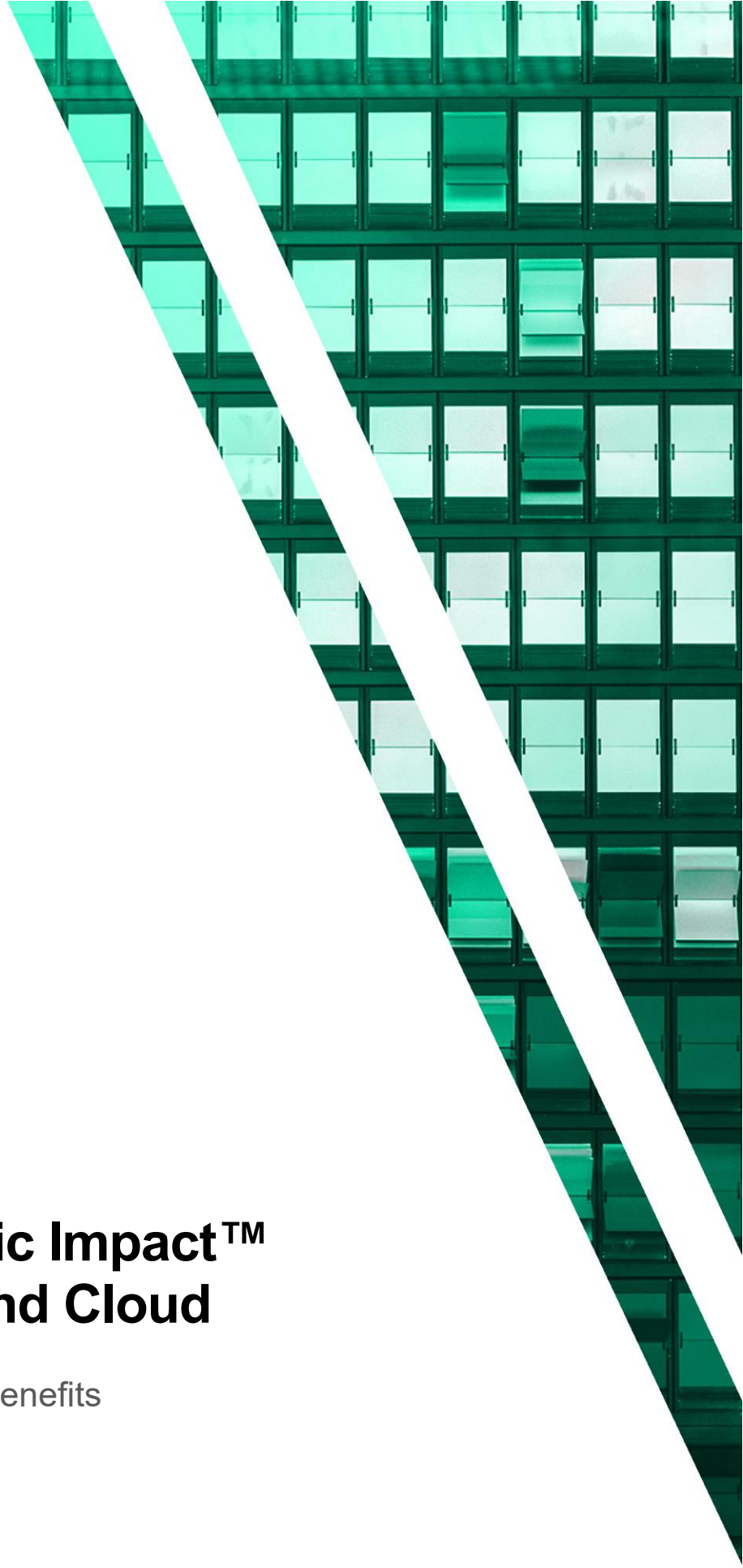
FORRESTER®



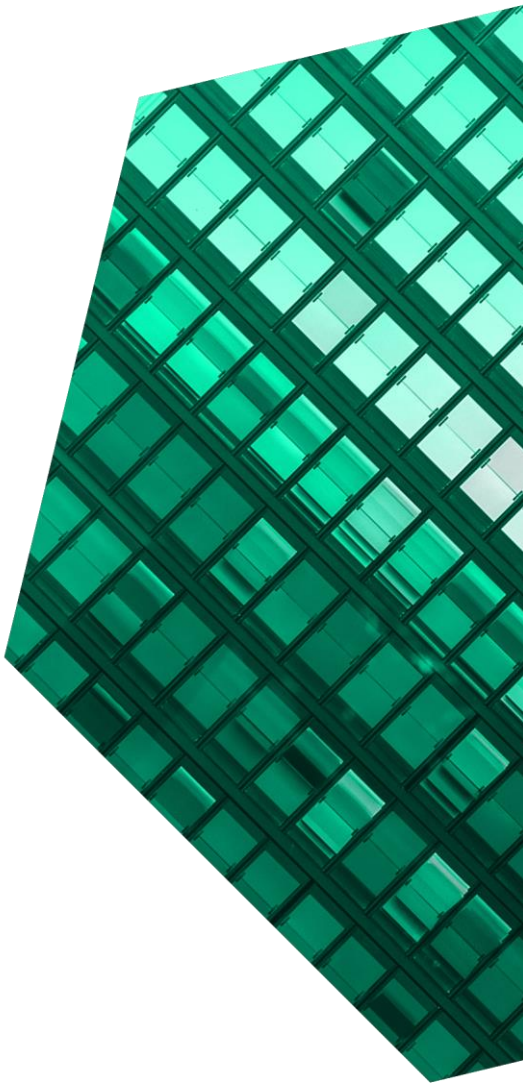
The Total Economic Impact™ Of Vercel's Frontend Cloud

Cost Savings And Business Benefits Enabled By Frontend Cloud

JANUARY 2024

Table Of Contents

Consulting Team: Jeffrey Yozwiak
Zahra Azzaoui



ABOUT FORRESTER CONSULTING

Forrester provides independent and objective research-based consulting to help leaders deliver key outcomes. Fueled by our customer-obsessed research, Forrester's seasoned consultants partner with leaders to execute their specific priorities using a unique engagement model that ensures lasting impact. For more information, visit forrester.com/consulting.

Executive Summary

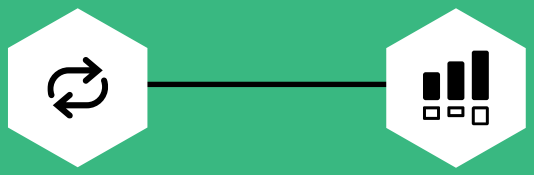
User experience is crucial to any online business. Performant, personalized, dynamic websites convert customers whereas slow and unreliable websites create abandonment and churn. With Next.js and Vercel, five front-end development teams built industry-leading websites while at the same time increasing developer productivity and lowering infrastructure costs.

Vercel's Frontend Cloud is a web application development and deployment platform. Vercel provides software teams with complete environments for developing, deploying, and iterating on high-performance, personalized web experiences. Vercel does this by automatically provisioning, scaling, and securing dynamic web content across global edge networks. The Vercel team also develops and maintains Next.js, an open-source framework for React, the popular JavaScript library for building user interfaces. (Although Next.js applications are particularly well-suited to being hosted on Vercel, Vercel also integrates with a variety of open-source frameworks and languages.)

