

WHITEPAPER

Requirements Traceability Benchmark

Executive Summary

Requirements traceability is a key aspect in requirements management software that has been around for decades, has been deployed by thousands of companies, and is required by most industry standards for complex product development.

The cost of failing to identify requirements related issues early in the lifecycle through traceability has been well researched, with INCOSE (International Council on Systems Engineering) and AVSI (Aerospace Vehicle Systems Institute) finding that late issue identification results in **16 to 100 times the cost**.

Since requirements define both product and services value and outcome, they are the only common language that ties together siloed business units (product development, marketing, development, legal, compliance, primary business unit, etc.) and the best way to track systems development progress and risks. Given how critical the product and services development process is for companies in today's world, one would think that measuring requirement traceability would be a top priority for management teams to reduce the risk of delays, cost overruns, defects, rework, audits, recalls, and failures.

And yet, no has ever measured traceability. The fundamental challenge to measuring traceability has been legacy tools that trap traceable information in siloes, preventing the measurement of traceability and the deployment of Live Traceability™ best practices across the lifecycle.

This report is the first measurement of traceability. Jama Software is the first to measure traceability thanks to our clients' participation in a benchmarking dataset of over 40,000 complex product and services development projects spanning Financial Services, Insurance, Healthcare, Telecom, Government, and more. All of this is made possible by our core product, Jama Connect®, which enables the largest community of end users using requirements management SaaS (Software as a Service) in the world.

To formally measure traceability, we have established the Traceability Score™. The Traceability Score measures the level of actual process adherence to the expected traceability model and can be used to compare performance across projects, teams, divisions, and companies. This score can also determine impacts to schedule, budget, cycle times, risk, and quality.

Top Performers in Traceability Score Outperform

Top quartile performers in Traceability Scores outperform bottom quartile counterparts significantly in measures of cycle time reduction and product quality. For example, top quartile performers successfully complete testing **2.5X faster** than bottom quartile performers and have more than a **2X advantage** on the rate of verified requirements.

Top 5 best practices of top performers

These business practices separate top quartile performers from the rest:

1. Focus on Live Traceability, not after-the-fact traceability
2. Integrate traceable data across best-of-breed tools
3. Make the Business Analyst function data-centric
4. Use model-based requirements to shorten and improve discovery
5. Manage by exception



Benchmarking Data



The main reason traceability has not been measured until now is the lack of available data. Legacy requirements management tools treated projects as unique documents with little or no data model consistency across projects nor a defined traceability model with relationship rules that continuously sync downstream activity to and from other best of breed tools.

As a result, there was no data available to measure traceability across a single project lifecycle, or across projects, teams, and organizations.

Jama Connect is the first and only requirements management SaaS to deliver Live Traceability via a traceability model that defines all traceable steps in the systems development process and continuously syncs data to and from best-of-breed tools. This enables Jama Connect to measure traceability at the project, product, division, and company levels. As part of the largest SaaS community of requirements management users, Jama Connect customers benefit from benchmarking their performance against aggregated and anonymized metrics.

This benchmarking analysis is based solely on aggregated and anonymized data from Jama Connect usage that was fully permissioned to be used for these purposes. This is the largest dataset ever assembled to analyze traceability. Here are some descriptive statistics for the dataset used for this benchmarking analysis:

- **50 million** unique product requirement items
- **6.7 million** unique test cases
- **2.9 million** item level status updates
- **360,000+** defects
- **40,000+** complex product development projects
- **6,000+** traceability models

Traceability Metrics

The first step to measure traceability is to define an operational definition of traceability that can then be measured. We start with the established definition for requirements traceability noted below and derive an operational definition that can be measured.

