Clojure Redeployed

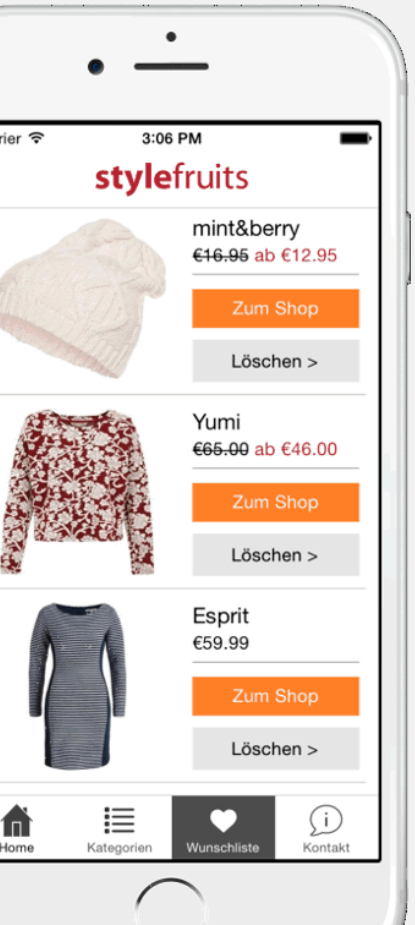
Jan Stępień @janstepien jan@stepien.cc

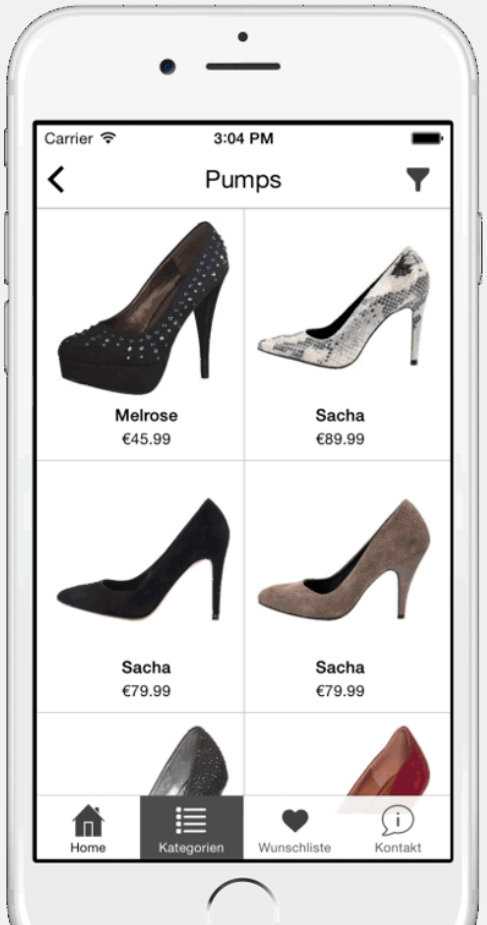

© iwillbehomesoon 2012, flickr.com/photos/jp_photo_online/6964634964

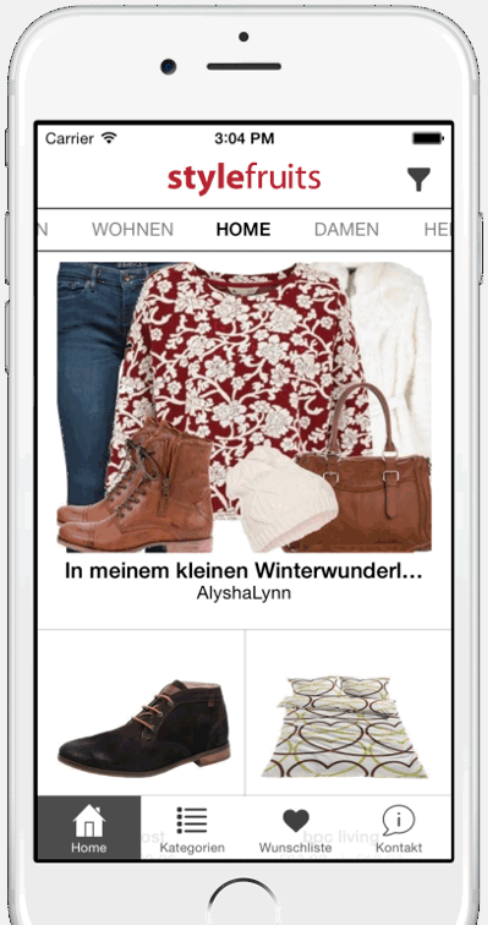
I work at

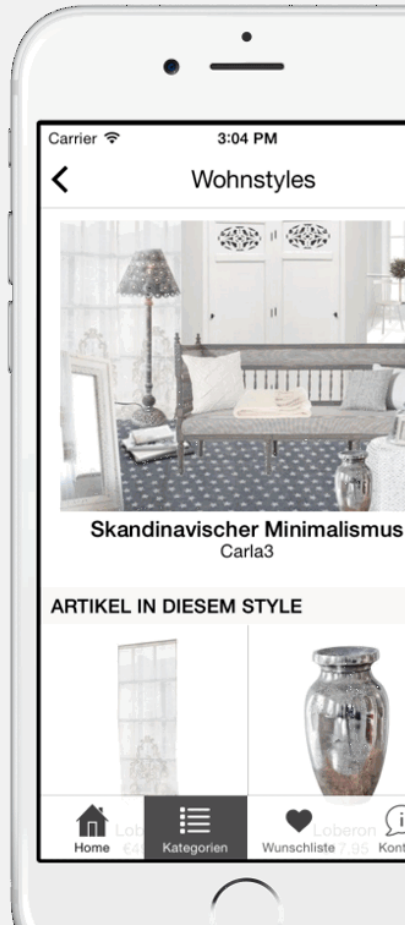
stylefruits

I work at

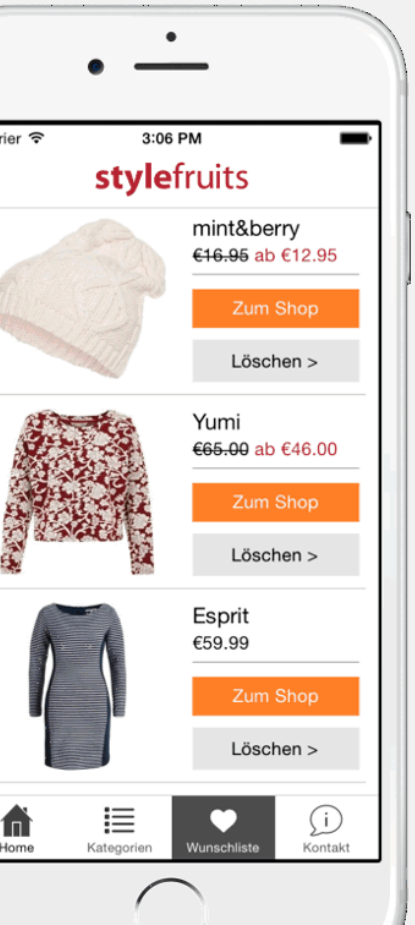

3:06 PM
stylefruits
mint&berry
€16.95 ab €12.95
Zum Shop
Löschen >
Yumi
€65.00 ab €46.00
Zum Shop
Löschen >
Esprit
€59.99
Zum Shop
Löschen >
Home
Kategorien
Wunschliste
Kontakt

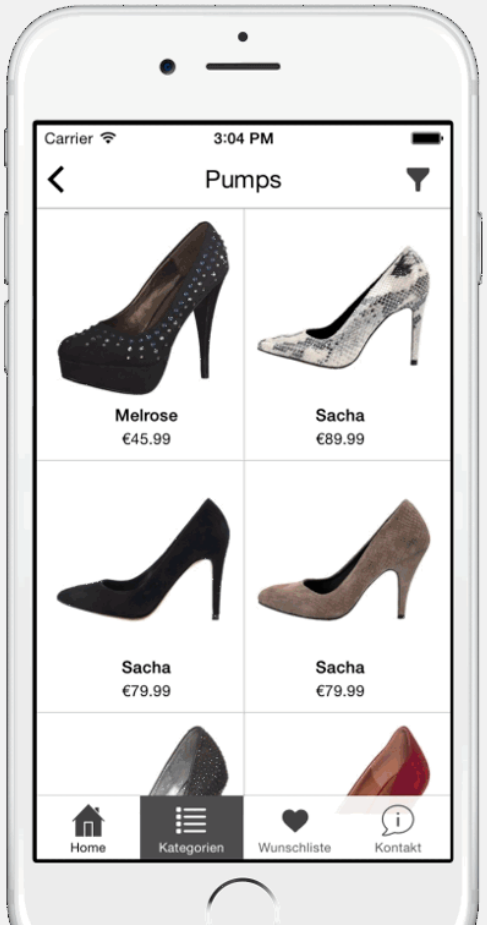

Carrier
3:04 PM
Pumps
Melrose
€45.99
Sacha
€89.99
Sacha
€79.99
Sacha
€79.99
Home
Kategorien
Wunschliste
Kontakt

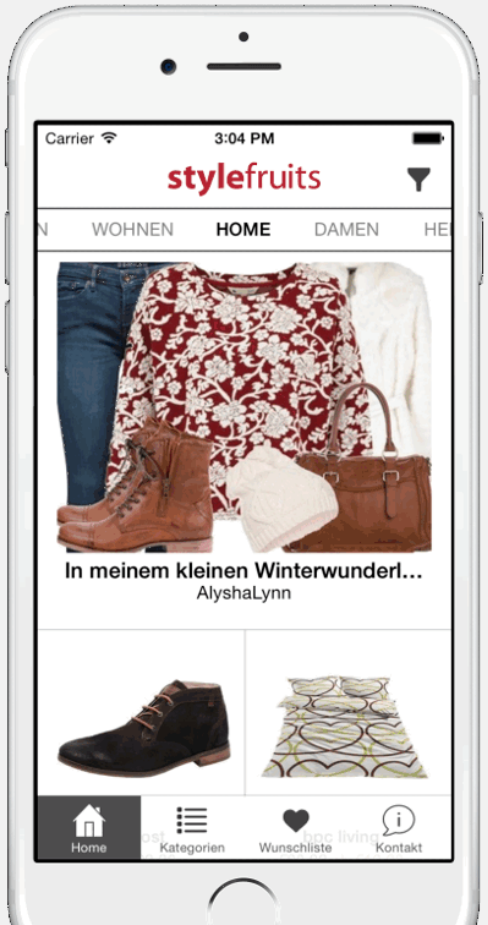

Carrier
3:04 PM
stylefruits
WOHNEN
HOME
DAMEN
In meinem kleinen Winterwunderl...
AlyshaLynn
Home
Kategorien
Wunschliste
Kontakt

