

Enterprise Data Management Solutions
October 2008



IBM **Information Management** software

Data management strategies to improve application testing in the IBM z/OS environment

Contents

2 Executive summary

3 Application testing presents challenges

5 Current practices in z/OS testing environments

6 Why formalize test data management?

7 Formalizing test data management with IBM Optim

13 IBM Optim test data management case study

15 Improving test data management delivers business value

Executive summary

The IBM® z/OS® mainframe environment has earned a hard-won reputation for superior application performance, availability and data security, while managing extremely large volumes of application data. To maintain a competitive advantage, organizations that manage their mission-critical operations on the IBM System z® platform must continually look for improvements in technology, processes and best practices.

One key area that requires constant improvement is application testing. The need to ensure the accuracy, reliability and quality of mission-critical z-series applications has never been more important. Why? Because organizations of all types depend on these applications to support daily business operations. Implementing new application enhancements, customizations and upgrades are essential for sustaining market leadership.

Designing efficient and effective data management strategies for testing applications in the z/OS environment presents many challenges. First, IT organizations are challenged to choose between the two most common approaches to building test environments, namely cloning production databases and writing custom extract programs. Each approach has advantages and disadvantages. An effective strategy would help deliver the best of both approaches, while minimizing costs.

In addition, z/OS applications often require a broad range of skills to capture test data from both relational and non-relational data sources. The ideal approach would maximize skills, while reducing costs. Lastly, there are increasingly strong business and legal reasons to protect the privacy of confidential information, especially in non-production (development, testing and training) environments. A best practices solution would provide data masking capabilities to protect privacy, while providing data that is valid for testing purposes.

So, how can your IT organization improve testing efficiencies and protect privacy in the z/OS non-production environments, while reducing costs throughout the application lifecycle? This white paper explains how a formalized Test Data Management strategy and the IBM® Optim™ Test Data Management Solution can help your organization deliver reliable applications at a lower cost to help maximize your investment in critical business applications.

Application testing presents challenges

Improving application testing in the z/OS environment can present both business and technical challenges. From a business perspective, mainframe application quality is easily and routinely reported by means of application performance metrics, percent availability, and/or the number of outages over a given time period. Organizations, applications and individuals are often evaluated and compensated directly based on the achievement of service level agreements. In addition, business units hold a common expectation that application quality should improve over time, even as new functionality is added.

From a technical perspective, developing and testing applications in the z/OS environment presents additional challenges:

- **Managing large production databases.** Large volumes of production data naturally give rise to large volumes of test data, increasing hardware (storage and CPU) costs and administrative burdens on skilled personnel. Developers often need to clone entire production databases or write complicated extract scripts just to create the test data. It is a challenge to create, manipulate and refresh the desired subsets, while preserving the integrity of the data throughout the development and testing environments. In addition, whenever the application data model is updated, any extract programs must be updated as well.

- **Creating test data from diverse data sources.** Application data in the z/OS environment can be spread across significantly dissimilar data sources. While workstation-based application environments might include different databases, these databases are likely to all be relational. In contrast, a mainframe application might easily have relational (IBM DB2®), hierarchical (IBM IMS™) and structured file (IBM VSAM® and sequential) data sources. Addressing these complexities requires additional skills. There may be no single tester or developer with all the skills needed to manage the test data for a given application. Agile enterprises need a solution that accesses relational data, as well as VSAM, sequential and IMS data, and correlates that data in a way that can be easily “understood” and managed.
- **Moving test data from one data source to another.** Creating test data for applications that include different hardware, operating systems and data models is complex enough. Syntactic and semantic differences, data compatibility and translation issues only add to this complexity. For example, it may be necessary to extract subsets of data from a production VSAM dataset and insert the data into a DB2 test database. This transition may cause translation problems with data types, such as dates and time-stamps. When working with several databases, you need additional capabilities for managing compatibility and translation differences automatically.
- **Protecting privacy.** With the increased focus on data privacy, the security measures inherent in the z/OS environment may not be appropriate for protecting privacy in non-production environments, where developers and testers need more access to “realistic” production data, not less. Encryption schemes provide a complementary technology, but are not a substitute for masking or disguising confidential data used in application development and testing. The capability to transform and de-identify sensitive data in the development and testing environments is recognized as a best practice for making the data useless to thieves and hackers.

