



Automated Test and Re-test (ATRT) Test Manager (TM)

Enabling Automated Software Testing for Heterogeneous Multi-Computer System Environments – also in the Cloud

ATRT OVERVIEW

■ Increase Productivity

Given the current challenges of declining budgets and decreasing schedules, ATRT will allow you to do more with less by increasing your team's productivity, i.e. based on our experience with large heterogeneous multicomputer system environments, using ATRT will increase your team's productivity by a factor of 5, allowing the manual testers to automate test procedures without having to be software developers themselves, for example. For your potential cost savings, please see our ROI calculator (based on actual programs) located on our www.idtus.com website

■ Scalable

ATRT has proven itself as a highly scalable solution whether you need to run 10 or 1000s of tests connecting to x number of displays and servers over days or weeks and whether you need to analyze 100s of

test outcomes or 1000s or more.

■ Support for Cloud Computing

ATRT can test systems independent of OS or platform and is one of the few automated testing solutions that will support your cloud computing environment.

■ Configurable and customizable – extensible

ATRT's eclipse-based plugin environment allows for ease of customization and configuration. Whether you need to write a plugin to your modeling tool, defect tracking tool or requirements management solution, etc., ATRT is easily extensible to allow for these additional integration features. Any desired tools enhancements and features are implemented on demand.

■ Secure and Trustworthy

ATRT has been installed and verified to meet the stringent security requirement of numerous Government programs.

MORE HIGHLIGHTS

- **Non-intrusive to the System under Test (SUT)** - doesn't modify the SUT configuration
- **Cross-platform and cross-OS compatible** – Web, Client Server, Handheld and Linux, Windows, Solaris
- **GUI Technology neutral** – independent of any GUI controls (custom or non-custom)
- **Scriptless Automated Testing** – no software development required
- **Keyword driven** - Simple clicks on specific test "action" icons generates code to add to the automated test baseline
- **Data driven** – allows the same test scenarios to be run repeatedly with different data sets (boundary testing; equivalence partitioning)
- **Model driven Automated Testing** – allows test flows to be generated via simple point and click model driven interface

Automated Test and Re-test (ATRT) Test Manager (TM)

Enabling Automated Software Testing for Heterogeneous Multi-Computer System Environments – also in the Cloud

IDT's ATRT solution addresses the complex testing challenges of software systems by providing an innovative technical solution that solves the unique testing problems of today's complex software systems. It is designed to support testing of a heterogeneous multi-computer environment while providing an integrated solution which includes a tool suite and engineering services that can be applied across the entire testing lifecycle.

Investment (ROI). Our approach will provide an opportunity to reduce the time and cost of software testing, improve software quality, and improve software test programs in measurable and significant ways over the testing life cycle.

How can my program benefit?
We specialize in providing solutions tailored to each customer's specific needs.

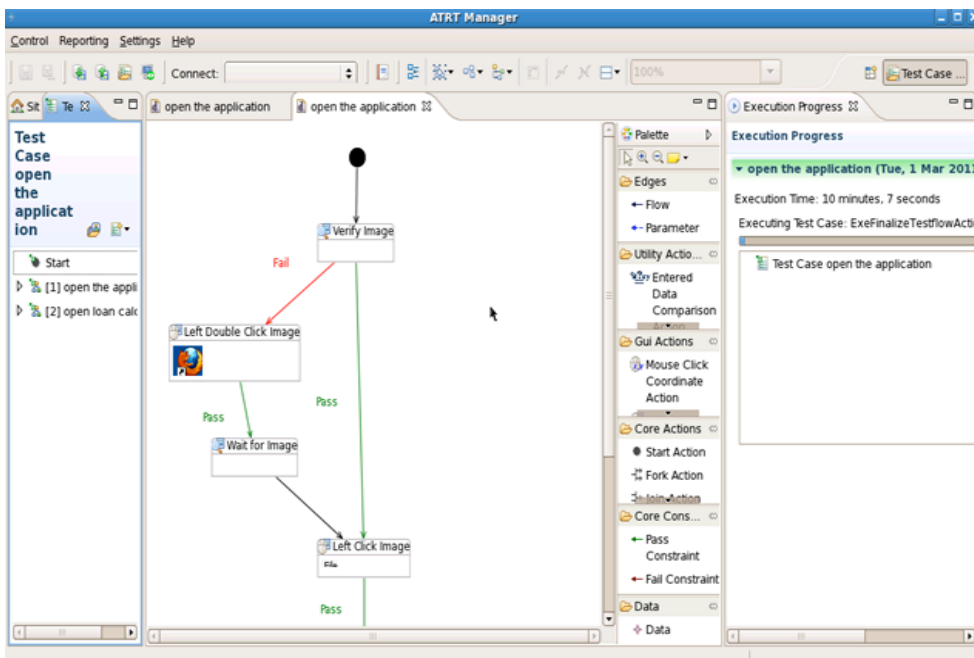
resources: automated tests can run unattended.

