

ETA (Easy Test Automation) Product Overview

Key Benefits

- Rapid Test Development Prior to Systems Under Test (SUT) availability.
- Integrated common language Test Plan with the automation that implement them.
- Reduced time-to-market by having a Just In Time inventory of properly verified automated tests prior to SUT availability.
- ISO Development process improvement by having Product Development staff review both the Test Plan Document and the Automation during the Test Plan review Phase.

Increasing demand for rapid development, test, and release of products to meet time-to-market targets has resulted in ever-increasing workloads for the Development and Verification teams throughout the hi-tech vendor community. While Developer tools have addressed many of the requirements for productivity-related enhancements, the tools in use within the Test and Verification discipline continue limping along with 1980's vintage technology. Isn't it time to enhance the technology and increase the productivity within your QA organization?

Etaliq's ETA product is designed to minimize the time spent on repetitive tasks within your QA organization. It provides a fully integrated IDE for your Test and Automation resources.

Etaliq's easy to use, patent pending, ETA tool and language provides the ability to integrate Test Plans and Cases with the Automation that implements them.

ETA also includes a device and testbed inventory, a fully integrated Test Plan and Case repository, a full featured programmable scheduler, an integrated indexed log and report viewer, and a summarized reporting facility for test results and device usage; all integrated in a client/server application model.

ETA provides a series of subsystems that integrate to form a state-of-the-art Test Automation IDE.

- GUI Application
- File Manager
- Syntax Checker
- Scheduler/Execution Manager
- Synchronized Log/Report Viewer
- Summarized Execution and Device Usage Reporting
- System Dispatcher

Test Automation Development

Traditionally, Test Plans are written using common language, by Subject Matter Experts (SMEs), describing the objectives of each test and their associated setup and steps. Some of these tests are then translated, by Test Automation Specialists, into scripts that verify the functionality described within each Test Case.

Etaliq's ETA product integrates this process by providing the ability to write the Test Automation directly into the common language Test Plans. ETA's easy to use command language reads like English and is used by SMEs during the Test Plan writing phase, prior to the availability of the new or enhanced System Under Test (SUT). It is no longer necessary to have separate specialists to create the automation.

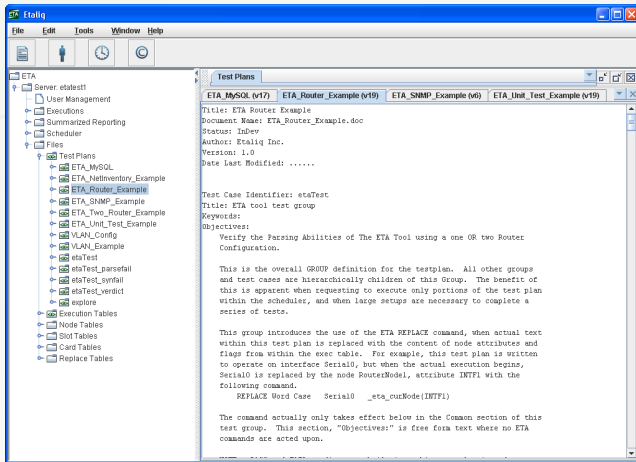
Additionally, ETA provides an SUT device simulation facility, which enables automation code verification and debugging prior to SUT availability. This is a large time saver that contributes significantly to meeting time-to-market objectives. This is a dramatic improvement over traditional Capture/Playback techniques that require access to the SUT in order to build and verify automated tests.

ISO Process Improvement

Integrating the Test Plan with the automation provides an additional side benefit, in that the Development Engineers responsible for the creation of the product or feature enhancement, actually review the test automation and not just the common language Test Plan. This contributes to dramatically fewer errors and inconsistencies between the Developer written Documentation and the Automated Test system. This review can now occur during the verification or Test Plan review process. ISO processes used in System Development, are thereby enhanced through the use of ETA.

Test Plans Always Match the Automation

Integrating the Test Plan with the automation provides yet another side benefit, in that the Test Plan and the Automation will never be mis-matched. Dual maintenance of the automation code and the Test Plan document is no longer required.



ETA's Command Language Is Simplicity

Automation is simply the repeating process of Send, Receive and Verify. ETA's command language simplifies the process of automated testing some of the worlds most complex systems in the Engineering community. This exemplifies how simple it is:

```
1 RESULTLIST(EXPECTED) "software version >= 3.4.1"
2 "ASIC Hw Pkg = xLCv42f"
3 SEND NODE1 "display system info" EXPECTED
```

The first command defines a result list named "EXPECTED", which contains two attribute verification requirements. The SUT attribute, "software version", must be greater than or equal to '3.4.1' and the SUT attribute "ASIC Hw Pkg" must be equal to 'xLCv42f'. The second command can be read as follows: SEND to "NODE1" the command "display system info" and verify that the response matches the "EXPECTED" result list.