Requirements Traceability Definition

Requirements traceability is a well-defined concept. The definition, mainly based on the IEEE association definition, is included here:

Pecification, to its subsequent deployment and use, and through periods of ongoing refinement and iteration in any of these phases). In the requirements field, traceability is about understanding how high-level requirements – objectives, goals, aims, aspirations, expectations, business needs – are transformed into development ready, low-level requirements. It is therefore primarily concerned with satisfying relationships between layers of information (aka artifacts). However, traceability may document relationships between many kinds of development artifacts, such as requirements, specification statements, designs, tests, models and developed components. For example, it is common practice to capture verification relationships to demonstrate that a requirement is verified by a certain test artifact.



Operational Definition of Requirements Traceability

As the definition of traceability above clearly articulates, traceability is a method to manage the product or service development lifecycle through data to achieve a desired outcome. This process management framework is analogous to other process management frameworks such as statistical process control (SPC) in quality engineering and sales pipeline management in CRM (customer relationship management) systems.

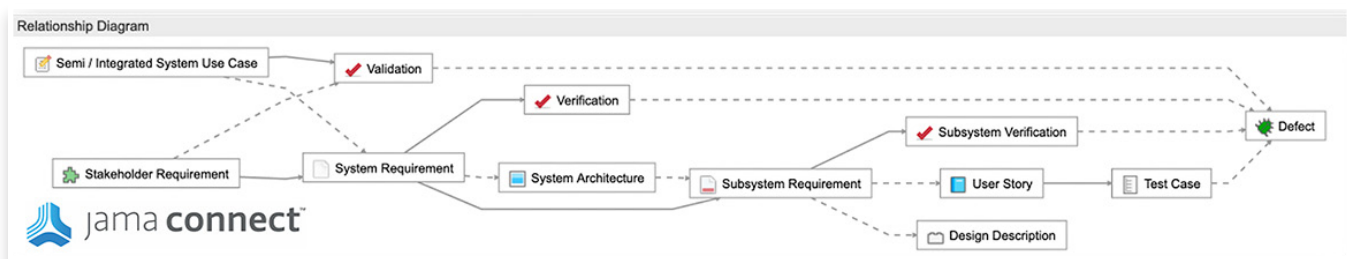
What these process management frameworks have in common is the following:

- Definitions of desired outcomes and failure modes
- A relational object to which all other process artifacts must link
- Defined elements that must be completed for each relational object
- Tracking mechanism to ensure elements are completed
- Measurement of successful attainment of desired outcome

For the product & service development process, the specific elements are as follows:

- Desired outcome is a verified and validated system with defects, rework, and failed tests as failure modes
- The relational object is a requirement
- Defined elements are the relationship rules in the systems development lifecycle
- The Traceability Score defines the percentage of expected elements in the model that have been followed
- Test status defines success or failure to reach the desired outcome

Combining these elements creates the traceability model – a complete definition of expected traceable relationships for each requirement in the project through the development lifecycle. An example of a traceability model in Jama Connect is shown below.



