Architecture in Motion

How Adyen achieved 100x

adyen

Bert Wolters - EVP Technology
bert@adyen.com

Traditional vs Today

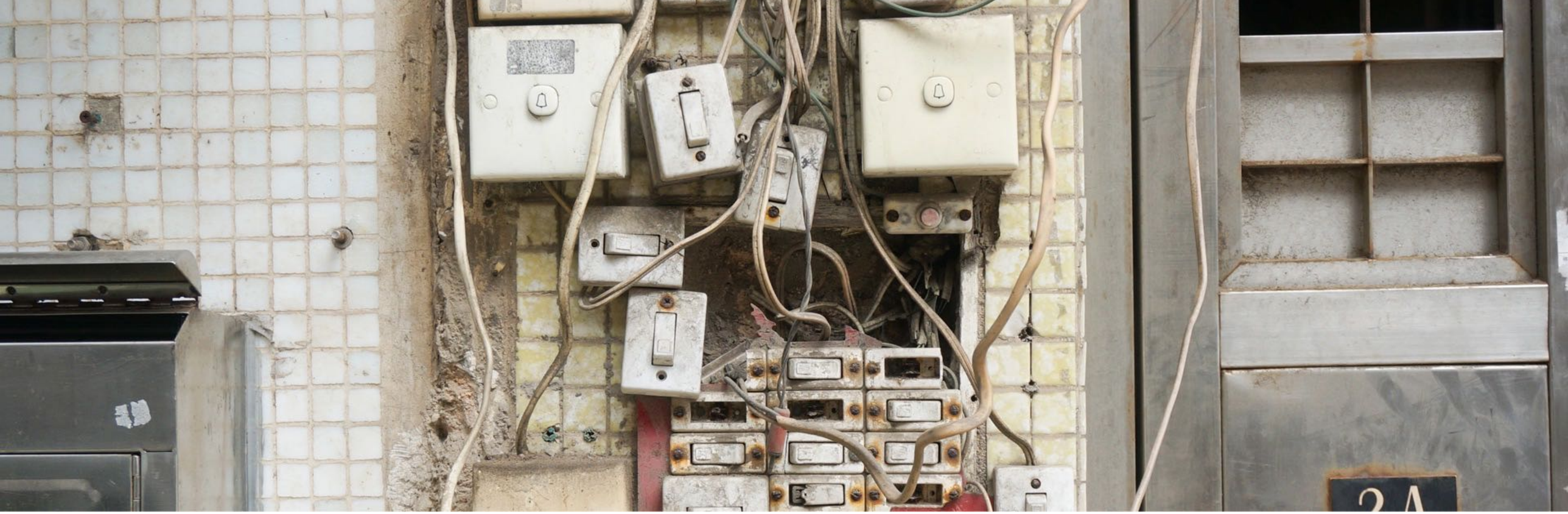
Customers are in full control

$1B

On Singles' Day in China, $1Billion was processed in 1 minute and 25 seconds

80%

On Black Friday in the U.S., nearly 80% of retailers' online traffic took place on mobile.

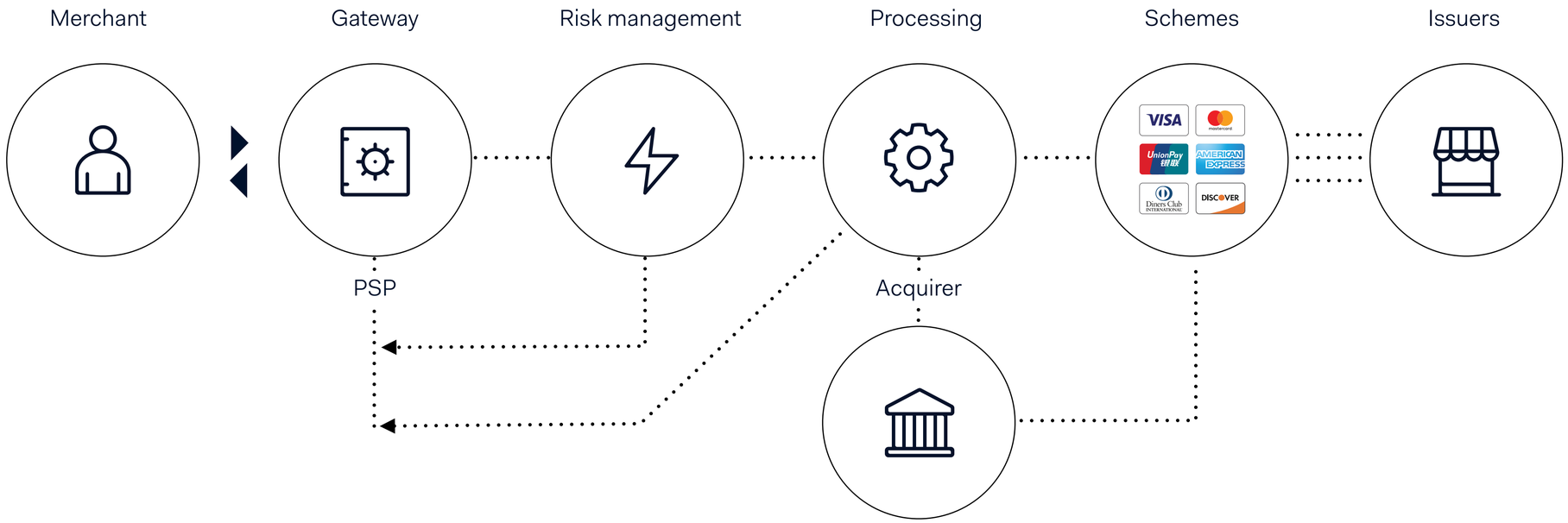


Companies face a lack of functionality, flexibility and innovation

Stuck with legacy technology that lead to poor user experience.

