

# Pourquoi une approche structurée du test ?



SOGETI



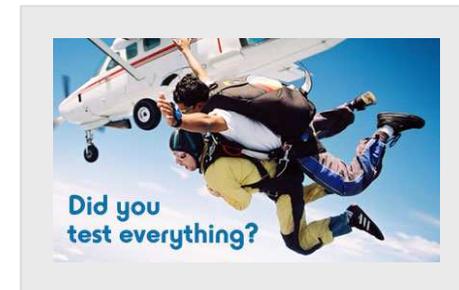
# Structurer le test pourquoi?

2016/06/16 16:22 bonjour!



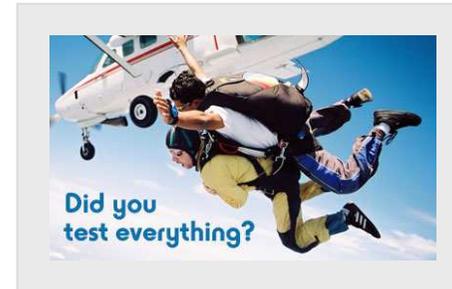


- Livrer un logiciel sans bug ?
- Détecter les anomalies le plus tôt possible et au moindre coût ?
- Livrer des outils et des processus de test réutilisables ?
- Prévenir les anomalies ?
- Améliorer la qualité logicielle ?
- Informer le client sur la qualité ?
- Elaborer une stratégie ?
- Identifier, organiser, hiérarchiser les risques ?
- Etre en capacité de prendre en compte des exigences ?
- Etc. . ...





- Tester un objet
  - Ne se limite pas à l'exécution uniquement
  - Implique une démarche structurée
  - Donne les éléments factuels quantitatifs et qualitatifs
  - Est un levier de décision
  
- Prendre en compte le processus de test dans sa globalité
  - Pré requis
  - Activités
  - Techniques
  - Méthodes
  - Livrables
  - Organisation
  - Outils
  - Reporting
  - Communication





- **Ce que l'on teste**
  - Paramétrage/configuration de logiciel applicatif
  - Logiciel système, développement spécifique, hardware
  - Composants, services
  - Chaînes de liaison, ponts migratoires
  - Procédures & outils
  - Processus de mise en service
  - Processus de déploiement
  - Documentation, supports utilisateurs, supports de formation
  
- **Ce que l'on veut vérifier**
  - Fonctionnalités
  - Performance
  - Sécurité
  - Maintenabilité
  - Etc.



Il faut tout  
tester et  
savoir  
s'arrêter





- **Quel est le profil du testeur?**
  - Développeur
  - Chercheur
  - Concepteur
  - Utilisateur
  - Analyste
- **Quelle expérience et quelle expertise**
  - Opérationnelle
  - Méthodologique
  - Technique,
  - Fonctionnel ou métier
- **Dépendance et indépendance des rôles?**
  - Développeur/testeur
  - Chercheur/testeur





- **Compétences**

- Fonctionnelles et métier
- Mise en œuvre d'indicateurs et d'outils de suivi
- Gestion d'un référentiel de test
- Développement
- Cartographie architecture

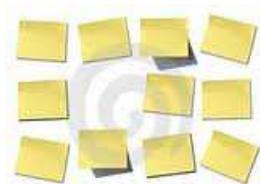


- **Outils**

- ✓ Automatisation des TNR
- ✓ Initialisation/chargement des données
- ✓ Identification & gestion des anomalies
- ✓ Reporting des faits marquants

- **Méthodes**

- ✓ Suivi des indicateurs
- ✓ Bonne vision des processus (ordonnancement)
- ✓ Procédures spécifiques au métiers
- ✓ TMap Next





- Réduction du « time-to-market ».
- Réduction des coûts des ressources de test.
- Réduction des coûts des infrastructures de test.
- Une meilleure vision de la qualité du produit.
- Un processus de tests mieux géré
  - Gestion du recouvrement entre les niveaux de tests
  - Automatisation adéquate du processus de tests
  - Réutilisabilité du dossier de tests et de la compétence acquise
- Confiance renforcée dans la qualité des objets à tester





SOGETI



EISTI

# Questions.....

Questions.....





# Exercice 1 - Mettre en œuvre les tests

EXERCICE 1 - METTRE EN ŒUVRE LES TESTS

## Contexte de l'exercice

Vous faites partie de la cellule de 10 experts de test au sein d'une société. Cette cellule est sollicitée pour tester un nouvel ordinateur avant que celui-ci soit distribué à des clients invités à un salon international des nouvelles technologies et de la mobilité.

Vous avez la responsabilité de vous prononcer sur la qualité de cet équipement avant qu'il ne soit envoyé en production puis distribué aux clients que votre société a convié à cet évènement.





## Ce qui vous est demandé :

- ✓ Précisez comment vous menez à bien cette tâche qui vous est confiée
- ✓ Quelle organisation vous semble la plus appropriée
- ✓ Précisez votre démarche, ce que vous comptez tester, comment et pourquoi ?
- ✓ Indiquer le nombre de jours nécessaire pour effectuer vos test avant que ce prototype soit envoyé en production



90 mn

## Restitution

A l'issue du travail en groupe vous présenterez le fruit de votre réflexion (ppt., .doc, ...)

