

STILE: a tool for parallel execution of E2E web test scripts

Dario Olanas, Maurizio Leotta, Filippo Ricca, Matteo Biagiola, Paolo Tonella

Introduction

Automated end-to-end (E2E) Web testing, relying on frameworks such as Selenium Web Driver, is commonly used to assess the quality of web applications. However, the resulting test scripts may require long execution times, due to their interaction with the browser GUI and backend services. To avoid repeated and costly setup of the Web application state, testers tend to build test suites whose test scripts depend on each other (i.e., one test case sets up the application state expected by another test case). The presence of dependencies forces an execution order and prevents the use of techniques like **test parallelization**.

Goal

The goal of our tool STILE [1] is to enable the **parallel execution** of E2E web test scripts, while ensuring the compliance of all execution schedules with the dependencies among the involved test scripts, allowing a **significant execution time reduction**.

From the test dependency graph to the warranted schedules

Before running STILE, we must know all the dependencies in the test suite and represent them in a graph called **test dependency graph**, where the nodes are the test scripts and the edges are the dependencies between them. We can do it with test dependencies detection tools like **TEDD** [2].

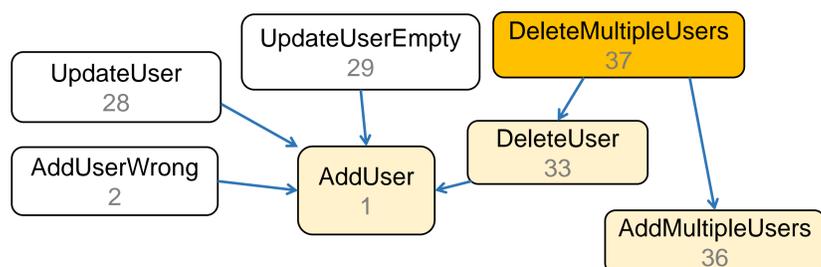
After that, we must extract from the graph a set of schedules, called **warranted schedules**, that:

1. Respect test dependencies
2. Contains every test script in the test suite

We can build a warranted schedule for a test script **t** by

1. Taking all the test scripts transitively reachable from **t** in the test dependency graph
2. Sorting them according to the original order of the test suite

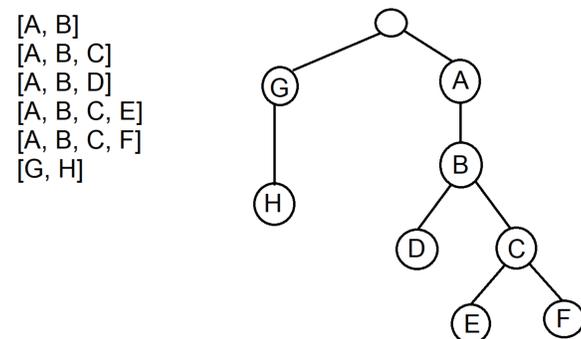
We can build a set of warranted schedules that contain every test script in the test suite by building a warranted schedule for each test script with no incoming edges in the test dependency graph



Part of the test dependency graph of one of our applications. Numbers represent the execution order of the test in the original order of the test suite. Highlighted nodes represent the warranted schedule for the test DeleteMultipleUsers

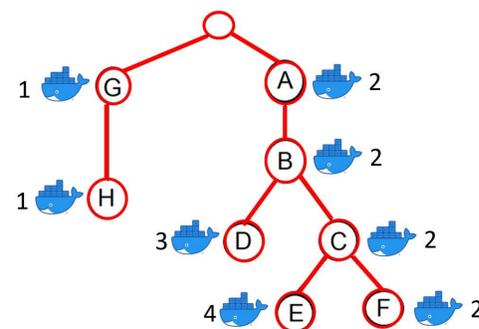
Efficient parallelization: from the warranted schedules to the prefix tree

Warranted schedules can be run in parallel, but it would be inefficient since they share many common prefixes. So, the solution is to represent them using a **prefix tree**.



STILE: parallel execution of the prefix tree

STILE runs the tests in parallel during a visit of the prefix tree. To avoid interferences between parallel executions, different branches of the trees are run against different instances of the system under test, that are created on demand using Docker.



Results

We evaluated STILE by running it on six test suites and obtained these execution time improvements, reaching a **maximum of 75%**:

Test suite	Sequential time (s)	Parallel time (s)	Time reduction (%)
Addressbook	1293	435	66.36%
Claroline	907	422	53.51%
Collabtive	1369	1031	24.66%
MantisBT	1028	261	74.65%
MRBS	936	510	45.55%
PPMA	706	483	31.55%

CONTACTS

Dario Olanas – dario.olanas@dibris.unige.it

REFERENCES

[1] D. Olanas, M. Leotta, F. Ricca, M. Biagiola and P. Tonella, "STILE: a Tool for Parallel Execution of E2E Web Test Scripts," in Proceedings of 14th IEEE International Conference on Software Testing, Verification and Validation (ICST 2021), 12-16 April 2021, pp.460-465, IEEE, 2021. DOI: 10.1109/ICST49551.2021.00060, ISBN: 978-1-7281-6836-4.

[2] M. Biagiola, A. Stocco, A. Mesbah, F. Ricca, and P. Tonella, "Web test dependency detection" in Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, ser. ESEC/FSE 2019. New York, NY, USA: Association for Computing Machinery, 2019, p. 154–164. DOI: 10.1145/3338906.3338948, ISBN: 978-1-4503-5572-8