

REACHING FOR THE STARS

SEMANTIC WEB (50728SEW6Y)

MARIT BEEREPOOT mbt297/10983430
JESSY BOSMAN jbn482/11056045
VINCENT DAMEN vdn340/11034734

"To infinity and beyond!"

- Buzz Lightyear



INHOUDSOPGAVE

1. Inleiding	4
2. Application Design	4
2.1 Goal	4
2.2 Users	4
2.3 Design	4
2.4 Walkthrough	5
3. Domain Modeling	6
3.1 Domain en scope	6
3.2 Methodology	7
3.3 Conceptualization	7
3.4 Ontology, class restrictions en bestaande ontologies	7
3.5 Inferences	8
4. Data reuse & querying	8
4.1 De externe bronnen	8
4.2 Waarom deze externe bronnen?	9
4.3 Data integreren	9
4.4 SPARQL queries	9
4.5 Inferences	10
5. Evaluatie	11

1. INLEIDING

Het universum is iets unieks, voor de meeste onaantastbaar en toch elke avond zichtbaar vanuit je tuinstoel. De sterren verlichten ons elke dag, maar toch is er nog geen linked data applicatie voor een visualisatie van dit mooie aanzicht. Daarom hebben wij besloten om dit te creëren. Door het gebruik van linked data is een dataset gecreëerd. Met behulp van diverse programmeertalen en deze dataset hebben wij onze eigen lucht vol sterren gebouwd. Ook onze melkweg verdient het om een mooi overzicht te krijgen. Daarom zijn een flink aantal manen, dwerg planeten en asteroïden genomen en ook gevisualiseerd. Om nog meer te leren over deze prachtige planetoïden is alle gegevens beschikbaar gesteld om te vergelijken. Kortom een prachtige applicatie om genoeg over te vertellen.

2. APPLICATION DESIGN

In dit hoofdstuk zal het doel, wat vastgesteld is voor het bouwen van de applicatie, worden besproken. Daarnaast zal de doelgroep worden gedefinieerd en wordt er onderbouwd waarom dit als doelgroep gezien wordt. Ook is er gedacht voor eventuele toepassingen van onze applicatie voor diverse doelgroepen. In dit hoofdstuk worden bovendien de design keuzes uitgelegd en beargumenteerd waarom hiervoor gekozen is. Er wordt ingegaan op eventuele interacties die mogelijk zijn in onze applicatie. Deze worden ook waar nodig beargumenteerd. Tenslotte volgt een sectie die een inzicht geeft in hoe er door de applicatie heen genavigeerd kan worden.

2.1 Goal

Het doel van de applicatie is het verkrijgen van informatie over hemellichamen en het vergelijken van verschillende hemellichamen. De applicatie is dus vooral gericht op het tonen van informatie van de verschillende hemellichamen. Dit wordt gedaan op verschillende manieren. De applicatie heeft ten eerste een interactief deel om de applicatie interessanter te maken. Alle hemellichamen worden namelijk in verschillende kaartweergaven getoond.

Daarnaast kan alle informatie van de hemellichamen met elkaar worden vergeleken waardoor gebruikers meer kennis kunnen opdoen van de eigenschappen van verschillende hemellichamen. Ook biedt de applicatie ook een wereldkaart met sterrenwachten en telescopen aan. Wanneer een gebruiker geïnteresseerd is geraakt in de verschillende hemellichamen, kan deze via dezelfde interface vinden waar hij/zij deze in het echt kan bekijken. De taken die de applicatie uitvoert zijn dus het ophalen en tonen van informatie over hemellichamen in verschillende weergaven en geeft de mogelijkheid

deze te vergelijken. Met deze applicatie wordt het dus mogelijk om informatie te kunnen vergelijken, wat niet veel voorkomt in linked data ontologies zoals DBpedia. Hiermee wordt geprobeerd om unieke diepte te geven aan de opgehaalde data.

2.2 Users

De doelgroep van de applicatie zijn eigenlijk alle mensen die geïnteresseerd zijn in het universum en de hemellichamen. Daarnaast zou het ook gebruikt kunnen worden door scholieren of studenten die sterrenkunde hebben, om inzicht te krijgen in de verschillende hemellichamen en hun eigenschappen. Omdat de interface overzichtelijk is en het grootste gedeelte van de navigatie via klikken gaat, zou de applicatie zelfs door kinderen gebruikt kunnen worden. De applicatie kan eigenlijk door iedereen die informatie van hemellichamen wilt verkrijgen gebruikt worden.

Deze applicatie is een antwoord op de behoefte van mensen om meer te weten over de verschillende hemellichamen. Doordat alle informatie van de verschillende planeten en sterren in een simpele applicatie te vinden zijn, is het makkelijk informatie hierover op te zoeken en de informatie van verschillende hemellichamen naast elkaar te zien. Op deze manier is het bijvoorbeeld mogelijk om te kijken of Jupiter groter is dan Mars (of juist niet). Daarnaast maakt het overzicht van sterrenwachten en telescopen het dus ook makkelijk om te vinden waar de hemellichamen in het echt bekeken kunnen worden.