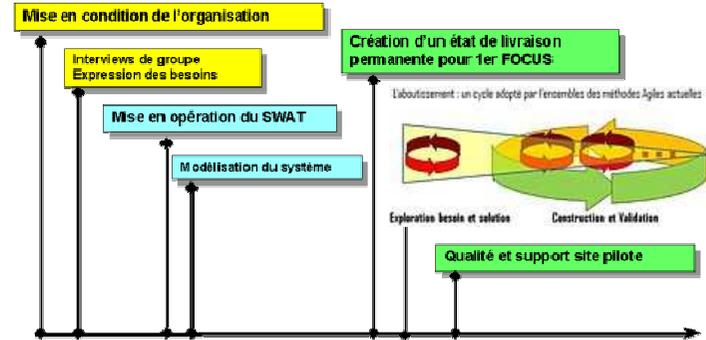
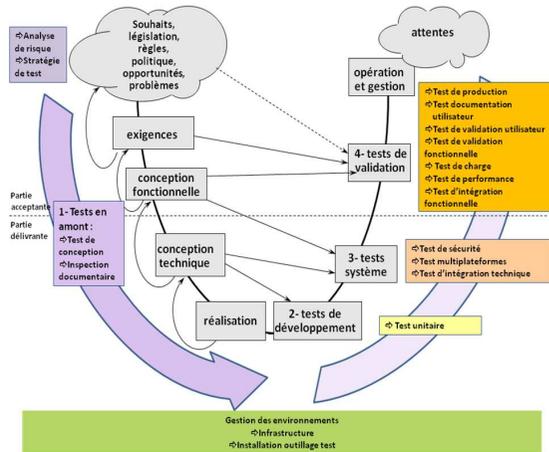
- ✓ Ultra Mobile PC - écran de 12 pouces
- ✓ Technologie wifi
- ✓ Autonomie estimée à 5h
- ✓ Processeurs AMD
- ✓ Equipée de carte vidéo avec sortie haute définition HDMI
- ✓ disque dur de 250 Go





