



Tech Notes

---

# Faster Application Development via Improved Database Change Management

Integrating Database Change Management with Software Development  
to Reduce Errors, Re-Work, and Testing Efforts

Embarcadero Technologies

November 2008

---

**Corporate Headquarters**  
100 California Street, 12th Floor  
San Francisco, California 94111

**EMEA Headquarters**  
York House  
18 York Road  
Maidenhead, Berkshire  
SL6 1SF, United Kingdom

**Asia-Pacific Headquarters**  
L7. 313 La Trobe Street  
Melbourne VIC 3000  
Australia

For a software development team, database changes and updates come from many different directions. A software developer may add a feature that requires a new column in a table. Another developer adds data to a code table. Testers add indexes to optimize new scenarios. Database administrators fix performance problems in production that need to be pushed back to development.

From design to development and from test to production and then back again, tracking database changes across the software development life cycle is a challenge that is not easily met by typical software development tools.

Database change management has a number of unique requirements that can only be addressed by specialized tools: preserving data when making structural changes, validating synchronicity in a replicated environment, managing reference data across test, development, and production environments, maintaining security differences, and much more.

A tool designed to handle database change management tasks can simplify processes, streamline complex tasks, reduce the chance for errors, and minimize downtime. You want to ensure that changes applied to production make it back into the database design, development, and test environments; you want to easily bundle changes together for migration between environments; you want to automate manual and repetitive tasks; and you want to keep track of tables, procedures, settings, and privileges.

Adopting reliable, repeatable, and efficient processes to manage database changes are vital to the success of any software project. Embarcadero® Change Manager™ puts the power of change tracking, schema comparison, software-generated synchronization scripts, and flexible reporting into the hands of the development team.

Incorporating database change management within the software development life cycle saves time, money, and resources. Particularly, Change Manager assists with the following tasks:

- Capture, Restore, and Report on Object Changes
- Ensure Accurate Test Environments
- Report and Recreate Projects

## CAPTURE, RESTORE, AND REPORT ON OBJECT CHANGES

It is important for each member of the software development team to know what is going on with the database during the development lifecycle. Developers need to ensure their development environments are synchronized with the latest code base and QA members need to be able to set up previous environments to test bug fixes.

Change Manager lets developers create schema archives to capture a baseline or “snapshot” of relevant objects. They can then report on the changes or create a schema compare job to identify the differences between their baseline and the new schema. With the help of Change Manager, teams can keep an accurate history of all database work to track progress or revert back to previous states.

With short release cycles and iterative development schedules, the database environment may change every day. It is an ongoing challenge for development and QA teams to ensure their software code stay synchronized with the latest database changes.

Using the Change Manager archive feature, you can capture snapshots of a master database and its DDL and use comparison jobs and email notifications to let development, documentation, quality assurance, and other groups know when the development database changes. The command-line interface allows you to invoke these jobs from a scheduler to run as frequently as necessary or use a generated ANT script to integrate with your continuous build process.

## USE SOURCE CONTROL

Source control and versioning are an invaluable part of any development process. Change Manager has flexible integration with a wide variety of source control software so that database baselines can be easily accessible and tracked within the development process.

With Change Manager, you can integrate with almost any source control product using the Eclipse Team API. Store archives, synchronization scripts, jobs, reports, and even data sources in the same source repository as your code base.

## SYNCHRONIZE DEVELOPMENT ENVIRONMENTS

Within the software development life cycle, teams use a number of different database environments for different stages of development. Changes within development databases, test databases, and staging databases all need to be tracked and pushed to other environments when necessary. In addition, performance improvements and configuration solutions found in production environments should make it back to development so that those improvements can be implemented in the next release.

With Change Manager, developers can ensure their development and test databases are up-to-date. You can capture changed objects and then generate alter scripts that automatically account for dependent objects. This saves time and reduces the possibility for errors because you don't have to figure out the correct syntax for carrying out changes. This is especially valuable in cases where a change requires a table to be dropped and recreated. Change Manager will ensure that the script also drops and recreates dependent objects such as indexes, primary and foreign key constraints, and more. It will even preserve data, renaming your original table, and moving data from it to the new version of the table.

## CREATE AUDIT TRAILS

By using Change Manager, developers can create an audit trail of their work, allowing them to easily communicate changes to the team's DBA or data modeler, who can then review and sign off on the developer's database changes.