One modern platform

Traditional value chain



This is Adyen

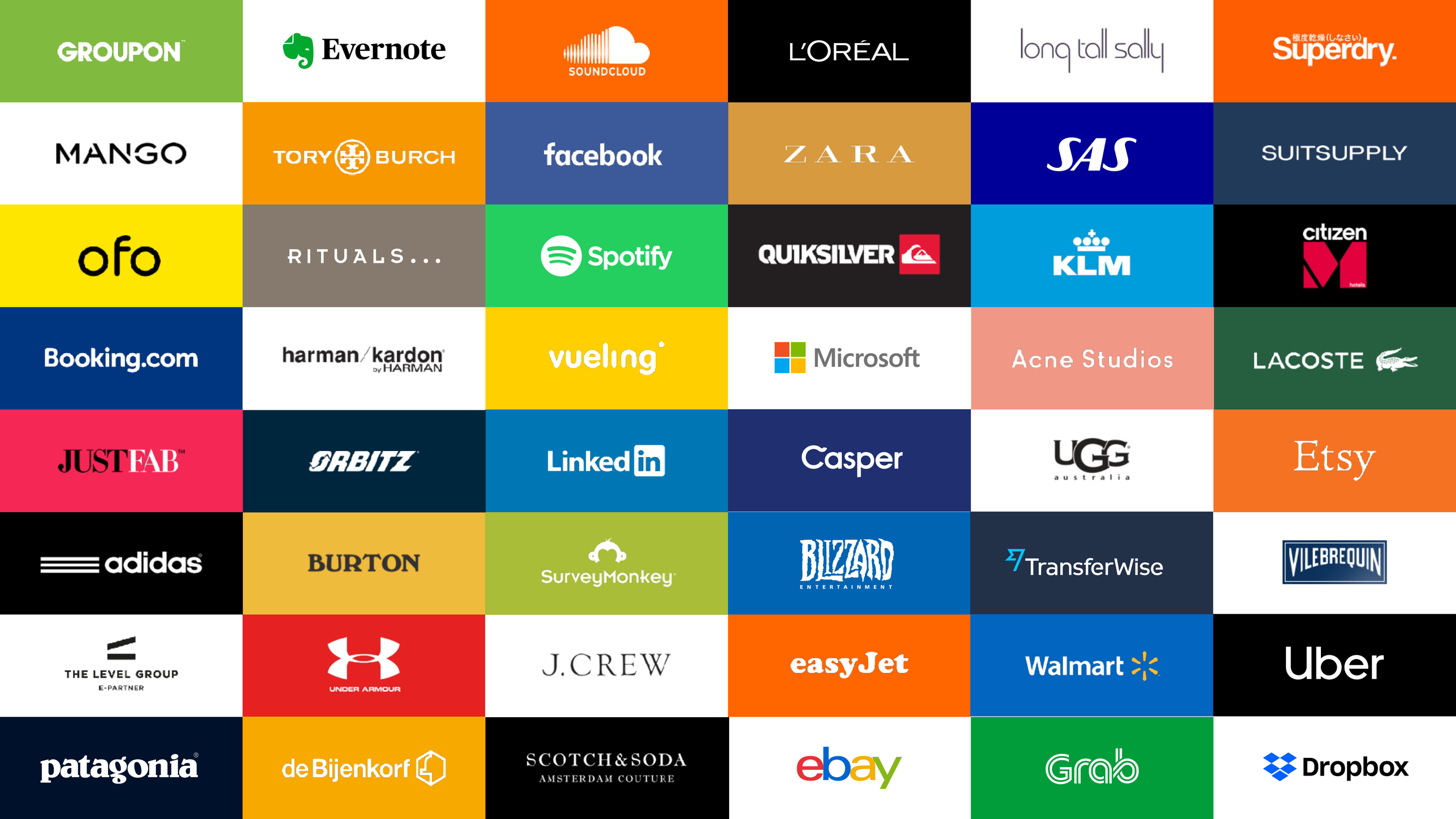
ALL TECHNOLOGY DEVELOPED IN-HOUSE

One platform, one contract, all sales channels

250+ payment methods

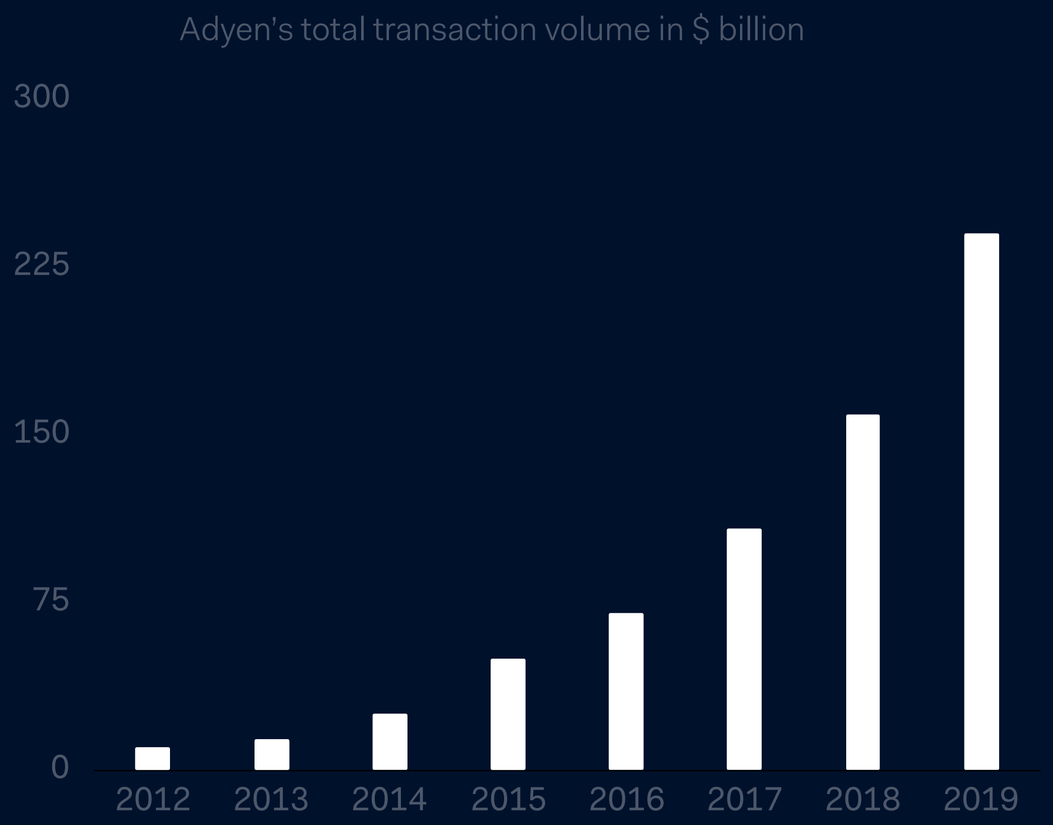
1200+ global employees

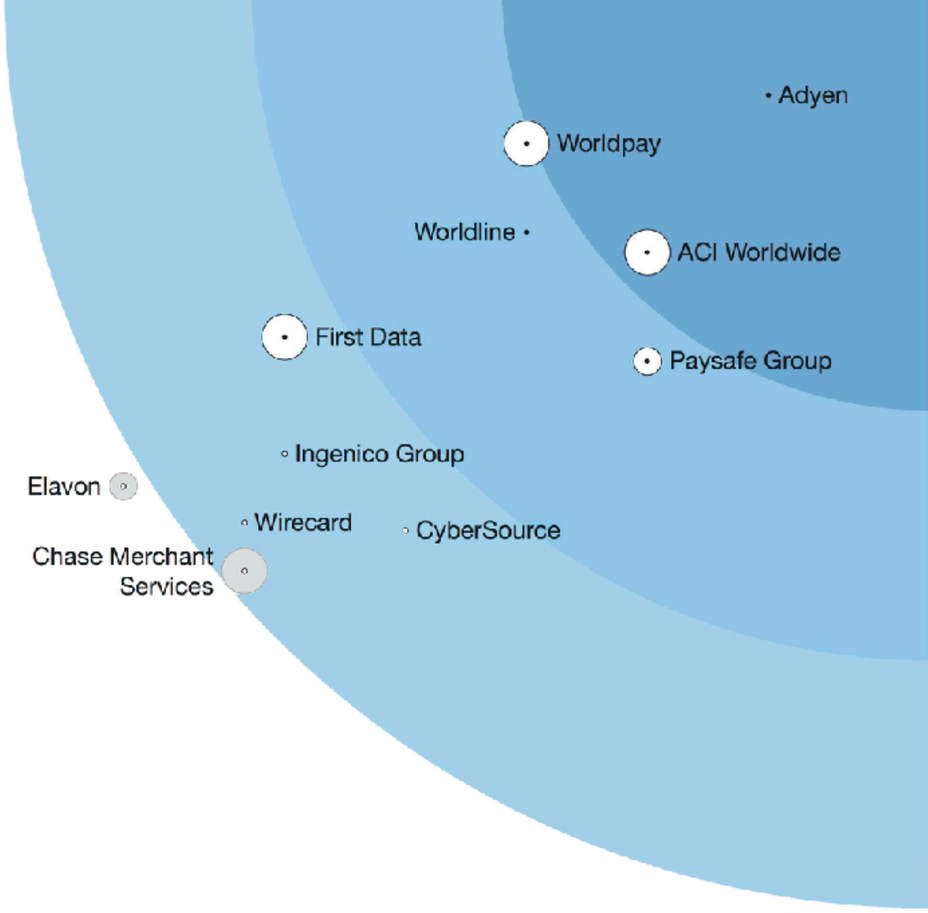
23 Global offices


GROUPON
Evernote
SOUNDCLOUD
L'ORÉAL
long tall sally
極度乾燥(しなさい)
Superdry.
MANGO
TORY BURCH
facebook
ZARA
SAS
SUITSUPPLY
ofo
RITUALS...
Spotify
QUIKSILVER
KLM
citizen
hotels
Booking.com
harman/kardon
by HARMAN
vueling
Microsoft
Acne Studios
LACOSTE
JUSTFAB
ORBITZ
LinkedIn
Casper
UGG
australia
Etsy
adidas
BURTON
SurveyMonkey
BLIZZARD
ENTERTAINMENT
TransferWise
VILEBREQUIN
THE LEVEL GROUP
E-PARTNER
UNDER ARMOUR
J.CREW
easyJet
Walmart
Uber
patagonia
de Bijenkorf
SCOTCH&SODA
AMSTERDAM COUTURE
ebay
Grab
Dropbox

CNBC

"The company behind Facebook, Uber and Netflix payments reveals huge transaction growth"





"Adyen is the best fit for global merchants that want to increase their overall payments performance, reduce their fraud rates and simplify their overall payments operations"

FORRESTER®

The Adyen Way of Engineering

We prioritize **current merchant experience** over future features

To think like the **merchant** we **go meet them**

We design for **20x**

Expose your work **early**

We are all **designers**, **architects**, **coders**, **testers**, **security officers** and **operations engineers**

We all make mistakes, but we **seek help** as soon as we find out

You own when, where and how **your code goes live**

Your code should be understandable **at 4am** under stress

We embrace **new technology** when it has clear benefits

Our tech stack is **open source** or **built in-house**

The Adyen Way of Engineering

We prioritize **current merchant experience** over future features

To think like the **merchant** we **go meet them**

We design for **20x**

Expose your work **early**

We are all **designers**, **architects**, **coders**, **testers**, **security officers** and **operations engineers**

We all make mistakes, but we **seek help** as soon as we find out

You own when, where and how **your code goes live**

Your code should be understandable **at 4am** under stress

We embrace **new technology** when it has clear benefits

Our tech stack is **open source** or **built in-house**

The Adyen Way of Engineering

We prioritize **current merchant experience** over future features

To think like the **merchant** we **go meet them**

We design for **20x**

Expose your work **early**

We are all **designers**, **architects**, **coders**, **testers**, **security officers** and **operations engineers**

We all make mistakes, but we **seek help** as soon as we find out

You own when, where and how **your code goes live**

Your code should be understandable **at 4am** under stress

We embrace **new technology** when it has clear benefits

Our tech stack is **open source** or **built in-house**

The Adyen Way of Engineering

We prioritize **current merchant experience** over future features

To think like the **merchant** we **go meet them**

We design for **20x**

Expose your work **early**

We are all **designers**, **architects**, **coders**, **testers**, **security officers** and **operations engineers**

We all make mistakes, but we **seek help** as soon as we find out

You own when, where and how **your code goes live**

Your code should be understandable **at 4am** under stress

We embrace **new technology** when it has clear benefits

Our tech stack is **open source** or **built in-house**

The Adyen Way of Engineering

We prioritize **current merchant experience** over future features

To think like the **merchant** we **go meet them**

We design for **20x**

Expose your work **early**

We are all **designers**, **architects**, **coders**, **testers**, **security officers** and **operations engineers**

