



Primary Research Report

In association with



Continuous Testing as a Digital Business Enabler

Lessons from leading practitioners

Freeform Dynamics, 2017

Executive Summary

Software delivery processes and systems, and the people involved with them, are under increasing pressure. Sometimes it's digital transformation, other times it's simply the challenge of keeping up with the demands created by ever more dynamic markets and an escalating pace of change. None of this is news, but it does provide an important backdrop to the discussion of how software delivery needs to evolve, especially given that traditional methods and approaches were never designed to deal with the fast-moving and unpredictable environment you are probably working in today.

But as you look to modern alternatives, like Agile Development and Continuous Delivery, it's important to ensure that you take a holistic and balanced approach. Yes, you need more speed and agility – business stakeholders are crying out for these – but it's no good delivering on those needs if that comes at the expense of quality, security and compliance. All you will be doing is storing up problems and risks that are likely to come back to bite you when you are least prepared.

With this reality-check in mind, the spotlight is increasingly shifting to the way you manage quality assurance, and particularly testing, in the modern, iterative software delivery cycle. Our aim in this report is to shed some light on what's important to consider as you transform and optimize testing practices in your environment, with the help of findings from a recent worldwide research study in which input was gathered from 923 IT and testing professionals. The detailed results are laid out on the following pages, but here are some key points to give you a feel for the kind of areas covered:

Modern methods are key, but the real aim is lifecycle integration

The majority of those participating in the study highlighted the importance of modern methods and approaches such as Agile, Continuous Delivery and DevOps. When put on the spot and asked about the single highest priority to drive improvement, however, nearly two thirds said it was about full lifecycle integration. Agile development is core to this, but so too is the idea of extending agility to both upstream and downstream activities via a Continuous Delivery framework with a heavy emphasis on automation.

New ideas in testing support the continuous, integrated vision

When asked about evolving testing requirements, a strong consensus emerged around the importance of automation, even though only one in five reported having a highly-automated testing environment. When looking to drive improvements, however, two ideas came through very strongly. The first was the principle of 'shift-left', the objective being to build testing into everyone's thinking and activity as early as possible, starting with requirements definition. Closely related is the idea of Continuous Testing, which may be defined as the embedding of testing as a fundamental and ongoing aspect of every activity through the software delivery cycle.

Get it right, and software testing actually becomes a business enabler

In order to close the loop with the discussion of how best to 'keep up', we assessed the impact of modern continuous testing practices on outcomes relevant to the business. A group of 'Leaders' who had made exceptional progress (see sidebar for details) demonstrated what could be achieved. They were more confident in meeting business needs and expectations in relation to speed, quality and efficiency, and were generally enjoying greater returns from their broader investments in Continuous Delivery.



DEFINITION OF

CONTINUOUS TESTING

The embedding of testing as a fundamental and ongoing aspect of every activity through the delivery cycle.



CONTINUOUS TESTING

LEADERS

20% of survey respondents, were identified as 'Leaders' based on ratings in relation to key enablers and capabilities captured on a scorecard:

- ✓ **Knowledge of modern delivery and testing techniques**
- ✓ **Foundational continuous delivery capabilities**
- ✓ **Tools and processes to enable test automation**
- ✓ **Breadth of test automation coverage**
- ✓ **Organization, culture and people-related**

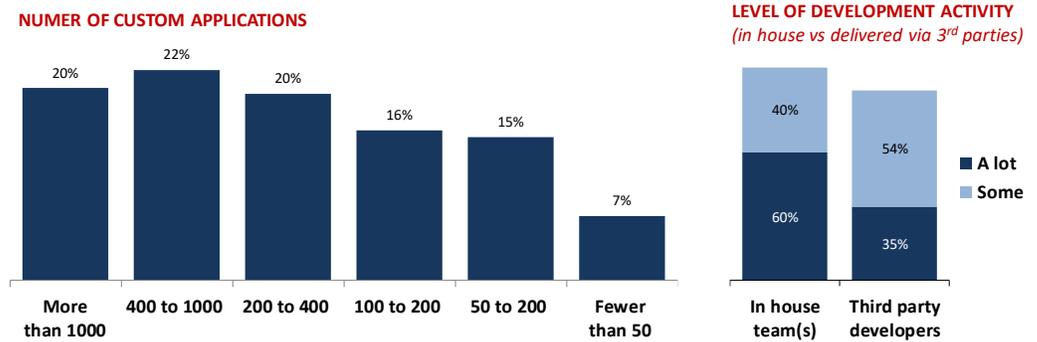
Leaders enjoy a boost to software delivery speed, quality and efficiency, and realize a range of other benefits, including a better return on their investments in continuous delivery.

A critical function, undergoing change

Development activity is not just about delivering new applications.

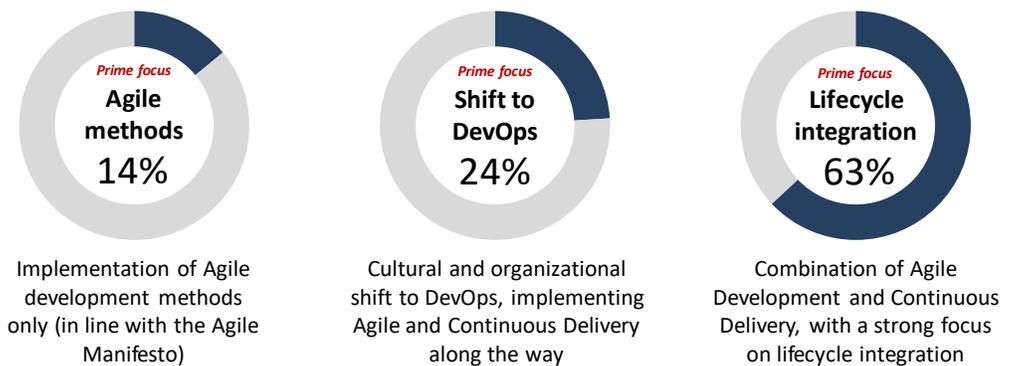
The fact that most organizations participating in our study do a lot of software development will not come as a surprise. Perhaps more important is a reminder that development activity is not just about delivering new applications, but also modifying and enhancing an often-sizeable custom application estate as requirements and technology evolve (Figure 1).

Figure 1
How many custom applications do you manage, and how much custom development do you do?



Against this background, it's understandable that the majority of study respondents tell us that modern approaches are critical, particularly the notion of creating a fully integrated and Agile-enabled Continuous Delivery lifecycle (Figure 2).

Figure 2
Which of the following statements best sums up your view of what's needed in your environment to optimize software delivery?



The picture we see here sums up the high-level imperative to move on from traditional waterfall methodologies and their rigid lines of demarcation that were conceived at a time when requirements changed far less quickly and constantly.