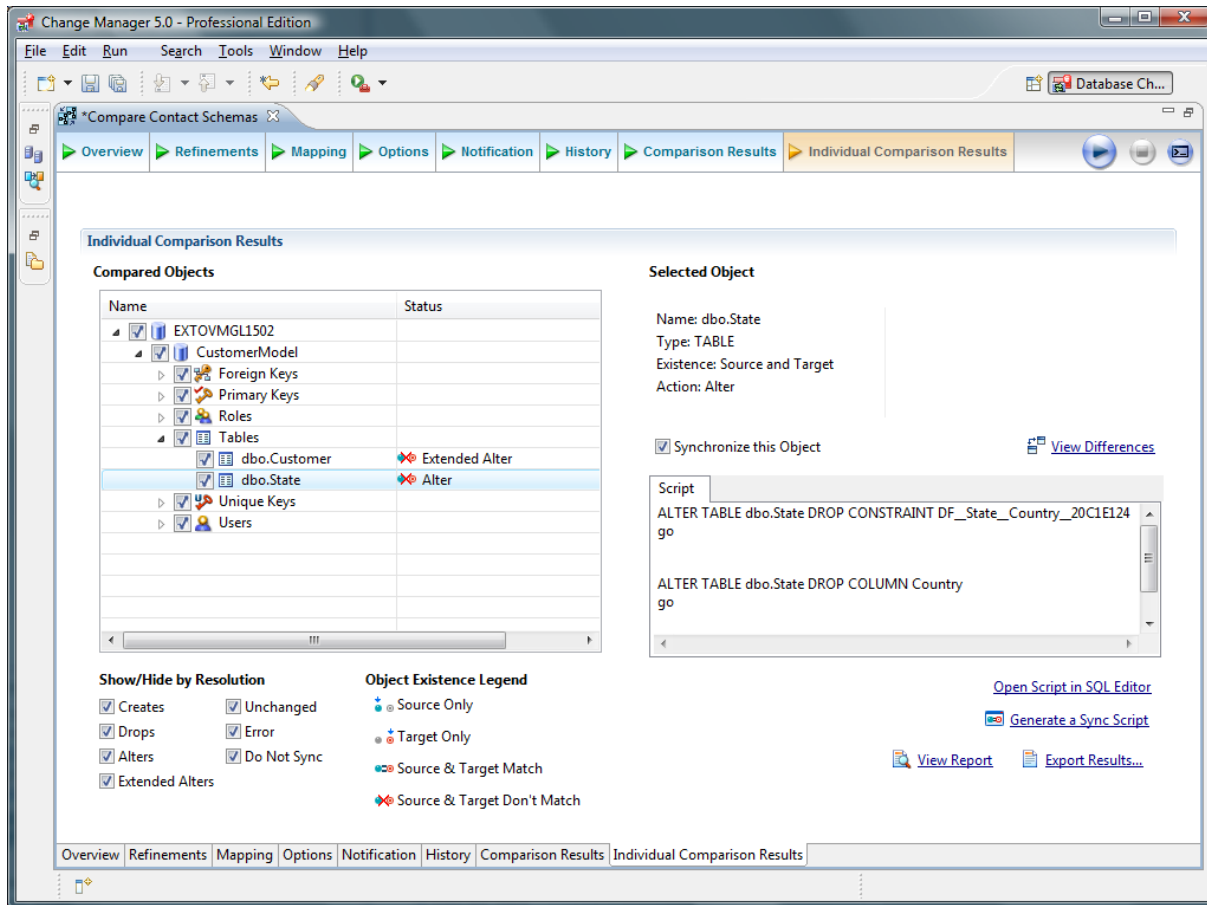


Figure 1 - Schema Results: Browse schema deltas, preview change DDL, and select objects to script

## ENSURE ACCURATE TEST ENVIRONMENTS

However good one's quality assurance plan may be, testing outputs are only reliable and accurate to the degree that the environments are reliable. It is time-consuming to build a test environment, and even after a great deal of effort, you may still find that certain changes were not properly migrated from development or production. At best, it delays other testing priorities, and at worst, this requires costly re-execution cycles.

In the case of code tables and reference data, having consistency between environments is a crucial aspect of maintaining accurate test environments. Keeping these types of data consistent is made simple with the Change Manager Data Compare feature, which can even compare data between database platforms.

For environments with multiple DBMS platforms, Change Manager enables team members to manage any type of database environment from within the same tool and with one user interface.

