

# DEVOPS - INTRO

G. Molines & S. Dahmoul

2018-2019



# INTRODUCTION

# Logistics

- Class starts at on time
- Cell phone switched off
- **One** conversation at a time
- Slack **#isa-devops** channel
- Deliveries: Github classroom

# Teaching Team

ISA



Anne-  
Marie  
**Dery**  
i3s



Laureen  
**Ginier**  
IBM



Philippe  
**Collet**  
i3s



Guilhem  
**Molines**  
IBM



Salah  
**Dahmoul**  
Orange

DevOps

# ISA + DevOps = a single module

Week	Friday Morning				Friday Afternoon			
	08:00 - 09:00	09:00 - 10:00	10:15 - 11:15	11:15 - 12:15	13:30 - 14:30	14:30 - 15:30	15:45 - 16:45	16:45 - 17:45
6	DevOps overview	TD DevOps on TCF			ISA overview		Project kick-off	
7	EJB, ORM		TD ISA on TCF		Test	TD Test	Project work	
8	Winter break							
9	View point	Project work			Artifactory	Distributed CI		
10	Interoperability & WS		Project work		Docker	Docker TD		
11	(buffer)	Project work			Multi-plan CI	Multi-plan CI		
12	Technical interview (Minimal & Viable Product)				Project work (unsupervised)			
13	Arch Dojo	Project work			DevOps Dojo	Project work		
14	Persistence	Project work			Compose	Project work		
15	Integration	Project work			Kubernetes	Project work		
16	Easter Break							
17	Project work (unsupervised)				Project work (unsupervised)			
18	Technical Interview (Almost-final Product)				Project work (unsupervised)			
19		Architecture Exam (3 hours)				DevOps Exam (3 hours)		
20								

- Shared case study and technological stack
- <https://github.com/collet/isa-devops/>



# DEVOPS ?

# Devs & Ops

Wall of  
Confusion

David is a DEVELOper !



David wants to  
maximize  
**change**

Peter is an OPERator !



Peter wants to  
optimize  
**stability**

# C.A.L.M.S. MODEL





Culture

Automation

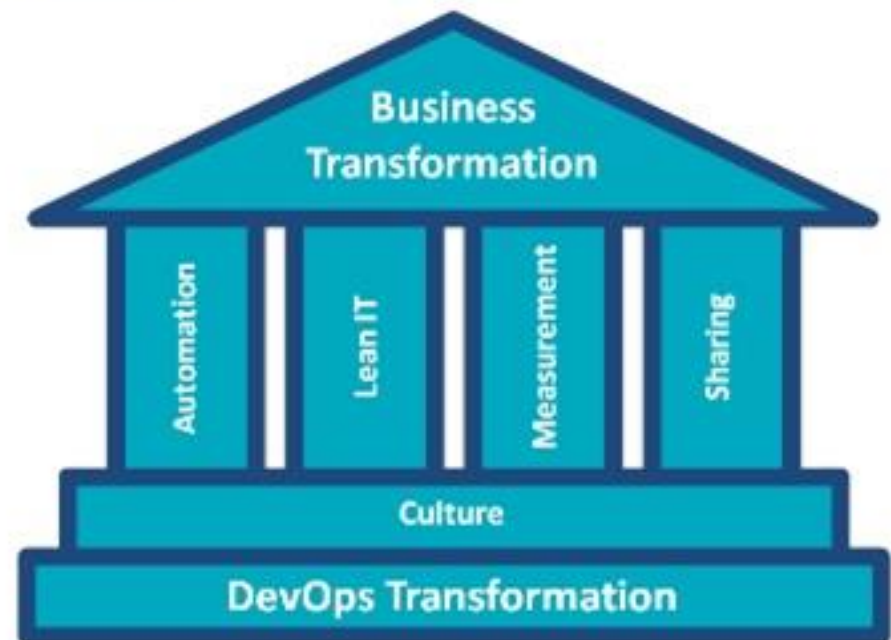
Lean

Measurement

Sharing

## The DevOps “calms” model

- Culture
- Automation
- Lean
- Measurement
- Sharing



# DELIVERING SOFTWARE

# Delivering software...

- In 1995, MS released Windows 95
  - Major breakthrough for consumers
  - Real multi-tasking with background apps
- How many service packs were released?


