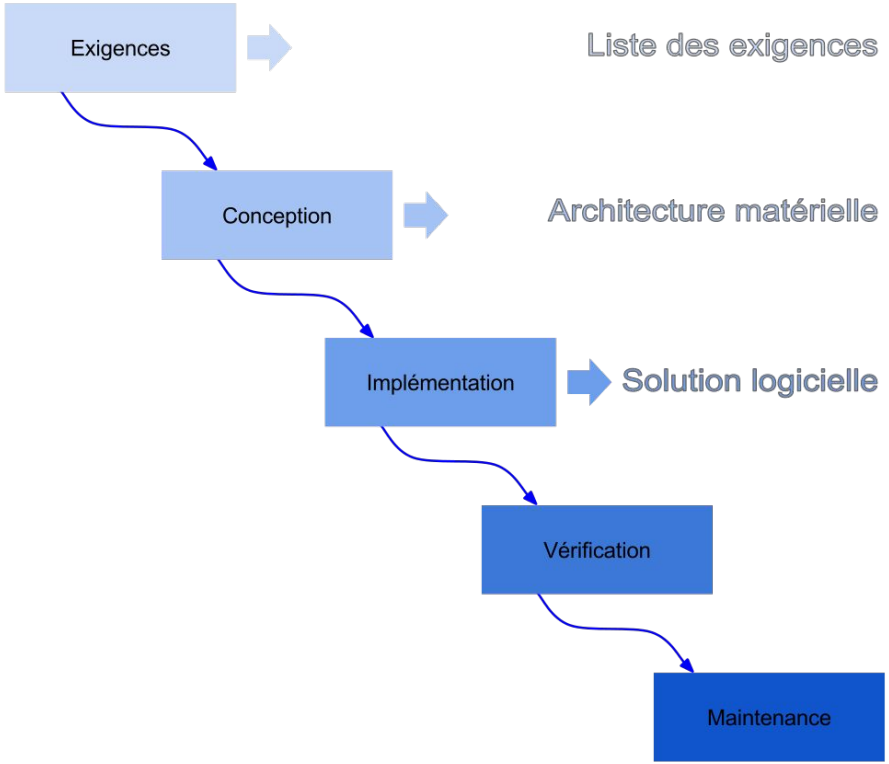
Sécurité Applicative

SDLC

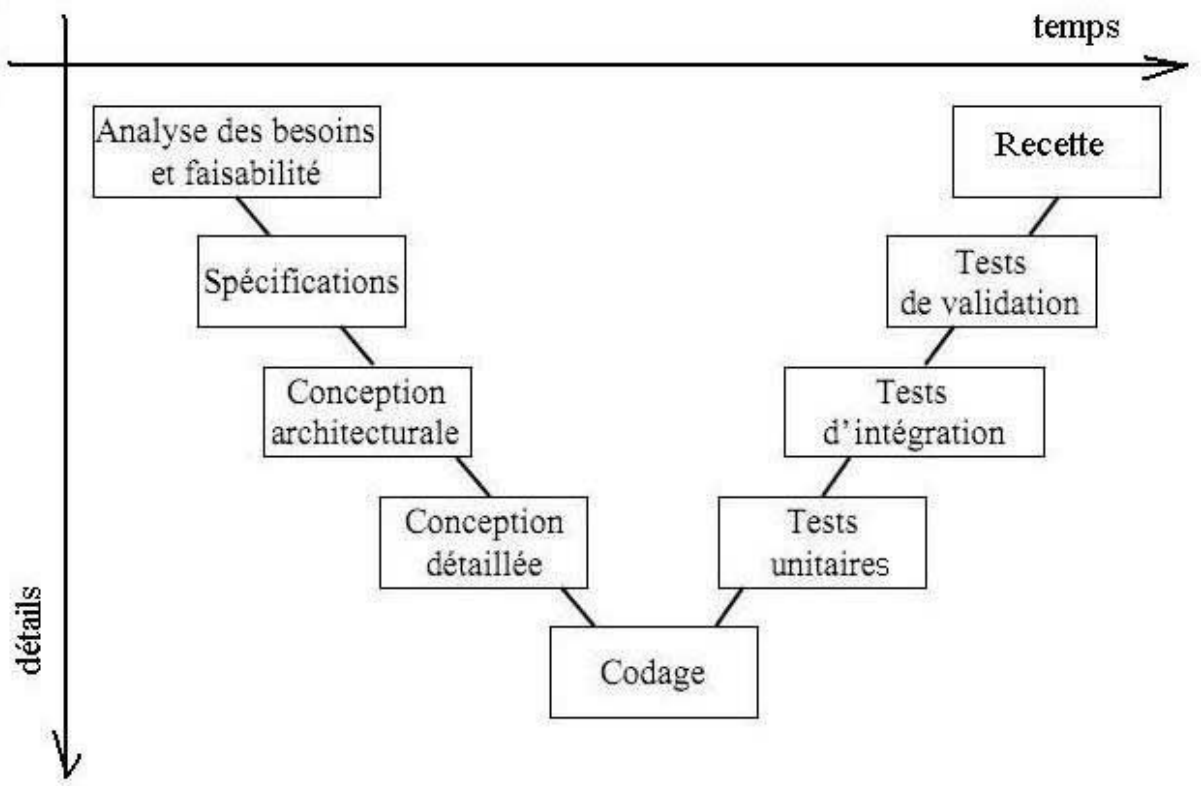
Ma. 8 Jan. 2019 - PHELIZOT Yvan

Cycle en V?

Au début, il y eu le “Warterfall”



Puis le cycle en V



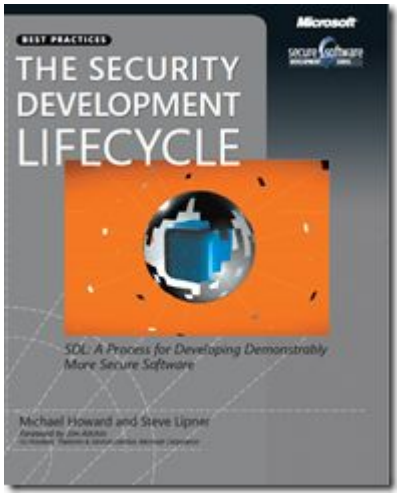
SDLC

Microsoft SDLC

- Security Development Lifecycle
- Intégrer la sécurité à toutes les étapes du projet
- But
  - Réduire le nombre de vulnérabilités
  - Réduire la sévérité des vulnérabilités restantes
- *“Un bug qui n’est pas créé lors du développement est un bug qui n’a pas besoin d’être retiré”*

Phases

1. Education
2. Exigences
3. Conception
4. Implémentation
5. Vérification
6. Déploiement
