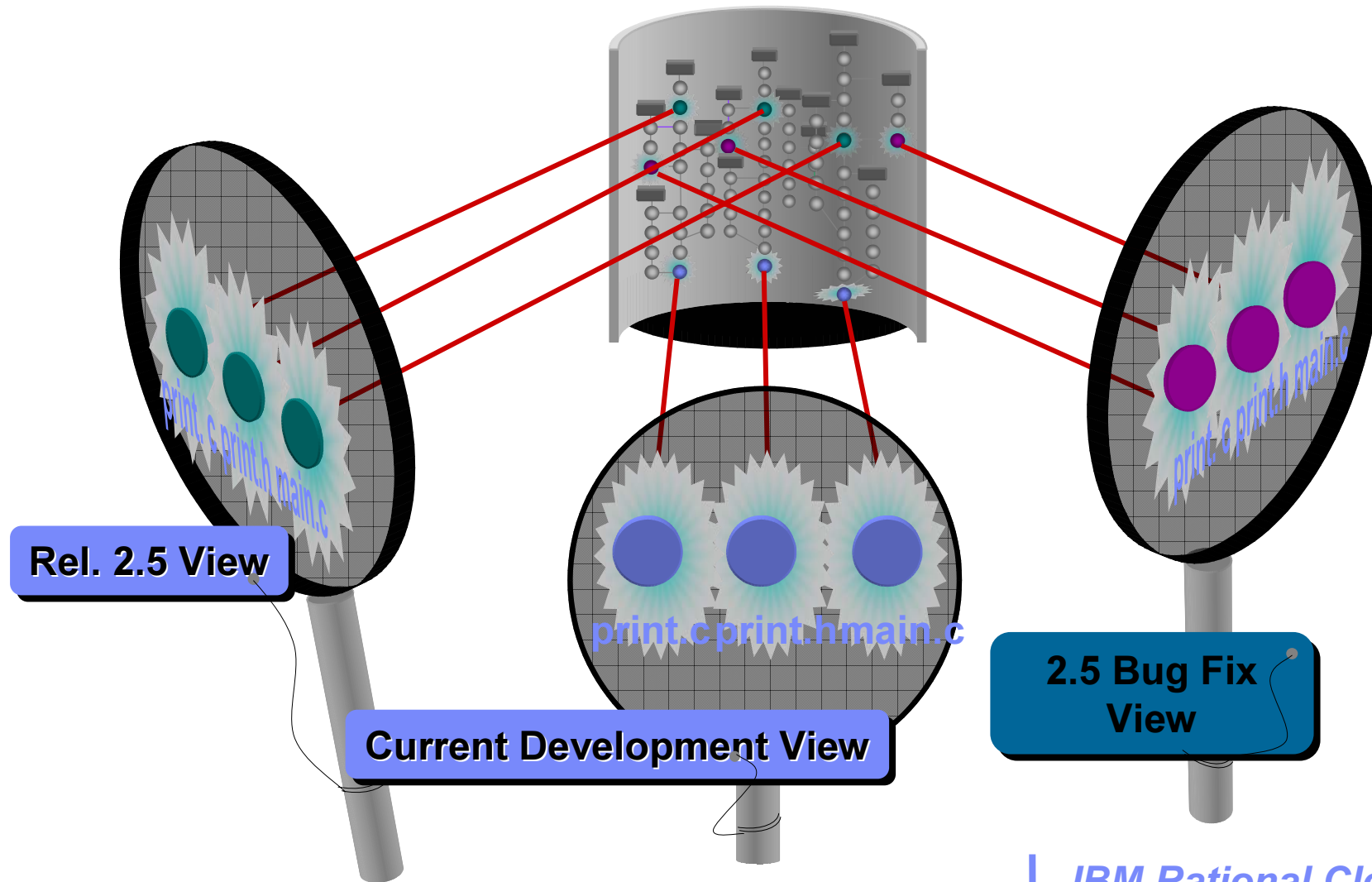
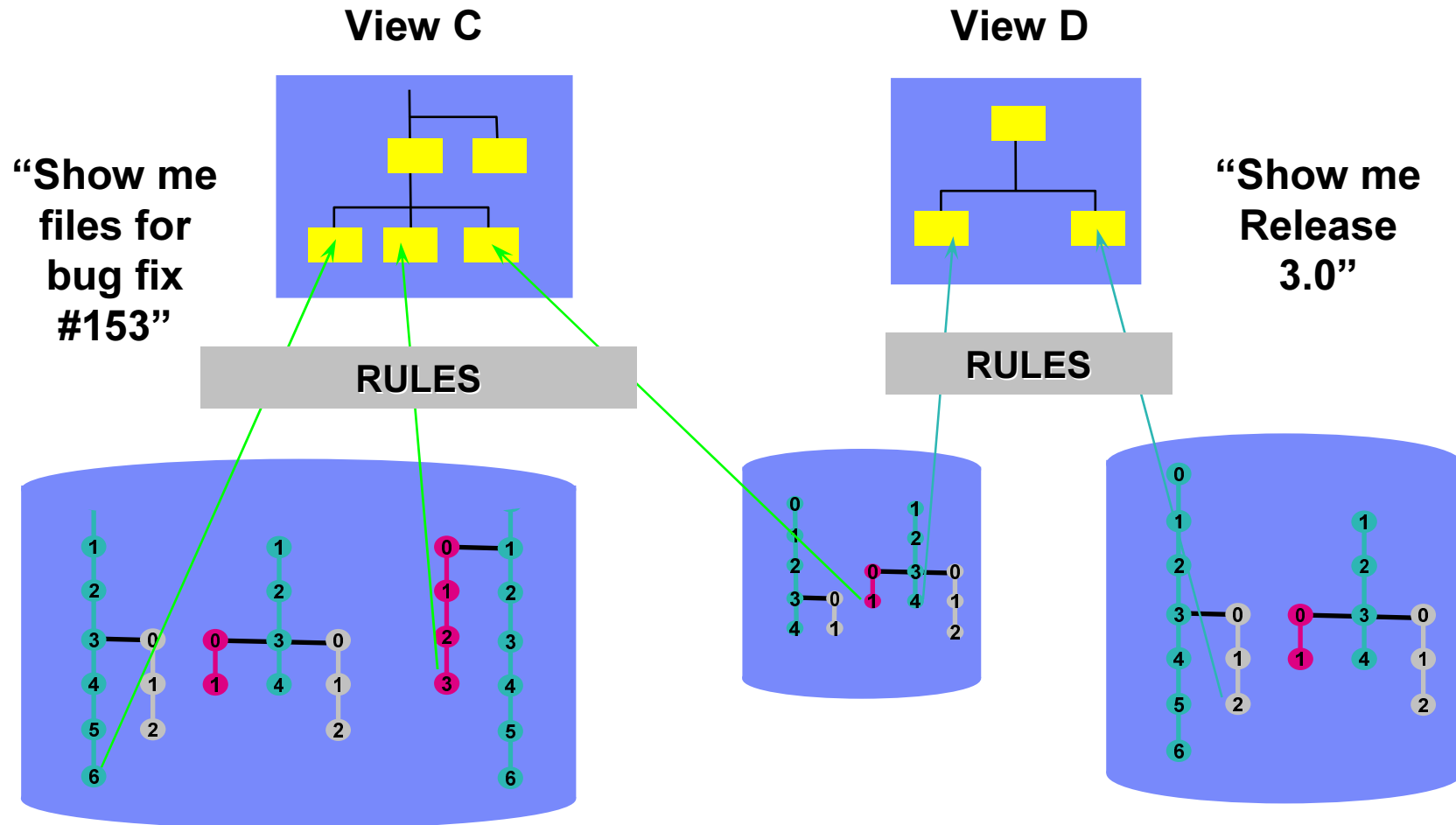


