

corewire

Helm

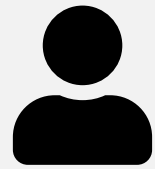
Helm Grundlagen

Folien-Hinweis

- Space, Page down: Nächste Folie
- Page up: Vorherige Folie
- ESC, o: Übersicht

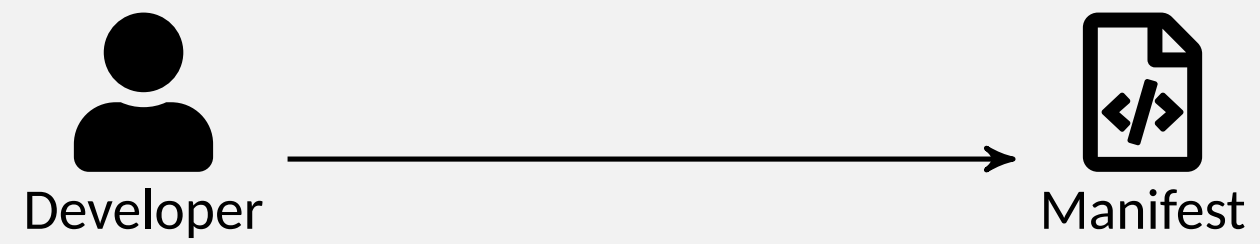
[Zur Kapitelübersicht](#)

