

DORA Metrics:

The Baseline For Engineering Excellence

As every organization has been accelerating digital transformation to support remote workforces and increasing demand for digital services, the brunt of that effort, of course, falls on engineering teams. While business reliance on engineering has grown, so has the demand for efficiency and accountability. To succeed in this environment, engineering leaders need to replace gut instinct with a data-driven approach to improve efficiency.

As engineering leaders try to build and run more efficient teams with less resources, they need insight into all aspects of software delivery across various tools to make data-informed decisions. Due to a lack of tools and quantitative metrics, many leaders resort to running their teams with intuition. They need more actionable metrics to effectively track the productivity of their teams and provide predictability for the business.

The 4 DORA Metrics

- ◆ Lead Time
- ◆ Mean Time to Restore
- ◆ Deployment Frequency
- ◆ Change Fail Rate

While lead time and deployment frequency measure software development velocity, mean time to restore and change fail rate measure the reliability of the software that is delivered. It's important to assess these qualities in tandem because software development velocity shouldn't come at the expense of quality or stability.

These four metrics provide a baseline for measuring the current tempo, rhythm, and responsiveness of an engineering organization.

What are DORA Metrics?

While engineering leaders have traditionally measured productivity by lines of code, code coverage, time spent on tasks, story points, and time estimates, these development metrics give teams a false sense of certainty, don't factor in external reasons and dependencies, and don't provide predictability or broader insights.

To determine the right software metrics to track the effectiveness and efficiency of software teams, the DevOps Research & Assessment (DORA) research program identified four key metrics that can be used to measure software delivery performance.



The DORA Metrics were published in what has become the de facto guide for software and DevOps leaders: "Accelerate – The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations"

by Nicole Forsgren Ph.D., Jez Humble and Gene Kim

Defining and Using DORA Metrics

Delivery Lead Time

What it is: The time from when development starts working on a task to the time the customer gets the feature. It is the time spent in implementation, integration, build, test, and deployment, and it does not include the time spent in backlog, prioritization, and design.

How to use it: Lead time helps organizations understand how quickly they can deliver software. With shorter lead times, teams can deploy to production in smaller increments, more frequently. This enables faster feedback on what is getting built and allows for quicker course correction. Conversely, longer lead times signify bottlenecks in the development process.

Mean Time To Restore (MTTR)

What it is: The time it takes to restore a failure in production, either an unplanned outage or a service failure.

How to use it: Measuring MTTR can be trickier than it appears because service failures and outages can be different types or severity, making the reporting of a single MTTR score incorrect and not valuable. For instance, in many organizations, minor and low-severity issues are never a priority to fix. Therefore, these low-priority issues should not be counted toward the final MTTR score.

Deployment Frequency

What it is: The number of times code or software is deployed or shipped to production.

How to use it: Depending on the type of application, for certain products it might not be possible or even necessary to ship code frequently to production. Software teams should evaluate the needs of their business and ensure that the velocity of their development process matches their business needs. However, the principles of lean and agile software development can still be applied by delivering software in small batches. As teams grow, it is critical to find a balance between how much and how often to deploy versus how stable the product is.

Change Fail Rate

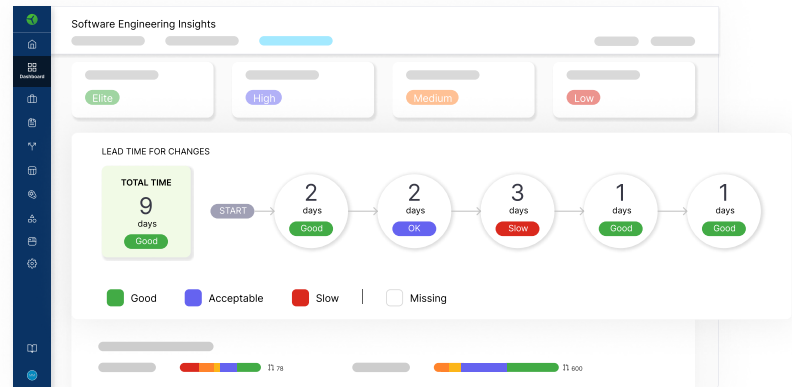
What it is: The percentage of deployments causing a failure in production, measuring the number of times a hotfix, a rollback, a fix-forward, or a patch is required after a software deployment or a service change. This metric is typically measured using data from the project management, version control, CI/CD, and incident tracking systems.

How to use it: Change fail rate can give organizations a sense of how frequently they are shipping code that causes issues. Ideally, the change fail rate should be as low as possible, indicating good quality code.

How Harness Software Engineering Insights™ Can Help

Harness Software Engineering Insights™ integrates with over 40+ tools across the DevOps toolchain to provide out-of-the-box software metrics and insights into the performance of engineering organizations.

Software Engineering Insights is purpose-built to address the engineering data challenge with sophisticated mechanisms to automatically manage the ever-growing plethora of data sources with good data hygiene.



Here's how you can track each DORA metric with Software Engineering Insights:

◆ Delivery Lead Time

Software Engineering Insights automatically correlates data across project management, SCM, CI/CD, and deployment systems, and provides accurate lead time information. It provides a detailed breakdown of time spent in each stage, so users can drill down and check which activity or step within the stage takes the most time to complete.

◆ Deployment Frequency

Software Engineering Insights is flexible enough to accommodate various types of deployment models and technologies. Whether the team being measured is pushing a package or container, Software Engineering Insights can measure deployment frequency.

◆ Mean Time To Restore (MTTR)

By integrating with more than 40 data sources, engineering leaders can measure specific types of failures across all their teams, giving them a much more standardized view of MTTR.

◆ Change Fail Rate

With Software Engineering Insights, the user can define the meaning of a failure, regardless of what system the failure is captured in. The user can also control the meaning of a hotfix, rollback, or fix-forward, whether it is in the SCM system or CI/CD tool.

Gain Actionable Insight with DORA Metrics

DORA Metrics are the foundation to understanding the efficiency and effectiveness of your engineering organization. With these foundational metrics established, engineering leaders can dig deeper into the root cause of slowdowns and quality issues, so they can make data-driven decisions and demonstrate alignment with the business and customer-centric outcomes.

Interested in learning more about how Harness Software Engineering Insights can help improve your engineering outputs?

[Request a demo](#)