

FORRESTER®

# The Total Economic Impact™ Of LaunchDarkly

Cost Savings And Business Benefits  
Enabled By LaunchDarkly

JANUARY 2021

## Table Of Contents

Consulting Team: Amy Harrison

<b>Executive Summary</b> .....	<b>1</b>
<b>The LaunchDarkly Customer Journey</b> .....	<b>6</b>
Key Challenges.....	6
Composite Organization.....	7
<b>Analysis Of Benefits</b> .....	<b>8</b>
Reduced Cost Of Preproduction Environment.....	8
Developer Productivity Savings .....	9
Avoided Costs Of Maintaining Homegrown Feature Management System.....	11
Reduction In Developer Recruitment Costs .....	12
Unquantified Benefits.....	15
Flexibility .....	15
<b>Analysis Of Costs</b> .....	<b>16</b>
Software License Fees.....	16
Deployment, Training, And Management Costs..	17
<b>Financial Summary</b> .....	<b>18</b>
<b>Appendix A: Total Economic Impact</b> .....	<b>19</b>
<b>Appendix B: Endnotes</b> .....	<b>20</b>



### ABOUT FORRESTER CONSULTING

Forrester Consulting provides independent and objective research-based consulting to help leaders succeed in their organizations. For more information, visit [forrester.com/consulting](https://forrester.com/consulting).

© 2021, Forrester Research, Inc. All rights reserved. Unauthorized reproduction is strictly prohibited. Information is based on the best available resources. Opinions reflect judgment at the time and are subject to change. Forrester®, Technographics®, Forrester Wave, RoleView, TechRadar, and Total Economic Impact are trademarks of Forrester Research, Inc. All other trademarks are the property of their respective companies.

## Executive Summary

Executive expectations require businesses to be customer-led, insights-driven, experience-focused, fast, and connected. Development organizations must also focus on agile and dynamic methods of operations. Modern technology operations practices help teams achieve faster reaction times to incidents, reduce risks in the IT enterprise, eliminate inefficiencies of both people and technologies, and, ultimately, deliver flexible, speedy, and high-quality services.<sup>1</sup>

As the demand for speed and flexibility grows at every layer of an organization, enterprises have turned to DevOps methods, driving a fundamental shift in how to approach core capabilities, organizational structures, processes, and tooling.<sup>2</sup> [LaunchDarkly](#) plays a crucial role in the DevOps process by helping teams execute on progressive release strategies. It allows development teams to control every feature from a centralized dashboard, and it can implement rule-driven automation to shut down errant code before problems multiply into a support incident. This significantly reduces release risk and stress.

LaunchDarkly commissioned Forrester Consulting to conduct a Total Economic Impact™ (TEI) study and examine the potential return on investment (ROI) enterprises may realize by deploying [LaunchDarkly](#).<sup>3</sup> The purpose of this study is to provide readers with a framework to evaluate the potential financial impact of investing in LaunchDarkly for their organizations.

To better understand the benefits, costs, and risks associated with this investment, Forrester interviewed representatives of four organizations with experience using LaunchDarkly. For the purposes of this study, Forrester aggregated the experiences of the interviewed customers and combined the results into a single composite organization.

Prior to using LaunchDarkly, developers tried leveraging code branching techniques, which can easily spiral out of control. In addition, branches

### KEY STATISTICS



Return on investment (ROI)  
**245%**



Net present value (NPV)  
**\$3.80M**

aren't good for feature work because they can insulate teams from important architectural changes like updating frameworks and common libraries.<sup>4</sup>

Organizations also tried building their own homegrown feature flag solutions. However, the cost to build, maintain, and update those homegrown systems led decision-makers to search for a better solution.

After the investment in LaunchDarkly, DevOps teams were able to use feature flags to merge and test code earlier in the development lifecycle. Using feature flags allows the tests to be run in a production environment with a small number of users. When the feature is ready, it can be released in a progressive manner to small groups of users at a time. If issues arise, the faulty component can be switched off, which improves potential customer experience issues. This significantly reduces the pressure on development teams because there are no longer big,

anxiety-producing launch events after working on code for months.

**“Developers now spend less time solving integration issues. There’s a lot less unplanned time when it gets to the end of the sprint. It’s more relaxed because they’ve been merging code frequently throughout the entire sprint. The code written at the start of the sprint is around the larger design, and towards the end, it is smaller and smaller. The code reviews and corresponding risks get lower as you approach the end of the sprint. Whereas when you’re not using feature toggling, the risks are higher at the end of the sprint. With LaunchDarkly, there’s a lot less unplanned work for big merges at the end and fewer issues in production.”**

*Chief architect, manufacturing*

## KEY FINDINGS

**Quantified benefits.** Three-year, risk-adjusted present value (PV) quantified benefits include:

- **Reduced cost of pre-production environment, saving \$3.37 million.** Software testing processes often require running user testing in a staging environment that closely replicates production, and this is very expensive to build and maintain. Using LaunchDarkly, the need for the staging environment is drastically reduced because developers can turn on code in the actual production environment to a very small number of users and resolve issues as they arise.

