



IBM Software Group

La solution IBM Rational de Gestion de Configuration et de Changements

GIPS, 28 mars 2007

Rational. software



@business on demand software

*Eric Hivon – Rational Sales Specialist
Jean-Philippe Durand - Rational IT Specialist
Jean-Michel Oliva – Rational IT Specialist*

IBM Software Group | Rational software



Plan

- Rappel des besoins de GIPS
- Présentation de la solution de gestion de configuration UCM
- Proposition d'architecture
- Démonstration
- Assistance au déploiement
- Conclusion



Rappel des besoins de GIPS

- Besoin d'un référentiel unique (gestion d'un environnement hétérogène)
- Processus piloté par la gestion des demandes
- Gestion des fabrications et des livraisons
- Environnement multisite
- Traçabilité, sécurité et reporting
- Facilité d'administration
- Souplesse et évolutivité de la solution



Plan

- Rappel des besoins de GIPS
- Présentation de la solution de gestion de configuration UCM
- Proposition d'architecture
- Démonstration
- Assistance au déploiement
- Conclusion



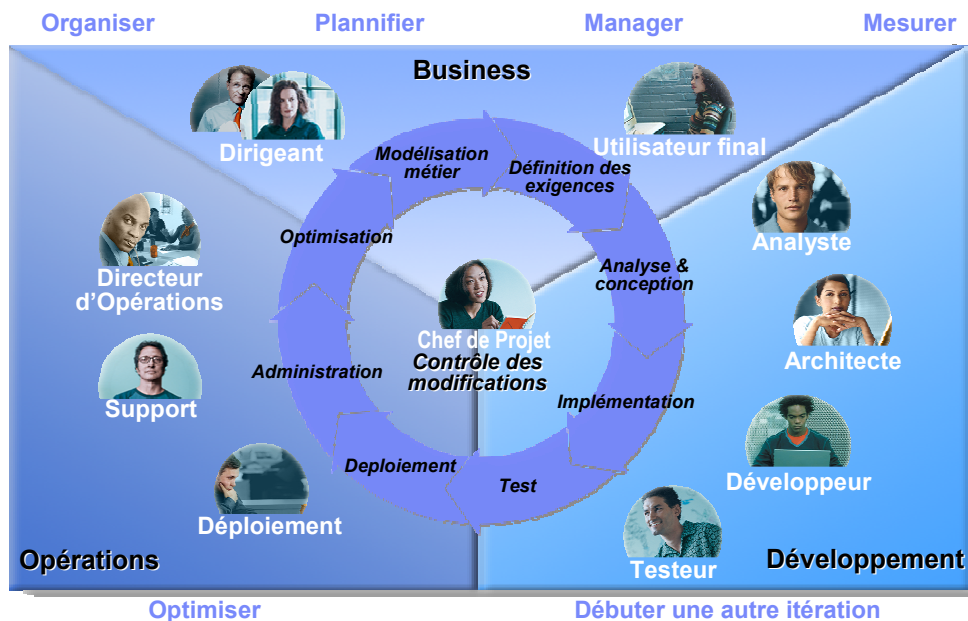
La solution de gestion de configuration UCM



- Introduction
- Solution IBM de gestion des configurations
- Solution IBM de gestion des changements
- Solution IBM de gestion des déploiements
- Solution IBM de gestion des tests
- Utilisation sur sites distribués
- Paramétrage et administration de la solution
- Conclusion



Le cycle de vie du développement logiciel



La plateforme de développement IBM Rational

Une solution complète, ouverte, modulaire et éprouvée



Rational Unified Change Management

Classée N°1 par IDC depuis plus de 4 ans !

Simplifier la gestion des changements

Modèle d'utilisation prêt à l'emploi
Automatise les meilleures pratiques
Approche basée sur les activités et les composants



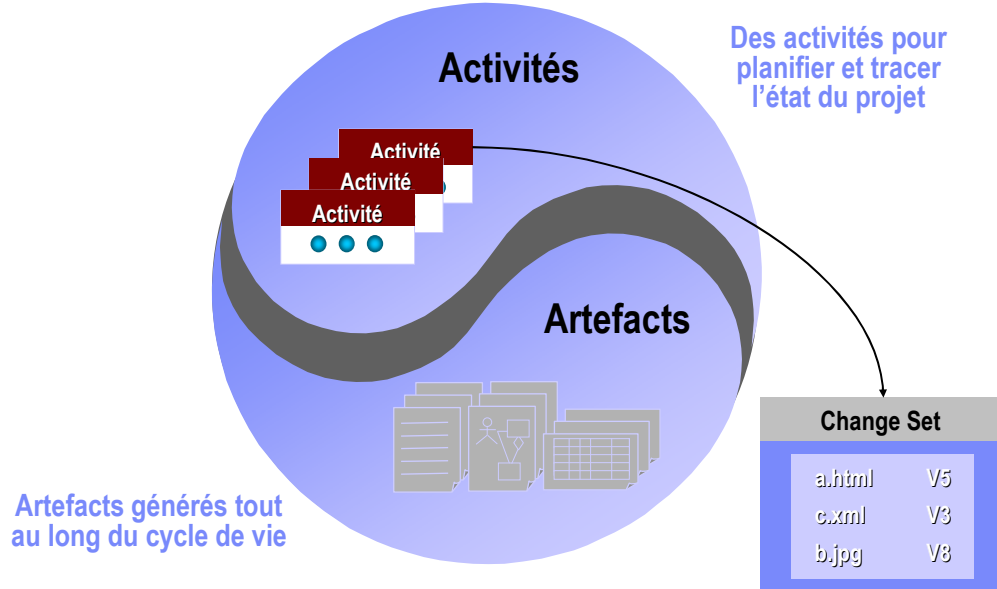
IBM Rational ClearCase



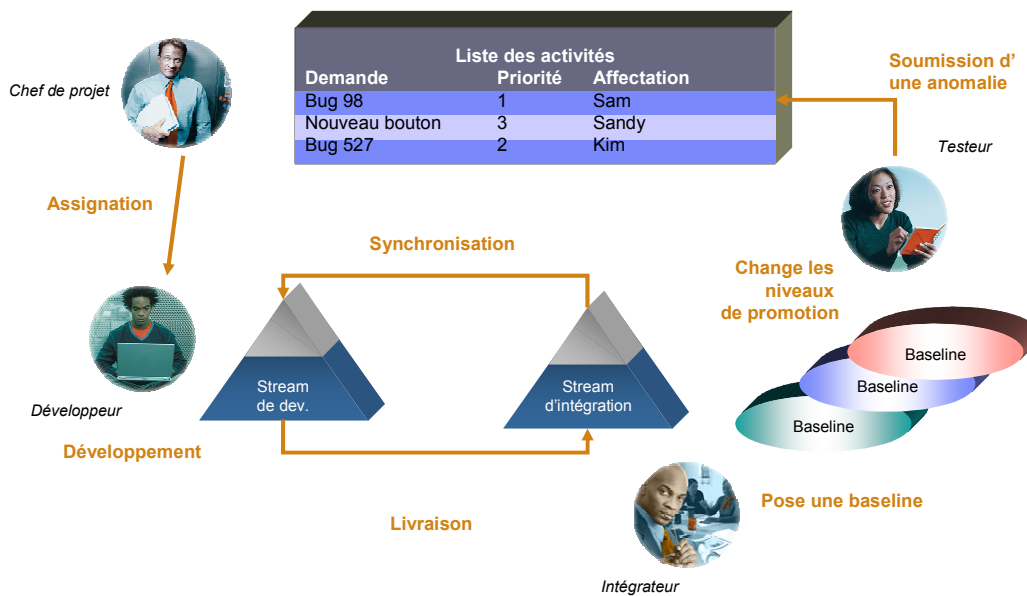
IBM Rational ClearQuest



Unified Change Management



Workflow



La solution de gestion de configuration UCM



- Introduction
- Solution IBM de gestion des configurations
- Solution IBM de gestion des changements
- Solution IBM de gestion des déploiements
- Solution IBM de gestion des tests
- Utilisation sur sites distribués
- Paramétrage et administration de la solution
- Conclusion

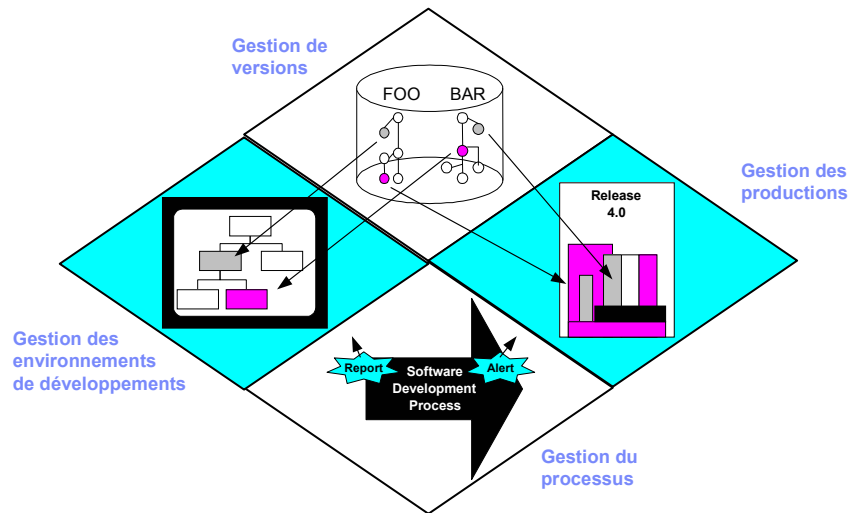


Gestion de configuration Vs Gestion de Version

- Gestion de configuration = processus automatisé
 - ▶ Gestion du travail en parallèle
 - ▶ Gestion des espaces de travail
 - ▶ Gestion automatisée des reports de corrections
 - ▶ Gestion du workflow : du développement vers la mise en production
 - ▶ Gestion des releases de logiciel et des versions de composants
 - ▶ Gestion des demandes de modifications (corrections, évolutions)
 - ▶ Gestion des fabrications



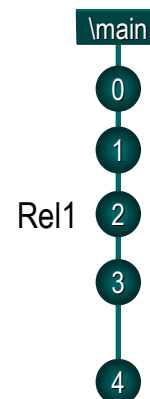
Qu'est ce qu'un outil de gestion de configuration ?



Rational ClearCase : le référentiel robuste et sécurisé

- Référentiel sécurisé (VOB)
 - ▶ Stocke tout type d'éléments (fichiers textes, binaires, tests, documentation, etc.)
 - ▶ Un élément est soit un fichier soit un répertoire

- Les éléments sont gérés en version
 - ▶ Mécanismes de réservation/libération
 - ▶ On ne peut pas modifier un fichier sans laisser de trace
 - Checkout
 - Checkin
 - Dans le cadre d'une « activité »
 - ▶ Chaque version d'élément est référencé par un identifiant unique
 - ▶ Gère aussi les versions de répertoires



Rational ClearCase : le référentiel robuste et sécurisé

- ClearCase gère les méta données associées aux éléments :
 - ▶ Attributs, Label, branches, hyperliens, triggers
 - ▶ Certaines méta données peuvent supporter des méta données

- ClearCase gère l'historique (qui, quand, quoi)
 - ▶ Chaque évènement sur un élément ou une méta donnée crée un enregistrement

Date	User	Name	Event Kind	Version	Comment
7/10/2006 7:02:09 AM	alex	...listTransactions.jsp	create version	\main\RatBankWeb_Int\4	
7/10/2006 6:42:45 AM	alex	...listTransactions.jsp	make hyperlink "Change" on version	\main\RatBankWeb_Int\4	Attached hyperlink "Change@1263@RatBankProjects".
6/25/2006 5:11:58 AM	alex	...listTransactions.jsp	destroy version on branch	\main\RatBankWeb_Int	Destroyed version "\main\RatBankWeb_Int\3".
6/19/2006 9:53:15 AM	alex	...listTransactions.jsp	destroy version on branch	\main\RatBankWeb_Int	Destroyed version "\main\RatBankWeb_Int\2".
4/2/2006 12:54:02 PM	alex	...listTransactions.jsp	create version	\main\RatBankWeb_Int\1	created by clearimport
4/1/2006 10:19:23 AM	alex	...listTransactions.jsp	import file element	\main\RatBankWeb_Int\1	created by clearimport
4/1/2006 10:19:23 AM	alex	...listTransactions.jsp	create version	\main\RatBankWeb_Int\0	created by clearimport
4/1/2006 10:19:23 AM	alex	...listTransactions.jsp	create branch	\main\RatBankWeb_Int	created by clearimport
4/1/2006 10:19:23 AM	alex	...listTransactions.jsp	create version	\main\0	
4/1/2006 10:19:23 AM	alex	...listTransactions.jsp	create branch	\main	
4/1/2006 10:19:23 AM	alex	...listTransactions.jsp	create file element	\main	created by clearimport

