

SOFTWARE PROJECT METRICS: DEFINITION, GUIDELINES, AND CONSIDERATIONS

Summary

While most people agree that metrics are important, choosing the right ones is a critical, but often very difficult process. This paper offers a set of guidelines and considerations when designing metrics for software, test, and documentation projects.

Introduction

Software project metrics are measurements, formulas, or constructs that help form raw data into patterns that can be used to improve the project outcome or process, identify trends, and compare the project to other projects or to earlier versions of the same project. They are often used to report project information to management as they often sum up areas of the project in a clear and easy to follow way.

Without them, it is often impossible to determine if a project is on track to meet important goals or deadlines. Furthermore, the effectiveness of improvements made to the process cannot be measured or compared.

The process of continuous improvement, where improvements made to each cycle of a software project make it better than the last relies on accurate, clear, and impartial measurements.

Metric qualities

Metrics fall into the following general categories:

Comparison metrics

Used to compare projects, or project elements, to previous versions of the project, or to other similar projects. The more similar these projects are to each other, the better and more accurate the comparison will be.

Tracking metrics

These are metrics that track the current progress of a project. For example, they might be graphs showing the amount of work completed or remaining, tables showing which components are complete, being worked on, or haven't been started. This kind of metric is generally more immediate than other types, and shows the current state of the project in the present, past, or very near term future.

Prediction metrics

These are metrics that show the expected state of the project, or project components at a future time. For example, they might be graphs showing the projected defect counts during an ongoing project, or customer defects over time for a project that has shipped. While prediction metrics are typically very “fuzzy” and subjective by nature, they are very useful to compare against the current tracking metrics to determine if things are going as expected and, if not, how far off they are.

Informational metrics

These are metrics that are show information about an area of the project, without an expected result. They are primarily used to generate discussion. For example, a pie chart that shows different categories for the current defects on a project can be used as an informational metric. By itself, it doesn’t mean much, but it generates discussion about why there were more defects in one category or another. This kind of metric is distinguished from a tracking metric by the lack of an expected right or wrong result for it. Many times informational metrics can be combined with each other to become a higher-level tracking or prediction metric.

Process metrics

These metrics are used to show the conformance or deviation of a project to an established process or standards. For example, if the process for a project calls for all defects to be fixed and closed 30 days after they are opened, the process metric for this might be a graph or table that shows which defects were opened each month, when they were closed, and a color-code to indicate which ones fell outside the 30 day window. If the process metrics for a project consistently show nonconformance, this may point to a problem with the process itself rather than the project it is measuring.

Most projects require metrics from all of these categories.

Recording the metrics

For metrics to be useful, everyone involved with the data collection and metric generation must be using the same data, and getting the same results. For example, when counting defects across the various components in a project, it is important that everyone involved in the counting has the same understanding of what constitutes a defect as opposed to a design change. If they don’t, the numbers will be off, and the metric will show incorrect results.

The only way to ensure common understanding is to fully document the definitions and process for deriving the metrics, as well as full definitions for the

data that feed into the metrics. It is important to document all of this in a common place and ensure that everyone involved reads and understands it before the data collection begins.

Do not spend too much time trying to get the terminology exactly right. It is more important to use the terms consistently than to find the right term.

Defining the metrics

To decide what metrics to use to measure a project, first decide what parts of the project are most important. Metrics that measure those aspects will also be very important. Start by breaking the project elements down into the following categories:

Things that must get done (key metrics)

These are the essential, core aspects of the project. If the product can't ship without the construction of five features, then measuring the number of features is an important metric. It is important to distinguish the things that must get done or the project fails from the things that are desirable, but which will not cause outright failure. By measuring the key project elements first, attention is focused on the most important part. If this part fails, the rest won't matter.

Things we want

It is human nature to produce and work toward more of anything that is measured in a visible manner. So, if we state that code features are important, we shouldn't be surprised when we come back a month later and there are a couple of hundred more code features, built at the expense of other aspects of the project that weren't measured. This is why it is necessary to identify those parts of the project that we really do want more of, and measure those. If we measure extraneous, unimportant things, we'll end up with more unimportant tasks consuming valuable project time to meet the metric. This problem can be mitigated by using multiple metrics in different areas, so that no single metric becomes the driver for the project activities and goals.

