Benefits of BenderRBT Test Automation

Document Version 6.0 6/1/16

Capability (Who, When & What)	Value (So)	Feature(s)
1 – Rigorous Test Case Design a) QA can learn to detect ambiguous specifications, incomplete specifications, and logically inconsistent specifications long before code is generated. b) QA can learn to create a more consistent level of high quality testing independent of the overall experience level of the tester. c) Testers can generate mathematically equivalent test cases to the set of functionality in all of the specifications. d) QA can receive assistance from RBT produced test cases that can factor in potential defects that can cancel each other out resulting in getting the right answer for the wrong reason. e) QA can view a comprehensive logic diagram to identify where diagnostic probe points should be inserted in the code for more accurate testing f) Provides QA with an optimized test library with the opportunity to identify twice the coverage for half the test cases. g) QA to receive information from RBT and its ability to identify generated test cases that will cover 80 to 90% of the code significantly shortening the time to achieve the 100% statement of branch coverage. h) QA can receive a set of negative tests as well as the full set of "go right" tests. i) Consistent test cases are available in describing the various inputs and outputs of the test (i.e., if the same variable at the same state is part of the inputs/outputs of more than one test, it will always be described by RBT in exactly the same way). j) QA can learn a cause-effect graphing process that helps identify system integration issues before design completion. k) Flexible test design approaches with addition of Quick Design l) Full constraint processing m) QA can determine what changes need to be made to existing tests when the application's rules change. n) Testers can dynamically see the logic flow by manually setting states in the cause-effect graph and having the states, forwards and backwards, extrapolated.	a) Reduce the cost and effort to resolve and reduces the amount of scrap and rework on the project. b) No matter how many testers you have, they all are going to come back with the same test case. c) Provides better wording of the test descriptions than wording in original specifications. c) Allows the test cases to be reviewed by the users/domain experts to validate the system or application rules and find defects in specifications by reviewing the test cases. c) Moves a major step of user acceptance earlier in the process before the code is written. d) Address defects that might be hidden because they are not directly observable. e) Increases code coverage, and decreases the likelihood of hidden errors canceling each other out. f) Allows companies to achieve major savings in the effort to script/implement test cases as well as reducing the time to run test cases, validate test cases, and maintain the test library. f) With one fourth the tests for equivalent coverage, this results in over a ten X reduction in the cost and effort to build executable scripts. g) Ensures that code coverage is 2 to 3 times better at delivery of the application. Most organizations only achieve 30% to 35%. For organizations that must meet test coverage criteria set by various government agencies (e.g. FAA, FDA, DOD). h) Helps ensure the system maintains data and transaction integrity even under adverse conditions. i) Avoids confusion in exactly how to set-up the test and what the results should be. Those not using RBT to design tests often use subtly or even significantly different wording to describe the same condition. j) Forces the tester to ask questions of how individual functions work and how those functions should work in relation to other functions. j) Numerous defects are detected which are not normally found until system integration testing. k) Configuration test design l) Ensures elimination of impossible test combinations while still supporting full negative testing. m) Protects the investment in the existing test cases.	• Logical Consistency Check • Cause-Effect Graphing test case design engine • Orthogonal Pairs design engine with Optimize option • Full constraint support including Exclusive, Inclusive, One and Only One, Requires, and Masks. • Evaluate Old Tests and Supplement Tests in the Run Tests Design option. • Neoning allows the tester to manually set the states in the Cause-Effect Graph and see the extrapolated states. Neoning also allows the tester easily determine why a functional variation is infeasible.

