

# API SUMMIT

## Event Driven APIs



~~Building~~ Real-Time Interfaces

# Defining



Axel Grosse

Head of API Management  
&  
Innovation Lead

Twitter: @apiaxl

LinkedIn: axelgrosse

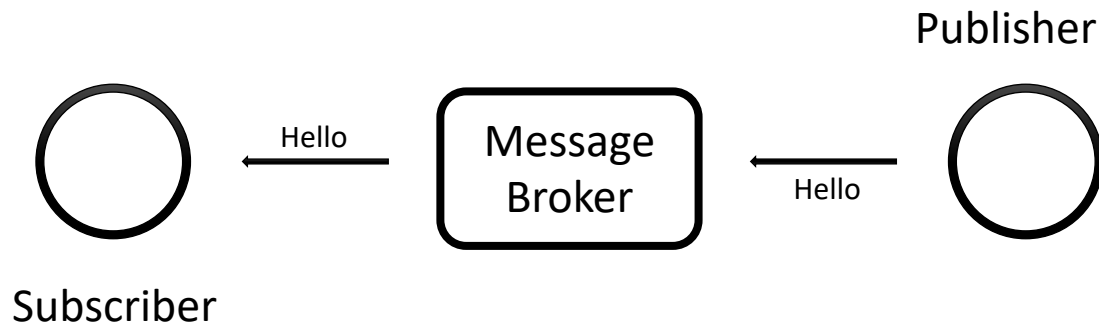
# Agenda

- Event Driven Grundlagen
- Events APIs Beschreiben
- Beispiele für Event APIs

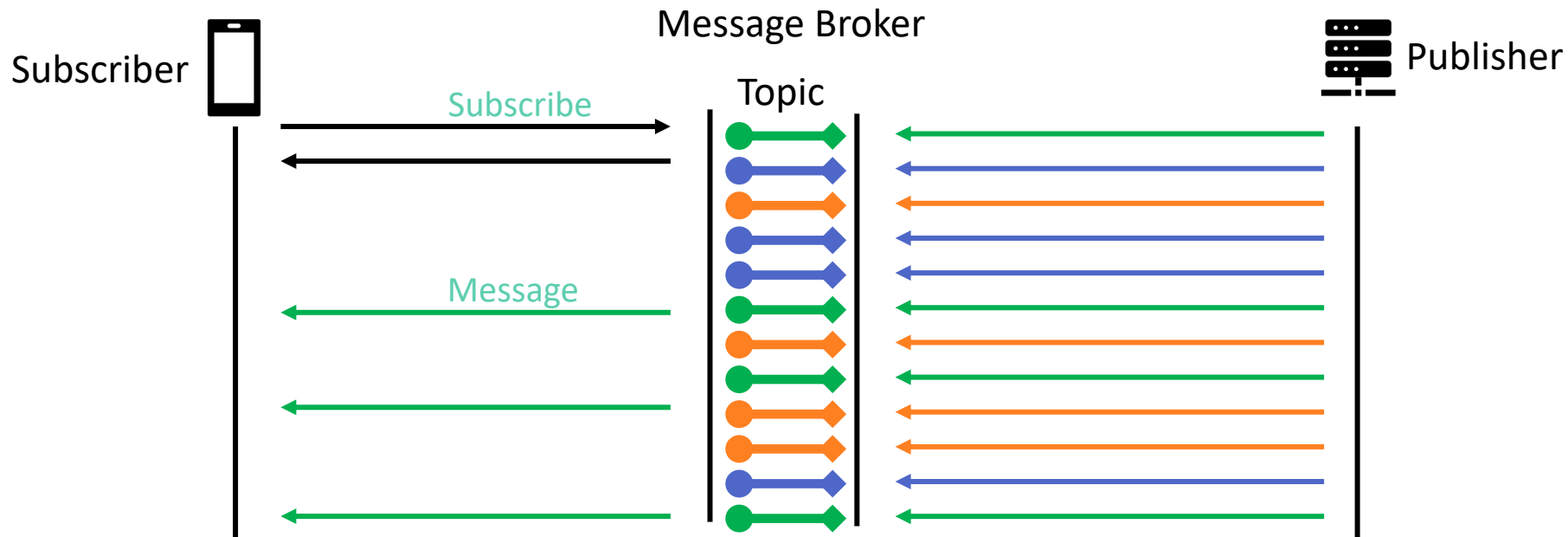


# Der Kern

## Event Driven Basics



# Der Kern



# MQ ... Da war doch was ??

Apache ActiveMQ

Apache Kafka

Financial Fusion Message Broker (Sybase)

Fuse Message Broker (enterprise ActiveMQ)

IBM MQ

JBoss Messaging (JBoss)

Microsoft BizTalk Server (Microsoft)

Oracle Message Broker (Oracle Corporation)

SAP PI (SAP AG)

TIBCO Enterprise Message Service

WSO2 Message Broker

AWS Simple Queue Service (SQS)

Microsoft Azure Service Bus (Microsoft)

