From Theoretical to Practical: How to Cull Repair and Return Data to Provide Field Based Feedback on Quality and Reliability

Gay Gordon-Byrne
TekTrakker
gbyrne@tektrakker.com

Premise

Reliability Engineers use their skills to build products that will not fail when deployed in the field. This is because vendors are motivated to deliver products which are not overly costly to keep in service.

Customers are also sensitive to failure and want to purchase devices that provide the function they ordered without interruption for their functional useful life[1]. At the end of the design/build process, the results of all design projections and modeling of reliability are experienced by customers in the field.

The most obvious place to look for data to feed calculations of failure in the field is repair records. Customers do not call for repair for fun – they usually report a problem first to the vendor, who them does some problem determination before dispatching a technician. Only those machines which have been agreed by the vendor to need a repair result in a technician dispatch. Effectively collecting repair records should result in the ability to validate initial projections of quality and durability, as well as reveal any immediate surprises. Monitoring the continued repair cycle will add to the knowledge base about the physical life of the product and its components, and attention focused on those components posting the most frequent failures.

More easily said than done. In my experience of over 30 years in the field dealing with large IT vendor and user organizations, feedback programs have been hampered by common problems which, when resolved, will allow better use of such records.

The problems are not mathematical, but practical. Repair, in warranty, or out, is often managed by teams not under the direct control of the OEM (Original Equipment Manufacturer). Systems designed to efficiently dispatch technicians and parts were not envisioned as data warehouses and frequently lack any data quality field validation or standards. Many are not internally consistent with other systems. Technicians and support staff will have been diligently reporting on the repairs made but in haphazard fashion with the result that key information is buried in text fields.

If these inconsistencies could be eliminated or effectively mitigated, Reliability Engineers would be able to complete the cycle of evaluation from testing through field experience and back.

Why follow repair and return data

Pre-deployment testing has its limits. It is Murphy's Law that something unexpected will go wrong. No amount of predictive modeling trumps real-life. Ultimately, until equipment is deployed in the field no one is fully confident that their work has borne fruit.

[1] Useful life is in the eye of the user. With IT products in conventional settings, useful life is often a synonym for refresh cycles. The physical product may be capable of years, or even decades, of additional service, but often the application systems themselves are obsolete long before the hardware is no longer functional. From a reliability and quality standpoint, the user wants uninterrupted performance to match their view of useful life. This makes it exceedingly difficult to create a standard.

Manufacturers that can improve their feedback loop to validate and improve upon test results by using repair and return data will be able to focus on the real causes of failure and seek to avoid those problems in the next iteration of the product.

Repair and warranty return data is a ready repository of all hardware problems in the field. The labor side of hardware support is a major expense for OEMs and equipment downtime a major headache for the user. Equipment that works as planned does not need repair during the warranty period. Returns made under warranty are tracked under the Return Merchandise Authorization (RMA) programs and leave a paper trail for every item. Culling this data and matching it to the known population of devices in use allows the calculation of Mean Time Between Failure of the Population – a direct measurement of failure rate which is both simple and effective as a starting point for homing in on failures experienced in the field.

Component details are almost always recorded somewhere in the repair process and should be used as a feedback mechanism to design and manufacturing. Finding the details and correlating them to the device is often extremely difficult but the opportunity to cull existing records exists. Finding a way to utilize these reports opens the opportunity to answer many questions about failure including if reported problems are random, if there is a documented percentage of all problems recorded against a particular part, or even begin building the "Bathtub Curve" of both whole product and component life.

As an added benefit, customers perceive Quality as a combination of performance execution and lack of problems. In the realm of hardware, lack of problems means lack of service calls for repair, even if the repair is covered by warranty. If there is any organizational doubt regarding the value of tracking repair experience, managers in the sales department are always aware that paying attention to the repair and service needs of equipment in the field is an excellent resource for improving customer service and supporting customer loyalty.

Field Reporting of Failure

During the warranty period, most equipment failures result in a return to the OEM. This data is often culled for feedback to engineering, but is often also done poorly. A little discipline on the part of the OEM and their repair team can improve the quality of data to the point that it can be trusted as a picture of equipment and component failure in the field.

As an example of the difficulties of using field data, I was involved with a very large financial services firm that deployed thousands of IBM blade servers. The products were failing at roughly twice the rate of other models. Even though the repair contract was not held by IBM, the repair company took it upon themselves to dig into the cause and uncovered a thermal problem. They told IBM, who in turn was able to make changes in new products to avoid that particular problem, and also instructed their field engineers how to mitigate the problem with existing equipment.

I use the example of IBM to illustrate that even the largest of manufacturers with the largest direct field engineering force in the world does not get all of its valuable feedback from within its own ranks. The more a manufacturer is disconnected from repairs in the field, the more difficult the task to utilize repairs records for this very important feedback.

