

IBM Software Group

Industrialiser la chaîne complète de fabrication

1ère partie - Les bénéfices de la solution logicielle IBM VisualAge Pacbase / Rational

Rational software

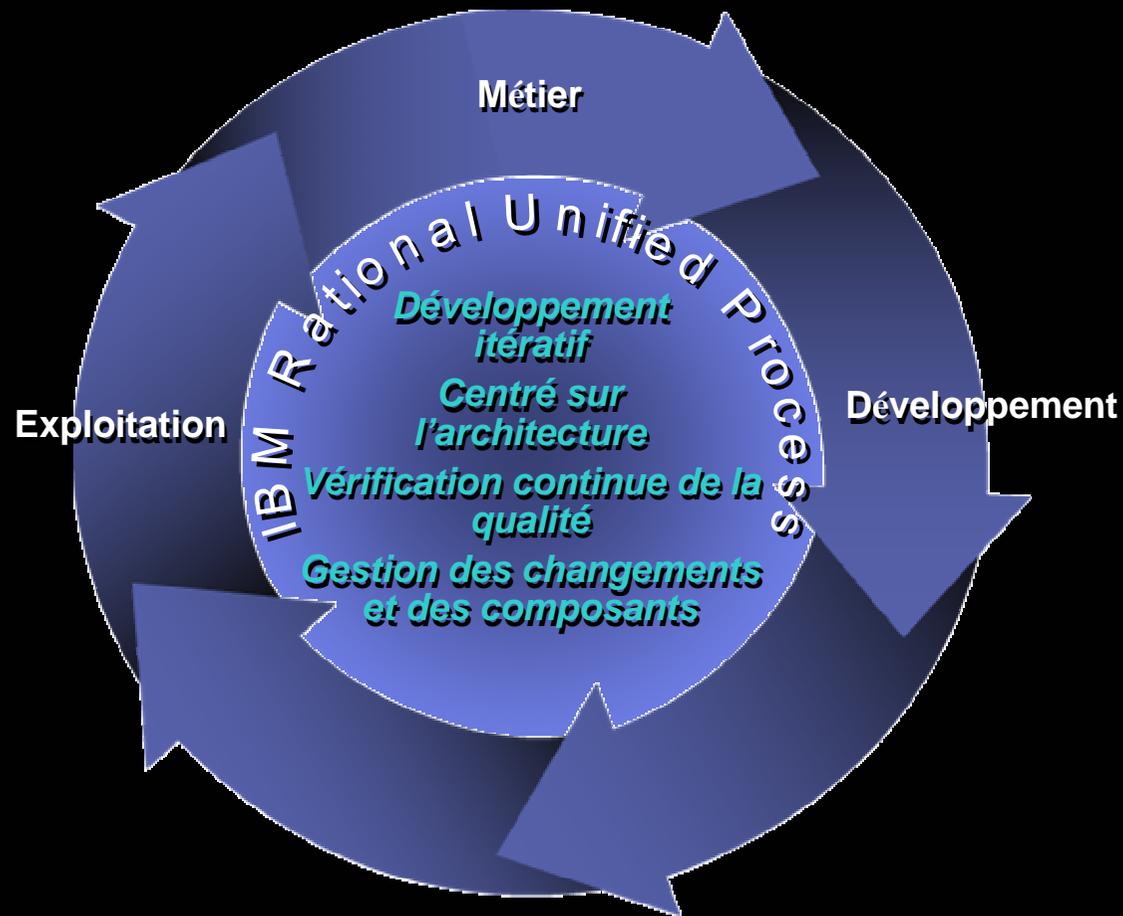
@business on demand software

Fernand Bonaguidi – fernand.bonaguidi@fr.ibm.com

Serge Bonnaud – serge.bonnaud@fr.ibm.com

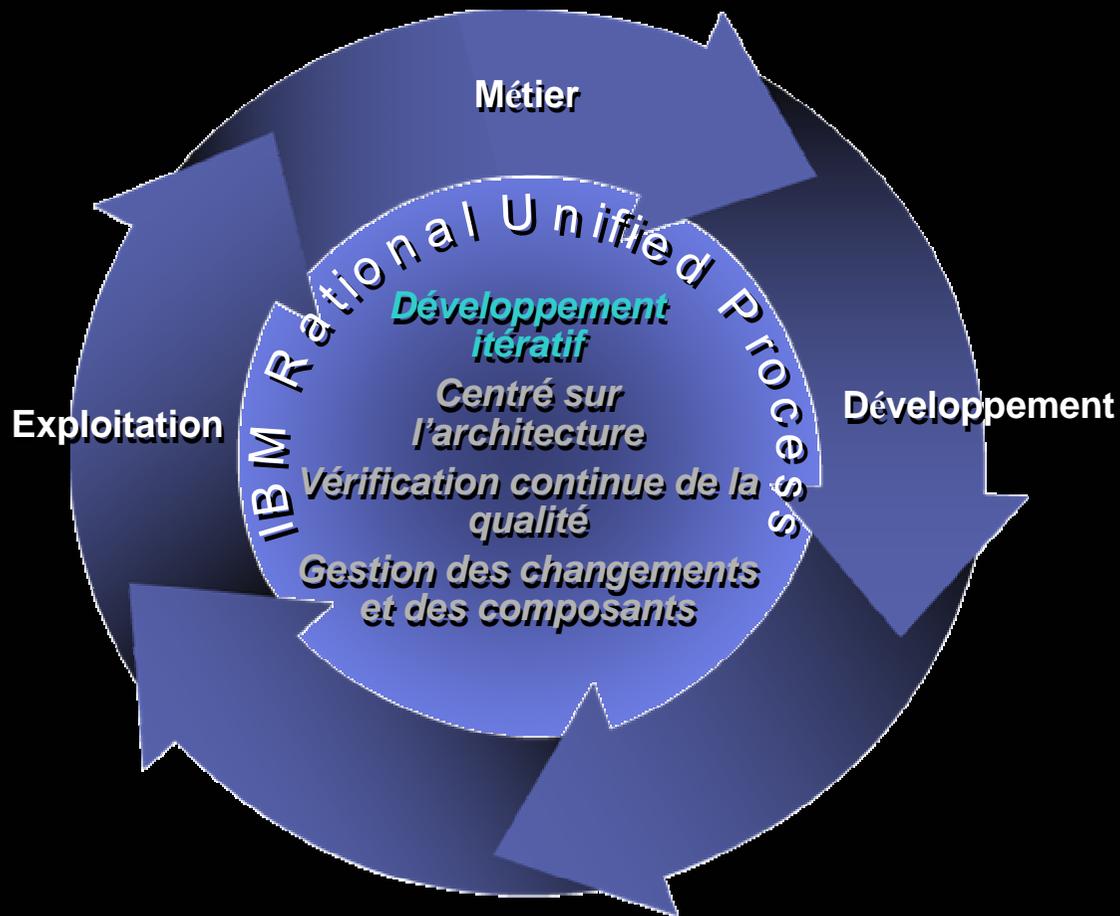
Quatre impératifs pour réussir vos développements

Un processus de développement guidé par le métier

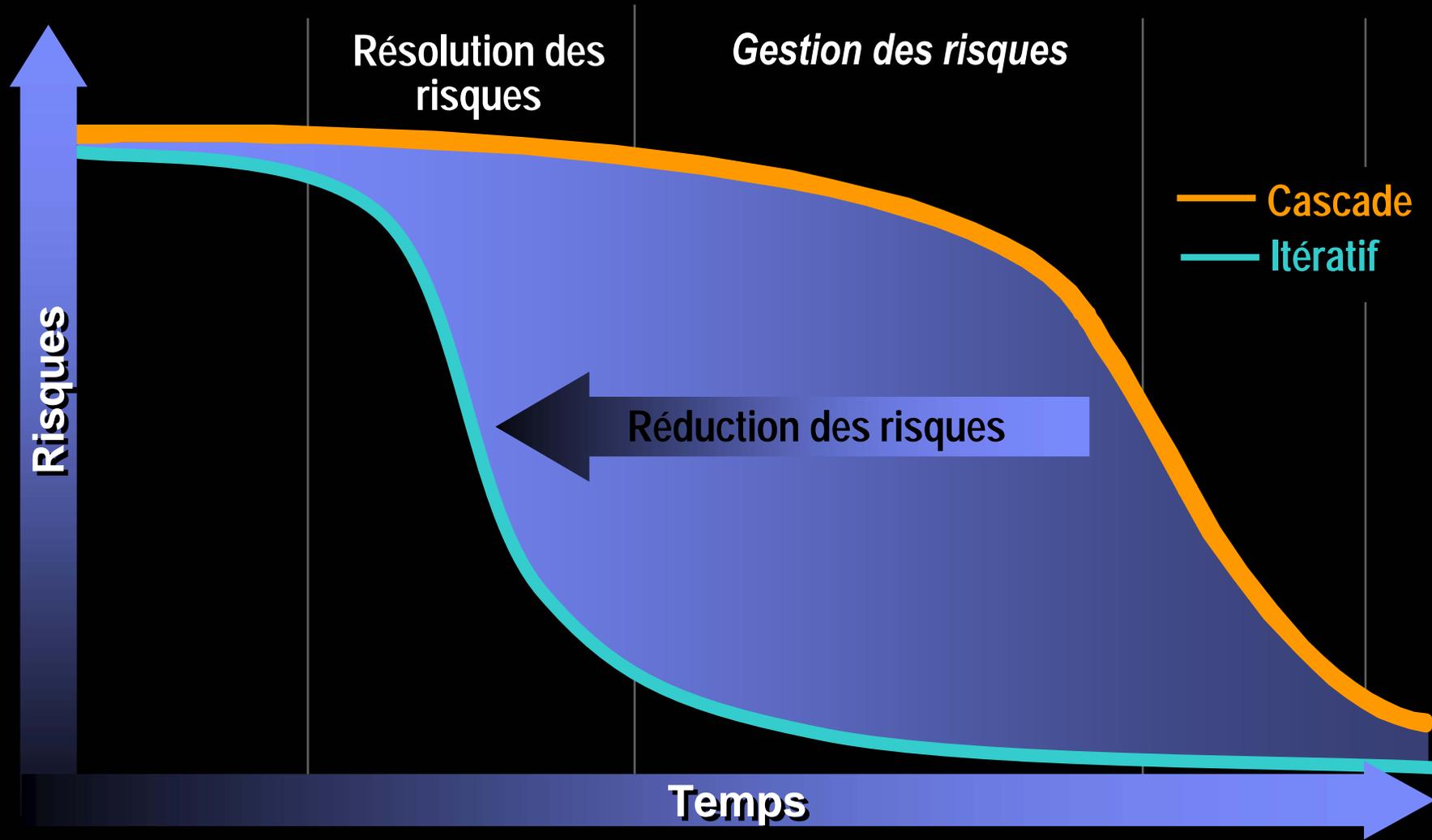


Quatre impératifs pour réussir vos développements

Un processus de développement guidé par le business

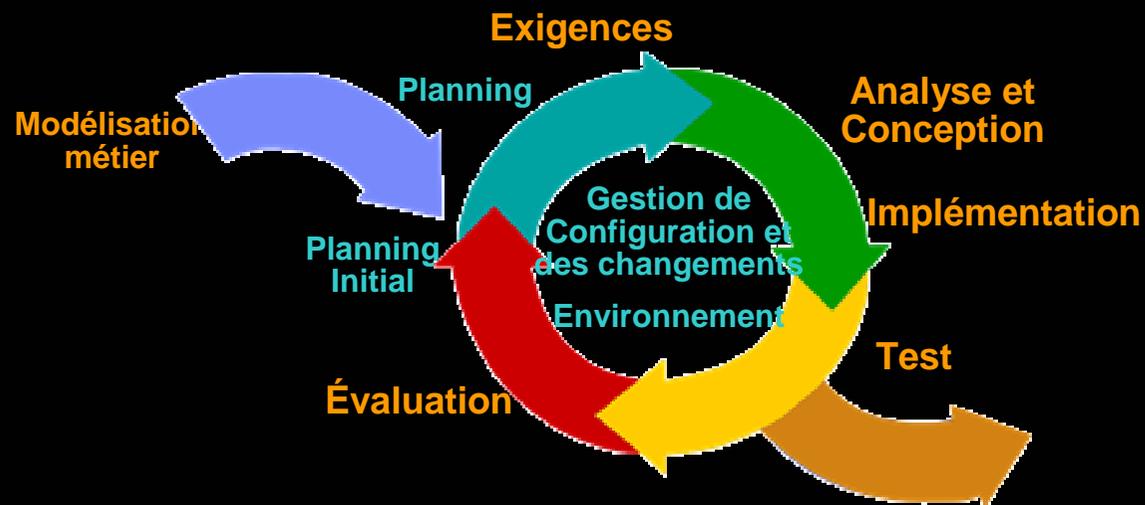
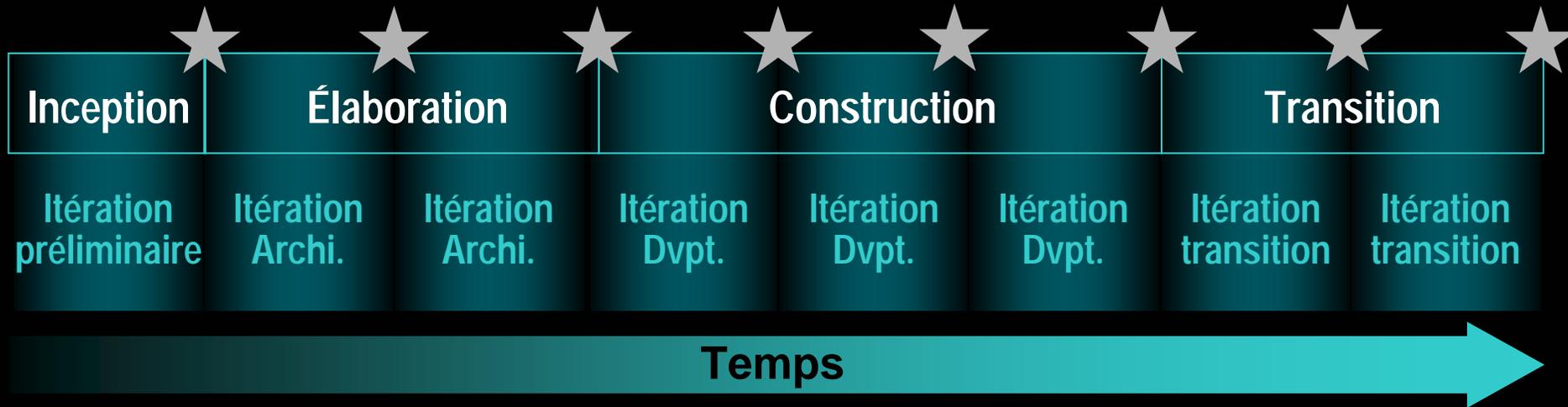


