Afbeelding met cirkel, Graphics, clipart, logo

Door AI gegenereerde inhoud is mogelijk onjuist.

**Opdrachtgever(s): Rob Smit**

**Klas:**  **HA5C**

**Groepsleden:** **Bastiaan Booij**

**Docenten: Ron Matena & Rob Smit**

**Versie:** v1.0

**Datum:** 19-03-2025

# 1 Voorwoord:

Voor u ligt het verslag van mijn ICT-project. Ik had de mogelijkheid voor de uitvoer van dit project niet gehad zonder de steun en begeleiding van mensen en organisaties die ik graag wil bedanken.

Allereerst wil ik mijn docenten, Rob Smit en Ron Matena, hartelijk danken voor hun begeleiding, feedback en het delen van hun kennis. Hun ondersteuning heeft mij geholpen met mijn groei en ontwikkeling binnen dit ICT.

Daarnaast wil ik het bad eendje bedanken, die mij op mysterieuze wijze door mijn problemen heen leidde.

Tot slot wil ik het Hondsrug College bedanken voor het bieden van de ruimte om mijzelf te ontwikkelen binnen ICT.

Ik hoop dat u met plezier dit verslag leest en dat het een goed beeld geeft van mijn leerproces en de behaalde resultaten.

Bastiaan Booij

Inhoudsopgave

[1 Voorwoord: 2](#_Toc194843888)

[2 Inleiding: 4](#_Toc194843889)

[2.1 Hoofdvraag: 4](#_Toc194843890)

[2.2 Deelvragen: 4](#_Toc194843891)

[3 Probleem/gewenste verandering: 5](#_Toc194843892)

[3.1 Probleemstelling 5](#_Toc194843893)

[3.2 Aan welke eisen moet de oplossing voldoen 5](#_Toc194843894)

[3.3 Wat is er onderzocht, welke oplossingen er gevonden zijn 5](#_Toc194843895)

[3.3.1 Welke workload er gebruikt zou worden: 5](#_Toc194843896)

[3.3.2 Hoe deze workload verdeeld zou worden/cluster: 5](#_Toc194843897)

[3.4 Hoe dit opgelost kan worden 6](#_Toc194843898)

[4 Resultaten: 8](#_Toc194843899)

[4.1 Wat voor workload moet er verdeeld worden? 8](#_Toc194843900)

[4.2 Hoe moet deze workload verdeeld worden? 8](#_Toc194843901)

[4.3 Hoe krijgen alle computers toegang tot de data? 8](#_Toc194843902)

[4.4 is er daadwerkelijk een winst in snelheid? 8](#_Toc194843903)

[4.4.1 De hoeveelheid computers: 8](#_Toc194843904)

[4.4.2 De kracht van elke computer: 8](#_Toc194843905)

[4.4.3 De grote van de workload: 8](#_Toc194843906)

[4.4.4 Conclusie: 9](#_Toc194843907)

[5 Realisatie 10](#_Toc194843908)

[6 Conclusie: 11](#_Toc194843909)

[7 Advies: 12](#_Toc194843910)

[8 Reflectie: 13](#_Toc194843911)

[9 Toekomst: 15](#_Toc194843912)

[10 Bronnen: 16](#_Toc194843913)

[11 Bijlagen: 17](#_Toc194843914)

# 2 Inleiding:

Dit project is ontstaan vanuit een idee van Rob Smit, wat inspiratie gaf om het verder uit te werken en tot leven te brengen. Het idee van het project was om een zware taak te verdelen over meerdere computers, zodat deze sneller verloopt

Het project zou een groep computers (een cluster) moeten kunnen laten samenwerken. Om die zware taak in een kortere tijd uit te laten voeren, bij zware taken kunt u denken aan, een render van een grote afbeelding of video, het uitvoeren van grote calculaties en het versnellen van de responstijd van een LLM (beter bekent als “AI”).

In het onderzoek kwam ik een applicatie genaamd “Flamenco” tegen, dit was een reeds bestaande oplossing voor het verdelen van een render-workload (een zware taak waar o.a. afbeeldingen van hoge resolutie worden geladen/gegenereerd).

In dit verslag wordt duidelijk wat uiteindelijk tot de keuze van Flamenco leidde, wat de moeilijkheden waren, welke successen er behaald zijn, en meer.

