This is only a test

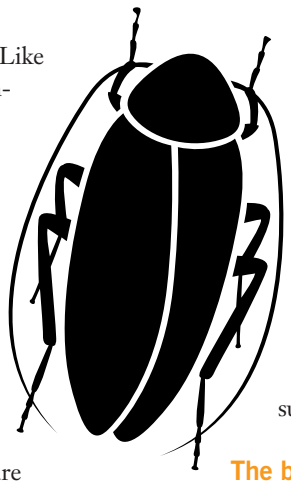
Dan Tebbutt explores the business case for software testing with a cross-section of industry experts

Dan Tebbutt is a senior writer with Editor Group and is one of Australia's most respected IT writers. He has a decade of international experience covering high-tech for leading publications such as *The Australian* and *Australian Personal Computer*. He has interviewed many of the world's leading technologists, including the inventors of the Internet, the Web and other important networking and computer technologies.
E-mail: dtebbutt@editor.com

Editor Group is a specialist writing and editorial services provider with recognised expertise in the information technology sector.
Visit: www.editor.com

Software testing has an image problem. Like tax returns, exercise and visits to the dentist, testing is a chore done grudgingly and not as regularly as recommended.

Whether it's called testing, debugging, quality assurance or verification and validation, its value is enormous and its implications range from staff turnover and sales success to legal liability and even business failure. The $5 billion One.Tel collapse arguably shows what can happen when software testing is neglected. A little more checking of the core billing software and One.Tel might have actually collected all the revenue its sales effort was generating.



A 2002 study by the National Institute of Standards and Technology (NIST) in the United States found that software bugs are so prevalent they cost the US economy an estimated US$59.5 billion annually – a 0.6 per cent drag on GDP. Over half this burden is borne by the end-users of software, with the rest shouldered by developers.

Not all errors can be removed, but NIST estimated that better testing could generate savings of around US$22.2 billion per annum.

With so much to gain, why is software testing not afforded higher priority? Testing is still regarded as difficult and time-consuming, an area where development time and spending can be economised when the pressure is on.

This view is increasingly outdated, with a broad spectrum of software industry figures agreeing that the business case for testing is now stronger than ever. With business performance hinging on stable and functional software infrastructure, independent software vendors (ISVs) and major software acquirers advocate thorough examination of every piece of code.

"If you can't afford to test software, don't build it – just don't even start," says Dr Graeme Port, vice-president of engineering at Melbourne-based multinational ManageSoft. "Without adequate testing it is just going to be grief for you, grief for your customers and grief for your support organisation. Just don't go there."

The business case for testing

The commercial justification for testing is all about risk mitigation. "The business case is really an avoided cost that's likely to occur if you don't test," explains James King, a business process consultant with AMP. He says testing should concentrate on areas of greatest risk, such as customer-facing applications and those with financial implications.

Inadequate testing essentially amounts to enlisting end-users as crash test dummies. This has a real cost, King says, with the credibility of the development team on the line every time a 'buggy' release is distributed. These doubts can lead to additional layers of governance at the insistence of customers, partners or regulators.

Buggy software is a major source of stress for employees. King says that this type of dissatisfaction drives up business costs through lost productivity, lower morale and increased staff turnover. Testing is a key strategic weapon in combating these issues.

Litigation and legal liability for faulty products is a primary driver behind testing. In a business

'If you test properly, you can spend less money and less time than if you didn't test'
James King, business process consultant, AMP

environment where unlimited liability is written into the majority of new contracts for enterprise software, testing is about more than mere inconvenience for users.

Adrian Di Marco, chief executive of Brisbane-based ISV Technology One, warns that inadequately tested software can leave the developer liable for enormous costs, including restoring corrupted customer data and paying for downtime. He says the software industry's failure to take testing seriously in the past is now coming home to roost.

"This industry has done a very poor job of meeting customer expectations and as a result, these days people go into relationships with vendors wanting a very strong, tight contract and they're not prepared to give them the benefit of the doubt."

