specifieke vragen er nog zijn vanuit het bedrijf. Deze vragen zijn beantwoord door middel van de vergaarde informatie en ervaring met Openstack. Door nader onderzoek te doen naar de werking van individuele componenten binnen Openstack zijn deze vragen beantwoord in het adviesrapport.

3.) De Keuze voor Openstack

Ten eerste is er tijdens het project keuze gemaakt een IaaS pakket, in dit geval was de keuze op Openstack gevallen. Om tot deze conclusie te komen is er onderzoek gedaan naar verscheidene bekende pakketten welke IaaS kunnen verzorgen. Deze zijn ge-shortlist tot een drietal mogelijkheden, namelijk Eucalyptus, Apache Cloudstack en Openstack.



Om een duidelijk beeld te krijgen van de individuele oplossingen is er per pakket onderzoek gedaan naar een aantal punten. Ten eerste hoe actief de community is, de community is het levensbloed van Open-Source projecten. In het geval dat een project geen steun krijgt vanuit de community of niet gestructureerd te werk gaat is er een kleine kans dat het een productie waardig resultaat oplevert tenzij er binnen het bedrijf een team dedicated developers aan kan werken om een eigen fork van het project op te zetten.

Hiernaast is er gekeken naar de daadwerkelijke functionaliteit die wordt geboden en hoe deze aansluit bij i3D.net. De drie projecten bieden beide globaal vergelijkbare diensten, namelijk het opzetten van een public en/of private cloud. Echter is het voor i3D.net niet alleen de directe functionaliteit van belang maar ook de community achter het produkt. Een produkt welke niet verder ontwikkeld word op het moment dat het in gebruik genomen wordt zal snel verouderd raken en zelfs onveilig zijn om in gebruik te houden. Ten eerste wordt daarom besproken welk product het beste voldoet aan de community eisen, waarna er gekeken wordt naar de functionele eisen van elk produkt. Ten slotte wordt er een keuze gemaakt voor 1 van de produkten.

3.1)Community analyse

Bij het verzamelen van informatie is het onderzoek “Open Source IaaS Community Analysis - Openstack vs OpenNebula vs Eucalyptus vs Cloudstack” van Qingye Jiang ^[13] geraadpleegd. Dit is een objectief onafhankelijk onderzoek naar de communities achter de respectievelijke open source producten. Dit onderzoek geeft een duidelijk inzicht op de huidige staat en groei van de communities aangezien dit onderzoek sinds 2011 per kwartaal uitgevoerd wordt. Er is duidelijk te zien op wat voor manier de gekozen pakketten zich ontwikkeld hebben over de afgelopen vierjaar. Deze informatie is waardevol om te voorspellen of het project de komende tijd levendig blijft of dat het project simpelweg niet van de grond af komt.

De informatie uit het onderzoek van Qingye Jiang is bovendien goed te combineren met een evaluatie van een succesvolle life cycle van Open-source projecten. Verschillende onderzoeken beschrijven de life cycle van Open-source project globaal als volgt:

● Phase 1: Concept

Er bestaat een idee waar een aantal mensen basisfunctionaliteit voor geschreven hebben en ideeën over beschrijven op een project website (Google Code, Codeplex etc.). Minimale documentatie is beschikbaar, vooral concepten.

● Phase 2: Bootstrap

Een paar mensen zullen serieus aan het product werken en hiermee experimenteren. Een kleine hoeveelheid documentatie kan beschikbaar zijn. De functionaliteit verschilt per omgeving.

● Phase 3: Early Development

Er worden door meerdere mensen kleine contributies gedaan naast de hoofd-contributors. Er wordt meer feedback verkregen door de groeiende groep geïnteresseerde en het project functioneert in meer omgevingen.

● Phase 4: Early Adoption

Vanwege een groeiend aantal deelnemers groeien ook de contributies. Het project beschikt inmiddels over een groot deel van de basisfunctionaliteit en is bruikbaar. Er wordt meer over gesproken op blogs en de kennisbasis wordt uitgebreid. Er wordt meer gewerkt aan de toegankelijkheid door het documenteren van API's, FAQ's en Getting Started guides.