7. Réponse au incident

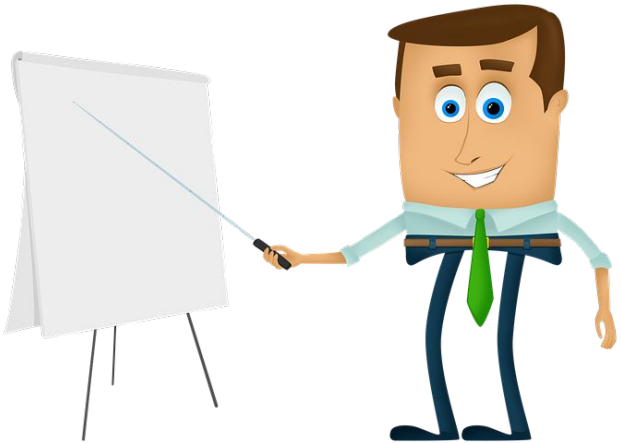


https://blogs.msdn.microsoft.com/microsoft_press/2016/04/19/free-ebook-the-security-development-lifecycle/

https://www.microsoft.com/en-us/SDL/process/training.aspx

1. Education

- Qui? Tout le monde!
  - Managers
  - MOA
  - Architecte
  - Développeur
  - Testeur
  - Exploitant
  - Utilisateur



Why the SDL has been so successful at reducing vulnerabilities ? There are two simple answers: ***executive support****, and* ***education and awareness****.*

2. Exigences

1 2 3 4 5 6 7

- Définition des exigences de sécurité
- Identification des risques pour la Sécurité et la Vie privée
- Bug Bar