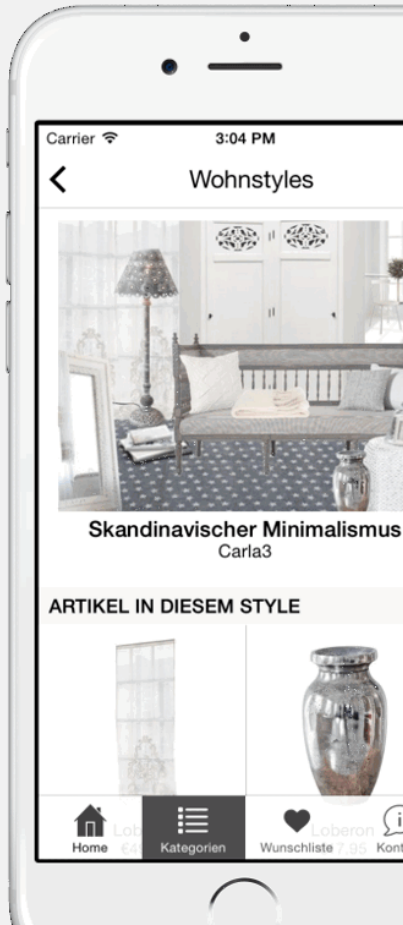

Carrier
3:04 PM
Wohnstyles
Skandinavischer Minimalismus
Carla3
ARTIKEL IN DIESEM STYLE
Home
Kategorien
Wunschliste

So how about…

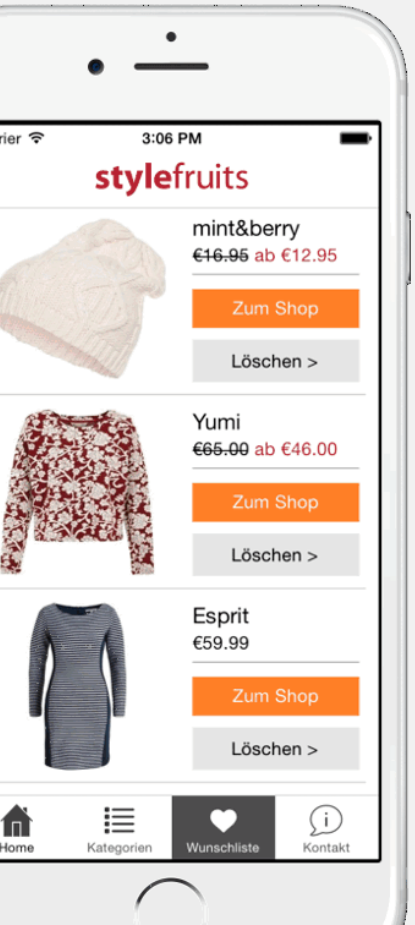

3:06 PM
stylefruits
mint&berry
€16.95 ab €12.95
Zum Shop
Löschen >
Yumi
€65.00 ab €46.00
Zum Shop
Löschen >
Esprit
€59.99
Zum Shop
Löschen >
Home
Kategorien
Wunschliste
Kontakt

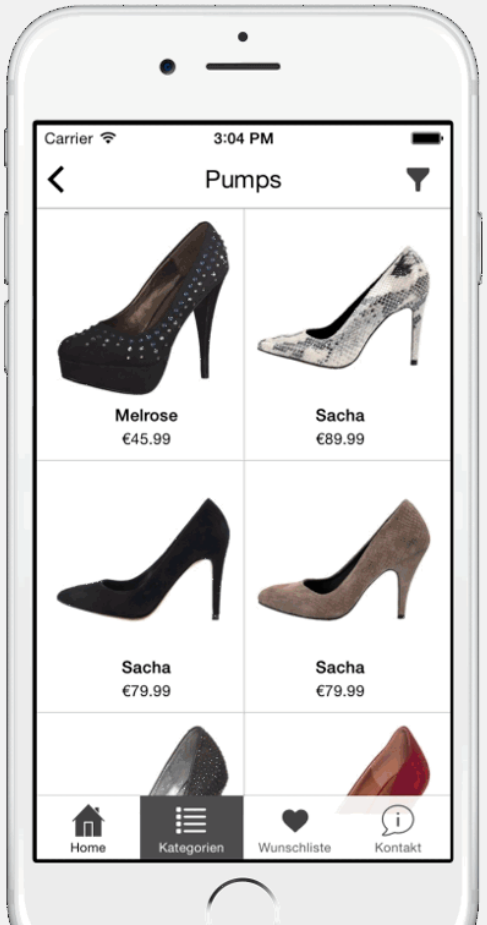

Carrier
3:04 PM
Pumps
Melrose
€45.99
Sacha
€89.99
Sacha
€79.99
Sacha
€79.99
Home
Kategorien
Wunschliste
Kontakt

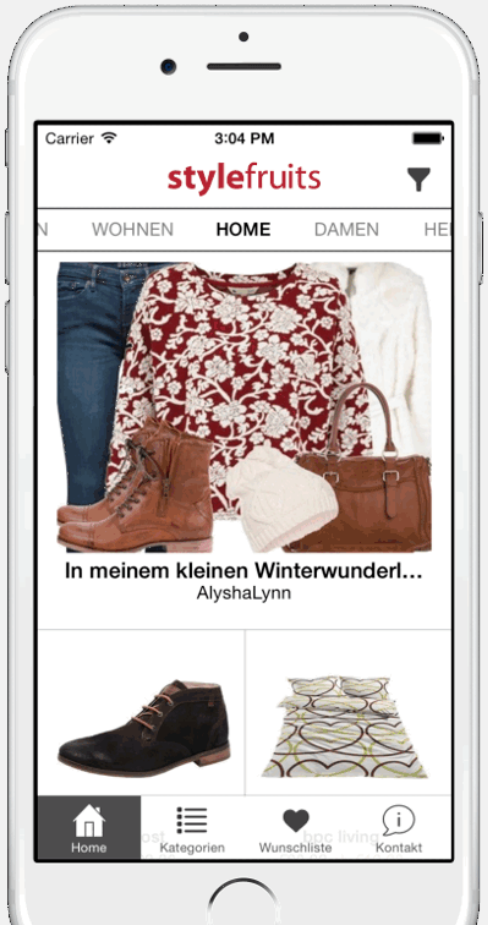

Carrier
3:04 PM
stylefruits
WOHNEN
HOME
DAMEN
In meinem kleinen Winterwunderl...
AlyshaLynn
Home
Kategorien
Wunschliste
Kontakt

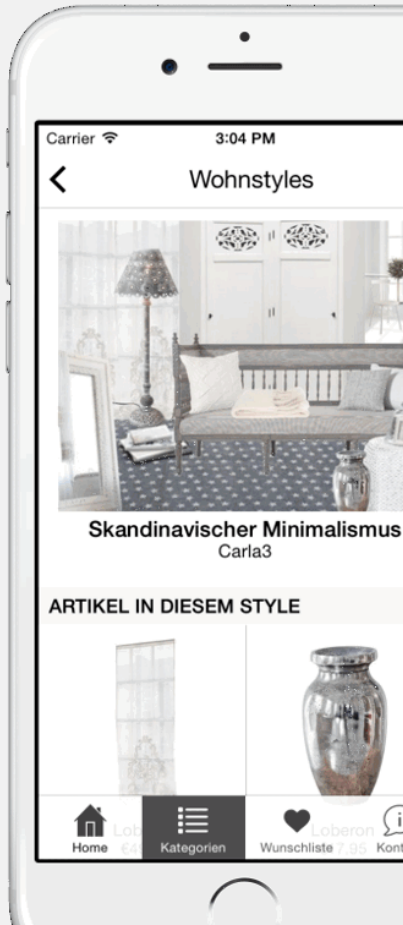

Carrier
3:04 PM
Wohnstyles
Skandinavischer Minimalismus
Carla3
ARTIKEL IN DIESEM STYLE
Home
Kategorien
Wunschliste

So how about… Clojure?