● Phase 5: Development

Meer deelnemers zijn serieus bezig met het project waardoor de core features sneller ontwikkeld worden. Er worden automatische builds opgezet waarnaast regelmatig binaries beschikbaar zijn, wat gepaard gaat met groeiende bekendheid. De wiki is op dit punt uitgebreid en bruikbaar en begint verschillende scenario's te beschrijven.

● **Phase 6: Adoption**

Meer mensen dragen bij in de vorm van patches en andere kleine toevoegingen, de user base groeit naar een paar honderd mensen en er wordt een regelmatig release schedule opgezet. Er worden gedetailleerde artikelen geschreven over de werking van functies en de wiki ziet regelmatige contributie.

● **Phase 7: Maturity**

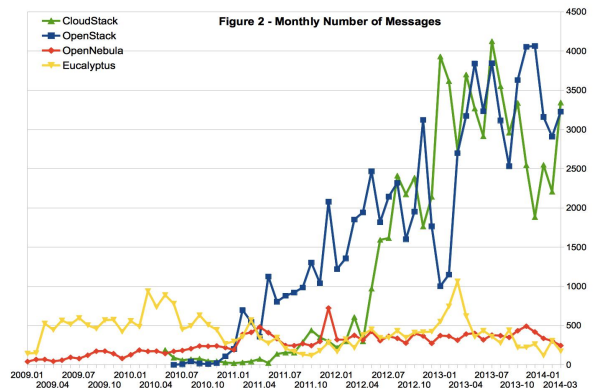
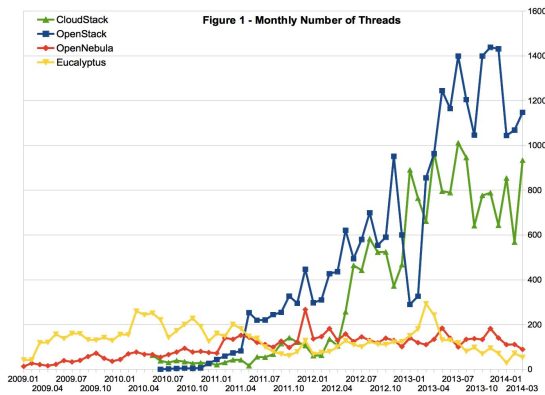
Vanwege de hoeveelheid patches en contributors wordt er een systeem opgezet om bijdragen en patches te evalueren en goed te keuren. Er worden releases opgezet voor verschillende omgevingen en contributie projecten worden samengevoegd.

● **Phase 8: Mainstream**

Tegen deze tijd zijn de originele contributors verder gegaan naar andere projecten of minder betrokken. Het project zelf groeit enorm in verschillende richtingen en consultancy bedrijven pakken het product op om er commerciële support op te bieden. Het project staat stevig en zorgt voor duidelijke toegevoegde waarde. De documentatie rondom het project groeit consistent met artikelen, handleidingen, blogs en evaluaties.

Door de pakketten te vergelijken met de fasen welke hierboven beschreven worden is een idee te krijgen van op welk punt in zijn levenscyclus het pakket is en wat te verwachten is van de toekomst. Met deze informatie is er gekeken naar de drie te evalueren pakketten. De belangrijkste punten uit het onderzoek van Qingye Jiang worden hier besproken.