Increased Testing Coverage: Use of ATRT enhances manual testing efforts via increased testing coverage.

Automated test outcome analysis and reporting: With the increase of test coverage comes an increase of test results that needs to be analyzed. ATRT provides an automated reporting capability on the increased test coverage and outcome, comparing expected versus actual results.

Enables testers to be more effective: ATRT can replace many of the manually mundane and labor intensive tasks performed today. This can free up testers to focus on the more challenging tasks at hand.

Automates hard to accomplish manual tasks: ATRT allows testers to accomplish what manual testing hardly can; such as memory leak detection under specific conditions; concurrency testing, to uncover race conditions; endurance, longevity, performance testing, and more.



ATRT Test Manager

KEY BENEFITS

Experienced Workforce: Our business is automated software testing. IDT principals have over 25 years experience in large scale automated software testing. We institute a structured approach toward automated software testing that starts with creating the business case i.e. calculating Return on

Investment (ROI). ATRT provides proven technology to reduce the time and cost of software testing while expanding test coverage leading to improved software quality, among other major benefits.

Time and Cost Reduction: Tester resources are freed up when tests are automated. Regression testing, endurance and longevity testing, are perfect examples of automated tests that can execute without unnecessary bogging down of test

UNIQUE FEATURES

Why ATRT? While numerous automated testing tools are available on the market, ATRT distinguishes itself from the rest by providing the following unique features required for software



Automated Test and Re-test (ATRT) Test Manager (TM)

Enabling Automated Software Testing for Heterogeneous Multi-Computer System Environments – also in the Cloud

testing - among many more common features:

Combining the power of GUI and non-GUI automated testing: ATRT provides GUI and message based testing capability for Systems Under Test (SUTs) when no GUIs are available or when GUI test automation is not beneficial, and Message Based testing is a viable solution. This feature is also very beneficial when GUI based automated testing needs to be enhanced with Message Based testing.

Distributed testing over a network: Automated tests can be executed over a network, for the test case where various GUI or message based outputs are dependent on each other over a network, e.g. GUI output 1 is a prerequisite to GUI output 2, or to test other serial or parallel message or GUI based interactions.

Concurrent testing support: Concurrent testing support is provided, for the test case where various test scenarios have to be run in parallel and race-conditions and other concurrency issues have to be detected, and simulation of live SUT interaction is required.

Batch Processing: A batch processing feature is provided for the case, when various test cases have to be executed together as a batch job as part of endurance or longevity testing.

USABILITY

ATRT has been designed with usability in mind. Testers require effective testing tools without having to be software developers. ATRT comes equipped with the following usability features:

Script-less Automated Testing: ATRT automatically generates software testing code for the tester through simple point and click features. With ATRT's "capture playback" feature, the baseline test code is generated behind the scenes and can then be reused for regression testing for each subsequent SUT release.