'Continuous' is now the watchword

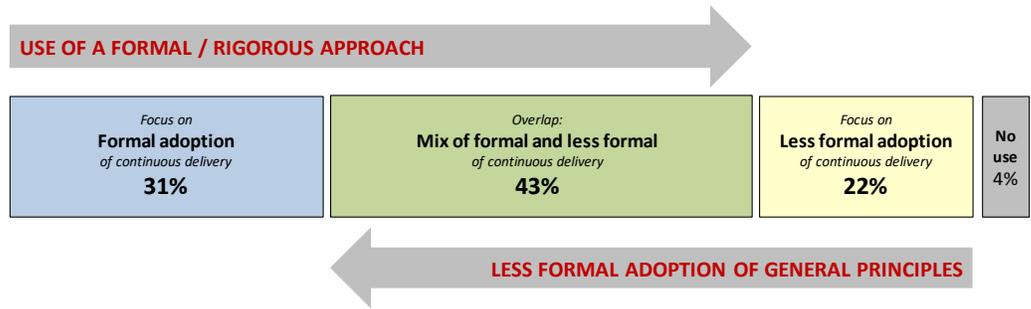
Continuous delivery provides the tangible process and automation framework that makes a joined-up, end-to-end, iterative approach practical and real.

Agile methods cover many 'upstream' requirements in the modern delivery cycle, and DevOps deals with the organizational, philosophical and culture dimension of delivery transformation. It's continuous delivery, however, that arguably provides the tangible process and automation framework that makes the joined-up, end-to-end, iterative approach practical and real, allowing a fast and constant flow of business value.

When it comes to implementation, some have taken a formal and rigorous approach to putting continuous delivery in place, while others have embraced the key principles of automating and integrating across the delivery cycle in a less formal manner (Figure 3).

Figure 3

What approach has your company taken to continuous delivery adoption?



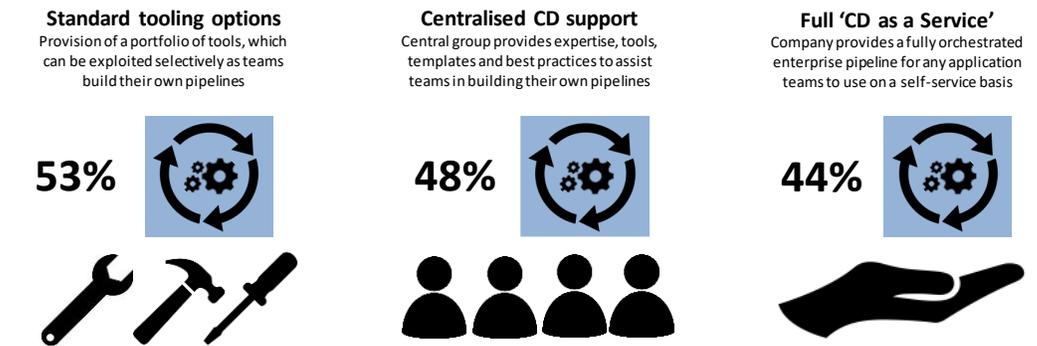
There is a need to maintain a degree of flexibility when implementing continuous delivery.

While it would be easy to assume that a formal approach is always best, this is not necessarily the case. Being organized and disciplined is good, but in some scenarios, too much rigor can be counterproductive, for example if it unduly constrains developers, operations staff and external parties involved in delivery, or forces them to work sub-optimally just because ‘the book’ dictates the way something must be done.

This need to maintain a degree of flexibility comes across when we see that each of the common continuous delivery support models has its place (Figure 4).

Figure 4

Which of the following do you currently use in order to support teams in their adoption and use of continuous delivery?



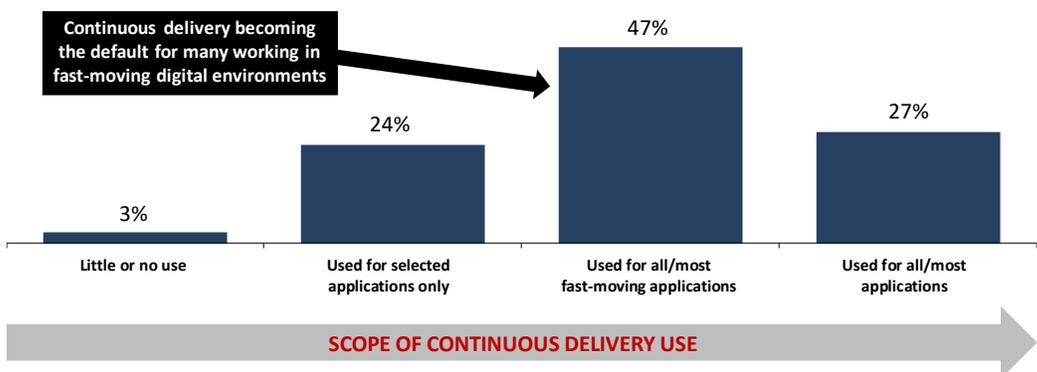
It is common for continuous delivery activity to be focused on faster-moving applications.

Green field digital developments may well be best served via the ‘CD as a Service’ approach, but it may be necessary to mix and match tools and techniques for projects concerned with packaged applications or established systems running in proprietary environments. There are then occasions, e.g. where developers are experimenting at the bleeding edge, when you need to allow pretty much total freedom. The important thing is to make sure that continuous delivery principles are applied wherever possible.

On that note, it is encouraging to see a significant number of study respondents already applying continuous delivery to all/most applications, validating the broad relevance of the approach. Having said this, even in our progressive study sample, it is more common for continuous delivery activity to be focused on faster-moving applications (Figure 5).

Figure 5

How broadly is continuous delivery currently used, whether formally or less formally?



Quality and efficiency are important benefits, as well as speed.

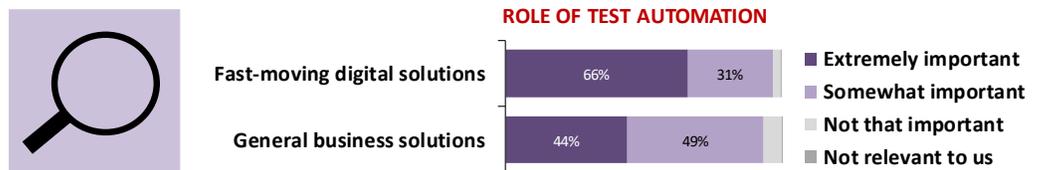
We'll be looking at the rationale for applying continuous delivery to applications with a slower cadence a little later in this report. Suffice it to say for now that benefits in terms of increased quality and efficiency appear from this research to be at least as attractive as simply speeding things up.

In the meantime, let's circle back to the main theme of software testing, and consider how the game is changing in that space.

Testing in the spotlight

Again, the research confirms something most of us could have guessed, but it is good to see coming through so positively anyway. In this case, it's a broad recognition of the importance of automation in the context of software testing, especially in relation to fast-moving digital solutions (Figure 6).

Figure 6
How important is it to automate testing in relation to the following?