→ 2 (yes, two)

(6 months and 1 year after release)


# Delivering software...

- In 2015, MS released Windows 10
    - 4 editions
    - Continuous, forced updates
  - How many service packs were released?
- couldn't count...  
(1 GB on first day)





What is the most important thing you  
need to deliver software  
continuously?



# Trust

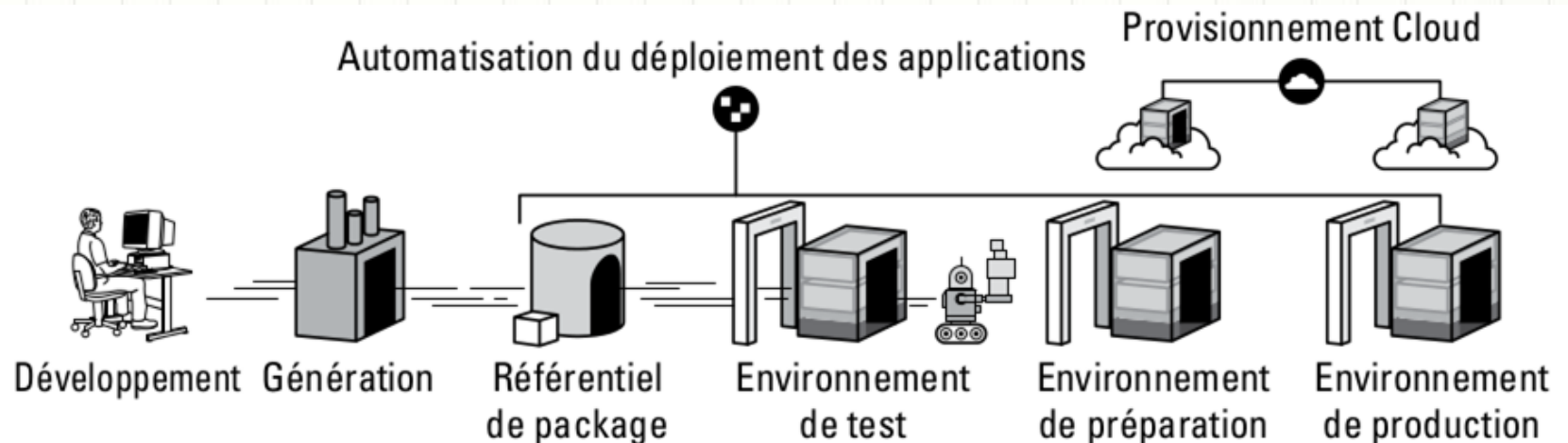


How do you get trust in your  
software delivery?

- Good architecture
- Development guidelines
- Project management
- Controlled processes
- Traceability of requirements
- Automation
- Repeatable **pipeline**
- Testing
- Testing
- And...

