White Paper, sponsored by LaunchDarkly | May 2024

Optimizing Software Delivery by Harnessing the Full Potential of Feature Flags

Jim Mercer
Research Vice President,
DevOps & DevSecOps, IDC

Katie Norton
Senior Research Analyst,
DevOps & DevSecOps, IDC

Table of Contents

CLICK ANY HEADING TO NAVIGATE DIRECTLY TO THAT PAGE.

IDC Opinion

The digital business era is upon us. Enterprises have already made progress in tackling digital transformation and are now transitioning to running viable digital businesses that rely on software applications to generate notable returns. As such, organizations must reconsider how they release their software since traditional methods will not provide the required capabilities, controls, and agility to be competitive in an unpredictable digital marketplace. Organizations must shift from ceremonial releases to more progressive, granular, agile, and feature-focused delivery strategies.

Feature flags are a key tool for enabling progressive delivery strategies. A feature flag (aka a feature toggle) allows control of new features and how they get implemented. Development teams can use the logic in feature flags to selectively turn portions of application code on and off to test new features without deploying new code. This approach empowers teams to gather valuable user feedback, make data-driven decisions, and respond quickly to any problems that may arise during the rollout, ultimately improving the overall quality and stability of the software.

To examine businesses' real-world experiences using feature flags, IDC interviewed three LaunchDarkly customers operating in different industries. All the customers operated complex software development and application environments across various sites and domains. These interviews probed areas such as driving forces, solution evaluation, application deployments, and lessons learned to understand how they used feature flags to enhance their ability to deliver new software innovations into the marketplace.

It should be noted that while the interview subjects were largely in software development, product management, and IT, there was much consideration for business outcomes driven by the use of LaunchDarkly feature flags and the initiatives they supported or made possible.

In This White Paper

This IDC White Paper provides some insights into some of the measurable business benefits highlighted by the LaunchDarkly customers, outlining four key feature flag use cases these customers identified:

- **De-risking releases and migrations** via controlled efforts where deployments are decoupled from releases
- Using **product experimentation** to turn application features on and off for groups of users and test different versions of a feature with different groups
- **Targeting and personalizing experiences** to empower product delivery teams with the confidence to deliver the right experiences to the right audiences
- **Optimizing mobile releases** by decoupling the process of deploying a feature to the app store from releasing that same feature to end users

Woven into each use case are LaunchDarkly's customer case studies:

Paddle is a London-based payment infrastructure provider with the mission of helping SaaS companies navigate the revenue journey at every stage. Founded in 2012, Paddle has grown to over 300 employees — over 100 in product and engineering. The Paddle all-in-one solution includes checkout, payments, reporting, tax, and compliance. Over 4,000 companies from around 245 countries and territories across the globe use Paddle to offload their operational complexities so that they can focus on their growth.

Vodafone is a leading technology communication company that serves more than 18 million mobile and fixed-line customers in the United Kingdom. Headquartered in Newbury, Berkshire, Vodafone plans to grow significantly to support its expanding digital services portfolio. By 2025, the company expects over 50% of all Vodafone employees to work in software engineering.

iPipeline is a leading provider of cloud-based software solutions for the life insurance and financial services industries. The company provides a SaaS-delivered solution that helps end customers accelerate and simplify insurance sales, compliance, operations, and support. iPipeline is headquartered in the United States, and of its 841 worldwide employees, nearly 70% are focused on application development or IT operations management. The iPipeline DevOps teams, which are geographically dispersed in the United States and the United Kingdom, support customer-facing applications with over 600,000 registered users globally.

Situation Overview

IDC data shows that modern digital enterprises are leaning toward progressive delivery. In a recent IDC's DevOps Survey, around two-thirds of respondents indicated they are expanding, using, or piloting progressive delivery, with 45% indicating they are using feature flags to do so (see **Figure 1**, next page).