We all make mistakes, but we **seek help** as soon as we find out

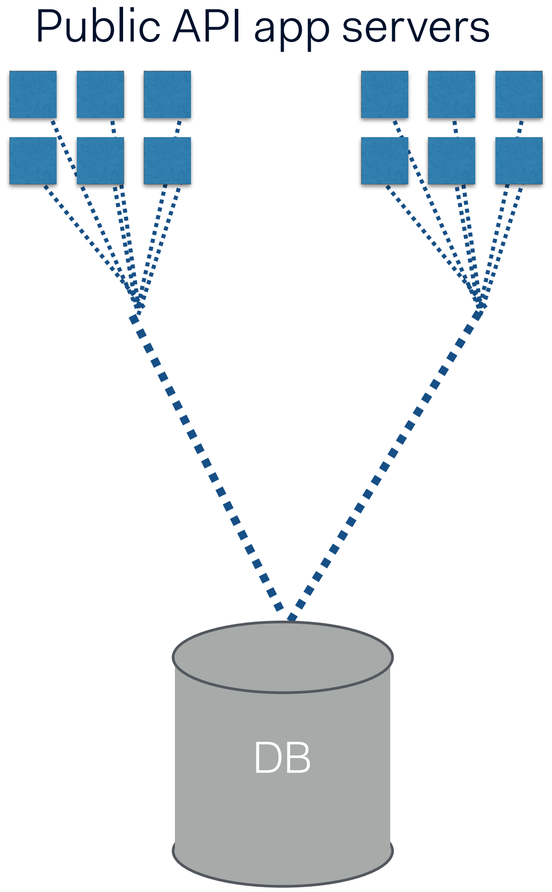
You own when, where and how **your code goes live**

Your code should be understandable **at 4am** under stress

We embrace **new technology** when it has clear benefits

Our tech stack is **open source** or **built in-house**

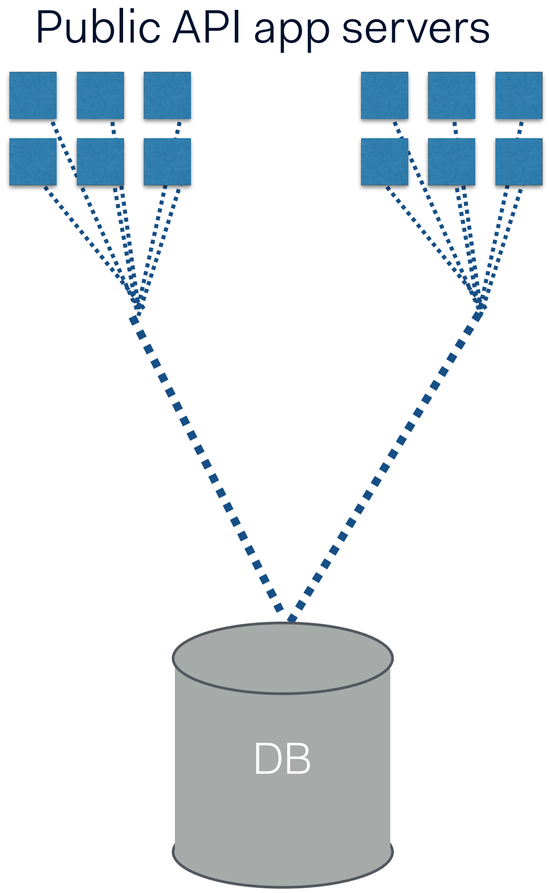
High-Level Architecture
(Bibit / pre-Adyen)



Shopper payment flow,
Risk analysis,
Payment authorization

Reconciliation,
Reporting

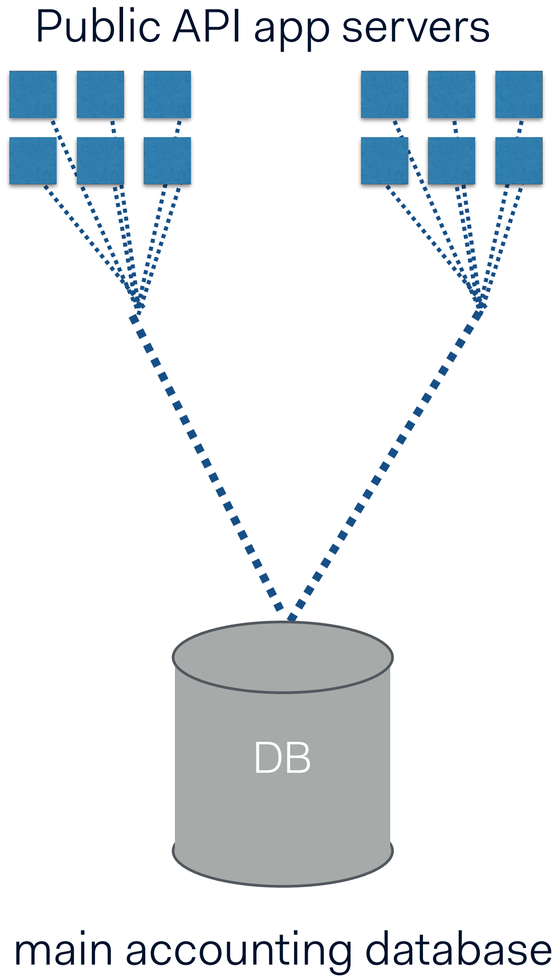
High-Level Architecture
(pre-Adyen)



???

???

High-Level Architecture
(pre-Adyen)