# Easy to Use: ClearCase Views

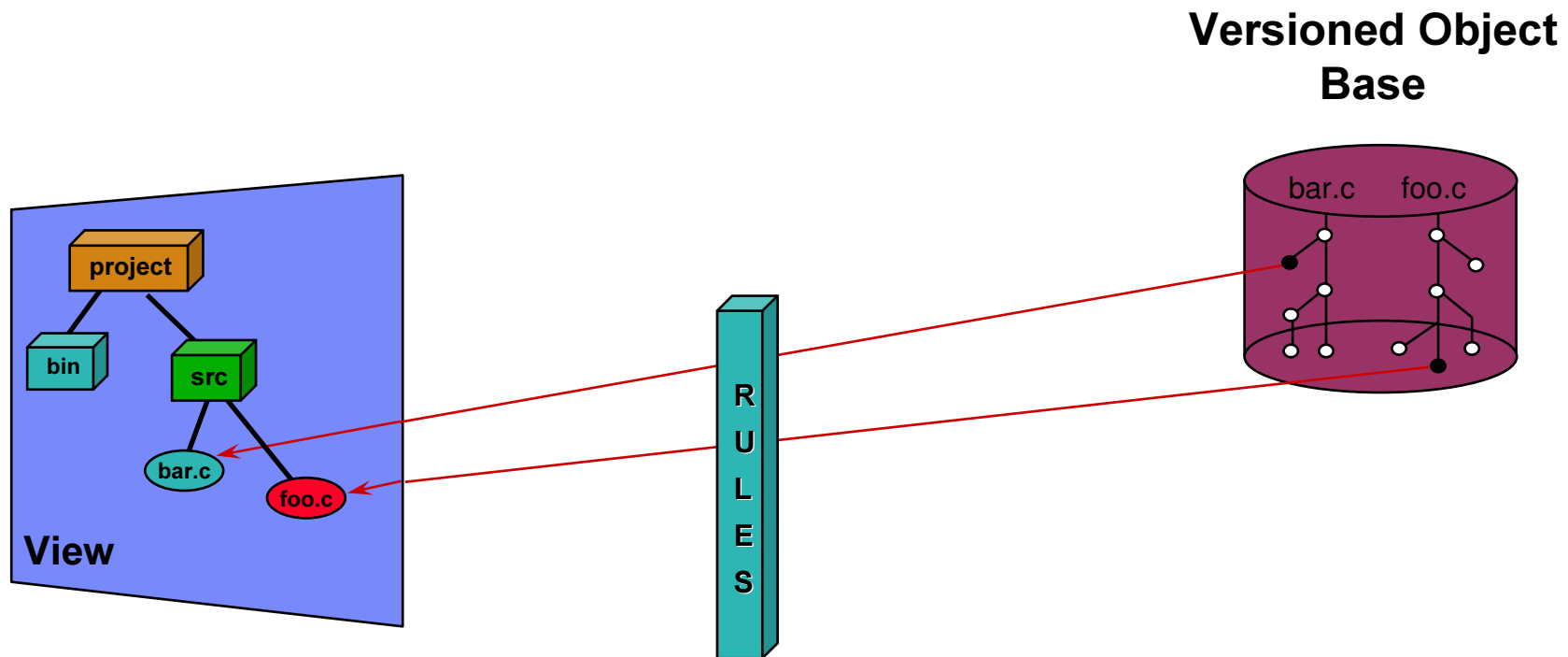


# ClearCase views are configurations



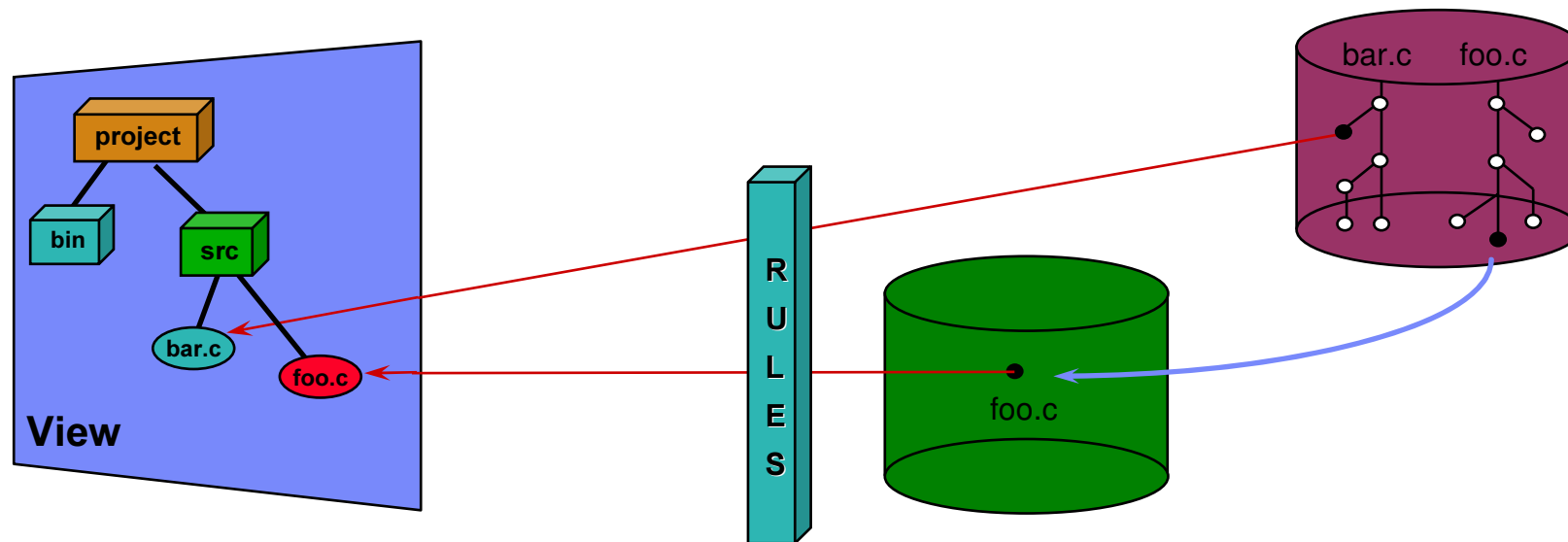
## Easy to use: Dynamic views transparent access

- View Before Checkout of foo.c

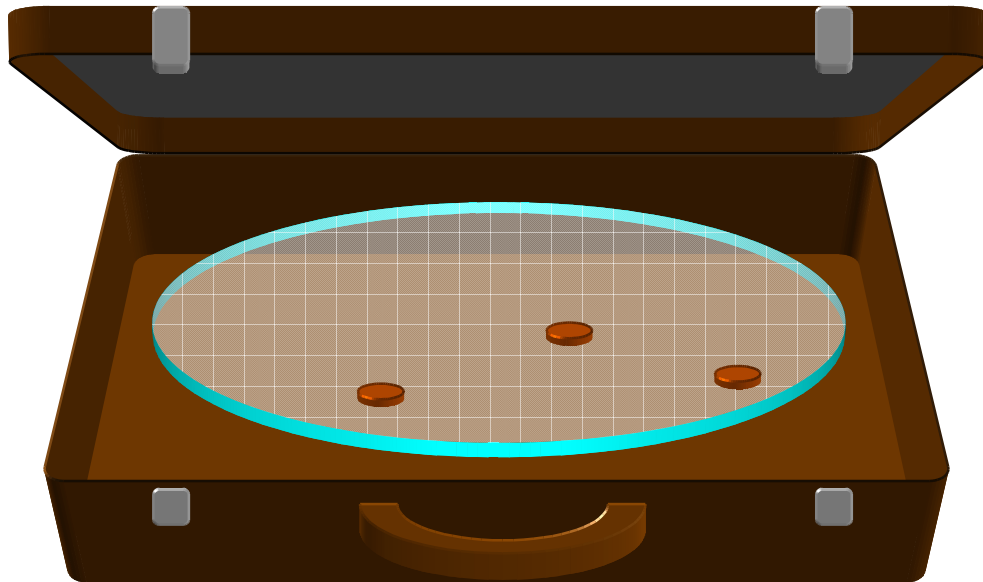


## Easy to use: Dynamic views transparent access

- View After Checkout of foo.c



## Easy to use: ClearCase snapshot views



- Snapshot Views enable reliable disconnected use
- Makes a controlled copy of the right configuration on your disk
  - Still under ClearCase control
  - Native filesystem access times
- Work away from office
  - Periodic updates only
  - Local builds
  - Automatically re-synchronizes your work with the ClearCase repository



# ClearCase/ClearQuest and Eclipse (CCRC)

The screenshot shows the ClearCase Remote Client interface. On the left is the 'ClearCase View Navigator' showing a tree view of project files. The main pane is titled 'ClearCase Details' and shows a file list with columns for Name, Size, Kind, and Modified Time. A tooltip for 'Fonts.java' shows its state as 'Loaded'. At the bottom, the 'ClearCase View Configuration, ClearCase History Browser, etc.' pane shows a table of files with columns for Name, State, and Path.

Name	Size	Kind	Modified Time
AdminFrame.java	3051	File Element Version	Oct 19, 2004
CDAdmin.java	2313	File Element Version	Oct 19, 2004
CDMaintPane.java	17189	File Element Version	Oct 19, 2004
CustomerMaintPane.java	17452	File Element Version	Oct 19, 2004
Fonts.java	3612	File Element Version	Oct 19, 2004
Logon.java		File Element	Oct 19, 2004
TaxMaintPane.java		File Element Version	Oct 19, 2004

Name	State	Path
.project	Loaded	C:\Documents and Settings\demo\stef_Rel3\Classics\Source\com\rational\cdshop\admin\project
bkg_gray.jpg	Loaded	C:\Documents and Settings\demo\stef_Rel3\Classics\html\images\bkg_gray.jpg
html	Loaded	C:\Documents and Settings\demo\stef_Rel3\Classics\html
images	Loaded	C:\Documents and Settings\demo\stef_Rel3\Classics\html\images
index.htm	Loaded	C:\Documents and Settings\demo\stef_Rel3\Classics\html\index.htm
maintitle_beethoven.gif	Loaded	C:\Documents and Settings\demo\stef_Rel3\Classics\html\images\maintitle_beethoven.gif
maintitle_brass.gif	Loaded	C:\Documents and Settings\demo\stef_Rel3\Classics\html\images\maintitle_brass.gif
maintitle.gif	Loaded	C:\Documents and Settings\demo\stef_Rel3\Classics\html\images\maintitle.gif
Privacy_Policy.htm	Loaded	C:\Documents and Settings\demo\stef_Rel3\Classics\html\Privacy_Policy.htm
ThankYou.htm	Loaded	C:\Documents and Settings\demo\stef_Rel3\Classics\html\ThankYou.htm