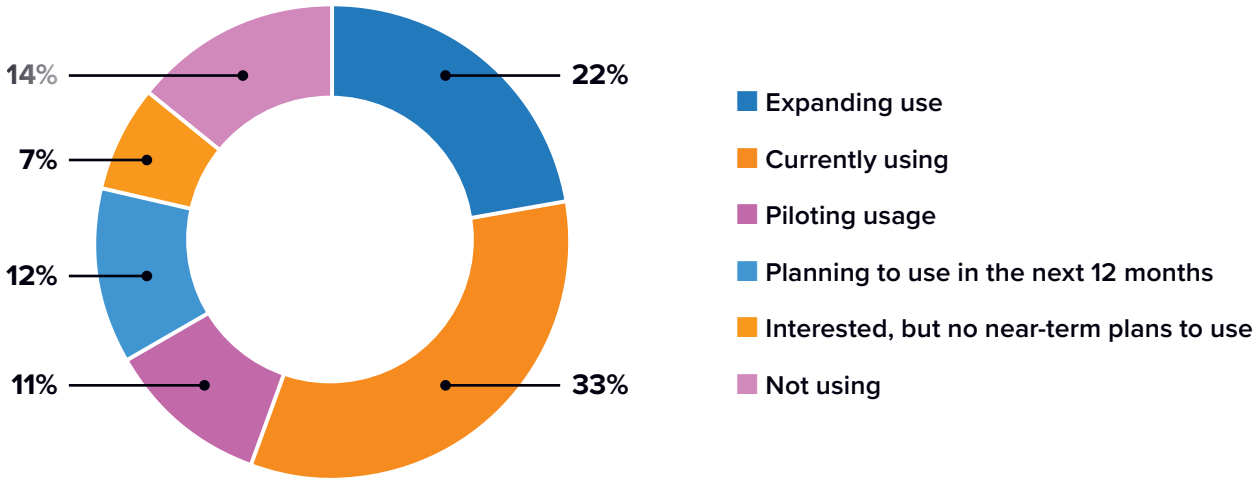
Feature flags have become increasingly popular among organizations for releasing software due to their numerous benefits. Companies across various industries have adopted feature flags to enhance their software development and release processes. Feature flags provide their DevOps teams with the flexibility, control, and agility needed to deliver high-quality software and iterate on features based on user feedback.

FIGURE 1

Software Development and Delivery Organization's Plans for Progressive Delivery

Is your organization using or planning to use progressive delivery?
What types are you currently using or planning to use?
(Percentage of respondents)

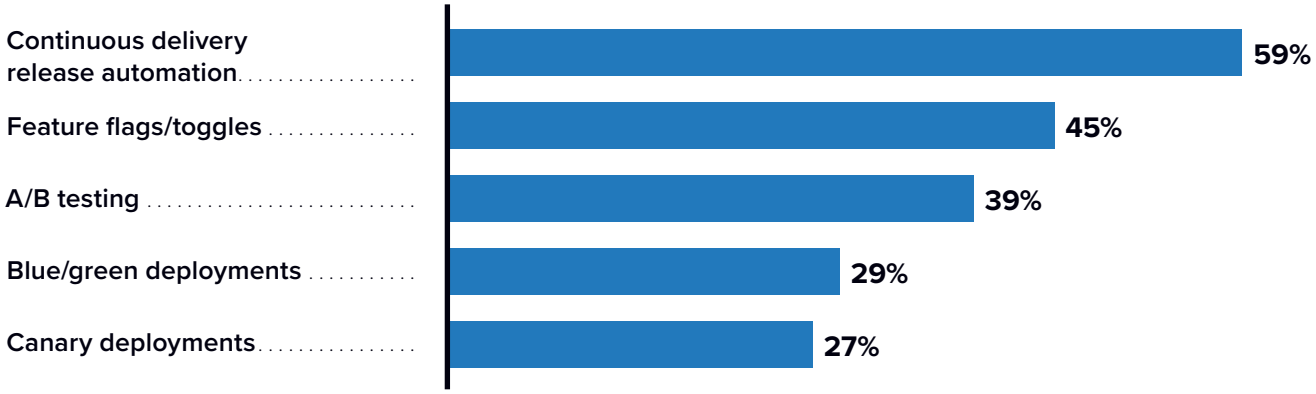
Organizations' Use of Progressive Delivery



