

***Kustomizons***  
***nos déploiements avec***  
***style 🌟 !***





**Google**  
**Developer Expert**

  
  
[@davinkevin](https://twitter.com/davinkevin) [@davinkevin.fr](mailto:davinkevin.fr)  




**GitLab Heroes**



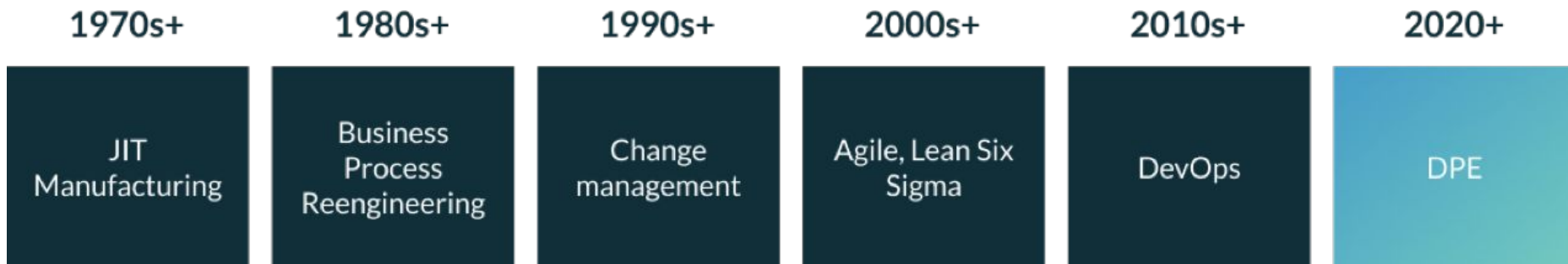


Our mission at **Gradle** is to **accelerate developer productivity** and  
**make developers happier**





What comes after DevOps?

