

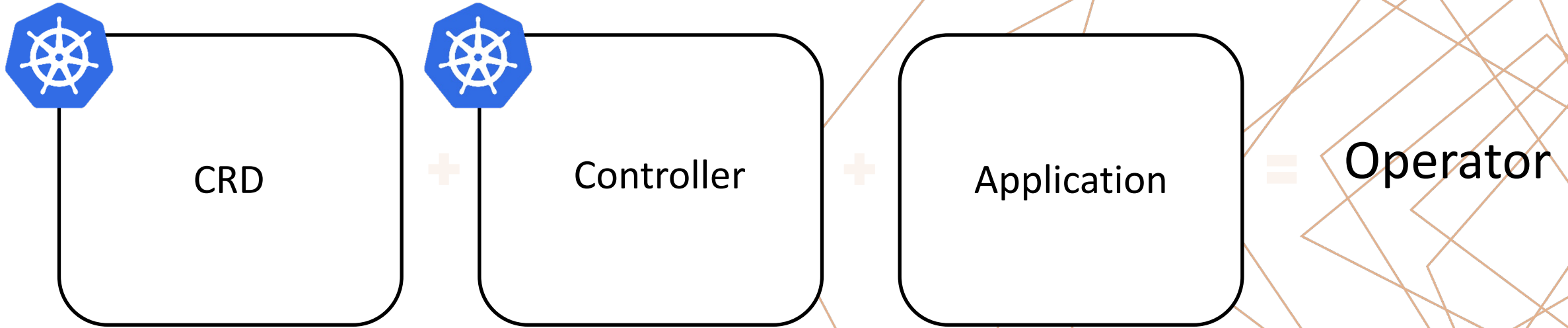
Abstract geometric lines in the top left corner, consisting of several overlapping, irregular polygons and lines in a light beige color.

ЧЕРЕЗ ТЕРНИИ К

KUBERNETES OPERATORS

ЧТО ТЫ ТАКОЕ?

“An Operator is a method of packaging, deploying and managing a Kubernetes application. A Kubernetes application is an application that is both deployed on Kubernetes and managed using the Kubernetes APIs and kubectl tooling.”



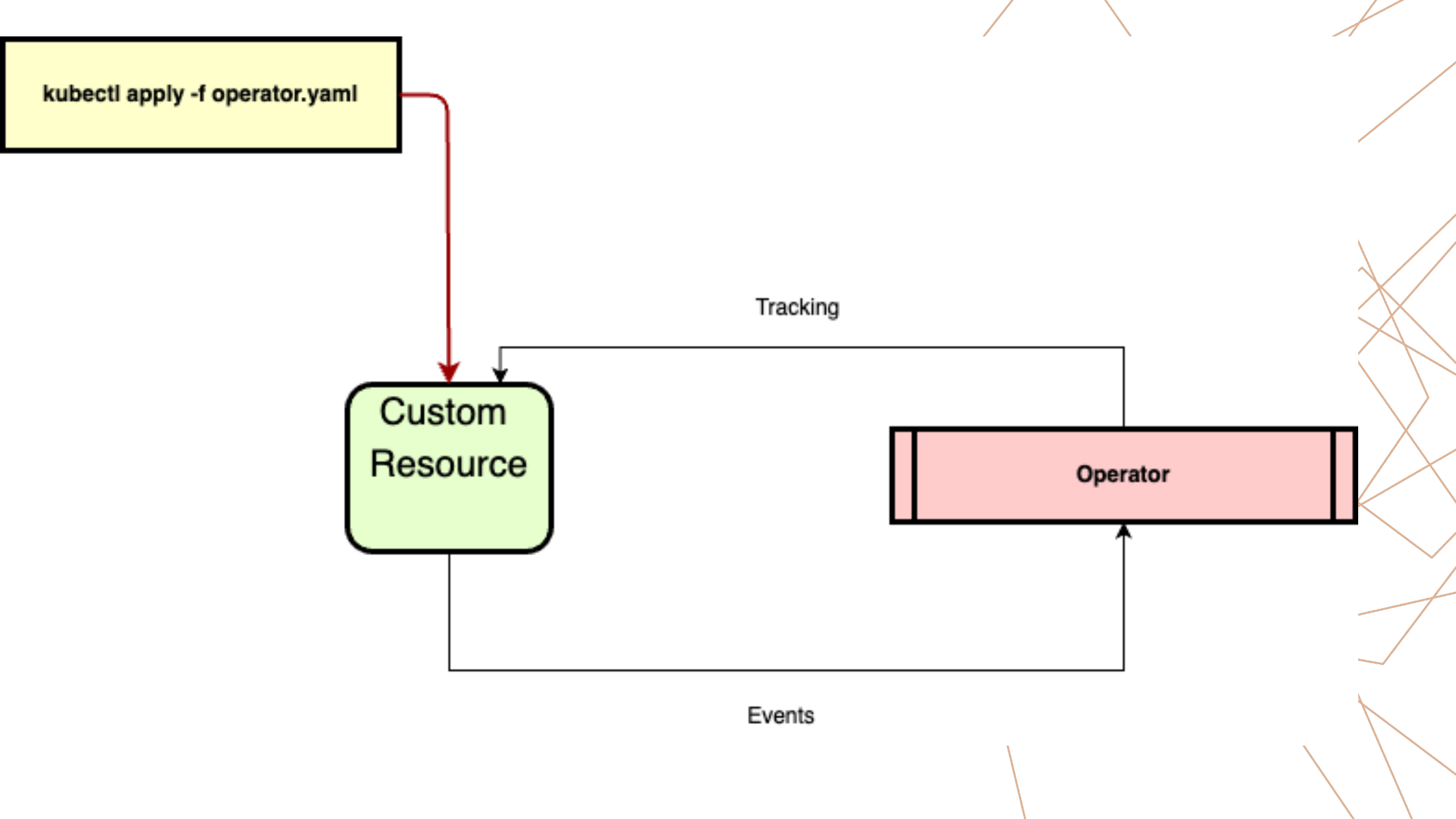
`kubectl apply -f operator.yaml`

Custom Resource

Operator

Tracking

Events



ОБРАБОТЧИКИ

@kopf.on.startup()

