



Our DevOps Approach

35%

of respondents said a lack of awareness of the business benefits was the greatest challenge that prevented them from adopting DevOps



At a time when IT organisations need to be operating in a faster, more seamless manner to help companies to thrive in more competitive markets, we can explain how introducing Edge's DevOps service can transform the way your company operates.

Implementing our DevOps service can transform your business processes, creating both value and cost savings.

The benefits include:

- Faster time-to-market/delivery times that improves your ROI
- Reduced support costs whilst increased quality and reliability of your service
- Early detection and faster correction of defects that helps provide the best service for your customers
- Improved collaboration between teams by improving the transparency required for effective decision making
- Transferring of team skills which enables constant learning and improvement

The challenges for testing as part of DevOps:

- A more dynamic landscape, needing better understanding of testing and development methods to decide what to test at any given point in time
- Closer working with operations, meaning a new 'language' to speak and new tools to understand
- A greater need to automate to handle the rate of change

How We Can Help

- Edge can provide your teams with the skills to meet these challenges, help them work in a more collaborative way and achieve the cultural change that is needed to support the DevOps philosophy
- We provide consultancy on understanding these challenges which will be catered for your own specific needs. In addition, we can help you manage the implementation and provide knowledge of the best tools to use

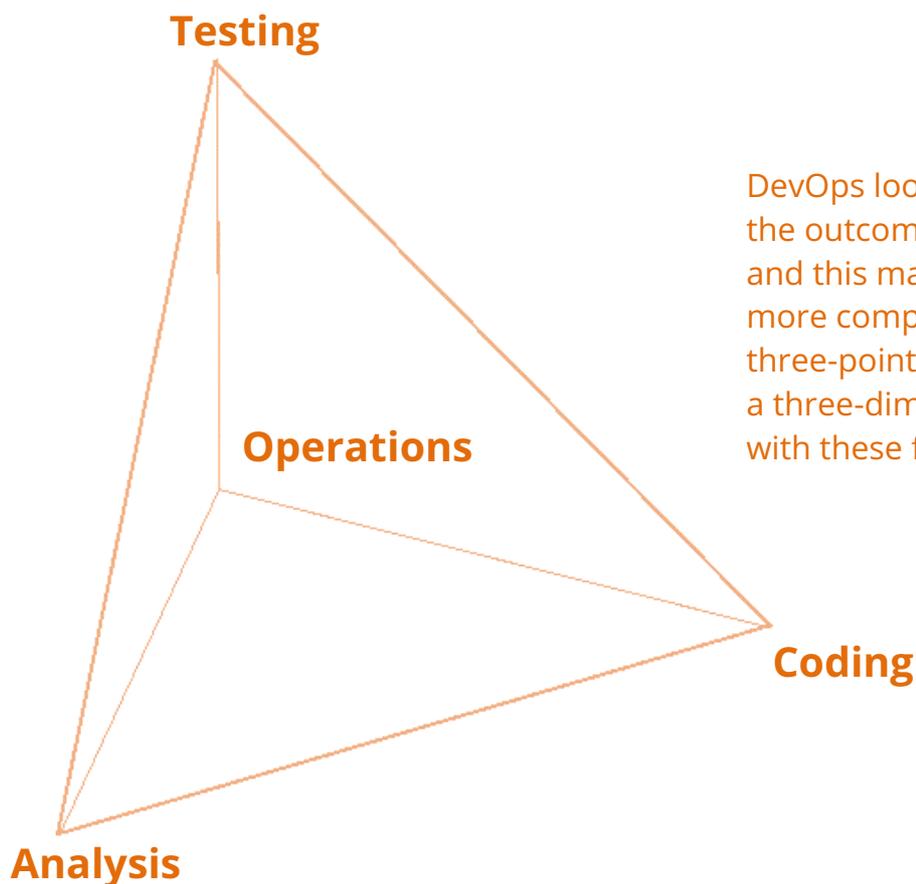
Our Approach

Within Edge we always push to be ahead of the wave with new approaches to development. We are utilising our depth of experience in test architecture, automation and other technical disciplines to support our clients in delivering quality in the demanding and accelerating context of DevOps.