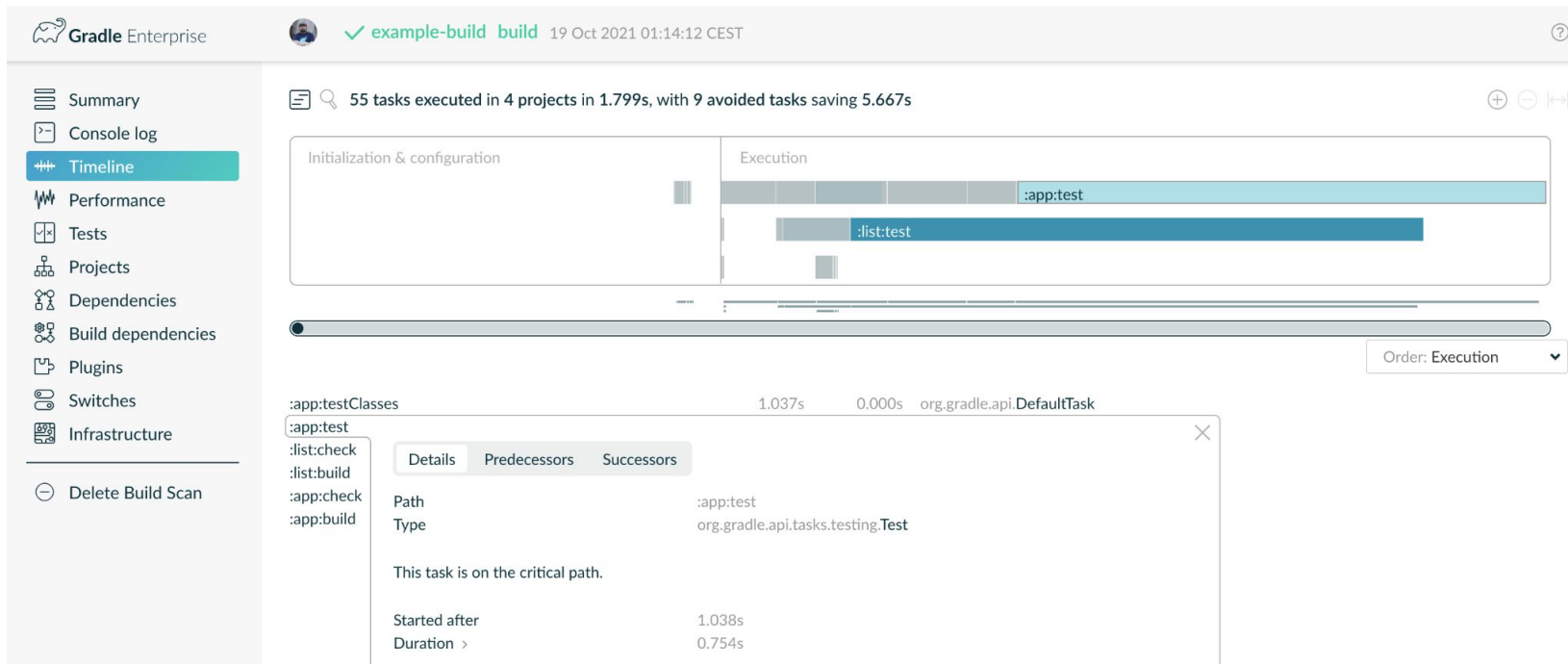


**Developer Productivity Engineering**



# “If you can’t measure it, you can’t improve it!”

by Peter Drucker

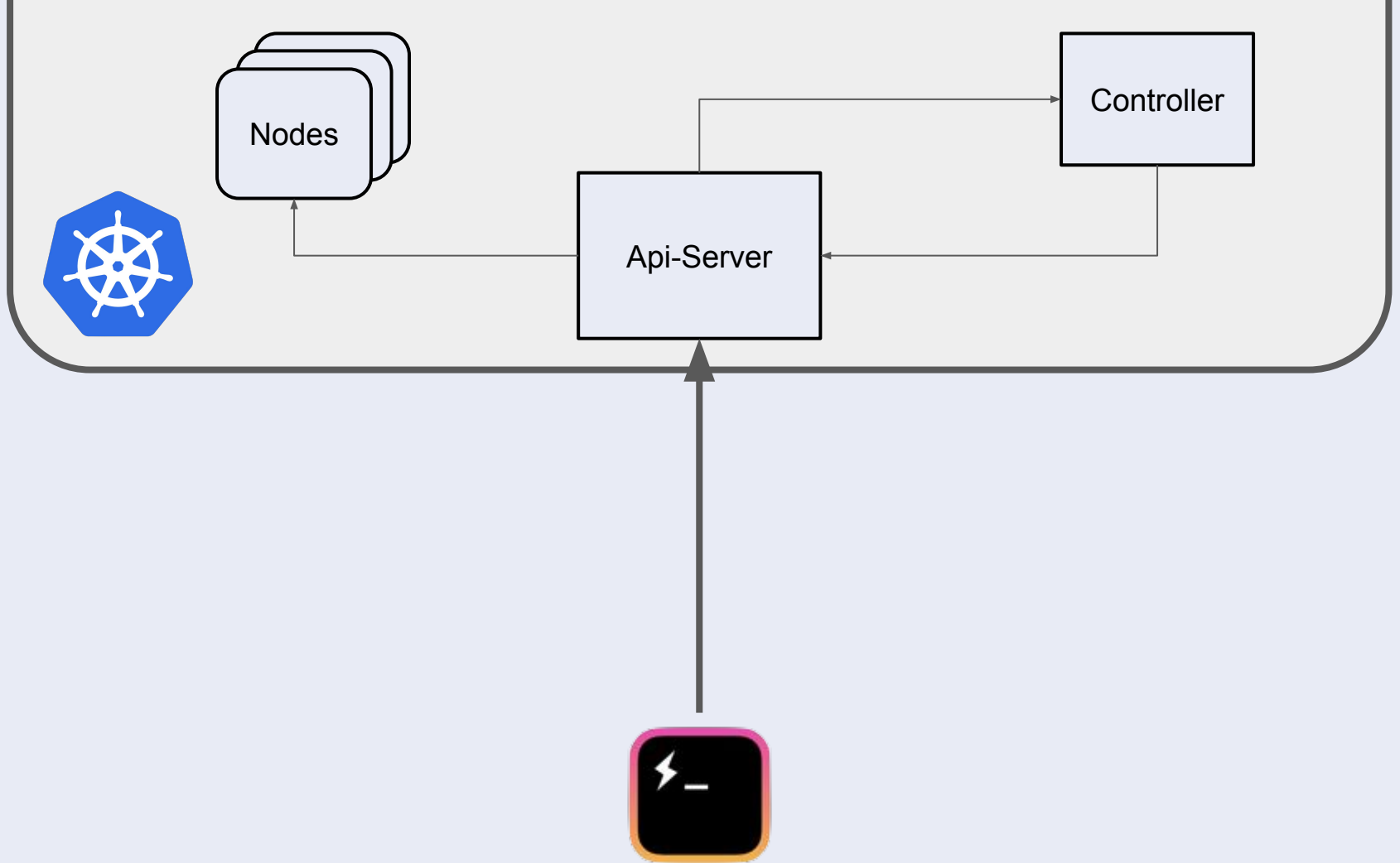








```
$ kubectl run backend --image=docker.bakery.com/backend
```



YOTAM OTTOLENGHI

# PLENTY MORE



AROUND THE WORLD IN 100 DISHES

**real chocolate**

*Chantal Coady*



*Apples for Sam*

Tessa Kiros



**jamie**  
does...

SPAIN ITALY  
MOROCCO SWEDEN  
GREECE FRANCE

PENGUIN  
MICHAEL  
JOSEPH

```
apiVersion: apps/v1      apiVersion: v1      apiVersion: v1
kind: Deployment          kind: Service        kind: ConfigMap
metadata:                 metadata:              metadata:
  name: backend           name: backend        name: backend
spec:                     labels:              data:
  selector:               app: backend      database.user: "foo"
    matchLabels:          spec:              stripe.user: "bar"
      app: backend
template:                 ports:
  metadata:               - name: http
    labels:               port: 8080
      app: backend        targetPort: 8080
spec:                     protocol: TCP
  containers:             selector:
    - name: backend       app: bakery
      image: docker.bakery.com/backend
      ports:
        - name: http
          containerPort: 8080

apiVersion: networking.k8s.io/v1beta1
kind: Ingress
metadata:
  name: backend
spec:
  rules:
    - host: api.bakery.com
      http:
        paths:
          - backend:
              service:
                name: backend
                port:
```