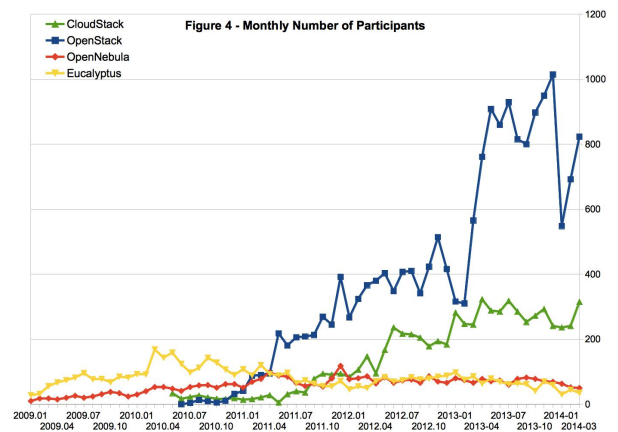
Threads en Messages



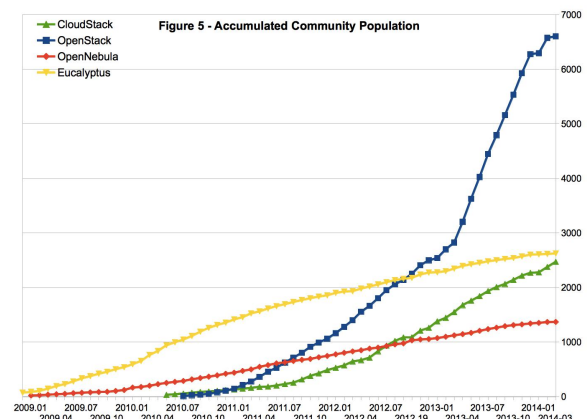
De bovenstaande grafieken (figuur 1 en 2) geven weer hoe actief de communities zijn wat betreft het posten van threads en messages. Er is hier duidelijk te zien dat de communities voor Cloudstack en Openstack sterk gegroeid zijn de afgelopen vier jaar. Alhoewel de Eucalyptus al langer bestaat dan Cloud- en Openstack is de activiteit van de community vrij constant gebleven tegenover de sterk toenemende activiteit van de concurrenten.

Participants en population

Deze gegevens worden interessanter als er gekeken wordt naar het aantal daadwerkelijke deelnemers aan de community welke in figuur 4 weergegeven wordt. In deze grafiek is te zien dat Openstack een groter aantal participanten heeft ten opzichte van Cloudstack, ongeveer 200% meer, en Eucalyptus, ruim 800%. In 2013 heeft met name Openstack een grote groei gekend Cloudstack heeft daarentegen al een aantal jaren een consistente kleine groei. Ten slotte heeft Eucalyptus gedurende de afgelopen 4 jaar een consistent ratio participanten gehad, welke bij de opkomst van Openstack lijkt af te nemen.

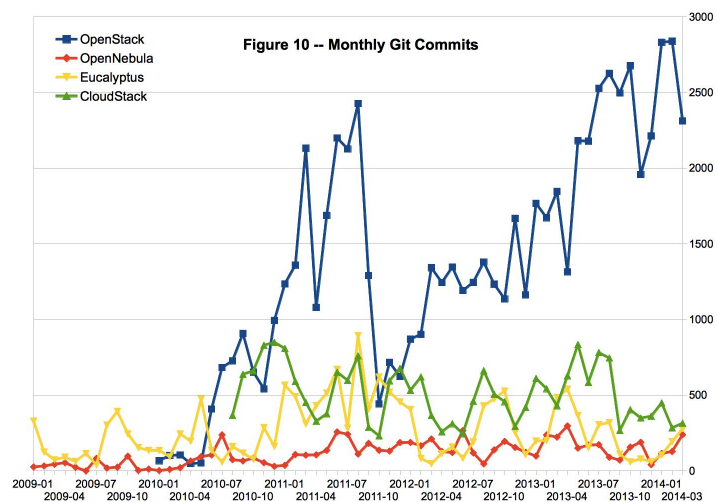


In Figuur 5 is tevens de groeiende interesse in Openstack te zien: deze grafiek geeft de totale community grootte weer. Er is duidelijk te zien dat de interesse in Openstack sinds 2013 sterk toegenomen is, waarbij Cloudstack op ongeveer de helft van deze groei zit boven de stagnerende groei van Eucalyptus.