Projects used to operate in a bubble to a greater or lesser extent, and a three-faceted view of activity was often used to understand what needs to be factored when developing a new product or system. Typically, this was some variation on:

- **Analysis**
- **Coding**
- **Testing**

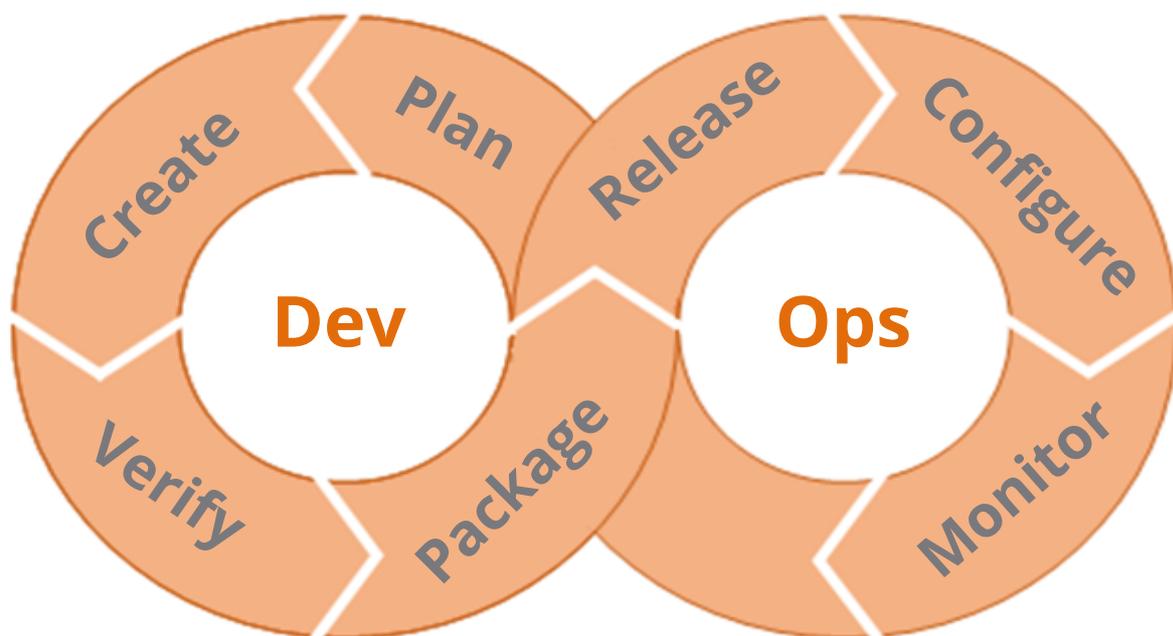
Indeed, agile has a practice known as 'the Three Amigos' that promotes this view of project work.



DevOps looks to bring the project and the outcome even closer together, and this makes the problem space more complex. Rather than a flat and three-pointed challenge we now have a three-dimensional problem space with these four cardinal points.

This shift away from contained projects to a more complex landscape of delivery chains impacts all aspects of delivery and creates new challenges for testing and assurance.

As an industry, we have already adapted to agile methods introducing shorter timescales and more frequent delivery; now we need to go a step further and push quality activity out into the full chain as a coherent activity.



The temptation when moving to a DevOps approach is to think that testing is limited to the “verify” link of the chain. On the contrary, quality needs to be embedded throughout the process in a meaningful way

plan



Whilst we prepare for another cycle of releasing software need to understand the implications for quality and testing, both in terms of what kinds of tests the delivery products will need but also whether any elements of the delivery process itself need enhancing to ensure quality outcomes

create



Throughout the creation activity we need to continually assess the products we are developing. This necessitates a focus on quality and effective implementation of techniques like TDD or BDD. Automation of low-level or discrete functionality is likely to be the initial focus here, however we must also address non-functional aspects of the system

verify



As well as assessing discrete elements of the system we also need to consider how it functions as a whole. Verifying broader use cases and user journeys should be our focus here, though the tools or techniques may be similar to those deployed during the creation of new assets. Ideally, we would blend activity with Create to maximise the benefits

package



The software product must be reliably deployable, so it is essential to test that. We also need to verify that builds can be rolled back if necessary

release



The software product must be reliably deployable, so it is essential to test that. We also need to verify that builds can be rolled back if necessary

Configure



Many systems can be greatly affected by configuration, potentially changing functional and non-functional behaviour through static or metadata. If there will be adjustments to configuration post deployment these also need testing appropriately

monitor



Monitoring is a process of continually testing the system through the use of automated or passive measures. We can help define and interpret these. We can also assess their impact on future solution requirements, ready for the next loop of development and operation

This more complex landscape demands higher capability to understand what should be done at each stage of the process, and react dynamically to those changing needs. It also requires a broader range of skills to interact and with and align to the delivery tools being used by a DevOps team.