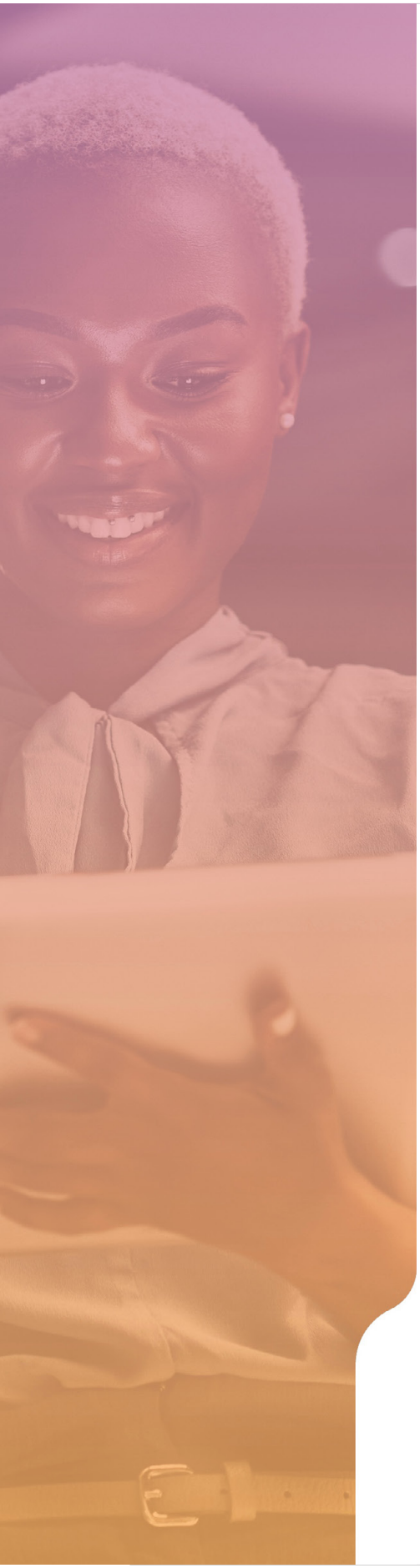
n = 311; Source: IDC *DevOps Survey,* November 2022

For an accessible version of the data in this figure, see Figure 1 Supplemental Data in the Appendix.

Progressive Delivery Methods Currently Used



Note: Respondents indicated organization is Expanding/Currently/Piloting usage of progressive delivery.
n = 206; Source: IDC *DevOps Survey,* November 2022, QE1a



Business Benefits

Beyond the technical aspects commonly associated with feature flags, LaunchDarkly customers also called out several key business benefits:

- More frequent application releases with enriched collaboration among internal teams and stakeholders across the delivery pipeline
- Improved production reliability for enhanced digital experience and customer loyalty
- Enhanced operational resilience with quantifiable improvements in the mean time, between failures and restoration
- Improved confidence across development and operations in application changes due to fast agile rollbacks and experimentation
- Improved developer and site reliability engineer experience due to uneventful application updates that have led to improved employee morale and retention
- Strong compliance and protection of critical customers personally identifiable information, assuring customers that their customers' data is safe and helping achieve SOC 2 compliance
- Measurable improvements in customer satisfaction ratings, net promoter scores, and renewals
- Faster time to market to enhance the ability to be agile enough to react to changing market dynamics and demands
- Operational cost improvements for both releasing and supporting production deployments
- Margin and profitability increases, enabling both fiscal and competitive rigidity

De-Risked Releases and Migrations

Using feature flags can ensure efficient and low-risk software releases, allowing developers to balance speed and safety. Feature flags empower development teams to release, test, and iterate on features with a higher degree of control. This approach enhances software development speed while minimizing the potential risks and ensuring a safer and more reliable software deployment process.

Without feature flags, developers must simultaneously release new features to all users and environments. If a feature negatively impacts the user experience or performance, it can result in a poor user experience for everyone. Since feature flags operate at the feature level, they can be enabled individually, making it easier to isolate failures. Also, when you turn off a problematic feature, you can still make other features in the software release available to your end users.

A key capability of feature flags in mitigating risk without impacting velocity is the separation of feature deployment from feature release. Pushing a release with several features can result in a rollback of the entire release if there is even a single bug. With feature flags, releases can proceed even if one feature is delayed or buggy by turning its flag off.

If a newly released feature introduces unexpected bugs or issues, feature flags provide a quick rollback mechanism, as developers can make changes without redeploying code. Developers can turn off the feature flag, effectively deactivating the feature for all users or reverting to the previous version. This rollback capability enhances the safety net during deployments.

As part of the company's modernization strategy, Vodafone set itself the goal of achieving 100 deployments a day, and the company aspired to reach a point where software development teams could safely and securely deploy as frequently as they wanted. However, Vodafone's software development teams were performing "big bang" releases once per quarter only four years prior. Decoupling software deployments from releases permitted the teams to deploy updates more frequently and gradually release individual features instead of rolling them out all at once.