Expectations set high

- short feedback loop
- continuous delivery
- automated and pauseless deployment
- …under load
- high availability, dynamic scaling

we've called it

Ogrom*

---

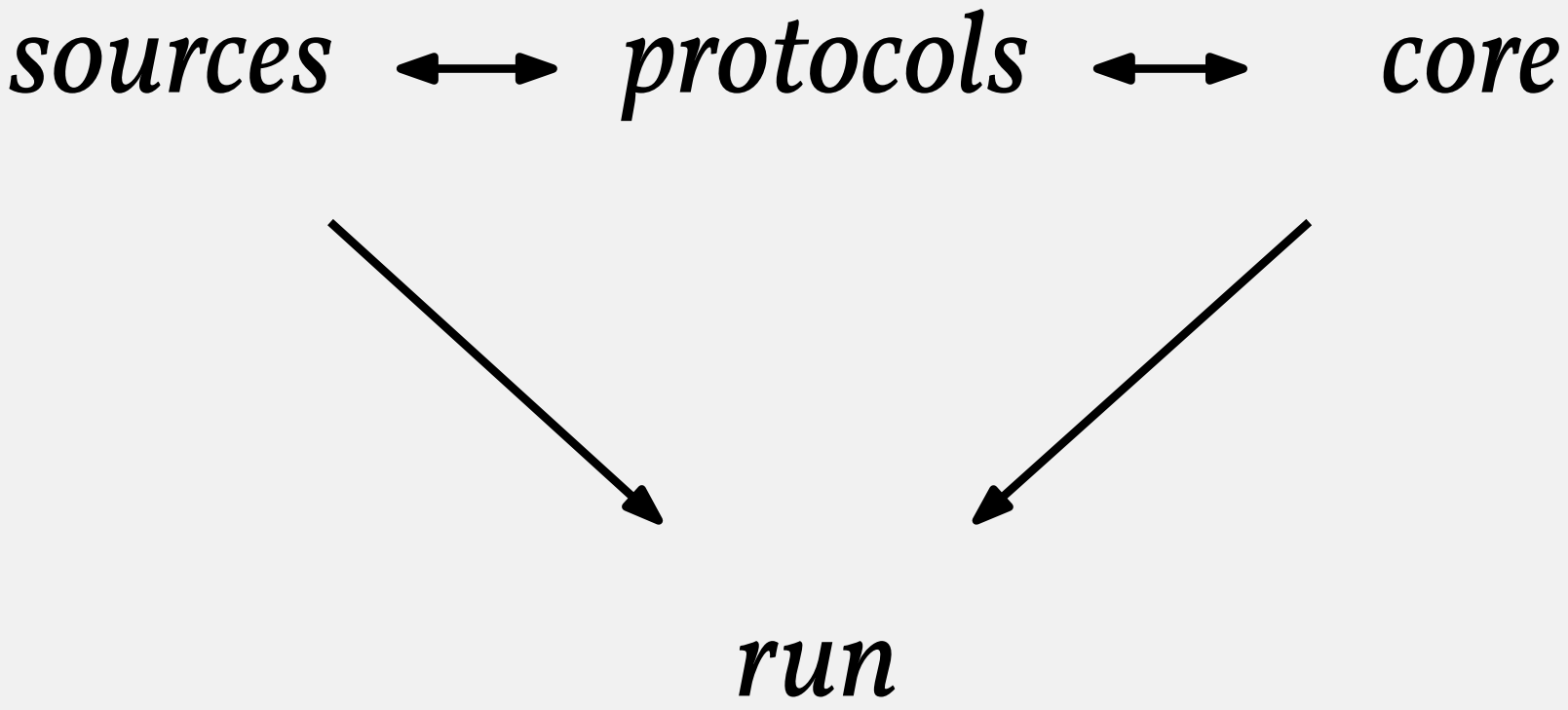
* *plethora, immensity, enormity*

Outline

1. Architecture
2. Deployment and Operations
3. Lessons Learned

Architecture

to use a bold word

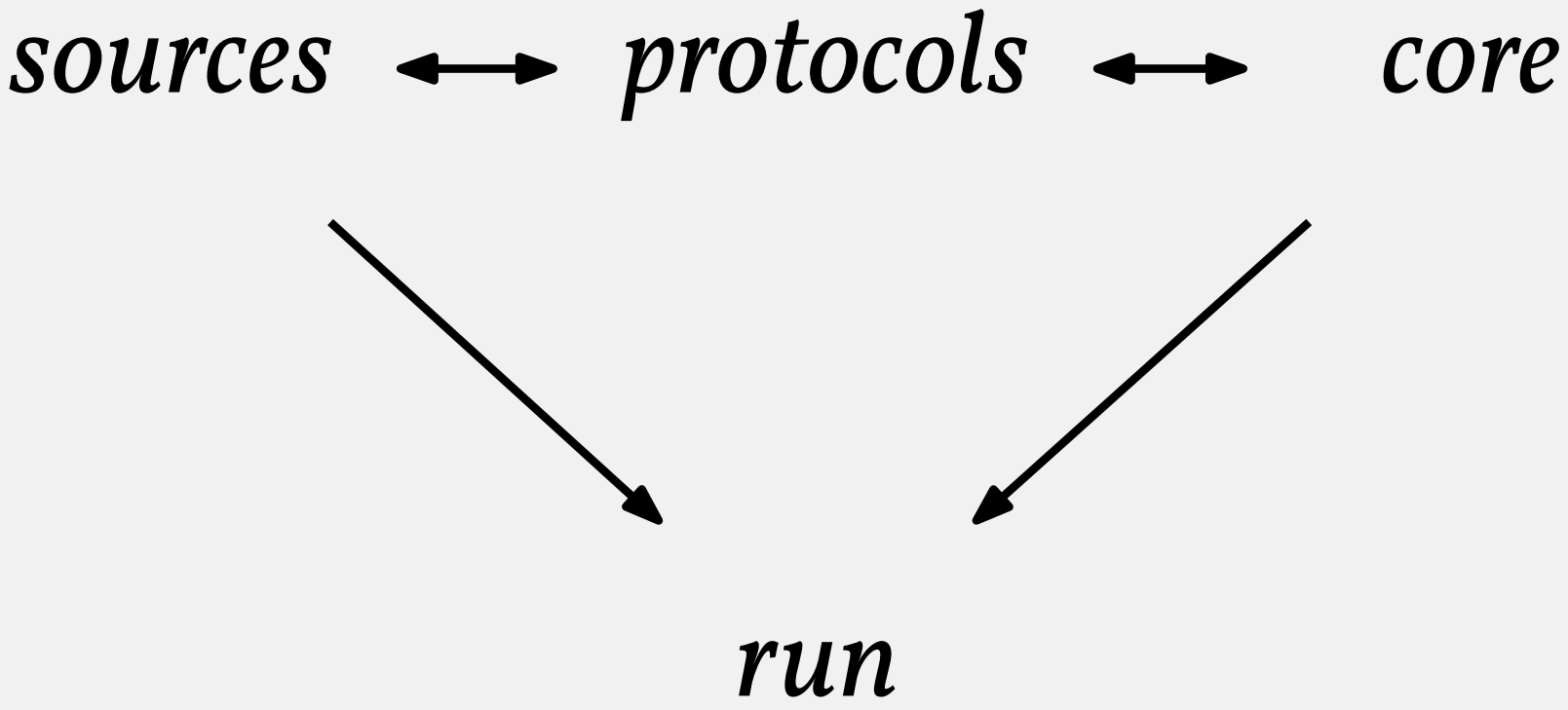

sources
protocols
core
run

protocols define interfaces

```
(defprotocol CollageSource
  (collage-by-id [ctx collage-id]))
```

sources wrap data sources

```
(reify CollageSource
  (collage-by-id [ctx collage-id]
    (fetch-from-mysql ctx collage-id)))
```


sources
protocols
core
run

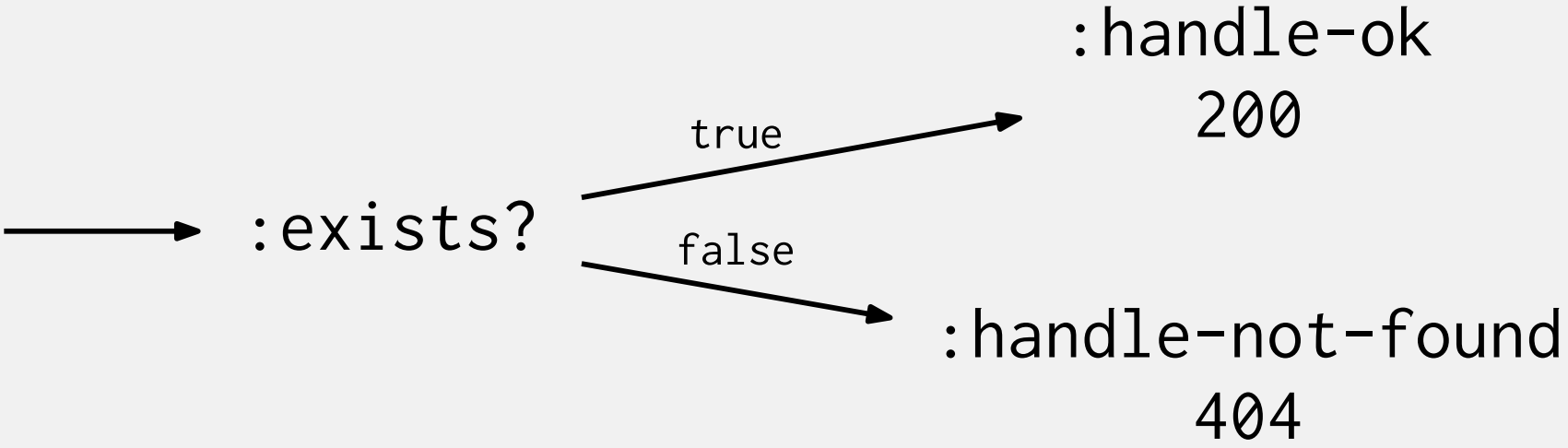
core defines HTTP endpoints

each *core* endpoint

- is independent
- has all dependencies injected
- is absolute-path-agnostic
- is a value

