

Mag Webpack zich nog steeds de koning der Javascript build tools noemen?

Vansteenkiste Maxim, Vanhaverbeke Cedric, Roobrouck Heidi

Hogeschool Gent, Valentin Vaerwyckweg 1, 9000 Gent

maxim.vansteenkiste@student.hogent.be

Abstract

Een Javascript build tool is een essentieel onderdeel bij het maken van moderne webapplicaties. Al jaren wordt Webpack gezien als de beste keuze hiervoor. Aangezien er genoeg andere opties zijn, vele met een nieuwe aanpak, is het wel eens tijd om die bewering te controleren. In een literatuurstudie wordt de geschiedenis en werking van een Javascript build tool geschetst. Daarna, aan de hand van een vergelijkende studie, worden drie populaire alternatieven tegenover Webpack geplaatst in rechtstreeks duel. Eerst wordt een nieuw project opgezet met de respectievelijke kandidaten. Daarna trachten we drie open-source projecten, elk met zijn eigen moeilijkheden, om te vormen van Webpack naar een tegenkandidaat. Doorheen dit gedocumenteerd proces, wordt er meer uitleg gegeven over de verschillende technologieën die aan bod komen. Tot slot wordt in de conclusie, rekening houdend met verschillende ob- en subjectieve factoren, de titel van deze proef beantwoord.

Introductie

De wijze waarop een webapplicatie gemaakt wordt, verandert continu. Het lijkt wel of er elke week een nieuw framework uitkomt met een iets andere aanpak dan al degene die hem voorgingen. Eén ding echter blijft al enkele jaren hetzelfde: de nood aan een build tool is er nog steeds, voor nu toch.

Dit onderzoek tracht de noodzaak, werking en toekomst van de Javascript build tool te schetsen. Dit aan de hand van een paar veel voorkomende implementaties ervan. Het uiteindelijke doel is om te bepalen of de huidige koning, Webpack, nog steeds het recht heeft om die titel op te eisen.

Experimenten

Er bestaan veel verschillende Javascript build tools. Ze allemaal vergelijken zou onbegonnen werk zijn. Gelukkig werken velen op zo goed als dezelfde manier. Vier Javascript build tools werden gekozen, vergeleken en getest in deze proef. Eerst wordt een nieuw project opgezet. Daarna vormen we drie open-source projecten die met Webpack opgezet zijn, om naar een andere kandidaat. Zo proberen we een antwoord te achterhalen op de titel van dit werk.

