



Unstable software tests cause issues that spread across projects

Researchers find that over half of projects in the popular OpenStack ecosystem are affected by shared instability from “flaky tests”

Fukuoka, Japan—In a study published in [IEEE Transactions on Software Engineering](#) on May 26, 2026, researchers from Kyushu University have found that “flaky tests,” which are unstable software tests that seem to randomly pass or fail, do not stay confined to the projects they originate in and often spread across entire ecosystems. After analyzing hundreds of interconnected projects in OpenStack, a widely used open-source cloud computing platform, the research team found that 55% of projects were affected by cross-project instability, resulting in a cumulative loss of 1,156 days of developer time.

Complex software systems, such as those used in cloud platforms, banking services, healthcare records, and government infrastructure, rely heavily on automated testing to ensure reliability. Each time a developer modifies code, automated tests run to confirm that nothing breaks. This process is known as Continuous Integration (CI) and allows software to evolve quickly while maintaining stability. Without it, even small errors could disrupt critical services that are used daily by millions of people.

However, not all test failures indicate real defects; “flaky tests” are a prime example. These tests behave unpredictably, passing in one run and failing in another without any code changes. As a result, developers are forced to spend time investigating false alarms and rerunning tests, requiring significant effort and computational resources. While companies like Microsoft and Google have reported high costs associated with flaky tests, most research has focused on individual projects. This leaves an important question unanswered: what happens in large, interconnected ecosystems where many projects share code, dependencies, and testing infrastructure?

In this study, the research team, led by [Assistant Professor Tao Xiao](#) and [Professor Yasutaka Kamei](#) from Kyushu University’s [Faculty of Information Science and Electrical Engineering](#), in collaboration with the University of Waterloo, Canada, as part of the Adopting Sustainable Partnerships for Innovative Research Ecosystem (ASPIRE) project, conducted a comprehensive analysis of the OpenStack ecosystem. They examined 649 projects, over 29,000 code reviews, and more than 73,000 code changes to understand how test instability behaves at scale.

The team found evidence of two key phenomena. The first is cross-project flakiness, where a single unstable test affects multiple projects. The second is inconsistent flakiness, where the same test behaves differently depending on the project in which it runs. In total, they identified 1,535 tests that caused failures across multiple projects and 1,105 cases in which flaky behavior varied across projects. Notably, around 70% of unit tests—which are typically designed to check small, isolated pieces of code—were found to exhibit cross-project instability, challenging assumptions about their reliability.

Importantly, the researchers found that instability was often caused by environmental and

system-level factors rather than problems in the code itself. These included timing-related problems in CI systems, temporary server problems or resource availability issues, mismatches in software dependencies, and inconsistencies in testing configurations across projects. Because many of these factors are shared across projects, flakiness can propagate widely.

As Kamei explains, “Our findings show that test instability is not a local issue but an ecosystem-wide problem. Addressing it requires coordinated efforts across projects, rather than isolated fixes, to reduce wasted development time and computational resources.”

The study also points toward practical improvements, such as standardizing CI environments, improving dependency management, and developing tools to detect and classify flaky tests early. These measures could help developers focus on real issues instead of repeatedly rerunning tests.

“Our work contributes to improving the reliability and efficiency of software development processes and paves the way for the development of intelligent, trustworthy testing infrastructures that support the growing demands of modern digital society,” concludes Kamei.

###

For more information about this research, see “Cross-Project Flakiness: A Case Study of the OpenStack Ecosystem,” Tao Xiao, Dong Wang, Shane McIntosh, Hideaki Hata, and Yasutaka Kamei, *IEEE Transactions on Software Engineering*, <https://doi.org/10.1109/TSE.2026.3685588>

About Kyushu University

Founded in 1911, [Kyushu University](#) is one of Japan's leading research-oriented institutions of higher education, consistently ranking as one of the top ten Japanese universities in the Times Higher Education World University Rankings and the QS World Rankings. Located in Fukuoka, on the island of Kyushu—the most southwestern of Japan’s four main islands—Kyushu U sits in a coastal metropolis frequently ranked among the world’s most livable cities and historically known as Japan’s gateway to Asia. Its multiple campuses are home to around 19,000 students and 8,000 faculty and staff. Through its [VISION 2030](#), Kyushu U will “drive social change with integrative knowledge.” By fusing the spectrum of knowledge, from the humanities and arts to engineering and medical sciences, Kyushu U will strengthen its research in the key areas of decarbonization, medicine and health, and environment and food, to tackle society’s most pressing issues.

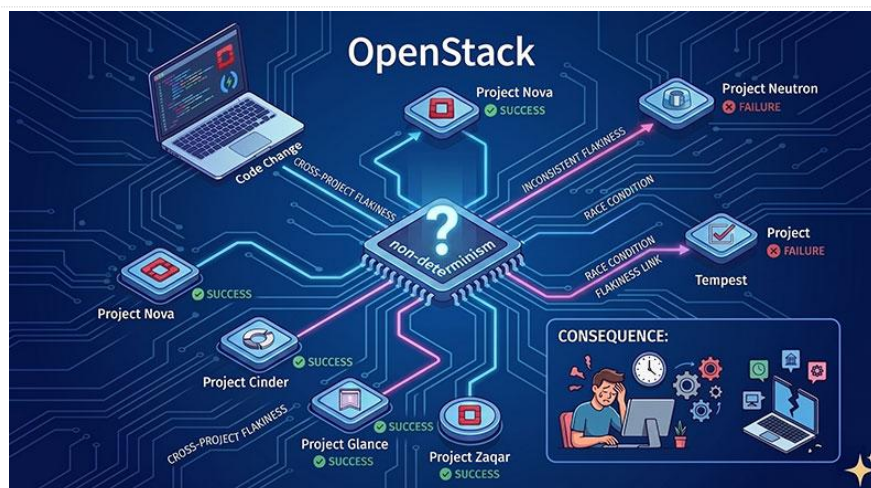


Fig.1 Flaky tests in the OpenStack ecosystem

Researchers identify cross-project and inconsistent flakiness in OpenStack and reveal that flaky tests are not confined to individual projects but propagate across projects within software ecosystems.

Source: Generated using Google Gemini

[Contact]

Yasutaka Kamei, Professor

Faculty of Information Science and Electrical Engineering

Tel: +81-92-802-3728

E-mail: kamei@ait.kyushu-u.ac.jp