

Meet the New Tester [PART 2]

This is the second and last article in a two-part series

To reduce the high overhead costs of defect management in the system testing phase we must look to methods that allow a much greater proportion of functional errors to be detected and corrected during the development phase. The authors argue that this goal can be achieved by introducing a policy of thorough testing at the development phase, with the help of a new type of tester.



**David Evans and
Nick Pointon**
of Cresta



The new tester will be an active and co-located member of the development team. By working alongside developers and helping the team to test code as it is written, the tester provides immediate feedback on quality and catches defects at a stage when they can be fixed quickly and cheaply.

Skills and qualities of the new tester

The role of the embedded tester is no small one. A Test Manager must choose carefully to find a candidate who is successful within a development team. For starters the tester must be personable, communicative and able to state their case dispassionately yet confidently to possibly defensive developers. He/she must also be

technically confident, and ideally should be able to read and understand program code, even if not required to write it. They must also have the confidence of their convictions and be able to hold their own in discussions about the readiness of the software. They must be steadfast and not bow to pressure from the development manager to 'let through' sub-standard code.

An active interest in the tools of the trade is a must; developers will be willing to use tools to support quality initiatives if they can be convinced of their worth and effectiveness. The more *au fait* the tester is with these tools, the more acceptance they are likely to win from the developers.

Automation skills are a definite bonus: the ability to code unit tests, write batch files or debug Perl scripts will encourage creative solutions to embedding quality processes, as well as earning hefty credibility with the developers.

The management structure

So where does our new tester sit within the management structure of the project? The key point to remember is that these people are dedicated testers. They may sit within the development team and may be executing something between Unit Tests and System Tests, but they must be answerable to the Test Manager. In this way the tester has a responsibility to uphold the testing ethos and to ensure that these aims are not subverted by any conflicting demands of the development team, such as meeting a delivery deadline.

As with all other phases of testing, development testing must adhere to a set of project standards. It is likely that these standards will differ from those for System or User Acceptance Testing (a sort of 'System Test Lite'), but the basic principles that apply to all testing are still relevant: the testing must be visible, accountable and documented in sufficient detail to provide

Technically confident

Able to read and understand program code

Confidence of their convictions

Steadfast

Active interest in the tools of the trade

Automation skills

confidence in the testing carried out, and if necessary to allow handover to another experienced tester.

This is not some special pre-System Test team, sitting apart from the developers; these are embedded testers. The testers must be co-located with the rest of the development team and get involved in the day-to-day activities of the developers, attending the developers' planning, design and progress meetings. Even more valuable, by sitting with the developers the new testers will hear all the informal information that passes around the team, much of which will contain vital information about the system's behaviour and pointers on where to direct testing efforts.

Although answerable to the Test Manager, the new tester will take direction on a day-to-day basis from the development team leader. Only the development team leader knows which packages are actively being worked on and which are ready to be tested, and can steer the work of the tester appropriately (see Figure 1).

The delivery out of the development team is now dependent on the tester successfully passing her tests. The tests to be executed, however, are directed by the test strategy or methodology, which is the responsibility of the Test Manager. This means that the development team leader appears to be dependent on something that is outside his control. For example he cannot affect the tests to ensure that they meet his delivery deadline. On occasions this can be frustrating for development team leaders and often they will exert undue influence to get tests changed or

removed. For this reason the new tester needs to be a senior tester with a strong personality, who is comfortable defending her testing principles; and of course she needs to be backed up and supported by the Test Manager.

Of course it is not really true to say that the development team leader is responsible for something that is outside his control and he shouldn't consider the presence of an embedded tester to be at variance with his objectives. His goal is to deliver high-quality software in a timely way, and the new tester is helping him to do this by providing accurate and unbiased feedback. The development team leader should welcome the new tester as a valuable resource in the team.

Peripheral advantages of the 'New Tester'

We talked initially about the cost savings that can be made by identifying defects earlier in the development life cycle. Having embedded testers is a key means of realising these cost savings, however there are a number of other benefits that will be enjoyed.

In some organisations there is an apparent conflict of interests between developers and testers. A divide, both physically and emotionally, can spring up between the two parties and a degree of 'finger pointing' on both sides can foster resentment between the two parties. This effect can be particularly pronounced where the development is outsourced to a third party, who is keen to maintain independence from the rest of the organisation.

Defect migration

Defects are introduced into a system because a mistake is made at a certain phase of the system's design or development. There are various types of mistake, and phases of development in which they are made. Examples of mistakes that introduce defects would include:

- an ambiguously specified requirement
- a flawed technical design decision
- an incorrect coding algorithm
- a fix for one defect found in testing that caused another elsewhere in the system

When a defect enters the system in one phase but is not removed until a later phase, if at all, it is known as a 'phase escapement.'

While it is better to find and remove defects in the testing phase, rather than letting them be discovered in production by users, it is best to remove defects in the same phase in which they are introduced. It is therefore best to apply quality control to the products of all phases, not just developed code. This is because the cost of defect removal increases dramatically for each phase that the defect remains.

The quality of the products from one phase directly affects quality in the 'downstream' phases. The impact increases as a defect escapes into later phases, because the size of the products of each phase increases dramatically. For example, a system may have hundreds of documented requirements, which leads to thousands of items in a technical specification, which leads to hundreds of thousands of lines of developed code, which leads to millions of lines of installed code.

