



Universität St.Gallen

EDPO-Group 4

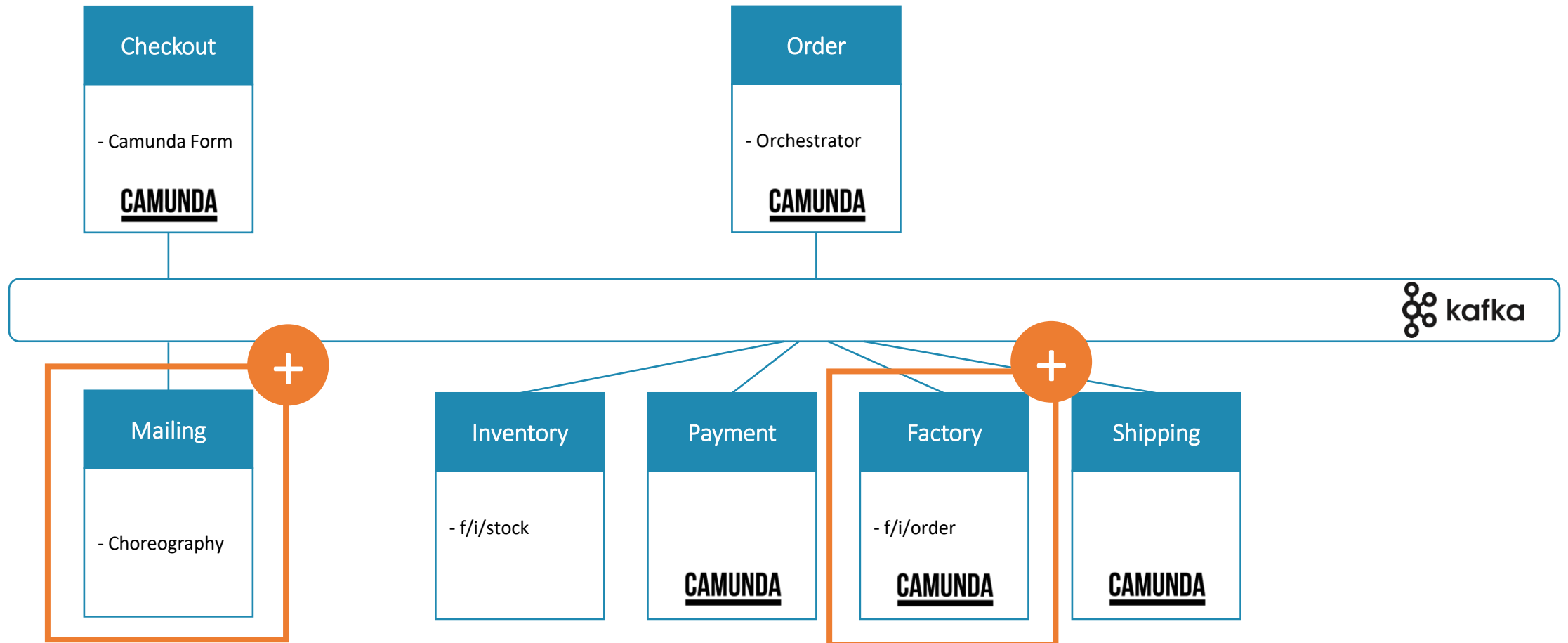
Assignment 1

Yasmin Lützel Schwab

Stefan Meier

18.04.2024

General Project Description



Camunda Form to initiate the Checkout process

Start process: chekout-flow

Customer Name*

Customer Address*

Article #1*