## 2.1 Hoofdvraag:

* Hoe is een server te realiseren die een workload kan verdelen over meerdere computers, en daardoor efficiënter en sneller kan werken?

## 2.2 Deelvragen:

* Wat voor workload moet er verdeeld worden?
* Hoe moet deze workload verdeeld worden?
* Hoe zorg je dat alle computers bij de data kunnen?
* Is er echt een winst in snelheid?

# 3 Probleem/gewenste verandering:

## 3.1 Probleemstelling

Een zware workload (taak) duurt vaak lang om te voltooien, maar wat als deze taak verdeeld kon worden, dat zou de taak sneller laten voltooien en daarmee tijd en geld besparen. Er bestaat al een applicatie met het doel dit probleem op te lossen, maar mist veel documentatie.

## 3.2 Aan welke eisen moet de oplossing voldoen

Als eerste eis moet er een werkend resultaat zijn, die een workload verdelen kan over meerdere computers, deze uitvoeren, en een resultaat kunnen presenteren

Verdere eisen zijn:

1. Het project moet zonder problemen kunnen draaien, en blijven draaien.

2. Het project moet, zodra hij ingesteld is, zonder verdere console commands beheerd kunnen worden.

## 3.3 Wat is er onderzocht, welke oplossingen er gevonden zijn

In het begin van het onderzoek was deze verdeeld in 2 grote stukken:

* Welke workload er gebruikt zou worden
* Hoe deze workload verdeeld zou worden/cluster

### 3.3.1 Welke workload er gebruikt zou worden:

Rob Smit had als workload een fractal (een soort wiskundig gegenereerde afbeelding) gesuggereerd, waarbij uit onderzoek bleek dat deze workload niet vaak gevraagd werd, en er een meer-gebruikte workload beschikbaar was, Blender.

### 3.3.2 Hoe deze workload verdeeld zou worden/cluster:

Eerst is er onderzocht of de bekende reeks van cluster-software, en een paar minder bekenden, zouden passen. Dit bleek in deze situatie niet het geval. De doorslag kwam wanneer beseft werd dat er te diep op de term “cluster” ingegaan was.

Een simpele definitie van een cluster is namelijk: “Een groep computers die met elkaar kunnen communiceren.” Dit betekent dat er geen software hoeft te zijn die alle servers op OS-niveau aanstuurt. Er kan namelijk ook een applicatie op elke computer draaien, die dan samenwerkt. Daardoor is uiteindelijk Flamenco verkozen als verdeel-software

Uit verder onderzoek naar flamenco bleek duidelijk dat de documentatie vooral gebrekkig was aan de Linux-kant. Dat, samen met het feit dat Linux minder zwaar draait (minder vraagt van de computer) dan Windows, leidde tot het besluit deze applicatie uit te gaan zoeken en documentatie te verbeteren waar dat kan.

Uit onderzoek is gebleken dat de Linux versie in zo verre anders is, dat de documentatie niet helpt, en je het eigenlijk zelf moet uitzoeken. Zo zegt de documentatie dat een bepaalde optie optioneel aangezet kan worden. En dat klopt, voor de Windows-versie. Op de Linux-versie bleek echter deze optie standaard aan te staan, dit geeft onduidelijke foutmeldingen die aangeven dat de fout iets te maken heeft met de NAS, en aangeven dat het aan de NAS-kant fout gaat. Door deze optie uit te zetten was het probleem verholpen en stuitte ik op het volgende probleem. Er zit dus een fout in de configuratie voor de Linux-versie van Flamenco die, zonder herstel, het programma onbruikbaar maakt

Flamenco deelt bestanden via een NAS, zo kunnen alle computers bij de bestanden. Maar flamenco benaderd de NAS op een bijzondere manier, wat het bestandssysteem van Ubuntu 22.04 niet kan verwerken, dit kost veel moeite om een stabiele verbinding te leggen tussen Flamenco en de NAS. Ook moet flamenco gelijk bij het opstarten de NAS zien als een bestaande locatie, anders maakt hij lokaal een “locatie” aan