Vercel commissioned Forrester Consulting to conduct a Total Economic Impact™ (TEI) study and examine the potential return on investment (ROI) enterprises may realize by switching to Vercel.[1] The purpose of this study is to provide readers with a framework to evaluate the potential financial impact of adopting Vercel at their organizations.

To better understand the benefits, costs, and risks associated with this investment, Forrester interviewed five representatives with experience using Vercel's Frontend Cloud. For the purposes of this study, Forrester aggregated the interviewees' experiences and combined the results into a single composite organization. The composite organization is a large enterprise with $1.5 billion in annual revenue. The composite organization's website has 250 million users and is supported by a team of 50 front-end developers as well as designers and marketers.

KEY STATISTICS



Return on investment (ROI)
264%

Net present value (NPV)
$9.53M

Before the interviewees' organizations migrated to Next.js and Vercel, their front-end development workflows were inefficient, and their websites' user experiences were poor. For example, at one organization, developers spent up to 40% of their time managing infrastructure instead of building new products and features for customers. Worse, poor website performance was hurting the business. Long page load times, frequent downtimes, poor search engine optimization (SEO), and lackluster user experiences all cost customers.

Using Next.js and Vercel, the interviewees' front-end teams built highly performant, content-rich websites dynamically personalized to their users. Vercel enabled efficient development workflows, and developer productivity improved. As the teams collaborated better and iterated faster, they delivered more features. And as the organizations enhanced

their websites, traffic and conversion rates rose. According to the interviewees, website user experience improvements directly contributed to business growth.

KEY FINDINGS

Quantified benefits. Three-year, risk-adjusted present value (PV) quantified benefits for the composite organization include:

- **Developers spend 90% less time managing front-end infrastructure.** Vercel simplifies and automates infrastructure-related tasks such as traffic routing, scaling, caching, performance optimization, and more. By switching to Vercel, the front-end engineering team saves tens of thousands of hours — time that it instead spends enhancing the composite organization's website and digital products. The time savings on infrastructure management alone are worth $2.4 million to the composite organization.
- **Developers spend 80% less time building and deploying code.** Faster build times and smoother releases with Vercel enable the front-end team to release more frequently and more confidently. Code quality also improves due to streamlined workflows and features like Preview Deployments. The time savings during code releases are worth $489,000 to the composite organization.
- **Developers release four times more major website enhancements and improve website performance by up to 90%.** The front-end team's velocity increases as developers spend more time building features and as the team releases more frequently and successfully. By using Next.js and Vercel, the front-end team improves core website performance metrics and creates a performant, dynamic, and personalized user experience. For the composite organization, these website enhancements drive business value by improving conversion rates and reducing abandonment.
- **Higher customer conversion rates generate $2.6 million in incremental profits.** Engaging, dynamic, and performant web experiences lead users to convert to customers at slightly higher rates. For the composite organization, even a conservative, marginal increase in conversion rates results in tens of millions of dollars in incremental revenue.
- **Higher website traffic generates $7.7 million in incremental profits.** The front-end team improves core web vitals (e.g., layout shift, page load times), technical SEO, and site uptime during high-traffic periods. More traffic at the top of the funnel translates into additional customers at the bottom. For the composite organization, these improvements are worth hundreds of millions of dollars in topline revenue.

> **"[Vercel] has been really transformative. The zero configuration and the Preview Deployments [enable] product folks, business stakeholders, [and] designers to interact and work with developers much more collaboratively than ever before. That's really helped transform not just the developer experience but [also] the stakeholder experience — the nontechnical experience — for other folks inside [our organization]."**
>
> *Principal engineer, e-commerce*

Unquantified benefits. Benefits that provide value for the composite organization but are not quantified in this study include:

- **Collaboration between developers and business stakeholders.** Preview Deployments and other Vercel features make it easy for nontechnical stakeholders to review development work pre-release. This facilitates rapid iteration and improves product quality.
- **Vercel enterprise support.** Using Next.js in partnership with Vercel — rather than another provider — unlocks additional benefits such as robust enterprise support, early access to new features, and opportunities to influence the evolution of Next.js.

Costs. Three-year, risk-adjusted PV costs for the composite organization include:

- **Incremental implementation of Next.js and Vercel over 12 months.** Switching to Next.js and Vercel is a major project that the front-end team spreads out over several months. The front-end team migrates the website incrementally, so the composite organization realizes value from its investments early on. The migration work is performed by only one-third of the front-end developers; most of the team continues to support the legacy website. Although the composite organization does not need to use Next.js and Vercel together (it could use Next.js with another vendor, or it could use Vercel with another framework), it migrates to both platforms to reap the advantages that they offer together. Migrating to Next.js takes most of the time; once the website is using Next.js, switching to Vercel takes comparatively little effort.
- **Part-time training on Next.js over three months.** Because Next.js is a new framework for developers at the composite organization, they devote some time to learning it (e.g., reading documentation).
- **Part-time management of Vercel by two developers.** One to two employees administer Vercel on an ongoing basis (e.g., handle updates). In total, the team spends 97% less time managing Vercel than it spends managing its prior front-end infrastructure.
- **Subscription to Vercel — including user seats, usage, optional features, and enterprise support — totaling $243,000 per year.** The composite organization pays for extra Vercel user seats (e.g., for contractors), and it subscribes to Vercel's highest level of enterprise support.

The representative interviews and financial analysis found that a composite organization experiences benefits of $13.13 million over three years versus costs of $3.60 million, adding up to a net present value (NPV) of $9.53 million and an ROI of 264%.