2.3 Design

Het design van de applicatie is op een dusdanige manier ontworpen dat het gebruik voor iedereen toegankelijk is. De applicatie bestaat uit verschillende onderdelen. Het eerste onderdeel is een menu, waarin kan worden gekozen welke informatie getoond moet worden. Er kan hier worden gekozen uit Stars, Planets, Telescopes & Observatories en Dwarfs, Asteroids & Moons. Alle onderdelen bestaan uit een kaart waar de entiteiten worden getoond, en waarbij er een popup met informatie komt als er wordt geklikt.

De kaarten voor de sterren, planeten en dwergplaneten, asteroïden en manen zijn random gegenereerd en geen exacte copy van de werkelijkheid. Hier is voor gekozen omdat het binnen de tijd niet haalbaar was om het sterrenstelsel precies na te maken. Het sterrenstelsel kan eigenlijk alleen correct worden afgebeeld als er een complete 3D weergave wordt gebruikt. Er kon wel gekozen worden om alleen de hemellichamen te laten zien die vanaf aarde zichtbaar zijn, maar hierdoor zouden er veel hemellichamen niet in de weergave komen en

daarnaast is er weinig duidelijkheid over waar asteroïden zich bevinden, omdat deze door de ruimte bewegen. De kaart van de telescopen en sterrenwachten is wel gebaseerd op werkelijke locaties.

Op de kaart kan er op een hemellichaam worden geklikt, waarna een popup verschijnt met de informatie over het gekozen hemellichaam. Hier wordt bijvoorbeeld de radius en de zwaartekracht van een hemellichaam weergegeven. Het design hiervan is bewust heel overzichtelijk gehouden. De abstract is uitvouwbaar, om zo onnodig scrollen te voorkomen. Daaronder staat de basis informatie van een hemellichaam in een overzichtelijke lijst weergave, zodat deze informatie goed te begrijpen is voor elke gebruiker. Daarnaast bestaat ook de mogelijkheid om verschillende hemellichamen met elkaar te vergelijken, door de informatie van twee hemellichamen naast elkaar te tonen.

De vergelijkings popup maakt het mogelijk om hemellichamen te zoeken, via de zoekbalk, en deze met elkaar te vergelijken. De informatie wordt getoond in een tabel, waardoor makkelijk kan worden gezien waar de verschillen in waardes liggen. Het overzicht laat alleen overeenkomstige informatie zien zodat er geen 'nutteloze' waardes worden weergegeven.

2.4 Walkthrough

De applicatie bevat een aantal verschillende mogelijke interacties voor de gebruiker. De eerste manier is via het hoofdmenu. Hier kan worden gekozen welke informatie er getoond wordt door op het onderdeel te klikken (afbeelding 1).

Een van deze manieren is via de sterrenkaart. Op deze sterrenkaart zijn alle sterren zichtbaar (afbeelding 2).

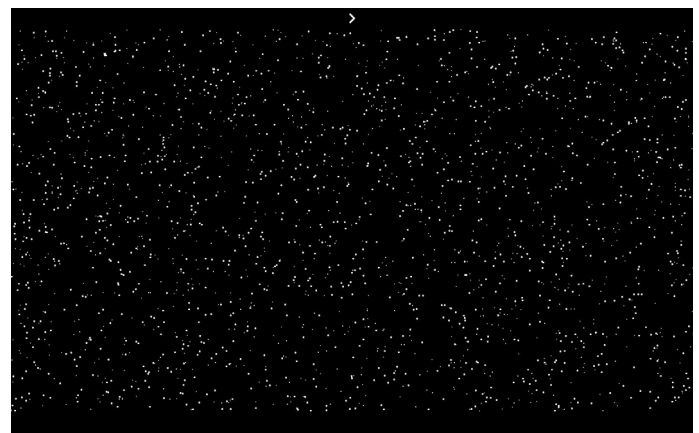
Als er op een ster wordt geklikt verschijnt de bijbehorende informatie. De belangrijkste informatie van een hemellichaam wordt dan getoond door middel van een pop-up op het scherm. Er kan op een knop worden geklikt waardoor er vergeleken kan worden met een andere ster (afbeelding 3).

Ook kan een gebruiker via de applicatie de hemellichamen met elkaar vergelijken. De gebruiker kan twee hemellichamen kiezen waardoor deze vergeleken kunnen worden. Dit kan worden gedaan door de naam van het hemellichaam in de zoekbalk te typen en op zoek te drukken. De informatie van de twee hemellichamen worden dan naast elkaar getoond. Op deze manier kan de gebruiker extra kennis opdoen, omdat de gebruiker er bijvoorbeeld achter kan komen dat Jupiter groter is dan Mars. De mogelijkheid om te kunnen vergelijken is dus vooral bedoeld om extra inzicht te krijgen in de verschillende hemellichamen (afbeelding 4).