- **Developer productivity savings totaling \$1 million.** The composite organization is able to achieve developer productivity savings on the day of release itself and in the hours saved resolving issues after release. Using LaunchDarkly, when new features are rolled out, they have already been tested in production, which surfaces fewer code errors. When issues do arise, developers are able to isolate and shut off the code causing the error, flipping the end user back, and causing no interruptions. Developers spend less idle time waiting for issues to surface at the time of release, and they benefit from reduced unplanned work in the following days.
- **Avoided costs of maintaining homegrown feature management system.** Interviewees described attempting to build their own homegrown feature management systems. After building them, they realized the systems would require continued resource to keep them up to date or to add new features. Interviewees said they chose LaunchDarkly as a best-of-breed solution because it is highly reliable, it integrates easily with Okta, and it complies with major security standards such as ISO 27001.
- **Reduction in developer recruitment costs.** Software developers are in high demand, and they are notoriously difficult to retain. A LinkedIn study found turnover rates of 21.7% for embedded software engineers.<sup>5</sup> The cost of recruiting, hiring, and training new engineers can cost 25% or more of a developer’s annual salary. LaunchDarkly contributes directly to the reduction of developer stress by decreasing the risk of massive errors being found at the point of merging code.

**Unquantified benefits.** Benefits that are not quantified for this study include:

- **Improved alignment with the business.** Utilizing feature flags allows for precise control of

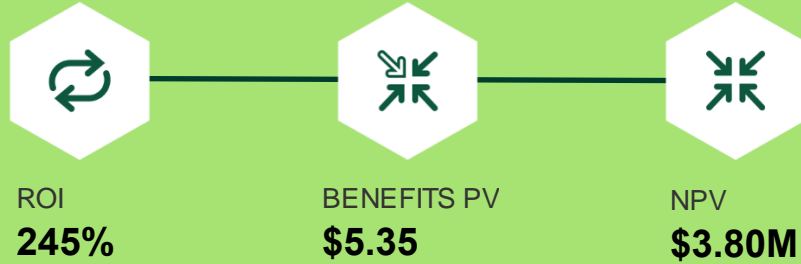
the deployment of features and choosing which users can access those features. Development teams and the business work together to deliver a better customer experience by engaging customers as early as possible in the development lifecycle. It also allows the business to deliver those features at the optimal time.

- **Improved customer experience.** LaunchDarkly provides the opportunity to test customer-facing applications in the client's own location without the added expense and intrusion of setting up a concurrent production environment.
- **Improved compliance with external governance.** An interviewee described how their organization used LaunchDarkly to enforce compliance standards. Because LaunchDarkly is SOC Level 2 compliant, it has GDPR privacy shield certification, and it is ISO 27001:2013 certified, the organization was able to change its risk category. It can now release during the day instead of overnight — much to the joy of the organization's developers.

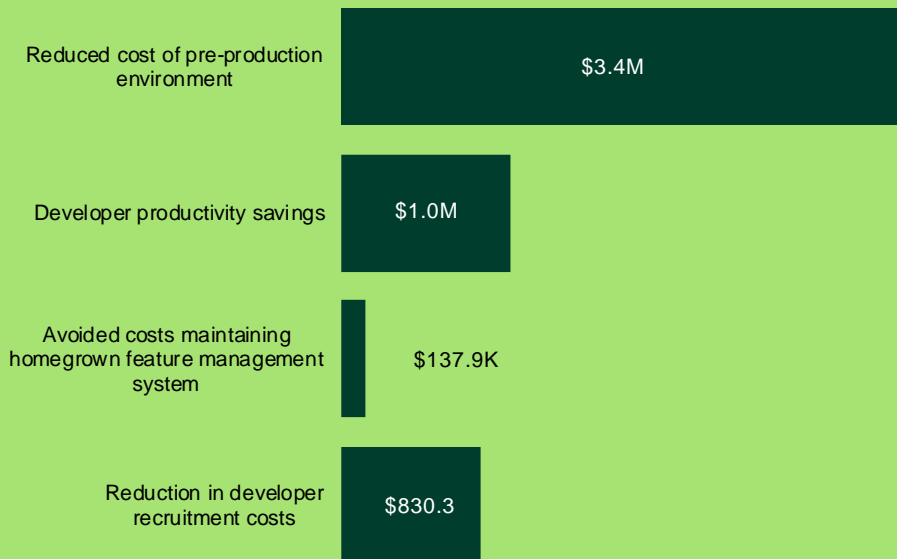
**Costs.** Risk-adjusted PV costs include:

- **Software license fees.** Customers purchase licenses from LaunchDarkly for an annual fee. The number of licenses purchased is based on the number of developers who use the product.
- **Deployment, training, and management costs.** Onboarding users to LaunchDarkly requires provisioning through Okta. According to interviewees, this takes 700 hours of time from an IT professional. For each user, this internal training process takes two hours. After onboarding, maintenance of LaunchDarkly requires 50% of the time of one FTE.

The customer interviews and financial analysis found that a composite organization experiences benefits of \$5.35 million over three years versus costs of \$1.55 million, adding up to a net present value (NPV) of \$3.80 million and an ROI of 245%.



### Benefits (Three-Year)



“ LaunchDarkly has become part of our development process and culture. Now that it’s here, it’s hard to think about how it would be without it.”

— Senior development manager, high tech

## TEI FRAMEWORK AND METHODOLOGY

From the information provided in the interviews, Forrester constructed a Total Economic Impact™ framework for those organizations considering an investment in LaunchDarkly.

The objective of the framework is to identify the cost, benefit, flexibility, and risk factors that affect the investment decision. Forrester took a multistep approach to evaluate the impact that LaunchDarkly can have on an organization.

### DISCLOSURES

Readers should be aware of the following:

This study is commissioned by LaunchDarkly and delivered by Forrester Consulting. It is not meant to be used as a competitive analysis.

