

# Functional Test Automation for Real-Time Embedded Software

Embedded software is at the heart of industrial innovation. Many functionalities of a product, be they genuine novelties or decisive improvements, make it possible to take or keep an advantage on your competitors.

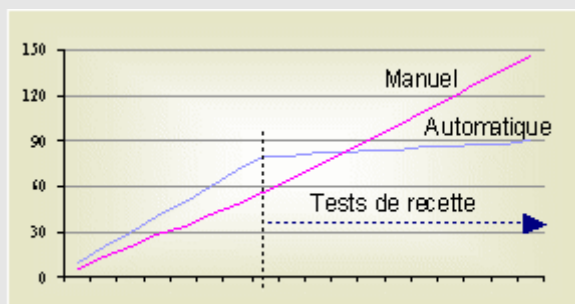
Let us take the example of the car industry. Many functions originated owing to the massive use of small embedded systems: GPS, ABS, speed regulator, etc. Embedded software is present in every segment of the economy and no one can figure out the limits of its expansion.

The increasing complexity of the applications, combined with the need for reducing development time, requires the use of modern methods of design, development and test. Software testing is undoubtedly a most tedious task. Nevertheless, a quick look at the breakdown of project expenditures shows that testing roughly accounts for an amount between 20 and 50% of the overall costs.

## TEST AUTOMATION

The validation of embedded software is mainly carried out manually. However, the advantages of automation are numerous in term of quality, cost reduction and development cycle time shortening. The covered fields, the accuracy and the repetitivity of the tests are pledges of better quality. The speed of execution 24 hours a day, 7 days a week, shorten the test duration at a critical phase of the development cycle. The automated testing is "at no cost" at the time of its execution, at the opposite of the manual method which generates a non-recoverable direct cost. Moreover, automation transfers a part of the effort of manual test upstream of the development cycle into a scripting activity, decreasing in a mechanical way the total duration of the development.

The main parameters for estimating the possible profit of automation are: the initial investment, the development cost of the test suites, the duration of manual testing operations compared to the duration of the same automated operations, and the number of times the test suites are executed. The curve opposite shows an example of comparison of the test efforts in the manual and automated configurations.



### Comparison of the Automated and Manual test efforts

The diagrams of effort in manual (pink) and automated (blue) modes cross in a point which depends on the application under test. This intersection can be during the 2nd passage of the whole set of validation tests. The profit, in term of man-days, grows then very quickly. It is often needed to execute validation tests several times, because errors are discovered at the time of validation testing and during the next phases. Automation appears particularly effective then.

## FAVOURABLE SITUATIONS

What are most suitable applications ?

The classes of applications providing a strong potential have at least one of the following characteristics:

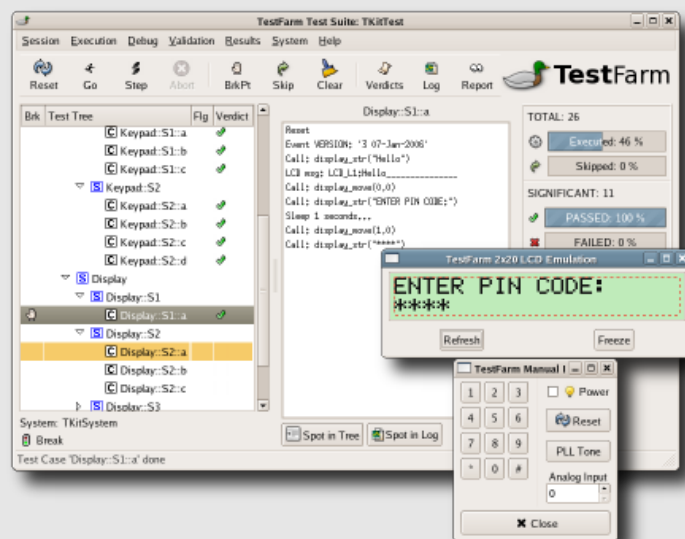
- High Reliability of operation required: test coverage and reliability of the results.
- Use of an incremental development cycle: re-use of the test suites.
- Application porting from a platform to another: re-use of the test suites.
- Development of a product line with similar functionalities: re-use of the test suites.
- Large scale diffusion: need for high test coverage and multiple test suites executions.
- High testing culture: sophistication of the scenarios of test.
- Reduced Time to market: speed of carrying out the test at any hour.

What types of test should be automated first ?

The answer is naturally very dependent on the type of application considered. However, generally, the following types of test are in essence of good candidates:

- The tiresome tests which claim a constant attention from the operator;
- The repetitive tests which, for example, differ from a value of parameter;
- The tests which require a complex instrumentation;
- The tests which are not possible otherwise;
- The tests that take a long time to run; the tests requiring high of stability and reliability;
- The tests that check time responses in an accurate manner;

Execution of a Test Suite on the **TestFarm**. platform.