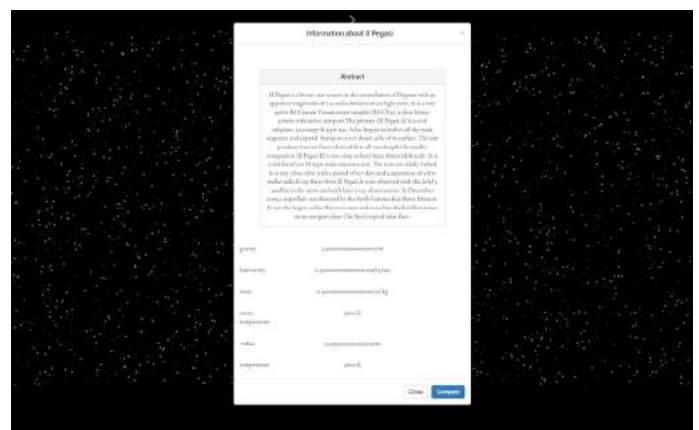
De kaart van de planeten en de kaarten van de andere



Afbeelding 1: Het hoofdmenu



Afbeelding 2: De sterrenkaart



Afbeelding 3: Informatie van een ster



Afbeelding 4: Vergelijkfunctie van de sterren

hemellichamen werken in principe hetzelfde als de sterrenkaart. Op de kaart van de dwarfs, asteroiden & moons zijn de kleinste cirkels de asteroiden, de grotere manen en de vier grotere de dwergplaneten (afbeelding 5). Op de kaart van de planeten zijn alle planeten ingeladen met behulp van afbeeldingen. In het midden is de zon te zien (afbeelding 6).

Aangezien deze kaarten hetzelfde werken is het ook hier zo dat er gebruik wordt gemaakt van pop-ups om de informatie weer te geven. Daarnaast is ook hier weer de vergelijkfunctie beschikbaar (afbeelding 7).

De informatie en de weergegeven eigenschappen zullen anders zijn dan die van sterren, omdat niet alle eigenschappen overeenkomen. Zo heeft een planeet bijvoorbeeld geen helderheid.

Om onnodig zoomen te voorkomen is het mogelijk om het abstract in te klappen, zoals hieronder te zien is (afbeelding 8).

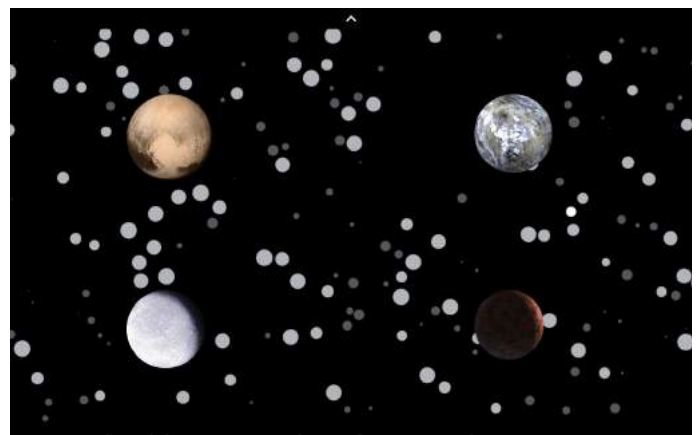
De gebruiker kan ten slotte op een kaart de locaties van alle telescopen en sterrenwachten bekijken. Op deze kaart kan de gebruiker ook inzoomen om een duidelijkere locatie te zien (afbeelding 9).

3. DOMAIN MODELING

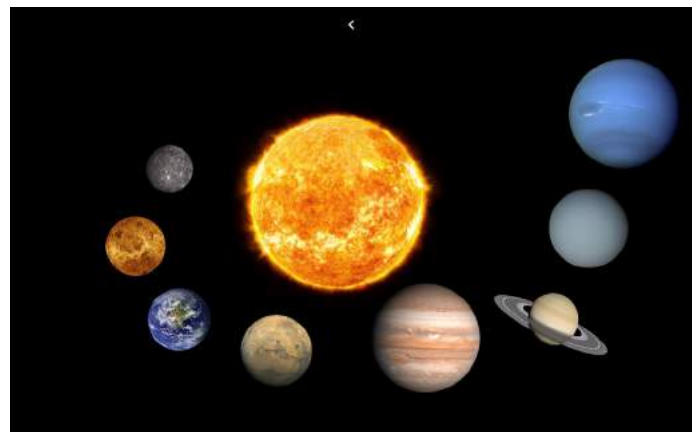
In dit hoofdstuk worden de domain en scope van de applicatie vastgesteld. Dit gebeurt aan de hand van zogeheten competency vragen. Daarnaast wordt er ingegaan over hoe de ontology gebouwd is en hoe deze in elkaar zit. De verschillende classes, class restrictions, inferences en externe bronnen zullen besproken worden.

3.1 Domain en scope

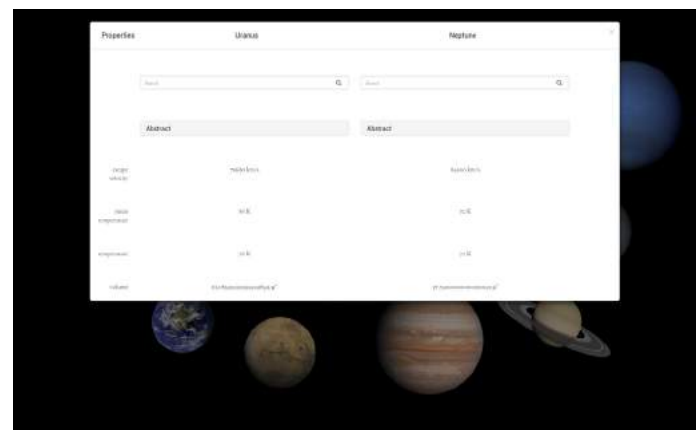
Het domein van de ontology is alles wat onder celestial body valt, de locaties van de sterrenwachten & de telescopen en de ontdekkers van de verschillende hemellichamen. De scope is vastgesteld aan de hand van een aantal competency vragen. De competency vragen luiden als volgt:



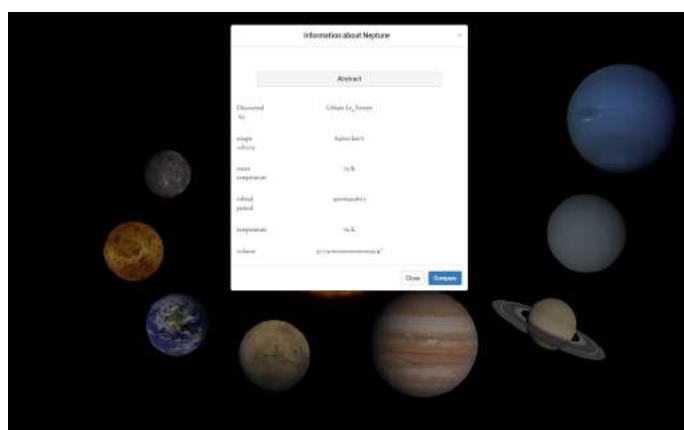
Afbeelding 5: De dwarfs, asteroiden & moons



Afbeelding 6: De Planeten



Afbeelding 7: Vergelijkfunctie van de planeten



Afbeelding 8: Inklapbare abstract



Afbeelding 9: Telescopen en sterrenwachten

- o Welke hemellichamen zijn er?
- o Is Gomez's Hamburger een single star, binary star of multiple star?
- o Op welke plekken is het mogelijk om met telescopen en sterrenwachten de hemellichamen te bekijken?
- o Is de temperatuur op Gomez's Hamburger groter of kleiner dan die van Canis Majoris?
- o Is Pluto een planeet of een dwergplaneet?
- o Is het volume van Pluto groter dan het volume van Mars?
- o Wat de omwentelingssnelheid van Uranus?
- o Wie is de ontdekker van de asteroïde Eris?
- o Van welke planeet is Europa een maan?

Door deze vragen te beantwoorden kwamen we erop dat de ontdekkers van de hemellichamen, basis informatie over de hemellichamen en sterrenwachten en telescopen nuttig zijn om toe te voegen aan de ontology, om de applicatie zo volledig mogelijk te maken.

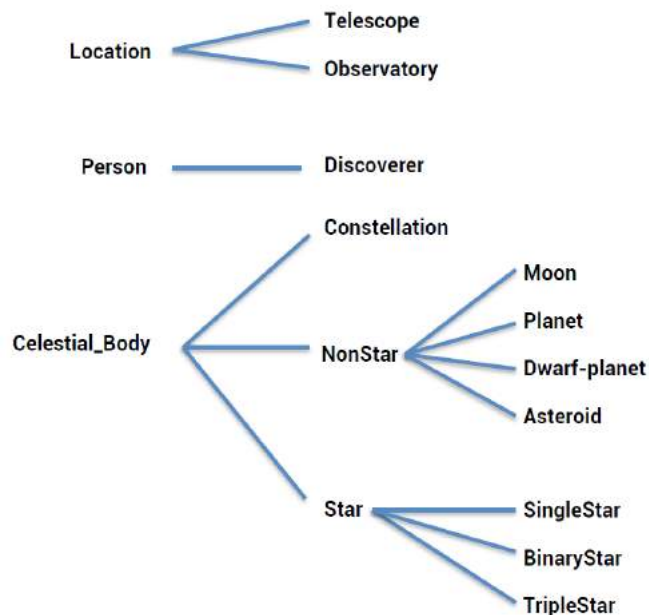
3.2 Methodology

De eerste stap die is ondernomen bij het maken van de ontology is verzinnen wat er precies in de ontology moet staan. Ons onderwerp stond al vast op de eigenschappen van verschillende hemellichamen. Aan de hand van het onderwerp zijn er drie bestaande ontologies gezocht. De eerste ontology is dbpedia, hieruit zijn de hemellichamen gehaald. De tweede ontology is linked geo data, hieruit zijn de sterrenwachten en telescopen gehaald. De derde ontology is wikidata, waar extra informatie over de sterren en planeten wordt uitgehaald.

Vervolgens is er gekeken naar hoe deze informatie onderverdeeld kan worden in classes. Aan de hand van classes zijn queries geschreven, om de informatie uit de ontologies te halen. Bij het schrijven van de queries zijn gelijk de classes en properties van de data zo gezet, dat deze overeenkomen met die in de ontology. Vervolgens zijn alle compiler errors en syntax errors uit de informatie gehaald, deze werden veroorzaakt door bijvoorbeeld "(" in de namen, waardoor de .ttl bestanden niet de juiste syntax bevatten. In de ontology is alles verder benoemd, de overige classes en subclasses zijn gemaakt en de restrictions zijn ingevuld. Zo is er bijvoorbeeld aangegeven welke eigenschappen een ster moet hebben om een ster te zijn met behulp van domain en range en kan een ontdekker alleen een persoon zijn.

