

L'AUTOMATISATION DES TESTS DANS UNE ORGANISATION AGILE

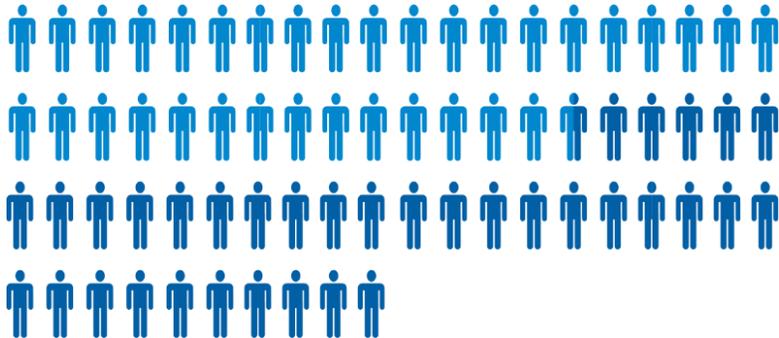


Michaël **GRANIER**
Stéphane **BATTEUX**



VSC TECHNOLOGIES EN CHIFFRES

EFFECTIFS



350 COLLABORATEURS
350 PARTENAIRES

35 ÉQUIPES de développement

1 PRA

ACTIF EN PERMANENCE
70% des internautes à Lille
30% à St Denis

+3500

SERVEURS

29

BILLETTS VENDUS / SECONDE
ODV 2015

1To

DE DONNÉES
TRAITÉES / JOUR

3 SITES : Lille, Paris
La Défense, Nantes



GROUPE VSC

80m voyages vendus en 2016

13m téléchargements de l'app

4,3 M€ volume d'affaire en 2016

12m Visiteurs uniques

par mois – 60% Mobile & Tablette

VSC TECHNOLOGIES A 2 MISSIONS



Apps
Mobiles



Sites
e-commerce



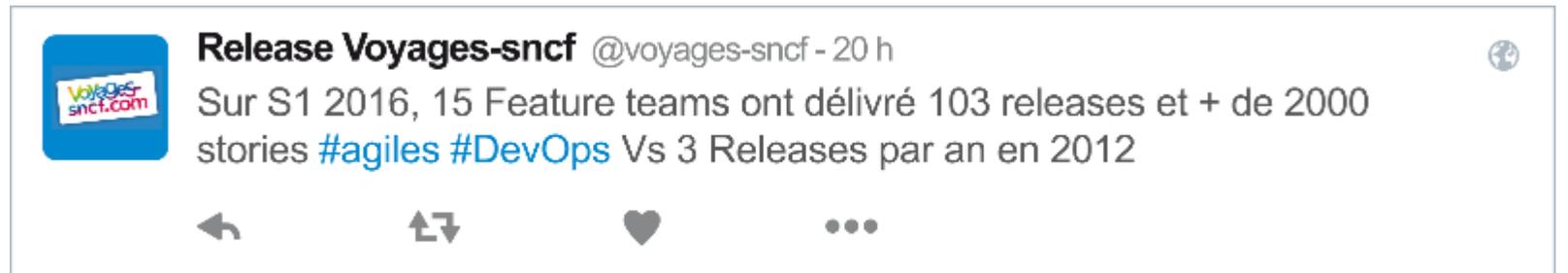
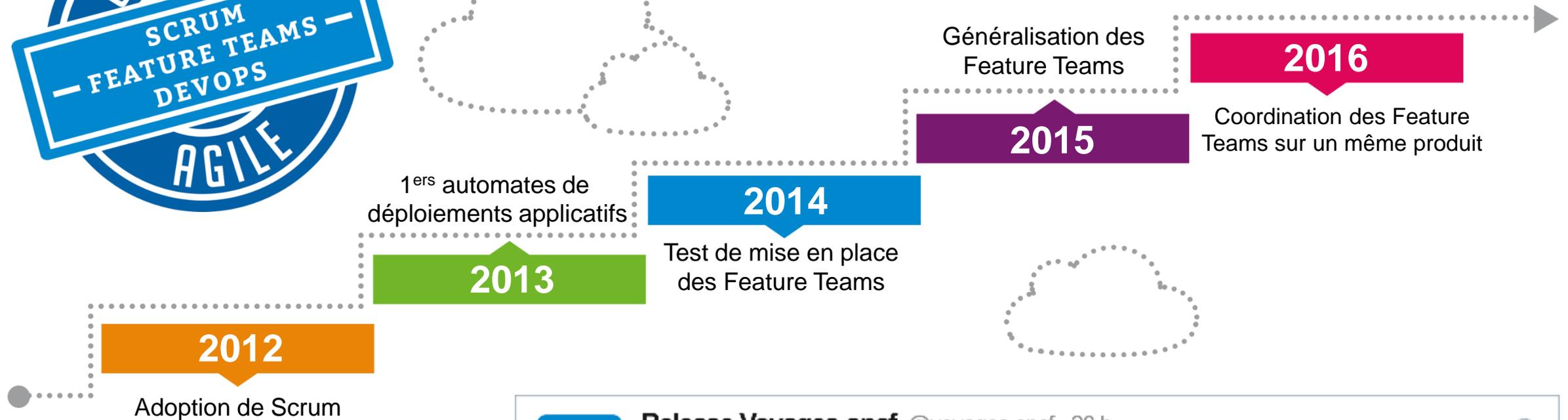
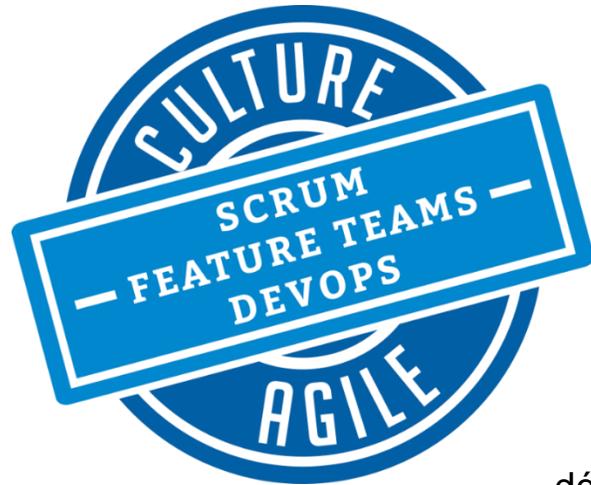
Sites
Web

Sites
e-commerce
Transport

Services

Apps
Mobiles

ÉVOLUTION DU CONTEXTE AGILE VSCT



LES ACTEURS DE L'AUTOMATISATION



SPÉCIFIER

DÉVELOPPER

VALIDER

EXÉCUTER

2012 - 2013

1 automaticien

Testeur

2014 - 2015

Crusaders

Testeur

UNE ÉQUIPE POUR L'AUTOMATISATION



Crusaders

4 Automaticiens