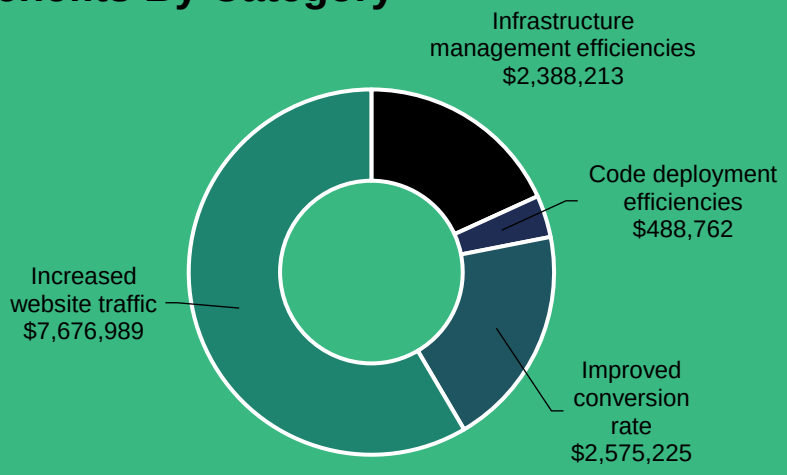
ROI
264%

BENEFITS PV
$13.13M

NPV
$9.53M

LESS TIME TO BUILD AND DEPLOY CODE
80%

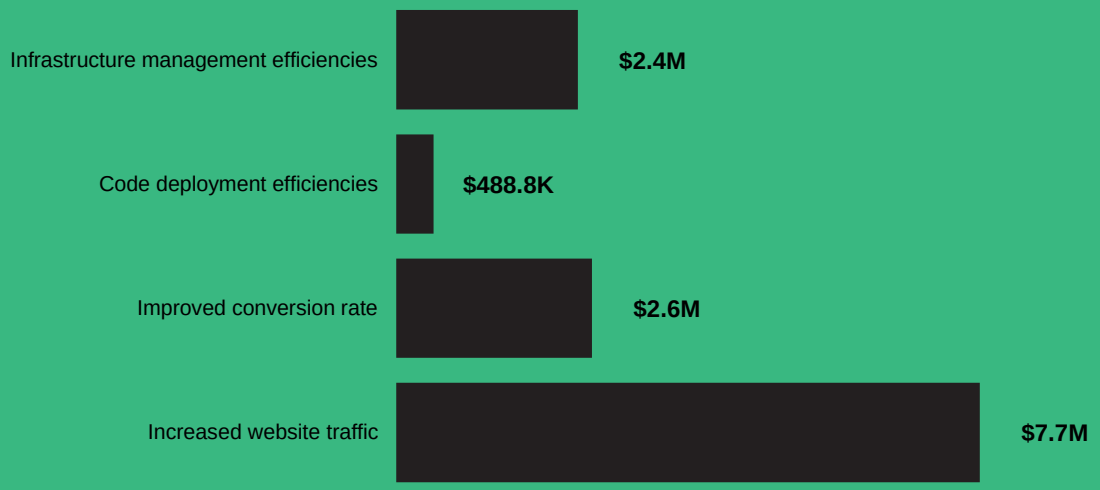
Benefits By Category



Benefits of Vercel related to **developer experience** include infrastructure management efficiencies and code deployment efficiencies. Together, these benefits comprise 22% of total benefits.

Benefits of Vercel related to **user experience** include improved conversion rate and increased website traffic. Together, these benefits comprise 78% of total benefits.

Benefits (Three-Year)



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 Frontend Cloud.

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 Frontend Cloud can have on an organization.

DUE DILIGENCE

Interviewed Vercel stakeholders and Forrester analysts to gather data relative to Frontend Cloud.

INTERVIEWS

Interviewed five representatives at organizations using Frontend Cloud to obtain data with respect to costs, benefits, and risks.

COMPOSITE ORGANIZATION

Designed a composite organization based on characteristics of the interviewees' 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 interviewees.

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 Vercel Frontend Cloud Customer Journey

Drivers leading to the Vercel investment

Interviews

Role	Industry	Region	Development Team
CTO	Media	Europe	25 developers
Senior engineer (front-end)	Financial services	North America	425 developers total (100 front-end developers)
Director of infrastructure	Financial services	Europe	30 developers total (15 front-end developers)
Manager of engineering	E-commerce	Global	30 developers total (20 front-end developers)
Principal engineer (front-end)	E-commerce	North America	10 developers

KEY CHALLENGES

The interviewees described implementing Vercel in a variety of situations. Their organizations had used mixtures of homegrown and off-the-shelf technologies, and it was common for development teams to include both in-house developers and contractors. Three interviewees described their legacy websites as monoliths — large, single codebases that were difficult to update. Although one organization hosted its website itself, most organizations had experience with other cloud providers before Vercel. Three of the organizations had even used Next.js with alternative cloud providers before switching to Vercel.

Despite these different circumstances, the interviewees reported several common challenges, including:

- **Development was slow and developers were frustrated.** The interviewees described several ways in which their legacy front-end technologies contributed to inefficient development workflows.
    - **Managing front-end infrastructure consumed 10% to 40% of developers' time.** The interviewees reported that their front-end teams spent between 10% and 40% of their time on fundamental tasks such as managing pre-production environments and releases and managing caching. The interviewees sought to save their developers' time so that they could spend more time building new features.

> **"I'm coming from a world in which there was so much pain just trying to do very basic things that everything in the Vercel world feels nice and effortless."**
>
> *Senior engineer, financial services*

    - **Releases were time-consuming and thus infrequent.** The interviewees said that their legacy front-end technologies made deploying code laborious. First, because websites were monoliths and

page sizes were large, creating builds could take hours. Second, monolithic codebases required extra coordination among developers.

For example, the director of infrastructure in financial services reported that deploying to production took 5 hours and that testing during integration — when developers merge their local work with the central codebase — was so painful that developers often neglected to do it. Similarly, the principal engineer in e-commerce said: "Due to the nature of the monolithic app ... to roll out a new feature, you needed to involve three, four, five different domain experts. … That was certainly a bottleneck."

Because releases required so much effort, they were few and far between. The interviewees believed that by reducing this effort, they could accelerate their teams' release cadences.

- **Change failure rates were as high as 30% to 75%, and rolling back changes was painful.** The manager of engineering in e-commerce reported that each release had as many as 30 failures, and the CTO in media said that releases had issues about 75% of the time. After resolving the problems, the development teams then had to perform lengthy release processes over again. In addition, rolling back (or undoing) changes was difficult. The senior engineer in financial services said that their team's only option when there was an issue was to roll back the entire release — hundreds of changes. Similarly, the manager of engineering in e-commerce described how rolling back changes took so long that when a release failed, the team would have to wait until the next day to redeploy.

> **"We felt that if we moved to Vercel, we could just focus on building components and shipping rather than on things that are just not our business: writing a lot of infrastructure and pipeline and feature-branching logic and caching logic and serving logic."**
>
> *Director of infrastructure, financial services*

- **Improving website performance was a business imperative.** Poor website performance hurt the interviewees' businesses in multiple ways. Interviewees described long page load times; for instance, the senior engineer in financial services said that 5% of users experienced page load times of 16 seconds. The senior engineer explained that the problem was the organization's legacy front-end technology: There was little caching, and loading the homepage required loading the code for every page on the site — including internal admin pages.

  A second common problem was poor SEO. For example, the CTO at the media organization said that of the site's hundreds of thousands of pages, only 2% to 3% ranked well in search engines. Pages took seconds to load, and most search engine algorithms factor in website performance.

  Uptime — a third common challenge — was particularly important to the interviewees from e-commerce organizations. The manager of

engineering in e-commerce explained that direct-to-consumer sales through the organization's website accounted for almost a quarter of the organization's total sales and were more profitable than sales in brick-and-mortar stores. At any given time, the website has thousands of visitors, but traffic is five to 10 times higher during peak shopping periods like Cyber Monday. Before Vercel, such traffic spikes caused the website to crash, which meant lost sales.

Beyond improving site performance, some interviewees sought to enhance users' experiences with personalization and new features, but they found themselves limited by their legacy technologies. The manager of engineering in e-commerce explained: "Our website [now] is heavily content-rich — lots of images, lots of moving things. It looks pretty. When we were on [our old provider], we couldn't do that level of customization. … Next.js is very well suited for the commerce world in that pages [can] be a lot more dynamic than [before]."

"The main [driver] we had from our business partners was site performance. And a lot of that is customer experience. … We needed to have a good-performing site."

