DSI international, inc.

eXpress™ Newsletter

Volume 31 Number 1 Spring 2025

The Four Levels of Diagnostic Correlation

Introduction

With the advent of digitally-integrated engineering, there is an increasing expectation that design-time diagnostic assessments be correlated with the actual diagnostics run in the field. When this correlation is levied as a requirement, suppliers are finding themselves confronted with a number of challenges.

The most daunting of these challenges are often systemic. For many organizations, diagnostic assessment and diagnostic development have traditionally been handled by separate teams, each employing distinct methodologies, data sets, and optimization criteria. Moreover, funding profiles often vary across disciplines, resulting in assessments that fail to capture the final state of the design, the latest version of the diagnostics, or the conditions under which the system will ultimately be maintained.

Nevertheless, even highly siloed diagnostic engineering efforts allow for some degree of diagnostic correlation, though the cost of doing this retroactively likely increases with the extent of systemic dysfunction. To better understand this, let's look at four levels of diagnostic correlation, from the simplest (most easily implemented) to the most speculative.

Level I: Test Usage Correlation

The most basic level of correlation involves comparing the tests used by implemented diagnostic strategies with those presumed during diagnostic analysis. While this task may seem straightforward, any uncovered discrepancies could not only undermine confidence in the diagnostic analysis but also signal deeper issues within the organization's diagnostic engineering process.

The most frequent reason for these discrepancies is insufficient coordination between the parties involved in the diagnostic engineering effort. Without proper coordination, analysts may mistakenly believe that certain tests have been developed and are accessible at the targeted maintenance level. On the other hand, the run-time diagnostics may incorporate case-based symptoms or rules not identified by the diagnostic analysts.

One might think that, by integrating feedback from both efforts into the overall diagnostic engineering process, an organization can effectively address and eliminate these discrepancies. For a diagnostic analysis to be effective, however, it should follow an agile process capable of incorporating periodic updates as the

continued on page 2

Inside this Issue...

Latest Software Versions

• ***eXpress***	8.1.0	3/25
• ***eXpress*** Design Viewer	2.2.4	7/24
• Run-Time Authoring Tool	6.4.3	4/25
• DSI Workbench	5.3.3	4/25
• TestDRIVE	2.5.1	10/24
• STAGE	Act II, Scene 3	11/23

from page 1

design evolves. Diagnostic analysis should commence early in the development process to allow timely feedback and ensure that any issues impacting diagnostic effectiveness are resolved before it's too late...yet this analysis must be revised beyond its peak period of usefulness to ensure alignment with the diagnostics. Despite employing a sufficiently agile process, maintaining up-to-date, *correlatable* analysis might be deemed 'out-of-scope' for a standard procurement contract and may need to be included as an explicit requirement.

Level II: Test Coverage Correlation

If the diagnostic analysis has been shown to use the same set of tests as the implemented diagnostics, the next level of correlation is to establish that both disciplines assume identical coverage for each test. The challenge here is that analysts and developers often think about tests in fundamentally different ways.

For analysts, tests coverage typically refers to the set of faults that a test can detect when it fails or the potential faults it can rule out when it succeeds. Diagnostic developers, on the other hand, often concentrate on the specific defects a test has been designed to uncover, rather than accounting for all possible issues that could trigger its failure. This difference is exacerbated by the fact that *test* developers typically focus not on test coverage, but rather on the specific signals or signatures being evaluated by individual sensors, Built-In Tests (BIT) or automated testing systems.

An effective way to address this issue is to invert the concept of test coverage: rather than prove that each test identifies the same faults, demonstrate that each fault produces the same fault signature—an identical set of failed tests. When model-based approaches have been used for both diagnostic analysis and diagnostic development, software can simply restructure the test coverage data into fault signature lookup tables that can be compared for inconsistencies. To ensure that the implemented tests perform as expected by diagnostic engineering, however, a fault insertion exercise will likely be required.

By introducing faults—either virtually or physically—into the system or device, one can identify precisely which tests fail for each specific fault. The resulting fault signatures can then be cross-referenced with those produced by diagnostic engineering to ensure that the tests used by the run-time diagnostics align with engineering expectations. Discrepancies may arise not only from differing interpretations of test objectives, but also from inconsistencies between the pass/fail criteria (e.g., tolerances) for each test and the diagnostic conclusions derived from them.

Using physical fault insertion to correlate test coverage for full systems will inevitably be both time consuming and costly. There are two ways in which this can be mitigated. The first involves defining a set of representative malfunctions and performing fault insertion for only the selected faults. If the resulting fault signatures align with those provided by diagnostic engineering, it can be inferred that test coverage is sufficiently correlated between the diagnostic analysis/development efforts and the fielded run-time diagnostics. The other approach would be to insert faults using a digital twin of the system. This adds an additional variable to the equation, however, as the correlation effort must now also confirm that the digital twin accurately replicates the performance of the physical system.

Level III: Correlation of Diagnostic Conclusions

After ensuring that the test coverage aligns, the next level of correlation would be the comparison of diagnostic conclusions. The ambiguity groups (sets of isolated items) identified by the diagnostic analysis must be identical to those isolated by the fielded diagnostics for the same fault signatures.

