

FORTEZZA


Continuous Delivery
Hype or must?

Rijkswaterstaat neemt
je mee in de praktijk

between

MEETUP

12 FEBRUARI

17:30 Kuiperweg 50
1101 AG
Amsterdam

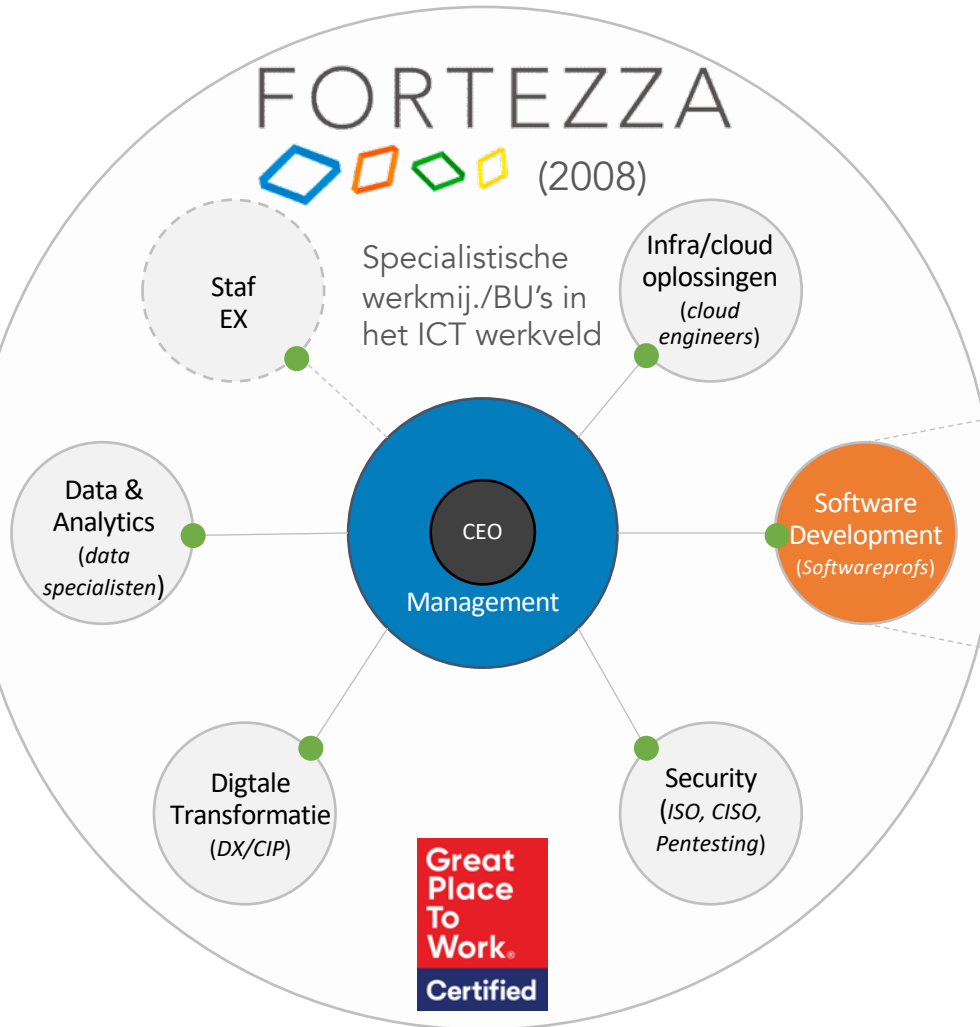
- 18:20 – Welkom door William Janse
- 18:30 – Presentatie Allard Baronner
- 19:30 – Presentatie Wijnand Top
- 20:30 – Borrel



Rijkswaterstaat
Ministerie van Infrastructuur en Waterstaat

FORTEZZA

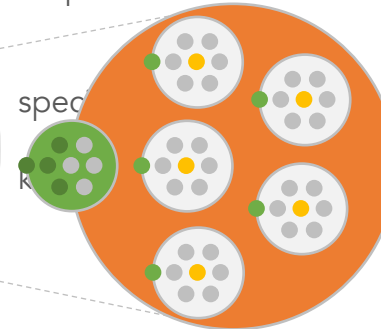
(2008)



Fortezza Softwareprofs (2015)

Belief: "Alles wordt software (Softwarization => DX)"

Purpose: "Werken aan een beter digitaal Nederland"



vanuit onze

- Java
- .NET kennis, kunde &
- Python
- Front-end
- DevOps
- Architectuur

===

Digitale Rechtspraak, Rijkswaterstaat, NDW, IL&T, VWS/PGB 2.0, SVB, NFI en veel meer...

===

Fortezza Meetups

- **GPTW als CI van Fortezza:**
 - Excelleren op de thema's Ontwikkelen & Inspireren.



- **Ontwikkelen:**
 - **Fortezza Meetups (> 6x per jaar);**
 - Intern diverse Kennissessies, Guild Nights en relevante Trainingen/Certificeringen;
 - Top opdrachten & dito opdrachtgevers;
 - Kwaliteit boven kwantiteit!
- **Inspireren:**
 - Met mooie leerzame ervaringen van en bij opdrachtgevers waar Fortezza mede i.s.m. Between een bijdrage levert.
- **Continuous Delivery; Hype or Must?**
 - 18:30 – Presentatie Allard Baronner
 - 19:30 – Presentatie Wijnand Top
 - 20:30 – Borrel



Rijkswaterstaat
Ministerie van Infrastructuur en Waterstaat

Uitdagingen

- Zo hebben we het altijd al gedaan
 - Dat hebben we nooit eerder gedaan
 - Daar zijn wij nog niet rijp voor
 - Dat maakt ons overbodig
 - Dat werkt niet in onze organisatie
 - Sta je er garant voor dat het werkt?
 - Waarom wijzigen, het werkt toch?
 - Dat is in strijd met ons beleid
 - Dat kost te veel
 - Dat is niet onze verantwoordelijkheid
 - Dat is een te grote verandering
 - Daar hebben we geen tijd voor
 - Het past niet in ons budget
 - Roei nu maar met de riemen die je hebt
 - Mag niet van Security/BIO
- Ja, maar...
 - Schrijf maar een business case
 - Laten we het maar in beraad houden
 - We hebben het ook zonder kunnen rooien
 - Is het al eens uitgeprobeerd?
 - Wat we hebben is goed genoeg
 - Dan moeten we de contracten openbreken
 - Infra accepteert dat nooit
 - Dan klopt het model/proces niet meer
 - Het heeft onze aandacht
 - Sinds wanneer ben jij een expert geworden?
 - Dat is ons probleem niet
 - Je bent zo enthousiast, jij houdt het hier niet lang vol
 - Het management wil dit toch niet



Onderwerpen



Theorie



Praktijk binnen Rijkswaterstaat



Rijkswaterstaat
Ministerie van Infrastructuur en Waterstaat

Continuous Delivery in de theorie

Continuous Delivery hype or must

Allard Baronner & Wijnand Top

Meetup 12 februari 2020

SOFTWARE- ONTWIKKELING

- Missiekritiek
- Complexer
- Omvangrijker
- Wendbaarder
- Kortere *Time-to-market*

Waterval -->

Agile -->

Continuous Delivery!

