

Building event-driven Microservices with Kafka Ecosystem

Guido Schmutz
London, 30.5.2018



 @gschmutz

 guidoschmutz.wordpress.com

BASEL ▪ BERN ▪ BRUGG ▪ DÜSSELDORF ▪ FRANKFURT A.M. ▪ FREIBURG I.BR. ▪ GENF
HAMBURG ▪ KOPENHAGEN ▪ LAUSANNE ▪ MÜNCHEN ▪ STUTTGART ▪ WIEN ▪ ZÜRICH

trivadis
makes **IT** easier. 

■ Guido Schmutz



Working at Trivadis for more than 21 years

Oracle ACE Director for Fusion Middleware and SOA



Consultant, Trainer Software Architect for Java, Oracle, SOA and Big Data / Fast Data

Head of Trivadis Architecture Board

Technology Manager @ Trivadis

More than 30 years of software development experience

Contact: guido.schmutz@trivadis.com

Blog: <http://guidoschmutz.wordpress.com>

Slideshare: <http://www.slideshare.net/gschmutz>

Twitter: [@gschmutz](https://twitter.com/gschmutz)

gschmutz 24.08.17 on April 18, 2017

Tags: flink (1), kafka (59), kafka-connect (4), kafka-streams (17), spark-streaming (31), storm (39), streams (4)

Last week in Stream Processing & Analytics – 18.4.2017

This is the 62nd edition of my blog series blog series around Stream Processing and Analytics!

Every week I'm also updating the following two lists with the presentations/videos of the current week:

- [Presentations from Slideshare](#)
- [Videos from YouTube](#)

As usual, find below the new blog articles, presentations, videos and software releases from last week:

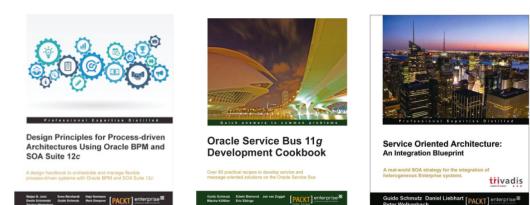
News and Blog Posts

General

- [Multi Master Replication For Geo-Distributed Data: It's more than you think by Ellen Friedman](#)
- [Understanding Indicators of Attack \(IOAs\): The Power of Event Stream Processing in CrowdStrike Falcon by Dan Brown](#)
- [Stream processing and messaging systems for the IoT age by Ben Lorica](#)

Apache Kafka / Kafka Streams / Confluent Platform

- [Creating a Data Pipeline with Kafka Connect API – from Architecture to Operations by Alexandra Wang](#)
- [Streaming Spring Boot Application Logs to ELK Stack – Part 1 by kaadayamuthu](#)
- [Streaming Spring Boot Application Logs to Apache Kafka – ELK/Kafka Stack – Part 2 by kaadayamuthu](#)



Building event-driven Microservices with Kafka Ecosystem

■ Agenda

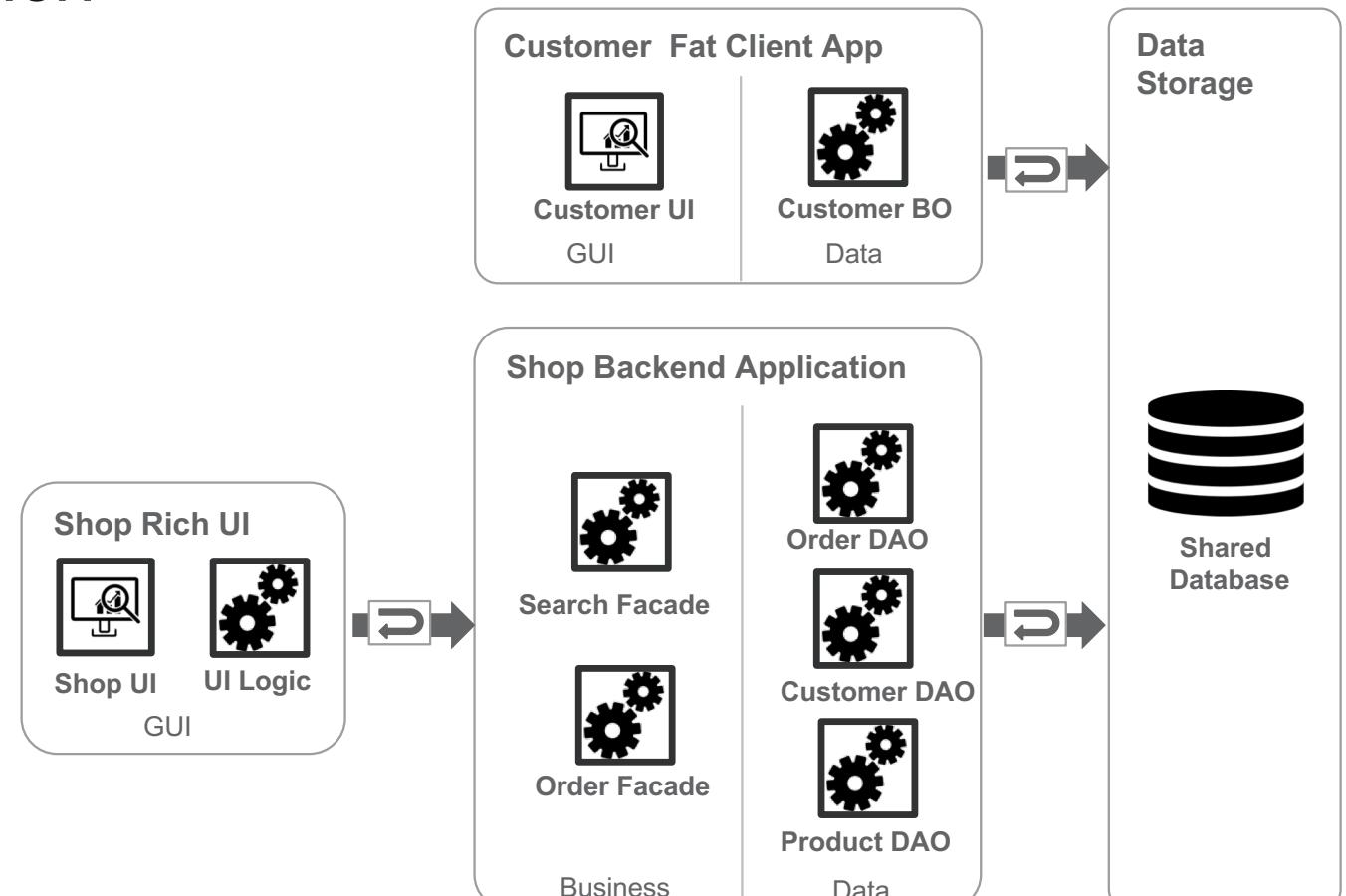
1. Where do we come from?
2. What are Microservices?
3. Why not Event Driven?
4. What about streaming sources?
5. What about integrating legacy applications?
6. What about (historical) data analytics?
7. Why Kafka for Event-Driven Microservices?
8. Summary

Where do we come from?