Helm

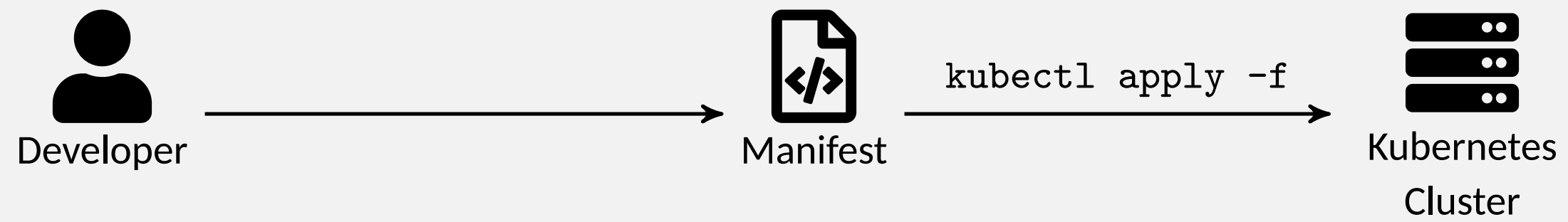


Developer

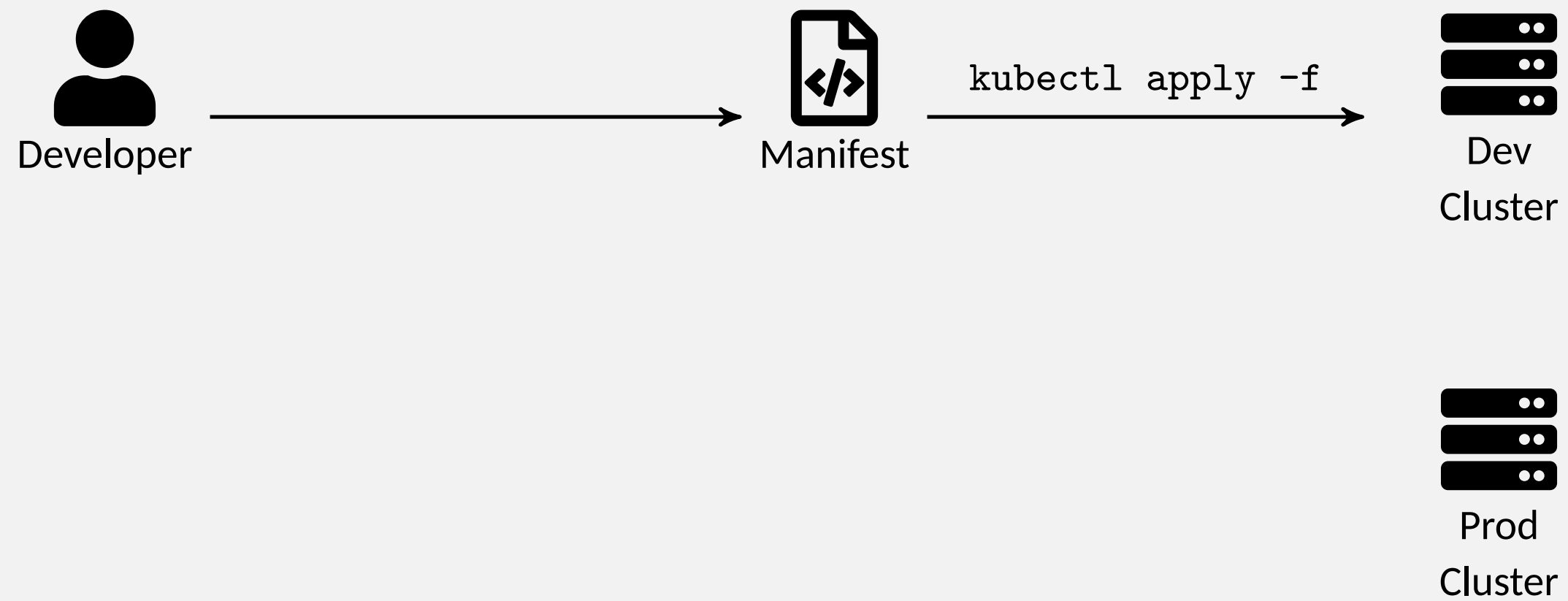
Helm