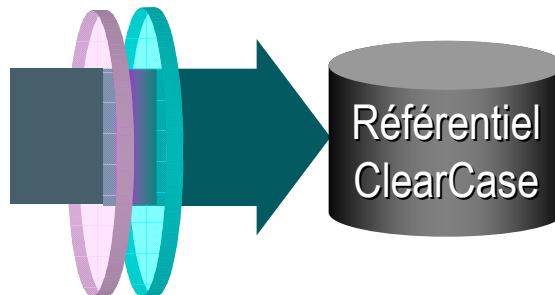


Une gestion très souple des espaces de travail

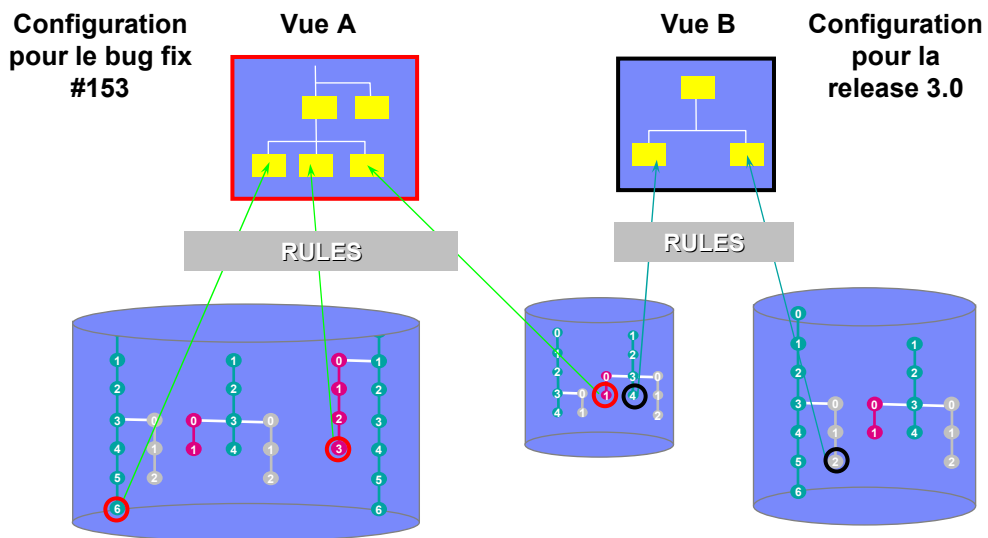
Name	Size	Modified	Version
debug	0KB	06/13/2001 21:41:19	\main\1
res	1KB	06/13/2001 21:41:41	\main\1
InvStat.sps	17KB	10/05/2001 16:02:30	
InvStat.clw	1KB	06/13/2001 21:40:12	\main\1
InvStat.cpp	2KB	10/06/2001 11:52:48	\main\Rel3_Integration\5
InvStat.def	1KB	06/13/2001 21:40:19	\main\1
InvStat.dsp	4KB	06/13/2001 21:41:13	\main\2
InvStat.dsw	1KB	10/05/2001 16:02:02	
InvStat.h	1KB	10/14/2001 11:56:19	\main\Rel3_Integration\1
InvStat.mak	7KB	06/13/2001 21:40:29	\main\1
InvStat.mcb	57KB	10/05/2001 16:17:24	
InvStat.oaf	1KB	06/13/2001 21:40:32	\main\1
InvStat.opt	52KB	10/05/2001 16:17:14	
InvStat.plg	2KB	06/13/2001 21:40:36	\main\1
InvStat.rc	3KB	06/13/2001 21:40:39	\main\1
InvStatInterface.cpp	3KB	10/05/2001 16:38:34	\main\Rel3_Integration\6
InvStatInterface.h	1KB	06/13/2001 21:40:45	\main\1
Resource.h	1KB	06/13/2001 21:40:49	\main\1
StdAfx.cpp	1KB	06/13/2001 21:40:52	\main\1
StdAfx.h	1KB	06/13/2001 21:40:56	\main\1
warehouse.cpp	3KB	10/05/2001 16:04:03	\main\Rel3_Integration\3
warehouse.h	1KB	06/13/2001 21:41:03	\main\1

Configuration de la vue = Règles de sélection

- des éléments des référentiels,
- et de la version de ces éléments

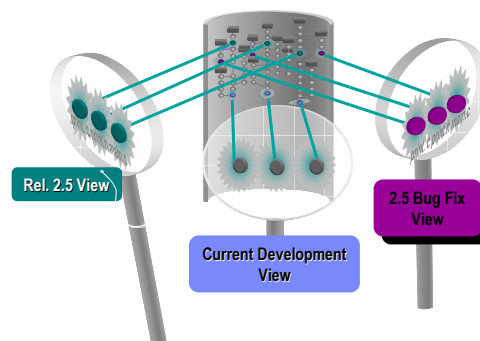


Les vues ClearCase sont des configurations



Les « Vues » ClearCase

- **Les vues Dynamiques**
 - ▶ Les versions des éléments sont accédés à travers le réseau (sans copie en local)
 - ▶ C'est une spécificité ClearCase
- **Les vues Snapshot**
 - ▶ Les versions des éléments sont copiés sur les disques locaux



Un utilisateur peut avoir à tout moment **plusieurs vues** sur plusieurs configurations



Intégrations avec les IDE

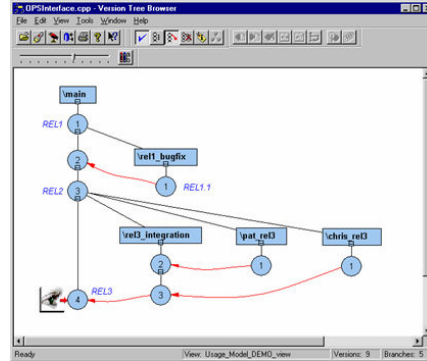


Intégration avec les progiciels

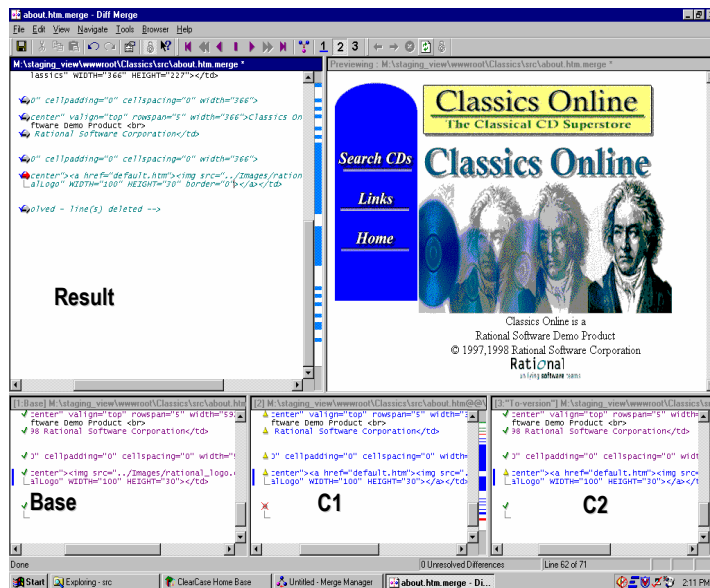
- Souvent, les progiciels gèrent directement leurs objets en base de données
 - ▶ il est difficile d'exporter et de gérer en configuration de manière unitaire ces éléments.
- Mode d'utilisation
 - ▶ Tous les fichiers de paramétrage externes, de spécification, de tests, etc. sont gérés en gestion de configuration
 - ▶ Les bases de données sont exportées lors des releases majeures pour être gérées en configuration dans ClearCase
 - ▶ Permet un retour à une configuration stable.
- Par contre, la gestion en versions des packages de déploiement (fichiers de déploiements générés par le progiciel) est prise en compte nativement par UCM

Gestion avancée du développement parallèle

- Simplicité de développement en parallèle
 - ▶ Espaces de travail privés
 - ▶ Synchronisation automatique avec l'équipe (deliver – rebase)
 - ▶ Fusions assistées voire automatiques
 - ▶ Report automatique des modifications sur plusieurs environnements
- Permet d'éviter de bloquer les développeurs durant les phases d'intégration
- Permet d'isoler des développements à risque (développement, intégration)
- Permet de créer des environnements de maintenances, de recettes, des variantes



Outil de fusion



Un gestionnaire étendu des productions

- Clearmake et Clearaudit
 - ▶ Génération des « objets dérivés » lors des builds ou commandes (binaires ou autres)
 - ▶ Chaque objet est associé à un « configuration record » contenant les informations suivantes :
 - Target, host, OS, date, time, vue, répertoire
 - Objets MVFS (gérés en configuration sous ClearCase)
 - Variables et options utilisées
 - Les scripts utilisés
 - Les dépendances externes explicites

```

Target bugsched.o built by akp.user
Host "oxygen" running SunOS 4.1.1 (sun4c)
Reference Time 11-Dec-
94.15:23:21, this audit started 11-Dec-
.94.15:23:39
View was neptune:/usr/people/akp/views/940615.vws
Initial working directory was /vobs/docaux/bgr/su
n4
-----
MVFS objects:
-----
/vobs/docaux/bgr/bugs.h@@/main/3 <17-Jun-
94.23:55:22>
/vobs/docaux/bgr/bugsched.c@@/main/2 <11-Dec-
94.15:23:04>
/vobs/docaux/bgr/libbgr/stint.h@@/main/2 <08-Sep-
94.10:06:04>
-----
Variables and Options:
-----
CC=/usr/bin/cc
CFLAGS=-I../libbgr -DBSD -DSCCS -g
RM=rm -f
SRC=..
-----
Build Script:
-----
rm -f bugsched.o ; /usr/bin/cc -c -I../libbgr -
DBSD -DSCCS -g ../bugsched.c
-----

```



Un gestionnaire étendu des productions

- ClearCase garantit 100% de reproductibilité grâce à une traçabilité complète des productions
 - ▶ Les « configuration record » sont stockés dans l'objet dérivé
 - ▶ Ils permettent de comprendre les différences entre productions
 - Comparaison des binaires (grâce à leur liste de composition)
 - Identification des éléments ayant évolué entre deux livraisons
- Clearmake permettent de partager des binaires entre plusieurs vues (évite de générer un binaire déjà compilé dans une autre vue)
- Clearmake permet de paralléliser et de distribuer automatiquement les fabrications
- Nécessite l'utilisation des vues dynamiques (MVFS)



Un gestionnaire étendu des productions

Exemple d'audit de script Ant capturé avec ClearAudit

```

Derived object: \UCMCOMP\auditant\dist\client.jar@07-Dec.18:53.2147484016
Target archive:jar built by nicholas.None
Host "IBM-739B2CC0B57" running NT 5.0 (i586)
Reference Time 07-Dec-04.18:53:05,
  this audit started 07-Dec-04.18:53:11
View was Ibm-739b2cc0b57:C:\Documents and Settings
  \nicholas\CCREP\VIEWSTO\Javaproject_dev_view.vws

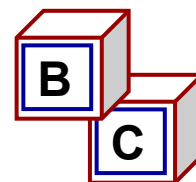
Initial working directory was Z:\UCMCOMP\auditant
-----
MVFS objects:
-----
\UCMCOMP\auditant\bin\classes\demo.class@07-Dec.18:53.2147483848
\UCMCOMP\auditant\bin\classes\mydemo.class@07-Dec.18:53.2147484010
Build Script:

```



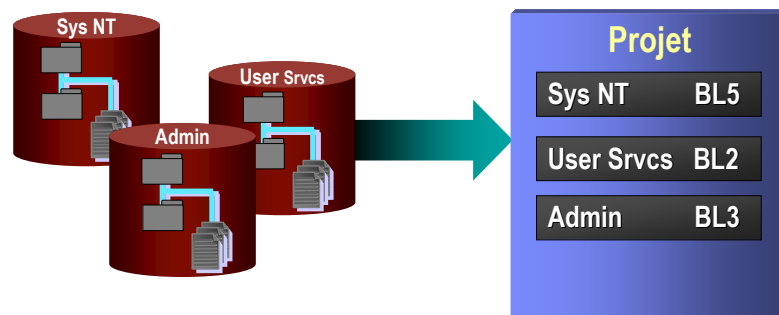
L'apport d'UCM : Élever le niveau d'abstraction