- **Sharing resources and skills for development and testing.** More often than not, application development and testing skills and resources are shared in the z/OS environment. For example, the skills of a system programmer, DBA, application developer and tester may all be required to setup and execute even the simplest test scenarios. Furthermore, multiple testers may more commonly be sharing the same databases. Capabilities that maximize skills and resources, while lowering costs, provide a clear advantage.

Current practices in z/OS testing environments

New test databases are created and refreshed each time an application in the z/OS environment is developed or modified. Because it is ideal to use realistic test data, the test database is usually a clone of the production database. The increased data volume that results from cloning an entire production database also increases the time, capacity requirements and CPU processing resources needed to run test scenarios.

It is much faster to test with smaller, realistic subsets that accurately reflect the production data, without adding overhead to the testing process. However, this approach would require writing extract scripts that have to be managed and maintained across a variety of data sources, as well as different programming skills and a dedicated staff, which increases costs.

Whether cloning or extracting data, it is an ongoing challenge to manage updates that occur as developers make changes to the current production data model. In addition, cloned or extracted data may not support the specific error and boundary conditions required to test exception handling. Special test cases may be required.

Iterative testing involves executing the application using the test database and verifying the results to validate that the application is functioning as designed. Any problems discovered must be resolved, and the test data must be refreshed before testing continues. In the z/OS environment, refreshing the test environment to a prior state can be complicated. After a test is executed, there is often no easy or automated method for verifying the results. This process is repeated throughout the various testing phases, from unit test through regression test, until the application is placed into production.

Why formalize test data management?

It is important to formalize the management of test data for the following reasons:

- **Meeting business requirements.** Business managers and project leaders are striving to improve application reliability, increase functionality, shorten the time to market, and reduce development and testing costs throughout the application lifecycle.
- **Meeting technical requirements.** Developers and testers often have different test scenarios and different needs. An individual developer needs only a small amount of data for unit testing versus system integration testing that requires much more data. A usability test is likely to need different data than a performance test. There are significant cost savings (hardware and skills) and productivity improvements if the data can be 'right-sized', tailored, standardized and shared for a given test.
- **Meeting regulatory requirements.** There are growing business and regulatory requirements to protect the privacy of personal and other data. Mainframe data is centrally managed, so expectations for the security and protection of that data are high. Test data represents a significant exposure. Mainframe production data is well guarded, but production copies for test or other non-production uses (sometimes even outsourced and sent overseas) are a major gap in the protective wall.

Best practices in data governance include formalizing the test data management process. Using proven test data management capabilities, like the IBM® Optim™ Test Data Management Solution for z/OS can provide data privacy protection in a way that is transparent to the application and to test-related personnel.

Formalizing test data management with IBM Optim

The IBM Optim Test Data Management Solution enables IT organizations to meet even the most complex application testing challenges by providing all the fundamental components of an effective test data management strategy.

What makes Optim unique? The Optim Relationship Engine™ is a unique technology that understands and processes related data from multiple data sources and ensures that each test subset is always referentially intact and logically complete. For example, in a subset of customer data, one customer may have items that are backordered, while another may not. From one customer to the next, the amount of data retrieved from any number of data sources will vary. But the Relationship Engine always extracts the right items for the right customers, complete and intact, every time.

Optim uses an active repository to store the user-defined business rules. The repository automatically captures relationship information defined to the database. Using the active repository, IT staff members can define, share and reuse different Access Definitions that specify, in essence, varying subsets of relational and non-relational data sources. Additional data model information can be populated from third party dictionaries or data modeling software. Finally, users can define relationships that exist, but are not known to the DBMS, such as application-managed relationships.

With Optim, organizations can design comprehensive testing strategies that include realistic test data, while improving productivity and overall application quality and reducing development costs. Optim capabilities allow users to migrate, browse, insert, load, transform, edit and compare complete subsets of related data (Figure 1).

