



GoMidjets CompBL

Easily Compare ClearCase UCM Baselines and Streams

Copyright © 2008-2009 GoMidjets. All rights reserved.
<http://www.gomidjets.com>

Administrator's Guide – Windows Version

Version 2.0

GoMidjets CompBL is patent-pending software.

Table of contents

| | |
|---|----|
| Conventions Used in This Book | 3 |
| How to Contact us | 3 |
| Summary of changes | 4 |
| GoMidjets CompBL Concepts..... | 5 |
| Key Features | 5 |
| GoMidjets CompBL installation..... | 6 |
| Pre installation..... | 6 |
| Installation..... | 6 |
| How to use GoMidjets CompBL | 6 |
| Save Report as Autoamtic Rule | 8 |
| How to use GoMidjets CompBL in command-line mode..... | 8 |
| For instance, assuming CompBL has a rule named "MainBanking" and you want to run it from command-line, please do the following:..... | 8 |
| Appendix A: ClearCase Terms | 9 |
| About components, baselines, activities, and streams..... | 9 |
| Components | 9 |
| Baselines | 9 |
| Activities..... | 10 |
| Streams | 10 |
| About recommended baselines in UCM..... | 10 |

Conventions Used in This Book

Throughout this book, we've used the following typographic conventions:

Constant width

Constant width in body text indicates a language construct, such as the name of a stored procedure, a SQL statement, an enumeration, an intrinsic or user-defined constant, a structure (i.e., a user-defined type), or an expression (like `dblElapTime = Timer - dblStartTime`). Code fragments and code examples appear exclusively in constant-width text. In syntax statements and prototypes, text set in constant width indicates such language elements as the function or procedure name and any invariable elements required by the syntax.

Constant width italic

Constant width italic in body text indicates parameter names. In syntax statements or prototypes, constant width italic indicates replaceable parameters. In addition, constant width italic is used in body text to denote variables.

Italicized words in the text indicate intrinsic or user-defined function and procedure names.

Many system *Italic*

elements, such as paths and filenames, are also italicized. URLs and email addresses are italicized. Finally, italics are used for new terms where they are defined.



This icon indicates a tip, suggestion, or general note.



This icon indicates a warning or caution.

When “UNIX” is mentioned in this book, it includes all UNIX operating systems and all Linux operating systems that ClearCase can run on.

How to Contact us

GoMidjets Inc.

68 Kaplansky St.

Petach-Tiqwa, Israel 49213

Tel: (972)-722-405-222

Facsimile: (972)-722-408-222

Email: info@gomidjets.com

Summary of changes

This edition (V2.0) includes the following changes:

- Locating differences between two streams
- Locating differences between a stream and a baseline
- Locating differences between two baselines
- Compare any combination by using composite baselines
- Compare any combination by using recommended baselines
- Reporting to Email
- Reporting to Excel
- Reporting to text files
- Open version tree browser from the report
- Saving rules
- Dynamic reports
- Automatic reports
- Enhanced support to "composite Baselines"
- Supports Windows server 2008, Windows Vista and Windows 7

GoMidjets CompBL Concepts

GoMidjets CompBL is a simple, fast and reliable software for automatic validation of the integrity of ClearCase infrastructures, on a daily basis.

GoMidjets CompBL's main purpose is to identify problems of ClearCase infrastructures (VOBs, views, global type, etc.) as early as possible, and ensures peace-of-mind.

GoMidjets CompBL works as follows:

1. The CM administrator defines the ClearCase VOBs server host, the type of checks, which VOBs to check and the frequency of checks.
2. CompBL executes the checks and recognizes problems, if they exist. This execution includes dbcheck, checkvob and rgy_check commands, and free space validation checks as well.
3. CompBL collects any problems found and sends them as a compact mail to the CM administrator. It may suggest solutions and provide links to the IBM knowledge-base

Key Features

- **Comparisons**
 - Compare differences between two baselines
 - Compare differences between two streams
 - Compare differences between a stream and a baseline
 - Compare any combination by using composite baselines
 - Compare any combination by using recommended baselines
- **Reports**
 - Filter out unnecessary data from reports
 - Drill down to include activities, components and file versions
 - Reporting to screen, including 'find' feature and well-known 'version tree' browser
 - Reporting to Email
 - Reporting to Excel
 - Reporting to text files
 - Open version tree browser from the report
- **Interface**
 - Enables easy installation - Start working in 1 minute
 - Set you own comparison rules for daily/nightly reports (static rules)
 - Set dynamic rules
 - Filter our unnecessary report inputs]
 - Supports any version of ClearCase
 - GUI interfaces
 - CLI interface provides a complete, automated process
 - Supports the following ClearCase versions, beginning with version 2003.06: ClearCase UCM, ClearCase LT, ClearCase MultiSite (each replica separately)
 - Supports the following operating systems: UNIX, Linux and Windows

