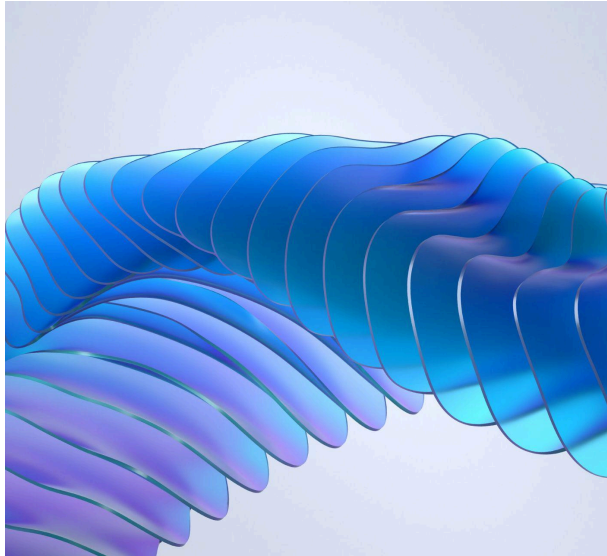


ReplicTest

IBM i Automated
End to End
Testing tool



Simplicity and efficiency are the predominant factors of this "engine" Testing tool for your IBM i applications



Native green screen 5250

IO + Navigation Encapsulation
Plug-in for IBM i Access Client Solutions (ACS)
Test speeded up to 200 x faster vs manually



Web Services

IO Encapsulation with initialisation and comparison
(GET, POST)
REST API, microservices



Application's DB tables

Automated DB Tables Encapsulation with reset & comparison data for all test environments (5250, web, batch)
Integrates RLA, SQL, ODBC, JDBC



Code Coverage

Details lines and % of code covered by single test case or merged lines and % by multiple tests
(Merging code coverage and having information dynamically is essential to validate quality ranks and govern flow processes)



Call pgm IO parm

IO parm Encapsulation with parameters
initialisation and comparison



Shrink Data

Creation of data subset with only the records effectively used by the test case.
(Enable & Increase tests replay frequency with a considerable reduction in turnaround times)

Enable & Automate your testing

Institutionalize quality and Save more than 88% of your time

ACCELERATE TIME TO MARKET

Full DataBase integration

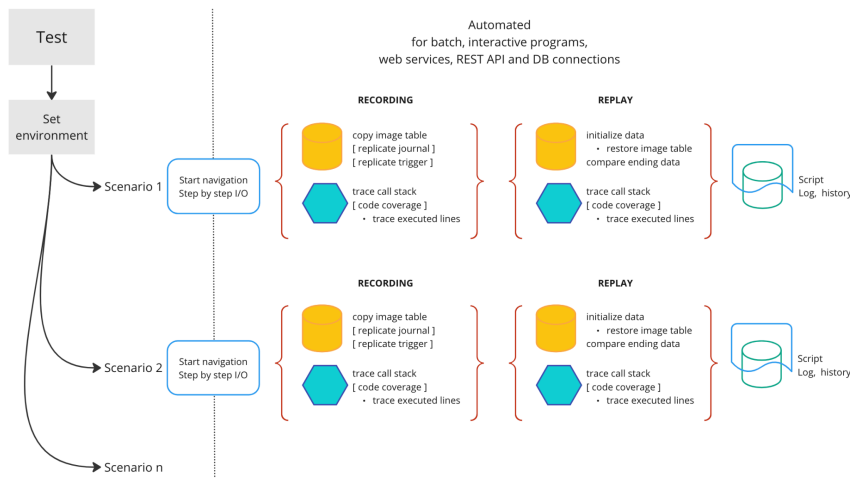
DB tables and files identification for images, restore and comparison are automatically handled.

Tests need to be replicable, which means that the data must be reset each time a test is replayed and many test tools, like ours, include this functionality. But the prerequisite is that all data files or tables are identified first. This prerequisite is often left to the end user or some tools just copy entire libraries with all files and tables.

ReplicTest does this automatically on the fly and only for the files and tables actually used by the specific test case. This is optimizing all processes in your IBM i (aka AS400), from prerequisite recognition to execution, doing and automating all the hard tasks for you.

Database subset for your tests

ReplicTest can validate data during tests and can be configured to shrink your database to only the records required for test cases. You can automate the cleanup or shrinking of test data, which is critical for keeping your test environments light and efficient, especially when executing frequent tests. This is a unique function that will save significant storage and can be deployed also for live data storage.



Automated for batch, web or interactive programs

The same command is used to process recording or replay of the test case. By default, the command automatically detects in which modality it should run; recording or playback and depending on the modality, different behaviors are processed to copy, restore or compare the image data.