BERAS  
KENCUR  
2600/90

KUNYIT  
ASAM  
2600/90

BANDREK  
2600/90

BE  
CALM SOAP  
ARTESANA  
65k

SOAP  
DAILY  
DETOX

KING OF  
FIRE SOAP  
ARTESANA

ALMOND  
321/90

WALNUT  
281/90

PINE  
NUT  
310/90

MACADAMIA  
140/90

RAISIN  
100/90

BLUEBERRY  
412/90

CRANBERRY  
250/90

PITTED  
PRUNE  
167/90

GOJI  
BERRY  
328/90

APRICOT  
131/90

PUMPKIN  
SEED  
150/90

COCOA  
BEANS  
500/90

MULTI  
115/90

MADU  
LENGKAS  
134/90

TEMULAWAK  
JAWA  
TUNGGAL  
150/90

ANGUR  
SULTAN  
1500/90

TEH  
REMPAH  
2600/90

JAHE  
TELANG  
1500/90

BROWN  
FLAXSEED  
105/90

GOLDEN  
FLAXSEED  
154/90

CHIA  
SEED  
240/90

PURE  
KAKAO  
POWDER  
650/90

BLACK  
HIMALAYA  
SALT  
165/90

SALT  
165/90







```
deploy bakery:
  stage: deploy
  image: docker.backery.com/kubectl
  script:
    - kubectl apply -k k8s/ --dry-run=client
    - kubectl apply -k k8s/ --dry-run=server
    - kubectl apply -k k8s/
```







# flux



# argo

```
apiVersion: source.toolkit.fluxcd.io/v1beta1
kind: GitRepository
metadata:
  name: example
  namespace: flux-system
spec:
  interval: 30s
  ref:
    branch: master
  url: https://gitlab.com/bakery/ops
```







```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
  namespace: bakery
  labels:
    app: bakery
    module: backend
    version: v1
  annotations:
    support_team: support@bakery.com
    support_ticketing: gitlab.com/bakery/backend/issues
    support_alerts: '01-02-03-04-05'
spec:
  ...
```



```
apiVersion: apps/v1
kind: Deployment
metadata:
  ...
spec:
  template:
    spec:
      containers:
        - name: backend
          image: docker.bakery.com/backend:v1.2.3-6b5048
```

...

containers:

- name: backend

...

env:

- name: DATABASE\_USERNAME  
valueFrom:  
  configMapKeyRef:  
    name: backend  
    key: database.username
- name: DATABASE\_PASSWORD  
valueFrom:  
  secretKeyRef:  
    name: backend  
    key: database.password

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
spec:
  replicas: 3
  template:
    spec:
      containers:
      - name: backend
        ...
      resources:
        requests:
          cpu: "1000m"
          memory: "2Gi"
```





```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
spec:
  strategy:
    rollingUpdate:
      maxSurge: 2
      maxUnavailable: 0
    type: RollingUpdate
  template:
    spec:
      containers:
        - name: backend
          ...
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
spec:
  template:
    spec:
      containers:
        - name: backend
          ports:
            - name: http
              containerPort: 8080
          readinessProbe:
            httpGet:
              port: http
              path: /healthz
```





```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
spec:
  template:
    spec:
      containers:
        - name: backend
          ...
          env:
            - name: LOGGING_FORMAT
              value: "GCP_JSON"
          ...
```

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: backend
```



















NGINX



```
apiVersion: {{ template "deployment.apiVersion" . }}
kind: Deployment
metadata:
  labels:
    app: {{ template "nginx-ingress.name" . }}
    chart: {{ template "nginx-ingress.chart" . }}
    heritage: {{ .Release.Service }}
    release: {{ template "nginx-ingress.releaseLabel" . }}
    {{ .Values.controller.componentLabelKeyOverride | default
"app.kubernetes.io/component" }}: controller

    {{- if .Values.controller.deploymentLabels }}
    {{ toYaml .Values.controller.deploymentLabels | indent 4 }}
    {{- end }}
name: {{ template "nginx-ingress.controller.fullname" . }}
annotations:
  {{ toYaml .Values.controller.deploymentAnnotations | indent 4 }}
spec:
  selector:
    matchLabels:
      app: {{ template "nginx-ingress.name" . }}
```





**Frederic**  **Branczyk** @brancz@hachyderm.io  
@fredbrancz



Replying to @bboreham @jimmyzelinskie and @adamhjk

I think in my mind it has boiled down to: helm charts are incredibly easy to consume, but just a horrorshow to produce and maintain. The overwhelming majority of people are users, therefore the strong voice for "I want to easily consume!".

12:49 PM · Feb 7, 2023 · **6,898** Views

---

**1** Quote Tweet   **15** Likes



**VALENTINE**  
Macarons  
1 stk. 15,-  
3 stk. 40,-  
9 stk. 120,-  
12 stk. 150,-  
15 stk. 185,-  
18 stk. 225,-  
30 stk. 375,-

**SWEET VALENTINE**  
Petit four  
Valg mellem:  
Alm. Sarah B.  
Kardé  
Chai  
Mocca  
Passion  
Hindbær  
1 stk. 15,- kr.  
3 stk. 40,- kr.