3. Conception

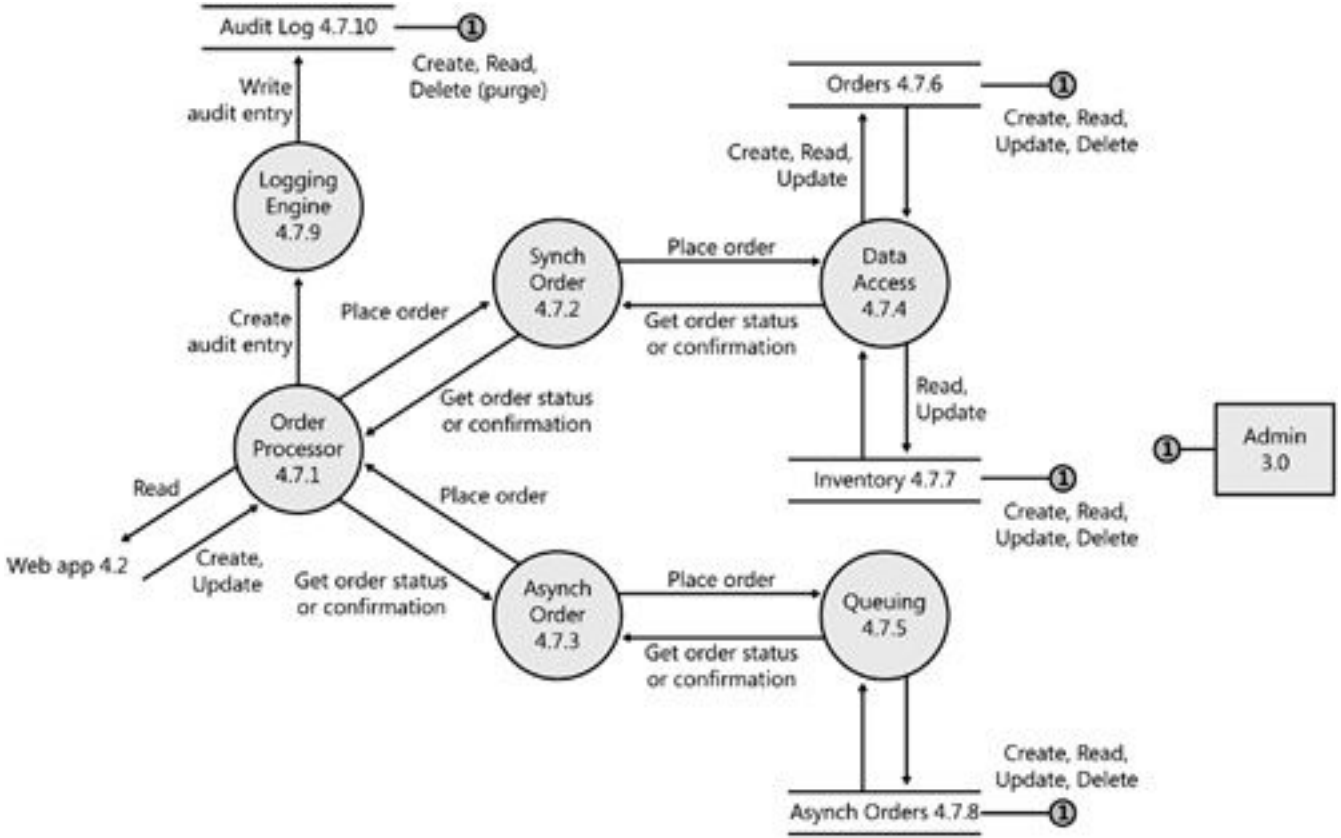
- “Secure by Design”
- Threat Modeling
- Solutions aux menaces identifiées
- Analyse et réduction de la surface d’attaque
- Revue de conception

“Threat Modeling at the design phase is really the only way to bake security into SDLC” - Michael Howard, Microsoft

STRIDE

- **S**poofing
- **T**ampering
- **R**epudiation
- **I**nformation Disclosure
- **D**enial of Service
- **E**levation of Privilege

DFD


Audit Log 4.7.10
Create, Read, Delete (purge)
Write audit entry
Logging Engine 4.7.9
Create audit entry
Order Processor 4.7.1
Read
Web app 4.2
Create, Update
Place order
Synch Order 4.7.2
Get order status or confirmation
Place order
Get order status or confirmation
Asynch Order 4.7.3
Orders 4.7.6
Create, Read, Update, Delete
Create, Read, Update
Data Access 4.7.4
Place order
Get order status or confirmation
Read, Update
Inventory 4.7.7
Create, Read, Update, Delete
Admin 3.0
Queuing 4.7.5
Place order
Get order status or confirmation
Asynch Orders 4.7.8
Create, Read, Update, Delete

DFD Flow

Type d'élément	Elément
External Entities	Pet Shop Customer (1.0) Anon (2.0)
Process	WebApp (4.2) User Profile (4.5) Membership (4.6)