Eclipse



UNIX/Linux



ISPF



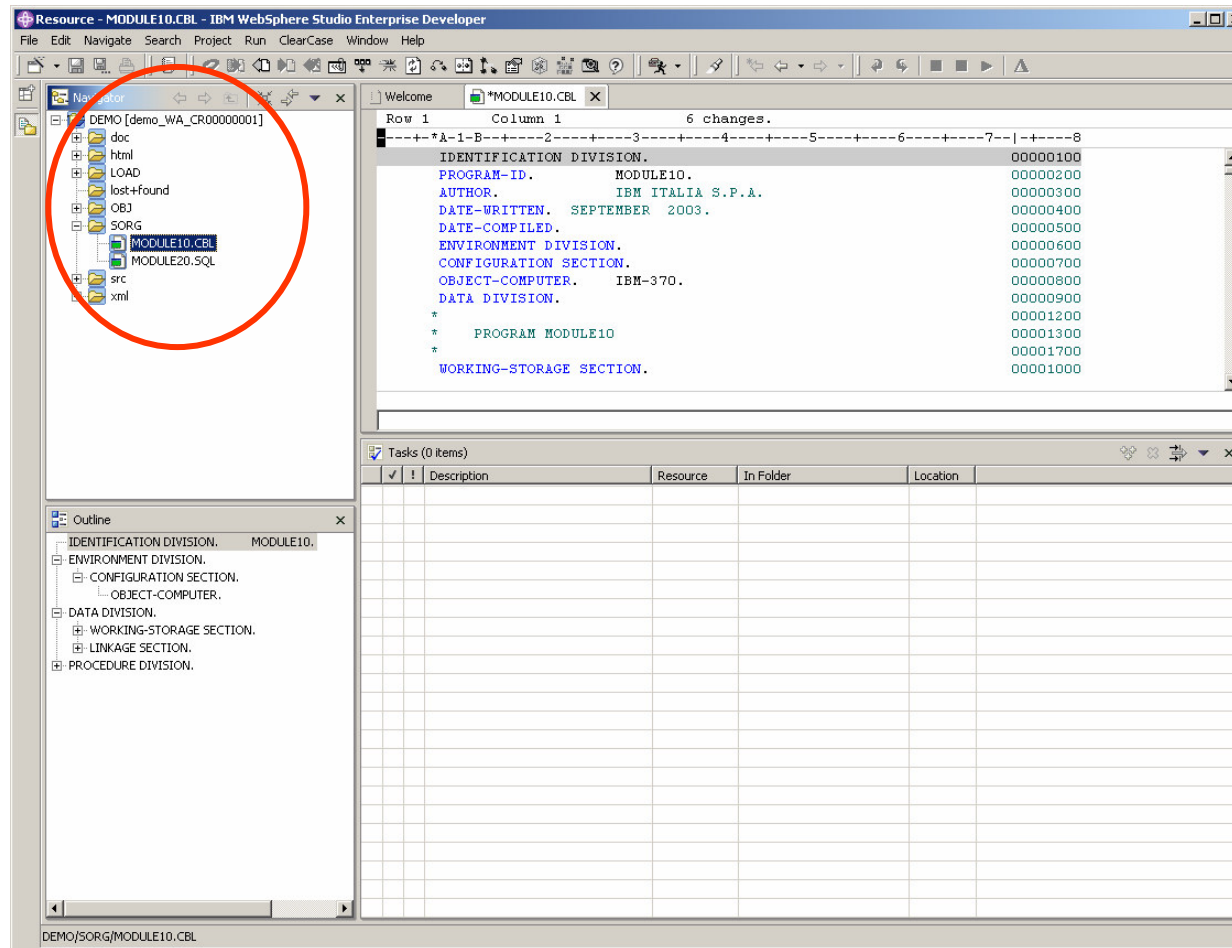
CLI



Browser



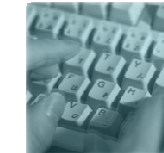
# ClearCase/ClearQuest and Eclipse (WDz - MVS)



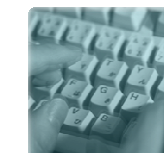
Eclipse



UNIX/Linux



ISPF



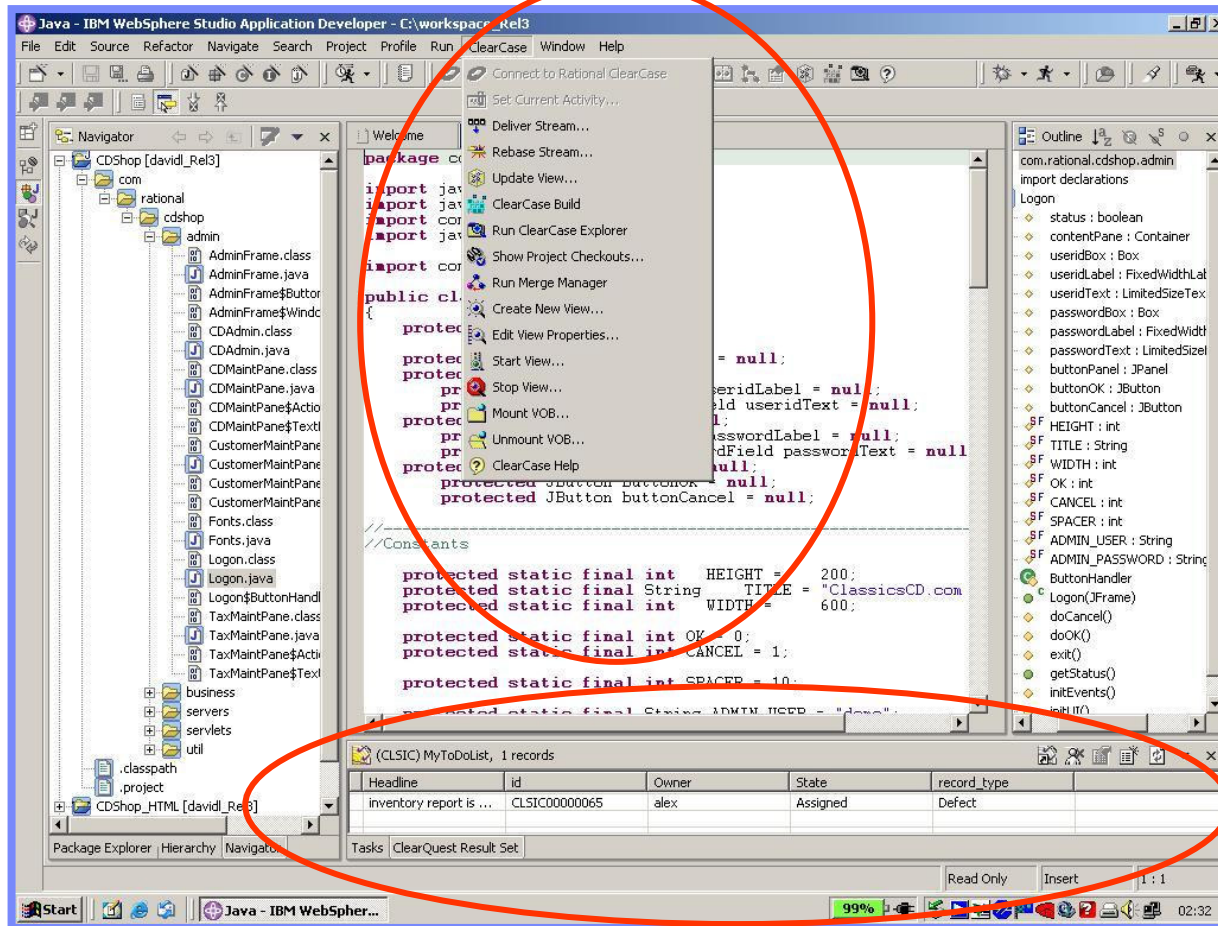
CLI



Browser



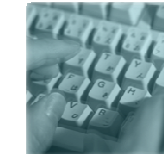
# ClearCase/ClearQuest and Eclipse (WSAD / RAD / RSA)



Eclipse



UNIX/Linux



ISPF



CLI

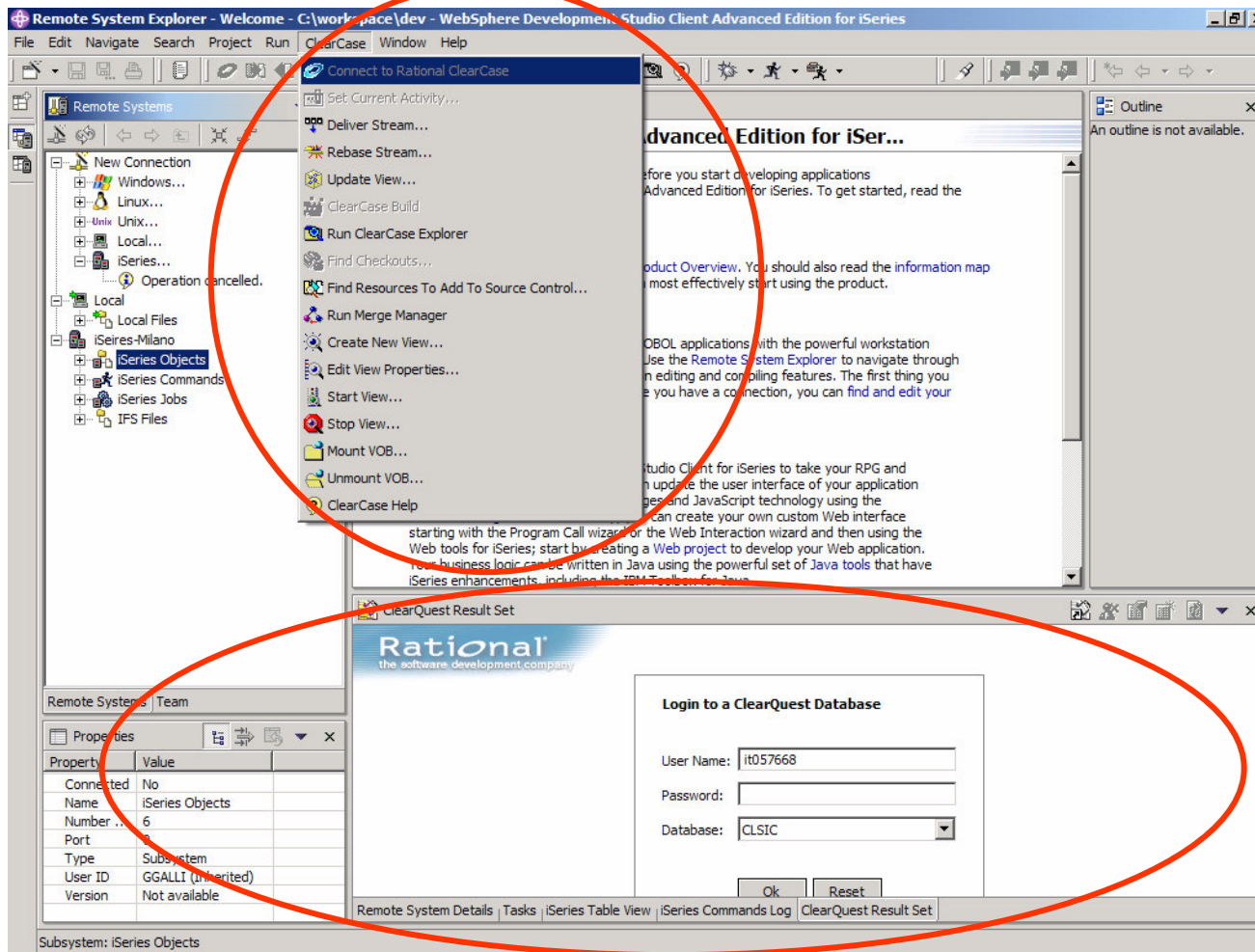


Browser





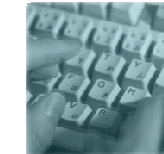
# ClearCase/ClearQuest and Eclipse (WDSC – AS/400)