- **Composant**
 - ▶ Ensemble de fichiers et répertoires définissant un sous-système
 - ▶ Les composants sont versionnés
 - ▶ Hiérarchie de dépendances entre composants
- **Projet**
 - ▶ Il contient un ensemble de composants en modification et/ou lecture
 - ▶ Permet le développement d'une release de l'application
- **Stream**
 - ▶ Environnement de travail (développement, intégration, maintenance,...)
 - ▶ Un projet comporte plusieurs streams (intégration, développement, recette, etc...)
 - ▶ Les Environnement peuvent être partagés ou non
- **Activités**
 - ▶ Tâche à effectuer liée à une demande (notion de granularité)
 - ▶ Tous les changements sont associés à une activité
 - ▶ L'association est gérée automatiquement par UCM
- **Baselines**
 - ▶ Configuration à un moment donné d'un composant ou d'un projet
 - ▶ L'état des baselines est configurable (niveau de promotion)



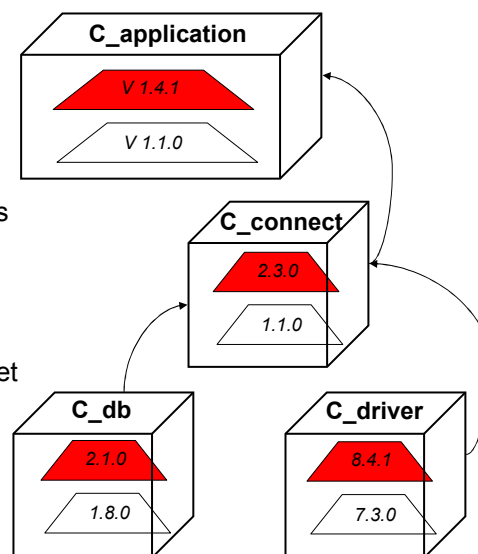
UCM : La gestion des composants

- Construction de baselines cohérentes
 - ▶ Les composants définissent la portée du projet
 - ▶ Les composants correspondent à des objets d'architecture de haut niveau (par ex. sous-systèmes)



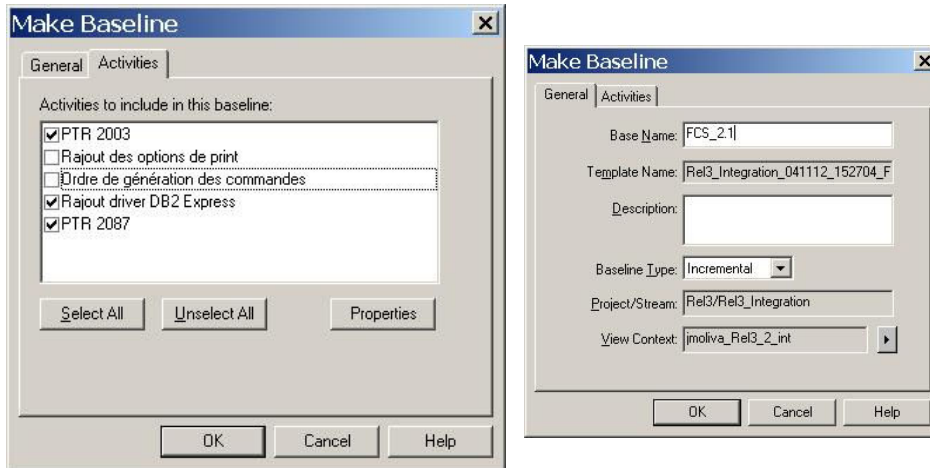
UCM : optimiser pour les intégrateurs

- Utilisation des baselines composites
- Simplification du processus d'intégration
- Optimisation du partage des composants (lecture seul ou lecture/écriture)
- Structuration des équipes
- Gestion automatique des dépendances et des conflits
- Dans un monde hétérogène, les composants appartiennent à des environnements différents

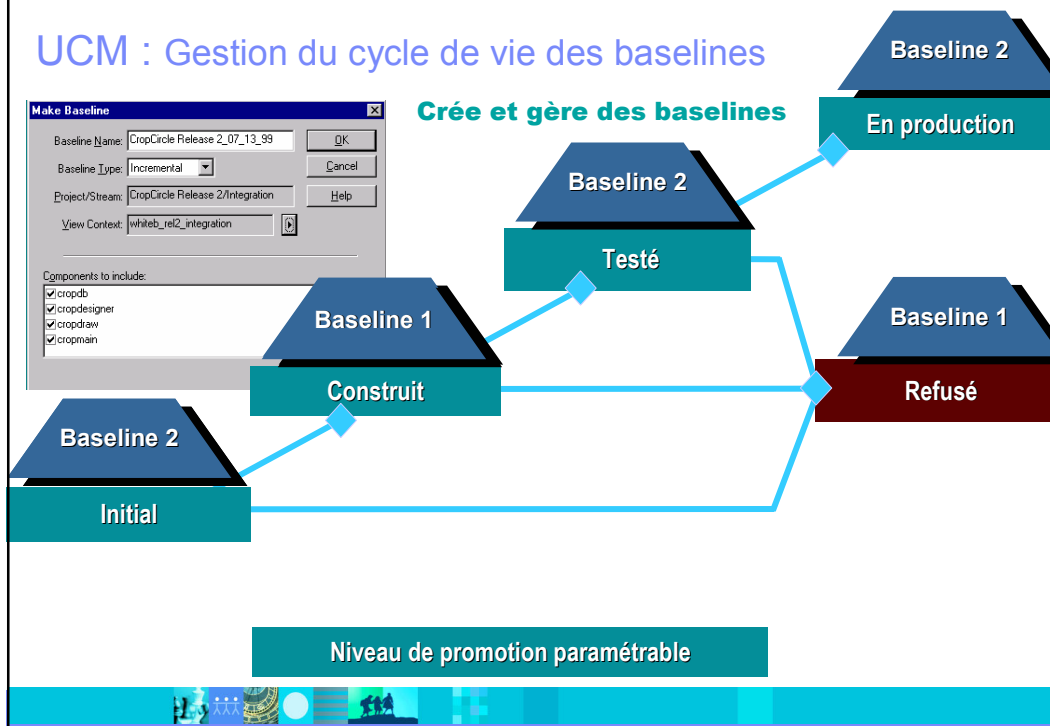


Gestion des baselines

- La création des baseline est piloté par les demandes

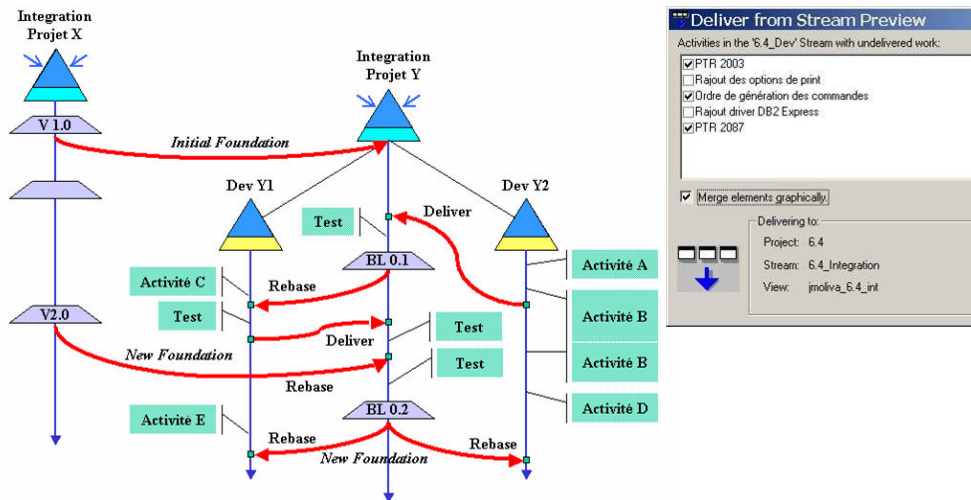


UCM : Gestion du cycle de vie des baselines



Gérer les échanges entre environnements

- Gestion des flux entre environnements (deliver, rebase)



Deliver from Stream Preview

Activities in the '6.4_Dev' Stream with undelivered work:

- PTR 2003
- Rajout des options de print
- Ordre de génération des commandes
- Rajout driver DB2 Express
- PTR 2087

Merge elements graphically

Delivering to:

- Project: 6.4
- Stream: 6.4_Integration
- View: jmoliva_6.4_int

Les projets organisent le développement

The image shows two screenshots of the 'Exploring ClearCase Projects' interface. The top screenshot shows a project tree with 'Release 2' selected, and a central pane showing 'Integration' and 'shirley_rel2' projects. An arrow points to this pane with the text 'Les projets organisent le développement'. The bottom screenshot shows the 'Components' pane with a tree view and icons for 'cropdb', 'cropdesigner', 'cropdraw', and 'cropmain'. An arrow points to this pane with the text 'Les projets organisent les composants'.

Facilité d'organisation des projets

Quelles sont les activités développées dans le projet ?

Quel est la configuration du projet ?

Propriétés de Intégration

Foundation baselines:

- ATM_SRC
- ATM_SRC_INITIAL
- INFOKEY_SRC_BL1
- TestIntegrationCC-CQ_REL1

Activities:

Headline	Owner
Setup UCM developer's lab	jmo
The word "bank" is misspelled in the ATM banner	jmo

Propriétés de Intégration

Components (select a component to display its baselines):

ATM_SRC

Baselines:

Title	Created On	Promotion Level
BL2	03/09/2001 13:12:38	INITIAL
★ BL1	03/09/2001 13:12:30	TESTED
ATM_SRC_INITIAL	03/09/2001 13:11:25	INITIAL

Où en est le projet ?

Component Tree

```

    graph TD
        Comp1[Component1@Test_IMO] --> C1_V1[C1_V1_105]
        C1_V1 --> C1_V2[C1_V2_105]
        C1_V2 --> Rel2[REL2_UCM_116]
        C1_V2 --> DevRel2[Integration project - DevRel2]
        DevRel2 --> DevRel2
        DevRel2 --> DevRel2
    
```

Compare Baselines

Baseline 1: RATLBANKMODEL_BL_2_INT

Component	Baseline
▲ RATLBANKMODEL	RATLBANKMODEL_2_INT
▲ RATLBANKMODEL_REL	RATLBANKMODEL_REL_2_INT

Baseline 2: RATLBANKMODEL_BL_6_INT

Component	Baseline
▲ RATLBANKMODEL	RATLBANKMODEL_6_INT
▲ RATLBANKMODEL_REL	RATLBANKMODEL_REL_5_INT

Baseline 2: RATLBANKMODEL_BL_6_INT

Name	Headline	Owner
▲ RBU00000021	Improve junit test output	alex
● RBU00000027	RatBankModel staging	alex
● RBU00000034	Updated model for new security policy	alex
● RBU00000071	Incorporate project files into staging area	alex
● RBU00000079	Implement release build Ant targets	alex

Suivi de l'état d'avancement réel du projet

- Combien d'anomalies graves ont été corrigées?
- Quel est l'état des évolutions prévues en itération n°1?

The screenshot displays two main components of the IBM Rational software interface:

- Report des demandes de maintenance:** A report titled 'Rapport des demandes de maintenance' showing details for two test cases:
 - TESTZ0000001 Correction de l'impression:** A 'Testée' (Tested) status. The description indicates that the development team has passed the product into production. It notes that the MDA tool is used to generate code, and the development team is responsible for ensuring that the code is generated correctly. The report includes a table with columns for 'ID', 'Description', 'Statut', and 'Demandeur'.

ID	Description	Statut	Demandeur
14	mettre à disposition des utilisateurs un formulaire de saisie	Completed	nom@company.fr
15	mettre à disposition des utilisateurs un formulaire de saisie	Completed	nom@company.fr
16	mettre à disposition des utilisateurs un formulaire de saisie	Completed	nom@company.fr
 - TESTZ0000004 Pas de formulaire de saisie:** A 'Submitted' status. The description indicates that the development team has not yet implemented the required functionality. It notes that the MDA tool is used to generate code, and the development team is responsible for ensuring that the code is generated correctly. The report includes a table with columns for 'ID', 'Description', 'Statut', and 'Demandeur'.