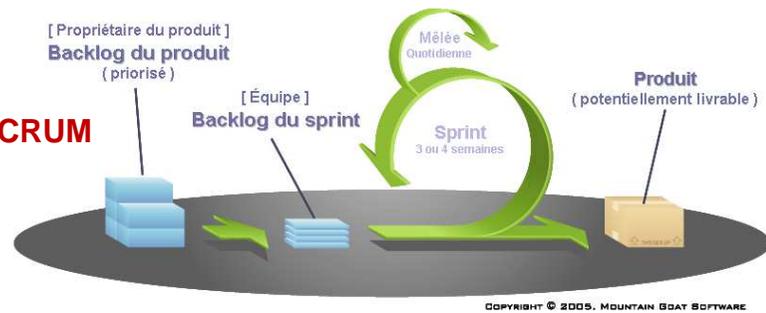
# Cycles de vie du projet

## Cycle en V

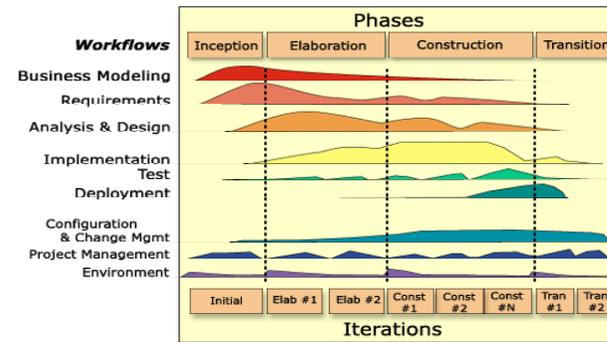


## RAD

## SCRUM

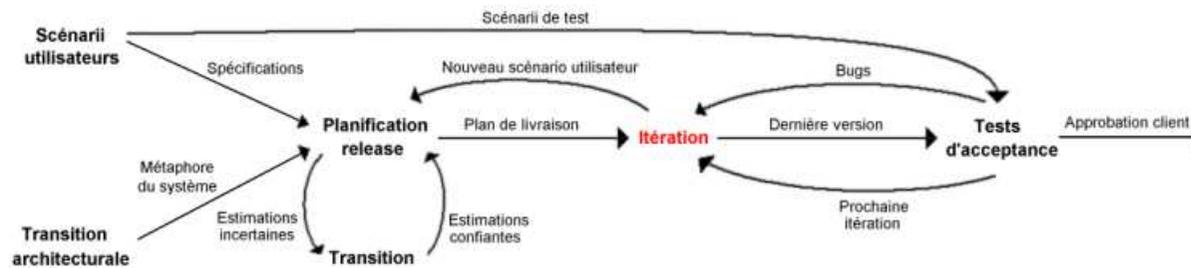


COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE

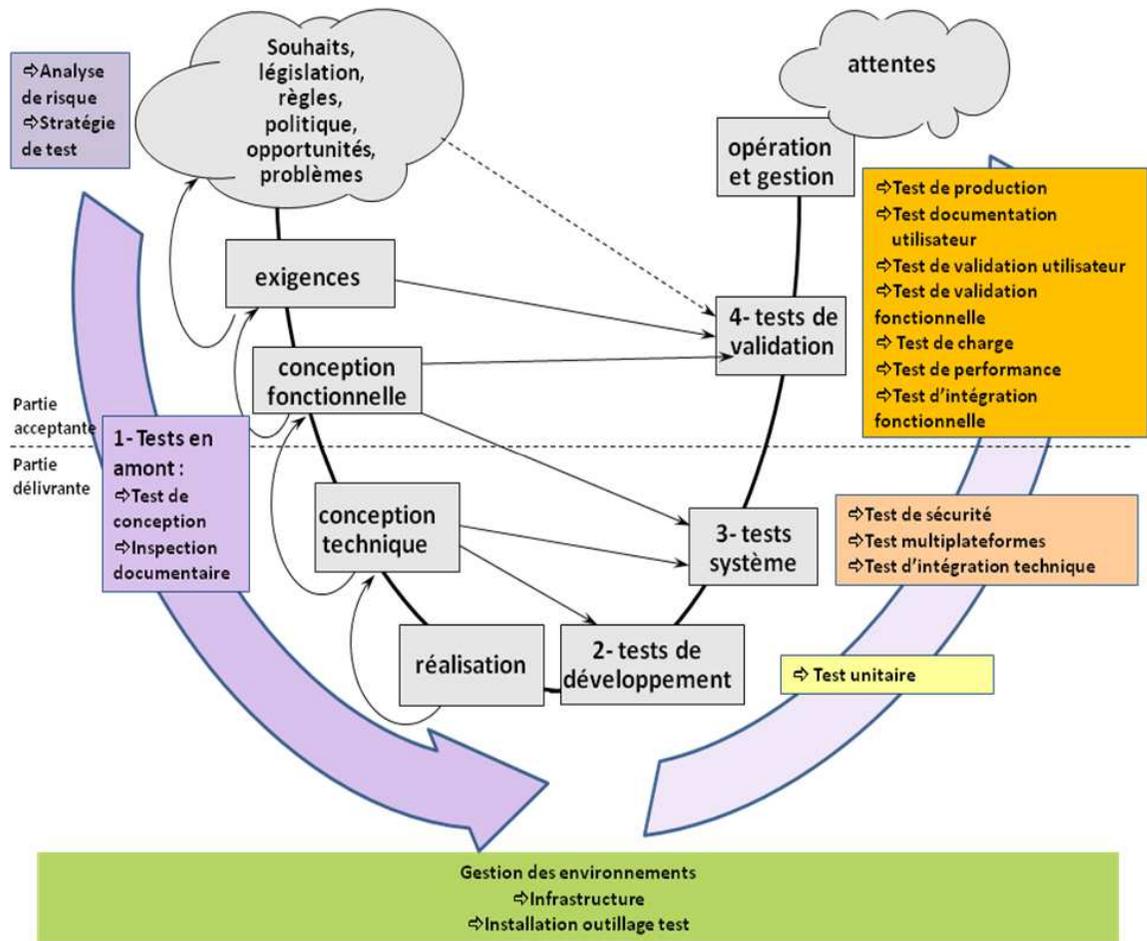


## RUP

## XP



# Le Cycle de vie projet - le cycle en 'V'





LOGEL 162 D01162 D062C10U2\*\*\*

Les tests sont-ils priorités?  
En fonction de quoi?

## Cycle de test en V

Si la vie du projet impose de diviser par deux le temps alloué au test, que se passe t-il?

Avons nous besoin d'un Référentiel de test?

Quel est notre Gouvernance des tests ?