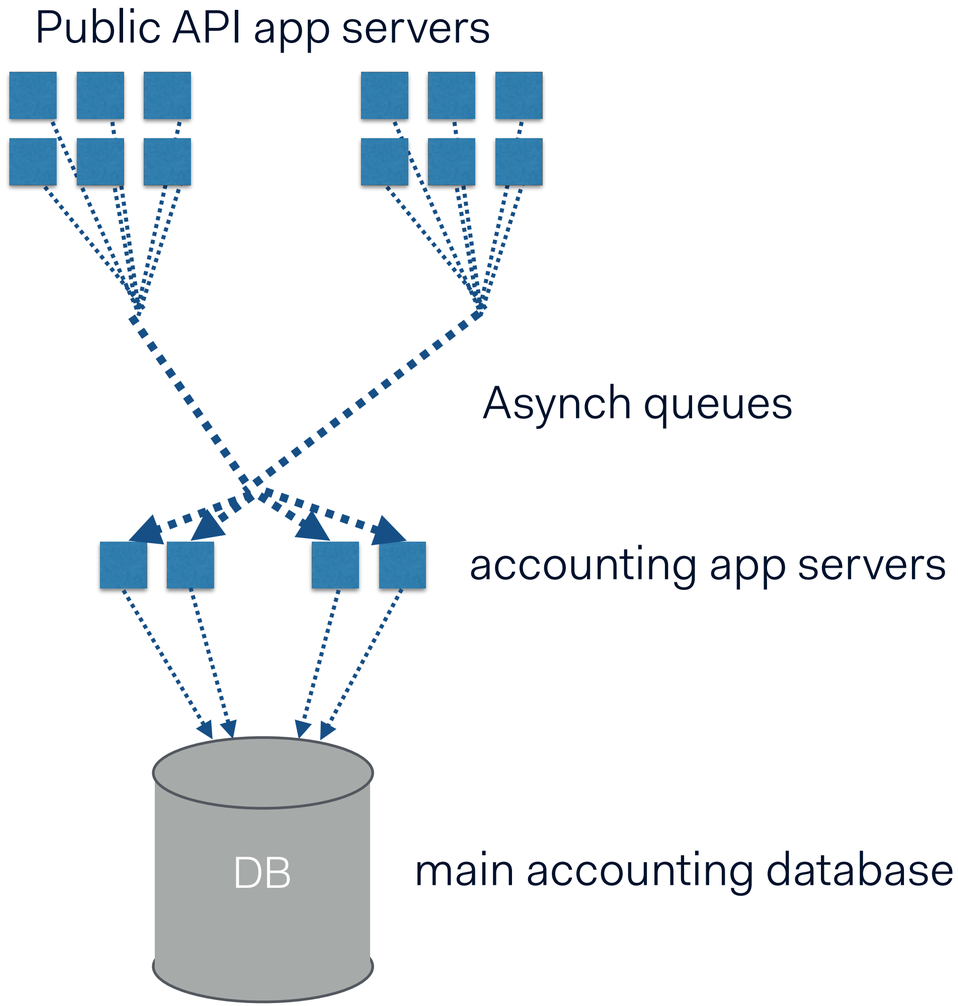


Availability

Consistency

Design for Breakage

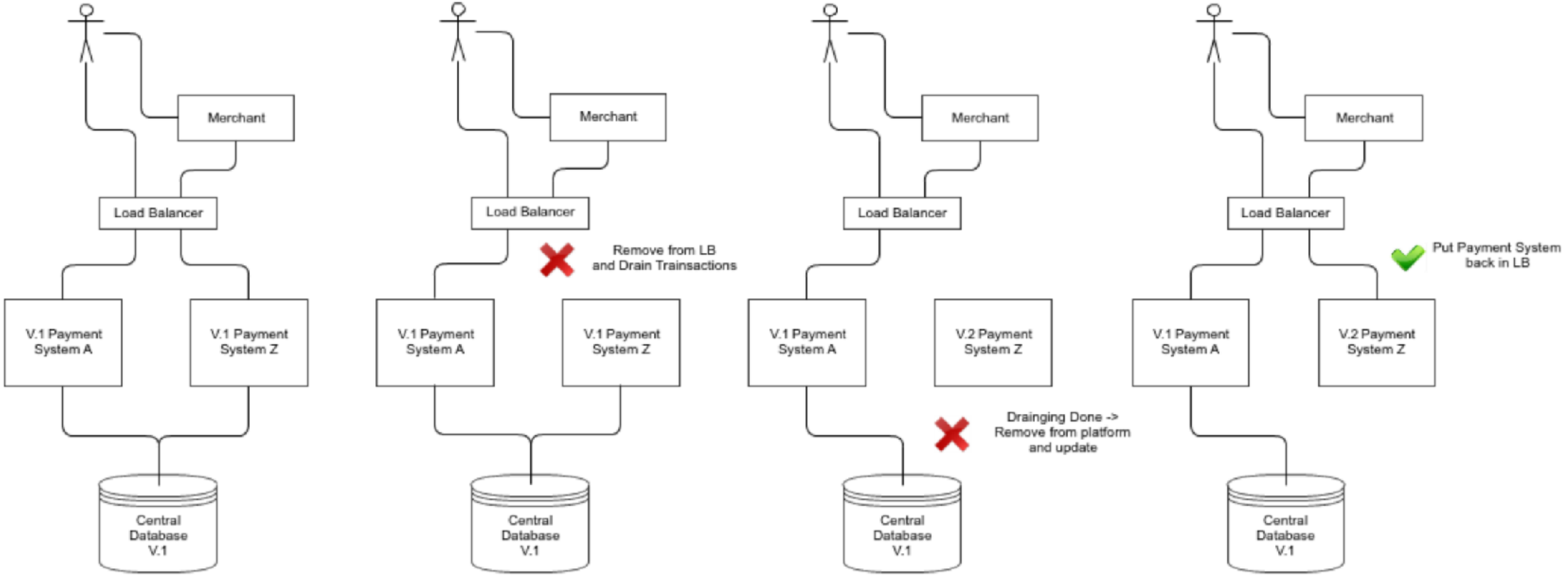
High-Level Architecture
(first years)



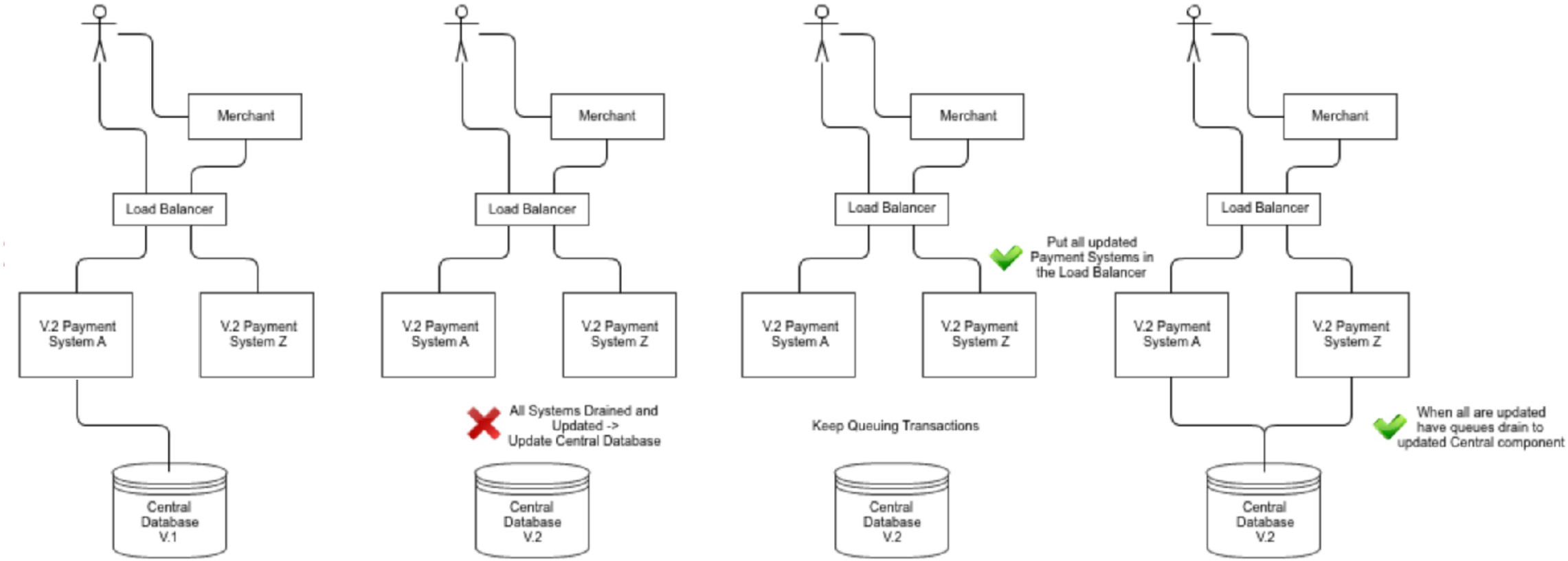
Stateless

Statefull

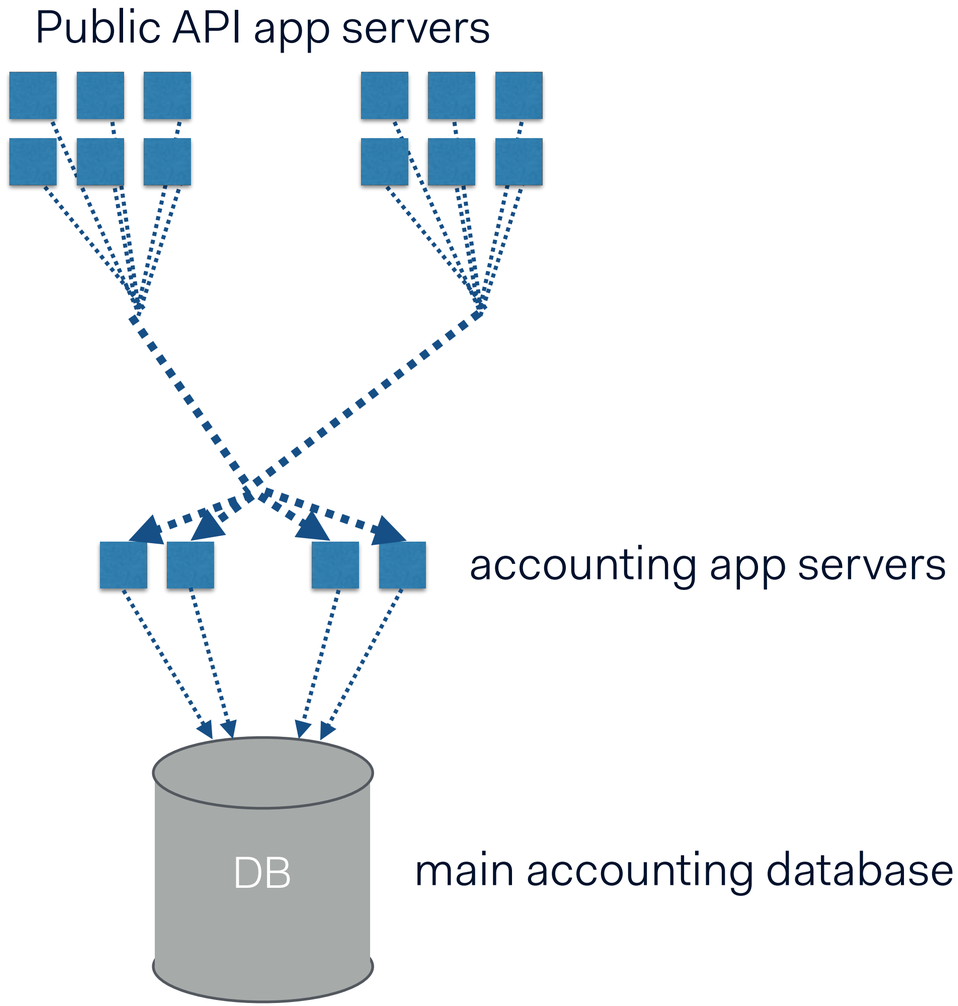
Front-end Maintenance



Database Maintenance



High-Level Architecture
(first years)