S'attaquer aux risques importants au plus tôt

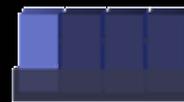
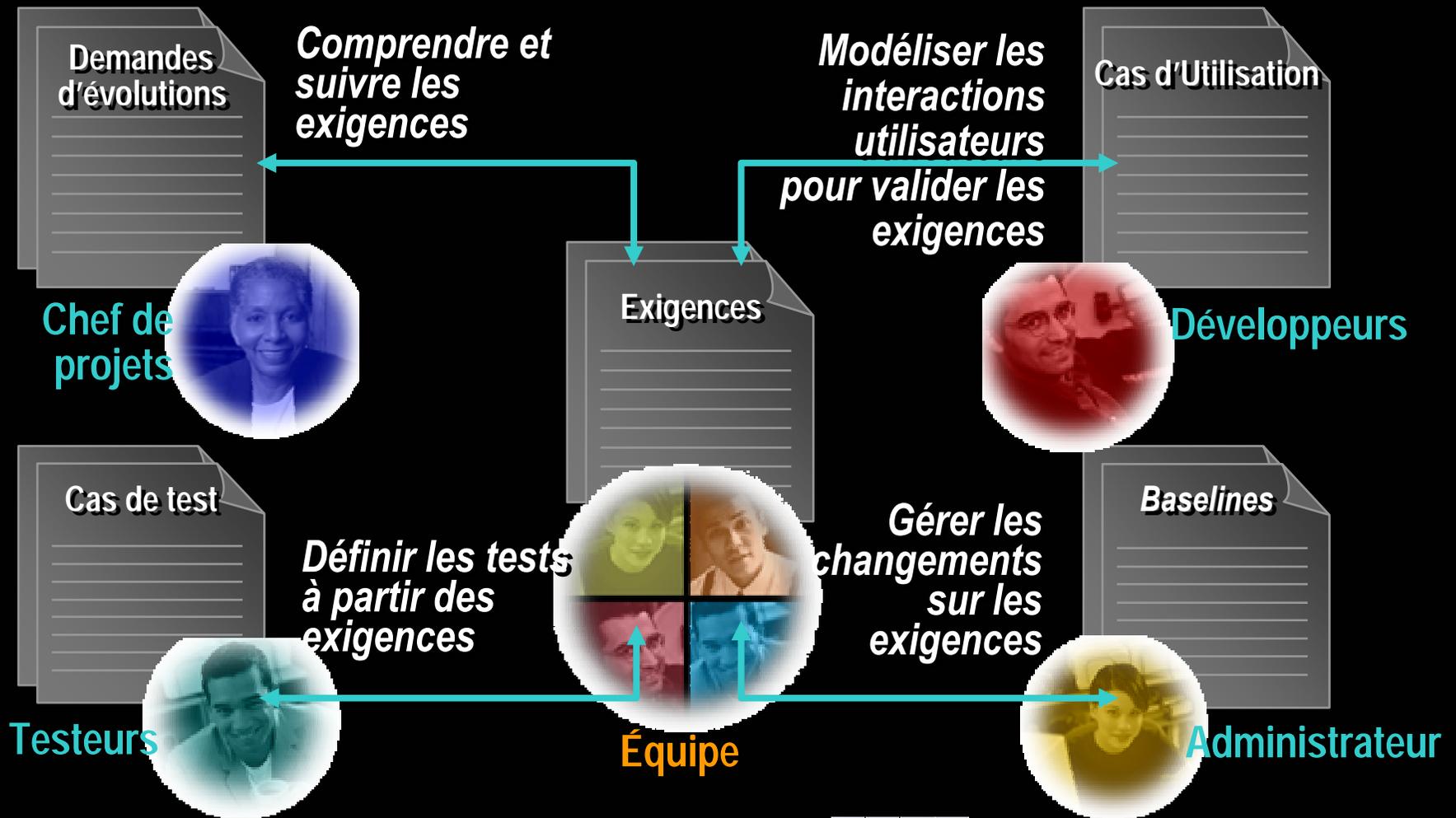


Itération et gestion des risques

Versions Exécutables



Les exigences pilotent le développement itératif



IBM Rational RequisitePro



Définir le périmètre de l'itération

Exigences	Priorité	Difficulté	Risque	Stabilité
EXG1: Sauvegarder / restaurer les critères de tri et de filtrage.	Assez haut	Bas	Bas	Haut
EXG2: Capacité de sauvegarder un document Requisite comme document Word.	Assez haut	Bas	Bas	Haut
EXG3: Capacité de voir les besoins éliminés dans une fenêtre.	Moyen	Assez haut	Moyen	Moyen
EXG4: Support des attributs du type de données Monnaie.	Moyen	Moyen	Assez bas	Moyen
EXG5: Support du type de document "Tout" (offre un moyen facile de définitions d'attributs communs à travers différents types de documents).	Assez haut	Moyen	Moyen	Assez haut
EXG6: Capacité de sélectionner un besoin visualisé et aller dans un document Word.	Assez haut	Moyen	Moyen	Assez haut
EXG7: Afficher un attribut de besoin dans le texte du document des besoins.	Moyen	Moyen	Moyen	Assez haut
EXG8: Nouvel assistant projet.	Assez haut	Haut	Assez haut	Moyen
EXG9: Création rapide d'un besoin (éviter le dialogue sur le besoin à la création).	Assez haut	Assez bas	Assez bas	Haut
EXG10: Autosauvegarde du projet (archives projet).	Moyen	Assez bas	Moyen	Moyen
EXG11: Modifier un ou plusieurs attributs pour un ensemble sélectionné de besoins.	Moyen	Assez haut	Moyen	Moyen
EXG12: Capacité de cloner la structure d'un projet pour permettre aux utilisateurs de créer des projets nouveaux à partir des anciens.	Haut	Moyen	Moyen	Bas
EXG13: Amélioration performance impression, identification des besoins.	Haut	Assez haut	Moyen	Assez haut
EXG14: Portabilité Windows XP.	Haut	Moyen	Haut	Haut



Mesurer la couverture des test par itération

Webshop - Rational TestManager - [Test Input]

File View Reports Tools Window Help

Test Case Reports

- Test Case Distribution
- Requirements Coverage
- Test Case Results Distributi...
- Test Case Results Trend
- Performance Testing Reports
- Compare Performance
- Performance
- Response vs Time
- Command Status
- Command Usage
- Listing Reports
- Build Listing
- Computer List Listing
- Computer Listing
- Configuration Listing
- Iteration Listing
- Log Listing
- Script Listing
- Session Listing
- Suite Listing
- Test Plan Listing
- User Listing

	Implemented Test Cases
Rational Project - RequisitePro Project	56
UC1 Browse Catalog	10
UC1.1 Browse CDs and Filter	0
UC1.2 Display CD homepage	0
UC1.3 Option to Browse Catalog	1
View Cat Page Basic	YES
UC1.4 Recommended Selection	0
UC1.5 Catalog Link	1
CatalogMain - Verify Sorting	YES
UC1.6 List available CDs	0
UC1.7 CD Details through a link	0
UC1.8 CD Details	1
UC1.8.1 CD Cover	1
Verify CD Cover Images	YES
UC1.8.2 CD Title	0
UC1.8.3 Performer and Orchestra	0
UC1.8.4 Review of the CD	0
UC1.9 Flow termination	0
UC1.10 Search by Selected Criteria	1
Catalog Search Shell	YES
UC1.11 Catalog Search	5
UC1.11.1 Composers	2
Bad Composer	YES
Search Partial Composer	YES
UC1.11.2 Composition	2
Search Partial Composition	YES
Bad Composition	YES
UC1.11.3 Performer	0
Multiple Sorts	YES
UC1.12 Initial entry	0
UC1.13 System Display	0
UC1.15 Pre-conditions	0
Close Browser	YES
UC2 Shop for CD	8
UC2.1 Select & Purchase	1
CatalogMain - Select Bach Concertos	

Exigence

Résultat de test

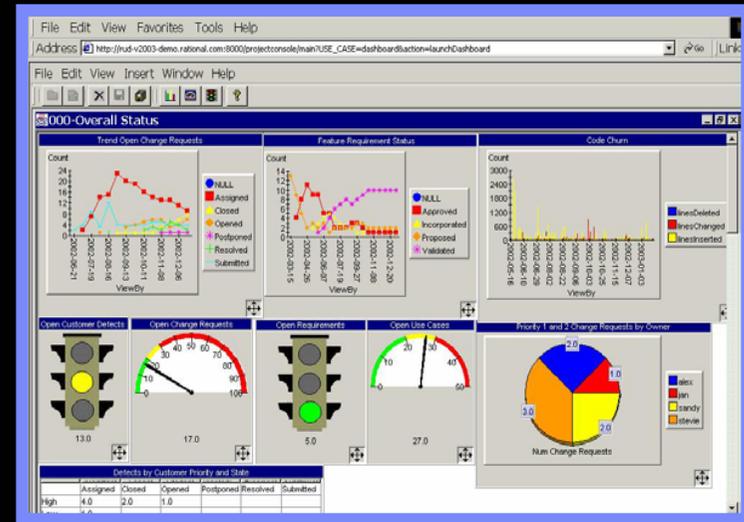
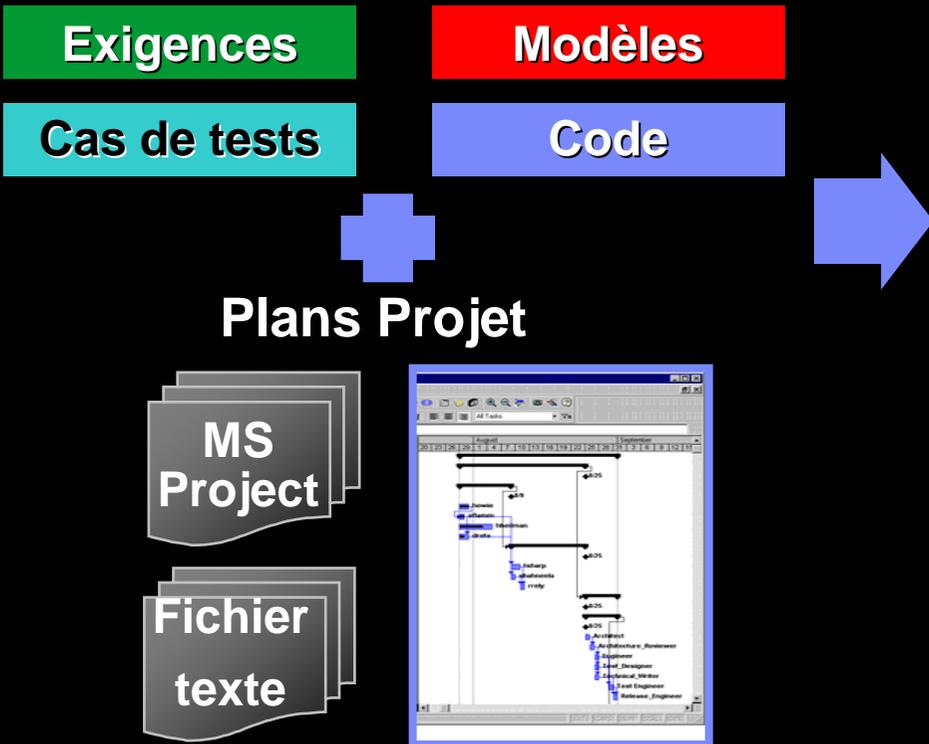
Test

Mesure de l'avancement et de la qualité à chaque itération

Un cycle de développement en boucle fermée

Collecte automatiquement les résultats dans les différents outils de développement