'Testing has to be really integrated through your methodology. It's not something that can be layered on top'

Adrian Di Marco, chief executive, Technology One

Counting the cost

Risk management expert Colin Cherry, who is program chair of AsiaSTAR, the region's leading software testing conference, says the value of testing is directly related to the cost of the application's failure.

"Think of testing as a form of insurance for your business," he says. Proper testing can help a company avoid breaching its legal obligations and ensure better business performance, and even the information gathered in the testing process is a very useful resource for risk mitigation activities.

'Think of testing as a form of insurance for your business'

Colin Cherry, program chair, AsiaSTAR

The cost of inadequate testing is felt across the organisation. Quality software can be the difference between products sinking or swimming, between sales success or failure, between excessive or minimal support costs and between corporate glory or disaster.

The NIST study suggests that few products are allowed to be sold with as many errors as software. "Without testing you are almost guaranteed to have faulty software," Port warns. Product failures can range from failing to make an on-sale date or operative deadline through to not delivering promised features or even a market-ready product.

Paul Beesley, vice president of solutions development for Mincom, Australia's largest commercial software developer, says testing is essential to meet rising customer expectations about software quality. "What customers thought was a good product from us two years ago would not apply today," he says. "There's a natural evolution in customer expectations."

"Just about every software developer has shipped a bad release at one time or another," admits Di Marco. The important question is how they responded and what changes resulted. Technology One distributed a troublesome upgrade many years ago, but the experience sheeted home the value of solid testing and review.

The enterprise software company now has numerous control points and a management culture that ensures corners are never cut during testing. "The lessons we learned are all common sense in hindsight, but when you are growing a business they can get bypassed," Di Marco says.

Impact on products and sales

Testing is not just about avoiding product failures. Testing can actually improve products by verifying that customer requirements are met, the architecture is sound and the solution performs satisfactorily. Software testing expert Dr Kelvin Ross adds that well-tested software is almost always easier to maintain, extend and upgrade in the future.

Dr Kelvin Ross

The ramifications of product failure immediately translate into the sales process. ManageSoft's Port says poorly tested products can have a deleterious impact on every stage of the sales cycle, from presales (an inability to compete in customer evaluations), to sales effectiveness (sales start to dry up), to after sales (dissatisfied customers won't provide sales references), and support (buggy products cost more to support).

The real indicator of success in testing is support costs. Di Marco says that thorough verification and validation (V&V) processes bring substantial benefits in terms of reduced support requirements. "If you don't do your testing properly, it has a huge impact on your support centre and you need a lot more people in support. Having resilient code means you can run a much smaller support centre."

Money spent fixing problems and supporting customers tends to detract from forward development, so it is almost always cheaper to test properly before software is released. AMP's King believes cutting ▶

corners in the V&V phase is a false economy. "You tend to save your money initially but then spend it over the next 12 months. It costs a lot more to fix something that's in production than while it's in development. If you test properly, you can spend less money and less time than if you didn't test."

Test early, test often

Early identification of problems through V&V is considerably cheaper than 'downstream' discovery of bugs.

The world's leading authority on software economics, University of Southern California academic Professor Barry Boehm, calculated that the cost of fixing a bug at the requirements stage is US$200 but this balloons 75 times over to a US$15,000 repair cost once an application is in production. Nevertheless, the NIST study found that the majority of errors are actually found downstream – late in the development cycle or during post-sales usage.

"If you sacrifice testing upfront, it comes back to haunt you and you pay the price further down the track," argues Ian Gray, group manager of research and development for Adelaide-based ISV Prophecy International. He says any design problems not exposed in testing result in timeframes slipping, poorly executed features and cost blowouts.

'Developers want to demonstrate that things work, but testers want to demonstrate that things don't work'

Ian Gray, group manager of research and development, Prophecy International

The bottom line

At the end of the day, testing is about quality assurance for software products – ensuring delivery of good products that fulfil customer expectations.