Principal engineer, e-commerce

INVESTMENT OBJECTIVES

The interviewees' organizations invested in Vercel to achieve two goals:

- **Improve developer experience.** The interviewees wanted to eliminate the inefficiencies caused by their legacy front-end technologies. They sought a platform that would be low-maintenance, so developers could spend less time managing infrastructure and more time shipping new features.
- **Improve user experience.** The interviewees saw that they could leverage Next.js to build highly performant, content-rich websites. They and their business stakeholders believed that attractive user experiences could impact their businesses — e.g., via better SEO, increased website traffic, higher conversion rates, and lower abandonment rates.

"Vercel [develops] Next.js, so ultimately they built their platform with Next.js in mind. It felt like a good fit."

Manager of engineering, e-commerce

The interviewees evaluated multiple technologies and built proofs-of-concept before deciding to adopt Next.js. Three of the organizations experimented with using Next.js on other cloud providers before they partnered with Vercel. Ultimately, as the director of infrastructure in financial services said, "To get the real benefits of Next.js, being on Vercel makes a lot of sense."

Most interviewees planned to incrementally migrate their organizations' websites to Next.js and Vercel over many months. Their teams would need to rewrite code, migrate content, retire legacy technologies, and more. (Only one organization did not take an incremental approach because its legacy website was managed by an outside contractor.)

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 five interviewees, 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 composite is a global organization with $1.5 billion in revenue. Half of the organization's revenue is generated via its website (the other half is generated via, e.g., physical channels and partnerships). Annually, the website has 250 million users and 3.75 million customers. Users browse the website meaningfully but do not contribute to revenue. On the other hand, customers do contribute to revenue — e.g., paying subscribers or shoppers who make purchases. The average annual revenue per customer is $200. Finally, the composite's IT department has 100 developers, of whom half are front-end developers. These developers collaborate with multiple designers and marketers to create the composite organization's website.

Deployment characteristics. The composite undertakes a strategic initiative to modernize and improve its website. Its legacy website is a monolithic app supported by both off-the-shelf and homegrown tools. The composite migrates to both Next.js and Vercel.

Key Assumptions

- **$1.5 billion in revenue**
- **250 million website users**
- **50 front-end developers**

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	Infrastructure management efficiencies	$960,336	$960,336	$960,336	$2,881,008	$2,388,213
Btr	Code deployment efficiencies	$196,538	$196,538	$196,538	$589,615	$488,762
Ctr	Improved conversion rate	$637,500	$1,062,500	$1,487,500	$3,187,500	$2,575,225
Dtr	Increased website traffic	$1,511,831	$3,425,500	$4,620,706	$9,558,038	$7,676,989
	Total benefits (risk-adjusted)	$3,306,206	$5,644,874	$7,265,081	$16,216,161	$13,129,189

INFRASTRUCTURE MANAGEMENT EFFICIENCIES

Evidence and data. The interviewees said that before Vercel, managing their websites' front-end infrastructures regularly consumed between 10% and 40% of their development teams' time. Example tasks included maintaining, updating, and troubleshooting technologies in the organizations' front-end stacks; writing and supporting custom infrastructure tools; and manually managing aspects of site performance, such as page caching, server utilization, and image optimization.

According to the interviewees, migrating to Vercel eliminated that effort. The Next.js framework and Vercel platform automatically handled front-end infrastructure management needs such as traffic routing, scaling, caching, optimization, and more.

The interviewees explained the benefits of Vercel as follows:

- The senior engineer in financial services said: "If you're using Next.js in Vercel, [you can take advantage of] framework-derived infrastructure. … You just use the framework, and then based on how you've written your app when you deploy it, the correct infrastructure is spun up for you. … A lot of the infrastructure needs are just handled."
- The manager of engineering in e-commerce said: "Somebody actually pinged me before this call. They want to put up a little site in Vercel. I know they'll be up and running by the end of the day. [With Vercel], you're not just getting hosting, although it's a very reliable hosting platform. It's everything [else you get]. It's the full package — the whole experience."

"Pretty much overnight, we quit worrying about infrastructure completely. We're really able to just focus on how we can differentiate our business and application in terms of user experience."

Principal engineer, e-commerce

The interviewees also quantified the following benefits from Vercel:

- Before Vercel, the media organization had three DevOps engineers dedicated to maintaining its internal systems. The CTO said that switching to Vercel freed up those engineers' time, and they now deliver product work.
- Similarly, at one of the financial services organizations, front-end developers had spent up to 30% of their time managing infrastructure before Vercel. According to the senior engineer, migrating to Vercel eliminated that effort entirely, allowing the developers to focus on product work. The senior engineer also estimated that without Next.js and Vercel, managing caching alone would have required two to three full-time equivalent (FTE) developer hires.
- At one of the e-commerce organizations, developers had spent 15% to 20% of their time on infrastructure maintenance before Vercel, and that time doubled during periods of peak traffic (e.g., holiday shopping seasons). The principal engineer said that using Vercel not only saved that time, but it also allowed a relatively small team to accomplish more. The principal engineer noted synergies between Next.js and Vercel: They estimated saving between three to four site reliability FTEs by running Next.js on Vercel rather than another provider.

"We no longer need this huge amount of [developer] resources to maintain a cloud infrastructure. The devs can just do what they do best."

CTO, media

"[As a developer], you're really only thinking about your app code. … You just implement a JavaScript function … and all the infrastructure is handled for you, all the monitoring is already in place, and all the technical complexity is very nicely abstracted away."

Senior engineer, financial services

Modeling and assumptions. For the composite organization, Forrester assumes:

- Before Next.js and Vercel, the composite organization's front-end developers spend about 20% of their time managing front-end infrastructure. The total time spent by the team sums up to 10 FTEs, although in practice, the work is distributed across all the front-end developers.
- After the composite implements Next.js and Vercel, front-end developers spend 90% less time managing infrastructure.
- In general, the developers capture and use 75% of the time they save for productive purposes (e.g., building new features). Invariably, they lose some time to less-productive purposes (e.g., coffee breaks, meetings, etc.).
- Developers earn a fully burdened annual salary of $150,000. (Fully burdened salaries include benefits and are higher than base salaries.)

Risks. This benefit is most likely to vary across organizations based on how much time developers spend managing legacy front-end infrastructure before Vercel. Organizations at which developers

Reduction in developer time spent managing front-end infrastructure

90%

spend more time maintaining legacy infrastructure may realize greater benefits.

Results. To account for these risks, Forrester adjusted this benefit downward by 5%, yielding a three-year, risk-adjusted total PV (discounted at 10%) of $2.4 million.

