


# Event Design


How do we know what events to create?

What data should each event contain?


Deciding on *what* events to publish and what data they should contain is *tricky*

A black arrow pointing downwards from the first box to the second box.

Depends on the purpose of your app and the features it implements

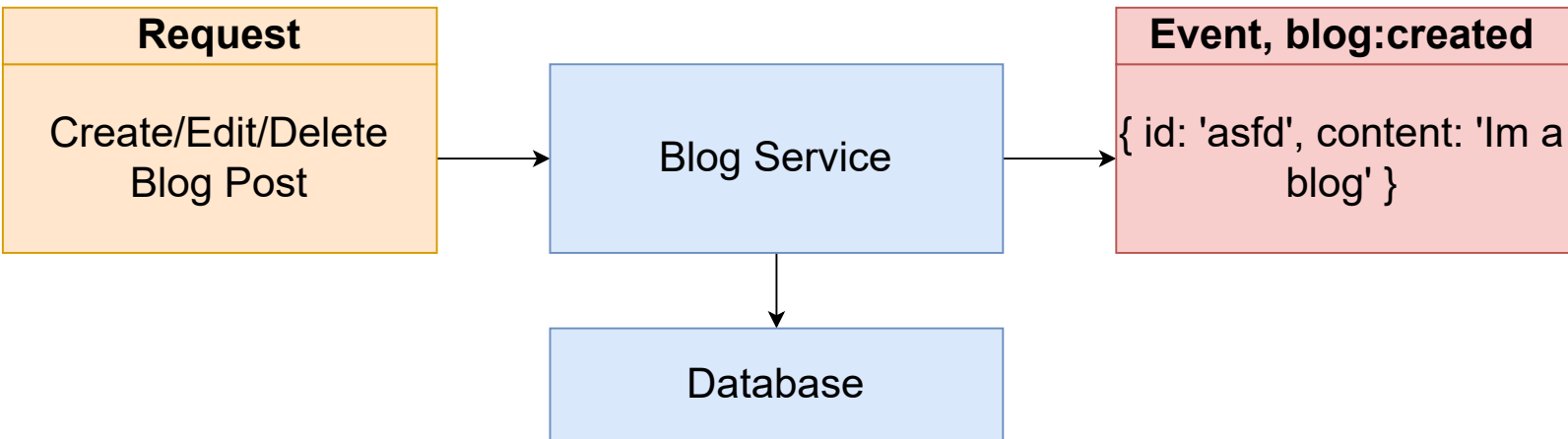
A black arrow pointing downwards from the second box to the third box.

Also involves thinking about concurrency issues!

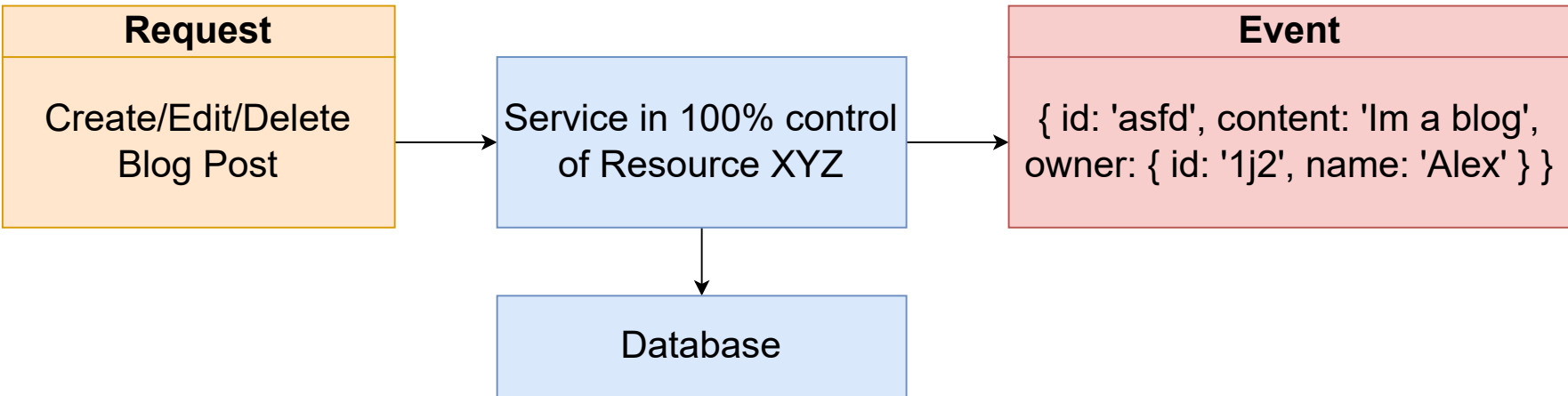
A black arrow pointing downwards from the third box to the fourth box.

