

# EXPERIMENT 3D VISUALISATIE

STEIN JONKER  
22-09-2022

MINOR DMP

Mario de Vries

# INHOUDSOPGAVE

1	Inleiding .....	3
2	Uitvoering.....	4
2.1	2D experiment .....	4
2.2	3D .....	4
3	Resultaten.....	8
3.1	Technisch .....	8
3.1.1	Leercurve .....	8
3.1.2	Performance.....	8
3.1.3	Functionaliteiten .....	8
3.2	Gebruikersvriendelijkheid.....	8
4	Conclusies .....	10
5	Bronnen .....	11

# 1 INLEIDING

In dit document beschrijf ik het proces en de resultaten van mijn eerste experiment. In dit experiment heb ik een 3D visualisatie gemaakt van Arnhem met daarop verschillende datasets gevisualiseerd.

Dit experiment heb ik uitgevoerd omdat het uitvoeren van mijn project een aantal onzekerheden heeft die ik graag wil onderzoeken:

- 3D visualisatie is voor mij redelijk nieuw. Het uitvoeren van een experiment is een goede manier om te leren hoe verschillende 3D concepten werken.
- In het experiment wil ik gebruik maken van de Arcgis API. Dit geeft mij inzicht hoe ik er goed gebruik van kan maken en welke eventuele beperkingen er zijn.  
Ook ben ik nieuwsgierig naar de performance van een 3D applicatie in een webomgeving.
- Ik vraag mij af in welke mate een 3D visualisatie overzicht biedt

De Arcgis API is de developer toolbox om gebruik te maken van Arcgis-kaarten in een webbrowser. Arcgis is een product gemaakt door Esri. De gemeente neemt dit product af en maken allerlei verschillende kaarten met deze software.

De data gemaakt met deze software kan ingeladen worden met de API. Sommige data wordt publiek gemaakt op het [open data portaal](#) van de gemeente, waarvan ik gebruik kan maken.

In dit document zal ik eerst toelichten hoe ik het experiment heb uitgevoerd. Daarna bespreek ik de resultaten die terugkoppelen aan de bovengenoemde punten. Het document wordt afgesloten met een conclusie en een bronnenlijst.

## 2 UITVOERING

### 2.1 2D experiment

Omdat zowel Arcgis als de 3D visualisatie voor mij nieuw is heb ik ervoor gekozen om eerst het probleem kleiner te maken. Door eerst een simpele 2D kaart te ontwikkelen kon ik vrij snel met belangrijke Arcgis concepten werken: Basemaps, featurelayers. De basemap is de kaart die gebruikt wordt als ondergrond. Dit kan bijvoorbeeld een satellietbeeld zijn of een stratenkaart. De featurelaag bevat informatie om te visualiseren, bijvoorbeeld locaties van banken/prullenbakken/trolleymasten.

### 2.2 3D

Nadat het mij gelukt was om een 2D kaart te ontwikkelen ben ik begonnen om Arnhem in 3D te visualiseren. Hiervoor heb ik gebruik gemaakt van de publieke data van [3D BAG](#).

Dit is een dataset van meer dan 10 miljoen gebouwen in Nederland, in 3D.

Deze dataset staat ook gepubliceerd in Arcgis Online, het portaal waar makkelijk data gedeeld kan worden.

De 3D BAG dataset is verkrijgbaar in twee verschillende versies: WGS en RD. Dit staat voor het coördinatensysteem wat gebruikt wordt. In een kaart moeten alle lagen gebruik maken van hetzelfde coördinatensysteem en als dat niet zo is dan moet de laag zich eerst projecteren naar een ander systeem. Om het experiment simpel te houden heb ik ervoor gekozen om te kiezen voor WGS, dit systeem wordt heel erg veel gebruikt en werkt ook samen met de basemaps die Esri aanbiedt.

