
Road To DevOps
Maarten van den Ende

Maarten van den Ende

Job

Consultant & Trainer

Contact

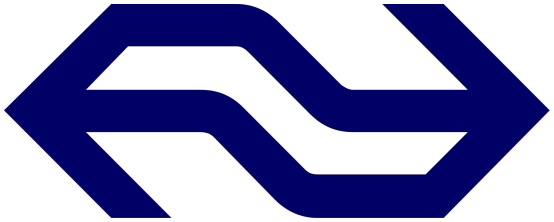
mvandenende@xebia.com

nl.linkedin.com/in/mjvdende

@mjvdende


ING

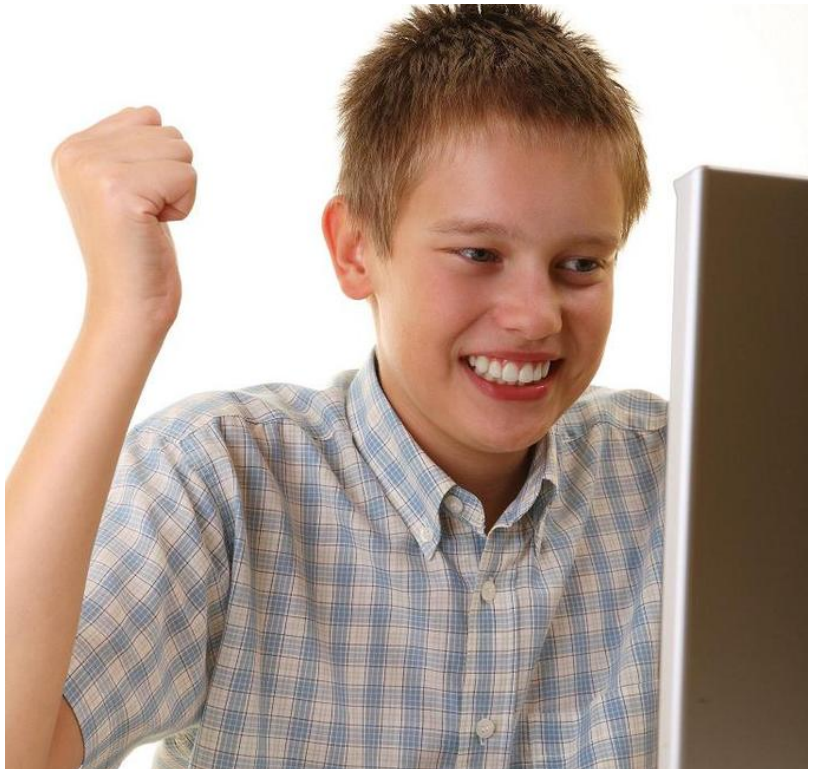

RBS
The Royal Bank of Scotland



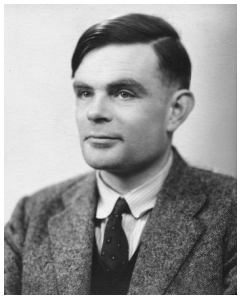

ABN·AMRO


ORACLE®


Protractor
end to end testing for AngularJS
Atlassian
MESOS
kibana
FitNesse
Se
docker
ANSIBLE
logstash
appium
JIRA
Gatling
STRESS TOOL
Atlassian
Stash
puppet
labs
VAGRANT
SoapUI
GitHub
CFEngine
Maven
git
amazon
web services™
CloudBees
gradle
VirtualBox
SUBVERSION
XL DEPLOY
Jenkins
Nexus
artifactory
NOLIO
php
MySQL™
CentOS
ubuntu
Java
JS
GRUNT
Apache
SOFTWARE FOUNDATION
CoreOS



- Started doing DevOps by accident 2009
- Never deployed to production
- Web focus



Roughly speaking those who work in connection with the [Automated Computing Engine] will be divided into its masters and its servants. Its masters will plan out instruction tables for it, thinking up deeper and deeper ways of using it. Its servants will feed it with cards as it calls for them. They will put right any parts that go wrong. They will assemble data that it requires. In fact the servants will take the place of limbs. As time goes on the calculator itself will take over the functions both of masters and of servants

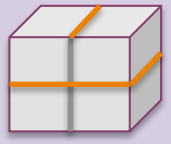
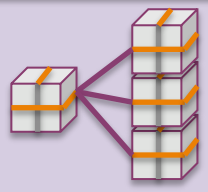
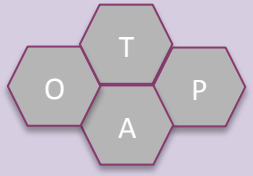
Turing, Alan. The Essential Turing, edited by B. Jack Copeland. Oxford University Press, 2004

Continuous Delivery

- Creëer cross functional product teams
- Teams beheren services gehele lifecycle
- Inclusief automatisering van bouw, test en deployment proces

