

Essentials of Test Data Management for Siebel

Optimized and Privatized Test Environments Ensure the Security and Maximize the Flexibility of Non-Production Siebel Instances

Steve Tallant

Director, Product Line Management

IBM

Executive Summary

Companies across industries depend upon the accessibility of customer data to support business processes and gain the upper-hand in a competitive market. But security risks challenge an organization's ability to successfully manage sensitive information, particularly in a testing environment. Today, companies acknowledge that protecting data privacy across the enterprise is essential for complying with privacy legislation and for gaining the trust of customers and business partners.

Around the globe, a wide range of government regulations have been enacted to protect personal information from misuse. In the United States, laws such as the Health Insurance Portability and Accountability Act (HIPAA) and Gramm-Leach-Bliley Act (GLBA) regulate the healthcare and financial services sectors, respectively. The European Union has established the Personal Data Protection Directive as the framework for privacy protection governing its member countries. Similar statutes exist in other countries as well.

British Telecom implemented Princeton Softech Optim™ Siebel Solution for Test Data Management that offers a variety of proven test data management capabilities and methods for de-identifying test data to protect data privacy and support regulatory compliance initiatives. These techniques span from basic – masking alphanumeric characters to substitute confidential data with de-identified data, to complex – methods such as arithmetic calculations, concatenated expressions, date aging and currency conversion. Clients can also incorporate site-specific data transformation routines, integrating the processing logic from multiple related applications and databases.

Companies can apply a single data masking solution across applications, databases and platforms, ensuring that all of their enterprise data is protected. Optim offers a range of techniques for substituting confidential data with de-identified data, giving organizations the ability to choose the most appropriate method for their needs. With Key Propagation capabilities, Optim maintains the referential integrity of information, allowing companies to utilize test data that is realistic and produce valid test results, while complying with privacy regulations.

Ensuring the accuracy, reliability and quality of your Siebel applications has never been more important. Why? Because companies across industries depend on Siebel to manage customer information, engender customer loyalty and drive their business initiatives. In turn, these applications rely on relational databases to store and manage the underlying customer and enterprise data.

The ability to deploy Siebel upgrades and enhancements, as well as manage, maintain and customize and these sophisticated applications is critical for achieving long-term business goals. Companies are striving to improve testing efficiencies and speed the deployment of reliable Siebel applications upgrades, while staying within tight IT budgets.

Now, more than ever, companies face new challenges when designing effective and efficient testing strategies for enterprise applications. Incomplete or flawed test data means inaccurate testing, which can lead to application failure and business disruption. No company wants to risk losing customers, market share, brand equity or revenue by deploying Siebel applications that have not been thoroughly tested.

More common approaches to building test environments include cloning Siebel production environments and writing custom extract programs. However, these methods are labor intensive, error prone and require extensive processing and network resources.

How can IT organizations improve testing efficiencies and reduce the total cost of owning and maintaining Siebel applications? This White Paper examines how proven test data management capabilities can help you ensure Siebel application reliability and achieve optimal business value from your CRM investment.

Importance of Siebel Application Testing

Your Siebel applications drive revenue and satisfy sophisticated marketplace initiatives. These mission-critical CRM applications have a huge strategic impact. As a result, there is increasing focus on application reliability to support the deployment of Siebel upgrades and enhancements, minimize unplanned downtime and provide superior customer service.

According to *Computerworld*, “buggy software is costing the economy billions of dollars.”¹ Software defects directly affect a company’s bottom line. Failure to properly test your Siebel applications through the various deployment phases is recognized as necessity. Can your business sustain the revenue impact and customer loss that could result from a Siebel application failure?

Improved Data Management Supports Application Testing

Improving the way you manage your enterprise application data in the testing environment can deliver dramatic results, including improved reliability, faster time to deployment, and reduced application upgrade and maintenance costs.

Speed Time to Deployment

Deploying reliable Siebel application upgrades and enhancements in a timely manner provides companies with a competitive advantage. Shorter upgrade and deployment cycles are desirable, but reliability testing takes time. This classic struggle between these two business objectives often means that deployment plans and “go live” dates slip. How important would it be to your business if you could get Siebel applications deployed faster without sacrificing accuracy and reliability?

Control Development and Quality Assurance Costs

Reducing the overall cost of installing, testing and deploying reliable Siebel applications can result in significant cost savings throughout the entire application ownership lifecycle. Siebel sites can realize direct cost savings and productivity gains by streamlining testing processes and allowing DBAs, developers and quality assurance testers to focus on additional priorities. Could your business benefit from reducing Siebel deployment costs and using IT resources more effectively?

