Verifide Technologies Inc

8/21/2014

PASS vs NI TestStand

A comparison of the leading test executive to our Enterprise Test Executive

Neil Baliga

Verifide Technologies, Inc.

Executive Summary

National Instruments is the market leader in test executive software with their TestStand product. Along with the wealthy ecosystem they provide, it offers a homogenous solution for those whose tool of choice is Labview or Labwindows.

The TestStand product also sports lots of features and configuration options including parallel execution and integration with switching etc.

The problem however, is that like other test executives on the market, it does not provide a complete end-to-end solution for the production environment. By this, I mean that it only provides a small piece of the overall test life-cycle.

Furthermore, increasing device complexities and volumes are requiring test organizations to strive for modularity and efficiencies in their process. The mechanisms by which test sequences are created, edited, and iterated are becoming significant cost drivers.

These constraints call for a new approach to testing, which TestStand and other test executives do not offer.

In this document, we shall compare and contrast our enterprise test executive PASS, with the current market leader NI TestStand.

Verifide Technologies Inc

8/21/2014

Test Executive Comparison

Feature	TestStand	PASS
Sequencing	NI implements sequencing with STEPS. Each step is autonomous and needs its inputs and outputs well defined. This makes it hard when needing to setup thousands of steps, or when needing to edit steps. Systems engineering needs to provide requirements that developers need to interpret, then implement – a more error prone system	PASS Implements table-driven sequences. User can create, edit, and copy/paste sequences in Microsoft Excel™. PASS Implements a parameter resolution engine that processes tables for test input parameters. This allows Sys. Engineering teams to develop ACTUAL sequences and test input parameters that are ran on the production floor. Increasing rate of return on this when implemented on higher complexity and volume.
Data Management	TestStand allows data storage to a database, but the DB structure has to be PRECONFIGURED and defined. Any changes to data, then require DB level changes. This is difficult to manage across product lines or testers because each product/test will store dissimilar data. TestStand does not have any built-in tools for trending or production metrics. Some 3r party products do integrate, but it requires data to be exported to their formats and their servers in order to execute the metrics.	PASS provides ENTERPRISE-grade data management that is truly scalable. The database builds itself as new data is stored using our dynamic database engine. Because data is not rigid, our tools allow generic access to pull data for trending, metrics, reporting, and selloff package generation – ALL OUT OF THE BOX. PASS provides RICH user interface applications to view raw data, view trend charts (export to excel for deeper analysis), view pivot grid metrics and analysis, verify sequence compliance, perform data audit and review capabilities etc.
Reporting	NI provides reporting tools to generate data reports from stored data. These reports, however are not data-bound reports and cannot be regenerated after	PASS has its own BUILT-IN reporting engine. It uses a data-bound approach to reporting whereby the developer creates a template with images, tables, charts, and labels etc. This is done in the included

	the fact unless the test and data acquisition is redone. There is no native ability to generate reports to PDF files directly. The reports make use of ActiveX controls that need to be resident on the system to view charts, else the report only shows tables – this is bad because reports are not purely transportable and deliverable to end customers. Integrating with other 3rd party reporting solutions is available, but the developer has to do additional work. Furthermore, there is no data browser that will allow you to browse for results and view reports in a single application	Report Designer that allows the developer to drag and drop placeholders to design the report. The report is then generated by passing test data (live, or at-rest) through the report template to generate the reports. This allow tests to have multiple report views of the same data, as well as the ability to REGENERATE reports on-demand. The PASS report engine allows generation of reports to print, print-preview, PDF file, or Image file. This can be done manually from the user interface, or programmatically through our Framework API.
Logging	NI provides simple trace logging. There are additional tools for logging that are included in marked-up packages like DIADEM. However, the logging mechanisms are mostly geared towards data logging and not activity logging. In complex systems, activity logging is very important to document the traceable history of the system activity (i.e. which tests were ran, which switches were thrown, over voltage conditions etc) There are no tools provided to view, filter, and data mine activity logs	PASS provides a comprehensive logging framework to monitor the status of the test system and tests. Our logging allows text as well as binary embedded data to be logged including information such as full error stack traces. The logging is designed for enterprise architectures where logs are monitored remotely – i.e. logs can be forwarded through TCP ports across firewalls for remote support etc. The log viewer application is feature rich and allows LIVE log message filtering by level and/or text with search text highlighting.
StripCharting	NI has additional tools for strip charting that are NOT included with TestStand.	PASS provides strip chart views of data log streams that are INTEGRATED into the Event and Alarm mechanisms included with the PASS software. This means that strip chart streams can be configured with any number of limits that can then trigger a system shutdown and

		put the system into a safe state – AUTOMATICALLY.
Services Framework	TestStand does not provide the ability to host your own components as services. Modern systems need services to be running to monitor and control test equipment and devices under test. To provide such capability, you will have to implement your own services architecture and TCP/IP inter-process messaging	PASS provides a built-in distributed services framework called the VServer which allows hosting of ANY .NET component. The services framework provides built-in IPC messaging, logging, resource locking, and error handling. This is the mechanism by which system functionality is created. Tests, drivers, utilities, and services are all hosted in these servers to provide truly modular and isolated components.
Event and System State Management	TestStand does not provide any system level state or asynchronous event handling infrastructure	PASS has a built-in event manager and state subsystem. It allows timer based events to be raised and handled (manually, and programmatically). The event subsystem allows capabilities to automatically shut down the system when critical events are not acknowledged, for example. The state and event management provides AUDIBLE alarms to alert operators that attention is required on the system.
Standard Data Formats	TestStand allows data storage to XML and databases, but it does not have a single unified data format that it uses. It has tools for ATML, which is a very RIGID data format and is difficult to parse with the human eye.	PASS implements a core data format called VDATA. This data structure is a tree based structure that allows infinite flexibility in how data and nodes are created and stored. It supports all scalar data types, arrays, tables, images, blobs, complex etc. The software comes with a RICH user interface to visualize the data including the ability to view and export tables, charts and images.

		It supports data protection features such as limits on entered values and password protection including encryption. It is the same data format used all across the PASS software for configuration files, interprocess messaging, data storage/retrieval, report templates, and sequence files as well.
Test Development	TestStand has really good integration with Labview. Development with other languages is not as streamlined and takes more API knowledge to make the most of the test executive. Non VI modules are TIGHTLY bound to the sequence. For example, using a .NET component version 1 in the sequence will require the same version when running the sequence. This is harder to manage from a configuration management point of view.	PASS provides development in any .NET compatible language. .NET DLL's are easily hosted in the VServer architecture and tests are FULLY isolated from the sequencer. The sequence files are just tables with cells. Development in non .NET languages such as Labview, Python etc. require the use of an adapter (included in our software) to call the module and pass input parameters etc to the module. The Labview code, then makes full use of the PASS API and can use all the features included in our software (data management, logging, events, resource locking, reporting, error handling etc)