Donna O'Neill, managing director of testing consultancy IV&V Australia, says the true value of software testing is frequently learned from experience. "Many developers release a disastrous product with lots of defects and it costs them a fortune to get it right. Then they start to get the message that maybe something is wrong with their process and they need a stronger quality focus."

Paul Beesley says the testing equation is simple: good, well-tested products enhance sales, while bad products add cost and divert attention from new developments. "The bottom line is that a happy customer will generate more revenue," he says.

For ISVs, testing is a matter of life or death, according to ManageSoft's Port. "Our survival is based on the quality of the product we build. If your testing is poor or inadequate, the maths is very simply: you kill revenue, you ramp up expenses and your bottom line is bathed in red ink."

How much to spend

Testing is almost certainly the area where skimping is most likely during development. While better projects maintain testing, Emprove managing director Erik Petersen believes the single biggest cause of project delays and budget overruns is not the testing itself but the tendency of managers to underestimate the time and effort required.

Erik Petersen

Decisions about how much time and money to spend on V&V are a trade-off between cost and functionality. While no testing can be disastrous, spending too much can stunt forward development. The overall objective is to reduce risk and raise the comfort level to a tolerable threshold that can be delivered at an acceptable cost.

Colin Cherry suggests there is no right or wrong approach, just a sliding scale of appropriate behaviour. "Unless you understand the real drivers of your business you will probably spend far more time and money on testing than you need to and you still won't reduce the risk sufficiently," he argues.

It's important to consider the financial or customer impact, says King at AMP, as well as the likelihood of any risks and the risk profile of target customers. But managers should also be mindful of the difficulty involved in fixing the problem.

For example, web-based applications can be updated quickly, whereas mainframe or complex middleware applications with lots of interfaces are slower and harder to fix, so more testing is required.

'If testing practices are done efficiently, they actually reduce the cost of development'

Donna O'Neill, managing director, IV&V Australia

Petersen adds that it is important to consider the consequences of failure: software that might affect safety, finances or legal liability should be the top priority for V&V.

It is much cheaper to fix problems earlier in the development cycle. Historical analysis

at Mincom, for instance, shows it is 15 to 20 times cheaper to find a bug internally – even as late as the user acceptance phase – than once the application enters production.

On the bright side, James King says experience actually makes testing easier – if the development team hangs together. "It definitely gets easier over time. You have more systems, so it's more complex, but you also have greater knowledge, more reuse, more scripts, existing test plans and the team is more familiar with the code," he says.

Similarly, the maturity of the development process and the development team can influence how many resources are devoted to testing. "If the developers already write very solid specs and review their work and have good coding standards, it often lessens the need for a testing group," Kelvin Ross says. "If their processes are weak, they are likely to produce buggy code to begin with."

Overhead or essential?

A major misconception associated with testing is that V&V is somehow an overhead expense that needs to be reduced. Industry figures argue that the value delivered by testing is clearly justified on a return on investment (ROI) basis – provided it is focused.

"To the uninitiated, it seems like an overhead expense, but that's because they don't understand the value of testing in terms of risk mitigation," Donna O'Neill says. "If testing practices are done efficiently, they actually reduce the cost of development – because it is cheaper to fix a problem earlier than later, and also because test feedback can be used to fine-tune sloppy development practices."

Developers such as Technology One and ManageSoft try to foster a culture where testing is seen as a core strategic element of the ISV's business (see case study page 34, 'Bug Lotto inspires ManageSoft'). "Building up capability in the testing organisation is every bit as important as building up capability in the development organisation," Port says.

Is it enough?

So do software developers spend enough on testing?

Ross says this is difficult to measure, since each development project is unique and there are no widely accepted industry benchmarks. He suggests the figure of one tester for every four developers as a starting point that should be adjusted up or down depending on the maturity of the team and its processes.

Technology One regularly commits around 25 per cent of its software development budget to V&V activities. Di Marco says the company follows a highly structured approach with checklists and regular ▶

HOW MUCH IS ENOUGH?