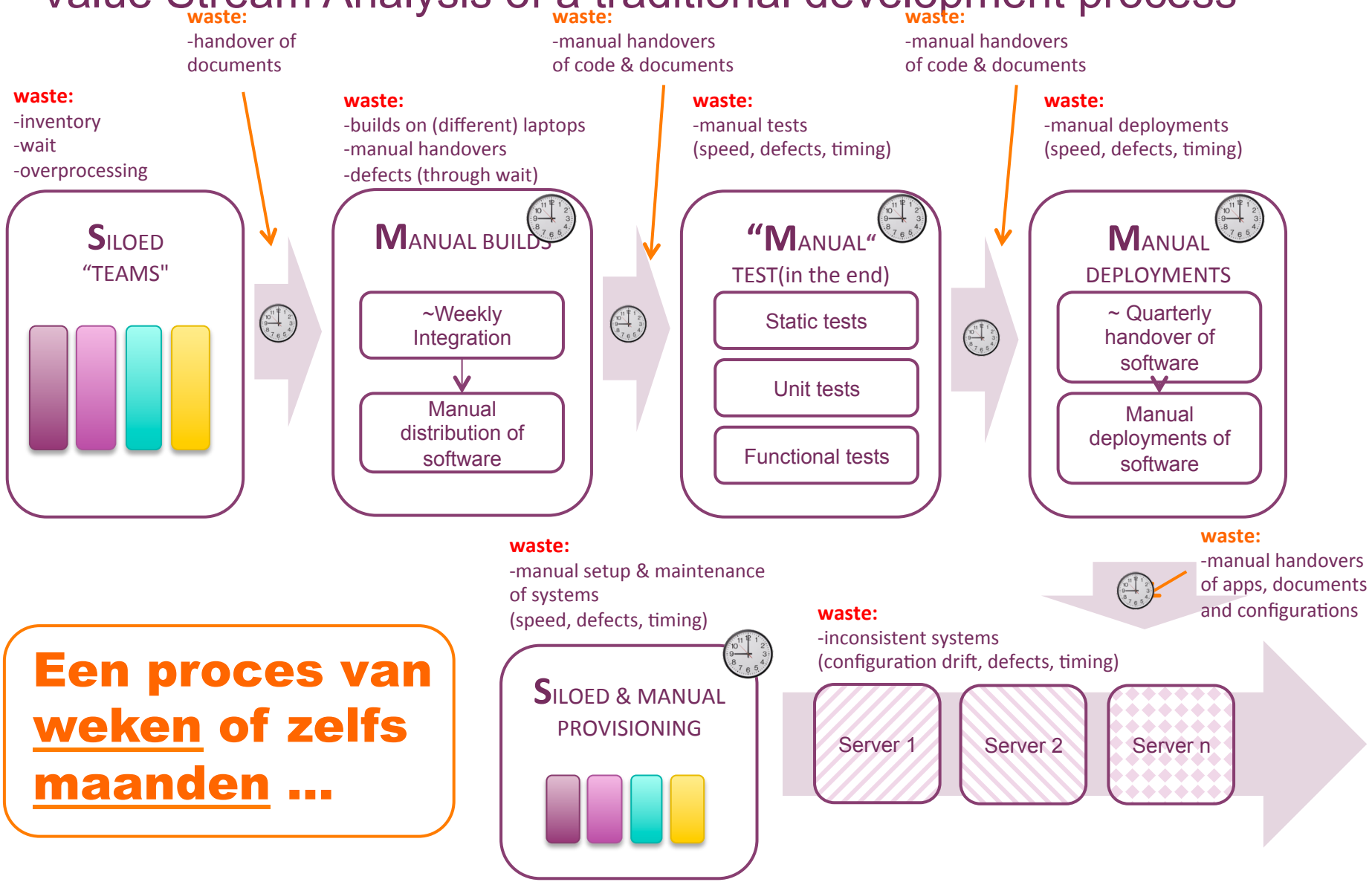
Remove Waste

FULLY AUTOMATED SOFTWARE DELIVERY PROCESS

AGILE PROCESS	AUTOMATED BUILD	AUTOMATED TEST	AUTOMATED DEPLOYMENT	AUTOMATED PROVISIONING
				
• Deliver fast • Deliver often • Do the right things	• Improve quality • Increase predictability	• Improve reliability • Repeatable • Reduce Cost • Increase speed	• Release insight • Reduce release time • Reduce errors • Less downtime • Cost reduction	• Reduce costs • Increase speed • Reduce risk • Reduce Cost
	Subversion, Jenkins, Nexus, Maven	*Fitnesse, Selenium, Xebium, NGrinder*	*Deployit*	

AGILE PROCESS

Value Stream Analysis of a traditional development process



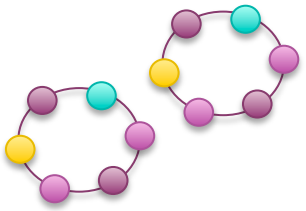
Een proces van weken of zelfs maanden ...

Value Stream Analysis of an optimized development process

characteristics:
-deliver fast
-deliver often
-do the right things

characteristics
-output: product

AGILE TEAMS



characteristics
-improve quality
-compile & construct continuously

AUTOMATED BUILD

- Continuous Integration
- Dynamic Software Library

characteristics
-continuous flow of code

characteristics
-improve quality
-test continuously
-reduce defects & costs

AUTOMATED TEST

- Static tests
- Unit tests
- Functional tests

characteristics
-continuous flow of code

characteristics
-release continuously (trust!)
-release consistently
-reduce costs & defects

AUTOMATED DEPLOY

- Release Man. (Light)
- Deployment Automation

characteristics
-continuous flow of apps & configurations

characteristics
-increase setup speed
-reduce cost
-reduce defects

AUTOMATED PROVISIONING

- Provisioning

characteristics
-consistent systems

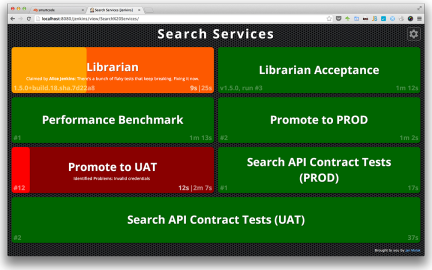
- Server 1
- Server 2
- Server n

€

characteristics
-up to date end-product @ customer

Een proces van minuten ...