Git contributie

Ten slotte zijn de maandelijkse git commits interessant. Uit de hoeveelheid Git commits is namelijk af te leiden hoeveel suggesties en toevoegingen er worden bijgedragen aan de code van een project door ontwikkelaars en leden van de community. In figuur 10 is te zien dat Openstack wederom actiever is dan de andere kandidaten. Eucalyptus en Cloudstack hebben beide een redelijk constante stroom commits. Als er echter vergeleken wordt met de hoeveelheid commits welke op Openstack uitgevoerd wordt, ligt deze significant lager. Naast een groter volume commits is er ook een duidelijk stijgende lijn te zien in het aantal commits voor Openstack: dit is conform met de stijgende hoeveelheid contributors welke in de eerdere grafieken weergegeven werd.



Conclusie community analyse

Als we kijken naar de gegevens in de bovenstaande grafieken, deze combineren met de informatie over Open-source lifecycles en de huidige staat van pakketten, dan kunnen er een aantal conclusies getrokken worden.

Ten eerste is het duidelijk dat als er naar volume word gekeken Openstack ver boven de competitie uitgedroeid is. Met een userbase welke groter is dan de andere pakketten gecombineerd, een groter aantal contributors en een constant groeiende community is Openstack wat dit betreft duidelijk de beste keuze.

Als we kijken naar op welk punt Openstack van zijn lifecycle is, kan er veilig gezegd worden dat Openstack zich tussen Maturity en Mainstream bevindt. Grote namen als Walmart, Paypal en NASA^[12,13] hebben zich aangesloten bij Openstack, er zijn communities dedicated aan het ontwikkelen van de verschillende sub-projecten van Openstack, consultancy en hosting bedrijven beginnen zich steeds meer te verdiepen in het aanbieden van Openstack en er zijn verschillende vertakkingen van Openstack ontstaan (Packstack, devstack, Helion, Mirantis). In deze fase kan er aangenomen worden dat het project zich zonder grote risico's verder zal ontwikkelen en dus een solide keuze zal zijn.

3.2) Functieanalyse

Naast de activiteit van de project is het belangrijk dat het project geheel voldoet aan de functionele eisen van i3D.net. Hiervoor zijn de specificaties per pakket naast elkaar gezet om dit te bepalen. Verdere functionaliteit is lastig in te schatten op theoretische basis en zal tijdens het Proof of Concept behandeld worden.

In deze sectie wordt er ingegaan op de theoretisch ondersteunde functionaliteit van de pakketten. Indien de volledige functionaliteit van een pakket getest wordt, moet deze voornamelijk tegenover de eisen van i3D.net getest worden in een praktische omgeving. Om dit te realiseren moet er een complete testomgeving opgezet worden welke vervolgens getest wordt op de gestelde eisen. In dit project zou dit teveel tijd en resources kosten waardoor gekozen is voor één project gebaseerd op de globale eisen een testomgeving op te bouwen.

De punten die de opdrachtgever als belangrijk heeft aangegeven, zijn:

- Hyper-V compatibiliteit
- AWS compatibiliteit
- API's zodat functionaliteit ingebouwd kan worden in het admin panel van i3D.net

Verder zijn alle pakketten vooraf gekozen op de algemeen criteria dat deze in staat zijn om private en public clouds te realiseren.

Door kritisch te kijken naar de beschikbare white papers en feature sheets is de volgende tabel opgesteld

	Functionaliteit	Hyper-V	AWS compatibiliteit	API functionaliteit
Pakket				
Eucalyptus		--	++	+
Cloudstack		+	++	++
Openstack		++	++	++

Aan de hand van de bovenstaande eisen is dus te zien dat Eucalyptus vrijwel direct afvalt door de vaste requirement van Hyper-V support. Verder zijn over het algemeen alle pakketten redelijk uitgebreid op het gebied van functionaliteit. Het doorslaggevende voor i3D is echter of de functionaliteit aan te passen is naar wens. In alle gevallen is dat in zekere mate mogelijk. Openstack heeft echter een toegewijde community die constant streeft naar het uitbreiden van de userbase en het betrekken van de community bij de development.