Partie acceptante

Partie délivrante

Exigences

Conception fonctionnelle

Conception technique

Réalisation

Opération et gestion

Tests de validation

Tests système

Tests de développement

TS, TVF, TR, TP

TS

TU, TIU

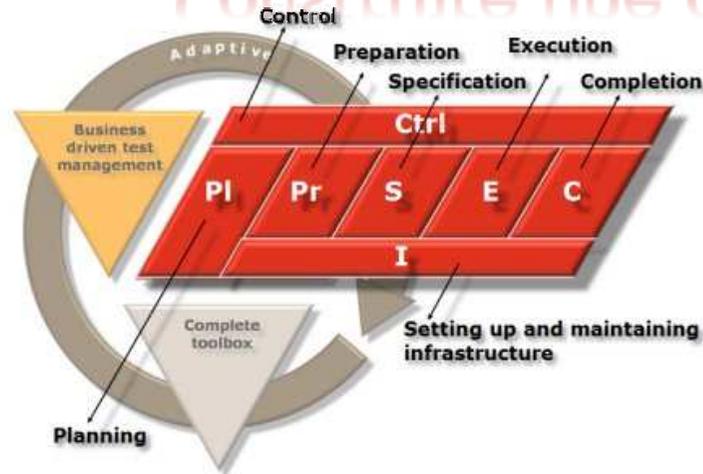




SOGETI



# Construire une démarche structurée



Partie acceptante

Exigences

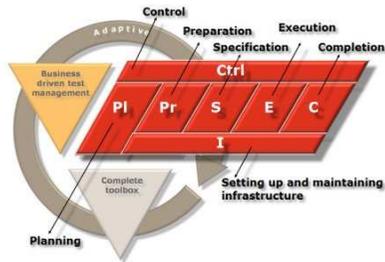
Opération et gestion



Partie délivrante



3 Niveaux de tests



# Gouvernance des tests

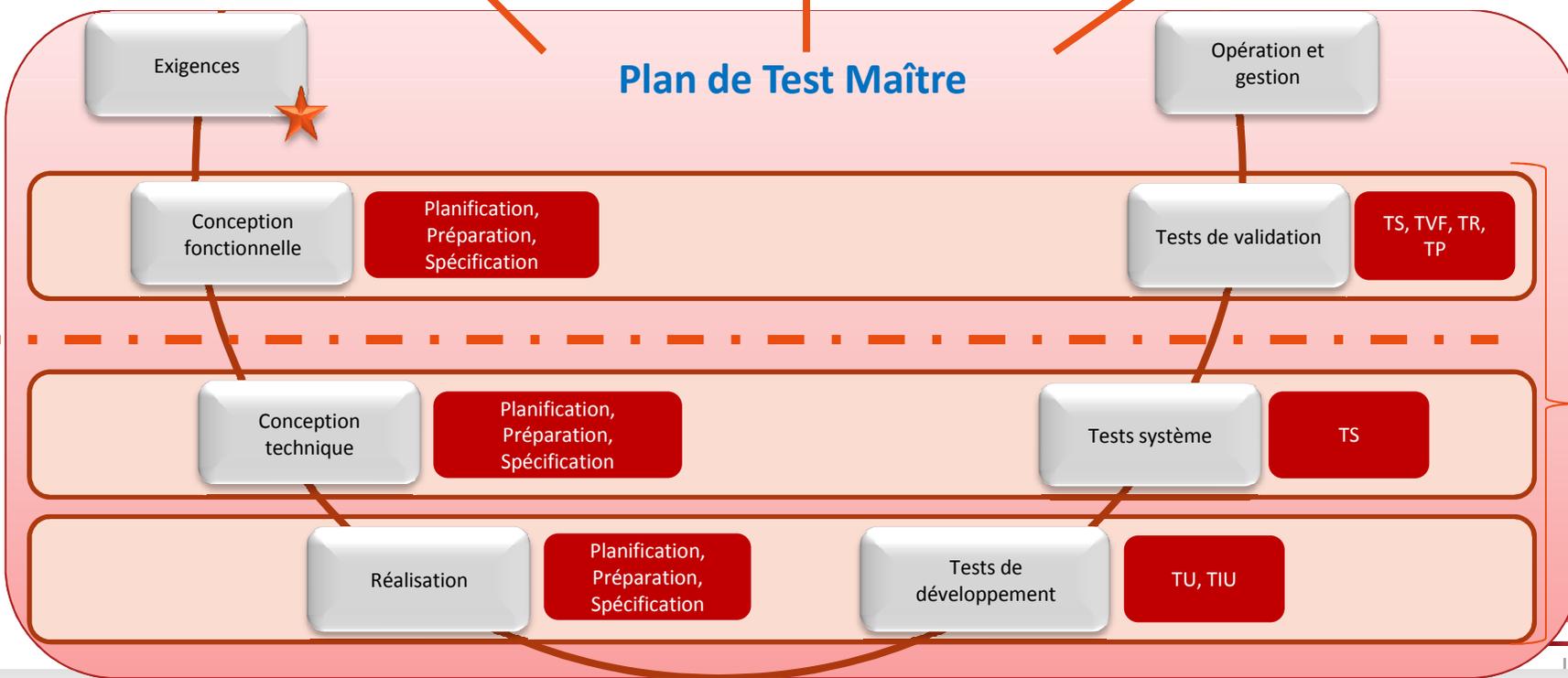
Garantir la maîtrise des risques

Mise en place d'une stratégie globale de test

Eviter les redondances

Partie acceptante

Partie délivrante



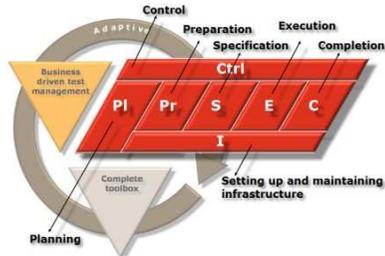
3 Niveaux de tests



SOGETI



# Stratégie de test & Analyse de risques

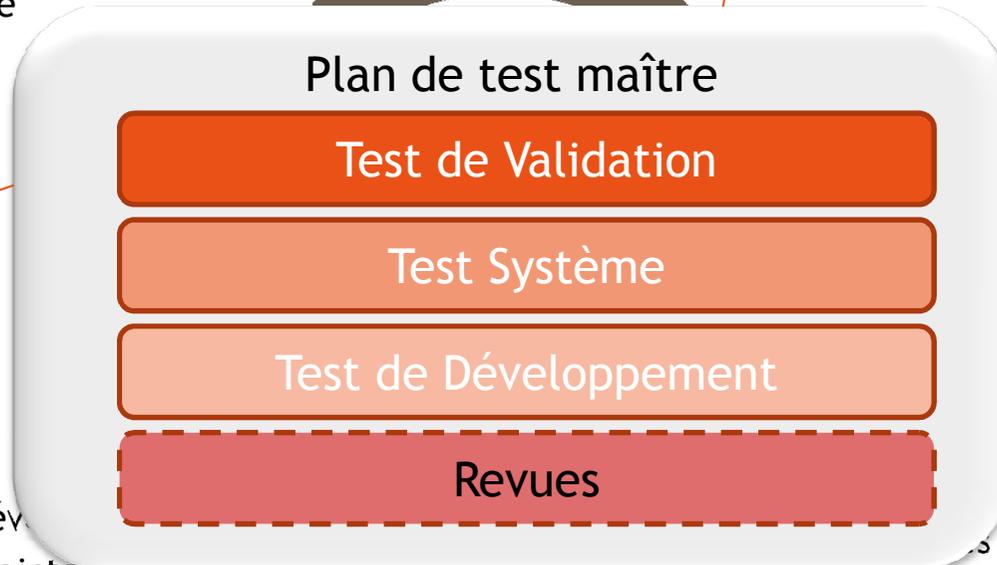


## Solutions

- Stratégie de test orientée risques
- Processus de test structuré
- Gestion de la couverture et de la profondeur de test
- Anticipation de la détection des anomalies
- Modèles réutilisables
- Outils

Optimiser le coût des tests (>30% des projets)  
 Anticiper la qualité perçue  
 Raccourcir les délais

Adaptable

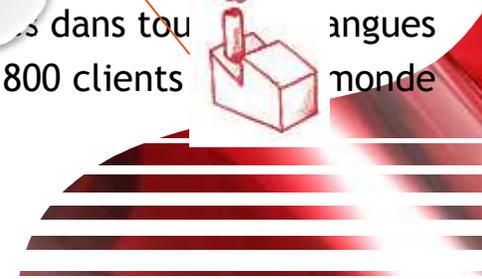


## Résultats obtenus

- Réduction des coûts de développement
- Réduction des coûts de maintenance
- Réduction de la charge de recette utilisateur
- Facteur qualité élevé (99%)
- Réduction des durées de tests (automation)

