

CENTRALE  
L Y O N

# Test de Logiciels

BTD/GL/Tests 1

CENTRALE  
L Y O N

## Principes (1/4)

Bertrand DAVID : Génie Logiciel

- Quelle est la définition exacte du test ?

Le test consiste à exécuter une application pour trouver les anomalies qui y sont enfouies ; vérifier que l'application fonctionne en fonction des spécifications et des données qu'elle manipule.
- Quelle est la différence entre le test et la mise au point ?

La mise au point est faite par les développeurs, elle joue le rôle de validation du développement. Le test consiste à vérifier que l'application fonctionne comme elle devrait.
- Existe-t-il une relation entre la qualité du logiciel et le test ?

Plan qualité constitue une démarche globale, le test une vérification locale.

BTD/GL/Tests 2

CENTRALE L Y O N	<h2>Principes (2/4)</h2>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● Peut-on tester une application sans l'exécuter ? Approche « boîte blanche » s'intéresse à la qualité du code</li><li>● Peut-on estimer le nombre d'anomalie ? Il est possible d'estimer ce nombre en fonction de critères techniques.</li><li>● Peut-on tester sans outil ? Tests manuels sont possibles et efficaces.</li></ul>
BTD/GL/Tests	3

CENTRALE L Y O N	<h2>Principes (3/4)</h2>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● Trouve-t-on une seule catégorie d'outils de test ? Non : tests fonctionnels, tests de non-régression, tests de performances</li><li>● Peut-on définir une stratégie de test ? Oui : selon la qualité visée, types d'anomalies à détecter, approches à utiliser (descendante ou ascendante), ...</li><li>● Comment tester des applications distribuées ? Il faut faire appel à des outils spécialisés, ...</li></ul>
BTD/GL/Tests	4

CENTRALE L Y O N	<h2>Principes (4/4)</h2>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● Peut-on évaluer le coût du test ? Oui, 30 à 50% du temps de développement.</li><li>● Qui participe aux tests ? Profil idéal : utilisateur expérimenté et motivé.</li><li>● Quand doit-on commencer à tester ? Le plus tôt possible pour éviter des anomalies bloquantes.</li><li>● Peut-on utiliser les outils de tests dans une démarche qualité ISO 9000 ? C'est une obligation de démontrer une démarche de tests claire et organisée.</li></ul>
BTD/GL/Tests	5

CENTRALE L Y O N	<h2>Définitions</h2>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● Tester consiste à exécuter un programme dans le but de faire apparaître les failles (anomalies) enfouies dans ce programme.</li><li>● Un test réussi est un test qui a permis de détecter une anomalie inconnue auparavant.</li></ul>
BTD/GL/Tests	6

CENTRALE  
L Y O N

## Aspects économiques

Bertrand DAVID : Génie Logiciel

- Vaut-il mieux investir dans le test et éliminer tous les bogues ou bien accepter leur présence et les éliminer au fur et à mesure ?
  - L'idéal qu'il n'y ait pas de bogues en adoptant une démarche de qualité très stricte
- Coût de l'anomalie
  - Chaque anomalie peut mettre en péril l'activité de l'entreprise, elle est donc proportionnelle aux dommages causés
    - Le développeur peut corriger immédiatement l'anomalie et diffuser la correction
    - L'application ne peut plus être utilisée car les dysfonctionnements sont considérables

BTD/GL/Tests 7

CENTRALE  
L Y O N

## Le coût des tests

Bertrand DAVID : Génie Logiciel

- 33% en développement
- 67 % en maintenance
- Plus on teste tôt, plus on a de chances de trouver à temps les erreurs importantes
  - Le test doit être intégré dans toutes les phases du cycle de développement
- Coûts des bogues détectés pendant le développement
  - Le coût du test est de l'ordre de 20% du coût total du développeur, du chef de projet et des outils spécialisés
  - Dans les phases d'intégration les bogues détectés coûtent 10 fois plus (car plus complexe à trouver et à corriger)
  - Dans la phase de recette on passe à 100 fois plus

BTD/GL/Tests 8

CENTRALE  
L Y O N

## Coût des bogues détectés après le déploiement

Bertrand DAVID : Génie Logiciel

- **Pour l'utilisateur :** coût direct et indirect de l'arrêt d'une partie de l'activité
- **Pour l'éditeur ou le prestataire :** coût de correction, frais d'un nouveau déploiement, paiement de pénalités
  - Il est donc vital d'investir dans les tests et dans des méthodes de développement améliorant la qualité.

