Efficient Preparation and Utilization of Test Data

Contents

1. Abstract

*With computers being the heart of today's world, applications being built need to be properly tested. Good **quality test data** is one of the major factors contributing to successful testing. Efficient **test data management** is imperative in ensuring software quality. The fact that test data plays a vital role not only in testing but also the entire software lifecycle process is often forgotten. By creating quality test data, defects can be detected at an early stage in the software lifecycle process which in turn helps to reduce cost and time to market and improves quality.*

But is it easy to create quality test data? How to manage it effectively? Can the effort for arranging test data be reduced? These are some of the questions that arise when project teams strategize about test data. While technology is enabling faster and richer data retention, the real challenge still lies in preparing quality test data and making good use of it.

The intent of this paper is to discuss an approach for the creation and utilization of test data thereby improving the quality and coverage of testing software applications.

2. Introduction

In today's world, all the organizations have three critical business goals: **Improving Business Agility**, **Increasing Revenue** and **Mitigating Possible Risks** to the business. The realization of these goals is the basis of success for any organization.

To help the organizations achieve these goals, IT and Business teams need to deliver quality products which are accepted by the customers. The IT team needs to deliver quality software on time and within the allocated budget. The basic building block of a coherent strategy for Software Quality is proper testing which in turn is based on appropriate and accurate test data. Most teams find that the hardest part of testing is finding the test data which fits the pre requisites for testing. Well-planned data provides flexibility and helps reduce the cost of testing and further maintenance quite a lot.

Generating the entire test data manually is not feasible as it is too slow and error-prone, and it can never prove the reliability of an application with the same level of confidence as real data. So, how can you acquire good quality test data, and how can it be managed effectively? This is where test data management becomes an important part of your overall testing strategy.

Some of the problems that arise with not having an effective test data management strategy are Inadequate testing, Increased time-to-market, Increased costs from redundant operations and rework and Non-compliance with regulatory norms on data confidentiality. Robust test data management processes are essential in maintaining applications and databases. In addition to this, the recent rise in identity theft, industry regulators and law makers continue to put pressure on organizations that prefer to use non standard techniques to provision test data.

3. Problem Statement

Test Data management and preparation of test data seems very simple but there are quite a few challenges involved:

- *Realistic data is difficult to collect* - With today's huge business applications; data is typically spread across multiple systems and databases. This makes data extraction a time-consuming process and also the testers have limited skills for dealing with the range of databases and schemas. It all adds up to a lot of lost time during the testing process.

- *Complexity of requirements* - Requirements are quite complex and thus preparation of test data for fulfilling all given requirements becomes very complex and may require understanding of various domains and systems.

- *High Storage costs* - As the number of business applications rises and the amount of data they handle explodes, storage maintenance costs are becoming a significant drain on IT budgets. Given the high cost of storage maintenance, your QA team needs to reduce the amount of data it stores and manages. It is not cost effective to clone and maintain an entire production database when you actually need just a relevant subset of the data for testing.

- *Using Non-Referentially Intact data* - It is hard to maintain the referential integrity of data when the data is taken out of a production environment or created manually.

- *Sudden Application Changes* - Sometimes the application may undergo a change suddenly and immediate requests for test data have to be catered by the testing team incorporating the changes that have occurred.

- *Data Confidentiality* - Social security numbers, credit card numbers and other personal and business information are an attractive target to hackers, data thieves and others. When production data is used for QA tests, we need to ensure that information is available only to authorized users.