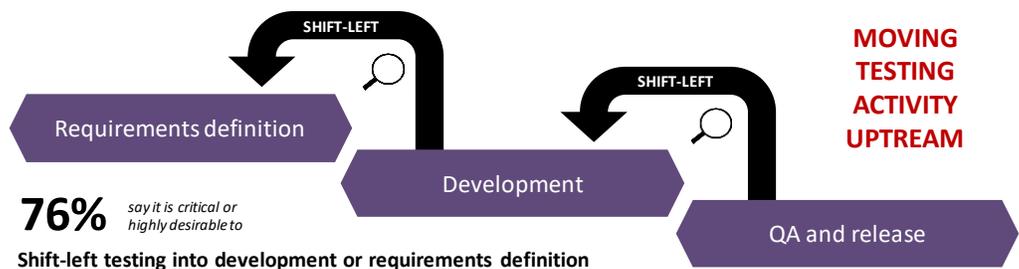
This automation imperative becomes particularly obvious when you consider that in a continuous delivery environment you should be aiming to embed testing as a fundamental and ongoing aspect of every activity throughout the delivery cycle – a practice increasingly referred to as ‘continuous testing’ (Figure 7).

Figure 7
How important is continuous testing to meeting evolving business needs and expectations?



One of the core principles of continuous testing is moving as much QA as possible to take place upstream in the delivery cycle, the idea being to ‘test early and often’. The term generally used for this is ‘shift-left’, and the prevailing view is that it's desirable, even critical, to extend testing activities not just into development, but right back into the requirements definition part of the cycle (Figure 8).

Figure 8
When implementing continuous testing, how important are the following?



A related finding from the study is that 63% highlight the importance of combining Agile development with Continuous Delivery, as the latter puts a strong focus on effective

Defining tests at the requirements stage, is not just more efficient, it also means a 'pass' is more likely to be relevant and meaningful in business terms.

integration of activities throughout the whole lifecycle. The point here is that focusing on Agile is not enough. Nor is just integrating with downstream activity, i.e. from development into release, deployment and operations. Upstream integration is critical too, right back to where requirements are being discussed and defined in collaboration with stakeholders. And the more you can also define relevant tests at this stage, the more likely it is that a 'pass' will mean something useful and relevant in business terms.

Important though shift-left and continuous testing are seen to be, study respondents also acknowledge that traditional testing facilities are often not geared up to enable this way of working, and you need the right tools applied in the right way (Figure 9).

Figure 9

How much would you agree or disagree with the following statements?



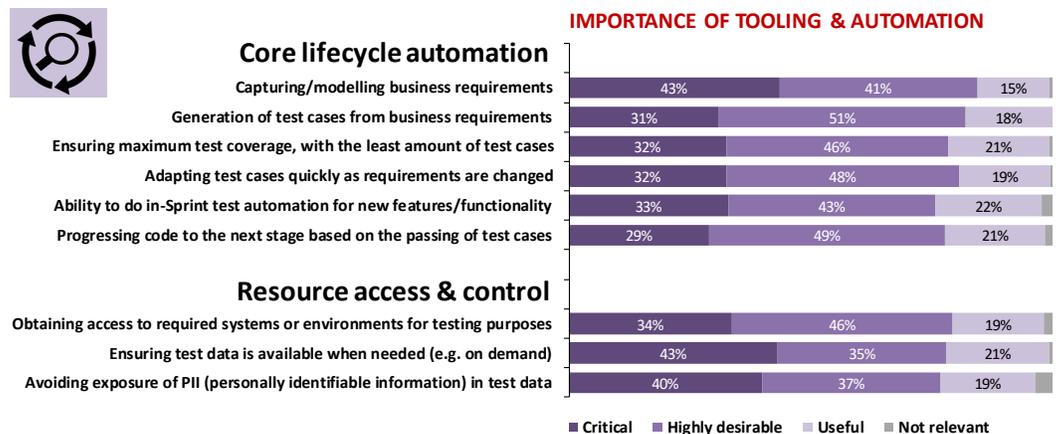
QA is no longer the sole responsibility of the testing team.

The mention of shifting mindset and culture here is noteworthy, as in a continuous delivery and continuous testing environment, QA is no longer the sole responsibility of the testing team – everyone becomes involved to one degree or another.

Drilling into more specific capabilities, the importance of testing and quality related automation is highlighted across a number of key functions (Figure 10).

Figure 10

And how important for continuous testing is good tooling and automation in these areas?



As part of this picture, it is comforting to see an appreciation of the need to deal with the question of test data management effectively.

The danger is that development and testing teams are tempted to take risks to speed up test data acquisition.

The requirement for speed and quality here is clear, i.e. to rapidly acquire/generate and refresh high quality test data sets that reflect the real world, but it's important not to let this overshadow the need for appropriate controls. The danger is that development and testing teams are tempted to take risks to speed up test data acquisition, e.g. by extracting data from live systems without taking proper steps to assure security and compliance. And it's not just potential reputational damage that we need to consider if things go wrong. Ever stricter privacy legislation, such as the EU's imminent General Data Protection Regulation (GDPR), means that fines relating to the mishandling of personally identifiable information in particular can lead to serious financial consequences.

It's not good enough to know what's important; you also need to act.

But knowing what's important is not the same as acting on the imperatives, so how have our study participants progressed in creating a modern continuous testing environment?

Progress towards the goal

Pulling together some of the threads we have been discussing, we can define a simple, high-level view of what needs to be in place to implement continuous testing effectively (Figure 11).

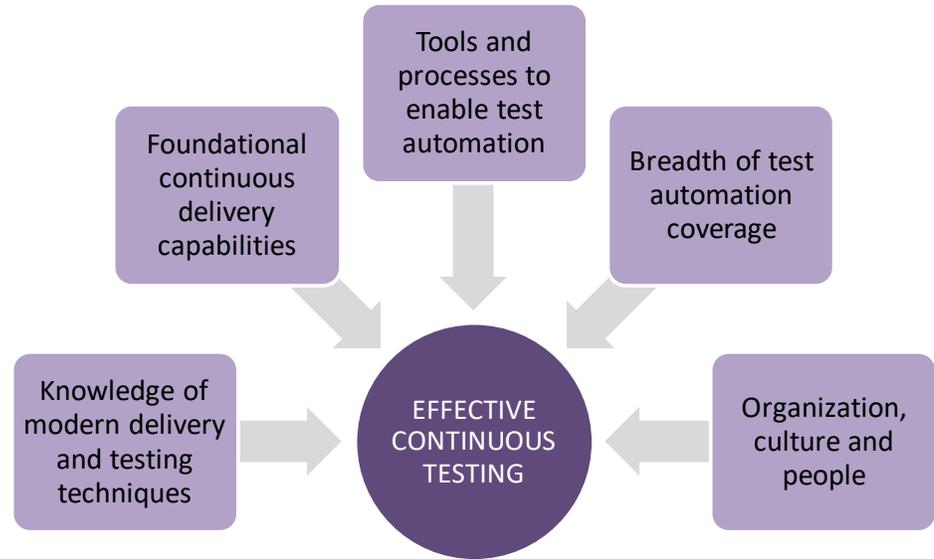


Figure 11
Driving success with continuous testing

Firstly, you need to be informed on the latest techniques and best practices. In the majority of cases, you must also be building on a reasonably solid foundation of more generic continuous delivery capability. Implementing basic test automation is worthwhile without this, but full continuous testing makes a lot more sense in a continuous delivery environment. We have already touched on specific tools and processes, but another key question is how broadly they are applied – hence the mention of test automation coverage. Lastly, building on the earlier discussion, we have a range of organization, culture and people-related requirements to consider.