**SWEET VALENTINE**  
Mini flødeboller  
Flere forskellige  
Pr. stk. 15,-  
3 stk. 40,-

**SWEET VALENTINE**  
Træstammer  
25,- pr. stk.



**Bartarbe**  
35 kr. stk

**SWEET VALENTINE**  
Kartoffelkage  
Pr. stk. 35,-  
35,- kr.

**SWEET VALENTINE**  
Kartoffelkage  
Pr. stk. 35,-  
Silk vanilla sponge with  
caramel, whipped cream  
and raspberry. Covered  
with marzipan and cocoa

**Bryllups kagen**  
chokoladekage-  
eller kagebunde  
Hindbærmousse  
Friske hindbær  
Chokoladekris  
Frosting  
Dryppende saltkaramel

**Rabarber**  
5-kuverts hindbær  
Pr. stk. 130,- kr.

**SWEET VALENTINE**  
5-kuverts chokoladekage  
på chokoladebunde med  
syltet appelsin  
Pr. stk. 130,- kr.



**SWEET VALENTINE**  
Vaniljemousse  
med chokoladekris  
38,- pr. stk.

**SWEET VALENTINE**  
Passionfrugtmousse på  
chokoladebunde  
38,- pr. stk.

**SWEET VALENTINE**  
Hindbær-mousse på  
chokoladebunde  
38,- pr. stk.

**SWEET VALENTINE**  
Hvid chokolademousse  
med lakrids på  
mandelbunde  
38,- pr. stk.

**SWEET VALENTINE**  
Hindbær-mousse på  
chokoladebunde  
38,- pr. stk.

**SWEET VALENTINE**  
Hvid chokolademousse  
med lakrids på  
mandelbunde  
38,- pr. stk.













▼ repo

- ⚙️ backend.yaml
- ⚙️ database.yaml
- ⚙️ frontend.yaml
- ⚙️ fs.yaml

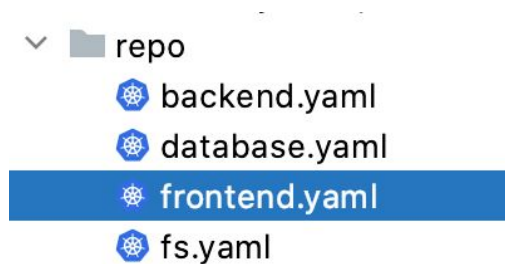
```
# backend.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
...
---
apiVersion: v1
kind: Service
metadata:
  name: backend
...
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: backend
spec:
  rules:
    - host: api.bakery.com
...

```

▼ repo

- backend.yaml
- database.yaml**
- frontend.yaml
- fs.yaml

```
# database.yaml
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: database
...
---
apiVersion: v1
kind: Service
metadata:
  name: database
...
---
apiVersion: v1
kind: ConfigMap
metadata:
  name: database
...
---
apiVersion: v1
kind: Secret
metadata:
  name: database
...
```



```
# frontend.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: frontend
...
---
apiVersion: v1
kind: Service
metadata:
  name: frontend
...
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: frontend
spec:
  rules:
    - host: bakery.com
...

```



- ▼ repo
  - backend.yaml
  - database.yaml
  - frontend.yaml
  - fs.yaml**

```
# fs.yaml
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: fs
...
---
apiVersion: v1
kind: Service
metadata:
  name: fs
...
---
apiVersion: v1
kind: ConfigMap
metadata:
  name: fs
...
---
apiVersion: v1
kind: Secret
metadata:
  name: fs
...
```



▼ repo

- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
- kustomization.yaml**

```
# kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization

resources:
- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
```







```
$ kubectl apply -k repo
```

```
$ kustomize build repo | kubectl apply -f -
```



▼ repo

- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
- kustomization.yaml**

```
# kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization

resources:
- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
```



```
# database.yaml
```

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: backend
data:
  url: "jdbc:postgresql://database:5432/bakery"
  username: "backend-user"
```

```
---
```

```
apiVersion: v1
kind: Secret
metadata:
  name: backend
data:
  password: "WxlXRGlHalk="
```

```
# kustomization.yaml
```

```
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
resources:
```

- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml

```
configMapGenerator:
```

- name: database
- literals:
  - url="jdbc:postgresql://database:5432/bakery"
  - username="backend-user"

```
secretGenerator:
```

- name: database
- literals:
  - password="YyWDiGjY"

```
# kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
...
```

```
configMapGenerator:
```

```
- name: database
```

```
  literals:
```

```
    - ...
```

```
  env:
```

```
    - ...
```

```
  files:
```

```
    - ...
```

```
...
```





```
# result.yaml
apiVersion: v1
data:
  url: jdbc:postgresql://database:5432/bakery
  username: backend-user
kind: ConfigMap
metadata:
  name: database-kb4tdd68h6
---
apiVersion: v1
data:
  password: WXlXR6lHaIk=
kind: Secret
metadata:
  name: database-76dm9c8t89
type: Opaque
```

```
# result.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
spec:
  template:
    spec:
      containers:
        - env:
            - name: DATABASE_URL
              valueFrom:
                configMapKeyRef:
                  key: url
                  name: database-kb4tdd68h6
            - name: DATABASE_USERNAME
              valueFrom:
                configMapKeyRef:
                  key: username
                  name: database-kb4tdd68h6
            - name: DATABASE_PASSWORD
              valueFrom:
                secretKeyRef:
                  key: password
                  name: database-76dm9c8t89
          image: docker.bakery.com/backend
          name: backend
```



▼ repo

- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
- kustomization.yaml**

```
# kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
namespace: feature-branch-XYZ
```

```
resources:
- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
```

```
...
```

```
...
apiVersion: v1
data:
  url: jdbc:postgresql://database:5432/bakery
  username: backend-user
kind: ConfigMap
metadata:
  name: database-kb4tdd68h6
  namespace: feature-branch-XYZ
...
```



▼ repo

- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
- kustomization.yaml**

```
# kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization

namePrefix: dev-
nameSuffix: -kevin

resources:
- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml

...
```

```
...  
apiVersion: apps/v1  
kind: Deployment  
metadata:  
  name: dev-backend-kevin  
...
```

▼ repo

- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
- kustomization.yaml**

```
# kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization

images:
- name: docker.bakery.com/backend
  newName: inner.bakery.com/backend
  newTag: 1.6.8
- name: docker.bakery.com/frontend
  newName: inner.bakery.com/frontend
  digest: f2fff79f2f2bf2d2418e3...70ccdfcc3f661a086b

resources:
- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
```

...

