Automatic test-generation technologies to limit excessive testing work in agile software development
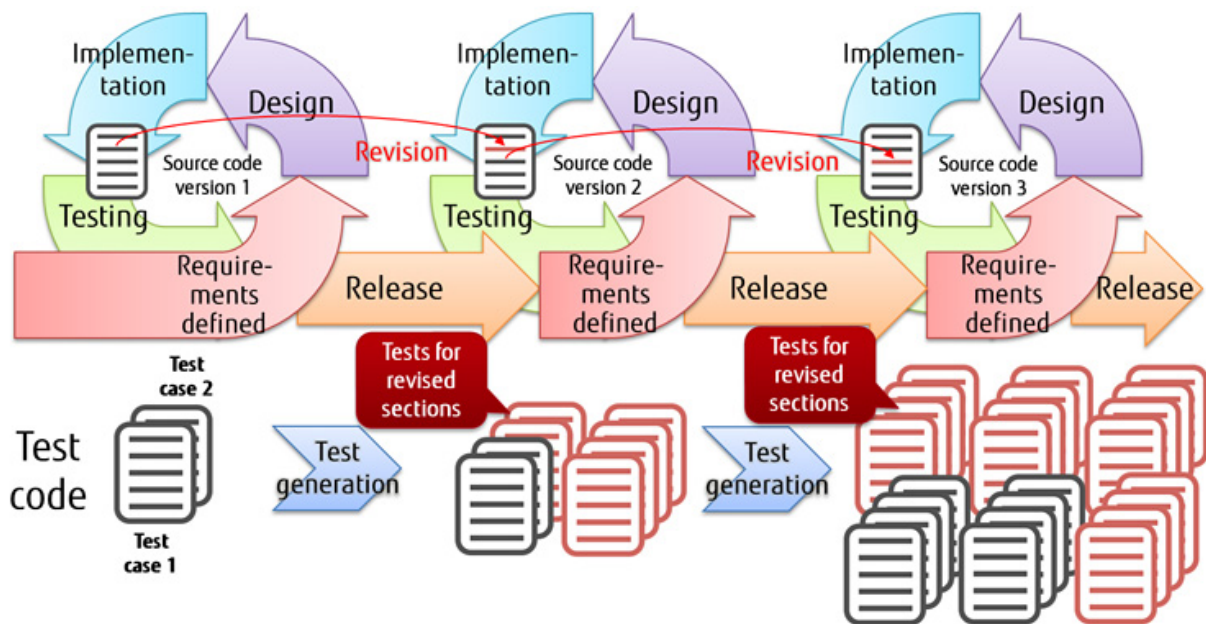
March 30 2016

Figure 1 In the case agile development is implemented with previous technology

Fujitsu Laboratories today announced the development of automatic test-generation technologies to limit excessive testing work and enhance efficiencies for agile development - an evolutionary approach to developing software which incrementally adds small-scale functions, and then one after another, makes them available. Automatic test-generation

technologies analyze source code and generate new test cases to exhaustively test software.

