

SIEMENS

Ingenuity for life

Big data testing

This paper describes the emergence of big data, discusses the main challenges of testing big data and examines the threshold for managing large quantities of test data with existing tools and resources.

Contents

Big data.....	3
The Excel dilemma.....	5
A solution: big data and the future of spreadsheets	6
Why use a commercial test management solution?	7
Polarion QA – The collaborative test management solution for big data	8
Conclusion	9

Big data

Big data is a big topic these days, one that has made its way up to the executive level. Most organizations may not yet fully understand what big data is, exactly, but they know they need a plan for managing it.

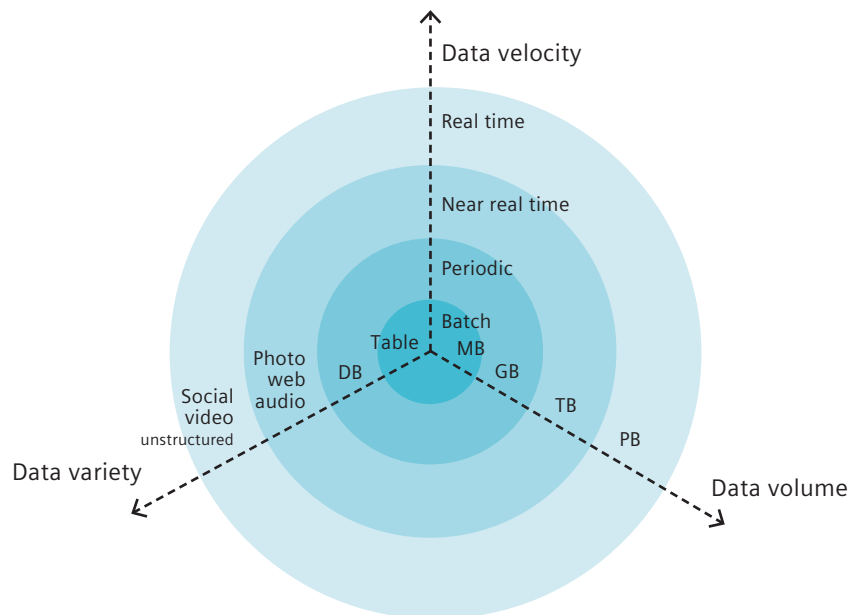
The accepted definition for big data talks about exploiting “data sets whose size is beyond the ability of commonly used tools to process it within acceptable time.” Big data is one of the most discussed topics in recent times. The volume of data we’re handling is growing exponentially, with the popularity of social media and the ever-growing internet of things, the mass of data produced by smart electric grids, intelligent traffic systems and other factors. According to IBM, “Every day, we create 2.5 quintillion bytes of data – so much that 90 percent of the data in the world today has been created in the last two years alone.”

As described by the Gartner information technology research and advisory company, big data is characterized by three Vs: volume, velocity, and variety. All are growing dramatically and we’re being asked to process data ever more quickly so we can respond to events as they happen, and data is coming from an ever wider array of channels, sensors and formats.

Data volume

The new norm is that more sources of data are added on a continuous basis. In the past, all company data was generated internally by employees. Currently, the data is generated by employees, partners and customers. For a specific group of companies, the data is also generated by machines; examples include data from hundreds of millions of smartphones and sensor readings from factories, pipelines and other sources. This data did not exist five years ago, and the result is that more sources of data with a larger volume of data combine to increase the amount of data that needs to be analyzed and tested. This is a major issue for those looking to put that data to use instead of just letting it disappear. Petabyte data sets are common these days, and exabyte is not far away.