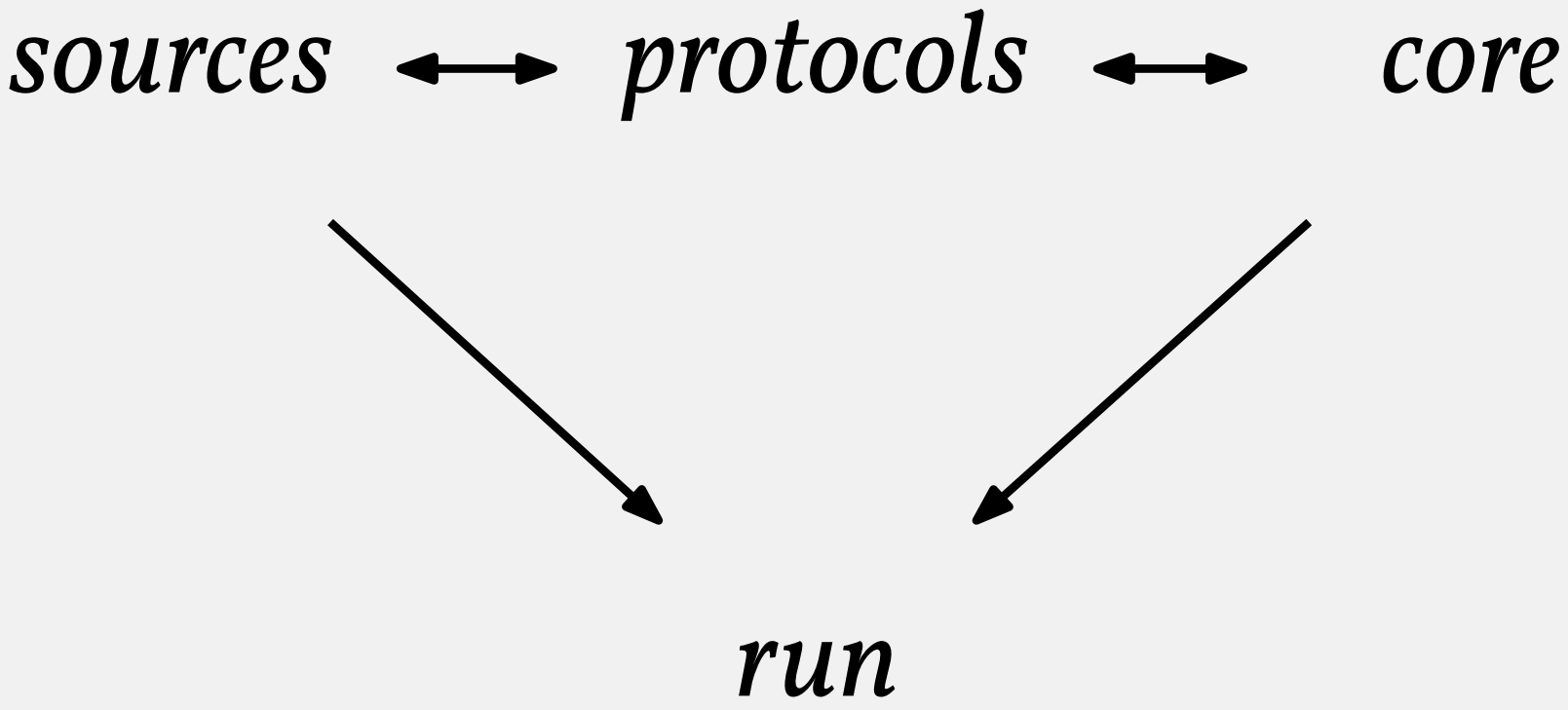
core uses Liberator

```
(defresource product
  :exists? find-product-or-return-nil
  :handle-ok render-product
  :handle-not-found render-error-message)
```

