

Chain logging in Go, onderzoek naar een gepaste framework en databank

Onderzoeksvoorstel Bachelorproef

Benjamin Van Iseghem¹

Samenvatting

Door de stijgende populariteit van de programmeertaal Go beginnen steeds meer bedrijven de taal te implementeren. Er worden ook steeds meer packages en interfaces beschikbaar gesteld dankzij de community die hier samen aan werkt. Een onderzoek is noodzakelijk om de meest performante combinatie van logging framework en database, in een situatie waarin een hoog aantal logs per minuut wordt geproduceerd, te vinden. Er zal een simulatie gemaakt worden waarin verschillende datanken en logging frameworks getest zullen worden op prestatie, gebruiksgemak en schaalbaarheid. Dit onderzoek biedt een meerwaarde voor bedrijven die een hoge dataflow willen opslaan. De verwachting van dit onderzoek is de juiste combinatie vinden voor deze situatie.

Sleutelwoorden

Onderzoeksdomein. Databanken en big data — SQL vs NoSQL— Go

Co-promotor

Jens De Valck² (Be-Mobile)

Contact: ¹ benjamin.vaniseghem.w3891@student.hogent.be; ² jens.de.valck@be-mobile.com;

Inhoudsopgave

1	Introductie	1
2	State-of-the-art	1
3	Methodologie	1
4	Verwachte resultaten	2
5	Verwachte conclusies	2

1. Introductie

Be-Mobile houdt zich bezig met het verwerken van traffic data. Hierbij wordt gebruik gemaakt van een pipeline van verschillende microservices en API's die alle binnenkomende data verwerken. Om het proces dat de data doorloopt duidelijker te maken en om eventuele problemen in de toekomst makkelijker op te sporen, zal er overal gepaste logging aan toegevoegd worden. Real time worden er dus veel logs per seconde gegenereerd. Daarom is het belangrijk om de juiste logging framework en databank te implementeren. Het werk beschrijft dus een databankonderzoek. Meer bepaald het op zoek gaan naar de meest geschikte combinatie van een logging framework en databank om een grote, constante flow van logs te centraliseren en op te slaan. Welke databank kan het best dit soort data opslaan. Welk logging framework is het meest geschikt in deze omgeving?

Elke situatie in verband met data is anders. Het is daarom belangrijk eerst eigen onderzoek te verrichten in plaats van zomaar een databank te implementeren omdat deze in een

andere situatie optimaal was.

2. State-of-the-art

Een eerste reden voor dit onderzoek is de populariteit van deze taal. Rouse (2017) stelt dat de groei hiervan te wijten is aan het feit dat Go een lichtgewicht, open source taal is die past in de microservices architecturen van vandaag. De programmeertaal Go is gereleased in 2012 en werd vooral gebruikt door Google (Golang, 2018). De taal zelf heeft een grote community achter zich gekregen en blijft in populariteit groeien. Een andere reden voor dit onderzoek is het gebrek aan soortgelijke onderzoeken. Ongetwijfeld zijn er al onderzoeken zoals deze uitgevoerd, maar niet met de bedoeling om deze te publiceren, binnen een bedrijf bijvoorbeeld. Vandaar dat deze dus niet beschikbaar zijn online. Garcin (2017) stelt dat Go reeds veel packages en interfaces heeft die gebruikt kunnen worden in database koppeling en logging. Aan opties dus geen gebrek. Daarom is het interessant om dit onderzoek uit te voeren.

3. Methodologie

Het onderzoek start met het onderzoeken welk soort databank het meest geschikt is voor het opslaan van log bestanden, een relationele of NoSQL databank. Eens een beslissing hierover gemaakt is, kan dieper worden gezocht naar welke van de gekozen soort databanken het beste passen bij deze soort data. Nadien zal ook een lijst van mogelijke frameworks gemaakt

worden die instaan voor het produceren van logs. De volgende stap in het proces is het maken van een webservice aan de hand van een vue.js frontend en Go backend dat op hoog tempo logs produceert. Dit om de echte omgeving te simuleren. Daarna zal elke databank om beurt getest worden in deze webservice om uit te maken welke het meest performant is in deze omgeving. De logging frameworks kunnen in deze webservice ook getest worden. De resultaten zullen dan statistisch verwerkt worden om een conclusie te maken.

4. Verwachte resultaten

Over resultaten kunnen er nog geen voorspellingen gemaakt worden, maar na een eerste literatuurstudie kan er verwacht worden dat een NoSQL databank het best zal fungeren in een omgeving zoals deze. De resultaten die geproduceerd zullen worden door de simulatie zullen over het algemeen niet erg verschillen tussen de verschillende databanken.

5. Verwachte conclusies

De combinatie van databank en logging framework is belangrijk en zal een grote rol spelen doorheen dit onderzoek. Het zal niet voldoende zijn om de meest performante databank uit te zoeken. Deze zal ook goed moeten kunnen samenwerken met de gekozen framework. Dit kan een extra moeilijkheid vormen doorheen het onderzoek.

Referenties

- Garcin, P. (2017). Surveying The Go Database Landscape. Verkregen van <https://www.activestate.com/blog/surveying-go-database-landscape/>
- Golang. (2018). The Go Blog. Verkregen van <https://blog.golang.org>
- Rouse, J. (2017). Why Go is skyrocketing in popularity. Verkregen van <https://opensource.com/article/17/11/why-go-grows>