```
""
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
spec:
  selector:
    ""
  template:
    ""
    spec:
      containers:
        - image: inner.bakery.com/backend:1.6.8
      ""
```

```
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: frontend
spec:
  selector:
    ""
  template:
    ""
    spec:
      containers:
        - image: inner.bakery.com/frontend@sha256:f2fff79f2f2bf2d2418e3...70ccdfcc3f661a086b
      ""
```



▼ repo

- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
- kustomization.yaml**

```
# kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
replicas:
- name: backend
  count: 3
```

```
resources:
- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
```

...

```
...
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
spec:
  replicas: 3
  selector:
    matchLabels:
      app: backend
  template:
    metadata:
      labels:
        app: backend
    spec:
      containers:
        - image: docker.bakery.com/backend
...

```

▼ repo

- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml
- kustomization.yaml**

```
# kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization

...

helmCharts:
- name: extra-component
  valuesInline:
    foo:
      bar: true
      noo: false
    another:
      value: 'xyz'
  releaseName: Xtra
  version: 1.2.3
  repo: https://gitlab.bakery.com/extra-component
```







▼  repo

▼  base













 backend.yaml

 database.yaml

 frontend.yaml

 fs.yaml

 kustomization.yaml

- ▼  repo
  - ▼  base
    -  backend.yaml
    -  database.yaml
    -  frontend.yaml
    -  fs.yaml
    -  kustomization.yaml
  - ▼  overlays
    - ▼  dev
      -  kustomization.yaml
    - ▼  prod
      -  kustomization.yaml

- ▼ repo
  - ▼ base
    - backend.yaml
    - database.yaml
    - frontend.yaml
    - fs.yaml
    - kustomization.yaml**
  - ▼ overlays
    - ▼ dev
      - kustomization.yaml
    - ▼ prod
      - kustomization.yaml

```
# base/kustomization.yaml
```

```
apiVersion: kustomize.config.k8s.io/v1beta1
```

```
kind: Kustomization
```

```
resources:
```

- backend.yaml
- database.yaml
- frontend.yaml
- fs.yaml

```
configMapGenerator:
```

- name: database
- ```
  literals:
```
- url=jdbc:postgresql://database:5432/bakery

- ▼ repo
  - ▼ base
    - backend.yaml
    - database.yaml
    - frontend.yaml
    - fs.yaml
    - kustomization.yaml
  - overlays
    - ▼ dev
      - kustomization.yaml
    - prod
      - kustomization.yaml

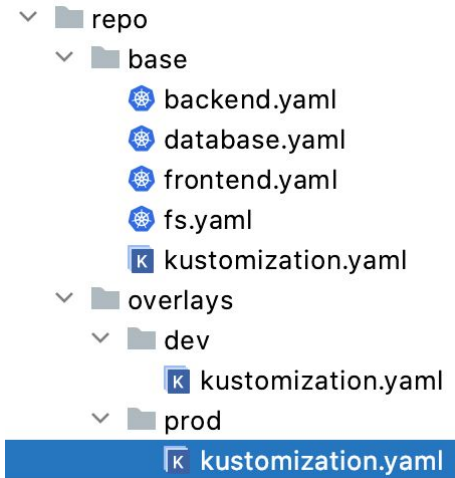
```
# overlays/dev/kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
resources:
- ../../base/
```

```
configMapGenerator:
- name: database
  behavior: merge
  literals:
- username="backend-dev-user"
```

```
secretGenerator:
- name: database
  literals:
- password="YyWDiGjY"
```





```
# overlays/prod/kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
resources:
- ../../base/
```

```
configMapGenerator:
- name: database
  behavior: merge
  literals:
- username="backend-prod-user"
```

```
secretGenerator:
- name: database
  literals:
- password="J9EtVV4K"
```

- ▼ repo
  - ▼ base
    - kustomization.yaml**
  - ▼ overlays
    - ▼ dev
      - kustomization.yaml
    - ▼ prod
      - kustomization.yaml

```
# base/kustomization.yaml
```

```
apiVersion: kustomize.config.k8s.io/v1beta1
```

```
kind: Kustomization
```

```
resources:
```

```
- gitlab.bakery.com/ops/base?ref=v1.2.3
```

```
configMapGenerator:
```

```
- name: database
```

```
  literals:
```

```
- url=jdbc:postgresql://database:5432/bakery
```



- ▼ repo
  - ▼ base
    - ⚙ backend.yaml
    - ⚙ database.yaml
    - ⚙ frontend.yaml
    - ⚙ fs.yaml
    - 📦 kustomization.yaml
  - ▼ overlays
    - ▼ dev
      - ▼ backend
        - ⚙ log\_as\_text.yaml
        - 📦 kustomization.yaml
    - ▼ prod
      - ▼ backend
        - ⚙ log\_as\_json.yaml
        - 📦 kustomization.yaml

```
# overlays/dev/backend/log_as_text.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
spec:
  template:
    spec:
      containers:
        - name: backend
          env:
            - name: LOG_FORMAT
              value: text
```



- ▼ repo
  - ▼ base
    - ⚙ backend.yaml
    - ⚙ database.yaml
    - ⚙ frontend.yaml
    - ⚙ fs.yaml
    - Ⓚ kustomization.yaml
  - ▼ overlays
    - ▼ dev
      - ▼ backend
        - ⚙ log\_as\_text.yaml
        - Ⓚ kustomization.yaml
    - ▼ prod
      - ▼ backend
        - ⚙ log\_as\_json.yaml
        - Ⓚ kustomization.yaml

