

# Welke technologie is het meest geschikt om in een IoT-applicatie data van microservices te gebruiken?

Desmet Ruben, Meersseman Maarten, Vandermeersch Sonia  
Hogeschool Gent, Valentin Vaerwyckweg 1, 9000 Gent

ruben.desmet.y7611@student.hogent.be

**Abstract**  
It's only a model. You don't vote for kings. Who's that then? We found them. Ni! Ni! Ni! Ni! The nose? On second thoughts, let's not go there. It is a silly place. Bloody Peasant! And the hat. She's a witch! Where'd you get the coconuts?

## Introductie

Tijdens dat ik dit onderzoek voerde, liep ik ook mijn stage in het bedrijf TVH. Al snel dit academiejaar kwam ik te weten dat ik hier mijn eerste werkervaring mocht opdoen. Toen er gevraagd werd om een onderwerp te zoeken voor deze bachelorproef, leek het mij handig om een onderzoek te voeren die interessant was voor TVH. Na een brainstorm-sessie met mijn stagementor en de product-owner van mijn stage kwam ik tot dit onderwerp.

Eerst wist ik nog helemaal niets over dit onderwerp. Gezien ik op mijn stage ook met een event-bus werk, was dit een ideaal moment om zaken bij te leren. Ons team van de stage gebruikt die van Google, namelijk Google Pub/Sub. Andere teams gebruiken Kafka en ik vernam dat vroeger ook nog RabbitMq gebruikt werd. Het leek mij ideaal om te onderzoeken welke technologie nu eigenlijk beter is.

## Experimenten

Dit onderzoek vergelijkt drie verschillende technologieën met elkaar. Deze zijn Kafka, Google Pub/Sub en RabbitMq. Om te testen welke van de technologieën die in dit onderzoek aan bod komen nu eigenlijk het beste is, zou je met veel zaken rekening moeten houden. Wat het beste is hangt vooral af van welke soort data je gebruikt. Ook de hoeveelheid transformaties die je data ondergaat voor dat ze uitgelezen worden of verzonden speelt een grote rol. Natuurlijk is het moeilijk om binnen de tijdspanne die er is voor dit onderzoek al deze factoren te gaan onderzoeken. Daarom spitst dit onderzoek zich toe op een bepaald scenario om daar conclusies uit te trekken. Voor het scenario werd er op voorhand beslist voor welk formaat van data we dit onderzoek voeren, dit zal een Json zijn. Ook zullen we het onderzoek drie keer uitvoeren, voor verschillende groottes van data. Een keer met 10 000, 100 000 en 1 000 000. Telkens wordt er gekeken hoeveel tijd er nodig is om een data object te verzenden en te ontvangen. Hierna word er gekeken naar het gebruik van memory. Hierbij wordt er geen rekening gehouden met de snelheid omdat de code die de memory ophaalt veel te veel tijd in beslag neemt.

Voor dit onderzoek zijn er twee grote applicaties nodig. Een producer en een consumer applicatie. De producer zorgt ervoor dat er per technologie een implementatie is waardoor een technologie data kan verzenden. In dit onderzoek zal dit voor Kafka, RabbitMq en Google Pub/Sub gebeuren. Elke keer deze applicatie opgestart wordt, zal er meegegeven worden voor welke technologie er data moet verzonden worden. Voor de tweede applicatie is er ook een implementatie nodig per technologie die ervoor zal zorgen dat er geluisterd wordt naar de topic/queue indien er nieuwe messages zijn. Deze verschillende consumers zullen tegelijkertijd luisteren naar hun eigen topic/queue. Deze applicatie moet dus niet apart opgestart worden.