GoMidjets CompBL installation

Pre installation

Prerequisites:

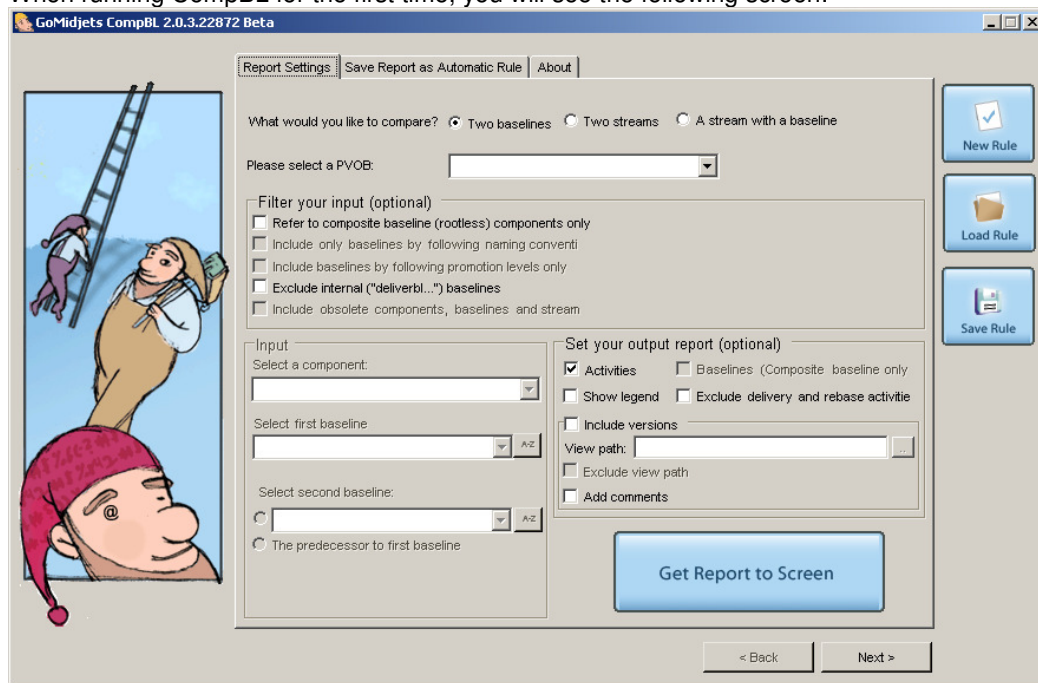
1. CompBL can be installed on any of the following Windows hosts: Windows 2000, Windows XP, Windows 2003 server, Windows Vista, Windows 2008 and Windows 7.
2. Administrative (local admin) privileges are required for Windows Server.

Installation

Just run the exe file.

How to use GoMidjets CompBL

When running CompBL for the first time, you will see the following screen:



The screen is consisted of tabs and buttons.

The tabs section is divided to the following tabs, as described below: "Report settings" and "Save Report as Automatic Rule".

You firstly have to choose what comparison you would like to run: A baseline with a baseline, a stream with another stream or a stream with a baseline. Then, you have to select your PVOB. A PVOB is considered as the name of your database. After selecting a PVOB, You have the option to filter our your input (component, streams and baselines), what can eliminate their lists and make your choice much easier.

These are the parameters you can filter out:

- **Refer to composite baselines (rootless) components only.** CompBL will not display non-composite baselines components in the components list.
- **Excluding internal baselines.** Rational ClearCase generates internal baselines on its on-going jobs. Those baselines are starting with "deliverbl" name. If you choose this option, CompBL will not show these baselines in the corresponding baseline lists.

Depend on your comparison choice, you have to select the input:

- **A Component.**
- **Streams.** CompBL suggests you to view your projects tree (very similar to the Project Explorer") and select your streams from there. You can use the right-click mouse to select streams, or by drag-and-drop.
When selecting a stream, you can select integration stream or development stream. CompBL automatically recognizes if that stream has recommended baselines, and it suggests you to use it accordingly.
- **Baselines.** Baselines are displayed after you select your component. CompBL offers you to refer to predecessor baseline (if exist) instead of selecting the second baseline. You can sort baselines by time or by alphabetical.