Traceability Score™

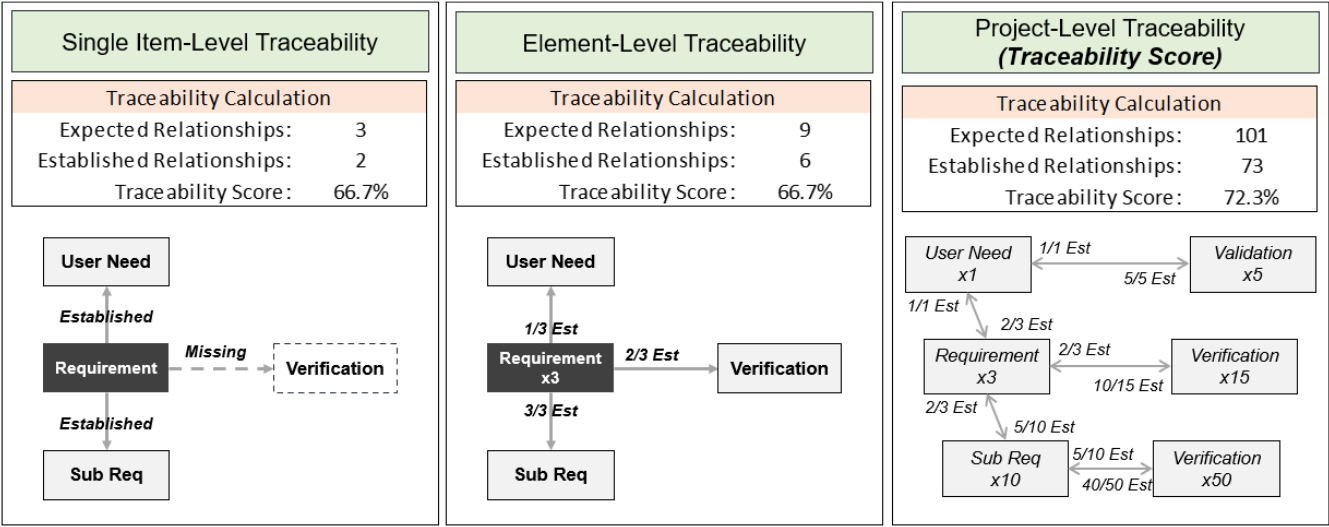
With our operational definition, we can now measure traceability and create a Traceability Score between 0-100%. This score can be calculated at any level in the traceability model (i.e., for an individual requirement, functional and non-functional requirements, system requirements, validation, test, etc.), but defaults to that of the entire project unless otherwise specified in this analysis. The formula for the Traceability Score is as follows:

Calculation

Traceability Score = # established relationships among model elements / # of expected relationships among model elements as specified by the project’s traceability model

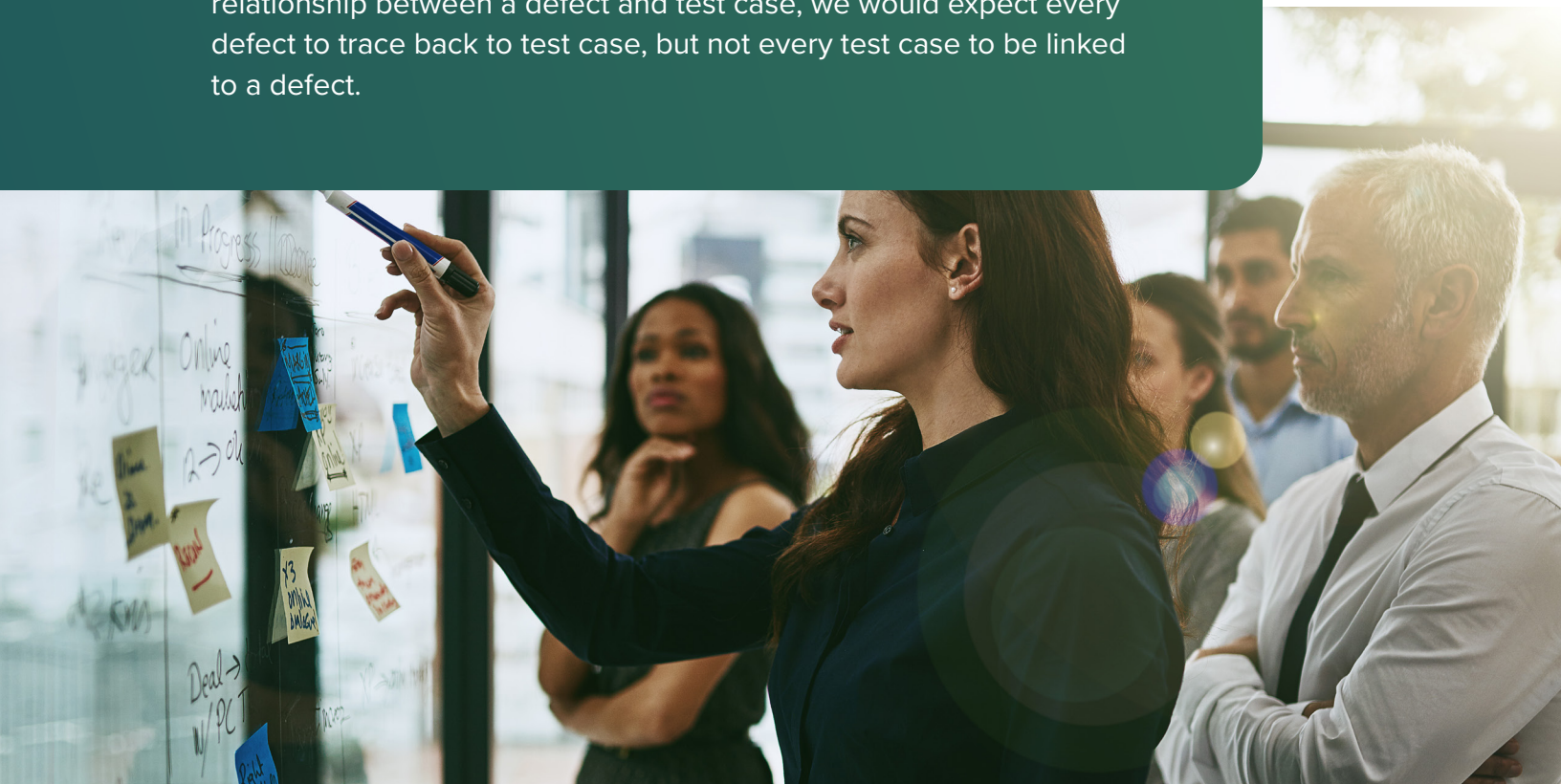
The following diagram provides an illustration for the buildup of the calculation:

- 1. At the individual requirement level, we can identify each expected relationship defined in a project’s traceability model (i.e., user needs defined by requirements, further refined by sub requirements, and test cases that should verify the requirement, etc.). We can then identify how many of these relationships have been established to get an individual requirement’s traceability.
- 2. As we go one level higher and measure traceability within a particular element type (e.g., user needs, requirements, tests, etc.) we can sum up the number of expected and established relationships across the set of items, giving us traceability at the element type level.
- 3. Finally, we can sum up the number of expected and established relationships across all element types, giving us the project’s total Traceability Score.



Details of score calculation

- By default, we exclude unused relationships from our traceability score (i.e., when no established relationships have been created for a particular relationship, we will exclude this relationship from the 'expected' relationships count as well). This allows for the Traceability Score to be more accurate in real-time operational use. We also ran the same analysis described below with a "strict" version of the Traceability Score counting all established relationships and the statistical results are similar comparing top and bottom quartile performers.
- When an element has multiple connections on the same relationship, this only counts as one established relationship (i.e., if a requirement is refined by five sub-system requirements, this still only counts as one expected relationship and one established relationship).
- In the case of a 'defect' model element, we only include upstream relationships in our calculation – as an example, if there is a specified relationship between a defect and test case, we would expect every defect to trace back to test case, but not every test case to be linked to a defect.





The focus of this research is to calculate the Traceability Score and its associated distribution across the dataset and determine correlation between Traceability Score and any positive and/or negative product or service outcomes. We explored correlations between Traceability Scores and defect detection, rework cycles, testing cycles, verification coverage, and item creation efficiency.

Normalization

To benchmark performance across projects, companies, and industries, we needed a definition of Traceability Score that followed a robust approach of actual behavior compared to expected behavior – which our definition provides. We also relied on Jama Connect’s configurable Traceability Model which allows each project to define the expected behavior. In this way, we were able to determine each project’s adherence to its own traceability model and compare project performance across different traceability models.

Performance Quartiles

With our analysis dataset established, we then performed the calculation outlined in the Traceability Score section of this paper for each project. We then calculated quartiles for these traceability scores across the dataset, and tagged each project with its associated quartile (‘Quartile 4’ denotes a traceability score in the 75-100% percentile, ‘Quartile 3’ denotes a traceability score in the 50-75% percentile, etc.)