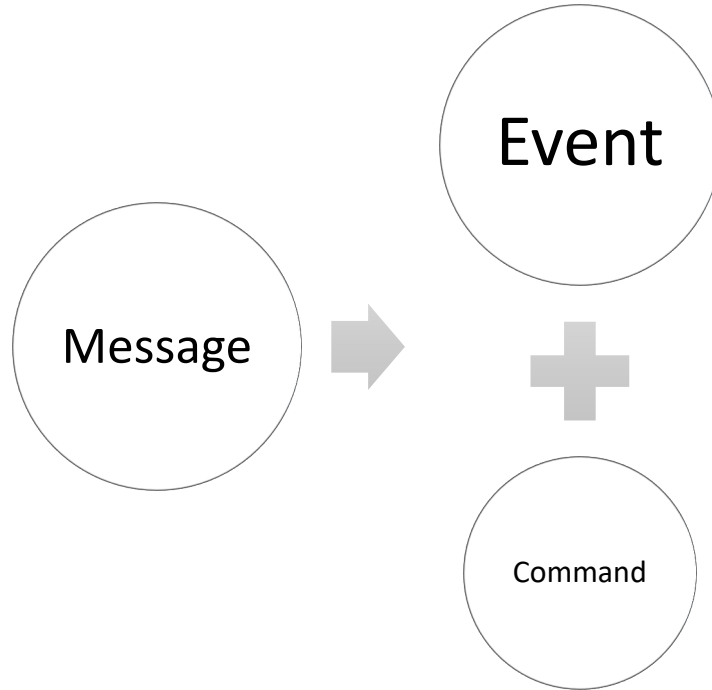
Oracle Message Broker (Oracle Corporation)

RabbitMQ

Redis

Solace PubSub+

# Warum Event und nicht Message



# Types of Events

Event Notification

Event-Sourcing

Event-Carried State  
Transfer

CQRS

Command Query Responsibility Segregation

Source: <https://martinfowler.com/articles/201701-event-driven.html>



# Warum reicht REST nicht?

## REST

- Schneller Austausch
- Beide Teilnehmer müssen online sein
- REST ist die Basis des Web
- Inhalt kann verloren gehen

## Events

- Einfacher um Prozesse zu beschreiben
- Event beschreiben Momente in denen sich etwas ändert
- Inhalt wird immer geliefert

## SYNCHRON

## ASYNCHRON

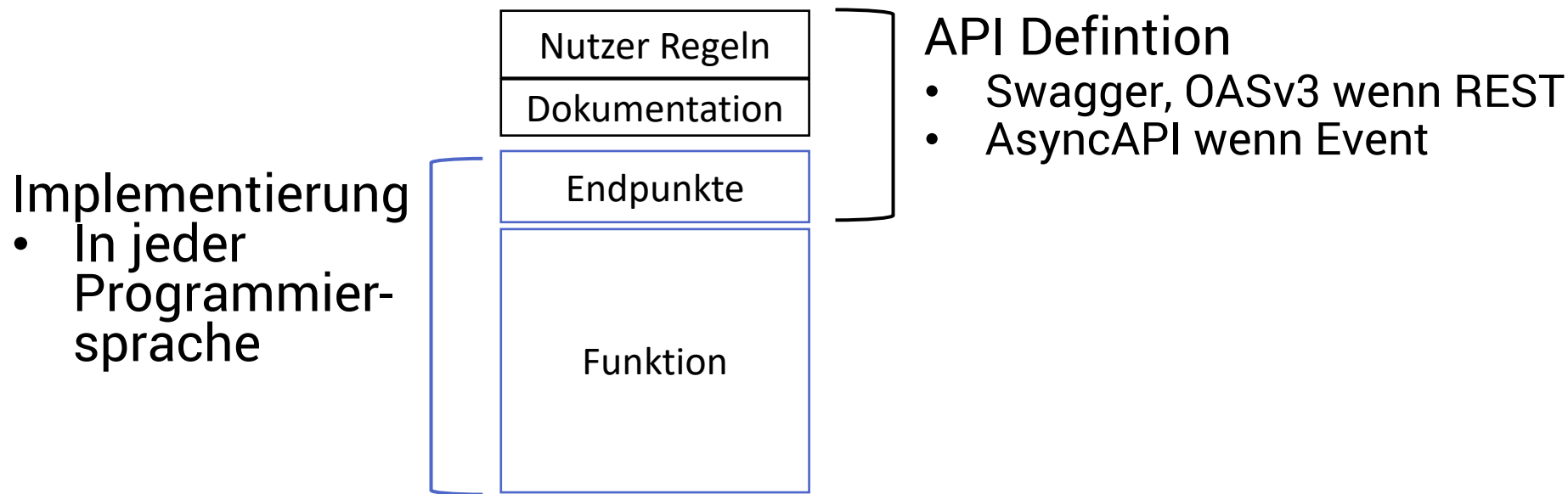
**Manche Prozesse lassen sich mit Events besser beschreiben**

# Warum Event Driven API ?

Event Driven APIs komplettieren REST APIs  
in der Aufgabe Integration über web fähige Technologien zu erlauben.

# Who is Who in the Zoo

API, Microservices, Swagger, OASv3, AsyncAPI, etc.



