froglogic

Squish Coco

Code Coverage

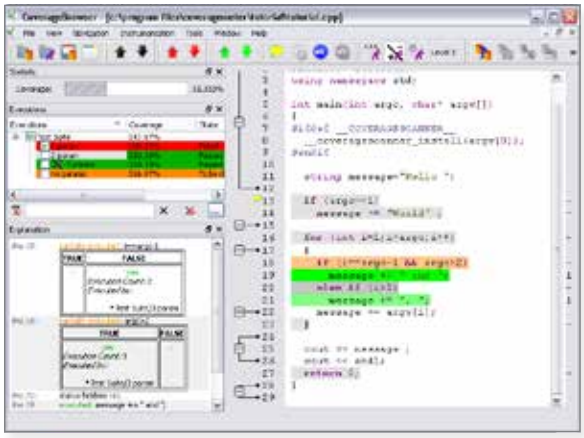
A complete, cross-platform code coverage toolchain for C, C++, C# and Tcl applications.

Squish Coco's code coverage toolchain provides wrappers for standard compilers to build instrumented versions of applications – with no source code changes necessary. Tests executed against an instrumented application produce data which Squish Coco analyzes. The analysis ensures testing coverage is achieved by:

- ☑ Discovering untested sections of code
- ☑ Identifying redundant tests
- ☑ Finding dead code

Squish Coco also:

- ☑ Determines the optimal test execution order, maximizing code coverage in a given time frame.
- ☑ Helps maintain coverage levels between releases: Comparing coverage differences between two application versions and identifying where new tests are needed.
- ☑ Provides advanced coverage analysis, merging and analyzing multiple execution reports.



*"Squish Coco is a test engineer's equivalent to a programmer's debugger, only instead of helping to **understand** bugs, Squish Coco helps to **reveal** bugs by highlighting untested statements."*

Sébastien Fricker, Lead Developer

Supported Platforms:

- Windows (32- and 64-bit)
- Linux (32- and 64-bit)
- Mac OS X (32- and 64-bit)
- Embedded Operating Systems
- UNIX (Solaris, AIX, ...)
- ... Inquire about additional platforms and custom toolchains

Use Squish Coco at every stage of testing:

- ☑ Unit
- ☑ Automated
- ☑ Manual
- ☑ Etc.

C++ code coverage analysis when the addressbook application is passed the `-h` (help) command line option.

```
1  int main(int argc, char **argv) {
2    QApplication app(argc, argv);
3    if (argc == 2) {
4      QString arg(argv[1]);
5      if (arg == "-h"  ||
6        arg == "--help") {
7        QTextStream out(stderr);
8        return 1;
9        out << "usage: "<<  argv[0] << "\n";
10     }
11   }
12   MainWindow mainWindow;
```

```
1   Executed code
0-1 argc == 2: was never false
1   Executed code
0-1 arg=="-h": was never false
0   arg=="--help": was never true or false
1   Executed code
1   Executed code
X   Unreachable code
-
-
0   Never executed
```

Result Summary

Lines executed:	4
Lines not executed:	2
Lines of dead code:	1
Expressions not covered:	3
Decision branches not executed:	2

The Squish Coco Toolchain:

- **CoverageScanner** analyzes and instruments C, C++, C# and Tcl applications.
- **CoverageBrowser** displays and manages coverage analysis data and results from a sophisticated GUI. (Available to commercial licensees only.)
- **Microsoft® Visual Studio Add-in** generates code coverage configurations for all available C, C++ and C# projects within the Visual Studio IDE.

Supported Coverage Levels:

- ☑ Function coverage
- ☑ Line coverage
- ☑ Branch Coverage
- ☑ Branch, Decision and Condition Coverage

More Information:

http://www.froglogic.com/squish/coco

contact
sales@froglogic.com
www.froglogic.com