At best, I can give you some rules of thumb and show you how I applied them to this project

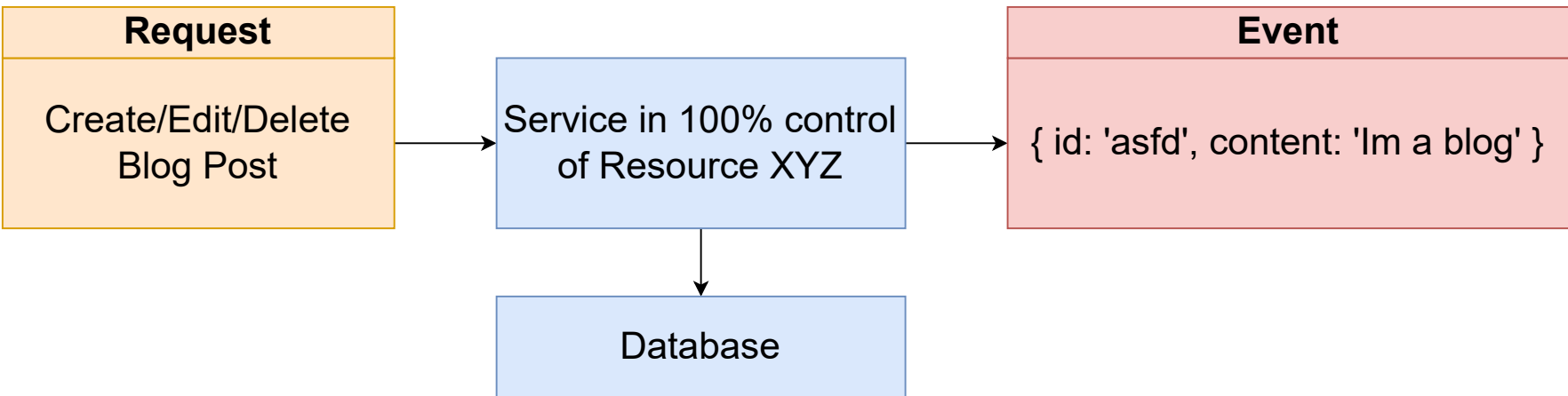
**Rule #1** - Make *one* service in charge of all aspects of a Resource. Emit events whenever changing that data



**Rule #2** - If you don't know how the event will be used, publish all available data about the resource



**Rule #3** - If you *do* know how the event will be consumed (and don't expect it to change soon), publish only the required info



## Tickets Service

event

## Orders Service

event

event

event

event

event

event

event

## Payments Service

event

## Expiration Service

# Data Held + Replicated by Each Service

## Tickets Service

tickets

## Orders Service

orders

tickets

## Expiration Service

## Payments Service

charges

orders

# Subject Names Format

**Name of resource**

**:**

**Thing that happened to it**



ticket

:

created



ticket

:

updated

order

:

created

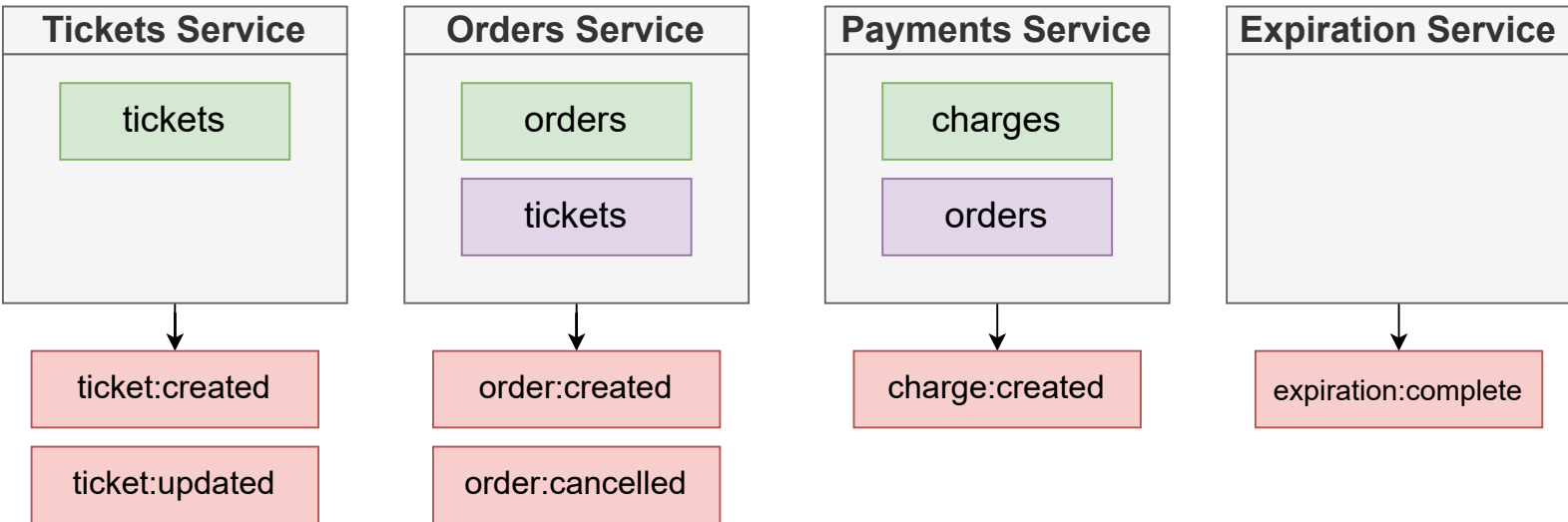
expiration

:

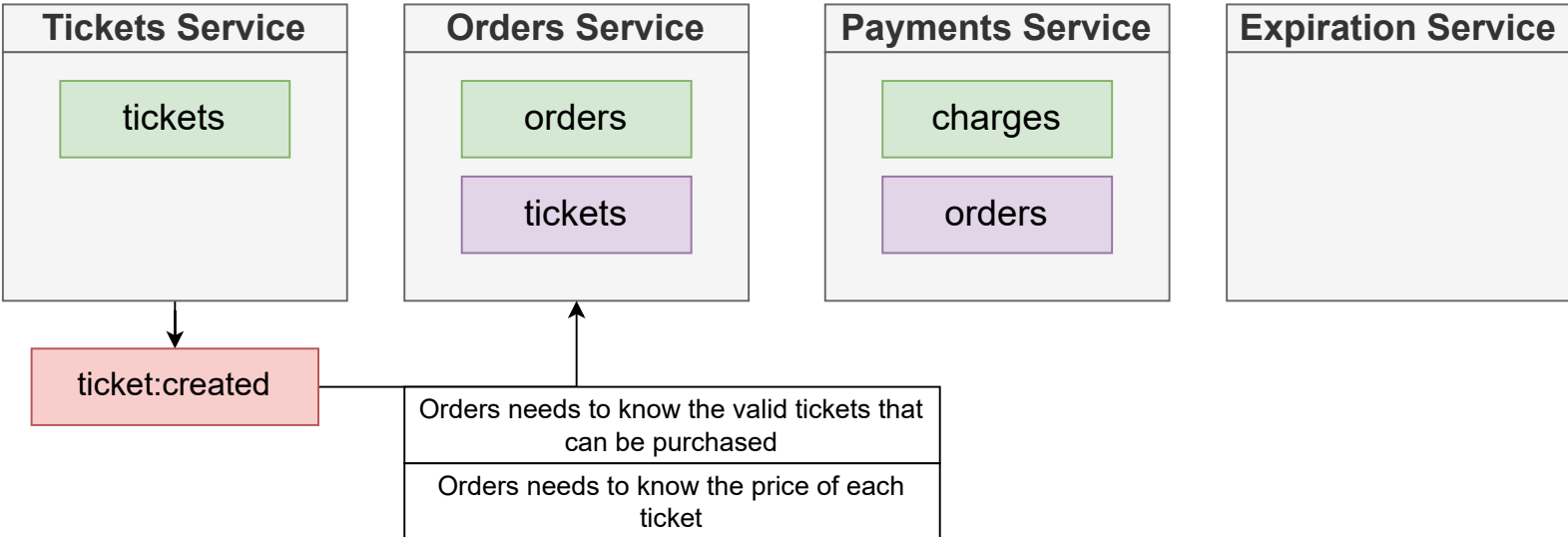
complete



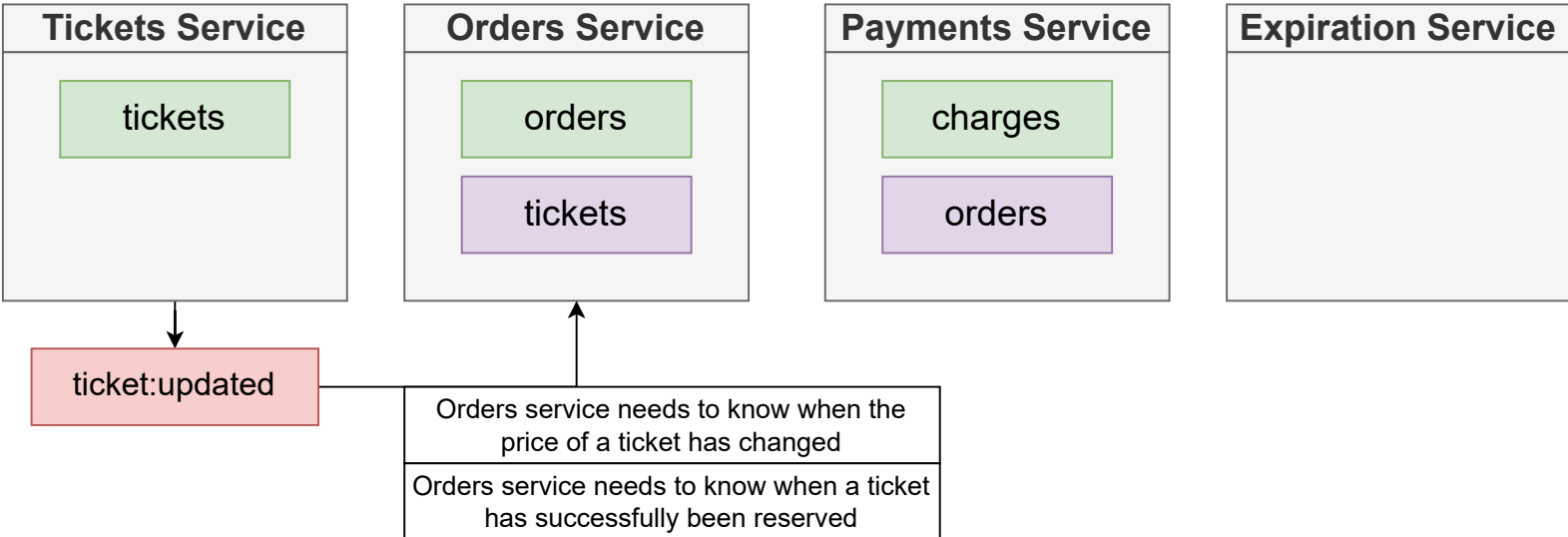
## Events Published by Each Service



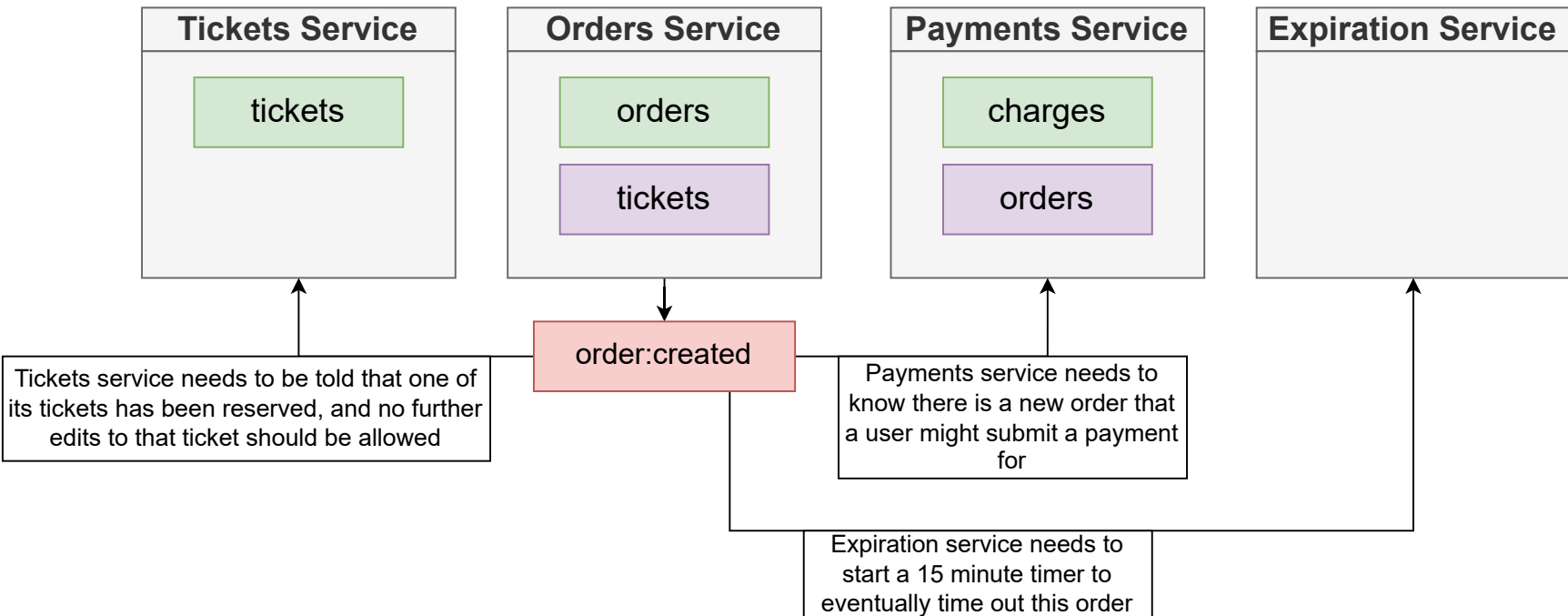
