“Diagnostic” Model-Based Reasoning

Discussion: Addressing the Decision-Making Value of MTBF Metric for “Availability”

Individually, we can challenge the utility (in design and/or in “practice”) of the MTBF metric, but getting past so many independently-prescribed interpretations of the metric, what is/are the end-goal(s) or usage intention(s) for the disciplinary-shared (RAM) metric? Is our objective ultimately to “design” for a “reliable product”? If so, how can we “use” the MTBF in assessing the design when comparing the proposed designed product(s) to alternative product designs in terms of (reducing failures) availability, operational success & cost of ownership of the fielded & maintained product(s)?

Alternatively, is our use of MTBF instead, intended to influence the “maintenance” (or “practice”) activities concerning the fielded product? Is our end objective to “use” (in performing maintenance decisions; “in-field practice”) a prediction method to compute expected failure frequency of the component(s) to bias future fielded experience(s) or maintenance corrective activities (repair/replacement)? Are we required to use or rely upon “engineering design” (predictive) data to influence maintenance (or practice) decisions or, instead, rely upon “empirical” (historical) data for sustainment (or advanced test) solutions? Where are we in the design development (or sustainment) process? What (or “when”) is the value that we are trying to formulate, extrapolate or exhume by exploiting the (re)use or calculation of the MTBF metric?

Let’s include parochial uses of the metric not involved in the “practice” or implementing of MTBF in the diagnostic solution:

How do we (seamlessly & affordably) honor past “critical system work product” requirements that base their calculations on the use of MTBF (think FHA or FTA, for example)? If we’re under contract to deliver either of these documents as CDRL items, we must “use” the MTBF metric in the performance of this required assessment product regardless of our deeming the metric to be inadequate or narrowly useful. Else, should we simply agree on a universally sanctioned use or applicability of the MTBF metric across multiple design disciplines? For CDRLs or FAA certification requirements? Can we be provided immediate directions to get there?

Is it a stretch, that if we (in referring to our fielded complex product(s) or system(s)) can’t unambiguously or accurately observe the failure(s), to suggest that the value of our investment into working extensive precision into the failure prediction metric(s) has been reduced? So, if the “Integrated” System doesn’t “see” (observable as constrained by Test Coverage minus Test Coverage Interference) the failure, was it recorded as a failure? Is this oversight a result of diagnostic engineering inadequacy? How would such uncertainty during design development impact NFF, CNDs, System Aborts and RTOK’s during the Sustainment Life-cycle? What is the ubiquitously understood methodology that categorize the “failure effects” of inadequate diagnostic integrity?

While keeping up with our tracking of historical events, how should we know if the replacement activity (replace a group of components) actually resolved all of the failures to the system? … Which of the replaced component(s) actually did not fail & was/were replaced with the (presumed-to-be-failed) component(s)? So, which failures should be recorded? How does this impact future calculations for component MTBF, if any? This can get much more complex by the way, as Reliability, Availability and Maintenance (RAM) do not independently subscribe to a paradigm that gracefully interprets the use of “Fault Groups”. Maintainability may typically perform a “LORA” (Level Of Repair Analysis), but it sure fails at accurately identifying failure interdependencies for complex integrated systems in the DDLC.

Additionally, the more complex integrated systems may have sustainment philosophies that vary based upon variable environmental mission readiness requirements for maritime as opposed to wartime criterion. Corrective Maintenance (Run-to-Failure) strategies may only be “preferred” for less critical “observed” or “sensed” failures, while the sustainment philosophy may also rely on Preventative (replace before the likelihood of occurrence or an “expected” failure – via CBM & RCM) sustainment strategy/ies for more critical or “CBM-triggered” replacements.

