

RIDGE

Interactive Development in the Age of Kubernetes

Misha Gusarov
Software Architect

Aug 2020

Ridge Cloud

Ridge is a massively distributed cloud.

Powered by a global data centers & CSP network, Ridge provides developers a public cloud experience at the edge.

Select Edge Data Center Partners



Porto Alegre	Hong Kong	Jaipur
Hamilton	Amsterdam	Ludhiana
Porirua	Singapore	Mumbai
Wellington	London	Mumbai
Sydney	San Jose	Sydney
Toronto	Dallas	Frankfurt
Toronto	New Jersey	Tokyo
Vancouver	Chicago	Amsterdam
Frankfurt	New York	
Frankfurt	Lagos	
Singapore	Abuja	
Slough	Karlskrona	
Reading	Haifa	
Seattle	Zagreb	
Sterling	Sydney	
Sterling	Melbourne	
Santa Clara	Toronto	
Secaucus	Hong Kong	
Chicago	Frankfurt	
Frankfurt	Amsterdam	
Stockholm	Hamilton	
London	Serangoon	Ashburn
Buffalo	Johannesburg	Dallas
Petah Tikva	London	Parsippany
Bogor Jawa	Ashburn	Chicago
Barat	Santa Clara	Los Angeles
Mumbai	Bangalore	Sunnyvale
Nashik	Bangalore	Toronto
Brasov	Ahmedabad	Johannesburg
Bucharest	Faridabad	Cape Town
Montreal	Ghaziabad	Hong Kong

81+

EDGE LOCATIONS
ONE CLOUD
ONE API



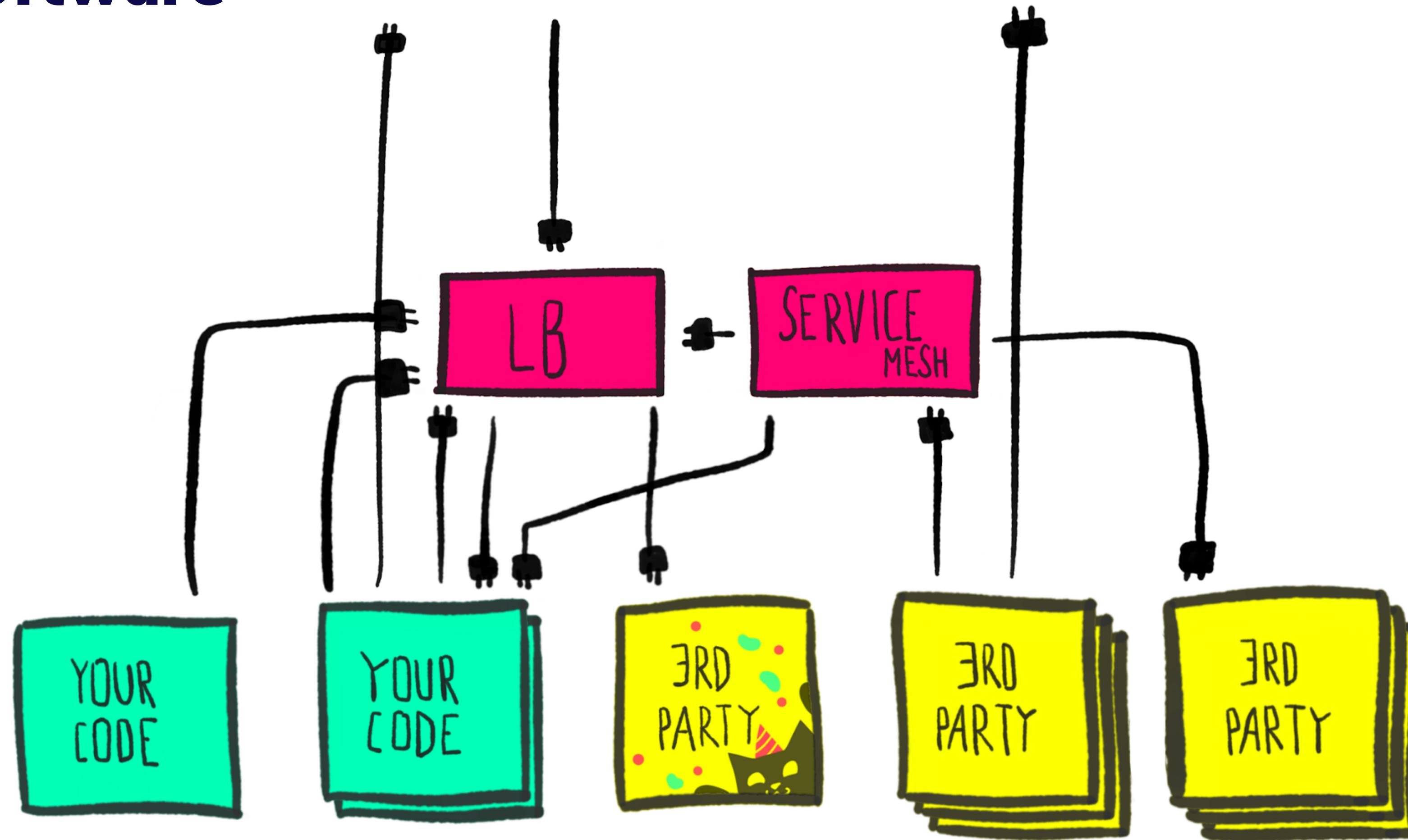
Interactive development, 80s style



```
ПК8000 BASIC 1987 версия 1.2
46873 Bytes free
Ok
10 PRINT "Hello wolrd!"
20 END
RUN
Hello wolrd!
Ok
10 PRINT "Hello world!"
RUN
Hello world!
Ok
■
```

color auto goto list run

Modern software



Ridge experience: development tools & problems

- Minikube et al
- Development clusters in cloud
- In-container debugging
- Slow edit-run-test loop
- Hard to start afresh
- Hard to debug
- Hard to manipulate state