1 PO / Testeur



Notre Vision

Rendre les tests disponibles
à l'ensemble des acteurs

- *Humains et automates*
- *De l'usine à la production*



Notre Objectif

Fournir la brique « Test » au
chantier de l'automatisation
de la chaîne de Continuous
Delivery



Notre Rôle

Accompagner les Feature
Teams dans l'automatisation
des tests

Gérer l'aspect transverse des
Frameworks d'automatisation



FRAMEWORKS D'AUTOMATISATION WEB

Sélénium VSC

- Sélénium 2 (Webdriver)
- Environnement Java
- Partagé entre 9 Feature Teams
- Tests de bout en bout
- Exécutable sur Firefox, Chrome & PhantomJS
- Déploiement continu grâce à Jenkins



Protractor

- Utilisé par une Feature Team sur le site UK <https://uk.voyages-sncf.com/en>
- Environnement NodeJs & JavaScript
- Applicatif full AngularJs
- Compatible Full Responsive
- Déploiement continu grâce à Jenkins



FRAMEWORKS D'AUTOMATISATION MOBILE



Espresso - IHM

- Directement intégré au projet de l'application Android
- Lancement sur émulateur Genymotion & terminaux physiques
- Géré par la Feature Team Android avec automaticien dédié
- Déploiement quotidien de la nightly-build



iOS ?