# Async API

## Ein neuer Standard für die Beschreibung von Event APIs

The purpose of AsyncAPI is to enable architects and developers to specify the event payload definition, channel name, application/transport headers and protocol– thus fully specifying the application's event-driven interface.

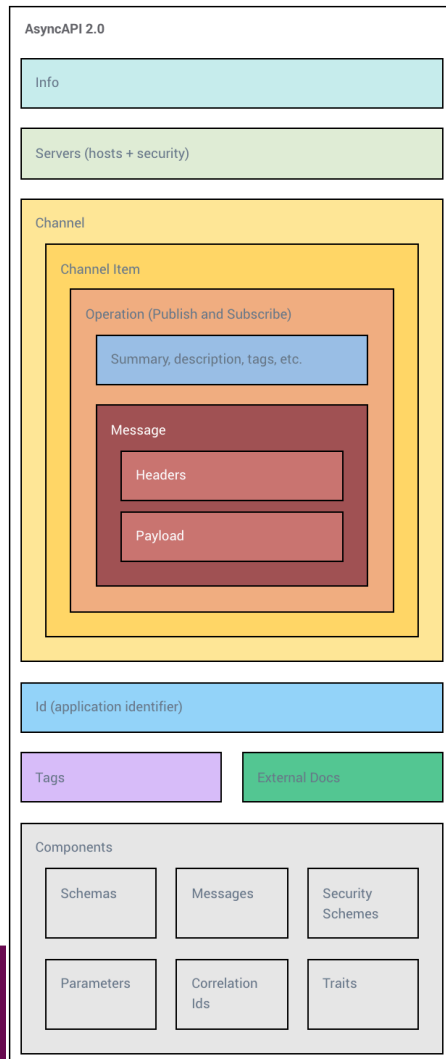
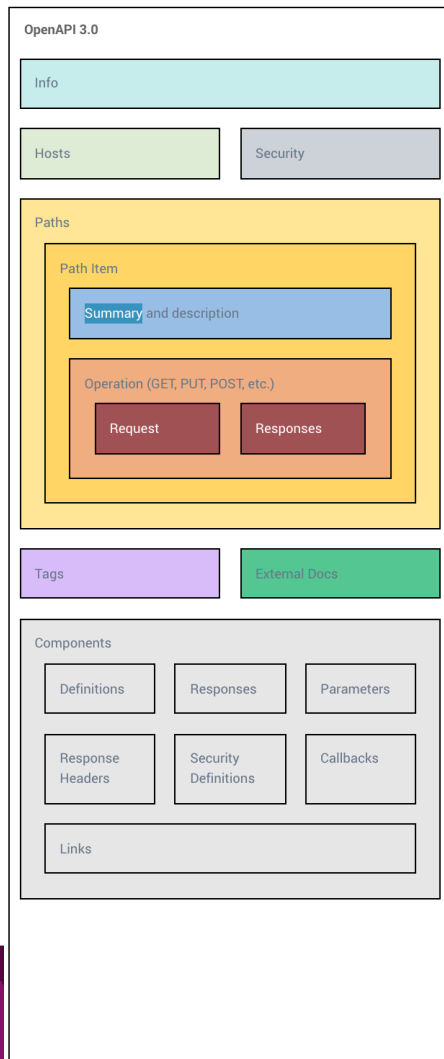
AsyncAPI is a common language for all the different protocols and schema formats, but we need to make it easy for developers to create new message-driven systems, in their programming language of choice.

<https://www.asyncapi.com/>

# Open API vs Async API

Foundation				Enablmant			
	Runtime	Development	Security	Governance	Discovery	Spec	Code Gen
REST	Web Server	Spring, Restlet, JAX-RS, node.js	OAuth OpenIDC	Apigee, Axway, Mulesoft	Istio/ Envoy, Netflix Eureka	Swagger Open API	Swagger.io
Event Driven	Broker	Spring Cloud Stream, Paho	OAuth, TLS	TBD	„Event Mesh“	Async API	Async API Generator

# Open API V3



# Async API 2.0

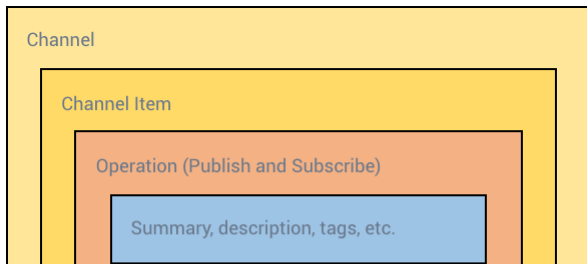
# Elemente von Async API

Info

Servers (hosts + security)

```
2  asyncapi: '2.0.0'
3  info:
4    title: Streetlights API
5    version: '1.0.0'
6    description: |
7      The Smartylighting Streetlights API allows you
8      to remotely manage the city lights.
9    license:
10     name: Apache 2.0
11     url: 'https://www.apache.org/licenses/LICENSE-2.0'
12
13  servers:
14    mosquitto:
15      url: mqtt://test.mosquitto.org
16      protocol: mqtt
17
```