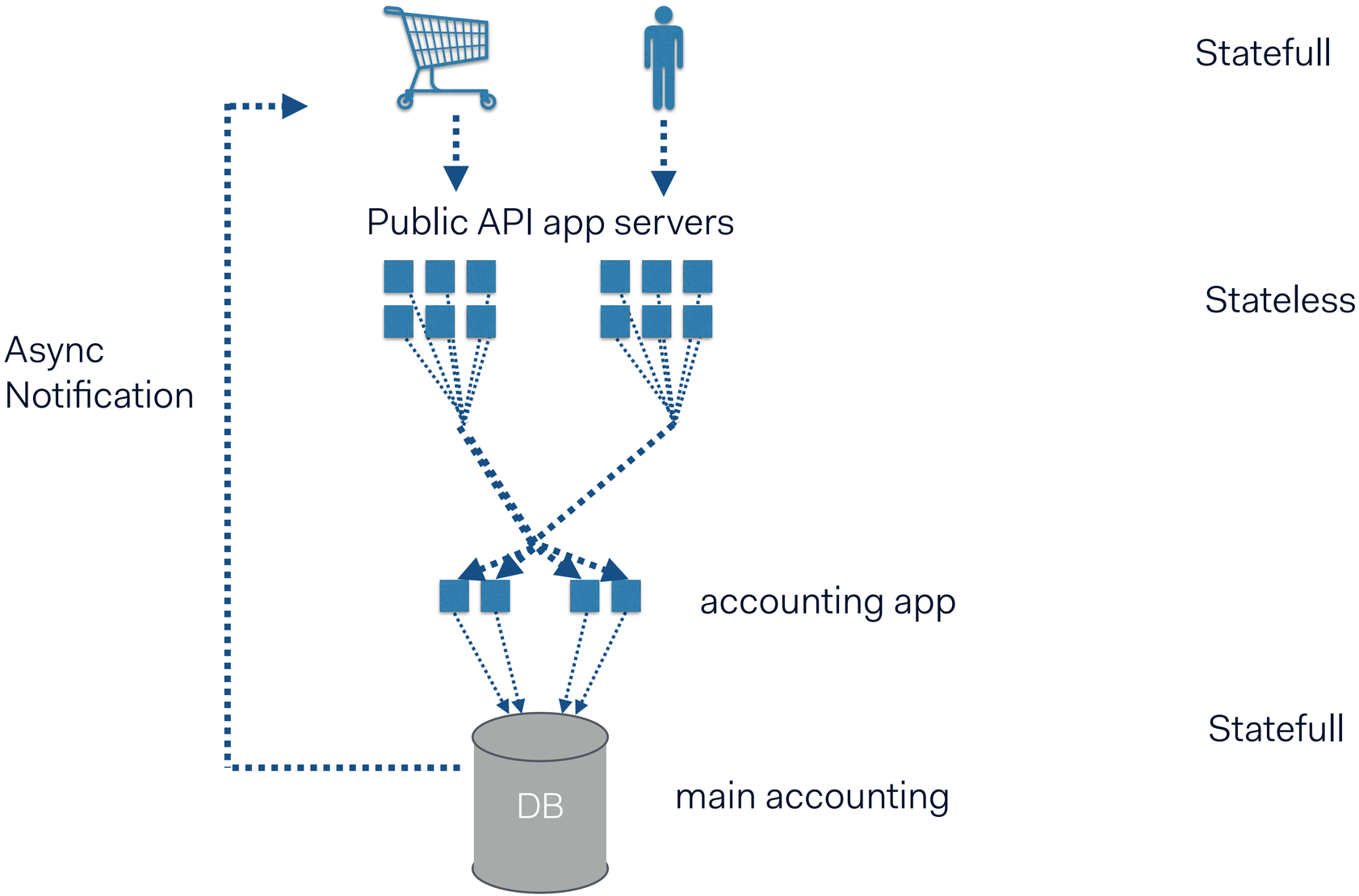
Stateless

accounting app servers

Statefull

main accounting database

Consequences of being Stateless



Consequences of being Stateless

Refunds

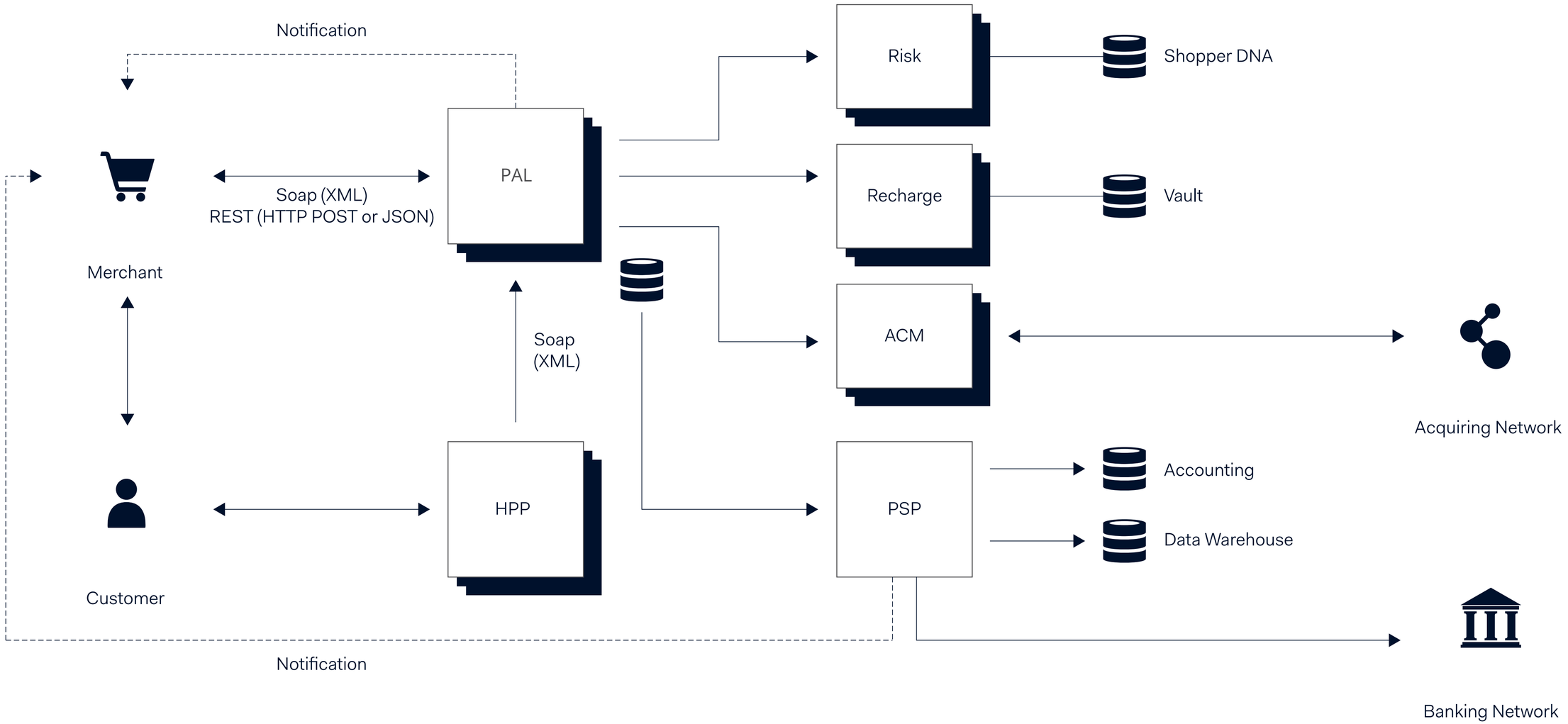
Request:

```
{
   "merchantAccount" : "TestMerchant",
      "modificationAmount" : {
      "value" : 500,
      "currency" : "EUR"
   },
   "originalReference" : "9313547924770610",
   "reference" : "YourModificationReference"
}
```

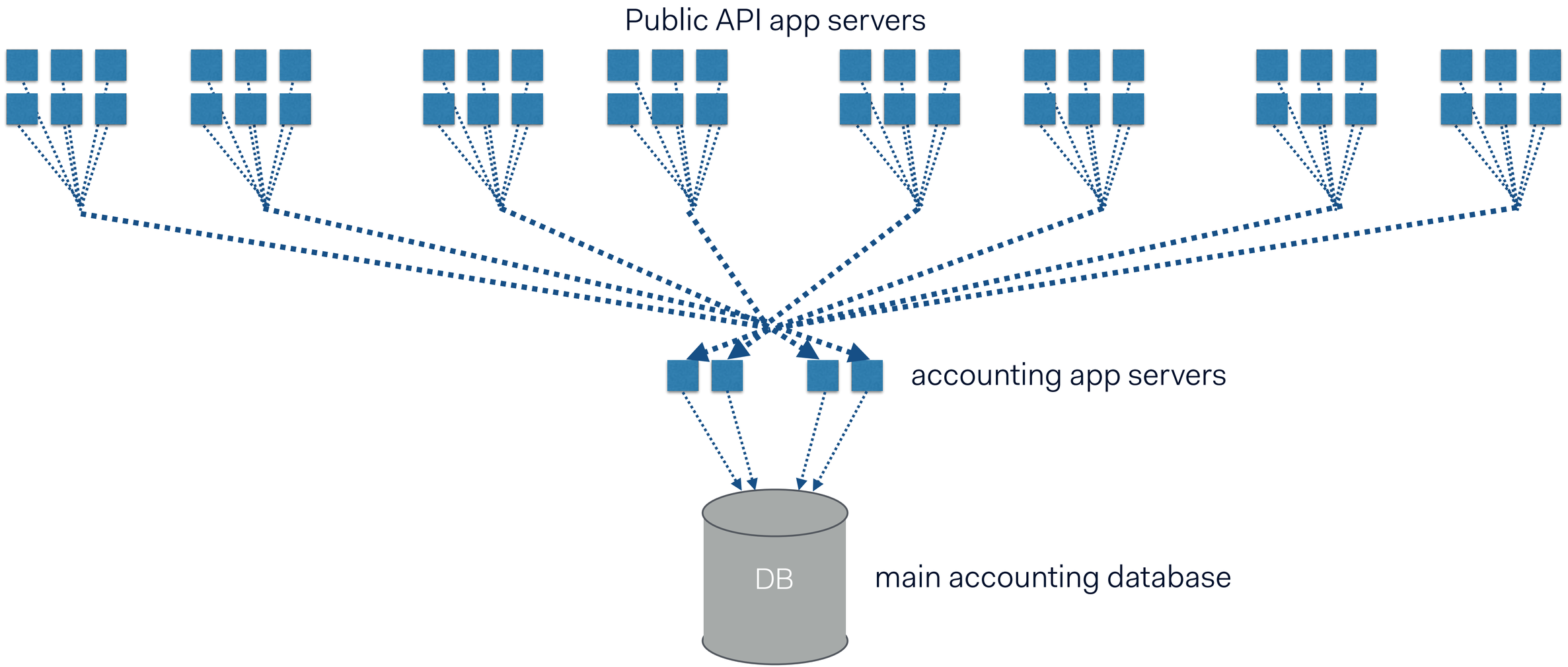
Response:

```
{
   "pspReference" : "8312534564722331",
   "response" : "[refund-received]"
}
```