Tableau de bord de projet

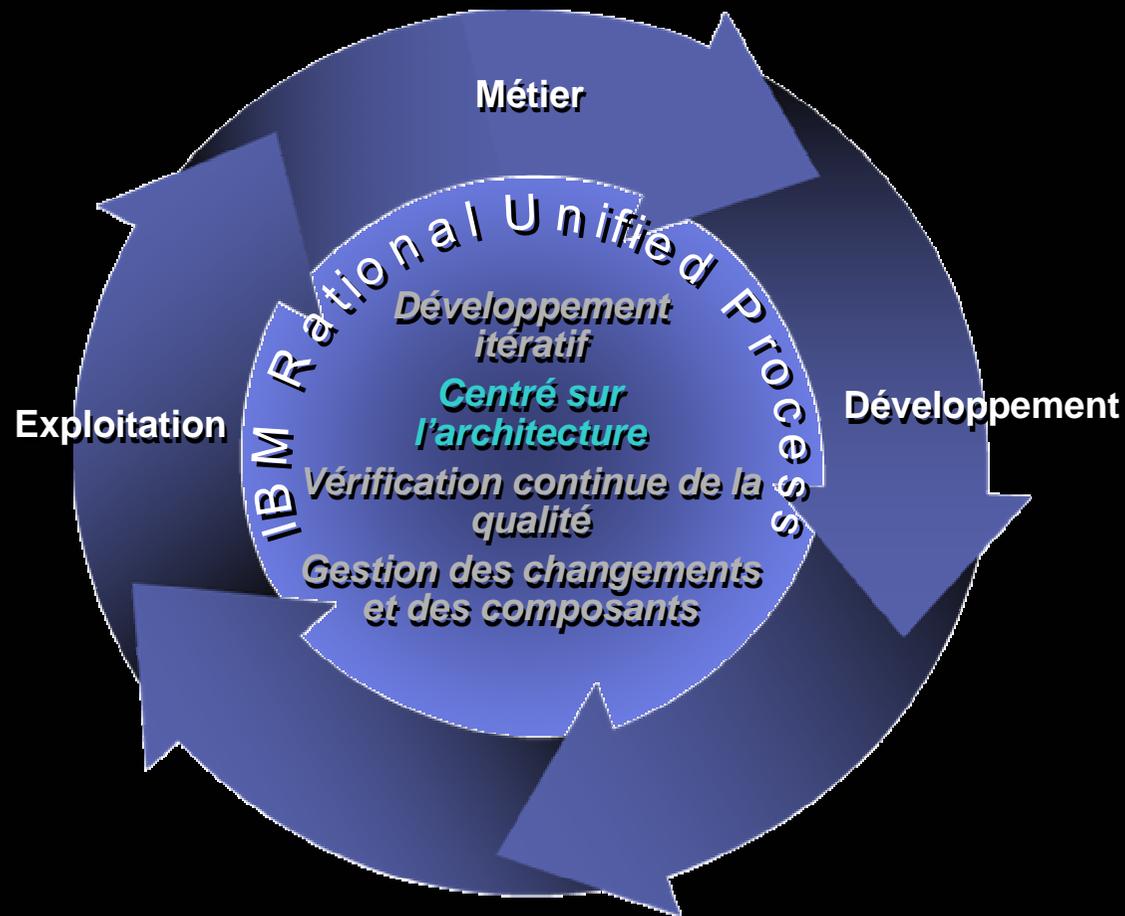


IBM Rational ProjectConsole



Quatre impératifs pour réussir vos développements

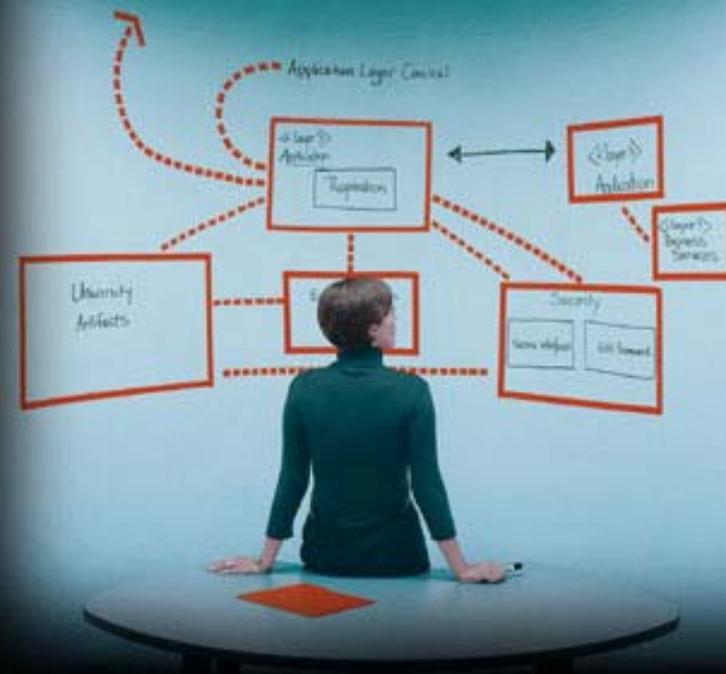
Un processus de développement guidé par le business



Moderniser le parc applicatif

- **Intégrer** en pensant flexibilité en cas de changement
- **Garantir** de bonnes performances
- **Conserver** ou améliorer l'Expérience Utilisateur
- **Gagner** du temps en réutilisant des composants ou des techniques d'intégration basées sur des standard
- **Préserver** l'intégrité et les investissements sur l'existant

Intégration de systèmes Existants

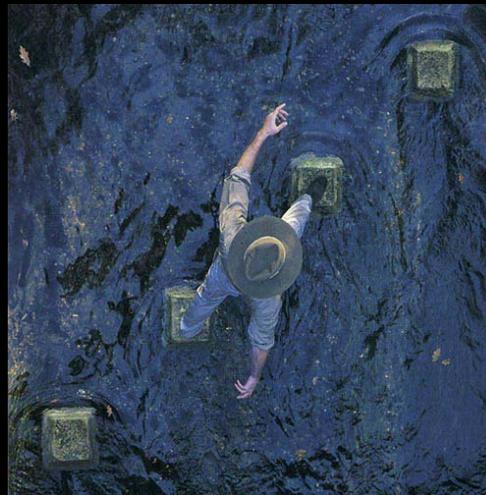


Intégrer les systèmes nécessite la maîtrise de l'architecture existante



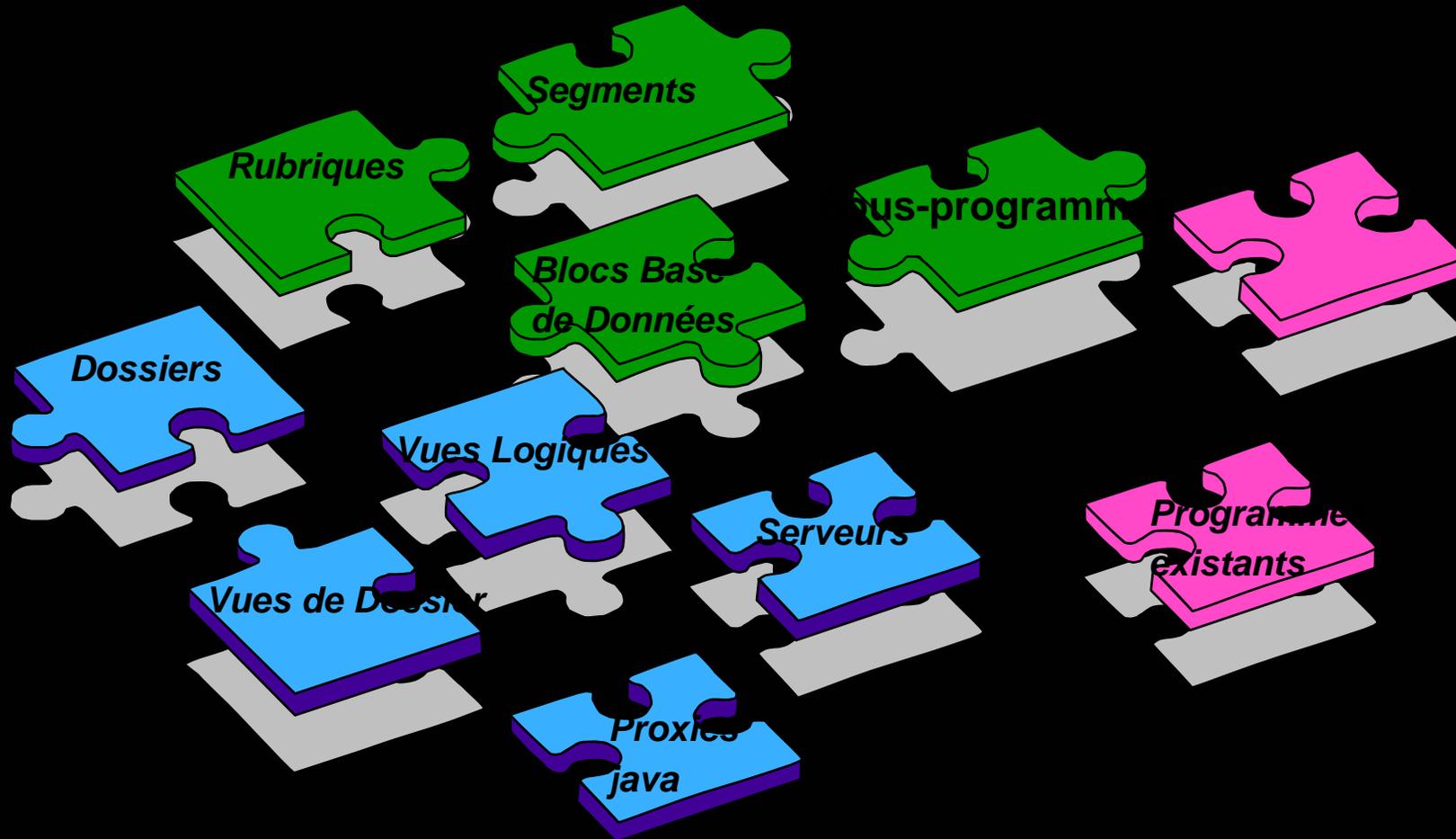
Démarche de modernisation

1. Analyse des actifs réutilisables existant
2. Choix du style de transformation
3. Définition des exigences à implémenter
4. Modélisation visuelle et génération VisualAge Pacbase
5. Test et déploiement de l'application finale



1. Analyse des actifs réutilisables : Entités VisualAge Pacbase

Réutiliser, Adapter, Transformer



1. Analyse des actifs réutilisables : démarche

■ Cartographie de l'existant

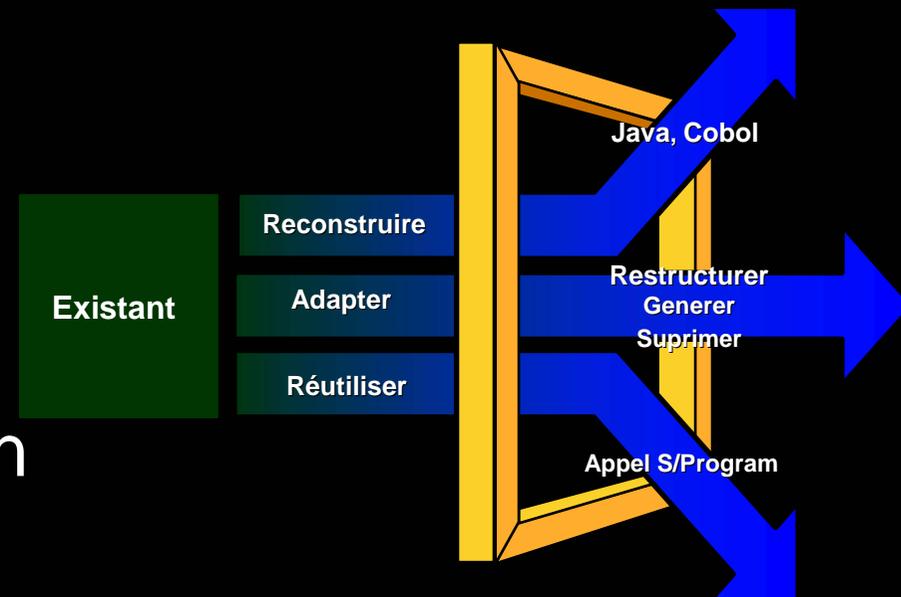
- Recenser
- Qualifier

■ Estimer les gains de réutilisation

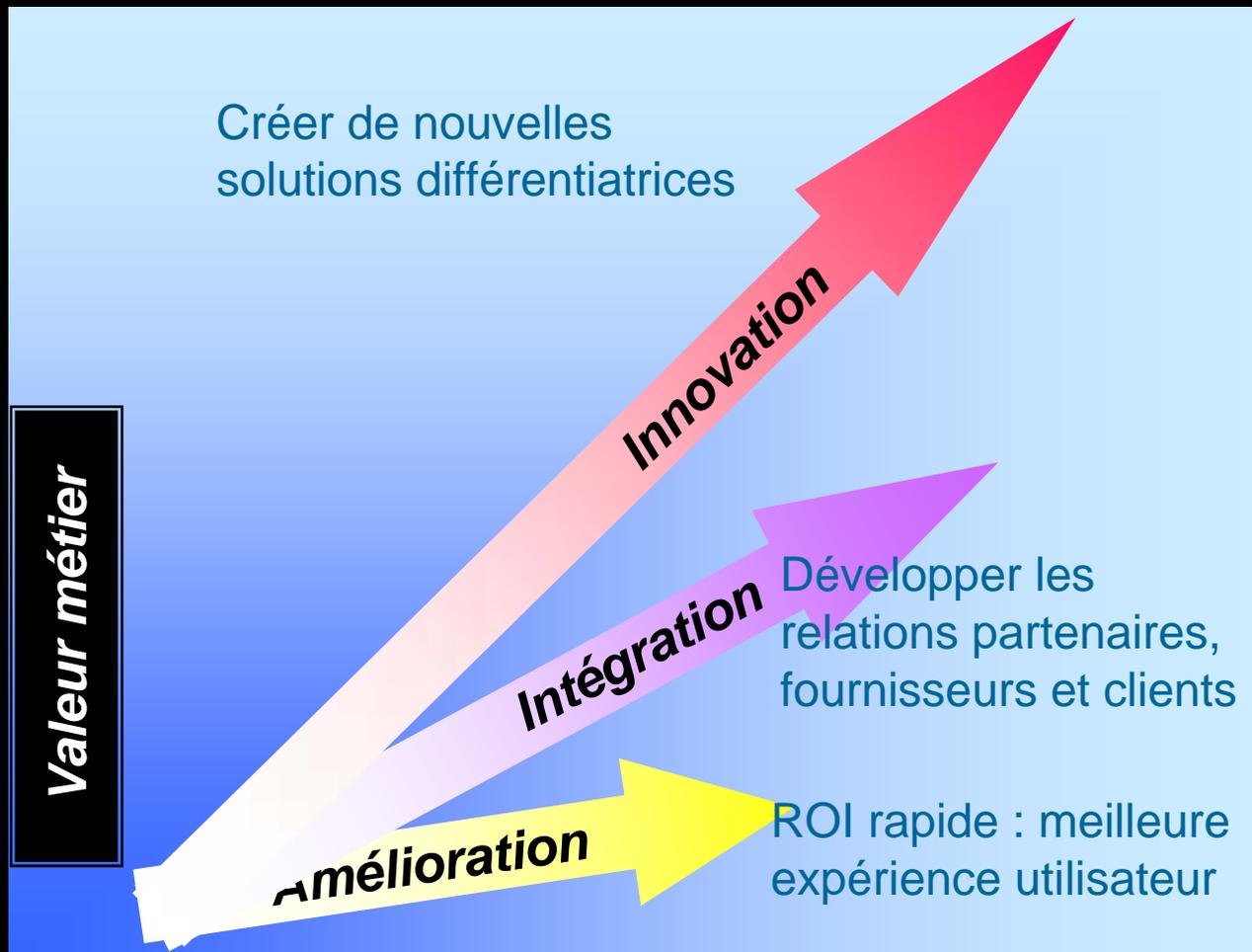
- Affecter un niveau qualité
- Définir les charges