## Faits

- Standard
- Langues
- monde
- Plus de 800 clients



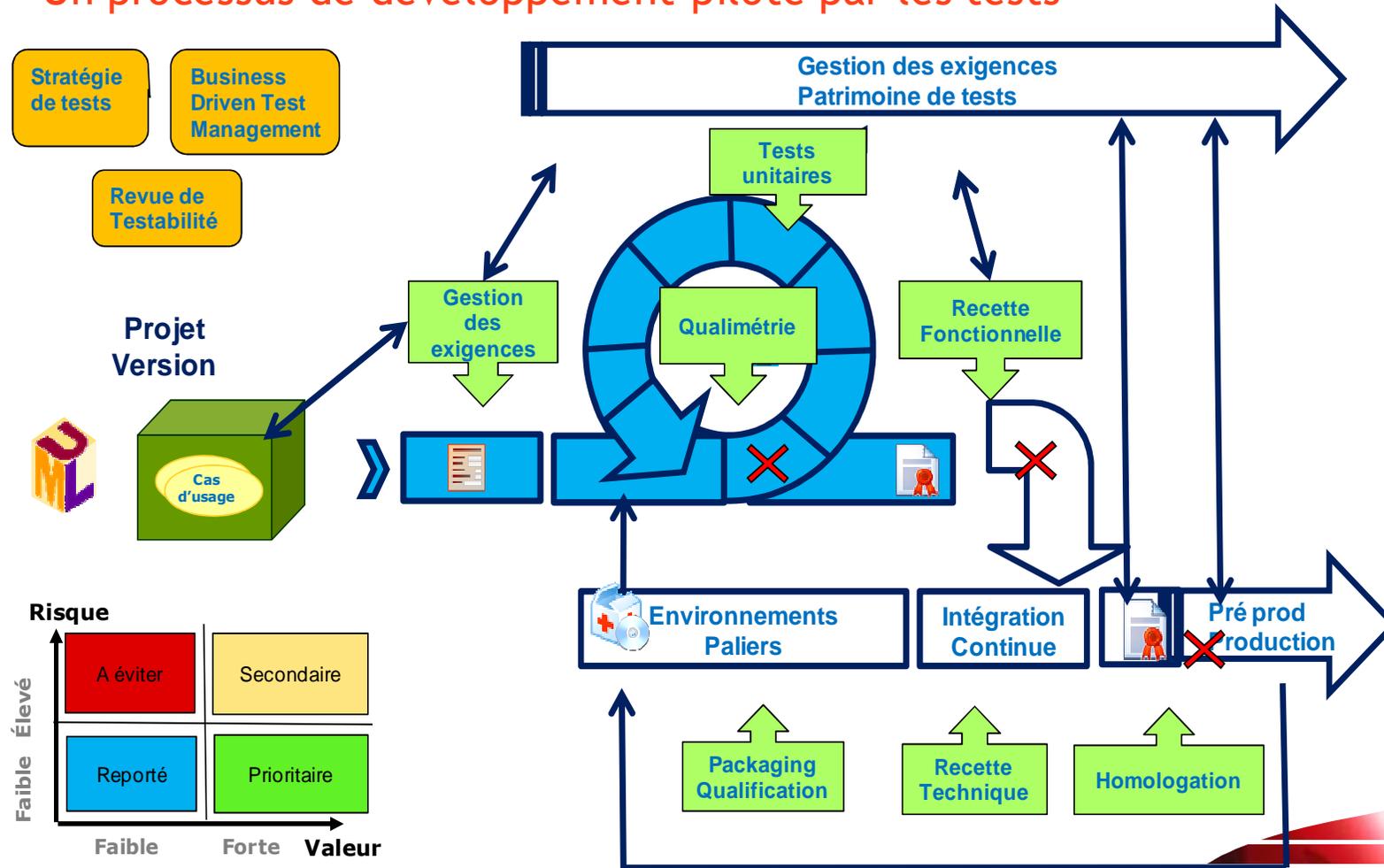


- Les méthodes "Agiles" permettent :
  - De concevoir des logiciels en impliquant au maximum le client.
  - De privilégier la communication et l'interaction qui en résulte.
  - De s'appuyer sur une démarche favorisant la réactivité et de piloter les projets par les enjeux et les risques.
  - D'avancer par jalons, par prototypage actif itératif et incrémental.
  - D'éviter la dérive et les échecs des projets (trop fréquents!).
- Les méthodes "Agiles" favorisent :
  - L'acceptation du changement.
  - La modification des priorités (Time-Box, Task-Box)
  - L'utilisation d'une planification flexible.
- Les méthodes "Agiles" ont pour nom :
  - RAD,
  - RUP,
  - XP,
  - SCRUM ...