IMPACT



Editor: Michiel van Genuchten
VitalHealth Software
genuchten@ieee.org



Editor: Les Hatton
Oakwood Computing Associates
lesh@oakcomp.co.uk

An Information and Tracking System for Inland Shipping

Therry van der Burgt and Allard Baronner

This issue's article describes the software that supervises transportation on Dutch waterways. Among other things, the article shows how changes in the investment level have affected the software's growth over the years. —*Michiel van Genuchten and Les Hatton*



WITH THOUSANDS OF vessels transporting over 350 million tons of cargo per year,¹ the Dutch inland waterways are the busiest in Europe. The Rotterdam area registers up to 250,000 vessel movements annually. Ever more cargo is moved by inland waterborne transport to alleviate road and rail congestion. A significant part of this cargo is dangerous goods.

To handle all this traffic, the Netherlands employs the Information and Tracking System for Inland Shipping (in Dutch, abbreviated as IVS90). Here we discuss IVS90 and the software that keeps it going.

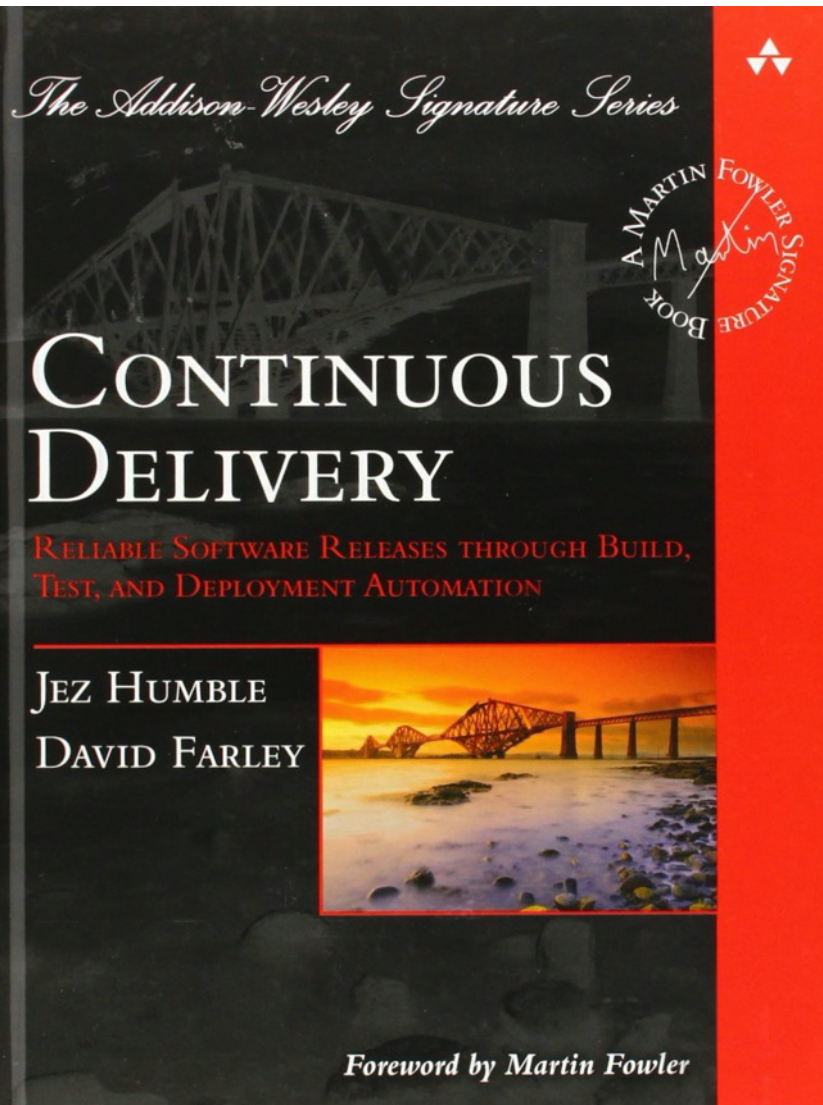
Rijkswaterstaat

Rijkswaterstaat manages and develops the main national infrastructure facilities on behalf of the Dutch Minister for Infrastructure and the Environment. Part of the infrastructure is the inland waterways: 3,462 km of canals and rivers with 84 locks and 1,092 bridges (176 of which are moveable).

A ship must report its vessel details, destination, and cargo when it passes through an area under vessel traffic management or when it wants to pass through a lock. This data is stored for safety and statistical reasons. Safety regulations state that passenger vessels and certain types of dangerous cargo can't be in the same lock chamber. To prevent ships from repeatedly having to declare their cargo and destination when passing successive locks, a system stores this data in a central place. This also enables fast retrieval of dangerous-cargo information in emergencies.

