Regression Testing in Plan 9: show us where it hurts with SpinalTAP

John (EBo) David

IWP9 2010 Seattle WA

SpinalTAP

- Regression Testing and Test Driven Development.
- Why Automate?
- The Testing Anything Protocol (TAP)
- SpinalTAP API
- TAP Extensions
- SpinalTAP Utilities

Regression Testing and Test Driven Development

- A software testing method which seeks to insure that new bugs were not introduced when making a change
- Include eXtreme Programming (XP), Agile, Test-Driven Development (TDD), ...
- Write tests first
- Automate and summarize

Why Automate?

- More than 50% of a projects life-cycles is in debugging and testing
- Bug fixes take 33 to 40 times as long
- Additional 15% to 30% investment results in 40% to 90% reduction in defect density
- Test suite itself is an asset

The Testing Anything Protocol (TAP)

- Developed by Larry Wall in the mid 80's for Perl language
- Separation of Producers and Consumers
- Simple
- No formal specification

TAP stream example:

```
Optional:    TAP version 13
Required:    1..3
             # some diagnostic
             ok 1 - Input file opened
             ok 2 - First line of the input valid
             not ok 3 - Read the rest of the file # TODO reason
```

Example TAP Summary

```
Test Summary Report
-------------------
t/iterators.t (Tests: 92 Failed: 8)
  Failed tests:  7-13, 15
t/nofork-mux.t (Tests: 6 Failed: 0)
t/regression.t (Tests: 4794 Failed: 103)
  Failed tests:  2, 5, 31, 34, 58, 61, 85, 88, 114, 118, 145-146,
                171-172, 200-201, 226-227, 252, 255, 278-279,
                308, 312, 338, 342, 368-369, 395-396, 422,
                425, 452, 454-455, 481, 484, 509-510, 538-539,
                563, 567, 593, 597, 623, 627, 653, 657, 683-684,
                686, 690, 716, 720, 746, 749, 775-776, 803-804,
                831-832, 835-837, 866, 870, 896-897, 923-924,
                926-927, 929, 955, 958, 984, 987, 1013-1014,
                1040, 1043, 1069, 1073, 1099, 1102, 1126-1127,
                1129, 1133, 1159, 1163, 1189-1190, 1192, 1196,
                1222-1223, 1226-1227, 1253, 1257

Plans=47 Tests=9370
Result: FAIL
```

SpinalTAP API

- plan(int num_tests); lazy_plan();
- ok(int bool, char *msg,...);
- diag(char *msg, ...);
- skip_next(char *reason,...);
- skip_block(int num_tests, char *reason, ...);
- skip(“reason”,

  ```
    ok(1,”test 1”);
    ...
  );
  ```

SpinalTAP API

- todo_next(char *reason,...);
- todo_block(int num_tests, char *reason, ...);
- todo(“reason”,

  ok(1,”test 1”);

  ...

  );

SpinalTAP API

- eq(void *a, void *b, int(*comp)(void *a, void *b), char *msg, ...);
- bail_out(char *reason, ...);
- die_on_fail(); restor_fail();
- pass(char *msg, ...); fail(char *msg,...);

SpinalTAP Extensions

- NAME test-name
- output_file(char *name);

SpinalTAP Utilities

- Prove – search for and run tests [WIP]
- TAPdancer – TAP stream parser
- TAPestry – history [WIP]
- Faucet – synthetic tap generator

Summary

- Regression testing is an important and valuable software engineering practice
- SpinalTAP provides a language and system agnostic testing tool for Plan 9

Acknowledgments

- Andy Armstrong, Gaurav Vaidya, and Curtis “Ovid” Poe for helpful suggestions and discussion of the IETF draft specification
- Devon H. O'Dell for introducing me to TAP
- Ron Minnich for being my GSoC mentor
- Erik van Hensbergen for point out that the drummer always dies:

```
#define drummer(X) bail_out("The Drummer always DIES! This one died %s",X);
```