3.3 Conceptualization

Op de afbeelding hiernaast is de indeling van de classes te zien. De belangrijkste class in de ontology is Celestial_Body, omdat dit de hoofdklasse is waaronder alle soorten hemellichamen met hun individuen vallen. Deze data is weer opgedeeld in de subclasses constella-



Afbeelding 8: Indeling classes

tion, nonStar en Star. Er zijn verschillende soorten sterren, daardoor kan de class Star worden opgedeeld in de subclasses singleStar, binaryStar en tripleStar. Er zijn veel verschillende soorten hemellichamen naast sterren en sterrenstelsels. De nonStar class is daarom opgedeeld in de subclasses Moon, Planet, Dwarf-Planet en Asteroid.

Er is heel veel gebruik gemaakt van properties. De properties name, minimumTemperature, maximumTemperature, volume, mass, abstract, surfaceGravity, escapeVelocity, surfaceArea, orbitalPeriod en atmosphere zijn gekoppeld aan de de individuals in de van subclasses en de class van nonStar. De properties name, abstract, temperature, mass, luminosity, gravity en radius zijn gekoppeld aan alle individuen van de subclasses en de class van Star. De property discoveredBy verbindt de instances van de class Person/subclass discoverer met de instances van de class nonStar. Hier wordt de inference gemaakt over welke persoon welke hemellichamen heeft ontdekt.

De classes Telescope en Observatory hebben de properties longitude, magnitude, position en naam. Deze classes zijn niet gekoppeld aan andere classes. De data is toegevoegd als een extra. Als een persoon enthousiast wordt van de applicatie, kan deze in de individuen van de class Location en zijn subclasses kijken om erachter te komen waar de verschillende hemellichamen van de class Celestial_Body in het echt bekeken kunnen worden.

3.4 Ontology, class restrictions en bestaande ontologies

Er worden verschillende class restrictions gebruikt in de ontology. Voorbeelden van class restrictions die zijn

gebruikt zijn:

- Een discoverer moet altijd een persoon zijn.
- Elke discoverer is alleen een discoverer als hij de eigenschap discovered heeft, die verwijst naar een instance.
- Location is disjoint with Person, een instance uit de class Location kan namelijk geen persoon zijn.
- Location is disjoint with Celestial_Body, een locatie kan namelijk geen hemellichaam zijn.
- Person is disjoint with Celestial_Body, een persoon kan namelijk geen hemellichaam zijn.

De ontology gebruikt verschillende externe bronnen. Een uitgebreide uitleg over waarom er voor deze bronnen gekozen is, is te vinden sectie 4.3. De volgende externe bronnen zijn gebruikt:

DBpedia: Er is gekozen om gebruik te maken van DBpedia omdat deze heel veel informatie bevat over sterren, planeten en andere hemellichamen. Bijna alle informatie uit de class celestial body is uit deze ontology gehaald.

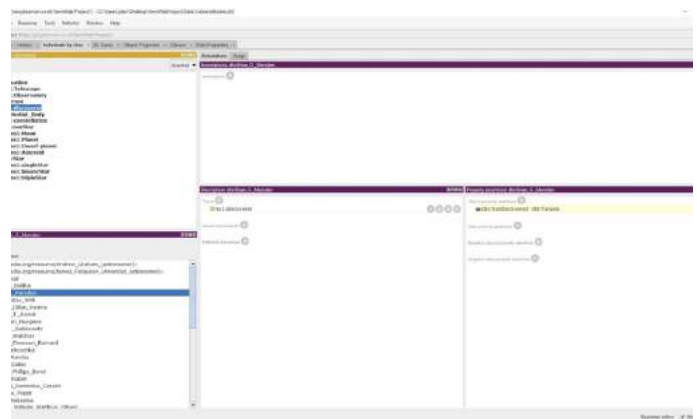
Linked Geo Data: Er is gekozen om gebruik te maken van Linked geo data om de observatories en de telescopes te vinden. Geo data beschikt over de coördinaten van deze plekken, hierdoor is het nuttig om deze informatie eruit te halen.

Wikidata: Er is gekozen voor wikidata omdat er hier veel informatie over de zon te vinden is, in andere linked data sets is dit niet zo uitgebreid gevonden. De wikidata ontology is dus vooral gebruikt om de celestial body class aan te vullen met informatie die niet in DBpedia staat.