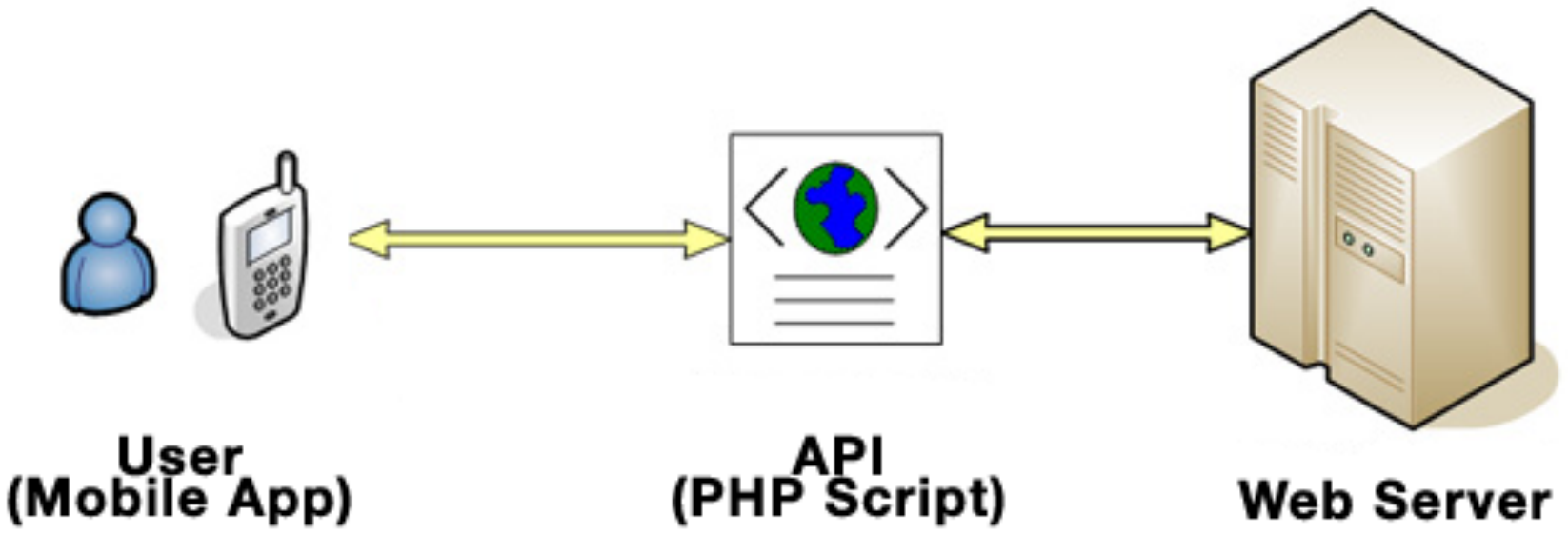
DevOps First Steps



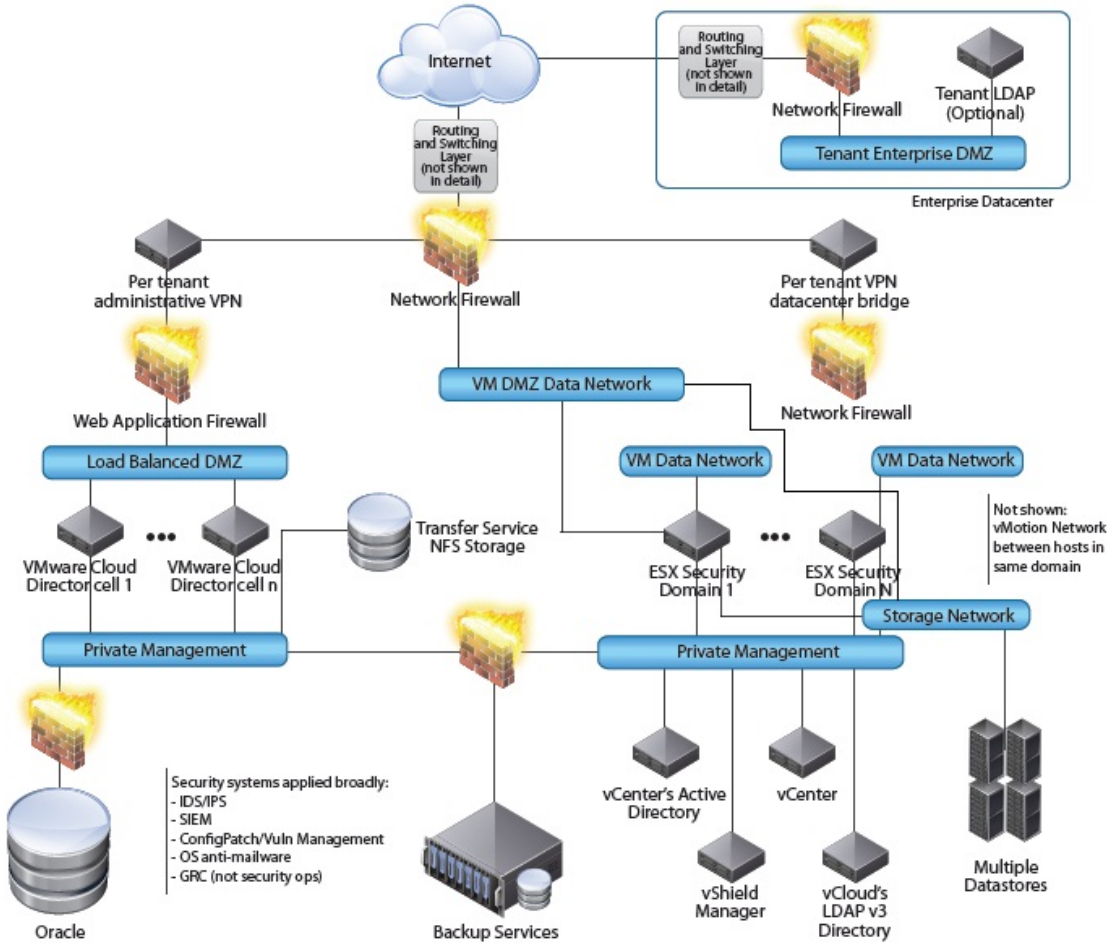
Anti Patterns

- Deploying software handmatig
- Alleen naar productie (like) wanneer development helemaal klaar was
- Manual config van productie omgevingen

Developer



Operations


Internet
Routing and Switching Layer (not shown in detail)
Network Firewall
Tenant LDAP (Optional)
Tenant Enterprise DMZ
Enterprise Datacenter
Routing and Switching Layer (not shown in detail)
Per tenant administrative VPN
Network Firewall
Per tenant VPN datacenter bridge
Web Application Firewall
VM DMZ Data Network
Network Firewall
Load Balanced DMZ
VM Data Network
VM Data Network
Not shown: vMotion Network between hosts in same domain
VMware Cloud Director cell 1
VMware Cloud Director cell n
Transfer Service NFS Storage
ESX Security Domain 1
ESX Security Domain N
Storage Network
Private Management
Private Management
vCenter's Active Directory
vCenter
Multiple Datastores
vShield Manager
vCloud's LDAP v3 Directory
Security systems applied broadly:
- IDS/IPS
- SIEM
- ConfigPatch/VuIn Management
- OS anti-mailware
- GRC (not security ops)
Oracle
Backup Services

Solution > DevOps

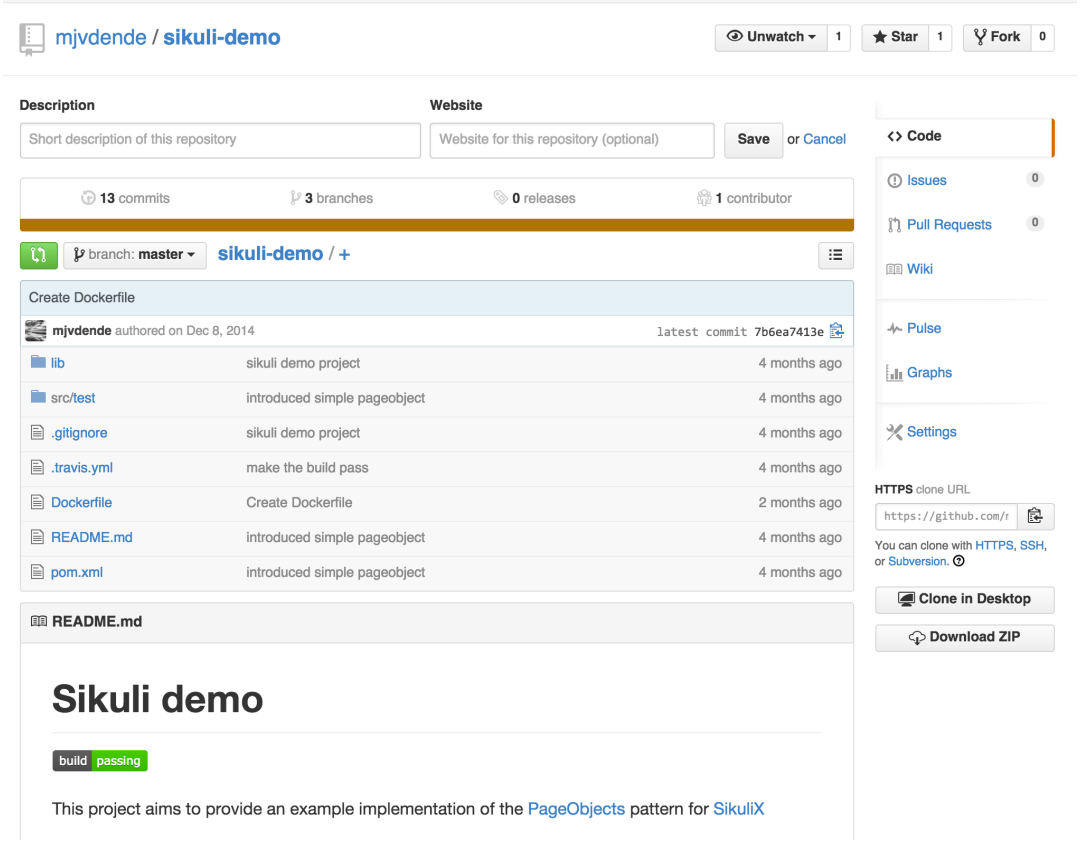
- It is a Movement
- CAMS
- Collaboration
- Communication
- Integration