Feature flags can also de-risk technology migrations, such as moving to a new database or from an on-premises server to the cloud. New technologies can be introduced gradually by toggling feature flags for specific functionalities, allowing for a step-by-step transition rather than a risky "big bang" approach. Feature flags also allow the migrated components to be isolated. This isolation minimizes the impact on the overall system, making it easier to identify issues during the migration process and limit their blast radius. Further, feature flags provide a quick rollback mechanism if issues or performance concerns emerge during a migration.

Paddle needed to rebuild the underlying infrastructure for the payments part of its platform significantly, which prompted it to explore feature management as a way to release in a granular fashion and minimize the impact of this migration on its customers. The implementation of LaunchDarkly began with the payments group for the migration to the new modern infrastructure.

Using LaunchDarkly, the team could direct traffic to a small portion of the customer base and gradually roll it out. The team then methodically increased traffic to the new infrastructure as the team's confidence improved, eventually rolling it out to the entire customer base. The Paddle team emphasized that when rolling out the newly rebuilt infrastructure for the payments part of its platform, LaunchDarkly feature flags "were a really, really important part" of making that successful.

Product Experimentation

For many development teams, it is often a guess what features will drive business value. The organization may spend resources developing features that do not resonate with its customers or do not meet market demands. As a result, resources may be allocated incorrectly, time to market may be delayed, and return on investment may be diminished. Feature flags can help organizations move to an innovative, data-driven software development model through experimentation.

Experimentation is a hypothesis-driven practice that applies the principles of the scientific method to software development and release. Experimentation with feature flags enables companies to test changes and determine if they're impacting the metrics that the organization cares most about by validating strategy with data. By experimenting, development teams can release the features with the highest probability of a positive outcome for the business and customer.

Measuring the business impact of development remains difficult for many organizations. Leadership may initially focus on measuring and defining software delivery performance — in other words, how fast teams are deploying, lead time for changes, mean time to recovery, or change failure rate. However, churning out software faster may not directly help the business. Thus, many organizations struggle with translating software performance metrics into business value KPIs such as customer satisfaction or net-new revenue streams.

Experimentation with feature flags can supply quantifiable insights into the impact of features. Developers can release two or more feature versions to distinct user groups and measure key business metrics such as user engagement, conversion rates, revenue, or any other relevant KPI. By analyzing the differences in outcomes, organizations can statistically assess the impact of features, providing quantitative data to understand how changes influence business goals.

Using metrics and feature flags, LaunchDarkly equipped the team at Paddle to make data-driven decisions about the impact of new features on end users without having advanced statistical knowledge. Natural fluctuations in customer traffic made it challenging to measure the impact of feature releases quantitatively.

With LaunchDarkly Experimentation, user traffic could be shaped, and differences could be observed. A select group of users could be routed to the new feature, with the rest using the existing platform, and the differences occurring between the two user cohorts are measured.

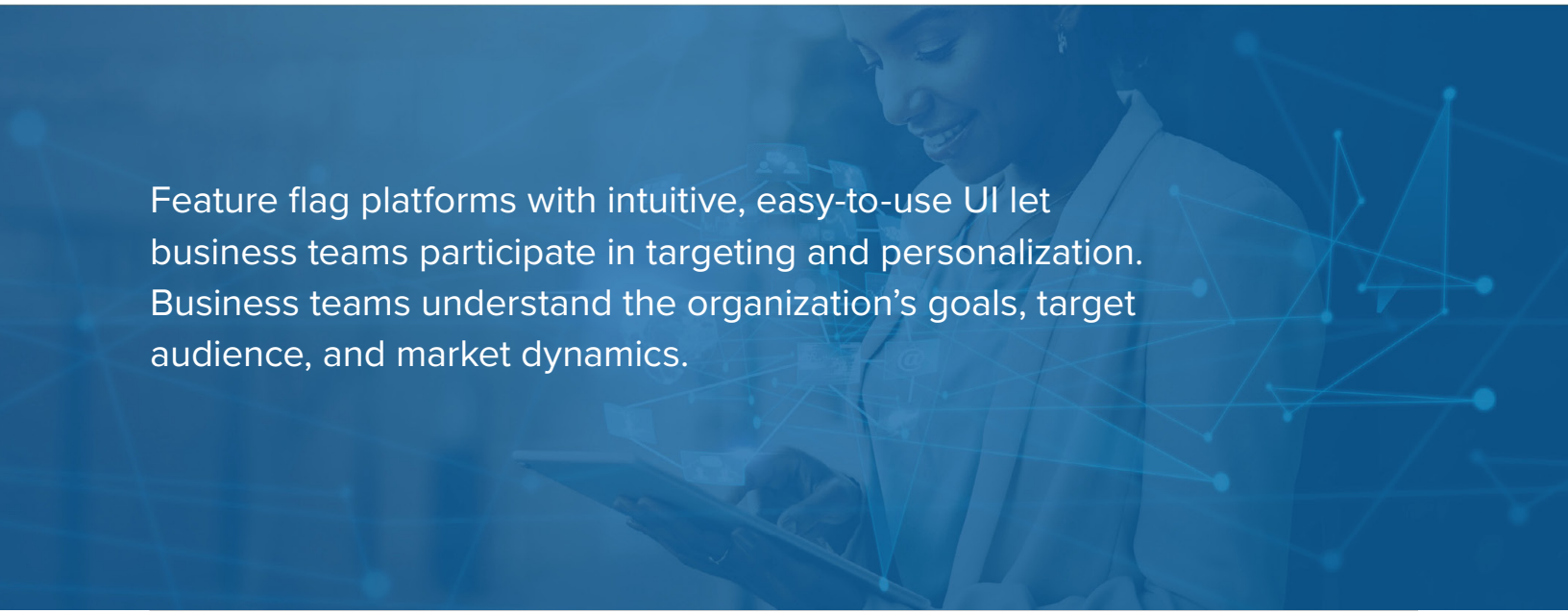
For example, A/B testing allowed Paddle to determine if there was an increase in shopping cart abandonment if a user interface component was changed from a link to a button. For Paddle, even a small improvement in checkout conversion can translate into large financial returns for its customers and vice versa.