# Testing

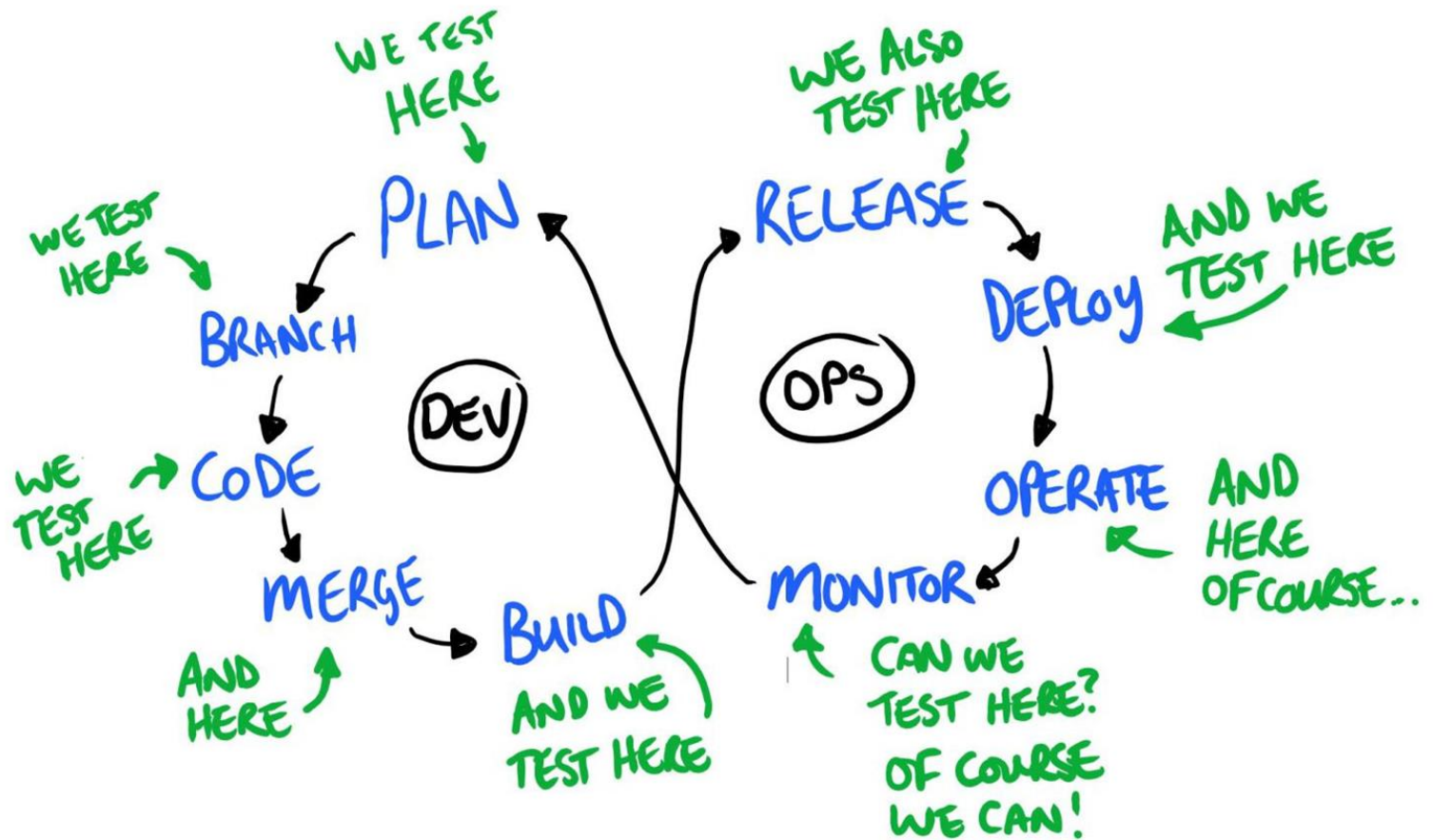
# How do you build a software delivery pipeline?