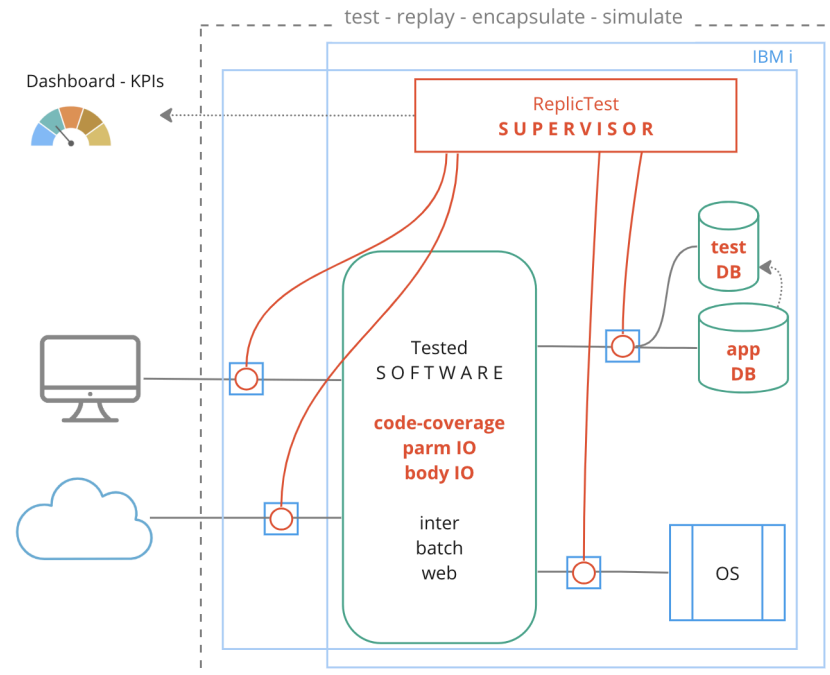
Holistic

Comprehensive testing in a holistic manner with our Test Encapsulation Engine, inter/batch/web (encapsulating DB, UI, IO parm, body parm and code coverage – with restorations and comparisons)

Our “Encapsulation Engine” is “omnivorous”: It integrates any languages, legacy or modern, RPG, CBL, CL, SQL, Java, PHP, .net, node, python etc... It can also integrate any objects like spoolfiles dtaara, dtaq, IFS objects, the joblog, etc... and more, when you change an object in your system, you will know which tests involved it and relaunch all of them.

Invokable as a wrapper to single test or test suites (series of tests). Autonomous or pluggable to DevOps pipelines, ACS, SQL Scripts, VS Code, RDj, Cypress or Selenium, PEX, iDoctor.

Remote tests from other platforms can also be launched & controlled with ReplicTest’s scripts and be monitored in the framework and dashboard.



Ally with AI on IBM i!

AI to leverage ReplicTest

Metadata and KPIs from ReplicTest can serve AI to:

Set Test Prioritization and Optimization.

Change test detection (When UI change, find patterns & suggest changes accordingly)

Predictive Defect Detection using previous test logs & statistics.

ReplicTest to Test AI

ReplicTest’s ability to ~truly~ automate testing and collect data across different layers of IBM i applications can support AI reinforcement learning models by providing a consistent environment, diverse data, and reliable feedback, in a continuous process.

Regression & workload testing, deployment and monitoring

Tests can be scheduled automatically, in sequence or in parallel, with same or different login

Command driven, tests can be launched automatically via different options, through other commands or via scripts in modular ways.

Tests status are logged including call stacks. In case of failure, monitoring configuration can leave the job open, or end it (and go/no go to a next test), logging all relevant messages and contextual metadata. Switch login to test different user authority access can be truly undertaken. (ReplicTest integrates all objects ownership and authority in replay mode as well).

User experience, KPIs and improvement

The user experience has been valued, for instance with the flexibility of correcting Recordings already done. Imagine you recorded a test case involving 30 screens and you realize you've entered the wrong data on screen 25, the tool enables the flexibility to correct it without having to redo the entire recording.

The same goes for web services and the integration of body messages. Another example is with timestamps that can automatically be bypassed for test replay comparisons or as in the case of code coverage that can be retrieved dynamically through an SQL UDF and also in an automated test driven workflow.

Tasks can be divided between End-User and Programmer. A non-programmeur end user can easily record a test case then a programmer can integrate it and deploy it.

Reporting can integrate a lot of information, from test status trends to message log with line statements, from code coverage percentage to time gained. All item information is available in a comprehensive database from which you can customize test reporting.

At last, we can also obfuscate datasets and present results in a dashboard.



Reporting, KPIs



Correcting script
Correcting recording
Correcting recording with data initialization
Splitting recording/replay script
Merging recording/replay script



Merging code coverage
• Lines of code executed

“Often, people ask what a realistic metric to measure design impact could be, and I’m convinced that metric is “time to value.” If your design influence isn’t decreasing overall time to value, you’re doing something wrong.”