Big data: expanding on three fronts at an increasing rate



Data source: Gartner

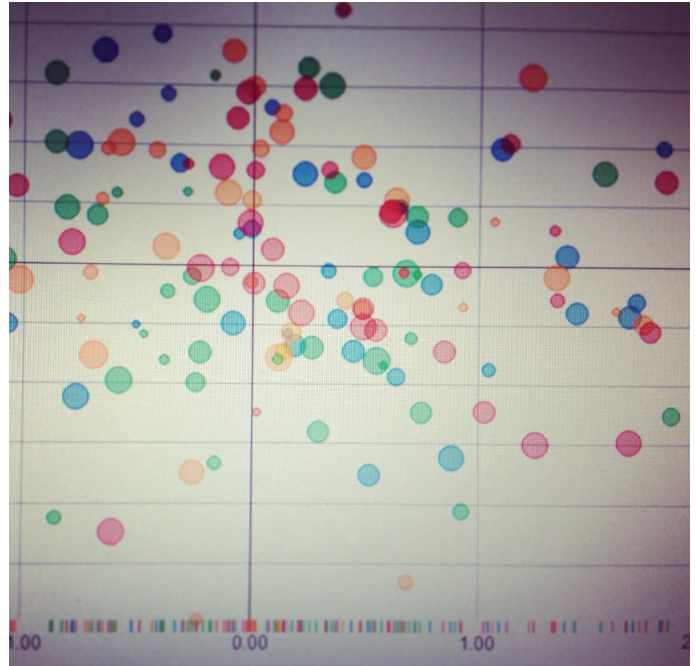
Data velocity

Big data can be described by its velocity or speed. Or you may think of it as the frequency of data generation or frequency of data delivery. For example, think of the stream of data coming from any kind of sensor (for example, thermometers sensing temperature, microphones listening for movement in a secure area or video cameras scanning for a specific face in a crowd). This isn't new; many firms have been collecting clickstream data from web sites for years and using the data to make purchase recommendations to web visitors. With sensor and web data flying relentlessly in real time, data volumes get large quickly. Even more challenging, the analytics that go with streaming data must make sense of the data and possibly take action – all in real time.

Data variety

Ranging from spreadsheet tables to databases, data structure has evolved to hundreds of different formats. We now deal with pure text, photo, audio, video, web, global positioning data, sensor data, relational data bases, documents, simple messaging system (SMS), PDF, Flash and other formats. The challenge is compounded by the fact that testers have limited or no control over the input data formats. Structure can no longer be imposed to keep control over the analysis. As new applications are introduced, new data formats come to life and must be dealt with during the software testing process.

This paper discusses the main challenges of testing big data and examines the threshold for managing large quantities of test data with existing tools and resources such as Microsoft® Excel®.



The Excel dilemma

Microsoft Excel is one of the flagship applications of the Microsoft Office® suite and by far the most popular tool used in software testing.

According to Naysawn Naderi (a program manager at Microsoft who played an integral role in defining and developing Microsoft Test Manager 2010), Excel, a tool developed in the 1990s, continues to be the most popular testing tool in the industry for manual testing. While some manual testers simply jot down their test cases on a piece of paper, many prefer Excel, since it can be re-used, is digital, and can be shared. However, it provides no customized or optimized testing experience. "It amazes me that testers live with it, as it is roughly equivalent to having a developer code using Microsoft Word," says Naderi.

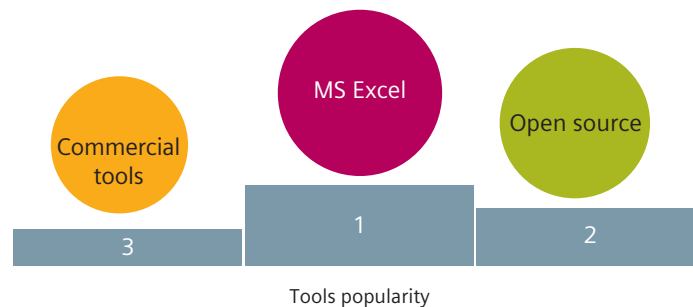
While many still use Excel to manage tests, it clearly leaves much to be desired. Its limitations include taking up a full window while testing, no options for collaborating effectively with the development team, no easy way of sharing what was tested, no ability of sharing steps between test cases and no linkage to a defect tracking or requirements system.