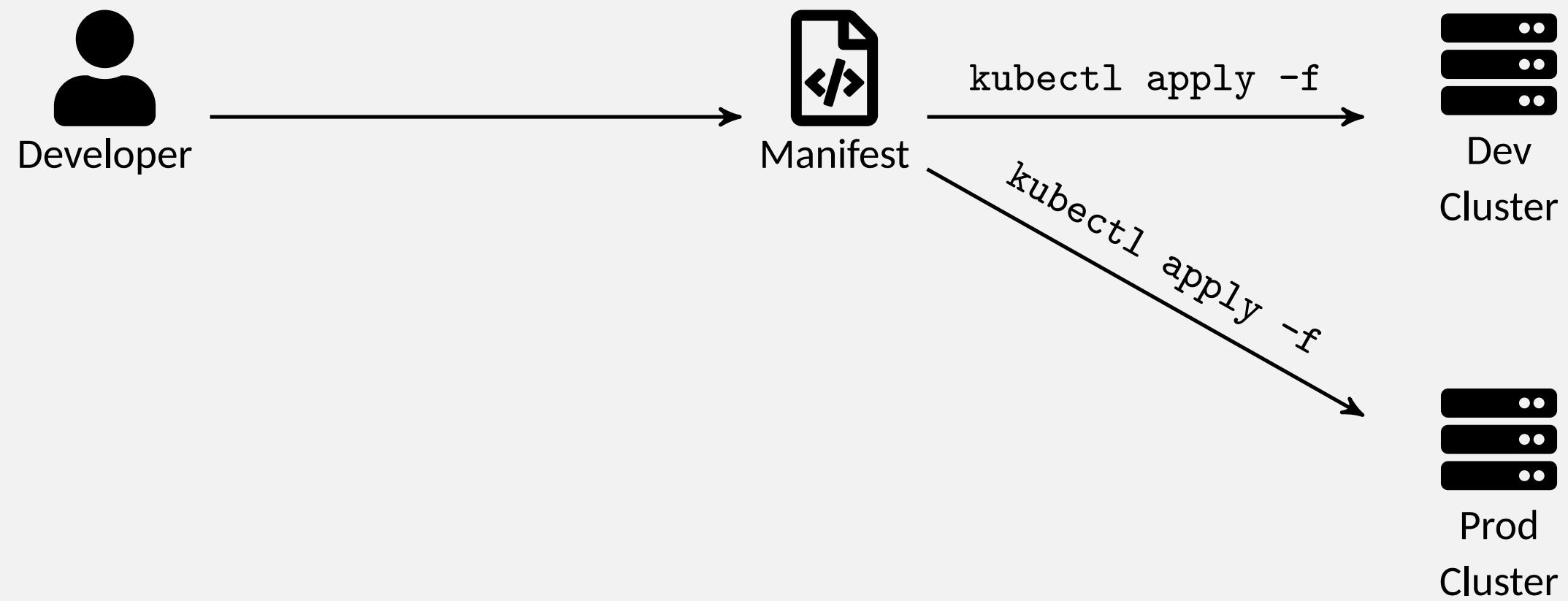
Helm



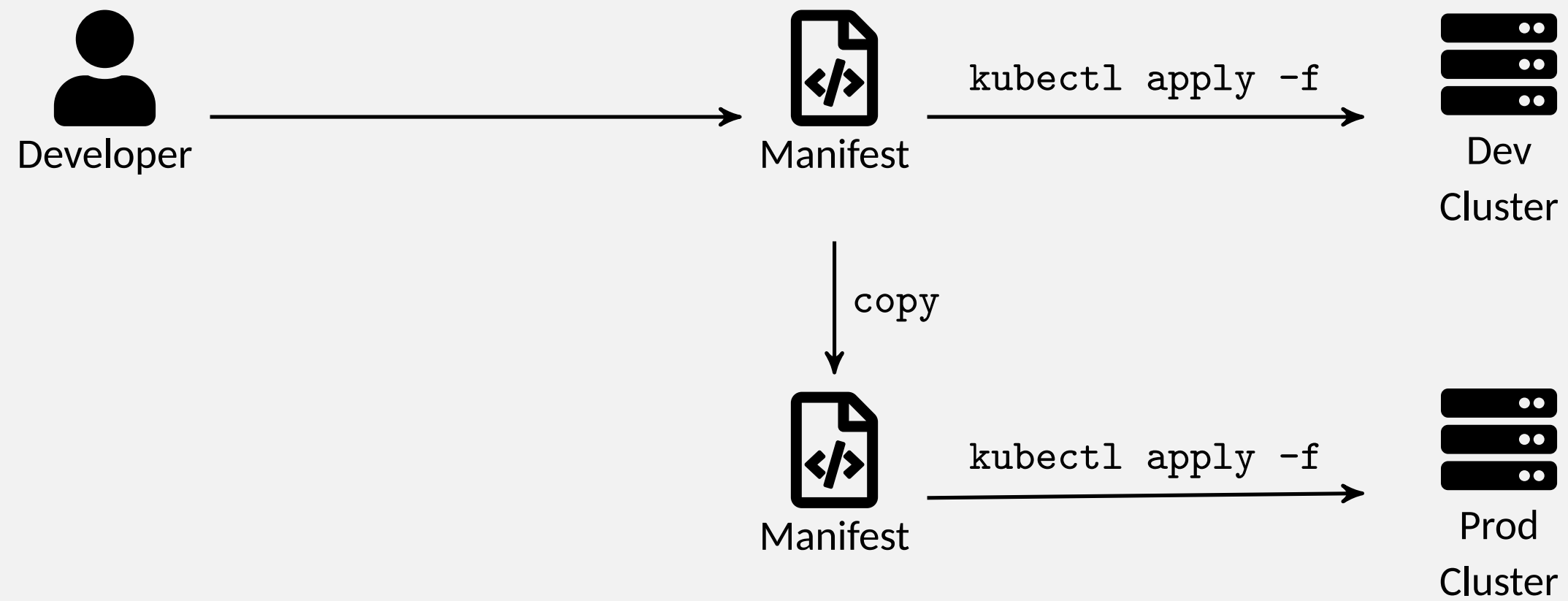
Helm