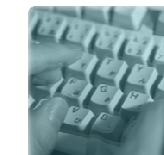
Eclipse



UNIX/Linux



ISPF



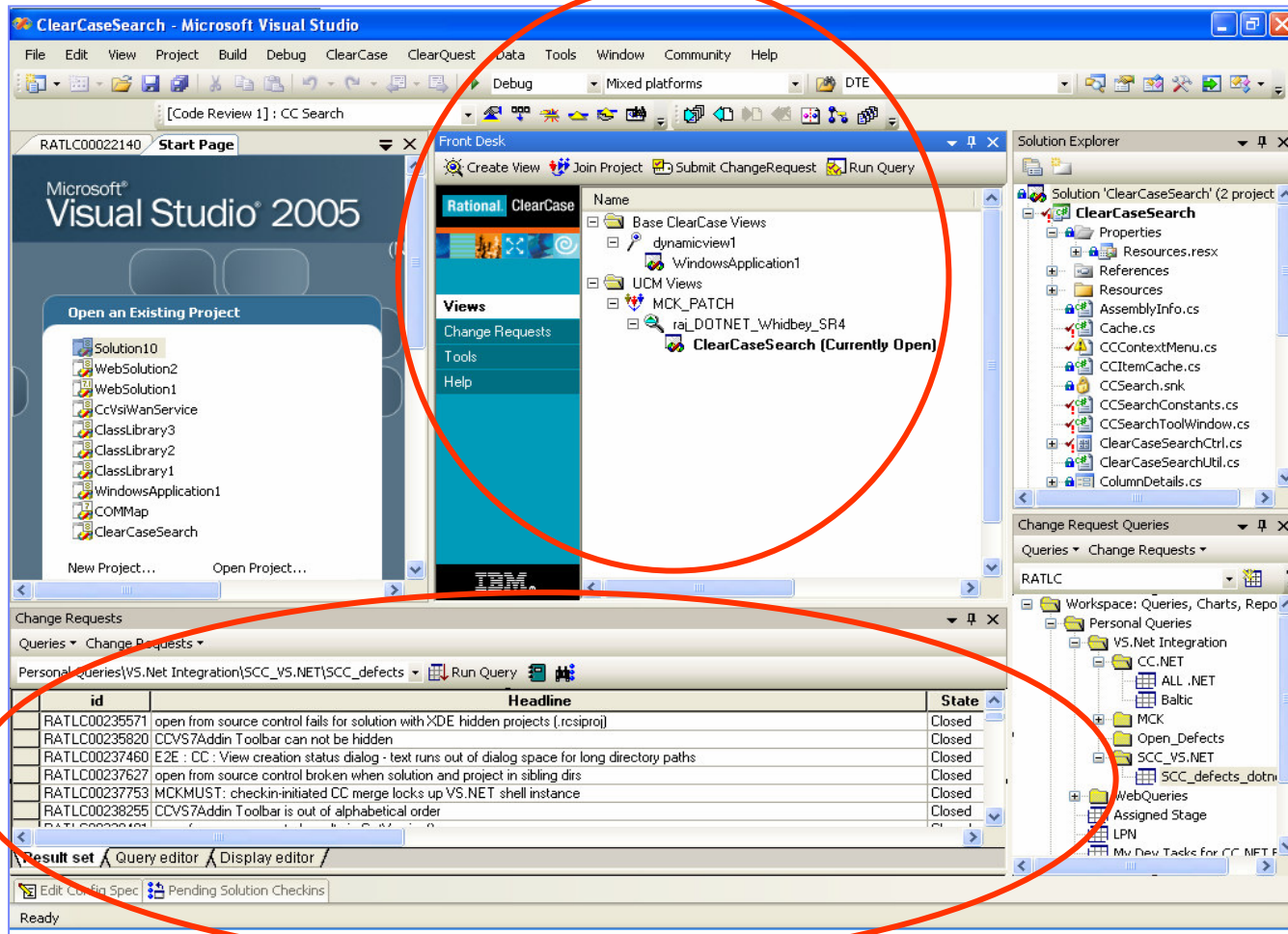
CLI



Browser



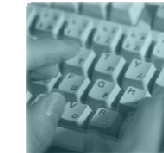
# ClearCase/ClearQuest and Microsoft IDEs (V\*)