**Verdere mogelijke problemen bij gebruik van Linux kunnen zijn, maar zijn niet gelimiteerd tot:**

Gebrek een cross-comptabiliteit met Windows, dit wordt veroorzaakt door een verschil in de bestandsystemen van Windows en Linux. Zo gebruikt Windows letters voor opslagapparaten (denk aan C:), terwijl Linux meestal drie letters gebruikt. Zoals: sda, sdb, sdc, maar ook nvme0n1. Het helpt ook niet dat Linux distro’s soms ook usb-opslagapparaten een andere naam geven. Het probleem zou zich bijvoorbeeld voordoen wanneer een Windows apparaat een taak naar een op Linux draaiende Flamenco installatie stuurt, en als opslaglocatie voor de voltooide taak bijvoorbeeld “C:\users\flamenco\documenten\resultaten” meestuurt.

Omdat Linux “C:” niet herkent, zal een van de volgende dingen voorkomen:

1. Flamenco geeft een error dat de resultaten niet kan opslaan
2. Flamenco laat een nieuwe map in zijn map aanmaken genaamd “C:\users\flamenco\documenten\resultaten”
3. Flamenco laat een nieuwe locatie aanmaken genaamd “C:/users/flamenco/documenten/resultaten”

Het verschil tussen situatie 2 en 3 zit hem in de “\” en “/”. Ik weet niet of Linux de “\” van Windows vertaald in zijn eigen “/”. Bij situatie 2 zou “C:\users\flamenco\documenten\resultaten” de naam van één map zijn. Terwijl bij situatie 3 “C:/users/flamenco/documenten/resultaten” de map “resultaten” in de map “documenten” in de map “flamenco” in de map “users” in de map “C:” in de root directory van Flamenco.

Gebrek aan duidelijke debug-output, zo geeft Windows een log waarin alle activiteiten en fouten worden weergegeven. Bij Linux kan ik deze (mogelijk door gebrek aan documentatie) niet vinden. Dit had mij mogelijk kunnen helpen bij het succesvol instellen van een Linux-gebaseerde cluster

## 3.4 Hoe dit opgelost kan worden

Het plan is om een documentatie te schrijven voor dit programma, wegens de problemen die ik ondervonden heb, kan ik niet een volledige documentatie schrijven. Wel kan ik alle ondervonden problemen en oplossingen documenteren en zorgen dat deze beschikbaar zijn voor mensen die vastlopen.

Verder moet ik helaas aangeven dat ik, wegens gebrek aan tijd overgeschakeld ben naar Windows. Wegens de betere compatibiliteit met Windows-apparaten, betere documentatie, en beter debug log, dit heeft bijvoorbeeld andere eigenaardige problemen aangegeven, zodat deze verholpen konden worden.

# 4 Resultaten:

## 4.1 Wat voor workload moet er verdeeld worden?

Ik heb gekozen voor blender renders, deze zijn ik de design en video-industrie vaak van belang, en bevatten een echt probleem.

## 4.2 Hoe moet deze workload verdeeld worden?

In Flamenco zijn er managers en workers. Een manager verdeeld de taken automatisch, en houd daarbij rekening met krachtigheid van machines, en beschikbaarheid daarvan. Een worker koppelt zichzelf aan een manager, wanneer deze manager vervolgens een taak krijgt wordt deze verdeeld over de workers, zodat deze uitgevoerd kunnen worden.

## 4.3 Hoe krijgen alle computers toegang tot de data?

De data bestaat uit meerdere delen, de originele workload, die naar de manager toe moet. De verdeelde stukken van die workload, die naar de workers toe moeten. En de resultaten die opgeslagen moeten worden. Alle workers en de manager moeten natuurlijk wel allemaal bij die data kunnen, hiervoor wordt een NAS gebruikt. Alle workers en de managers verbinden ermee, en hebben allemaal toegang tot het bestand. Flamenco past het bestand nog aan, zodat hij beter verdeeld kan worden, vervolgens wordt dat bestand uitgevoerd door de aangewezen workers.

## 4.4 is er daadwerkelijk een winst in snelheid?

In een simpel antwoord zijn die er niet, dat hangt af van meerdere factoren.

Hierbij kunt u denken aan: de hoeveelheid computers, hoe krachtig elke computer is en zelfs hoe groot de workload is

### 4.4.1 De hoeveelheid computers:

In principe klinkt het logisch hoe meer computers, hoe beter de taak verdeeld kan worden, en dus hoe sneller hij afgerond is. Er zit echter wel een addertje onder het gras.