- UITesting à l'étude dans la Feature Team
- Crainte sur le ROI

SoapUI – Web services

- Automatisation du backend mobile
- Factorisation du code en Building Blocks
- Scripting en Groovy



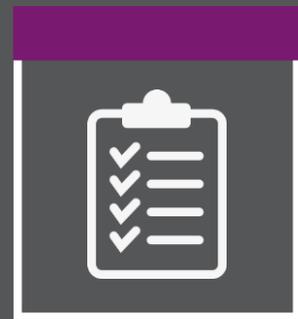
CARACTÉRISTIQUES DE NOS FRAMEWORKS



Rapport clair



DataDriven



Preuves de test

Tout environnement



Source control

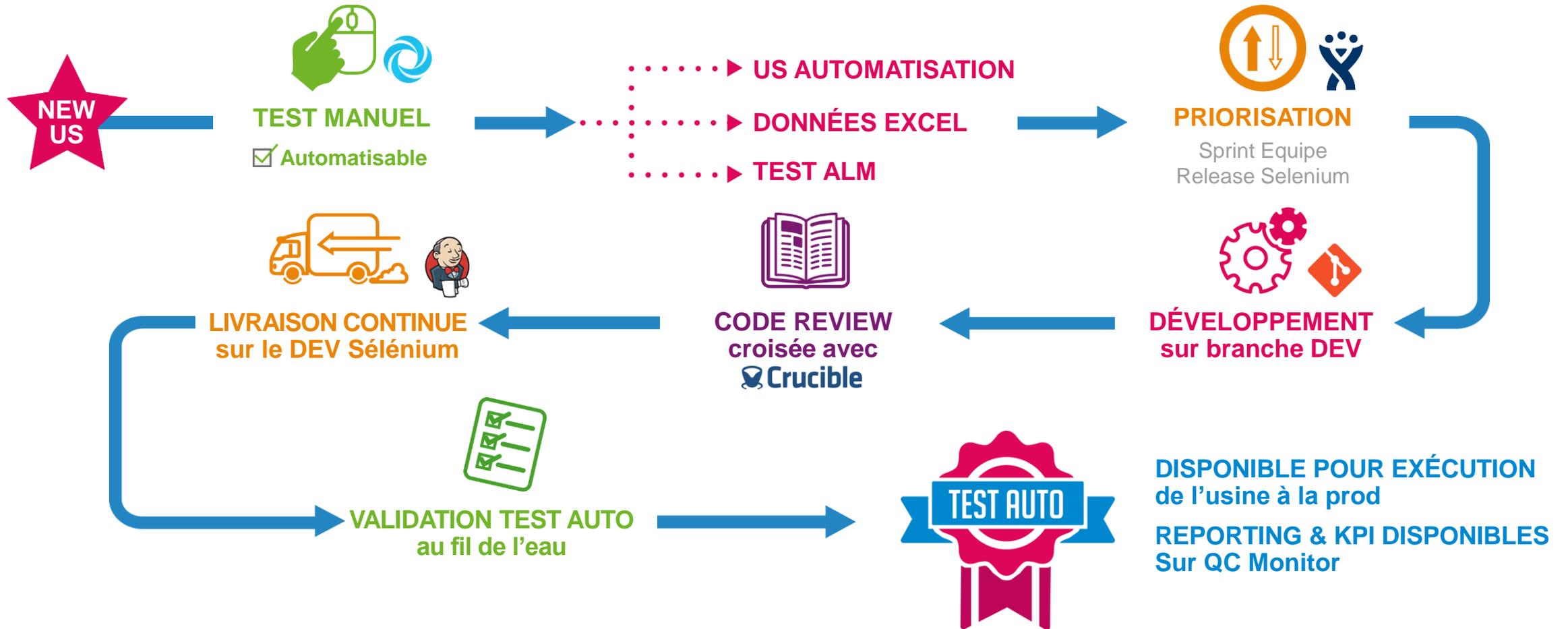
PROCESSUS DE COORDINATION

CRÉATION D'UN TEST AUTO



PROCESSUS DE COORDINATION

CRÉATION D'UN TEST AUTOMATISÉ



LES ACTEURS DE L'AUTOMATISATION



SPÉCIFIER

DÉVELOPPER

VALIDER

EXÉCUTER

2012 - 2013

1 automaticien

Testeur

2014 - 2015

Crusaders

Testeur

2016

Testeur

Testeur

Développeur

Testeur

Testeur

Crusaders – Formation / Code Review / Maintenance Framework / Gestion Livrables / Veille

LES DIFFICULTÉS RENCONTRÉES



COMMUNICATION



Silotage Feature Team : chacun se concentre uniquement sur son périmètre applicatif
Vision différente de l'automatisation entre les développeurs et testeurs

VALIDATION DES TESTS AUTO



Incompréhension du concept de la validation des tests auto
Mise à dispo tardive des tests développés

AUTOMATISATION DANS LES FEATURE TEAMS



Démarrage tardif
Peu priorisé par le Product Owner



FAIBLE ADHÉSION DES FEATURE TEAMS AUX FRAMEWORKS PRÉ-EXISTANTS

Crés dans les équipes tests sans consultation des besoins dévs