Infrastructure Management Efficiencies

Ref.	Metric	Source	Year 1	Year 2	Year 3
A1	Front-end developers	Composite	50	50	50
A2	Percentage of time spent managing front-end infrastructure before Vercel	Interviews	20%	20%	20%
A3	Developer FTEs managing front-end infrastructure before Vercel	A1*A2	10.0	10.0	10.0
A4	Time spent managing front-end infrastructure before Vercel (hours)	A3*2,080	20,800	20,800	20,800
A5	Reduction in front-end infrastructure management time with Vercel	Interviews	90%	90%	90%
A6	Time spent managing front-end infrastructure after Vercel (hours)	A4*(100%-A5)	2,080	2,080	2,080
A7	Average developer salary (fully burdened)	TEI standard	$150,000	$150,000	$150,000
A8	Developer hourly rate (fully burdened)	A7/2,080	$72	$72	$72
A9	Percentage captured	TEI standard	75%	75%	75%
At	Infrastructure management efficiencies	(A4-A6)*A8*A9	$1,010,880	$1,010,880	$1,010,880
	Risk adjustment	↓5%			
Atr	Infrastructure management efficiencies (risk-adjusted)		$960,336	$960,336	$960,336
	Three-year total: $2,881,008			**Three-year present value: $2,388,213**	

CODE DEPLOYMENT EFFICIENCIES

Evidence and data. The interviewees described workflows for releasing code that — before Vercel — were time-consuming and error-prone. The interviewees reported long build and deploy times as well as high change failure rates caused by inefficient workflows. Moreover, rolling back changes was difficult and required developers to perform lengthy release processes over again.

The interviewees reported that using Vercel eliminated this effort. Build and deploy times fell drastically, and developers used Vercel features such as Preview Deployments and instant rollbacks to improve quality. Instead of large, irregular, and risky releases, they became incremental, frequent, and safe.

"Generally, [deploys] are more comfortable, a lot safer, and we just feel more confident with our code. That's huge — the developer experience [benefits are] huge."

Manager of engineering, e-commerce

The interviewees explained the benefits of Vercel as follows:

- The manager of engineering in e-commerce described how workflows enabled by Vercel improved quality: "The developer experience is just so much better. … When a developer makes their code change and pushes it, the build happens behind the scenes. … [Another] big efficiency change is that QA can test the code in a deployed environment without having it impact other areas. … Ultimately, we're just catching bugs earlier and often. And that's been a huge, huge win. So now when we go to production, it's generally quite uneventful, which is great."

"Developers definitely seem happy with the build times alone. They're just delighted with how quickly they can get their code deployed out."

Manager of engineering, e-commerce

- The CTO in the media industry described similar quality improvements after Vercel: "We can release any time of the day or night knowing quite comfortably that we're not going to break anything when we go live. Beforehand, there was always a risk. Now, you can just be a developer and build stuff and test it. … When we press the button to go live, we know exactly what's going live. [And] we can very quickly decide whether a release is a success or not and roll back if we need to."

  The CTO added: "Releases are a lot smoother than before, when we literally just chucked features out into the world and hoped they worked. [Now], all features go through the same [process]. It doesn't matter how big or small the feature is. All features go into a feature branch. The feature branch gets tested against various environments, and then it goes out into the wild. That process is very different from what we had before, and for us, it's working quite nicely."
- The director of infrastructure in financial services said: "If we wanted to do [twice as many] deployments a day, then that is not only possible

but much more likely. … We can scale to a much higher frequency of [deployments] now because it's so hands-off, whereas before, it was hands-on."

Development time with Vercel

4x faster

The interviewees also quantified the following benefits from Vercel:

- The manager of engineering at an e-commerce organization said that releasing code was four times faster with Vercel. Before Vercel, the organization had run Next.js on another platform. Switching to Vercel reduced build times from 20 minutes to 3 minutes. In addition, code quality improved. Before Vercel, each deploy had averaged around 30 failures, and fixing the issues had regularly required 1.0 to 1.5 FTEs. However, switching to Vercel changed this. Developers used Vercel's Preview Deployments to test more frequently, and because builds happened on every commit and pull request, there were fewer conflicts. The manager of engineering cited a third benefit of Vercel: Before Vercel, rolling back changes took 20 minutes; with Vercel, rolling back changes took seconds.
- The CTO at the media organization said that before Vercel, the team had released irregularly, and the releases had had issues 75% of the time. However, since Vercel, the team's release cadence has increased to twice per week, and the team is delivering more features than before: About one-third of the tickets in each release are minor fixes or tweaks, and the other two-thirds are tied to feature requests from the business. The CTO attributed these improvements to Vercel. With Vercel, the time to deploy code fell to 40 minutes, and any developer on the team could release to production. The team also benefited from easily standing up test environments and different feature branches with Vercel; by leveraging these capabilities, the team reduced its change failure rate and prevented countless hours of rework.
- The senior engineer in financial services described several workflow improvements since Vercel. Before Vercel, iteration cycles were long, and testing changes was painful. For example, local builds took 8 minutes, and integration testing took more than an hour. As a result, changes were often not well-tested. This created problems downstream: Every release included hundreds of changes, and if a release contained even a single issue, the entire release had to be rolled back.

  Migrating to Vercel alleviated these problems. Developers have leveraged Vercel's Preview Deployments to iterate quickly and improve quality; the team created 11,000 Preview Deployments per month. Build and deploy times were also faster: Any team can have a pipeline to production in 1 to 2 hours without any help from the larger team.

Modeling and assumptions. For the composite organization, Forrester assumes:

- Each release involves work from five developers (10% of the front-end team).
- Before Vercel:
  - Due to long build and deploy times, each release takes an average of 2 hours.
  - Since releases are so time-consuming, the team averages 1.5 releases per week.

    - Unfortunately, 75% of releases fail due to bugs or other issues.
    - On average, fixing these issues takes the release team 10 hours.
    - Once the issues are fixed, the release team must spend another 2 hours deploying code to production.
- After Vercel:
    - The time to build and deploy code falls by 83% — from 2 hours to 12 minutes.
    - The front-end team increases its release cadence to twice per week.
    - The change failure falls by 98% as only 2% of releases fail.
- In total, the development team spends about 94% less time on code deployments after Vercel. The team realizes these time savings despite increasing its release frequency.
- As before, developers earn a fully burdened annual salary of $150,000 and capture 75% of the time saved.

Build and deploy times

Before Vercel: **2 hours**

After Vercel: **12 minutes**

Risks. This benefit is most likely to vary across organizations based on:

- Average build and deploy times before Vercel.
- Change failure rate before Vercel.
- Average time to remediate failed releases.

Organizations do not need both high build times and high change failure rates to realize this benefit. Organizations can still see benefits if only one condition is true.

Results. To account for these risks, Forrester adjusted this benefit downward by 10%, yielding a three-year, risk-adjusted total PV of $489,000.