Figure 1: Logo Kafka



Figure 2: Logo Google Pub/Sub



Figure 3: Logo RabbitMq

## Conclusies

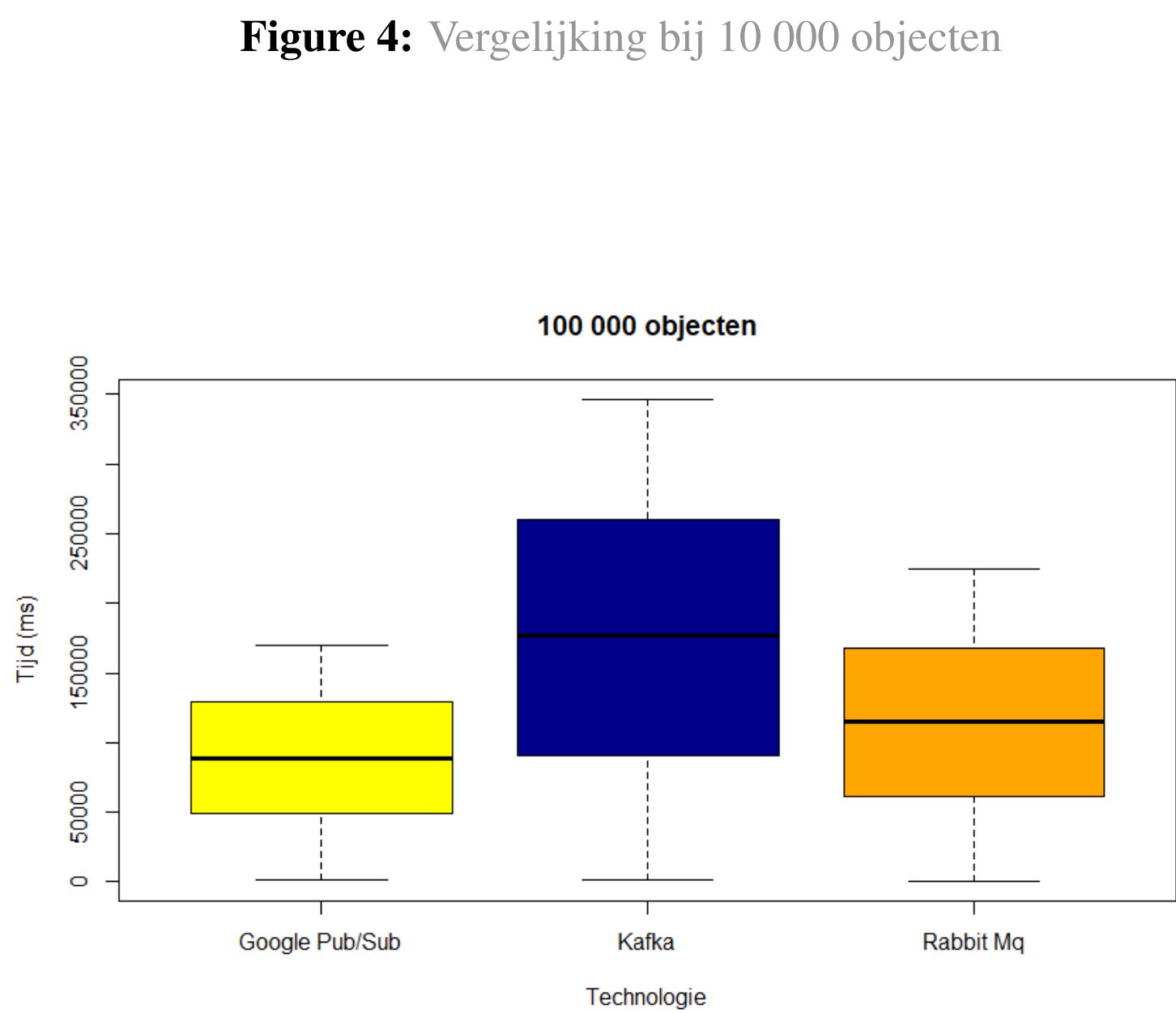
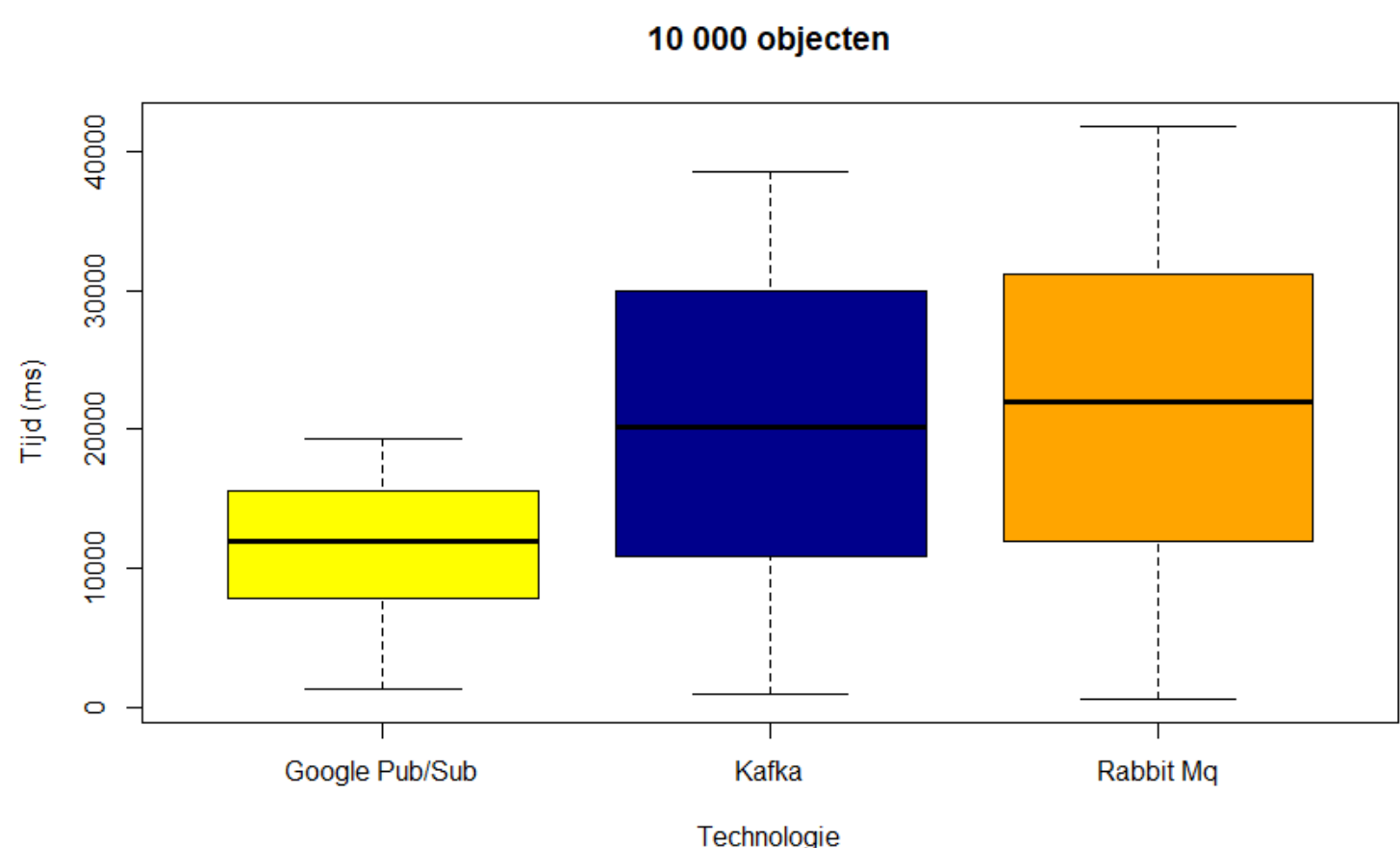