Narrowing the metrics down

A large list of metrics will require too much time and effort to collect and measure. Furthermore, if some of the metrics are not particularly useful, they will interfere with the message provided by the ones that are key. Because of this, it is often necessary to distill the list down to only the most important ones. Start with keeping all of the key metrics, and narrow down the number of others. To help with this, use the following guidelines:

Data that is easy to measure

Data and metrics are not the same thing. Data is used by the metric to produce meaningful information. It is important to choose metrics that do not require data

that is too difficult, time-consuming, or subjective. Otherwise, too much project time will be used to create and update them. The exception to this is a metric that is only expected to be updated once, or once a cycle. For example, comparing the number of total defects found during a product development cycle with the number of defects found in the field after it ships may be difficult, but it is only done once. In this case, the usefulness of the metric may outweigh the time and effort required to collect the data for it.

Things we want, not things we don't want

It is often easier to see the things that get in the way of a project than the things that move the project along. However, most people are happier and more productive working toward the things they want, rather than working *away* from the things they don't want. Because of this, it is better to measure the positive, desired outcomes instead of the impediments. The exception to this is anything that would compromise the core elements of the project and cause it to completely fail.

Limit the number of informational metrics

Informational metrics are metrics that don't have a desired outcome. Some of these are useful for generating discussion, but too many of these are distracting and waste time. Sometimes moving these kinds of metrics to the end of a project, after it ships, can make them more useful for planning, while less burdensome than they would be during the active production cycle.

Decide what form the metric should take when reported

Charts, tables, graphs, and summary text bullets are all possibilities for reporting methods. Making it look good is far less important than making it clear and easily understood. The 3D bi-modal bar chart may look impressive, but if your audience can't understand what it means quickly and easily, the usefulness is lost. Simple charts and tables tend to be easier to quickly comprehend than text.

Decide how often the metric should be updated and reported

The metric itself is the most important part, but if it is reported too frequently, or not frequently enough, the usefulness will be lost. Key metrics should be reported often, as a failure in one of these that is discovered toward the end of a project will be very expensive.

Some metrics should be measured weekly, and reported monthly. Some should be measured more often, but reported only when they fall outside of a range.

Metrics that are hard to determine, or which require hard to find data, should be reported less often.

The higher the level, the better the metric

The more data behind a metric, the more accurate it will be. This is because a single incorrect data point will have less influence if there are many other correct points to smooth out the result. For example, metrics at a component level, when rolled up to a project level, are often more accurate. Conversely, metrics that are taken down to the individual level are often very inaccurate and misleading. This is why metrics typically don't work well for measuring things like individual performance, but do work to measure overall productivity rates for an entire project.

Use several metrics, not just one

Multiple metrics work better than just one for the same reason more data works better than less data. If one metric is reporting a skewed result, the other metrics will smooth out the result. Five metrics that show everything is going well, and one that shows it is not, is a cause to question the metric, not the project.

Use metrics to praise, not punish

When used positively, metrics can inspire and motivate. For example, giving out awards or thanks for driving down defect levels increases teamwork. Punishing the group that had the highest levels will cause people to work around the metric and incorrectly report data. In the end, all metrics rely on at least some data collection. If people are scared of the metrics, they will find a way to manipulate the data.

Improve the metrics

Metrics are often used to make small corrective improvements to a project or process and then measure the results to see if it worked. Over time, this can lead to significant improvements. These corrective improvements should also be made to the metrics themselves. As more data is collected and the actual results are compared against the metric results, it should be possible to improve the metrics. Metrics that prove to be less useful over time can be discarded and replaced.

Conclusion

Metrics are an often misunderstood tool. Taking the time to carefully define appropriate metrics for your project or area ensures that the process of continuous project improvement is possible. When carefully defined and used, they can be effective, accurate, motivators of behavior.