RBCS
TIME TESTED.
TESTING IMPROVED.
www.RBCS-US.com

Challenges of Testing with Production Data

This article is excerpted from Chapter 3 of Rex Black's book Managing the Testing Process, 3e.

A number of RBCS clients find that obtaining good test data poses many challenges. For any large-scale system, testers usually cannot create sufficient and sufficiently diverse test data by hand; i.e., one record at a time. While data-generation tools exist and can create almost unlimited amounts of data, the data so generated often do not exhibit the same diversity and distribution of values as production data. For these reasons, many of our clients consider production data ideal for testing, particularly for systems where large sets of records have accumulated over years of use with various revisions of the systems currently in use, and systems previously in use.

However, to use production data, we must preserve privacy. Production data often contains personal data about individuals which must be handled securely. However, requiring secure data handling during testing activities imposes undesirable inefficiencies and constraints. Therefore, many organizations want to anonymize (scramble) the production data prior to using it for testing.

This anonymization process leads to the next set of challenges, though. The anonymization process must occur securely, in the sense that it is not reversible should the data fall into the wrong hands. For example, simply substituting the next digit or the next letter in sequence would be obvious to anyone—it doesn't take long to deduce that "Kpio Cspxo" is actually "John Brown"—which makes the de-anonymization process trivial.

In addition, Kpio Cspxo and other similar nonsense scrambles make poor test data, because they are not realistic. The anonymization process must preserve the usefulness of the data for localization and functional testing, which often involves preserving its meaning and meaningfulness. For example, if the anonymization process changes "John Brown" to "Lester Camden," we still have a male name, entirely usable for functional testing. If it changes "John Brown" to "Charlotte Dostoyevsky," though, it has imposed a gender change on John, and if his logical record includes a gender field, we have now damaged the data.

Preserving the meaning of the data has another important implication. It must be possible to construct queries, views, and joins of these anonymized data that correspond directly to queries, views, and joins of the production data. For example, if a query for all records with the first name "John" and the last name "Brown" returned 20 records against production data, a query for all records with the first name “Lester” and the last name “Camden” must return 20 records against anonymized data. Failure to honor this corollary of the meaning and meaningfulness requirement can result in major problems when using the data for some types of functional tests, as well as any kind of performance, reliability, or load test.

Even more challenging is the matter of usefulness of the data for interoperability testing. Consider three applications, each of which have data gathered over years and describing the same population. The data reside in three different databases. The three applications interoperate, sharing data, and data-warehousing and analytical applications can access the related data across databases. These applications can create a logical record for a single person through a de facto join via de facto foreign keys, such as full name, Social Security number, and so forth.

If the anonymization process scrambles the data in such a way that these integrity constraints break, then the usefulness of the anonymized data for interoperability testing breaks. Meaningful end-to-end testing of functionality, performance, throughput, reliability, localization, and security becomes impossible in this situation. For many of our clients, preserving the usefulness of the data for interoperability testing poses the hardest challenge.

In addition, the anonymization process must not change the overall data quality of the scrambled data. This is subtle, because most production data contains a large number of errors. Some have estimated the error rate as high as one in four records. So, to preserve the fidelity of the test data with respect to production data, the same records that have errors must continue to contain errors. These errors must be similar to the original errors, but must not allow reverse engineering of the original errors.

A good test-data set has the property of maintainability, and so the anonymized data must also. Maintainability of test data means the ability to edit, add, and delete the data. This includes at the level of individual data fields and records, and, if applicable, across the logical records that might span multiple databases. To have the property of maintainability, the anonymization of the production data should not make maintenance of the data impossible, of course, but furthermore it should not make maintenance of the data any more difficult or time-consuming than maintenance of the production data.

Two other practical challenges arise with the process of anonymization itself. The first is the time and effort required to carry out the anonymization process. One client told an RBCS consultant that they only refreshed their test data from production every 12 to 18 months, because the test-data refresh process, including the anonymization, required 4 to 6 person-months of effort and typically took an entire month to complete. In an organization where staff must charge the time spent on tasks to a particular project, few project managers felt compelled to absorb such a cost into their budgets.

The next practical challenge of anonymization relates to the need to operate on quiescent data. In other words, the data cannot change during extraction of the to-be-anonymized data. This is nothing more complex than the usual challenge of backing up databases, but the people involved in producing the anonymized test data must be aware of it.

Options for production-data anonymization include both commercial and custom-developed tools. The selection of a data anonymization tool is like the selection of any other test tool. One must assemble a team, determine the tool options, identify risks and constraints for the project, evaluate and select the tool, and then roll out the tool. In this case, these activities would typically happen in the context of a larger project focused on creating test data entirely or in part through the anonymization of production data. Our experience with RBCS clients has shown that such a project requires careful planning, including identification of all requirements for the anonymized data and the anonymization process. An organization planning such a project should anticipate investing a substantial amount of time and perhaps even money (should commercial tools prove desirable). Trying to do a production-data anonymization project on the cheap is likely to result in failure to overcome many of the challenges discussed here. However, with careful planning and execution, it is possible for an organization to use anonymized production data for testing purposes.