Building event-driven Microservices with Kafka Ecosystem



■ Traditional Approach



sync request/response

Building event-driven Microservices with Kafka Ecosystem

trivadis
makes IT easier. ■ ■ ■

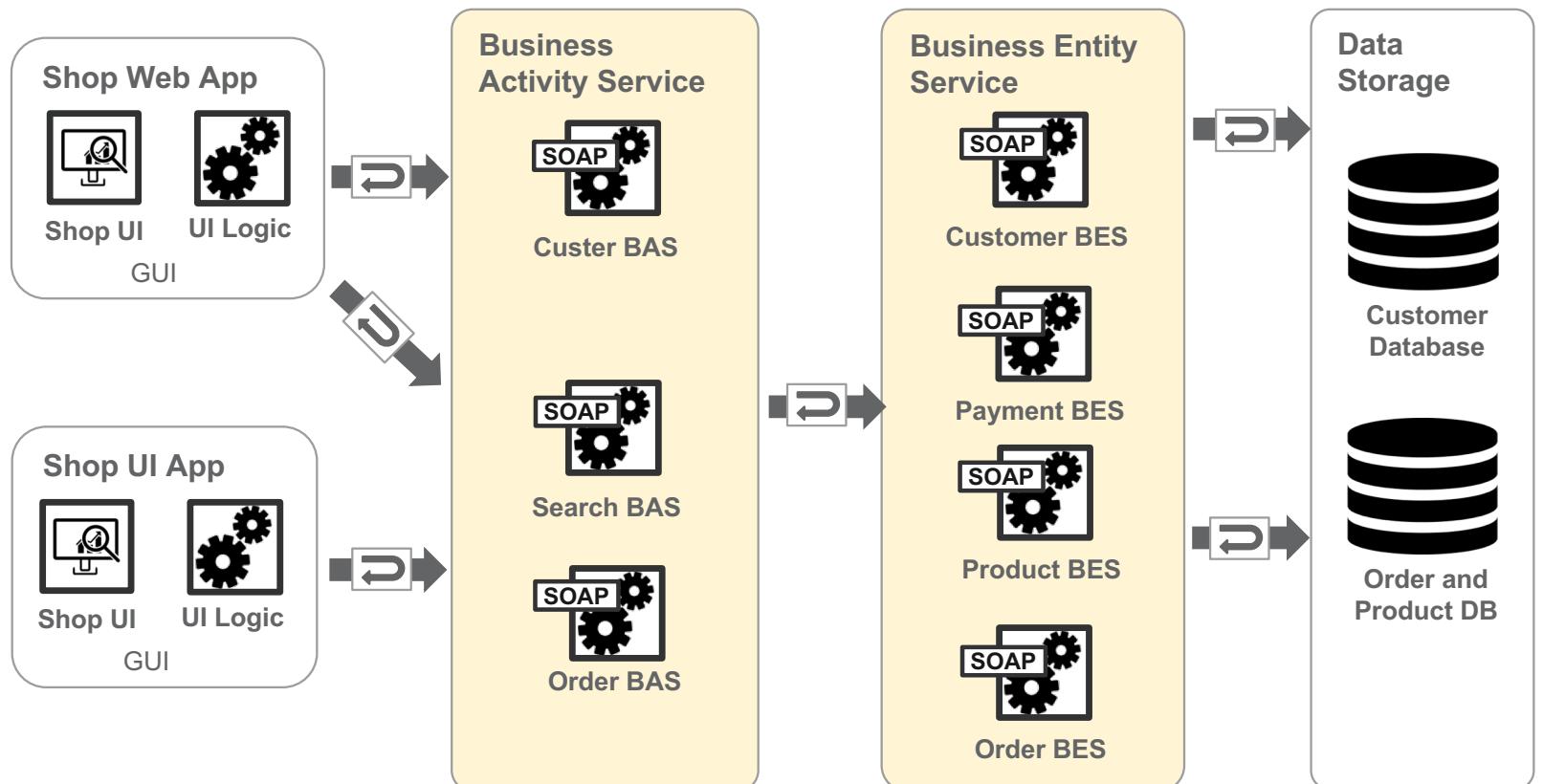
■ SOA Approach

Contract-first
Web Services

Technical layers
offer their own
interfaces

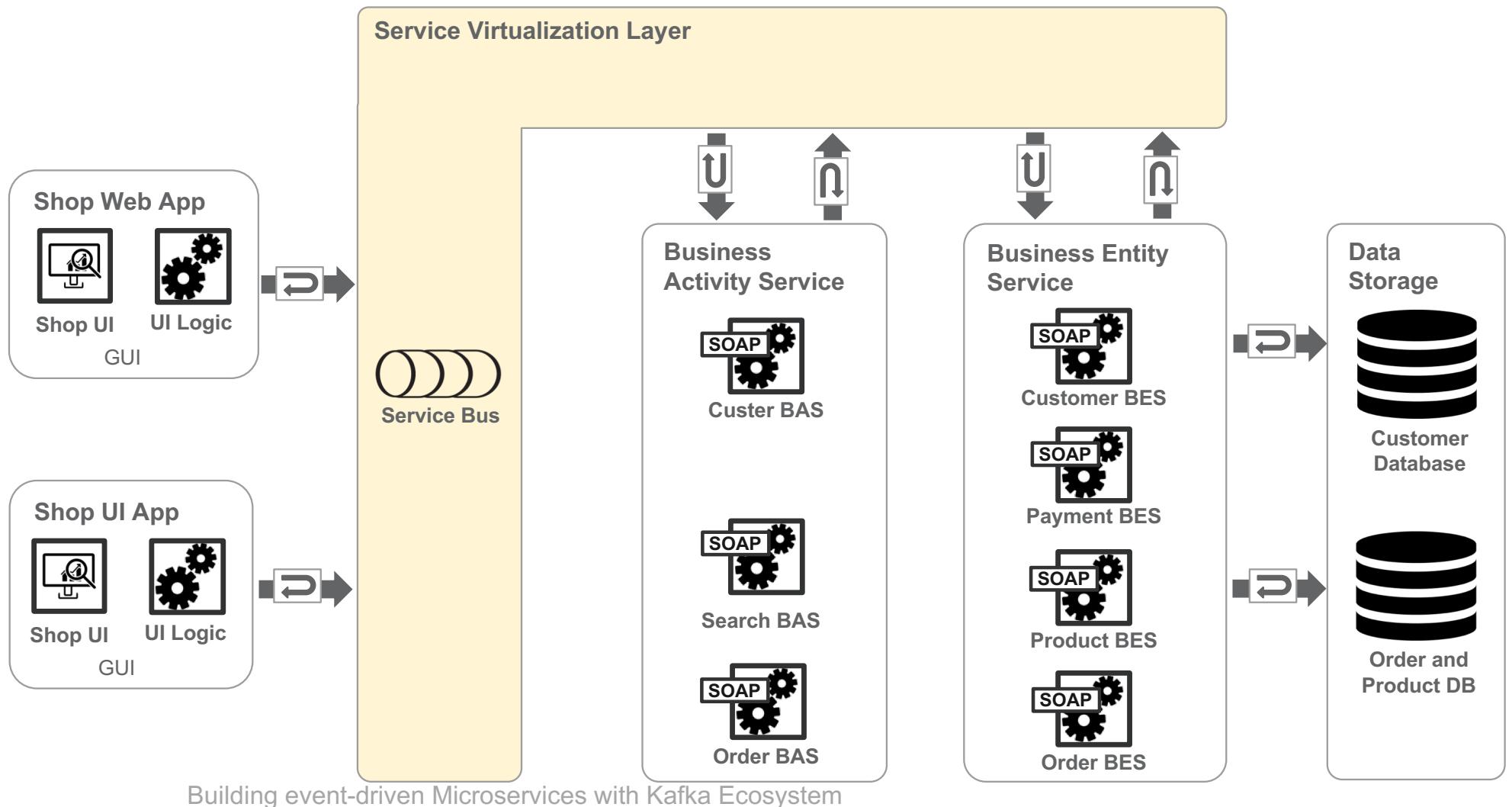
Reuse on each
level

Lower layer
often wraps
legacy code



Building event-driven Microservices with Kafka Ecosystem

Virtualized SOA Approach



What are Microservices?

Building event-driven Microservices with Kafka Ecosystem



■ What are Microservices?

Tightly Scoped behind clear interfaces

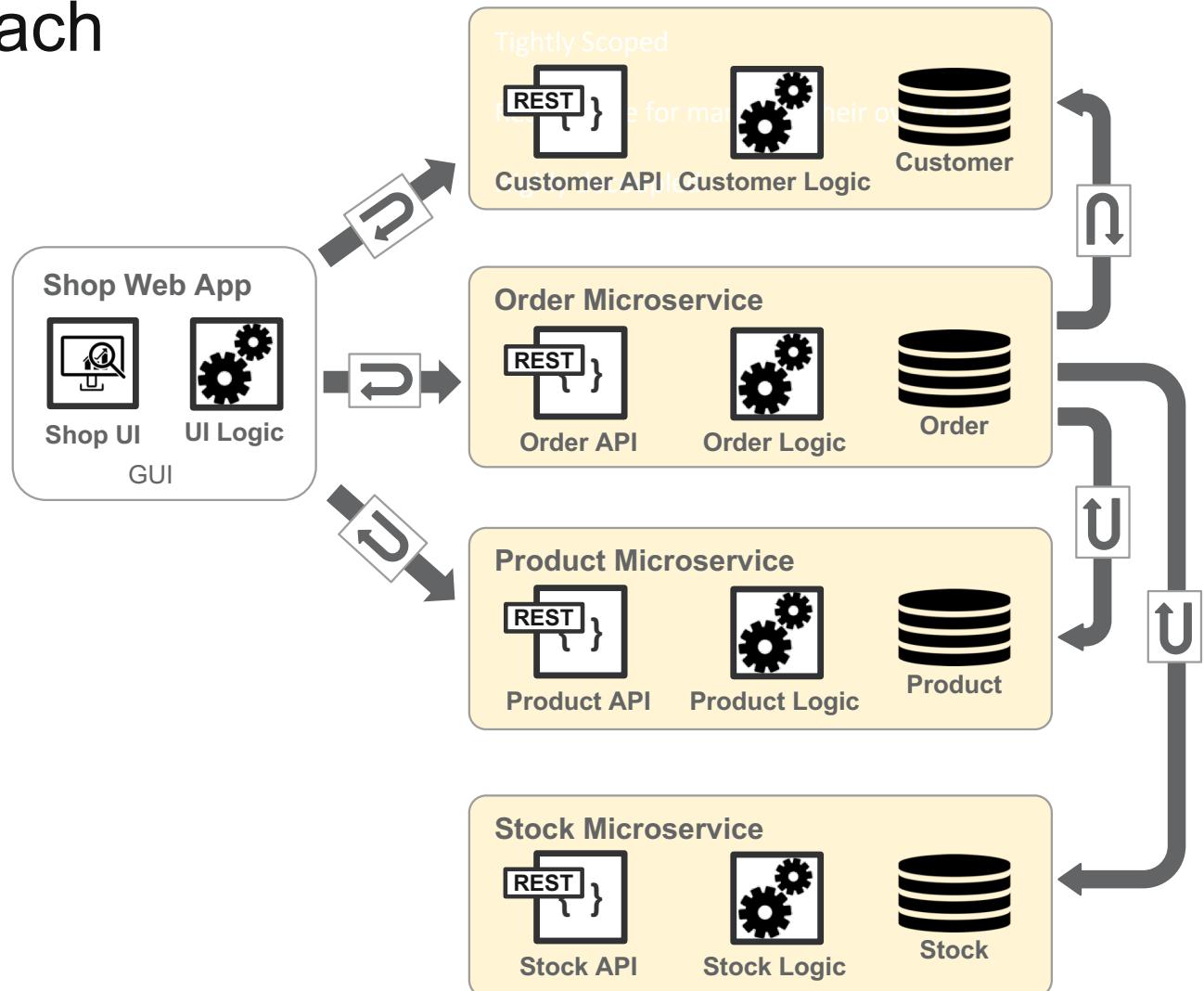
Responsible for managing their own data (not necessarily the infrastructure)