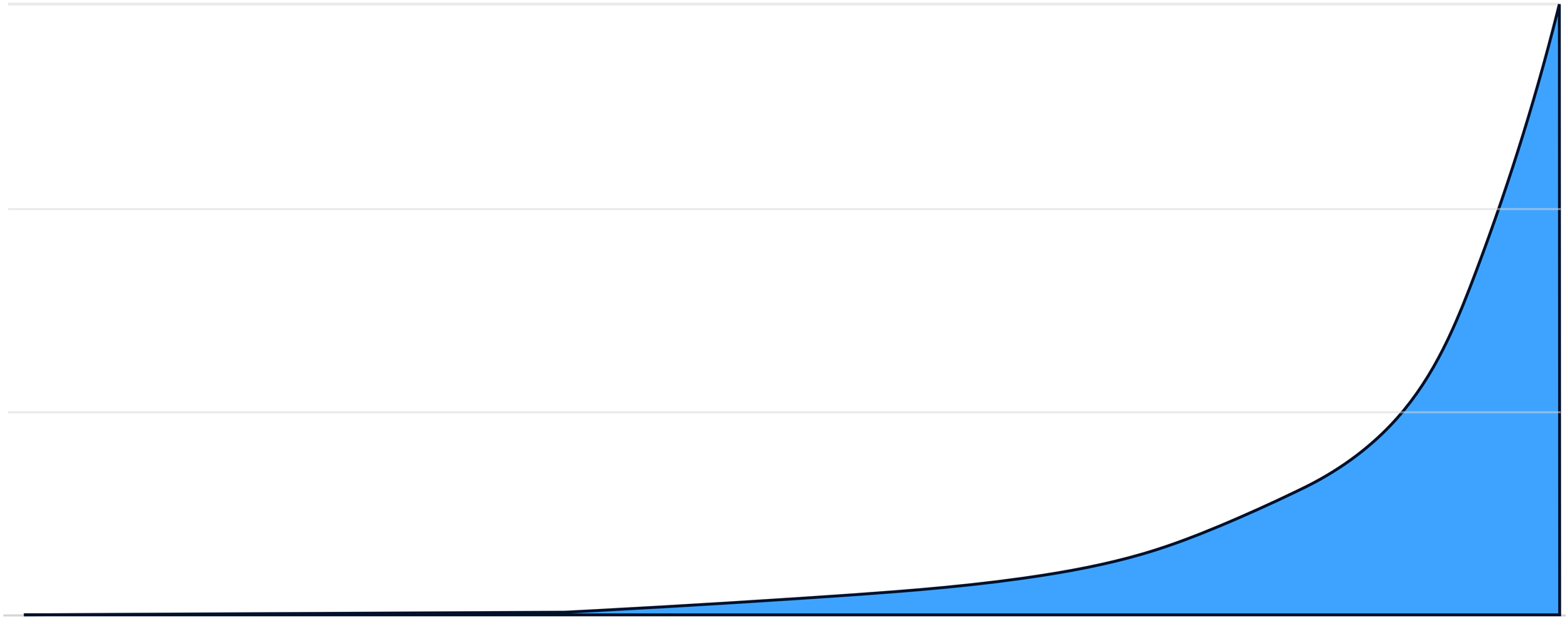
Architecture Diagram – Authorisations



The Challenge



Growing Exponentially
over $100b processed annually



Scalability Challenges

API (micro)services designed to be highly redundant and stateless. Scale linearly with more hardware.

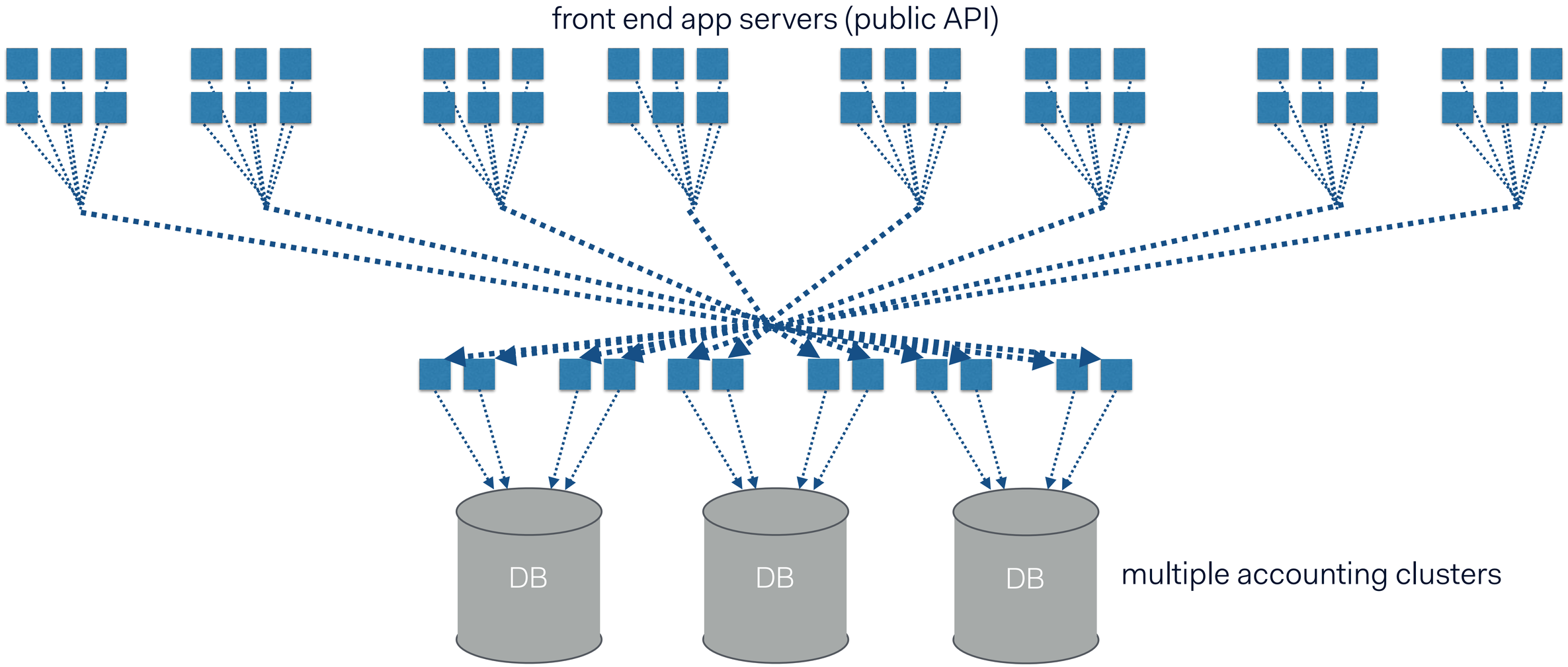
However main payment accounting system was running > 70TB on a single PostgreSQL instance at up to 25k tps.

At 2-4x, optimisation and/or bigger hardware solve the problem. At 20x this is no longer sufficient and requires re-architecting.

Marbles

Generating Reports/Batch Files

The Solution / The NEW Challenge



More jars...

PspReference
Refunds

Request:

```
{
   "merchantAccount" : "TestMerchant",
      "modificationAmount" : {
      "value" : 500,
      "currency" : "EUR"
   },
   "originalReference" : "9313547924770610",
   "reference" : "YourModificationReference"
}
```

Response:

```
{
   "pspReference" : "8312534564722331",
   "response" : "[refund-received]"
}
```


pocket83

Streaming Framework

Accounting DB's should be insert only

Reduce I/O and CPU in the main DB (cache thrashing / spilling to disk)