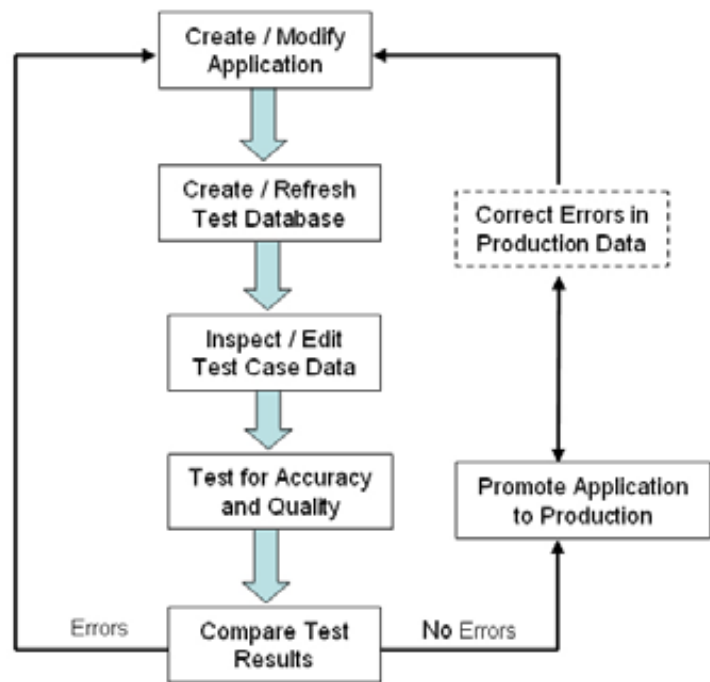


Figure 1. IBM Optim improves every stage of the application testing process.

After a new application or enhancement is ready for testing, Optim helps create the testing environment, allowing developers and testers to inspect and add test case data needed to force error and boundary conditions. Next, iterative testing results can be compared to automatically validate expected and unexpected results. Optim’s extract and insert capabilities reduce the time to refresh that test database. Once results are as expected, the new application can be promoted into production.

Creating realistic, right sized test databases. Optim's subsetting capabilities allow you to create realistic data to test application functionality and performance. Right-sized test databases are small enough to support rapid test runs and large enough to accurately reflect the variety in your production data. For example, a test case may require extracting all information relating to specific customer accounts in a particular sales region. The extract must include all customer identifying information, as well as order and payment history, and the data must be extracted from many interrelated data sources (see Figure 2).

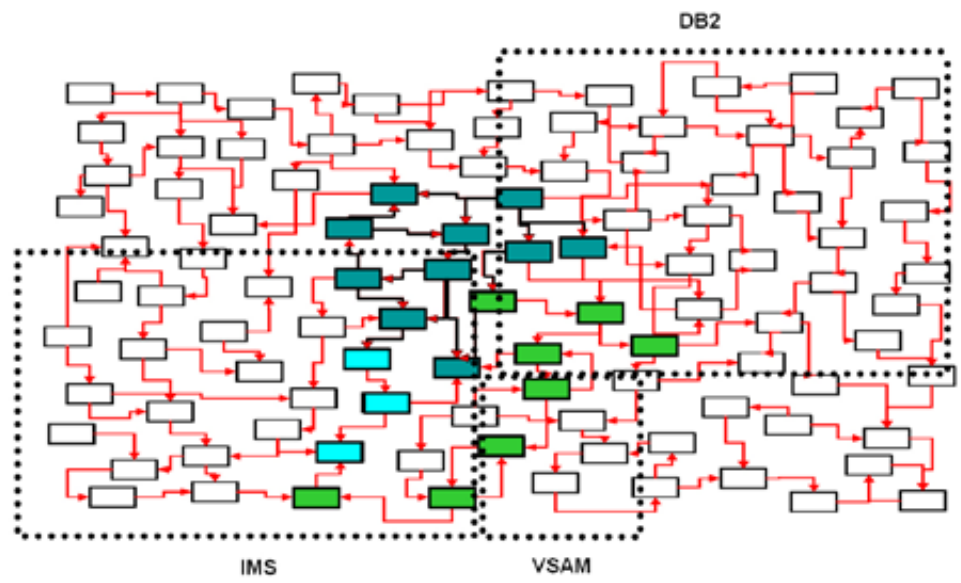


Figure 2. Extracting a precise subset of related data from interrelated data sources.

With Optim, you can extract and move precise subsets of data from DB2, IMS, VSAM or sequential file data sources, based on your specifications, without writing custom programs. This capability keeps skill costs much lower than either cloning or programmatic extract approaches. Using subsets of test data also lowers capacity and CPU costs.

Optim references data from each different data source as “tables” and manages the differences, automatically. Simply specify the selection criteria to identify the precise set of related data you need to test a particular application function. Optim captures the data and optionally the DB2 metadata that defines tables, columns, referential integrity relationships, views, triggers and so on. Optim includes metadata (DB2 DDL) in extract processing, which enables replicating test environments and accommodating data model changes quickly and accurately.