### 4.4.2 De kracht van elke computer:

De computer moet Flamenco draaien, en CPU en/of GPU (en RAM) beschikbaarheid overhebben zodat het daadwerkelijk een taak kan uitvoeren.

### 4.4.3 De grote van de workload:

Zolang de taak groot genoeg is, zijn er geen minimale eisen. Als je een uur over een taak zou doen, maar een andere pc doet de rest in een uur scheelt het alsnog wat, maar veel taken duren langer. Duurt een taak voor vele pc maar 10 seconden, dan is het niet logisch on een computer in de cluster te zetten die er een uur over doet. Dat werk averechts, waar de taak langer wordt wegens die pc.

### 4.4.4 Conclusie:

In veel situaties is er een merkbare winst in snelheid tot voltooiing van de taak, wel moet er oplettend gekeken worden naar:

* Of de computer niet, in verhouding tot de andere workers te traag is.
* Of er meerdere computers beschikbaar zijn.
* Of de taak groot genoeg is om door de cluster te laten uitvoeren.

5 Realisatie:

Dit project is te verdelen in de volgende onderdelen:

- Een netwerk waar alle apparaten elkaar kunnen bereiken

- Een sterke computer die de manager en een worker kan draaien

- Drie computers met een worker installatie erop (voor proof of concept)

- Een NAS die de data kan delen met alle workers en de manager

- Een apparaat die de taken naar de manager stuurt (persoonlijke laptop)

Na het onderzoek is een kleine test-setup gemaakt van 3 Nucjes (kleine computertjes) waarvan één ingesteld als NAS en 2 met Flamenco. Er waren wat problemen in de setup, dit lag aan de processorcapaciteit van de NUC waarop de manager draaide.

Hierna is een setup met een server (HP Proliant dl380 gen 8) als flamenco manager en worker, een NUC als Nas, en 2 Dell optiplex 9010 pc’s als workers. Ook is een demo-NUC aangemaakt, die bij een live demonstratie kan laten zien wat de limitaties zijn van dit project.

Voor de demo is vervolgens een klein netwerk met een wifi-router, en een kleine switch, deze zorgen ervoor dat alle workers, de NAS, de manager, en een opdracht-gevend apparaat (in dit geval een laptop) via wifi.

6 Conclusie:

Tijdens het schrijven van dit verslag is er nog geen voltooid resultaat, echter zijn er wel conclusies die getrokken kunnen worden.

Zo kan geconcludeerd worden, op het moment van schrijven, het gebruik van Linux vermeden moet worden. De documentatie is te onvolledig om een stabiel resultaat te krijgen.

Ook kan geconcludeerd worden dat er minimaal 1 computer een verhoogde RAM en CPU-capaciteit moet hebben. Het draaien van een manager is een zwaar proces.

Tot slot moet gezegd helaas gezegd worden, dat wegens de moeite voor het opzetten van Flamenco. Het voor veel bedrijven aantrekkelijker is om een betaalde onlineservice te gebruiken.

# 7 Advies:

Het advies voor de opdrachtgever luidt het volgende:

1. Voor de manager is Windows aan te raden als OS, deze geeft live log output, die problemen kan aangeven, dit doet Linux niet
2. Het is voor kleinschalig gebruik aantrekkelijker om een betaalde onlineservice te gebruiken.
3. Het best kunt U een Manager op Windows, en de workers op Linux draaien, omdat Linux lichter draait
4. Het beste is om je NAS te laten delen via SMB, aangezien dit stabiel is, en geen connectieprobemen krijgt met flamenco

# 8 Reflectie:

Uit de vele uren die ik in dit project heb gestoken komen natuurlijk wat verbeterpuntjes.

Zo heb ik mijn tijd niet helemaal even verdeeld. Ik heb ontdekt dat als ik mij focus op een probleem, dat ik probeer kleine stapjes terug te zetten, terwijl het soms handiger is om grotere stappen terug te zetten. En even van een afstandje te kijken wat er eigenlijk wel/niet goed en fout gaat. Dit is één van de momenten dat ik het moeilijker maak voor mezelf, dan het hoeft te zijn. Waarom ik dit doe is mij nog onduidelijk, maar het feit dat ik nu weet dat dit een probleem is, bied mij de mogelijkheid hieraan te werken.

