

# KafMonGoSQL

## *Team 1*

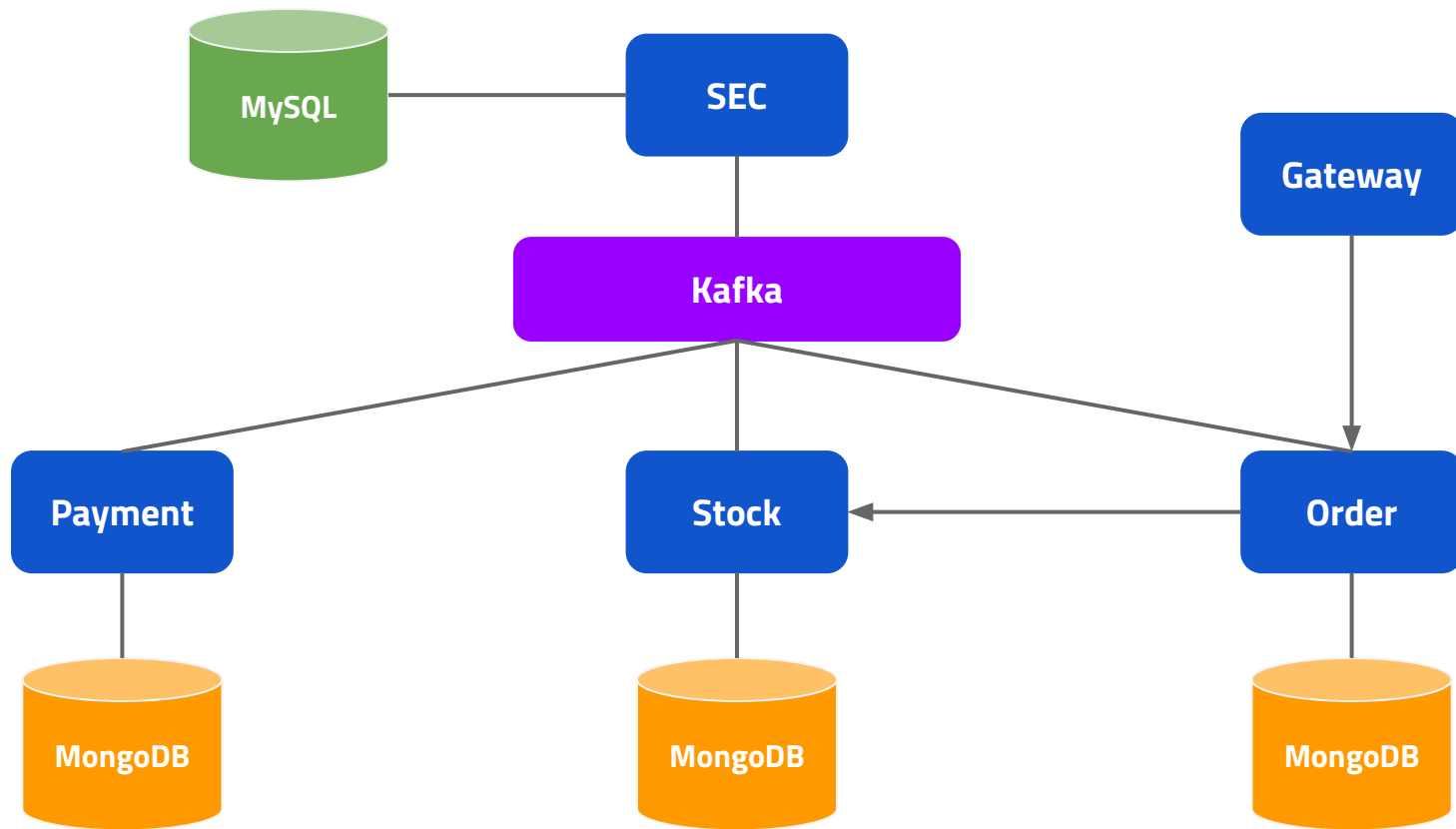
Rahul Kochar, Gijs Paardekoooper, Deepali Prabhu, Tobias van den Hurk