Statistical Approach

In order to confirm the statistical validity of our findings, we leveraged two statistical approaches. For results based on comparisons between the means of two different traceability score quartiles, we used two-sample t-tests. These t-tests were one-sided and used an alpha value of 0.05 for confirmation/rejection of the null hypothesis. For results based on correlation between two variables, we used Spearman’s Correlation, as the relationship between variables often did not meet the criteria of having a strictly linear relationship. These Spearman Correlation tests used an alpha value of 0.05. For several of our correlation and hypothesis tests including Median Time to Execute Test Cases, Median Time from Test Start to Defect Detection, and Median Time to Identify the Set of Defects, we removed outlier datapoints using the IQR (Interquartile Rule) method.

Benchmark Results: Descriptive Statistics

This is the first time traceability has been measured and benchmarked. Here are some initial observations and interpretations looking across the dataset:

- **Median Traceability Score = 35%**
This low score is not unexpected given that traceability has never been measured before. As this report increases Traceability Score awareness and the impact on outcomes, we expect median scores to rise rapidly.
- **Top 10% of performers average Traceability Score = 87%**
Top performing companies are already achieving impressive Traceability Scores and seeing the benefits of fewer defects, delays, rework, and cost overruns.
- **Top quartile performers have a 7x higher average traceability score than bottom quartile performers.**
The performance variance between top and bottom performers is dramatic. The competitive advantage for top quartile performers is significant and sustainable unless bottom performers rapidly improve.
- **Within companies, project performance can vary significantly**
It is not uncommon for Traceability Scores within a company to range significantly as each project team has significant control over how their development process is managed. In fact, 57% of companies have projects in more than one quartile.



Correlations & Hypothesis Test Results

As a process management tool, the value of a Traceability Score is to quantify actual adherence to the specified approach. To determine best practices from the data, statistical tests were run to understand how differing levels of project adherence to Live Traceability can impact desired outcomes.

As we have shown, the Traceability Score measures actual adherence to the defined traceability model. Anecdotal evidence abounds to support this intuition, but the dataset has been lacking to conduct statistical tests to test this hypothesis. Using our dataset, we were able to determine that Traceability Scores exhibit statistically significant correlations to the following outcomes *and* rejected the null hypothesis that these correlations were purely random.

Outcomes

High-Level Finding	Specific Metric	Validated Hypothesis
Faster Time to Market	'Testing' Cycle Time	Median Time to Execute Test Cases (2.6X Faster)
	'Defect Discovery' Cycle Time	Median Time from Beginning of Testing to Defect Detection (5.3X Faster)
		Median Time to Identify the Set of Defects (2.1X Faster)
Higher Quality	Verification of Requirements	Pct Requirements with Verification Coverage (1.9X Higher)
		Pct of Requirements Verified with Passed Test Case (2.1X Higher)
	Test Case Success Rate	Initial Test Case Failure Rate (2.4X Lower)
		Final Test Case Failure Rate (2.9X Lower)

Faster time to market

The first three tests focus on how Traceability Scores impact cycle time. Do higher Traceability Scores lead to faster test case execution and defect identification? This is a fundamental value – that earlier detection of defects leads to fewer delays and much lower cost to correct.

We measured the following times below and noted performance improvements in top versus bottom performers of **2.1X to 5.3X**. Higher Traceability scores were found to lead to faster test case execution and defect detection having passed both of our statistical tests.

- Median Time to Execute Test Cases (2.6X faster)
- Median Time from Test Start to Defect Detection (5.3X faster)
- Median Time to Identify the Set of Defects (2.1X faster)

Higher quality

The last three tests focus on how Traceability Scores impact quality. Do higher Traceability Scores lead to a higher quality product? This is yet another key value – that a commitment to test case creation and execution leads to a higher degree of requirement verification and product quality.

We measured the following aspects of testing and verification below and noted performance improvements in top versus bottom performers of **1.9X to 2.9X**. Higher Traceability scores, having passed both of our statistical tests, led to more tests being completed and a higher percentage of passed tests.