BTD/GL/Tests 9

CENTRALE  
L Y O N

## Conséquences juridiques

Bertrand DAVID : Génie Logiciel

- Relations éditeur-utilisateur
- Cas du logiciel sous plastique :
  - l'éditeur ne peut pas garantir que son logiciel soit exempt de toute anomalie et ne peut prédire le contexte d'usage - il se décharge de toute responsabilité directe ou indirecte liée à l'usage du logiciel par la formule :
- Garantie et responsabilités :
  - « LLL » et la documentation qui l'accompagne (consignes d'emploi comprises) sont fournis dans l'état où ils se trouvent et sans aucune garantie. En cas de support défectueux, un autre exemplaire sera délivré par « CCC » sur demande.
  - « CCC » décline toute responsabilité découlant d'un dommage direct ou indirect en relation avec l'utilisation ou l'impossibilité d'utilisation de « LLL », y compris les dommages entraînés par la perte de bénéfices, l'interruption des activités, la perte d'informations et autres, et ce même si « CCC » a été informé de la survenance ou de l'éventualité de tels dommages.
  - Comme l'acheteur n'achète pas le logiciel, mais une licence d'utilisation, l'éditeur retire sa responsabilité sur son exploitation.

=> L'acheteur devient responsable des dommages liés par l'usage d'un produit dont il n'a pas la maîtrise

BTD/GL/Tests 10

CENTRALE L Y O N	<h2>Relations client-SSII (1/2)</h2>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● Les contrats des relations entre SSII et donneurs d'ordres contiennent une rubrique ayant la forme suivante :<ul style="list-style-type: none"><li>• Jeux d'essais en vue de recette<ul style="list-style-type: none"><li>• « FFF » devra fournir des jeux d'essais permettant de s'assurer que le programme mis en œuvre par elle contient bien aux besoins de « CCC »</li><li>• Les jeux d'essais seront sous la responsabilité de « CCC » même si le « FFF » est amené à participer à leur mise au point.</li></ul></li><li>• Réception du programme<ul style="list-style-type: none"><li>• Le logiciel réalisé par « FFF » sera considéré comme conforme après mise en place sur les systèmes de « CCC » et contrôle de la validité à la suite d'un traitement effectué sur les jeux d'essais</li><li>• « FFF » communiquera les résultats des essais à « CCC » qui devra s'assurer qu'ils sont conformes aux prestations demandées.</li><li>• Si les jeux d'essais n'ont pas été remis dans les conditions prévues, ou si « CCC » n'a pas fait de réclamation écrite dans les dix jours qui suivent l'envoi des résultats, les programmes fournis seront considérés comme satisfaisants et acceptés en l'état par « CCC ».</li><li>• La réception des résultats, sans réclamation ou contestation, vaudra recette définitive et la responsabilité ultérieure de « FFF » se trouvera déchargée en l'absence de réserves justifiées et formulées par par « CCC »</li></ul></li></ul></li></ul>
BTD/GL/Tests	11

CENTRALE L Y O N	<h2>Relations client-SSII (2/2)</h2>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>• Registre des anomalies<ul style="list-style-type: none"><li>• Il sera tenu un registre sur lequel « CCC » indiquera toutes les anomalies ou incidents remarqués concernant les logiciels en utilisant le classement qu'elle a défini dans le document</li><li>• Les représentants de « FFF » indiqueront les solutions adoptées pour y remédier, les résultats obtenus et les recommandations nécessaires</li></ul></li><li>• Garantie<ul style="list-style-type: none"><li>• Les logiciels standard éventuellement fournis sont garantis pendant une période de XXX mois à compter de la réception définitive de l'ensemble du système.</li><li>• Cette garantie couvre toute anomalie (bogue) constatée, la remise gratuite de toute nouvelle version de logiciels (release) et la mise à jour (update) gratuite en cas de modification des règles comptables, fiscales ou juridiques mettant en cause la pertinence du logiciel livré.</li><li>• Pendant la durée de la garantie, la maintenance sera assurée par téléphone ou, si nécessaire, sur site dans le cas où « CCC » rencontrerait un problème particulier et ce sans aucun frais pour « CCC ».</li><li>• Les prestations de maintenance sont prévues à titre gratuit pendant la période de garantie.</li></ul></li></ul> <ul style="list-style-type: none"><li>● Il est souhaitable d'inclure dans le CC l'utilisation d'outils</li></ul>
BTD/GL/Tests	12

CENTRALE  
L Y O N

## Loi du 19 mai 1998

Bertrand DAVID : Génie Logiciel

- Le 25 juillet 1985 les institutions européennes arrêtaient une directive relative au rapprochement de législations des Etats membres en matière de responsabilité du fait des produits défectueux.
- 13 ans après, la France vient finalement d'intégrer cette directive dans sa législation par la loi du 19 mai 1998. Le Code civil comprend désormais 18 nouveaux articles instituant un régime de responsabilité sans faute du fait des produits défectueux.

