

# Cloud-programmeren





THE TWELVE-FACTOR APP

# Twelve-Factor App



## **Manifesto van ontwikkelaars Heroku**

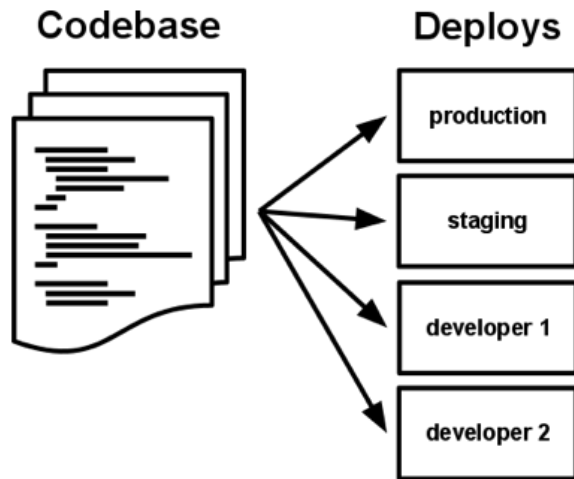
Hoe moderne, web-based, cloud-ready applicaties te bouwen

Interessante stellingnames

# Twelve-Factor App



## 1 – Eén codebase



- een repo
- meerdere deploys
- meerdere klanten

## 2 – Expliciet declareren en isoleren van afhankelijkheden

- bestaan tools en libraries nooit veronderstellen
- package manager zoals RubyGems, Maven, etc.

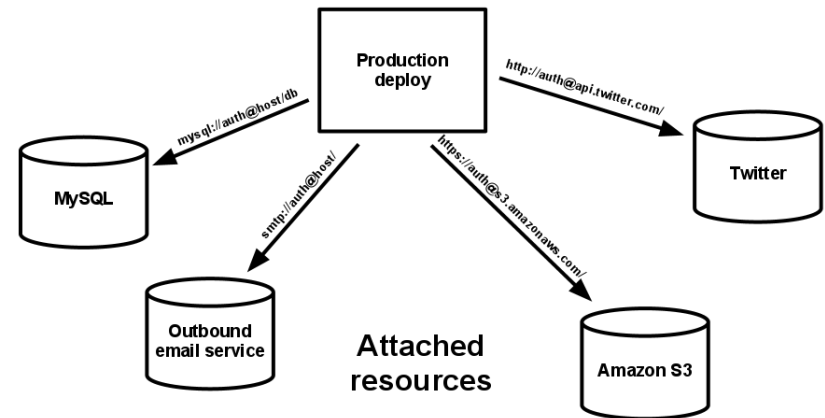
# Twelve-Factor App



## 3 – Configuratie in de environment

- stricte scheiding configuratie en code
- denk aan DTAP
- combi met config file mogelijk

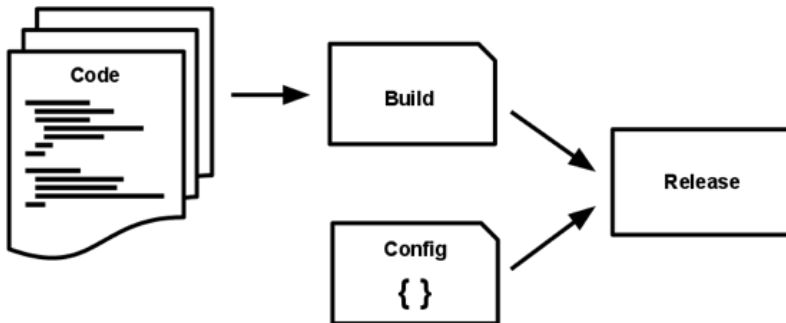
## 4 – Ondersteunende services als *attached resources*



# Twelve-Factor App



## 5 – Scheiding build, release, run



- release en run zonder developer mogelijk
- rollback makkelijk

## 6 – Een of meerdere stateless processen

- bv. één web server proces per server
- state in data store
- statelessness maakt robuust

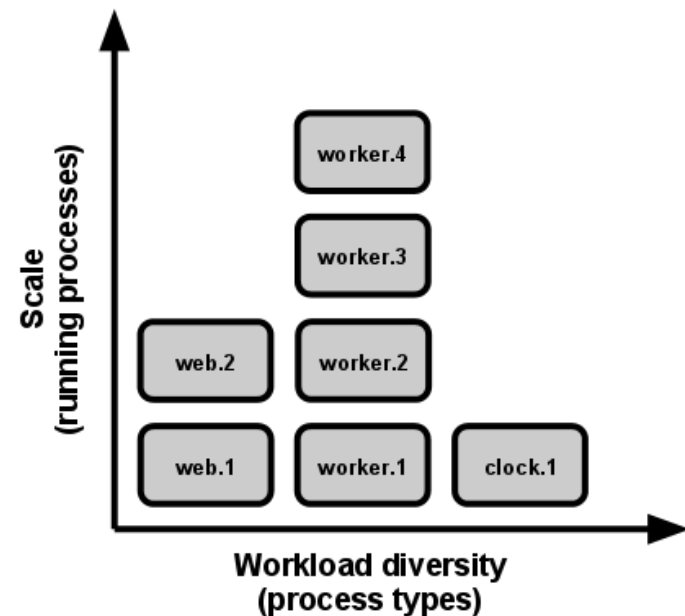
# Twelve-Factor App



## 7 – Self-contained, service beschikbaar via poort

- dus niet module in Apache/Tomcat/...
- `http://localhost:5000/`
- maakt elke app mogelijk een ondersteunende service