# Technologies

- **Go** - taking advantage of its parallel computing purpose
- **Kafka** - used for managing SAGAs and microservice messaging
- **MongoDB** - database used by Order, Stock and Payment
- **MySQL** - database used to recover SAGA logs

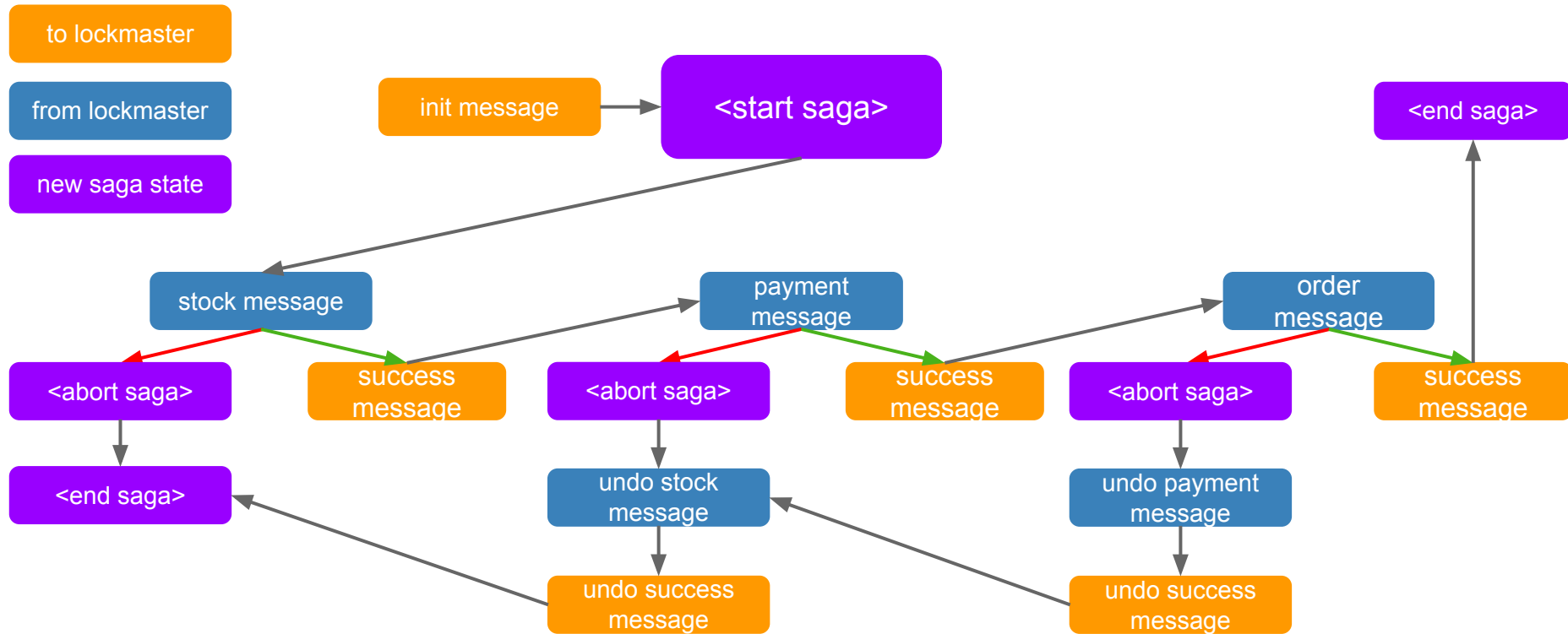
# General Structure



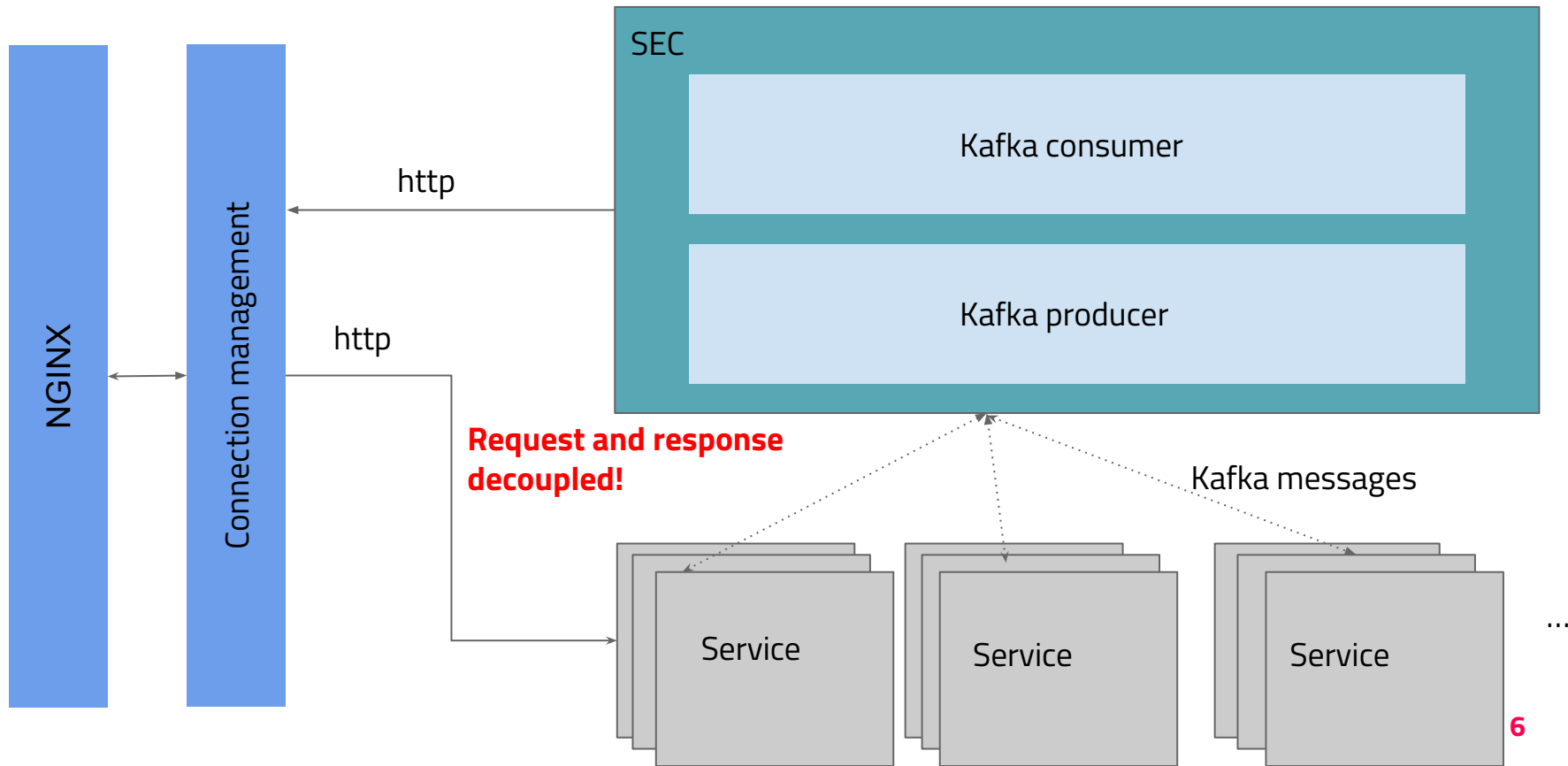
# Sagas

- Temporary inconsistency
- SEC - Saga Execution Coordinator
- MySQL log for abort