After that, you have the option to set your output report:

- **Activities.** Displays differences in terms of activities and Include activities in the report.
- **Baselines.** For composite baselines, displays differences in terms of member baselines. When used with the –activities option, displays each member baseline followed by activities in the member baseline.
- **Show legend.** Add to report a legend that explains the meaning of difference arrows (">>", "<<", "->" and "<-") as described in appendix A below.
- **Exclude "Delivery" and "Rebase" activities.** This feature enables you to filter out the ClearCase internal activities which starts with "Delivery" and "Rebase" names.
- **Include Versions:** Displays differences in terms of versions. If you want to enhance the report so it includes the filenames and versions which are part of the activities – check this option. If you check it, you may have to specify the following:
 - **View Path.** You can specify the view path by clicking on the "..." button and then you will get a "browse for folder" wizard.
 - **Exclude View Path:** Check this option if you want to exclude the view path string in the report.
 - **Add comments.** Include change-set comments into your report, beside the version other details.

That's it. Now you can press "Get Report to Screen" and immediately get your report.

When you read the report, you can find for a string and use the right-click menu to select all text and then copy & paste it to another application.

If you choose to include versions in your report, you can right-click on a version and see its ClearCase version-tree browser.

CompBL enables you to take your report one step ahead and save it as a rule, what means that you can save it and run it on an automatic basis, daily, nightly or hourly, depends on your needs.

Save Report as Autoamtic Rule

The screenshot shows the 'Save Report as Automatic Rule' dialog box in the GoMidjets CompBL 2.0.3.22872 Beta application. The window has a title bar with the application name and standard Windows window controls. On the left side, there is a cartoon illustration of a man in a red hat and a woman climbing a ladder. The main area contains the following fields and controls:

- Report Settings** tab selected.
- Please enter rule name:** A text input field.
- Set as active rule**
- Please select at least one of the following reports:**
 - Email**: Includes fields for **Recieptents** and **Copy (cc)**.
 - Text File**: Includes a **Filename** field.
 - Excel**: Includes a **Filename** field.
- Overwrite if already exist** (two instances, one for Text File and one for Excel).
- Test Rule** button with a filmstrip icon.
- Schedule Rule on a Daily Basis** section:
 - Text: "You can schedule this rule to run automatically on a daily basis. Press here to set it for this rule..."
 - Schedule Rule** button with a clock icon.
 - Schedule All Rules** button with a clock and multiple arrows icon.
- General Email Properties** section:
 - Outgoing Mail Server (SMTP):** **Send a Test Email >>** button.
 - Advanced Email Settings** button.
 - Save Email Settings** button.
 - Text: "Use advanced settings if your mail server requires authentication".
- Navigation buttons: **< Back** and **Next >**.
- On the right side, there are three buttons: **New Rule** (checkmark icon), **Load Rule** (folder icon), and **Save Rule** (floppy disk icon).

Rule Name. Please enter a rule name. This is a unique name

Active Rule – Mark this checkbox if you plan to run all active rules by a scheduler. CompBL enables you to run on active rules at a time.

Email – This feature enables you to send the report to predefined recipients (seperated by a comma). You can enter also recieptents who get the email as a copy (cc).

TIP: If you select this feature, you have to set the email properties. See below.

Text File – Save the report to a text file. Please sign its file name. By defaultl, CompBL creates a new file if it's not exist yet. If it's already exist, it append the report to the end of the file. If you mark the "overwrite if already exist" checkbox, it will overwrite the file and creates a new file.

Excel – Save the report to an Excel file (csv format), each line into a row. Please sign its file name. By defaultl, CompBL creates a new file if it's not exist yet. If it's already exist, it append the report to the end of the file. If you mark the "overwrite if already exist" checkbox, it will overwrite the file and creates a new file.

Test Rule – This feature enables you to test your rule before running it on a regular basis. Same results.