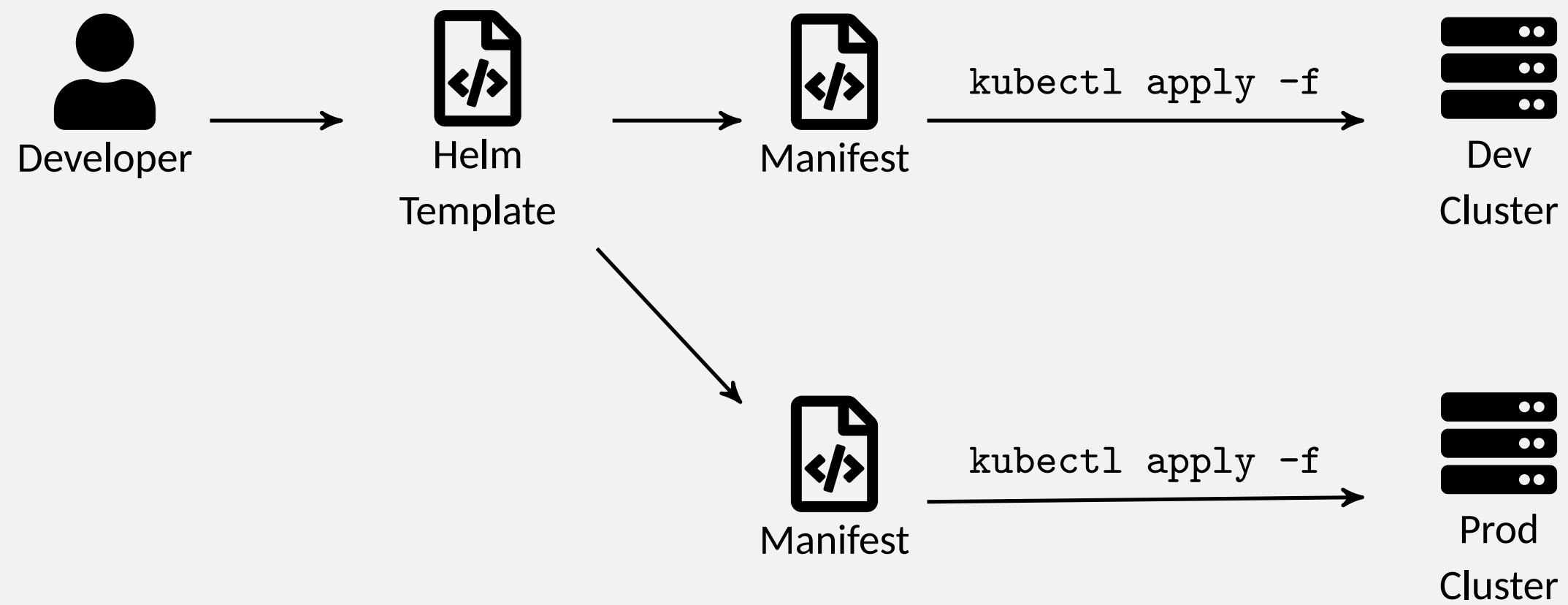
Helm



Helm



Helm



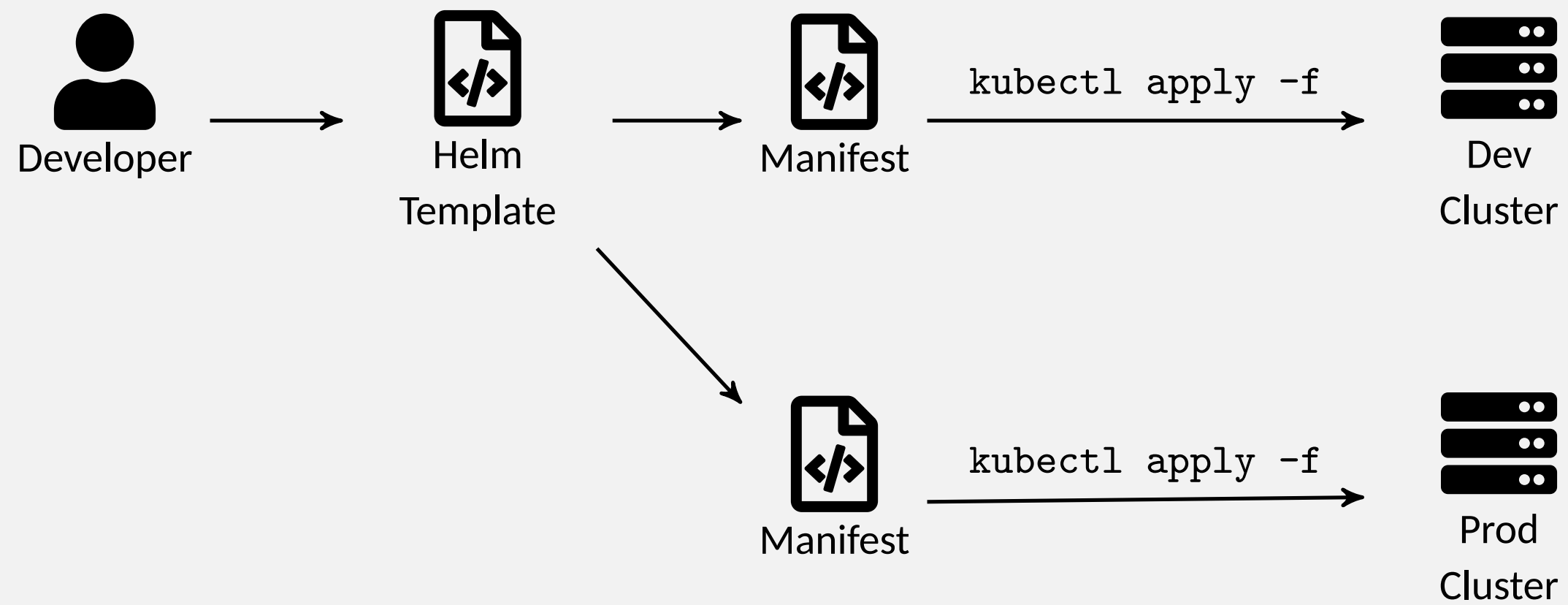
Helm Einführung

- Paketmanager für Kubernetes
- Mit Templates können Manifeste auf verschiedene Umgebungen angewendet werden