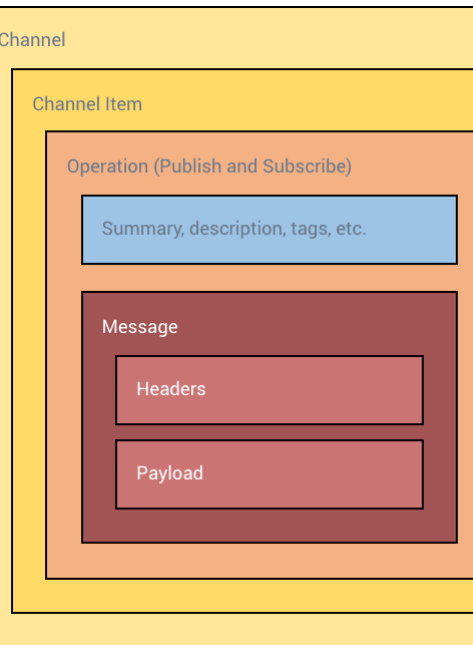
# Elemente von Async API



```
1 | channels:
2 |   light/measured:
3 |     publish:
4 |       summary: Inform about environmental lighting conditions for a particular
5 |       operationId: onLightMeasured
```

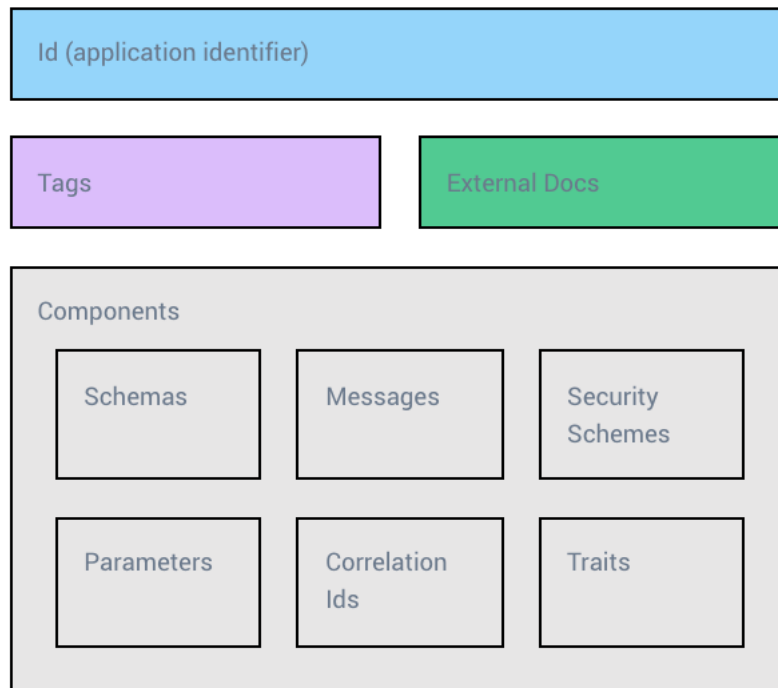


# Elemente von Async API



```
23 message:
24   payload:
25     type: object
26   properties:
27     id:
28       type: integer
29       minimum: 0
30       description: Id of the streetlight.
31   lumens:
32     type: integer
33     minimum: 0
34     description: Light intensity measured in lumens.
35   sentAt:
36     type: string
37     format: date-time
38     description: Date and time when the message was sent.
```

# Elemente. von Async API



# Werkzeuge

Async API Editor

Code Generator Service

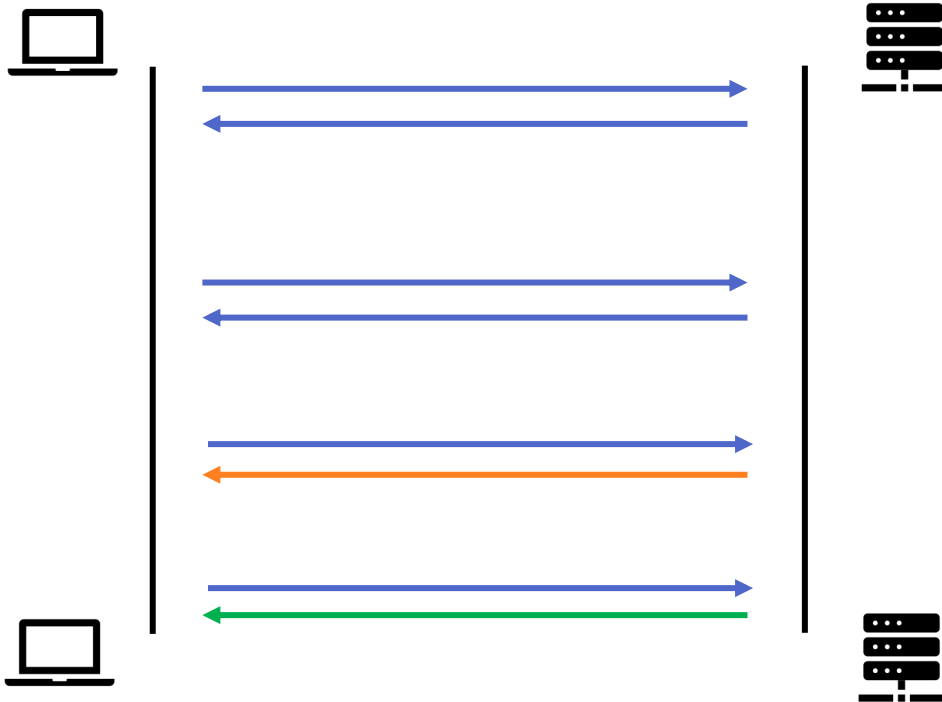
[genaueres findet ihr unter https://github.com/asynccapi](https://github.com/asynccapi)