the thing you want

Amount #1*

how many you want

Article ID #2

another thing you want

Amount #2

how many of the other thing you want

Article ID #3

another thing you want

Amount #3

how many of the other thing you want

Submit

Back

Close

Start

Orchestration vs. Choreography



Checkout Service

→ Order Service is choreography



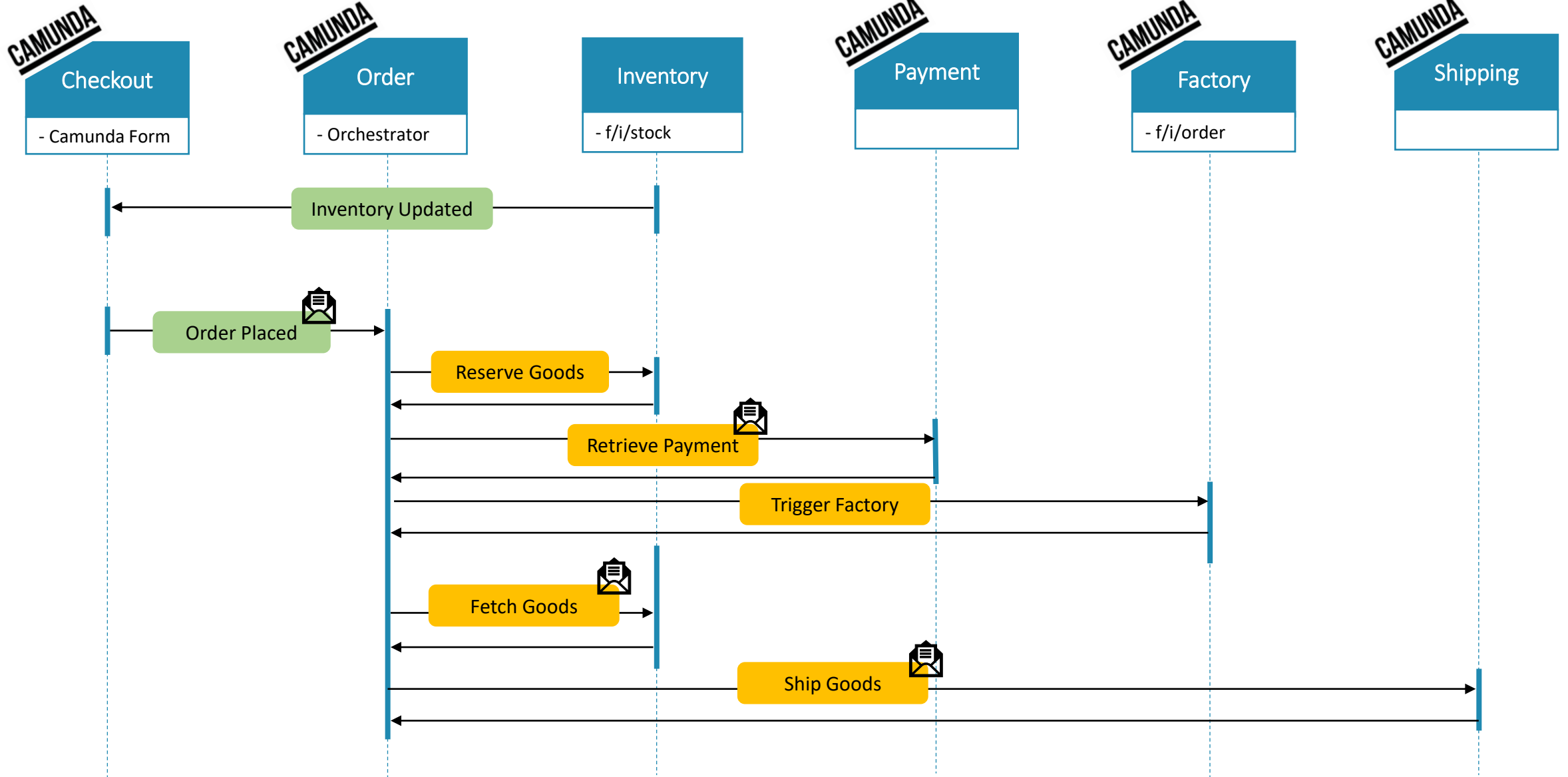
Once Order placed

→ Orchestrated by Order-Service



Any Service

→ Mailing Service is choreography



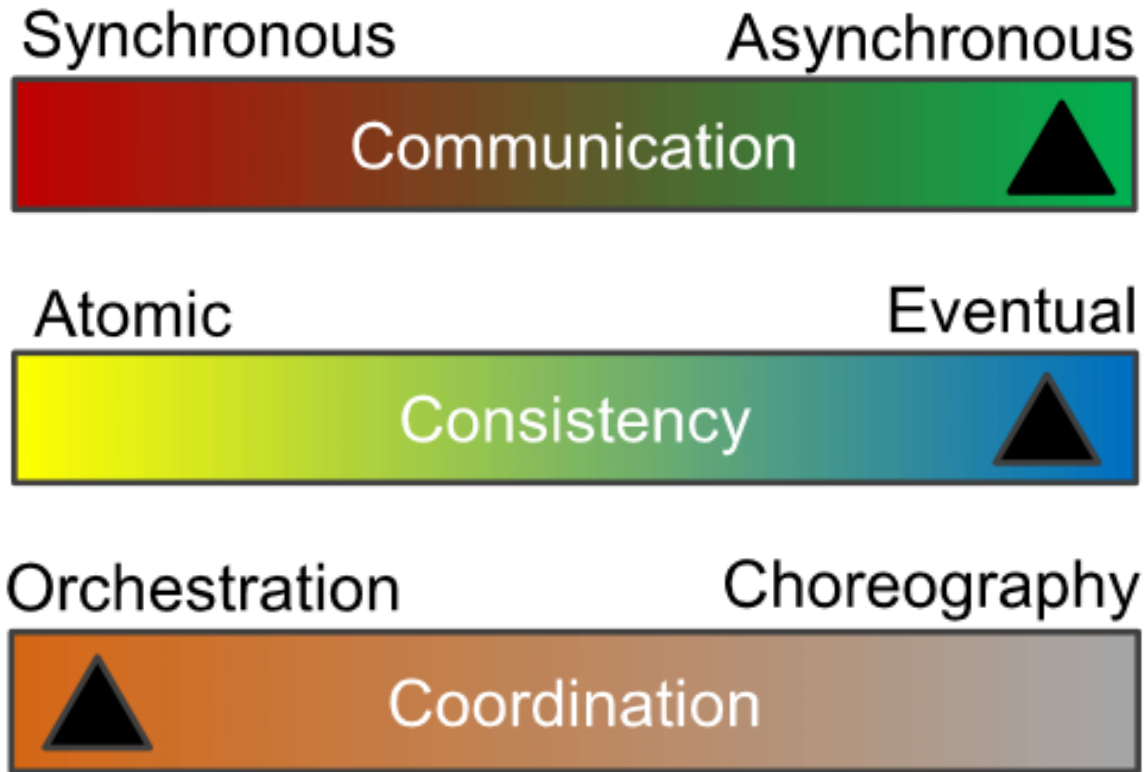
Event

Command

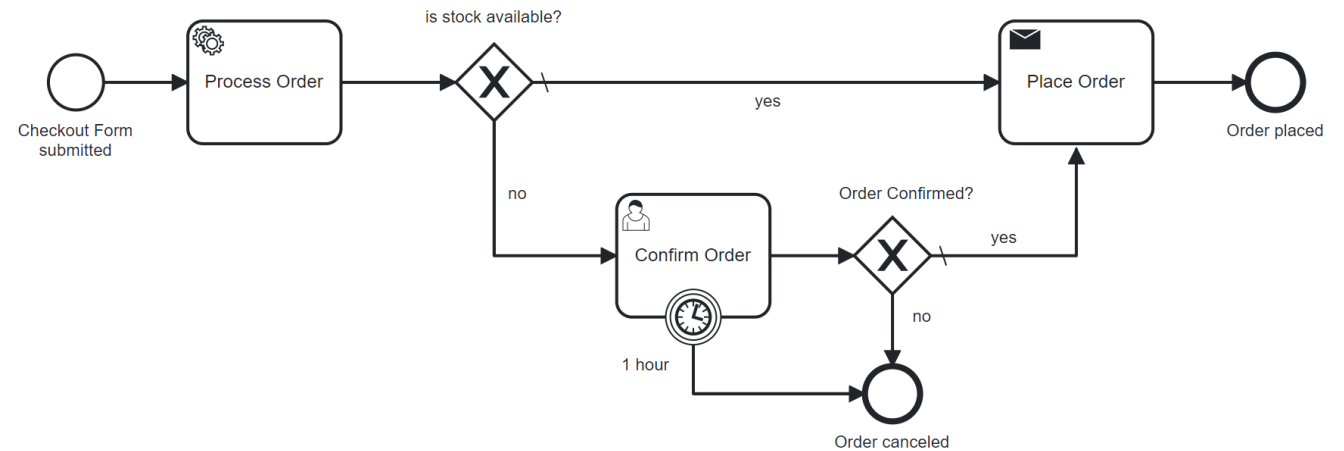
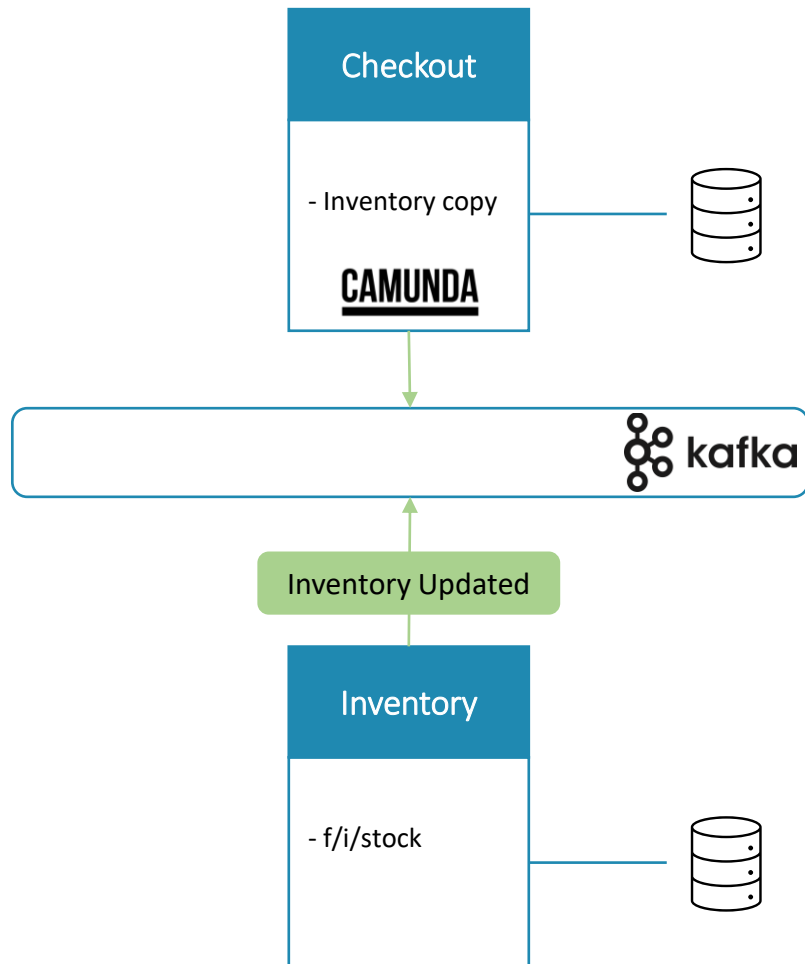