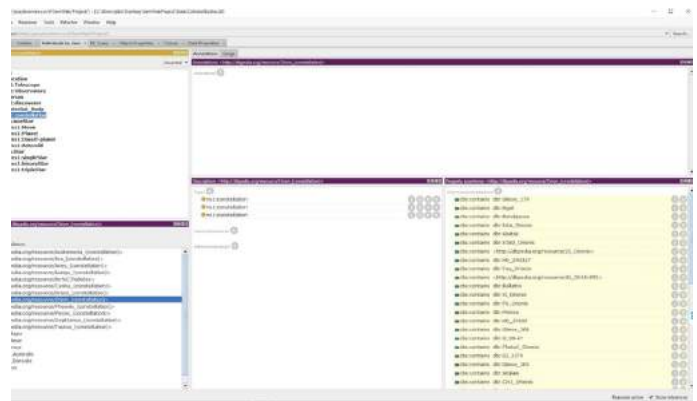
3.5 Inferences

De applicatie produceert nuttige inferenties die essentieel zijn voor de applicatie. Een voorbeeld hiervan is dat elke ster aangeeft in welk sterrenstelsel/constellation deze zich bevindt. Hierdoor komen alle sterrenstelsels automatisch onder constellation te staan. Wanneer er op een instance in constellation wordt geklikt, worden ook alle sterren die zich in dit sterrenstelsel bevinden getoond. De instances in de class constellation worden dus volledig gegenereerd uit de andere linked data.

Een ander voorbeeld van een nutte interferentie heeft te maken de class discoverer. Bij elke discoverer komt ook te staan welke hemellichamen deze heeft ontdekt. Dit komt doordat de hemellichamen de property discoverer hebben, waarin de ontdekker van de hemellichamen staat. Hiernaast zijn twee afbeeldingen toegevoegd waar te zien is dat er inferences zijn. De inferences worden in sectie 4.5 uitgebreider besproken.



Afbeelding 9: De ontology met de reasoner uit



Afbeelding 10: De ontology met de reasoner aan

4. DATA REUSE AND QUERYING

In dit hoofdstuk wordt ingegaan op welke externe bronnen er gebruikt zijn. Hierna zal worden beargumenteerd waarom er voor deze bronnen gekozen is. Ook zal er ingegaan worden op hoe de data uit deze bronnen geïntegreerd is in de ontology en applicatie. Tenslotte zullen de SPARQL queries die hierbij geholpen hebben en de verschillende inferences besproken worden.

4.1 De externe bronnen

Er is gebruik gemaakt van verschillende externe bronnen. De eerste is DBpedia. Deze bron heeft een SPARQL endpoint, namelijk <http://dbpedia.org/sparql>. DBpedia is een bron vol met informatie van Wikipedia. Deze informatie is gestructureerd, waardoor het gezien kan worden als linked data. Voor de applicatie is de data van DBpedia gebruikt die betrekking heeft tot de hemellichamen.

Wikidata is de tweede externe bron waar gebruik van is gemaakt. Deze bron heeft ook een SPARQL endpoint, namelijk <https://query.wikidata.org/sparql>. Wikidata is een open kennisbron die zowel kan worden gelezen als bewerkt door mensen en machines. Voor de applicatie is van Wikidata alleen aanvullende informatie gebruikt, die niet in DBpedia te vinden was.

Ten slotte is er ook nog gebruik gemaakt van Linked

Geo Data. Deze externe bron heeft ook een SPARQL endpoint, namelijk <https://linkedgeodata.org/sparql/>. Deze bron houdt zich bezig met een ruimtelijke dimensie toe te voegen aan het semantic web, door informatie te gebruiken van OpenStreetMap en deze om te zetten in RDF. De informatie van Linked Geo Data is gebruikt voor het gedeelte van de applicatie waar de verschillende sterrenwachten en telescopen.

4.2 Waarom deze externe bronnen?

Er is gebruikt gemaakt van DBpedia omdat dit een van de meest complete linked data sites is. Over bijna elk onderwerp is wel linked data te vinden. Dit was dan ook, samen met wikidata, een van de weinige bronnen die veel/buikbare informatie bevat over hemellichamen. Wikidata is vooral gebruikt om deze informatie aan te vullen. Wanneer DBpedia niet alle informatie over een bepaald onderdeel heeft, zoals over de zon, kan Wikidata deze informatie verlenen.

Linked GeoData was noodzakelijk voor het gedeelte van de applicatie met de sterrenwachten en telescopen. Het handige van Linked GeoData is namelijk dat er coördinaten te vinden zijn bij de verschillende items. Hierdoor zijn deze makkelijker op een kaart te plaatsen. Daarnaast heeft Linked GeoData ook het meest complete overzicht van de verschillende sterrenwachten en telescopen in de vorm van linked data.

Voor het kiezen van de externe bronnen is dus vooral gekeken naar welke informatie de bron bevat en hoe deze informatie geïntegreerd kan worden in de applicatie. Er is gelijk gekeken naar hoe hier het beste eindproduct uit kan komen.

4.3 Data integreren

De eerste stap die is ondernomen om de data te integreren in de applicatie is queries schrijven. Deze queries zijn geschreven met behulp van Yagui om de informatie uit de SPARQL endpoint te halen en dit te downloaden in RDF. Vervolgens is deze informatie in de ontworpen ontology geplaatst (meer informatie hierover is te vinden in sectie 3.2). Hier is de reasoner over gerund, waardoor ook inferences zijn ontstaan. Hierna zijn de data triplets in Stardog geplaatst. De data die ingeladen wordt in Stardog krijgt nieuwe inferred triples door de reasoner van Stardog.