IVS90

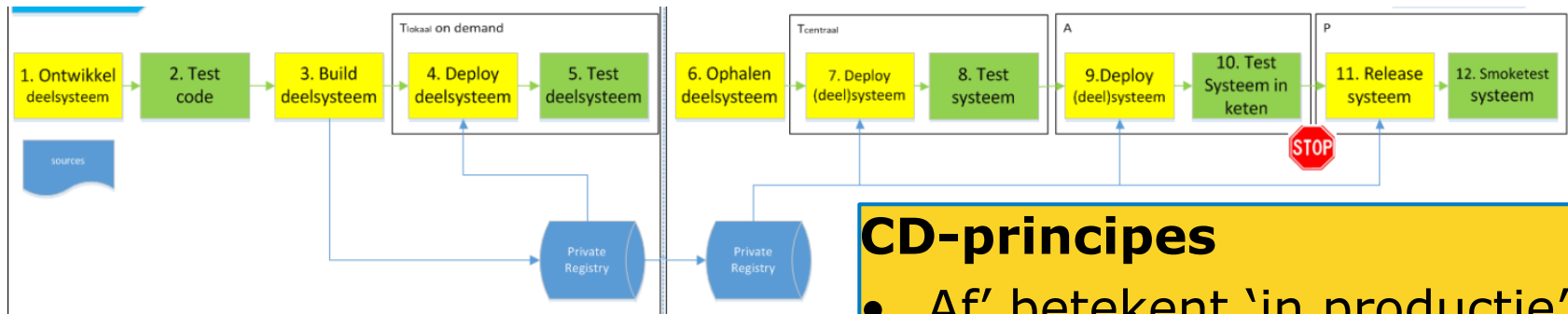
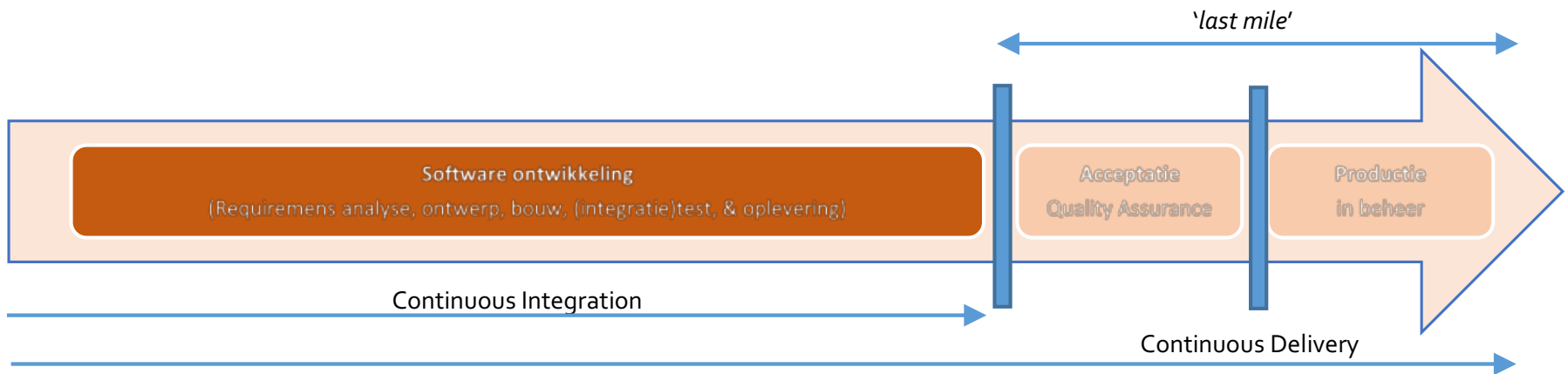
Around 1994, IVS90 became operational in one of the busiest areas of inland shipping in the Netherlands. Currently, the system is used nationwide and is connected to the systems of neighboring countries and the larger harbor authorities such as Rotterdam and Amsterdam. Because the large locks, bridges, and vessel traffic service (VTS) centers operate



“Continuous Delivery is het **vermogen** om **veilig** op **elk moment** softwareaanpassingen op de **productie-omgeving** te kunnen releasen en daarmee te ontsluiten voor de **eindgebruiker.**”

(Fitzgerald & Stol, 2014; Humble & Farley, 2010; Humble, Molesky, & O'Reilly, 2015; Krusche & Alperowitz, 2014; Neely & Stolt, 2013).

Procesuitbreiding



CD-principes

- Af' betekent 'in productie'
- Herhaalbaar en betrouwbaar proces

Procesautomatisering



PERIODIC TABLE OF DEVOPS TOOLS (V2) (V1) EMBED DOWNLOAD ADD

1 Fm Gh Github	3 Os Gt Git	4 Pd Dm DBmaestro	11 Fm Bb Bitbucket	12 Os Lb Liquibase	19 Os Gl GitLab	20 En Rg Redgate	21 Os Mv Maven	22 Os Gr Gradle	23 Os At ANT	24 Os Fn FitNesse	25 Fr Se Selenium	26 Os Ga Gatling	27 Fr Dh Docker Hub	28 Os Jn Jenkins	29 Pd Ba Bamboo	30 Os Tr Travis CI	31 Pd Gd Deployment Manager	32 Os Sf SmartFrog	33 Os Cn Consul	34 Os Bc Bcfg2	35 Os Lxc Linux Containers	36 En Rs Rackspace	37 Os Sv Subversion	38 En Dt Datical	39 Os Gt GrunT	40 Os Gp Gulp	41 Os Br Broccoli	42 Fr Cu Cucumber	43 Os Cj Cucumbersjs	44 Fr Qu Qunit	45 Os Npm npm	46 Fm Cs Codeship	47 Pd Vs Visual Studio	48 Fm Cr CircleCI	49 Fr Cp Capistrano	50 Fr Ju JuJu	51 Os Rd Rundeck	52 Os Cf CFEngine	53 Fr Ds Swarm	54 Os Op OpenStack	55 En Hx Helix	56 En Dp Delphix	57 Fr Sb sbt	58 Os Mk Make	59 Os Ck CMake	60 Fr Jt JUnit	61 Fr Jm JMeter	62 Fr Tn TestNG	63 Os Ay Artifactory	64 Fm Tc TeamCity	65 Fm Sh Shippable	66 Os Cc CruiseControl	67 En Ry RapidDeploy	68 Fm Cy CodeDeploy	69 En Oc Octopus Deploy	70 En No CA Nollo	71 Os Kb Kubernetes	72 Fm Bx Bluemix	73 En Cw ISPW	74 En Id Idera	75 Os Msb MSBuild	76 Os Rk Rake	77 Fr Pk Packer	78 Os Mc Mocha	79 En Xltv XL TestView	80 Os Jm Jasmine	81 Os Nx Nexus	82 Os Co Continuum	83 Fm Ca Continus CI	84 Pd So Solano CI	85 En Xld XL Deploy	86 En EB ElasticBox	87 Fm Dp Deploybot	88 En Ud UrbanCode Deploy	89 Fr Fl Fleet	90 En Os OpenShift	2 Fm Aws Amazon Web Services	5 En Ch Chef	6 En Pu Puppet	7 Os An Ansible	8 En Sl Salt	9 Os Dk Docker	10 Pd Az Azure	13 Os Ot Otto	14 En Bl BladeLogic	15 Os Va Vagrant	16 Fr Tf Terraform	17 Os Rk rkt	18 En Gc Google Cloud Platform
----------------------	-------------------	-------------------------	--------------------------	--------------------------	-----------------------	------------------------	----------------------	-----------------------	--------------------	-------------------------	-------------------------	------------------------	---------------------------	------------------------	-----------------------	--------------------------	-----------------------------------	--------------------------	-----------------------	----------------------	----------------------------------	--------------------------	---------------------------	------------------------	----------------------	---------------------	-------------------------	-------------------------	----------------------------	----------------------	---------------------	-------------------------	------------------------------	-------------------------	---------------------------	---------------------	------------------------	-------------------------	----------------------	--------------------------	----------------------	------------------------	--------------------	---------------------	----------------------	----------------------	-----------------------	-----------------------	----------------------------	-------------------------	--------------------------	------------------------------	----------------------------	---------------------------	-------------------------------	-------------------------	---------------------------	------------------------	---------------------	----------------------	-------------------------	---------------------	-----------------------	----------------------	------------------------------	------------------------	----------------------	--------------------------	----------------------------	--------------------------	---------------------------	---------------------------	--------------------------	---------------------------------	----------------------	--------------------------	------------------------------------	--------------------	----------------------	-----------------------	--------------------	----------------------	----------------------	---------------------	---------------------------	------------------------	--------------------------	--------------------	--------------------------------------