Masking and transforming test data while maintaining data relationships.

Capabilities for de-identifying confidential data allow you to protect privacy, while providing the necessary realistic test data to developers and quality assurance staff. Optim provides flexible technology for mapping source and destination columns and capabilities for using a variety of transformation functions and algorithms. For example, with IBM® Optim™ Data Privacy Solution, it is easy to de-identify customer identification numbers by simply applying a random or sequential number function. More sophisticated masking capabilities allow you to use substrings, concatenated expressions or table lookup functions. The capability to include user-defined data transformation programs provides even greater flexibility to satisfy complex or site-specific requirements.

All of the methods described are effective techniques for de-identifying test data. However, there is an added complication because data relationships must be maintained. Specifically, the tester requires the capability to propagate a masked data element in one record to all of the related records.

Key propagation is necessary to retain the referential integrity of the transformed data. Otherwise, data relationships are severed, test data is rendered inaccurate, and application testing yields invalid results. Optim's key propagation can be used to enable independent testing in shared databases. Testers can create and work on independent copies of the "same" test data in a shared environment by reusing the same extracted data with different key values (and changing the other values if necessary).

Optim's data transformation functions also provide the capability to "manufacture" new or changed 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 or other changes are made to the data model.

Preserving data integrity. Whether testing with relational or non-relational data, unless you can preserve the integrity of the data, application testing will be flawed. Using realistic, referentially intact subsets of production data from multiple data sources promotes accurate and efficient testing and reduces capacity requirements. This data integrity must be preserved even when data is masked and propagated across the testing environment.

Optim respects the Referential Integrity (RI) rules enforced within the database and your applications. Typically, the application-enforced RI is more complex. For example, the application may include relationships that use compatible, but non-identical, data types, composite and partial columns and data-driven relationships. Optim has proven capabilities for handling every type of relationship. Optim maintains both database and application referential integrity with 100 percent accuracy, even for the most complex application data models.

Creating targeted test scenarios. Creating realistic subsets of related test data from a production database is a reasonable start, but it is sometimes necessary to edit the data to force specific error conditions or to validate specific processing functions. Optim includes comprehensive relational editing capabilities that not only simplify the tasks necessary to create this special data, but also make it easy to browse data and resolve application errors. The capability to insert rows and edit database tables directly improves productivity and accuracy. The ability to browse data in its relational or business context also provides a clear way to envision the data relationships and structure of the application data model.

Validating the test results. Without the capability to automatically compare the test data before and after a test run, validating test results and identifying changes is next to impossible. The only way to truly achieve this goal is to deploy an automated capability for comparing the baseline test data against results from successive test runs. Speed and accuracy are essential. The ability to compare only a related subset of test data is desirable.

Optim provides the capability to compare subsets of related DB2 data and to identify the differences automatically. With DB2 relational data, this means more than comparing row to row. It means using data model intelligence to compare related

sets of rows. For example, after developing a new product rate table, you can compare the “before” and “after” test results to validate that rate changes are being applied appropriately. This capability allows multiple testers to use one test database and compare only the portion of the database that is their individual testing focus.

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. Test data management capabilities for saving and reusing extract, insert and load processing specifications help streamline the testing process and maintain a consistent and manageable test environment.

Automating application testing. Developing an effective test strategy requires both automated testing processes and test data management. Automated testing software emulates user interactions that exercise and test application functionality. Test data management provides appropriate and realistic test data to ensure that automated test scripts exercise all paths of application logic and generate reliable results. When deployed together, automated testing and test data management capabilities provide a powerful, efficient combination that increases the value of automated testing.

IBM Optim test data management case study

Cetelem, a subsidiary of the BNP Paribas Group, offers consumer credit and associated financial services to clients around the world. To support its continued business growth, Cetelem relies on customer-focused enterprise applications to support its daily operations around the world.

The core mainframe applications are managed in an IBM z/OS and OS/390® operating environment. The production application database contains over 600 DB2 tables consuming more than 1 terabyte of capacity, and the qualification and user acceptance testing environments together consume more than 24 gigabytes of capacity.

Cloning large databases for testing purposes was a lengthy process, especially for processing iterative testing scenarios and refreshing the test environment. As a result, it became more time consuming and costly to create and manage multiple test environments. It was also more difficult to validate the reliability of new functionality and to complete testing processes in time to deliver that functionality to business users, partners and customers.