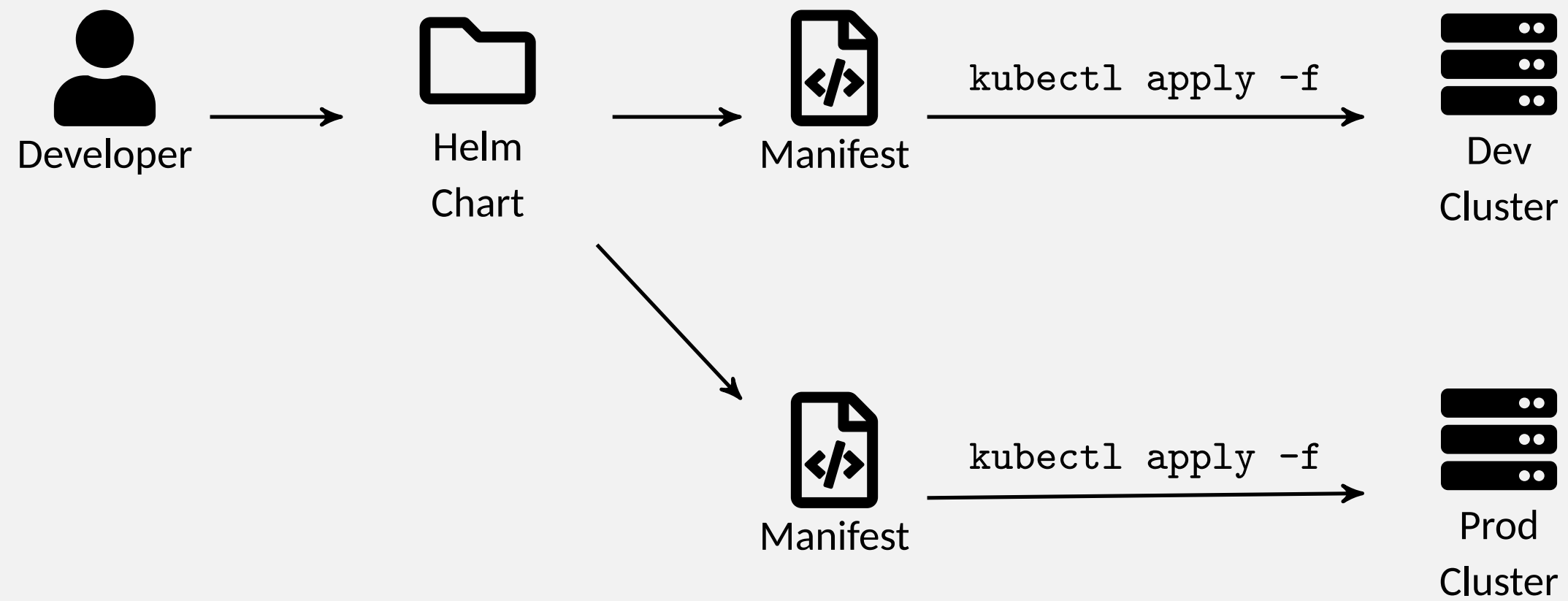
Helm Template

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-deployment
spec:
  replicas: {{ .Values.replicaCount }}
  selector:
    matchLabels:
      my-label: my-app
  template:
    metadata:
      labels:
        my-label: my-app
    spec:
      containers:
        - name: my-container
          image: nginx:latest
```

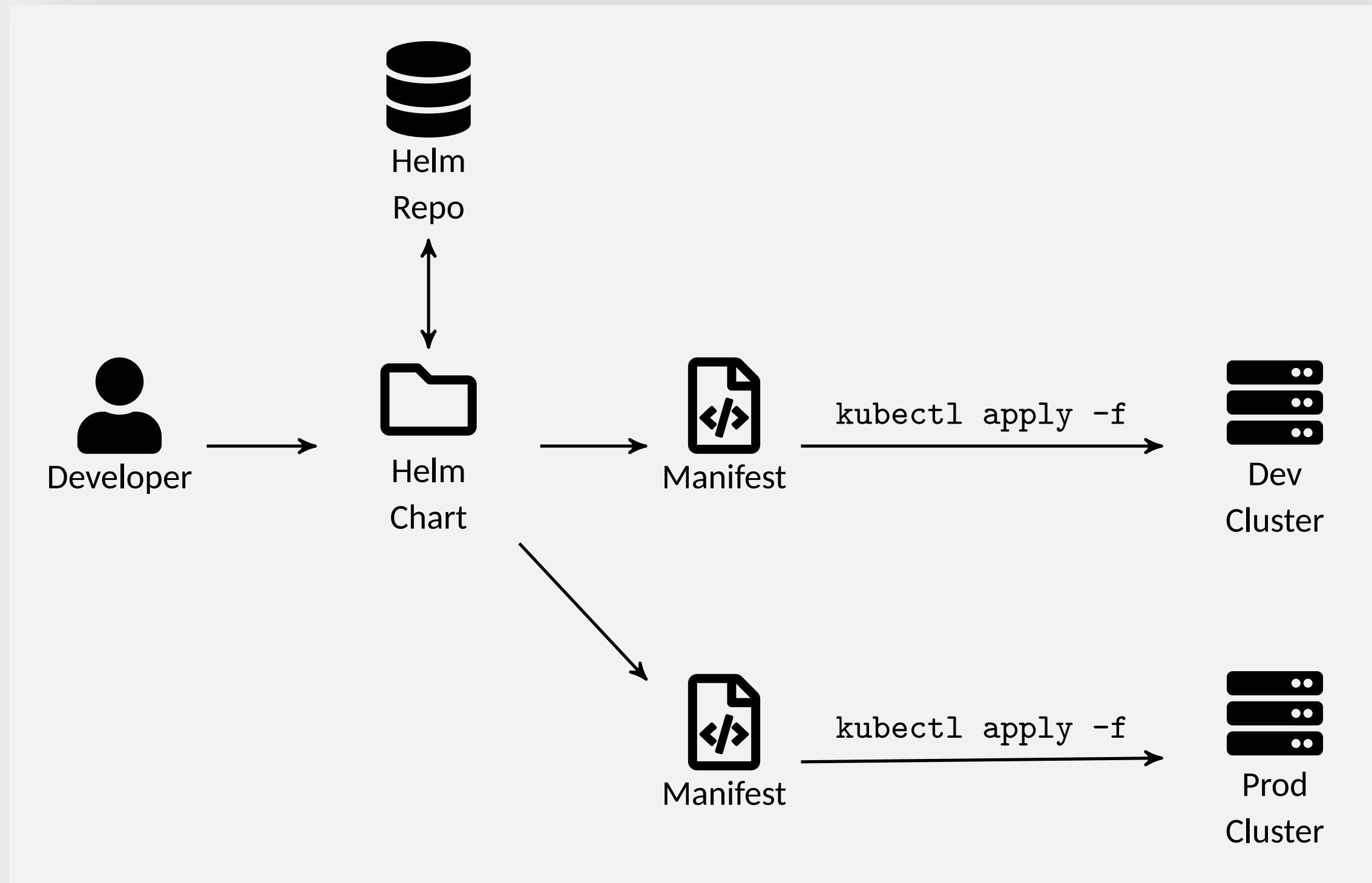
Helm Charts



Helm Charts



Helm Charts



Helm Chart

Sammlung von Templates und Metadaten zur Beschreibung von
Kubernetes-Ressourcen

Chart-Repositories Verwalten

Repository hinzufügen

```
helm repo add bitnami https://charts.bitnami.com/bitnami
```

Repositories anzeigen