Type d' élément	Elément
Data Flow	Web pages read by Web application (4.3 →4.2) Anonymous user request/response (2.0 →4.2→2.0)
Data Store	WebApp configuration data (4.1) Web pages (4.3) Use profile data (4.8)

DFD Flow + STRIDE

Type	S	T	R	I	D	E
External Entity	X		X			
Data Flow		X		X	X	
Data Store		X	T	X	X	
Process	X	X	X	X	X	X

DREAD

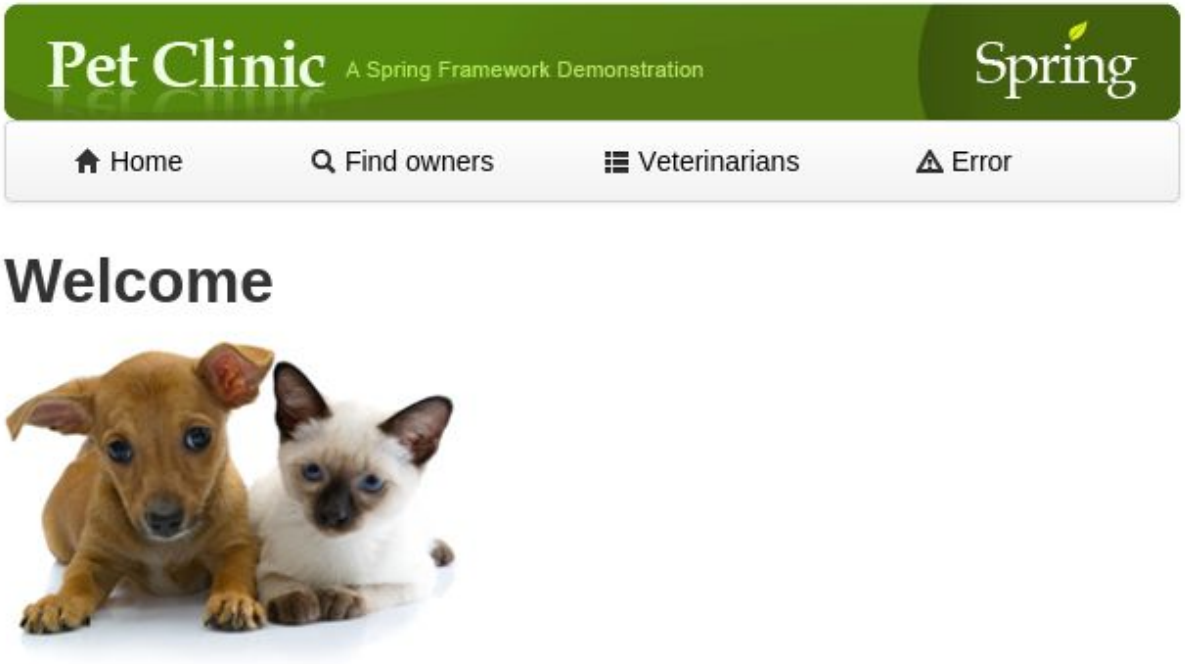
- **D**amage Potential
- **R**eproducibility
- **E**xploitability
- **A**ffected Users
- **D**iscoverability

Risk_DREAD = (D + R + E + A + D) / 5

⇒ Gestion du risque

EoP & Cornucopia

Pet Clinic



spring by Pivotal.

4. Implementation

- Bonnes pratiques
- Outils validés
- Security Guidelines
- Secure coding

Remember, even the most secure design is rendered by a low-quality and insecure implementation, regardless of the number of security features the product employs

5. Vérification

- Sécurité & CI : SAST & DAST
- Test pour la sécurité
- Security Push

6. Déploiement

- Construction du plan de réponse aux incidents
- Revue de sécurité finale

7. Réponse

- Point d’entrée
- Triage
- Relation
- Création du fix
- Test
- Disclosure
- Livraison
- Amélioration Continue