## THE BENEFITS OF TEST AUTOMATION

---

Quality improvement of the delivered software. Software quality improvement covers many aspects of which much is difficult to quantify. How to determine the number of defects that automation made it possible to detect before the field test? Which profits bring the "repetitiveness" of the tests? These are complex questions, although it is possible to agree on the fact that indeed, the delivered software will have less defects. Automation makes it possible to stress a software application over a long period of time at nearly no cost. Such an operation would not be also thorough manually.

Reduction of the testing workload

- The execution time in the automated mode is generally shorter. It comes from the complexity of the testing procedure and the number of parameters to be captured at the same time. A factor 5 between the durations of the automated and manual testing is very often seen. This ratio, very dependent of the application, is advisable to estimate. To achieve that, it is enough to time the manual realization of a test, including the initial conditioning and the valued recording of the result. As a second stage, one estimates the duration of the same operations performed by a machine.
- The automated testing is carried out 24 hours a day, 7 days a week. By taking the example of a series of tests requiring the workload of 10 man-days for an operator testing 7 hours per day, it will be carried out, if automated, in a little more than one day (factor 5). Indeed the number of hours "machine" is of  $10 \cdot 7 / 5 = 14$  hours. One can also use several test workstations in parallel, thus reducing as much the overall testing duration. Moreover, if the execution of the series of tests is launched at 6 p.m., the test report is available the next morning at 8 a.m.

Development cost reduction

- An operator producing a series of tests corresponds, in financial terms to a direct cost of some man-days. Comparatively, a click on an icon is enough to launch the execution of test suites and get test reports.
- Defects that are discovered early are less expensive to correct.
- When the customer detects an error which it is necessary to correct as fast as possible, the automated testing reduces considerably the reaction time

# THE CONTRIBUTION OF AUTOMATED TESTING TO THE DEVELOPMENT PROCESS

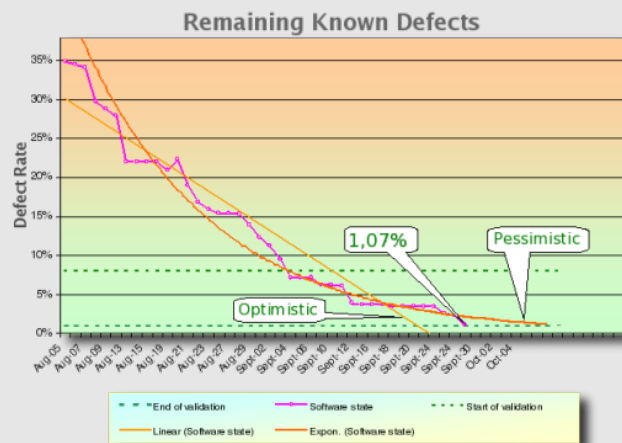
Better efficiency of the development process

The Validation test aims at the demonstration that the software, through its implementation into its target hardware, carries out indeed its specification. The test report is, to some extent, a proof given to the customer, whether (s)he is internal or external. The work methodology that comes with TestFarm makes it possible to establish the link between an element of specification and its corresponding test case. As a result, the matrix of traceability between the specification and the test result of is implicit.

Better control of the development process

The advantage of the validation testing is that it informs in a precise way about the real state of the software. One conceives consequently easily, that earlier in the cycle of development this type of test is carried out, better is the knowledge of the state of the software. One can imagine that the environment of development is organized so that it is possible to carry out a "validation test" on a functionality under development. One obtains thus, in an automatic way, an impartial state of the level of development of the functionality under scrutiny.

Test automation allows to perform an objective estimate of a product delivery date, by continuously measuring its maturity during the development phase.



Undeniable support to incremental development practices

Incremental development consists in building a stable software application with minimal features. Once this operation is carried out, new features are added one by one (in an incremental way) until obtaining the complete application. This mode of development makes it possible to show a conclusive model earlier in the cycle of development and gets uncontested points of visibility. In this mode of development, the automation of the test is capitalized with each increment, since the test relating to the previous increments is at no cost. Moreover, the concept of partial test so-called "non-regression" on the previous increments, is profitably replaced by the complete test.

Reduction of the development cycle time

The automation of the validation testing transfers upstream a large part of the workload located at the end of the development cycle. Indeed, the test cases writing effort is carried out before the phase of test. One thus reduces mechanically the total duration of the development. The validation testing generally being at a moment when there is "no time left", the pressure is on the shoulders of the "testers". Automation avoids this pitfall and returns a more immediate result towards the development team.

|  |   |
|--|---|
| Test results reliability                       | The automated test is not subjected to the risks of the manual procedure. An operator will not carry out the same test twice in the same way, as for instance for the timings of the manipulations. Moreover, after a few hours, one can reasonably think that its vigilance weakens. On the automated side, one obtains an unequalled "repetitivity" and an error free recording of the test results. In the manual mode, the test report is never totally certain because of the quantity of verdicts "correct" obtained, nothing certifies that no error occurred. |
| Better skill of the team in charge of the test | Automation makes disappear the boring part of the test, to the profit of an activity of thinking and coding. Automation authorizes operations which are very difficult, even impossible, to achieve by an operator. This new freedom reinforces the motivation and the creativity of the team   |