

# TIOBE SOFTWARE SOFTWARE QUALITY GATES

---

An Introduction to Quality Gating

## Head office

TIOBE Software BV  
Victory House II Esp 401  
5633 AJ Eindhoven,  
The Netherlands  
Phone: +31 (0)40 400 2800  
Info@tiobe.com



**TIOBE**  
⟨ the software quality company ⟩

# SOFTWARE IS EVERYWHERE

Or as they say: software is eating the world. The huge demand for software means that there is also a huge demand for software developers. Since this resource pool of software developers is not unlimited, it is tempting to lower the bar and allow less skilled and experienced software developers to make the software of tomorrow. All this comes with a threat for the software world: the risk that the quality of delivered software is too low. Software bugs might have huge financial consequences or human lives might even be at stake.

Fortunately, there are ways to safeguard software quality. One of them is the introduction of software quality gates. At various stages of the software development cycle there are automated checks that will stop the software development conveyor belt if the quality appears to be too low. Great, but what stages need to have such quality gates and what needs to be measured? This brief article shows how software quality gates can be introduced in an organization to make sure that the quality of software stays at a high level.

## 7 STEPS

The overview to the right might seem like a lot of effort. Fortunately, TIOBE has paved the way for the first 5 steps. It has defined a set of metrics and tools to measure software quality based on the ISO 25010 standard. On top of that they have also defined a qualification model called the TIOBE Quality Indicator (TQI). [See the TIOBE quality indicator for more details.](#) In short, the TQI combines the results of all measured metrics to come to an indication of software quality. The results range from level A (very good quality) to level F (very bad quality). In the rest of this document we will focus on the last 2 steps: defining quality targets and enable quality gates.

- 1 **Select a quality model**
- 2 **Define software metric criteria**
- 3 **Select software metrics**
- 4 **Select tools to measure**
- 5 **Define qualification**
- 6 **Define quality targets**
- 7 **Enable quality gates**

# QUALITY GATES

---

Let's start with defining quality targets. If an organization is not quality minded yet, introducing quality targets should be done with caution. Be careful not to be too ambitious, because you only get one chance to do it right. An easy starter is to go for relative targets, i.e. stating that things should not get worse. This is something all stakeholders, even project managers with tight release schedules, can accept.

The most common first target we have experienced to work well in practice, is to allow no new severe coding standard violations to be introduced. The reason for this is that coding standards can be measured in a fast way and its violations are usually easy to fix. This means that it is an accessible and user-friendly way to get familiar with software code quality targets.

Once the team is ready for a new challenge, new targets can be defined, e.g. deciding that the code coverage is not allowed to drop under 50%, or prohibiting the code from containing any severe compiler warnings.

Quality targets are great, but without proper quality gating it is like trying to learn swimming by reading a book. A quality gate is passed if a certain quality target is met at a certain stage in the software development process.

The faster a quality target can be measured the closer the quality gate can be to the software developer. For instance, if you don't accept any compiler warnings you can put your quality gate just before developers deliver their code to the software archive or even before that, i.e. letting the build fail if there are any compiler warnings left.

Metrics that take more time, such as measuring the branch coverage of unit tests, can be performed during a nightly build and are typically executed after delivery.

# CONCLUSION

---

Defining the right targets and making sure they are checked at the right place in the software development process is not easy. We are happy to help you!

**Head office**  
TIOBE Software BV  
Victory House II Esp 401  
5633 AJ Eindhoven,  
The Netherlands  
Phone: +31 (0)40 400 2800  
[Info@tiobe.com](mailto:Info@tiobe.com)