De pagina's waarop de data te zien is in de applicatie, zijn gemaakt met behulp van HTML, CSS, Python, Javascript en Json. Om de data nuttig en bruikbaar te maken zijn de pagina's opgedeeld in verschillende categorieën. Het is opgedeeld in de categorieën Stars, Planets, Telescopes & Observatories en Dwarfs, Asteroids & Moons. De data wordt vanuit stardog naar de applicatie gelad-

en. Dit wordt gedaan met een combinatie van Python, Json en Javascript. Javascript stuurt namelijk een http request, met een SPARQL query, naar stardog om de data uit stardog te halen. Json zorgt er vervolgens voor dat javascript de data als tekst genereert en in het HTML plaatst. Python zorgt ervoor dat de query resultaten er op de juiste manier uit komen. Hiervoor is het tutorial.py bestand uit de opdracht van week 1 gebruikt.

Python is gebruikt voor de Stars en Dwarfs, Asteroids & Moons. Deze bestonden uit veel individuen, dus handmatig visualiseren was geen optie. Om toch een visualisatie te maken is Python gebruikt. Deze genereert een dictionary met alle namen en een willekeurig, uniek x,y coördinaat. Deze coördinaten zijn zo gegenereerd dat ze niet kunnen overlappen. Deze worden vervolgens met Canvas en Javascript getekend in het HTML. Om niet allemaal dezelfde grote sterren te hebben, zijn de grootte van de sterren random berekend in Javascript. Op deze manier lijkt het alsof er naar een sterrenhemel wordt gekeken.

Voor het onderdeel Telescopes & Observatories is gebruik gemaakt van JS map by amCharts. Deze map zorgt ervoor dat de coördinaten, die zijn opgeslagen in de triplets, kunnen worden gevisualiseerd op een kaart.

Bij de sectie voor planeten wordt er gebruik gemaakt van canvas en javascript om te visualiseren. Canvas tekent verschillende wiskunde berekende hitboxes waar vervolgens de afbeeldingen van de planeten overheen getekend worden. Elke hitbox krijgt de naam van de bijbehorende planeet mee. Als er wordt geklikt op een planeet wordt de positie van de muis uitgerekend om te bepalen welke planeet is geselecteerd. Deze vindt vervolgens met behulp van een query de juiste informatie bij de naam. Deze informatie wordt teruggestuurd naar de browser, waar javascript het in een popup plaatst met behulp van de functie innerHTML.

4.4 SPARQL queries

Om de data uit de ontology te halen zijn een aantal SPARQL queries geschreven. Op de volgende pagina is een screenshot te vinden van verschillende queries die worden gebruikt door de applicatie. Hieronder worden de verschillende voorbeelden besproken. Dit is echter maar een fractie van het aantal queries dat daadwerkelijk gebruikt is voor de applicatie.

In de eerste query is het de bedoeling dat de longitude en latitude van de telescopen eruit gehaald worden, bovendien worden de longitude en latitude gekoppeld aan de naam zodat de applicatie voortaan de naam gebruikt in plaats van de instance naam.

De tweede, derde en vierde query haalt de triplets van de manen en de asteroïden op uit stardog. De manen en

```

<textarea id='get_telescopes' class='sparql' style='display:none' rows='10'>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cbs: <http://example.com/Celestial/>
SELECT ?name ?latitude ?longitude

WHERE {
?name rdf:type cbs:Telescope.
?name cbs:latitude ?latitude.
?name cbs:longitude ?longitude.
}
</textarea>

<textarea id='get_singleStars' class='sparql' style='display:none' rows='10'>
<textarea id='get_binaryStars' class='sparql' style='display:none' rows='10'>
<textarea id='get_tripleStars' class='sparql' style='display:none' rows='10'>
<textarea id='get_asteroids' class='sparql' style='display:none' rows='10'>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cbs: <http://example.com/Celestial/>
SELECT ?OCB ?prop ?ans

WHERE {
?OCB rdf:type cbs:Asteroid.
?OCB ?prop ?ans
FILTER(?prop!=rdf:type)
}
</textarea>

<textarea id='get_moons' class='sparql' style='display:none' rows='10'>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cbs: <http://example.com/Celestial/>
SELECT ?OCB ?prop ?ans

WHERE {
?OCB rdf:type cbs:Moon.
?OCB ?prop ?ans
FILTER(?prop!=rdf:type)
}
</textarea>

<textarea id='get_sun' class='sparql' style='display:none' rows='10'>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cbs: <http://example.com/Celestial/>
SELECT ?name ?prop ?ans

WHERE {
<http://www.wikidata.org/entity/Q525> ?prop ?ans
<http://www.wikidata.org/entity/Q525> cbs:name ?name
FILTER(?prop!=rdf:type)
}

```

Afbeelding 11: De SPARQL queries

de asteroïden vormen samen met de dwergplaneten een deel van de applicatie, te vinden onder Dwarfs, Asteroids & Moons. In de eerste regel van de where statement van de query wordt aangegeven dat alle hemellichamen (OCB wordt hier gebruikt als afkorting voor other celestial bodies) van het type maan/asteroïde moeten worden opgehaald. De regel daarna geeft aan dat er van elk hemellichaam alle properties (?prop) en hun waarden (?ans) moeten worden geselecteerd. De derde en vierde query verschillen van elkaar als in dat de vierde query gebruik maakt van de data van Wikidata, en dus ander endpoint heeft.