However, under an agile development approach, frequent, repeated changes to the source code result in a large number of test cases and excessive testing work. Fujitsu Laboratories have now developed automatic test-generation technologies that can efficiently search for the variables in source code that impact the sections of code where new software features are added, and locally revise and add to portions of the tests at the variable level. In a trial that applied these technologies to a revision of open source software with approximately five thousand lines of source code, Fujitsu Laboratories confirmed that it was possible to dramatically reduce the increase in test code to 1/24th of previous levels. This will enable efficiencies in development through test automation, which was previously difficult to apply to agile development. It will also greatly reduce increases in testing time caused by repeatedly adding functions in a brief period, thereby shortening overall development times.
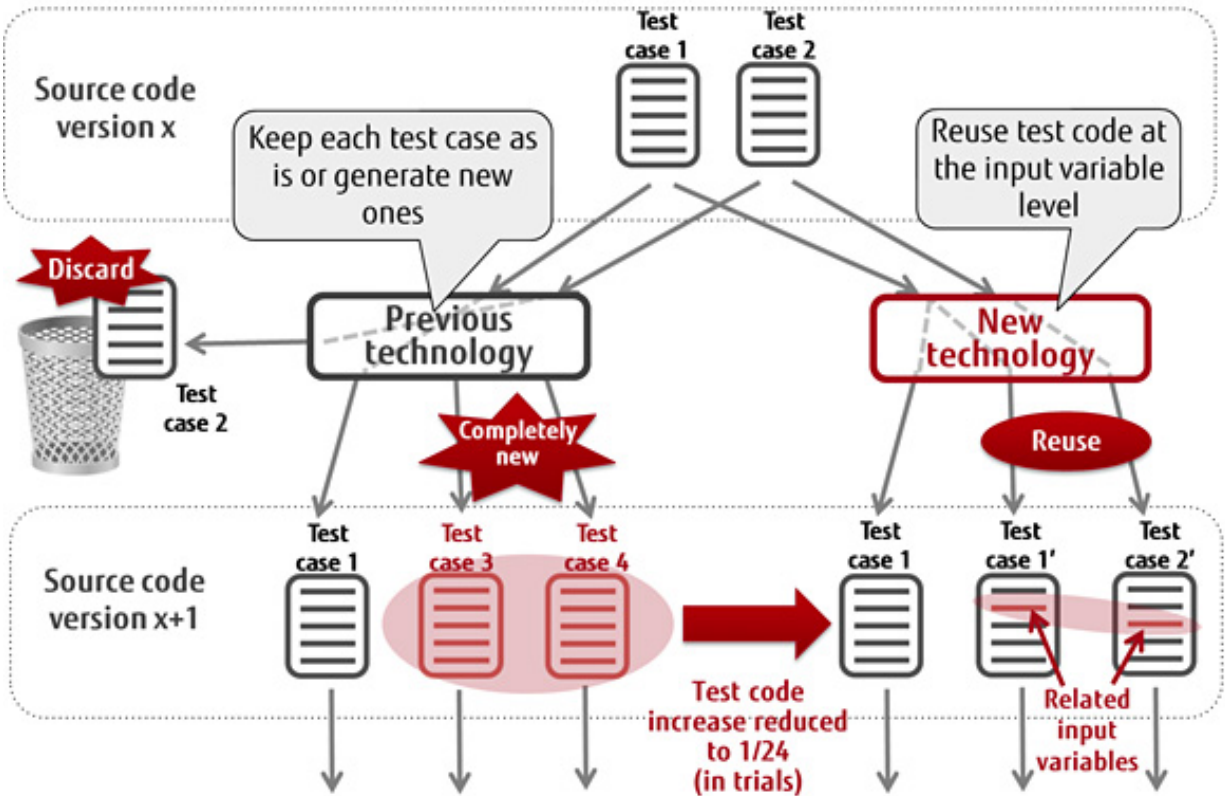
Development Background

Figure 2 Test code regeneration technology

Agile development is advancing as an approach to software development that can quickly and flexibly respond to changes in requirements and specifications by adding, in a short period, small-scale functions, and then making them available one after another. Fujitsu Laboratories are pioneering the development of automatic test-generation technologies for agile development, which automatically and exhaustively test software by analyzing source code.

Issues

When source code enhancements are made, with conventional automatic

test-generation technology, there may be cases where previously available test cases cannot be used to test the enhancement. In that event, it is necessary to generate and add a new test case. This means that in agile development, where revisions are frequent, the number of added test cases becomes excessive, and tasks such as handling test cases, checking test results, and test case maintenance become a problem (Figure 1).