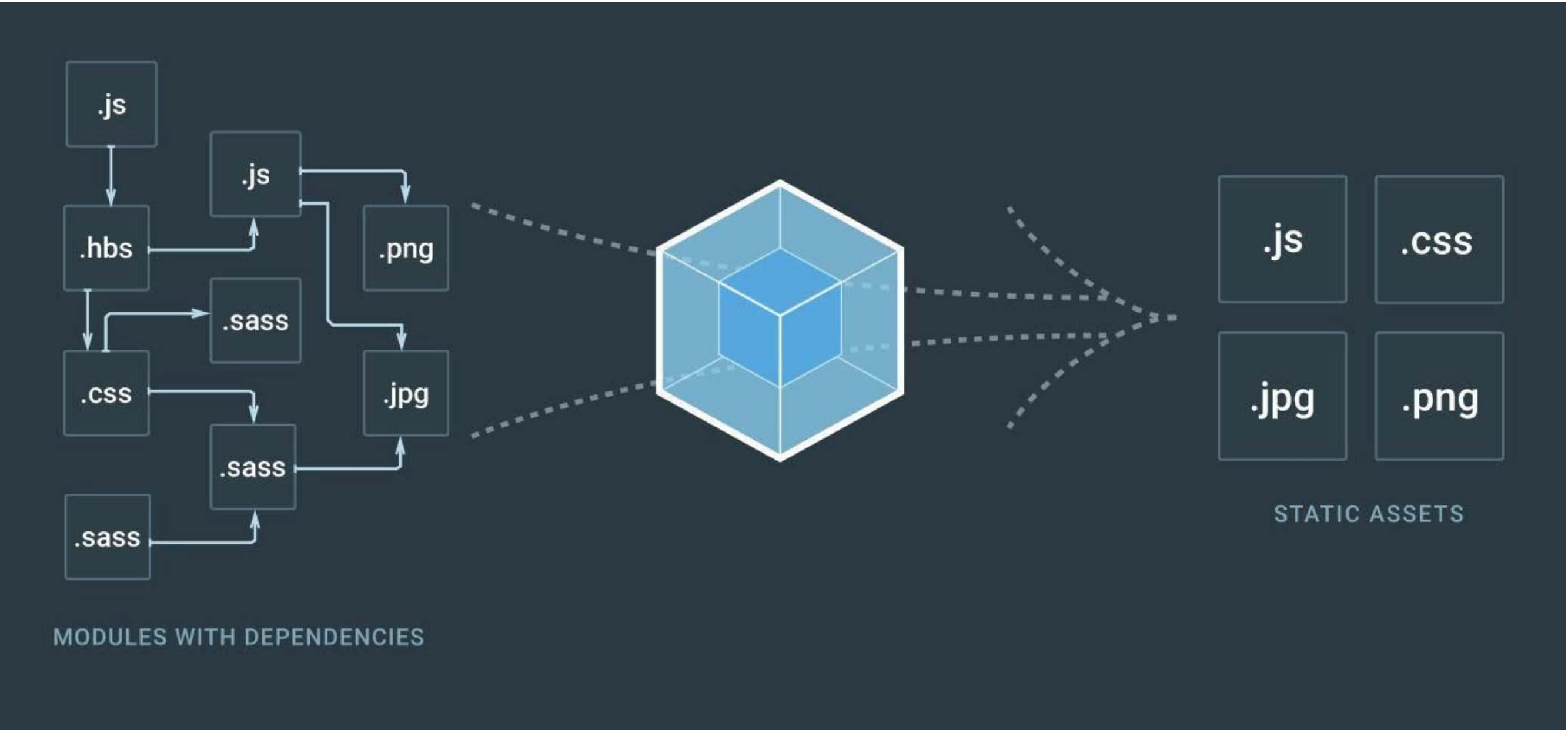


Figure 1: Essentiële functie module bundlers (Webpack, g.d.-c)

Conclusies

De hoofdvraag van deze proef, of Webpack nog steeds mag gezien worden als de koning der Javascript build tools, kan beantwoord worden met een duidelijke ‘nee’. Zelf binnen zijn eigen categorie van module bundlers, blijkt Parcel, in de meeste gevallen, een betere optie te zijn. Vite bewees dat de ongebundelde build tools de toekomst zijn. Module bundlers zijn hierdoor echter nog niet voorbijgestreefd. Zelfs wanneer browsers en Node.js ESM modules volledig ondersteunen en alle open-source packages er volledig mee geschreven zijn, zullen de voordelen van module bundlers hun plaats in de Javascript wereld garanderen. Voor de nabije toekomst zullen module bundlers en ongebundelde build tools hand in hand moeten samenwerken om het beste van beide werelden samen te brengen. Vite lijkt dit het best te begrijpen. Het is dus wel duidelijk dat Vite de beste optie is voor zowel een nieuwe webapplicatie te ontwikkelen, als een bestaand een nieuwe, snellere en modernere motor te geven.

Toekomstig onderzoek

De wereld van webapplicatie ontwikkeling staat nooit stil. Het lijkt wel of er bijna iedere week een nieuw framework uitkomt. Hoewel het dus op dit moment nog zo is dat een build tool een module bundler nodig heeft, zal dat over enkele jaren misschien niet het geval zijn. De manier hoe developers hun webapplicaties bouwen of bouwen blijft onderhevig aan veranderingen en dus zal verder onderzoek nooit veraf zijn.