Conclusie Functie Analyse

Aan de hand van de globaal verwachte functionaliteit is Openstack de meest geschikte keuze. Alhoewel Cloudstack bijna dezelfde functionaliteit biedt als Openstack, was er veel onduidelijkheid over de beloofde functionaliteit en in hoeverre deze beschikbaar was. Omdat transparantie erg belangrijk is en er geen tijd is om voor onverwachte problemen tijdens het project te staan, is er daarom gekozen voor Openstack. Openstack biedt een duidelijk beeld van wat er te verwachten valt indien Openstack in gebruik genomen wordt en geeft aan wat er in ontwikkeling is met een planning voor verwachte implementatie.

3.3) Advies: Welk pakket is geschikt voor i3D.net?

Na het bekijken van de functionaliteit en de community's van alle project kan geconcludeerd worden dat voor i3D.net het Openstack pakket het beste past. Dit pakket heeft een grote en actieve community welke ondersteund wordt door grote toonaangevende bedrijven. De community heeft inmiddels een vaste development cycle opgezet en heeft in principe een vaste groep developers welke aan hun eigen sub-projecten werken. De actieve community en snelle development is voordelig voor i3D.net in verband met de snelle patches voor problemen in Openstack en fixes voor kritieke issues.

De openheid van Openstack zorgt er tevens voor dat indien i3D genoeg in het project ziet, zij hier hun eigen development team aan kunnen toewijden om zelf bij te dragen aan het project. Om deze redenen is er dus besloten om met Openstack verder te werken aan het Proof of Concept. Met dit Openstack Proof of Concept kunnen verder vragen en situaties getest worden.

4.)Advies voor i3D.net

Na het onderzoeken van verschillende pakketten en het opzetten van een Openstack Proof of Concept kunnen er een aantal conclusies getrokken worden over de vraag “Wat voor IaaS oplossing past het beste bij i3D.net en hoe beïnvloedt dit de diensten van i3D.net”.

Ten eerste kan er het advies om verder te gaan met Openstack. In alle uitgevoerde testen is Openstack op het moment duidelijk het beste naar voren gekomen. Het is een sterk groeiend product wat op het moment al sterke core functionaliteit heeft en snel groeit door middel van support van grote spelers als NASA, Paypal, Walmart en HP. Als er gekeken wordt naar hoe actief de community op het moment is en hoe het staat met de groei van Openstack tegenover de gemiddelde life cycle van Open-Source projecten is er tevens geen vermoeden dat het Openstack project binnenkort verlaten wordt of buiten development raakt.

Als er gekeken wordt naar het in gebruik nemen van Openstack zijn er een aantal punten sterk aan te raden. Ten eerste is er helaas geen tijd geweest om een opstelling uit te werken welke voor productie geschikt zou zijn. Vanwege de grootte van het Openstack project wordt er daarom ook aangeraden om een dedicated team van 2-3 medewerkers in te zetten om het Openstack project uiteindelijk te realiseren voor i3D.net. Op deze manier kan de workload voor alle onderdelen van Openstack verspreid worden en kan er per medewerker meer in-depth kennis opgedaan worden van de configuratie van elk sub-project van Openstack.

Verder is er tijdens gesprekken met medewerkers gebleken dat veel automation op het moment via scripting in het admin panel in combinatie met PXE verloopt. Vanwege de architectuur van Openstack wordt er sterk aangeraden om een automation platform op te zetten welke gebruikt kan worden voor het uitrollen en configureren van nieuwe Openstack nodes. Dit voorkomt een hoop repetitief handmatig werk wat gemakkelijker door een automation applicatie als Puppet, Chef of Salt uitgevoerd zou kunnen worden.