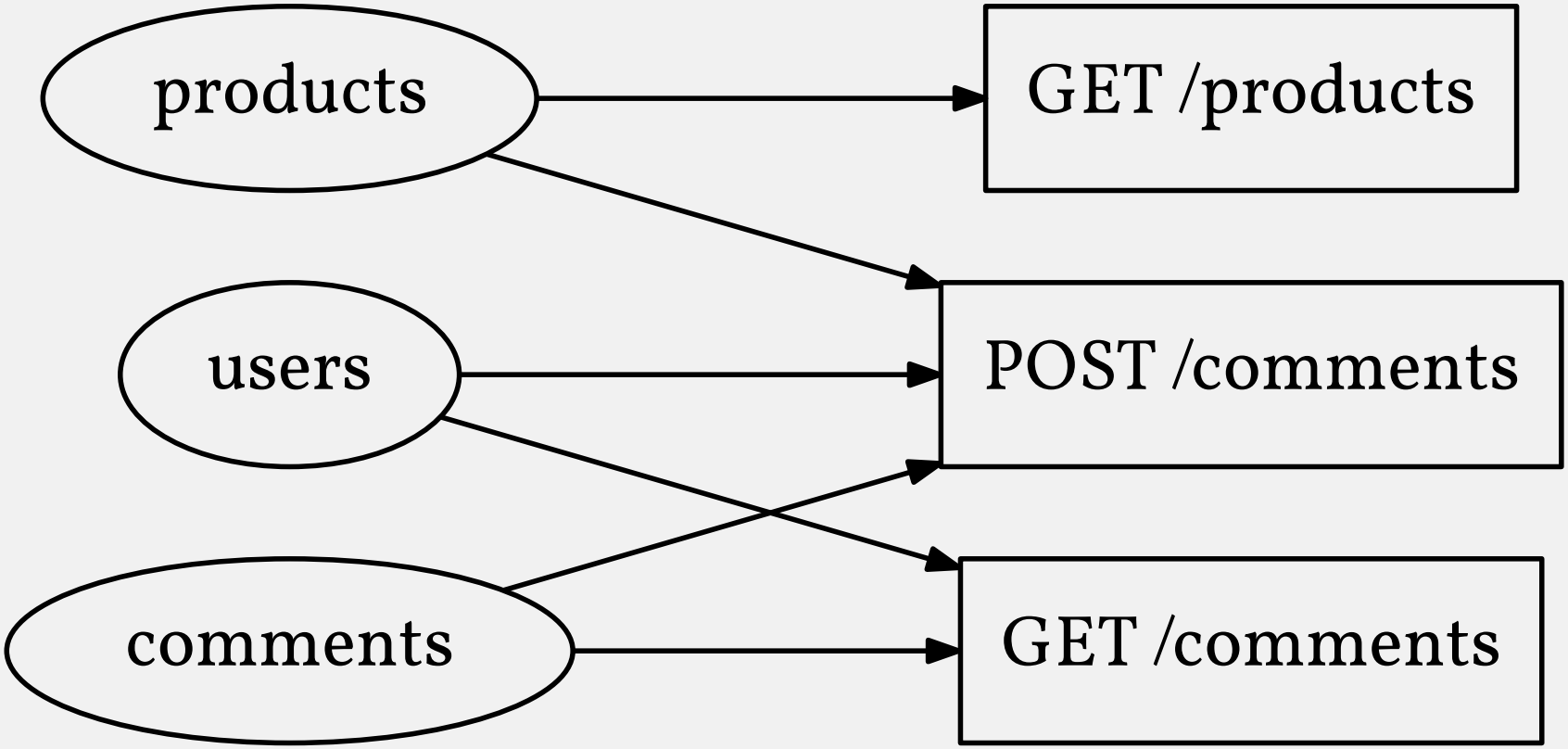

:exists?
true
false
:handle-ok
200
:handle-not-found
404

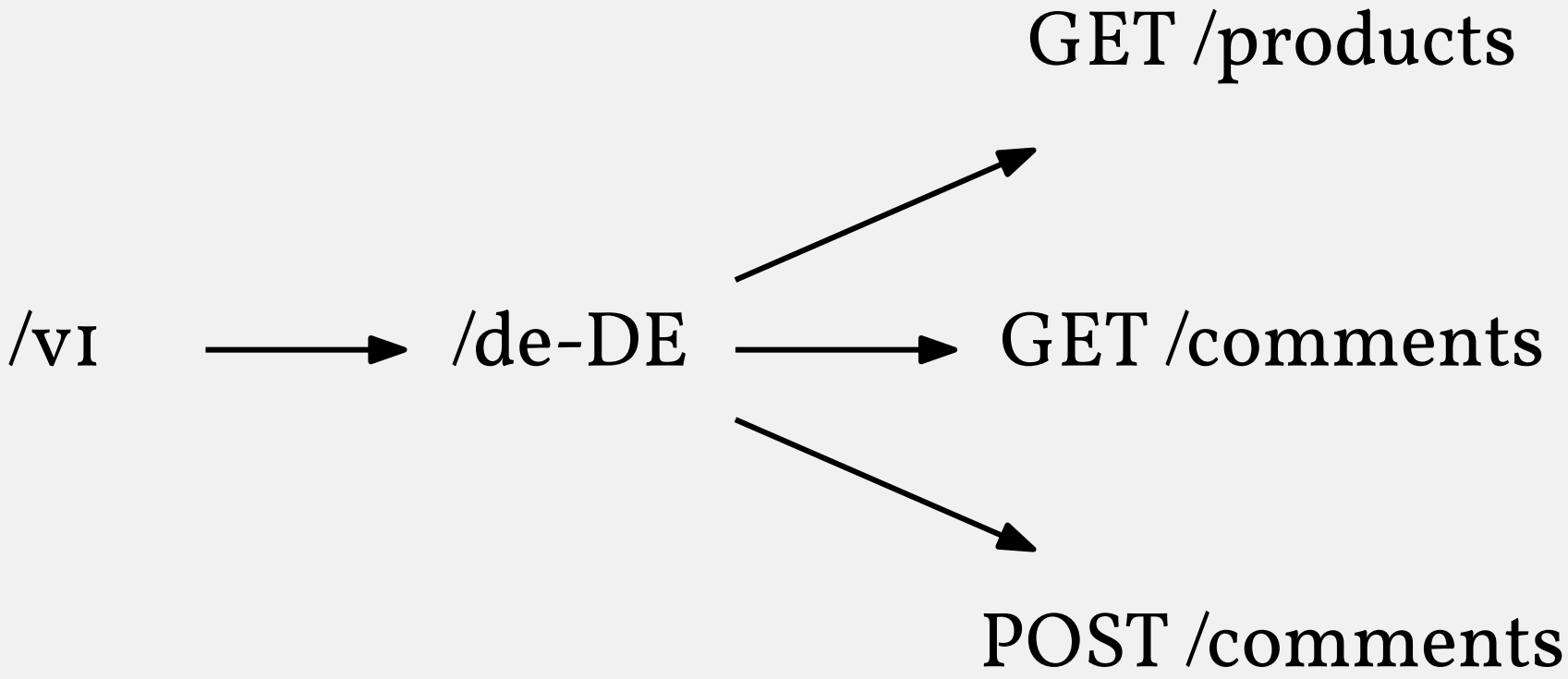
core uses bidi

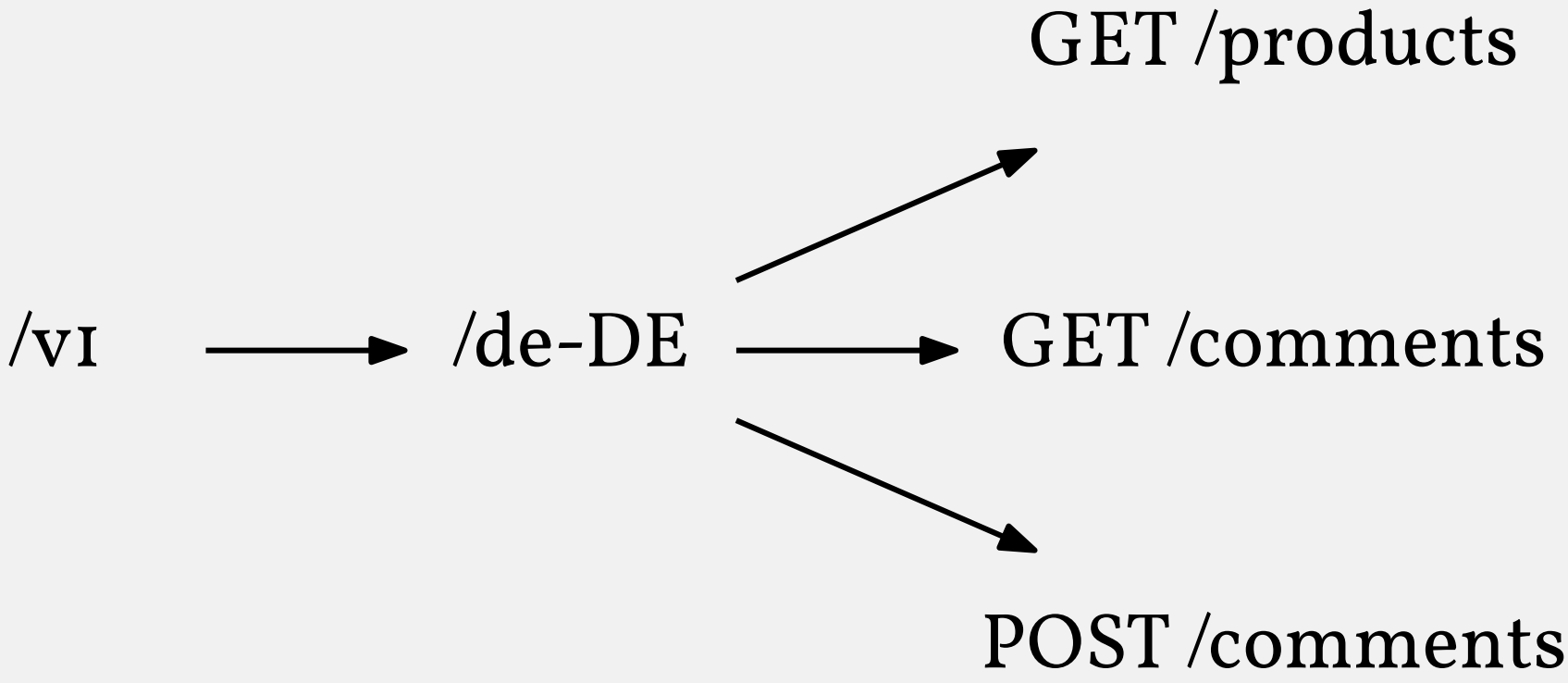
```
(def routes
  ["/"
    {"product" product
     "ratings" ratings}])
```