- Percent of Requirements with Verification Coverage (1.9X higher)
- Percent of Requirements Verified (2.1X higher)
- Initial Test Case Failure Rate (2.4X lower)
- Final Test Case Failure Rate (2.9X lower)



Using the Traceability Scores to identify top performing projects, we were then able to determine the business practices followed to achieve top performance. Here are the top five most common best practices among top performers.

BEST PRACTICE #1

Focus on live traceability not after the fact traceability

Top performers understand that higher levels of traceability mean adherence to the process and better outcomes. This is often expressed as having moved from after-the-fact traceability to Live Traceability.

After-the-fact traceability occurs after the product has been developed and is typically a highly manual effort to try and re-create artifacts to demonstrate traceability that should have occurred during the development process but did not. This effort is undertaken solely for complying with industry standards and satisfying auditor requests for demonstration of process maturity.

Live Traceability occurs in real time as product and service development progresses to improve overall productivity (by ensuring business analysts and developers across disciplines are always working off the most recent and correct versions) and to reduce the risk of negative product outcomes (delays, defects, rework, cost overruns, recalls, etc.) through early detection of issues.

BEST PRACTICE #2

Integrate traceable data across best-of-breed tools

Live Traceability is only possible if you have the necessary traceable information integrated into a traceability model. Top performers have focused on integrating traceable data trapped in best-of-breed tools, desktop, homegrown tools, and spreadsheets into a traceability model managed by a requirements management system. This focus on traceable data has provided the means to achieve high traceability scores and manage the development effort by exception to achieve superior outcomes.

BEST PRACTICE #3

Make the business analyst function data centric

Top performers have a strong commitment to functions focused on business analyst process improvement that spans siloed teams. These companies have moved beyond lengthy meetings and emails as the only tools to determine status and ensure coordination among teams. Instead, the business analyst function deploys a data centric mindset and Live Traceability to ensure every analyst has access to the most up-to-date information in the system and to improve process performance.

BEST PRACTICE #4

Use model-based requirements to shorten and improve discovery

An important phase of any development project is discovery – in which meetings happen, learning progresses, hypotheses are tested, and requirements are generated. Top performers follow a model-based approach by consistently defining objects and establishing relationships among requirements. This model-based approach results in stakeholders and business analysts thinking early on about interaction, impact and how to test – a significant improvement over large, unproductive meetings on reviewing requirements documents without any connective thread.

BEST PRACTICE #5

Manage by exception

Business analyst management on top-performing projects focus on identifying exceptions to expected process behavior as early as possible and taking corrective action. This is a classic “management by exception” technique that goes back to the origins of statistical process control. This emphasis on process and identifying that the origin of most downstream negative outcomes (defects, rework, delays, cost overruns) is process exceptions (uncovered requirements, drift, missing test cases, etc.) occurring earlier in the engineering process.

Determine Your Own Traceability Score

The first step to apply these findings is to baseline your Traceability Score.

- Are you ahead or behind industry peers in Traceability Score?
- How do projects across your company compare on Traceability Score?
- How does Traceability Score correlate to outcomes specific to your company?
- How can you use the Traceability Score to manage the systems development lifecycle on a daily basis to improve quality and time to market?

For those that are not already Jama Connect users, a [Traceability Diagnostic](#) is available to provide a quantitative assessment of your current level of Live Traceability and offer recommended steps to improve.





Jama Software® is focused on maximizing innovation success in multidisciplinary engineering organizations. Numerous firsts for humanity in fields such as fuel cells, electrification, space, software-defined vehicles, surgical robotics, and more all rely on Jama Connect® requirements management software to minimize the risk of defects, rework, cost overruns, and recalls. Using Jama Connect, engineering organizations can now intelligently manage the development process by leveraging Live Traceability™ across best-of-breed tools to measurably improve outcomes. Our rapidly growing customer base spans the automotive, medical device, life sciences, semiconductor, aerospace & defense, industrial manufacturing, consumer electronics, financial services, and insurance industries. To learn more, please visit us at jamasoftware.com.