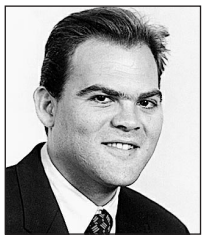
Deciding how much to spend on software testing is tricky, as the commercial value, importance and technical nature of each project is unique.

As a guideline, Coles Myer would typically allocate from 15 to 50 per cent of the budget for a new IT development to testing. Factors to consider include the level of technology being deployed, the development team's capability and the degree of process compliance.

When modifying an existing application, testing can vary from 20 to as much as 80 per cent of the total maintenance effort, according to Coles Myer project manager Louis Liew. "This variation is mainly due to the regression testing effort to ensure we don't break existing applications," he explains.

"Quality applications are an important business tool that enable us to serve customers better and manage day-to-day business operations effectively and efficiently. Testing allows us to reduce business risk, which translates into a lower cost of doing business and in turn enables us to provide more competitive product offerings to our retail customers."

Louis Liew



Chris Carter

management review. "Testing has to be really integrated through your methodology. It has to be done all the time; it's not something that can be layered on top."

Mincom suggests an even higher figure – up to half of the development time spent and 30 to 40 per cent of the budget – owing to the complexity of its enterprise resource planning products for large utilities and mining companies. In comparison, ManageSoft has fewer software engineers dedicated to testing because it strongly emphasises the duty of coders to conduct first-level testing.

On the acquirer side, AMP estimates that testing normally accounts for around 25 per cent of the budget for general IT projects, but this falls for web-based rapid application development (RAD) work or maintenance upgrades. Experience at Virgin Blue suggests a similar figure (see case study page 36, 'Virgin Blue testers flying high').

As a consensus figure, Chris Carter, managing director of test management solutions specialist Planit, suggests that testers would claim 40 per cent but around 30 per cent is reasonable for most projects. "At the end of the day you get what you pay for," he says. "Spend anything less than 20 per cent and you're asking for trouble."

Do most organisations make sufficient allowance for testing in practice? Most start out with the best intentions, but testing is often squeezed for time and resources towards the end of the project. "Testing is the first thing to be sacrificed when the pressure is on and it takes a strong project manager to fight to keep the quality in," says Gray.

O'Neill and Cherry both believe it's rare to find adequate allocation for V&V, so the real question should centre on identifying the risks of not doing so and deciding where cutbacks can be made with least risk.

Likewise, Ross suggests that many organisations wrongly view testing as something that detracts from their ability to deliver applications. "Many people think they would rather put on another developer, but that

Dr Graeme Port

CASE STUDY

Bug Lotto inspires ManageSoft

A sense of fun drives the push for thorough testing and quality assurance at ManageSoft, a Melbourne-based multinational formerly known as Open Software Associates. Patented software distribution and management technology is the driving force behind this ISV's global success, which includes international partnerships with Hewlett-Packard, Microsoft and Siemens and a blue chip client base spanning finance, telecommunications, law, defence, government and education.

ManageSoft uses the Rational Unified Process development model, which calls for frequent iterative releases with testing throughout the product development cycle.

Earlier this year, while developing its ManageSoft for Windows Deployment 2.0 extension product, the company decided to add extra spice to the testing process. During the construction phase, six iterative releases were built, packaged and sent for quality assurance on a weekly basis.

That's when the fun started.

To encourage thorough testing, ManageSoft devised a game called Bug Lotto. For each iterative release, the developers had to document the features and estimate how many bugs the QA team would find. Bugs were weighted according to seriousness and impact.

The QA group would then subject the release to a rigorous suite of tests, trying to shake out defects, inconsistencies and bugs among the 55,000 lines of code. After three days of testing, a bug report was collated and scored.

If the number of bugs equalled or exceeded the developers' estimate, QA would take home a generous Tattslotto ticket. When the testers could not find as many bugs as the programmers predicted, the code jockeys would take home the ticket.