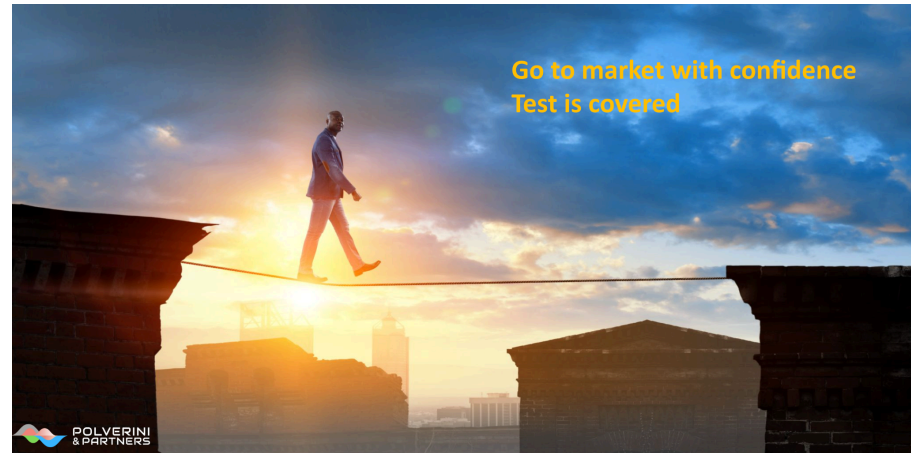
*Juan J. Ramirez
Product Design Lead @Netflix*

Integrate KPIs, Metrics and Trends into Your Software Development Lifecycle

Case Study

The tool enables a testing governance of high accuracy in correlation with impact analysis of lines of code to be changed for specific projects like, for example Field Resize.

It is also totally integrable with our DevOps solution.



"The innovation of ReplicTest is remarkable because it is removing what represented a roadblock for many IBMi shops to achieve testing. For the quality of the code, the consistency of the data and significant time savings."

Birgitta Hauser - IBM i Champion

"Besides the traditional automated regression testing using ReplicTest sql-scripting feature to invoke app processes, we also use ReplicTest in our refactoring project. Refactoring is all about testing; making small changes, testing, additional changes, again testing etc... Using the metadata ReplicTest makes available developers can compare the before and after situation easily. All database access is recorded including the call stack at that event. By analyzing the data provided we are able to trace any overhead in the process. "Why are we querying this table?". Together with the Code-Coverage feature we were able to remove obsolete code and make our application much faster.

About our product: ECI's EasyOrder is an IBM i web based e-commerce and order management solution and is used in a variety of industries. It covers the entire supply chain and the solution involves a variety of electronic procurement systems to purchasing systems with many customizations."

Rick Bovet - Development manager at ECI Software Solutions

Investment & ROI

Programmers spend from 20% up to 50% of their daily time on testing

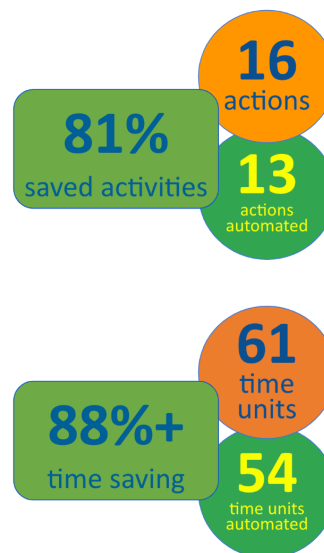
88%+ time saved on effective testing

88%+ gained time for production development

Automate & institutionalize quality integration

(* Following the math you will save from 88% of 20% to 88% of 50%, which makes an average of 30.8% of your programmers time. But how would you quantify the value of the confidence and quality gained?)

Steps to take for TESTING	Actions	Time taken with manual execution (time-units: 0-5)	Time taken with ReplicTest (time-units: 0-5)
Setting up the test case	Establish all test case actions in sequence (first time only)	3	3
Setting up the test environment	Login	1	0
	Identification of the data tables used for the test case	5	0
	Data initialization	4	0
	Save a copy of the data tables at the beginning and end of the test case	3	0
Execution of the test case	Perform the test the first time	2	2
	Perform the test <i>n</i> times	5	0
Test quality	Get code coverage	5	0
	Merge code coverage	5	0
Analysis of the result	Intercept input & output data	5	0
	Get tables data outcome	4	0
	Compare before/after I/O data	3	0
	Compare before/after tables data	3	0
Reporting	Create reports	5	2
	Record history	4	0
	Create KPIs	4	0



Functional testing

Ensure quality for all your app releases and accelerate TTV & TTM.



Workload testing

Ensure effective scalability with differentiated parallel testing.



Security testing

Ensure effective object or data protection from non authorized user access.
Encompass RCAC and system authority features.



Empowered code coverage

Full code coverage metadata and code coverage merging (generated directly in IFS and DB, no RDi dependency)



Create test variations with the same initial data

To get strong coverage you will often need to run the same test with different inputs. The tool integrates a feature to start a test with the same initial data and gives you back the control to continue with different inputs and create variations.



User friendly and efficient

End users (even non-programmers) can use ReplicTest to record test scenarios and a programmer can integrate them. One unique command to record and replay your test scenarios.



IDE or DevOps integration

The tool can be integrated with any IDE or DevOps structure. Command's driven and DB queryable. ReplicTest scripts can also be integrated with VS Code or SQL scripts.



www.polverinipartners.com

Solution Strategies – Digital Transformation – Cybersecurity

IBM i ISV Advisory Council member since 2005
OpenPOWER Foundation Associate since 2015
Modernization IBM Redbooks Author

IBM Champion Data & AI
Enterprise Design Thinking certified
Register of technological innovation experts - MIMIT