Running distributed applications locally



Ideal

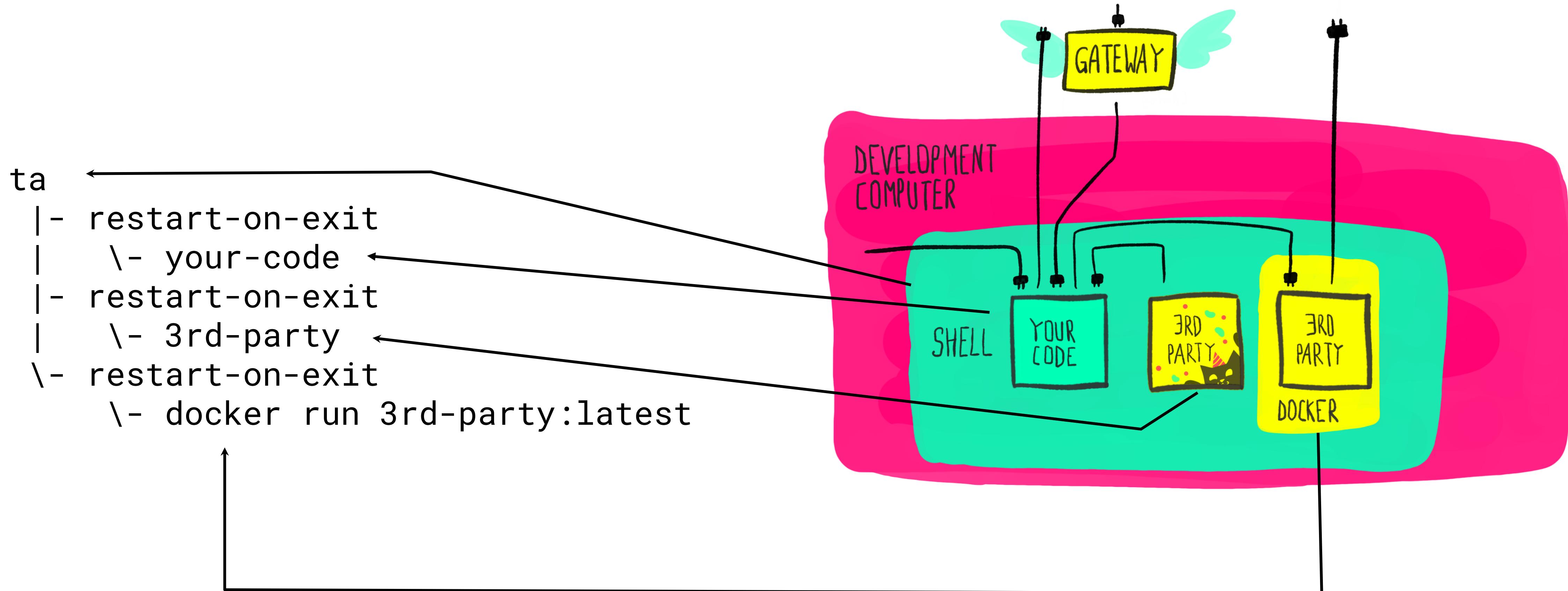
```
PK8000 BASIC 1987 версия 1.2
46873 Bytes free
Ok
10 PRINT "Hello wolrd!"
20 END
RUN
Hello wolrd!
Ok
10 PRINT "Hello world!"
RUN
Hello world!
Ok
■
```