Windows



UNIX/Linux



ISPF



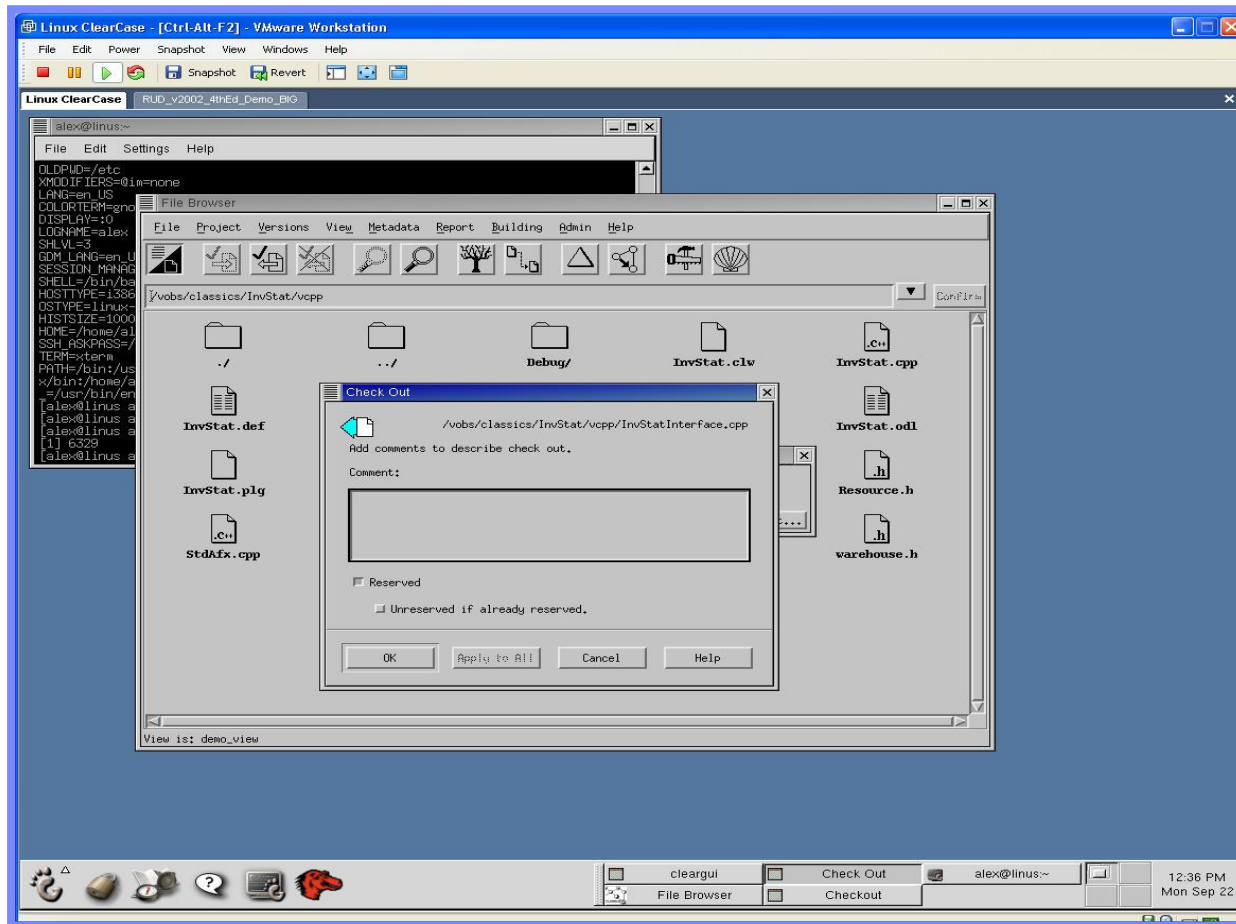
CLI



Browser



# ClearCase/ClearQuest and Linux



Eclipse



UNIX/Linux



ISPF



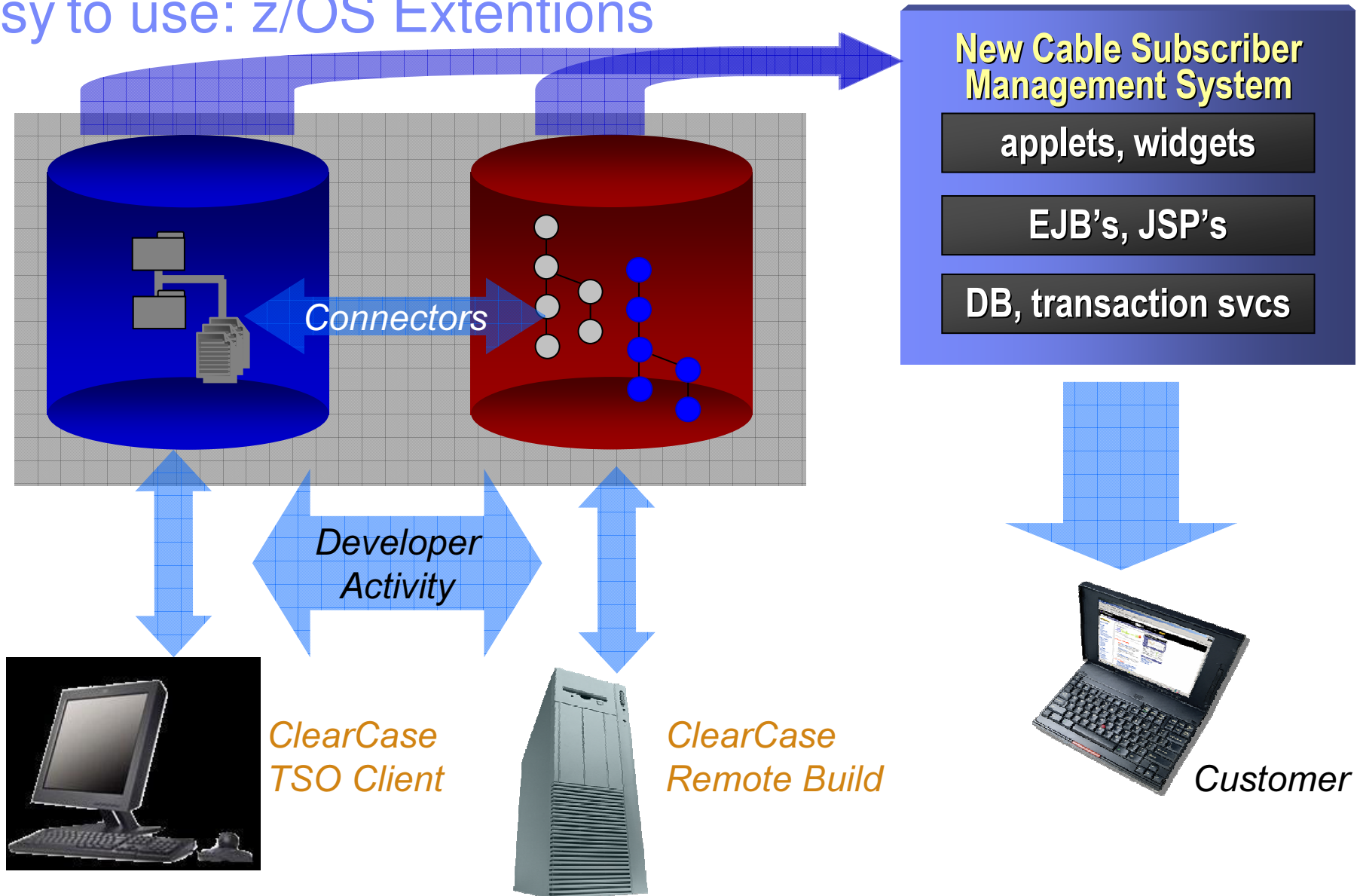
CLI



Browser



# Easy to use: z/OS Extensions



# ClearCase z/OS Extensions – TSO Client Feature

```
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  
  
M@ b 09/014
```



Eclipse



UNIX/Linux



ISPF



CLI



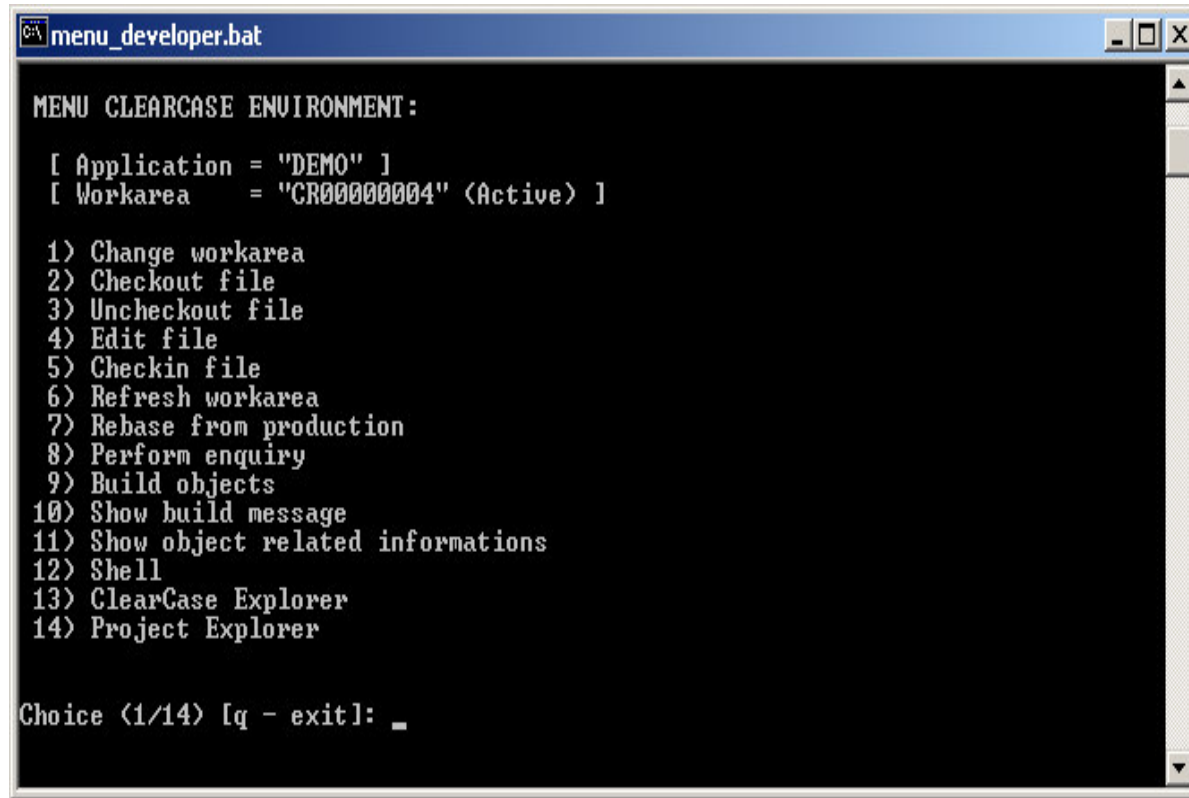
Browser



# ClearCase z/OS Extensions - Remote Build Feature



## ClearCase/ClearQuest and Command Line Interface



```
menu_developer.bat
MENU CLEARCASE ENVIRONMENT:
[ Application = "DEMO" ]
[ Workarea   = "CR00000004" (Active) ]

1) Change workarea
2) Checkout file
3) Uncheckout file
4) Edit file
5) Checkin file
6) Refresh workarea
7) Rebase from production
8) Perform enquiry
9) Build objects
10) Show build message
11) Show object related informations
12) Shell
13) ClearCase Explorer
14) Project Explorer

Choice (1/14) [q - exit]: _
```

(example of custom interface)



Eclipse



UNIX/Linux



ISPF



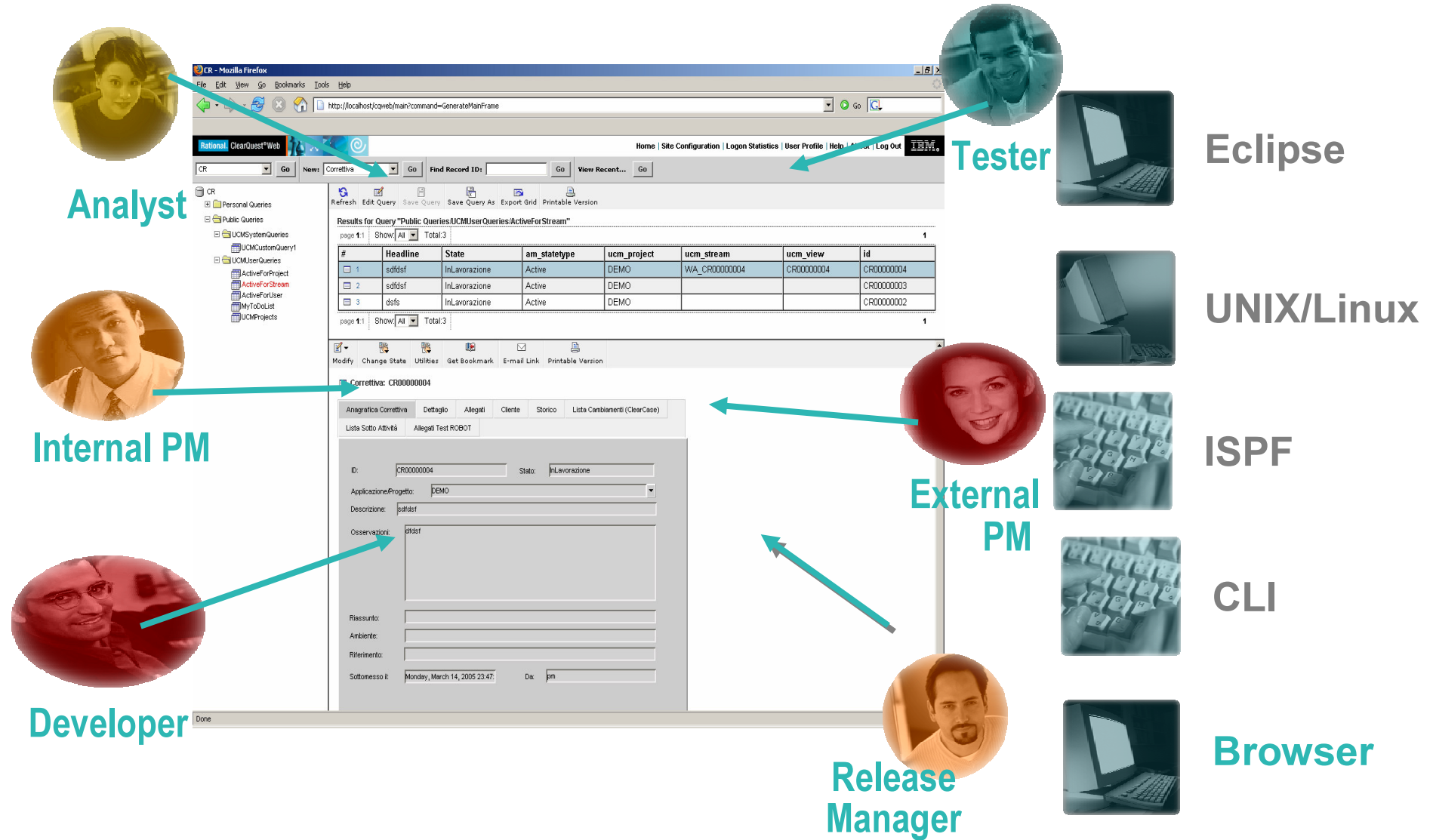
CLI



Browser



# ClearCase/ClearQuest and Web Interface





## Accelerate productivity with parallel development

### IBM Rational ClearCase and IBM ClearQuest...

- Make parallel development practical
  - Private workspaces, automatic coordination with rest of team
  - Work simultaneously on multiple releases
  - Coordinated workflows across the team
  - Secure, versioned software assets
- Help teams collaborate and communicate
- Benefit the entire development team
  - Developers
  - Analysts
  - Content contributors
  - Integrators
  - Testers
  - Project Managers