The final score? QA won 6–2 on the eight features developed, but ManageSoft vice president of engineering Dr Graeme Port says that's actually a great result for the coders. "The fact that engineering won anything shows that they were exercising the code very hard before it went to QA."

'If you can't afford to test software, don't build it – just don't even start' **Dr Graeme Port, ManageSoft**

Port says Bug Lotto inspires healthy competition and encourages programmers to take responsibility for the quality of their work. "You don't want things to be adversarial between the development and QA teams, but you do want each to be doing the best job they can in their area," he says.

Visit: www.managesoft.com

can actually exacerbate the problem because no one may be evaluating whether the developers are making progress or heading in the right direction," he says.

The problem, in essence, comes back to the image of testing as an optional extra rather than an essential ingredient in the development mix. At the coalface, testing is often not given due recognition: when things go well developers take the credit, but when things go badly the testers take the blame.

A testing set of skills

Software testing is now developing into a discipline in its own right, with greater professionalism and unique skills that differentiate testers from other software engineers. To date, however, the contribution of software developers is not widely recognised.

Experts agree that testers generally have a broader outlook that emphasises strategic thinking more than technical know-how. "We're looking for people who understand the business processes and who subsequently have an interest in IT," says Mincom's Beesley. "Ultimately it's not so much the testing tool they use as the business knowledge we are after."

Skills that are highly sought in testers include analytical skills, critical thinking, risk assessment, prioritisation and methodical behaviour, attention to detail and advanced problem solving. "You need to question what you see before you, to try to find what might be missing or incorrect," Donna O'Neill says.

As a lecturer in advanced software engineering at Griffith University and a consultant, Kelvin Ross spends a lot of time encouraging students and clients to see the difference between testing and development roles. "You need people with an aggressive attitude trying to find bugs and break the system. That is not typically a coder mindset," he explains.

Ian Gray says this difference typically comes down to the fact that developers want to demonstrate that things work, whereas testers want to demonstrate that things don't work.

Beyond their technical proficiencies, testers need to have what Colin Cherry calls 'soft skills', including active listening abilities, interpretation of requirements and strong oral and written communications. "The role of a tester is often to provide a communication channel between the IT and business communities, so the ability to communicate on both levels is a critical skill," observes Planit's Carter.

Moreover, testers need political deftness, since they often serve as bearers of bad tidings. "You need to deliver bad news gracefully with tact and diplomacy and still walk away smiling," says Martin Harris, ACT general manager of IT recruitment specialist Candle Australia.

THE BUSINESS CASE FOR SOFTWARE TESTING

Why developers should test

- Testing early in the development cycle costs as much as 75 times less than fixing a problem in production
- Poor testing puts a developer's credibility on the line and leads to additional governance costs
- Solid V&V reduces the burden on the support centre
- Well-tested products are easier to upgrade and maintain
- Money spent fixing poorly tested applications detracts from forward development
- Deciding how much to spend is a trade-off between cost, functionality and risk
- Developers need to foster a culture that recognises testing and avoids cutting corners
- Happy customers buy more software, while poorly tested products squander goodwill
- Testing protects the developer and the customer
- Regular testing throughout the development cycle delivers better products

Why acquirers should test

- Research suggests bugs cost the US economy US$59.6 billion annually
- Testing is insurance against risks that could lead to legal liability or business failure
- Solid testing is essential to meet rising customer expectations about software quality
- Core business applications are slow to change, so more testing is required
- Acquirers need to be involved early, from verifying requirements documents to validation through user acceptance testing
- Top priorities for testing include customer-facing applications and those with financial impact
- Testing reduces risks associated with business-critical software

Certification: some belated recognition

Another issue for the testing community is the absence of generally recognised professional standards. With no easy way to tell testers apart, it is still very easy for anyone to hang up their shingle as a tester. What matters most today, according to recruitment agents like Harris, is experience, references and possibly some sort of product-level certification.