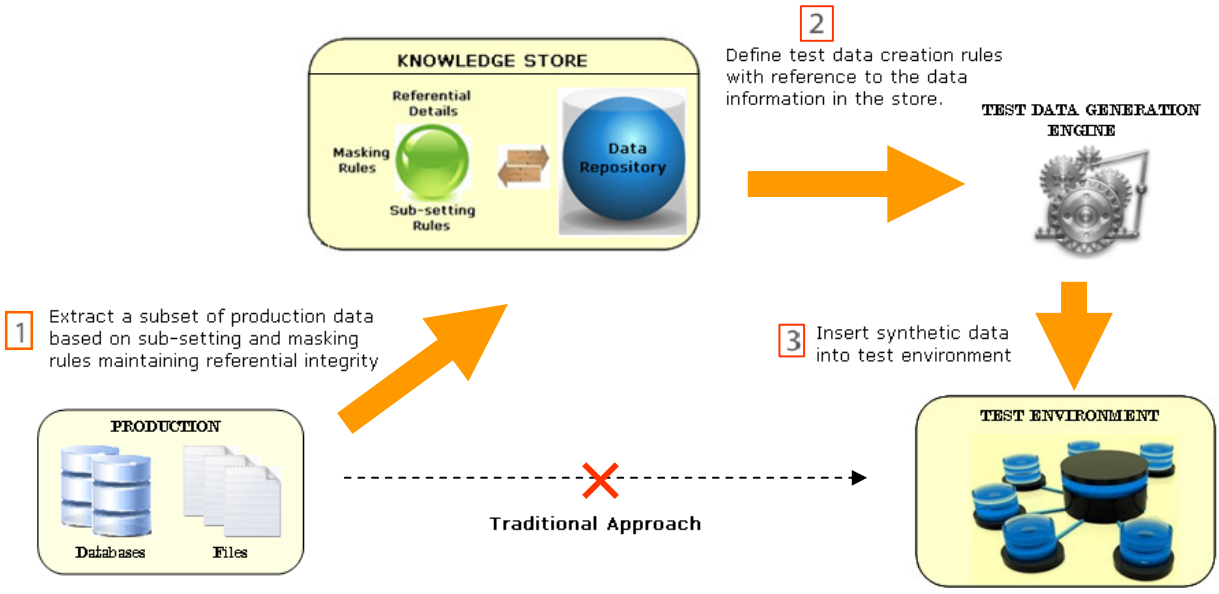
- *Test data exhaustion* - During testing cycles, there are chances that test data gets used and cannot be reused again.

4. Proposed Solution

It is critical to generate accurate test data so as to test how an application would behave in production. Good quality test data is the key to testing existing and new applications. The traditional approach of extracting data from production and loading it to the test environment is not desirable as the costs involved with this activity is high. Manually creating test data would require a lot of effort. What we require is an approach which would reduce the test data management efforts ensuring a smooth and well defined workflow.

Below is a test data management approach (Refer figure 1) for the effective preparation of test data.

Figure 1: A test data management approach



Knowledge Store

Extraction of data from production to test environment as and when required is not a good approach. Instead, we propose to have an intermediate repository which can serve as a base database and act as reference for the test data generation engine which creates and loads the data to the test environment whenever required. This repository is referred in our solution as the "Data Repository".

The data repository is a relational database like Oracle into which the data extracted from production is stored. Data in the production environment is analyzed and using this information all referential details is identified. Masking rules are created to ensure compliancy to privacy regulations by depersonalizing sensitive information such as

credit card numbers, social security numbers etc. Sub-setting rules are prepared to extract a consistent subset from the production environment. The combination of data repository, rules (masking, sub-setting) and referential details is called as the "Knowledge Store".

Referentially intact data is extracted based on sub-setting rules and privacy information is removed from the extract using masking rules prepared based on business values. The data in the knowledge store can be replenished by moving data from production to the data repository when required.

Synthetic Data Generation

With the information available in the data repository, fresh synthetic data will be prepared using data generation tools available in the market such as Datamaker etc. and then loaded to the test environment. With such an approach data provisioning will be easily handled since the data is generated separately for the various requests.

The biggest advantage of synthetic data generation is when test data is required for test cases with specific requirements or when the necessary data is not available in the test environment (E.g. data for testing negative scenarios). In such scenarios, the traditional approach of extracting data from production might not be the solution. Test data for negative scenarios can be created by specifying the required conditions in the test data generation engine. Also, it is easier to generate huge volumes of data for performance testing using this technique.

By making good use of the information available in the knowledge store, generation of synthetic data for testing becomes very easy. This also solves the problem of privacy sensitive information since the data generated, even though resembles production is purely fictive.

