Property-Based Testing in the Wild!

Exploring Property-Based Testing in Java: An Analysis of jqwik Usage in Open-Source Repositories

Introduction

Property-based testing (PBT) checks if the code's **properties** hold for a wide range of **generated inputs**. When a test fails, frameworks simplify the input by **shrinking** it, helping developers pinpoint the cause.

```
@Property(tries = 1000)
void sameSizeAfterReverse(@ForAll List<Integer> x) {
    assert x.size() == ListUtils.reverse(x).size();
}
```

Initially developed in Haskell, developers use PBT across languages for domains like automotive systems [1], OS kernels [2], and machine learning [3]. PBT aids in **bug detection** and **verification** and serves as **executable documentation**.

Research Questions

Properties:
RQ1. What sort of properties do PBTs generally check?
RQ2. How are these properties usually expressed?
RQ3. What role does PBT play in the project's correctness guarantees and bug-finding strategies?

Generators and Shrinking:
RQ4. How and when are generators implemented?
RQ5. In which cases is shrinking support explicitly added?

Methodology

We searched for **open-source** Java projects that use **jqwik** on **GitHub**, **SourceGraph**, and **Maven Repository**, based on stars and presence of jqwik dependencies or imports. We then applied **open coding** to analyze and classify test properties qualitatively through multiple review passes.

Results

So far, we have analyzed **seven repositories** and found **84 PBTs**, out of which:

- 60 can be decomposed,
- 33 use custom generators,
- 0 use custom shrinkers,
- 46 have filtered input, and
- 9 assert exceptions.

Other remarks:

- We found **four integration tests** and **no environment tests;** the rest focus on unit-level properties and functionality.
- In most cases, property-based tests represent **less than 2%** of the total tests.
- The **number of tries** per test varied significantly, ranging from 25 to 100,000, though most tests used the default 1,000 tries.
- Only two repositories used jqwik's **assumption feature**.

Label Distributions

- Mutation
- Invariant
- Round Trip
- Test Oracle
- Structural Induction
- Hard to Prove, Easy to Verify
- Different Paths
- Idempotence

Threats to Validity

- **Time constraints:** The 10-week duration of our project limited the number of repositories we could analyze, potentially making our results less representative.
- **Test complexity:** Some PBTs were difficult to interpret, which may have led to misclassifications or overlooked errors.
- **Repository limitations:** We restricted our analysis to open-source repositories, excluding larger projects that may rely more on jqwik or use it more effectively.

Conclusions

Despite its potential, property-based testing is **underutilized** in open-source Java projects, typically representing less than 2% of their test suites, indicating **limited adoption** and **impact**.

- **RQ1 & RQ2:** Developers mainly test properties related to **state changes** and **behavioral consistency**, focusing on categories like **mutation**, **invariant**, and **round trip**.
- **RQ3:** PBT plays a **minor, supportive role** in testing strategies, only being applied to isolated cases where it offers clear benefits.
- **RQ4:** Developers usually implement custom generators in combination with **input filtering** for classes **explicitly made for testing.**
- **RQ5:** Developers do not implement custom shrinkers, which suggests they find jqwik's **default shrinkers sufficient** or are **unaware** that they can create their own.

[1] T. Arts, J. Hughes, U. Norell, and H. Svensson, "Testing AUTOSAR software with QuickCheck," in *2015 IEEE Eighth International Conference on SoftwareTesting, Verification and Validation Workshops (ICSTW)*, Graz, Austria: IEEE, Apr. 2015. doi: 10.1109/icstw.2015.7107466.
[2] A. Santos, A. Gunha, and N. Macedo, "Property-based testing for the robot operating system," in *Proceedings of the 9th ACM SIGSOFT International Workshop on Automating TEST Case Design, Selection, and Evaluation*, Lake Buena Vista FL USA: ACM, Nov. 2018, pp. 56–62. doi: 10.1145/3278186.3278195.
[3] V. H. S. Durelli, R. Monteiro, R. S. Durelli, A. T. Endo, F. C. Ferrari, and S. R. S. Souza, "Property-based Testing for Machine Learning Models," in *Proceedings of the 9th Brazilian Symposium on Systematic and Automated Software Testing*, SBC, Sep. 2024, pp. 39–48. doi: 10.5753/sast.2024.3791.

Author: Harald Toth (htoth@tudelft.nl)
Supervisors: Andreea Costea, Sára Juhošová
Delft University of Technology
CSE3000: Research Project