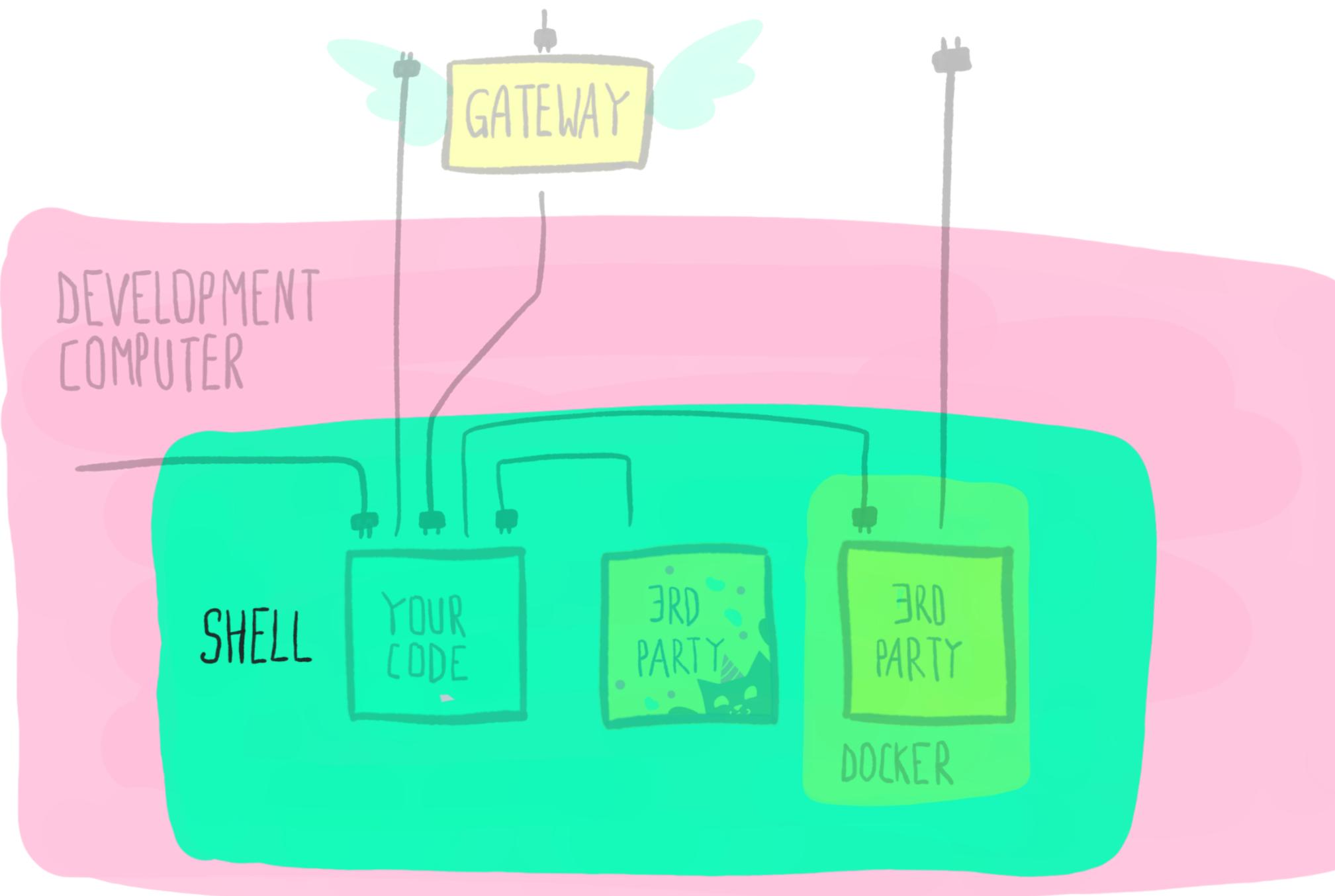
color auto goto list run

Realistically achievable

```
SunOS solaris 5.10 Generic_147147-26 sun4u sparc
SUNW,Sun-Fire-V210
    7:40pm  up 80 day(s),  2:15,  3 users,  load
average: 0.03, 0.02, 0.02
dottedma@solaris$ vi hello.c
"hello.c" 6 lines, 75 characters
dottedma@solaris$ cc hello.c
dottedma@solaris$ ./a.out
Hello wordl!
dottedma@solaris$ vi hello.c
"hello.c" 6 lines, 75 characters
dottedma@solaris$ cc hello.c
dottedma@solaris$ ./a.out
Hello world!
dottedma@solaris$
```

Running distributed applications locally (2)





```
$ ta start
$ ta stop
$ ta destroy
$ ta attach
$ ta detach
```

[zsh@bohr:~/interactive/ta]%. /ta

Welcome to test application for interactive development.

URL: http://127.0.0.1:8080/

Common keys:

- Ctrl-] 0..9 Select window 0..9
- Ctrl-] p,n Select previous, next window
- Ctrl-] w Select window interactively
- Ctrl-] [Enter scrollback/copy mode (ESC to exit)
- Ctrl-s to search in scrollback in this mode
- Ctrl-] d Detach from the terminal (keep running in background)

CLI command is 'ta', run 'ta -h' for help.

This is tmux(1), so `man tmux` is useful.

Dealing with windows:

- * Exited programs will be marked in status line.
- * Ctrl-C sends SIGINT. It will stop the program inside, you can run your own (e.g. under debugger).

:

<Test Application> 0:help 1:static 2:traefik 3:backend-with-db[exited] 4:d>

[zsh@bohr:~/interactive/ta]%. /ta

(0) - ta: 7 windows (attached)

(1) -> 0: help* (1 panes) "bohr.malta.dottedmag.net"

(2) -> 1: static (1 panes) "bohr.malta.dottedmag.net"

(3) -> 2: traefik (1 panes) "bohr.malta.dottedmag.net" ██████████

(4) -> 3: backend-with-db[exited] (1 panes) "bohr.malta.dottedmag.net"

(5) -> 4: db[exited] (1 panes) "bohr.malta.dottedmag.net"

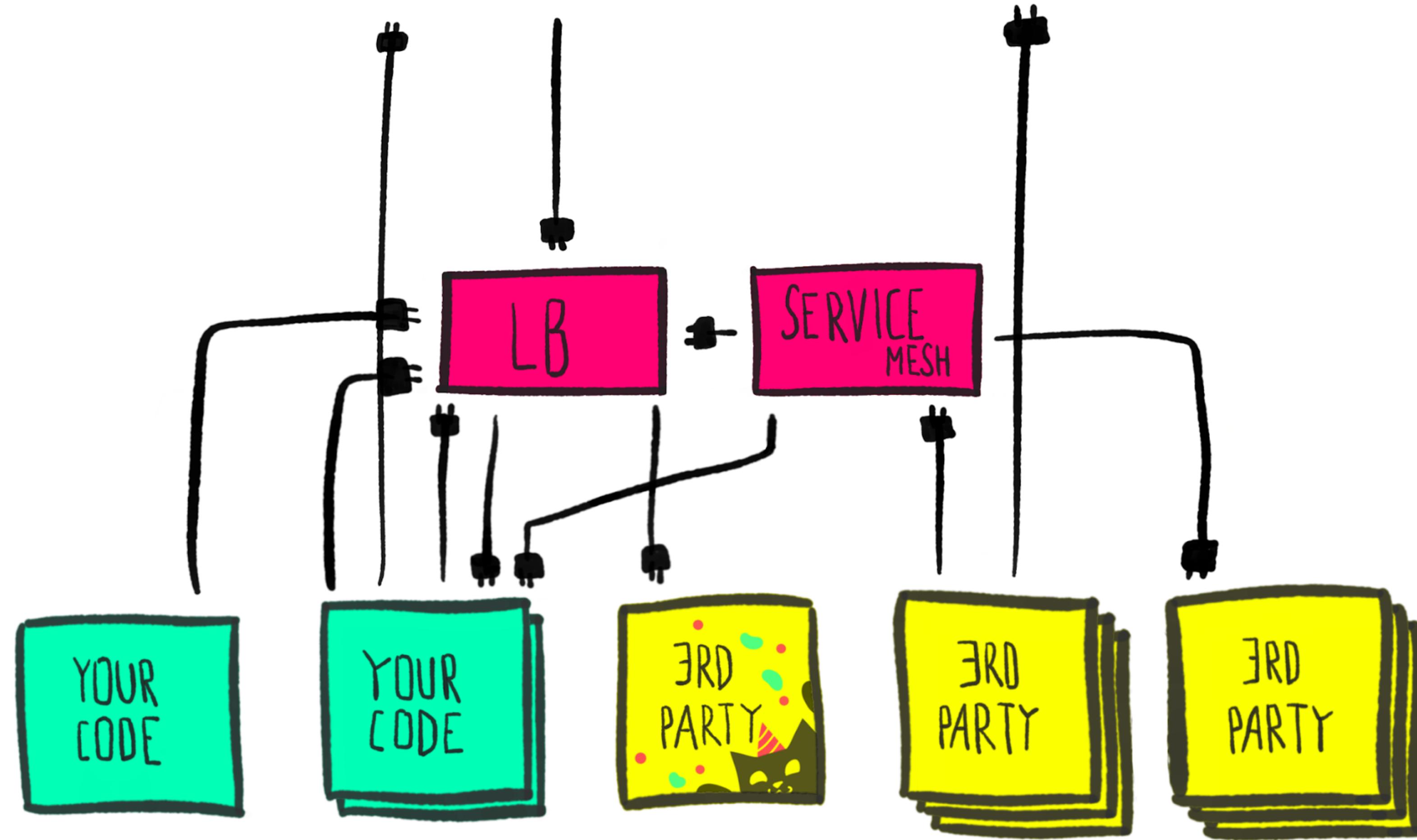
(6) -> 5: backend-without-db[exited] (1 panes) "bohr.malta.dottedmag.net"