```
# overlays/dev/kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization

resources:
- ../../base/

patches:
- path: backend/log_as_text.yaml

...
```

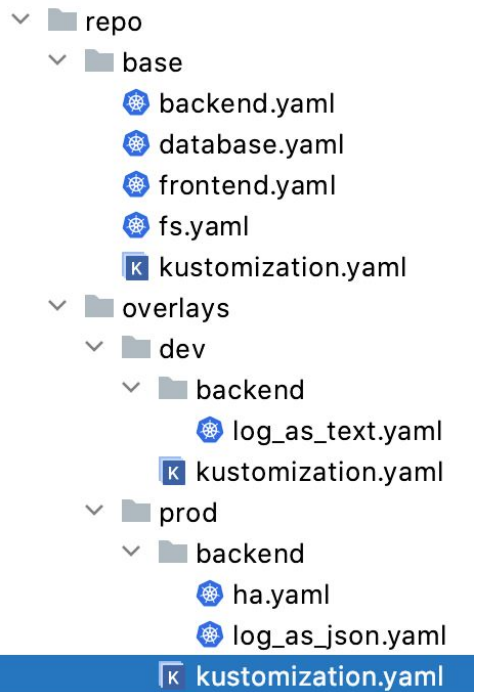
```
$ kubectl apply -k repo/overlays/dev/
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    ...
  name: backend
spec:
  selector:
    ...
  template:
    ...
    spec:
      containers:
      - env:
        - name: LOG_FORMAT
          value: text
        - name: DATABASE_URL
          valueFrom:
            configMapKeyRef:
              key: url
              name: database-7mt876452m
        - name: DATABASE_USERNAME
          valueFrom:
            configMapKeyRef:
              key: username
              name: database-7mt876452m
        - ...
      image: docker.bakery.com/backend
      name: backend
```

- ▼ repo
  - ▼ base
    - ⚙️ backend.yaml
    - ⚙️ database.yaml
    - ⚙️ frontend.yaml
    - ⚙️ fs.yaml
    - 📄 kustomization.yaml
  - ▼ overlays
    - ▼ dev
      - ▼ backend
        - ⚙️ log\_as\_text.yaml
        - 📄 kustomization.yaml
    - ▼ prod
      - ▼ backend
        - ⚙️ ha.yaml
        - ⚙️ log\_as\_json.yaml
        - 📄 kustomization.yaml

```
# overlays/prod/backend/ha.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
spec:
  template:
    spec:
      affinity:
        podAntiAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            - labelSelector:
                matchExpressions:
                  - key: app
                    operator: In
                    values:
                      - backend
              topologyKey: topology.kubernetes.io/zone
```





```
# overlays/prod/kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
resources:
- ../../base/
```

```
patches:
- path: backend/ha.yaml
- path: backend/log_as_json.yaml
```

```
...
```

```
$ kubectl apply -k repo/overlays/prod/
```

```
...
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: backend
    name: backend
spec:
  selector:
    ...
  template:
    metadata:
      labels:
        app: backend
    spec:
      affinity:
        podAntiAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            - labelSelector:
                matchExpressions:
                  - key: app
                    operator: In
                    values:
                      - backend
              topologyKey: topology.kubernetes.io/zone
      containers:
        - env:
            - name: LOG_FORMAT
              value: json
          ...
```





- ▼ repo
  - ▼ base
    - ⚙️ backend.yaml
    - ⚙️ database.yaml
    - ⚙️ frontend.yaml
    - ⚙️ fs.yaml
    - Ⓚ kustomization.yaml
  - ▼ overlays
    - ▼ dev
      - ▼ backend
        - ⚙️ log\_as\_text.yaml
      - ▼ frontend
        - 📄 ingress\_host.yaml
        - Ⓚ kustomization.yaml
    - > prod

```
# overlays/dev/frontend/ingress_host.yaml
- op: replace
  path: /spec/rules/0/host
  value: dev.api.bakery.com
```

- ▼ repo
  - ▼ base
    - backend.yaml
    - database.yaml
    - frontend.yaml
    - fs.yaml
    - kustomization.yaml
  - ▼ overlays
    - ▼ dev
      - ▼ backend
        - log\_as\_text.yaml
      - ▼ frontend
        - ingress\_host.yaml
    - kustomization.yaml
  - > prod

```
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
resources:
- ../../base/
```

```
patches:
- path: backend/log_as_text.yaml
```

```
patches:
- path: frontend/ingress_host.yaml
  target:
    group: networking.k8s.io
    version: v1
    kind: Ingress
    name: frontend
```

...

```
$ kubectl apply -k repo/overlays/dev/
```

```
...
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  labels:
    app: frontend
    module: frontend
  name: frontend
spec:
  rules:
  - host: dev.api.bakery.com
    http:
      paths:
      - backend:
          service:
            name: frontend
            port:
              number: 80
```

...