■ Démarrer rapidement un nouveau projet

- ▶ Préparer la conception
- ▶ Préparer les équipes



2. Choix du style de transformation



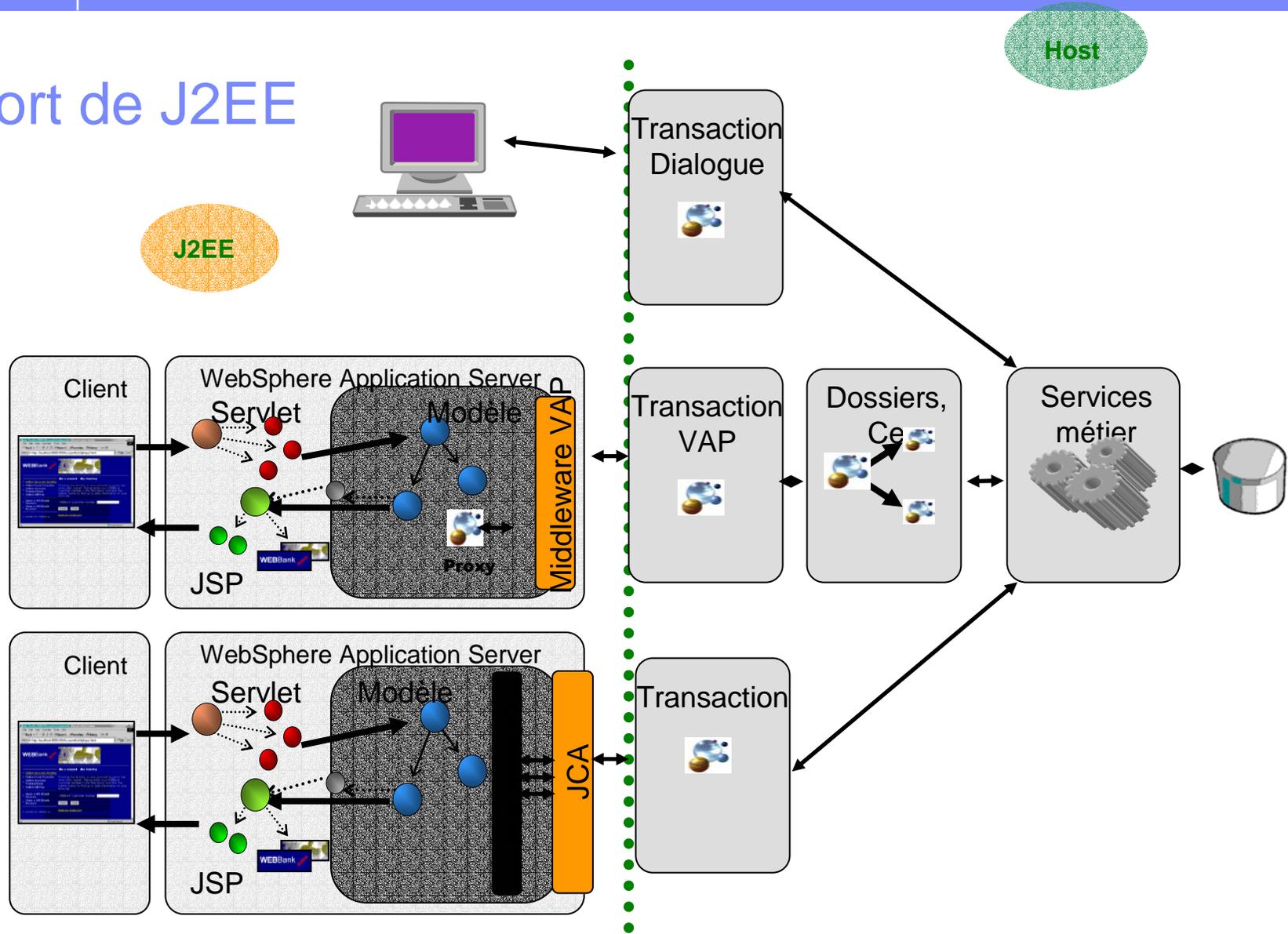
Rendre **modulaire** et réutilisable certains applications critiques afin de pouvoir moderniser les processus métier principaux.

Intégrer les applications et développer de nouvelles interfaces utilisateurs. Moins risqué que la stratégie de remplacement total.

Améliorer l'interface utilisateur des applications critiques pour réduire les coûts de formation et améliorer l'expérience utilisateur.

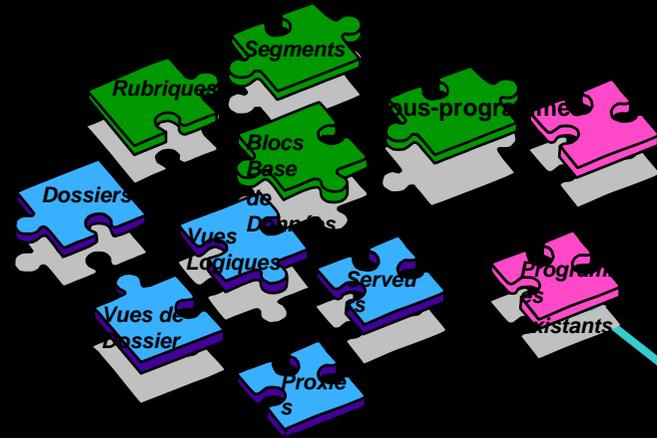


Apport de J2EE



3. Définition des exigences à implémenter

Redécouvrir les cas d'utilisation à posteriori



Entités VisualAge Pacbase

Comment est ce utilisé ?

Que fait le système ?



Analyste

Manuel, defects, logs



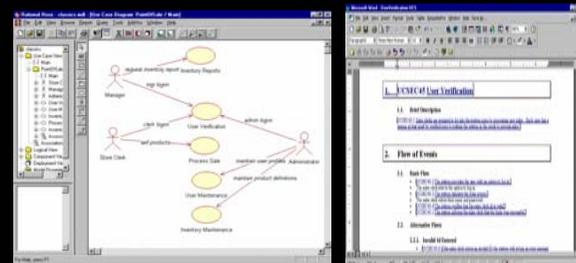
Utilisateurs, support

Quelles exigences ?



Documents, données

Cas d'Utilisation



Qu'est-ce qu'un cas d'utilisation ?

“...une séquence d'actions réalisée par le système et qui produit un résultat observable pour un acteur individuel du système.”

Rational Unified Process

“...décrit les exigences systèmes dans leur contexte d'utilisation.”

Schneider & Winters, Applying Use Cases: A Practical Guide

Avant les cas d'utilisation

- ◆ Le système doit permettre l'insertion de cartes de crédit
- ◆ Le système doit permettre l'impression de reçus
- ◆ Le système identifie les utilisateurs par un mot de passe numérique

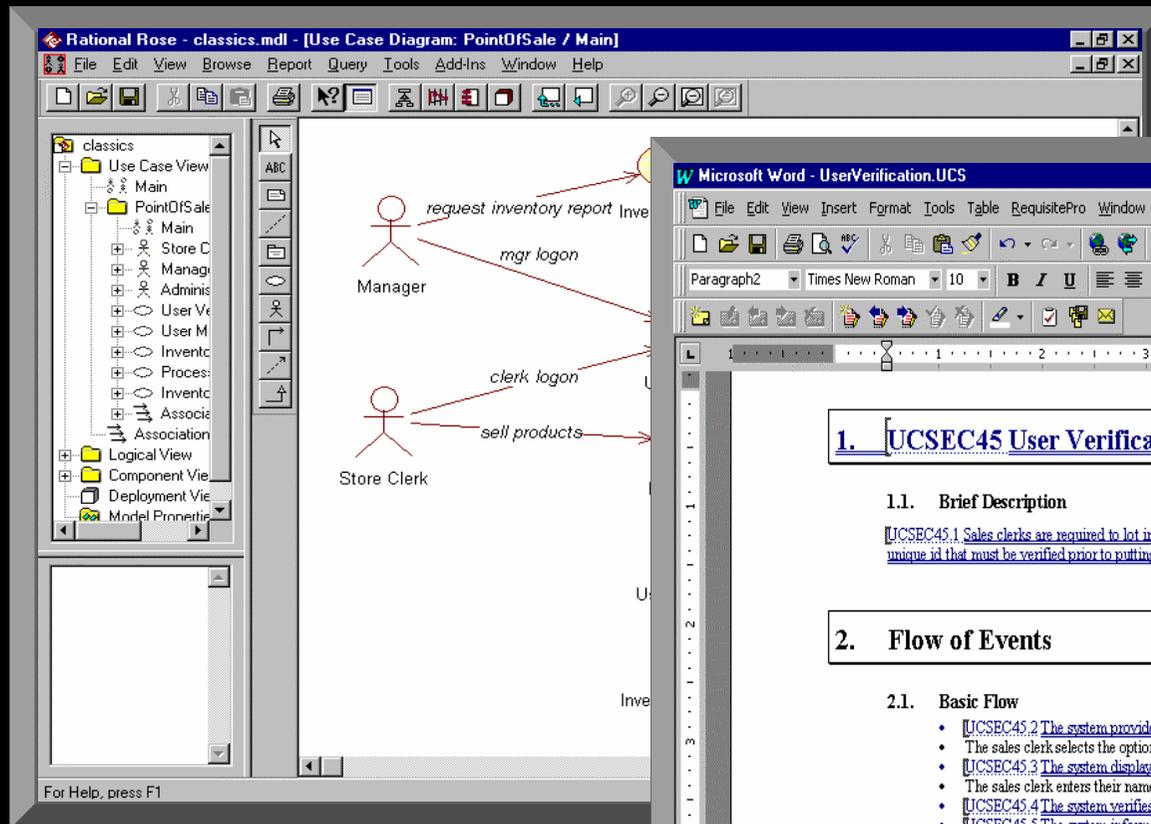
Après les cas d'utilisation

- ◆ L'utilisateur insère la carte
- ◆ Le système demande le code
- ◆ L'utilisateur entre son code
- ◆ Si le code est valide, le système propose : retrait ou consultation
- ◆
- ◆ L'utilisateur demande à avoir un reçu



Les cas d'utilisation sont ...

graphiques



Mais aussi
textuels

1. [UCSEC45 User Verification]

1.1. Brief Description

[UCSEC45.1 Sales clerks are required to log into the system prior to processing any sales. Each user has a unique id that must be verified prior to putting the system in the mode to process sales.]

2. Flow of Events

2.1. Basic Flow

- [UCSEC45.2 The system provides the user with an option to log in.]
- The sales clerk selects the option to log in.
- [UCSEC45.3 The system displays the login screen.]
- The sales clerk enters their name and password.
- [UCSEC45.4 The system verifies that the sales clerk id is valid.]
- [UCSEC45.5 The system informs the sales clerk that the login was successful.]

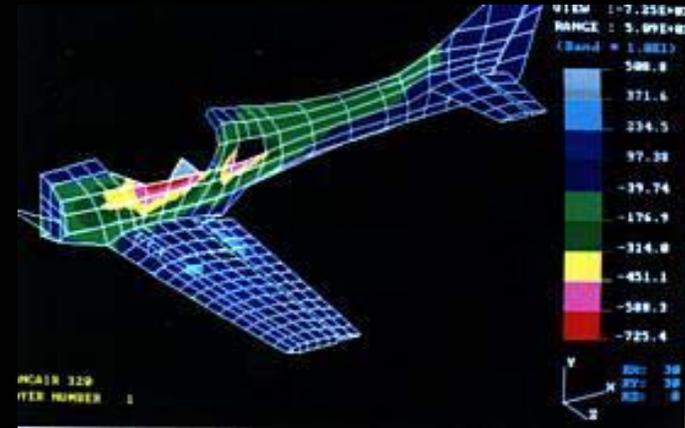
2.2. Alternative Flows

2.2.1. Invalid Id Entered

- [UCSEC45.6 If the sales clerk enters an invalid ID the system will return an error message]

Pourquoi modéliser ?