Crés avant l'organisation en Feature Team
Historique conséquent et choix techniques



PRÉSENCE / MAINTENANCE DU LEGACY RESTANT



LES LEÇONS TIRÉES



AMÉLIORER LA COMMUNICATION AUTOUR DE L'AUTOMATISATION

Sensibiliser les développeurs aux problématiques du test

Mettre des KPIs et les communiquer largement



ACCOMPAGNER LES FEATURE TEAMS À L'AUTOMATISATION

Objectifs d'équipe sur l'automatisation

Rex sur l'importance de l'anticipation des tests automatisés



ÉCOUTER LES DEMANDES DES FEATURE TEAMS

Pour faire évoluer l'existant
Pour démarrer une nouvelle solution



APPLIQUER LES BONNES PRATIQUES DE L'AGILITÉ POUR L'AUTOMATISATION

DSM
User Story d'automatisation des tests

Rétrospective
Déploiement continu
Tester au plus tôt



ARA

AUTOMATION REST API



Tests en local

Chaque testeur peut exécuter les tests sur sa machine



Tests parallélisés

Les tests lancés à distance sont répartis sur plusieurs robots



Tout framework

Les tests sont centralisés sur ARA

ARA - INTÉGRATION

HP ALM

- Exécute les tests automatisés à la demande des testeurs
- Les résultats et preuves de test sont archivés pour la postérité



Rundeck

- Exécute les tests auto pour valider le déploiement des plateformes



Jenkins

- Exécute les tests tous les jours
- Les tests IHM sont pris en compte pour la météo applicative