Code Deployment Efficiencies

Ref.	Metric	Source	Year 1	Year 2	Year 3
B1	Code releases before Vercel	Composite	78	78	78
B2	Percentage of front-end development team involved per release	Composite	10%	10%	10%
B3	Front-end developers involved per release	A1*B2	5	5	5
B4	Average time to build and deploy code per release before Vercel (hours)	Interviews	2	2	2
B5	Time spent releasing code before Vercel (hours)	B1*B3*B4	780	780	780
B6	Release failure rate before Vercel	Interviews	75%	75%	75%
B7	Failed releases before Vercel	B1*B6	59	59	59
B8	Average time to remediate a failed release (hours)	Interviews	10	10	10
B9	Average time to build and deploy again after a failed release (hours)	B4	2	2	2
B10	Time spent remediating failed releases before Vercel (hours)	B7*B3*(B8+B9)	3,540	3,540	3,540
B11	Subtotal: Time spent on code releases before Vercel (hours)	B5+B10	4,320	4,320	4,320
B12	Code releases after Vercel	Interviews	104	104	104
B13	Reduction in time to build and deploy code per release with Vercel	Interviews	83%	83%	83%
B14	Average time to build and deploy code per release after Vercel (hours)	B4*(100%-B13)	0.3	0.3	0.3
B15	Time spent building and deploying code after Vercel (hours)	B12*B3*B14	156	156	156
B16	Reduction in release failure rate with Vercel	Interviews	98%	98%	98%
B17	Release failure rate after Vercel	B6*(100%-B16)	2%	2%	2%
B18	Failed releases after Vercel	B12*B17	2	2	2
B19	Time spent remediating failed releases after Vercel (hours)	B18*B3*(B8+B9)	120	120	120
B20	Subtotal: Time spent on code releases after Vercel (hours)	B15+B19	276	276	276
B21	Time saved releasing code after Vercel (hours)	B11-B20	4,044	4,044	4,044
B22	Developer hourly rate (fully burdened)	A8	$72	$72	$72
B23	Percentage captured	A9	75%	75%	75%
Bt	Code deployment efficiencies	B21*B22*B23	$218,376	$218,376	$218,376
	Risk adjustment	↓10%			
Btr	Code deployment efficiencies (risk-adjusted)		$196,538	$196,538	$196,538
	Three-year total: $589,615			**Three-year present value: $488,762**	

INCREASED FEATURE DELIVERY

Evidence and data. Interviewees said that their front-end teams delivered more work with business value — e.g., new features or performance improvements — after their organizations switched to Next.js and Vercel. As described in Benefit A: Infrastructure Management Efficiencies and Benefit B: Code Deployment Efficiencies, developers saved time with Vercel. However, in addition to these developer productivity benefits, the interviewees reported faster release cadences and higher change success rates. The result was that the front-end teams significantly improved their organizations' websites in multiple ways — from faster page loads to net-new features.

"By using Next.js [with] Vercel, you unlock all these other features that Next.js and Vercel offer that can push the envelope even further from both a developer experience standpoint and a performance standpoint."

Senior engineer, financial services

The interviewees explained the benefits of Vercel as follows:

- The senior engineer in financial services described how Vercel's Preview Deployments feature helped the team to increase velocity: "Now, we can loop in designers, we can loop in managers, we can loop in product, and we can test changes even before they are merged. … The dev experience that produces just gives you a really nice feedback loop; with the ability to loop in non-dev folks, [development] is much more seamless. [So] you're seeing teams iterate faster. … There's not really a bottleneck in Next.js and Vercel, whereas the bottlenecks in our other system would often push even small things to take a week or longer."
- The manager of engineering in the e-commerce industry said: "We're releasing features faster, for sure. We're now releasing about every two weeks. When we were with [our old cloud provider], we would probably release once a month just because it was such an arduous task. Now, we release every other week." The manager of engineering also described how using Vercel helped the front-end team deliver a major enhancement to the organization's website: "We've released this new feature — a new part of the site — and there are a lot of new features coming. That [feature] was a huge win for us. … If we weren't on Next.js and Vercel, I don't think we would have been able to do it. … We were able to [use] the same repo [and] reuse components. … If we had still been on [our old provider], we wouldn't have done that."

"Features can be iterated on as quickly as the team is able to."

Senior engineer, financial services

The interviewees also quantified the following benefits from Vercel:

- The senior manager in financial services reported that core web vitals had improved by 75% since the organization switched to Next.js and Vercel. Before, core web vitals had been in the bottom 95% — partly because the average uncompressed page size was 5 MB. Even worse,

5% of pages took 16 seconds or longer to load. After switching to Vercel, the team reduced page sizes across the site by optimizing over 7,000 images. Since Vercel, the pages that used to take longer than 16 seconds to load now load in under 2 seconds.

- The director of infrastructure in financial services noted improvements of 20% to 90% in key website performance metrics. For example, the front-end team reduced the homepage size from 186 KB to less than 20 KB. The director of infrastructure also reported the following improvements in site performance metrics: Cumulative layout shift (CLS) fell from 0.15 to 0.01 (well below the recommended 0.10); first-page load times fell by about 200 milliseconds (thanks to better bot management and edge caching with Vercel); and largest contentful paint (LCP) fell by more than 400 milliseconds.
- The principal engineer in e-commerce reported 75% shorter development timelines. They compared building specific features in the organization's legacy environment to building the same or similar features with Next.js and Vercel. First, a product customization tool that had originally taken four months to develop was rebuilt in one month by a smaller team. Similarly, a new type of product page that might have taken four to five months to develop before was released in five weeks. In addition, technical SEO work that would have three sprints before was completed in days. The principal engineer explained: "[Because of] the amount of effort required to implement a lot of the technical SEO stuff … it never made it onto the priority list in the old world."

Modeling and assumptions. Forrester assumes:

- Each release at the composite organization includes about 100 code changes. These changes include bug fixes as well as work to support new features.

"At that moment of inspiration, you can kind of effortlessly just test things. … [In one case], within 2 hours, they had done what they had budgeted an entire quarter to do. Stuff like that has been really powerful and impactful."

Senior engineer, financial services

"Moving to Vercel has really opened the doors to getting our content team new features. … Everyone is happier, from our stakeholders to our content editors to our developers — and hopefully, our customers as well."

Manager of engineering, e-commerce

- The front-end team needs to release around 2,000 changes to complete a major website enhancement. Major website enhancements include new product features, performance improvements, and other substantial projects to improve the experience of website users.
- Before Vercel, the front-end team completes only one major website enhancement per year. This is because the front-end team spends much of its time managing infrastructure and releases instead of building new features. To be specific:

  - Because releases are so time-consuming, the front-end team averages only 1.5 releases per week.
  - In addition, 75% of changes fail.
  - As a result, the front-end team only successfully completes 1,950 changes — just enough to constitute a major website enhancement.
- After Vercel, the front-end team makes five major website enhancements per year. As the developers save time, velocity increases. To be specific:
  - The release cadence increases to twice per week.
  - With Vercel, 98% of changes succeed.
  - As a result, the front-end team successfully completes 10,244 changes — more than enough to constitute five major website enhancements.
- The composite organization benefits only from the four incremental website enhancements that using Vercel makes possible.
- In Year 1, each major website enhancement increases site performance by 10%. In Year 2, each enhancement increases performance by 7.5%, and in Year 3, each enhancement increases performance by only 5%. Thus, the front-end team sees diminishing marginal returns from its work. Improving the website is easy at first; e.g., many of the changes are "low-hanging fruit." However, as the website becomes better, improving it even further becomes harder and requires more development effort.
- Thus, the front-end team improves website performance by 40% in Year 1, 30% in Year 2, and 20% in Year 3. In Year 2, the cumulative improvement in performance is 70%; in Year 3, the cumulative improvement in performance is 90%.
- Benefit C: Improved Conversion Rate and Benefit D: Increased Website Traffic increase as the front-end team improves the composite organization's website. Both benefits reflect the business value of the front-end team's increased velocity by capturing the economic impact of the incremental website enhancements — i.e., the new features and other improvements — that the front-end team delivers.