This variable replacement strategy further muddies the quality of the data acquired in the tracking of the serviceable life ("TTF", Time To Failure & "RUL" Remaining Useful Life) of replaced, interdependent non-failed components. Often, more complex integrated systems use BIT to indict (presumed-to-be) failures at the integrated system(s) level (typically, observed as failure effects in a specific mission phase of operation), which may or may not be reconfirmed as an actual failure during (during I or D-Level) sustainment testing.

Prioritizing Design Influence Benefits for Ensuring Decision-Making Data Quality:

Most of these, and an ocean of other factors, impede the accuracy and comprehensiveness of solely relying of tracking and reporting failures. Furthermore, the variable, program-independently-prescribed corrective actions ("FRACAS") data for the data mining & computing of MTBF values (to 6-9 decimal digits) can often work into a FRACAS-leveraged sustainment inaccuracy. Need to "initially" ensure component replacements were "required" to resolve complex system failure(s). Otherwise, FRACAS "practices" can saliently mischaracterize failure resolutions as valid sustainment tactics. Worse yet, planning to use FRACAS as a primary driver to "mature" (at uncontrollable costs) sustainment decision-making as a reactive-diagnostic approach to sustainment activities (practices) is not a solution that counter-balances inadequate diagnostic design (non-critical) strategy decisions.

If it appears that I may be banging on the relevance of diagnostics design when concerned about metrics that, is some role are involved in the playfield for the determining or the system-impact of "failures" (including the MTBF metric), KNOWING the diagnostic integrity of the (complex system) design is a companion prerequisite. Otherwise, this is where we unknowingly invite costly back-end-driven costs into unwittingly reworking the sustainment decisions based upon conjecture of resolutions that may not be comprehensive, accurate and overly-presumptuous, which can be better addressed at another time). FRACAS activities ought to be for sustainment tweaking and tracking. Period.

Establishing A Robust Diagnostic Design & Support "Disciplinary-Interdependence" Environment:

Systems Engineers too often fail at seeking the value from diagnostics engineering to leverage investment into companion multi-disciplinary design development work products that not only can behave as (incidental) "cross-validation" methodologies, but also open doors to a vast array of interactive, design-development-based "disciplinary-hybrid" (including selective stochastic-generated) metrics that we can explore offline, if desired.

Exceptional Diagnostics Engineering Extends far beyond the Boundaries of 2-Demensional Spreadsheets:

Folks that have extensive systems engineering background coupled with expertise in RAM, ought to be the first in line to beat down the bushes daily with the value of KNOWING the "TEST COVERAGE". If we're concerned about how to better consider the impact of system failures in assessment or deployment "practices", we must first be concerned with the ability to identify (detect AND isolate) failures through "test coverage" requirements, analyses & optimization in the (integrated system(s)) design development activities. Traditional spreadsheet approaches are painfully inadequate for affordable & effective assessment & management of comprehensive test coverage of interdependent functional and failure effect propagation characteristics characterized by today's complex or large integrated system(s) diagnostics design (ISDD) involving multiple design tools, techniques, output detail, requirements, etc. from various design activities and/or for inter-organizational programs.

What does the test (place or mechanism, sensor, etc. to query for proper functionality or lack thereof) cover, in terms of what diagnostic conclusions can be made from that pass or failed test. What (fault group) is called into suspicion, what other events (inherent to the design, including an "integrated systems' design) may "interfere" with the accuracy of that test?

Is the (integrated systems') design capable of isolating to, in fact, a single component and discern if the failure is the loss of a non-critical function or a more safety critical function (ability to "uniquely isolate" functions/failure effects), or does it need to?