Nouvelle loi sur la responsabilité du fait des produits défectueux :

- Le producteur est responsable du dommage causé aux personnes ou aux biens par un défaut de son produit et ce pendant dix ans à compter de la mise en circulation du produit.
- Question : Logiciel est-il un produit ?

BTD/GL/Tests 13

CENTRALE  
L Y O N

## Le cas des logiciels

Bertrand DAVID : Génie Logiciel

- **Sécurité et défaut**
  - Le bogue d'un logiciel à l'origine d'une perte de données constituera-t-il un défaut causant un dommage aux biens parce qu'il n'offre pas la sécurité à laquelle on peut légitimement s'attendre ?
  - Cas d'un logiciel intégré dans un appareil : « composant » du produit.
  - La notion de bogue fait référence à un dysfonctionnement par rapport à des exigences définies au préalable plutôt qu'à la notion de sécurité.
  - Il faut donc préciser les conditions d'usage offrant le maximum de sécurité pour que cet article de la loi ne puisse s'appliquer.

BTD/GL/Tests 14

CENTRALE L Y O N	<h2>Risques indirects</h2>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● <b>Installation défectueuse</b><ul style="list-style-type: none"><li>• Problèmes induits par l'installation d'une nouvelle application sur les applications déjà installées.</li></ul></li> <li>● <b>Corruption de données</b><ul style="list-style-type: none"><li>• Un utilisateur altère des donnée d'un autre, notamment dans le cas d'utilisation en réseau.</li><li>• Le concepteur de l'application sera tenu pour responsable de cette perte.</li></ul></li></ul>
BTD/GL/Tests	15

CENTRALE L Y O N	<h2>Objectifs et limites du test</h2>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● <b>Les conséquences économiques et financières démontrent la nécessité de tester les applications développées.</b></li> <li>● <b>Peut-on tester complètement une application ? NON</b></li> <li>● <b>Objectifs :</b><ul style="list-style-type: none"><li>• Rechercher les anomalies présentant le plus grand risque pour l'application</li><li>• Se concentrer sur les types d'anomalies les plus bloquantes</li></ul></li></ul>
BTD/GL/Tests	16

CENTRALE  
L Y O N

## Activités de test

Bertrand DAVID : Génie Logiciel

- Démarche globale
  - Préparation
  - Planification
  - Exécution
  - Analyse
- Situations
  - Environnements centralisés
  - Cycle de V pour des grands projets
  - Applications orientées objet
  - Applications client-serveur : Prototypage incrémental (RAD Rapid Application Development)
  - Applications Web - Intranet
  - Progiciels de gestion (ERP)
  - Applications multimédia

BTD/GL/Tests 17

CENTRALE  
L Y O N

## Catégories de tests

Bertrand DAVID : Génie Logiciel

- Boite blanche (développement et validation de performances)
- Boite noire (recette d'application)
- Boite grise (conversions)

BTD/GL/Tests 18

CENTRALE L Y O N	<h2>Boite blanche</h2> <p>(développement et validation de performances)</p>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● <b>Créer des jeux d'essais validant le code de l'application</b><ul style="list-style-type: none"><li>• Tous les chemins ont été parcourus au moins une fois</li><li>• Les conditions vrai/faux ont été validées</li><li>• Les boucles sont exécutées correctement</li><li>• Les valeurs limites sont correctement définies</li><li>• Il n'y a pas de fuite mémoire</li></ul></li></ul>
BTD/GL/Tests	19

