The Client

Our client is the investment advisory firm, managing assets for institutional and private clients worldwide valued at $42.4 billion in assets and $34.2 billion in Institutional/Private Client Assets.

The Challenge

Our client generated all financial information and reports via manual calculations delivering their final reports in Excel format. This did not satisfy customer needs and they threatened to take their business to competitors where they could get reports in industry standard format. The challenge: Implement a Financial Reporting Solution within a year. To achieve this, our client needed to build an infrastructure and a data warehouse that would store data to generate ad hoc and canned reports in a timely manner. But, the biggest challenge was how to validate the reports and ensure that all the calculations were accurate. The solution was to find a Testing Partner with extensive experience validating financial reports. This is exactly what RentTesters offered, experienced Financial Reporting QA Engineers with a strong financial background and the technical skills to deliver a high-quality bullet-proof reporting solution.

The Solution

The RentTesters team began by assessing the work and offered a Free Rapid Assessment to understand the scope of work, schedule, and the resourcing needs. We assembled a team of eight: one QA Lead and seven Senior QA Engineers. Our QA lead was responsible for the overall Test strategy, Test Planning, Daily Status reporting, and day-to-day team management.

Our team assessed reporting requirements, data sources, and data target. We also reviewed source to target maps and came up with 800+ test cases that focused on ensuring:
Data Completeness – ensuring all expected data is loaded.

- Compare record counts between source data and the data loaded to the warehouse as well as rejected records.
- Compare unique values of key fields between source data and data loaded to the warehouse. This is a valuable technique that points out a variety of possible data errors without doing a full validation on all fields.
- Utilize a data profiling tool that shows the range and value distributions of fields in a data set. This can be used during testing and in production to compare source and target data sets and point out any data anomalies from source systems that may be missed even when the data movement is correct.
- Populate the full contents of each field to validate that no truncation occurs at any step in the process. For example, if the source data field is a string (30) make sure to test it with 30 characters.
- Test the boundaries of each field to find any database limitations. For example, for a decimal (3) field include values of -99 and 999, and for date fields include the entire range of dates expected. Depending on the type of database and how it is indexed, it is possible that the range of values the database accepts is too small.

Data Transformation - ensuring all data is transformed correctly according to business rules and/or design specifications.

- Create a spreadsheet of scenarios of input data and expected results and validate these with the business customer. This is a good requirements elicitation exercise during design and can also be used during testing.
- Create test data that includes all scenarios. Elicit the help of an ETL developer to automate the process of populating data sets with the scenario spreadsheet to allow for flexibility because scenarios will change.
- Utilize data profiling results to compare range and distribution of values in each field between source and target data.
- Validate correct processing of ETL-generated fields such as surrogate keys.
- Validate that data types in the warehouse are as specified in the design and/or the data model.
- Set up data scenarios that test referential integrity between tables. For example, what happens when the data contains foreign key values not in the parent table?
- Validate parent-to-child relationships in the data. Set up data scenarios that test how orphaned child records are handled.

Data Quality – Ensuring ETL application correctly rejects, substitutes default values, corrects or ignores and reports invalid data.

- Reject the record if a certain decimal field has nonnumeric data.
- Substitute null if a certain decimal field has nonnumeric data.
- Validate and correct the state field if necessary based on the ZIP code.
- Compare product code to values in a lookup table, and if there is no match load anyway but report to users.
Case Study: Data Warehouse & Report Testing
Investment Banking

Performance & Scalability - Ensuring data loads and queries perform within expected time frames and that the technical architecture is scalable.

- Load the database with peak expected production volumes to ensure that this volume of data can be loaded by the ETL process within the agreed-upon window.
- Compare these ETL loading times to loads performed with a smaller amount of data to anticipate scalability issues. Compare the ETL processing times component by component to point out any areas of weakness.
- Monitor the timing of the reject process and consider how large volumes of rejected data will be handled.
- Perform simple and multiple join queries to validate query performance on large database volumes. Work with business users to develop sample queries and acceptable performance criteria for each query.

Reporting UI Testing Verify Reports User Interface

- Ensure that the generated report is in readable format, and verifies all labels, headers and footers are in place.

Data Calculations

- Ensure that all data in reports is calculated according to the formulas specified in a requirements document.

Integration Testing

- Ensures that the ETL process functions well with other upstream and downstream processes.

Compatibility Testing

- Ensures compatibility on different browsers & versions and Operating Systems.

User-acceptance Testing

- Ensures the solution meets users’ current expectations and anticipates their future expectations.

Regression Testing

- Ensures existing functionality remains intact each time a new release of code is completed.
Case Study: Data Warehouse & Report Testing
Investment Banking

The Benefit

- High quality data warehouse and reporting solution helped Client to retain their customers and acquire new customers.
- Achieved optimal performance for complex reports.
- Great user experience.
- Meet project deadlines.

Customer Quote
“Your robust testing strategy and experienced QA team helped our firm achieve its quality goals.”
---CTO, San Diego based Investment firm

For more on how RentTesters can help your company improve testing processes, check us out online at www.renttesters.com or call 1.888.789.1482.