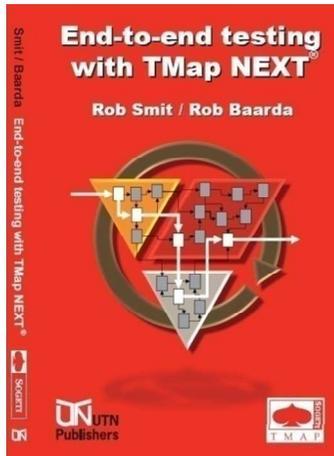


## Un processus de développement piloté par les tests

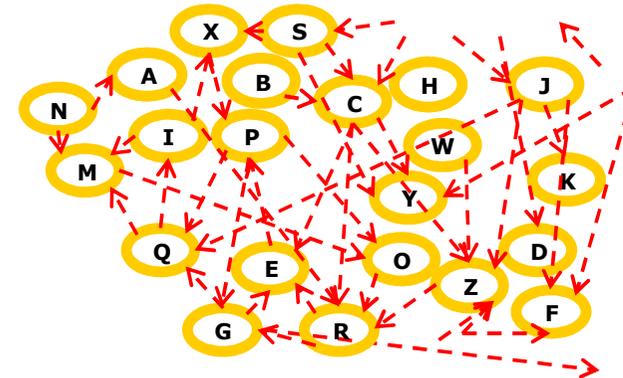




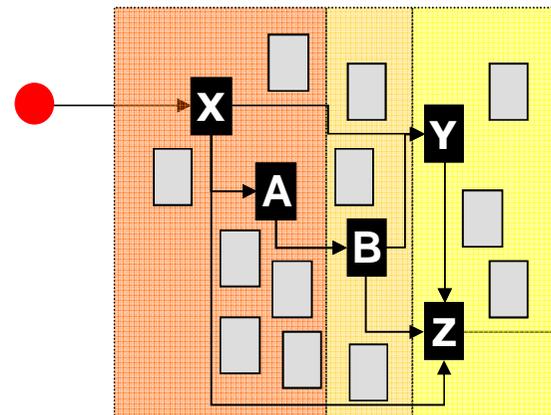
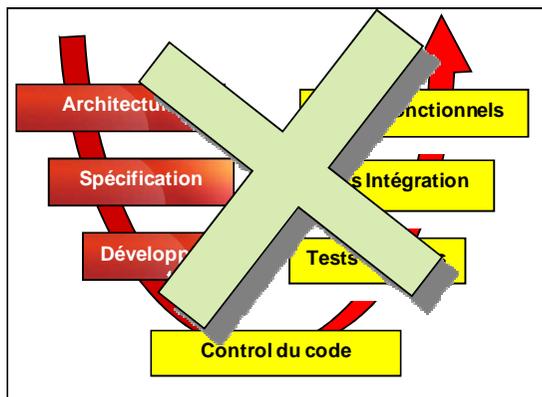
## E2E : End to End Testing



Contexte: les systèmes sont bien souvent plus interconnectés qu'il n'y paraît, ce qui rend le test long et complexe



Dans ces conditions, E2E testing devient la solution

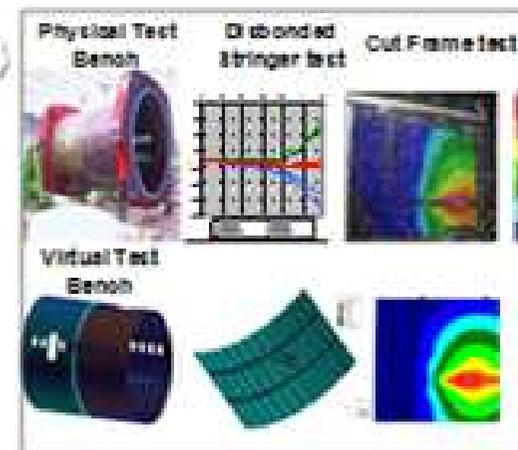
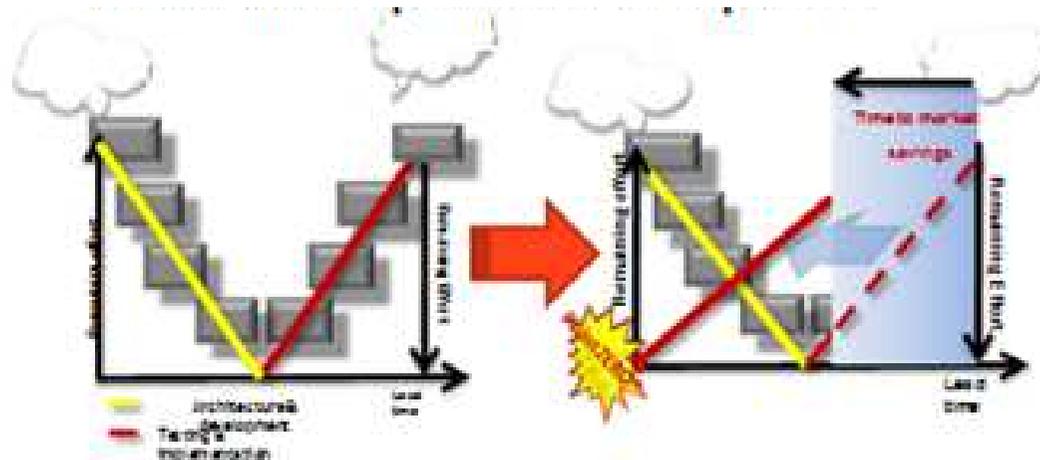
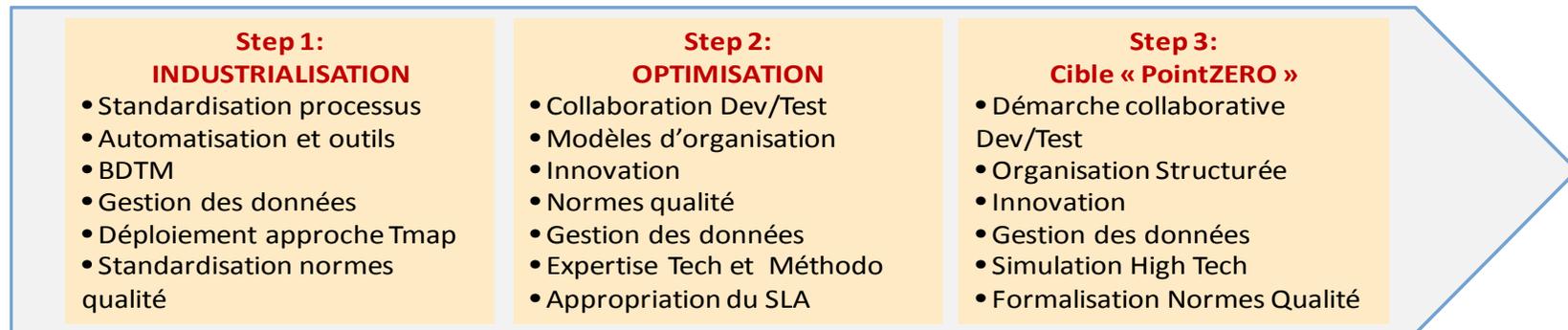