Forrester makes no assumptions as to the potential ROI that other organizations will receive. Forrester strongly advises that readers use their own estimates within the framework provided in the report to determine the appropriateness of an investment in LaunchDarkly.

LaunchDarkly reviewed and provided feedback to Forrester, but Forrester maintains editorial control over the study and its findings and does not accept changes to the study that contradict Forrester's findings or obscure the meaning of the study.

LaunchDarkly provided the customer names for the interviews but did not participate in the interviews.



### DUE DILIGENCE

Interviewed LaunchDarkly stakeholders and Forrester analysts to gather data relative to their use of feature flags.



### CUSTOMER INTERVIEWS

Interviewed five decision-makers at four organizations using LaunchDarkly to obtain data with respect to costs, benefits, and risks.



### COMPOSITE ORGANIZATION

Designed a composite organization based on characteristics of the interviewed organizations.



### FINANCIAL MODEL FRAMEWORK

Constructed a financial model representative of the interviews using the TEI methodology and risk-adjusted the financial model based on issues and concerns of the interviewed organizations.



### CASE STUDY

Employed four fundamental elements of TEI in modeling the investment impact: benefits, costs, flexibility, and risks. Given the increasing sophistication of ROI analyses related to IT investments, Forrester's TEI methodology provides a complete picture of the total economic impact of purchase decisions. Please see Appendix A for additional information on the TEI methodology.

# The LaunchDarkly Customer Journey

■ Drivers leading to the LaunchDarkly investment

Interviewed Organizations			
Industry	Region	Interviewee	Number of LaunchDarkly users
Manufacturer	North America (US)	Chief architect DevOps leader	200
Banking	Asia Pacific (Australia)	Platform engineer, mobile internet banking	100
High tech	North America (US)	Senior development manager	75
Software	Asia Pacific (New Zealand)	Lead portfolio architect	850

## KEY CHALLENGES

As organizations adopt a DevOps approach to application development and they move towards a continuous integration and continuous delivery, previous development processes quickly become untenable. Whether trying to deliver a single, stressful, massive release of code or managing a forest of code branches, previous processes become onerous to support.

The interviewees' organizations struggled with similar challenges, including:

- **Rudimentary techniques like branching proved unwieldy.** Creating a branch process as a way of keeping code separate can quickly spiral out of control. Not only does a team need to create the branch, but updating and integrating to the main branch and then deleting it after completion adds time and complexity. A DevOps leader for a manufacturer said: "Teams were working to keep functionality disabled by using strategies like long-lived branches. From a business perspective, that meant a slower integration of the features and risk at the point where we have the light integration into master branches."
- **End-user testing was risky.** Developers and their business partners worried about the process

of gathering end-user feedback for fear that the release of code would trigger errors. The DevOps leader described how this delayed the overall development process. She said: "This didn't give the business flexibility to get quick end-user validation. Because they were not able to easily demonstrate features and gather customer feedback while still in development, release delays ensue."

**"We had all these really large features we wanted to implement and deliver to customers. We weren't sure if the traditional way of deploying those features was going to work. The traditional way entails going off to a secret room, coding up this new feature for months and months and months, and then one day just kind of pushing it out into production and keeping our fingers crossed that everything works."**

*Senior development manager, high tech*



- **Homegrown systems require ongoing investment.** Whether a team builds a simple branching system or a full-blown feature management system, there is cost involved. Beyond the initial development costs, teams must maintain what they have built. The DevOps leader said: “Every time you build something, there is always an element of sustenance, engineering, and ensuring it continues to operate correctly.” As organizations want the ability for more functionality like environment toggling or user-based switching, the time vs. cost tradeoffs grow. A chief architect with a manufacturer said, “When I went to go to user-based switching, everything started getting out of control in terms of effort to maintain.”
- **Each release caused the development teams significant levels of stress.** Interviewees universally described the pain of pushing the button to release a new feature, then waiting for things to break. A senior development manager for a high-tech organization said, “It’s hard to quantify stress relief in dollars, but I could

**“Developers could work on releases for three to four months. If one release causes a problem and it’s market-sensitive, it can hurt your brand. We didn’t have the ability to disable a feature quickly if something went wrong. There was a lot of pressure and nervousness with each release, but it’s much calmer with LaunchDarkly. If there are no side effects, great. If there are, we can just turn it off, and it’s safe. It’s all much less stressful.”**

*Platform engineer, financial services*

quantify that in stress points. In the past, there used to be lots and lots of anxiety.”

### COMPOSITE ORGANIZATION

Based on the interviews, Forrester constructed a TEI framework, a composite company, and an ROI analysis that illustrates the areas financially affected. The composite organization is representative of the interviewees’ companies, and it is used to present the aggregate financial analysis in the next section. The composite organization has the following characteristics:

**Description of composite.** The organization is a multichannel bank that provides both B2B and B2C services. The bank has more than 15 million customers, and it is based in the US.

#### Key assumptions

- **Multichannel bank**
- **15M customers**
- **B2B and B2C offerings**
- **Growing steadily over three years**

**Deployment characteristics.** The bank has been growing steadily through organic customer growth and acquisition. In the first year of using LaunchDarkly, the mobile app and web development teams purchase licenses. After seeing the success of those customer facing teams, back-end systems teams start using LaunchDarkly feature flags. The composite organization has 100 licenses in Year 1, 500 in Year 2, and 1,000 in Year 3.