Should be highly decoupled

Independently deployable, self-contained and autonomous

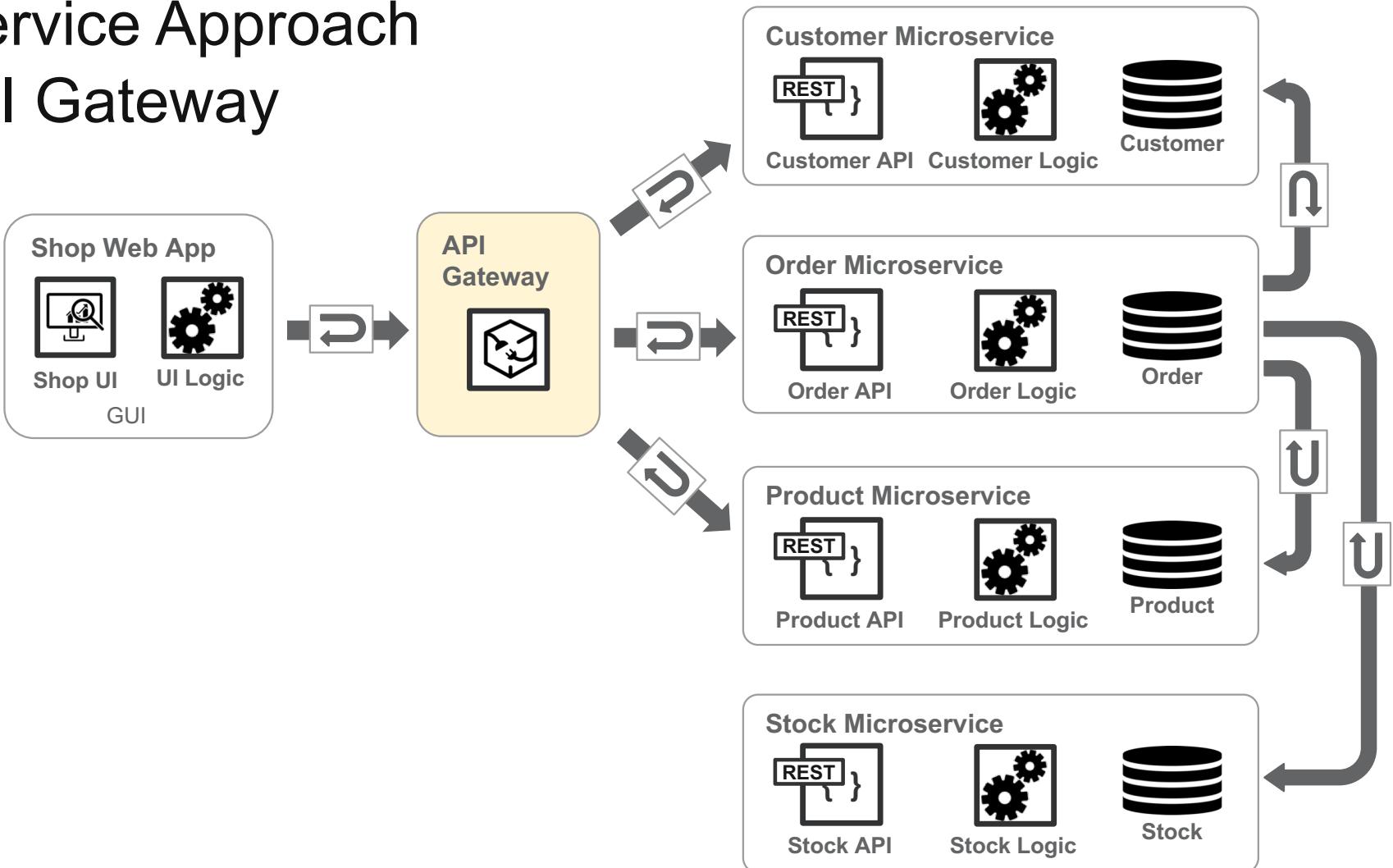
SOA done right ?!

■ Microservice Approach



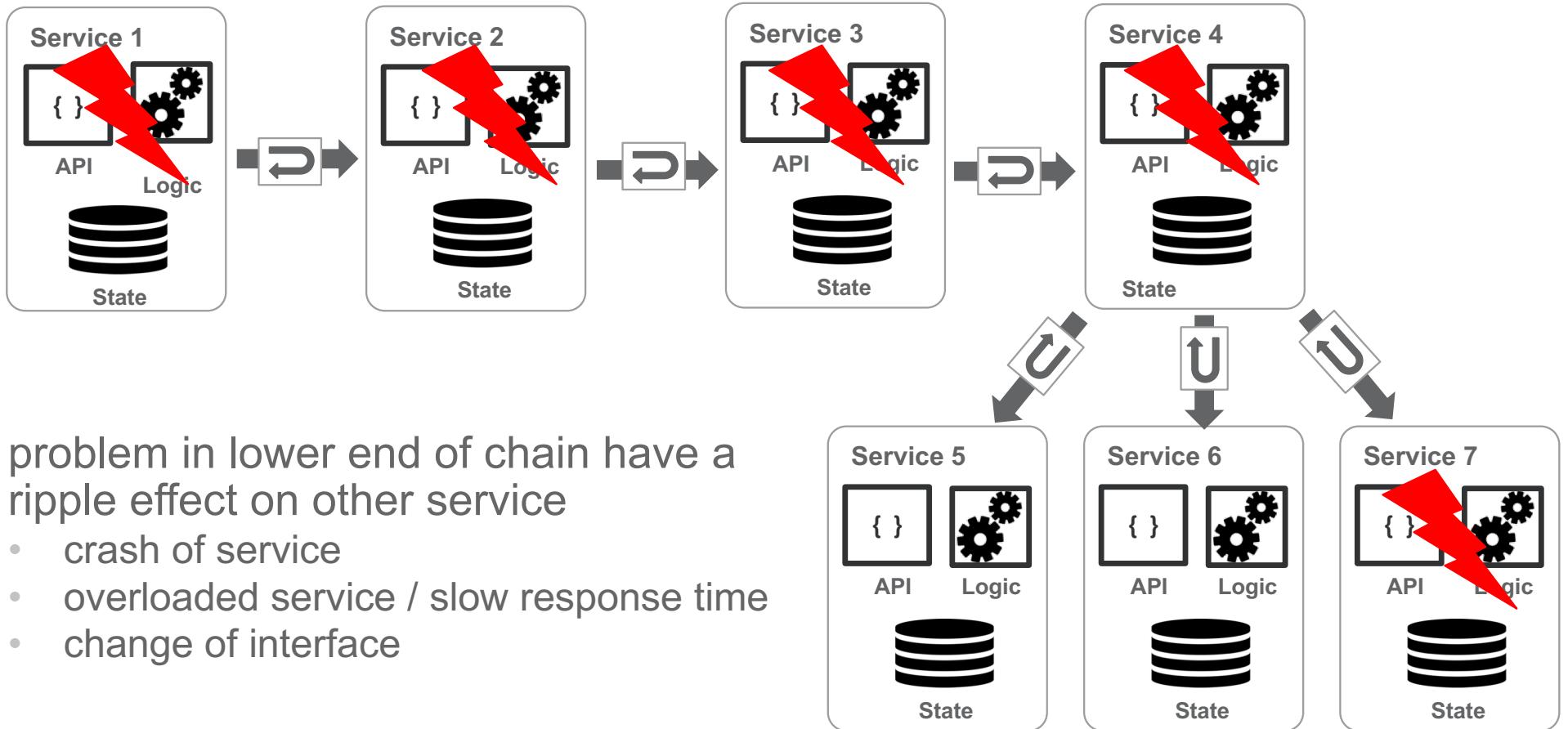
Building event-driven Microservices with Kafka Ecosystem

■ Microservice Approach with API Gateway



Building event-driven Microservices with Kafka Ecosystem

■ Synchronous World of Request-Response leads to tight, point-to-point couplings

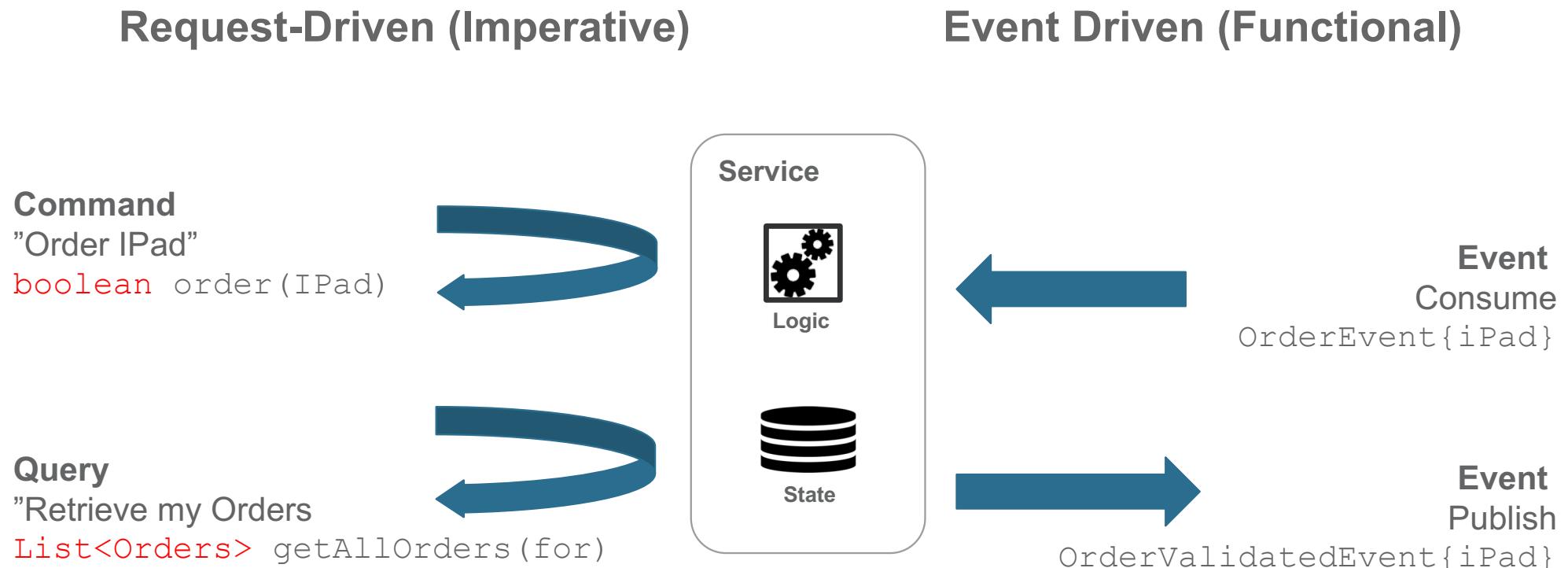


Why not Event-Driven?