## 8 – Horizontaal schalen volgens process model



# Twelve-Factor App



## 9 – Snelle startup, cleane shutdown

- snelle startup, denk enkele seconden
- SIGTERM
- crash-bestendig

## 10 – dev/prod gelijkheid

- Code
- *Tijd*: uren i.p.v. weken
- *Personeel*: zelfde mensen i.p.v. andere
- *Tools/services*: zelfde i.p.v. andere
- vagrant, chef, docker



# Twelve-Factor App



## 11 – logs: alles naar stdout

- geen routing of opslag doen.
- laat een andere service de log verwerken, filteren, doorsturen, etc.
- ELK: Elasticsearch, Logstash, Kibana

## 12 – admin taken als eenmalige processen

bv.

- database migratie
- model inspectie via REPL
- scripts vanuit repo
- op de server zelf, of identieke environment



# Heroku is een PaaS-leverancier



- **IaaS: Infrastructure-as-a-service**
  - prompt in the sky: lege server
- **PaaS: Platform-as-a-service**
  - Mercedes, rode loper
    - configuring
    - monitoring
    - patching
    - etc.
  - Heroku, Pivotal, Amazon BeanStalk, Google App Engine, Azure Websites, etc.

# Heroku



Fast deployment

```
$ heroku create sushi
Creating sushi... done
http://sushi.herokuapp.com/ | git@heroku.com:sushi.git

$ git push heroku master
----> Heroku receiving push
----> Rails app detected
----> Compiled slug size is 8.0MB
http://sushi.herokuapp.com deployed to Heroku
```

# Heroku - scaling



dev-days-paris-2012-jr0cket [Open Application](#)





Performance **Resources** Activity Collaborators Settings

Dynos

☒ web java \$JAVA\_OPTS -Dspring.profiles.active=prod -jar target/d... **\$970.50**

**28**

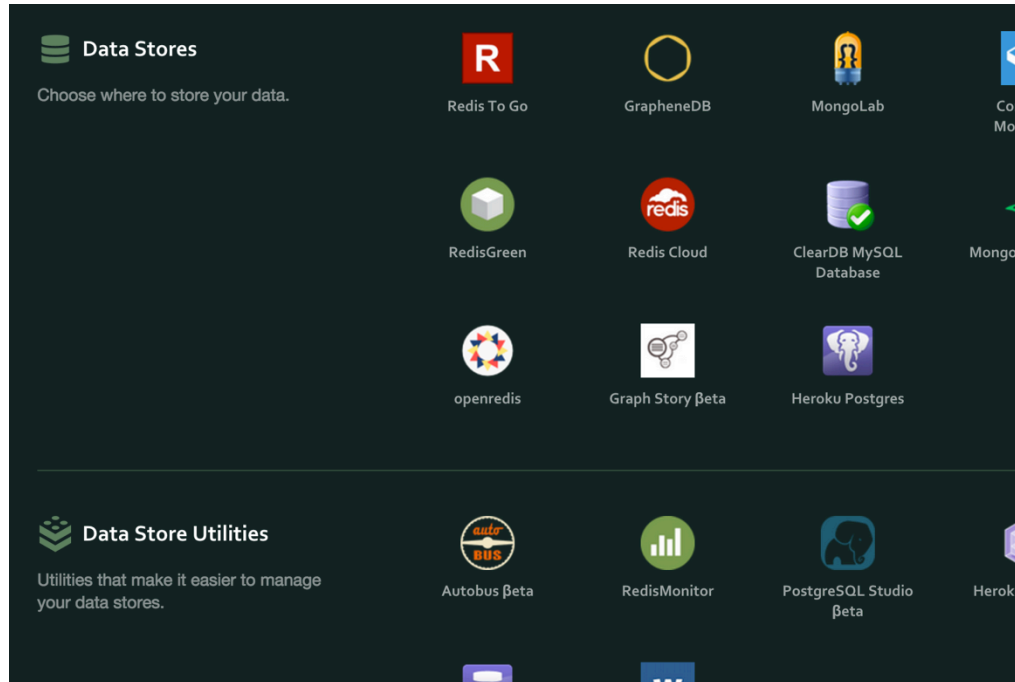
Add-ons

-  Heroku Postgres Dev :: Black Free
-  Kerosene Test Alpha
-  PG Backups Plus Free
-  [Get Add-ons](#)