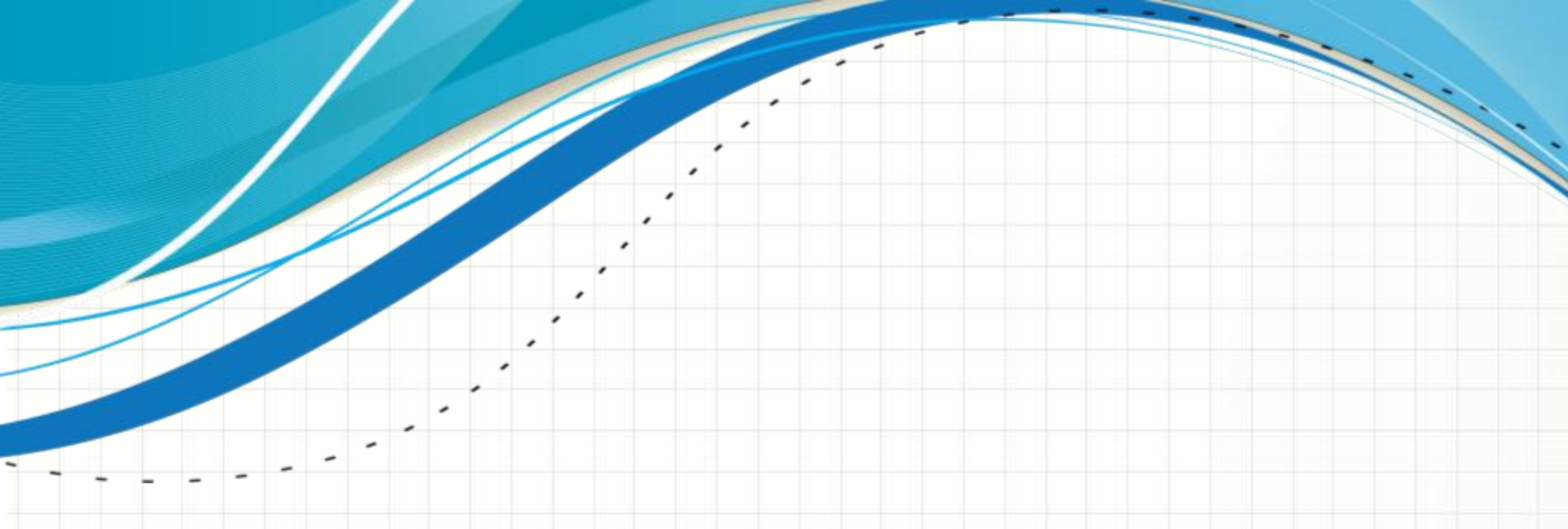






How do you write good tests?





# CONTINUOUS INTEGRATION

# Single student, simple project

- Maven?
- SCM?
- Code and click

# Group of students, simple project

- Ant / Maven
- Github
- Code and click
- Build on local machine
- Ticket management ?