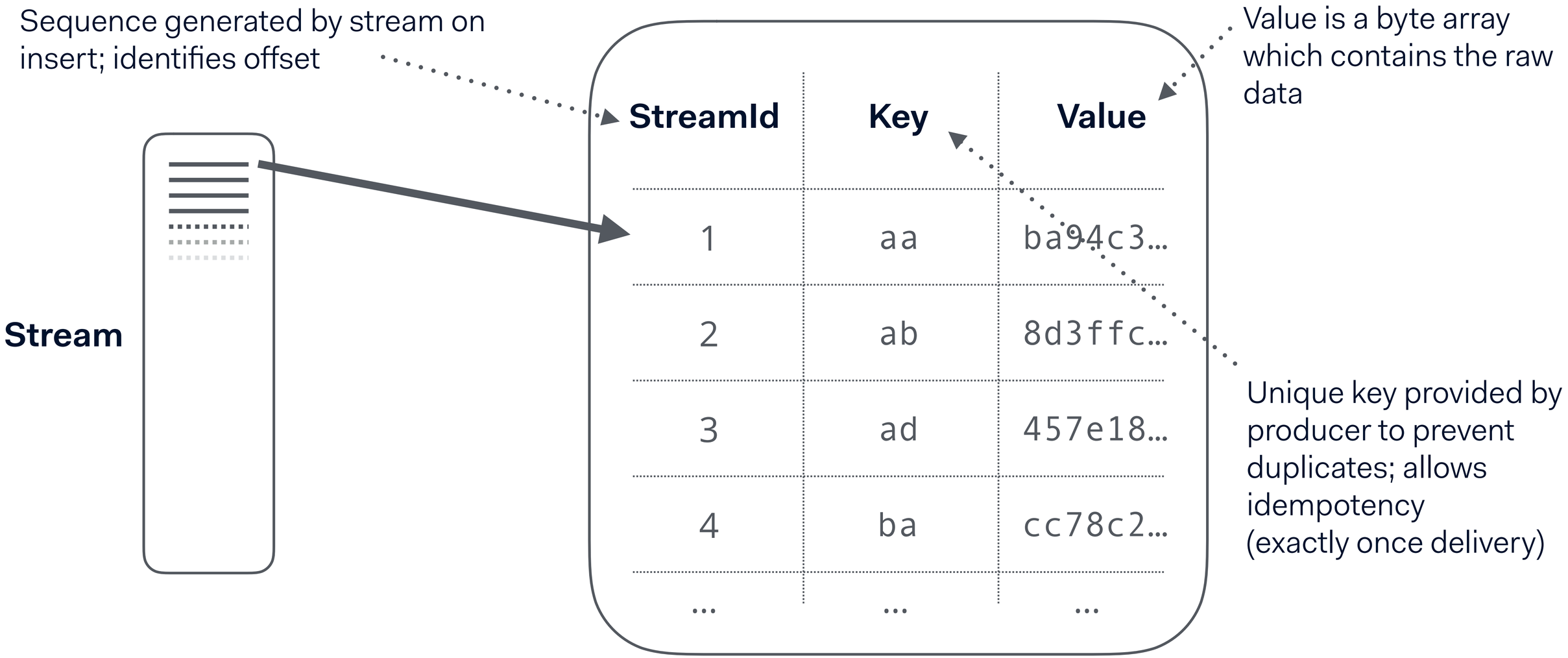
Exactly once delivery (Kafka started to support this since 2017)

Prevent multi-shard queries

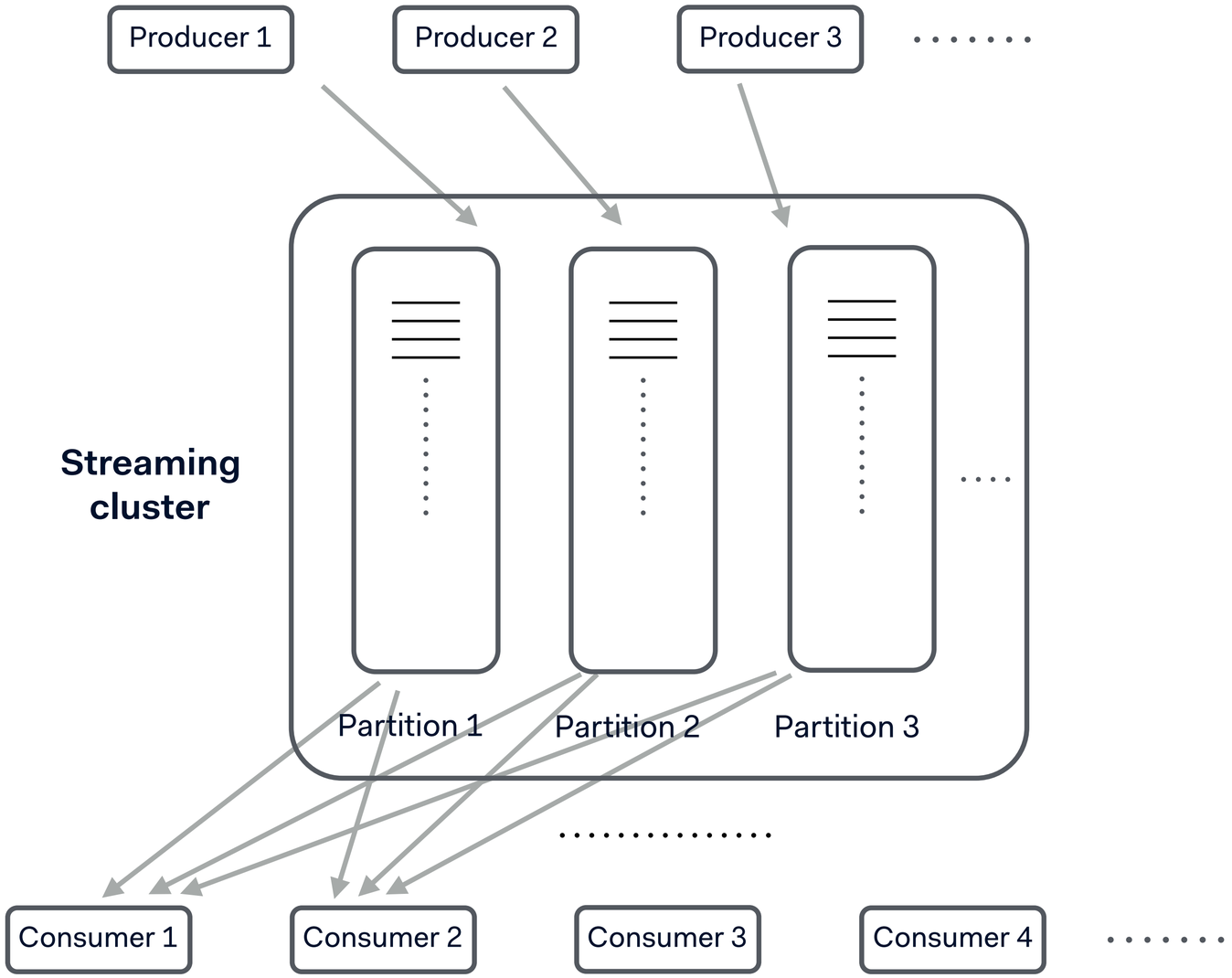
Stream Contents


```
"@type" : "com.adyen.protocol.stream.journalstream.JournalStreamItem",
   "journalStreamItemType" : "Journal",
   "journal" : {
      "bookingDate" : 1493908683696,
      "journalId" : 227274316270,
      "postDate" : 1493908682341,
      "lines" : [
         {
            "registerTypeId" : 23,
            "quantity" : 10500,
            "accountId" : 378293,
            "batch" : {
               "accountId" : 378293,
               "periodEndDate" : 1493935200000,
               "registerTypeId" : 23,
               "batchId" : 506220414,
               "periodBeginDate" : 1493848800000
            },
            "unitId" : 840,
            "journalLineId" : 117964524937,
            "batchId" : 506220414
         },
```