Figure 5: Vergelijking bij 100 000 objecten

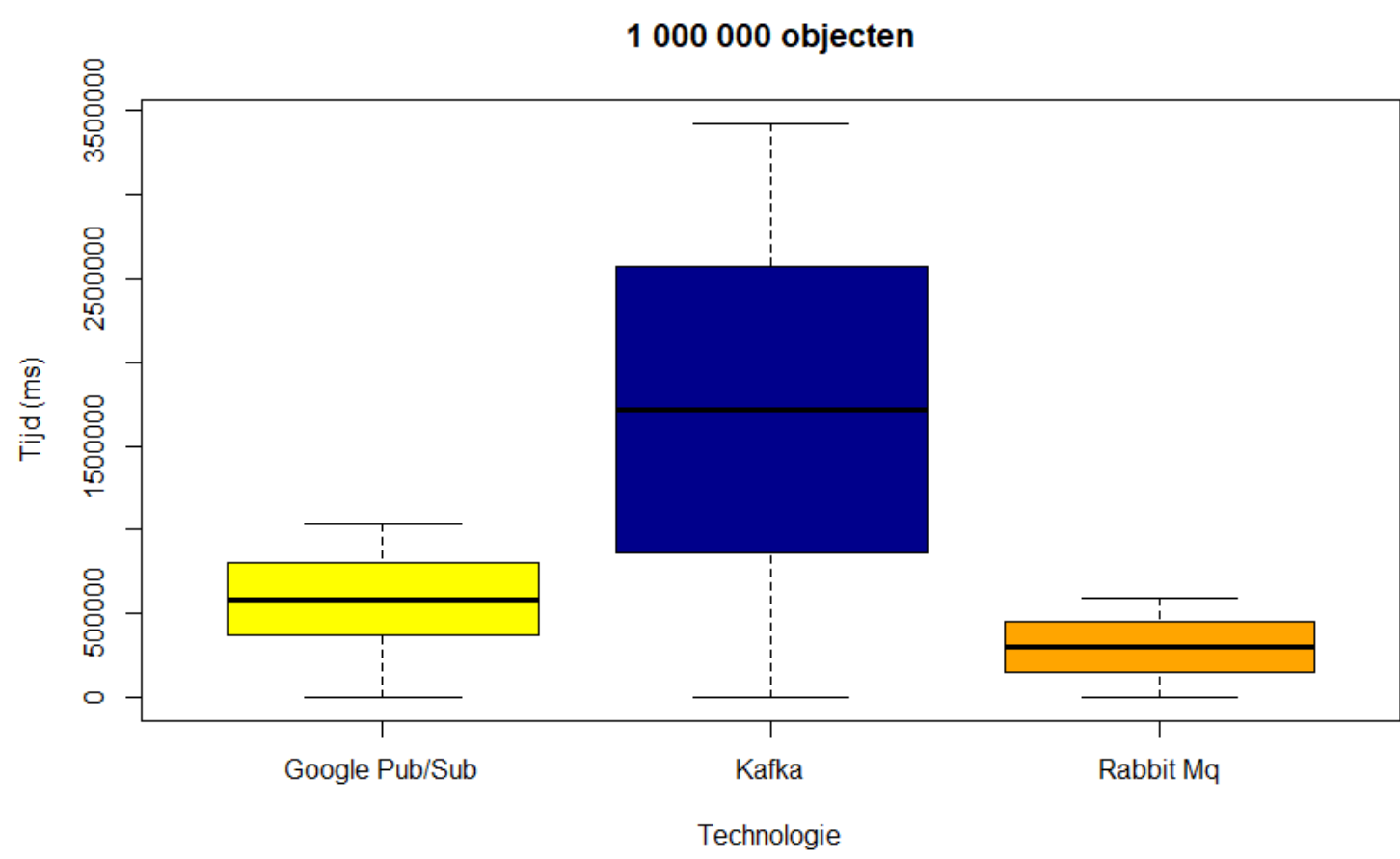


Figure 6: Vergelijking bij 1 000 000 objecten

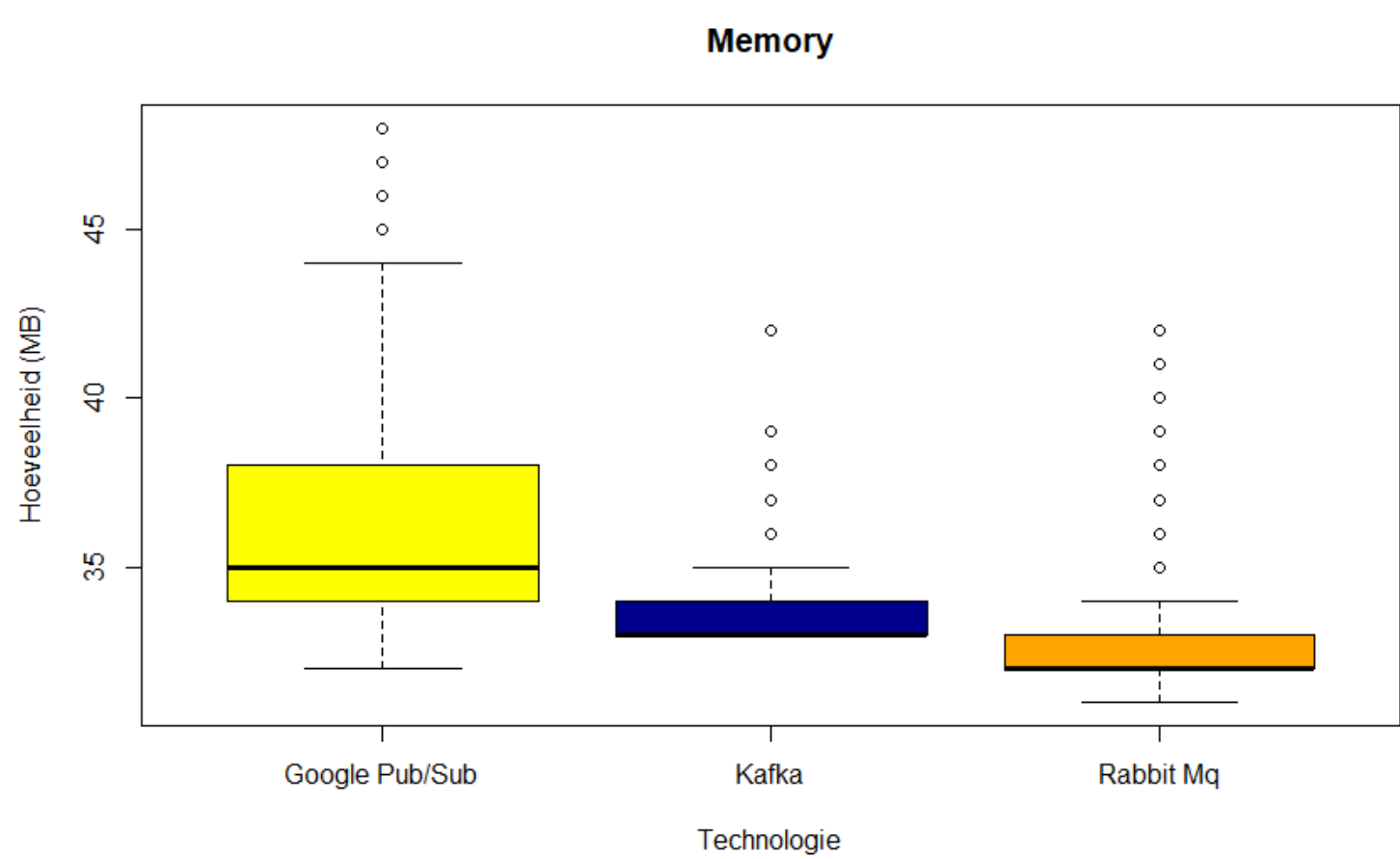


Figure 7: Vergelijking geheugen

Wanneer snelheid voor een bedrijf het belangrijkste is, dan lijkt Google Pub/Sub het beste. Tenzij je heel veel objecten hebt (meer dan 1 000 000) dan lijkt RabbitMq sneller te werken. Het geheugengebruik is het best bij RabbitMq maar Kafka doet het niet veel slechter. Bij Kafka ben je het meest zeker dat je effectief al je verzonden data meteen ontvangt. Hier slaagt RabbitMq het slechtste op. Voor TVH specifiek lijkt het belangrijkste dat alle data zeker meteen ontvangen wordt. Omdat Kafka hierop het beste scoort en bijna de beste is op basis van geheugengebruik, lijkt Kafka de beste technologie om te gebruiken binnen dit bedrijf.

## Toekomstig onderzoek

Als verder onderzoek zou je dit onderzoek meerdere keren kunnen uitvoeren om te bepalen hoe frequent verzonden data niet meteen aankomt. Een andere mogelijkheid is om de applicaties niet lokaal maar in de cloud te laten draaien. Op die manier ben je niet afhankelijk van een pc of laptop.