“Our staff wanted to implement more efficient and cost-effective testing processes that would shorten the time for creating and managing multiple test environments,” said Michèle Davain, DBA Manager within the Technical Department at Cetelem. “We also wanted to implement test data management strategies that would allow us to accommodate frequent changes and customizations, without sacrificing quality.”

“Because Optim enables us to define accurate subsets of related data specific to each application, we have more flexibility within our testing organization and can provide better quality control,” said Davain. “With Optim subsetting, we have implemented a change management solution and a quality-oriented process for our entire DB2 mainframe platform. Optim capabilities support intensive use by more than 310 developers and quality assurance testers, who can now easily create and refresh multiple test environments on demand, while ensuring accuracy and reliability.”

“IBM Optim has improved the productivity and the quality of our processes for creating and maintaining our testing environments and meeting the specialized

requirements for each specific project. We have also started using Optim's data masking capabilities to protect privacy in our development and testing environments," noted Davain. "Optim's user-friendly and flexible capabilities have quickly gained the approval of our development and quality assurance teams. The performance and the robustness of Optim are very much appreciated."

Improving test data management delivers business value

Optim provides a single solution for managing enterprise application data throughout every stage of the information lifecycle. Now you can implement a formalized and proven test data management strategy to speed application deployment by streamlining the way you create and manage test environments:

- Extract referentially intact subsets of data with 100 percent accuracy to build realistic and "right-sized" test databases. Eliminate the expense and effort of maintaining multiple database clones and lower costs associated with capacity and CPU resources.
- Integrate relational and non-relational test data from multiple related data sources (DB2, IMS, VSAM and sequential files). Eliminate the need for special extract programs and associated costs.
- Insert or load subsets of related data to quickly build realistic test databases. Update or refresh the test data consistently to preserve the integrity of the test environment.
- De-identify and transform sensitive data in the development and testing environments to minimize the potential for unauthorized disclosure and support compliance with regulatory requirements for data privacy.

- Transform data to support changes in the application data model.
- Browse and edit DB2 test data to force error conditions and resolve problems. Reviewing data in its relational business context provides a clear vision of the data model.
- Compare DB2 test data before and after exercising the application to validate expected test results and identify anomalies automatically and with pinpoint accuracy.
- Integrate test data management with automated testing to realistically test application functionality with complete data accuracy.

You also benefit from knowing that Optim scales to support your current and future test data management needs. Optim supports all leading enterprise databases and operating systems, including IBM DB2®, Oracle®, Sybase®, Microsoft® SQL Server®, IBM Informix®, IBM IMS™, IBM VSAM®, Microsoft Windows®, UNIX®, Linux® and IBM z/OS®. And it supports the key ERP and CRM applications in use today – Oracle® E-Business Suite, PeopleSoft® Enterprise, JD Edwards® EnterpriseOne, Siebel® and Amdocs® CRM, as well as your custom and packaged applications.

About IBM Optim

IBM® Optim™ enterprise data management solutions focus on critical business issues, such as data growth management, data privacy compliance, test data management, e-discovery, application upgrades, migrations and retirements. Optim aligns application data management with business objectives to help optimize performance, mitigate risk and control costs, while delivering capabilities that scale across enterprise applications, databases and platforms. Today, Optim helps companies across industries worldwide capitalize on the business value of their enterprise applications and databases, with the power to manage enterprise application data through every stage of its lifecycle.

For more information

To learn more about IBM Optim enterprise data management solutions, contact your IBM sales representative or visit: www.optimsolution.com.



© Copyright IBM Corporation 2008

IBM Software Group
111 Campus Drive
Princeton, NJ 08540-6400
U.S.A.
www.optimsolution.com

Produced in the United States of America
10-08
All Rights Reserved.

DB2, IBM, the IBM logo, IMS, Informix, Optim, the Relationship Engine, VSAM and z/OS are trademarks or registered trademarks of the IBM Corporation in the United States, other countries or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. UNIX is a registered trademark of The Open Group in the United States and other countries. SQL Server and Windows are registered trademarks of Microsoft Corporation in the United States and other countries.

All other company or product names are trademarks or registered trademarks of their respective owners.

References in this publication to IBM products, programs or services do not imply that IBM intends to make them available in all countries in which IBM operates or does business.