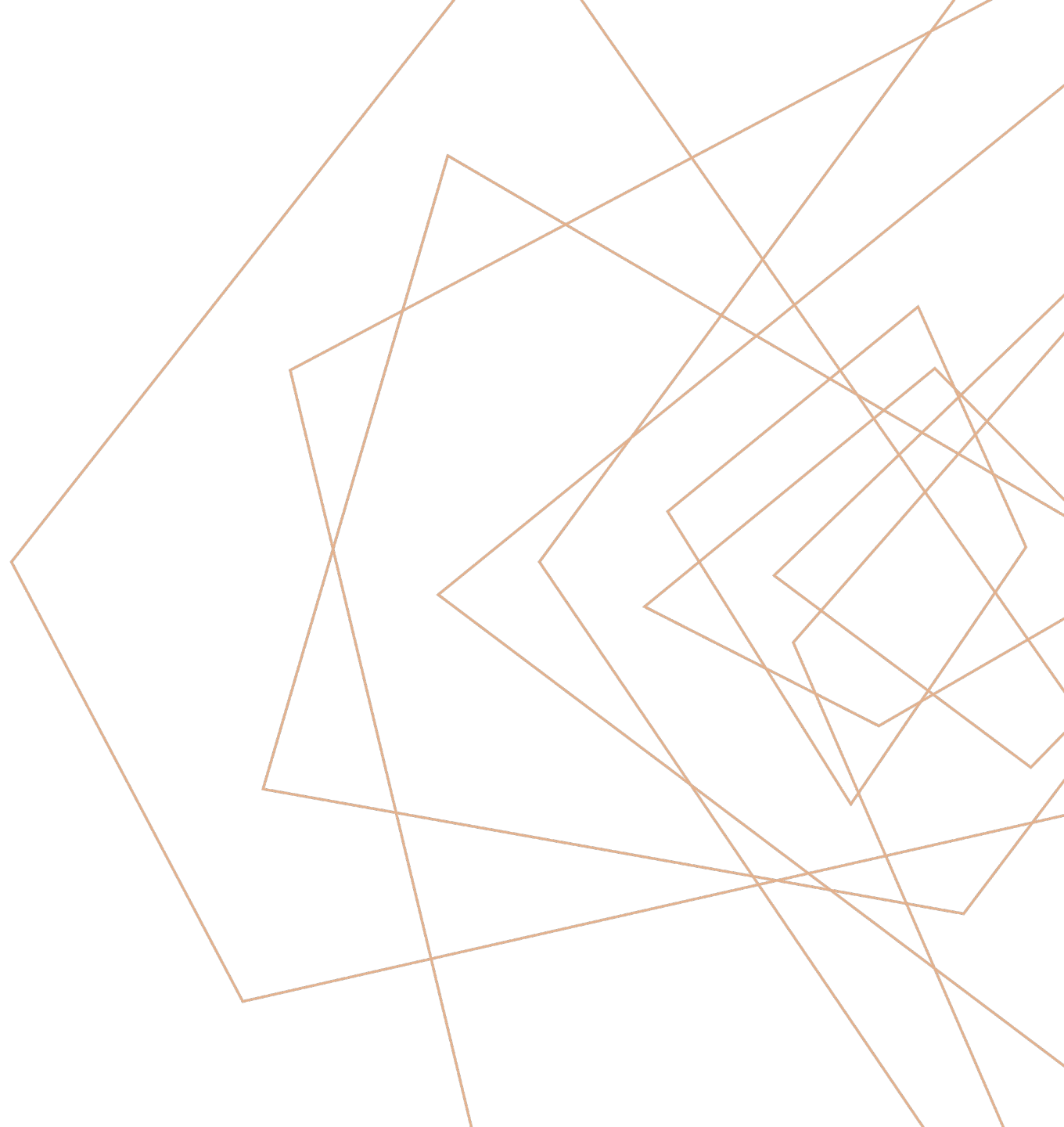
@kopf.on.login()

@kopf.on.create()

@kopf.on.update()

@kopf.on.delete()

@kopf.on.cleanup()



@KOPF.ON.STARTUP()

Обработчики запуска фактическая задача не запускаться до тех пор, пока все обработчики запуска не завершатся успешно.

@KOPF.ON.LOGIN()

Обработчики для авторизации в Kubernetes cluster, возможно использовать переменные окружения, cli конфигурацию, сервисный аккаунт или внешние сервисы

@KOPF.ON.CREATE()

Обработчики отслеживающий появления в
кластере CR отвечающий спецификации

@KOPF.ON.UPDATE()


Обработчики отслеживающий изменения в
пользовательском ресурсе

@KOPF.ON.DELETE()

Обработчики для удаления и очистки
дочерних и родительских ресурсов
созданным оператором