# Analysis Of Benefits

■ Quantified benefit data as applied to the composite

Total Benefits						
Ref.	Benefit	Year 1	Year 2	Year 3	Total	Present Value
Atr	Reduced cost of preproduction environment	\$678,600	\$1,282,050	\$2,263,500	\$4,224,150	\$3,377,056
Btr	Developer productivity savings	\$63,360	\$216,000	\$1,025,856	\$1,305,216	\$1,006,853
Ctr	Avoided costs of maintaining homegrown feature management system	\$55,440	\$55,440	\$55,440	\$166,320	\$137,871
Dtr	Reduction in developer recruitment costs	\$66,132	\$330,660	\$661,320	\$1,058,112	\$830,252
	Total benefits (risk-adjusted)	\$863,532	\$1,884,150	\$4,006,116	\$6,753,798	\$5,352,032

## REDUCED COST OF PREPRODUCTION ENVIRONMENT

**Evidence and data.** One interviewee described how their organization’s development process previously required running user testing in a staging environment, closely replicating production. As the code grew, so did the costs of the infrastructure, management, and support needed to run it.

Implementing LaunchDarkly drastically reduced the need for a staging environment. It enabled developers to use feature flags to turn on code in the actual production environment to a very small number of users and to resolve issues as they arise. As issues were resolved, developers could slowly

grow the population of beta testers using new features in production over time.

The senior development manager said: “Today, the code usually lives in our staging environment for maybe an hour to run a set of automated tests before it goes to production. We no longer have [to pay for] our own infrastructure. We just go to the production line. It allows us to have this progressive delivery model. Instead of having a dozen engineers testing code, we can have hundreds if not thousands of our own users testing it for us.”

**Modeling and assumptions.** The financial model assumes:

- The composite organization previously needed to spend \$35,000 per month running its hosted infrastructure costs for its staging and preproduction environments.
- Two IT system administrators were required to manage and maintain the preproduction infrastructure.
- The fully burdened annual salary of an IT administrator is \$167,000.

**“It’s not just cost from these 50 [preproduction] machines, but all the costs involved of engineers having to spend their time to debug issues or to monitor and maintain machines.”**

*Senior development manager, high tech*

**Risks.** The results for the reduced costs of the preproduction environment may vary based upon the following risks:

- The size of the organization and the amount of infrastructure required to run the preproduction environment.
- The number of features being tested and released.

- The cost of an IT system administrator.

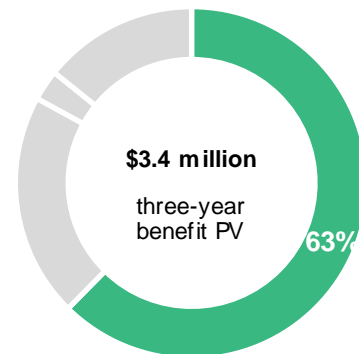
To account for these risks, Forrester adjusted this benefit downward by 10%, yielding a three-year, risk-adjusted total PV (discounted at 10%) of \$3.4 million.

Reduced Cost Of Preproduction Environment					
Ref.	Metric	Calculation	Year 1	Year 2	Year 3
A1	Reduced preproduction monthly hosted infrastructure costs	Interviews	\$35,000	\$70,000	\$140,000
A2	Number of administrators running preproduction environment	Interviews	2.0	3.5	5.0
A3	Annual fully loaded salary	Salary.com	\$167,000	\$167,000	\$167,000
At	Reduced cost of preproduction environment	$(A2 \cdot A3) + A1 \cdot 12$	\$754,000	\$1,424,500	\$2,515,000
	Risk adjustment	↓10%			
Atr	Reduced cost of preproduction environment (risk-adjusted)		\$678,600	\$1,282,050	\$2,263,500
<b>Three-year total: \$4,224,150</b>			<b>Three-year present value: \$3,377,056</b>		

### DEVELOPER PRODUCTIVITY SAVINGS

**Evidence and data.** The composite organization is able to achieve developer productivity savings on the day of release and in the hours saved resolving issues after release. The benefits include:

- **Fewer developers spend unproductive time waiting for problems to arise.** The senior development manager described said: “We get to the point where Tuesdays and Thursdays, everyone will get together in a call and basically wait for us to push the button. We sit around checking logs and the website, and we inevitably wait for things to break. We go back to make fixes, redeploy again, or make a monkey patch to the existing production environment to make these changes quickly.” Using LaunchDarkly, when new features are rolled out, they have already been tested in production. The same



interviewee said: “It’s no longer this big anxiety event; it’s literally just that this feature we’ve worked on for six months has been released. It’s business as usual.”

- **Fewer cooks in the kitchen make smaller messes.** When a release gets delayed because of errors in code, more people need to get involved to solve the issues. A platform engineer in the banking industry said that her

organization’s mobile application team of 10 saved three to four days with each release. She said: “When the release drags on, a lot of people get involved. For example, let’s say there are three scrum teams. You need to involve the

**“We are supposed to launch every Tuesday and Thursday, but a lot of times we don’t get to do the Thursday launch because we’re still busy fixing all of the problems that arose on Tuesday.”**

*Platform engineer, financial services*

release team, the team that caused the problem, and experts to fix it. Using LaunchDarkly, if you can’t fix it, turn it off, and figure it out business as usual, you stop the snowball of people.”

**Modeling and assumptions.** The composite organization is on a DevOps transformation journey and it is moving to more continuous releases of code. In Year 1, it launches larger features less frequently. In Year 2, it launches monthly. And in Year 3, it launched weekly.