sources
protocols
core
run

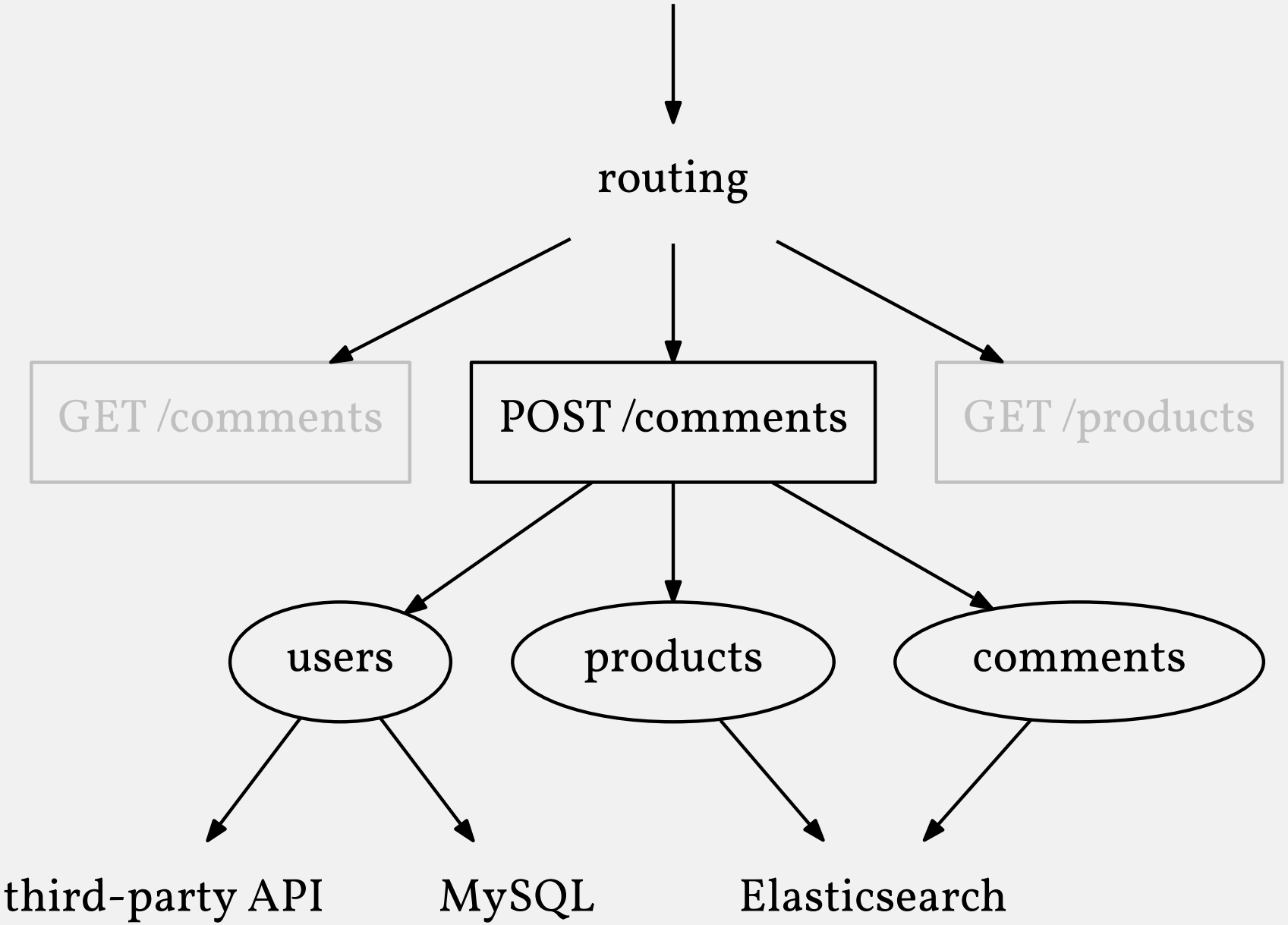
run wires everything together



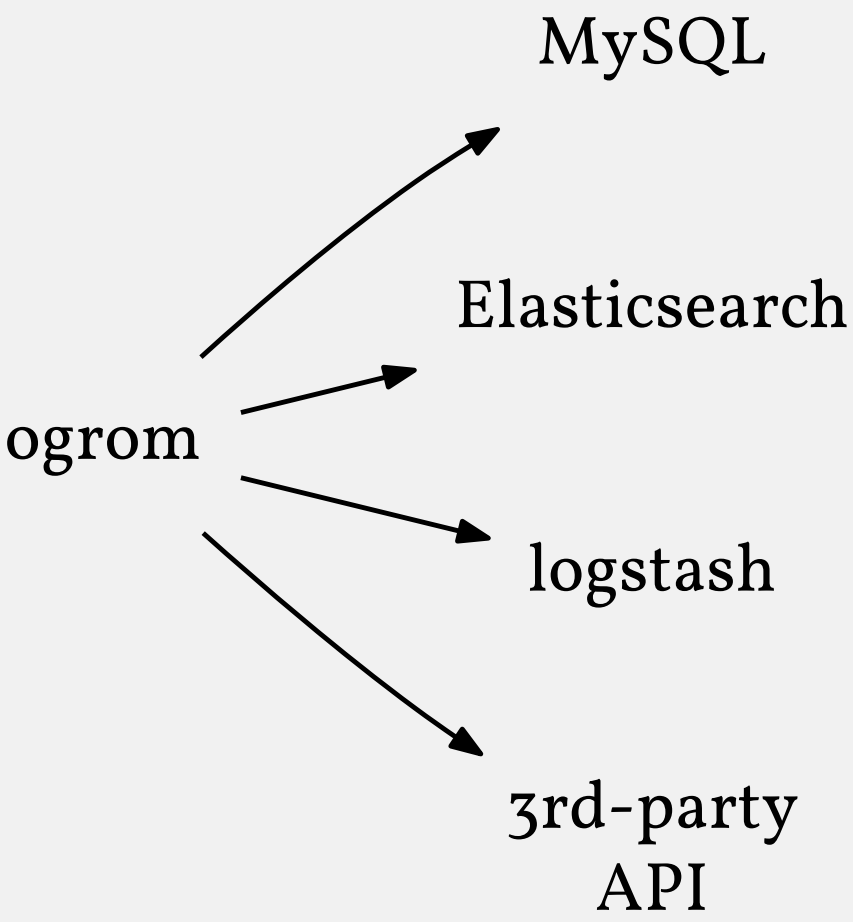

GET /products
/v1
/de-DE
GET /comments
POST /comments

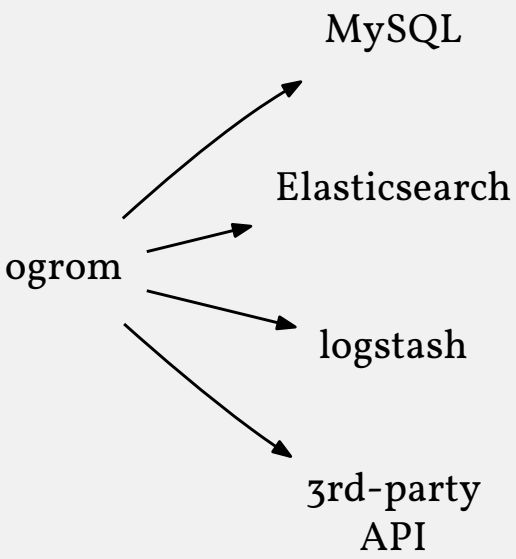


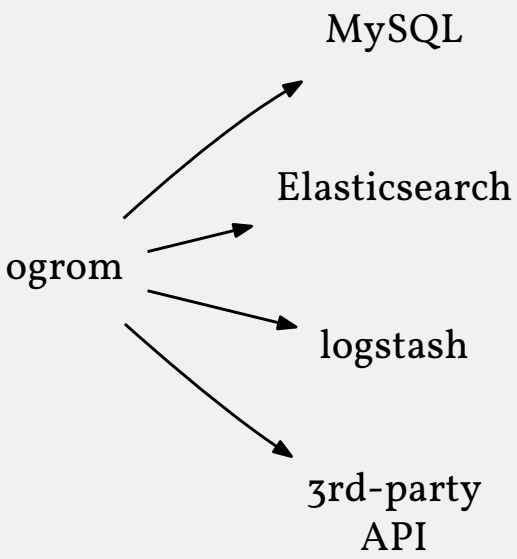
GET /v1/de-DE/comments

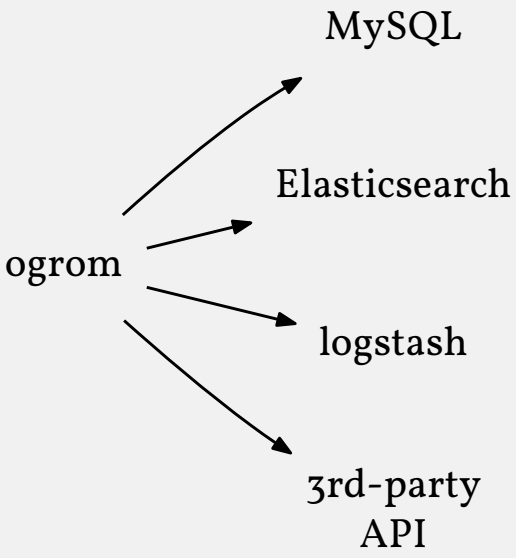

routing
GET /comments
POST /comments
GET /products
users
products
comments
third-party API
MySQL
Elasticsearch

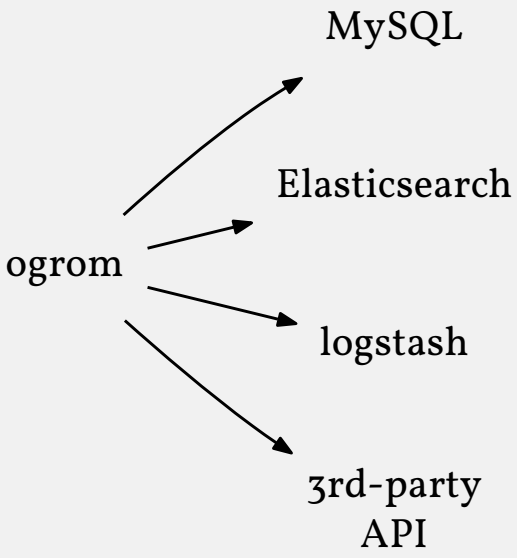
run instantiates a *self-contained* service