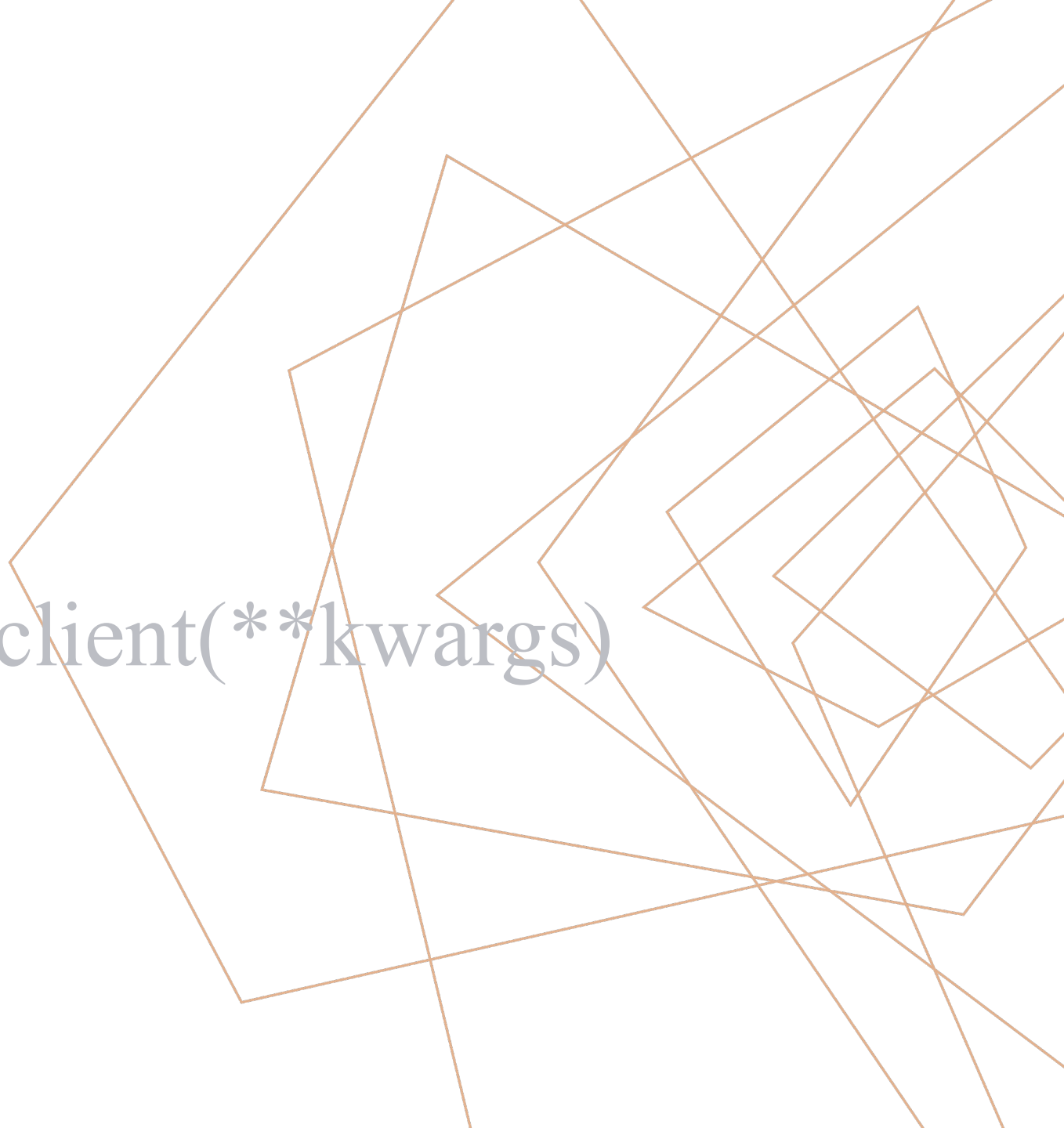
@KOPF.ON.CLEANUP()

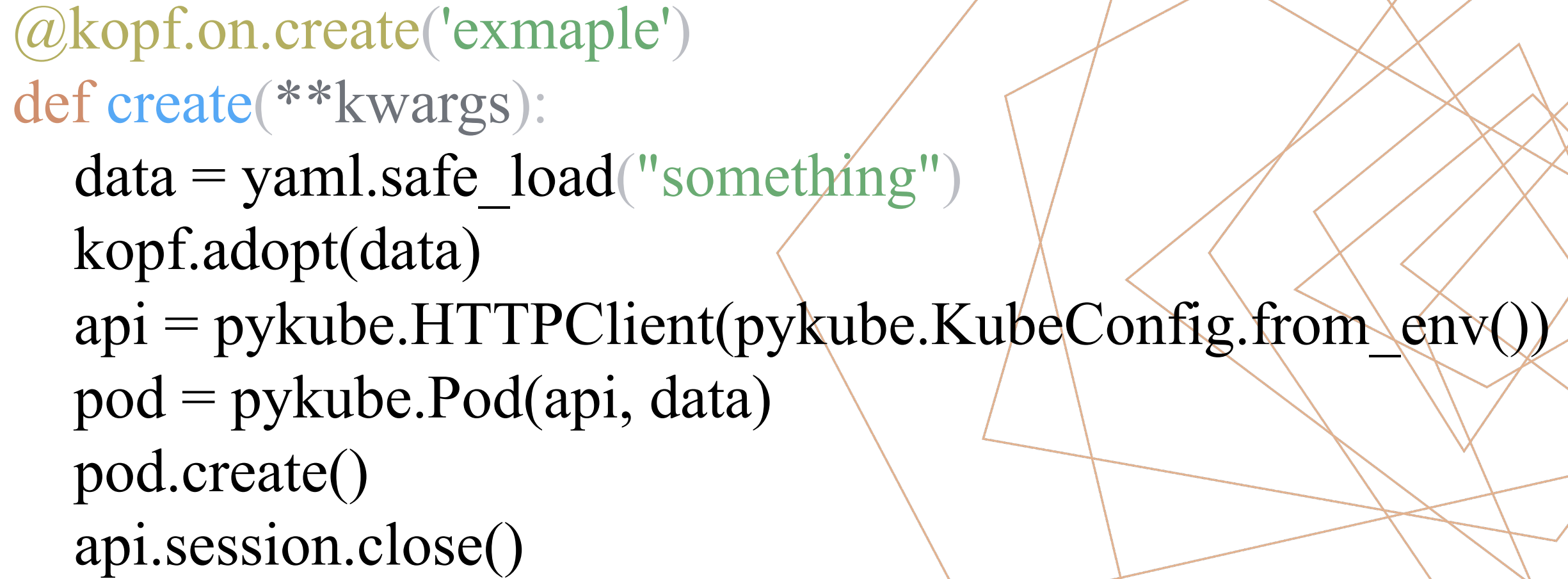
Обработчики очистки выполняются, когда оператор завершает работу, либо по перехвату исключения,



```
@kopf.on.startup()  
async def wait_for_operator(logger, **kwargs):  
    logger.info('Waiting for start')  
    await asyncio.sleep(3)
```

```
@kopf.on.login()
def login(**kwargs):
    return kopf.login_via_client(**kwargs)
```