# Interaktion und API am Beispiel

Wie beschreibe ich welches Protokoll als API

- Long Polling
- Webhook-Callback
- Websockets
- Sever Side Events
- MQTT
- AMQP

# Long Polling



## Pro

- Standard HTTP GET
- Einfach zu implementieren

## Cons

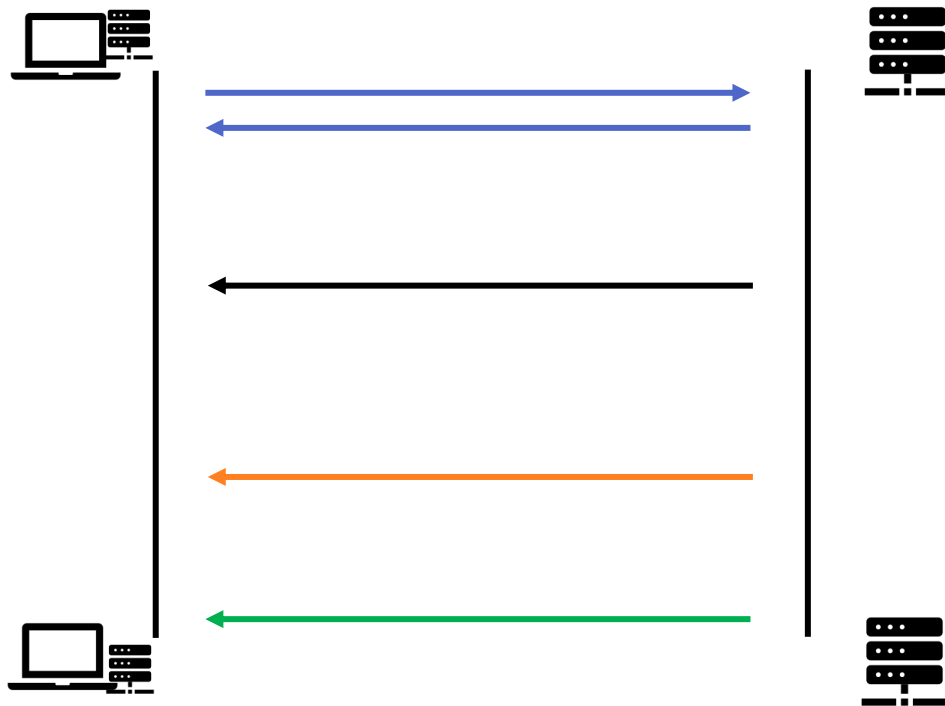
- Kosten Intensiv
- Verpasst meist den Moment