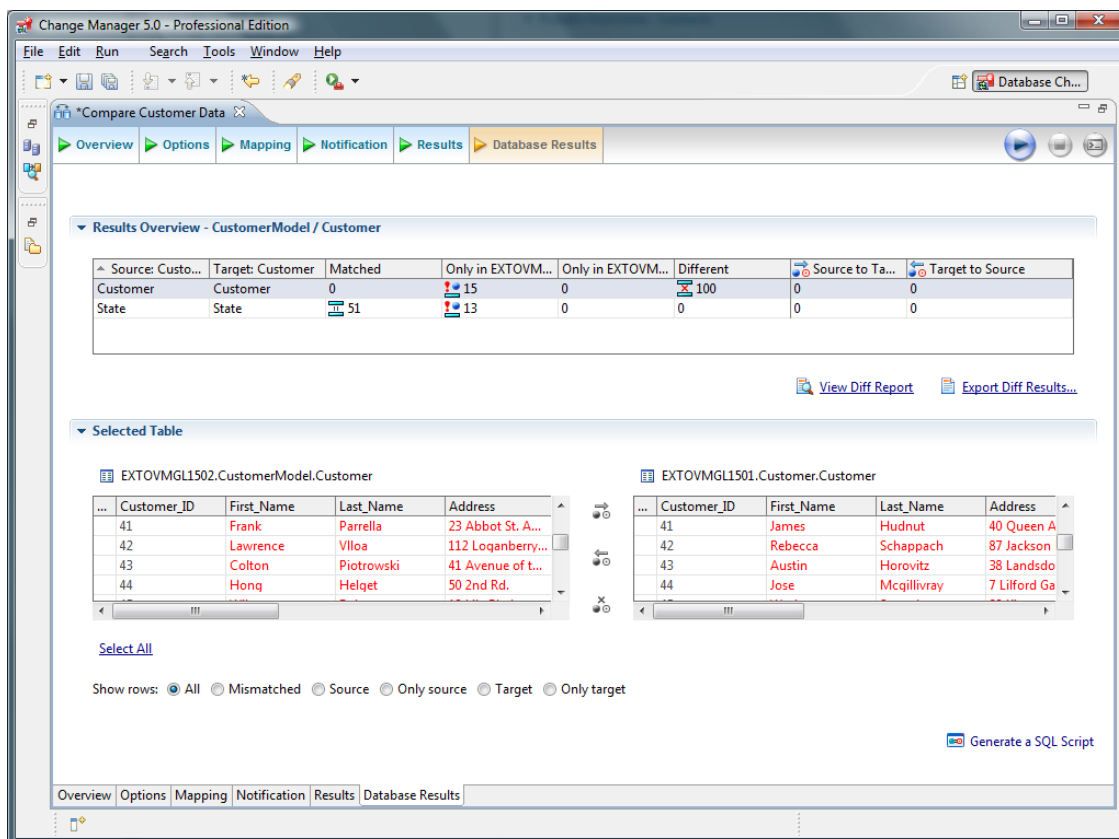


Figure 2 - Data Results: Compare and synchronize data between heterogeneous databases

Change Manager can help ensure that each environment is up to date with the latest schemas, configurations, and reference data. It can also provide versioning for the test environment's schema and database configuration, so that you have an audit trail showing exactly what the environment looked like during various test cycles. Just as with development snapshots, the test snapshots can be compared to view differences, and to roll forward or back between versions.

In addition to establishing and managing the integrity of test environments, Change Manager provides the following QA-centric features:

- Archive and compare database configuration settings to ensure consistency of application-related settings: character sets, locale and time zone, sort order, etc.
- Compare and view differences between two live databases, for example: development and test. Create a script using the comparison as a basis to roll new and modified schema objects forward into test.
- Grab an archive of an upcoming release's database from source code control and analyze differences between it and the current release, as represented in the test environments.
- Compare a master database of an upcoming release with all of your test systems using a one-to-many comparison job to ensure that the entire test lab is on the right version for release testing.
- Use data compare not merely to validate reference data, but to automate the verification of test results. The data compare feature easily compares between a schema containing expected results and the QA database at the end of the test cycle (containing the actual results) to ensure that the application is interacting correctly with the application data.

## REPORT AND RE-CREATE PROJECTS

If someone finds a bug in build 357, you must be able to recreate the application as it appeared in that build—database and all. A good database change management tool makes it easy to retrieve a particular build, so that you can use it to test the corresponding version of the application.

Creating database archives for use in regression testing and being able to report on database changes from build to build helps ensure that the software development process is what you think it is: a traceable evolution of change.

Change Manager reports give you the flexibility of using whatever format fits your process. XML reports can easily be parsed by your own code or served up to browsers, while HTML files can be posted with your team's wiki. Change Manager report formats include HTML, PDF, XML, CSV, and RTF.

Some advanced uses of Change Manager include the ability to integrate custom plug-ins for configuration tracking and notification purposes. You can run multiple jobs to retrieve configuration information about all of your test environments and load them into a database using a custom plug-in. Its versatility means easier integration into your existing tools and processes.

## CONCLUSION

Managing database changes throughout the development process can be a daunting task, but having the right tools makes the difference. Development teams are already under tough deadlines without having to spend time and resources manually managing changes to the database. They need flexible tools that fit within their existing processes, styles, and methodologies.

Embarcadero Change Manager lets development teams streamline communication, ensure accurate tracking, and automate of many manual and repetitive tasks. With a powerful, yet intuitive interface, Change Manager offers quick and easy processes for most tasks but also provides advanced features when you need fine-grain control and flexibility. By placing the power of comparing schemas, data, and configurations; of capturing archives and monitoring for changes; of reporting and sharing; of integrating with source code control, development teams will see measurable results in productivity.



Embarcadero Technologies, Inc. is a leading provider of award-winning tools for application developers and database professionals so they can design systems right, build them faster and run them better, regardless of their platform or programming language. Ninety of the Fortune 100 and an active community of more than three million users worldwide rely on Embarcadero products to increase productivity, reduce costs, simplify change management and compliance and accelerate innovation. The company's flagship tools include: Embarcadero® Change Manager™, CodeGear™ RAD Studio, DBArtisan®, Delphi®, ER/Studio®, JBuilder® and Rapid SQL®. Founded in 1993, Embarcadero is headquartered in San Francisco, with offices located around the world. Embarcadero is online at [www.embarcadero.com](http://www.embarcadero.com).