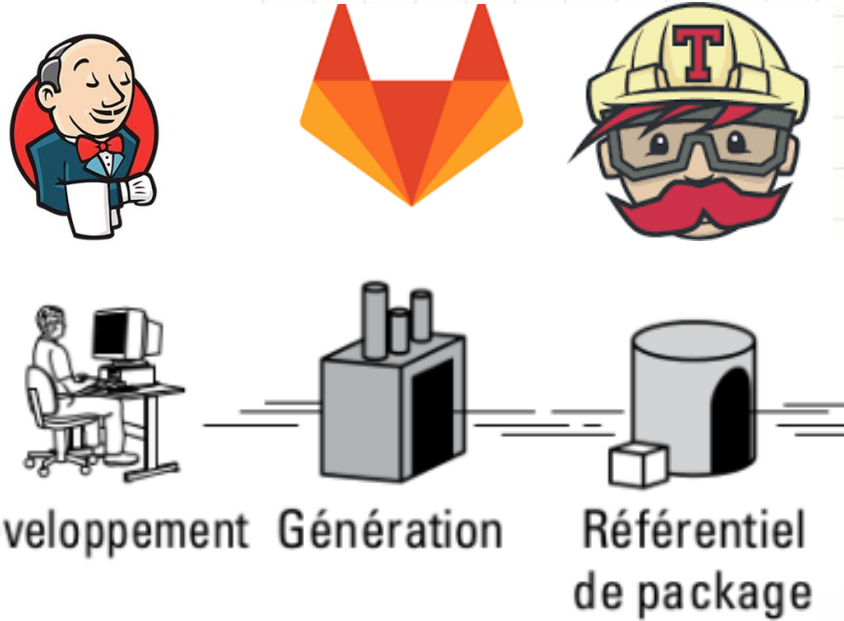
# Industrial complex project

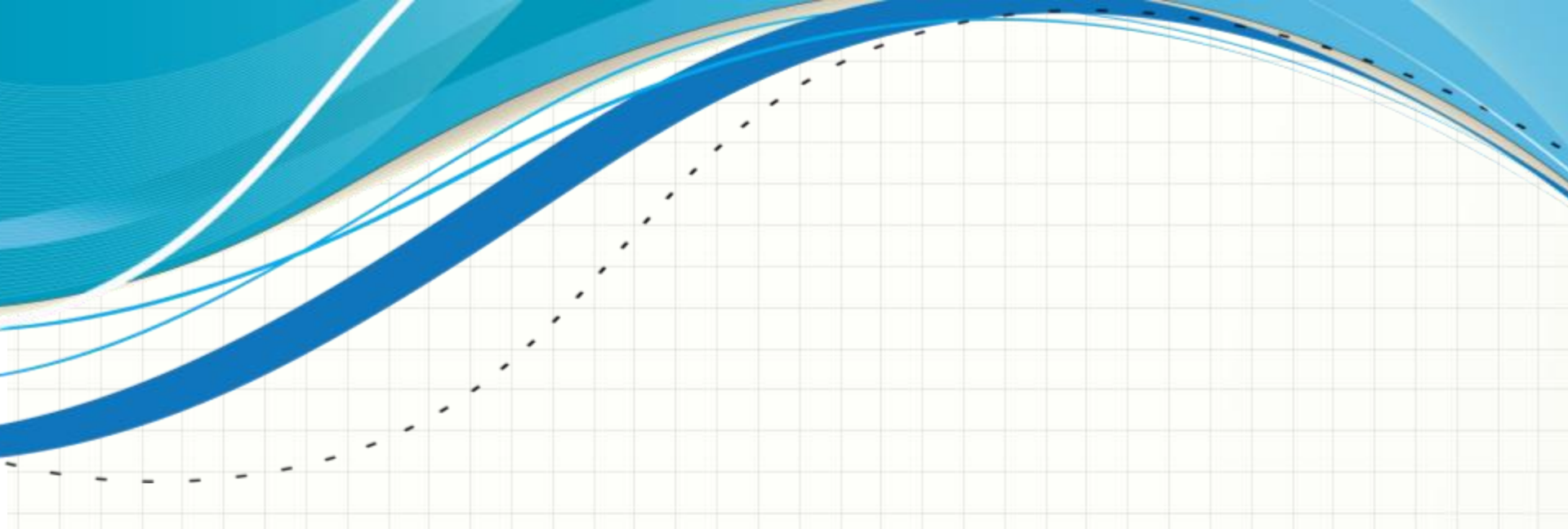
- 20 – 200 contributors
- One release / year
- One patch / month
- > 20M LOCs
- Deployment complexity
- → What do you need?

# Industrial complex project

- Componentization
- Independence of builds
- Each component tested
- Clear quality indicators
- Requirement traceability
- Fast builds
- Each contributor only builds what they code

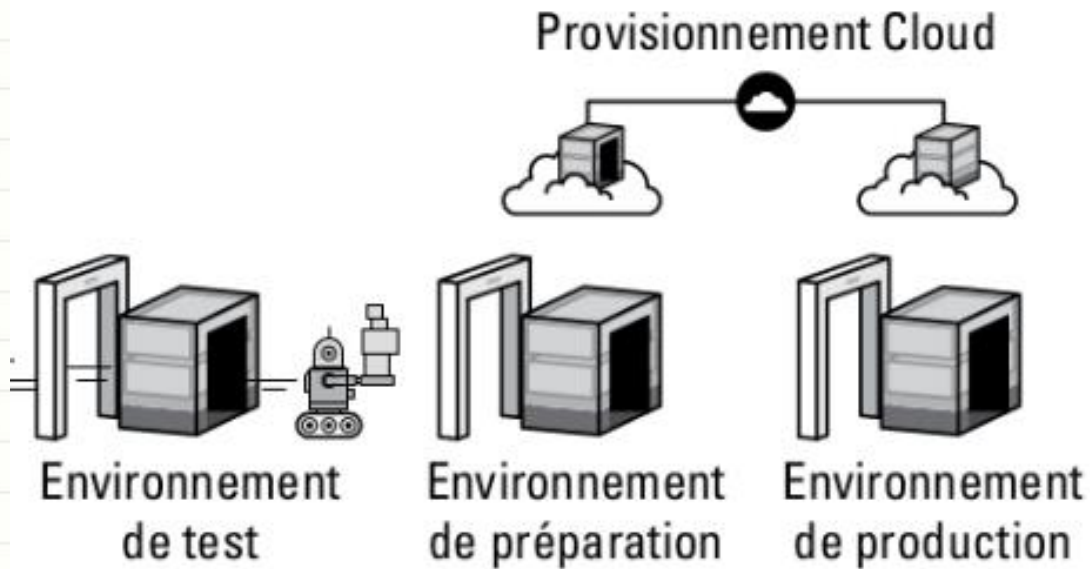
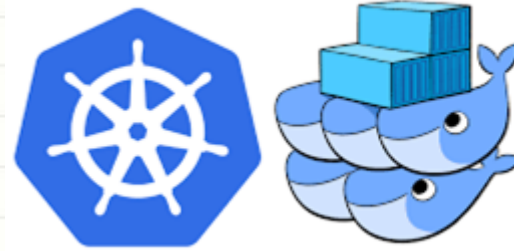
# Tools





# CONTINUOUS DELIVERY/DEPLOYMENT

# Tools







# Questions ?