Test Driven Development

Steven Bucksbaum, February 5, 2011

Quantitative Development Approach

Develop Test Harnesses as Code is advanced: Test code and associated program code are developed at the same time. The increasing number of test methods is a measure to management that program coding is moving forward. Thus, coding is progressing.

Management Reports:
1. Number of tests coded
2. Number of test run.
3. Number of successful and number of failed tests.
4. Number of tests run should grow over time, as tests are coded.

Development approach:
1. Developer thinks of the code or the functionality to be coded.
2. Designs the test code
3. Designs the program code
4. Runs the test
5. Re-factors program code to correct test results.

Automated testing of this kind provides data for reporting to management:
1. As program code expands, the number of test methods also expands. Report the total number of test methods to management. Over the time, the number should grow.
2. As formal program builds occur, the number of tests run grows.
3. With each formal program build, run the tests. The out come of each test, success or failure, is placed in a database.
4. Reports are generated of failed tests and successful tests. These numbers are reported to management on a periodic basis.
5. Engineer responsible for the code/test reviews both the program code and the test code to determine the error and then corrects the error.

Continual Testing:
1. Find errors sooner
2. Provides quantitative data for reporting to management
3. Gives management data that coding is progressing as the number of tests written grows over time.
4. Improved software quality

Disadvantages:
1. Developers must write more code. Test code and program code.
2. Database and web site infrastructure necessary to support the testing effort.