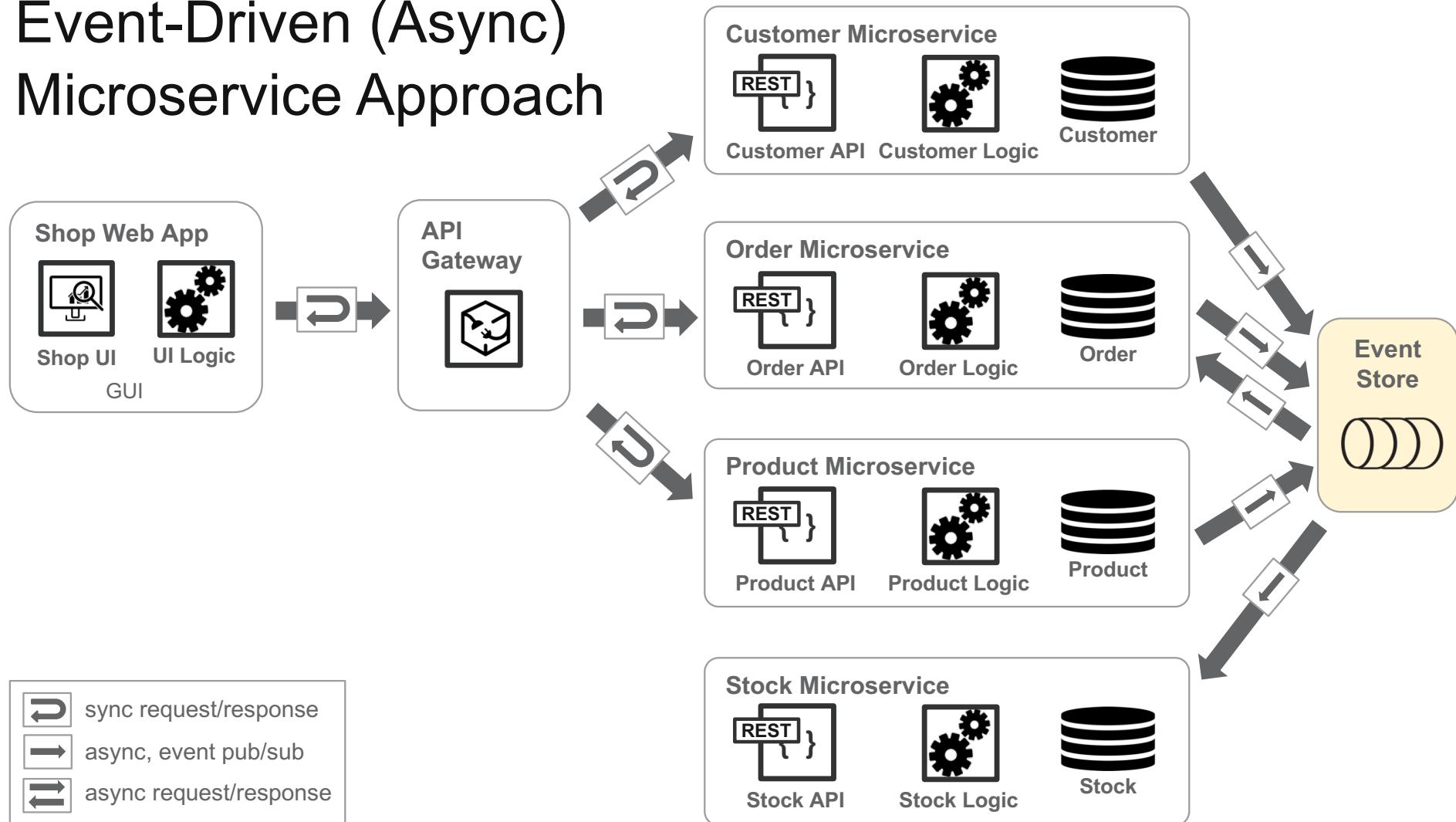
Building event-driven Microservices with Kafka Ecosystem



■ 3 mechanisms through which services interact



■ Event-Driven (Async) Microservice Approach



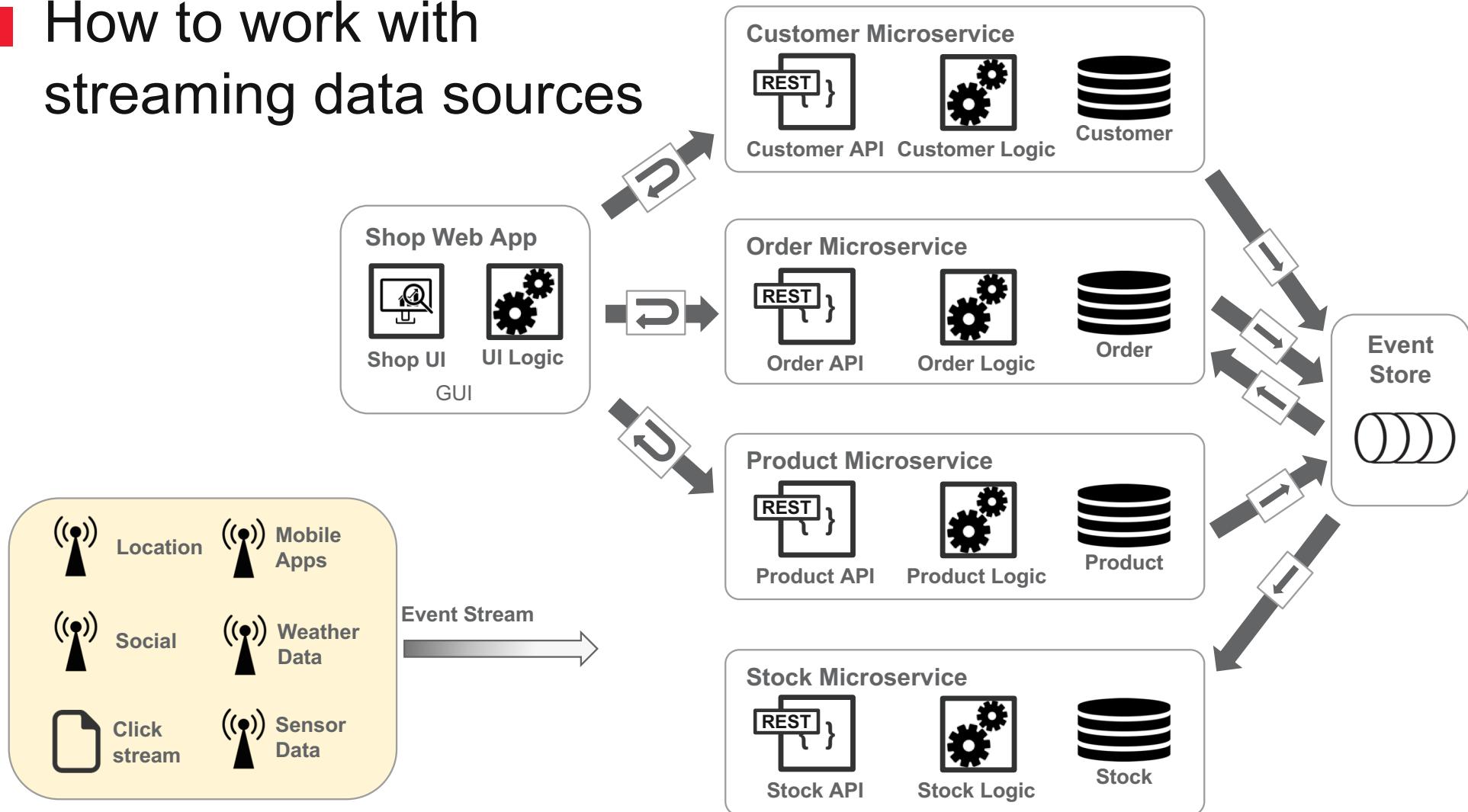
Building event-driven Microservices with Kafka Ecosystem

What about streaming sources?

Building event-driven Microservices with Kafka Ecosystem

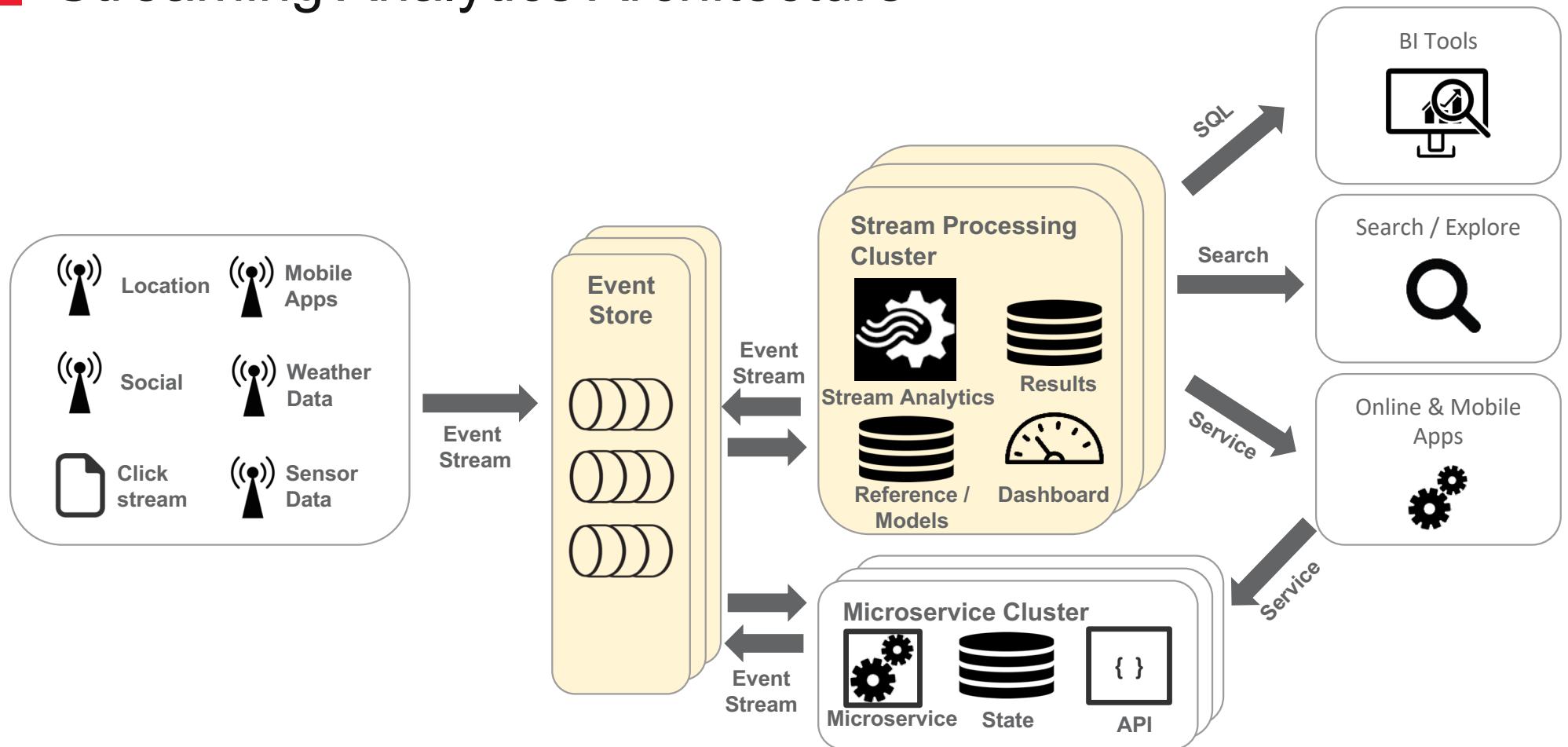
trivadis
makes **IT** easier. 