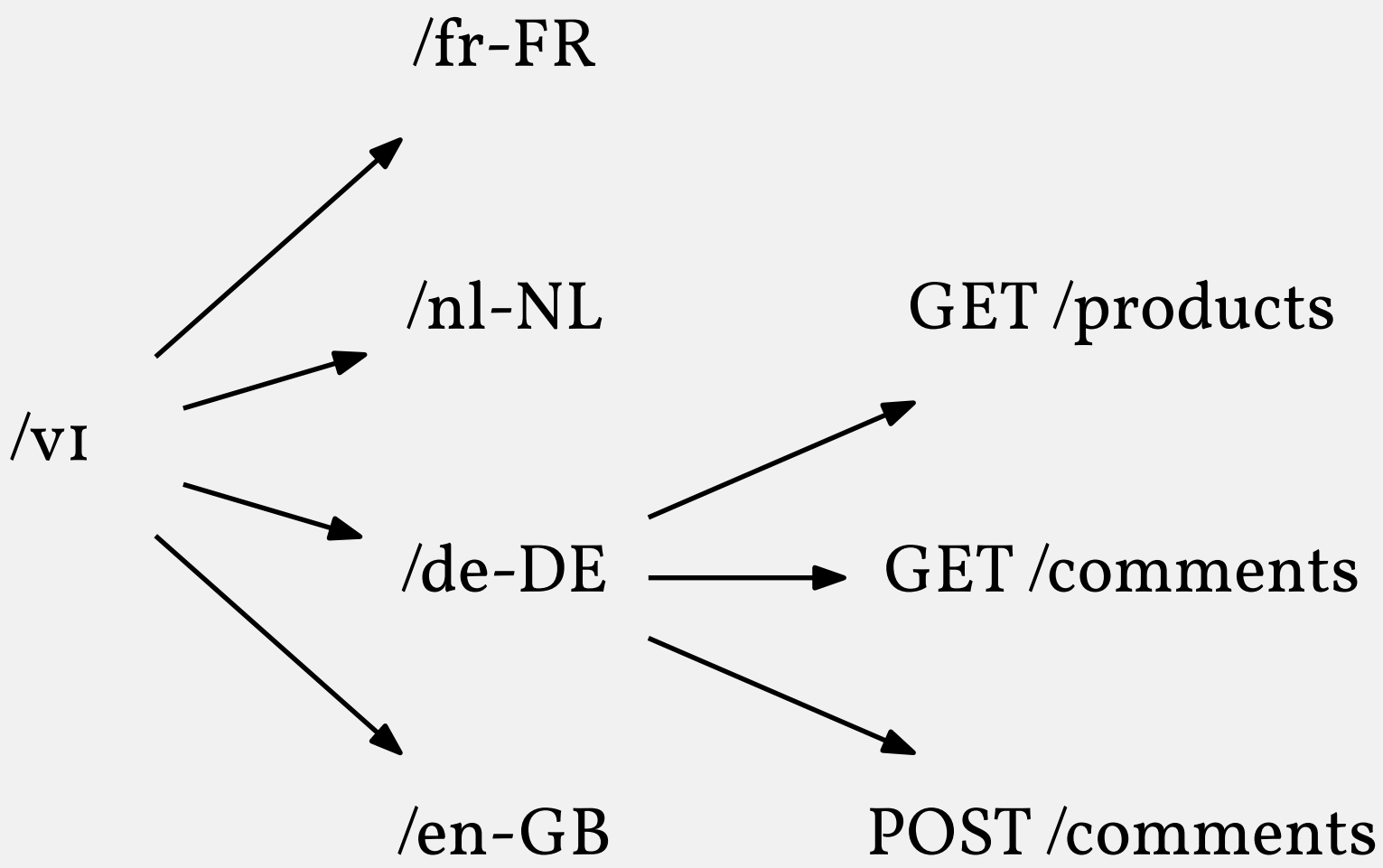


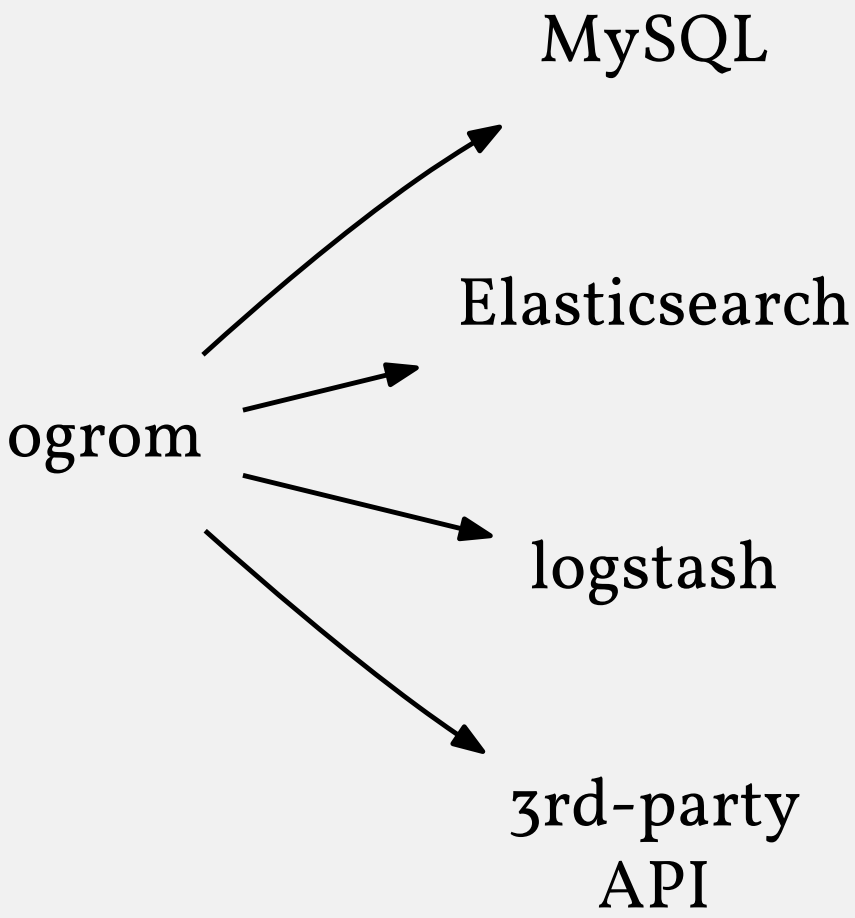

MySQL
Elasticsearch
ogrom
logstash
3rd-party
API


MySQL
Elasticsearch
ogrom
logstash
3rd-party
API


MySQL
Elasticsearch
ogrom
logstash
3rd-party
API


MySQL
Elasticsearch
ogrom
logstash
3rd-party
API

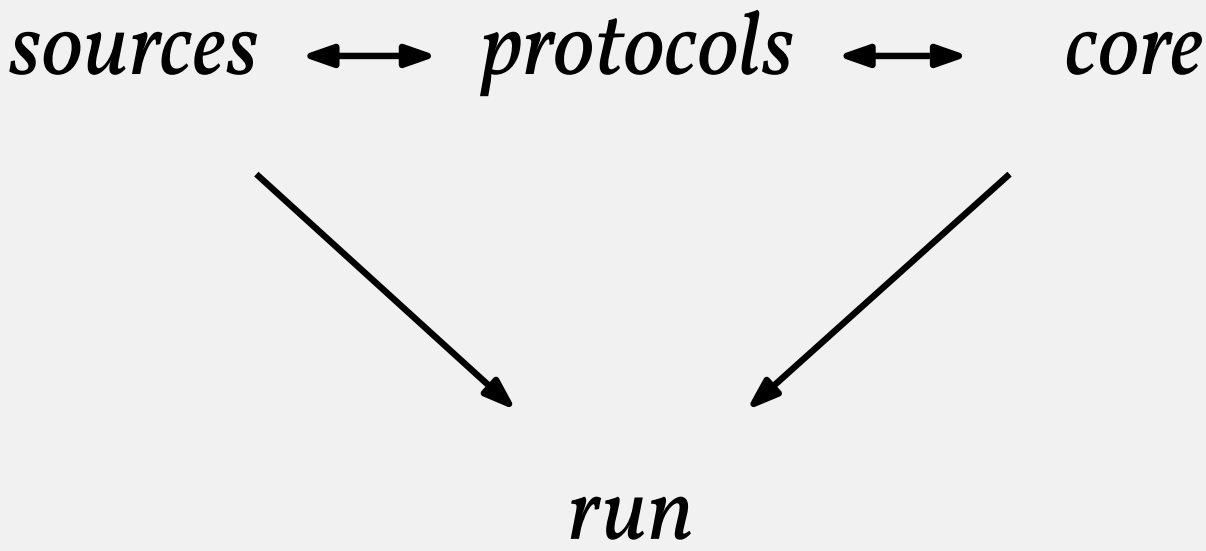

/fr-FR
/nl-NL
GET /products
/vi
/de-DE
GET /comments
/en-GB
POST /comments


MySQL
Elasticsearch
ogrom
logstash
3rd-party
API

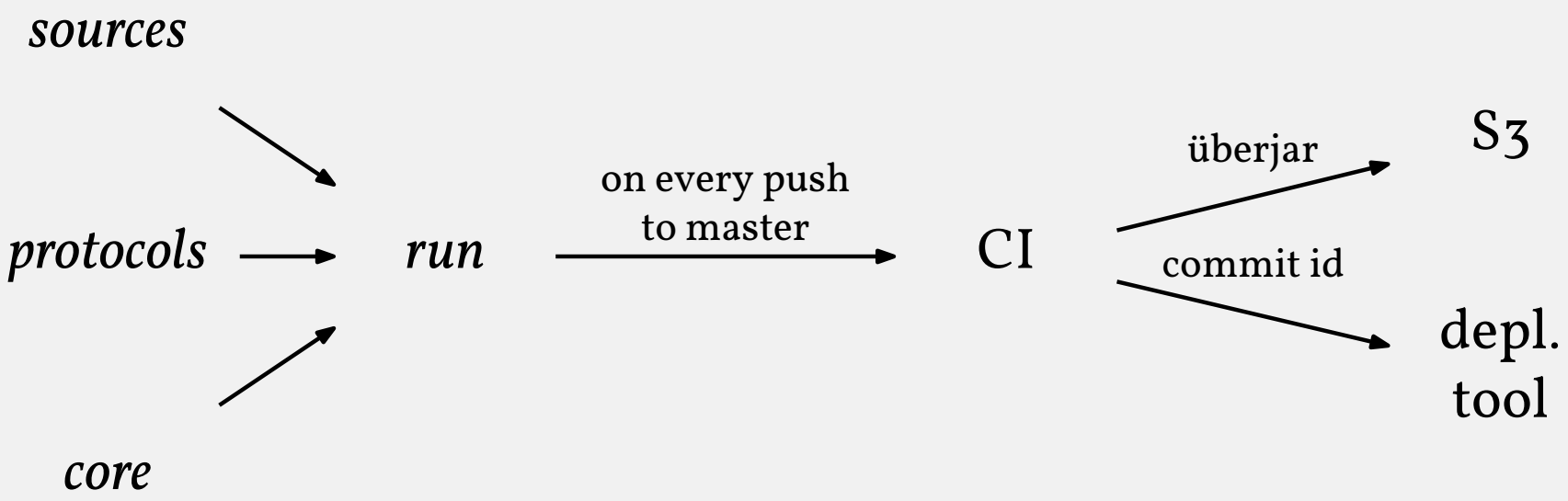
Deployment
and Operations

Deployment used to be cumbersome

1. Create an überjar
2. Upload to *n* boxes
3. Restart processes
4. Juggle load balancers

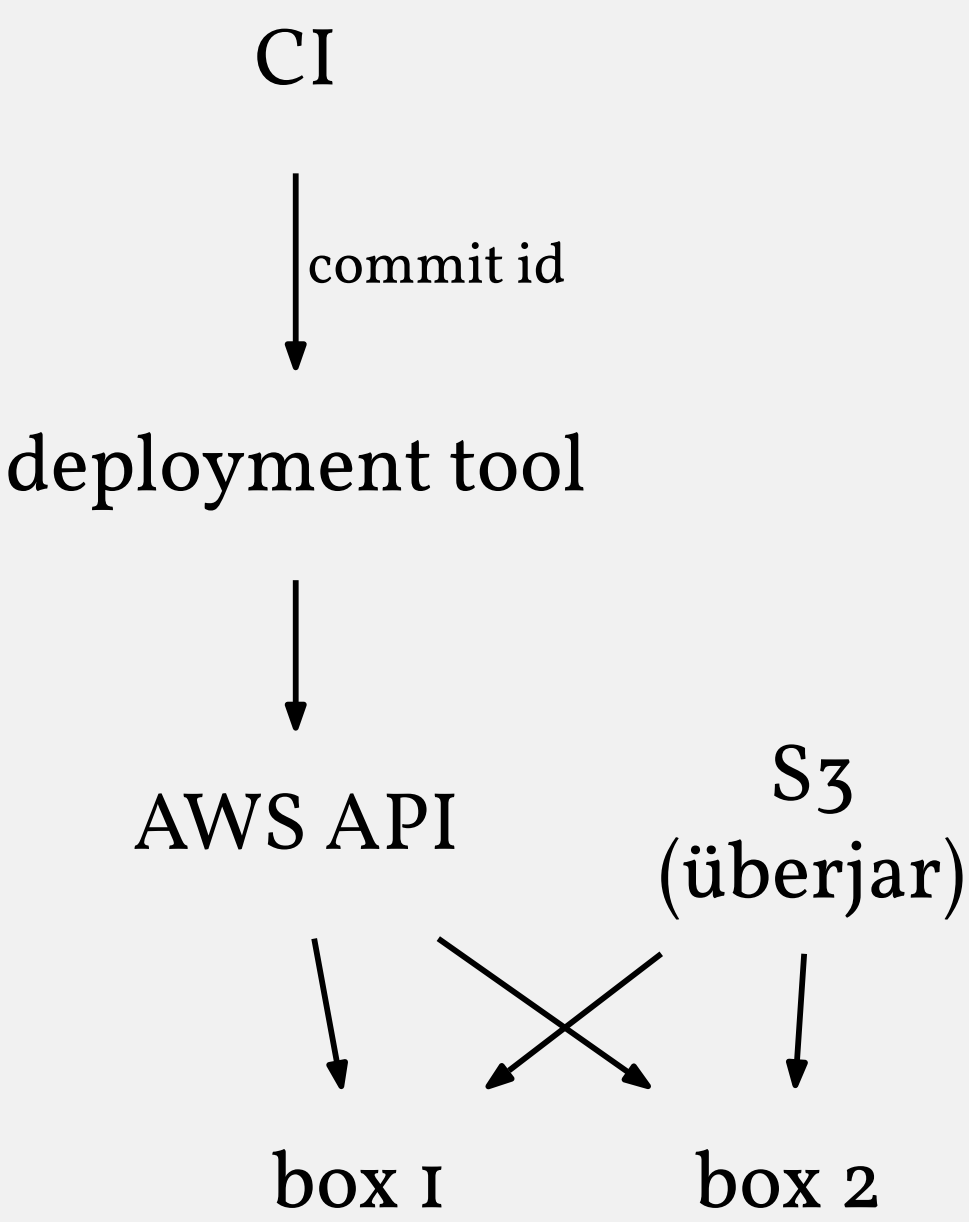


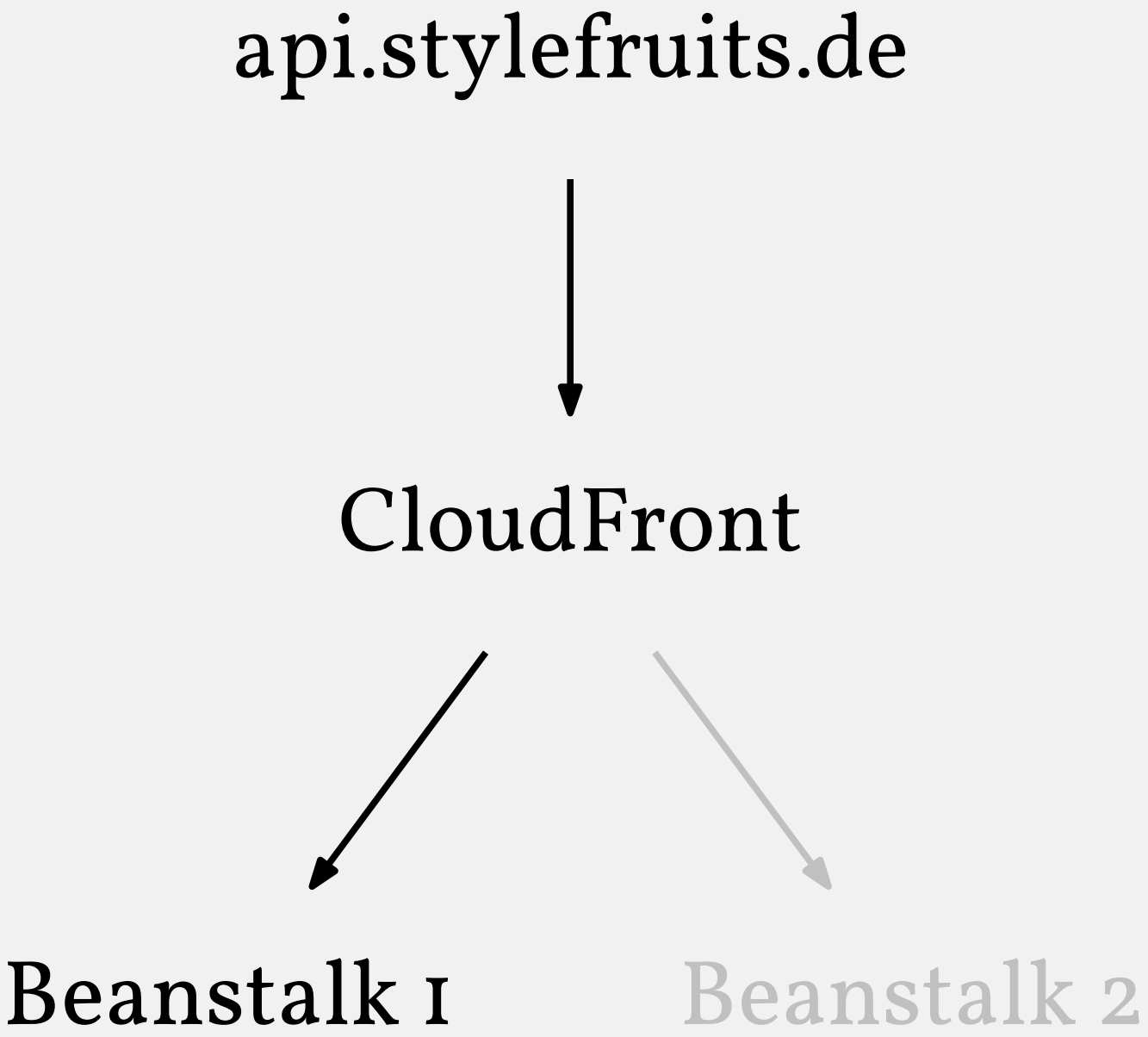
- *sources*, *protocols*, *core* are sem-versioned
- *run* is deployed straight from master


sources
protocols
core
run
on every push
to master
CI
überjar
commit id
S3
depl.
tool

Elastic Beanstalk and its environments

- EB wraps EC2 and ELB
- environments with instances
- runs Docker containers


CI
commit id
deployment tool
AWS API
S3
(überjar)
box 1
box 2

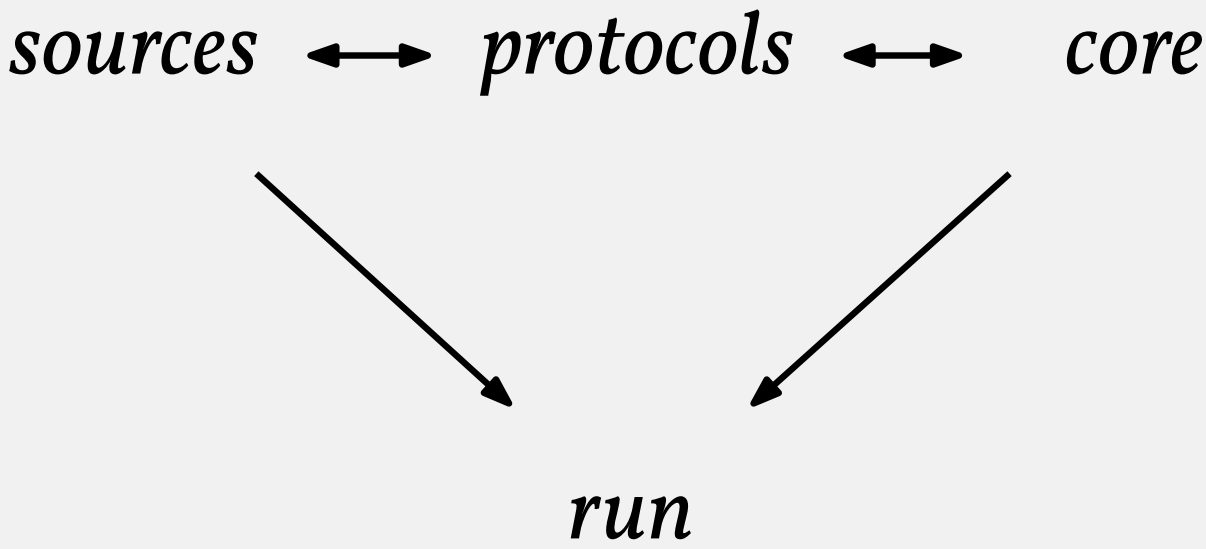