# Heroku - addons



- <https://addons.heroku.com/>

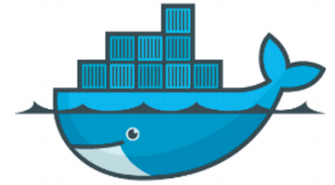


# Heroku tegen hun eigen Twelve Factors



1. Eén codebase
2. Expliciet declareren en isoleren van afhankelijkheden
3. Configuratie in de environment
- 4. Ondersteunende services als *attached resources***
5. Scheiding build, release, run
6. Een of meerdere stateless processen
- 7. Self-contained, service beschikbaar via poort**
- 8. Horizontaal schalen volgens process model**
9. Snelle startup, cleane shutdown
10. Dev/prod gelijkheid
11. Logs: alles naar stdout
12. Admin taken als eenmalige processen

# Docker – Twelve Factors



1. Eén codebase
2. Expliciet declareren en isoleren van afhankelijkheden
3. Configuratie in de environment
4. Ondersteunende services als *attached resources*
5. **Scheiding build, release, run**
6. **Een of meerdere stateless processen**
7. **Self-contained, service beschikbaar via poort**
8. Horizontaal schalen volgens process model
9. Snelle startup, cleane shutdown
10. **Dev/prod gelijkheid**
11. Logs: alles naar stdout
12. Admin taken als eenmalige processen



# Veel verschillende cloudproviders

## RankCloudz Online - Customizable Cloud Vendor Evaluation

Dev & Test  
Infrastructure

Virtual Data  
Center

Enterprise  
Apps &  
Hosting

Storage &  
Backup

Big Data &  
Analytics

### Quick Tips

Click on any Requirement name in the left side panel below to see evaluation based only that requirement.

All Vendors



Create In-depth report

### Technical or Business Requirements

Priorities  
(0 - 10)

Reset

Application Services

5

Compute

8

Data Protection

5

Deployment Management

7

Ease of Use

8

Governance and Regulat...

3

High Availability & Disas...

6

Infrastructure

5

Integration with Existing ...

7

Operating Cost

9

Performance

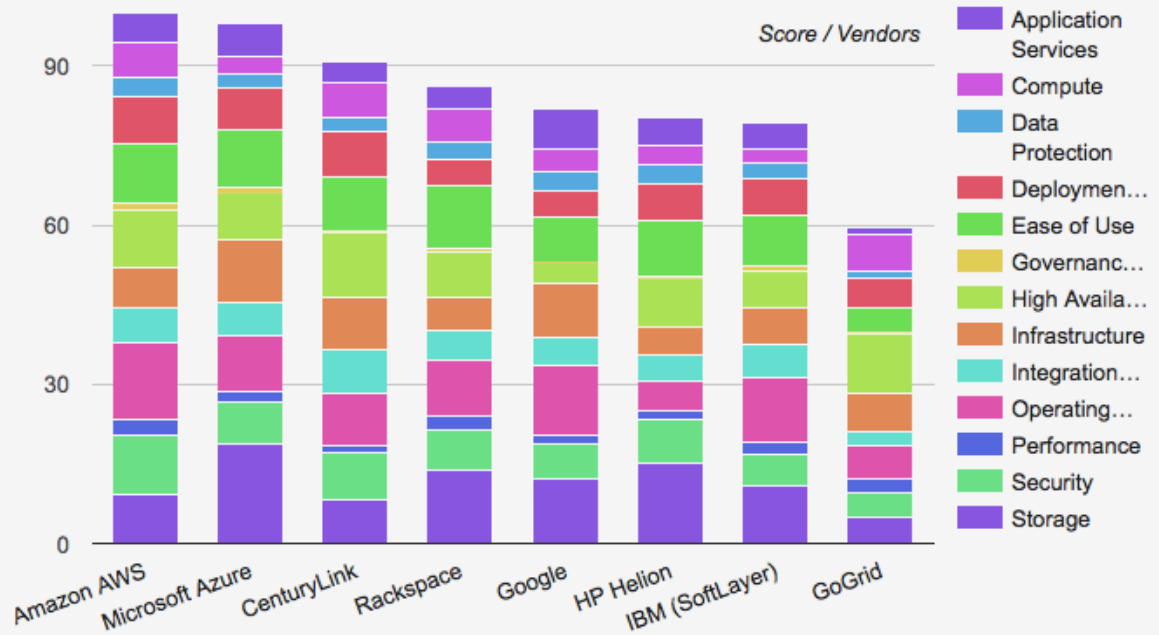
6

Security

5

Storage

6



# Factoren bij keuze cloudleverancier

- Kosten: belangrijk maar moeilijk in te schatten
- Openheid: Sommige leveranciers (Amazon, Microsoft) schrijven bepaalde software voor. Andere werken met OpenStack
- Privacy: Amerikaanse leveranciers staan **waarschijnlijk altijd** onder toezicht van NSA.