The costs associated with resolving new functionality issues in the Siebel production environment are as much as 10 to 100 times greater than if these issues are resolved early during the application development and deployment processes. Even worse, when your business users and customers are the first to discover application issues, you risk losing revenue opportunities and customer loyalty. Could your company afford the increased direct expenses and indirect opportunity costs of releasing an unreliable Siebel upgrade?

¹ Sue Hildreth, “Buggy Software: Up From a Low-Quality Quagmire,” *Computerworld*, July 25, 2005.

Lower Infrastructure Costs

In Siebel application upgrade and testing environments, there are typically several clones of the original production database. How many database copies do you use? It is important to consider that as data is duplicated, costs increase proportionally. Using right-sized databases rather than clones can help you save or reclaim storage assets. In addition, by reducing the size of test databases you will also reduce the processing power needed to perform testing activities. Could your business benefit from using smaller, right-sized Siebel test databases?

Deploying Reliable Siebel Applications

Siebel sites are realizing that better planning is needed to satisfy the rigorous testing objectives necessary to deploy reliable Siebel upgrades and enhancements. Designing a comprehensive testing strategy is often taken for granted. Organizations also recognize the need for proven test data management and automated testing software to manage the quality assurance process cost effectively. Siebel application testing is fast becoming a high priority because resolving problems early in the deployment stages can significantly reduce costs.

Many organizations are reluctant to make a significant investment in testing, allocating the money to other areas. However, without a commitment to quality assurance, the business consequences could be significant. It is necessary to understand the impact of unacceptable Siebel application accuracy and reliability on critical business objectives, such as generating revenue and ensuring customer satisfaction.

Protecting your investment in your Siebel applications by using effective enterprise data management capabilities and repeatable testing processes will ultimately increase your return on investment. In today's market, reliability and quality are major competitive differentiators. To remain competitive, testing your Siebel application upgrades and enhancements before deployment must be a high priority.

Why is Siebel Application Testing a Challenge?

Whether your Siebel applications rely on an Oracle, UDB, DB2 or SQL Server database, this complex relational database technology introduces many challenges in the testing process. The Siebel application data model contains hundreds of base tables, as well as interrelated tables associated with numerous business objects, such as Service Requests, Opportunities, Activities, Orders, Invoices, Payments, Marketing Campaigns and so on.

Without the appropriate test data management capabilities, developers often need to write sophisticated extract programs just to create the needed Siebel test data. It is a challenge to navigate the numerous tables, rows and columns to create, manipulate and refresh the desired database subsets. It is also difficult to develop an extract program that considers relationships defined and enforced within your Siebel applications. Finally, whenever the application or database is updated, the extract program must be updated as well.

After a test is executed, there is still no clear and simple method to verify that the results are valid. Adding to the complexity is the challenge of handling heterogeneous database management systems and differing data models. Clearly, you need comprehensive testing capabilities that support these relational complexities and “remember” to account for them in every extract, compare or update operation.

Improving Your Siebel Testing Environment

The quality of the Siebel test environment is just as critical as the quality of the test data. So, what is needed to improve Siebel application testing?

- **Using Realistic Test Data** – Building and populating realistic test databases requires the capability to extract and move precise subsets of related data based on your specifications. Maintaining referential integrity with 100 percent accuracy is essential for the simplest and most complex data models. Working with predefined realistic subsets of Siebel data improves testing and overall application quality.
 - **Protecting Privacy** – Comprehensive capabilities for transforming and de-identifying test data are required. This added flexibility allows for creating special test data and is especially important to comply with government regulations for protecting privacy.
 - **Improving Testing Efficiencies** – As the content of the test database is modified during the iterative testing process, it diverges further from the baseline test data, resulting in a less than optimal test environment. Without capabilities for saving and reusing the Siebel test data, it is more difficult to refresh the test database to ensure accurate results on subsequent test runs. The ability to save and reuse extract, insert and load processing specifications streamlines the testing process and ensures a consistent and manageable test environment.
 - **Accommodating Changes in the Data Model** – Unless the extract process includes metadata, there would be no way to accommodate changes in the Siebel data model during the testing phases. For example, metadata documents the structure of the database including the tables, columns, relationships, views, triggers and so on. Including metadata in extract processing ensures that you can recreate the test data and accommodate data model changes quickly and accurately.

- **Forcing Error Conditions** – Creating test data to force error conditions requires intelligent browsing and relational editing capabilities. Relational browse features improve the ability to perceive data relationships clearly and to identify and resolve problems quickly. The ability to insert rows and edit database tables directly improves productivity and accuracy. Multi-level undo capabilities are essential.
- **Comparing Test Results** – The ability to identify data anomalies and inconsistencies during testing is essential to the overall quality of the application. The only way to truly achieve this goal is to deploy an automated capability for comparing the “before” and “after” images of your test data. Speed and accuracy are essential. Automated compare processing saves time and ensures that you can identify problems that would otherwise go undetected.