Culture ams

- Break the silos

cAutomationms

- Automate all the things



ca Measurements

- Issues
- Builds and Tests
- Code Analytics
- Deploy Statistics

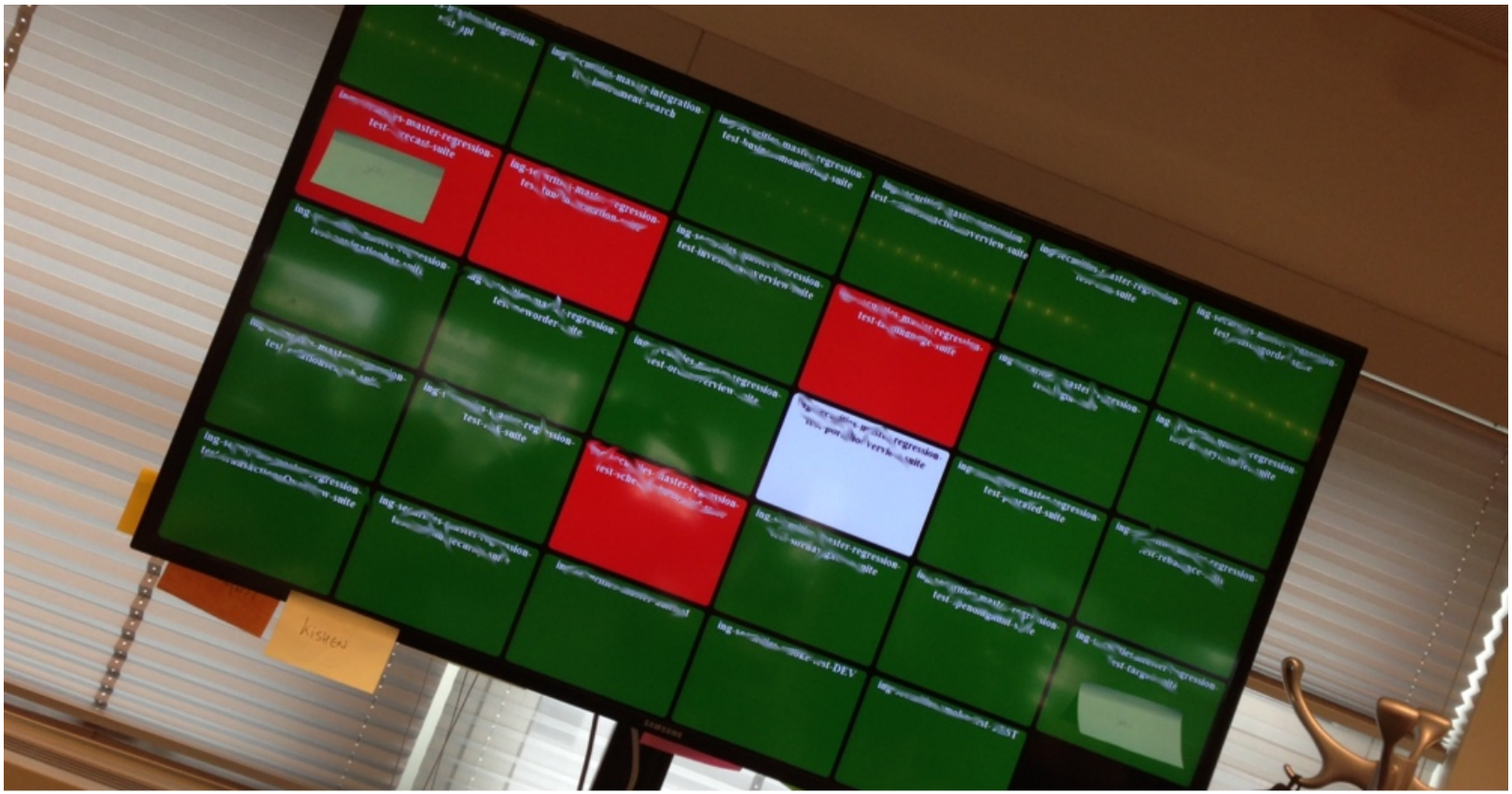
ca Measurements

- Operating System Statistics
- Middleware statistics
- Application Metrics
- New Feature impact > A/B testing
- Availability metrics

camSharing

- Deel metrics in organisatie
- Publiceer Code / Open source
- Tech Fridays / Meetup.com

DevOps Second Attempt



Build

- 3000+ Unit tests uitgevoerd voor commit
- Na elke commit CI

Test

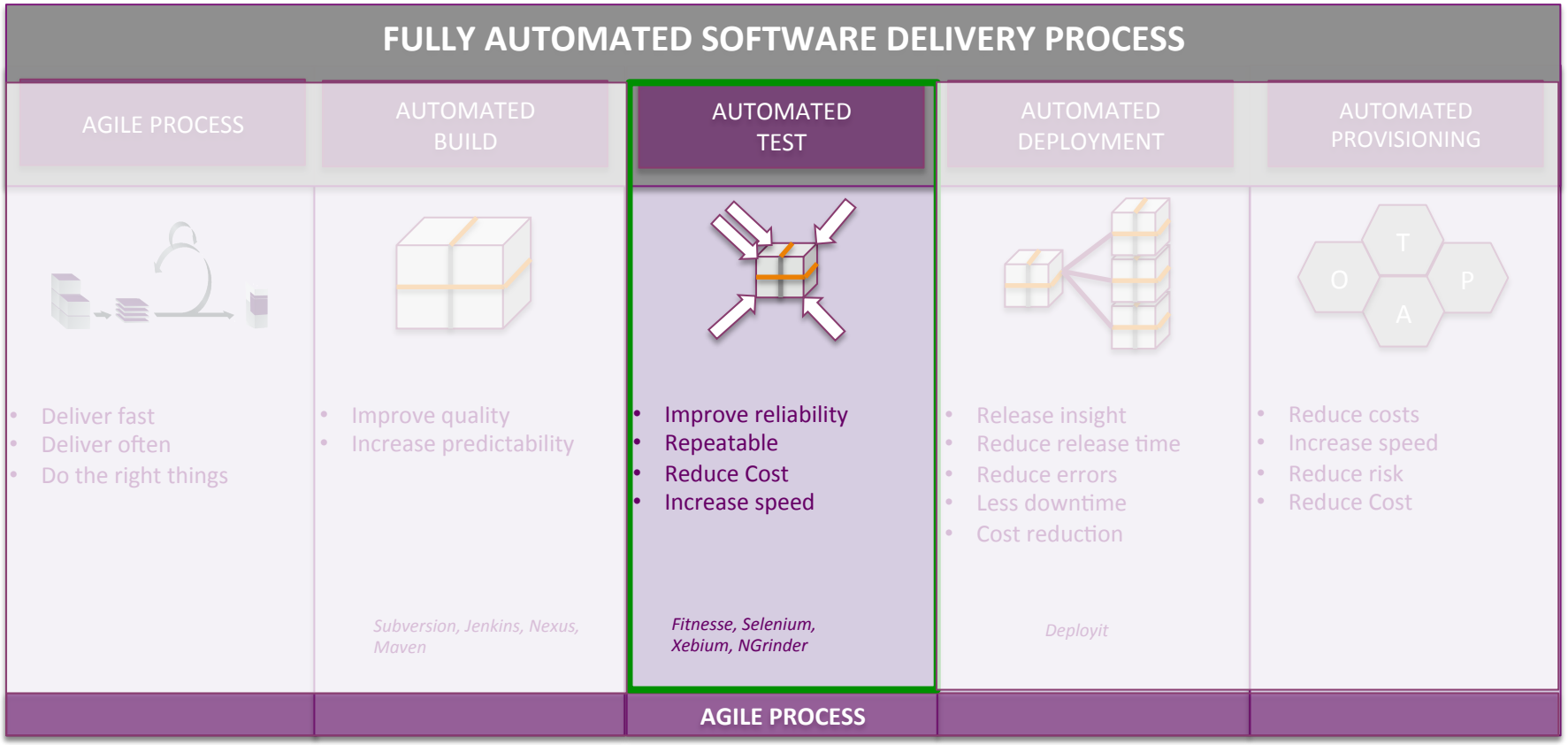
- Na elke commit 300+ automated functional tests 