The LaunchDarkly Experimentation UI highlights how the test is performing, which informs go-to-market communications and proves the effectiveness of changes with their customer base. Having reliable data provides reassurance that the changes Paddle makes to the platform will have a net-positive benefit for customers.

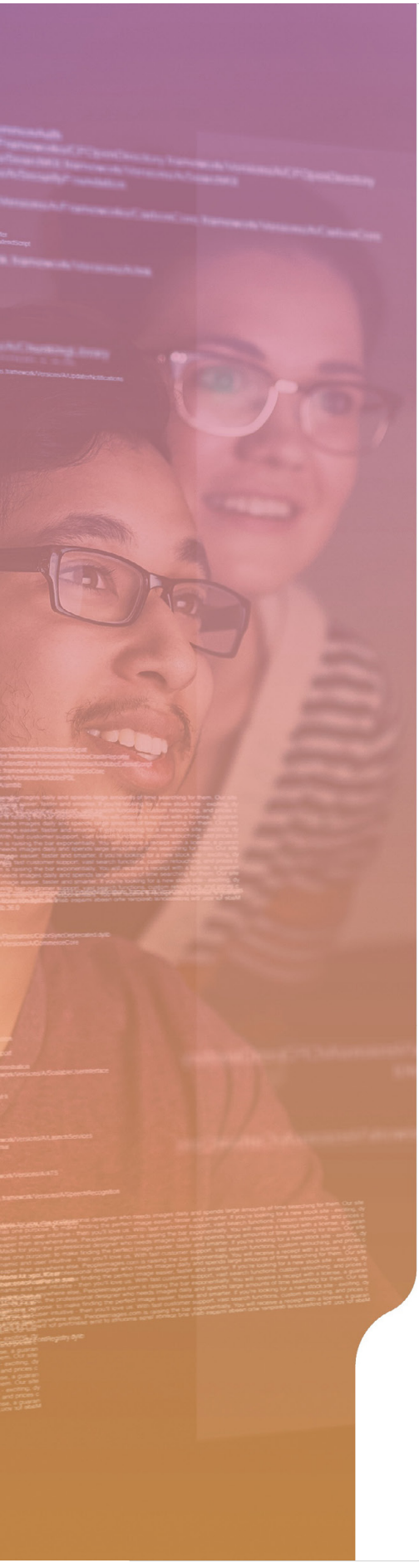
Targeting and Personalizing Experiences

Today, end users are demanding and impatient, expecting a personalized user experience tailored to their needs and usage. User experience expectations are growing daily, and today's exceptional user experience is tomorrow's expected user experience. A customized user experience can enhance user engagement, satisfaction, and overall application or feature success. It can make the difference between making a sale or losing business to a competitor.