Om de dataset te kunnen gebruiken moet deze worden geüpload worden naar een feature/scene/tile-service. Welke service je nodig hebt is afhankelijk van hoe de data gemodelleerd is in Arcgis Pro. Een featurelayer moet worden gepubliceerd naar een feature-service en een tilelayer naar een tileservice, enzovoorts.

In het geval van de 3D gebouwen betreft het een scenelayer en daarom een scene-service. Deze hoef ik zelf niet te publiceren omdat 3D BAG dat ook al heeft gedaan. Deze service is toegankelijk via een URL. Deze service kan ik implementeren door de code op de volgende pagina:



```

// Create the layer that will hold all information about the 3D buildings
const buildings = new SceneLayer({
  url: 'https://tiles.arcgis.com/tiles/nSZVuSZjHpEZZbRo/arcgis/rest/services/BAG_3D_WGS/SceneServer',
  id: 'buildings'
})

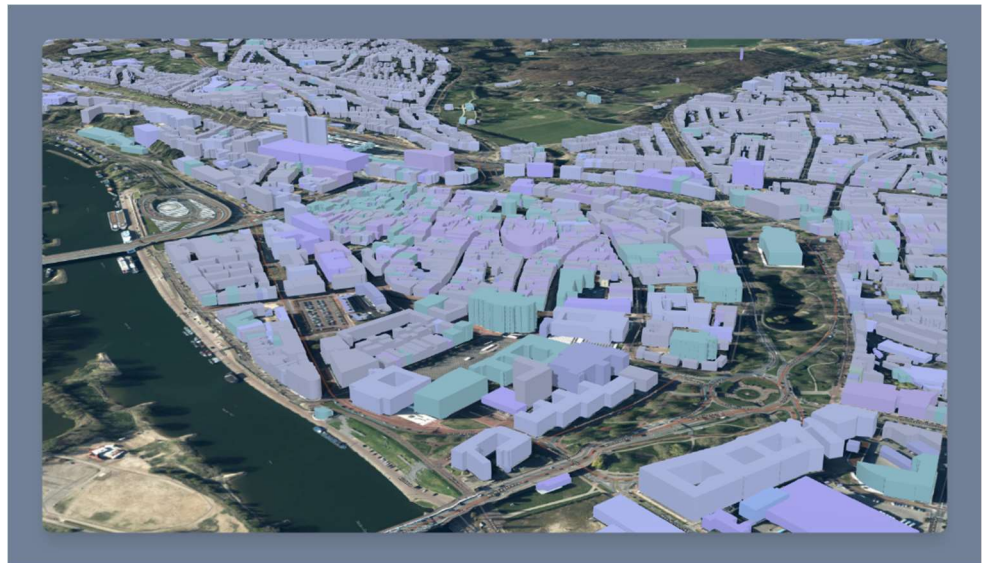
// Create the map and add the building layer
const map = new WebScene({
  // The hybrid basemap is a mix of satellite imagery and street names
  basemap: 'hybrid',
  layers: [buildings]
});

// Create a view with the map in it and show it in the HTML element with id viewDiv
const view = new SceneView({
  container: 'viewDiv',
  map: map,
});

```

*Figuur 1 Toevoegen van gebouwen aan een 3D kaart*

Deze code geeft het effect zichtbaar in Figuur 2. 3D BAG biedt ook een kaart voor windturbines en bomen, deze heb ik ook opgenomen in het experiment en die kunnen op dezelfde manier worden toegevoegd aan de kaart.



*Figuur 2 Resultaat 3D BAG Gebouwen*

De volgende stap is om wat infrastructuur van de gemeente Arnhem te visualiseren. Op het moment van schrijven heb ik nog geen toegang tot interne datasets. Daarom heb ik gebruik gemaakt van het [opendata portaal](#) van de gemeente. Het downloaden van de datasets gaat hier erg makkelijk, maar ik heb de Arcgis services nodig. Via de URL <https://geo.arnhem.nl/arcgis/rest/services/> kon ik de service URL voor elke dataset achterhalen. Ik heb gekozen om eerst banken te visualiseren, deze wordt aangeboden via de GeoJSON-service. GeoJSON is een standaard geografische objecten met JSON te encoden (Hobu Inc, et al., 2016).