■ How to work with streaming data sources



Building event-driven Microservices with Kafka Ecosystem

■ Streaming Analytics Architecture



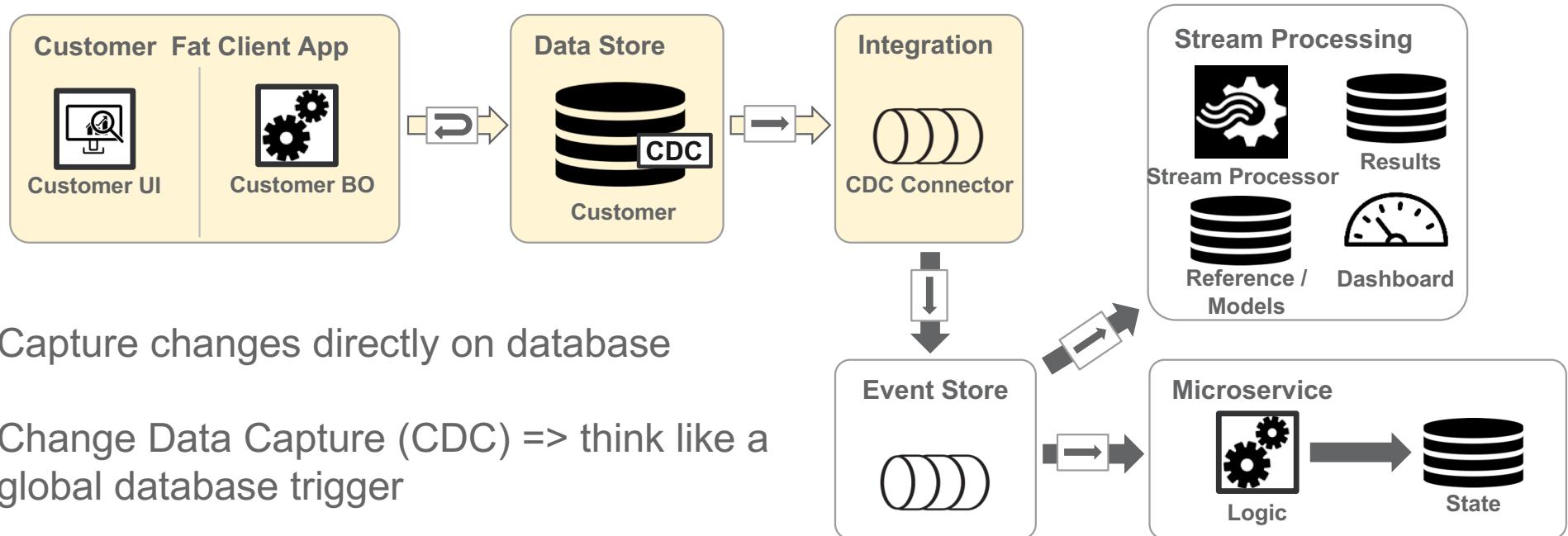
Building event-driven Microservices with Kafka Ecosystem

What about integrating legacy applications?

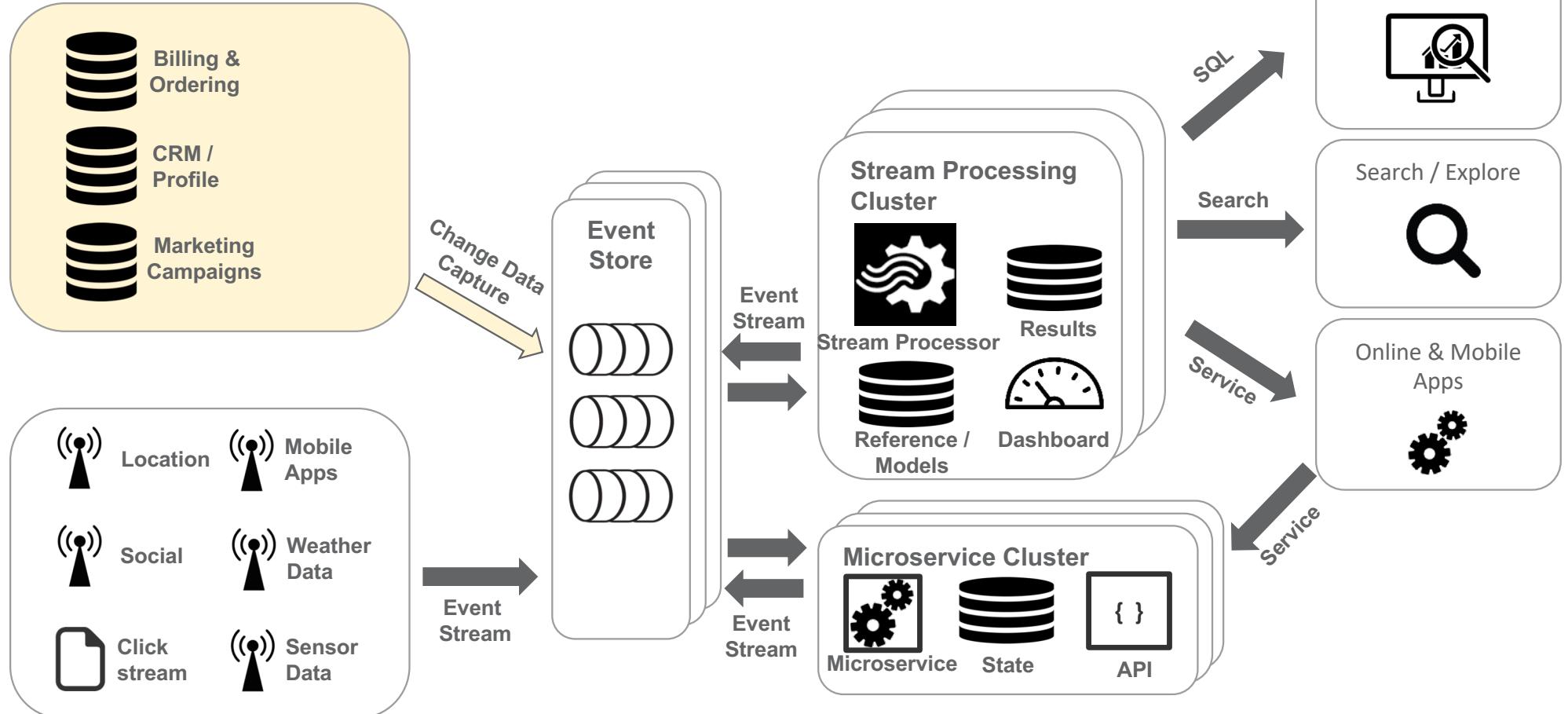
Building event-driven Microservices with Kafka Ecosystem



■ Integrate existing systems through CDC



■ Integrate existing systems through CDC



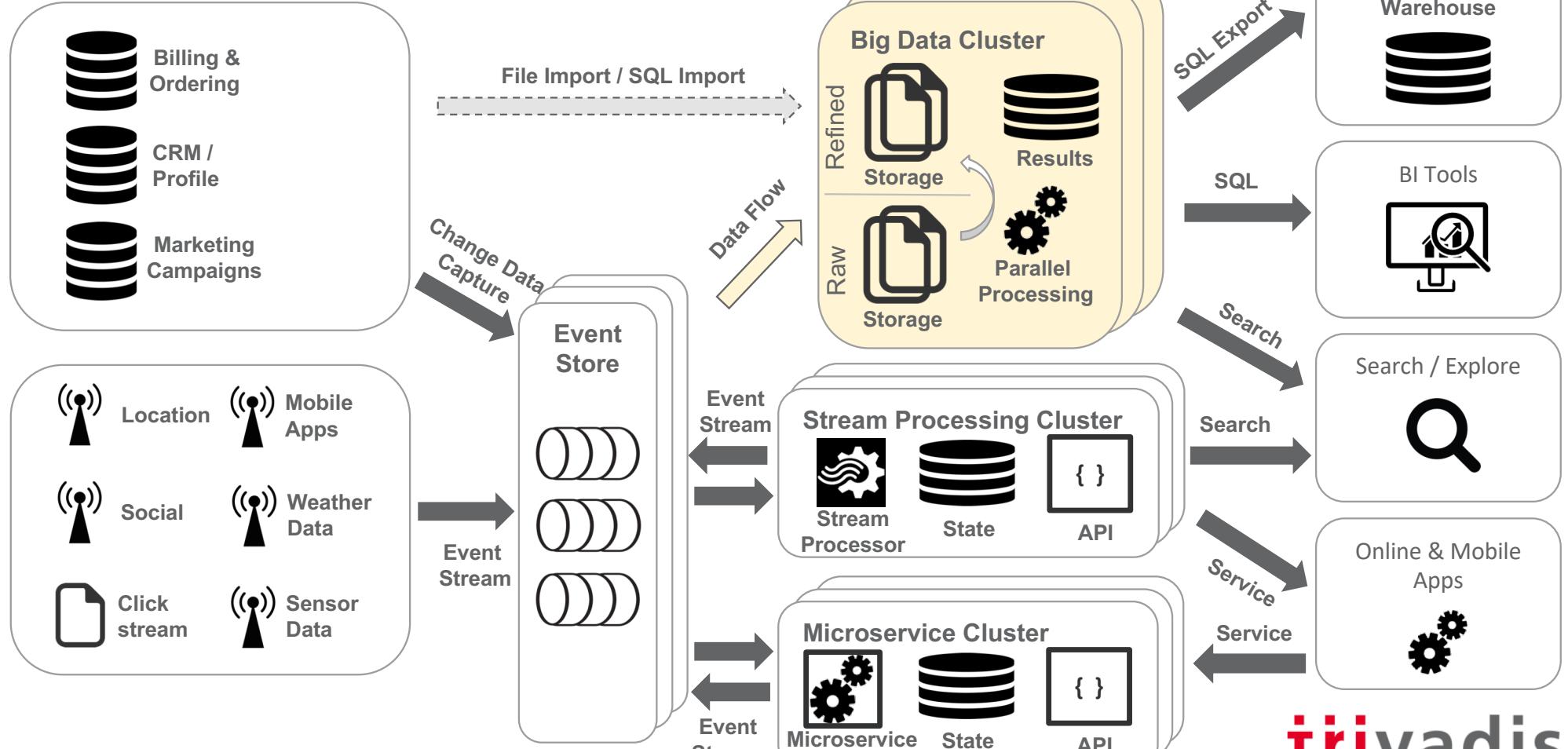
Building event-driven Microservices with Kafka Ecosystem

What about (historical) data analytics?