Newly Developed Technology

Now, by efficiently analyzing the relationships between input variables and the revised sections of the source code, so that test code can be reused at the input variable level, Fujitsu Laboratories have developed automatic test-generation technologies that limit the increase in the number of test cases due to source-code revisions. The newly developed technologies have the following features:
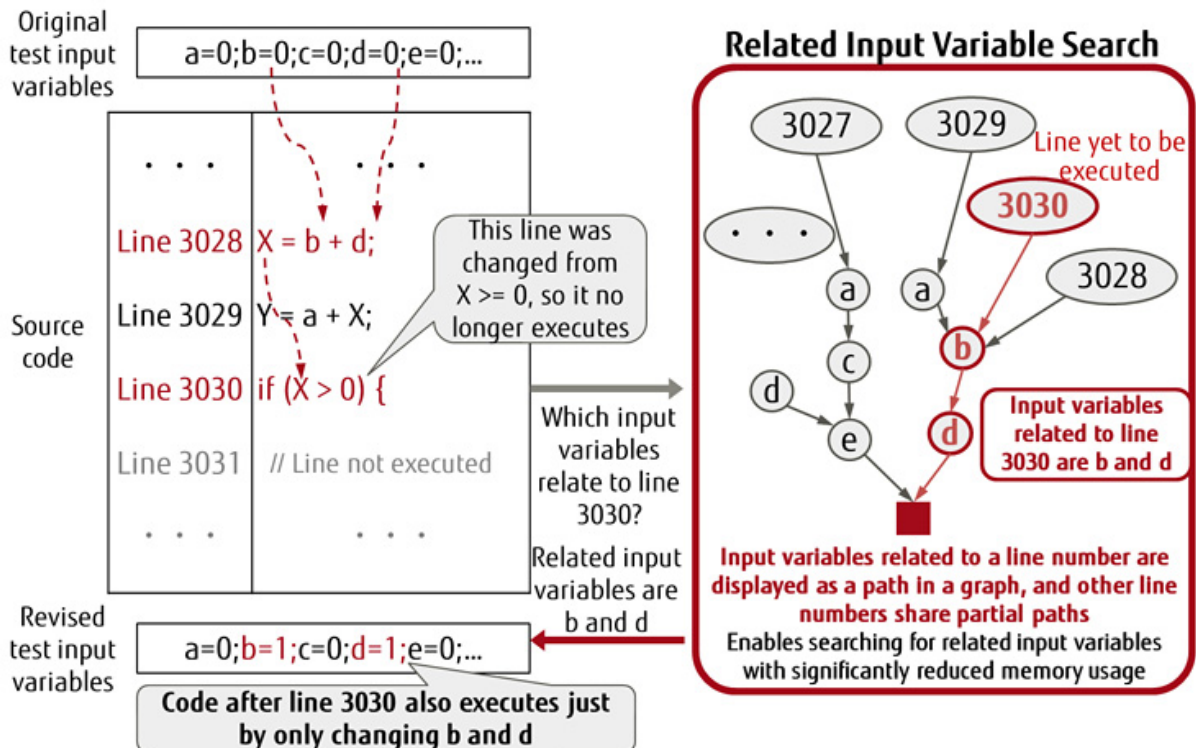
Figure 3 Related input variable search method

1. Technology that reuses test code at the input variable level

By revising the test code only for input variables that are related to the changed sections of source code, Fujitsu Laboratories have developed test-code regeneration technology that can limit the increase in test cases (Figure 2). With previous technology, in response to source-code revisions, each test case had to be reused as is, a newly regenerated test case had to be added, or the original had to be replaced, and all such choices had to be made at the test case level. With this newly developed technology, as the test code is rewritten at the individual input variable level, it is possible to regenerate test cases efficiently.

2. Technology to efficiently find the input variables related to a line of source code

Identifying the input variables related to revised sections of source code requires a large amount of memory (Figure 3). Fujitsu Laboratories developed a technology that can find the input variables related to a source code line while greatly reducing the amount of memory used, by representing the input variables related to a line of code as paths in a graph and partially sharing these paths with other lines of source code.

Effects

In a trial that applied this technology to a revision of open source software of approximately five thousand lines, it was shown to reduce

the increase in test code to 1/24th of previous levels. With this technology, it becomes possible to use automatic test-generation technology with agile development, shortening development times.

Future Plans

Fujitsu Laboratories will enhance this technology through internal trials, aiming for commercialization in fiscal 2016.

Provided by Fujitsu