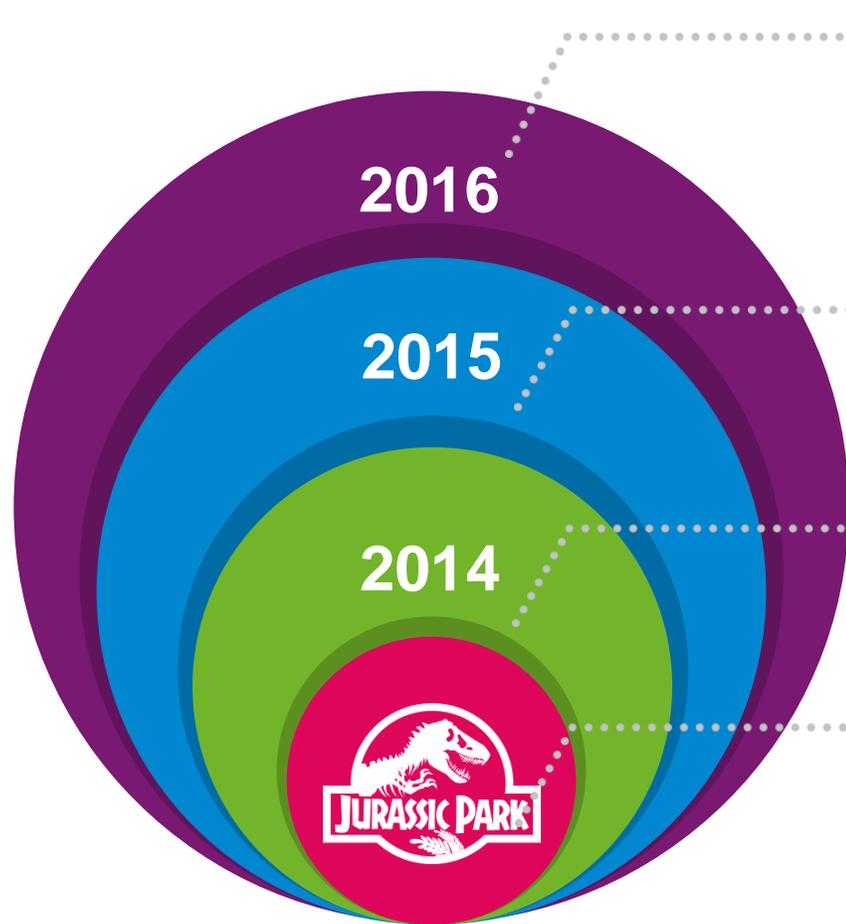


Elasticsearch + Kibana

- Facilite l'analyse de grandes quantités de tests



UTILISATION DES TESTS AUTOMATISÉS



+ 2016

- Lancement des tests automatisés au déploiement des plateformes
- Génération de données de test
- Exécution et résultats depuis Jenkins

+ 2015

- Exécutions quotidiennes analysées par les Feature Teams
- Validation des instances de prod

+ 2014

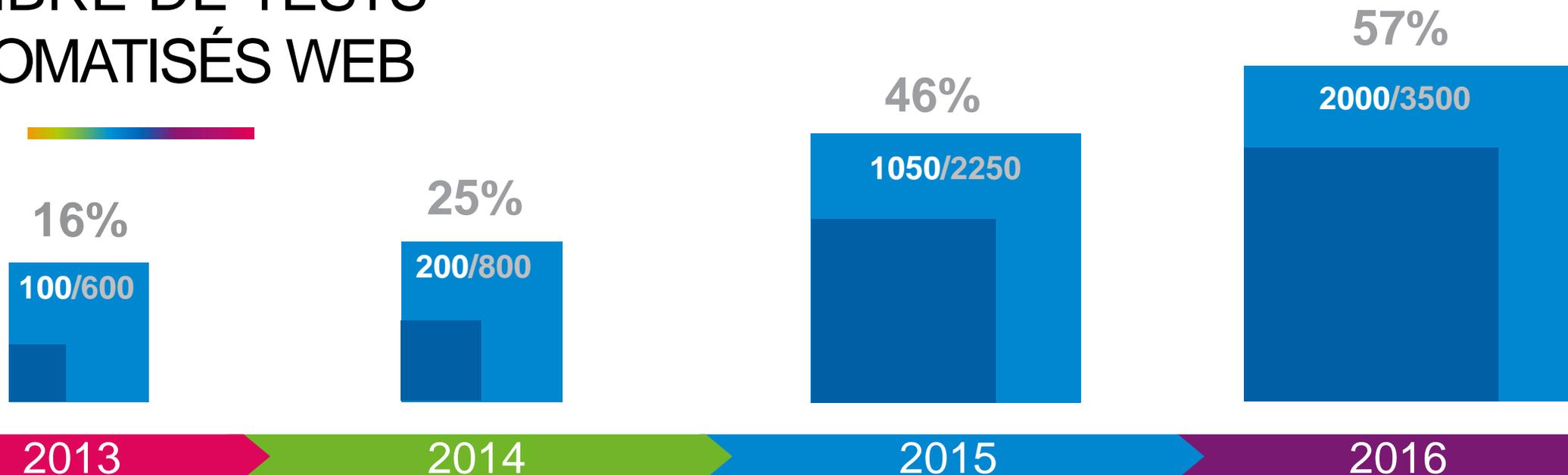
- Exécution et résultats des tests Sélénium depuis HP ALM