# Activity-Based Change Management

## Unified Change Management (UCM)

- UCM Activities automatically manage Change Sets
  - Activities in IBM Rational ClearQuest track IBM Rational ClearCase assets
  - Developers, Integrators, Testers work with Activities, not low-level assets

### IBM Rational ClearQuest

Manages activities

- ◆ To Do Lists
- ◆ Workflow

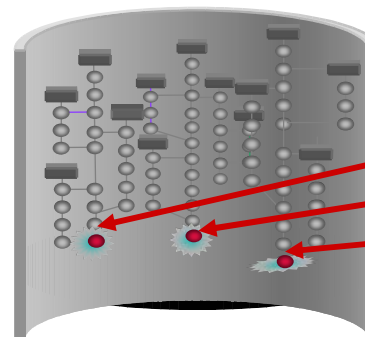
### IBM Rational ClearCase

Manages assets

- ◆ Versioning: code, models, XML, HTML
- ◆ Parallel development

### IBM Rational ClearQuest: Organized Activities

Request	Priority	Owner
Special Promo	1	Terry
Bug 527	2	Sandy
Add GUI button	2	Kim



### Change Set

Special Promo

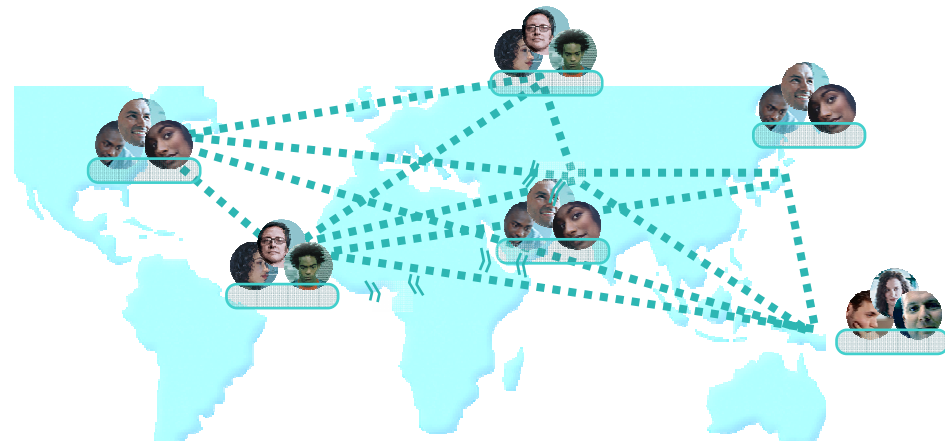
- a. html V5
- b. xml V3
- c. jpg V8



## Manage Distributed Development

Rational CCM solutions provides a variety of ways to manage and deploy your distributed deployment environments.

- ClearCase and ClearQuest Multisite
- ClearCase Web and ClearCase Remote Client (CCRC)
- ClearQuest Web



# Rational CCM Enterprise wide management tools

## Requirements, Change Sets, Build Information, Reporting all out of the box

The screenshot shows the IBM Rational ClearQuest application. At the top, there is a menu bar (File, Edit, View, Actions, Query, Window, Help) and a toolbar. Below the toolbar is a table of defects with columns: id, Headline, Owner, State, and Submit\_Date. The table contains three rows of defect data. Below the table, there are tabs for PQC, Test Data, and Environment. A 'View Requirement' dialog box is open, showing details for a requirement named 'Web Interfaces Compliance' associated with project 'ClassicsCD Web Shop' and tag 'FEAT13'. The dialog includes fields for Project, Tag, Name, and Text, along with buttons for OK, Cancel, Print Record, and Actions.

id	Headline	Owner	State	Submit_Date
CLISIC00000083	Link to "Home" does not go to ClassicsCD home page	jan	Opened	8/14/2002 8:34:01 AM
CLISIC00000084	Registered user cannot proceed to Cashier	jan	Submitted	8/15/2002 8:37:22 AM
CLISIC00000085	Wrong quantity of items placed in Shopping Cart	sandy	Assigned	8/17/2002 2:04:28 AM
CLISIC00000086	Fails poorly if browser doesn't allow cookies	jan	Closed	8/11/2002 4:07:28 AM

The screenshot shows the IBM Rational ClearQuest application. On the left is the 'ClearQuest Navigator' with a tree view of various reports and charts. The main area displays a table of 'All Defects - ClearQuest Query Results' with columns for id, State, and Headline. Below the table, there is a 3D bar chart titled 'Active Defects by Owner'. The chart shows the number of active defects for three owners: 'user', 'QE', and 'engineer'. The 'engineer' owner has the highest number of active defects.

id	State	Headline
CLISIC00000001	Opened	spelling error in login screen
CLISIC00000002	Resolved	sales tax incorrect if item deleted from purchase
CLISIC00000003	Resolved	cancel sale doesn't correctly reprint screen
CLISIC00000004	Resolved	want more help on inventory report
CLISIC00000005	Resolved	columns out of alignment
CLISIC00000006	Opened	delete item not working correctly
CLISIC00000007	Resolved	override price does not work
CLISIC00000008	Resolved	alt-C does not invoke cancel operation
CLISIC00000009	Resolved	http://www.ibm.com/nodeout.shtml

The screenshot shows the IBM Rational ClearCase application. On the left is the 'ClearCase Version Tree Browser' showing a tree structure of versions. The main area displays a table of files with columns: Name, Size, Kind, and Modified. A 'View Defect' dialog box is open, showing details for a defect named 'CLSIC00000070'. The dialog includes fields for UCM Project, Stream, View, and Change Set, along with buttons for OK, Cancel, Print Record, and Actions.

Name	Size	Kind	Modified
lost-found	0	Direct...	9/26/2006
foo.txt	20	File El...	10/9/2006

The screenshot shows the IBM Rational ClearCase application. On the left is the 'ClearCase Version Tree' showing a tree structure of versions. The main area displays a 'ClearCase Details' window showing a diagram of the version tree. The diagram shows a sequence of versions (0, 1, 2, 3) and their relationships, including a 'derive' relationship between versions 1 and 2.