Feature flags are key in creating targeted and personalized experiences for users because they allow developers to control the visibility of features based on user attributes, behaviors, or other conditions. This segmentation allows for targeted feature releases to specific user groups.



Feature flag platforms with intuitive, easy-to-use UI let business teams participate in targeting and personalization. Business teams understand the organization's goals, target audience, and market dynamics.



Feature flags also enable progressive delivery. Developers can use feature flags to gradually release features to different user segments. This phased approach helps gather feedback and monitor the impact of features on specific user groups before a broader release, ensuring a personalized and optimized experience.

LaunchDarkly feature flags considerably impacted how the iPipeline application developers built new application features. Since iPipeline could now release features incrementally using a blue-green style methodology, the company improved the frequency of releases from monthly to multiple times per week. The increased release frequency enabled faster customer feedback, allowing DevOps teams to respond more to evolving customer needs.

Feature flag platforms with intuitive, easy-to-use UI let business teams participate in targeting and personalization. Business teams understand the organization's goals, target audience, and market dynamics. They often have insights into user preferences, feedback, and demographic information, allowing them to control feature flag targeting, which ensures personalized experiences align with overarching business objectives and enhance feature releases' relevance and impact.

When business teams are empowered to control targeting and personalization, development teams don't need to handle routine adjustments. As a result, the decision-making process becomes more efficient, and business teams can make timely changes without requiring extensive technical assistance. Allowing business teams to control feature flags also fosters collaboration between technical and non-technical teams. This cross-functional collaboration ensures that technical decisions align with business priorities, promoting a more cohesive and aligned approach to feature management.

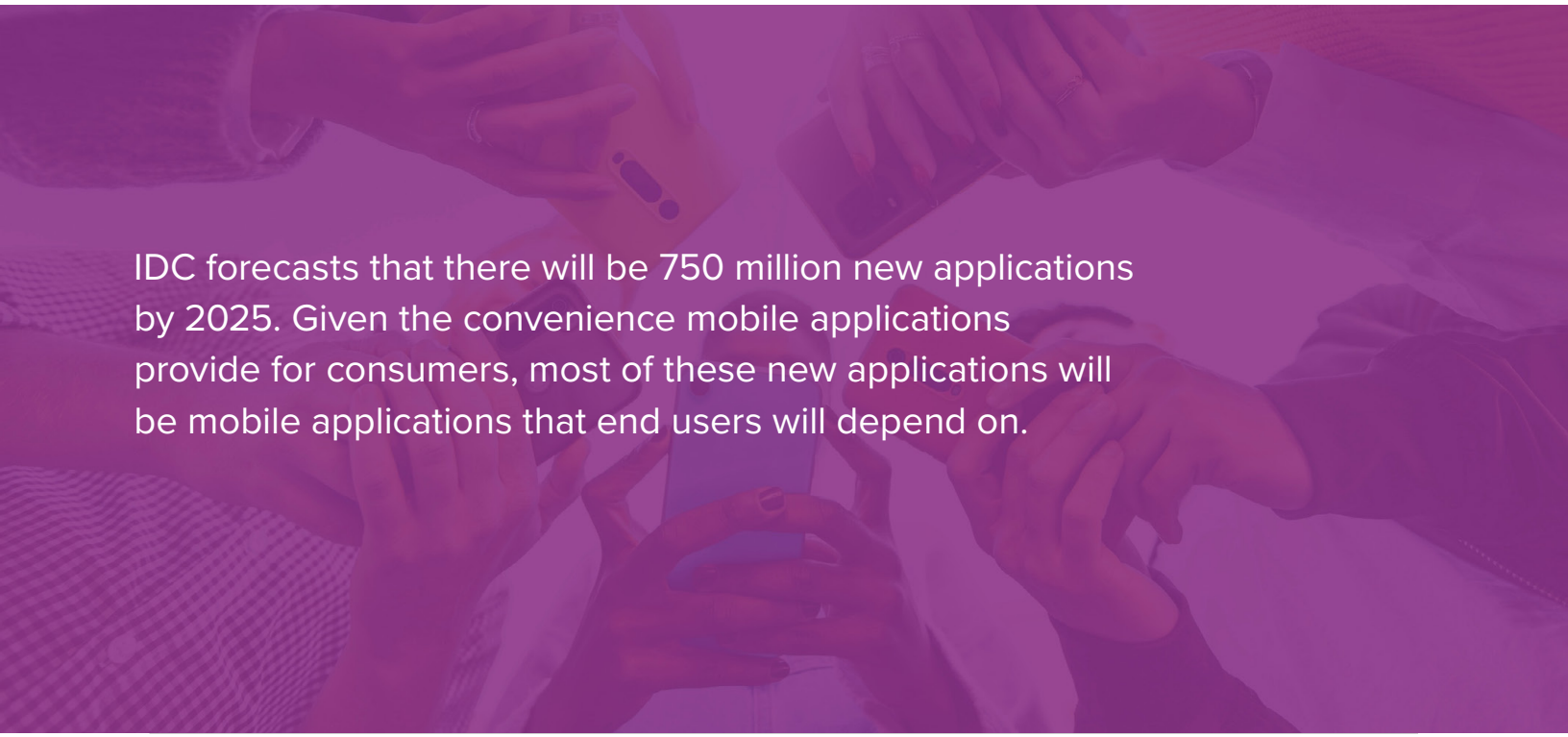
At Paddle, enabling product managers to participate in the release processes empowers them to better control releases, coordinate launch timings, and create feedback loops informing product direction. With LaunchDarkly, Paddle product managers can see hour-by-hour differences in the behavior of the software and the end customers, providing high levels of confidence before changes are broadly released to all the platform users.