There are other systemic problems I have observed as well. My personal opinion is that companies that follow their repairs and returns are often limited not just by the difficulties of using repair data, but also by attitude. No one wants to admit failure, and our natural tendency as humans is to deflect blame.

As an example of the disadvantages of playing the blame game, I worked with a client to evaluate the causes of problems within their call centers as they felt they were paying excessively for repair contracts. Among the many products we tracked, there was a specialty network device that was posting repair activity at a very high rate compared with other products. I contacted the company and discussed the failure reports with their head QA engineer – who assured me they didn't have any problems. Everything was the fault of the end user. He entirely missed the point. The client was very unhappy and the product was in jeopardy of being removed, but he resisted the very concept that his product was imperfect.

This has been my experience all too often. The instinct is to diminish discussion of uncomfortable problems leads not to better reliability, but to unhappy customers. This particular vendor was willfully ignoring a high rate of return and refusing to do the deep dive to understand the problem so that the customer experience would be improved. Without the dive, they had no hope of making a correction – be it software or hardware.

Human Engineering Recommendations

Recommendation #1: Avoid the temptation to test equipment for "No Fault Found" and blame the end user. The equipment buyer didn't return the item because it worked for them, they had a problem. One of the best ways to get a picture of problems which are recurring and not just random is to track, carefully, all returns. If the records are being kept correctly, it should easy to spot trends within products as well as trends within customers.

Recommendation #2: Encourage the forthright analysis of field based failure reporting as a discipline over time from the moment of installation. (Infant mortality is a different metric[2]). Changes in repair rate over time can be an early warning system for systemic component failure which might otherwise be a surprise and a reputation killer. Moreover, one of my favorite axioms applies here. "Bad news does not improve with age" I have never seen a product which improves on its own. (Just as my teenage son's car did not magically lose its dents by ignoring them)

Data Quality Recommendations

Recommendation #1: Much of the important detail already captured by repair records is buried in text fields. Text is not only difficulty to cull, but the types of decisions made to exploit keyword searching and other locating techniques are themselves limitations on locating the full spectrum of repair detail. In my experience, the best reporting makes use of summary fields with restricted formatting and restricts the use of free form text to commentary.

This recommendation is difficult to implement because many systems have to be adjusted to demand data quality. Without being commanded to work with better standards, as with a requirement based on the Reliability Engineering needs of the manufacturer, then repair records will remain no better than electronic punch list for the Help Desk to delete.

The degree of detail that can be delivered back to the engineer is entirely at the mercy of the repair system design. If the system is written to allow technicians to enter anything they choose to write, the resulting data may be worthless. One of our clients was using a contract team for their desktop support and that

[2] In my experience, infant mortality is far more likely a problem of packaging and shipping than of reliability. Once a product is staged and tested by the end user deployment team, (current is applied), the overall failure rate is observable separate from shipping and packaging problems. Infant Mortality is also often mis-understood as the desire of manufacturers to make quick iteration changes, particularly interoperability and software changes, on a new application for products in the early stages of product life. This is product improvement, not infant mortality.

company had developed its own internal "slang" for problems which would crop up in repair records. We struggled to understand "BSOD" as a recurring repair, until we were told it stood for "Blue Screen of Death". Unfortunately, Blue Screen of Death is a symptom, not a repair, and we remained unable to resolve the repair records as planned.

Many companies are not capturing repair records in such a way that their own management knows which models of equipment were repaired never mind the part type or the part number. Computer systems are wonderful repositories of data, but without consistency the power of the computer cannot be utilized to make even the simplest of calculations.

Recommendation #2: Seek a way to connect the volume of assets deployed to the repair database. This is the missing link for producing a calculation of MTBF of the population, as without the known deployed population, useful calculations cannot be made. The distinction has to be made that products shipped may not yet be installed, and in the case of products sold through distributors, may be permanently disconnected from the deployed population. OEMs providing products strictly through distribution will find making this connection challenging. The most promise for OEM connection to the population is the repair contract "Entitlement" process whereby the warranty clock is started and the customer is reasonably assumed to have installed the equipment.

Sources of Repair Reporting

Within the warranty period, most OEMs are able to require reporting from the service teams under their control. This holds true for both badged employees and subcontractors. The types of information needed to extract field based reliability measurements are already being captured as a normal business process for the repair industry. Nothing new is needed, other than the attention to detail recommended above.

Even within the warranty period there are gaps in reporting which will need attention. Only the OEM can determine how large a gap they have in their warranty reporting. For example, a warranty program may be missing a percentage of products which are thrown away and not returned for replacement. This is more common with very low cost products than higher value units. Only the OEM can determine if the data gaps are too large to calculate good results.

The big problem comes at the end of the warranty period when repair reporting becomes scattered and the OEM loses the data stream from their warranty program. Repairs are still being made, just not reported through the warranty channel. There are different data sources to be culled to continue the evaluation of failure rate over time.