- Code Analysis 

Deploy

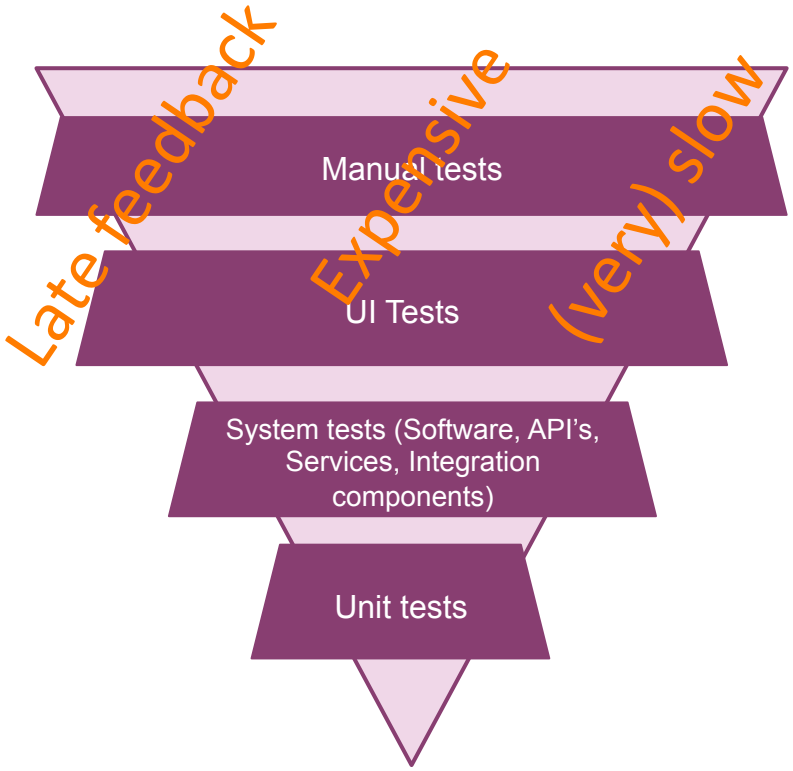
- CD > Deploy nightly build naar productie-like environment
- Smoke tests
- Manual tests, production data
- Elke twee weken release naar productie

Automated Test



TEST Automation

”automate your tests!” – a very costly ANTIpattern!



http://watirmelon.com/2012/01/31/introducing-the-software-testing-ice-cream-cone/

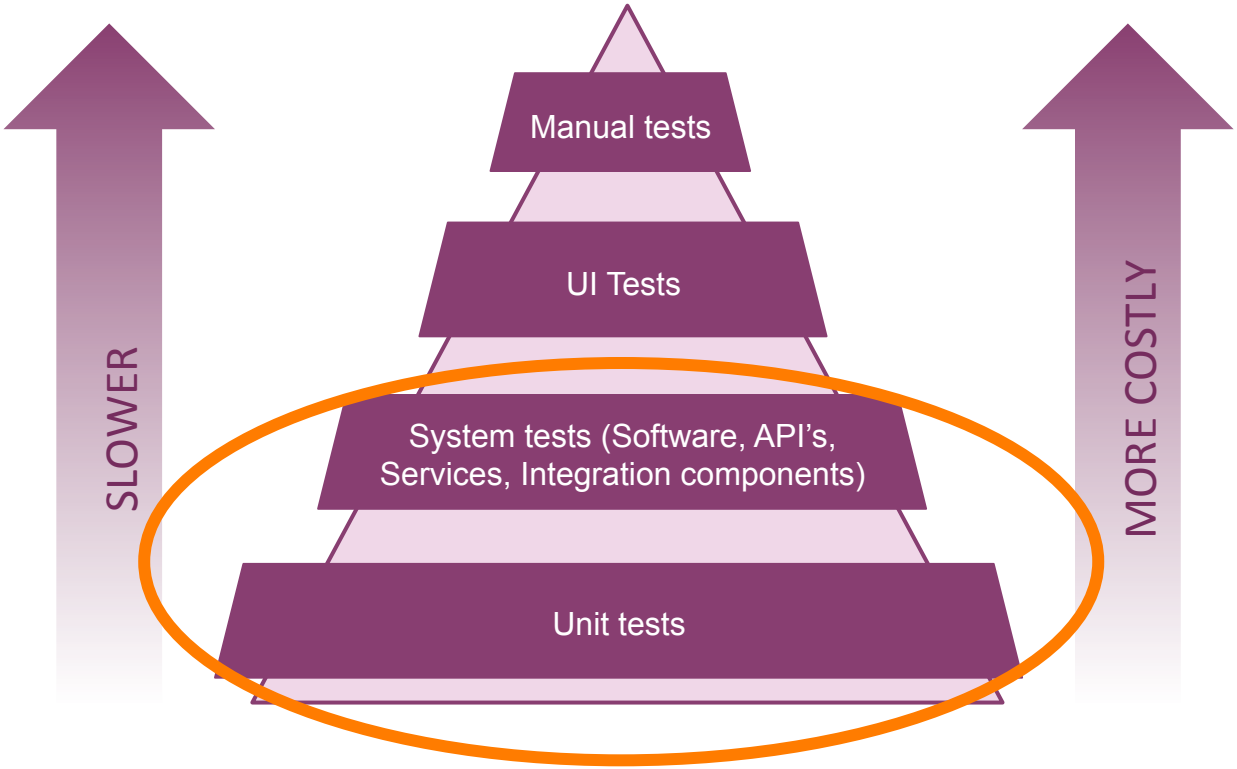
TEST Automation
”automate your TESTS!”