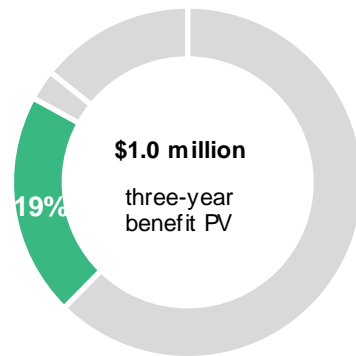
- Each release involves several teams totaling 30 developers.
- Those 30 developers previously spent all of their time waiting for problems to arise and fixing them.
- After implementing LaunchDarkly, the teams are able to reduce the amount of time fixing problems by 50% in Year 1, and that increases to 95% in Year 3.

- 50% of the productivity improvement is recaptured and put toward other value-added tasks.
- Because LaunchDarkly allows for testing and releases to smaller numbers of people, the development team is able to save 32 hours of time due to a reduction in unplanned work. Therefore, there were many fewer problems on smaller pieces of code to resolve.

**Risks.** The savings from the reduction in unplanned work could vary with:

- The amount of code released and the number of issues that arise.
- The variances in salaries for members of the development team in different regions.
- The complexity of the problems that arise.

To account for these risks, Forrester adjusted this benefit downward by 10%, yielding a three-year, risk-adjusted total PV of \$1.0 million.



Developer Productivity Savings					
Ref.	Metric	Calculation	Year 1	Year 2	Year 3
B1	Number of releases per year	Interviews	4	12	52
B2	Number of developers involved	Interviews	30	30	30
B3	Hours waiting for and solving problems day of release	Interviews	8	8	8
B4	Percentage of time saved solving problems	Interviews	50%	75%	95%
B5	Fully burdened hourly salary (developer)	PayScale	\$80	\$80	\$80
B6	Productivity recapture	Assumption	50%	50%	50%
B7	<b>Developer productivity savings day of release</b>	<b>B1*B2*B3*B4*B5*B6</b>	<b>\$19,200</b>	<b>\$86,400</b>	<b>\$474,240</b>
B8	Hours saved per release in unplanned work (solving problems)	32 hours	32	32	32
B9	Team of developers performing unplanned work	Interviews	10	10	10
B10	<b>Savings due to reduction in unplanned work</b>	<b>B1*B8*B9*B5*B6</b>	<b>\$51,200</b>	<b>\$153,600</b>	<b>\$665,600</b>
Bt	Developer productivity savings	B7+B10	\$70,400	\$240,000	\$1,139,840
	Risk adjustment	↓10%			
Btr	Developer productivity savings (risk-adjusted)		\$63,360	\$216,000	\$1,025,856
<b>Three-year total: \$1,305,216</b>			<b>Three-year present value: \$1,006,853</b>		

**AVOIDED COSTS OF MAINTAINING HOMEGROWN FEATURE MANAGEMENT SYSTEM**

**Evidence and data.** Some interviewees said their organization attempted to build its own homegrown feature management system. Others spoke about maintaining libraries of branches. Another interviewee described how four software engineers spent 50% of their time for an entire year building a homegrown system, but the organization only assigned 25% of one FTE’s time to maintain it. A lead portfolio architect for a software organization said: “We only actually had about maybe 25% of one engineer for ongoing maintenance. What that meant was that we didn’t add new features to it, and that led

to some problems.” The interviewee said this was a key factor in exploring a best-of-breed solution.

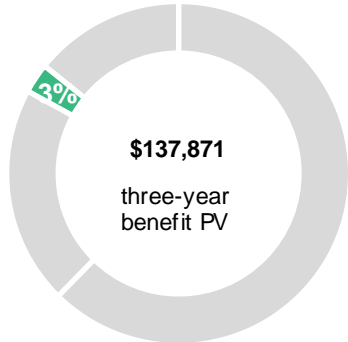
In addition to saving on maintenance, the interviewees’ organizations now have a system that is ISO 27001-certified. The system also:

- Is highly reliable with excellent performance.
- Integrates easily with Okta single sign-on.

Lack of maintenance of homegrown system led to future problems that resulted in system downtime.

**Modeling and assumptions.** The financial model assumes:

- Because the cost of building the system is a sunk cost, the model looks at the ongoing costs beginning with 25% of the fully burdened salary of a fully burdened engineer to maintain a homegrown system.



- Ongoing hosting fees for the homegrown feature management system are \$20,000 per year.

**Risks.** Potential risks that could impact this benefit include:

- The number of engineers assigned to support a homegrown system.
- The variances in salaries for members of the development team in different regions.
- The size of the system and the costs for hosting it.

To account for these risks, Forrester adjusted this benefit downward by 10%, yielding a three-year, risk-adjusted total PV of \$137,871.

### Avoided Costs Of Maintaining Homegrown Feature Management System

Ref.	Metric	Calculation	Year 1	Year 2	Year 3
C1	Ongoing engineer support costs to maintain feature managementsystem	25% of FTE	\$41,600	\$41,600	\$41,600
C2	Hosting fees	Interviews	\$20,000	\$20,000	\$20,000
Ct	Avoided costs of maintaining homegrown feature managementsystem	C1+C2	\$61,600	\$61,600	\$61,600
	Risk adjustment	↓10%			
Ctr	Avoided costs of maintaining feature management system (risk-adjusted)		\$55,440	\$55,440	\$55,440
<b>Three-year total: \$166,320</b>			<b>Three-year present value: \$137,871</b>		

### REDUCTION IN DEVELOPER RECRUITMENT COSTS

**Evidence and data.** Each of the interviewees referred to the stress developers felt with each major software release. When asked about the value of LaunchDarkly to the organization, stress reduction was near the top — if not at the top — of the list.