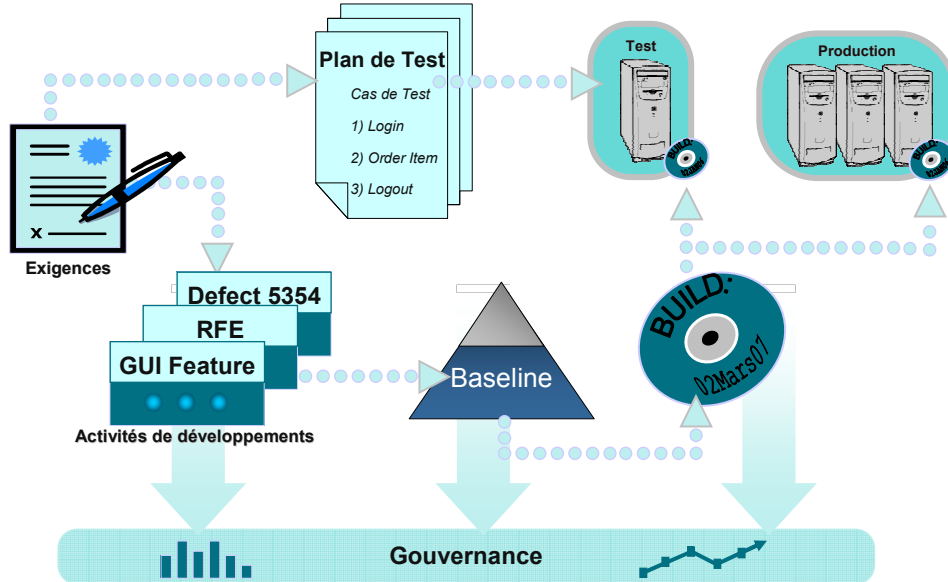
ID	Description	Statut	Demandeur
9	mettre à disposition des utilisateurs un formulaire de saisie	Completed	nom@company.fr
10	mettre à disposition des utilisateurs un formulaire de saisie	Completed	nom@company.fr
11	mettre à disposition des utilisateurs un formulaire de saisie	Completed	nom@company.fr
- Defect Aging by State (Defect):** A window showing a pie chart titled 'Defects by State and Severity'. The chart displays the distribution of defects across different states and severity levels. The legend indicates:
 - 1-Critical (Red)
 - 2-Major (Orange)
 - 3-Average (Green)
 - 4-Minor (Blue)
 - 5-Enhancement (Purple)

La solution de gestion de configuration UCM



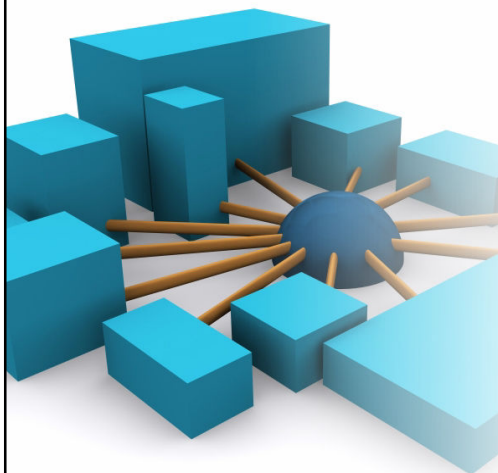
- Introduction
- Solution IBM de gestion des configurations
- Solution IBM de gestion des changements
- Solution IBM de gestion des déploiements
- Solution IBM de gestion des tests
- Utilisation sur sites distribués
- Paramétrage et administration de la solution
- Conclusion

Gestion de la traçabilité



La traçabilité tout au long du cycle de développement

La gouvernance des processus de développement, de déploiement et de test



IBM Rational ClearQuest

Changements - Déploiements - Tests

Traçabilité Complète

- Lien vers la Gestion des Exigences
- Gestion des tâches et des activités
- Lien avec la Gestion des Fabrications
- Gestion des Déploiements
- Gestion des Tests
- Gestion des Demandes

Rational ClearQuest : au delà de la gestion des anomalies

- Recueil et trace tous les types de demandes de changement
 - ▶ Définition de tout type d'enregistrement
 - ▶ Liens entre enregistrements
- Assure la communication entre les membres de l'équipe au travers du cycle de vie des demandes
- Signatures Électroniques & Audit Trail
- Solution prêt-à-l'emploi
- S'adapte facilement à votre processus
 - ▶ Entièrement et très facilement paramétrable
- Facile et puissant à utiliser
 - ▶ Requêtes, graphiques dynamiques
 - ▶ Rapports (Crystal Reports)
- S'installe partout, pour toute taille d'équipe



Contraintes réglementaires et compliance

- "Audit Trail"
 - ▶ Historisation possible de tous les changements
- Signatures Électroniques
 - ▶ Authentification renforcée sur certaines actions

The top screenshot displays the 'Field History' window. It contains a list of changes with the following details:

- START Fields Changed by admin, Close->Closed, 24 February 21
- Record was Electronically Signed by: Sig#2
- Old value of Resolution: <NULL>
- New value of Resolution: Enhancement Request
- Old value of State: Submitted
- New value of State: Closed
- END Fields Changed by admin, Close->Closed, 24 February 21
- START Fields Changed by admin, Submit->Submitted, 24 Feb
- Old value of Headline: <NULL>
- New value of Headline: Project # 2 Defect
- Old value of Owner: <NULL>
- New value of Owner: admin

The bottom screenshot shows the 'Electronic Signature' window. It includes the following fields:

- Signature Username: Old Sig#1
- Signature Password: [Redacted]



Communiquez en temps réel l'état du projet

- Requêtes et graphiques
 - ▶ Faciles à créer et à modifier grâce aux assistants
 - ▶ Graphiques : répartition, tendance,...

- Rapports
 - ▶ Moteur de génération basé sur Crystal Reports
 - ▶ Exportation vers de multiples formats (Excel, Word, HTML,...)

- Règles de Notification par Email
 - ▶ Notification automatique sur changements d'états et événements
 - ▶ Configurable via l'interface graphique
 - ▶ Pas de programmation nécessaire !

Defect Detail Report
17/06/2002

PLCIC00000097

Status	Submitted	Priority	1-Resolve Immediately	Severity
Submit_Date	14/06/02	Submitter	Owner	
Headline	The System Sometimes Crashes When Ordering CDs			
Description	When ordering a CD the system sometimes crashes for no reason.			

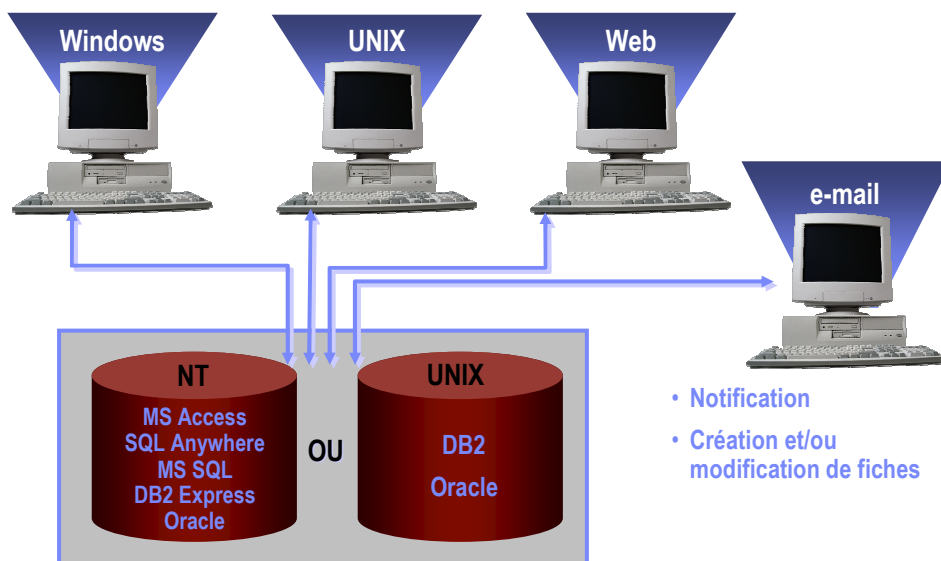
PLCIC00000092

Defects by Priority and Owner

Owner	High Priority	Medium Priority	Low Priority	Total
admin	11	0	0	11
alex	0	5	2	7
chris	0	5	4	9
dale	0	5	1	6
devon	0	2	1	3
jan	0	4	1	5
sandy	0	2	4	6



ClearQuest est Flexible



UCM: Améliore la communication et l'efficacité

- Le Chef de Projet est capable de
 - ▶ Extraire des métriques sur le projet
 - ▶ Équilibrer les charges
 - ▶ Communiquer les priorités
- Tous les membres de l'équipe savent
 - ▶ Quelles sont les activités prioritaires
 - ▶ Quelles sont les activités en cours
- Les projets sont pilotés par les demandes



Développeurs



Analystes



Testeurs



Intégrateur



Chef de projet



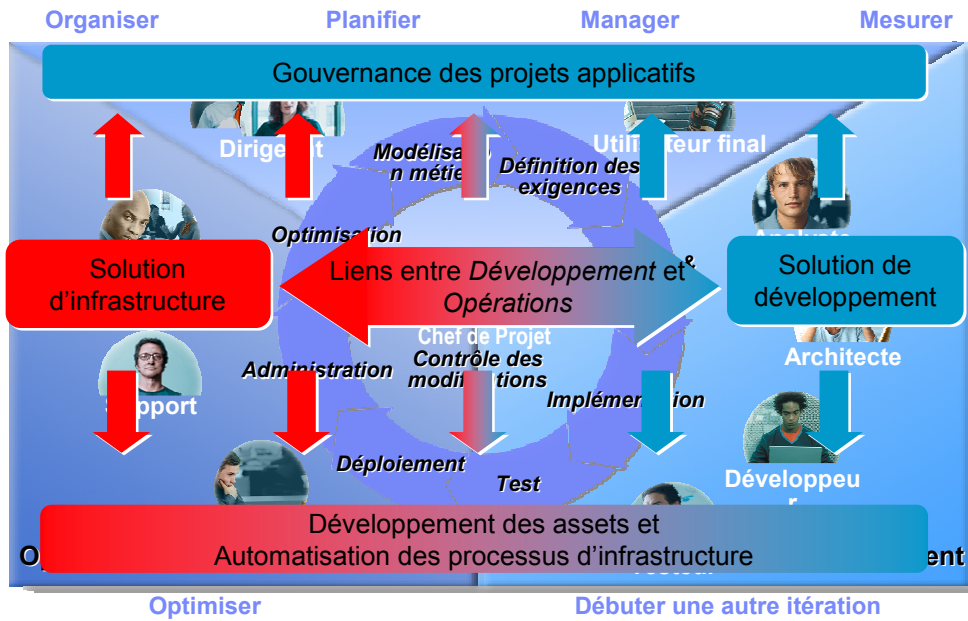
La solution de gestion de configuration UCM



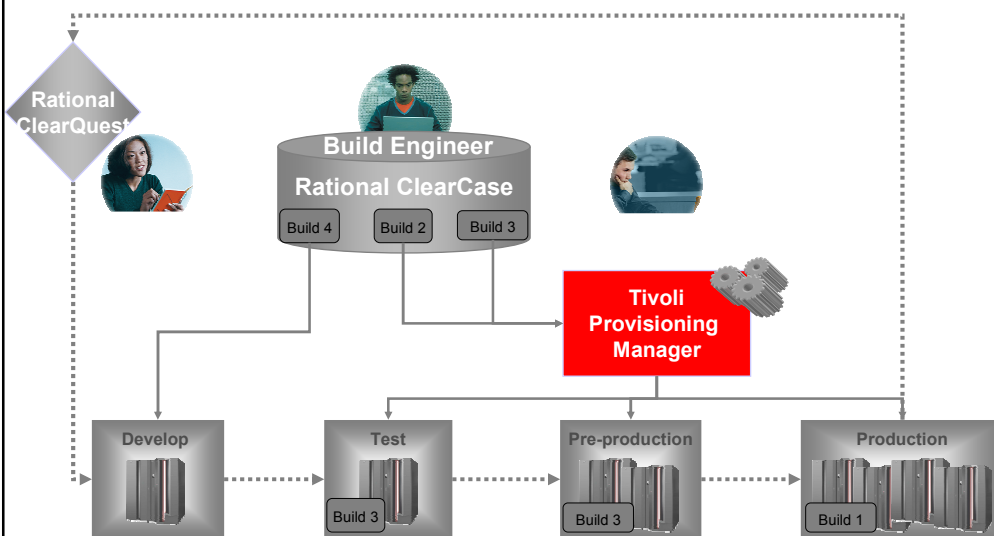
- Introduction
- Solution IBM de gestion des configurations
- Solution IBM de gestion des changements
- Solution IBM de gestion des déploiements
- Solution IBM de gestion des tests
- Utilisation sur sites distribués
- Paramétrage et administration de la solution
- Conclusion