Os Open Source Fr Free Fm Freemium Pd Paid En Enterprise
SCM CI Deployment Cloud / Iaas / Paas BI / Monitoring
Database Mgmt. Repo Mgmt Config / Provisioning Release Mgmt Logging
Build Testing Containerization Collaboration Security

91 En Xlr XL Release	92 En Ur UrbanCode Release	93 En Bm BMC Release Process	94 En Hp HP Codar	95 En Au Automic	96 En Pl Plutora Release	97 En Sr Serena Release	98 Pd Tfs Team Foundation	99 Fm Tr Trello	100 Pd Jr Jira	101 Fm Rf HipChat	102 Fm Sl Slack	103 Fm D	104 Pd P	105 En E
106 Os Ki Kibana	107 Fm Nr New Relic	108 Os Ni Nagios	109 Os Zb Zabbix	110 En Dd Datadog	111 Os El Elasticsearch	112 En Ss StackState	113 En Sp Splunk	114 Fm Le Logentries	115 Fm Sl Sumo Logic	116 Os Ls Logstash	117 Os Gr Graylog			

CD-principes

- Automatiser alles
- Bouw kwaliteit in



Vertrouwen!

CD-principes

- Iedereen is verantwoordelijk
- Doe wat moeilijk is eerst
- Als het pijn doet, doe het vaker
- Continue verbetering



Effecten CD op softwareontwikkeltrajecten

SCOPE

- Betere aansluiting bij klanten en eindgebruikers door:
 - Kort cyclische feedback en sturing
 - Gebruik van real user data in een representatieve omgeving

KWALITEIT

- Minder refactoring
- Minder bugs
- Meer betrouwbare releases
- Softwarefouten in productie sneller hersteld

TIJD

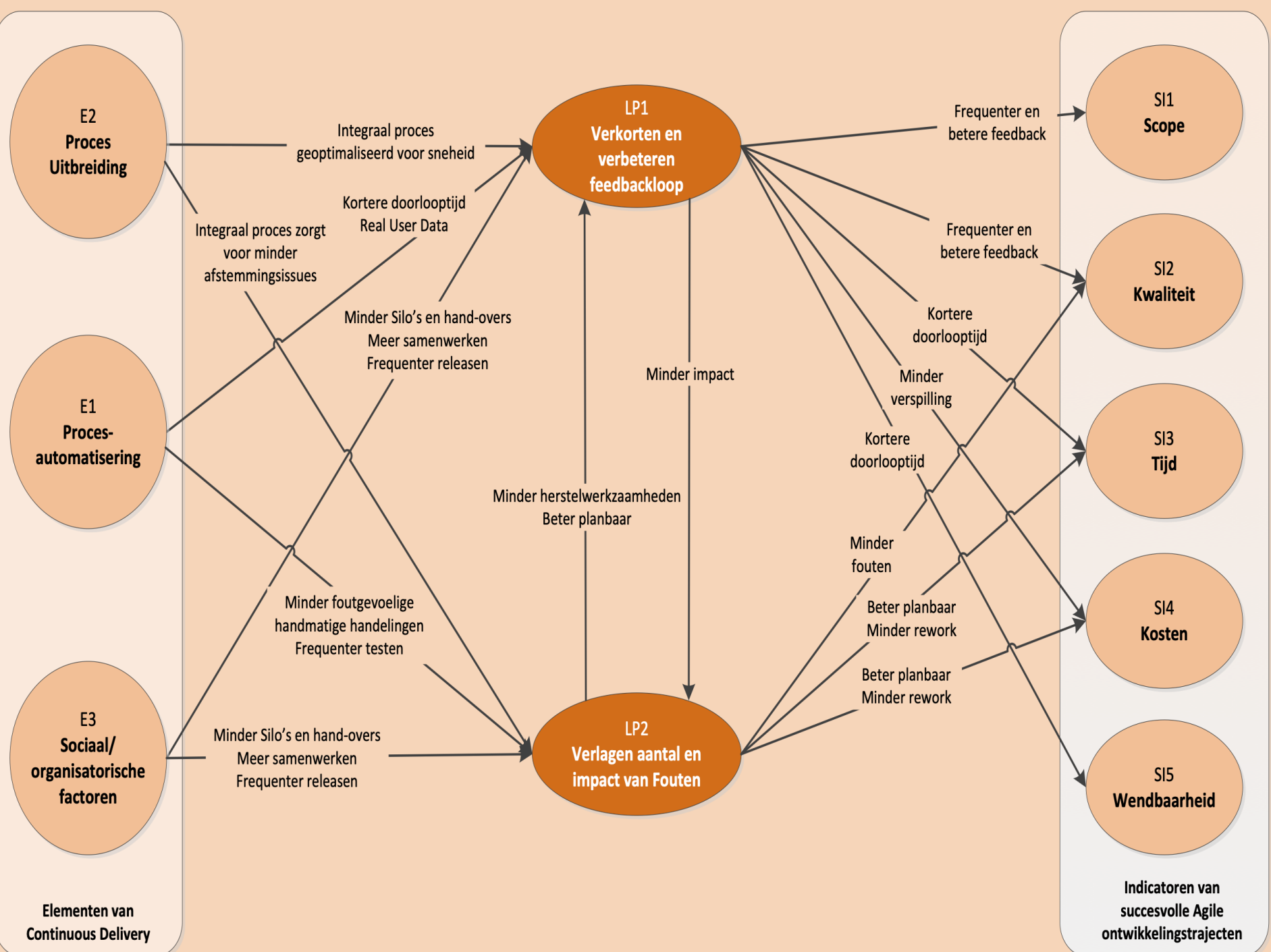
- Hogere ontwikkelsnelheid*
- Snellere time-to-market*