This situation is changing, with a growing move towards a consultative approach. Harris says this new breed of test analyst offers a more disciplined approach that addresses V&V through a broad, systematic approach.

Ironically, Harris credits the technology industry slowdown with stirring up interest in quality assurance and testing. Candle is now seeing greater demand for test analysts than ever before. "Previously the testing arena was sparse on people. They were spread about as part of larger projects and not necessarily recognised as a specialist skill. Since the 'tech wreck' there is a lot more recognition: people want quality testing specialists." ▶

'Previously the testing arena was sparse on people. They were spread about as part of larger projects'

Martin Harris, ACT general manager of IT recruitment, Candle Australia

Harris advocates certification as a big step forward for the V&V sector. Currently the most widely recognised qualification is the Information Systems Examinations Board (ISEB) range of certificates in software testing. ISEB operates according to an internationally agreed syllabus with independent assessment.

S|E|A| National's Certified Software Test Professional (CSTP) course is a newer standard, now rolling out nationally after two years of pilot development. Candle already advocates CSTP to contractors and clients, believing it offers an edge in the recruitment market.

Recognised certification could also drive salaries upwards. The 2002 Australian Computer Society Remuneration Survey Report found an average package of $77,625 for engineers working in testing and analysis – more than general programmers and analysts ($73,060) but considerably less than database administrators ($83,399).

AsiaSTAR's Cherry says the strength of CSTP will come from focusing on the core competencies of software testers together with defined practical deliverables and outcomes. He suggests testers should aim for at least 10 days per year of professional development and training activities, including seminars, online learning, formal instruction and conferences.

Most ISVs and software acquirers don't currently insist on certification because on-the-job skills are more highly prized. Most welcome certification as an excellent avenue for professional development – a theme emphasised by Kelvin Ross, one of the driving forces behind the development of CSTP.

"Testing is now a lot more recognised than it used to be, but it is still often seen as a poor cousin to programming," Ross says. "That is starting to change and a lot of companies are now seeing the risk mitigation benefits of testing."

Changing perceptions requires patience, but in time the true value of software testing is destined to be recognised. ■

CASE STUDY

Virgin Blue testers flying high

No airline in history has enjoyed such spectacular growth in its first three years as Virgin Blue. While the new carrier certainly enjoyed a measure of luck, its success reflects outstanding execution on a business strategy based around speed, flexibility and modern technology.

The ability to quickly build, test and deploy software applications is at the heart of Virgin Blue. The Brisbane-based airline maintains a team of five software engineers to develop and test new applications.

The speed of the Virgin Blue business – and the pressure on its development team – is illustrated by the 'Stocking Stuffers' project. The company came up with the idea of selling gift vouchers that customers could give as Christmas presents. The difficulty for the development team was that Christmas was fast approaching.

"The whole idea went from conception to production in two weeks," recalls Virgin Blue development manager John Morrison. "It was a cross-departmental project, so there was a lot of work in finalising the requirements statement."

Coding was completed in three days, thanks to reuse of many base elements from existing applications. This was followed by three days of testing, which was mainly concerned with ensuring the application was easy to use for internal customers (Virgin Blue sales agents) and that it gathered all the information required for the product.

'The amount of money you spend on testing is a fraction of what you stand to lose if things go wrong' **John Morrison, Virgin Blue**

"Most testing centred on verification of functionality, rather than load testing," Morrison says. "Accuracy, reliability and security are far more important than performance for browser-based solutions where many of the performance issues are outside your control."

Such a short lead-time would often place a squeeze on testing, but Morrison says Virgin Blue insists on thorough testing before releasing a new application. "The amount of money you spend on testing is a fraction of what you stand to lose if things go wrong."

Usually Virgin Blue reserves about 25 per cent of the time on a software development project for testing. The team works closely with internal customers in acceptance testing to ensure requirements are met. "The key to good software is the design of the requirements document and thorough testing – if you are proactive about those, your applications will be successful," Morrison says.

Visit: www.virginblue.com.au