Supporting Your Siebel Testing Requirements

Each phase, from unit testing through system integration and acceptance testing has unique requirements and varying levels of complexity. Typically, each time a new Siebel version is released, new test databases are created. Since it is ideal to use “realistic” test data, the test database is usually a clone of the production database. Special test cases may be added before testing begins.

Iterative testing involves executing the application using the test database and verifying the results to ensure the application is working as designed. Any problems discovered must be resolved, and the test data must be refreshed, before testing continues. This process is repeated throughout the various testing phases (unit, integration, system, load, regression and acceptance testing) until the Siebel upgrade is migrated into production.

In addition to finding and resolving application errors before deployment, another important goal is to create a repeatable testing process that improves application quality, reduces time to market and minimizes costs. So how can you find a test data management solution that meets your needs?

Improving Testing Efficiencies

The solution must provide proven capabilities to extract, browse, insert, load, transform, edit and compare complete subsets of related data. Siebel sites can design comprehensive testing strategies that include realistic test data while improving productivity and overall application quality and reducing costs (see Figure 1).

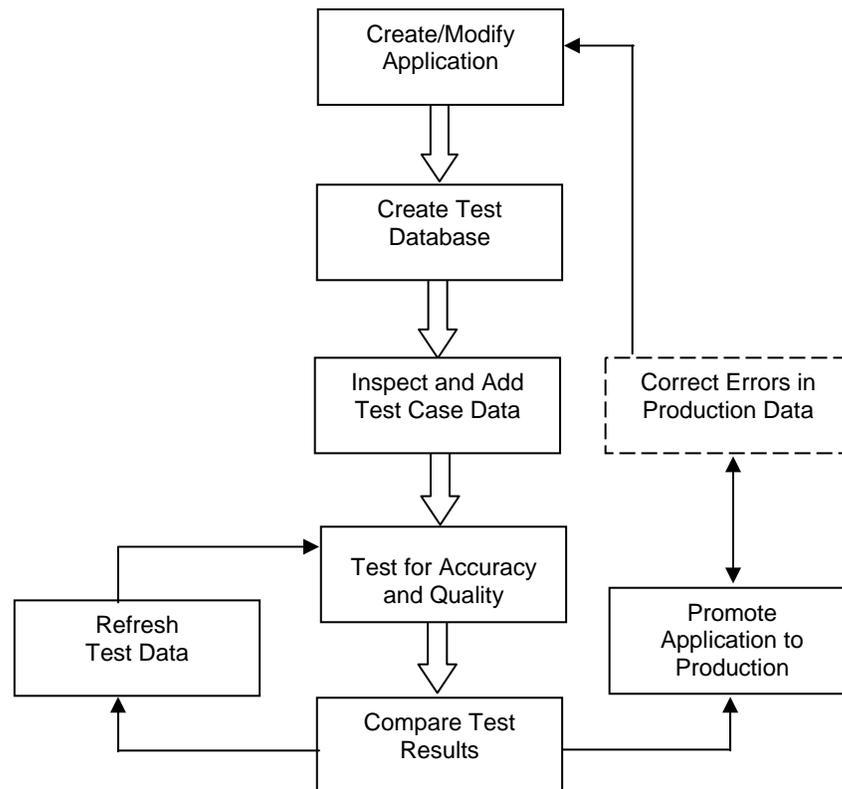


Figure 1. Test Data Management improves every stage of the application testing process.

Masking and Transforming Siebel Test Data

With the increased focus on data privacy, the ability to transform and de-identify sensitive Siebel data in the development and testing environments is critical to prevent data breaches and severe penalties. The solution provides flexible technology for mapping source and destination columns and capabilities for using a variety of transformation functions and algorithms.

For example, it would be easy to de-identify customer identification numbers or social security numbers by simply applying a random number function. More sophisticated capabilities would allow you to use substrings, sequential numbers, date aging, currency conversion, or table lookup functions. The capability for including user-defined data transformation programs would provide even greater flexibility to satisfy complex or site-specific requirements.

All of the methods described are effective techniques for de-identifying test data. However, in testing relational database applications, there is an added complication. Specifically, the tester requires the ability to propagate a masked data element in a parent table to all of the related child tables in the database. Key propagation is necessary in order to retain the referential integrity of the transformed data. Otherwise, the relationship between parent and child tables is severed, test data is inaccurate, and application testing yields invalid results.

Key propagation also provides the ability to “manufacture” new sets of test data. This technique is helpful when additional data is required for testing, as is the case when testing new applications or testing applications when new tables are added to the data model