# Long Polling API Definition

Normale OASv3 oder OASv2-Swagger Definition

ohne besondere Einträge für Events

# Webhook-Callback



## Pro

- Standard HTTP POST/PUT
- Effizienter als Long Polling

## Cons

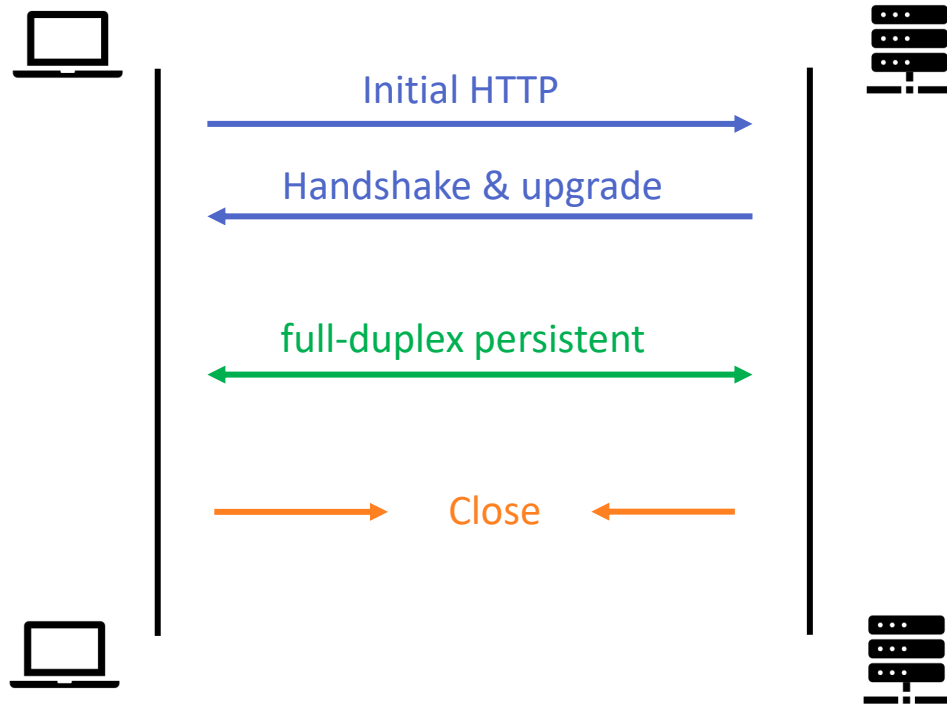
- Aufwänding auf der Client Seite

# Webhook-Callback API Defintion

Normale OASv3 Definition mit Beschreibung  