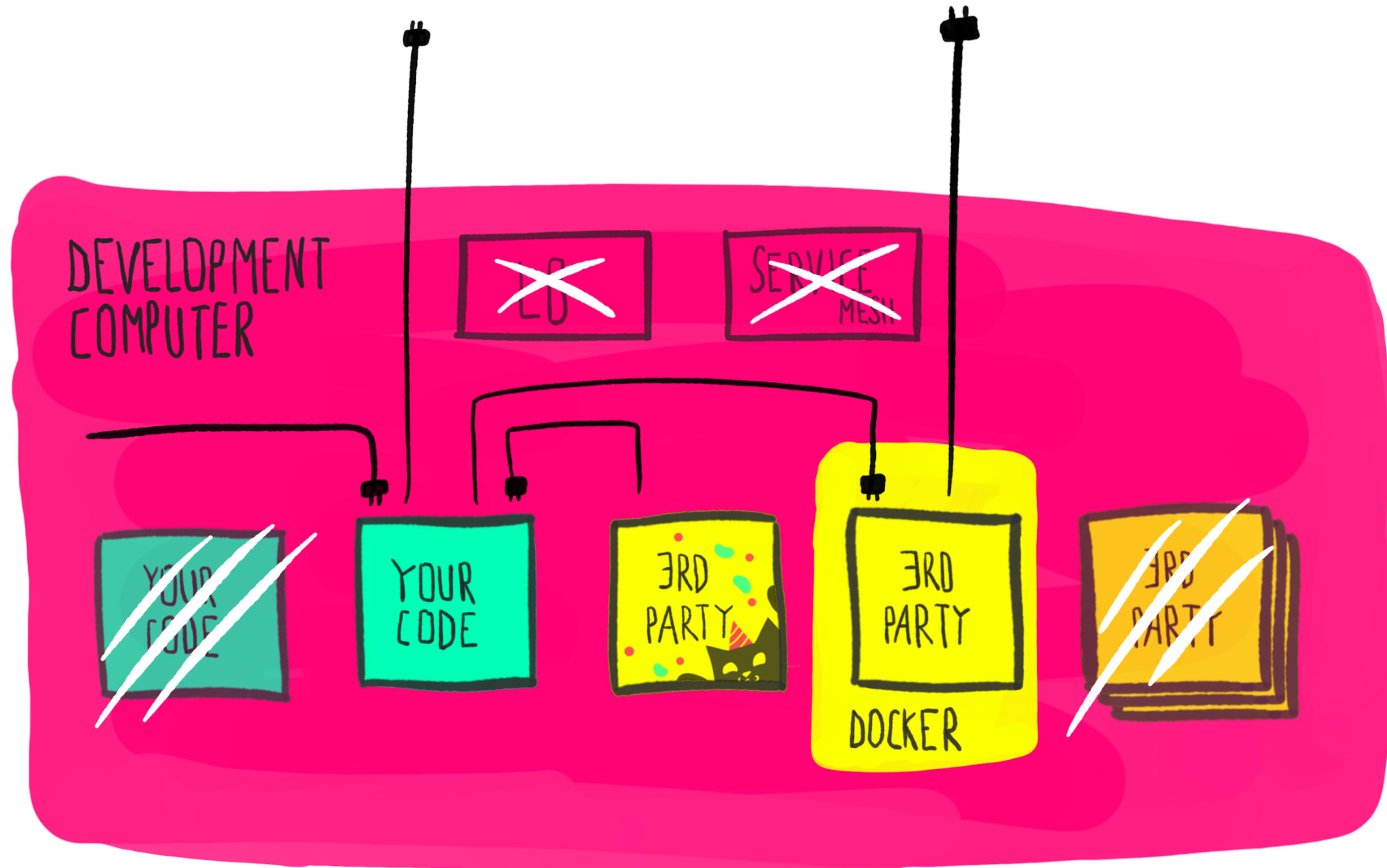
(7) -> 6: init[done] (1 panes) "bohr.malta.dottedmag.net"