- Simplifier le développement
- Communiquer sans ambiguïté
- Comprendre la structure
- Élever le niveau d'abstraction
- Faire des prévisions
- Formaliser le problème et sa solution
- Piloter la fabrication finale
- Analyser l'impact

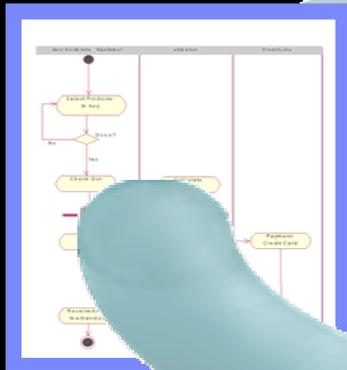


La modélisation permet de maîtriser l'architecture

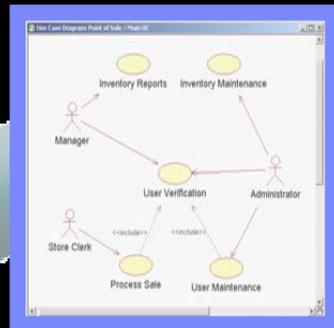


Que modéliser ?

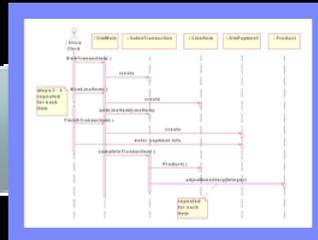
Modélisation
métier



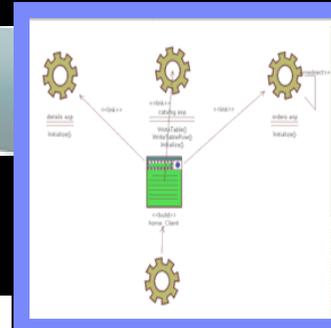
Modélisation
des exigences



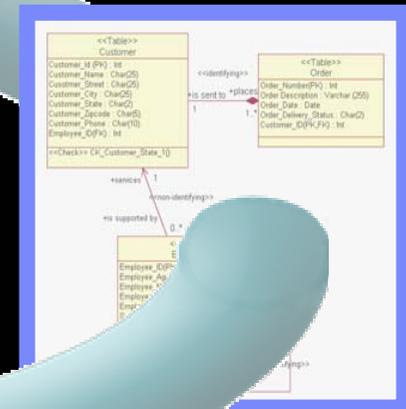
Modélisation
de l'application



Modélisation
du web



Modélisation
des données



UNIFIED!

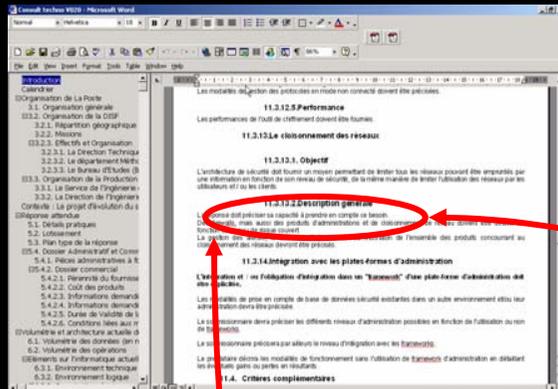
*Ne pas oublier la modélisation libre

Un langage unique pour l'équipe

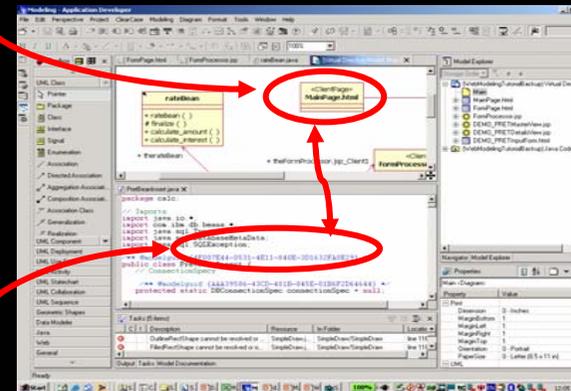


Modélisation, exigences et test : quelle relation ?

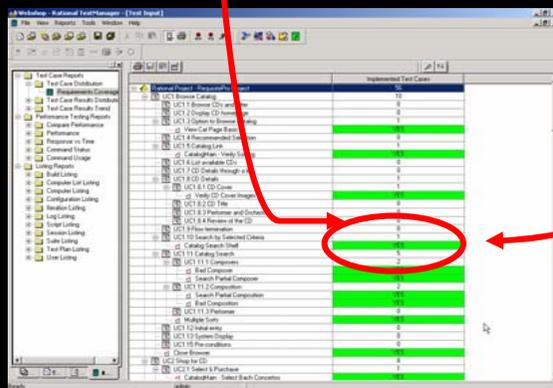
La spécification



La conception visuelle



Le code source



La plan de test



4. Modélisation et génération dans VisualAge PacBase

Des cas d'utilisation aux entités VisualAge Pacbase

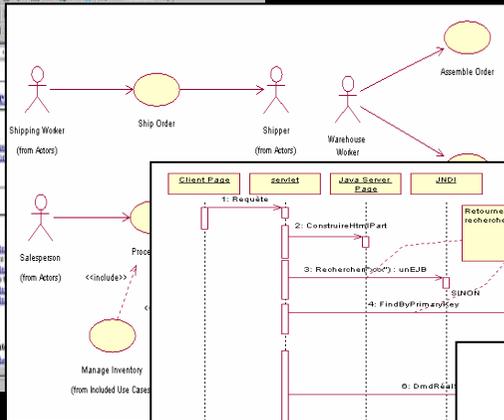
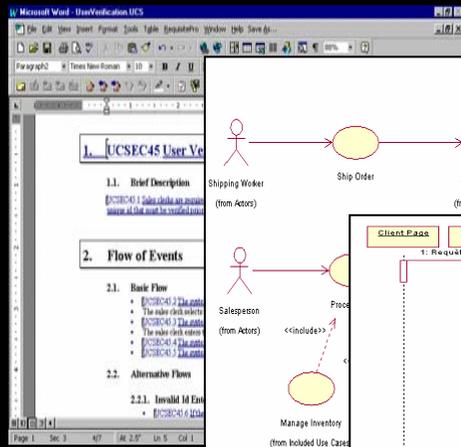
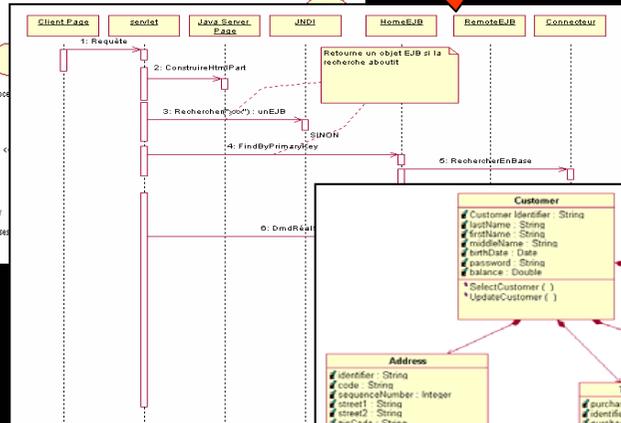
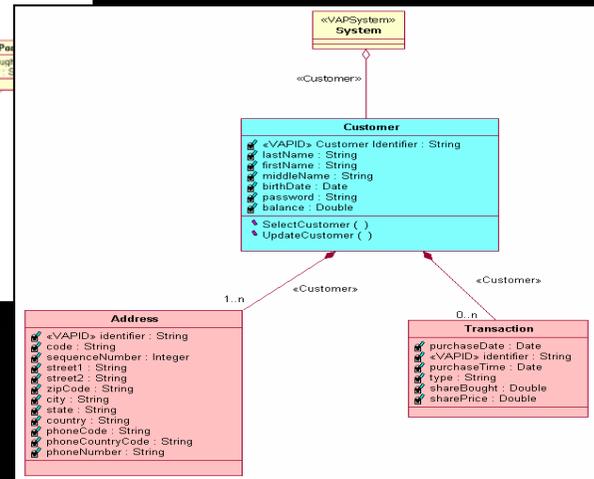
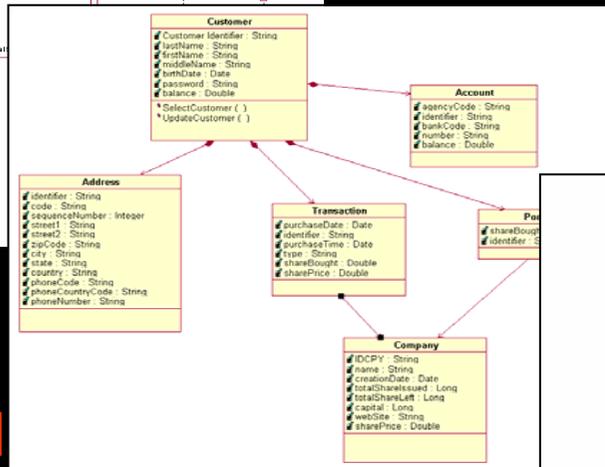


Diagramme de séquence



Modèle métier

Diagramme de classes analyse

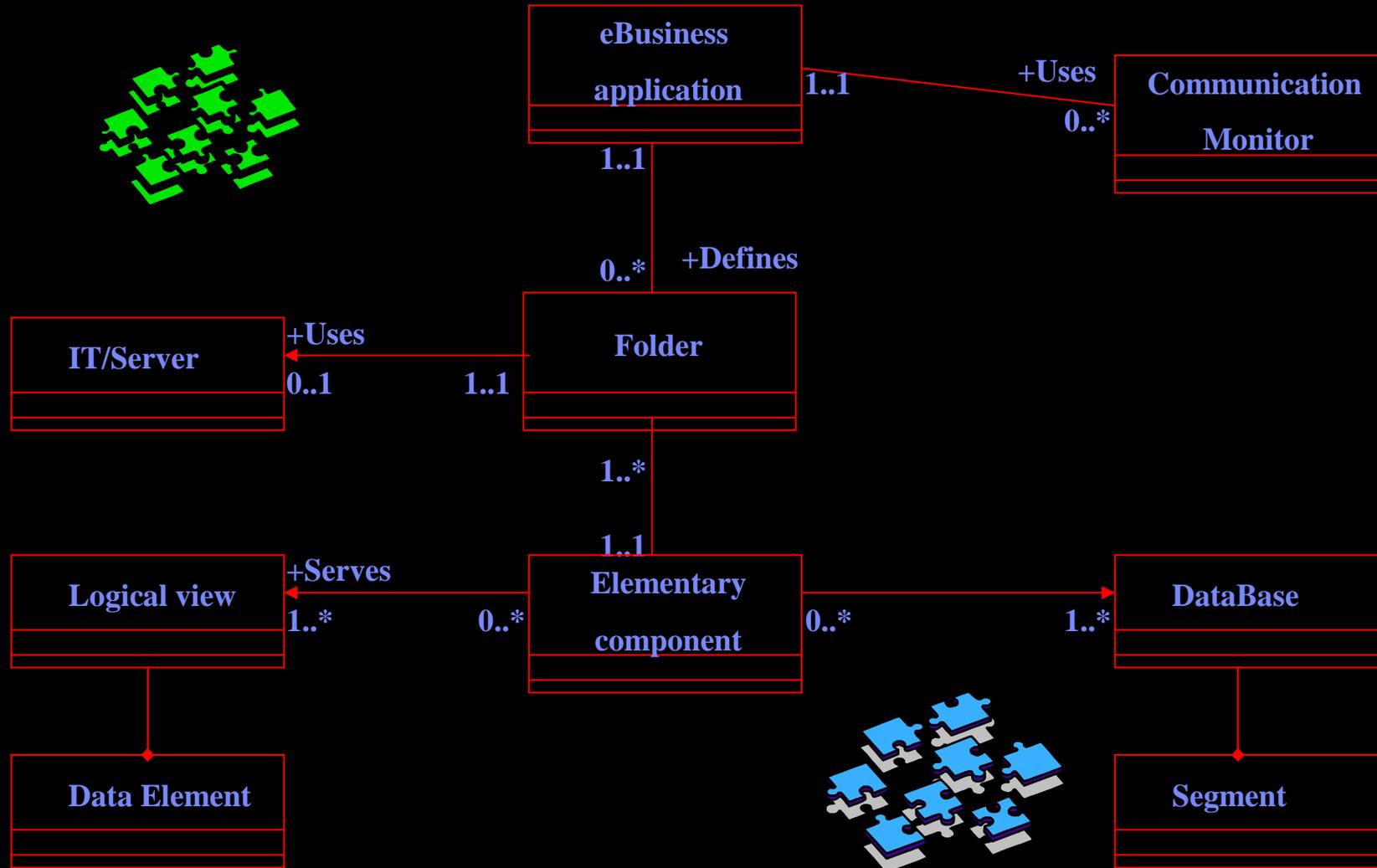


Modèle technique VAP

Entités VisualAge Pacbase



Le metamodèle e-Business de VisualAge Pacbase



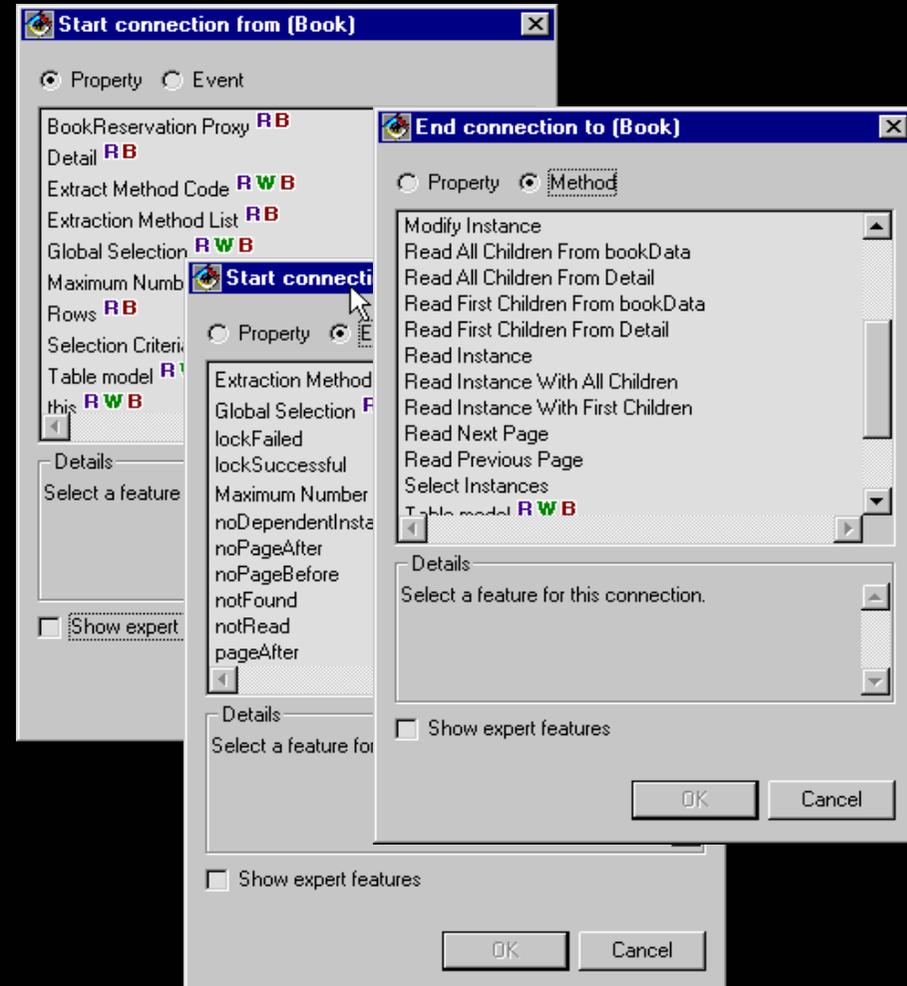
L'objet Proxy

■ Méthodes

- ▶ Services "serveur" (accès aux données, règles de gestions)
- ▶ **Services locaux** (cache local, contrôles, ...)