CENTRALE L Y O N	<h2>Boite noire</h2> <p>(recette d'application)</p>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● <b>Tests boite noire et non-régression en phase de recette</b><ul style="list-style-type: none"><li>• Il s'agit de tests fonctionnels</li><li>• Le respect des exigences</li><li>• Détection de fonctions incorrectes ou manquantes</li><li>• Les incohérences au niveau de l'interface</li><li>• Les erreurs d'accès aux données</li><li>• Les problèmes de performance de l'application</li><li>• Les conditions initiales ou finales d'une fonction</li></ul></li></ul> <p style="text-align: center;">L'intérêt principal des tests « boite noire » est qu'ils valident les fonctionnalités du logiciel plus que la qualité du développement. Ils sont donc plus proches des préoccupations quotidiennes des utilisateurs.</p>
BTD/GL/Tests	20

CENTRALE  
L Y O N

## Boite grise (conversions)

Bertrand DAVID : Génie Logiciel

- **Combinaison des tests « boîte blanche » et « boîte noire » dans le cas des conversions (an 2000, euro, ...)**

Deux étapes :

- Phase d'analyse : boîte blanche - pour comprendre
- Phase de réalisation : boîte noire - pour valider

BTD/GL/Tests 21

CENTRALE  
L Y O N

## Tests unitaires et tests d'intégration

Bertrand DAVID : Génie Logiciel

- **Tests unitaires : composants individuels**
  - Concepts spécifiques
  - Driver de test : logiciel spécifique définissant le cadre de test
  - Bouchon (stub) : module remplaçant les modules dépendant de l'unité qui doit être testée
- **Trois approches :**
  - Approche descendante (top/down)
  - Approche ascendante (bottom/up)
  - Approche modules isolés

BTD/GL/Tests 22

CENTRALE  
L Y O N

Bertrand DAVID : Génie Logiciel

## Approche descendante (top/down)

- L'approche top/down consiste à tester chaque unité indépendamment de chaque unité appelante, en partant du haut de la hiérarchie. Chaque module appelé est remplacé par un bouchon.
- Avantages :
  - On effectue simultanément des tests unitaires et des tests d'intégration
  - On valide dès le début les fonctions les plus importantes de l'application, car ces tests sont basés sur les besoins fonctionnels
  - La structure de ces tests pourra être réutilisée dans les tests de non-régression
- Inconvénient
  - Utilisation de bouchons, dont la mise en œuvre peut être complexe.

BTD/GL/Tests 23

CENTRALE  
L Y O N

Bertrand DAVID : Génie Logiciel

## Approche ascendante (bottom/up)

- Les tests bottom/up sont basés sur l'approche inverse. On commence à tester les modules élémentaires et on remonte dans la hiérarchie
- On utilise pour cette méthode les drivers de test et non plus les bouchons
- Avantage :
  - Utilisation de fonctionnalités détaillées et utilisation des résultats des tests dans les niveaux supérieurs
- Inconvénients :
  - Les modules unitaires sont souvent développés plus tard que les niveaux supérieurs
  - Le contrôle sur les tests devient plus complexe
  - Les relations entre les modules peuvent devenir complexes dans des applications graphiques ou web.

BTD/GL/Tests 24

CENTRALE L Y O N	<h2>Approche modules isolés</h2>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● Le module est testé indépendamment des autres et on combine les deux approches top/down et bottom/up.</li><li>● On remplace ainsi chaque module appelant par un driver de test et chaque module appelé par un stub.</li><li>● Il est ainsi facile de tester chaque module, quels que soient la complexité de l'application et son impact sur les autres modules.</li><li>● En revanche, on perd la dynamique de l'application, et les tests d'intégration devront être conçus et réalisés indépendamment des tests unitaires</li></ul>
BTD/GL/Tests	25

CENTRALE L Y O N	<h2>Tests de performance</h2>
Bertrand DAVID : Génie Logiciel	<ul style="list-style-type: none"><li>● Temps de réponse dans les conditions normales</li><li>● Temps de réponse dans les conditions extrêmes<ul style="list-style-type: none"><li>• Volume</li><li>• Conditions extrêmes d'exploitation (charge utilisateur, charge réseau, manque de périphériques, ...)</li></ul></li><li>● Outils :<ul style="list-style-type: none"><li>• Simuler les utilisateurs virtuels</li><li>• Enregistrement des scripts de simulation</li><li>• Exécution des scénarios de simulation des utilisateurs</li></ul></li></ul>
BTD/GL/Tests	26

CENTRALE  
L Y O N

## Ordonnancement des tests

- Planification des tests
- Estimation des ressources
- Approche modules isolés
- Tests unitaires
- Tests d'intégration
- Tests système et tests fonctionnels
- Tests d'installation
- Tests de recette
- Bêta tests

Bertrand DAVID : Génie Logiciel

BTD/GL/Tests

27