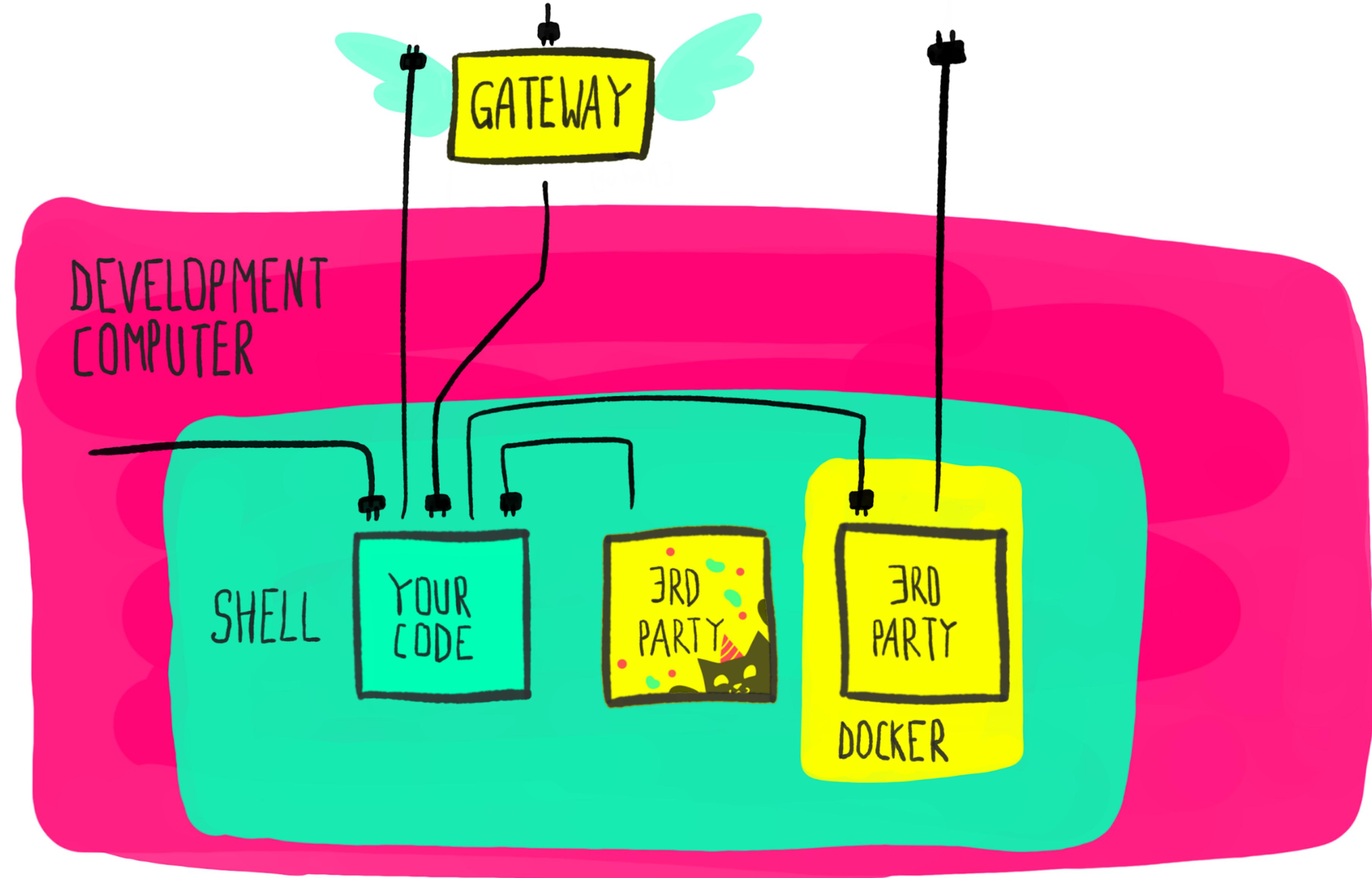
2 (sort: index)
64 --configfile=../src/traefik/traefik.toml --log=true --log.level=debug
time="2020-08-19T20:02:37+02:00" level=info msg="Configuration loaded from f
'/Users/dottedmag/interactive/src/traefik/traefik.toml"
■