der Callback URL für Events



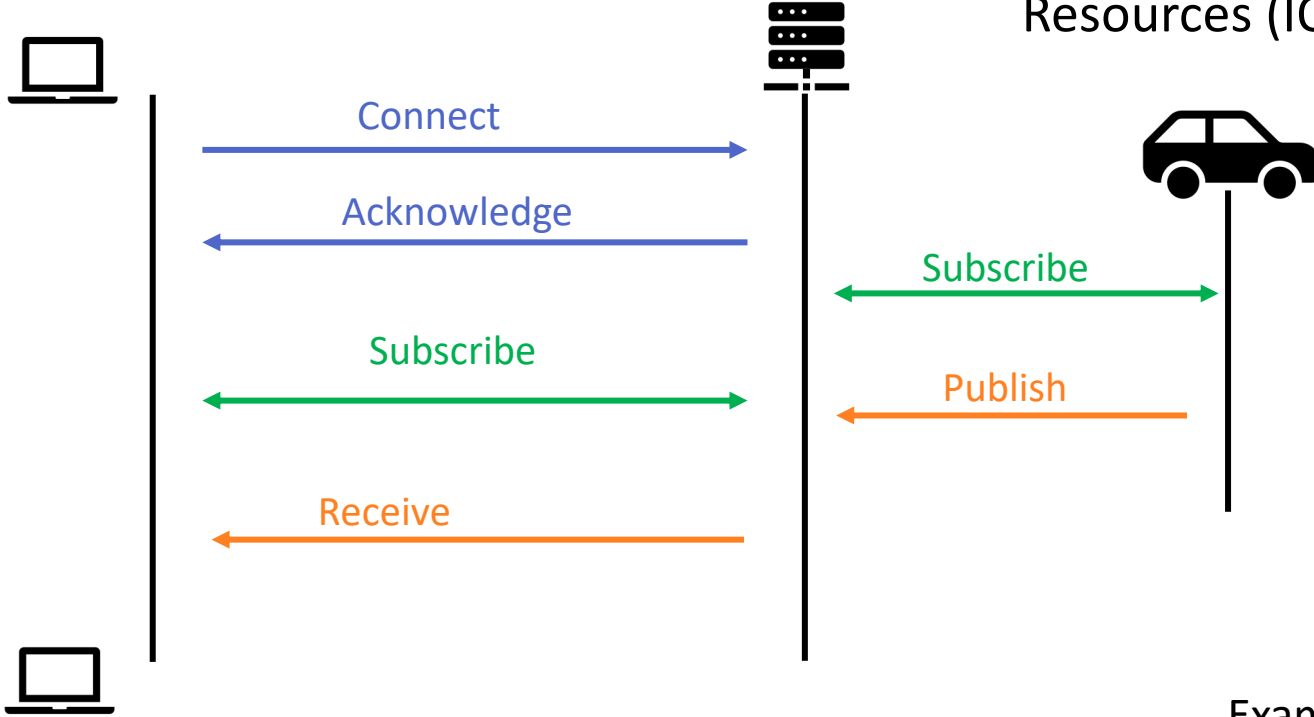
# Websockets



- Bi-direktionales Protokoll
- Nah am TCP
- Upgrade handshake von HTTP

Example: Blockchain

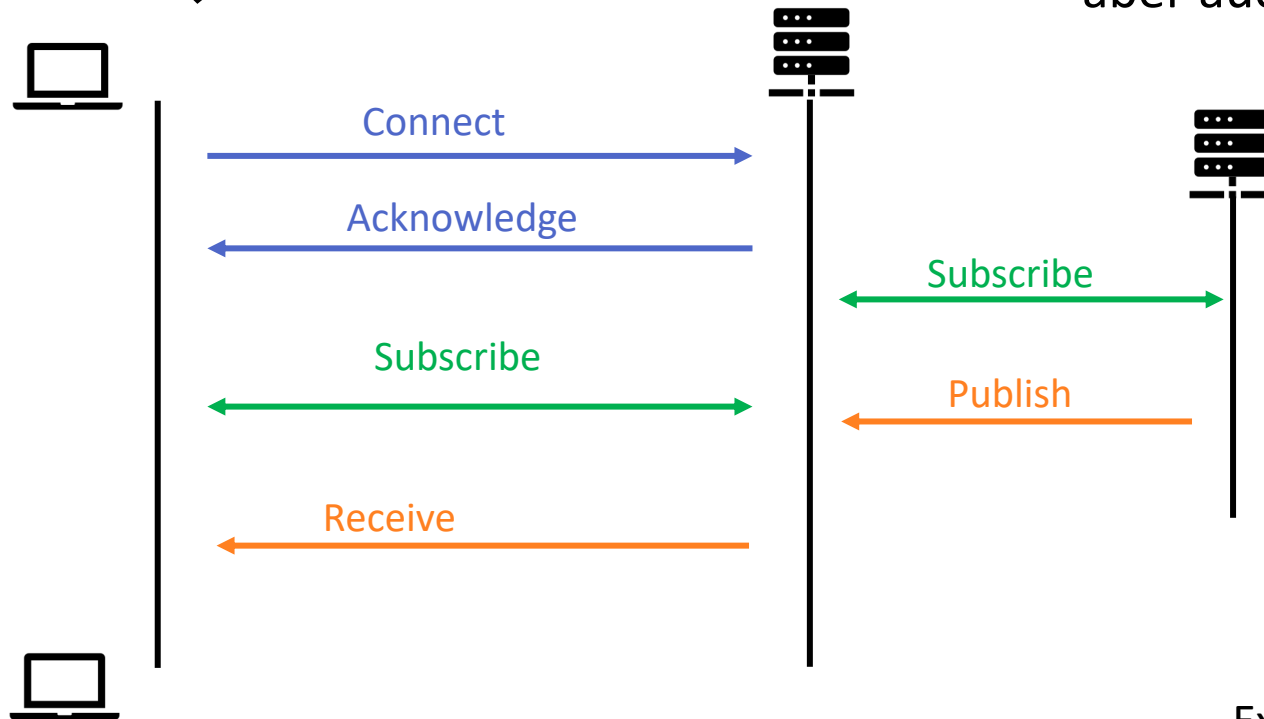
# MQTT



- MQTT ist gedacht für Clients mit begrenzten Resources (IOT Devices)