# Le Cycle de vie projet Virtual Testing

## Modèle PointZERO® :

Combinaison du savoir-faire Simulation et Testing afin d'améliorer les





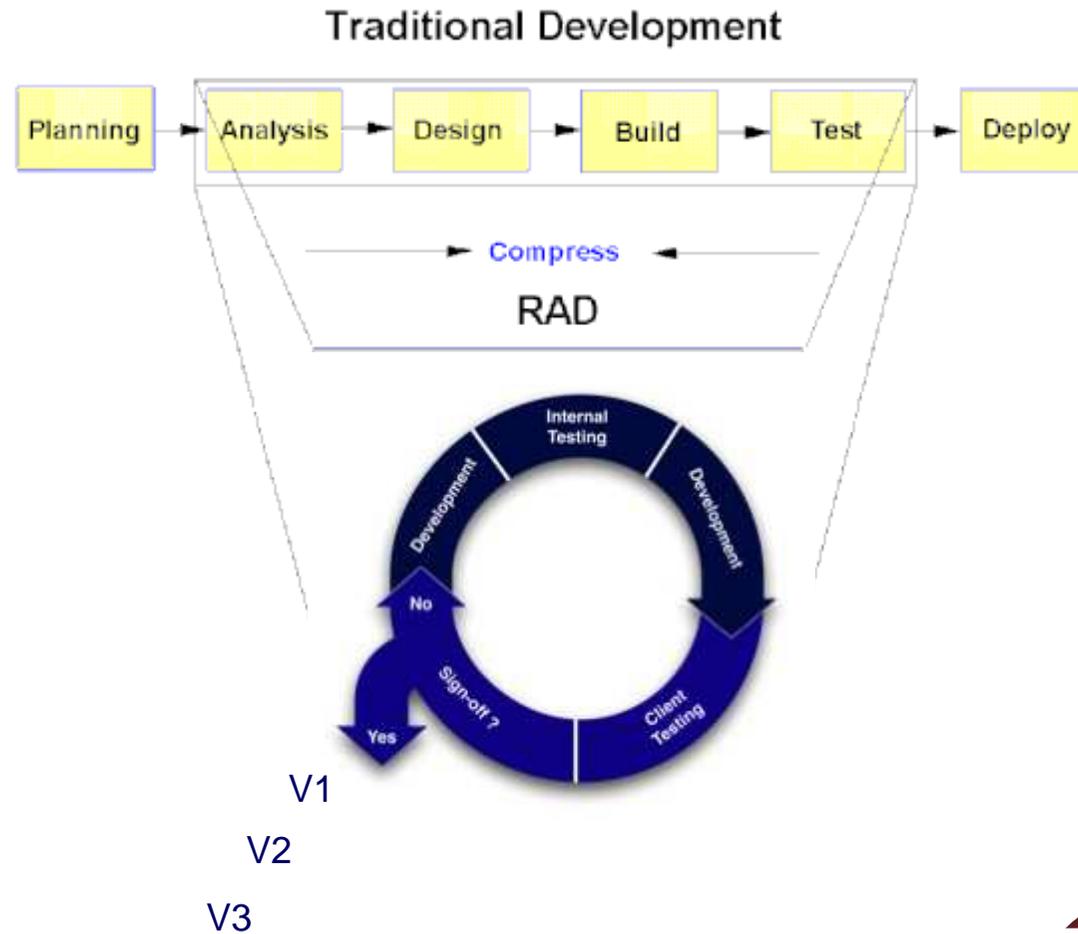
## Rapid Application Development - RAD

- **Objectif**
  - Réduire le cycle de développement
    - Réduction de la charge (gain d'argent) et des délais (gain de temps)
  - Garantir l'adéquation entre la qualité du produit fini et les besoins des utilisateurs
- **Principe**
  - La phase de lancement permet :
    - De détailler le planning et les objectifs du projet.
    - De mettre au point l'organisation, les modalités de communication et de management.
  - La phase de cadrage fonctionnel permet
    - De déterminer avec le client, le périmètre fonctionnel du projet.
    - De produire un premier sous-ensemble fonctionnel (maquettes),
  - La phase d'itération permet :
    - De produire des versions intermédiaires (construction progressive des scénarios fonctionnels).
    - D'identifier pour chaque version des tâches de conception technique, de réalisation et de tests.
  - La phase de mise en œuvre permet :
    - De finaliser l'application
    - De produire une version V1.0 pour présentation en réception (recette).