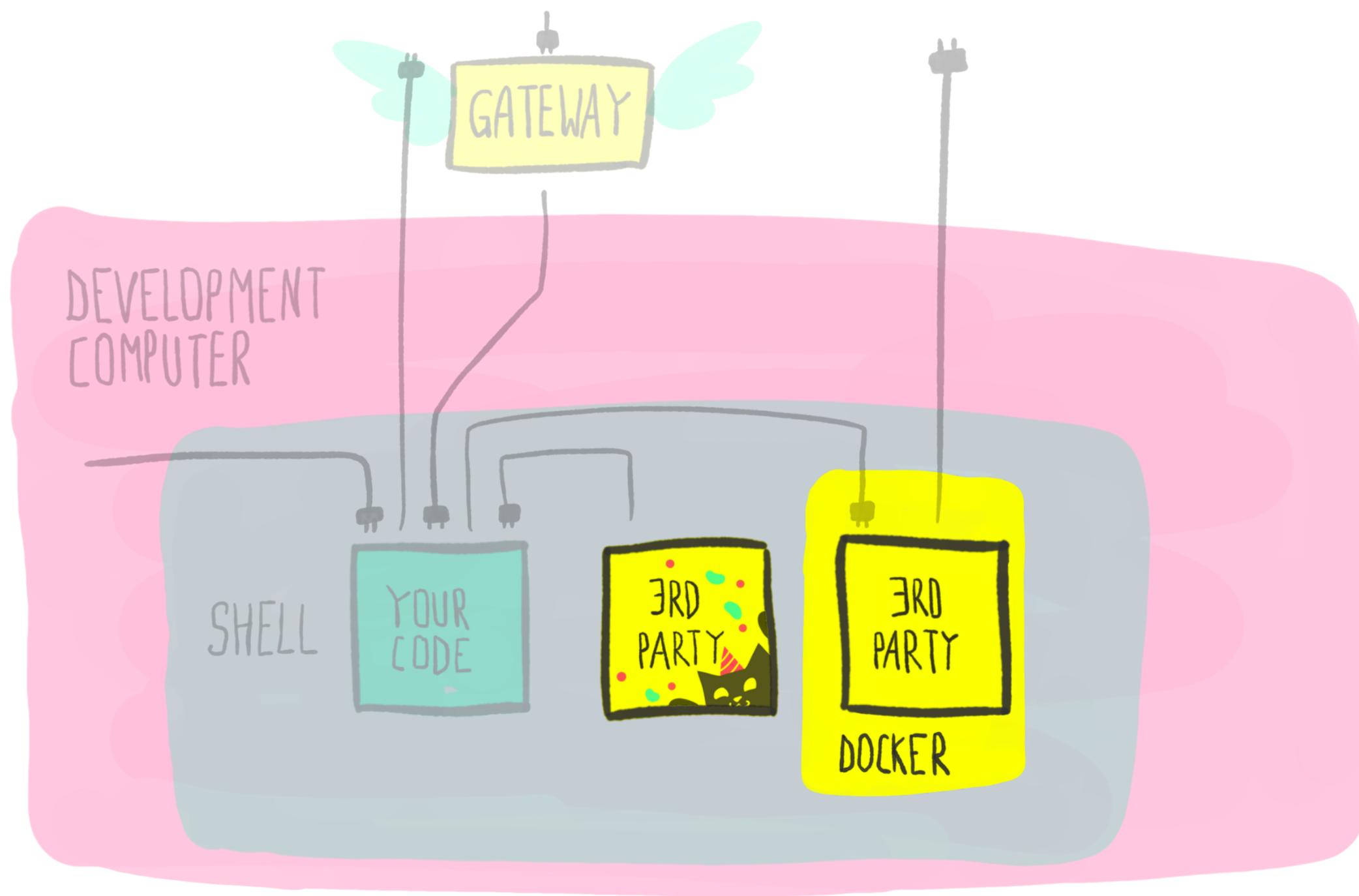
0

<Test Application> 0:help 1:static 2:traefik 3:backend-with-db[exited] 4:d>

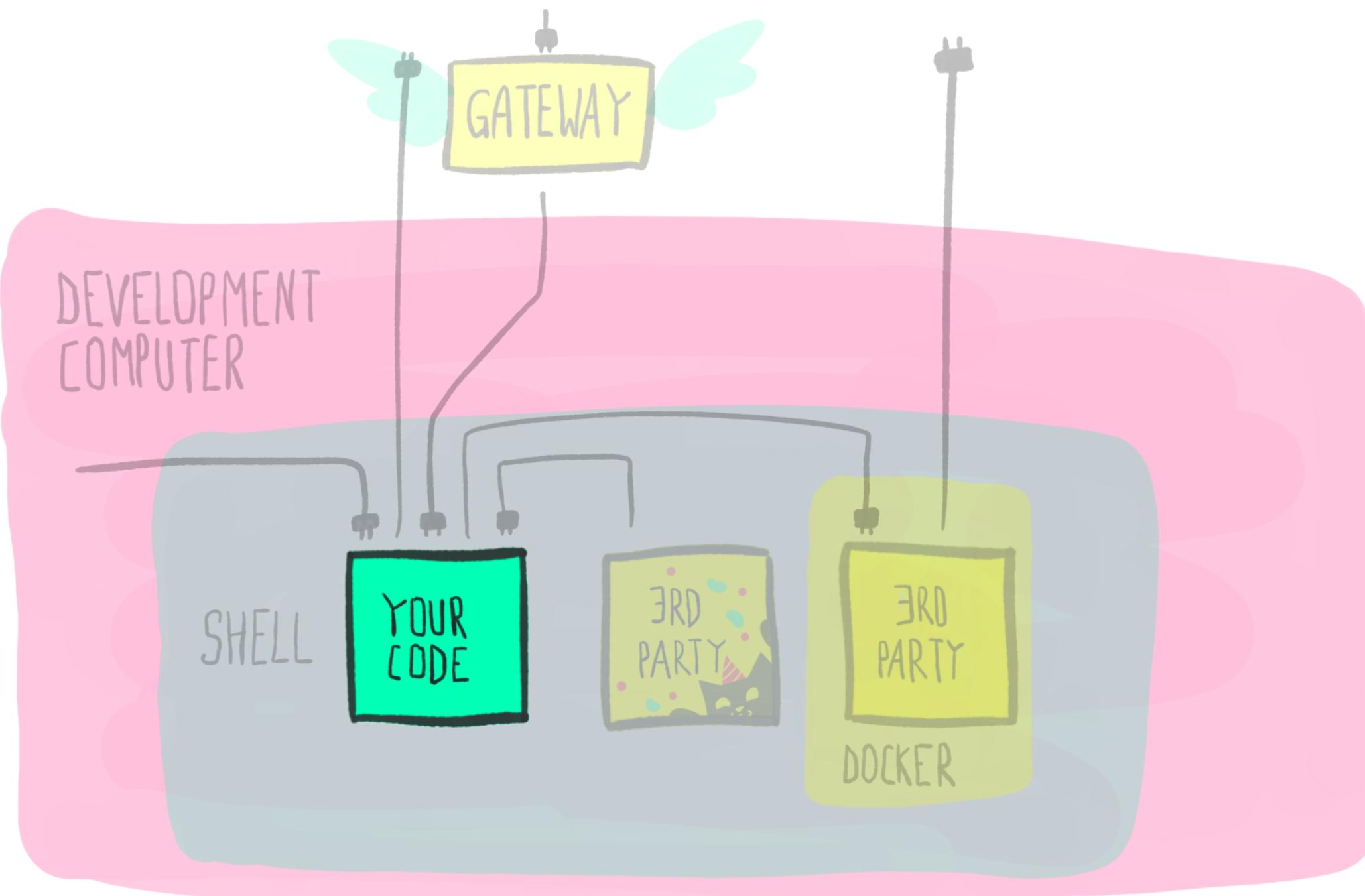








- Example in test app
-
- Remove —
 - Run natively Træfik
 - Replace it PostgreSQL → BoltDB
 - Container PostgreSQL



```
[zsh@bohr:~/interactive/ta]%.ta
Current directory is /Users/dottedmag/interactive/ta
Running parts/010-backend-with-db.service

+ cd ../src/backend
+ exec ta-gorun backend --boltdb-file=/Users/dottedmag/interactive/ta/_ta/data/backend.db
+ name=backend
+ shift
+ go build -o /var/folders/bw/6hyp_vyj68v973qbhlcvcvcr0000gn/T//ta/backend .