Mobile Releases

In the digital business era, mobile applications have become crucial to an organization's strategy. For users and customers, convenience is king — they want everything to be more accessible, faster, and personalized. For many companies, mobile applications are the new storefront and are the primary channel for customer engagement.

IDC forecasts that there will be 750 million new applications by 2025. Given the convenience mobile applications provide for consumers, most of these new applications will be mobile applications that end users will depend on.

However, the release process for mobile applications has unique challenges that feature flags can address. For example, app stores ensure a safe and consistent user experience, but they also restrict the amount of control organizations have over their apps due to review processes. Feature flags allow developers to decouple the process of deploying a feature to the app store from releasing that same feature to end users. By wrapping code with feature flags, developers can toggle off features that aren't ready for end users. Users will not see any toggled "off" features once the app is approved by the app store and made available to download, allowing for a more strategic release of new features.



Customers also have little tolerance for slow or buggy mobile app experiences. Following app store approval processes without using feature flags can mean an app release with a bug that could impact customers for hours or days. Users could be lost during that time, and negative reviews could damage the app's reputation. Buggy features can be killed instantly by feature flags, preventing them from being used until the development team can identify what went wrong.

Mobile app testing also has unique challenges because of the fragmented mobile ecosystem of devices, platforms, and operating system versions. Feature flags allow developers to target features by device type, operating system, application version, or other parameters to ensure the right user gets the right experience.

LaunchDarkly has been key to ensuring Vodafone customers get the best possible digital experience. Vodafone development teams can resolve incidents instantaneously when a new feature impacts a production system. Since new features can be released gradually, isolating the problem is straightforward. Using the LaunchDarkly platform, developers can revert problematic changes with minimal customer impact. With the problem isolated, developers can quickly and confidently address the underlying software bug before re-releasing the fixed code into production.

Challenges/Opportunities

CHALLENGE:

Temptation to create a homegrown feature flag solution

One of the common pitfalls of feature flags is that, on the surface, organizations think it is something they can build on their own. However, this effort quickly becomes overblown with too much time spent building and maintaining the homegrown solution. Compounding this challenge are frustrated developers and an expensive opportunity cost when they are pulled away from creating new features to deal with the toil of working with a homegrown feature flag solution.

OPPORTUNITY:

Seek an enterprise-grade solution

It behooves organizations to avoid the temptation to build homegrown feature flag capabilities and seek out a robust feature flag solution that can enable them to move faster while de-risking application releases, allowing experimentation, and personalizing application experiences across platforms.

CHALLENGE:

Technical debt

Because feature flags are embedded in the application code, once the feature has been released and verified, the feature flag logic must eventually be removed from the code. While the actual coding involved with cleaning up feature flags is straightforward, it adds technical debt for developers to go back and clean up the code. Also, if not tracked properly, the feature flag Boolean logic could accumulate over time, creating needless performance overhead.

OPPORTUNITY:

Flag management

Seek out a feature flag solution that provides built-in capabilities to help enterprise organizations identify stale or inactive flags and integrates easily with tools across the development and DevOps pipeline to ensure teams can eradicate obsolete feature flags. Schedule regular review cycles on a monthly or quarterly basis to identify feature flags to archive. LaunchDarkly provides the options to deprecate, archive, and delete feature flags.