*OEM Extended Service Agreements***:** Many OEMs offer post-warranty or extended service agreements as a profit center. These service records are usually tied to a known population of devices since they are almost always administered to the serial number level. Where the OEM has direct knowledge of the repairs made and also the known population of devices under contract, the calculation of MTBF of the population and the component details should be easy to make. The proviso is always that the data be sufficiently well-organized.

Third Party Service: OEMs often offer service contracts and then dish off the physical repair to any of hundreds of independent companies known as 3rd Party Maintainers (TPMs) or Independent Service Providers (ISPs). Most such contracts are also written with a contractual obligation for a delay the client will accept before the technician and associated parts arrive. These are called "Service Level Agreements (SLAs)". Most OEMs have written their systems to use a time stamped ticketing process so that they can report back to the end user how well they met their time obligations. As a result of the feedback demand regarding time, the OEM sub-contract to the labor provider usually includes an automated data feedback program so that the ticketed problem can be closed out and the contract satisfied.

However, the data captured by the actual service provider is only as good as is required in the contract. Manufacturers can stipulate the level of detail captured by the direct or subcontract employee, but if the contract is lax or silent, the results are unlikely to be satisfactory. (As with the "BSOD" example.)

Board Level Repair: Behind the scenes of both the OEM and the ISP are companies that specialize in board level repair. These specialists are valuable because they can help a service provider (or a self-maintainer) keep a stock of tested spares ready for use over many years. OEMs can tap into this resource for themselves where they hold the repair contract. If this is not already being done, it represents another excellent source of component failure data.

Underwriters: Warranties are often sold by retailers under their own names such as Best Buys "Geek Squad". These warranties are supported by insurance underwriters and not by the OEM or the retailer. There is data held by these companies that could be tapped if the contracts were made to gain access.

Population Data Sources

The other half of the MTBF (Population) calculation is the population. There is no point in attempting to calculate failure rate without a known population. This is the major road block for using repair records. Repair records can be cleaned up, but there is no substitute for a known population.

Warranty &Entitlement: OEMs that connect the installation or acceptance of product into their warranty system are in an excellent position to cull the population size for each model. In cases where there is a multi-year warranty, even if some of the warranty period is covered by a different division of the company, the correlation is probably "close enough". As a practical matter, a small percentage of uncertainty will wash out in volume. So long as reporting is consistently "close enough", then the continuity of records would allow the calculation of a "Bathtub Curve" unique to the organization. (Better levels of consistency are needed to share reporting.)

Post Warranty Population: By focusing on the failure rate of a discrete population, the population does not have to be large to be useful. OEMs should not give up on calculating MTBF of the population just because they control the repair of only a small subset of their original warranty agreement. We found that by resolving all asset reporting into "Asset Months", the dataset builds rapidly over time and is also easily aggregated with other reporting on the same device.

For example, with a population of 100 of model ABC, over a 12 month period that is 1200 "Asset Months". An OEM supporting ten clients each with 100 units would then be seeing the repairs made against the equivalent of 12,000 units in the field for one month, or one device over 12,000 months. It has been our experience that very few IT devices post better than 240 months MTBF, so 12,000 asset-months of data will deliver a useful result. Since the result is population based, and the causes of the failures are known, the results can be used to spot the distribution of component failures between such things as hard drives, fans, memory cards, power supplies, etc. If failures were truly random, one would see a wide distribution of parts, but if failures were consistently with the hard drives, then it becomes clear where attention needs to be paid.

Limitations: Using field reporting is limited to the data at hand. Changing the types of data captured and asking repair organizations to adopt new standards may take years, if not longer. Using field data involves dealing with what can be learned and using that experience to the best of one's ability.

For example, we have discovered that very few repair organizations are collecting part number information consistently. When part numbers are recorded, they are often easily typographically flawed, and equally important, many distributors use different part numbers for the same part in order to be protected from price shopping. Rather than fight for better data (we always try) we prefer to resolve the category of part (Memory, CPU, NIC, Fan, etc.) and the type of repair (unit swap, adjustment, and part replacement) in order to capture the most that is possible from existing systems. Units which are swapped without a part type posted are lumped together as “unspecified” so that we do not lose any data and can still provide a result.

Summary: Using repair and service records kept in electronic databases can be a valuable source of data for reliability engineers to validate reliability projections. OEMs that are not fully in control of their repair activities can collect necessary data from their subcontractors. Other resources may be needed to help resolve the many syntactical and typographical challenges of preparing the data. Engineers need to be forthright and accepting of failure in the field as the best way to make continuous improvement.

About the Author

Gay Gordon-Byrne is Founder and President of TekTrakker®, originally built as a database of Information Technology (IT) hardware failure rates driven by repair records and now being applied to reliability questions in the emerging Smart Grid. Her experience with reliability (and lack thereof) is from the repair perspective where the quality of equipment as designed and manufactured is validated in the field. She believes firmly in the worn adage “One cannot manage what one cannot measure” (Lord Kelvin).