Mailing: send status Mail

Parallel Saga



Event-carried State Transfer



Decoupling, Reduced load on inventory



Replicated Data → eventual consistency

Event: OrderPlacedEvent (from Checkout)

Command: RetrievePaymentCommand (from Order-Camunda)

Event: CheckAvailableStockEvent (from Order-Camunda)

Event: CustomerCreditUsedEvent (from Payment-Camunda)

Event: PaymentReceivedEvent (from Payment-Camunda)

Event: GoodsAvailableEvent (from Inventory)

Event: GoodsAvailableEvent (from Inventory)

Event: GoodsAvailableEvent (from Inventory)

Event: GoodsAvailableEvent (from Inventory)

Event: GoodsAvailableEvent (from Inventory)

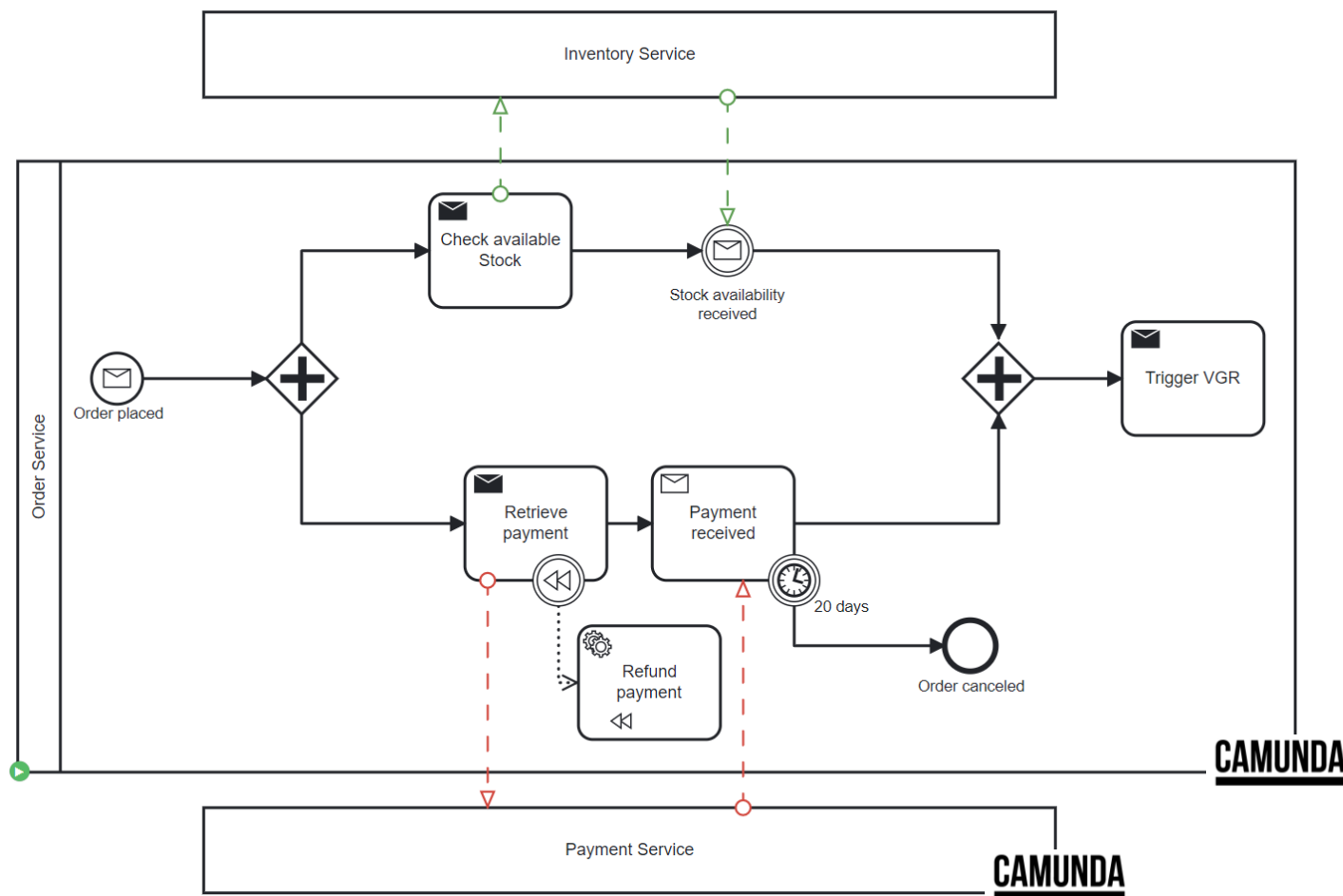
Event: GoodsAvailableEvent (from Inventory)

Event: GoodsAvailableEvent (from Inventory)

Event: GoodsAvailableEvent (from Inventory)

Event: GoodsAvailableEvent (from Inventory)

Event: GoodsAvailableEvent (from Inventory)



Parallel Gateway, check the inventory while payment is retrieved



Message duplication

Event flow

Event: OrderPlacedEvent (from Checkout)

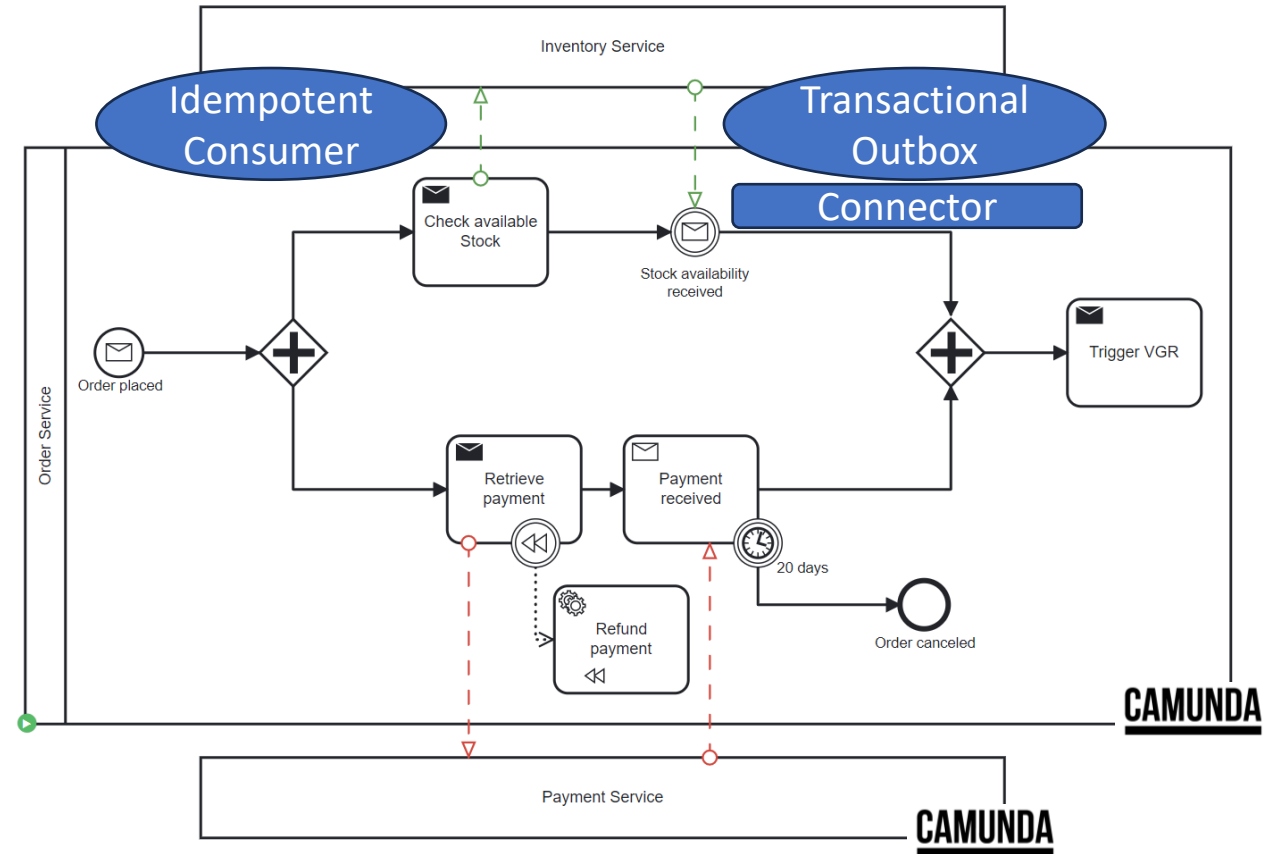
Command: RetrievePaymentCommand (from Order-Camunda)