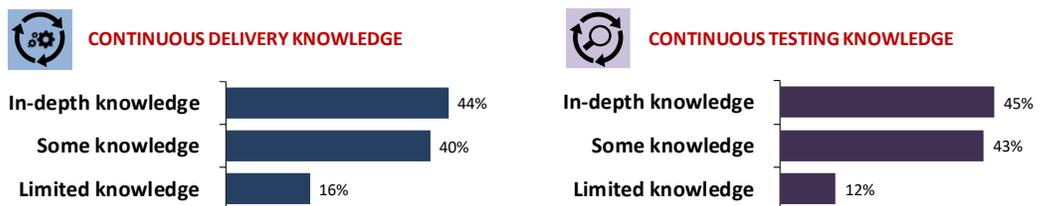
A balanced and holistic approach is necessary for the effective implementation of continuous testing.

So, let's take each of these areas in turn, and see how well study participants shape up in relation to each. Once we have done this, we'll look at how capability overall impacts the level of results achieved in line with relevant business outcomes.

Knowledge of modern techniques

In line with the advanced nature of the sample, the level of knowledge reported is very respectable in relation to both continuous delivery and continuous testing, albeit a little variable (Figure 12).

Figure 12
How familiar are you with these concepts?



Suppliers are often very willing to discuss requirements and fill in knowledge gaps as you explore the value of their solutions.

Suppliers have the advantage of being able to look across many different customer scenarios.

The important thing to bear in mind with both of these topics is that they are relatively easy to understand at a high level. Indeed, if you explain the concepts to experienced IT professionals who have never heard them before, they generally ‘get’ the idea pretty much straightaway. This can sometimes lead to a false sense of confidence, however. The reality is that success depends on effectively dealing with detailed process and tooling requirements at the next level down.

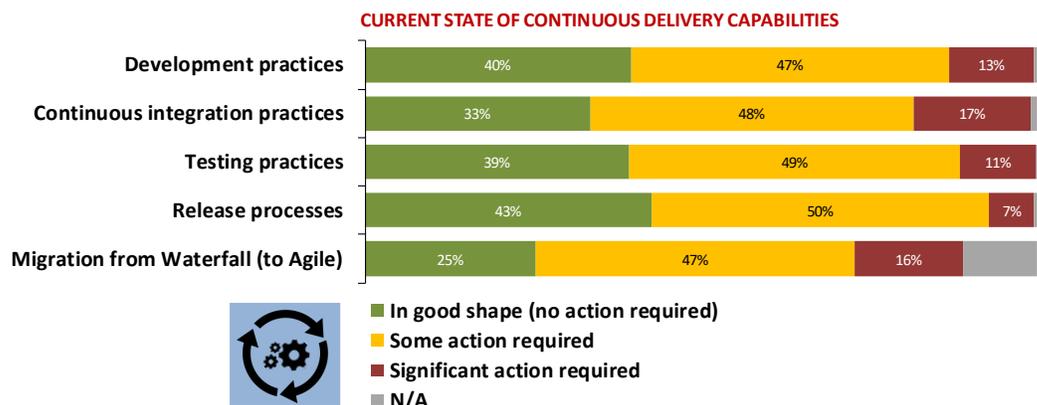
The good news is that many sources of information exist to help you get up to speed, not least some of the relevant open source communities, many of which are very lively.

It’s worth noting that commercial suppliers are often very willing to discuss requirements and fill in knowledge gaps as you explore the value of their solutions. Key players often make highly experienced consultants available during the buying cycle, whose expertise goes way beyond the tools being offered. These individuals have the advantage of being able to look across many different customer scenarios, so they will typically have a good sense of what’s likely to work best in your environment. And getting input from multiple suppliers who can each contribute a different perspective is a great way of getting a rounded view of both your requirements and the ‘art of the possible’.

Foundational continuous delivery capabilities

Looking at the picture in relation to generic continuous delivery capability, while some claim to be in very good shape, many tell us they have work to do (Figure 13).

Figure 13
How much does your company need to drive change and/or improvement in the following areas to achieve full continuous delivery capability?



From a continuous testing perspective, this picture highlights that some organizations might struggle to implement continuous testing based on their current state of maturity.

For a more detailed discussion of continuous delivery principles, processes and tools required for success in this area, please see our separate paper entitled “Taking DevOps to the next level; It’s all about continuous delivery” (available from the Freeform Dynamics website). Suffice it to say for now that one of the big requirements is a framework that enables end to end integration up and down the entire delivery chain. In many respects, continuous testing simply extends this idea, though in itself it needs a more specific set of tools, integrations and practices.

Testing tools and practices

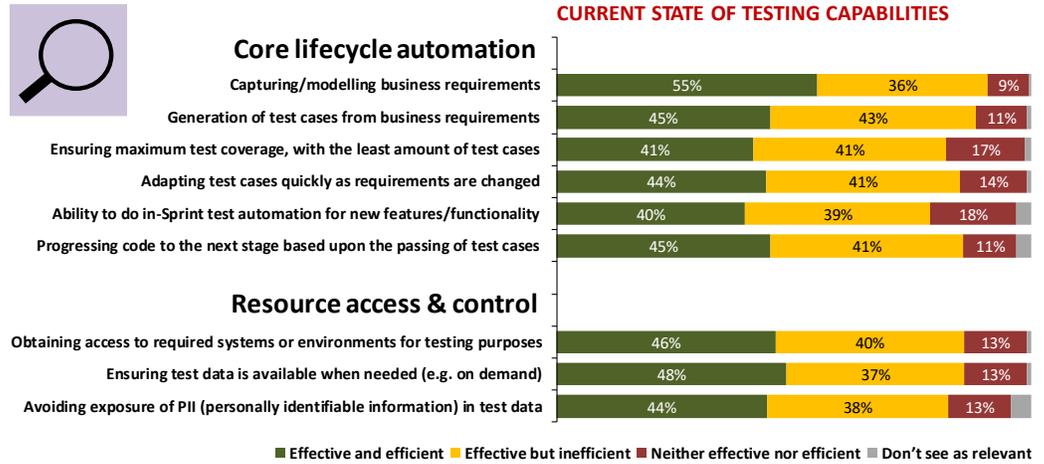
Turning to testing-specific capability, most study participants say that what they have in place already does the job. This isn’t surprising given that you need to take care of pretty much everything previously listed in one way or another.

Continuous testing builds on and extends the idea of continuous delivery.

Efficiency gaps prevail.