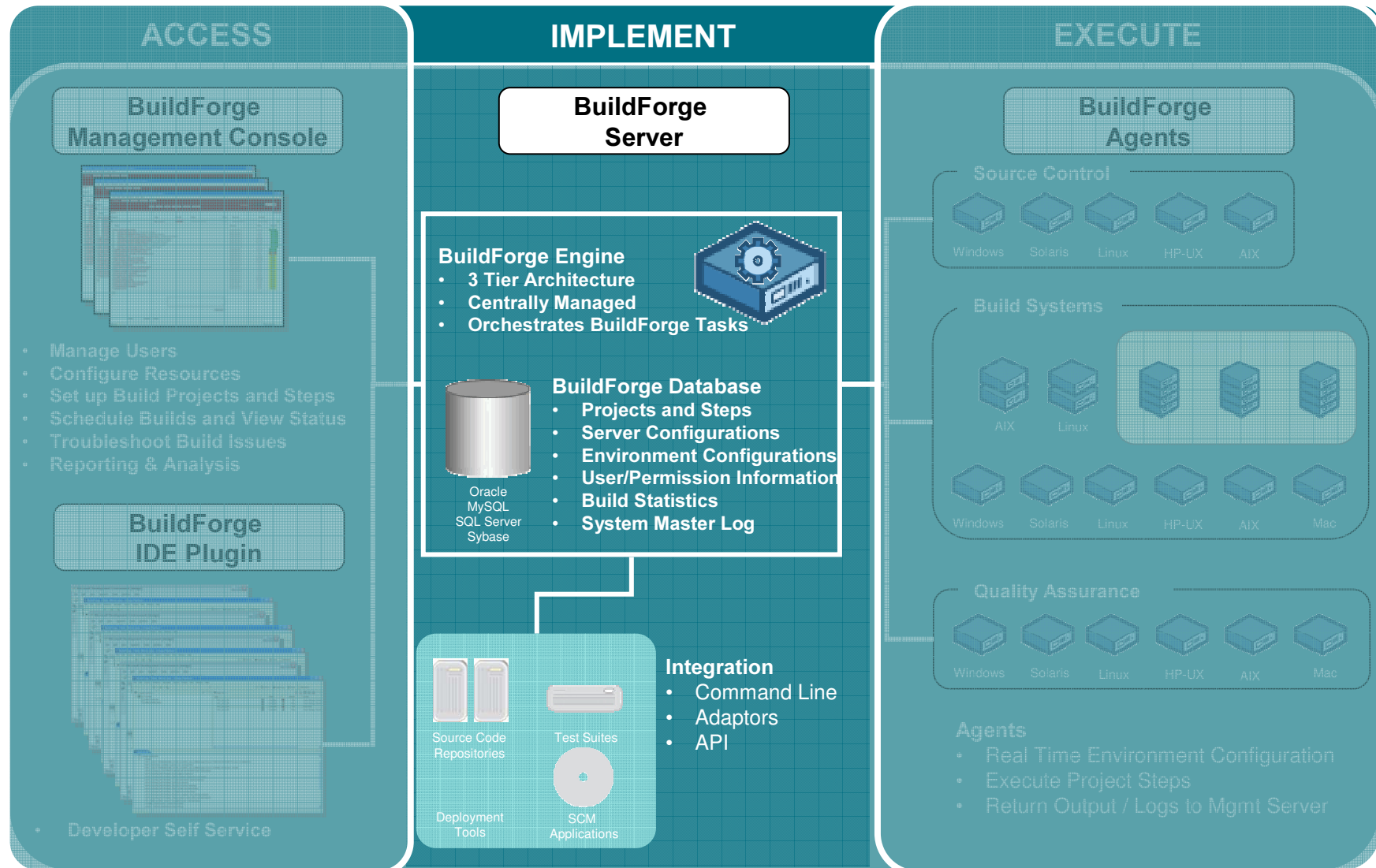


## IBM Rational ClearCase Integrations:

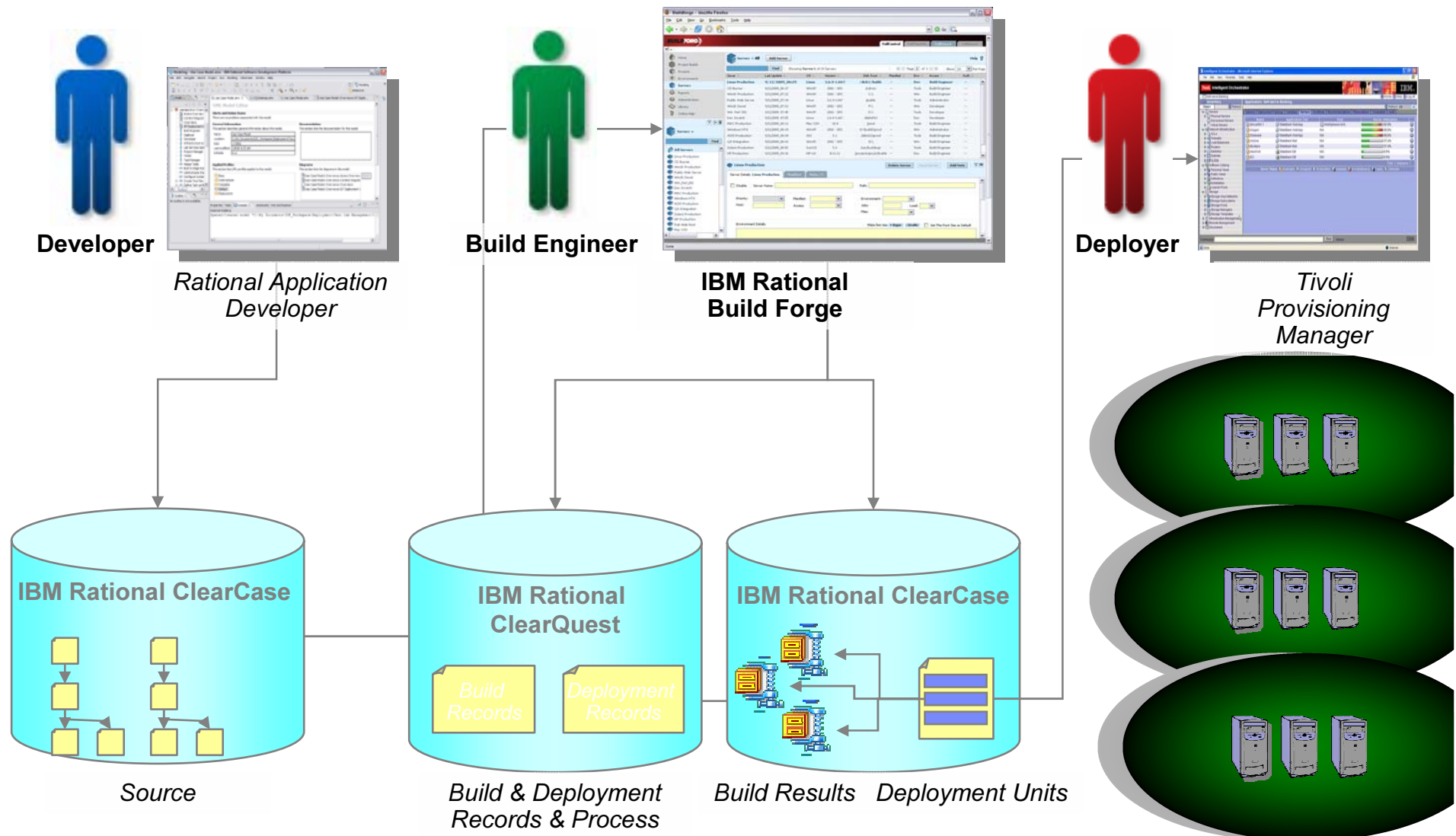
- Rational ClearQuest
- Rational BuildForge
- Rational Rose
- Rational Functional, Manual, Performance Testers
- RequisitePro
- Microsoft VS\*®
- Sybase PowerBuilder®
- Microsoft Word®



# IBM Rational BuildForge: Functional Architecture



# Build & deploy



## Terms



### Server

A Server is a logical resource in the BuildForge environment that represents any physical system running the BuildForge Agent.

A Server is defined within the BuildForge system with certain properties, such as a default path.

### Project

A set of Steps which can be run in the BuildForge system. A project can be run repeatedly; each run generates a new Build.

### Step

A component of a Project. A Step stores one or more commands which can be executed on a Server. Each command may itself launch an executable file, a batch file, or script that launches many other commands.

A command is anything that can be invoked from the command line of a given server (through an Agent) based on the security privileges configured.





## Environments

The BuildForge system allows you to manage environment variables separate from your Projects, Servers, and Steps

- Define Environment Groups
  - Containing one or more Environment Variables
- You can assign Environment Groups to Servers, Projects, and Steps
  - At runtime, the system assembles the final working environment for a Step from all of the relevant Groups assigned to the Server, Project, and Step



## Bill of Materials

- The BuildForge system generates a Bill of Materials (BOM) after each build
  - Build Results
  - Notes
  - Environments
  - List of build files
  - Code Changes
- The BOM contains information about the steps in the build and the changes to files that resulted from it.
- Uses of BOM:
  - Understanding the contents of a new build
  - Serve as an audit solution for your build and release process
  - Complete documentation of a build's contents.



## Filtering

- BuildForge, by default, determines the success or failure of any command by the exit status it returns
  - zero as success
  - one as failure
- Certain commands always return a zero exit status
  - For example, a command like *net use* prints a list of mapped network drives; the command always succeeds although the list may not contain desired drives
- Filters are a set of regular expressions that are used to parse the output of a step command to determine success or failure



## Terms



### **Dot (.) Commands**

The BuildForge system includes a set of special commands called Dot Commands that typically can be used as the body of a Step.

The system intercepts commands that are preceded by a period and uses them to perform special functions within the BuildForge system.



## Sampling of Commands Available

- .put
- .get
- .break
- .edit
- .load
- .lock
- .mkdir
- .monitor
- .include
- .strsub
- .retry
- .rget
- .rmdir
- .rput
- .scan
- .sleep



## Notification

- E-mail notifications are always sent to Groups, not individual users
- Projects level Notification
  - Start
  - Pass
  - Fail
- Steps level Notification
  - Pass
  - Fail



## Security

Concepts in the BuildForge security system:

- Users
- Access Groups
- Permissions
- Ownership



## Security (cont'd)

- Role-based system
  - *Access Groups* represents a role a *User* has in organization
  - Roles have *Permissions* and *Ownership*
- User access is determined by the **union** of the *Permissions* of all groups the user belongs to
- Combination of *Permissions* and *Ownership* define what a group can do and/or see
  - To edit a give Project, a User would have to have *Ownership* on the Project AND *Permissions* to Edit Projects
- Example
  - User who is a member of the Guest group (and no other groups) sees only Projects which have the Guest group assigned as their *Access* property
  - User can only launch projects with Guest access





## Ownership vs Permissions

### Permissions

- System level
  - Add/Delete Projects
- Multiple Access Groups can have a single permission
  - Build Engineer & System Administrator can have a permission to edit Access Groups

### Ownership

- Resource level
  - Projects, Steps, Environment Groups
- Only a Single Access Group can have ownership over a resource



## Overview of Adaptor Configuration

- Define source code *Interfaces* for the systems you use
- Create *Links* between those interfaces and specific projects
- *Schedule* the projects to run, as often as you want your system to check for changes
- *Run* each project once



## Chaining Projects

You can link or embed one Project within another Project in the BuildForge environment through a feature called **Chaining**.

### Two Types of Chaining

- Inline Chaining
- Pass / Fail Chaining



## Terms



### Library Projects

From the point of view of the BuildForge system, a library project is simply a projects whose Server property is set to *None*.

These projects are intended to be chained within other projects and therefore use the server of the Step that calls them.

### Server Pool

A Server Pool is a group of Servers with similar build properties (running the same operating system and with the same compilers available, for example).

The BuildForge system can run a Project on a different Server in the same Pool when the default Server for the Project is busy.

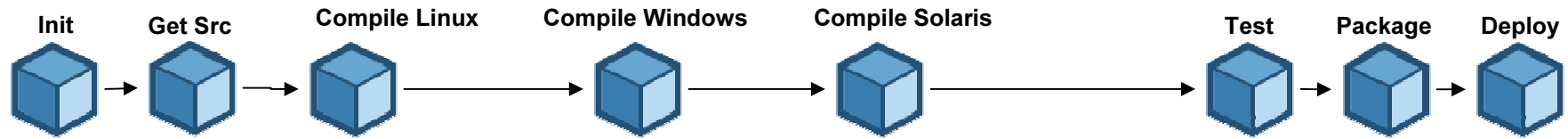
### Threading

Threading allows for multiple Steps to run in parallel, thereby increasing the speed of your build.