The different scenarios to be considered for test data generation are:

- *Default or empty data set* – This is necessary to test for proper error handling in the application.
- *Valid data set* – Check if the application is as per requirements and validates the data inputs.
- *Invalid/Illegal data set* – Check application behavior for negative test scenarios and to test the acceptance of data in incorrect format.
- *Performance/Load/Stress data set* – This would require large amount of data for performance related analysis.

This way creating separate data sets for each test condition and generating data based on the available information will ensure complete test coverage.

To summarize,

1. Extract relationally intact production data into the knowledge store using the defined sub-setting and masking rules.
2. Classify the test cases into separate data sets based on the requirements.
3. Define test data creation rules in the data generation engine with reference to the data available in the knowledge store.
4. Generate synthetic data and load the data into the test environment as per requirements.

   This way data can be easily generated and loaded into the test environment as and when required.

5. Future Direction

Going forward with this approach, the knowledge store in time attains a self sustained state and allows test data modeling to be carried out with minimal dependency on the production database thereby reducing overall costs.

Furthermore using this proposed solution, an end to end test data management workflow called ***''Test Data on the Go"*** can be easily created. A centralized test data management provisioning team manages and controls the overall data management activities of the entire organization. IT teams will be able to create test databases on demand from the convenience of their machines by placing a request to the central team. Specific test data will then be extracted based on a collection of templates which are already defined. Requesters simply specify the target environment and the specific data will be delivered to the test environment. If a required template does not exist, then requestors can place a new request for the required template and it will be executed by the provisioning team.

Requests will be executed in queues based on priority and executed online without too much overall impact on existing environments. With precise data being delivered, the teams will be able to improve testing efforts. Several requests can be raised and completed on demand in a single day. Reservation of data will also be carried out if required to ensure that data overlap does not take place across requests. Creating of test databases in turn takes much less time since the templates defined is shared and reused across teams. Such a workflow can drive and promote overall harmony in test data management processes within an organization.

6. Conclusion

Today's organizations are constantly trying to meet the requirements of an ever-changing market and demand the same from their IT teams. Manually creating data will never allow this to be done efficiently and effectively. However, using our proposed solution offers a way out by creating data which is entirely synthetic. This allows the data to be easily configurable and conforming to all industry regulations. Using the knowledge store, we are able to easily construct a pictorial representation of how the data should look like and then use this information to create new data with all the characteristics of production data. Also by using coverage techniques such as All Pairs, synthetic data can also ensure that proper coverage of tests are done by creating all possible valid and invalid data combinations. The final result is a low volume data set which will be perfect for high quality testing without incurring huge costs.

The use of the knowledge store provides a central location which is easily accessible by everyone in the organization. It enables parallel development, vastly reducing the time needed to fix any errors. Synthetic data can greatly improve the efficiency of data management techniques and provides the following benefits:

- Data can be easily created from scratch and thus can be tailor made to suit specific requirements.
- The Knowledge store maintains referential and relational integrity and can be used as a reference.
- Synthetic data increases the coverage of test scenarios and reduces overall disk storage.
- Synthetic data creation reduces chances of manual error.
- Knowledge store provides a one stop shop for performing all activities related to test data.
- Synthetic data creation allows data to be reused for future tests.

Appendices

Appendix A – References

http://www.grid-tools.com

http://www-01.ibm.com/software/data/data-management/optim-solutions/

http://www.solix.com/solix_EDMS_test_data_management.htm

http://www.oracle.com

Appendix B – Authors

Ajay Kumar Kachottil (Ajay_kacottil@infosys.com) is a Test Analyst at Infosys Technologies Ltd. He is specialized in Test Data Management and has worked on many large testing programs, especially in the Banking sector.

Anil Kumar Appukuttan (Appukuttan_Kumar@infosys.com) is a Test Analyst at Infosys Technologies Ltd. He is specialized in Test Data Management and has worked on many large testing programs, especially in the Banking sector.

Abhishek Shanker (Abhishek_Shanker@infosys.com) is a Project Manager at Infosys Technologies Ltd. He has rich experience in Managing Large Testing programs and in Test Automation mostly with Banking and Retail sector clients.