Figure 14
Thinking of your current testing tools and processes, particularly your reliance on manual vs automated activity, how would you sum up your current capability in the following areas?

The more pertinent question in the context of continuous testing, however, is process efficiency. In this respect, indications are that even in our more advanced sample, many are still relying on inefficient manual processes (Figure 14).



But whatever the claims about tooling and process efficiency, what really matters is how much automation is actually taking place.

The variation in test automation coverage is extremely wide-ranging, with many still relying on manual processing for much of their testing activity.

Test automation coverage

This next survey finding is one of the most important ones in the whole study. When you cut straight to the heart of one of the key requirements and look at how much testing activity is automated in practice, the variation in responses is extremely wide-ranging.

Focusing on fast-moving digital applications, where the need for automation is arguably most acute, only one in five indicate very high coverage, i.e. where 80% or more of test procedures are automated (Figure 15).

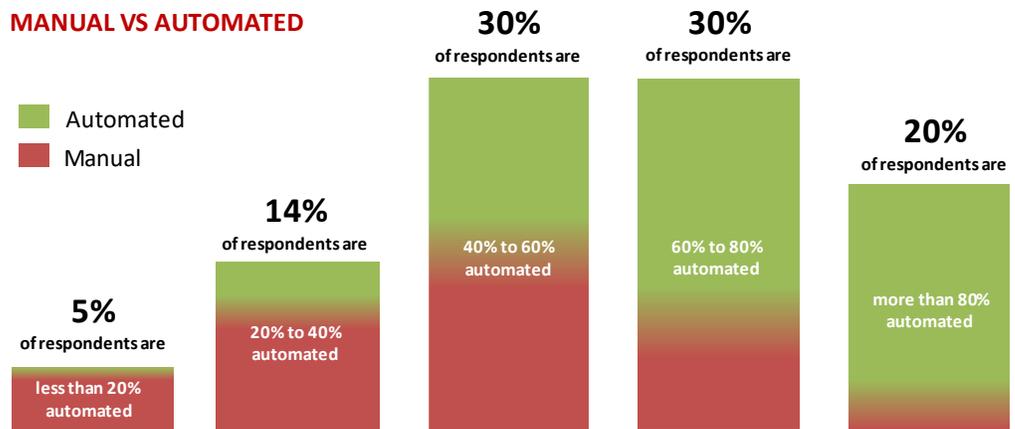


Figure 15
What percentage of your testing procedures are automated in relation to fast-moving digital applications?

The picture we see here suggests a continuing reliance on manual testing processes by many, which is clearly far from ideal. Apart from the additional time and effort needed to design, maintain and execute test procedures, manual processing inevitably leads to more human errors, which means yet more time and effort spent on remediation. All in all, manual testing is generally bad news when it comes to delays, costs and disruption.

Zooming out, this data provides yet more confirmation that many are still some way off having enough basic building blocks in place to implement continuous testing effectively.

Many are still some way off having enough basic building blocks in place.

Driving cultural change can be challenging.

Figure 16
How would you sum things up in these areas?

Organization, culture and people

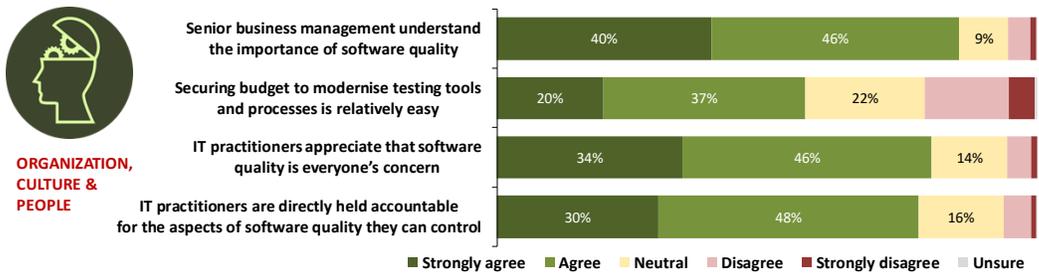
Environmental and organizational questions were touched on at two points in the survey questionnaire. Firstly, in relation to continuous testing readiness; while respondents on average are making good progress with regard to team collaboration, driving cultural change is proving more elusive (Figure 16).



Also evident from this chart is the fact that it can be hard to gain management buy-in to support quality-related plans and activities. Turning to another set of questions in the survey, we then see a familiar picture, in which senior business management might appreciate the need, but are less forthcoming in their willingness to fund the investment required to meet it (Figure 17).

It can be hard to gain management buy-in.

Figure 17
How much would you agree or disagree with the following statements?



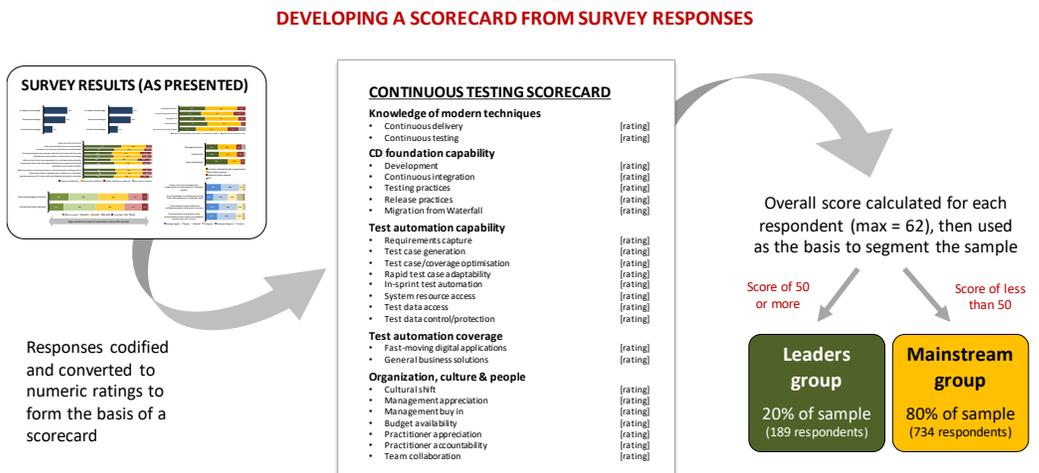
According to responses, IT practitioners themselves are generally reasonably quality-aware, with many of them assuming at least some level of responsibility and accountability in this area. In the context of an advanced sample filling out an online survey, however, it should be noted that the difference between ‘Strongly agree’ and ‘Agree’ can be quite significant. This brings us to our continuous testing scorecard.

The scorecard approach is great for assessing current capability in key areas.

Assessing overall effectiveness

In order to understand the relationship between capability and results, we derived a scorecard from the survey responses, and used this to segment respondents (Figure 18).

Figure 18
Scorecard development and segmentation of research respondents based on their continuous testing capability



Continuous testing
‘Leaders’ out-perform
their counterparts
significantly across a
number of key areas.