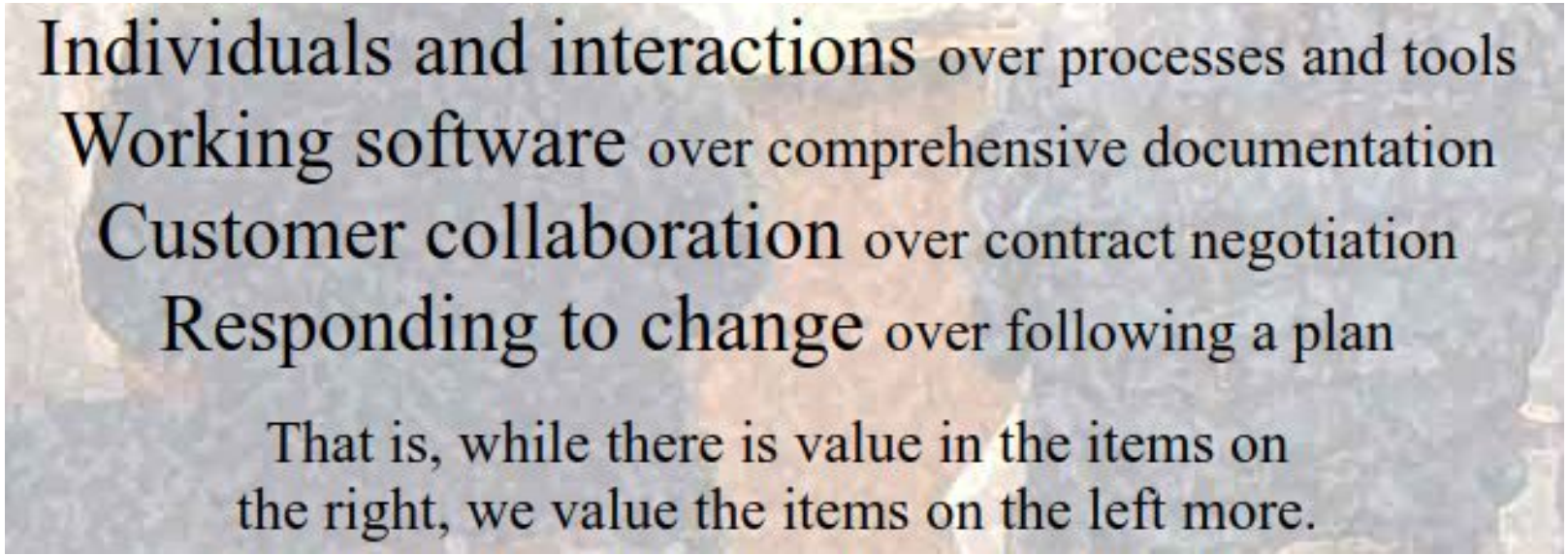
Problèmes du cycle en V?

Problèmes

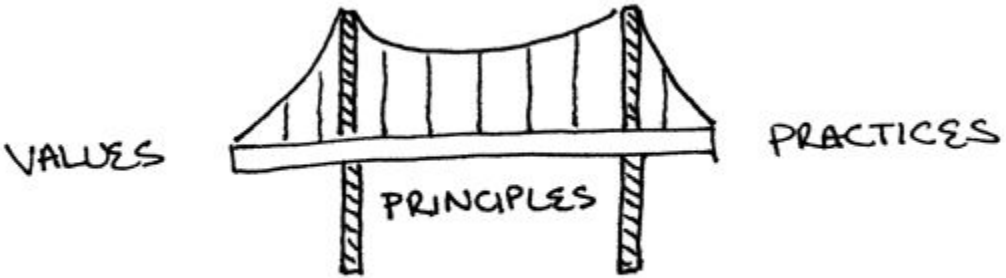
- Délai
- Adaptation aux changements
- Tracabilité
- Cloisonnement
- Hypothèse: connaissance pure et parfaite
- Coût

Agile?
Agilité?

Agile Manifesto et les 4 valeurs



12 principles



Des méthodes...

- SCRUM
- Kanban
- RUP
- XP
- SAFe
- ...

Voire n’importe quoi...

Pourquoi?