Risks. These calculations will most likely vary across organizations based on how front-end teams organize their work. How development teams define, measure, and plan work is highly variable and often specific to organizations. Nevertheless, teams considering investing in Next.js and Vercel should still evaluate the potential impact of increased front-end development velocity at their organizations — regardless of how they define velocity.

In addition, the business value of the incremental work front-end teams deliver is likely to vary across organizations. However, this risk is accounted for in the calculations for Benefit C: Improved Conversion Rate and Benefit D: Increased Website Traffic.

"[Development has] drastically accelerated now versus in the old world."

Principal engineer, e-commerce

Increased Feature Delivery

Ref.	Metric	Source	Year 1	Year 2	Year 3
R1	Code releases before Vercel	B1	78	78	78
R2	Changes per release	Composite	100	100	100
R3	Release failure rate before Vercel	B6	75%	75%	75%
R4	Failed changes before Vercel	R1*R2*R3	5,850	5,850	5,850
R5	Successful changes before Vercel	R1*R2-R4	1,950	1,950	1,950
R6	Changes per major website enhancement	Composite	2,000	2,000	2,000
R7	Major website enhancements released before Vercel	R6/R5	1	1	1
R8	Code releases after Vercel	B12	104	104	104
R9	Release failure rate after Vercel	B17	2%	2%	2%
R10	Failed changes after Vercel	R8*R2*R9	156	156	156
R11	Successful changes after Vercel	R8*R2-R10	10,244	10,244	10,244
R12	Major website enhancements released after Vercel	R11/R6	5	5	5
R13	Incremental major website enhancements after Vercel	R12-R7	4	4	4
R14	Website performance increase per major website enhancement	Interviews	10.0%	7.5%	5.0%
R15	Website performance increase from major website enhancements	R13*R14	40%	30%	20%
R16	Subtotal: Cumulative website performance increase after Vercel	R16PY+R15	40%	70%	90%

IMPROVED CONVERSION RATE

Evidence and data. Interviewees said that the website enhancements made after switching to Next.js and Vercel contributed to material business wins for their organizations. By adding features, improving website performance and increasing personalization, the organizations created better user experiences for prospective customers. After the enhancements went live, sales through the organizations' websites increased. Based on a variety of factors — from A/A tests to market trends — the interviewees confidently attributed the higher conversion rates they saw to specific website improvements.

"[We] continue to unlock more performance wins, which we've correlated historically to conversion wins."

Senior engineer, financial services

Remaining competitive is critical for organizations in the fast-growing e-commerce space; having strong product offerings is no longer sufficient. According to Forrester research, roughly one-third (33%) of US online adults and nearly half (46%) of US online young adults ages 18 to 24 believe that it is important for websites to provide personalized product recommendations informed by their browsing history.[2] To convert users to customers, websites must offer frictionless user experiences that feel personal and relevant.

The interviewees explained the benefits of Vercel as follows:

- The manager of engineering in e-commerce said: "In comparison to our previous website, the Vercel version just feels quicker — it loads instantly, and it's content-rich. It's super-fast. After migrating to Vercel, our team immediately felt the difference, and I think our users and customers feel the same way because when you visit our site, just it feels snappier— there is no doubt about it."

  The manager of engineering continued: "Like any e-commerce [company], we make the most money by selling direct-to-consumer [DTC]. So it's imperative that we keep our [website] stable and performant and just attract as many people as we can through that channel. Having a platform that's stable and reliable is huge for us because every minute that we lose [to slow site performance or downtime] is potentially lost revenue."

  They concluded: "[Since moving to Vercel], we have seen an uptick in the big KPIs — performance of the site, page load [times], … conversion rate — along with the number of users on the site. [And] we have definitely seen an uptick in sales."

- The senior engineer in financial services explained the business impact of higher conversion rates at their organization: "We're talking about millions of dollars just from using a new framework and architecture and a very low-effort [migration]. … When you're having performance issues, a performance lift is very easily tied to business impact."

- The principal engineer in e-commerce directly attributed improved conversion rates to the website enhancements: "I can say confidently that most of the lift [in conversion] was due to website improvements. Since the big COVID peak [in DTC sales], the overall industry was down by 30% last year. [But] despite these macroeconomic conditions and a smaller marketing budget, our sales have remained stable since we released [the website on Vercel]. … From what I can see, the lion's share of [the lift] was from the web improvements."

The interviewees also quantified the following benefits from Vercel:

- The senior engineer at a financial services firm said that, after migrating specific high-traffic pages to Vercel, the organization saw a 13% improvement in site performance that correlated with a 5% increase in conversion rates. For this organization, the improvement was worth millions of dollars.
- The principal engineer in e-commerce reported that direct-to-consumer sales increased by 62% in the first year after the organization deployed its new site. The interviewee attributed this growth to user experience improvements made possible with Vercel.

"The look, feel, [and] performance of the site were all much more modern — instantly. We received a lot of great feedback from users on that — performance especially being really near and dear to us as engineers. And from a business perspective, during the first year of the new site, we saw direct-to-consumer sales grow about 62%. So that kind of indicated to us that the improved user experience and performance were actually affecting the bottom line."

Principal engineer, e-commerce

Modeling and assumptions. For the composite organization, Forrester assumes:

- The composite organization's website has 250 million users. These users are active on the site, but only some of them are paying customers.
- Before Vercel, users convert to customers at a rate of 1.5%.
- The front-end team improves the website as modeled in the previous section, Increased Feature Delivery.
- The website enhancements lead to slight increases in the user-to-customer conversion rate. Although cumulative website performance improves significantly (e.g., 90% in Year 3), the direct impact on the conversion rate is extremely small (modeled at 5% of the cumulative performance increase). The conversion rate increases to 1.53% in Year 1, 1.55% in Year 2, and 1.57% in Year 3.
- The average annual revenue per website customer is $200.
- The composite organization has an operating margin of 5%. (Operating profit is income minus both production costs and operating costs — e.g., rent, salaries, etc. An operating margin of 5% might be typical for an e-commerce company.)[3]

"From the user's perspective, the experience is night and day."

Principal engineer, e-commerce

Risks. This benefit is most likely to vary across organizations based on:

- Website users before Vercel.
- User-to-customer conversion rate before Vercel.

- Improvement in conversion rate due to website enhancements with Vercel.
- Average revenue per website customer.
- Operating margin.

Forrester took an exceptionally conservative approach to modeling this benefit in several ways:

- The user-to-customer conversion rate before Vercel is on the low end of industry benchmarks.
- The improvement in conversion rate due to website enhancements is conservative.
- Benefits are calculated based on operating margin rather than, for example, gross profit margin. (For potential benefits calculated based on gross profit margins for several industries, see Appendix B: Potential Gross Profits By Industry.)
- The operating margin for the composite organization is typical for an e-commerce organization but lower than the market average of 12%.[4]

Thus, organizations may easily realize greater benefits than those calculated for the composite.

Results. To account for these risks, Forrester adjusted this benefit downward by 15%, yielding a three-year, risk-adjusted total PV of $2.6 million.