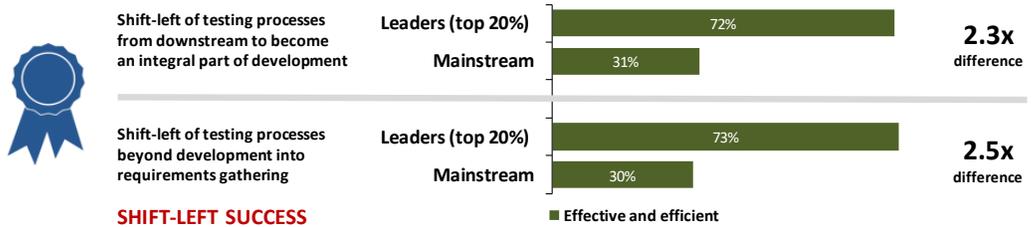
The scorecard we used is presented in its full form in Appendix B, and we will later discuss how you can use this to assess your own position as a starting point for driving improvement.

In the meantime, pulling out the top 20% of respondents based on the scorecard results allows us to get a feel for the kind of advantages that good continuous testing capability translates to in terms of outcomes. In quite a few areas these ‘Leaders’ outperform their counterparts in the ‘Mainstream’ group (comprised of the other 80% of the sample).

Significantly greater ‘shift-left’

On average, Leaders are 2.3x more likely to have succeeded in shifting testing activity upstream into the development environment, and the difference is 2.5x when it comes to left-shifting back into the requirements domain (Figure 19).

Figure 19
How would you sum up your current capability in the following areas?



This kind of left-shifting provides Leaders with a big advantage because they can start to think of QA and testing from the outset in relation to any project. Not only does this provide advantages in terms of trapping issues and anomalies early (making them cheaper to address), it also means that testing activity is more likely to be directly aligned with business needs and objectives.

Left-shifting provides Leaders with a big advantage because they can start to think of QA and testing from the outset on any project.

From an automation and integration perspective, Leaders are undoubtedly exploiting the ability of modern tools to automatically generate test cases directly from requirements specifications, ensuring maximum coverage with the minimum number of test cases. Being able to view testing activity and results through a business lens is then great for providing meaningful stakeholder visibility of status, progress and risk. There’s far less need to reverse engineer technical definitions before going into a review meeting.

Let’s also not forget that shift-left is equally important downstream in the delivery cycle. For example, embedding tests relating to operational readiness into the release process rather than running them post deployment can save a lot of time and distraction.

Leaders are much more confident in their ability to handle quality and speed requirements.

Significant boost in confidence

As a result of shift-left, but also the minimization of errors and increased throughput that stems from automation in general, Leaders report a 2.4x and 1.9x boost in confidence in relation to quality and speed of delivery respectively (Figure 20).

Figure 20
Confidence in quality of output and speed of delivery



More applications becoming business critical leads to a greater intolerance of failure, and more acute impatience in relation to speed of delivery.

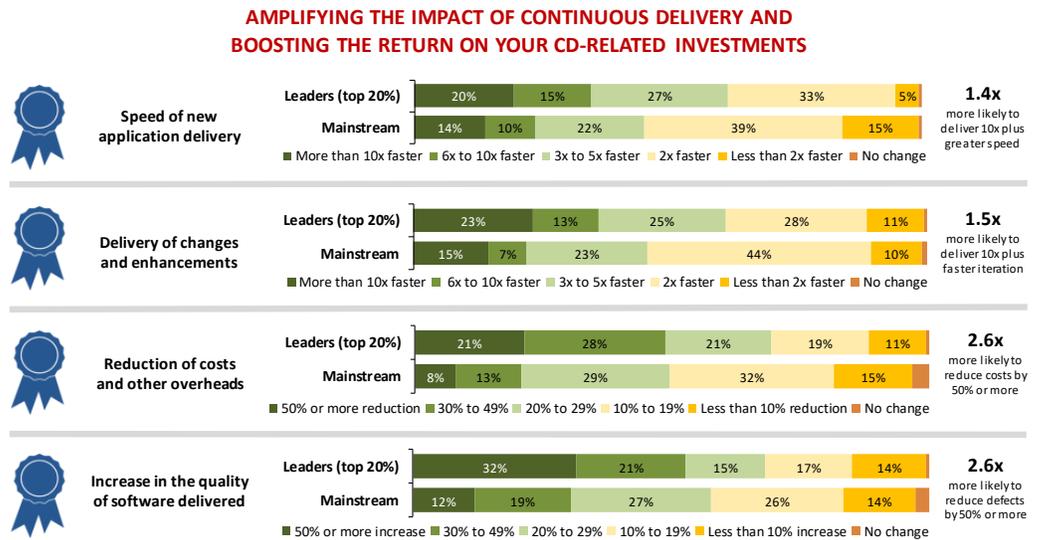
These increases in overall confidence are particularly noteworthy for anyone involved in digital transformation. As everything from customer engagement systems, through employee empowerment solutions, to core operational business processes become increasingly digitized, this inevitably leads to higher expectations. More applications and services become genuinely business critical, which in turn results in a greater intolerance of failure, and more acute impatience in relation to building and rolling out both new applications, and application fixes and enhancements.

But is this increased confidence justified? After all, it's easy to provide an optimistic view when completing an online survey. Circling back to some of the advantages stemming from continuous delivery in general, it isn't difficult to find corroborating evidence.

More effective continuous delivery

So far, we have largely discussed continuous delivery as a foundation and prerequisite for implementing continuous testing. Turning this on its head, a key question is whether implementation of test automation and continuous testing drives greater returns from your continuous delivery efforts and investments at a higher level. The evidence suggests that it does (Figure 21).

Figure 21
What kind of improvement have you experienced (or would you expect) on average from the adoption of continuous delivery (compared to traditional methods)?



There's a lot on this chart, but it's worth taking a minute to study it and work out what it is telling you. In essence, what we are seeing is that returns from continuous delivery implementation are boosted significantly in relation to a number of important outcomes.

Implementation of test automation and continuous testing drives greater returns from your continuous delivery efforts and investments.

Speed of delivery is enhanced for both new applications and the rollout of application changes and enhancements. The difference we see between Leaders and Others here probably doesn't fully reflect what can be achieved in relation to fast-moving applications as we specifically asked for the 'average' improvement across the board. Low cadence applications simply won't be taking advantage of the potential delivery speed increases, which will bring the average down.

All applications, however, can benefit from increases in efficiency of delivery and quality of output. The combination of shift-left testing (trapping and fixing issues early) and the elimination of manual processing (less time and resources, and fewer human errors to resolve) really can have a dramatic impact on cost reduction and the number of defects making it through to production. This is something to bear in mind when formulating the

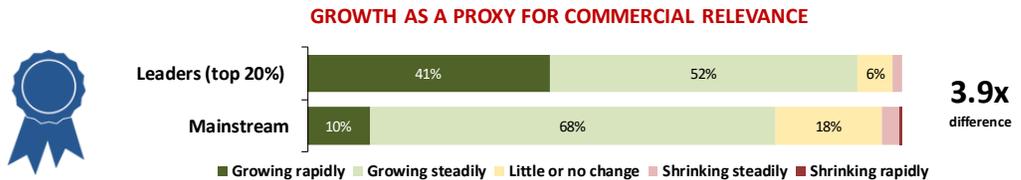
Continuous testing is one of those things that successful companies implement as part of their ongoing quest for competitive advantage.

business case for investment to modernize your testing environment, as unlike speed, payback here potentially comes from your whole custom application estate.