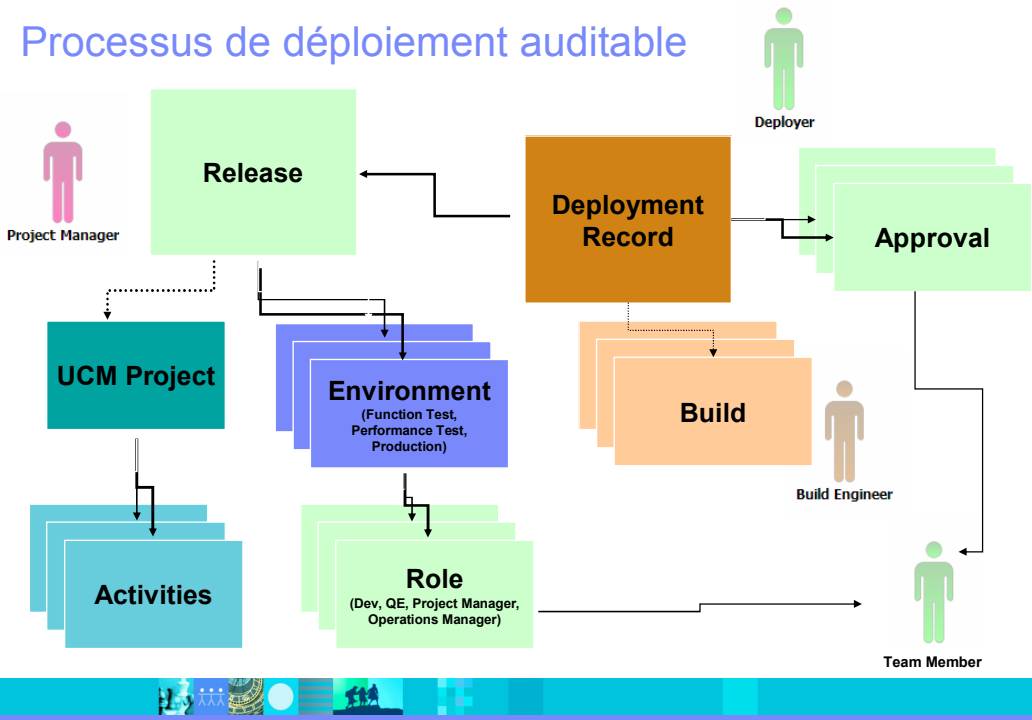
Le cycle de vie du développement logiciel



Processus de gestion des builds et déploiements



Processus de déploiement auditable



Deployment Record

The screenshot shows the Deployment Record interface with the following fields and annotations:

- State:** Ready (Annotation: État de la fiche de déploiement)
- Environment:** Unit Test (Annotation: Environnements cibles en fonction du workflow)
- Approval Status:** No Approval Required (Annotation: Status des approbations (non nécessaire dans ce cas) fonction du cycle de vie Avec éventuellement des e-signatures)
- Deployment_Unit_Reference:** ccase:/vobs/test_vob/duxml?OREF=oid:3A1a232d3a.101211da.8ef1 (Annotation: La version du "Deployment Units" identifie exactement les éléments du builds à déployer => Deployment Unit "template")

Other visible fields include: id (d200000008), Headline (Version 0.1 App), Submitter (admin), Submit Date (8/24/2005 11:53:39 AM), Owner (user), Application (RD_Bank_App), and Description.

Exemple de Deployment Unit

```
<?xml version="1.0" encoding="UTF-8" ?>
<tracking:deployment_unit
  xmlns:xsi="http://www.ibm.com/xml/SDK/Schema/instance"
  instanceID="build_20051102-20051102091635-0500" version="1.0"
  name="My Test Application" xmlns:tracking="http://www.ibm.com/deployment/schema/deploymentTracking.xsd">
  <history changeOperation="Decorate" changeNote="" user="medawson" hostname="melamine.yourco.com"
    datetime="2005-11-02T09:16:35-0500" />
  <history changeOperation="Generate" changeNote="" user="medawson" hostname="melamine.yourco.com" datetime="2005-11-02T09:13:38-0500" />
  <file path="/vobs/test_vob/shlib/foo.so" versionURI="ccase:/vobs/test_vob/shlib/foo.so?
    OREF=oid%3A286b43cd.291e11da.8af5.00%3A01%3A80%3Aa6%3Ad4%3A39%40
    vobuuid%3Aaf2b2c77.291a11da.8ae0.00%3A01%3A80%3Aa6%3Ad4%3A39
    &registry_server=melamine.yourco.com
    &region=thedomain_r_d_unix"
    csrule="main/LATEST"
    orig_path="foo.so" />
  ... other file path elements ...

  <CCOrigin>
    <ConfigSpec> element * CHECKEDOUT element * /main/LATEST</ConfigSpec>
  </CCOrigin>

  <BuildLog path="/vobs/test_vob/build_log_20051102.txt" versionURI="ccase:/vobs/test_vob/build_log_20051102.txt?
    OREF=oid%3A142327c7.4bac11da.8678.00%3A01%3A80%3Aa6%3Ad4%3A39%40
    vobuuid%3Aaf2b2c77.291a11da.8ae0.00%3A01%3A80%3Aa6%3Ad4%3A39
    &registry_server=melamine.yourco.com
    &region=thedomain_r_d_unix"
    vendor="IBM Build Logging" />
</tracking:deployment_unit>
```



Automatiser les déploiements avec Tivoli Provisioning Manager

The screenshot shows the Tivoli Provisioning Manager (TPM) interface. On the left, a 'Workflow Status' table is visible with columns for 'Execution id', 'Name', 'Status', and 'Datetime'. Below it, a yellow box contains the text: "Le statut de déploiement de TPM est enregistré". On the right, a browser window displays the 'Intelligent Orchestrator' interface, showing a tree view of servers and a table of server utilization. A callout box with the text "URL to TPM Web UI" points to the TPM Web URL field in the left window.

Execution id	Name	Status	Datetime
00000008			

Name	Application Tier	Pool	Server Utilization
ibm-p800-1	StateBank WebApp	WebSphere4-A05	92.9%
Oregon	StateBank WebApp	N/A	88.8%
Delaware	StateBank WebApp	N/A	88.8%
Arizona	StateBank Mail	N/A	77.4%
Montana	StateBank Mail	N/A	77.4%
NewYork	StateBank DB	N/A	29.6%
DC	StateBank DB	N/A	29.6%



La gestion des déploiements

- **ClearQuest** permet de piloter le déploiement de vos applications (C++, Java, Web) grâce aux packages :
 - ▶ [Build Tracking](#)
 - ▶ [Deployment Tracking](#)
- **ClearCase** permet de générer les templates des "Deployment Units" en fonction des environnements utilisés.
- Le package [TPM](#) de **ClearQuest** permet d'alimenter *Tivoli Provisioning Manager* avec les "Deployment Unit".
- Ces packages sont directement disponibles avec **ClearQuest**
[Documentation des packages de déploiement](#)



La solution de gestion de configuration UCM



- Introduction
- Solution IBM de gestion des configurations
- Solution IBM de gestion des changements
- Solution IBM de gestion des déploiements
- Solution IBM de gestion des tests
- Utilisation sur sites distribués
- Paramétrage et administration de la solution
- Conclusion



Améliorer la démarche qualité

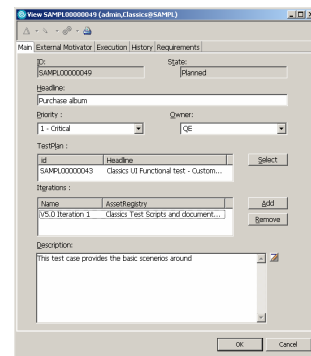
Ce que nous constatons souvent :

- Pas de centralisation des tests
 - ▶ Aucun partage des artefacts des tests entre le développement et la qualification
 - ▶ Trop peu de Tests de non régression (anomalies et "bugs cachés") lors des montées versions
 - ▶ Augmente le nombre d'aller – retour
- Scripts et environnements de tests très hétérogènes
 - ▶ Approche manuelle et non industrielle
 - ▶ Processus souvent peu ou pas défini
 - ▶ La mise en place des tests n'est pas systématique
 - ▶ Tests empiriques (risques/criticité, délais, prédictibilité) et non industriels



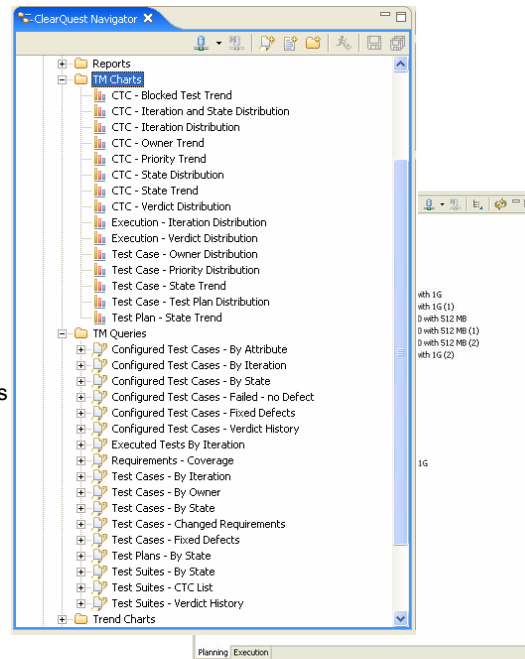
ClearQuest Test Management

- Solution de gestion des tests basée sur **ClearQuest**
 - ▶ Gestion des plans de tests et des cas de tests associés
 - ▶ Planification des campagnes de tests
 - ▶ Exécution par lancement des tests manuels et/ou automatiques
 - ▶ Gestion des résultats, des demandes de correction et du reporting
 - ▶ Rapports de couverture
- Intégration avec tous les outils de tests
 - ▶ Eclipse TP/TP
 - ▶ Rational Manual Tester
 - ▶ Tests Unitaires (JUnit)
 - ▶ Rational Functional Tester
 - ▶ Rational Performance Tester
 - ▶ Extensions pour supporter tout type de test
- Directement intégré sous Eclipse
- Solution directement disponible avec **ClearQuest**



Métriques CQTM

- Requêtes, graphiques et rapports prédéfinis
 - ▶ Rapport de couverture / suspicion
 - Impact d'un changement d'exigence sur les cas de test
 - Impact d'un changement ou d'une anomalie sur les cas de test
 - ▶ Status
 - Cas de test planifiés / implémentés
 - Status Ok / Ko
 - Cas de test Ko sans anomalie
 - Exigences sans cas de test
 - ▶ Tendance au cours du temps
 - Nombre de cas de test
 - Nombre de cas Ok / Ko



La solution de gestion de configuration UCM



- Introduction
- Solution IBM de gestion des configurations
- Solution IBM de gestion des changements
- Solution IBM de gestion des déploiements
- Solution IBM de gestion des tests
- Utilisation sur sites distribués
- Paramétrage et administration de la solution
- Conclusion



ClearCase MultiSite : autour du monde !



- Favorise le partage du travail entre plusieurs sites
 - Les référentiels sont "répliqués" sur les différents sites
- Supporte toute topologie et montée en charge
- Solution entièrement automatisée
- Chaque site travaille indépendamment des autres
- Largement utilisé
- Robustesse et fiabilité reconnues

Console d'administration MultiSite

The screenshot displays the Rational MultiSite Administration console. The main window shows the 'Synchronization Status for VOB: \dmm_testvobC1' as of Mon, 27 Jan 2003 16:28:06 UTC. It indicates 3 total replicas.