- The DevOps leader with a manufacturer said: “By reducing some of that risk, it’s helping teams smooth out the process and reduce stress. Using

**“I can tell you that I left my last organization especially because of stress, but [decision-makers] never recognize it as a problem. It’s almost like Stockholm syndrome; they’re so used to it that they think it’s just the way things are.”**

*Senior development manager, high tech*

techniques like feature toggling reduces the risk of delayed releases. Teams are more comfortable to make much more confident commitments, which has an impact on the overall business.”

- The chief architect with a manufacturer said, “The issue of developer stress is hard to quantify, but it’s real.”
- The senior development manager with a high-tech organization said, “I think feature flags are one of those things that can reduce stressful releases and they are true game-changers.”

Software developers are in high demand, and they are difficult to retain. A LinkedIn study found turnover rates of 21.7% for embedded software engineers.<sup>6</sup> The cost of recruiting, hiring, and training new engineers is expensive, and it has a material impact on a company’s bottom line. LaunchDarkly contributes directly to the reduction of developer stress by decreasing the risk of massive errors being found at the point of merging code. In turn, this will improve employee experience for teams using feature flags.

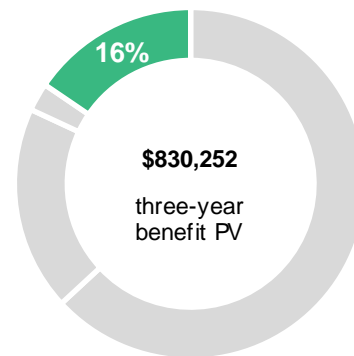
**Modeling and assumptions.**

- The cost of employee turnover is 20% of the annual fully loaded salary.
- The number of developers corresponds with the number of LaunchDarkly licenses sold.
- Forrester assigned a 22% turnover rate for software engineers.
- Forrester attributed LaunchDarkly with 10% of the credit for retaining the reduction in recruiting costs.

**Risks.** The actual results of this benefit may vary with:

- The overall size of the organization.
- The variances in salaries for members of the development team in different regions.
- The actual voluntary turnover rates for each organization.

To account for these risks, Forrester adjusted this benefit downward by 10%, yielding a three-year, risk-adjusted total PV of \$830,252.



### Reduction In Developer Recruitment Costs

Ref.	Metric	Calculation	Year 1	Year 2	Year 3
D1	Cost of replacing a software engineer	A3*20% of annual fully burdened salary	\$33,400	\$33,400	\$33,400
D2	Total number of software engineers using LaunchDarkly		100	500	1,000
D3	Percent turnover for software engineers	LinkedIn	22%	22%	22%
D4	Percent credit given to LaunchDarkly	Forrester data	10%	10%	10%
Dt	Reduction in developer recruitment costs	$D1 * D2 * D3 * D4$	\$73,480	\$367,400	\$734,800
	Risk adjustment	↓10%			
Dtr	Reduction in developer recruitment costs (risk-adjusted)		\$66,132	\$330,660	\$661,320
<b>Three-year total: \$1,058,112</b>			<b>Three-year present value: \$830,252</b>		



## UNQUANTIFIED BENEFITS

Additional benefits that customers experienced but were not able to quantify include:

- **Improved alignment with the business.** Utilizing feature flags allows for precise control over the deployment of features and choosing which users can access those features. This allows development teams to form better alignment between their groups and the business, ultimately delivering a better customer experience. The senior development manager in the high-tech industry said: “Our R&D teams are closely aligned with product marketing, product management, sales, and customer engagement teams. To ensure what we release is high-quality and meets customer needs, we engage with our customers as early as possible. We inform sales of what’s available to launch, and then our customer teams reach out to their customers and turn on the features. This gives us more engagement with the customer, rather than just turning things on under the hood.”
- **Improved customer experience.** Organizations continually look for ways to create better experiences for their customers. The DevOps leader with a manufacturer explained how LaunchDarkly improved their organization’s ability to collaborate with customers without the added expense and intrusion of setting up a concurrent production environment. They said: “LaunchDarkly has allowed for feature validation with customers enabling it without setting up a separate environment. It’s right there in their own working systems. It’s much less invasive.”
- **Improved compliance with external governance.** A platform engineer for a banking organization noted that the regulatory body that governs the banking industry requires service-level agreements (SLAs) for delivering services to customers. The regulations say that if consumers are unable to access their accounts

for a certain number of days, they can submit a proposition to remove the banking license. The platform engineer said: “We were able to change our risk category. Before [using] LaunchDarkly, we were only able to release during the night. Now we can release during the day, and we don’t have to stay up all night to do releases. That makes our teams very happy.”

## FLEXIBILITY

The value of flexibility is unique to each customer. There are multiple scenarios in which a customer might implement LaunchDarkly and later realize additional uses and business opportunities, including:

- **Increased use of LaunchDarkly for A/B testing.** Interviewees described the various methods their organizations utilized to perform usability testing such as customer visits, paper prototypes, and mock UIs. These methods are manual and time-consuming. Interviewees also expressed interest in using LaunchDarkly internally as part of their organizations’ release processes and running A/B testing to give the business ample time to incorporate user feedback into features.
- **Enabling the maturity of DevOps practices.** Feature flags are a tool that DevOps leaders use to increase application velocity. They allow for launching complex feature sets incrementally versus in biweekly sprints. This is one tool in a DevOps kit that allows leaders to execute on a continuous delivery vision.