Correlating diagnostic conclusions is more challenging than test coverage correlation, though both activities follow the same procedure: faults are inserted into the run-time system and the resulting diagnoses (sets of suspected items) are compared with the corresponding ambiguity groups from the diagnostic analysis. This approach enables the correlation of diagnostics based on different methodologies. The correlation is considered successful if the diagnostics consistently isolate the inserted faults to the expected set of suspected items.

There are a number of reasons why run-time diagnostics might yield different results than the diagnostics used for testability and other design-time analyses. First of all, the diagnostic strategies often serve distinct purposes. For instance, testability analysis produces maintenance-oriented metrics highlighting diagnostic issues that might potentially impact availability or life-cycle cost. These metrics are based on a diagnostic approach that attempts to identify the root cause of each failure and isolate it to the smallest set of items for the specified level of repair. Conversely, embedded diagnostics prioritize the reporting of system status and the initiation of actions to mitigate critical malfunctions. Due to these differing objectives, the resulting diagnostic conclusions can vary significantly.

Diagnostic strategies might also be based on different underlying assumptions. Effective run-time diagnostics are designed to consistently and accurately identify issues, even when faced with multiple simultaneous malfunctions. Testability analysis, on the other hand, is based on entirely different premises. The equations used to calculate fault detection and isolation—metrics documented in both military and IEEE standards—serve as the foundation for Testability requirements in nearly all government procurement contracts. These equations assume that only a single malfunction exists as the system is diagnosed. The diagnostics used for these analyses not only presuppose a single fault, but are often optimized based on that assumption. This reflects the fundamental aim of testability analysis: not to forecast diagnostic performance in real-world scenarios, but to demonstrate that the design is structured to enable efficient and reliable diagnostics.

Unfortunately, diagnostic conclusions from strategies designed for specific objectives or based on differing assumptions often prove to be irreconcilable. For projects that adopt an agile approach to model-based diagnostic analysis, however, tailored correlation-ready diagnostic conclusions can be generated by adjusting various software settings to match the objectives of the run-time diagnostics. Since these conclusions are derived from the same model used by the diagnostic analysis, their alignment with run-time diagnostic outcomes serves to validate the consistency of the analysis

Level IV: Correlation of Diagnostic Performance

Industry pundits have occasionally argued that logistics disciplines should be held to more rigorous standards by ensuring that the results of design-phase analyses are systematically compared with the real-world performance of implemented systems. For diagnostic engineers, this process would involve evaluating the real-world rates of detected and undetected failures, unambiguously isolated faults, false removals and diagnostic-related false alarms against design phase projections.

While the intent behind such correlations is commendable, they inevitably fall short as an effective means of validating diagnostic analyses. This limitation arises not only from the previously mentioned discrepancies in methodology, objectives and assumptions but also from inherent challenges in translating design characteristics into accurate predictions of system behavior.

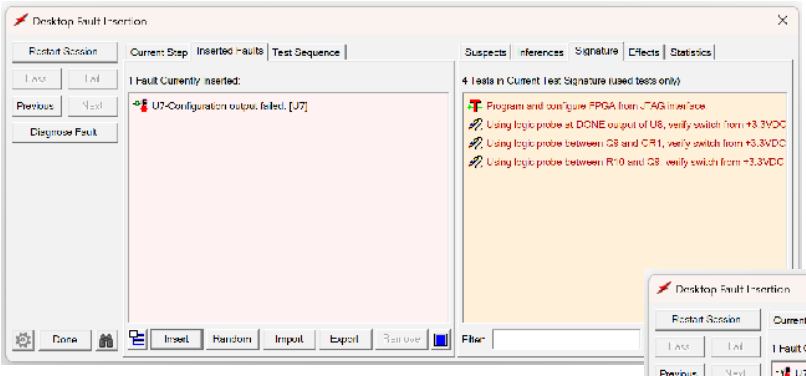
As a straightforward example, let's look at fault detection metrics. Analysts can easily determine the percentage of possible system faults that can be detected by a given diagnostic strategy. To convert these percentages into projections, reliability data is used to assign weights to various failures. This approach ensures that faults with higher expected failure rates exert a greater influence on the projected rate, while those with lower occurrence rates have a minimal impact.

There are several problems with this approach, however. Most importantly, failure rates are notoriously inaccurate—especially for newly developed components. Raw failure rates must also be adjusted to account for the relative usage of each component (another act of probabilistic guesswork). Worse yet, the frequency that a given component fails may change based not only upon its usage, but also operating conditions, the existence of other failures and the occurrence of unanticipated events.

The manner in which a system is maintained significantly impacts how it fails. When a fault occurs and the associated component is replaced, not only is the immediate issue resolved, but the remaining useful life of the component is effectively reset. This reset greatly reduces the likelihood of less common failures. Anticipatory replacements driven by preventative or predictive maintenance also impact each component's effective failure rate, resulting in further differences between reliability-weighted performance projections and real-life system behavior.