4.5 Inferences

Uit de verschillende data van de sterren, hemellichamen en sterrenwachten/telescopen komen verschillende inferences. De eerste inference die ontstaat heeft betrekking tot de sterren en de sterrenstelsels. De data van een ster heeft een eigenschap 'constellation'. Deze geeft aan in welk sterrenstelsel de ster zich bevindt. Door deze eigenschap en de class 'constellation' aan elkaar te koppelen komen alle sterrenstelsels automatisch onder de class constellations te staan. Door deze inference komen dus eigenlijk alle sterrenstelsels van de sterren in de ontologie te staan, zonder dat deze zijn toegevoegd met behulp van een query.

The top screenshot shows a SPARQL query in Stardog:

```

1 PREFIX cbs:<http://example.com/Celestial/>
2 PREFIX viny:<http://vinctdame.nl/>
3 SELECT ?discoverer ?celestialBodies
4
5 WHERE {
6   ?discoverer rdf:type cbs:discoverer .
7   optional{?discoverer viny:hasDiscovered ?celestialBodies.}
8 }
9
10
11
12

```

The results table shows:

discoverer	celestialBodies
dbpedia:Asaph_Hall	
dbpedia:Audouin_Dollfus	
dbpedia:Brian_G_Marsden	
dbpedia:Carl_Gustav_Wirtz	
dbpedia:Charles_Dillon_Ferrine	

The bottom screenshot shows a similar query using Wikidata:

```

1 PREFIX cbs:<http://example.com/Celestial/>
2 PREFIX viny:<http://vinctdame.nl/>
3 SELECT ?discoverer ?celestialBodies
4
5 WHERE {
6   ?discoverer rdf:type cbs:discoverer .
7   optional{?discoverer viny:hasDiscovered ?celestialBodies.}
8 }
9
10
11
12

```

The results table shows:

discoverer	celestialBodies
dbpedia:Asaph_Hall	dbpedia:Dionysius_moon
dbpedia:Asaph_Hall	dbpedia:Phobos_moon
dbpedia:Audouin_Dollfus	dbpedia:Janus_moon
dbpedia:Brian_G_Marsden	dbpedia:Tartarus
dbpedia:Carl_Gustav_Wirtz	dbpedia:433_Eros

Afbeelding 12: Stardog met reasoner uit

Afbeelding 13: Stardog met de reasoner aan

Een ander voorbeeld van een inference komt voor bij de ontdekkers van de hemellichamen. Een deel van de hemellichamen heeft de eigenschap 'discoverer', waarin de ontdekker staat. Daarnaast staan alle ontdekkers van de hemellichamen in de class discoverer. Hierdoor komen alle ontdekkers van de hemellichamen (die bekend zijn in de data) automatisch in elkaar in verbinding te staan, wanneer de reasoner aanstaat. Deze inference zorgt er dus voor dat bij de ontdekkers ook de hemellichamen komen te staan die zij ontdekt hebben en andersom.

Hiernaast zijn screenshots te zien, bijgevoegd als bewijs van de inferences. Dit is aangetoond met behulp van stardog. Op de screenshots is te zien dat wanneer reasoning aanstaat de ontdekte hemellichamen verschijnen als eigenschap van de class 'discoverer'.

5. EVALUATIE

Het project was een erg leerzaam en leuk proces. De voorbereidingsfase was echter wel de meest lastige fase voor het samenwerken. We zijn lang bezig geweest met het zoeken van een onderwerp, aangezien er linked data beschikbaar is over zoveel verschillende dingen. Veel van onze ideeën hadden echter weinig bronnen in de vorm van linked data. Hierdoor zijn we uiteindelijk op het visualiseren van hemellichamen en telescopen gekomen.

Toen we eenmaal een idee hadden konden we goed doorwerken. We begonnen snel met het schrijven van queries en het in elkaar zetten van de ontology. Echter kwamen er wat tegenslagen, aangezien de data van de sterren van DBpedia afgehaald werd.. Na lang zoeken naar een oplossing is het uiteindelijk wel gelukt om deze data uit een archief van DBpedia te halen. Dit heeft ons echter wel opgehouden, aangezien de sterren een belangrijk onderdeel van de applicatie zijn.

De focus van het project lag bij ons wel vrij snel bij de applicatie in plaats van de ontology. Hierdoor moesten wij last minute nog wat aanpassingen maken aan de ontology, om te zorgen dat deze er zo goed en duidelijk mogelijk uitziet.

Tijdens het project werd er eigenlijk geen gebruik gemaakt van een taakverdeling. We werkten met z'n drieën tegelijk aan alles. Echter hadden we wel allemaal een eindverantwoordelijkheid. Die van Vincent en Jessy lag bij de applicatie en de ontology, en die van Marit bij het verslag. Dit werkte perfect.

Uiteindelijk hebben we een applicatie gemaakt waar wij trots op zijn. We vonden het vooral erg leuk om bezig te zijn met het bouwen van de applicatie. Echter hadden wij als wij meer tijd hadden gehad graag nog willen kijken naar het 3D maken van de kaarten van hemellichamen, om de experience van de applicatie beter te maken. Daarnaast hadden wij nog iets meer aandacht willen geven aan de ontology, om deze nog verder uit te breiden en nog meer informatie te kunnen laten zien.