CHALLENGE:

Cultural resistance to feature flags

Feature flags can fundamentally change your company culture and how your team operates, releases, and deploys application changes into production. Every culture has certain norms and can be resistant to change. This behavior means any attempt to optimize how things are done will almost certainly be met with fear and opposition.

OPPORTUNITY:

Demonstrate value early

Build early and incremental value using feature flags by starting small. Do not attempt a “big bang” approach to adopting feature flags and overcommit to first experiences. Carefully select a well-suited project or application to use as part of an initial pilot project and use this to build a culture of feature flag experimentation, prioritizing customer experiences and shifting from delivering application releases to delivering features.

Conclusion

In the era of digital economies, the speed of products and services entering the market will grow at an increasing rate. "The fast eat the slow" is common — innovators grasping developing market trends, creating products and services that meet market needs, capitalizing on their timing to iterate and improve and, ultimately, gaining a competitive advantage. Organizations rely on software to drive internal efficiencies, grow revenue, and achieve other strategic business outcomes.

Therefore, it behooves organizations to improve the speed at which they can deliver new application features to the market. These organizations must rethink how they release their software since traditional methods will not provide the required capabilities to compete in the new digital paradigm. Forward-looking organizations must examine how feature-level management via feature flags can give them the business agility needed to compete.

As part of evaluating feature flag solutions, these organizations should evaluate the LaunchDarkly Feature Management platform and the enterprise scalability and business benefits it could provide.

Appendix: Supplemental Data

This appendix provides an accessible version of the data for the complex figure in this document. Click "Return to original figure" below the table to get back to the original data figure.

FIGURE 1 SUPPLEMENTAL DATA

Software Development and Delivery Organization's Plans for Progressive Delivery

Organizations' Use of Progressive Delivery

	Organizations' Use of Progressive Delivery
Expanding use	22%
Currently using	33%
Piloting usage	11%
Planning to use in the next 12 months	12%
Interested, but no near-term plans to use	7%
Not using	14%
Don't know	0%

n = 311; Source: IDC DevOps Survey, November 2022

Return to the original figure

About the IDC Analysts

Jim Mercer
Research Vice President, DevOps & DevSecOps, IDC

Jim is a research vice president within IDC's DevOps and DevSecOps Solutions research practices. Jim's core research includes topics such as rapid enterprise application development, modern microservice-based packaging, GitOps, application security, software supply chain security, and automated deployment and life-cycle/management strategies as applied to a DevOps practice. In addition, he examines how the move to DevOps methodologies impacts enterprise use of open source and preferences for using on-premises computing and development platforms versus public cloud services. He looks at how organizations prioritize DevSecOps and use automation to insert security assessments into the DevOps delivery pipeline (i.e., shift left).

More about Jim Mercer

Katie Norton
Senior Research Analyst, DevOps & DevSecOps, IDC

Katie is a senior research analyst within IDC's DevOps and DevSecOps research practices. With her background in research administration and data analytics, Katie takes a data-first approach in her market analysis. Katie covers topics such as rapid enterprise application development, integrating security into the software development life cycle, and automated deployment and life-cycle/management strategies. In the area of DevSecOps, she explores how DevOps teams take ownership of security and examines the drivers of DevSecOps adoption. Katie has particular interest in adjacent topics such as mainframe DevOps, software supply chain security, feature flags/progressive delivery, and GitOps.

More about Katie Norton

IDC Custom Solutions

IDC Custom Solutions produced this publication. The opinion, analysis, and research results presented herein are drawn from more detailed research and analysis that IDC independently conducted and published, unless specific vendor sponsorship is noted. IDC Custom Solutions makes IDC content available in a wide range of formats for distribution by various companies. This IDC material is licensed for external use and in no way does the use or publication of IDC research indicate IDC's endorsement of the sponsor's or licensee's products or strategies.

IDC Research, Inc.
140 Kendrick Street, Building B, Needham, MA 02494, USA
T +1 508 872 8200

idc.com @idc @idc

International Data Corporation (IDC) is the premier global provider of market intelligence, advisory services, and events for the information technology, telecommunications, and consumer technology markets. With more than 1,300 analysts worldwide, IDC offers global, regional, and local expertise on technology and industry opportunities and trends in over 110 countries. IDC's analysis and insight helps IT professionals, business executives, and the investment community to make fact-based technology decisions and to achieve their key business objectives.