```
helm repo list
```

Index aktualisieren

```
helm repo update
```

Charts Suchen

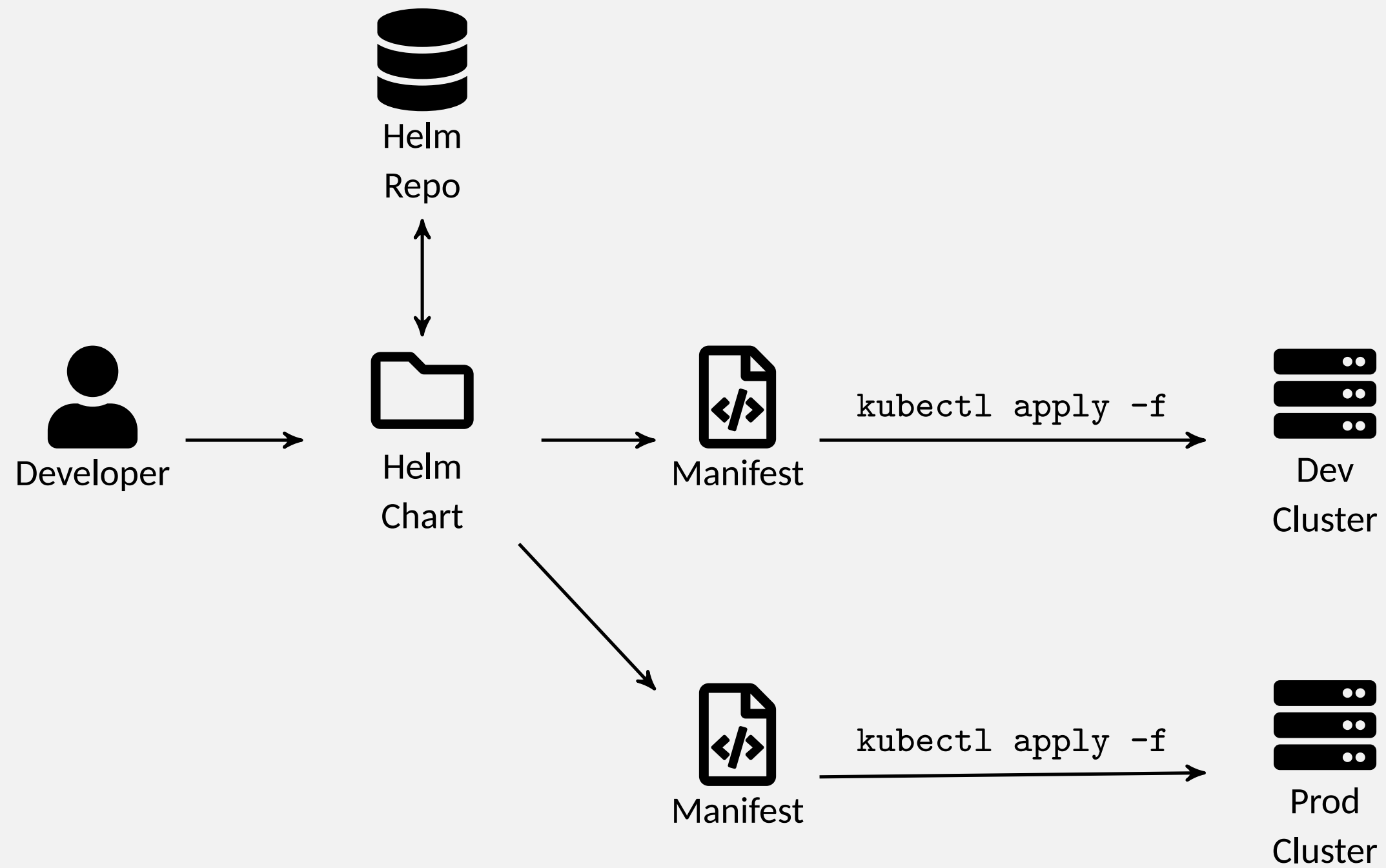
Im lokalen Repo-Index suchen