Benefits of BenderRBT Test Automation

Capability (Who, When & What)	Value (So)	Feature(s)
2 – Test Coverage Measurement a) QA can quantify the functional test coverage based on what test cases passed and failed. b) A subset of tests can be identified with maximum coverage.	a) Provides management with quantitative measure of the quality of the application. Go / no go decisions are made on a quantitative basis function by function. b) Allows testers to decide which test to implement first. c) Optimizes test coverage when time is limited.	• Coverage analysis utility • Less Tests feature
3 – Test Exporting a) QA can export the test descriptions to test management tools. b) QA can export the test descriptions to most of the major playback tools in the form of comments in shell scripts. This helps create self- documenting tests saving scripting time. c) QA can export the test data to Excel via a comma delimited file	a) Saves redundant data entry because it is described once in RBT and that description is replicated across all the RBT test cases. b) Exported tests to playback tools and test library managers, saving an enormous amount of redundant entry. c) Reduce rework time by ensuring immediate notification of requirement changes, keeping all stakeholders up to date on the project. d) Create self-documenting tests which saves scripting time. e) Allows easy integration with non-standard tools f) Allows flexible integration of test information in other documents	• Exports to: ➢ Quality Center ➢ TOSCA ➢ Generic Export via the OMG's TestIF format ➢ Generic Export via comma delimited files. ➢ Generic Export via rich text format ➢ Generic Export via XML
4 – Test Document Generation a) QA can generate ancillary documentation information that may be necessary which saves testers a great deal of time. b) QA can generate test cases in various formats in batch mode, script mode, and DOD 498 format.	a) Saves testers time. a) Produces test definition matrix and test coverage matrix with one mouse click instead of hours spent building an Excel spreadsheet. b) Ensures that the tests are in the format most useful to needs of the project. Switching between formats only requires one mouse click. This is especially useful when doing projects for the federal government since they require formats necessitating a lot of work to create and maintain.	• Coverage matrix • Test definition matrix • Logic diagram • Statistic report • Test in batch or script format • Function variation list

Capability (Who, When & What)	Value (So)	Feature(s)
5 – Test Coverage Analysis a) Stakeholders can protect the investment in test cases already built, whether designed by RBT in a prior release or done independently of RBT. b) Testers can run "What if "scenarios for functional coverage that can be used to determine the optimal subset of test cases to build.	a) Analyzes how much the existing set of test cases cover the rules of a new or revised function. a) Identifies what additional test cases are now needed, what modifications need to be done to existing test cases, and which test cases are now redundant. a) Allows testers to maintain an optimal test library while minimizing rework of tests already built. b) Ability to prioritize the testing effort b) By focusing on the tests which give the greatest coverage first, a reasonably stable version of a given component can be created for use in early entry into system level integration test. b) Assist when projects find tight time or resource constraints limit how many tests can be implemented and run.	• Evaluate old tests • Design supplemental test • Coverage analysis utility • Less Tests feature
6 – Requirements Management Integration a) Relationships between system specifications and test specifications are available. b) Users can view RBT test cases from a requirements management tool. c) Users can view the status of the most current successful functional test coverage.	a) Identifies impacted tests when a specification changes. b) The test cases are often more detailed and easier to read than the original specifications. Users use this view to ensure that what is being built matches their expectations. b) Provides mechanism to move user acceptance testing earlier in the life cycle long before code has been written. b) Users do not have to be familiar with RBT as they are viewing outputs produced by it. c) Ability for users to make informed decisions as to whether the application is ready to deploy c) Users do not have to be familiar with RBT as they are viewing outputs produced by it.	• Generates a Functional Specification from the Cause-Effect Graph model which can be exported in rich text format.

Capability (Who, When & What)	Value (So)	Feature(s)
7 – Requirements Based Testing Process a) QA can learn a full process for testing the specifications. b) Stakeholders can be involved in Ambiguity Review training that can help in uncovering numerous classes of defects in the specifications at reading pace. c) QA can learn and use Cause-Effect Graphing steps that when applied by the testers can help identify more subtle issues with the specifications.	a) Minimize user ramp-up time by providing quick and comprehensive training. a) Ensure that they are correct, complete, unambiguous, and logically consistent. b) It is a useful skill for users and developers in addition to the testing staff. Staff can be trained in less than a day to apply this step. c) Uncovers unclear rules for the sequencing of the steps in a system (i.e., unclear precedence relations). c) Uncovers problems with aliases in the naming of data and processes – an object being referred to by more than one name across different sections in the specifications or two or more different objects being referred to by the same name.	• Bender Learning Institute • RBT Quick Start • RBT-SP (service provider)