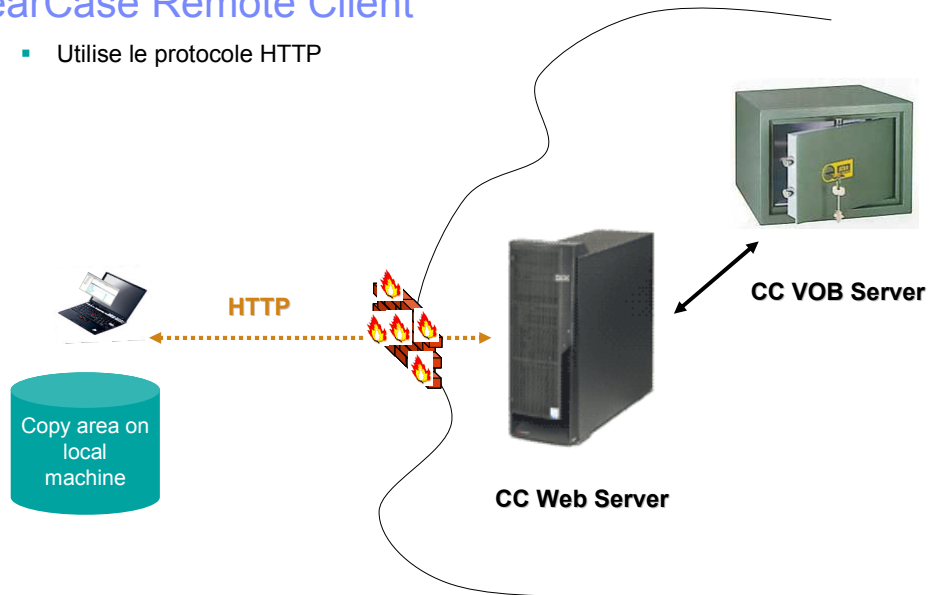
Replica	r1	r2	r3
r1 dmm_testrepC1 Botall	—	0/0	2/5
r2 dmm_testrepC2	2/8	—	0/0
r3 dmm_testrepC3	2/8	0/0	—

Below this, it shows 'Most Recent Synchronization Attempts for Replica-dmm_testrepC1 /Server-roadrunner' as of Mon, 27 Jan 2003 16:28:06 UTC. It lists 3 sync attempts:

Sync Ended	Status	Sibling Replica	Sibling Replica Server	Kind	Type
Mon, 27 Jan 2003 11:30:02 UTC	Failed	dmm_testrepC3	salty	import	VOB packet
Mon, 27 Jan 2003 11:15:02 UTC	Failed	dmm_testrepC2	astoria	export	VOB packet
Mon, 27 Jan 2003 11:30:02 UTC	Failed	dmm_testrepC3	salty	export	VOB packet

ClearCase Remote Client

- Utilise le protocole HTTP



La solution de gestion de configuration UCM



- Introduction
- Solution IBM de gestion des configurations
- Solution IBM de gestion des changements
- Solution IBM de gestion des déploiements
- Solution IBM de gestion des tests
- Utilisation sur sites distribués
- Paramétrage et administration de la solution
- Conclusion



Personnaliser IBM Rational ClearQuest

- S'adapte à votre processus
 - ▶ Modèle à états entièrement personnalisable
- S'adapte à votre utilisation : personnalisation complète des informations
 - ▶ Champs (de la base de données)
 - ▶ Formulaires de saisies
 - ▶ Requêtes, graphiques
 - ▶ Formats de rapports
- Interface programmable COM, Perl et Java (interopérabilité)



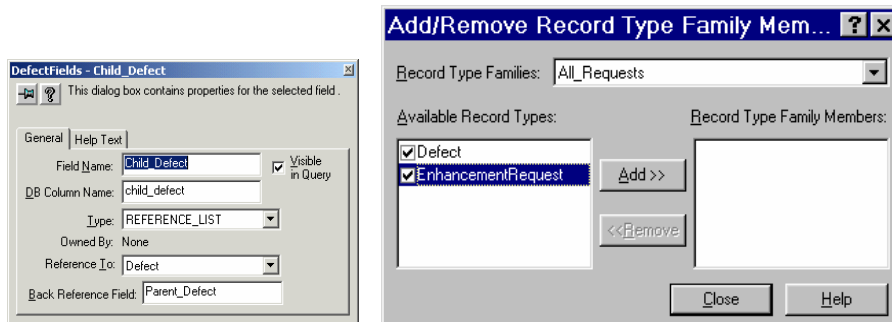
Personnaliser IBM Rational ClearQuest

- Choisir entre les schémas disponibles "out of the box"
- Étendre les fonctionnalités en utilisant les packages prédéfinis:
 - ▶ Enregistrements pré définis:
 - ▶ Clients
 - ▶ Améliorations
 - ▶
 - ▶ Notification mail
 - ▶ Intégrations outils
 - ▶ Gestion des fichiers attachés
 - ▶ Gestion de l'historique, des notes
 - ▶ Audit et signatures électroniques
 - ▶ Package de gestion des builds et des déploiements
 - ▶ Package de gestion des tests



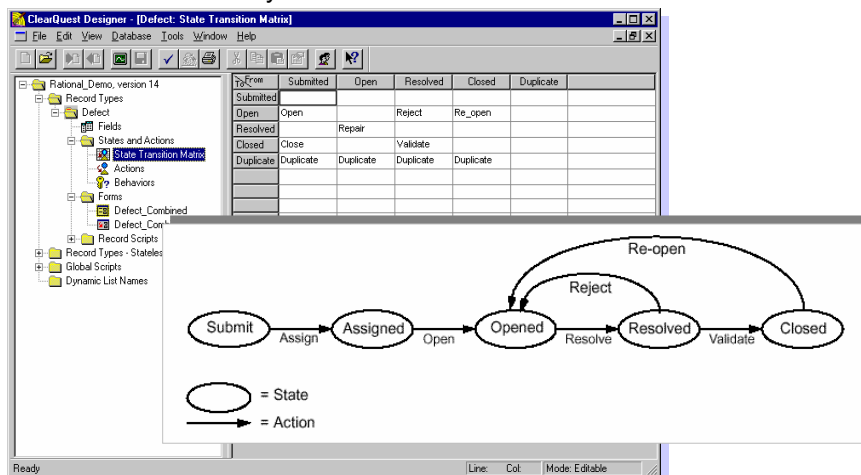
Types d'enregistrements configurables

- Création de multiples types d'enregistrements
 - ▶ Avec états (processus, défaut)
 - ▶ Sans états (sans processus, info client)
 - ▶ Familles d'enregistrements
 - ▶ Liaison entre enregistrements (granularité des demandes)



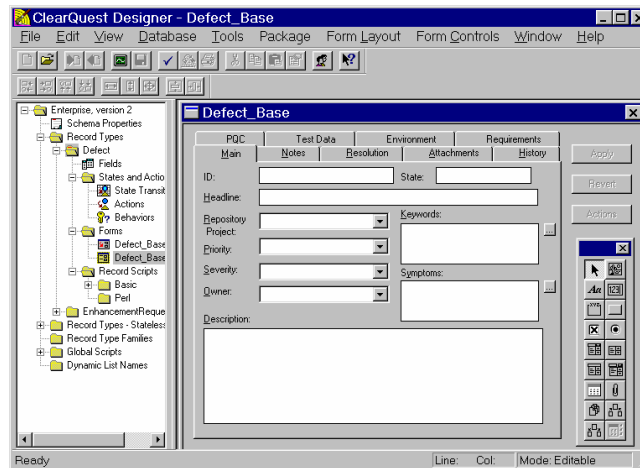
Matrice des états configurable

- ClearQuest utilise une machine à états pour contrôler le flux des fiches de votre système

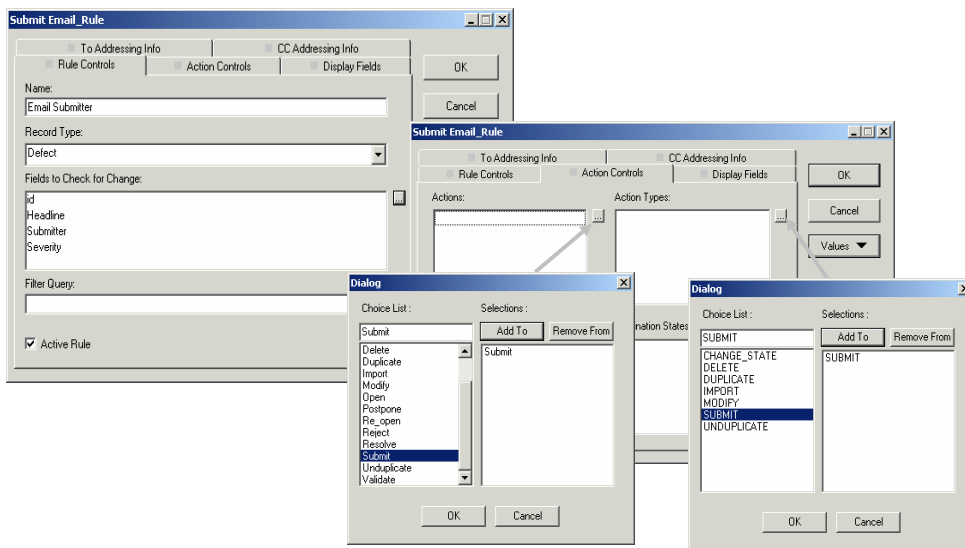


Modifier les formulaires

- Editeur intégré de formulaires (drag & drop)



Notification par e-mail



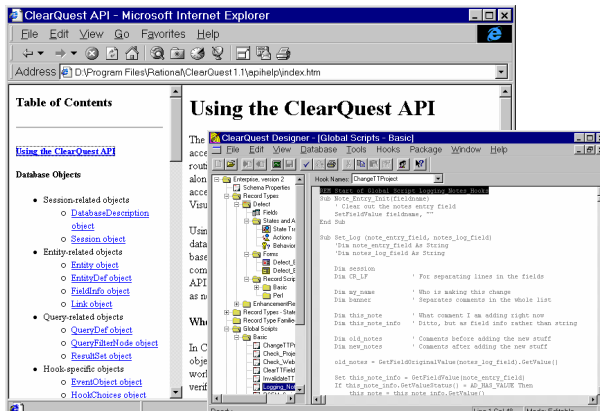
Possibilités d'intégration avec ClearQuest

- ClearQuest est un outil ouvert
 - ▶ Il permet d'appeler des applications depuis l'interface de ClearQuest ou d'effectuer des contrôles, validation, etc...
 - Sur des actions spécifiques
 - Sur des hooks (modifications de champs, validation, etc.)
 - Via des ActiveX
 - ▶ ClearQuest peut être appelé depuis l'extérieur
 - Via une API complète et documentée (COM, Perl, Java)
 - Toutes les actions sont disponibles au travers de l'API
 - IBM Rational Team API : API Java (standard JSR 147) commune à ClearCase, ClearQuest, RequisitePro
 - ▶ Via au service MailReader (création et modification de fiche par envoi de mails formatés)



Étendre votre système

- Automatiser et contrôler à l'aide de hooks sur les champs, les actions, etc...
- Éditeur intégré de scripts
- VBScript et Perl
- Documentation avec exemples intégrés
- Base de scripts : <http://www.ibm.com/developerworks/rational/library/4236.html>



Possibilités d'intégration avec ClearQuest

- Appel d'une application dans le cadre d'une fonction ClearQuest

```
Function Defect_Calculator(param)
    ' Param as Variant

    Dim MyShell
    Dim shell_cmd

    shell_cmd="C:\WINDOWS\system32\calc.exe"
    set MyShell=CreateObject("WScript.Shell")
    MyShell.Run shell_cmd, 1, 1
End Function
```

- ▶ Connexion de la fonction à l'action ClearQuest « Calculer »

Action Name	Type	Access Control	Initialization	Validation	Commit	Notification	Record Scripts
Submit	SUBMIT	All Users	BASIC	BASIC			
Open	CHANGE_STATE	All Users		BASIC			
Resolve	CHANGE_STATE	All Users		BASIC			
Assign	CHANGE_STATE	All Users		BASIC			
Créer Aide de Référence	RECORD_SCRIPT_ALIAS	N/A	N/A	N/A	N/A	N/A	
Calculer	RECORD_SCRIPT_ALIAS	N/A	N/A	N/A	N/A	N/A	PopUpFiche/Aide Calculer
ModifHistory	BASE	All Users		BASIC			
Analysér	CHANGE_STATE	All Users					