■ Propriétés Données applicatives

- ▶ Données techniques (number of instances per page, ...)
- ▶ Événements
 - Fin de liste, erreur



Test unitaire des méthodes de l'objet Proxy

VisualAge Pacbase Tools - Folder Unary Tests

File View Selected Actions locales Actions serveur Window Help

Instances

FOCLNT

Actions locales

Actions serveur

Browse locations...

Sélection sur un noeud

Sélection simultanée sur plusi.

Gestion de la pagination

Emission des mises à jour

Gestion du verrouillage logiqu.

Gestion des instances dépendan.

Gestion des Services Utilisate.

Gestion des conversations asyn.

Gestion des sous-schémas

selectInstances

readInstance

readInstanceAndLock

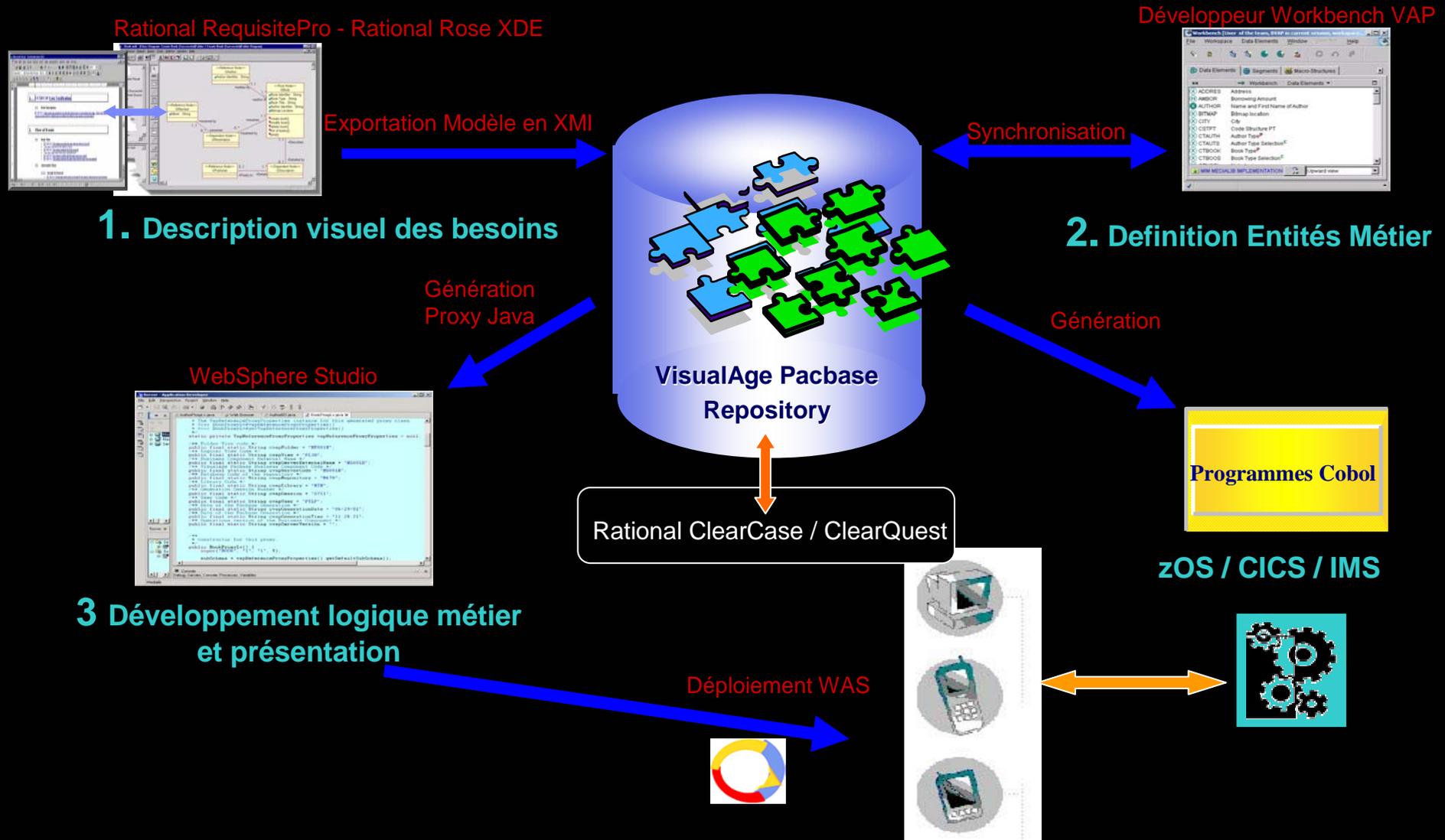
Nu.	Customer name	Corporate Name	Street Name	City	Zip Code	Type de client
3			Porto-Vecchio	Nantes	13000	P
4			Basket	Nantes	44000	N
5			España	Toulouse	31000	N
6			CHILI	Nice	05000	P
7			Californie	Paris	75000	P
8						

Attribute Label	Attribute Value
Customer Number	3
Customer name	BAUQUEL
Corporate Name	Philippe
Street Name	rue à Trégaté
City	Lyon
Zip Code	69000
Type de client	N

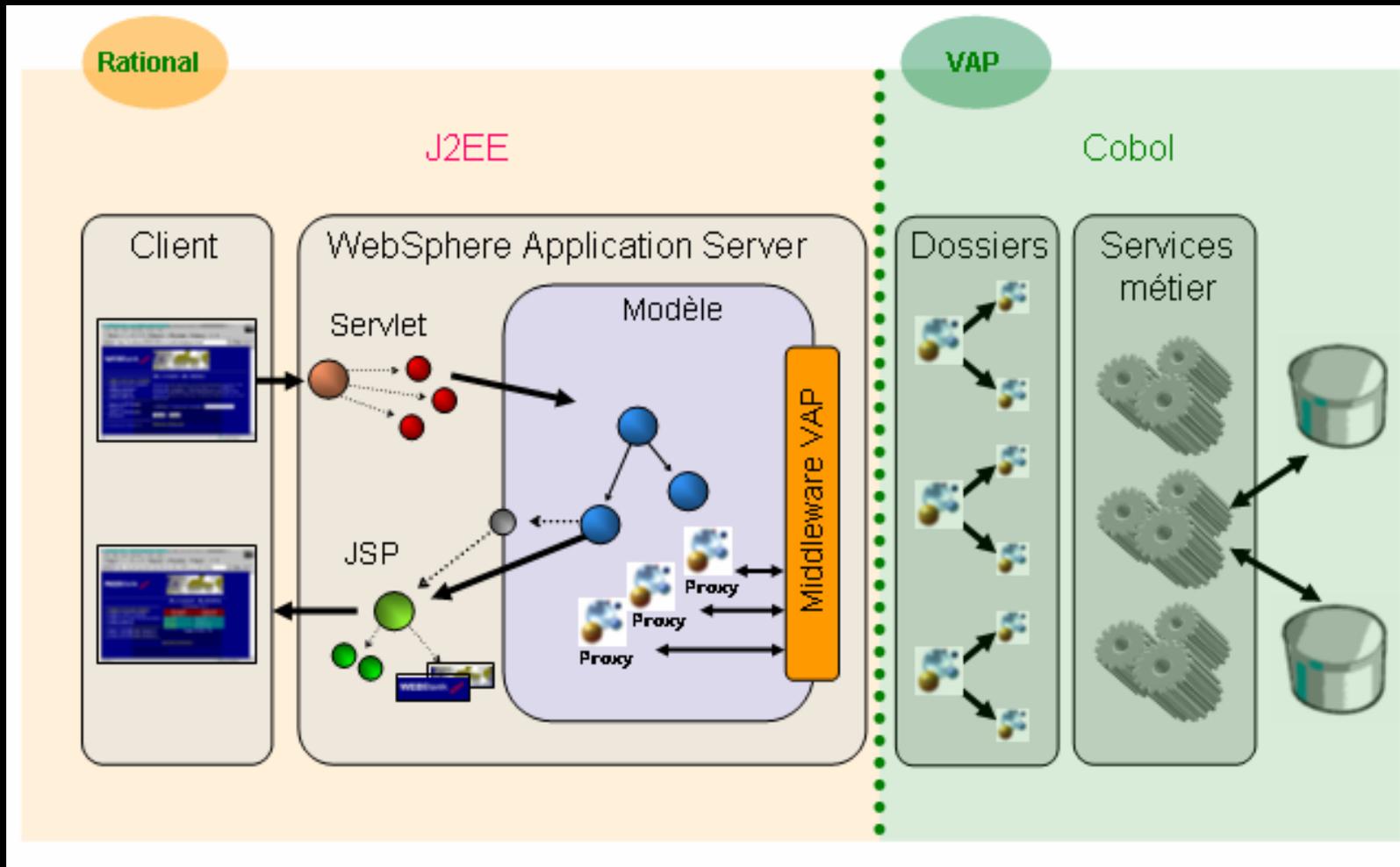
Location: Localhost-Fic-Local



Résumé : scénario d'utilisation de la démarche



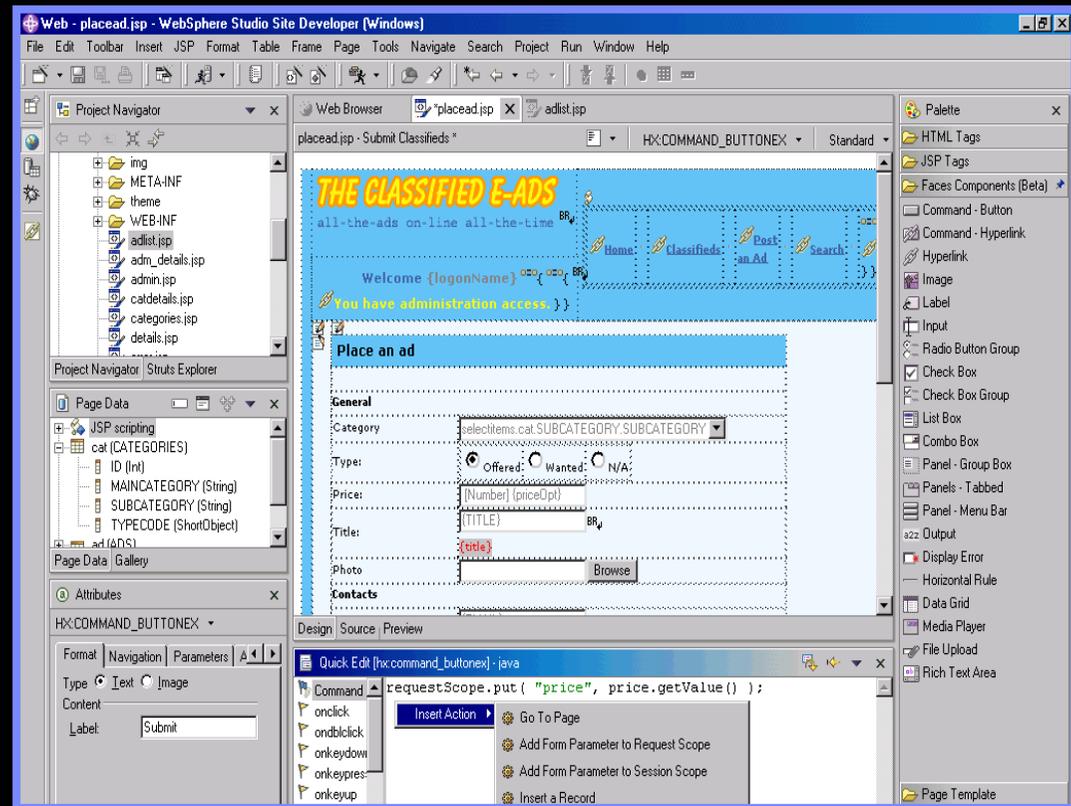
Développement de la partie J2EE



Développement rapide architecturé basé sur les standards

Flexibilité, évolutivité et performance

- Développement **rapide** et **simplifié** mettant en œuvre automatiquement les bonnes pratiques J2EE
- S'appuie sur des standards
- **Ouverture, ergonomie, confort** du poste de travail
- **Intégration** sur tout le cycle de vie des activités