These are two of only thirty ETA commands that completely simplify the test automation process. It is no longer necessary to struggle with Tcl or Perl syntax, regular expression parsing constructs, or errors that only become apparent when running the scripted automation. ETA has completely simplified the Automation Development process such that separate Subject Matter Experts and Automation Specialists are no longer required.

ETA File Manager

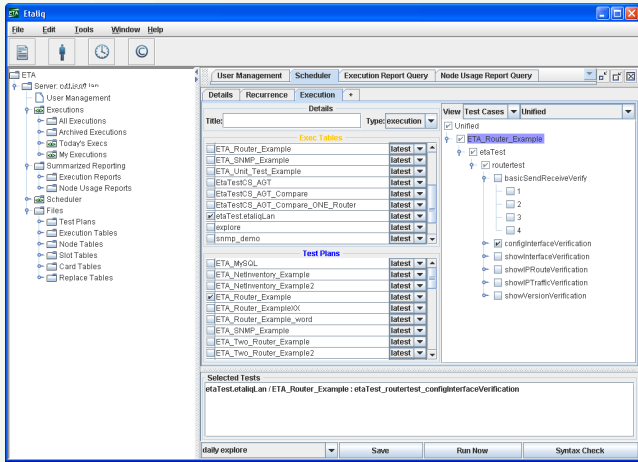
All Test Plans, Device Definition tables, Execution Parameter files, and Hardware specifications, are maintained by the ETA File Manager. All versions of all files are maintained in a repository for easy reference and use.

ETA Syntax Checker

ETA includes a Syntax Checker that verifies the proper use of the command language, in combination with the Device Definitions tables, Execution Parameter files, and Hardware specifications, prior to commencing every execution. This facility can also be called on demand by ETA users while writing their Test Automation. This functionality saves a tremendous amount of time over traditional scripting languages and ensures the best possible use of Test Resources, including Nodes/SUT's, Traffic Generators, Workstations, as well as Database, File, and Web Servers.

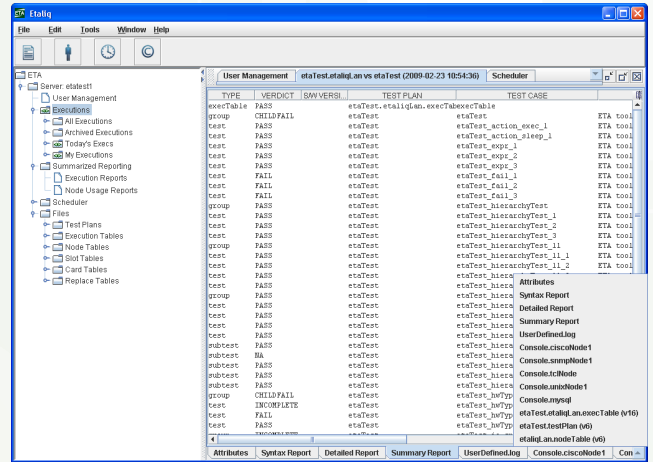
ETA Execution Manager

ETA's Execution Manager provides real time access to execution logs and reports, on the fly, as they are generated. The execution Manager creates a series of files for each execution including an Execution Attribute Record, a Summary Report showing individual Test Case Verdicts, a Detailed Report detailing the steps of each case, a Console Log per device used in the execution as well as the actual Test Plan, Device Definitions and Execution Parameters files used in the execution.



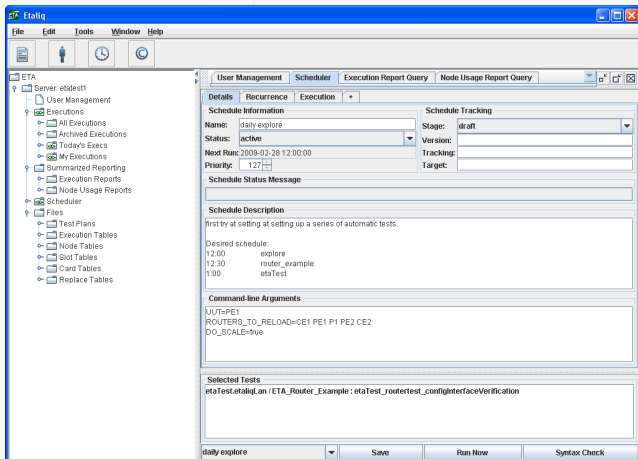
ETA Synchronized Log and Report Viewer

ETA's execution logs and reports are fully indexed, not only to each other, but to the Test Plan, Device Definition and Execution Parameters files themselves. The Detailed Execution report is fully indexed to the all other files created within the execution.