Création d'une demande depuis l'extérieur

```
Dim oAppli As CQApplication
Dim oSession As CQSession

Private Sub Form_Load()
    Set oAppli = New CQApplication
    Set oSession = oAppli.GetSession
    oSession.Logon
End Sub

Private Sub Command1_Click()
    Dim session
    Dim MyEntity

    ' Création d'un nouveau "Defect"
    Set session = oSession.OAdSession
    Set MyEntity = session.BuildEntity("Defect")

    ' Modification des champs
    MyEntity.SetFieldValue "Headline", Text1.Text
    MyEntity.SetFieldValue "Description", Text2.Text
    MyEntity.SetFieldValue "Severity", Combo1.Text

    ' On récupère l'ID de la fiche créée
    ID = MyEntity.GetFieldValue("Id").GetValue

    ' On commit la création dans la base ClearQuest
    MyEntity.Validate
    MyEntity.Commit
    MsgBox "La fiche " & ID & " a été créée avec succès"
End Sub
```



Possibilités d'intégration avec ClearQuest

- Exemple d'appel externe de ClearQuest

- ▶ Connexion à ClearQuest

```
Set oAppli = New CQApplication
Set oSession = oAppli.GetSession
oSession.Logon "", "", "test", "maquettel"
```

- ▶ Exécution d'une requête et affichage des résultats

```
Set workspaceObj = session.GetWorkSpace
Set QueryDef = workspaceObj.GetQueryDef("Public Queries/All demandes")
Set resultSet = session.BuildResultSet(QueryDef)
resultSet.Execute
```

```
Status = resultSet.MoveNext
Do While Status = 1
    firstColumn = resultSet.GetColumnValue(2)
    ...
```

- ▶ Affichage d'une fiche

```
Set TheForm = oSession.CreateForm("Envt", Num)
```



Possibilités d'intégration avec ClearQuest

- Exemple d'utilisation de l'API JNI de ClearQuest

```
import com.rational.clearquest.cgjni.*;

public class JniTest {
    public static void main(String[] args) {
        try {
            // Load the native implementation of the Java API for ClearQuest
            System.loadLibrary("cgjni");
            CQClearQuest cq = new CQClearQuest();
            try {
                CQSession session = cq.CreateUserSession();
                session.UserLogon("admin", "", "SAMPL", "2003.06.00");
                CQEntity entity = session.GetEntity("Defect", "SAMPL00000040");
                System.out.println("State = " + entity.GetFieldValue("State").GetValue());
            } catch (Exception e) {
                // TODO handle exception here
                System.out.println(e);
            }
        } catch (UnsatisfiedLinkError e) {
            e.printStackTrace();
        }
    }
}
```



Possibilités d'intégration avec ClearQuest

- Exemple d'utilisation du mail reader

- ▶ Exemple de soumission d'une fiche de type « defect »

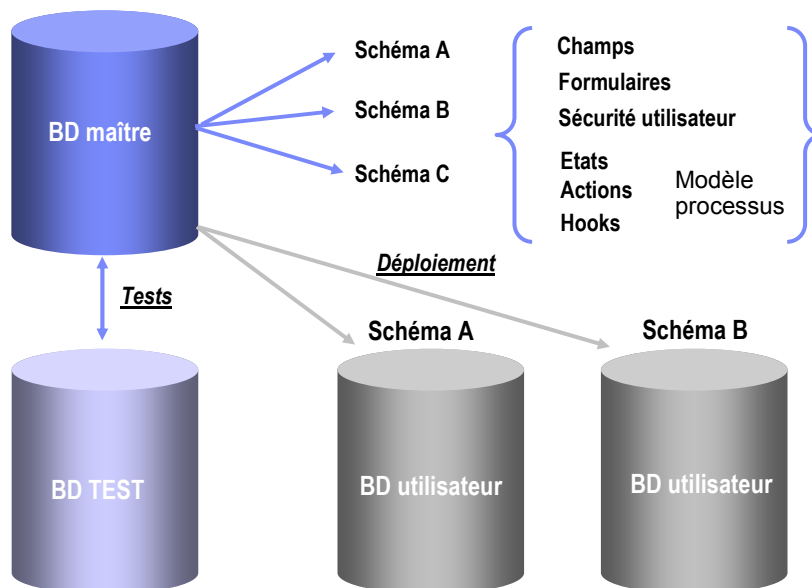
```
Subject:
defect submit
Body:
Headline: inventory report is not running correctly
Severity: 1-Critical
Project: ClassicPOS
{Description: When running an inventory report, application crashes if more than
50 items are included in the report. }
Priority: 2-Give High Attention
```

- ▶ Exemple de clôture du defect n°SAMPL00001234

```
Subject:
defect closed SAMPL00001234
Body:
Note_Entry: Followed the steps outlined in the description, and confirmed that
the error no longer occurs.
Test Method: Manual
```

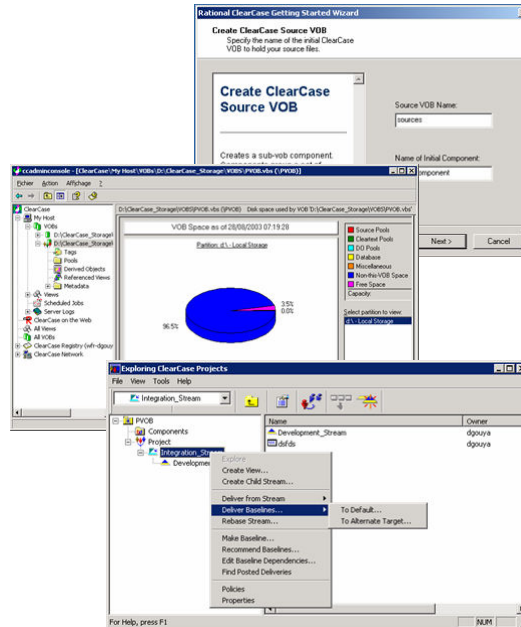


Schémas et bases de données utilisateur



Facilité d'administration de ClearCase

- **Un assistant d'aide** au démarrage vous permet d'initialiser et de configurer facilement votre référentiel
- **Une console d'administration** regroupe l'ensemble des fonctionnalités de surveillance et de gestion du référentiel
- **Un explorateur de projet** centralise la définition et la gestion des projets, des composants, des espaces de travail et des livraisons



Puissance de paramétrage de ClearCase

- Simplicité de paramétrage d'UCM
- Puissance de l'utilisation des méta données (attributs, links, etc...)
- Performance de l'utilisation des triggers (événementiel)
 - ▶ Pré opération (Contrôle)
 - ▶ Post opération (Notification)
 - ▶ Toutes les opérations sont supportées
 - ▶ Liste exhaustive de toutes les informations disponibles

La solution de gestion de configuration UCM



- Introduction
- Solution IBM de gestion des configurations
- Solution IBM de gestion des changements
- Solution IBM de gestion des déploiements
- Solution IBM de gestion des tests
- Utilisation sur sites distribués
- Paramétrage et administration de la solution
- Conclusion



Les avantages de la solution UCM (1)

- Pour les développeurs
 - ▶ Rapidité de création des espaces de travail
 - ▶ Sélection automatique des bonnes versions sur lesquelles travailler
 - ▶ Simplifier l'accès simultané à différents espaces de travail
 - ▶ Gestion automatique des branches dans le cas de développements en //
 - ▶ Intégration directe avec les IDE

 - ▶ Connaître automatiquement les tâches sur lesquelles travailler grâce aux priorités définies dans ClearQuest
 - ▶ Couplage automatique des modifications avec les activités (activité courante)
 - ▶ Suppression des erreurs de livraison grâce à la livraison des activités



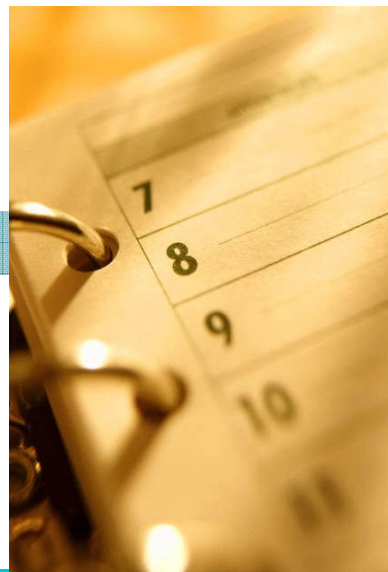
Les avantages de la solution UCM (2)

- Pour les intégrateurs
 - ▶ Faciliter de création des baselines par sélection des activités
 - ▶ Faciliter d'audit grâce aux changeSet des activités
 - ▶ Gestion des builds et des niveaux de promotion des baselines
 - ▶ Gestion des projets par versions de composants
 - ▶ Gestion des déploiements au travers des packages ClearQuest
 - ▶ Facilité de mise à disposition des versions à tester
- Pour les chefs de projets
 - ▶ Connaître l'avancement des tâches (mise à jour automatique de l'état des activités gérées par ClearQuest – ouvert, résolue, intégré, etc.)
 - ▶ Savoir qui fait quoi (liste des activités par projet, par stream, par team, par développeur)
 - ▶ Meilleure vision et meilleure anticipation pour conduire correctement les projets



Plan

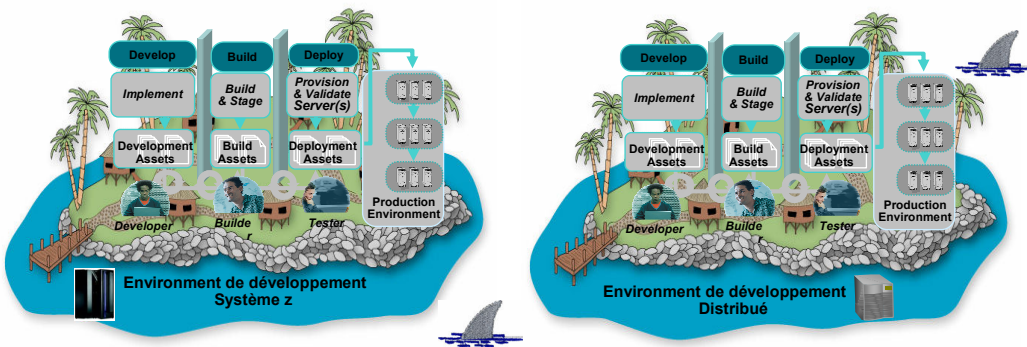
- Rappel des besoins de GIPS
- Présentation de la solution de gestion de configuration UCM
- Proposition d'architecture
- Démonstration
- Assistance au déploiement
- Conclusion



La réalité d'aujourd'hui – ce que vous avez ...