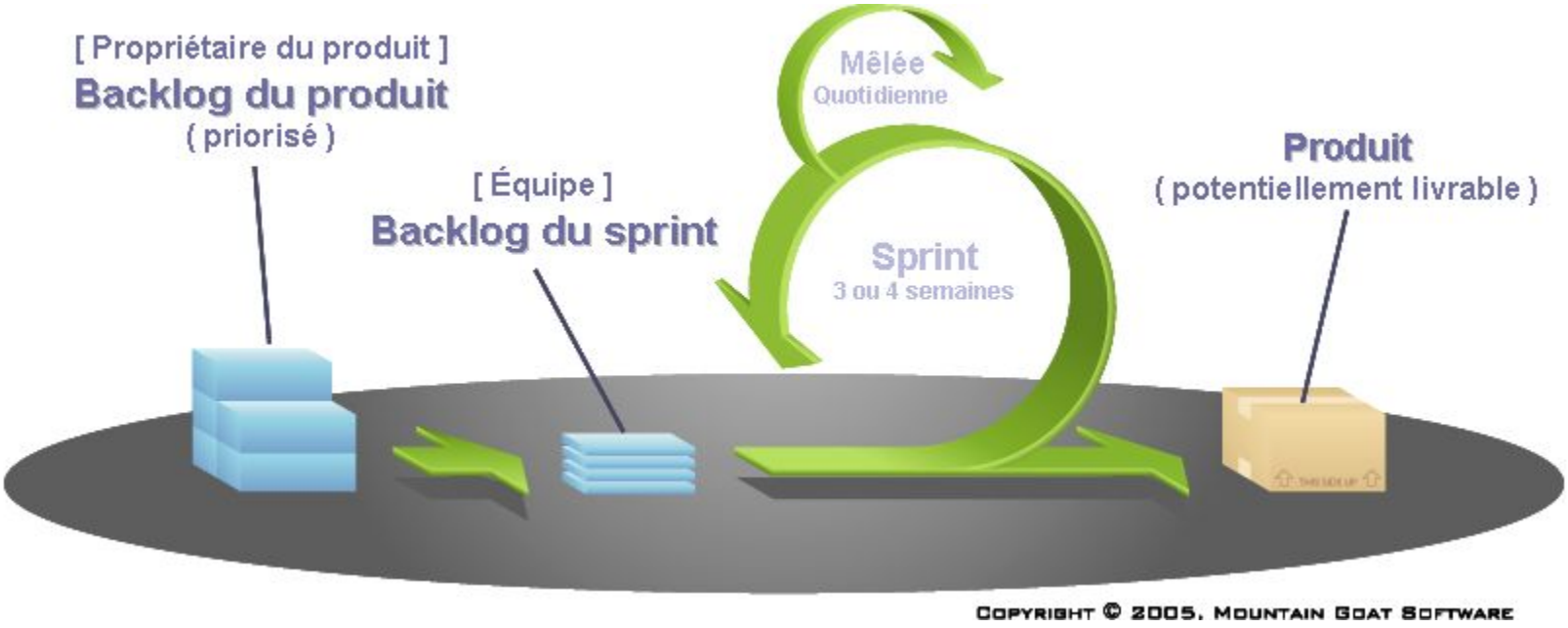
Problèmes?

Problèmes

- Accompagnement du changement
- “Cross-functional team”
- Quid de la sécurité?
- Le produit n’est pas fini!

SCRUM

SCRUM


[Propriétaire du produit]
Backlog du produit
(priorisé)
[Équipe]
Backlog du sprint
Mêlée
Quotidienne
Sprint
3 ou 4 semaines
Produit
(potentiellement livrable)
Copyright © 2005, Mountain Goat Software

Cérémonie

Chaque sprint:

- Sprint planning
- Démo
- Rétrospective

Chaque jour:

- Daily meeting

Epic/Thème/US/Tâche

Equipe

Product Owner

Scrum Master

Externes

SDL for Agile Development

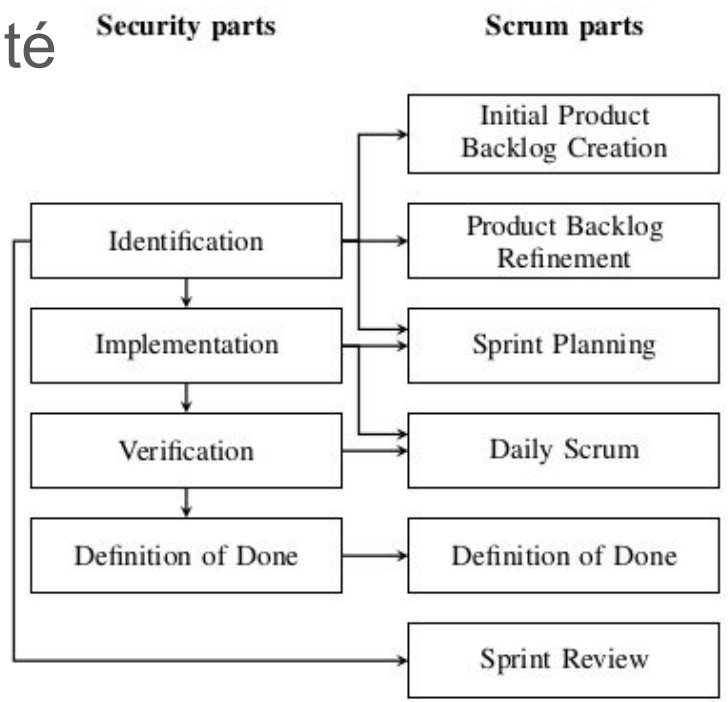
- Changement d’approche
  - Équipe de sécurité comme des facilitateurs
  - Excellence technique
- Avantages: Correction au plus tôt
- Inconvénients: le produit n’est jamais “fini”/figé
- Approche incrémentale de la sécurité