● Préhistoire

- Release : Quelques tests de NR avec QTP



EVOLUTION DU NOMBRE DE TESTS AUTOMATISÉS WEB



- Historique avec QTP
- Peu maintenu

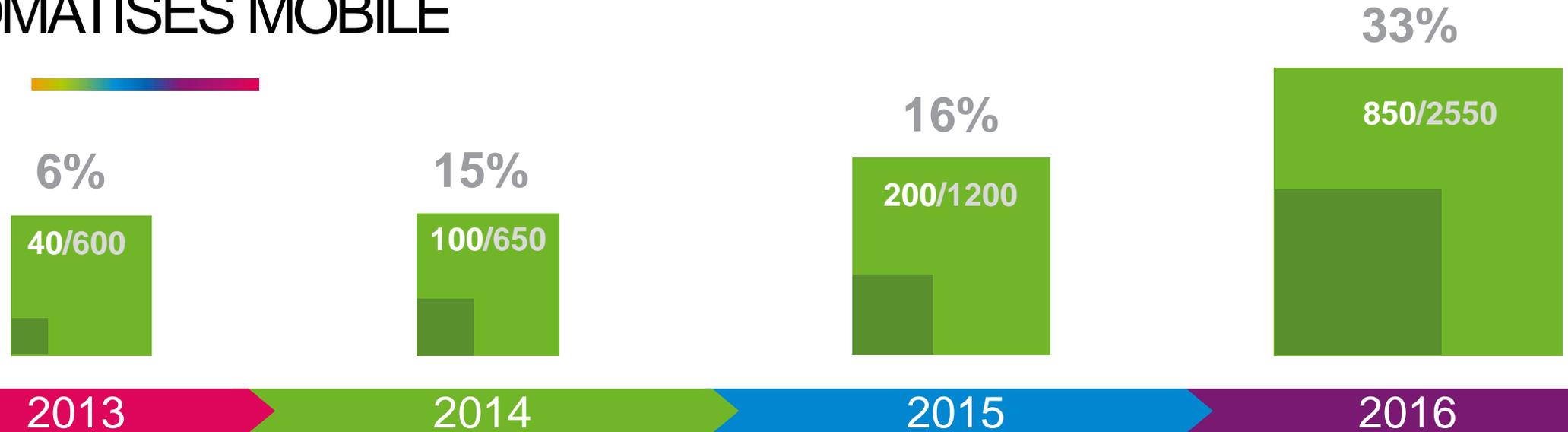
- Passage à Sélénium
- Création d'une équipe pour l'automatisation
- Début de la démarche DevOps

- Renforcement de l'équipe
- Objectif annuel sur le % d'automatisation
- Mise en place de KPI disponibles pour tous

- Intégration des Feature Teams dans l'automatisation
- Renfort d'automatiseurs dans les Feature Teams
- Sprint et intégration continue pour les tests développés



EVOLUTION DU NOMBRE DE TESTS AUTOMATISÉS MOBILE



- Quelques tests SoapUI
- En dehors de QC
- Peu maintenu

- Un testeur / automaticien sur les tests web services

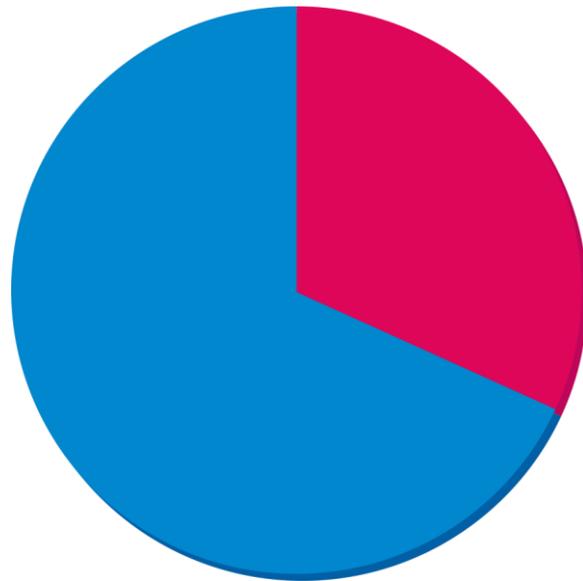
- Refonte du projet d'automatisation SoapUI
- Mise en place de la solution Espresso pour Android
- Mise en place de KPI disponibles pour tous