Example: Streetlight

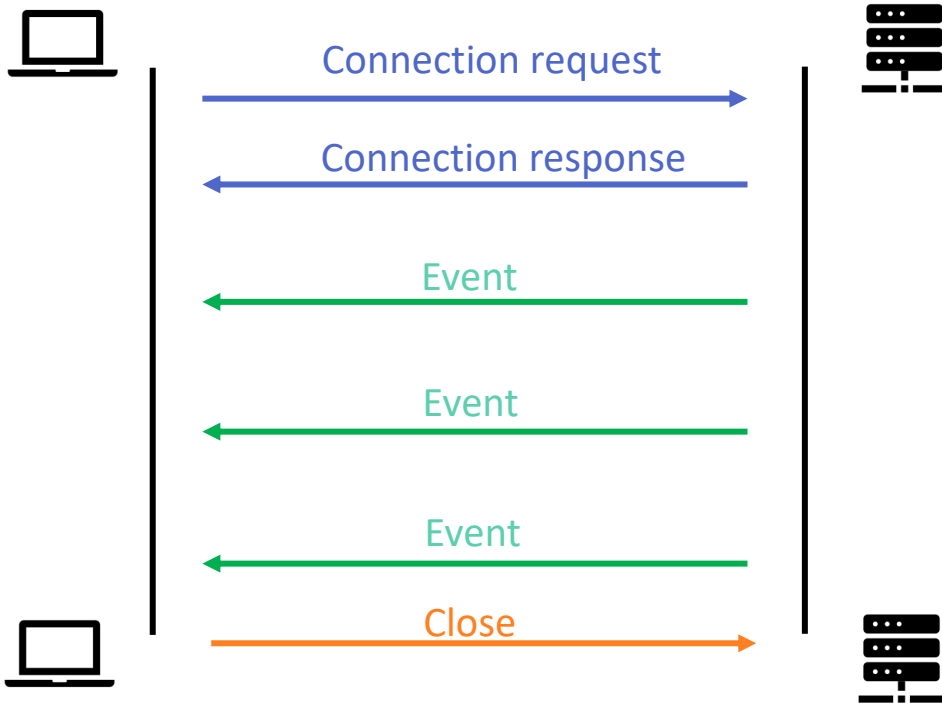
# AMQP



- Mehr features als MQTT aber auch mehr overhead

Example: RabbitMQ

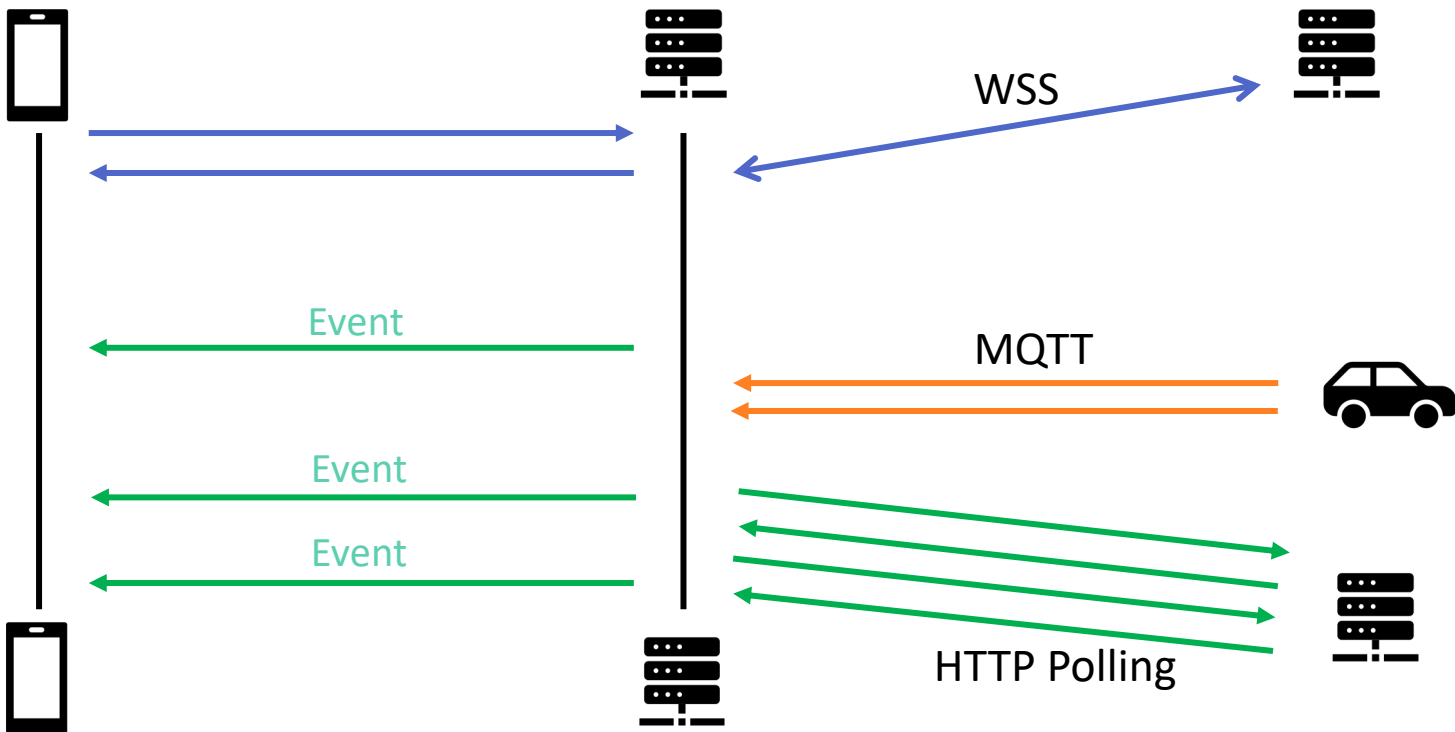
# Server Side Events



- Effizientes uni-direktionales
- Vordefinierte Fehler Behandlung
- HTTP-basiert für real time data streaming
- Einfache Client Implementierung

Example: Gitter

# Bridge the Gap - Streams API



FINAL  
THOUGHTS

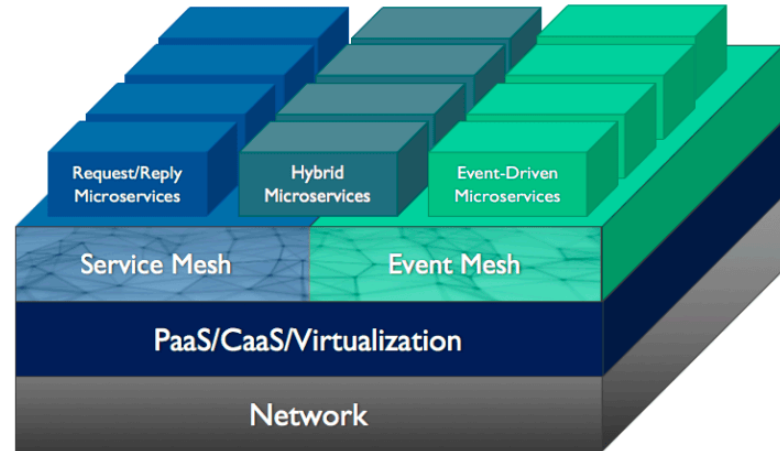




# Event Mesh – Work in Progress

Eine Ebene um die Brücke zu  
Event Driven Microservices  
zu schlagen mit Fokus auf

- Security
- Auto Discovery
- Governance  
load balancing
- Resilience



Source: <https://solace.com/what-is-an-event-mesh/>

# API First

- Beschreiben sie ihr Interface bevor Sie an die Implementierung gehen.
- Erzeugen Sie eine simple Testimplemetierung (Mockup).
- Verifizieren Sie ihre Idee mit den voraussichtlichen Nutzern.
- Gehen Sie iterativ vor und fragen lieber öfter, das ergibt kleinere Änderungen und sie bleiben am Ball.
- Ihre Nutzer sind auch Entwickler !



# Thank You

Email an **axel@axway.com** **Betreff: EventAPI**

Für Quellcode

oder

<https://github.com/axelgrosseaxway/event-driven-api>