

Microservice Integration Patterns op SAP order-to-cash proces

Van Damme Lyva, Nicolas Pauwelyn, Karine Samyn

Hogeschool Gent, Valentin Vaerwyckweg 1, 9000 Gent

lyva.vandamme.y7102@student.hogent.be

Abstract

Dit onderwerp, hoe microservices invloed hebben op het SAP order-to-cash proces, werd voorgesteld door delaware. Dit werd gekozen omdat deze nieuwe technology een interessante invloed kan hebben op de order-to-cash proces. Dit is een manier om een proces robuuster te maken. In de meeste software wordt gebruik gemaakt van één grote databank of meerdere databanken die in staan zijn om meerde services te voorzien van data. Bij microservices wordt voor elke service een aparte databank opgesteld. Dit is maar een klein deeltje van een microservice. De microservice moet voldoen aan business requirements. SAP zelf heeft ook al veel ondernomen omtrend microservices. Eén van hun oplossingen is Kyma. Maar de belangrijkste vraag is namelijk: Hoe microservice integration patterns een SAP order-to-cash beïnvloedt.

Introductie

0.1 Waarom microservices?

De architectuur maken aan de hand van microservice patroon of aan de hand van monolithic? Een vraag die wel vaker gesteld wordt. Afstappen van het iets dat we kennen? Een sprong in het donkere, het onbekende. Een microservice architecture brengt voordelen en nadelen met zich mee, net zoals monolithic. Maar waarom kiezen we de microservice techniek boven die van monolithic? En welke passen heeft SAP al genomen om hun klanten de mogelijkheid van microservices aan te bieden?

0.2 Waarom order-to-cash proces?

Het order-to-cash proces is een veel gebruikt proces binnen bedrijven. Dit proces heeft veel invloed op het succes van een bedrijf. Er kan aangetoond worden hoe het proces beïnvloedt wordt door microservices. Welke obstakels er waren bij het uitdenken van de architectuur. Waar waren er problemen, waar moest er meer nagedacht worden over een oplossing? Een order-to-cash proces wordt over verschillende afdelingen gebruikt. Dit is een reden om dit proces uit te kiezen. Het proces bevat onderdelen die verschillende doelen hebben.

Experimenten

Deze bachelorproef was een theoretisch onderzoek naar de invloed van microservices op een order-to-cash proces. Er werd onderzoek uitgevoerd naar op welke manier er moest worden overgeschakeld worden. Welke communicatie methode er kan gebruikt worden tussen de microservices. Het volledige proces werd opgesplitst. Er werd nagegaan welke requirements elk onderdeel van het proces had. Er werd een uitgebreide literatuurstudie gedaan over microservices. Dit is een breed onderwerp. Er is diep ingegaan op bepaalde eigenschappen van microservices. De term zelf is heel breed.

Sectie met figuur

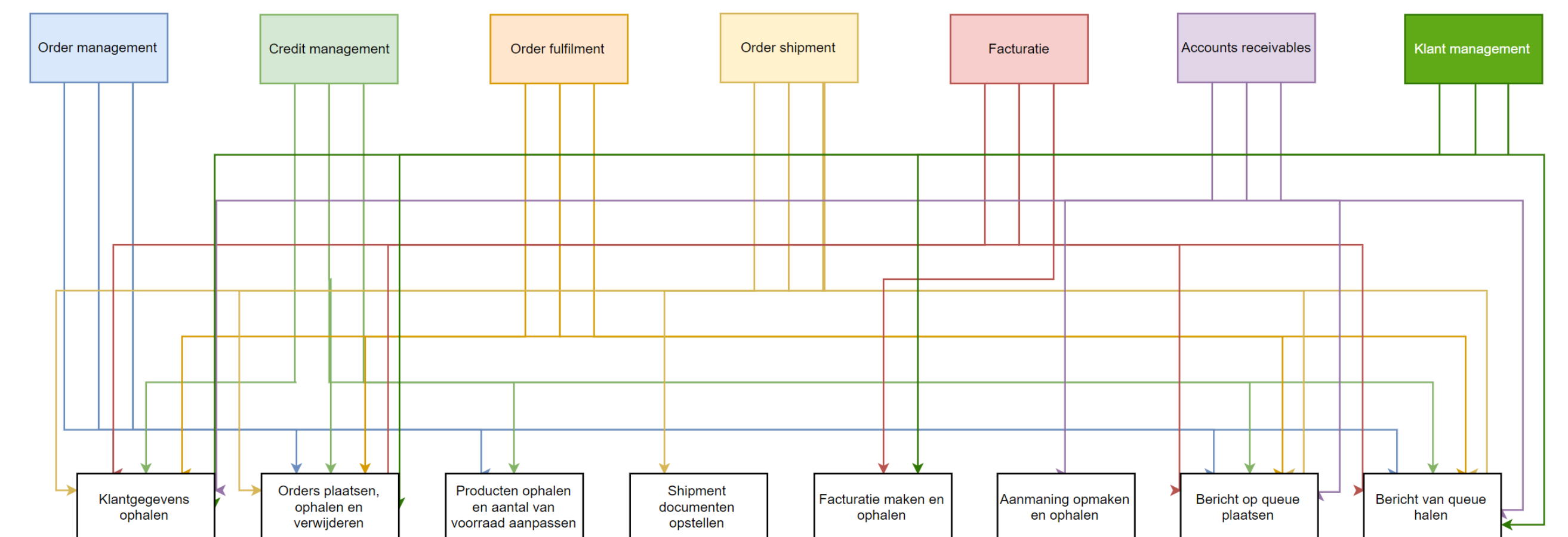


Figure 1: Het order-to-cash proces waarvan elk onderdeel zijn nodige microservices aanspreekt.

Conclusies

Microservice is een architectuur en ideologie waarin logica en de requirements, die te vinden zijn bij een monolithic, terugkomen. Dit heeft invloed op de overschakeling naar microservices. Deze manier van implementatie moet aangepast worden bij het gebruik van microservices. Dat kan een obstakel zijn. Er wordt een andere manier van denken en organiseren gevraagd binnen een de IT-afdeling. De teams worden hervormd. Een team bevat niet meer de kennis van de volledige architectuur, enkel over de microservice waaraan zij werken.

De performance zal stabiel zijn bij pieken van orders. Omdat de microservices onafhankelijk zijn van elkaar. De onderdelen van het OTC proces kunnen onafhankelijker functioneren. Valt er een onderdeel uit of tijdelijk niet beschikbaar dan heeft dat geen grote gevolgen voor het proces in het algemeen. Het onderdeel ervoor kan nog altijd data op de queue plaatsen. Maar de onderdelen na het onbeschikbare deel, zal wel wat hinder ondervinden. Maar niet zodat het gehele proces plat zal liggen.

Toekomstig onderzoek

In de toekomst kan er een proof of concept gemaakt worden. De verschillende communicatie methodes kunnen verder onderzocht worden. De manier om authenticatie en autorisatie toe te passen, kunnen nog dieper worden onderzocht. Deze studie probeerde zo veel mogelijk eigenschappen van microservices aan te halen. Dit zorgt ervoor dat er niet op elk deel diep kan worden ingegaan. In de toekomst kan elk deel van microservices dieper worden onderzocht en uitgetest.