```
helm search repo nginx  
helm search repo bitnami/nginx --versions
```

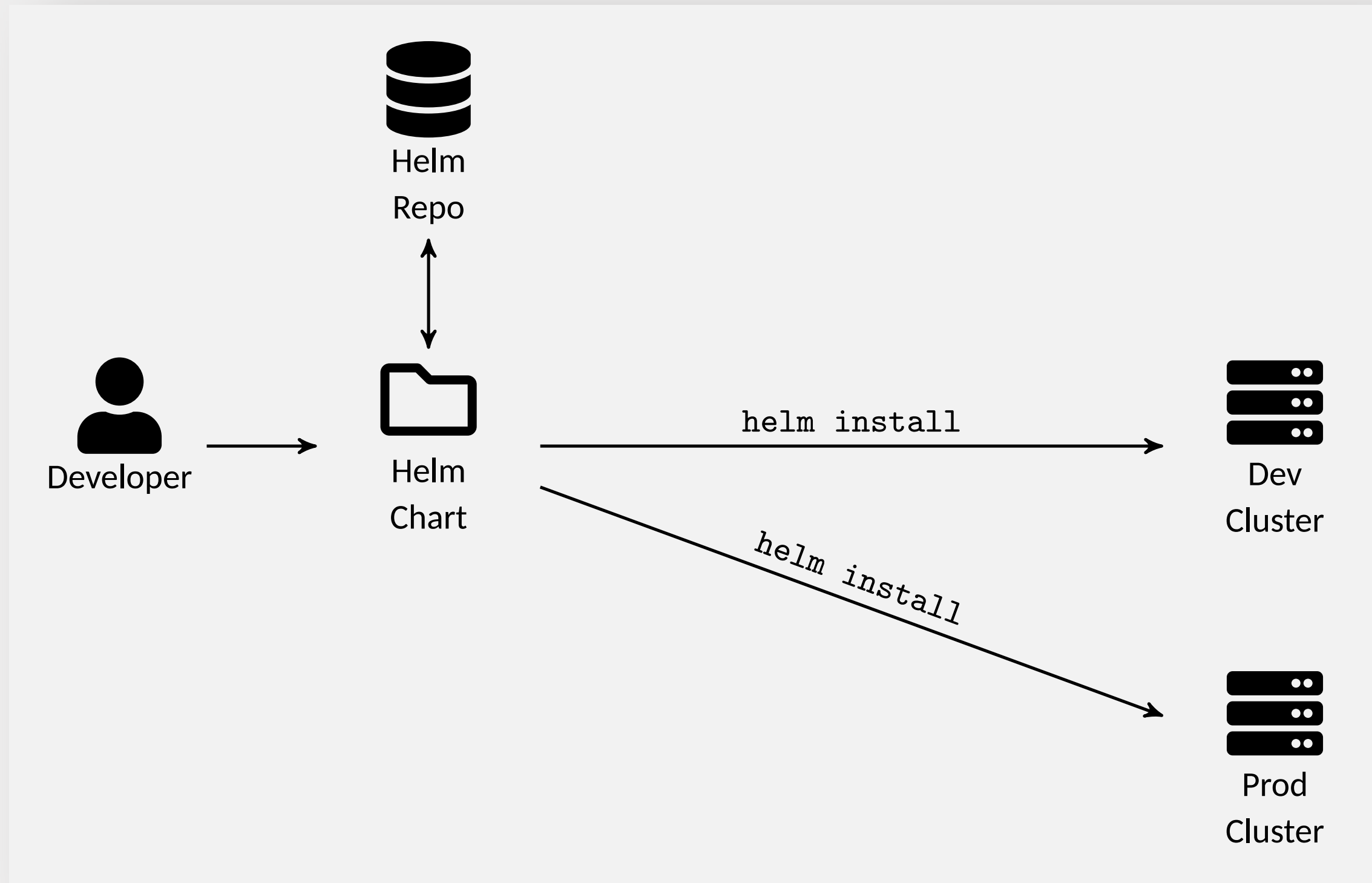
Chart-Metadaten anzeigen

```
helm show chart bitnami/nginx
```

Helm Release



Helm Release



Helm Release

- Instanz eines Helm Charts
 - in einem Kubernetes-Cluster
 - mit eigenem Namen
 - mit eigener Version

Helm Release

- Besteht aus:
 - Manifesten im Cluster
 - Metadaten als Kubernetes-Secret

Release Installieren

```
helm install web bitnami/nginx \
  --namespace web \
  --create-namespace
```

Release-Liste anzeigen

```
helm list --all-namespaces
```

Release-Status prüfen

```
helm status web --namespace web
```

Release Aktualisieren

```
helm upgrade web bitnami/nginx \
  --namespace web \
  --version 18.2.0
```

*Hinweis: Ohne Values-Datei werden die
Default-Werte des Charts genutzt*

Verlauf und Rollback

Historie anzeigen

```
helm history web --namespace web
```

Rollback auf Revision 1

```
helm rollback web 1 --namespace web
```

Nach Rollback Zustand prüfen

```
helm status web --namespace web
```

Release Entfernen

```
helm uninstall web --namespace web
```

Kontrolle

```
helm list --all-namespaces
```

Typischer Basis-Workflow

1. `helm repo add ...`
2. `helm repo update`
3. `helm search repo ...`
4. `helm install ...`
5. `helm upgrade ...`
6. `helm history ...`
7. `helm rollback ...`
8. `helm uninstall ...`

Demo: Helm Basics

Lab

- Gehen Sie auf: <https://learn.corewire.de>
- Navigieren Sie nach:
 - Helm
 - Aufgaben
 - Helm Grundlagen

Zur Kapitelübersicht

- Vorheriges Kapitel: [Pod Management](#)
- Nächstes Kapitel: [Helm Values](#)