Building event-driven Microservices with Kafka Ecosystem



■ Streaming & (Big) Data Analytics Architecture



Building event-driven Microservices with Kafka Ecosystem

Why Kafka for Event-Driven Microservices?

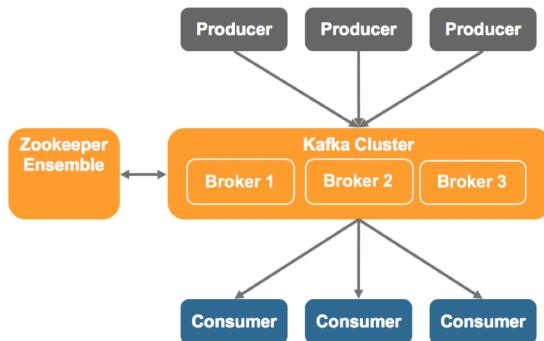
Building event-driven Microservices with Kafka Ecosystem



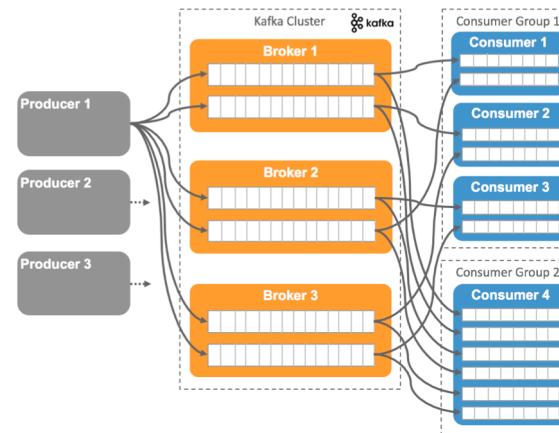
Apache Kafka – A Streaming Platform



High-Level Architecture



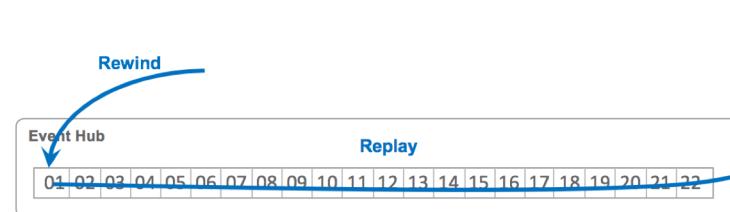
Scale-Out Architecture



Distributed Log at the Core

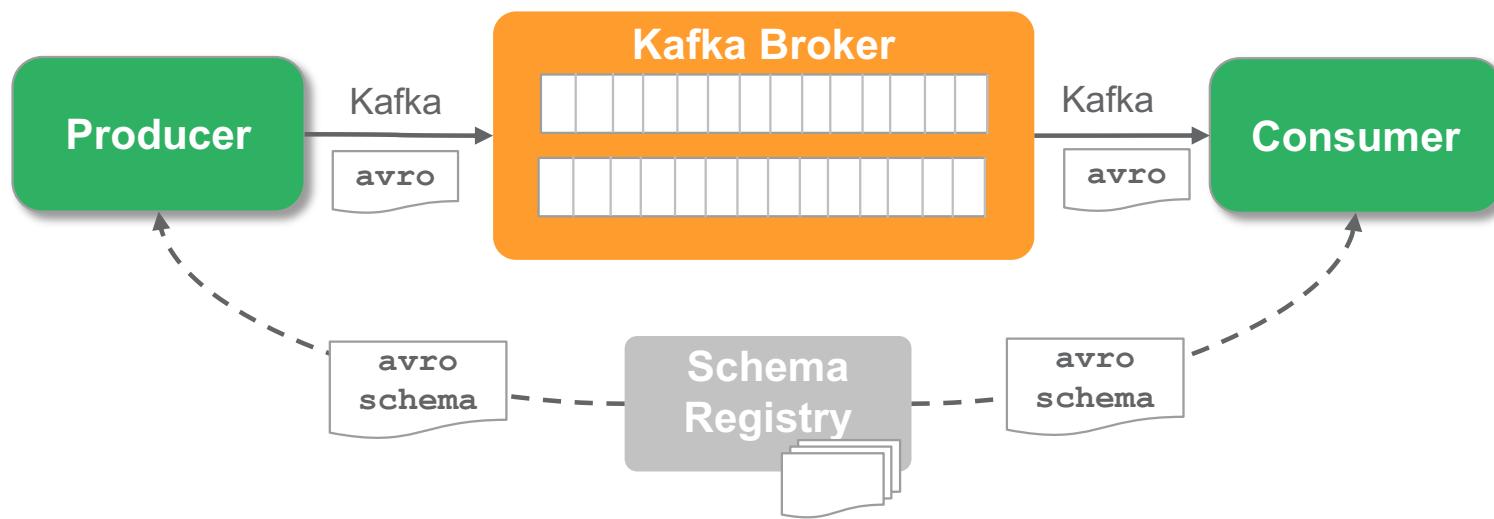


Logs do not (necessarily) forget



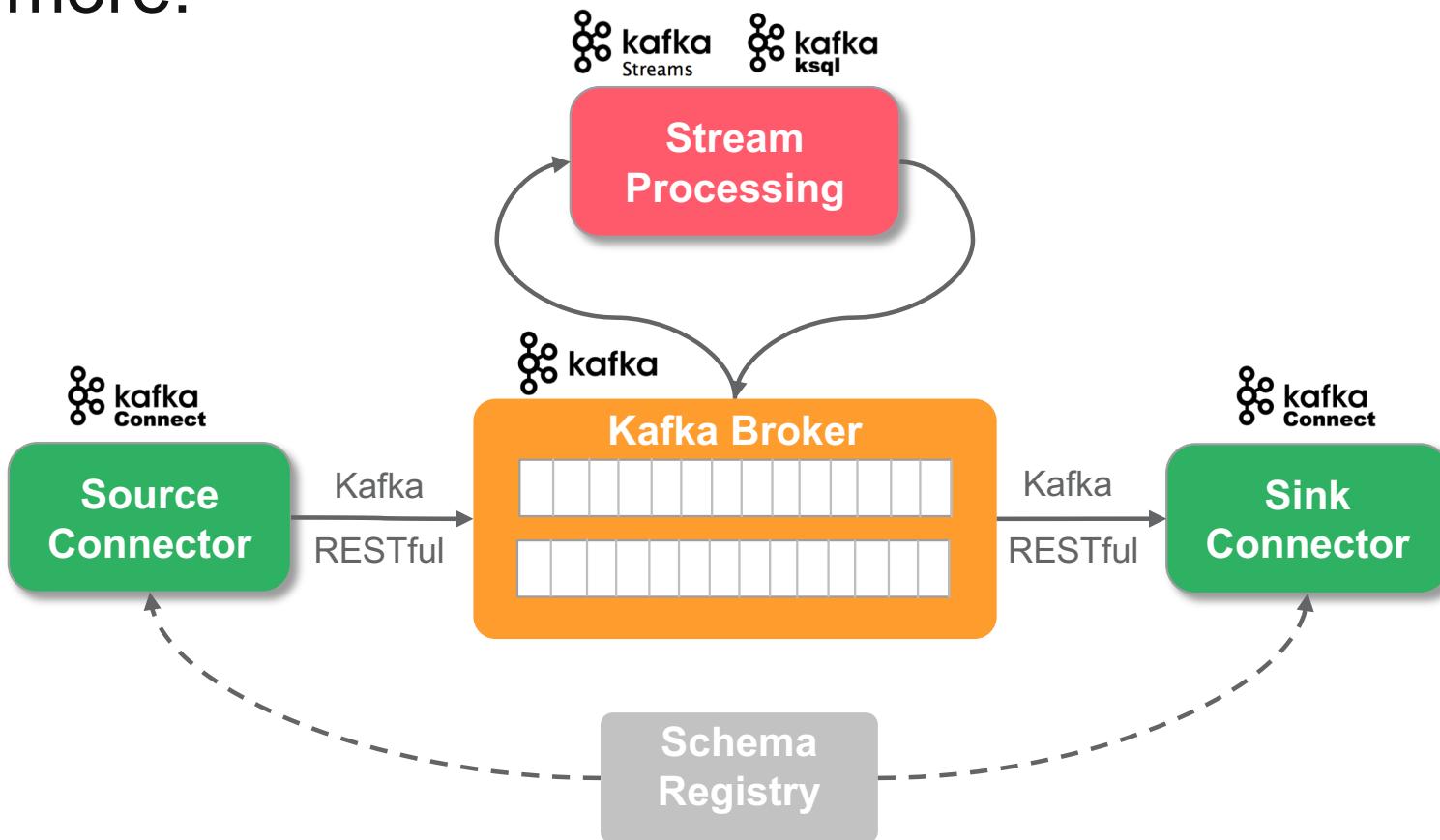
Building event-driven Microservices with Kafka Ecosystem