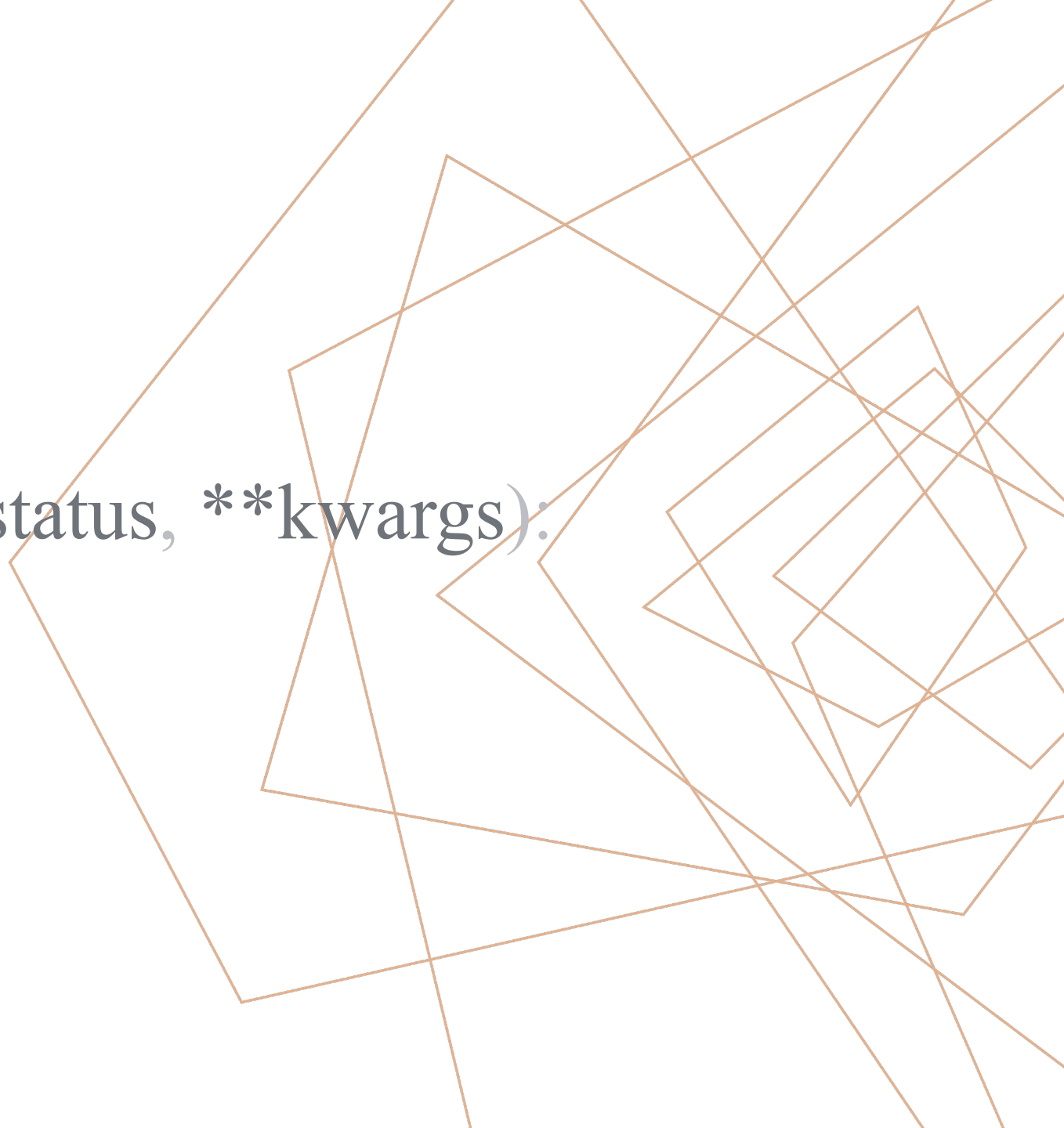
```
@kopf.on.create('exmaple')
def create(**kwargs):
    data = yaml.safe_load("something")
    kopf.adopt(data)
    api = pykube.HTTPClient(pykube.KubeConfig.from_env())
    pod = pykube.Pod(api, data)
    pod.create()
    api.session.close()
```


```
@kopf.on.update('example')
```

```
def update(body, meta, spec, status, old, new, diff, **kwargs):  
    print(list(diff))
```



```
@kopf.on.delete('example')
def delete(body, meta, spec, status, **kwargs):
    do something
```

An abstract geometric pattern consisting of multiple overlapping, irregular polygons and lines in a light brown or tan color. The lines intersect to form various shapes and spaces, creating a complex, layered visual effect on the right side of the page.



```
@kopf.on.cleanup()  
def cleanup(logger, **kwargs):  
    logger.info("Cleaning up in 3s...")  
    time.sleep(3)
```

apiVersion: apiextensions.k8s.io/v1

kind: CustomResourceDefinition

metadata:

name: demo.devoops.io

spec:

scope: Namespaced

group: devoops.io

names:

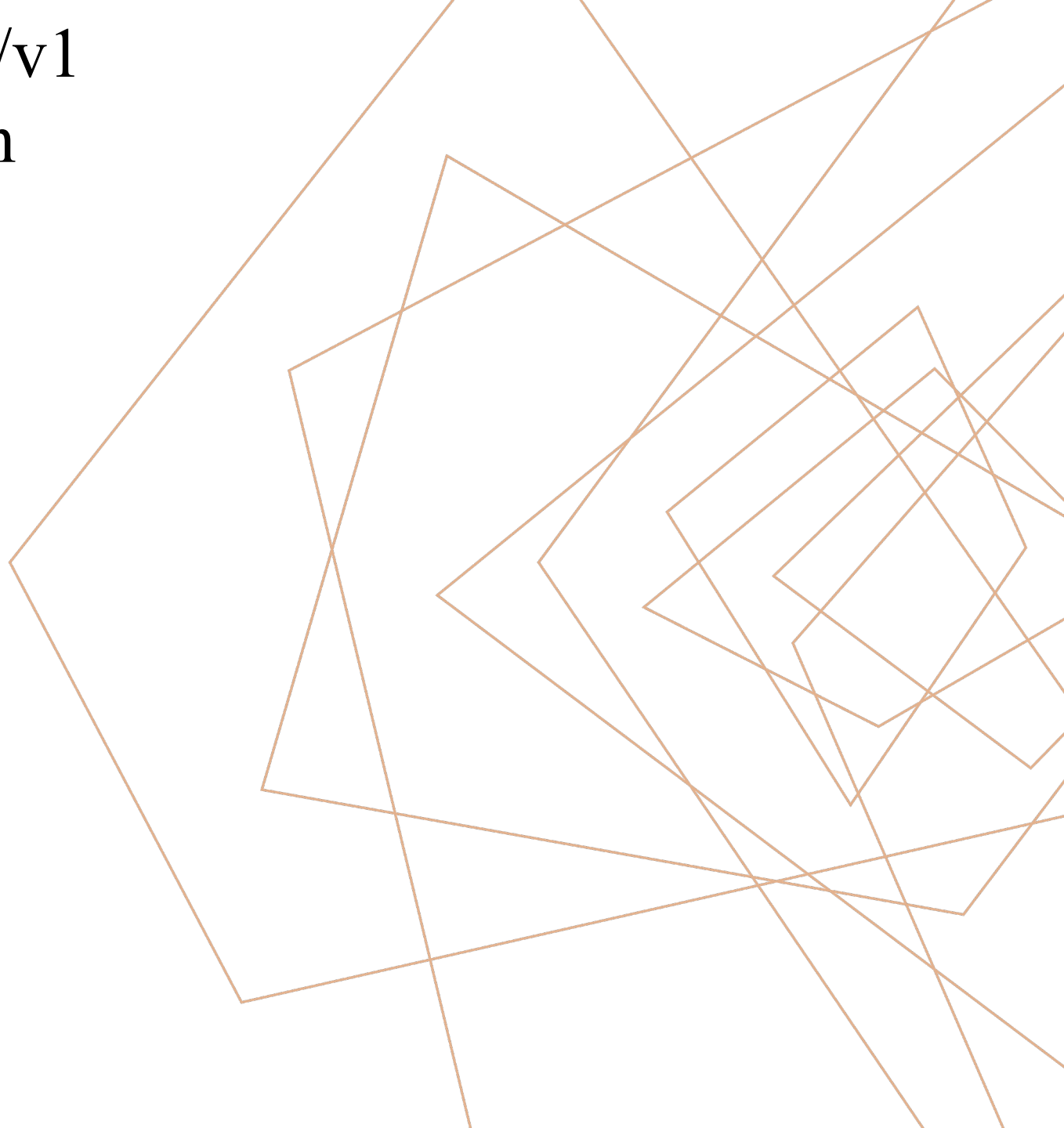
kind: demo

plural: demo

singular: demo

shortNames:

- d



apiVersion: apiextensions.k8s.io/v1

kind: CustomResourceDefinition

metadata:

name: demo.devoops.io

spec:

scope: Namespaced

group: devoops.io

names:

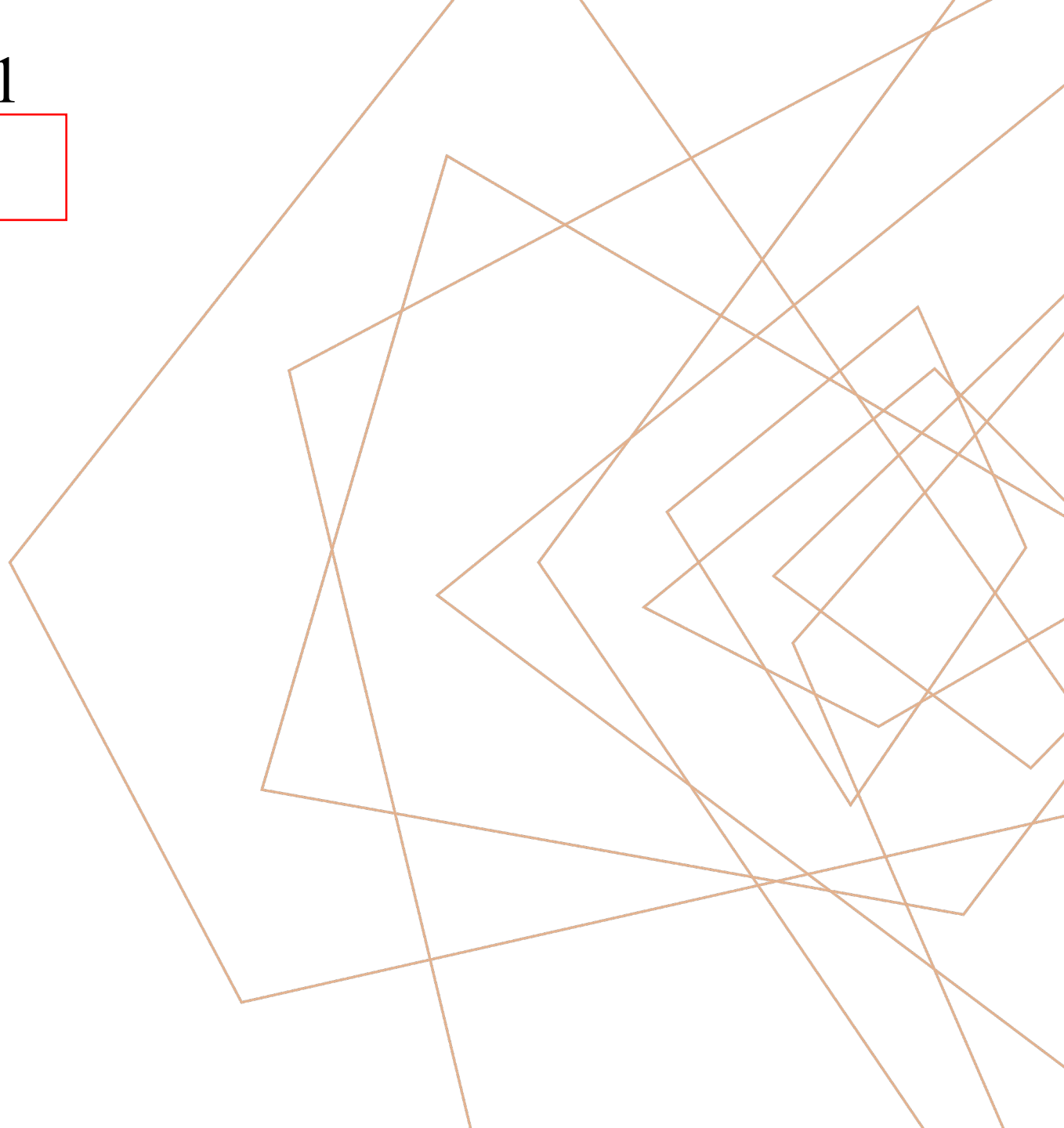
kind: demo

plural: demo

singular: demo

shortNames:

- d



apiVersion: apiextensions.k8s.io/v1

kind: CustomResourceDefinition

metadata:

name: demo.devoops.io

spec:

scope: Namespaced

group: devoops.io

names:

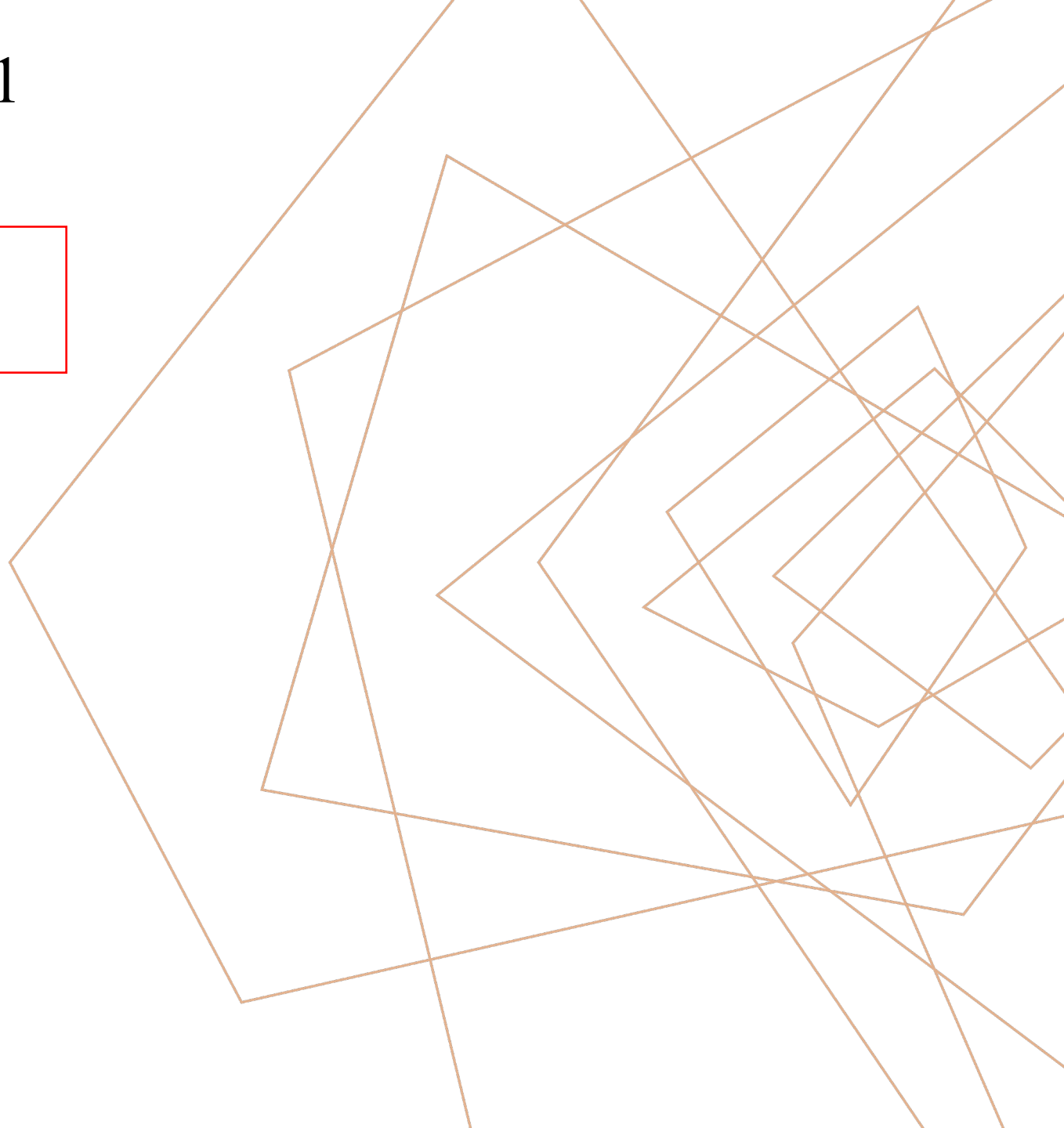
kind: demo

plural: demo

singular: demo

shortNames:

- d



apiVersion: apiextensions.k8s.io/v1

kind: CustomResourceDefinition

metadata:

name: demo.devoops.io

spec:

scope: Namespaced

group: devoops.io

names:

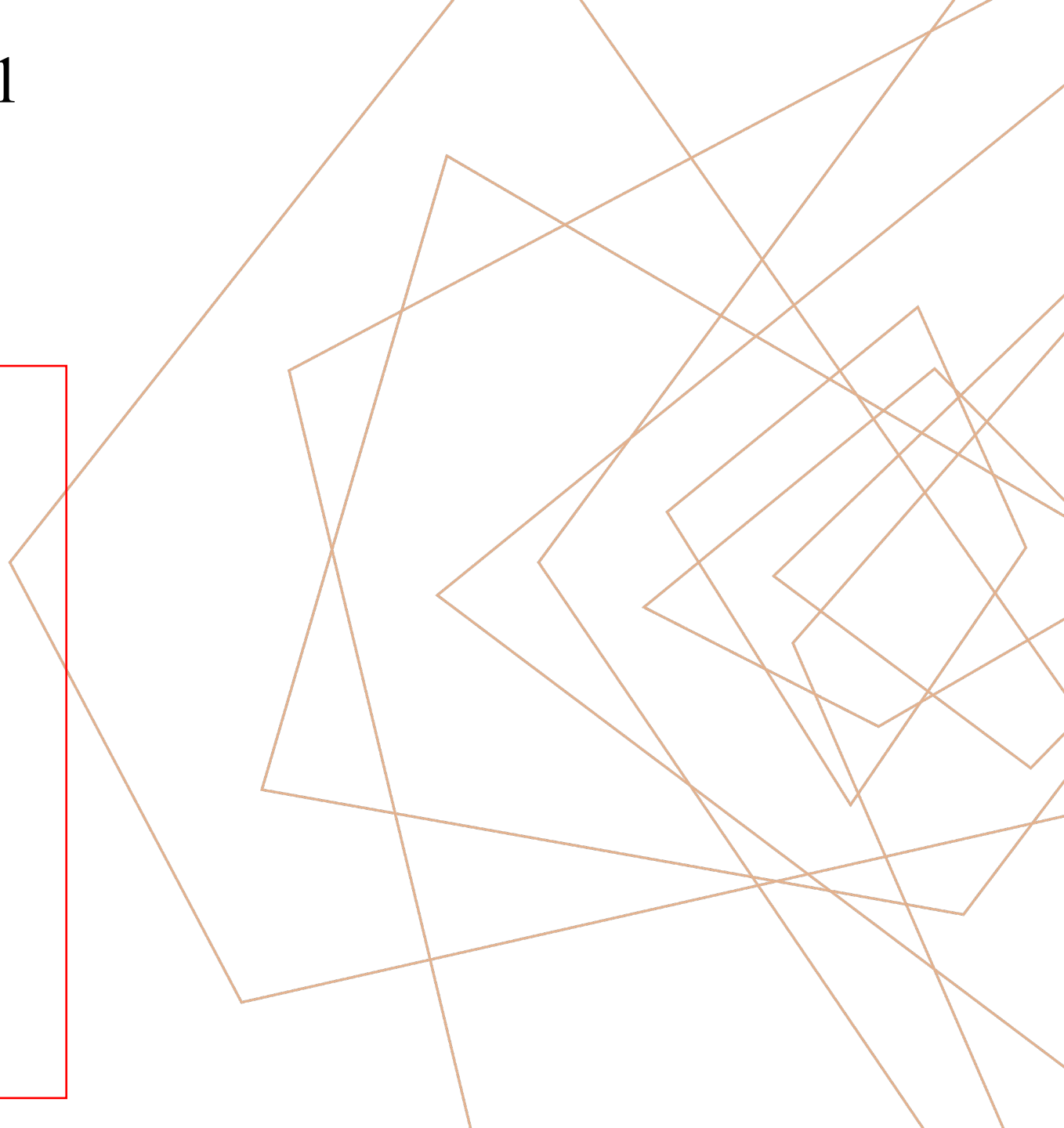
kind: demo

plural: demo

singular: demo

shortNames:

- d



versions:

- name: v1

served: true

storage: true

schema:

openAPIV3Schema:

type: object

properties:

spec:

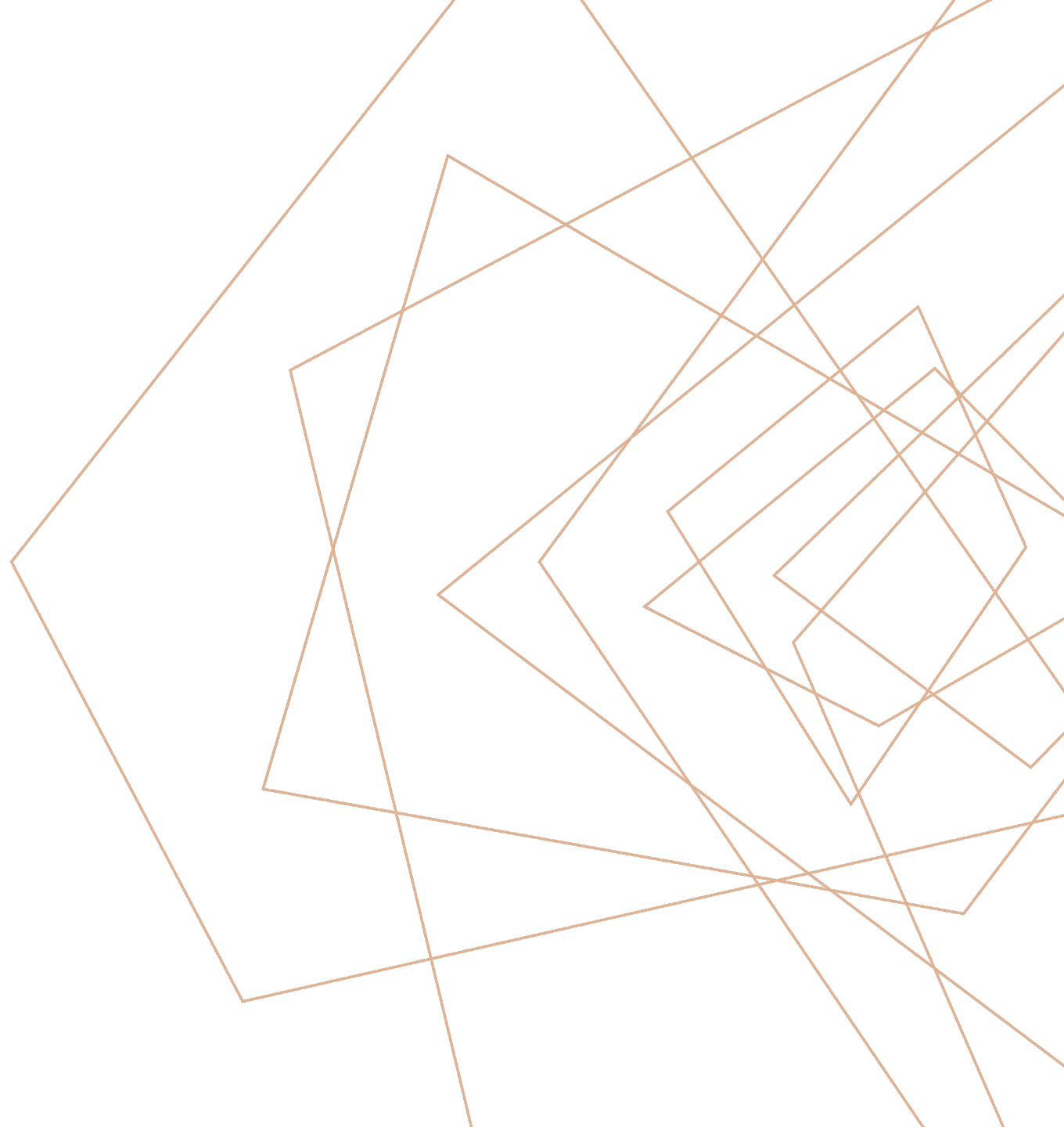
type: object

properties:

name:

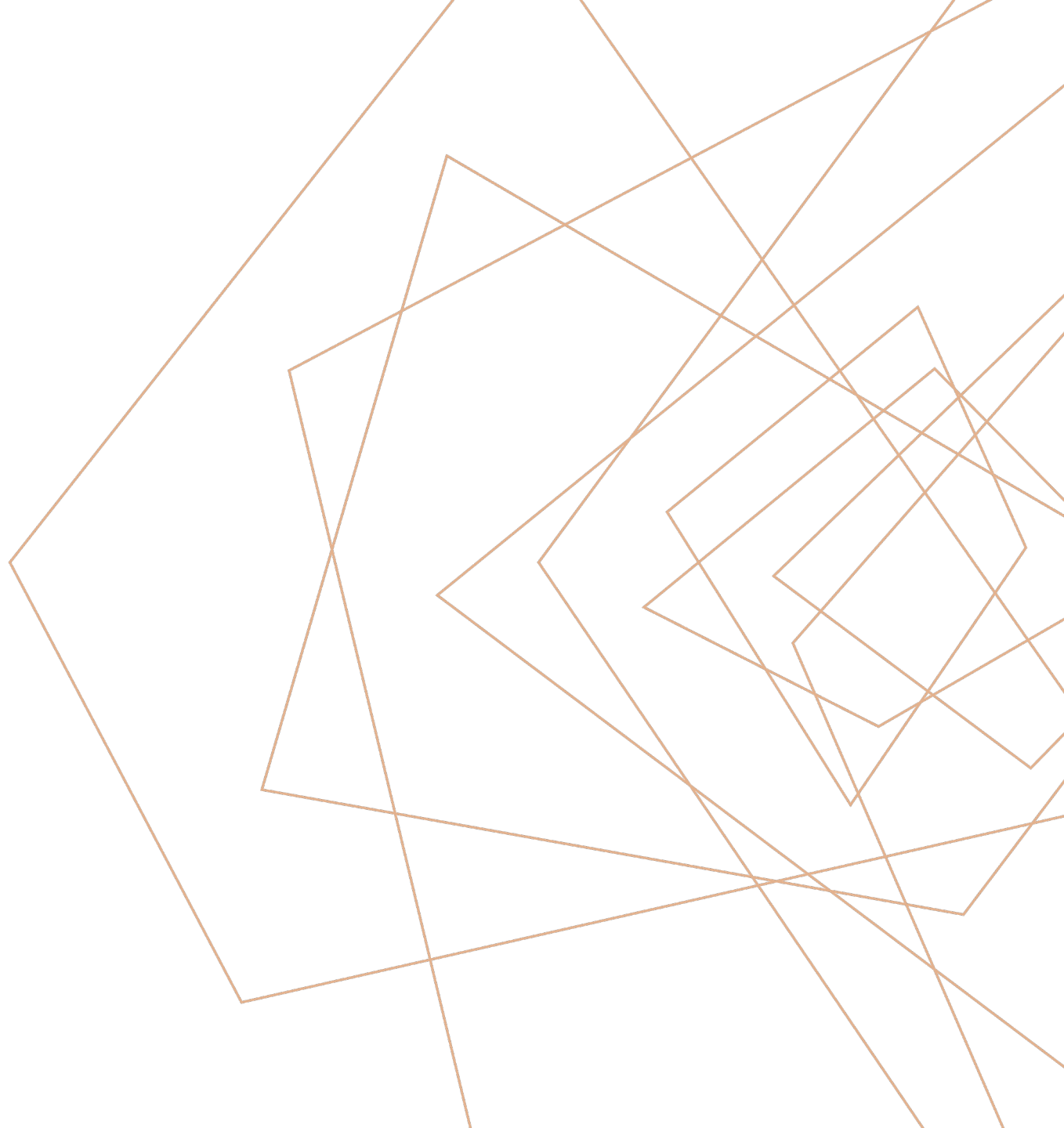
type: string

default: None



ТЕСТИРОВАНИЕ

- pyunit tests
- E2e
- Ручное?????



The background features a complex, abstract geometric pattern composed of numerous thin, light brown lines. These lines intersect to form a variety of overlapping polygons, including triangles, quadrilaterals, and pentagons. The pattern is most dense on the right side of the image, where it creates a sense of depth and complexity, while the left side is mostly clear white space.

ПРАКТИКА/DEMO