How to use GoMidjets CompBL in command-line mode

```
CompBL.exe -exec -rule=rulename[,rulename,[rulename]]...
```

For instance, assuming CompBL has a rule named "MainBanking" and you want to run it from command-line, please do the following:

```
"C:\program files\GoMidjets\CompBL.exe" -exec -rule=MainBanking
```

Appendix A: ClearCase Terms

The following symbols are used to describe the results of a comparison:

| Symbol | Meaning |
|--------|---|
| << | The item is present only in <i>baseline1</i> or <i>stream1</i> . |
| >> | The item is present only in <i>baseline2</i> or <i>stream2</i> . |
| <- | The activity is present in both of the items being compared and is newer in <i>baseline-selector1</i> or <i>stream1</i> . |
| -> | The activity is present in both of the items being compared and is newer in <i>baseline2</i> or <i>stream2</i> . |
| none | There are no differences between the objects being compared. |

About components, baselines, activities, and streams

ClearCase UCM uses baselines, activities, and streams to control modifications to a project's components.

Components

UCM users rely on a project manager to create a project, structure the project's shared resources (files and directories) into components, and then specify baselines, activities, and streams that guide the team in modifying and integrating the components.

A UCM component is a collection of shared resources that the team develops, integrates, and releases as a unit (a class library, for example, or a user interface module). A UCM project includes one or (usually) more components, which are often shared with other projects. A component is usually created as a top-level directory in a VOB. A VOB can include one or more components, but a component cannot contain other components.

Note: You cannot use the ClearCase Remote Client to create, modify, or view an existing component. To work with a component, you must use tools that are available only on the native Rational ClearCase client. For more information about components, see the ClearCase documentation.

Baselines

A baseline specifies one version of each element in a component and represents a version of the entire component at a particular stage of its development. As work on a component progresses, the project manager periodically creates new baselines and gives each one a promotion level that indicates its maturity or readiness for use by the team. When you rebase a UCM ClearCase view, you update its configuration to select the versions selected by a particular baseline (usually the project's recommended baseline).

You can use the ClearCase Metadata Explorer Navigator to create and recommend baselines in a UCM project.

Activities

In a ClearCase UCM view, any time you add a resource to source control or modify a resource that is already under source control, you must associate your action with a UCM activity, which identifies a set of versions created during a specific development task. When a UCM project is enabled for Rational ClearQuest®, activities are created and maintained as records in a Rational ClearQuest database. When a UCM project is not enabled for Rational ClearQuest, activities are created and maintained as metadata in a ClearCase Project VOB.

Each activity includes a headline, an activity ID, and a change set. The headline is a text string, the activity ID is a unique identifier generated by ClearQuest or ClearCase, and the change set specifies one version of each file that you modified (or added to source control) while working on the activity.

When you create a new activity or select an existing one to associate with an operation, that activity becomes the current activity for the ClearCase view in which you are working. Each ClearCase UCM view can have no more than one current activity.

An activity is not limited to a particular scope of work, although UCM project managers usually establish guidelines that help users assign work to an activity. For example, a project manager may decide that the versions that you create while fixing a defect or adding a new feature constitute an activity. An activity could also encompass all of the changes required to port an application to a new operating system or hardware platform.

You can use the ClearCase Metadata Explorer Navigator to view activities in a UCM view.

Streams

Every UCM ClearCase view is bound to a stream, which includes a set of activities and baselines and determines which versions of elements are in the view. Every project has an integration stream, to which all project members deliver their work. In addition, a project can define one or more development streams that allow individuals to work in greater isolation before delivering their changes to the integration stream. When you join a single-stream UCM project, you create an integration view, which is bound to the project's integration stream. When you join a multiple-stream UCM project, you create an integration view and a development view. The development view is bound to a development stream.

In a single-stream project, all team members work on activities in an integration view and complete them by checking them in to the integration stream. As soon as a change is checked in, any developer can see it by updating his integration view. In a multiple-stream project, individuals work on activities in a development view and periodically deliver them to the project's integration stream. Before the results of these activities can be seen by other team members, the activities must be included in a baseline and the team members must rebase their development streams to that baseline. With a multistream UCM project, allows developers more control over their development environment.

About recommended baselines in UCM

Baselines are used to ensure that developers stay in sync with each other's work in a UCM project. A new baseline includes the work developers have delivered to the integration stream since the last baseline. It represents a snapshot of the changes made to a particular component in the context of a particular stream: it is a version of a component.

Note: Baselines should be thoroughly tested before they are approved to be recommended.

A recommended baseline is a set of versions that have been identified as stable. This baseline establishes the current starting point for the project, and when you rebase or join a

project this is the baseline that is used as the foundation for your development (or child) stream; unless you specify a different baseline to be used.

Note: An integration stream's default recommended baselines are its foundation baselines. A development stream has no default recommended baselines.