Na het achterhalen van de GeoJSON-service URL was het toevoegen erg gemakkelijk, de code is te zien in Figuur 3.

```
const benches = new GeoJSONLayer({
  url: 'https://geo.arnhem.nl/arcgis/rest/services/OpenData/Banken/MapServer/0/query?outFields=*&where=1%3D1&f=geojson'
});

// Create the map, just like before but with the benches layer added
const map = new WebScene({
  basemap: 'hybrid',
  layers: [buildings, benches]
});

// Create the view and add it to HTML, just like before
```

Figuur 3 Toevoegen banken kaart

Na het toevoegen van banken heb ik op vergelijkbare manier kaartlagen toegevoegd van:

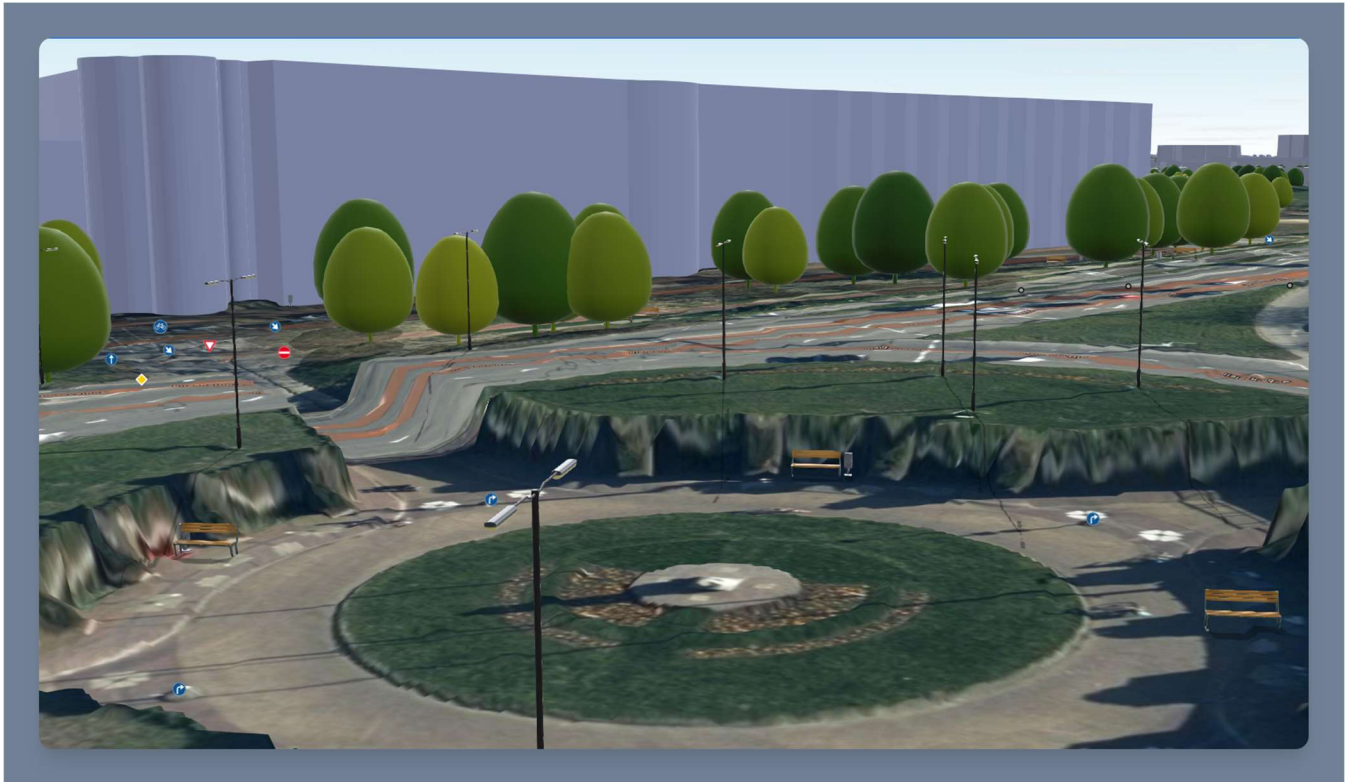
- Trolley masten
- Verkeersborden
- Afvalbakken
- Speeltoestellen
- Windturbines
- Bomen