Typically, testers use Excel spreadsheets (or, less typically, Word tables) to record testing steps, expected results and pass/fail state in the required timeframes. They compile the results of testing either manually or with scripted programs that import the Excel spreadsheets or Word tables, process the test results, and then produce somewhat usable reports. However, anyone attempting to use Excel spreadsheets or Word tables to manage manual test scripts will run into these problems:

- The need to constantly scroll horizontally and vertically makes these methods inefficient and difficult to use
- The typically wide formats of Excel spreadsheets or Word tables and the need to reproduce headings on each page make it difficult to print reports
- It is difficult to organize scripts by grouping testing steps
- There is no standard way to identify expected results within test scripts
- There is no standard way to report test results
- It is difficult or impossible to re-use test script lines or groups of lines in different test scripts
- Spreadsheets or tables are not easily shared across multiple locations
- With row and cell limits, spreadsheets or tables are not scalable
- Limited data validation
- Minimal reporting capability

There are many shortcomings that make Excel a substandard test management software tool, and the characteristics of big data make Excel virtually unusable. It is true that Excel has been used on big data for years, but not directly on big data. With a row limit of around 1 million, Excel certainly cannot be used as a direct window into continuous volumes of data. Alternatively, test analysts use Excel for front-end views or other subsets of the original data set that have been heavily cleansed and normalized.

A tool that solves these problems would enable test departments to take full advantage of their enlightened decision to incorporate manual test scripts into the application development process.



A solution: big data and the future of spreadsheets

First, software engineering teams need to adopt new technologies and to learn – or hire – new skill sets to work with data at this scale. New roles such as data scientists and data technologists are arising to meet these challenges. Training and cross-discipline collaboration can also help to raise everyone's data IQ.

A software solution should solve the needs of manual testers who must manage large amounts of data, are typically short of time, work as a part of a team, collaborate extensively with developers and are passionate about application quality. To begin, testers need a tool that just allows them to test manually. Specifically, they need a tool that enables them to perform exploratory testing, author manual test cases, run them while focusing on the application under test, fast forward through the uninteresting parts, file defects, share test results with their team and validate that bugs are fixed without having to worry about managing large volumes of test data and various data formats.

Second, since manual testers and developers work collaboratively in the development process, each requires an integrated toolset that provides visibility into each other's activities and allows them to work efficiently together.

Third, manual testers need a tool that will provide them with a workflow of going from a manual test to an automated test to a test that is run as a part of a build process, with minimal to no interruption.

Why use a commercial test management solution?

A great test management application will reduce the cost of managing big data by providing structure in the form of test cases. Many organizations have their test scripts scattered across multiple documents and formats. This quickly becomes cumbersome when trying to track test execution, and reporting metrics around test passes turns into a painful exercise of tallying results from multiple sources.

Reduce cost of managing test suites by mitigating the pain of managing test cases

A quality test management system should make it easy to import test cases stored in Excel and create structured test cases. Once those cases are imported it should be easy to group those tests into lists, and execute and track the results of test list runs, thereby eliminating the consolidation of stats from multiple Excel sheets.

Reduce time for executing your test

Manual test scripts require a lot of time for setup and configuration before the true testing begins. In many test environments it is common for test scripts to be written, modified and run on an ad hoc basis, where no thought is given to formal approval and release for the test scripts that need to be run. It's not uncommon to spend 10 to 20 minutes doing setup work for a test that takes only a few moments to complete. Advanced testers will often cobble together utilities and small scripts to perform these setup actions. In addition, scripting can be more complex and time-consuming when dealing with larger and more frequent data structure changes. Ideally, a test management solution should simplify and reduce the time and complexity of creating and managing test scripts.

Polarion QA – The collaborative test management solution for big data

You can manage and aggregate all of your testing activities from one central quality management platform. Most QA departments already use multiple testing tools, work with defined processes, and do not want to change their processes because of the introduction of a new tool.

Siemens PLM Software's Polarion QA™ software provides you with all the benefits of test management software and minimal interruption to your existing environment while reducing cost and improving the quality of your software. With Polarion QA it is easy to create test cases and to link them back to your software requirements without disrupting your current workflow.

Excel round-trip

Recognizing the fact that most organizations continue to use Excel in their testing environments, Polarion QA offers the most robust integration that enables testers to easily import existing Excel-based test data to quickly bring those tests into Polarion QA where they can edit, track and execute them with ease, and with none of the limitations of Excel. Bringing manual tests into Polarion QA also improves management of manual tests with version control.