Erg langzaam, inconsequent feedback. Duur. Late feedback.

Langzaam, getest na deployment, saai, late feedback.

Snel, getest na deployment, Direct feedback.

Snel, getest voor deployment, direct feedback, “in detail”.



Daarom: test first !!!

Test Automation

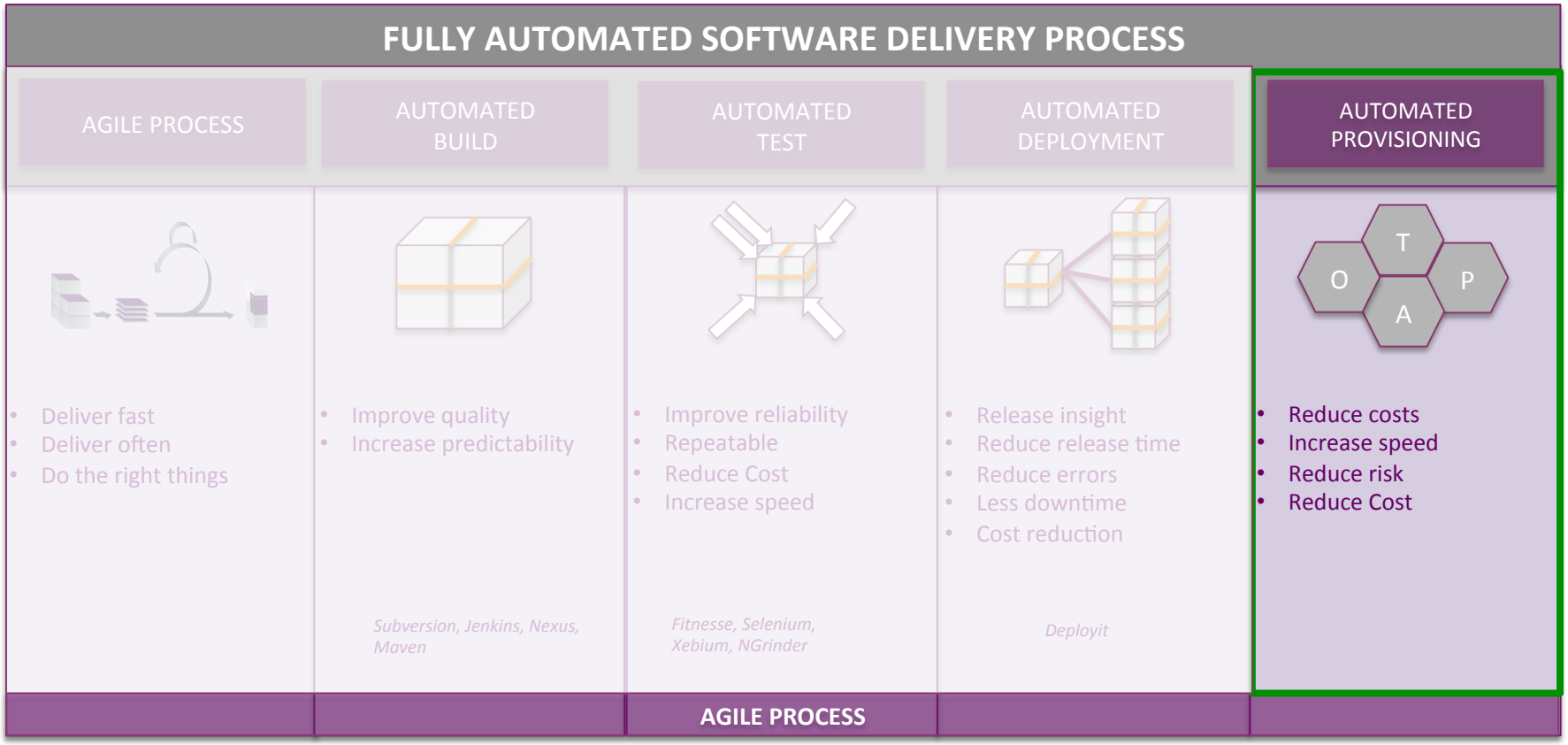
”automatiseer testen!” – in multi disciplinaire Feature teams!

- In Silo’s testen is langzaam, erg duur
- Testers en developers moeten samenwerken aan testen schrijven, uitvoeren, automatiseren en onderhouden.

  - “Whole team testing”
  - Test code is productie code

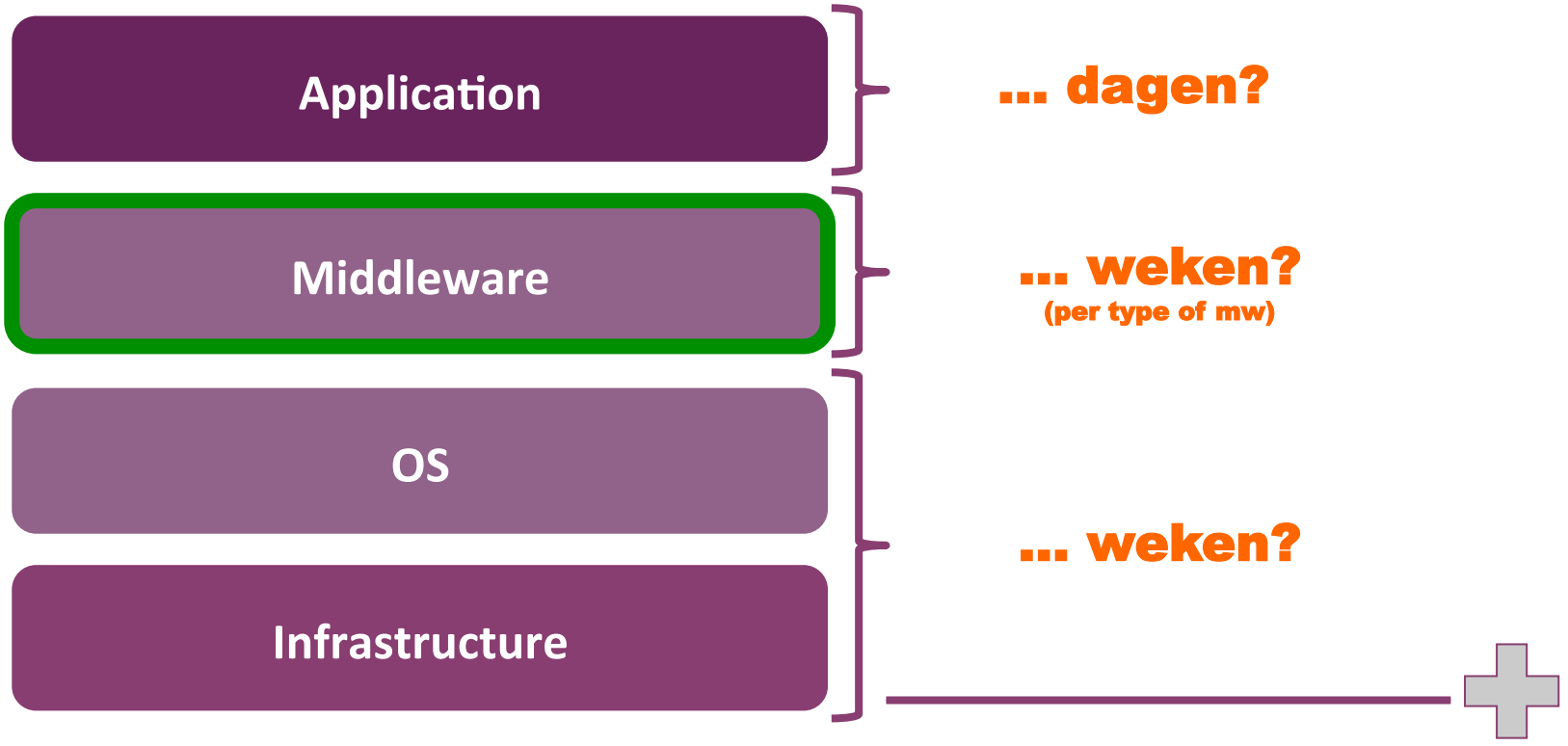
- Tests moeten gemaakt worden voordat er een regel code is geschreven