Ook had ik wat dieper moeten nadenken over wat de term “cluster” inhield en dat dat niet per se een groep computers die met een stuk speciale software elkaar kunnen beheren hoeft te zijn. Maar dat het ook gewoon een paar computers (op hetzelfde netwerk) die samenwerken kunnen zijn. Als ik mij dit eerder beseft had, had dat mij tussen de 20 en 30 uur gescheeld aan onderzoek en experimentatie. Dit valt ook terug te koppelen aan het “stapje terug doen” probleem van hiervoor. Zo ben ik met de eerste definitie voor “cluster” waar ik aan dacht verder gaan werken. Maar had ik deze eigenlijk wat breder moeten bekijken.

Nu iets wat meer met slordigheid, of er gewoonweg niet aan denken is. Mijn uren documenteren ging wat minder goed dan gehoopt, zo vergat ik regelmatig uren te documenteren. Dir realiseerde ik me vervolgens later. Vervolgens maak ik een snelle notitie van mijn uren. Met de intentie die in het logboek te zetten. Dit vergeten liep echter zo uit de hand dat ik op een gegeven moment 2 logmoeken had, één in Excel, en één in kladblok (dit start sneller op). Het herstellen van dit logboek, alles ik Excel zetten dus koste echter onnodig veel tijd, die beter gespendeerd was aan dit verslag.

Verder is er de onderzoeksfase die voor 80% overbodig was. Zoals al eerder gezegd focuste ik mij op de verkeerde definitie van “cluster”, met als resultaat dat ik ongeveer 80% van de tijd van mijn onderzoek in het uitzoeken van de beste cluster-software stak. Het “stapje terug” zal hier ook gelopen hebben.

Voor positieve opmerkingen, ook al was ik al 4 lessen met hetzelfde probleem bezig. En had ik al behoorlijk wat stappen teruggezet gaf ik niet op, ik heb ondanks de tegenslag mijn doorzettingsvermogen vastgehouden. Aangezien dit een vaardigheid is waarvan ik dacht dat ik die niet echt had/beheerste, behalve in specifieke situaties. Dit maakt mij op dat gebied best trots dat ik dat blijkbaar beheers.

In eerste instantie had ik een goed doordachte planning, weliswaar beter in mijn hoofd dan op papier, maar hij was er wel. Ik was echter onder de indruk dat ik binnen 100 uur een werkend systeem kon produceren, helaas bleek het gebrek aan documentatie en console daar de das om te draaien. En heb ik niet tot nauwelijks tijd om vóór de presentatie nog een uitgebreide documentatie te maken. Ik heb echter op dit moment het plan om verder te werken aan deze documentatie, ook na mijn examens.

Ondanks alle tegenslagen kan ik zeggen dat ik blij, en trots ben op wat ik geleerd heb, zowel over mezelf als over mijn project. Maar ben ik me ook bewust geworden van dingen waar ik voortaan rekening mee moet houden. Zo moet ik flexibeler plannen, “een stap terug doen” en me bewust zijn over mijn logboeken, en het bijhouden daarvan. Ik ben vooral blij met de kwaliteiten die ik heb waarvan ik dacht dat ik die niet beheerste. Ik kijk daarom ook met een positieve blik over het project heen.

# 9 Toekomst:

Mogelijk wordt dit project voortgezet. Dit zal inhouden dat er verdere pogingen gedaan zullen worden on een werkende en stabiele Linux installatie en daar vervolgens een documentatie voor te maken.

Na het leveren van een duidelijke documentatie voor Linux zijn er geen verdere toekomstplannen met Flamenco.

# 10 Bronnen:

<https://blenderartists.org/> (verschillende forums)

[blender.org - Home of the Blender project - Free and Open 3D Creation Software](https://www.blender.org/)

[Flamenco](https://flamenco.blender.org/)

[Reddit](file:///C:\Users\SmiR\AppData\Local\Microsoft\Windows\INetCache\Content.Outlook\661828UH\reddit.com)  (verschillende forums)

# 11 Bijlagen:

Alle bestanden zijn beschikbaar ik GitHub, hiervoor gebruikt u de volgende link:

<https://github.com/BasBaas08/workload-sharing-cluster>