+ exec /var/folders/bw/6hyp_vyj68v973qbhlcvcvcr0000gn/T//ta/backend --boltdb-file=/Users/dottedmag/interactive/ta/_ta/data/backend.db
unknown flag: --boltdb-file
Usage of /var/folders/bw/6hyp_vyj68v973qbhlcvcvcr0000gn/T//ta/backend:
--boltbdb-file string      BoltDB file
--listen string            Listen address (default ":4444")
--postgresql-url string   PostgreSQL URL to connect
unknown flag: --boltdb-file

<<exited with code 2>>
Press Enter to restart

<Test Application> <atic 2:traefik 3:backend-with-db[exited] 4:db[exited] 5>
```

fix code → press “Enter” → done

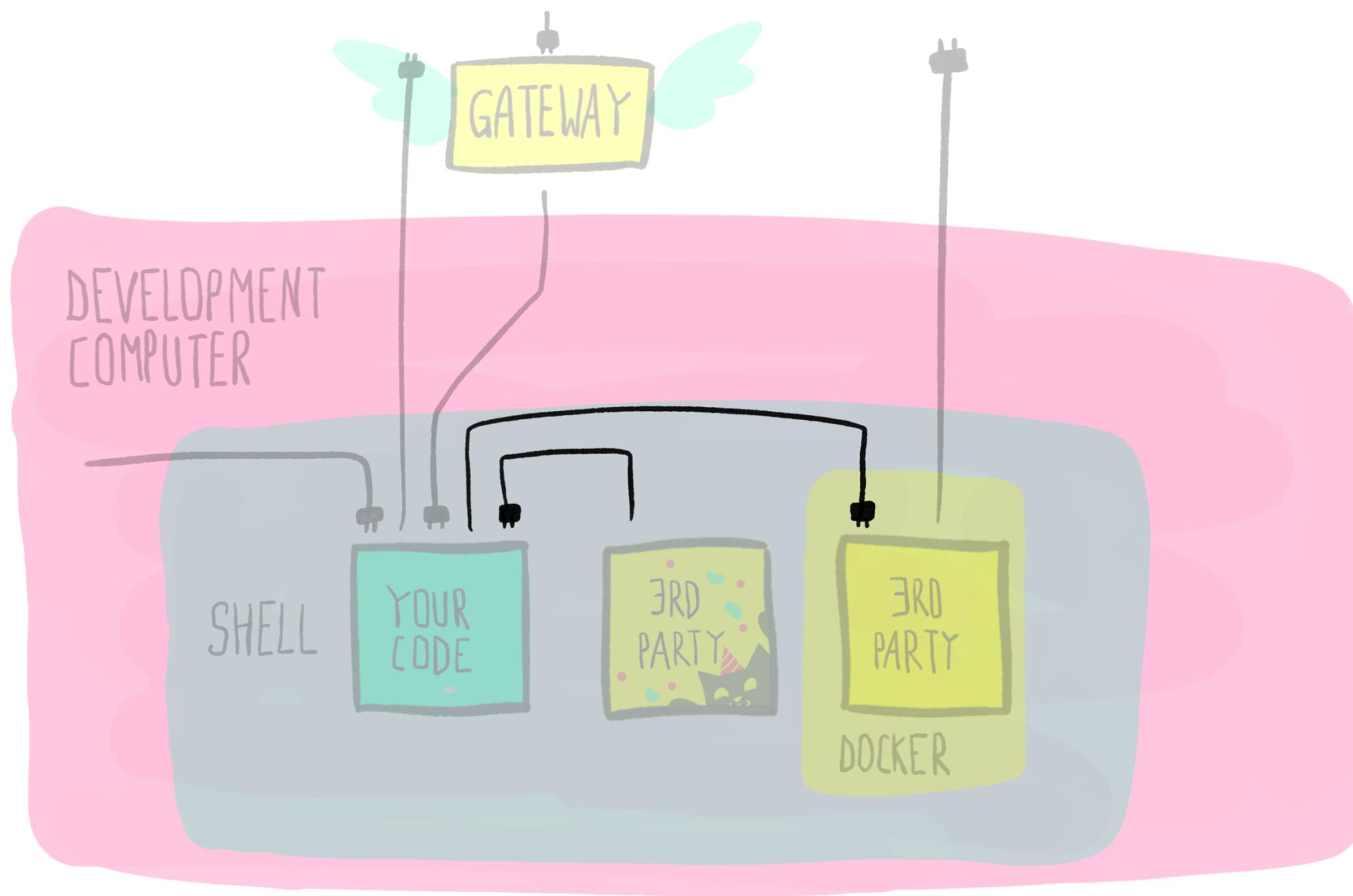
Project-specific. Ideally:

- go run
- hot code reload

```
[zsh@bohr:~/interactive/ta]%.ta
Current directory is /Users/dottedmag/interactive/ta
Running parts/010-backend-with-db.service

+ cd ../src/backend
+ exec ta-gorun backend --boltdb-file=/Users/dottedmag/interactive/ta/_ta/data/backend.db
+ name=backend
+ shift
+ go build -o /var/folders/bw/6hyp_vyj68v973qbhlcvcvcr0000gn/T//ta/backend .
+ exec /var/folders/bw/6hyp_vyj68v973qbhlcvcvcr0000gn/T//ta/backend --boltdb-file=/Users/dottedmag/interactive/ta/_ta/data/backend.db

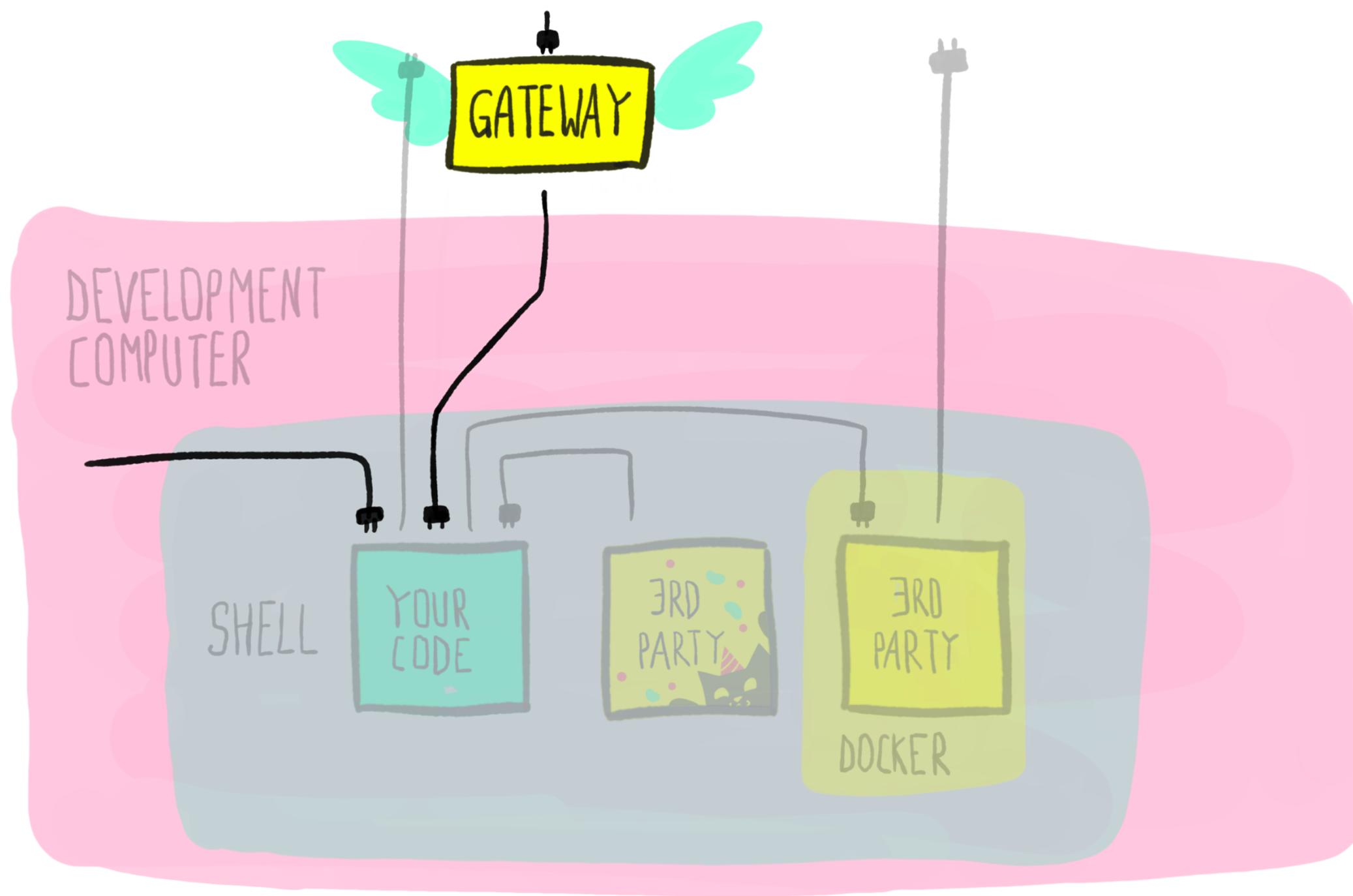
<Test Application> <1:static 2:traefik 3:backend-with-db 4:db[exited] 5:bac>
```



```
serviceA --listen=:4444
```

```
serviceB --listen=:44445 --serviceA-url=http://localhost:4444/
```

```
frontend: listen on 8000, use http://localhost:4444/ and  
http://localhost:44445/
```



- Gateway VM + HTTP server + SSH forwarding
- Gateway VM + SSH forwarding
- ngrok.io or similar

State



- One dir
 - via configuration
 - via Docker mounts
 - Logs too
- `ta destroy` removes `_ta`



Sync with production configuration



- Sync manually
- Generate
 - Prod configs → dev configs
 - Abstract description → prod configs, dev configs



Q&A

Ridge website ridge.co

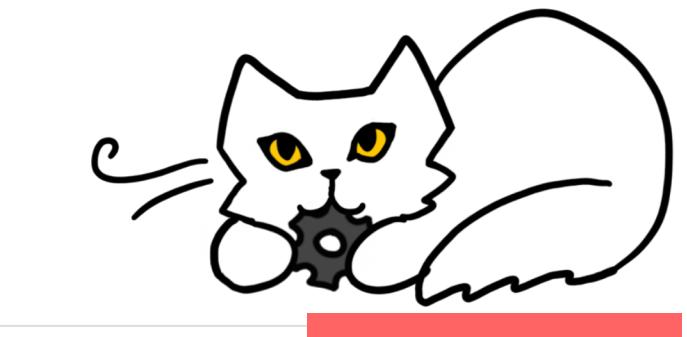
Me@ridge misha@ridge.co

Misha Gusarov dottedmag.net

Test application

<https://github.com/ridge/interactive>

Kittens courtesy of





Thanks