Op basis van het Proof of Concept kan er onder andere geconcludeerd worden dat de functionele eisen welke i3D.net wil behalen met Openstack grotendeels te bereiken zijn. Alle basisfunctionaliteit welke van een cloud verwacht zou worden is te implementeren via Openstack, echter zijn er een aantal later besproken optionele punten waar Openstack nog niet aan voldoet. Ten eerste was er de vraag of Shared Storage een mogelijkheid was voor Openstack. Tijdens het uitvoeren van dit project is Shared Storage nog geen ondersteunde functie van Openstack, echter is het sub-project Manila inmiddels opgezet om deze functionaliteit toe te voegen aan Openstack. De verwachting is dat dit binnen de komende 3 releases officieel toegevoegd wordt aan Openstack.

De volgende vraag betrof het draaien van virtuele machines geïsoleerd op compute nodes, deze vraag gaat gepaard met de vraag of local storage te gebruiken is met Openstack. De kracht van Openstack ligt hem in het samenvoegen van gedistribueerde hardware naar 1 cluster. Daarom is er na

het onderzoeken van deze vragen ook geconcludeerd dat VM's geheel gecentraliseerd draaien wel mogelijk is, maar sterk af te raden is. Openstack heeft een tweetal projecten gebaseerd op storage; Cinder en Swift, deze gaan er beide van uit dat er gebruik gemaakt wordt van centrale storage pools waar virtuele machines ruimte uit toegewezen krijgen. Vanwege de techniek hoe deze projecten zijn ingericht is er dus ook niet rekening gehouden met individuele pools per node. Dit is mogelijk uit te voeren, maar kost erg veel administratieve handelingen in Openstack om dit werkend te maken en is daarom ook af te raden. Openstack heeft de voorkeur om hardware in grotere pools in te delen, zoals een high performance pool en een low performance pool. Ook is er interesse in het delen van CPU kracht tussen nodes. Dit is een feature welke voor Openstack nog niet beschikbaar is en waar ook nog geen duidelijke plannen voor zijn in de toekomst.

Ten slotte is er nog de vraag geweest hoe Openstack omgaat met high availability. Hierbij moet gekeken worden naar de kracht van cloud computing. Cloud computing biedt namelijk de mogelijkheid om vele generieke "workers" op te zetten welke individueel geen impact op de functionaliteit van een service hebben mocht deze uitvallen, er is hooguit een tijdelijk disruptie in het managen van de workers. Hierom ligt de focus bij Openstack dus ook niet op het high available houden van VM's. Wat wel high available opgezet kan worden is de backend van Openstack, namelijk de databases services, storage en API services. Deze kunnen door middel van een generieke Pacemaker stack welke zorgt voor failovers waar nodig.

Concluderend kan er dus gezegd worden dat Openstack de juiste keuze is voor i3D.net en dat met een geschikt team en een duidelijke doelstelling voor de Openstack cloud dit een zeer sterke additie in het aanbod van i3D.net oplevert.

Verklarende woordenlijst

Term	Verklaring
AWS	Amazon web services
Cloud hosting	Het via het internet op aanvraag beschikbaar stellen van hardware, software en gegevens
colocatie	Het aanbieden van rackspace
Hyper-V	Virtualisatie oplossing van Microsoft
Hypervisor	Een stuk hard- of software dat virtuele machines creëert en aanstuurt
i3D.net	Rotterdamse managed hosting provider
IaaS	Infrastructure-as-a-Service
KVM	Linux virtualisatie oplossing
NIC	Network Interface controller
Node	Fysieke machine in het Openstack netwerk
VMWare	Een Amerikaans softwarebedrijf dat diverse producten uitbrengt die virtualisatie mogelijk maken.
PoC	Proof of Concept
SaaS	Software-as-a-service
PaaS	Platform-as-a-service
SLA	Service Level Agreement
Git	Een gedistribueerd versiebeheersysteem
Chargeback	Kosten rekenen voor de gebruikers van de door hen geconsumeerde ICT-diensten