```
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
resources:
- ../../base/
```

```
configMapGenerator:
- name: bakery
  behavior: merge
  literals:
  - DEFAULT_DOMAIN=test1.api.bakery.com
```




```
replacements:
- source:
    kind: ConfigMap
    name: bakery
    fieldPath: data.DEFAULT_DOMAIN
  targets:
  - select:
      kind: Ingress
      name: frontend
    fieldPaths:
    - spec.rules.0.host
    - spec.tls.0.secretName
    - spec.tls.0.hosts.0
    options:
      create: true
```

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  labels:
    app: frontend
    module: frontend
  name: frontend
spec:
  tls:
  - hosts:
    - test1.api.bakery.com
      secretName: test1.api.bakery.com
  rules:
  - host: test1.api.bakery.com
    http:
      paths:
      - backend:
          service:
            name: frontend
            port:
              number: 80
```

...





- ▼  repo
  - >  base
  - ▼  overlays
    - >  google\_cloud
    - ▼  on-prem
      - >  base
      - ▼  overlays
        - >  site\_a
        - >  site\_b
        - >  site\_c
    - >  scaleway

- ▼ repo
  - > base
  - ▼ overlays
    - ▼ google\_cloud
      - > base
      - ▼ overlays
        - > int
        - > prod
    - ▼ on-prem
      - > base
      - ▼ overlays
        - ▼ site\_a
          - > base
          - ▼ overlays
            - > int
            - > prod
        - ▼ site\_b
          - > base
          - ▼ overlays
            - > int
            - > prod
        - ▼ site\_c
          - > base
          - ▼ overlays
            - > int
            - > prod
    - ▼ scaleway
      - > base
      - ▼ overlays
        - > int
        - > prod

- ▼ repo
  - > base
  - ▼ overlays
    - ▼ google\_cloud
      - ▼ base
        - ▼ backend
          - log\_format\_gcp.yaml
          - kustomization.yaml
      - ▼ overlays
        - ▼ int
          - kustomization.yaml
        - > prod
    - ▼ on-prem
      - > site\_a
      - > site\_b
    - ▼ scaleway
      - > base
      - ▼ overlays
        - > int
        - > prod

```
#overlays/google_cloud/base/backend/log_format_gcp.yaml
```

```
apiVersion: apps/v1
```

```
kind: Deployment
```

```
metadata:
```

```
  name: backend
```

```
spec:
```

```
  template:
```

```
    spec:
```

```
      containers:
```

```
        - name: backend
```

```
          env:
```

```
            - name: LOG_FORMAT
```

```
              value: gcp
```

- ▼ repo
  - > base
  - ▼ overlays
    - ▼ google\_cloud
      - ▼ base
        - ▼ backend
          - ⚙️ log\_format\_gcp.yaml
          - kustomization.yaml**
    - ▼ overlays
      - ▼ int
        - kustomization.yaml**
      - > prod
  - ▼ on-prem
    - > site\_a
    - > site\_b
  - ▼ scaleway
    - > base
    - ▼ overlays
      - > int
      - > prod

```
#overlays/google_cloud/base/kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
resources:
- ../../../../base/
```

```
patches:
- path: backend/log_format_gcp.yaml
```



- ▼ repo
  - > base
  - ▼ overlays
    - ▼ google\_cloud
      - ▼ base
        - ▼ backend
          - ⚙ log\_format\_gcp.yaml
          - 📄 kustomization.yaml
      - ▼ overlays
        - ▼ int
          - 📄 kustomization.yaml
    - > prod
  - ▼ on-prem
    - > site\_a
    - > site\_b
  - ▼ scaleway
    - > base
    - ▼ overlays
      - > int
      - > prod

```
#overlays/google_cloud/overlays/int/kustomization.yaml
```

```
apiVersion: kustomize.config.k8s.io/v1beta1
```

```
kind: Kustomization
```

```
resources:
```

```
- ../../base
```

```
configMapGenerator:
```

```
- name: database
```

```
  behavior: merge
```

```
  literals:
```

```
    - username="backend-int-user"
```

```
secretGenerator:
```

```
- name: database
```

```
  literals:
```

```
    - password="0to70rdN"
```



- ▼ repo
  - > base
  - ▼ overlays
    - ▼ google\_cloud
      - > base
      - > overlays
    - ▼ on-prem
      - > base
      - ▼ overlays
        - ▼ site\_a
          - ▼ base
            - ▼ backend
              - auth\_with\_openldap.yaml
              - kustomization.yaml
              - openldap.yaml
            - > overlays
          - ▼ site\_b
            - ▼ base
              - kustomization.yaml
            - > overlays
          - ▼ site\_c
            - ▼ base
              - ▼ backend
                - auth\_with\_openldap.yaml
                - kustomization.yaml
                - openldap.yaml
              - > overlays
      - ▼ scaleway
        - > base
        - > overlays



- ▼ repo
  - ▼ base
    - ⚙️ backend.yaml
    - ▼ components
      - ▼ openldap
        - ▼ backend
          - ⚙️ auth\_with\_openldap.yaml
          - kustomization.yaml**
          - ⚙️ openldap.yaml
        - ⚙️ database.yaml
        - ⚙️ frontend.yaml
        - ⚙️ fs.yaml
        - k** kustomization.yaml
    - > overlays

```
# base/components/openldap/kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1alpha1
kind: Component

resources:
- openldap.yaml

patches:
- path: backend/auth_with_openldap.yaml
```