Flexibility would also be quantified when evaluated as part of a specific project (described in more detail in [Appendix A](#)).

# Analysis Of Costs

■ Quantified cost data as applied to the composite

Total Costs							
Ref.	Cost	Initial	Year 1	Year 2	Year 3	Total	Present Value
Etr	Software license fees	\$0	\$121,000	\$473,000	\$913,000	\$1,507,000	\$1,186,860
Ftr	Deployment, training, and management costs	\$52,360	\$92,752	\$137,632	\$152,592	\$435,336	\$365,070
	Total costs (risk-adjusted)	\$52,360	\$213,752	\$610,632	\$1,065,592	\$1,942,336	\$1,551,930

## SOFTWARE LICENSE FEES

**Evidence and data.** The LaunchDarkly platform is licensed to customers for an annual fee. The licenses are typically based on the number of developers who will engage with the product.

### Modeling and assumptions.

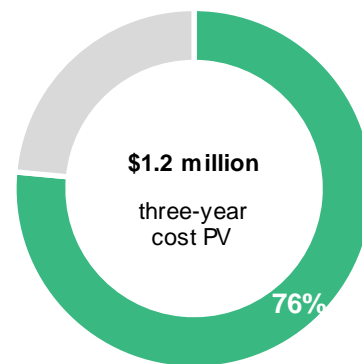
The cost of implementing LaunchDarkly for the composite organization is based on the following:

- A per-user license charge.
  - Year 1: 100 users
  - Year 2: 500 users
  - Year 3: 1000 users
- A fee for access to the platform.
- Where applicable, LaunchDarkly offers a volume discount, but that is not reflected in this model.

**Risks.** The costs may vary based upon:

- The number of licenses purchased.
- Potential volume discounts.

To account for these risks, Forrester adjusted this cost upward by 10%, yielding a three-year, risk-adjusted total PV (discounted at 10%) of \$1.2 million.



## Software License Fees

Ref.	Metric	Calculation	Initial	Year 1	Year 2	Year 3
Et	Software license fees	LaunchDarkly	\$0	\$110,000	\$430,000	\$830,000
	Risk adjustment	↑10%				
Etr	Software license fees (risk-adjusted)		\$0	\$121,000	\$473,000	\$913,000
<b>Three-year total: \$1,507,000</b>			<b>Three-year present value: \$1,186,860</b>			

## DEPLOYMENT, TRAINING, AND MANAGEMENT COSTS

**Modeling and Assumptions.** Interviewees described the following categories of costs:

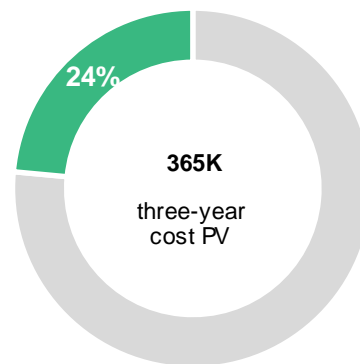
- **Internal deployment costs:** Interviewees described processes in which onboarding users required provisioning them through Okta. For the composite organization, this takes a total of 700 hours at \$68 per hour (fully burdened) of an IT professional's time.
- **Training:** LaunchDarkly provided training onsite. Interviewees said their organizations took a train-the-trainer approach to rolling this out. The internal training took 2 hours per user. These costs encompass both the training that LaunchDarkly provides and the follow-up training LaunchDarkly provides internal employees

**Ongoing internal management costs:** One full-time FTE dedicates 50% of their time to the ongoing maintenance and adoption of LaunchDarkly.

**Risks.** The costs for deployment, training, and management will vary based on:

- The size and scope of the organization deploying LaunchDarkly.
- The variances in salaries for members of the development team in different regions.

To account for these risks, Forrester adjusted this cost upward by 10%, yielding a three-year, risk-adjusted total PV of \$365,070.



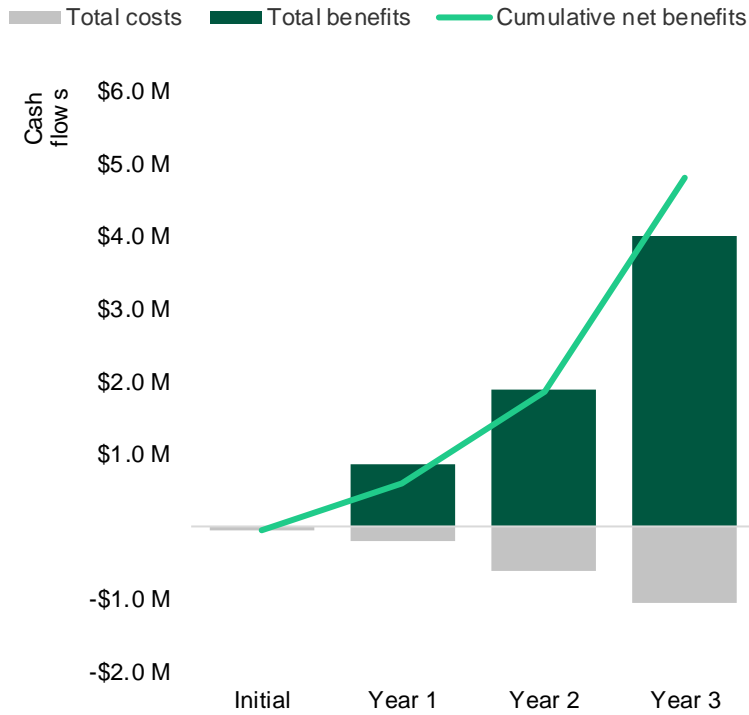
### Deployment, Training, And Management Costs