# Checkout Saga



# API Gateway




# Results

# Locust Test





# Locust Test

 LOCUST

HOSTlocalhost:8080

STATUSSTOPPED  
New test

RPS331.9

FAILURES1%

Statistics

Charts

Failures


Exceptions

Current ratio

Download Data

Type	Name	# Requests	# Fails	Median (ms)	90%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
POST	/orders/addItem/{order_id}/{item_id}	10144	0	4	6	21	4	2	118	0	61.8	0
POST	/orders/checkout/{order_id}	6612	550	2200	3000	3800	2262	742	4267	0	40.7	5.4
POST	/orders/create/{user_id}	6899	0	3	5	15	4	1	131	184	37.7	0
DELETE	/orders/removeItem/{order_id}/{item_id}	1208	0	4	6	23	4	2	51	0	7.5	0
POST	/payment/add_funds/{user_id}/{amount}	5970	0	3	5	13	3	1	80	14	30.8	0
POST	/payment/create_user	7029	0	3	5	13	3	1	77	106	36.2	0
POST	/stock/add/{item_id}/{number}	10751	0	3	5	13	3	1	105	0	54.1	0
GET	/stock/find/{item_id}	556	0	3	5	13	3	2	30	118	3.3	0
POST	/stock/item/create/{price}	10878	0	3	7	920	50	1	1015	115	56.5	0
POST	/stock/subtract/{item_id}/{number}	556	0	3	5	11	4	2	41	0	3.3	0
Aggregated		60603	550	3	1500	3100	258	1	4267	56	331.9	5.4