api.stylefruits.de
CloudFront
Beanstalk 1
Beanstalk 2

Lessons Learned
and Future Work

Feedback loops are shorter

- Productive development environment
- Bug fixes and features can be deploy the same morning they were asked for

Splitting wasn't strictly necessary



Docker and empowered developers

- Developers are in control over deployment
- Reduced dependency on the ops team
- Production env ≈ development env

There are some trade-offs

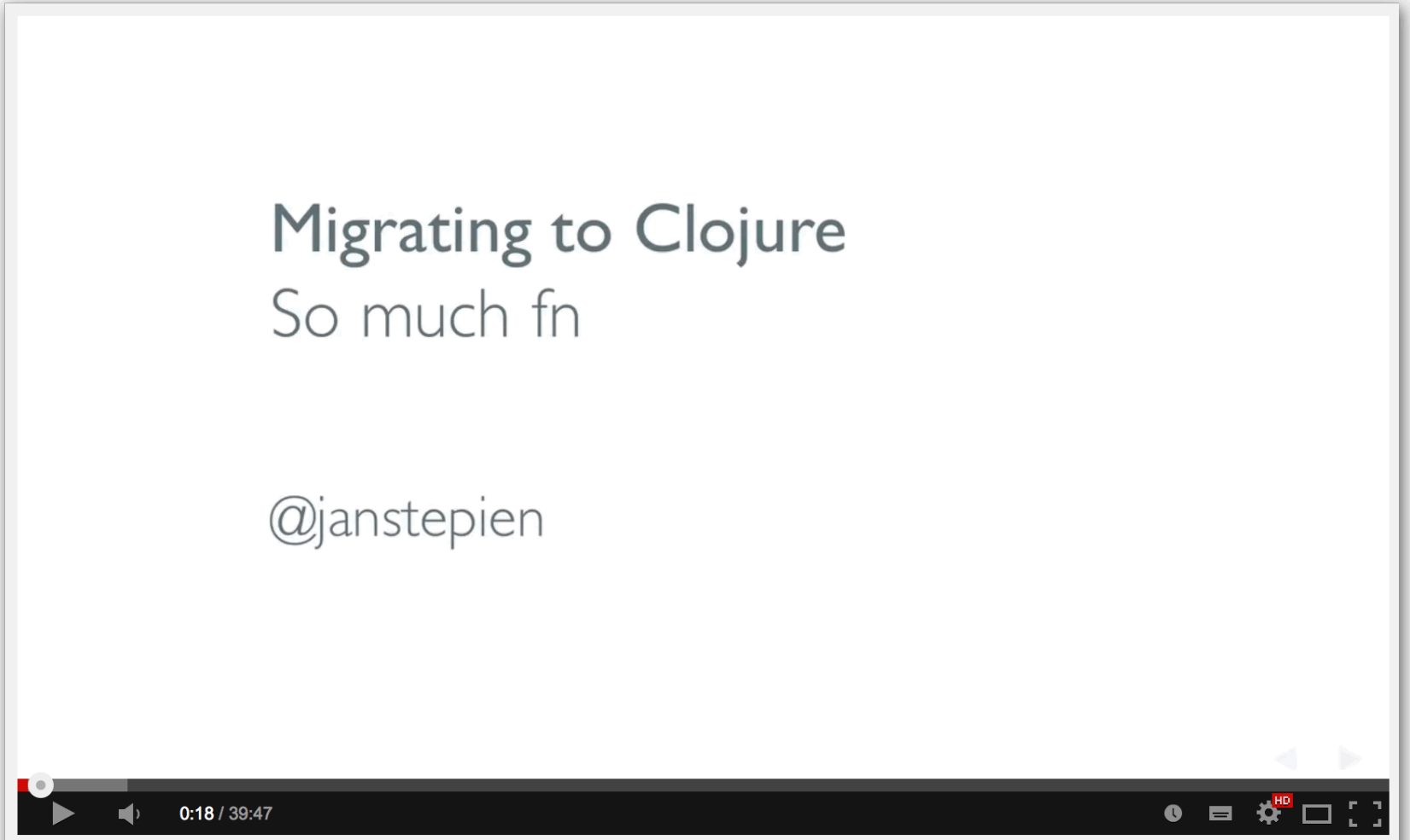
- Beanstalk allows opening only a single port when running Docker containers
- Deployment under load vs. JIT compilation
- The deployment takes a while: Clojure compile times, JVM start-up time, etc.
- There's no staging environment

Ogrom is an improvement

- Client teams are happy with it
- It performs well under load and is durable
- Our main web app uses the Ogrom too now

Plans for the future

- Ring Swagger
- Clojure 1.7 and its reduced compilation time
- Clojure(Script) client API



Jan Stępień - MIGRATING TO CLOJURE. SO MUCH FN

wrocloverb

Subscribe 628

421 views

Add to Share More

7 0

Clojure Redeployed

Jan Stępień @janstepien jan@stepien.cc