Ref.	Metric	Calculation	Initial	Year 1	Year 2	Year 3
F1	Internal deployment team costs	700 hours*\$68 per hour	\$47,600			
F2	Ongoing internal management costs	50% of FTE		\$70,720	\$70,720	\$70,720
F3	Number of new licenses			100	400	500
F4	Training	2 hours of training *\$68 per hour *number of new licenses		\$13,600	\$54,400	\$68,000
Ft	Deployment, training, and management costs	F1+F2+F3	\$47,600	\$84,320	\$125,120	\$138,720
	Risk adjustment	↑10%				
Ftr	Deployment, training, and management costs (risk-adjusted)		\$52,360	\$92,752	\$137,632	\$152,592
<b>Three-year total: \$435,336</b>			<b>Three-year present value: \$365,070</b>			

# Financial Summary

## CONSOLIDATED THREE-YEAR RISK-ADJUSTED METRICS

### Cash Flow Chart (Risk-Adjusted)



The financial results calculated in the Benefits and Costs sections can be used to determine the ROI and NPV for the composite organization's investment. Forrester assumes a yearly discount rate of 10% for this analysis.

**These risk-adjusted ROI, and NPV values are determined by applying risk-adjustment factors to the unadjusted results in each Benefit and Cost section.**

### Cash Flow Analysis (Risk-Adjusted Estimates)

	Initial	Year 1	Year 2	Year 3	Total	Present Value
Total costs	(\$52,360)	(\$213,752)	(\$610,632)	(\$1,065,592)	(\$1,942,336)	(\$1,551,930)
Total benefits	\$0	\$863,532	\$1,884,150	\$4,006,116	\$6,753,798	\$5,352,032
Net benefits	(\$52,360)	\$649,780	\$1,273,518	\$2,940,524	\$4,811,462	\$3,800,102
ROI						245%

# Appendix A: Total Economic Impact

Total Economic Impact is a methodology developed by Forrester Research that enhances a company's technology decision-making processes and assists vendors in communicating the value proposition of their products and services to clients. The TEI methodology helps companies demonstrate, justify, and realize the tangible value of IT initiatives to both senior management and other key business stakeholders.

## TOTAL ECONOMIC IMPACT APPROACH

**Benefits** represent the value delivered to the business by the product. The TEI methodology places equal weight on the measure of benefits and the measure of costs, allowing for a full examination of the effect of the technology on the entire organization.

**Costs** consider all expenses necessary to deliver the proposed value, or benefits, of the product. The cost category within TEI captures incremental costs over the existing environment for ongoing costs associated with the solution.

**Flexibility** represents the strategic value that can be obtained for some future additional investment building on top of the initial investment already made. Having the ability to capture that benefit has a PV that can be estimated.

**Risks** measure the uncertainty of benefit and cost estimates given: 1) the likelihood that estimates will meet original projections and 2) the likelihood that estimates will be tracked over time. TEI risk factors are based on "triangular distribution."



## PRESENT VALUE (PV)

The present or current value of (discounted) cost and benefit estimates given at an interest rate (the discount rate). The PV of costs and benefits feed into the total NPV of cash flows.



## NET PRESENT VALUE (NPV)

The present or current value of (discounted) future net cash flows given an interest rate (the discount rate). A positive project NPV normally indicates that the investment should be made, unless other projects have higher NPVs.



## RETURN ON INVESTMENT (ROI)

A project's expected return in percentage terms. ROI is calculated by dividing net benefits (benefits less costs) by costs.



## DISCOUNT RATE

The interest rate used in cash flow analysis to take into account the time value of money. Organizations typically use discount rates between 8% and 16%.

## Appendix B: Endnotes

---

<sup>1</sup> Source: “Deliver Better IT Assurance With Modern Technology Operations,” Forrester Research, Inc., January 6, 2021.

<sup>2</sup> Source: “Nimble To The Core: Drive Continuous Delivery And DevOps At A Holistic Level,” Forrester Research, Inc., November 20, 2018.

<sup>3</sup> Total Economic Impact is a methodology developed by Forrester Research that enhances a company’s technology decision-making processes and assists vendors in communicating the value proposition of their products and services to clients. The TEI methodology helps companies demonstrate, justify, and realize the tangible value of IT initiatives to both senior management and other key business stakeholders .

<sup>4</sup> Source: Dan Radigan, “Feature branching your way to greatness,” Atlassian Agile Coach (<https://www.atlassian.com/agile/software-development/branching>).

<sup>5</sup> Source: Michael Booz, “These 3 Industries Have the Highest Talent Turnover Rates,” LinkedIn Talent Blog, March 15, 2018 ([business.linkedin.com/talent-solutions/blog/trends-and-research/2018/the-3-industries-with-the-highest-turnover-rates](https://business.linkedin.com/talent-solutions/blog/trends-and-research/2018/the-3-industries-with-the-highest-turnover-rates)).

<sup>6</sup> Source: Paul Petrone, “See The Industries With the Highest Turnover (And Why It’s So High),” LinkedIn Learning Blog, March 19, 2018 (<https://www.linkedin.com/business/learning/blog/learner-engagement/see-the-industries-with-the-highest-turnover-and-why-it-s-so-hi>).

FORRESTER®