Automated Test



Automated Provisioning

”automate system setup” – hoe lang duurt provisioning in uw organisatie?



.... Voegt toe aan elk project:

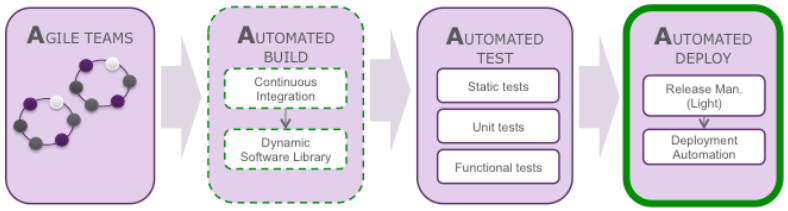
?? Maanden lead tijd

Automated Provisioning

”automate system setup” – technical landscape can become complex

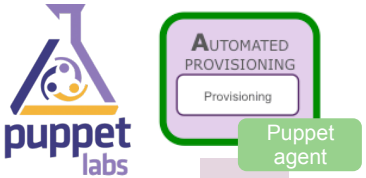
Automated Provisioning

”automatiseer system setup”



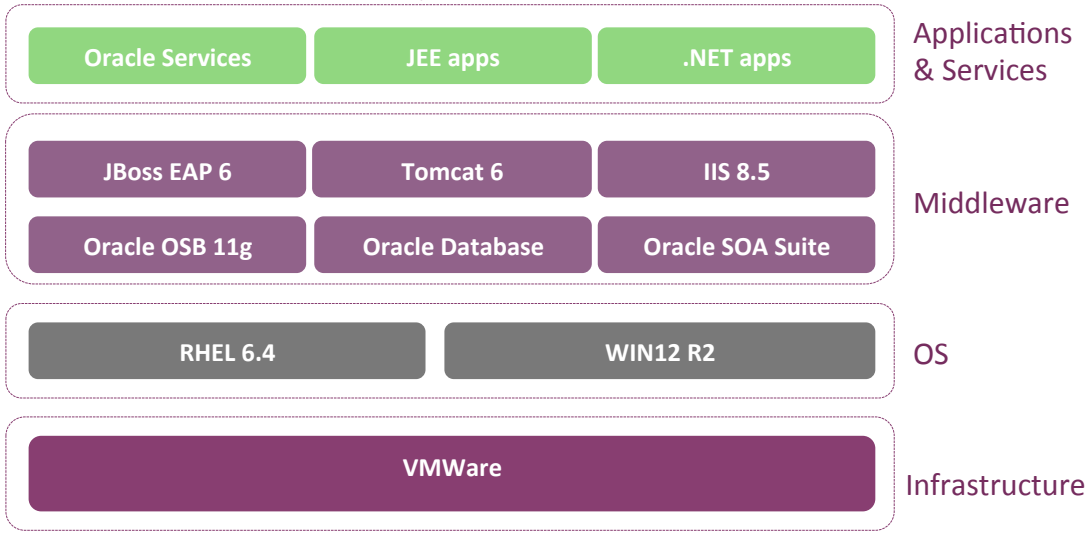
Deploy

in 10 minuten…



Provision

in 30 minuten…



Automated Provisioning

”automate system setup” – Configuration Management Tooling

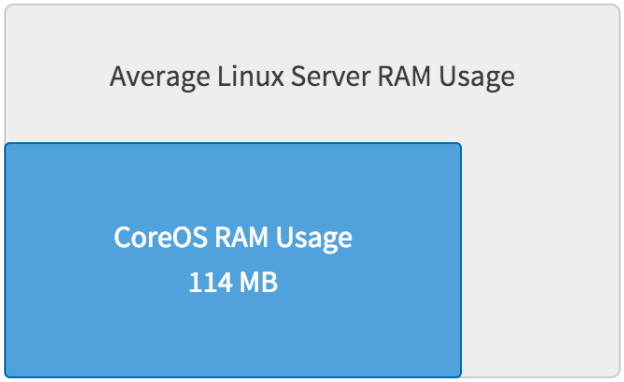
		
Python	Ruby	Python
YAML	DSL	YAML
Master	Master/Agent	Master/Minion
Python API	no API	Python API
Push based	Pull based	Push based
Executed in order	Executed as specified	Executed in order
>200 core types	48 core types	>200 core types

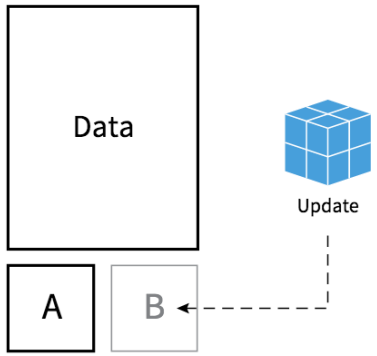
DevOps done right

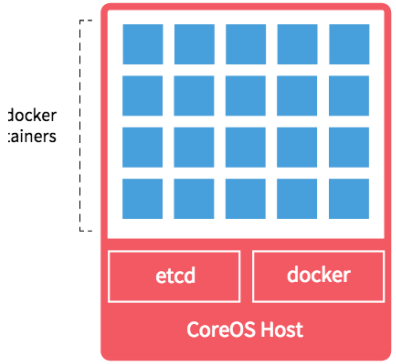
- New breed OS
  - CoreOS
  - Redhat Atomic
- Api for developer to provision environments
- Docker Containers