Ce qui ne change pas

- Education à la sécurité
- Les équipes de sécurité comme facilitateurs
- L’analyse de risque
- La partie implémentation (secure coding, bonnes pratiques, …)
- La réponse en cas d’incident

Secure Scrum

- Extension de Scrum pour la sécurité
- 4 composantes
  - Identification des US sensibles du Backlog
  - Implementation
  - Vérification : les membres sont apte à tester
  - DoD



ANSSI et l’agilité

“Intégrer la sécurité numérique en démarche Agile”

“Agilité & Sécurité Numériques”

Exercice: Enjeu au niveau sécurité?

User stories	***D***	***I***	***C***	***P***
Un client transmet son identifiant, sa position et son numéro de téléphone				
Un client peut commander un taxi				
Un client peut calculer le prix d'une course				
Un administrateur peut enregistrer un taxi				

Besoin important en termes de sécurités *, très important **

Des besoins de sécurité...

User stories	***D***	***I***	***C***	***P***
Un client transmet son identifiant, sa position et son numéro de téléphone	*	**	**	
Un client peut commander un taxi	*	**	*	*
Un client peut calculer le prix d'une course		*		*
Un administrateur peut enregistrer un taxi		*		*

Exercice:
Quels types de problèmes peut-on imaginer?

- Evénements redoutés?
- Impacts métier?
- Gravité?

… aux événements redoutés

Evénements redoutés	Impacts métier	Gravité
Le système ne répond pas	Expérience utilisateur dégradée ⇒ **Perte de clients**	*
Un opérateur de taxis émet de fausses positions	Qualité de service dégradée ⇒ **Perte de clients**	*
Un taxi fait une course d'approche en pure perte	Perte de confiance des taxis ⇒ **Désengagement taxi**	**

Exercice : Evil/Abuser User Story?

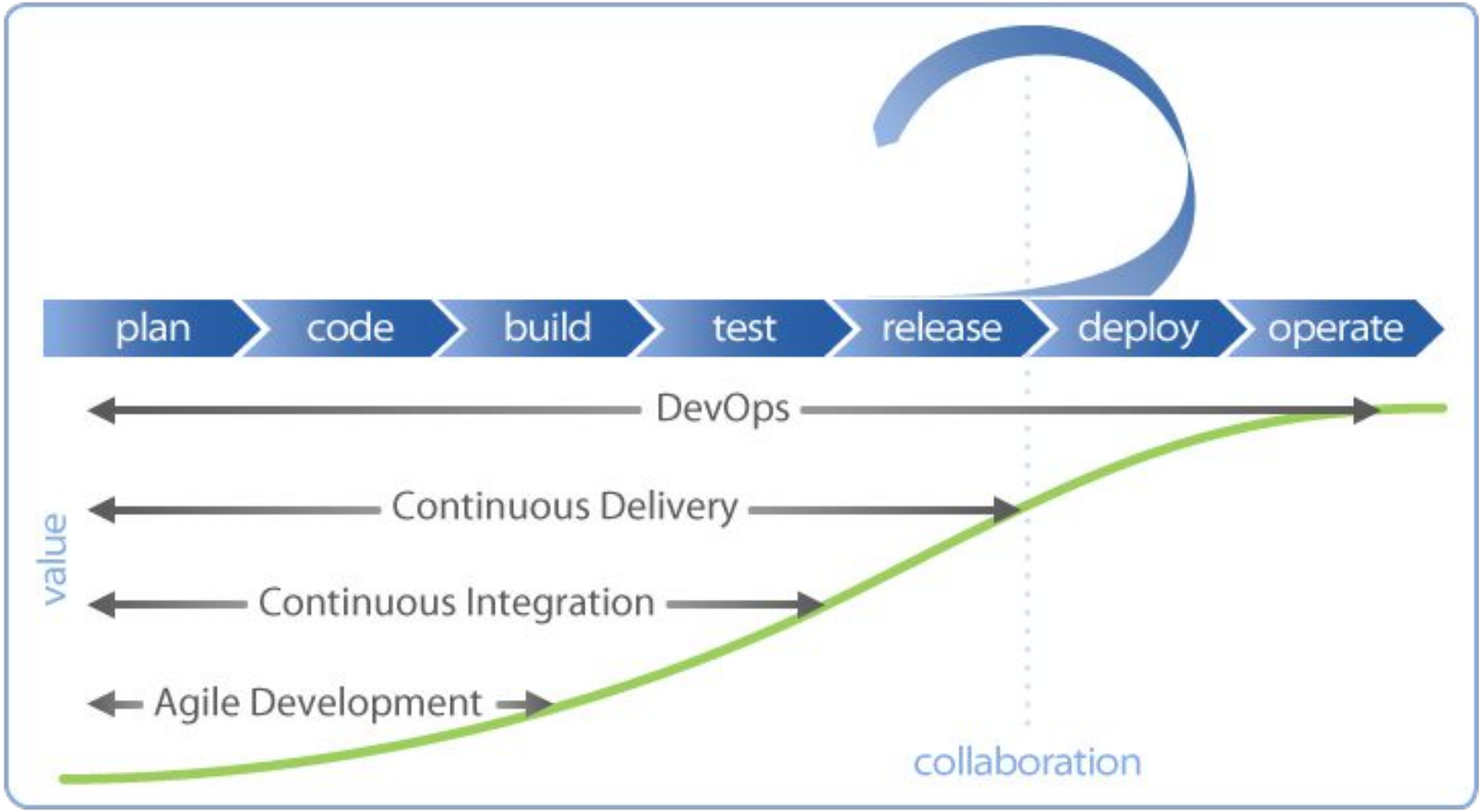
Livrer un logiciel non fini?