Keyword driven Automated Testing: Keyword driven automated testing reduces test creation time significantly, and empowers the tester to generate automated test code via simple clicks of test related test "action" icons that have to be executed as part of a test scenario, generating code behind the scene and is automatically added to the automated test baseline. Keyword driven testing allows for conformity of testing scripts across the board and increases maintainability of automated test scripts.

Data Driven Testing: Input test data can be grouped into boundaries: Valid, invalid, boundary, one off boundary and so forth. Manual testers have learned that most defects congregate around boundary values of data input.

However, the manual testing challenge is that there is often not enough time to cover all the boundaries or equivalence classes of test data input combinations. With this data driven testing feature one test scenario can be reused over and over with different data values and data combinations.

Model-driven Automated Testing: To further increase its ease of use the ATRT is equipped with a model-driven automated testing capability. This powerful feature allows complex test flows (loops, conditionals, etc.) to be designed via a model driven interface via simple point/click/drag, i.e., test creation and execution via a workflow/flowchart user interface.

KEY TECHNOLOGY

Extensible Technology – We use an Eclipse based framework with extensible functionality provided through plug-ins, in contrast to some other applications where functionality is hard coded. Customers may extend UI aspects (layout, menus, toolbars, etc.) as well as functionality.

Extensible Reporting Framework - Business Intelligence and Reporting Tools (BIRT) integration allows for a flexible and extensible reporting framework. Users can design custom reports including an export capability to various reporting formats (pdf, Word, html, etc.) is provided.



Automated Test and Re-test (ATRT) Test Manager (TM)

Enabling Automated Software Testing for Heterogeneous Multi-Computer System Environments – also in the Cloud

Virtual Network Computing (VNC) – VNC technology allows the ATRT Test Manager’s installation to be non-intrusive to the System under Test (SUT), i.e. it doesn’t modify the SUT configuration, which is often a pre-requisite to testing of large commercial software systems.

Cross-platform and cross-Operating System (OS) compatible – Many organizations are responsible for testing SUTs that run on various platforms across various Operating Systems. Instead of having to rely on purchasing different tools to be compatible with the different environments, ATRT Test Manager is cross-platform and cross- OS compatible: It is compatible with Web, Client Server, Handheld platforms and Linux, Windows, Solaris OSs.

GUI Technology neutral – Many of the currently available automated testing tools depend on GUI Technology and for example, while one tool is compatible with GUI control A, it might not be compatible with GUI control B. New GUI controls are being developed constantly and the testing tool vendors are having a difficult time maintaining tool compatibility. The ATRT is independent of any GUI controls (custom or non-custom) and is GUI technology neutral.

XML based data stores – Test data artifacts are stored in XML format, which allows for industry standard

cross-compatible data exchange and data representation.

OTHER NOTEABLE FEATURES

Test Management: Full-featured Test Management tool features, allowing for manual and automated test case documentation, broken down into test case -> test step -> detailed test step and so forth.

Requirements Traceability Matrix (RTM): In order to track progress an RTM feature is provided that allows for traceability of all test artifacts back to the requirements.

Enhanced Error Debugging: The debug feature provides the automated tester with instant inside knowledge of a failure, i.e. expected vs. actual screenshot results are displayed for comparison, differences are highlighted. Additionally, the tester can replace a failed step with the baseline, if the SUT changes are expected. During an automated test run debug can be run in parallel to the test executing, thus enhancing the debug efforts.

Optical Character Recognition (OCR): Sometimes text needs to be extracted from a non-editable image field on the SUT’s GUI and then be compared to another field element on another part of the SUT’s GUI, or another GUI altogether. ATRT’s Test Manager has solved this issue by providing an OCR capability that allows users to “extract” the non-editable image data to be able to compare it.

Automated Capture using X,Y coordinates: For quick and easy automated test creating and for SUT’s whose GUI image positions rarely change, an automated capture feature based on x,y coordinates is provided.

Connecting to a different SUT component at the sub-step level: When a test sub-step requires a change in connection to a different SUT component, ATRT’s powerful IP address/connection feature allows a test sub-step to be modified and connected at the most granular level, i.e. the test sub-step level.