Voor sommige van deze kaartlagen heb ik een renderer toegevoegd. Dit verandert de manier waarop een bank/ afvalbak/ speeltoestel wordt getoond. Hierdoor kun je bijvoorbeeld een 3D model van een bank of afvalbak neerzetten, in plaats van een simpele cirkel.

Het resultaat hiervan is te zien in Figuur 4. Hier is ook te zien dat de data geen gegevens bevat over rotatie. Alle objecten zijn daarom naar het noorden geroteerd en dat is problematisch voor bijvoorbeeld de twee bankjes aan de linker- en rechterzijde. Deze staan namelijk over de stoep heen gedraaid.

Daarnaast heb ik ook een 'Elevation layer' in de kaart geladen. Dit is een kaart met de hoogtes van de grond. Dit betekent dat de heuvels in het noorden van Arnhem zichtbaar zijn in 3D. Echter is er geen data bekend van tunnels of bruggen, daarom zakt de weg in Figuur 4 ineens weg.

Na het toevoegen van de hoogteinformatie wordt het ook interessant om de ElevationProfile widget te implementeren. Een demo van deze functionaliteit is te vinden [op deze website](#).



*Figuur 4 Renderers en hoogtekaart Airborneplein*

## 3 RESULTATEN

Dit hoofdstuk zal ik opsplitsen in twee onderdelen: de resultaten met een technische invalshoek en uit een gebruikersvriendelijkheid perspectief

### 3.1 Technisch

#### 3.1.1 Leercurve

Wat mij opviel in het gebruik van de Arcgis Javascript API is het gemak van gebruik. Het was even zoeken hoe het zit met feature/ scene lagen en de verschillende coördinatensystemen van verschillende kaarten, maar wanneer ik dit begreep was het programmeren van de verschillende kaartlagen heel erg gemakkelijk. Het experiment heb ik binnen twee dagen uit kunnen voeren, dit is inclusief het zoeken naar de datasets, het leren over coördinatensystemen en het leren over de verschillende soorten Arcgis services.

#### 3.1.2 Performance

Wat mij opviel in de performance van de kaart is het hoge verbruik van het computer-geheugen. Wanneer alle kaarten ingeladen neemt mijn browser 5 gigabyte RAM in. Dit is voor een browser erg veel en kan tot problemen leiden wanneer er meerdere geheugen-intensieve programma's tegelijk openstaan. Een hoger geheugenverbruik is te verwachten met een 3D applicatie, maar 5 gigabyte is veel. Ik verwacht dat het geheugenverbruik te verlagen is door de dataset voor gebouwen, bomen en windturbines te verkleinen; deze bevat nu data van heel Nederland maar alleen het gebied rond Arnhem is belangrijk.

Het navigeren over de kaart verliep soepel, hoewel het inladen van de verschillende data afhankelijk is van de internetsnelheid. Op het stads kantoor, waar mijn applicatie vooral gebruikt zal gaan worden heb ik de laadtijden niet als storend ervaren.

#### 3.1.3 Functionaliteiten

De functionaliteiten aangeboden door de Arcgis Javascript API zijn erg indrukwekkend. Er zijn veel verschillende gebruiksklare widgets zoals bijvoorbeeld de ElevationProfile, zoekbalk en een lageselectie.

Er zijn nog veel meer widgets die ik niet gebruikt heb, bijvoorbeeld de [slice](#) widget om doorsnedes te maken. Bovendien kunnen widgets ook zelf ontwikkeld worden.