IBM WebSphere Studio Application Developer



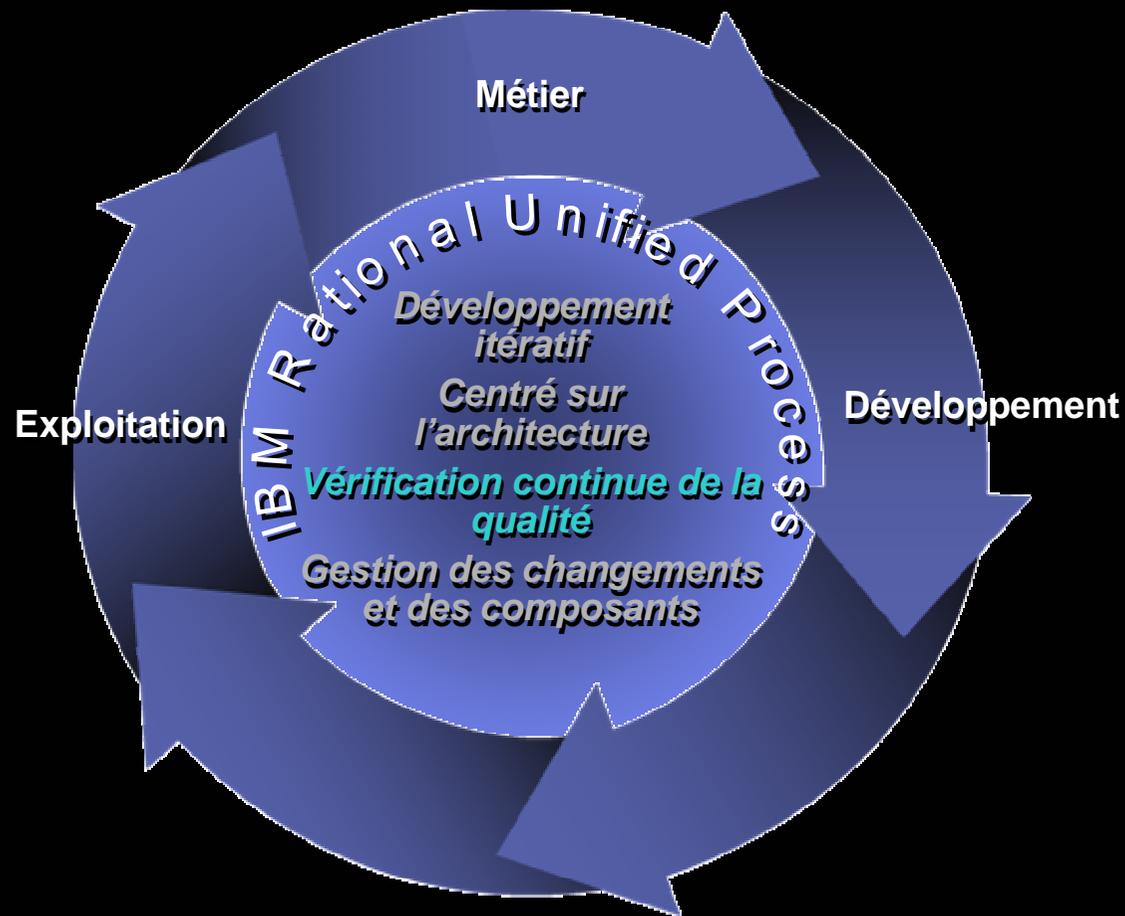
Les bénéfices de la démarche

- **Démarrage** plus rapide
 - ▶ Analyse de l'existant
- Application **répondant aux besoins** des utilisateurs
 - ▶ L'apport des cas d'utilisation
- Analyste et développeur plus **centrés** sur le métier
 - ▶ Approche par les modèles
- **Analyse** d'impact possible
 - ▶ Traçabilité et analyse de la couverture des exigences
- **Accroître** la qualité des applications finales
 - ▶ Automatisation par les outils (WebSphere Studio et plug-ins)
 - ▶ Intégration des activités dans un environnement unique



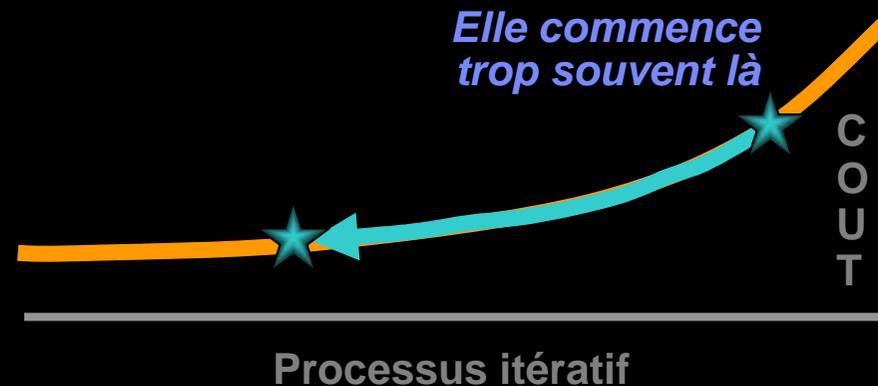
Quatre impératifs pour réussir vos développements

Un processus de développement guidé par le business

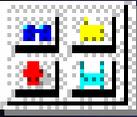


Vérification continue de la qualité

- Faire de la qualité une exigence centrale de votre **processus**.
- **Vérifier** que les exigences sont alignées avec les besoins du métier et des clients.
- **Valider** la qualité tôt et souvent.
- **Gérer les changements** pour garantir la réussite du déploiement.
- **Piloter** pour identifier, isoler et réagir aux violations du niveau de service.



Construire des tests de non régression automatiques

-  Définir cas de test
-  Enregistrer actions
-  Insérer une validation
-  Editer script (optionnel)



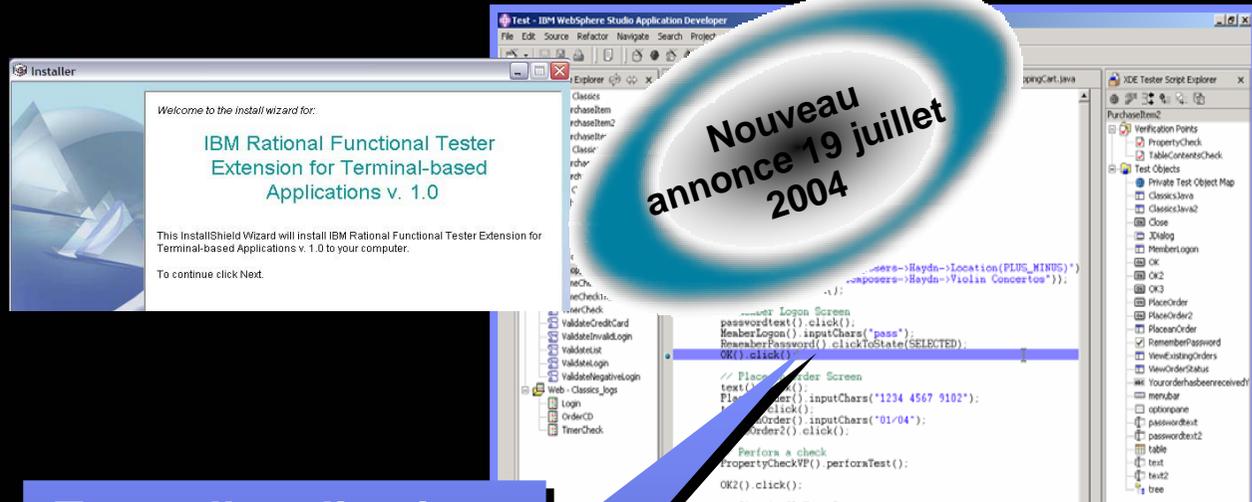
- Les objets peuvent changer en apparence, position, contenu, temps
 - *Oriented Recording*[®] rend les scripts de test plus robustes, quand les applications changent, ou à travers les configurations



Tester automatiquement les fonctionnalités

La solution IBM pour développer, maintenir et exécuter les tests fonctionnels et de non-régression

- Une seule solution pour vos tests d'applications Web et mainframes
- Améliorer la communication entre les développeurs et testeurs
- Faciliter la maintenance des tests
- Java comme langage de tests
- Technologie brevetée de reconnaissance graphique ScriptAssure™



**Tests d'applications
3270 et 5250**

**Activités de test
intégrées directement à
l'intérieur de
WebSphere Studio ou
Eclipse**

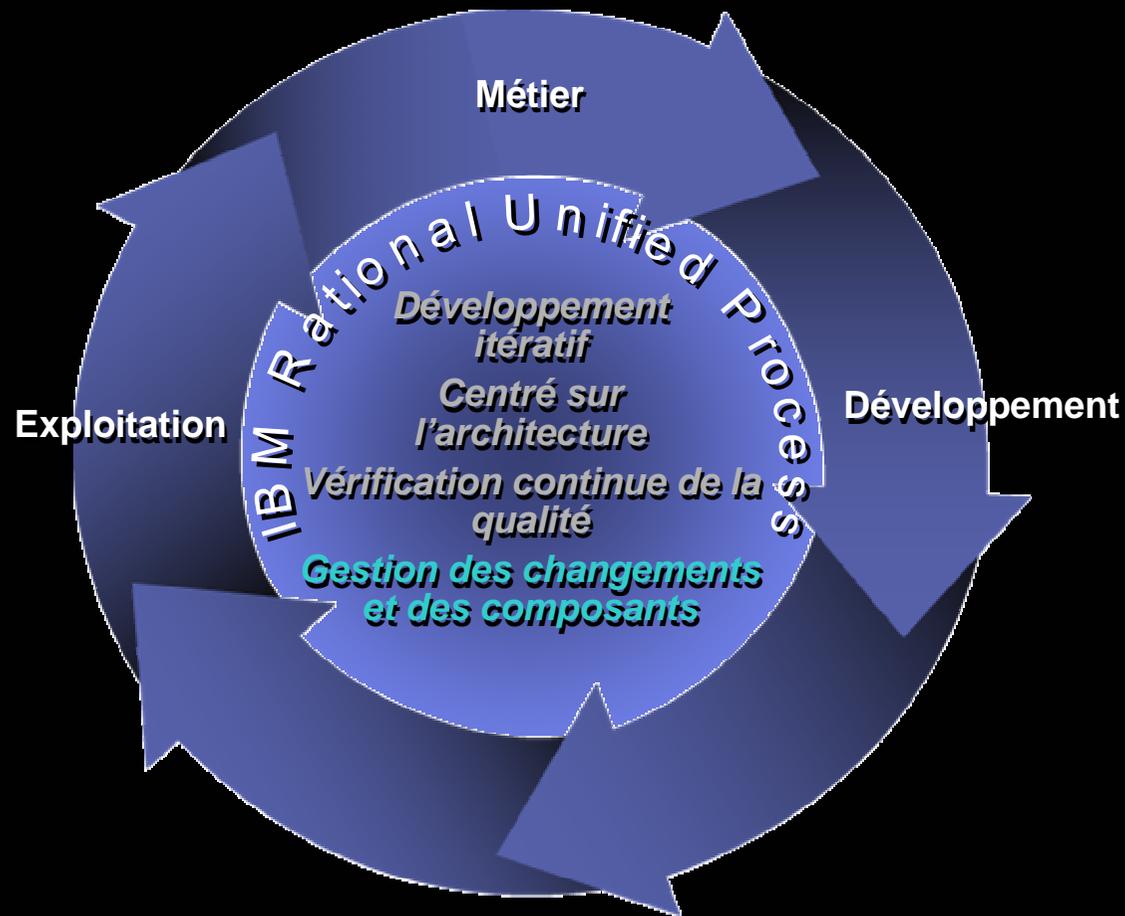


IBM Rational Functional Tester Extension for Terminal-Based Applications



Quatre impératifs pour réussir vos développements

Un processus de développement guidé par le business

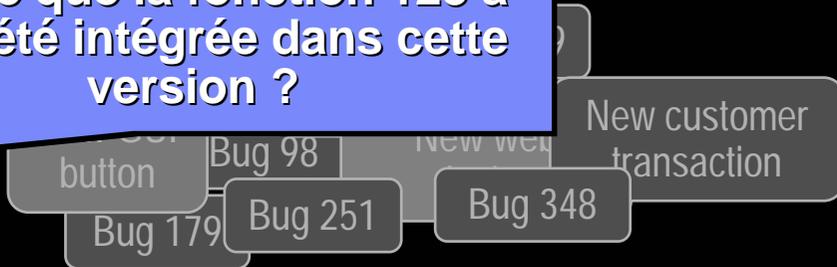


Gestion des changements : le défi



Analyste

Est-ce que la fonction 123 a bien été intégrée dans cette version ?



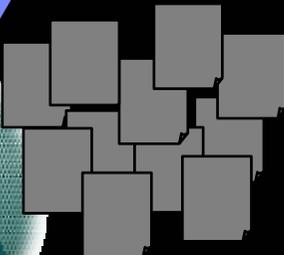
Chef de projet

Combien de bugs de sévérité 1 restent-ils dans cette version ?

Je n'ai pas pu oublier de fichier ... !



Developpeurs

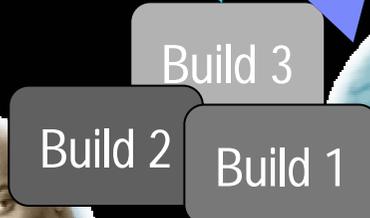


Pourquoi cet exécutable ne marche pas ?