Alignment with revenue growth

The last correlation we will look at in relation to our Leader group is in the area of revenue growth. Clearly, it is hard to draw a direct link between continuous testing capability and overall company performance, but a difference of almost 4x is unlikely to be pure coincidence (Figure 22).

Figure 22
How is your company's revenue changing?



What's going on here is an alignment between progressive behavior and success. Agile organizations with a change friendly culture tend to naturally invest in advanced technology solutions that help them compete more effectively. Continuous testing, almost certainly as part of any continuous delivery approach, is simply one of those things that successful companies implement as part of their ongoing quest to gain and maintain competitive advantage.

We would therefore encourage you to explicitly assess your current testing environment and where necessary prioritize the relevant investments and transformations.

Final thoughts and recommendations

The results of this study confirm that continuous delivery is key to success when it comes to delivering software effectively in today's fast-moving digital environment. The findings also provide clear evidence that you are unlikely to enjoy the full benefits of continuous delivery unless you modernize your testing tools and practices. We would therefore encourage you to explicitly assess your current testing environment and where necessary prioritize the relevant investments and transformations. If dealing with testing requirements is an afterthought, it's more likely that you will end up with suboptimal solutions, and get stuck in perpetual catch-up mode.

In order to help you on your continuous delivery and continuous testing journey, the scorecard we presented earlier has been captured in its full form in Appendix B. We have deliberately not dictated the way in which you should weight and score the individual elements because what's appropriate will vary significantly from company to company. What's most important is using a scheme that's both meaningful and practical for you to populate with actual values. Even if you use a traffic light approach, you'll quickly get a feel for where gaps and disjoints exist.

Quality and testing in today's fast-moving continuous delivery environment is quickly becoming everyone's responsibility, so management need to ensure their people are properly enabled.

Our last piece of advice while you do this is to remember that software quality and testing in today's fast-moving continuous delivery environment is quickly becoming everyone's responsibility. Whether you are a business analyst, developer, release manager or operations professional, you have a role to play in the quality chain. Based on the principle that responsibility without control is ineffectual, however, it's incumbent on business and IT management to ensure that everyone is appropriately enabled.

Appendix A

Research Sample

The study upon which this report is based was designed, executed and interpreted by Freeform Dynamics Ltd in collaboration with CA Technologies. Data was gathered from 923 respondents via an online survey completed in Spring 2017.

The sample distribution for the online survey was as follows:

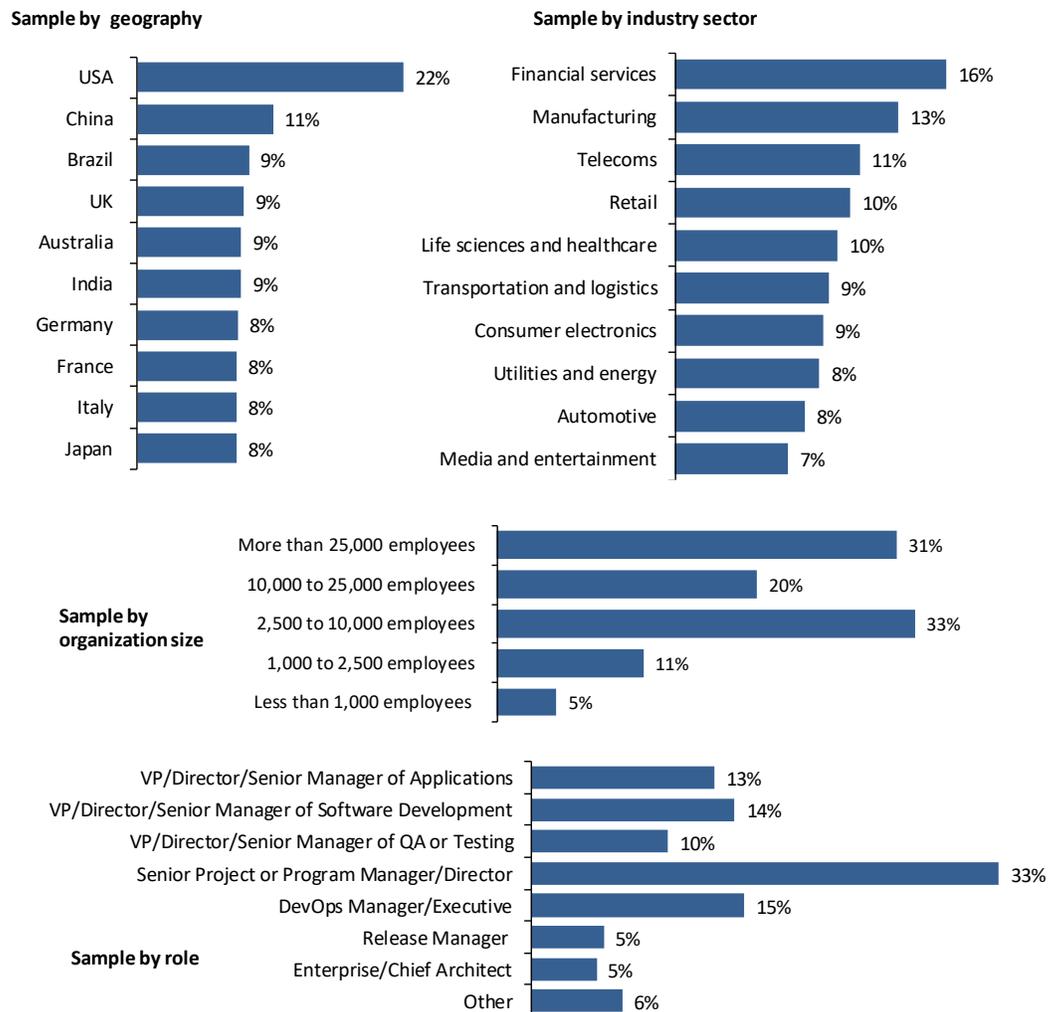


Figure 23
Composition of research study sample

Note on methodology

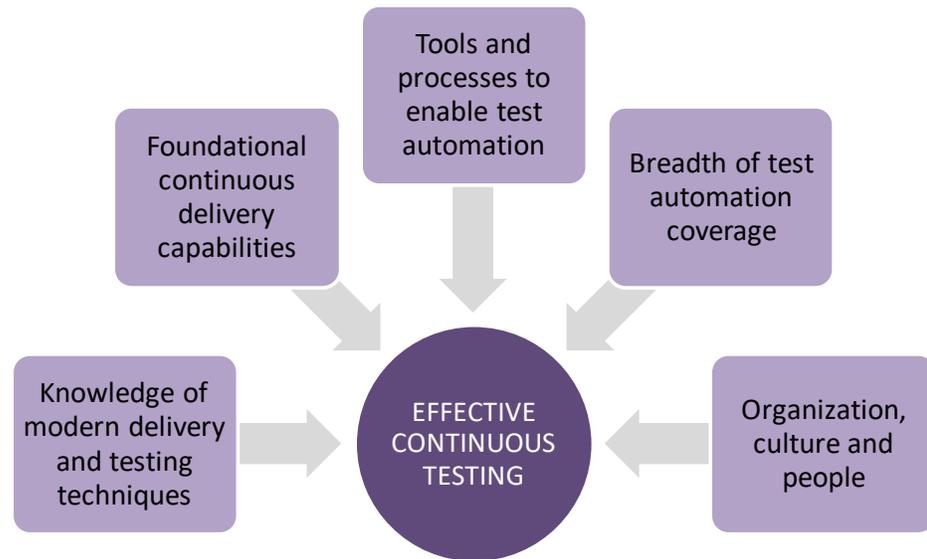
The survey was conducted online and respondents ‘self-selected’ into the study. We must therefore be aware of possible sample bias towards more advanced respondents who are generally more enthusiastic and more likely to respond to a research call to action. This does not affect the commentary or conclusions contained in this report, but should be borne in mind when considering the data in another context.

Appendix B

The Continuous Testing Scorecard

In the main body of this report, we highlighted a number of factors that determine success with Continuous Testing. Here is a reminder of what these are (Figure 24).

Figure 24
Driving success with continuous testing



While it's impossible to be exhaustive when reviewing enablers in this way, and we of course also need to recognize that a degree of generalization and simplification is unavoidable, the research suggests that this is a pretty good starting point to conduct a high-level readiness assessment.

With this in mind, we have adjusted some of the questions we asked in our survey so you can use it to form a meaningful view of your own position. The result is the scorecard presented on the next page, which has been designed to be an actionable, working document for you to use in your review and planning activities.

As we put this together, we took care to maintain consistency of scoring and weighting with the scheme used in the original research. This means once you have filled out the scorecard and totaled your ratings, you'll be able to compare yourself with our study sample. If you achieve a score of 50 or more, then you can congratulate yourself as this puts you in the 'Leader' category – i.e. the top 20%. If you get 40 or above, you can feel good about being above average, but that doesn't mean you don't have work to do. Obviously if you score below 40 then you are behind the curve compared to your counterparts, so maybe driving improvements in the way you manage software quality should be placed higher on your investment agenda.

So, while as a byproduct of a research a study, the scorecard clearly doesn't represent a definitive assessment tool, we hope it will at least allow you to form a quick view of where you are and provide useful input in your planning and transformation activities.

CONTINUOUS TESTING READINESS SCORECARD

Knowledge of modern techniques

How familiar are these concepts to your organization?

Score *Max*

Continuous delivery	<input type="text"/>	5
Continuous testing	<input type="text"/>	5

Rate the level of knowledge that exists on a scale of 0 to 5, where 0 = None and 5 = Extensive

Foundational continuous delivery capabilities

Thinking about what's needed to become fully effective in your use of continuous delivery, how would you sum up the state of your current practices in these areas?

Score *Max*

Software development	<input type="text"/>	2
Support for agile methods	<input type="text"/>	2
Continuous integration	<input type="text"/>	2
Software testing	<input type="text"/>	2
Release management	<input type="text"/>	2

Score yourself for each item in the list as follows:

2 = In good shape in this area
1 = Some improvement required
0 = A lot of improvement needed

Test automation and enablers and capabilities in more detail

Turning to test automation pre-requisites and enablers, how would you rate your current capabilities in the following areas?

Score *Max*

Capturing/modelling business requirements	<input type="text"/>	2
Generation of test cases from business requirements	<input type="text"/>	2
Ensuring maximum test coverage, with the least amount of test cases	<input type="text"/>	2
Adapting test cases quickly as requirements are changed	<input type="text"/>	2
Ability to do in-Sprint test automation for new features/functionality	<input type="text"/>	2
Progressing code to the next stage based upon the passing of test cases	<input type="text"/>	2
Obtaining access to required systems or environments for testing purposes	<input type="text"/>	2
Ensuring test data is available when needed (e.g. on demand)	<input type="text"/>	2
Avoiding exposure of PII (personally identifiable information) in test data	<input type="text"/>	2

Rate what's in place for each item in the list based on the following:

2 = Effective and efficient
1 = Effective but inefficient
0 = Severely lacking in general

Test automation coverage

What percentage of your test automation procedures are automated in relation to the following types of application?

Score *Max*

Fast-moving digital solutions	<input type="text"/>	5
General business solutions	<input type="text"/>	5

Score on automation coverage:

5 = 80% plus 2 = 20 to 40%
4 = 60 to 80% 1 = Less than 20%
3 = 40 to 60% 0 = All manual

Organization, culture and people

How would you rate your company in relation to the following?

Score *Max*

Achieving the necessary cultural and mindset shift	<input type="text"/>	2
Management appreciation of the need for software quality	<input type="text"/>	2
Securing management buy-in to implement modern methods	<input type="text"/>	2
Securing the budget required to fund necessary improvements	<input type="text"/>	2
IT practitioner appreciation of need for software quality	<input type="text"/>	2
Individual IT practitioner accountability for quality of output	<input type="text"/>	2
Effective team collaboration on testing activities	<input type="text"/>	2

Allocate scores for each item in the list as follows:

2 = Full achievement of goal
1 = Partial achievement of goal
0 = Little or no achievement

HOW DID YOU DO?

Add all of your scores to derive your continuous testing readiness rating and compare yourself to our study sample

TOTAL *Max*

62

50 or more means you're a Leader
40 or more is above average
Less than 40 is below average

About Freeform Dynamics

Freeform Dynamics is an IT industry analyst firm. Through our research and insights, we aim to help busy IT and business professionals get up to speed on the latest technology developments, and make better-informed investment decisions.

For more information, and access to our library of complementary research, please visit www.freeformdynamics.com.

About CA Technologies

CA Technologies (NASDAQ: CA) creates software that fuels transformation for companies and enables them to seize the opportunities of the application economy. Software is at the heart of every business in every industry. From planning to development to management and security, CA is working with companies worldwide to change the way we live, transact, and communicate—across mobile, private, and public cloud, distributed and mainframe environments. Learn more at www.ca.com.

Terms of Use

This document is Copyright 2017 Freeform Dynamics Ltd. It may be freely duplicated and distributed in its entirety on an individual one to one basis, either electronically or in hard copy form. It may not, however, be disassembled or modified in any way as part of the duplication process. Hosting of the entire report for download and/or mass distribution by any means is prohibited unless express permission is obtained from Freeform Dynamics Ltd or CA Technologies. The contents contained herein are provided for your general information and use only, and neither Freeform Dynamics Ltd nor any third party provide any warranty or guarantee as to its suitability for any particular purpose.