■ Apache Kafka – Schema Registry



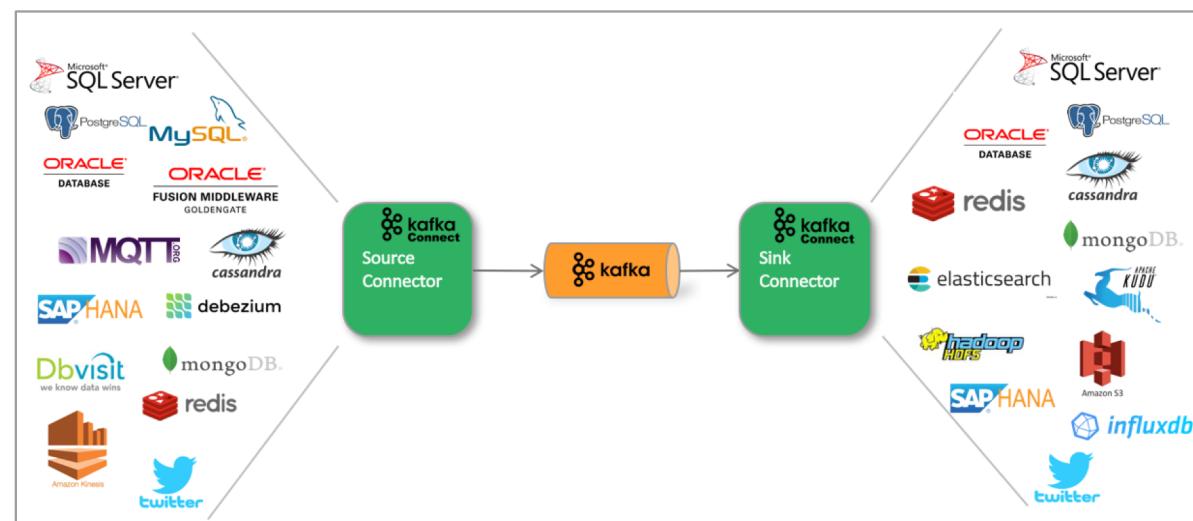
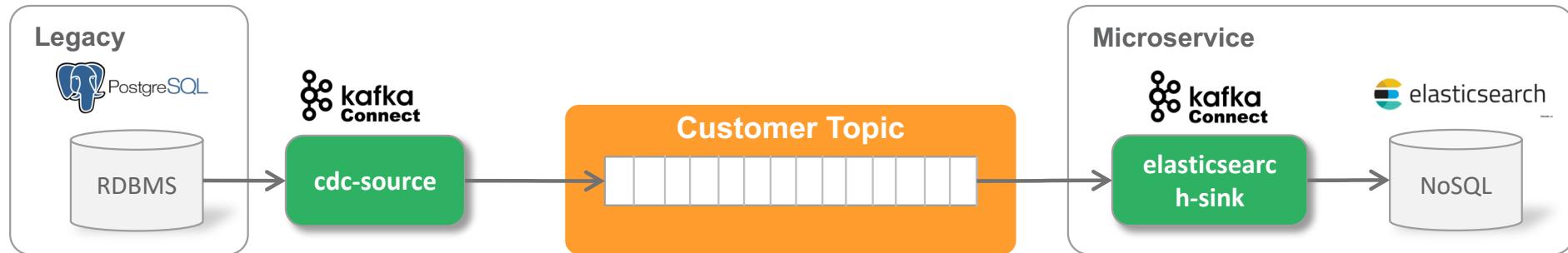
Building event-driven Microservices with Kafka Ecosystem

Apache Kafka – scalable message processing and more!



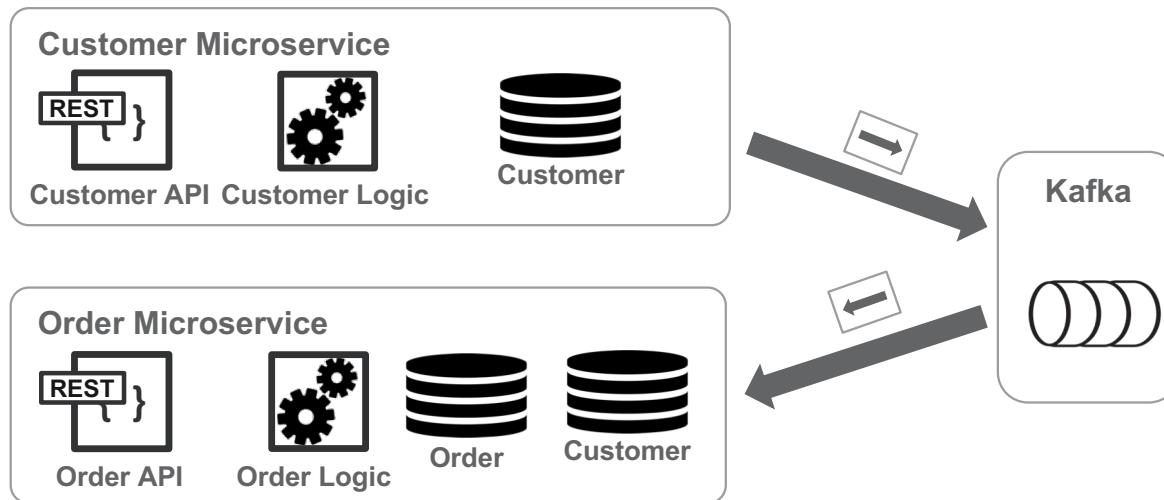
Building event-driven Microservices with Kafka Ecosystem

■ Change Data Capture (CDC)



Building event-driven Microservices with Kafka Ecosystem

Demo

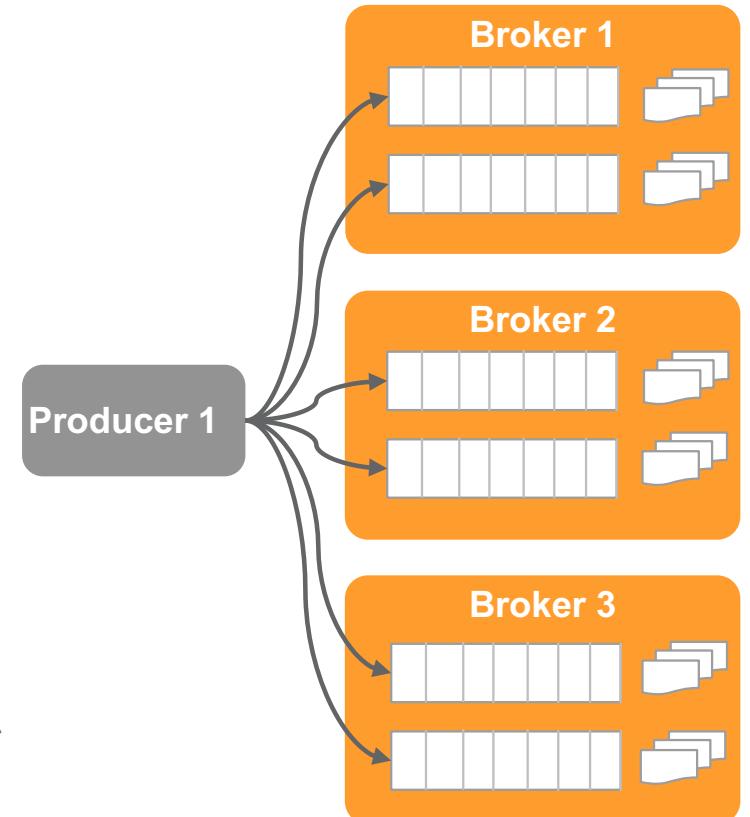


Building event-driven Microservices with Kafka Ecosystem

■ Hold Data for Long-Term – Data Retention

1. Never
2. Time based (TTL)
`log.retention.{ms | minutes | hours}`
3. Size based
`log.retention.bytes`
4. Log compaction based
(entries with same key are removed):

```
kafka-topics.sh --zookeeper zk:2181 \  
    --create --topic customers \  
    --replication-factor 1 \  
    --partitions 1 \  
    --config cleanup.policy=compact
```



■ Topic Viewed as Event Stream or State Stream (Change Log)

Event Stream

2015-10-02	11,Peter,Muster,3010,Berne
2016-10-04	12,Paul,Steffen,8001,Zurich
2016-12-02	21,Lisa,Meier,3043,Ittigen
2017-05-03	11,Peter,Muster,3015,Berne
2017-05-03	21,Lisa,Steffen,8001,Zurich
2017-07-03	11,Peter,Muster,3052,Zollikofen

State Stream (Change Log Stream)

2015-10-02	11,Peter,Muster,3010,Berne
2016-10-04	12,Paul,Steffen,8001,Zurich
2016-12-02	21,Lisa,Meier,3043,Ittigen
2017-05-03	11,Peter,Muster,3015,Berne
2017-05-03	21,Lisa,Steffen,8001,Zurich
2017-07-03	11,Peter,Muster,3052,Zollikofen

Keep Topics in Compacted Form

0	1	2	3	4	5	6	7	8	9	10	11
K1	K2	K1	K1	K3	K2	K4	K5	K5	K2	K6	K2
V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	