New breed of OS

- Atomic, Ubuntu Core en CoreOS

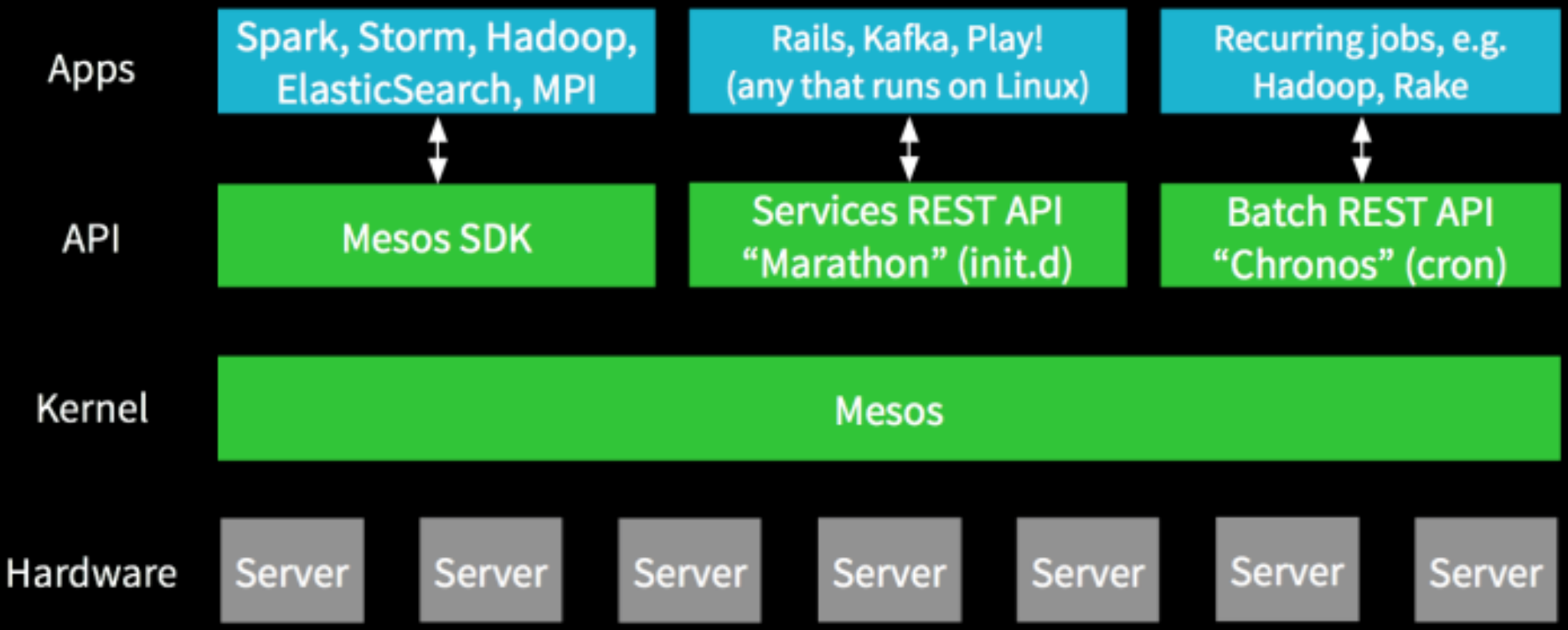






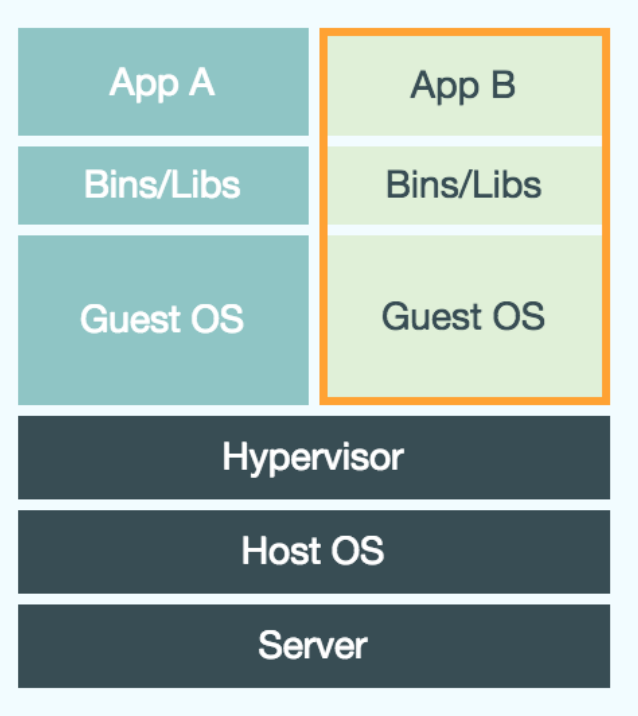
Api for dev Mesosphere

- Mesos
  - Master
  - Slave
- Marathon
  - Initd system
- Hadoop, Kafka, Spark

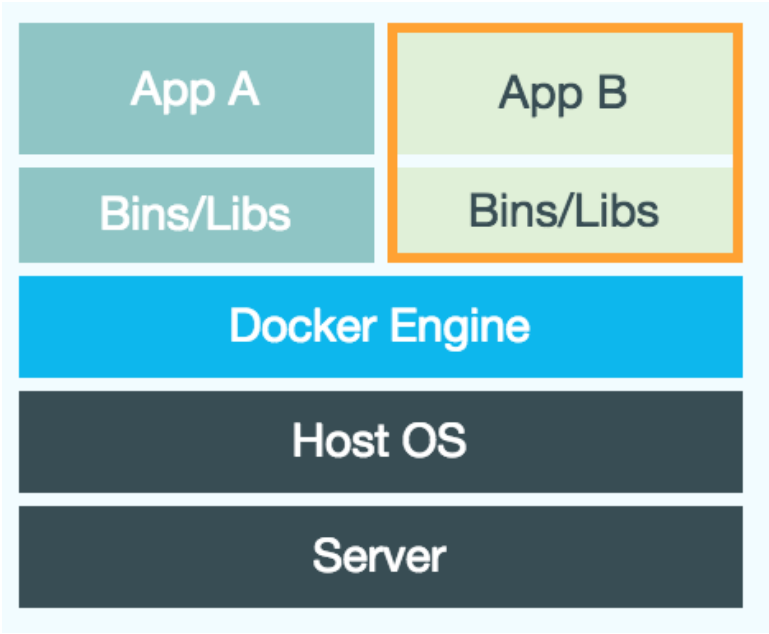

Apps
Spark, Storm, Hadoop, ElasticSearch, MPI
Rails, Kafka, Play! (any that runs on Linux)
Recurring jobs, e.g. Hadoop, Rake
API
Mesos SDK
Services REST API "Marathon" (init.d)
Batch REST API "Chronos" (cron)
Kernel
Mesos
Hardware
Server
Server
Server
Server
Server
Server
Server

Introducing Docker

Virtual Machine



Docker



Create CD pipeline

- https://github.com/mjvdende/vagrant-mesos

Installeer git, virtualbox en vagrant (+plugins)

```
$ git clone https://github.com/mjvdende/vagrant-mesos.git
$ cd standalone && vagrant up
```

TL;DR

- Zonder test automatering geen continuous delivery
- Iedereen in het team draagt bij aan automatisering
- Manueel testen beperken tot een minimum
- Build > Measure > Learn

Thank you!
Questions?

@mjvdende - mvandenende@xebia.com