Continue to use Excel when required

Companies can also leverage their existing environments with document round-trip capabilities of Polarion. Changes made outside of Polarion QA can be imported back seamlessly while preserving the original formatting. All tests can still be executed via Excel while taking advantage of Polarion QA's advanced features, including detailed traceability from software requirements to defects, forensic-level auditing and customizable reporting.

Maximize actual testing time with workflow automation

Polarion QA's workflow engine and template creator help to quickly accomplish the same tasks within one consolidated environment. If an existing test script creates the setup, you can bring it into the manual test directly. You can also write code to handle database initialization or system configuration.

Increase velocity while reducing risk

The challenges of big data management result from the expansion of all three properties, rather than just volume alone – however, the sheer amount of data to be managed becomes a major issue. Ironically, as the quantity of test data increases, the expected rate of testing such data increases as well. With increased velocity, Polarion QA can mitigate risk while accelerating the quantity of data consumed during the testing process. For example, when a tester creates a bug, Polarion QA can make it extremely rich (with video information, a readable log of the actions that took place in the application under test, links to the test case that caused it, and audit logs), and can present it to developers in an actionable way, where they can immediately jump to the line of code that failed. If developers are presented with bugs that are irrefutable to reject, they will fix them, and the state of application quality will improve. Further, when the developers fix the bug, they may change some of the lines of code, which will cause some tests that previously passed to fail. Using Polarion QA, you can track the lines of code your tests cover and recommend that testers rerun the test cases that cover the code that has changed from build to build.

Conclusion

Big testing gets its bigness from three distinct things:

1. A willingness to experiment with big, bold ideas – to try something that may fail. Risk is mitigated by testing such ideas on a small scale, not by avoiding daring new concepts.
2. The pursuit of experimentation is championed by management – it's a big deal, an integral part of the culture. High failure rates of experiments, as long as they are run well, are not treated as failures of the individuals running them. On the contrary, aggressive testing is recognized and rewarded.
3. A large number of people throughout the organization are empowered to run experiments in their work. This is what gives big testing its scale, allowing a large number of experiments to be run across different facets of the company.

Conventional thinking and software tools clearly do not meet the requirements of being able to properly manage and test big data. For almost 20 years, Microsoft Excel has been the de facto test application. It cannot be replaced entirely. The best strategy is to reframe the problem and solution, and provide true test management software that will exceed the limitations of Excel.

Big data is good. It's an amazing source of new hypotheses for marketing. But to truly unlock the value from big data, companies must embrace big data testing.

Siemens PLM Software

Headquarters

Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

Americas

Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8499

Europe

Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

Asia-Pacific

Suites 4301-4302, 43/F
AIA Kowloon Tower,
Landmark East
100 How Ming Street
Kwun Tong, Kowloon
Hong Kong
+852 2230 3308

About Siemens PLM Software

Siemens PLM Software, a business unit of the Siemens Digital Factory Division, is a leading global provider of product lifecycle management (PLM) and manufacturing operations management (MOM) software, systems and services with over 15 million licensed seats and more than 140,000 customers worldwide. Headquartered in Plano, Texas, Siemens PLM Software works collaboratively with its customers to provide industry software solutions that help companies everywhere achieve a sustainable competitive advantage by making real the innovations that matter. For more information on Siemens PLM Software products and services, visit www.siemens.com/plm.

www.siemens.com/plm

© 2016 Siemens Product Lifecycle Management Software Inc. Siemens and the Siemens logo are registered trademarks of Siemens AG. Excel and Office are trademarks or registered trademarks of Microsoft Corporation. ALM, D-Cubed, Femap, Fibersim, Geolus, GO PLM, I-deas, Insight, JT, NX, Parasolid, Polarion, Solid Edge, Syncrofit, Teamcenter and Tecnomatix are trademarks or registered trademarks of Siemens Product Lifecycle Management Software Inc. or its subsidiaries in the United States and in other countries. Other logos, trademarks, registered trademarks or service marks belong to their respective holders.

55671-A4 8/16 F