KOSTEN

- Lagere kosten in ontwikkel*- en beheerfasen

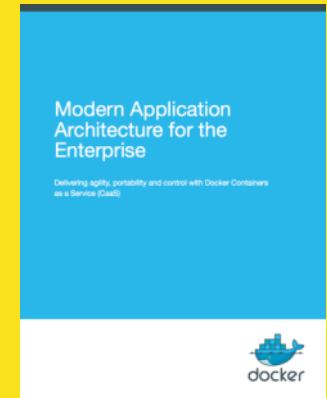
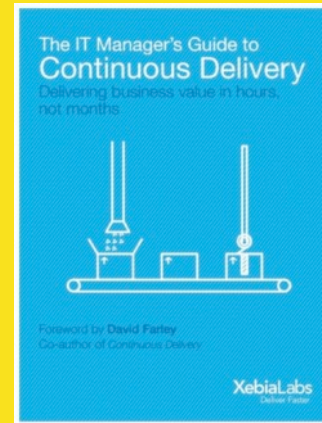
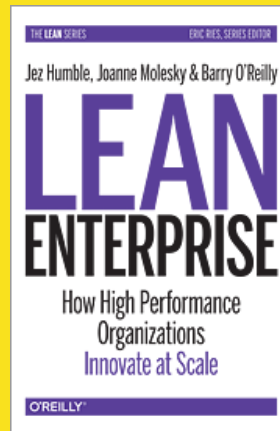
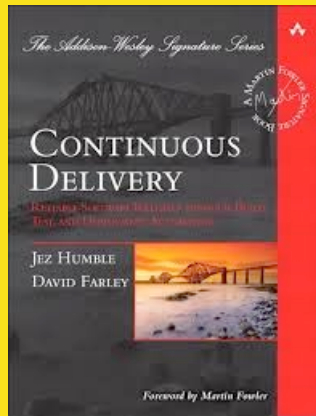
WENDBAARHEID

- Aanpassingen sneller ontsloten aan eindgebruiker





Inspiratie



<https://thesis.eur.nl/pub/41263/A.R.-Baronner.-scriptie.pdf>

<http://go.xebialabs.com/rs/xebialabs/images/Continuous-Delivery.pdf>

https://www.docker.com/sites/default/files/WP_ModernAppArchitecture_04.12.2016_1.pdf



Rijkswaterstaat
Ministerie van Infrastructuur en Waterstaat

Continuous Delivery in de praktijk

Software ontwikkeling in het scheepvaart management domein.

Allard Baronner & Wijnand Top

Meetup 12 februari 2020



Continuous Delivery in de RWS-praktijk



Onze applicatie(s)