### 3.2 Gebruikersvriendelijkheid

Het is mij opgevallen dat het lastig is om exact te vinden waar je op zoekt: een specifieke bank of trolley mast. Zeker als er gebruik gemaakt wordt van een renderer om een 3D bank te plaatsen, valt het overzicht weg. Met de standaard 2D cirkels die geprojecteerd worden behoud je dat overzicht wel.

Deze cirkels zijn niet altijd standaard aan te klikken, dus extra informatie wordt niet standaard getoond. Dit is wel makkelijk op te lossen, door gebruik te maken van een [popup template](#). Dit is een popup die opent op het moment dat je de cirkel aanklikt om meer informatie te geven over het object.

Ook zou het een mooie toevoeging zijn om beter te kunnen zoeken. Er kan op dit moment worden gezocht op adressen, waardoor je naar dit adres navigeert. Een mooie toevoeging lijkt mij een functionaliteit waarop je alle objecten in een radius van een locatie kan zien, ongeacht de



geselecteerde lagen. Zo voorkom je belangrijke objecten te vergeten doordat een kaartlaag is uitgeschakeld.

Een andere mogelijkheid is om de objecten toe te voegen aan de zoekbalk resultaten, net zoals adressen. Het is wordt dan wel even zoeken waar je op moet zoeken, bijvoorbeeld een ID. De vraag is of die voor iedereen al bekend is vóór het zoeken.

Wat ook opviel is dat objecten erg lastig te zien worden wanneer ze als 3D object worden weergegeven, zoals in Figuur 4. De cirkels zijn van veel verder af beter zichtbaar. Wel zouden de cirkels vervangen kunnen worden met bijvoorbeeld icoontjes, zodat het duidelijker zichtbaar is wat voor soort object de cirkel betekent.

## 4 CONCLUSIES

Het experiment in zijn algemeen is een succes, maar de applicatie vereist meer verfijning, met name in de performance en de mate van gebruikersvriendelijkheid.

De kaart gebruikt op dit moment veel geheugen, zo'n 5 gigabyte. De applicatie is soepel in navigatie en laadtijden zijn acceptabel op het stadskantoor. Het verbruik in het geheugen is hoog, het zou heel mooi zijn als deze wat verlaagd kan worden. Dit zou kunnen door sommige datasets te verkleinen om alleen data rond Arnhem te laten bevatten, in plaats van heel Nederland.

De functionaliteiten van de Arcgis Javascript API zijn makkelijk te gebruiken en bieden veel verschillende mogelijkheden. Ook staat de API toe om eigen widgets te schrijven waardoor het systeem makkelijk uit te bereiden is met op maat gemaakte functionaliteiten.

In het gebied van gebruikersvriendelijkheid valt er nog veel te winnen. Alleen de locaties van bankjes, prullenbakken en speeltoestellen is niet voldoende om het overzicht te creëren. Zeker wanneer de 3D modellen worden getoond zijn de locaties van objecten lastig te zien omdat het minder opvalt. Een cirkel of icoontje werkt veel beter.

Ook is het zoeken naar een specifiek object, of objecten in een gebied lastig: je kunt op dit moment alleen zoeken op adressen. Een verbetering zou zijn om de objecten toe te voegen in de zoekresultaten, zodat je bijvoorbeeld op het ID van een prullenbak kan zoeken.

Een functionaliteit om alle aanwezige objecten in een gebied te tonen lijkt mij ook een goede toevoeging. Het is makkelijk om objecten over het hoofd te zien wanneer er veel kaartlagen aanwezig zijn omdat je er veel uitzet. Door deze functionaliteit krijg je voor een locatie van interesse alles te zien.

Het overzicht in een 3D visualisatie is er wel maar van het overzicht-niveau naar een detail-niveau vereist nog wat werk.

---

## 5 BRONNEN

Hobu Inc, Cadcorp, Doyle, A., Butler, H., Daly, M., Gillies, A., . . . Planet Labs. (2016, August). *RFC 7946: The GeoJSON format*. Opgehaald van Rfc Editor.