If a mistake is made that leads to a defect in a single requirement, this can have a 10-fold impact on the design and a 100-fold impact on the code. If such a defect is not discovered until the test phase, it could therefore require changes to hundreds of lines of code. If, however, the same defect were found and corrected during the requirements specification phase, phase escapement would be prevented. As a result, the cost of correcting the requirement will be negligible because it is before design and coding has been affected.

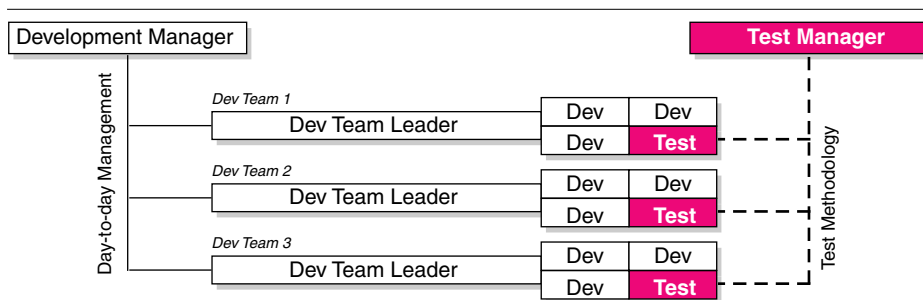


Fig. 1 Management structure

Embedding testers in the development team breaks down barriers between the two teams. The natural interactions that happen as a result of sitting in among the developers will foster a spirit of understanding and create personal relationships between the two teams' members. This, in turn, will promote co-operation between the teams and can help the project work as a coherent unit.

The embedded testers also serve to raise the profile of testing within development. By making each development team responsible for the quality of their software, where that quality is independently verified, means that each team has to ensure that the software it produces functions correctly. It prevents a 'that'll do' mentality as developers cannot throw the code 'over the wall' until it has correctly passed its tests. In essence, it keeps the developers honest.

The new testers' tests are, by their very nature, the exit criteria from development into System Testing. In many organisations, exit criteria for each phase are elaborately set out during the planning phase only for them to be ignored when the project gets underway. With the independent stamp of approval required from the embedded testers to exit development, the exit criteria are a visible and integral part of the process. The project management can choose to disregard them, but it cannot pay them lip service and just forget about them.

By embedding testers within the development team, the develop-test-fix cycle becomes a much tighter loop. The testers are able to provide very swift feedback to the developers, and the developers can provide fixes very quickly, without the need for extensive change control procedures. Normally only unit tests

enjoy this level of near instantaneous response, but unit tests are not produced by professional testers and are not designed with the same attention to detail and the business-oriented mindset that professional testers' tests are.

The result is cleaner and more robust builds, which give System Test a much greater chance of success. Given that the System Test phase is the one that is most commonly cut short by fixed delivery deadlines, any improvement in quality going into System Test will reduce the total cost of testing and result in fewer production problems ■

nick.pointon@cresta.net;
david.evans@cresta.net

**Subscribe to
Professional Tester Magazine**



We doen in ICT.

En we hebben iets te melden.

Sogeti
Klaar voor de toekomst

Sogeti is één van de top-5 ICT-bedrijven van Nederland. Met 2.100 medewerkers en een historie van 35 jaar staan wij garant voor interessante opdrachten en klantomgevingen. Het ontwerpen, realiseren, implementeren, testen en beheren van ICT-oplossingen behoort tot onze core-business. We staan voor kwaliteit en vakmanschap. Dit is zichtbaar in onze dienstverlening en in de door ons ontwikkelde en tot standaard uitgegroeide methoden zoals DYA®, Regatta®, TMap® en TPI®.

De divisie Software Control (SC) maakt de risico's die worden veroorzaakt door fouten in software inzichtelijk en beheersbaar. Met onze dienstverlening op het gebied van Testen en Quality Assurance geven wij klanten inzicht in risico's die nog in een systeem zitten en bieden wij hulpmiddelen om daar gecontroleerd mee om te gaan. Onze methoden TMap® en TPI® worden wereldwijd gebruikt en zijn in veel talen vertaald. In Nederland is Software Control met 450 medewerkers de grootste gespecialiseerde eenheid op het gebied van Testen en QA. Met als resultaat dat wij de mooiste en beste opdrachten mogen uitvoeren en zeer gericht kunnen investeren in zowel dienst- als kennisontwikkeling.

Wij zijn doorlopend op zoek naar nieuwe collega's voor de functies:

- Testcoördinatoren
- Testengineers
- Testmanagers

Sogeti staat voor resultaat door gepassioneerd ICT-vakmanschap. We hebben daarin een naam hoog te houden. We zoeken voor deze functies vakmensen met minimaal een HBO-werk-/denkniveau en aantoonbare ervaring die klinkende resultaten kunnen behalen. Wij bieden je goede arbeidsvoorwaarden en uitstekende opleidings- en certificeringsmogelijkheden in een dynamische werkomgeving. Ben jij de gepassioneerde vakman of vakvrouw die we zoeken? Kijk dan voor meer informatie over de mogelijkheden, vacatures en sollicitatieprocedures op www.tmap.net of www.sogeti.nl.

Sogeti Nederland B.V. Postbus 263, 1110 AG Diemen
Tel (020) 660 66 00 Fax (020) 698 14 37 www.sogeti.nl

SOGETI
Realisme in ICT