Continuous Delivery met leveranciers



Agile software ontwikkeling binnen RWS



Continuous Integration & Testing



Uitdagingen



Ambities

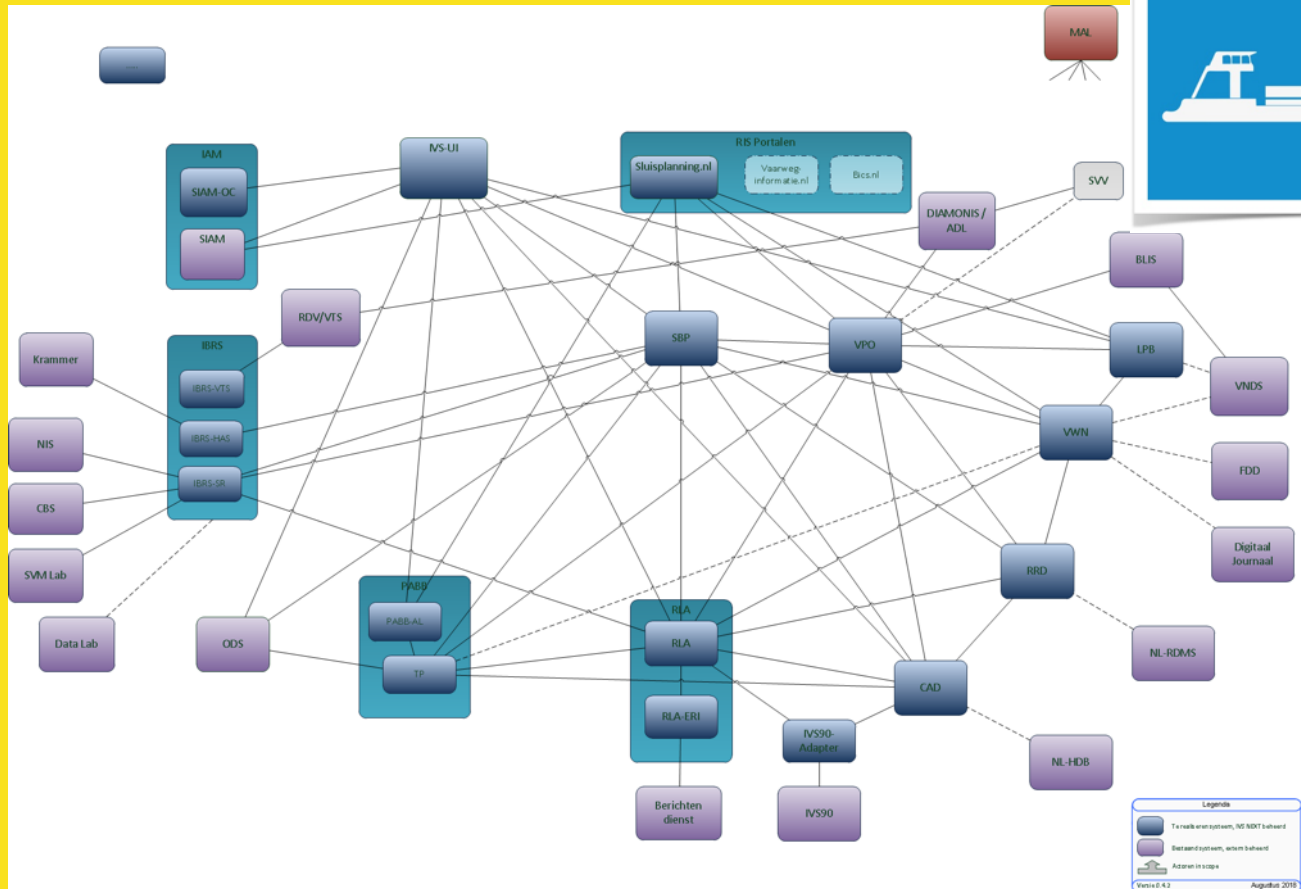


RWS als IT-leverancier

Video verwijderd i.v.m. bestandsgrootte



Introductie IVS-Next





Componenten vanuit verschillende bronnen

Ontwikkeling



Delivery

Leverancier ICTU



RWS CD team



Leverancier B

Team orange

Team green

Team Neon



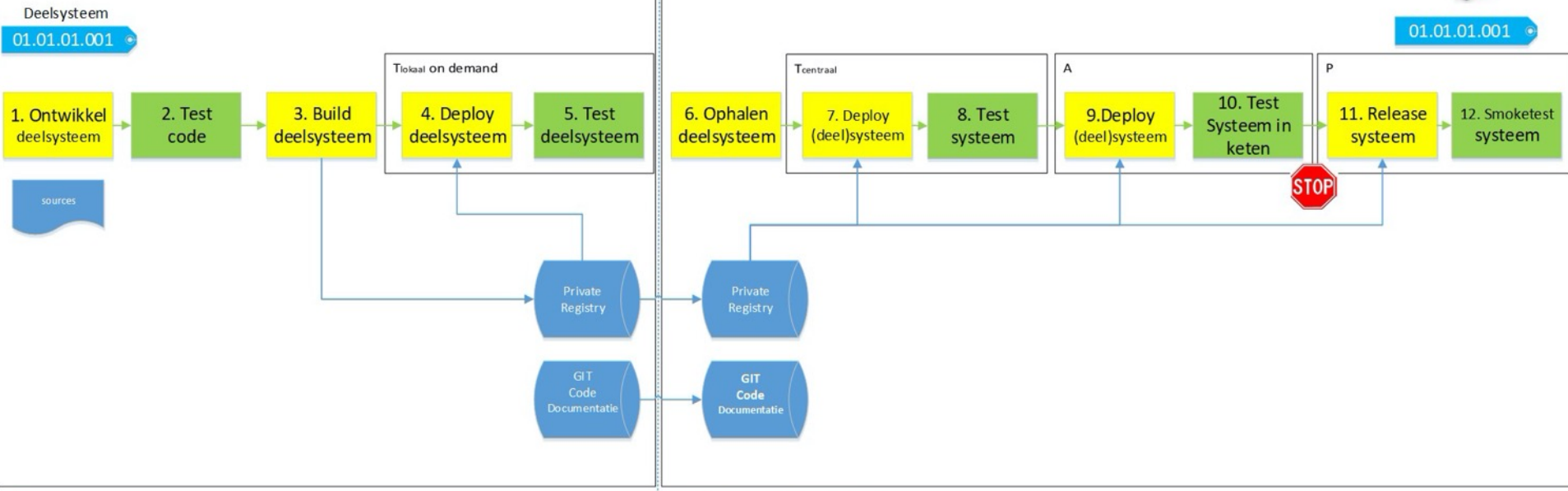
Orchestratie

Continuous Integration (CI) en Continuous Testing (CT)

Continuous Delivery (CD) en Continuous Testing (CT)



01.01.01.001



Monitoring, Alerting & Logging

Informatie & Audittrail



Overdracht CI naar CD

```
{
- stub-bdn: {
  - artifacts: {
    releaseZip: "https://.../infoboard/artifact?component=stub-bdn&version=1.62.0.200114031519&project=91",
    dockerCompose: "https://.../infoboard/artifact/docker-compose.yml?component=stub-bdn&version=1.62.0.200114031519&project=91",
    dockerComposeOverride: "https://.../infoboard/artifact/docker-compose.override.yml?component=stub-bdn&version=1.62.0.200114031519&project=91",
  },
  version: "1.62.0.200114031519",
  age: 2363386,
  name: "stub-bdn",
  project: "91",
  humanized: "a month"
}
}

```

[+ - View s](#)



Jenkins





Releases



ELKE SPRINT EEN
RELEASE



EXPERT GROEP



AUTOMATISCHE
KETENTEST



Agile software ontwikkeling binnen RWS

- Locatie
- Mensen
- Tools



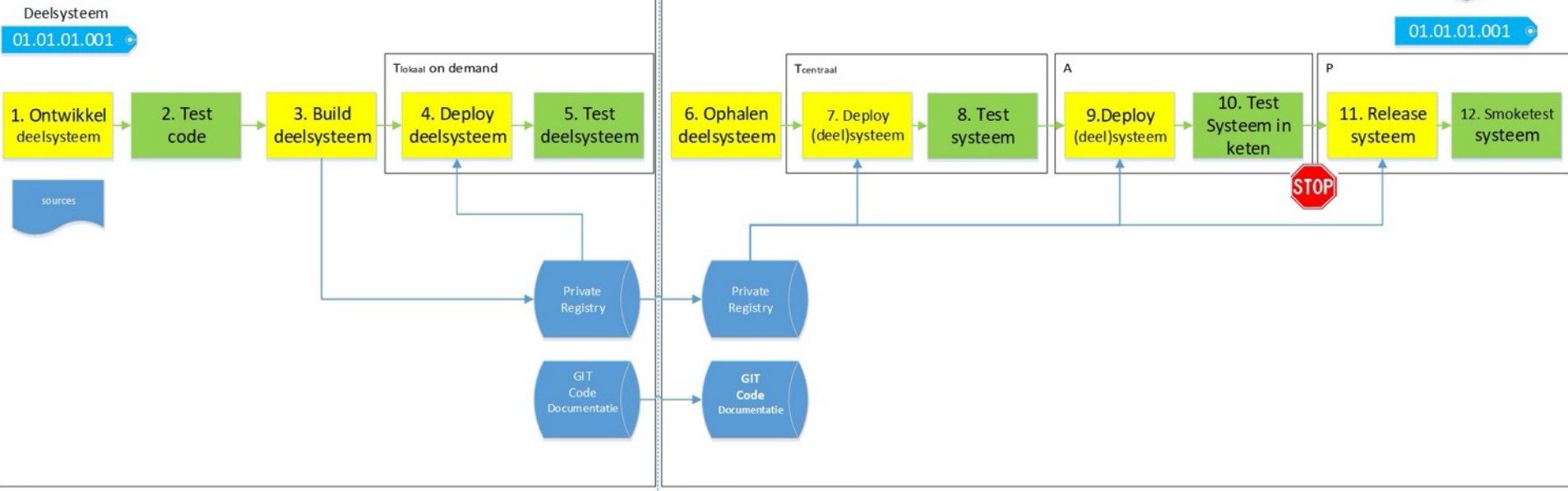
Orchestratie

Continuous Integration (CI) en Continuous Testing (CT)

Continuous Delivery (CD) en Continuous Testing (CT)



01.01.01.001



Monitoring, Alerting & Logging

Informatie & Audittrail



Continuous Integration (CI straat)

Het automatisch;

1. samenvoegen van code
2. bouwen van de applicatie
3. op verschillende niveaus testen

dit met als resultaat werkende distribueerbare software.