Balancing Maintenance "mix" to Optimize Sustainment Objective(s):

Diagnostic design decisions (strategies) are often based or impacted by the assigned failure rates (implying the use of MTBF in design assessment, once again) of components as part and parcel to the design. This is important design opportunity to discover & ensure the critical functions are not "grouped", for example, with less critical failures, that in such event, will force a more drastic corrective action. Here is where we can move to more advanced metrics and assess the diagnostic design integrity for (corrective or preventative "maintenance") actions that result in "False Mission Aborts" vs. "True Mission Aborts". Else, "True Diagnostic Alarms" vs. "False Diagnostic Alarms", etc. (via DSI's "STAGE" seamless Operational Support simulation). These assessments would be pulled directly from the identical diagnostic design of the integrated system, containing design input from multiple design disciplines to generate new metrics not otherwise possible from any discipline independently. That said, investment into Reliability-based predictions metrics (including MTBF) can be leveraged to provide input and value for optimizing (dRAM) design-based decisions rather elegantly, whereas such input may be a false path if used to independently drive maintenance actions (or "practices").

When all of the functional interrelationships and failure effect propagation is "consumed" into the integrated systems diagnostic design, we can move our diagnostic alternatives quite a few giant steps forward - particularly on complex or "Big Boy", large Integrated System(s) design(s).

Evolving with Back-End Test Technology:

Let's allow the diagnostic strategies and solutions to evolve with currently available (back-end "test", etc.) technology over time without losing traceability to the Systems' Operational Requirements document. This endeavor will require the gathering of information for making decisions from either "Design Knowledge" (including full knowledge of test coverage, at a minimum) and "Historical Knowledge"(updated and maintained concurrently), but present the field experience the opportunity to include both. In this manner, we can use MTBF (failure rates used in design-based assessment) to spur or seed other related purposes (a myriad of "availability" assessment metrics or enriched before reuse in concurrent variant designs, ensuing designs, etc.)'

Avoid Repeating Sustainment Errors Resulting from "Double-Downing" on Inferior Diagnostic Resolutions:

Tracking dissatisfaction may be noble, but ensuring we have performed a superior job in the detecting and indicting the failed component(s) within the (multiple levels of integrated) system(s) is the first duty. This would entail the knowledge of "test coverage" (including BIT, or any embedded Health Management capability, etc.) and "test interference" (the ability to not assume that every test is able to "uniquely isolate" a single function) within the integrated system(s). Reliability Engineering does not concern itself with "fault groups", which means the lowest level of test is consumed within the diagnostic integrity of the integrated system(s), we will replace the fault group (constituency). When we replace the fault group(s), we may typically replace non-failed component(s) with "presumed-to-be-failed" component(s). We can, therefore, need to consider sparing for our lack of diagnostic savvy, which is too simple to avoid experiencing so many undesirables or dissatisfactions that were actually caused by our lack of diagnostic engineering. Diagnostic Engineering can do far more than just the above described snippet to greatly improve dissatisfied customers.

Even should we track dissatisfied customers, have we really tried to get to the cause of the dissatisfaction? The customer may not be aware that maybe he could have save some significant expense or gained some significant increase in operational availability if the integrated system(s) (or supplied, fielded product) was efficiently designed from both a reliability and a diagnostic engineering perspective. NFF, CND's RTOK's and a litany of labels that identify the

dissatisfaction from the inability to accurately and consistently perform effective diagnostics - to the component(s), function(s) or failure effect(s) within the fielded integrated system(s).

For complex or large integrated system(s), which contain subsystem designs and lower level assemblies and all the way down to the functions on the components, or each level of repair as designated for the sustainment requirement(s), we can capture all of the functional and failure propagation of the interrelated pieces of the integrated system(s) design (COTS or sensitive designs included as "replaceable entity"). But once we capture these interdependencies, we can immediately "push" the entire capture design into a simulation environment that will, based on the diagnostic integrity of the integrated system(s), enable over a hundred simulations to forecast failures within the constraints of the diagnostic design. So, failures within fault groups will trigger replacement as one possible "maintenance resolution". In this manner, and for any lifetime sustainment period chosen, we can simulate failures of the integrated system AND with the impact of maintenance. The impact of a maintained integrated system forever changes the failure characteristics of the integrated system(s).

Topic Notes

Prepared by:
Craig DePaul
DSI International, Inc.