## Events Published by Each Service



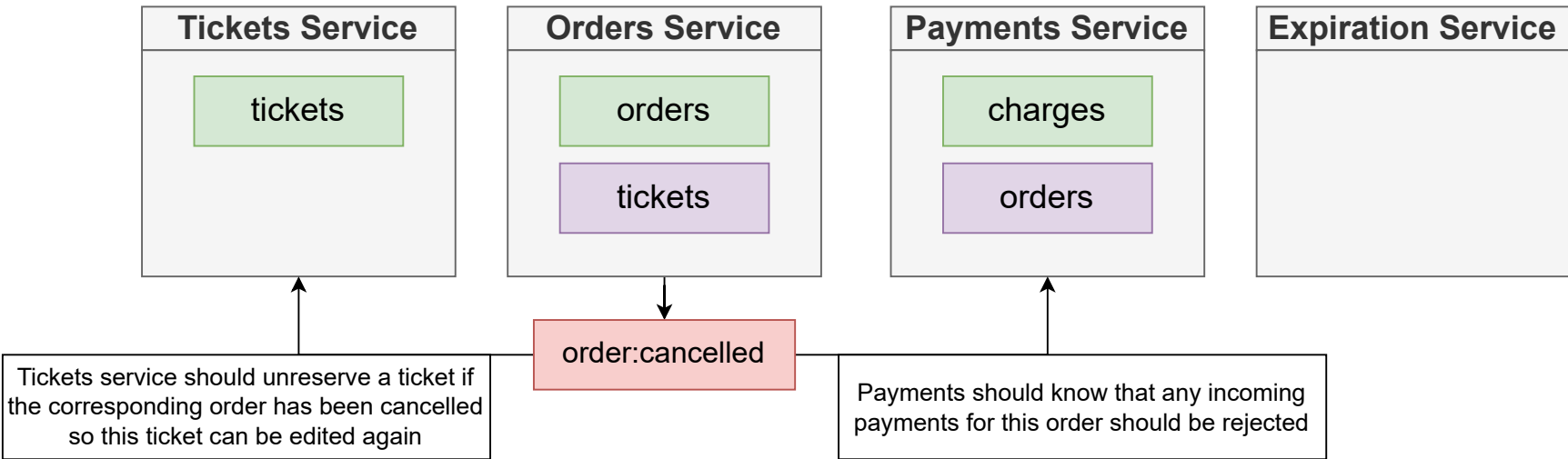
## Events Published by Each Service



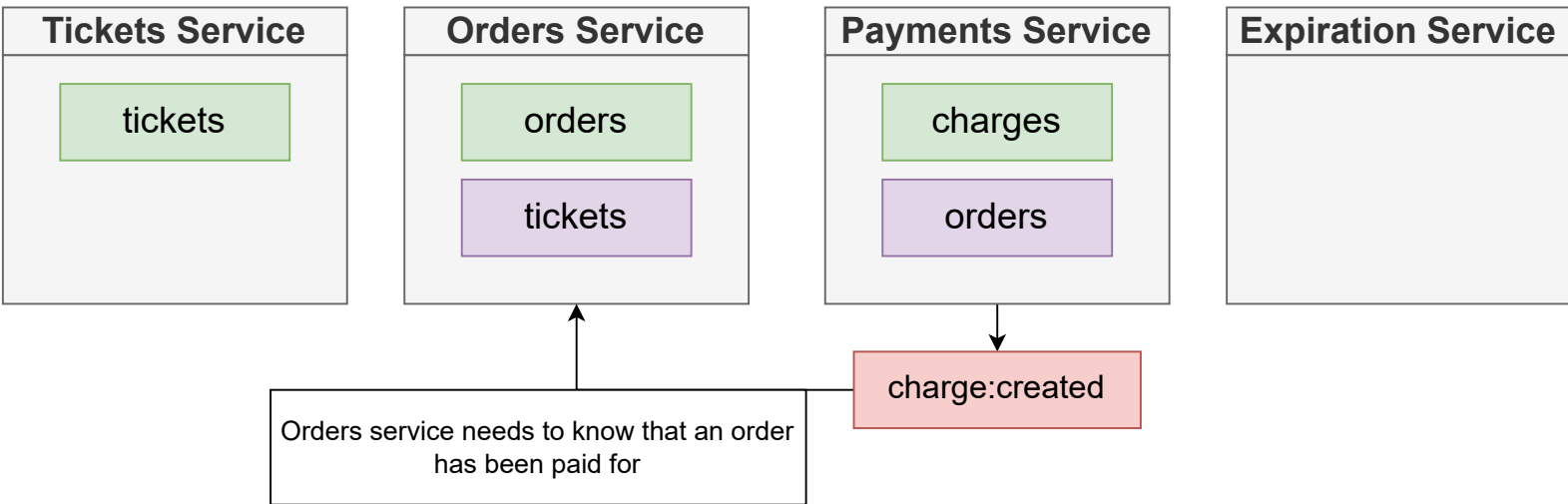
## Events Published by Each Service



## Events Published by Each Service



## Events Published by Each Service



## Events Published by Each Service