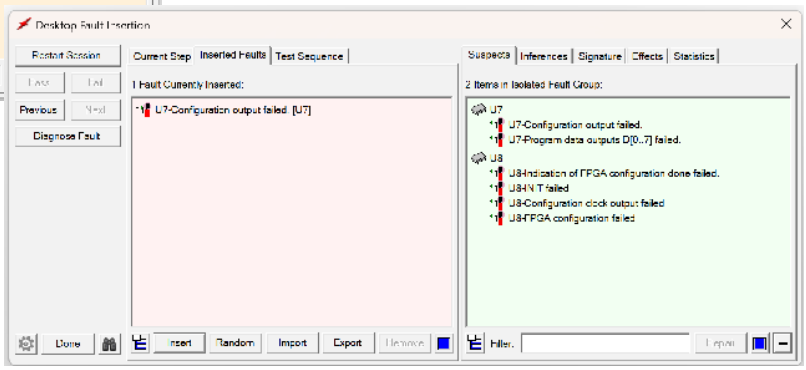
Does this suggest that the correlation of diagnostic performance is a lost cause? Absolutely not. While it may not be an effective technique for validation, this correlation provides a valuable way to pinpoint opportunities for design improvement. Variances between observed and predicted failure rates, test/repair times, and other sustainment-related characteristics, can help highlight opportunities to refine diagnostic strategies and optimize test and maintenance procedures—ultimately resulting in improved mission capability with a reduced maintenance footprint.

Desktop Fault Insertion™ & Diagnostic Correlation



The ***eXpress*** Desktop Fault Insertion capability aids diagnostic engineers in identifying and investigating discrepancies between the ***eXpress*** diagnostics and an implemented diagnostic strategy.

One useful feature of this dialog is the ability to insert one or more faults and examine the fault signature—the set of failed tests—that would result (example at left).



This dialog also allows you to compare fault groups isolated by ***eXpress*** with those identified by another diagnostic. Insert one or more faults and click on the "Diagnose Fault" button. You can then examine the isolated fault group (shown at right), view the associated test sequence, and even review test-by-test the inferences leading to the diagnosis.

DSI international, inc.

eXpress User's Group 2025

Virtual User's Group Meeting

June 18, 2025

7:00 – 11:30 a.m. PDT

We have a dynamic and informative agenda in store:

Presentations by Industry Specialists

Overview of Digital Diagnostic Engineering

Demonstrations of New Features, including

- Capella2eXpress
- Mission Phases in STAGE
- False Alarm Calculations

Discussion of Future Software Plans

Contact DSI today at info@dsiintl.com
or register online at www.dsiintl.com/EUG25

Recent Software Releases

eXpress
Version 8.1

FTA Module Now Automatically Included
Available Capella to ***eXpress*** Translation

RTAT
Version 6.4.3

Now with Automated View Mapping

DSI Workbench
Version 5.3.3

Now with TCP/IP Interface

Coming Soon

STAGE
Act III, Scene 1

Now with Mission Phases

***eXpress* Maintenance Module**
Mark III

Now with False Alarm Calculations

Schedule of Events

Course Number	Prerequisite	Description	Dates	Location	POC
CE-345 (repeat)	none	Continuing Education: Creating and Mapping Views in DSI Workbench	April 15, 2025 One 90-minute session	Virtual: Webex	info@dsiintl.com
CE-356	none	Continuing Education: Failure Rates & Probabilities in ***eXpress***	May 6, 2025 One 90-minute session	Virtual: Webex	info@dsiintl.com
TLS-100	2 hours home study prior to first session (video)	System Diagnostics Concepts and Applications Basic Modeling & Introduction to Testing	Starting May 12, 2025 Eight 4-hour sessions (Mon-Thu for 2 weeks)	Virtual: Webex In Person: Orange, CA	info@dsiintl.com
CE-357	non	Continuing Education: Using Custom Symbols in ***eXpress*** & DSI Workbench	June 3, 2025 One 90-minute session	Virtual: Webex	info@dsiintl.com
EUG-25	N/A	***eXpress*** User's Group 2025	June 18, 2025 4 hours	Virtual: Webex	info@dsiintl.com

World Wide Representatives

United States
DSI International, Inc.
(714) 637-9325
info@dsiintl.com
www.dsiintl.com

Canada
Acutronic Inc.
(647) 258-6390
marco@acutronix.com
www.acutronix.com

United Kingdom
Spherea
Technology Ltd.
+44 (0)1202 868585
chris.gorringe@spherea.co.uk
www.spherea-technology.co.uk

France
Spherea
Test & Services
+33 (0)5 34 55 43 23
testability@spherea.com
www.spherea.com

Japan
SDK, Inc.
+81-(0)44-322-8460
i-yoshii@sdk.jpn.com
http://sdk.jpn.com

South Korea
Realtimewave Co, Ltd
+82-2-572-9471/2
sales@realtimewave.com
www.realtimewave.com

China
MTCS
Systems Engineering Co. Ltd
+86-10-5881-6565
sales@mtcs.com.cn
www.mtcs.com.cn

The eXpress™ Newsletter is published semi-annually by DSI International