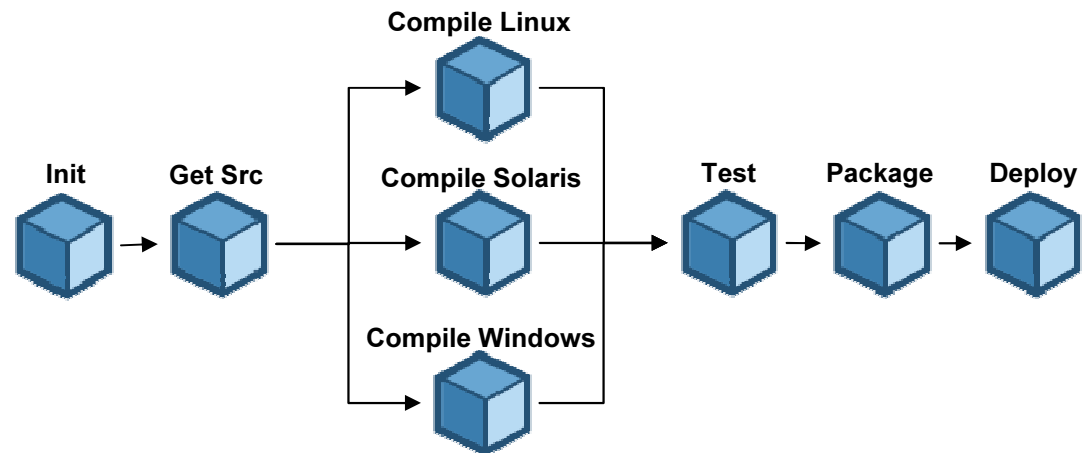


# Example

## Without Threading



## With Threading



## Benefits – Server pooling

- System can switch Build to alternate Server if default server is down or busy
- System will distribute processing load across farm of Servers automatically
- System can run a single Step on several Servers, in parallel, using **Broadcasting**



## Advanced Build Management Features

- Dynamic Server Management
  - Dynamic Pooling
    - Logically group hardware based on user criteria- department, similar characteristics, etc.
    - Used for load balancing, fault tolerance, easier administration
  - Dynamic Server Allocation
    - Server inventory contains manifests- properties of a server used for dynamic selection
    - Optimally select build servers at execution time
      - Flexible selection criteria: can be general or detailed
    - Attribute types include: built-ins, static, dynamic
    - Support for user-defined types
    - Server manifest refresh on-demand or scheduled
    - Add new servers on network without having to manually configure or update any projects



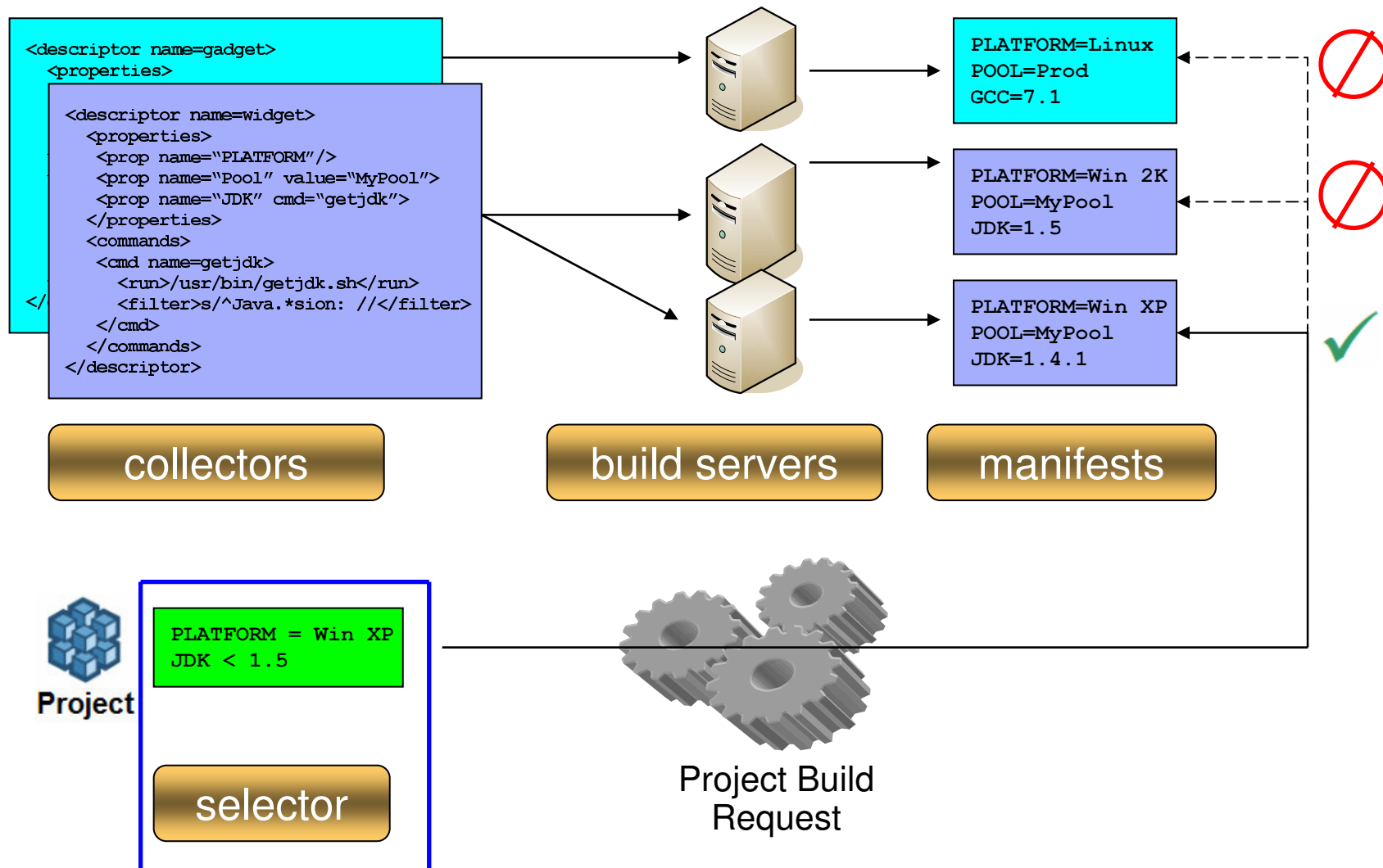
## Advanced Build Management Features (cont'd)

- Load Balancing
  - Control maximum number of jobs per server, per user
  - Utilize idle hardware for requests
- These advanced build management features introduced in 7.0 are available via Enterprise Edition only





# Workflow Overview of Collectors / Selectors / Manifests





## Extreme Change Management

- Common, distributed and synchronized repository
- Common process and UI for heterogeneous platforms (integrated with IDEs)
  - MVS
  - Windows (VAJ – WSAD/RAD – WD/z – Eclipse – MS V\* - WDSC)
  - Unix / Linux
- Ability to work on parallel Change Requests
  - Enhancement, Maintenance, Bugfixing
- Ability to rebase all parallel developments with production environment
- Ability to work on the same task/activity/request with multiple users
- Increase the level of automation for the SCM process
- High value functionalities for critical applications:
  - Impact analysis
  - Out-of-date mechanism for builds -> software consistency



## Agenda

- Defining Change and Configuration management
- Business value of Change and Configuration management solutions
- Challenges-solutions in Change and Configuration management
- Capabilities of Rational Change and Configuration management tools
- BuildForge Demo
- ECM Demo
- Closing, Q&A and Thanks



## For More Information...

- **IBM Rational Clearcase:**

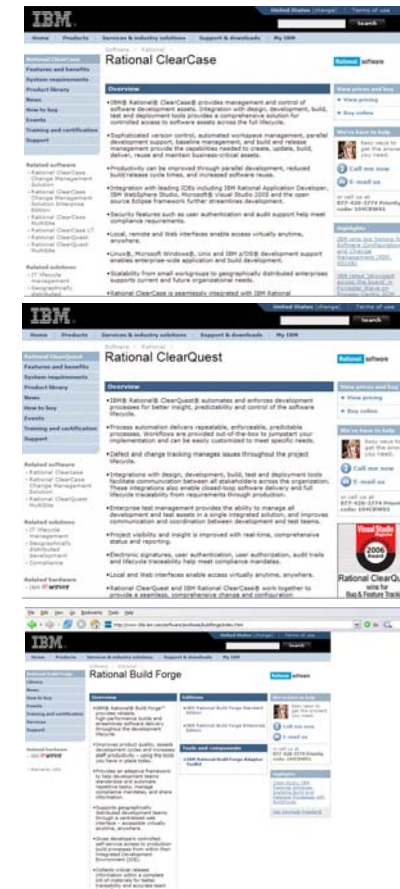
<http://www-306.ibm.com/software/awdtools/clearcase/>

- **IBM Rational ClearQuest:**

<http://www-306.ibm.com/software/awdtools/clearquest/>

- **IBM Rational BuildForge:**

<http://www-306.ibm.com/software/awdtools/buildforge/index.html>



## Additional resources

- Eclipse
  - <http://www.eclipse.org>
- IBM developerWorks for hints, tools, tips and tricks
  - <http://www-106.ibm.com/developerworks/platform/>
- Demos, evaluations, upgrades, patches, hot fixes, add-ins, plug-ins, models, scripts, and documentation
  - <http://www-136.ibm.com/developerworks/rational/downloads/>
- IBM Rational Application Developer
  - <http://www-306.ibm.com/software/awdtools/developer/application/index.html>





# IBM Software Development Platform

*Paolo Cravino*    [paolo\\_cravino@it.ibm.com](mailto:paolo_cravino@it.ibm.com)

*Oreste Egidio*    [oreste\\_egidio@it.ibm.com](mailto:oreste_egidio@it.ibm.com)