Event: CheckAvailableStockEvent (from Order-Camunda)

Event: CustomerCreditUsedEvent (from Payment-Camunda)

Event: PaymentReceivedEvent (from Payment-Camunda)

Event: GoodsAvailableEvent (from Inventory)



Received CheckAvailableStockEvent

MessageListener: Check if stock available

Added to outbox: Message [type=GoodsAvailableEvent,

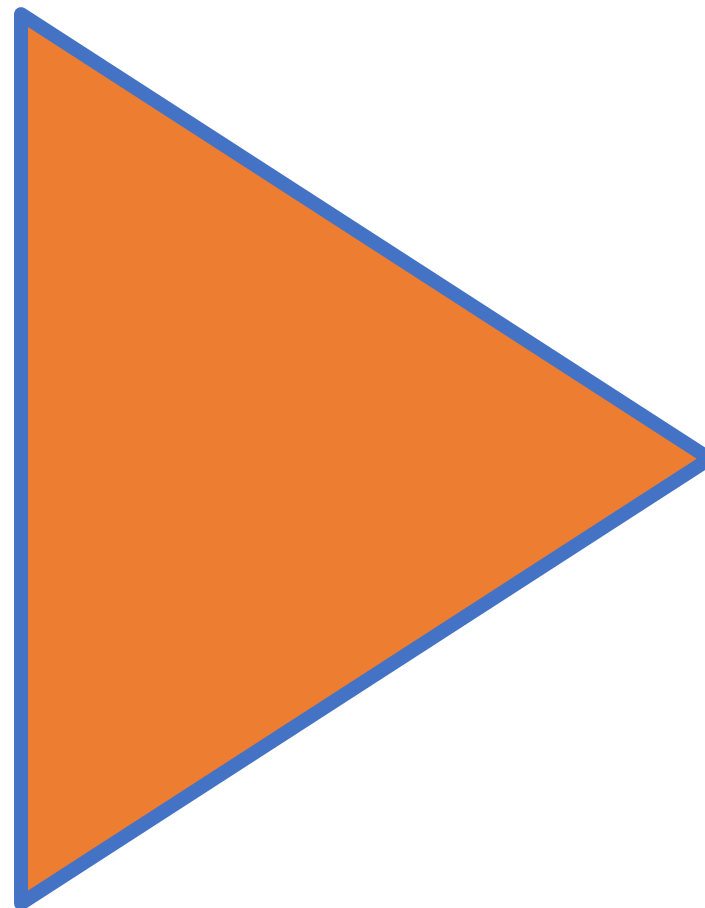
Outbox Poller: Successfully sent message from outbox

Received CheckAvailableStockEvent

Duplicate CheckAvailableStockEvent detected for refId:., skipping processing.

❌ Message duplication
Inventory does not use Camunda

DEMO



Q & A