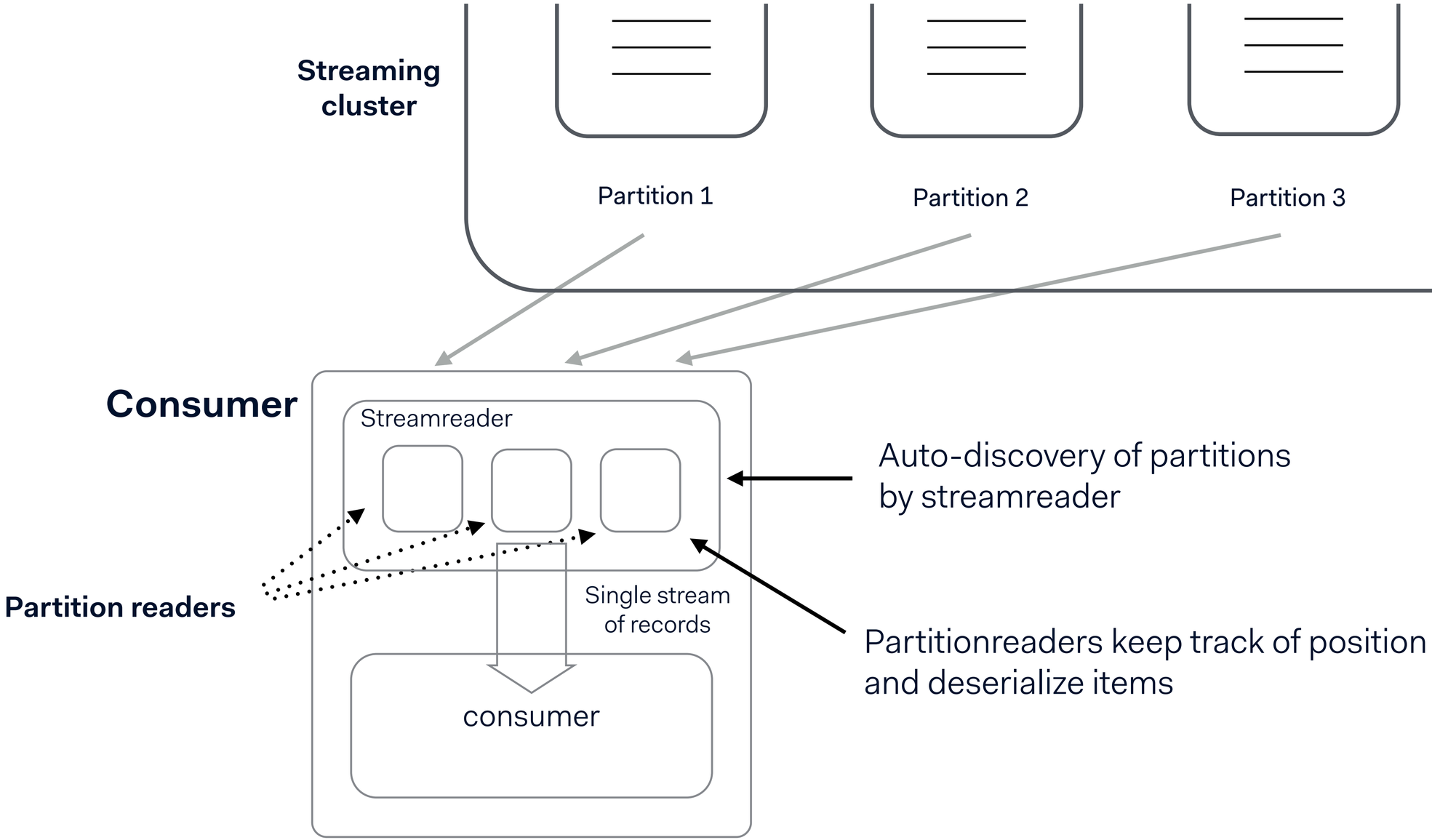
Streaming is an Idempotent Log



A stream consists of multiple partitions

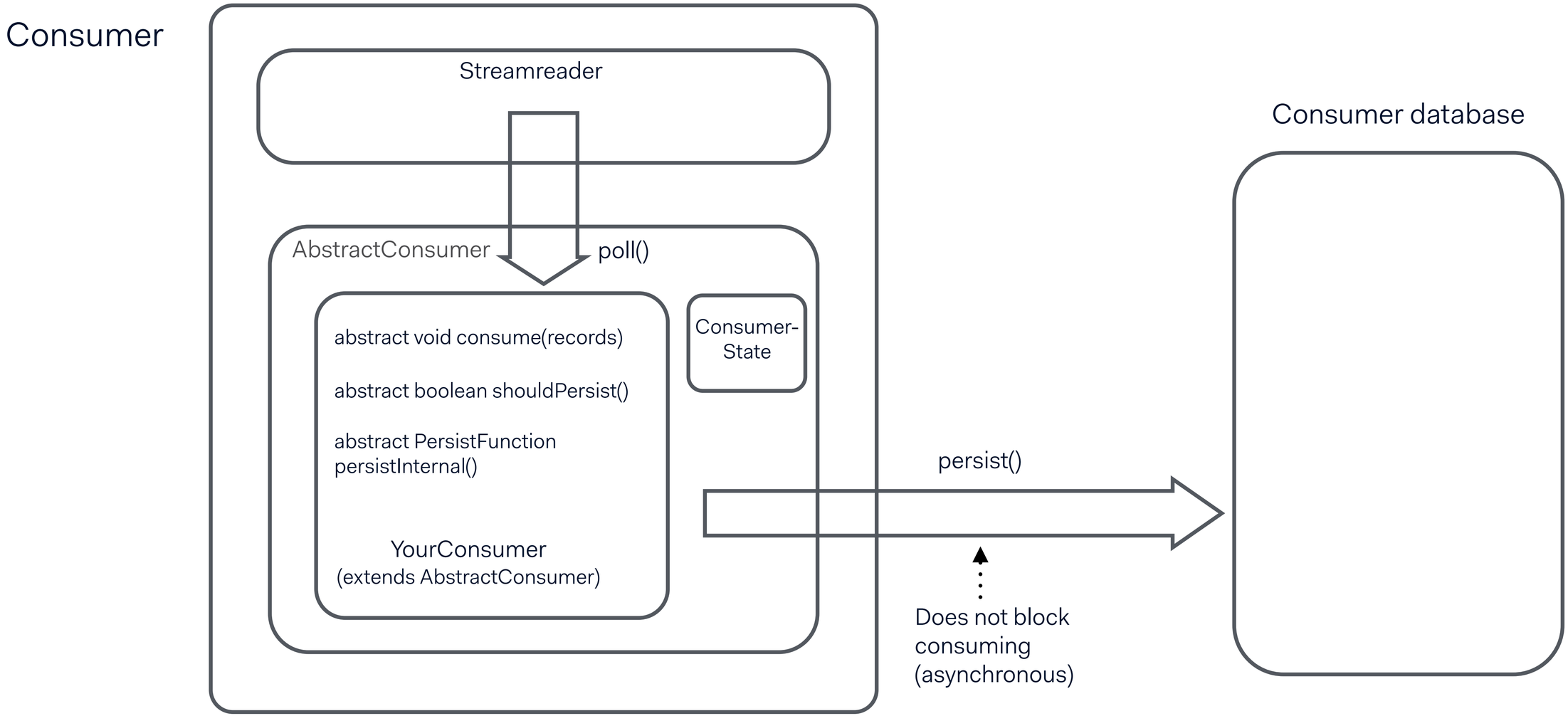


Streamreader hides complexity of reading partitions



Extend AbstractConsumer to implement a new consumer

AbstractConsumer communicates with streamreader and keeps track of consumerState



Implementation

- Built on PostgreSQL (ACID)
- Denormalised data consumes lots of bandwidth/storage
- Very fast (de)serialisation and compact serialised representation
- Using FastJSON (Alibaba) with Zstandard (Facebook) compression achieves similar performance/byte size
- Still produces > 200Gb/day!

Putting it into Production

Results after Deployment

First customers live on multi-cluster in January 2017 with no customer impact

Streaming handles virtually all data functions which would have required multi-cluster queries.

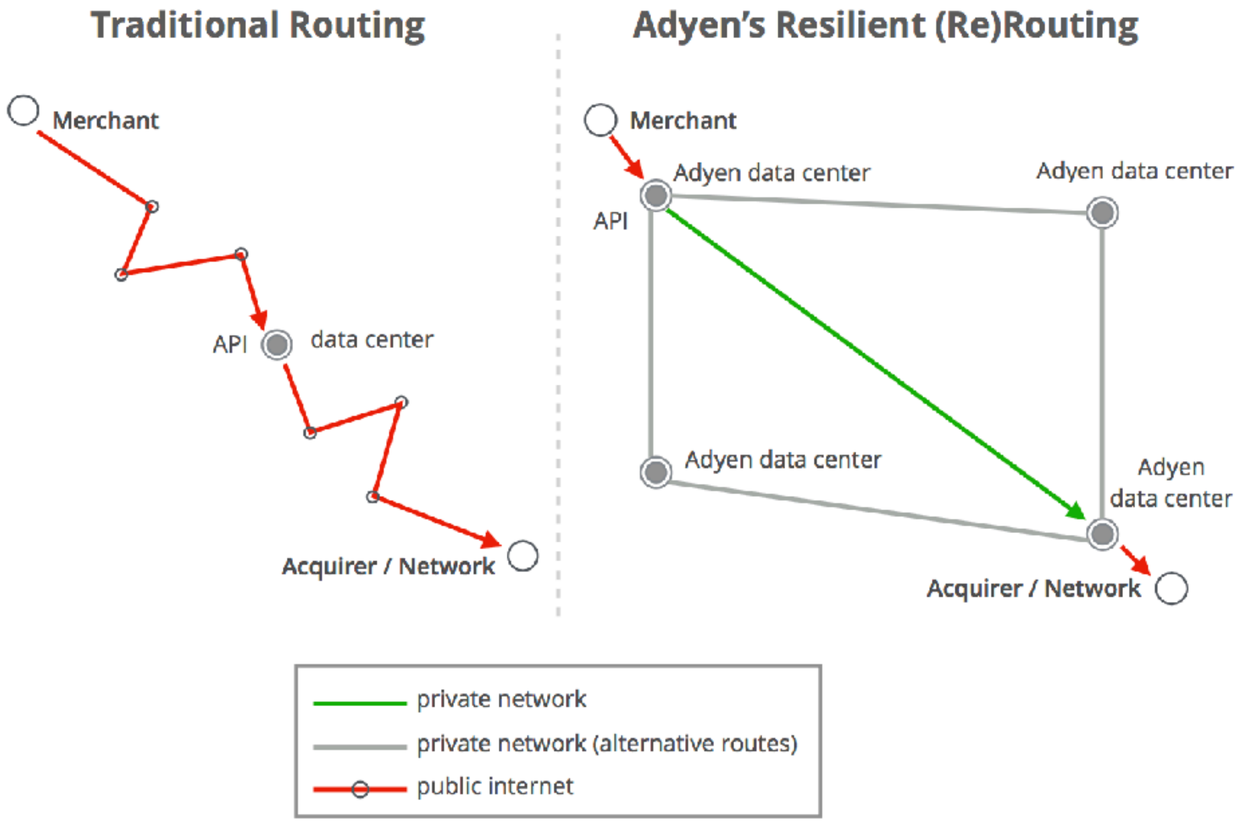
DB tps on primary cluster dropped from 25k to 8k in less than two months

Global Data Centers

Leased lines: MPLS links

Improving Latency and Resilience

- Multiple data centers per region to provide resilience against internal and external issues
- Merchants connect into data center in their own geographical region minimising latency and hops over public internet
- Sysmon recovery and health-based rerouting choose optimal path for each transaction



Architecture in Motion

How Adyen achieved 100x

adyen