## Rapid Application Development





## Rational Unified process

- **Objectif**

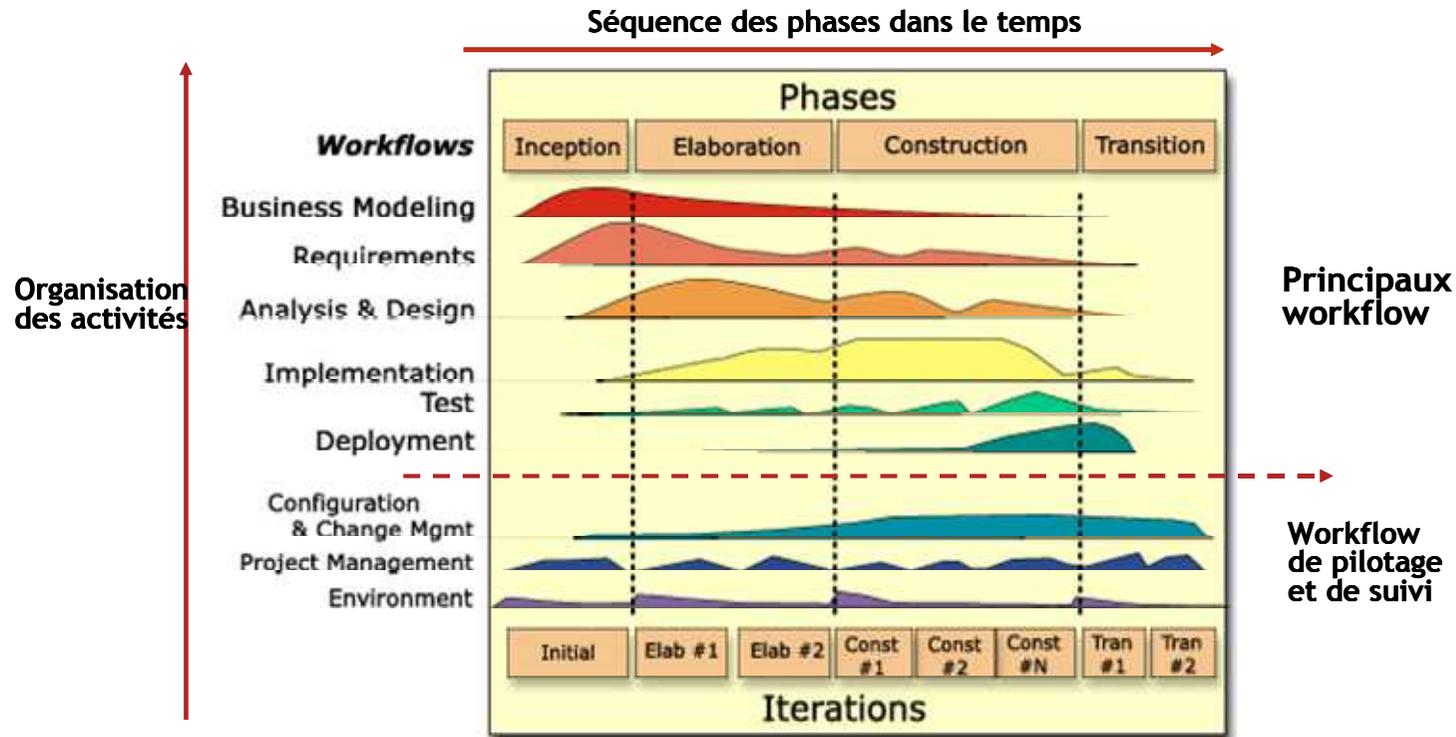
- **Organiser un projet et toutes les activités associées dans le temps.**
  - En utilisant une méthode itérative et incrémentale qui s'apparente à un outil.
  - En s'appuyant sur 4 phases (Inception, Elaboration, Construction, Transition)

- **Principe**

- **Méthode qui repose sur 6 piliers**
  - 1°) Développement itératif
  - 2°) Modélisation visuelle
  - 3°) Architecture basée sur des composants
  - 4°) Gestion des exigences
  - 5°) Vérification continue de la qualité
  - 6°) Gestion des modifications et de la configuration



## RUP : présentation du processus



- **L'axe vertical** représente la séquence des phases dans le temps (le cycle de vie du processus). La fin de chaque phase est marqué par des jalons permettant de valider des décisions stratégiques et de vérifier l'atteinte des objectifs majeurs de chaque phase.
- **L'axe horizontal** représente l'organisation du travail (disciplines, groupe d'activités, rôles, ...).



## XP: Extreme Programming

- **Objectif**
  - Rendre le projet plus flexible et ouvert aux changements.
- **Principes :**
  - Puisque la revue de code est une bonne pratique, elle sera faite en permanence (par un binôme) ;
  - Puisque les tests sont utiles, ils seront fait systématiquement avant chaque implémentation ;
  - Puisque la conception est importante, elle sera faite tout au long du projet ;
  - Puisque la simplicité permet d'avancer plus vite, la solution la plus simple est privilégiée;
  - Puisque l'intégration des modifications est cruciale, elles seront prises en compte (si nécessaire plusieurs fois par jour) ;
  - Puisque les besoins évoluent vite, plusieurs cycles de développement très rapides sont mis en œuvre pour une adaptation permanente aux changements.
- **Cycles rapides de développement**
  - Des itérations de développement de quelques semaines
  - Une phase d'exploration détermine les scénarios clients qui seront fournis pendant chaque itération.





Cycle de vie	Forces	Inconvénients
"V"	<ul style="list-style-type: none"><li>▪ Cohérence pas à pas.</li><li>▪ Vision globale.</li></ul>	<ul style="list-style-type: none"><li>▪ Ergonomie imprécise.</li><li>▪ Peu évolutif</li><li>▪ Effet tunnel.</li></ul>
RUP	<ul style="list-style-type: none"><li>▪ Cohérence entre l'organisation du travail et la séquence des phases dans le temps</li><li>▪ Méthode incrémentale et itérative</li></ul>	<ul style="list-style-type: none"><li>▪ pas d'inconvénient « structurel », mais implique d'adapter le processus à la typologie du projet</li></ul>
"RAD"	<ul style="list-style-type: none"><li>▪ Pilotage par les délais et les coûts.</li><li>▪ Adéquation produit / besoin.</li></ul>	<ul style="list-style-type: none"><li>▪ Qualité en "juste assez".</li><li>▪ Évolutivité du produit gérée tardivement.</li></ul>
XP	<ul style="list-style-type: none"><li>▪ Cycle de développement très court.</li><li>▪ Fonctionnement en binôme (développeur/testeur)</li></ul>	<ul style="list-style-type: none"><li>▪ Peu de formalisation des cas de test, scénario, etc. Les re-test parfois difficiles.</li></ul>