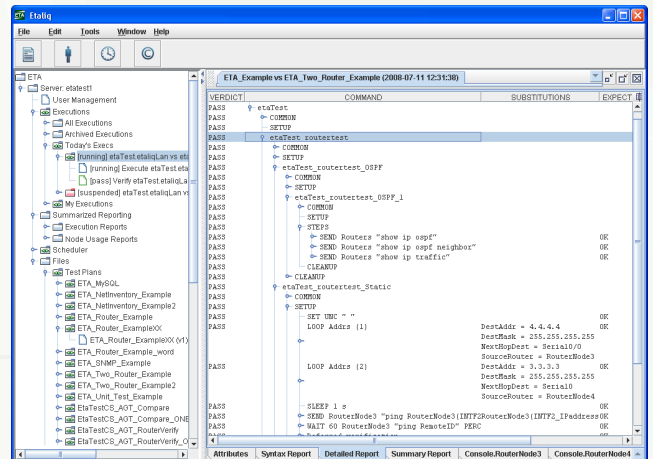


ETA Scheduler

ETA includes a full function scheduler for unattended execution of Test Plans. Schedules can drive either one time executions or recurring schedules. Recurring schedules can be programmed to operate daily or weekly or monthly, or every Tuesday and Saturday, or per month on the 1st and 15th calendar day.



This unique ETA feature is invaluable when attempting to re-construct a customer found problem months after a product release. The goal is to find out why the problem was not discovered in automation and ensure that a test is either created or updated to ensure the error does not occur again. With ETA it is no longer necessary to hunt through CVS or RCS, for the various script and library versions, get a copy of the retained report and console logs, and match these to where the automation needs to be updated to fill the gap.

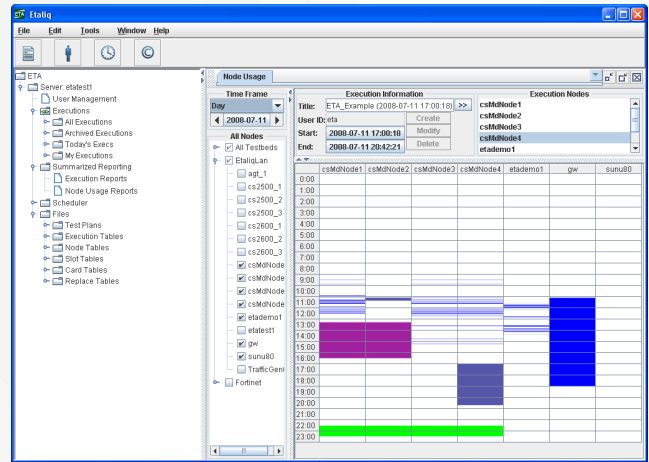


ETA Summarized Reporting

Manager and Engineers are always looking for summarized reports that show the current state of the SUT product according to the latest Test Results. ETA provides a completely customizable summarized reporting capability. Users can request per execution/daily/weekly or monthly summary reports for all Test Plan or Case verdicts. Users can request various filters be included for these executions. A project manager may only be interested in the execution results for a particular project or a Director may only be interested in production execution results against a particular target. The ETA Summarized Execution Reporting facility is completely customizable to your individual needs.

Execution Report		2008				TOTAL
		7	8	30	31	
ETA_NetInventory_Example	CHILDFAIL	0	8	0	1	10
	FAIL	0	8	0	1	10
	PASS	0	22	0	1	23
ETA_Router_Example	CHILDFAIL	18	51	4	0	73
	FAIL	17	43	4	0	64
	PASS	84	168	13	0	265
	UNCLEAR	6	11	0	0	17
ETA_SNMP_Example	CHILDFAIL	1	26	0	0	27
	FAIL	0	21	0	1	22
	INCOMPLETE	1	0	0	0	1
	PARENTFAIL	4	0	0	0	4
	PASS	0	33	0	3	36
TOTAL		122	321	17	5	558

Individual Managers may be interested in the hardware usage statistics about the devices under their charge. Their goal may be to ensure that their hardware is fully used before adding additional capacity, and budget. ETA provides a graphical view to device usage where different colors show when devices are used by automation, locked out for manual devtest, or unused and therefore available for additional tasks.



Contact Us:

Etaliq Inc.
4B-2548 Sheffield Road
Ottawa, ON K1B 3V7

<http://www.etaliq.com>
sales@etaliq.com
support@etaliq.com
Phone: (613) 241-1385
Fax: (613) 241-1523