Des îlots de développements pour Mainframe et distribués

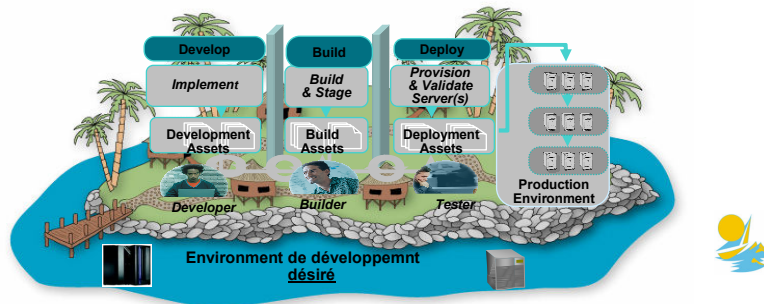
- ▶ Des processus uniques dans chaque îlot; faible communication entre îlots
- ▶ Bonnes pratiques et processus non partagés entre îlots
- ▶ Différents outils et infrastructures créés des compétences inflexibles, ce qui inhibe la flexibilité de l'IT
- ▶ Fardeau financier à maintenir la cohérence des deux environnements



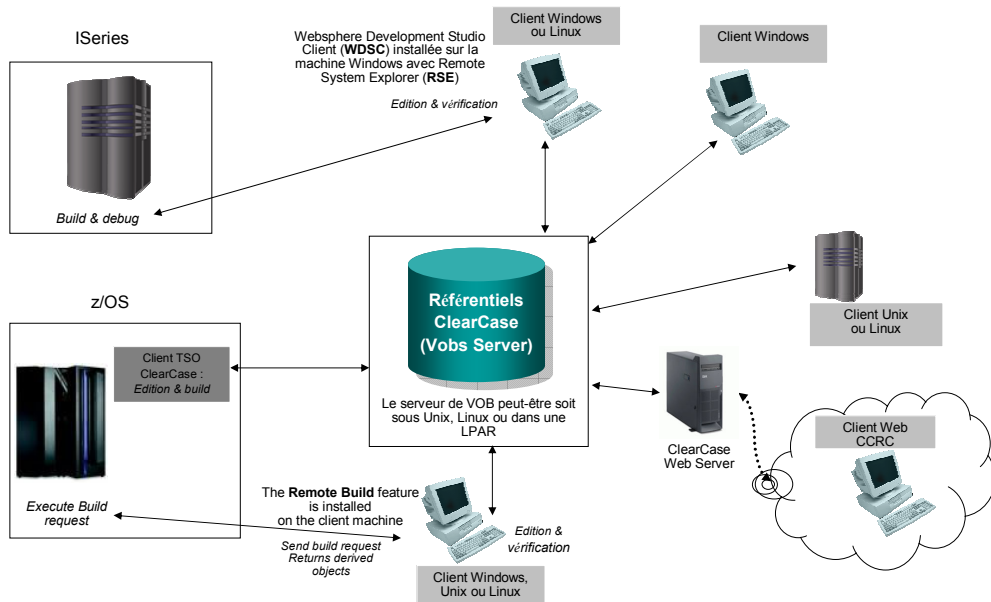
Etape future – ce que vous voulez ...

Environnement de développements partagé entre Mainframe et distribués

- ▶ Exploiter une seule infrastructure pour les développements Mainframe ET distribués. Gestion des développements multi-sites.
- ▶ Bénéfice d'une réduction des coûts dûs à l'élimination d'outils et processus dupliqués
- ▶ Pilotage de la gestion des demandes
- ▶ Améliorer communication et traçabilité et reporting de bout en bout
- ▶ Améliorer la flexibilité de l'IT car les compétences des équipes peuvent être accrues à travers l'organisation



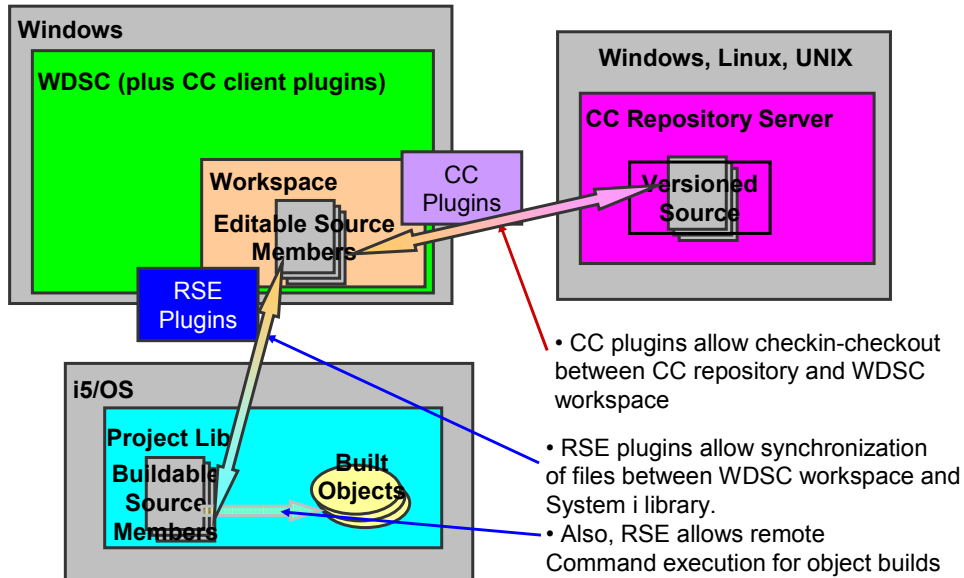
Architecture proposée



Comment travailler avec les ISeries

- Pour les développements autour d'Adelia
 - ▶ Utilisation d'une base de donnée pour sauvegarder les artefacts
 - ▶ Versionning des exports de base de données lors des releases importantes
- Pour les autres developments :
 - ▶ Nous recommandons d'utiliser WSDC (Websphere Studio Developer Client) et RSE (Remote Server Extension) afin :
 - d'éditer les fichiers sur le poste de développement (Checkout - Edit - Verify - Checkin)
 - Utiliser WSDC pour se connecter au serveur ISeries et construire et déboguer les programmes.

Utilisation de WDSC avec ClearCase - System i



Comment travailler sur z/OS

Il y a deux manières de travailler sur z/OS :

- Le « **client TSO** »
 - ▶ Il fournit accès au référentiel ClearCase au travers de l'interface native ISPF sur le mainframe.
 - ▶ Le processus de gestion de configuration est consistant entre le mainframe et le distribué – utilisation de UCM.
- Le « **Remote Build** »
 - ▶ Cela permet de soumettre des requêtes d'exécution à partir d'un client ClearCase vers le z/OS (MVS, USS). Les objets dérivés sont retournés sur le client ClearCase où ils peuvent être versionnés.
 - ▶ Il est aussi possible d'utiliser les fonctionnalités d'audit de ClearCase (clearmake et clearaudit)

Utilisation du client TSO

```

Command ==>
Rational Clearcase Mainframe Connectors TSO Client
Select Action for Member: 2.2.1

Option ==>
1. Check Out          4. Compare With Previous Version
2. Check In          5. Show Version Tree
3. Undo Check Out    6. Add to Source Control

Comment for check out and check in option:

PF01=ClearCase Help          PF03=Return to ViewFile List
    
```

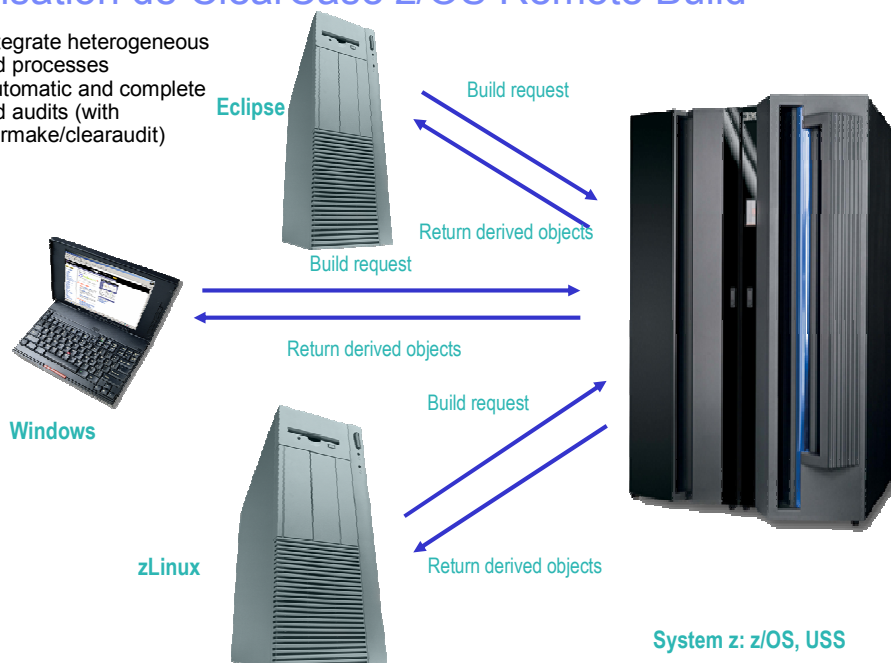
Accéder et gérer les éléments avec ClearCase UCM à partir d'ISPF

Les utilisateurs TSO peuvent utiliser ClearCase dans un environnement familier.



Utilisation de ClearCase z/OS Remote Build

- Integrate heterogeneous build processes
- Automatic and complete build audits (with clearmake/clearaudit)





TSO Client

Remote Build



Plan

- Rappel des besoins de GIPS
- Présentation de la solution de gestion de configuration UCM
- Proposition d'architecture
- Démonstration
- Assistance au déploiement
- Conclusion



Les Services IBM Rational : quel objectif ?

▪ S'assurer du succès des projets de nos clients

▶ Avant les choix d'outils

- Par des études de l'adéquation des solutions avec les besoins

▶ Pendant le déploiement

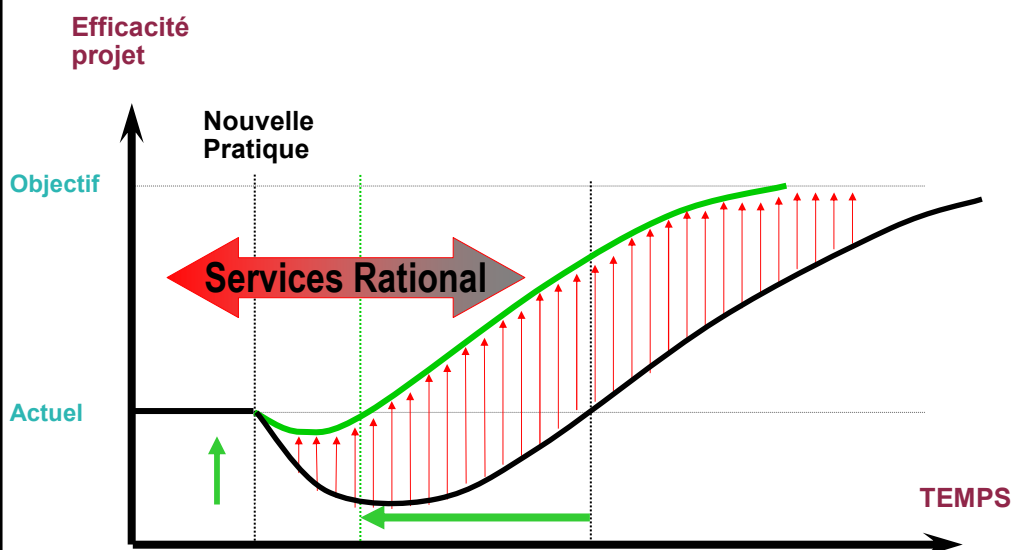
- En accompagnant les utilisateurs dans la mise en œuvre des solutions
- En mettant en place un transfert de compétence

▶ Après le déploiement

- En accompagnant les projets dans leurs étapes importantes
- En proposant un support supplémentaire



Mise en œuvre :



Approche des déploiements UCM

- **Un déploiement associé à un transfert de compétence coordonné en cinq étapes**
 - ▶ Définition de l'équipe projet IBM Rational
 - ▶ Formation de l'équipe chargée du déploiement
 - ▶ Préparation du plan de déploiement global (PGCL)
 - ▶ Accompagnement de l'équipe système pour les premiers déploiements
 - ▶ Accompagnement global et support des équipes système locales



Définition du PGCL

Plan de Gestion de Configuration Logicielle

- Le PGCL permet de définir clairement :
 - ▶ L'infrastructure matériel et réseau et l'environnement logiciel
 - ▶ Les processus de gestion des demandes changements
 - ▶ Les processus statiques et dynamiques de gestion des configurations
 - ▶ Les procédures de suivi et d'audit de ces processus
 - ▶ Le déploiement des processus et des outils
 - ▶ Les tâches de maintenance, de packaging et d'installation
 - ▶ Le support aux utilisateurs
 - ▶ La formation des nouveaux projets
 - ▶ La définition des rôles et les responsabilités



PGCL : Définition des rôles et responsabilités

- Dans les équipes IT:
 - ▶ **Administrateur ClearCase/ClearQuest & Système** (installation et maintenance des serveurs, des licences et des repositories, gestion des users)
 - ▶ **Responsable de la GCL** (définition du PGCL, organisation des composants, définition des packages de déploiement, paramétrage, support aux utilisateurs, formation, guides d'utilisation, interface technique IBM Rational)
- Dans les équipes HelpDesk
 - ▶ **Responsable des installations**
- Dans le management:
 - ▶ **Sponsor du projet de déploiement**
- Dans les équipes projets:
 - ▶ **Chefs de projet**
 - ▶ **Intégrateurs, Testeurs, Développeurs**



Plan

- Rappel des besoins de GIPS
- Présentation de la solution de gestion de configuration UCM
- Proposition d'architecture
- Démonstration
- Assistance au déploiement
- Conclusion



MERCI