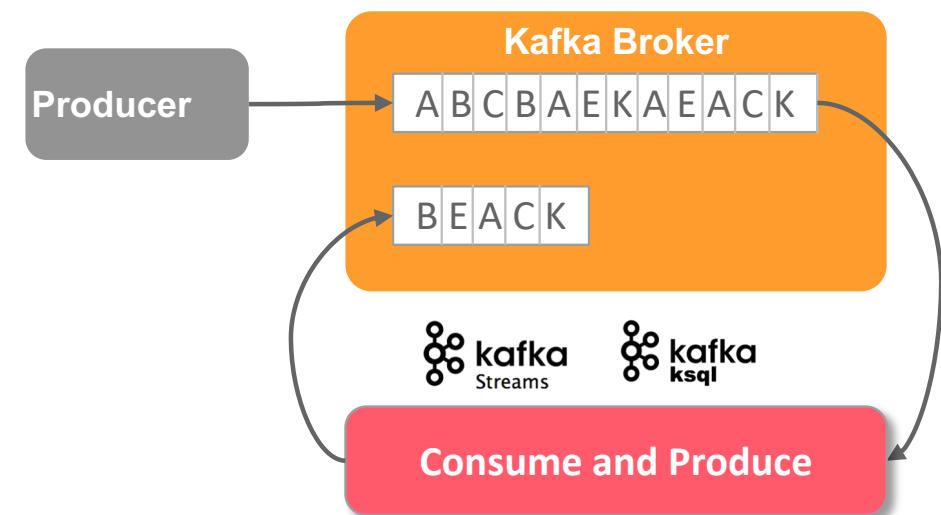
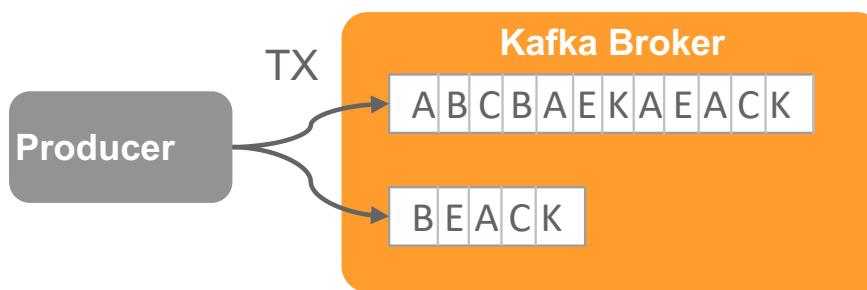
Compaction

Offset	3	4	6	8	9	10
Key	K1	K3	K4	K5	K2	K6
Value	V4	V5	V7	V9	V10	V11

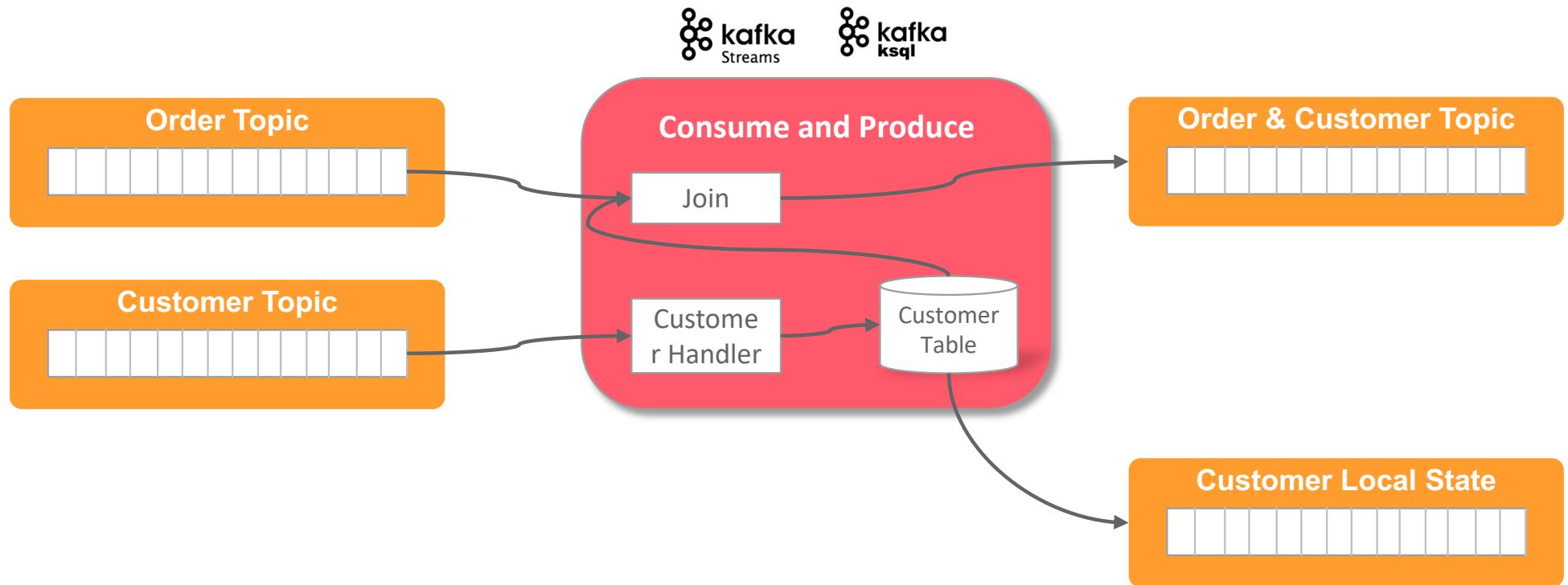


■ Keep Topics both in Original and Compacted Form

OR



Enrich Stream with Static Data with Kafka Streams

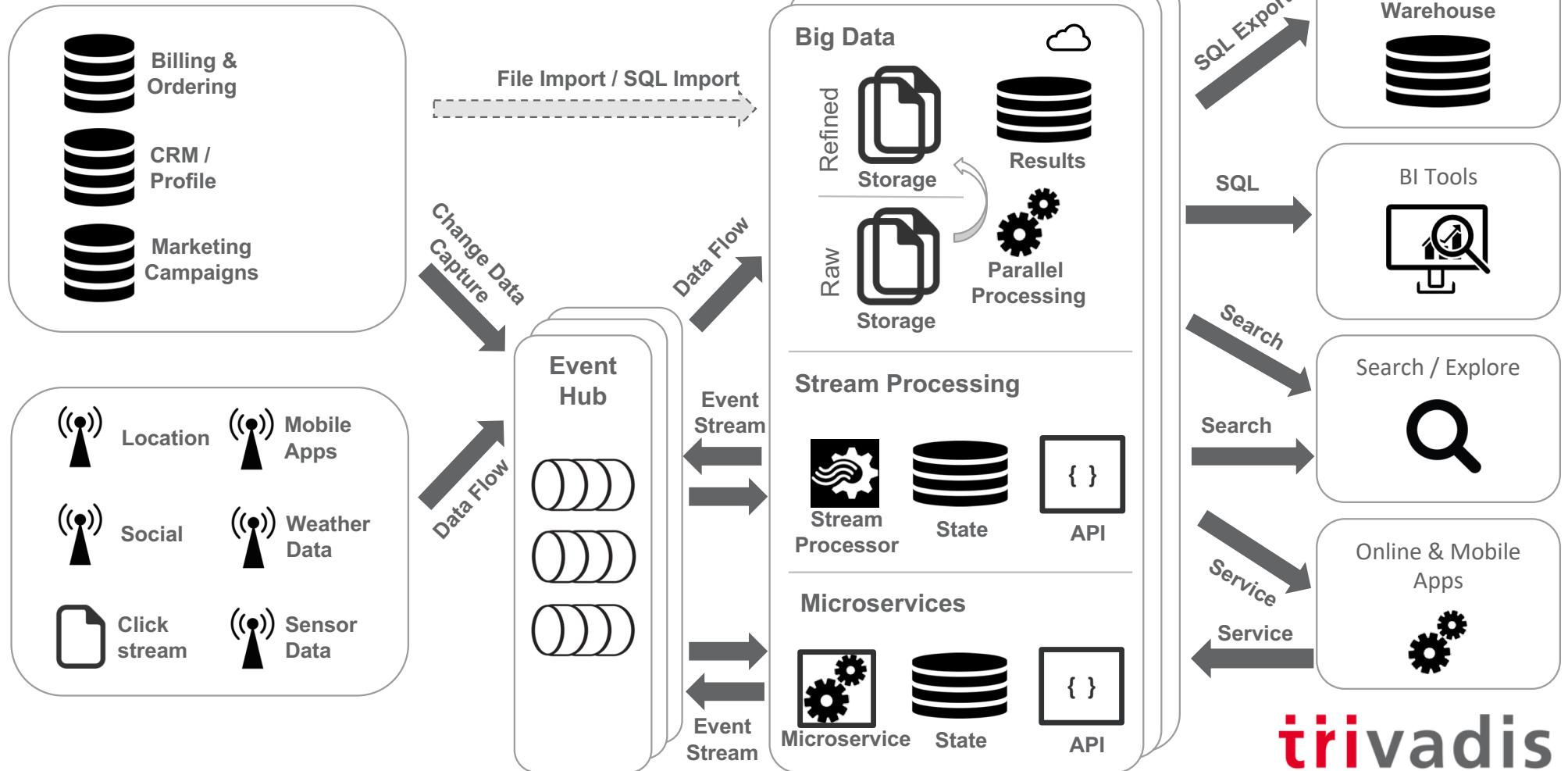


Summary

Building event-driven Microservices with Kafka Ecosystem

trivadis
makes **IT** easier. 

■ Summary



■ Summary

- service autonomy is key in a Microservices Architecture!
- not all communication need to be synchronous => separate into
 - commands
 - events
 - queries
- Kafka is well suited as an event broker / event store
 - brings many more interesting features beyond just “message passing”

■ References

Microservices Blog Series, Ben Stopford, Confluent:

- <https://www.confluent.io/blog/tag/microservices>

Apache Kafka for Microservices: A Confluent Online Talk Series:

- <https://www.confluent.io/landing-page/microservices-online-talk-series/>

Turning the database inside-out with Apache Samza, Martin Kleppmann, Con

- <https://www.confluent.io/blog/turning-the-database-inside-out-with-apache-samza/>

Event sourcing, CQRS, stream processing and Apache Kafka: What's the connection?, Neha Narkhede, Confluent:

- <https://www.confluent.io/blog/event-sourcing-cqrs-stream-processing-apache-kafka-whats-connection/>

Immutability Changes Everything, Pat Helland, Salesforce:

- http://cidrdb.org/cidr2015/Papers/CIDR15_Paper16.pdf

Commander: Better Distributed Applications through CQRS and Event Sourcing, Bobby Calderwood:

- <https://www.youtube.com/watch?v=B1-gS0oEtYc>

Technology on its own won't help you. You need to know how to use it properly.



Building event-driven Microservices with Kafka Ecosystem

trivadis
makes **IT** easier. 