MVS : Minimal Viable Secure Software

DevSecOps

Aujourd’hui...

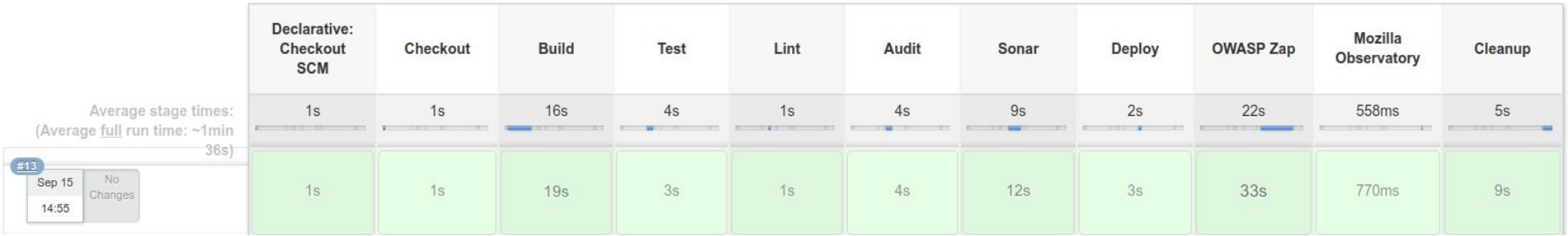
- Des cycles de développement en de plus en plus courts
- Des déploiements facilités par le cloud et la “containerisation” des applications
- Moins de distance entre le développement et la production


plan
code
build
test
release
deploy
operate
DevOps
Continuous Delivery
Continuous Integration
Agile Development
value
collaboration

Sécuriser la chaîne de livraison

	Avantages	**Inconvénients**
Analyse statique	Analyse complète du code Précis Rapide	Limité (quelques lignes de codes) Faux positif
Analyse dynamique	Teste l'outil dans sa globalité	Les cas de tests ne sont pas forcément exhaustifs

Exemple avec Jenkins



	Declarative: Checkout SCM	Checkout	Build	Test	Lint	Audit	Sonar	Deploy	OWASP Zap	Mozilla Observatory	Cleanup
Average stage times: (Average full run time: ~1min 36s)	1s	1s	16s	4s	1s	4s	9s	2s	22s	558ms	5s
#13 Sep 15 14:55 No Changes	1s	1s	19s	3s	1s	4s	12s	3s	33s	770ms	9s

https://github.com/cotonne/misc-101/

Agilité, la sécurité et...

- La réglementation