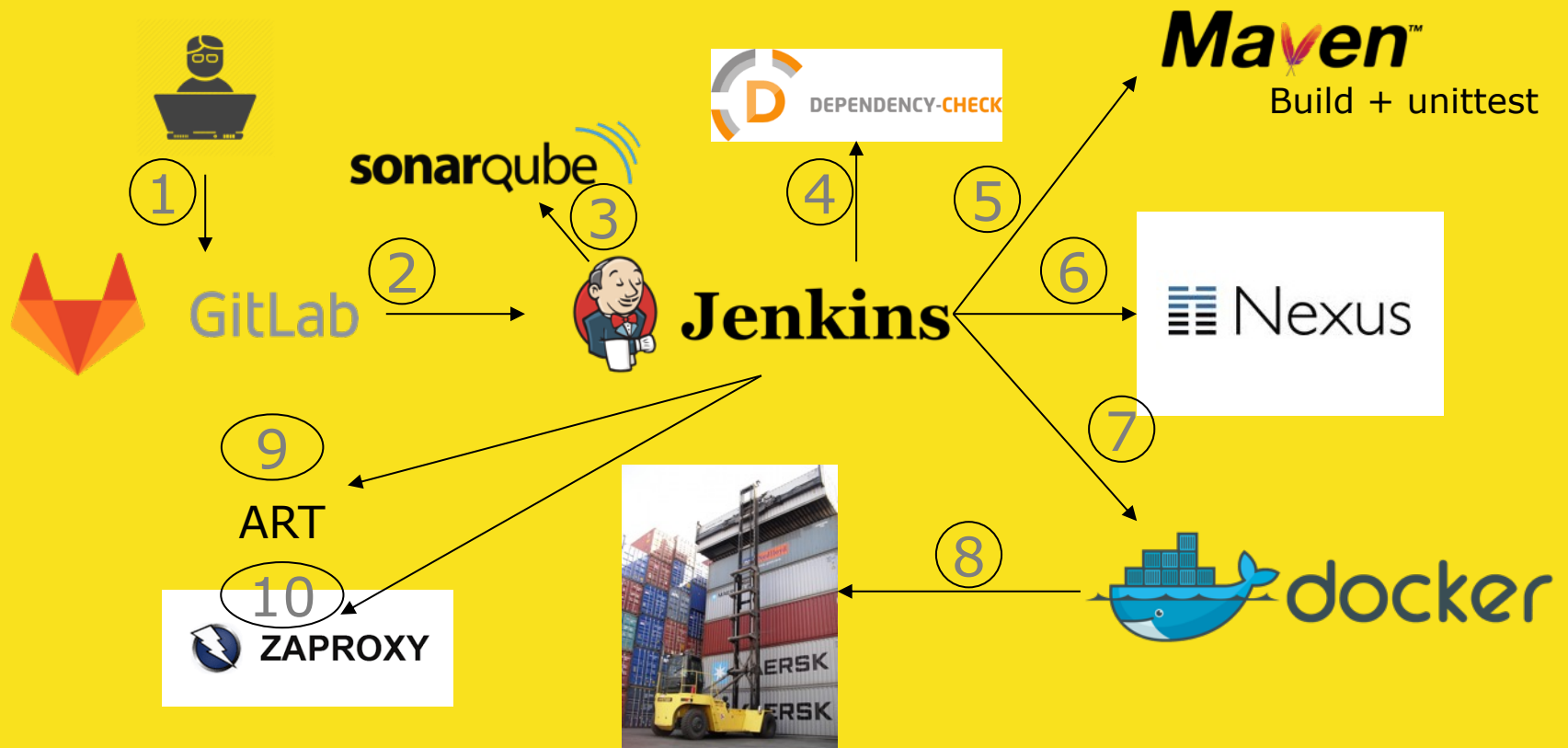


Wat gebeurt er in de CI pipeline

Stap	Tools	Actie
1. Ontwikkel deelsysteem	GitLab	Opslaan, versioneren en reviewen van code.
2. Test & scan code <ul style="list-style-type: none">• static code analysis• Dependency scans	SonarQube Owasp	Controle: kwaliteit code gebruikte libraries.
3. Build deelsysteem	Jenkins, Maven, Docker	Bouwen van uitvoerbare applicaties.
4. Deploy deelsysteem	Docker, Portainer	Opstarten van applicatie om testen uit te kunnen voeren.
5. Test deelsysteem <ul style="list-style-type: none">• Dynamic Application Security Test (PEN test)• Functional & regression	ZAP Testx, RobotFramework (ART)	Scan op kwetsbaarheden. Controle van functionaliteit.