Intégrateur

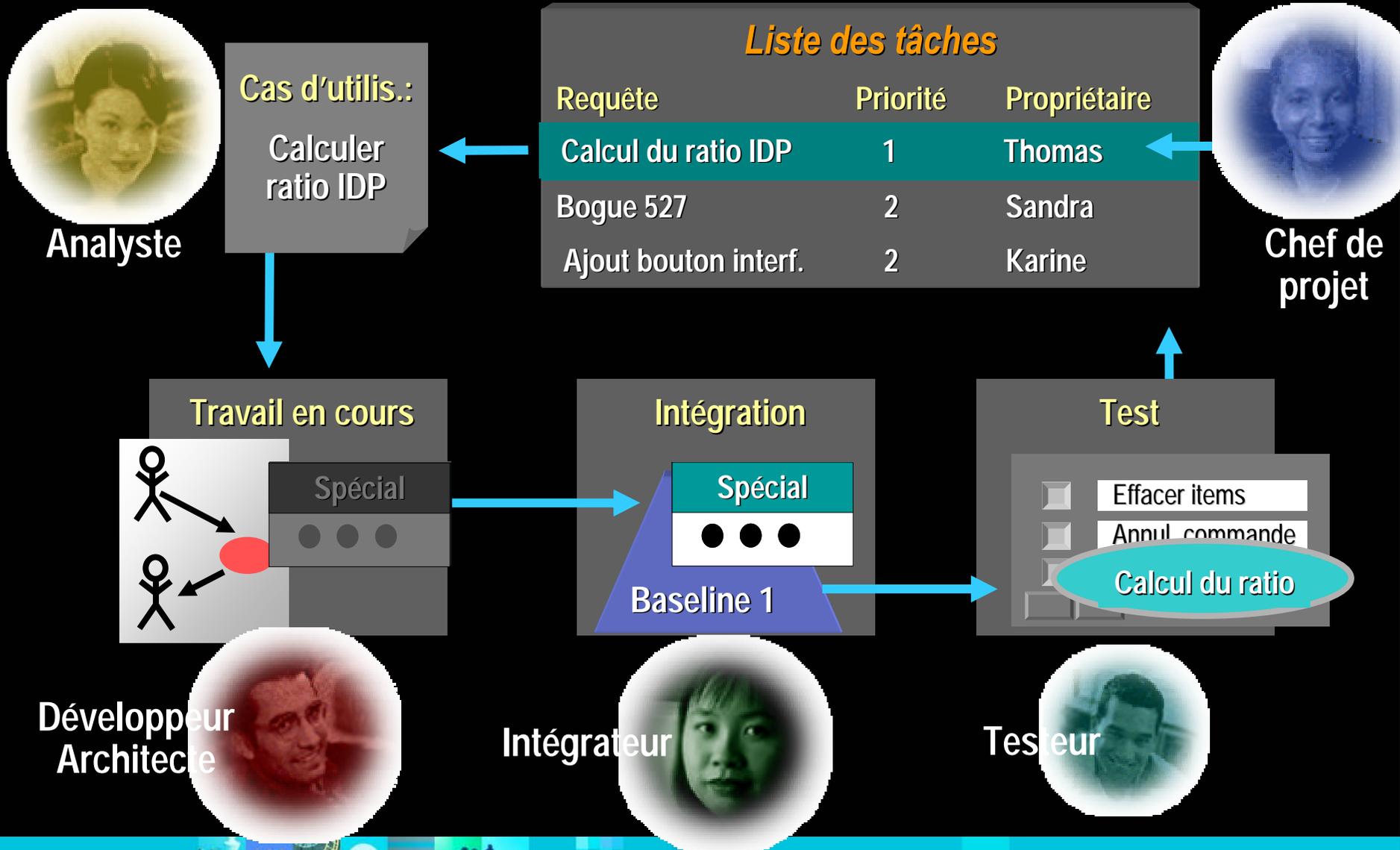
Est-ce que le bug 823 est corrigé dans cette version ?



Testeurs



Cycle de vie d'une demande de changement



Piloter par les demandes

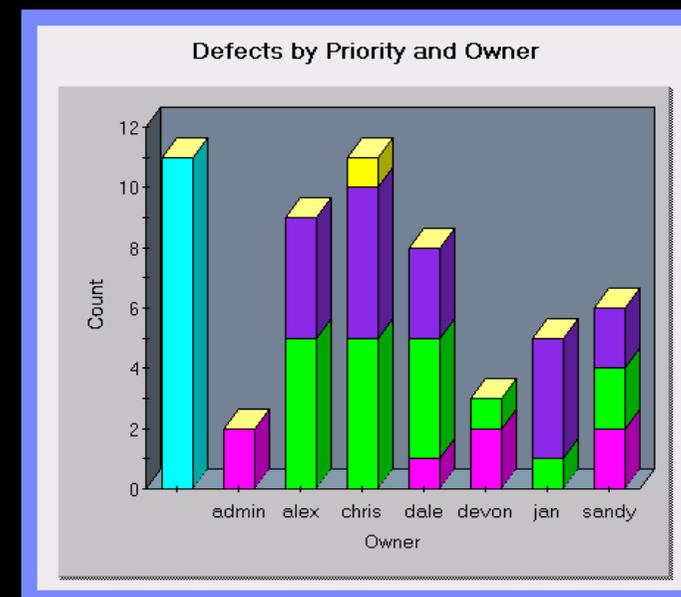
- Gestion des demandes et des ressources
 - ▶ Affectation des ressources
 - “Quelle répartition des demandes dans l'équipe ?”
 - “Qui est disponible pour corriger ce problème ?”
 - ▶ État du projet
 - “Combien de défauts de priorité 1 ?”
 - ▶ Tendances
 - “Quelle durée moyenne de correction d'un bug ?”
 - “Combien de temps pour implémenter ces évolutions ?”

The screenshot shows the Rational ClearQuest interface. The top part displays a table of requests with columns for ID, Owner, State, and record_type. Below the table, a detailed view of a specific request is shown, including fields for ID, Headline, Subj Project, UCM Project, Priority, Severity, Owner, and Description.

ID	Owner	State	record_type
CLISIC0000019	alex	Assigned	Defect
CLISIC0000020	alex	Assigned	Defect
CLISIC0000021	alex	Assigned	EnhancementRequest
CLISIC0000022	alex	Assigned	EnhancementRequest
CLISIC0000023	alex	Assigned	EnhancementRequest
CLISIC0000024	alex	Assigned	EnhancementRequest
CLISIC0000025	alex	Assigned	EnhancementRequest
CLISIC0000026	alex	Assigned	EnhancementRequest
CLISIC0000027	alex	Assigned	EnhancementRequest
CLISIC0000028	alex	Assigned	EnhancementRequest

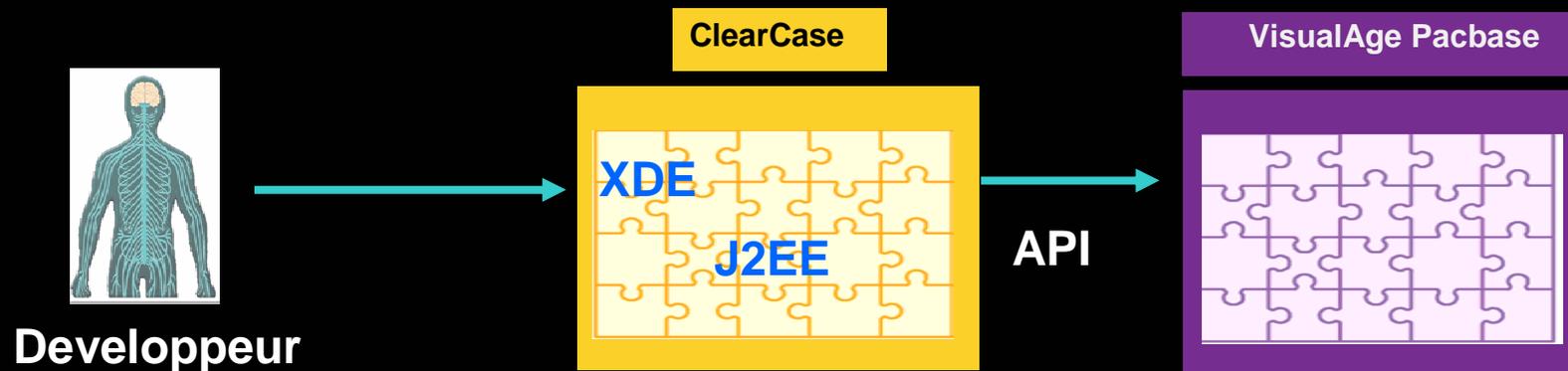
Detailed view of request CLISIC0000022:

- ID: CLISIC0000022
- State: Assigned
- Headline: Site "hangs" when server is down
- Subj Project: Webshop
- UCM Project: [empty]
- Priority: 2 Give High Attention
- Severity: 1 Critical
- Owner: alex
- Description: When the ClassicCD.com server is down, visitors to the site will eventually time out in their browser. We need to provide a backup server that would at least notify the user that we're currently experiencing problems at our site.



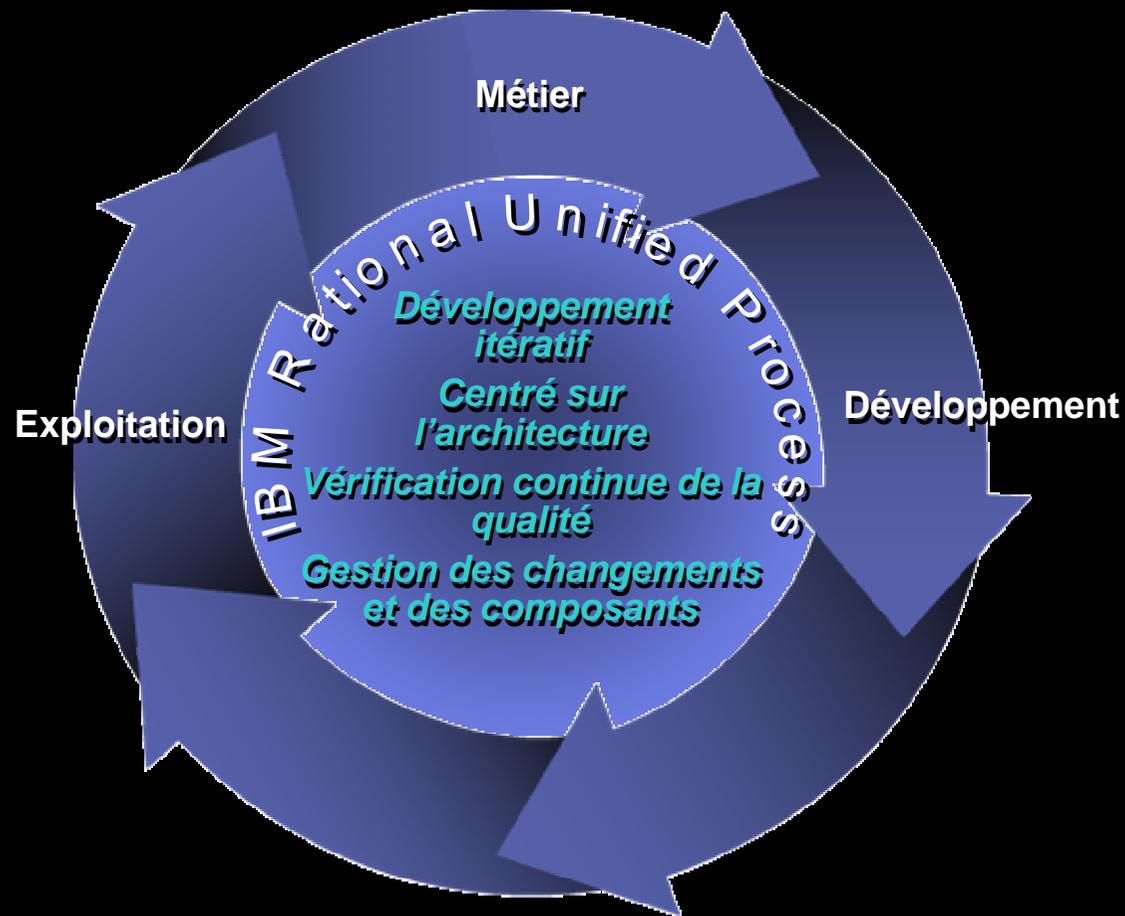
Couplage IBM Rational ClearCase et VisualAge

- Les outils sans référentiel sont sous contrôle ClearCase.
- Les outils avec référentiel sont couplés à ClearCase via une API ("couplage fort")
- Synchronisation des actions de réservation et de libération des éléments à modifier



Quatre impératifs pour réussir vos développements

Un processus de développement guidé par le business



IBM Rational Software Development Platform

	Découverte et re-documentation du métier et des applications	Planification des projets	Gestion de la sous-traitance	Conformité aux standards et réglementations	Modernisation d'applications
Dev. Itératif RUP	X	X	X	X	X
Gestion des exigences RequisitePro		X	X	X	
Architecture WBI, Rose/XDE	X				X
Gestion des changements ClearCase/CQ	X				
Develop Va Pacbase/WSAD	X		X	X	X
Tests Test Studio			X	X	



La plate-forme de développement logiciel : les fonctions



Analyste

Modélise, simule et pilote les processus métier



Architecte

Modélise les applications et les données



Développeur

Construit visuellement les programmes, et génère le code



Testeur

Conçoit, crée, et exécute les tests



Chef de Projet

- Applique un processus commun
- Suit l'avancement du projet
- Gère les exigences
- Gère les changements et les composants
- Gère la qualité

TEAM UNIFYING PLATFORM

Mobile et Embarqué

*DB2
Lotus
Tivoli
WebSphere*

Serveurs IBM et autres

Partenaires

Microsoft

Standards Ouverts

Services IBM