# Locust Test

 LOCUST

HOST  
localhost:8080

STATUS  
STOPPED  
New test

RPS  
217.3

FAILURES  
1%

Statistics

Charts

Failures

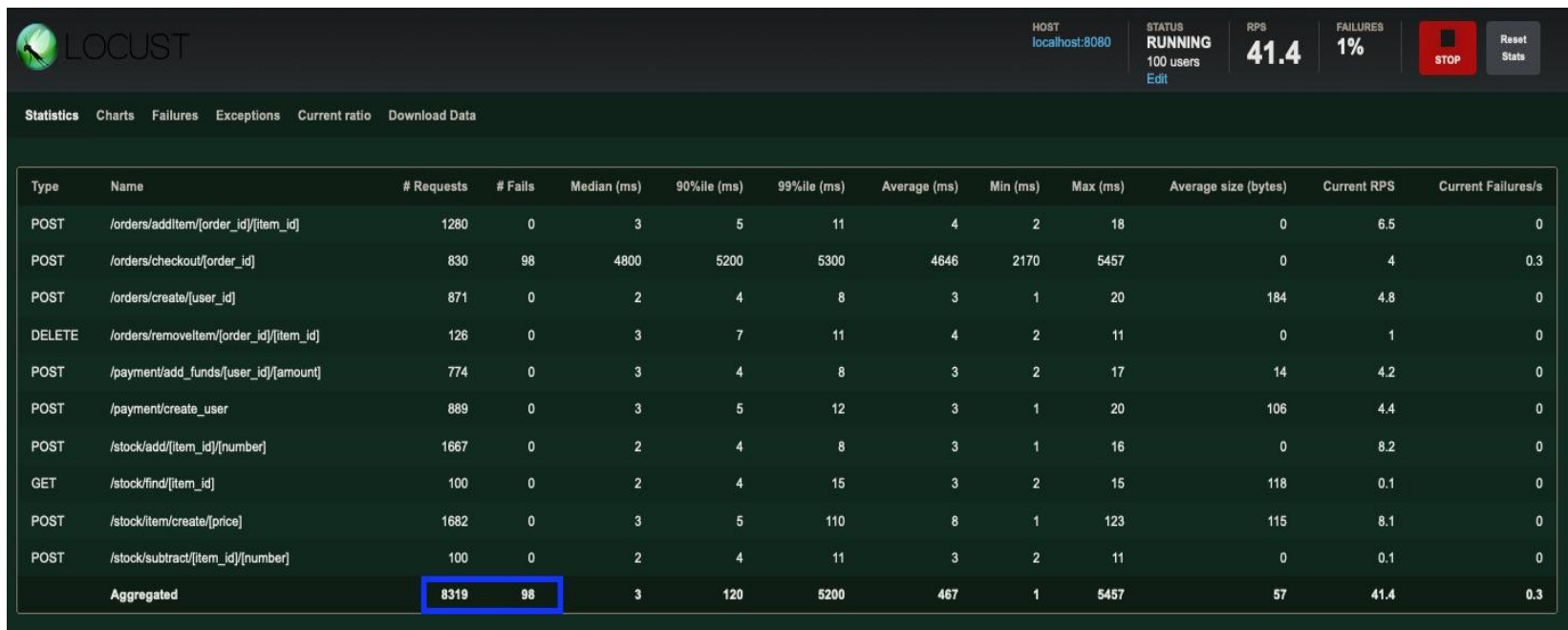
Exceptions

Current ratio

Download Data

Type	Name	# Requests	# Fails	Median (ms)	90%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
POST	/orders/addItem/[order_id]/[item_id]	7294	0	3	5	12	4	2	160	0	37.1	0
POST	/orders/checkout/[order_id]	4787	340	2100	3000	3600	2143	501	3849	0	21.1	1.8
POST	/orders/create/[user_id]	5032	0	2	4	10	3	1	163	184	26.2	0
DELETE	/orders/removeItem/[order_id]/[item_id]	1046	0	3	5	11	4	2	28	0	5.2	0
POST	/payment/add_funds/[user_id]/[amount]	4219	0	2	4	9	3	1	119	14	23.1	0
POST	/payment/create_user	5108	0	2	4	10	3	1	120	106	26.2	0
POST	/stock/add/[item_id]/[number]	7877	0	2	4	9	3	1	112	0	36.8	0
GET	/stock/find/[item_id]	348	0	3	4	10	3	2	16	118	2.6	0
POST	/stock/item/create/[price]	7949	0	3	5	830	41	1	903	115	36.4	0
POST	/stock/subtract/[item_id]/[number]	348	0	3	4	8	3	2	8	0	2.6	0
Aggregated		44008	340	3	1300	3000	243	1	3849	56	217.3	1.8

# Locust Test



# Locust Test



LOCUST

HOST  
localhost:8080

STATUS  
**RUNNING**  
100 users  
[Edit](#)

RPS  
**39.4**

FAILURES  
**1%**



[Reset](#)  
[Stats](#)

[Statistics](#) [Charts](#) [Failures](#) [Exceptions](#) [Current ratio](#) [Download Data](#)

Type	Name	# Requests	# Fails	Median (ms)	90%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
POST	/orders/addItem/[order_id]/[item_id]	1221	0	3	5	11	4	2	18	0	5.8	0
POST	/orders/checkout/[order_id]	798	97	4800	5200	5300	4648	2170	5457	0	5.3	1
POST	/orders/create/[user_id]	836	0	2	4	7	3	1	20	184	3.4	0
DELETE	/orders/removeItem/[order_id]/[item_id]	117	0	3	7	11	4	2	11	0	0.3	0
POST	/payment/add_funds/[user_id]/[amount]	743	0	3	4	8	3	2	17	14	3	0
POST	/payment/create_user	856	0	3	5	12	3	1	20	106	4.1	0
POST	/stock/add/[item_id]/[number]	1601	0	2	4	7	3	1	16	0	7.9	0
GET	/stock/find/[item_id]	98	0	2	4	15	3	2	15	118	0.5	0
POST	/stock/item/create/[price]	1623	0	3	6	110	8	1	123	115	8.6	0
POST	/stock/subtract/[item_id]/[number]	98	0	2	4	11	3	2	11	0	0.5	0
Aggregated		7991	97	3	120	5200	468	1	5457	57	39.4	1

## In Hindsight

- Don't over engineer
- Use off-the-shelf products (serverless) instead of trying to build them
- Better communication and coordination
- Start working on Kubernetes earlier to avoid issues that arise when migrating from local development at a later stage

**I WILL LIKE TO CALL AN EXCEL**



**SPREEDSHEET A  
"DATABASE"**

# Ideal SAGA