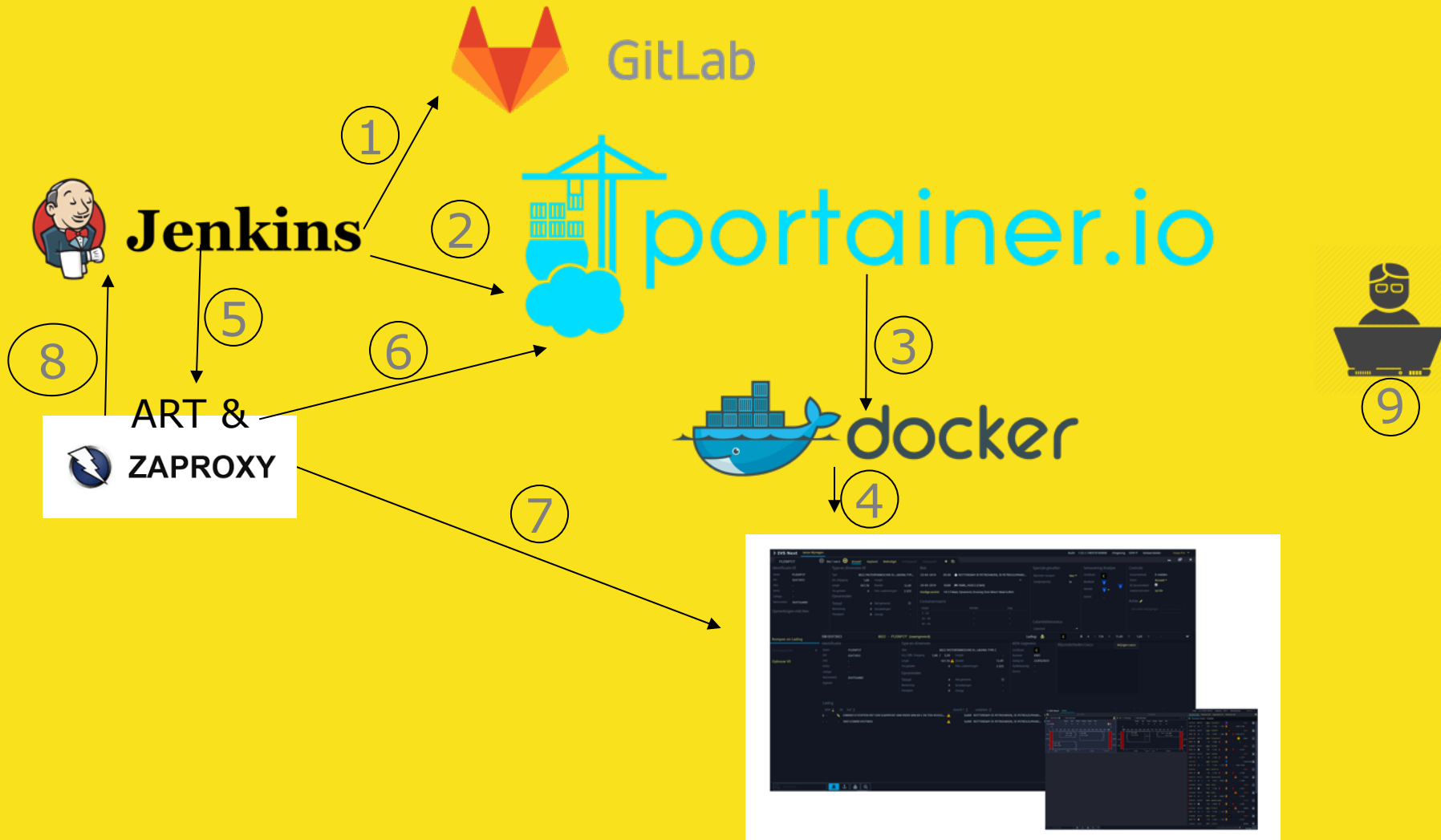


De Continuous Integration straat





Automatische regressietest (ART)





Kwaliteitsdashboard

Kwaliteitsrapportage ICTU/IVS Next

Metrieken ▾

Filter metrieken...



Toon ▾

Trend

Help

Rapportage van 11-2-2020 11:10

Algemeen

Componenten (master branch)

IVS GUI (master branch)



VPO (master branch)



RRD (master branch)



IBRS Krammer (master branch)



VWN (master branch)



ERI (master branch)



CAD (master branch)



IBRS Standard Record (master branch)



SBPN (master branch)



RLAN (master branch)



RLA-CRR (master branch)



RLA-RPN (master branch)



BDN (master branch)



LPB (master branch)



RLA Messages (master branch)



Id

Trend
(7 dagen)

Status

Meting

Norm

Comment



ABLM-1



214 van de 214 unittesten slagen. [\[SonarQube\]](#)

Alle unittesten slagen.



ABLM-2



Het meest recente unittest rapport van ABL (master branch) is 0 dagen oud. [\[SonarQube\]](#)

De leeftijd van het unittest rapport is maximaal 3 dagen oud. Meer dan 7 dagen is rood.



ABLM-3



ABL (master branch) unittest line coverage is 99%. [\[SonarQube\]](#)

Minimaal 90% van de regels code wordt gedekt door unittests. Lager dan 80% is rood.



ABLM-4



ABL (master branch) unittest branch coverage is 86%. [\[SonarQube\]](#)

Minimaal 80% van de code branches wordt gedekt door unittests. Lager dan 60% is rood.



ABLM-5



Alle 247 regressietesten van ABL (master branch) slagen en geen van de regressietesten is overgeslagen. [\[Jenkins testreport\]](#)

Alle regressietesten slagen.

Dit is een combinatie van unit-, integratie- en automatische regressietesten.



ABLM-6



De regressietest van ABL (master branch) is vandaag gedraaid. [\[Jenkins testreport\]](#)

De regressietest is maximaal 3 dagen geleden gedraaid. Langer dan 7 dagen geleden is rood.

Dit is een combinatie van unit-, integratie- en automatische regressietesten.



ABLM-7



ABL (master branch) ART statement coverage is 85%. [\[JaCoCo coverage rapport\]](#)

Minimaal 80% van de statements wordt gedekt door geautomatiseerde functionele tests. Minder dan 70% is rood.



Uitdagingen



Standaardisatiedruk



Verandering



Nieuwe tools en expertises



Ambities

Opschalen

- Meer applicaties intern ontwikkelen.
- Meer afdelingen ondersteunen.

Continuous Improvements

- Verplaatsen CI omgeving naar Azure (of andere cloud).
- Kubernetes voor ART omgevingen en OTAP.
- Simplify builds met bijvoorbeeld GitLab runners.
- Meer (opensource) (security) testen (zoals Clair).
- Kotlin



Behoudendheid

- Zo hebben we het altijd al gedaan
- Dat hebben we nooit eerder gedaan
- Daar zijn wij nog niet rijp voor
- Dat maakt ons overbodig
- Dat werkt niet in onze organisatie
- Sta je er garant voor dat het werkt?
- Waarom wijzigen, het werkt toch?
- Dat is in strijd met ons beleid
- Dat kost te veel
- Dat is niet onze verantwoordelijkheid
- Dat is een te grote verandering
- Daar hebben we geen tijd voor
- Het past niet in ons budget
- Roei nu maar met de riemen die je hebt
- Mag niet van Security/BIO
- Ja, maar...
- Schrijf maar een business case
- Laten we het maar in beraad houden
- We hebben het ook zonder kunnen rooien
- Is het al eens uitgeteerd?
- Wat we hebben is goed genoeg
- Dan moeten we de contracten openbreken
- Infra accepteert dat nooit
- Dan klopt het model/proces niet meer
- Het heeft onze aandacht
- Sinds wanneer ben jij een expert geworden?
- Dat is ons probleem niet
- Je bent zo enthousiast, jij houdt het hier niet lang vol
- Het management wil dit toch niet



Tools	
GitLab	https://about.gitlab.com/
Jenkins	https://jenkins.io/
SonarQube	https://www.sonarqube.org/
OWASP Dependency Scan	https://owasp.org/www-project-dependency-check/
Maven	https://maven.apache.org/
Nexus	https://www.sonatype.com/product-nexus-repository
OWASP Zap	https://owasp.org/www-project-zap/
Portainer	https://www.portainer.io/
Qualitytime	https://github.com/ICTU/quality-time
TestX	https://github.com/ICTU/
RobotFramework	https://robotframework.org/