- Renfort automaticien dans la Feature Team Mobile

LES TESTS NON AUTOMATISABLES



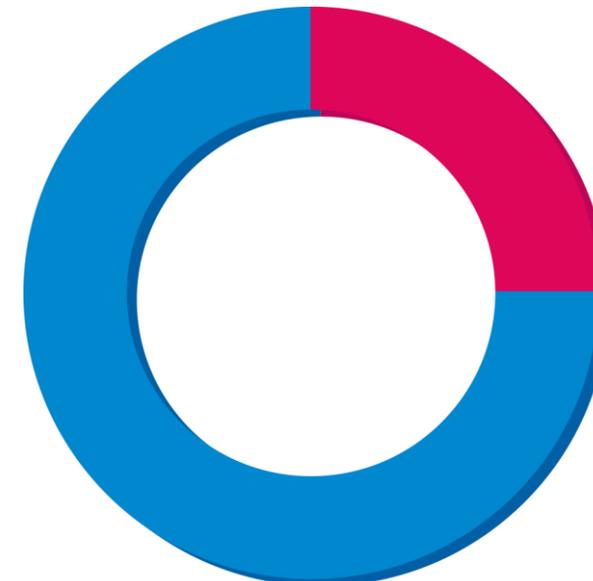
SUR WEB



SOIT **1600** TESTS



SUR MOBILE



SOIT **650** TESTS

TOUT N'EST PAS AUTOMATISABLE



En cours
de POC



Feature Flipping



Web responsive



A étudier



Multicanal



Web sur mobile



Accessibilité



Asynchrones



ROI très
discutable



Validation des emails



Vérification graphique

LES VICTOIRES DE L'AUTOMATISATION



VÉLOCITÉ

2

RÉACTIVITÉ

Détection des régressions
en amont

Sur l'exécution des tests
200 tests IHM en 30 minutes

Validation 1 ferme de production
3 jours▶ *0,5 jour*

Release du legacy
3 mois▶ *1 mois*

3

QUALITÉ

Plus grande valeur ajoutée des
testeurs sur la validation manuelle

Pour nous, l'automatisation permet de *conserver*
un niveau **OPTIMUM de qualité** dans un contexte agile et exigeant

PROBLÉMATIQUES 2017 – 2018

DÉSIRS

- 😊 Réduire le Time To Market
- 😊 Toujours plus de tests automatisés
- 😊 Exécuter et analyser tous les jours
- 😊 Garder la maîtrise sur le budget test

CONTRAINTES

- 😞 Tests automatisés parfois instables
- 😞 Il faut tester sur un environnement iso prod (et sans bouchon)
- 😞 Temps de maintenance important
- 😞 Temps d'analyse important

Comment faire évoluer et enrichir sereinement le patrimoine de test ?

UN SYSTÈME EXPERT DE TEST



ANALYSE AUTOMATIQUE

Cause du test en échec ?

Est-ce une anomalie applicative ?

Le problème est-il reproductible ?

Quelle équipe faut-il prévenir ?



INDICE DE CONFIANCE

Le livrable est-il acceptable ?

Le déploiement de la plateforme est-il ok ?

Peut-on ouvrir la plateforme aux testeurs ?

EXEMPLE



RÔLE

En tant que
testeur

SITUATION

Lorsque je rencontre
une erreur

« Une erreur de paiement est
survenue »

ACTION

Je regarde
si le même test

avec « Carte Paiement VISA »

fonctionne

CONCLUSION

Si oui cela
invaliderait les causes

« API Paiement KO »
« Configuration plateforme KO »

C'EST À VOUS