- ▼ repo
  - ▼ base
    - ⚙️ backend.yaml
    - ▼ components
      - ▼ openldap
        - ▼ backend
          - ⚙️ auth\_with\_openldap.yaml
          - 📄 kustomization.yaml
          - ⚙️ openldap.yaml
        - ⚙️ database.yaml
        - ⚙️ frontend.yaml
        - ⚙️ fs.yaml
        - 📄 kustomization.yaml
      - ▼ overlays
        - > google\_cloud
        - ▼ on-prem
          - > base
          - ▼ overlays
            - ▼ site\_a
              - ▼ base
                - 📄 kustomization.yaml
        - > scaleway

```
#  
overlays/on-prem/overlays/site_a/base/kustomizat  
ion.yaml  
apiVersion: kustomize.config.k8s.io/v1beta1  
kind: Kustomization  
  
resources:  
- ../../../../base  
  
components:  
- ../../../../base/components/openldap
```

# Composition > Inheritance



- ✓  bakery
  - >  base
  - ✓  overlays
    - >  bakery.com
    - >  int.aws.bakery.com
    - >  int.azure.bakery.com
    - >  int.gcp.bakery.com
    - >  int.scaleway.bakery.com
    - >  team-candy.dev.bakery.com
    - >  team-oven.dev.bakery.com









containers





SOPS

## SOPS: Secrets OPerationS

Simple and flexible tool for managing secrets

 40 followers  <https://getsops.io>  @getsops

Verified

```
apiVersion: sops.davinkevin.fr/v1alpha1
kind: SopsSecret
metadata:
  name: database
  annotations:
    config.kubernetes.io/function: |
      container:
        image: davinkevin.fr/krm-sops-inflator
        envs:
          - "AGE_KEY"
```

```
data:
  password:
ENC[AES256_GCM,data:ADB8h5z7Cs44BXCbz4ppcStylP26/CbK/corr
A=,iv:ZJ1m7/yfsjUfJM+cjI5GPEEppEh3M/CYpo7oqErCGyw=,tag:d
Nem/u+RV26s8o6yhWL2ew=,type:str]
sops:
```

...



containers

```
apiVersion: sops.davinkevin.fr/v1alpha1
```

```
kind: SopsSecret
```

```
metadata:
```

```
  name: database
```

```
  annotations:
```

```
    config.kubernetes.io/function: |
```

```
    container:
```

```
      image: davinkevin.fr/krm-sops-inflator
```

```
      envs:
```

```
        - "AGE_KEY"
```

```
data:
```

```
  password:
```

```
ENC[AES256_GCM,data:ADB8h5z7Cs44BXCbz4ppcStyLP26/CbK/corr  
A=,iv:ZJ1m7/yfsjUfJM+cjI5GPEEppEh3M/CYpo7oqErCGyw=,tag:d  
Nem/u+RV26s8o6yhWL2ew=,type:str]
```

```
sops:
```

```
  ""
```



```
#!/bin/bash
set -euo pipefail

mkdir -p $XDG_CONFIG_HOME/sops/age/
echo $AGE_KEY > $XDG_CONFIG_HOME/sops/age/keys.txt

KFN_INPUT="$WORKING_DIR/input.yaml"
cat > $KFN_INPUT

SOPS_SECRET="$WORKING_DIR/secret.yaml"
cat $KFN_INPUT | \
  yq e '.functionConfig' - | \
  yq '.kind = "Secret"' - | \
  yq '.apiVersion = "v1"' - \
  > $SOPS_SECRET

sops -d $SOPS_SECRET
```

```
# repo/overlays/dev/kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

```
...

generators:
- sops.secret.enc.yaml
```

```
# repo/overlays/dev/sops.secret.enc.yaml
apiVersion: k8s.bakery.com/v1
kind: SopsSecret
metadata:
  name: database
data:
  password:
ENC[AES256_GCM,data:x3ZhXnPrdJ6D+6wF0BmV
spbJaYmgD9u6ZSU/mUztq4zu0vdX4BmRozVwvps=
,iv:WCnyc/9gjnVKucgIWPdNwLX0KQ+k2s54xJxK
mAfbRSw=,tag:pPbdw7DC0PReoa9HU7k5AQ==,ty
pe:str]
sops:
```

```
...
```

```
apiVersion: v1
kind: Secret
metadata:
  name: database
data:
  password: "Z1U5RHIwZm83WkN...Q4em9GdTgK"
```

```
# repo/overlays/dev/kustomization.yaml
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
```

The diagram illustrates a Kustomization workflow. A central Kustomization resource (top) points via arrows to two target resources. The left target is a Secret resource (bottom-left), and the right target is a transformed Secret resource (bottom-right). The Kustomization resource lists the Secret and a transformer as its resources.

```
resources:
- sops.secret.enc.yaml

transformers:
- sops-transformer.yaml
```

```
# repo/overlays/dev/sops.secret.enc.yaml
apiVersion: v1
kind: Secret
metadata:
  name: database
data:
  password:
ENC[AES256_GCM,data:x3ZhXnPrdJ6D+6wF0BmV
spbJaYmgD9u6ZSU/mUztq4zu0vdX4BmRozVwvps=
,iv:WCnyc/9gjnVKucgIWPdNwLX0KQ+k2s54xJxK
mAfbRSw=,tag:pPbdw7DC0PReoa9HU7k5AQ==,ty
pe:str]
sops:
```

```
apiVersion: v1
kind: Secret
metadata:
  name: database
data:
  password: "Z1U5RHIwZm83WkN...Q4em9GdTgK"
```







[@davinkevin](https://twitter.com/davinkevin) [@davinkevin.fr](https://www.instagram.com/davinkevin)



# *Question(s) ?*



# WE'RE HIRING



Gradle Enterprise



Gradle Build Tool

[www.gradle.com/careers](https://www.gradle.com/careers)