Improved Conversion Rate

Ref.	Metric	Source	Year 1	Year 2	Year 3
C1	Website users before Vercel	Composite	250,000,000	250,000,000	250,000,000
C2	User-to-customer conversion rate before Vercel	Composite	1.50%	1.50%	1.50%
C3	Website customers before Vercel	C1*C2	3,750,000	3,750,000	3,750,000
C4	Improvement in user-to-customer conversion rate due to website enhancements with Vercel	C2*R16*5%	0.03%	0.05%	0.07%
C5	User-to-customer conversion rate after Vercel	C2+C4	1.53%	1.55%	1.57%
C6	Website customers after Vercel	C1*C5	3,825,000	3,875,000	3,925,000
C7	Incremental website customers after Vercel	C6-C3	75,000	125,000	175,000
C8	Average revenue per website customer	Composite	$200	$200	$200
C9	Incremental revenue from bottom-of-funnel website improvements	C7*C8	$15,000,000	$25,000,000	$35,000,000
C10	Operating margin	Composite	5.0%	5.0%	5.0%
Ct	Improved conversion rate	C9*C10	$750,000	$1,250,000	$1,750,000
	Risk adjustment	↓15%			
Ctr	Improved conversion rate (risk-adjusted)		$637,500	$1,062,500	$1,487,500
	Three-year total: $3,187,500			**Three-year present value: $2,575,225**	

INCREASED WEBSITE TRAFFIC

Evidence and data. Before Vercel, the interviewees' organizations lost website traffic due to site performance. Poor SEO, long page load times, and extended periods of site downtime all deterred visitors. Site performance issues were especially problematic during major events and holidays — e.g., Black Friday and Cyber Monday — when website traffic spiked. The inability to attract and sustain a high volumes of traffic meant that the organizations lost out on potential revenue.

"In terms of experience, you're getting the page quicker, it's rendering quicker — it's a better experience, even on low-end devices. … The pages are just much smaller. … Overall, our website is a better experience for users."

Director of infrastructure, financial services

With Vercel in place, the interviewees' front-end teams improved page load times, SEO, and site uptimes — thereby reducing bounce rates. By engaging more customers at the top of the funnel, the organizations increased revenues.

The interviewees explained the benefits of Vercel as follows:

- The manager of engineering in e-commerce explained how, after investing in Vercel, their team was able to manage spikes in website traffic around the holiday season with ease: "Last year [was a] complete nonevent, which is the best thing we could ever say about the holiday period. It's our biggest time of year. It's when we do our most sales. When we were [with our prior solution], we just were not confident. We were always worried. … But with Vercel, we don't even have to worry about scaling. [Traffic spikes are] just nonevents."
- The CTO in the media industry said: "Stability is the number one [goal]. We have peaks of traffic around [major news stories]. The joke before was that we've never had [a major news story] where the website hasn't broken. [After Vercel], our website just worked through [a major event]. So stability was paramount to me — [now that our site is on Vercel], it just works."

"We're seeing lightning-fast [load] times. We're very happy."

Senior engineer, financial services

The interviewees quantified the benefits of Vercel as follows:

- The manager of engineering in e-commerce said that before Vercel, website traffic during the holiday season often caused the site to crash. However, during holiday periods after the organization switched to Vercel, the site handled five to 10 times its usual traffic without any downtime. Direct-to-consumer sales via the organization's website account for nearly 25% of its total sales.
- The principal engineer in e-commerce explained that their organization's website was down for 5 to 8 minutes every day before Vercel. Since migrating the website to Vercel, the organization has experienced only two incidents of downtime over two years.
- The director of infrastructure in financial services said that once the organization switched to

Vercel, website impressions nearly doubled — increasing from 3 million to 5 million impressions a day. Thanks to improved SEO, the organization was able to invite more users to the site and convert them to customers. Faster page load times (e.g., through edge caching) not only improved SEO but also reduced bounce rates.

"Having a platform that's stable and reliable is huge for us because every minute that we lose is potentially lost revenue."

Manager of engineering, e-commerce

Modeling and assumptions. For the composite organization, Forrester assumes:

- Before Vercel, as many as 15% of website visitors bounce due to poor site performance.[5] For example, long page load times cause visitors to navigate away before becoming users. Similarly, downtime incidents deter visitors.
- The front-end team improves the website as modeled in the Increased Feature Delivery section.
- The website enhancements lead to corresponding reductions in the bounce rate. The bounce rate falls to 9% in Year 1, 4% in Year 2, and 1% in Year 3 as more website visitors become active users.
- The incremental website users convert to customers as described in Benefit C: Improved Conversion Rate.

Risks. This benefit is most likely to vary across organizations based on:

- Website traffic before Vercel.
- Percentage of website visitors who bounce before Vercel.
- Reduction in bounce rate due to website enhancements with Vercel.

Results. To account for these risks, Forrester adjusted this benefit downward by 15%, yielding a three-year, risk-adjusted total PV of $7.7 million.

Increased Website Traffic

Ref.	Metric	Source	Year 1	Year 2	Year 3
D1	Website users before Vercel	C1	250,000,000	250,000,000	250,000,000
D2	Percentage of website visitors who bounce before Vercel	Assumption	15%	15%	15%
D3	Website visitors before Vercel	D1*(100%+D2)	287,500,000	287,500,000	287,500,000
D4	Reduction in bounce rate due to website enhancements with Vercel	D2*R16	6%	11%	14%
D5	Percentage of website visitors who bounce after Vercel	D2-D4	9%	4%	1%
D6	Website users after Vercel	D3*(100%-D5)	261,625,000	276,000,000	284,625,000
D7	Incremental website users after Vercel	D6-D1	11,625,000	26,000,000	34,625,000
D8	User-to-customer conversion rate after Vercel	C5	1.53%	1.55%	1.57%
D9	Average revenue per website customer	C8	$200	$200	$200
D10	Incremental revenue from top-of-funnel website improvements	D7*D8*D9	$35,572,500	$80,600,000	$108,722,500
D11	Operating margin	C10	5.0%	5.0%	5.0%
Dt	Increased website traffic	D10*D11	$1,778,625	$4,030,000	$5,436,125
	Risk adjustment	↓15%			
Dtr	Increased website traffic (risk-adjusted)		$1,511,831	$3,425,500	$4,620,706
	Three-year total: $9,558,038			**Three-year present value: $7,676,989**	

UNQUANTIFIED BENEFITS

Interviewees mentioned the following additional benefits that their organizations experienced but that they were not able to quantify:

- **Collaboration between developers and business stakeholders.** Prior to investing in Vercel, developers and business users worked in silos that hindered effective collaboration on new features. With Vercel's advanced CI/CD pipelines and integration with digital asset management systems, teams were able to provide feedback and work in iterative cycles more quickly — ultimately releasing better, higher-quality features that met the standards of the technical and creative elements of the business.
    - The CTO in the media industry said: "[With Vercel], we can show our business users the features that we've got ready to go. Before, we used to tell them about upcoming features, but they never saw them until we went live. It's a big change to be able to show what we are working on to the product team and start a conversation with them ahead of time. That is a massive win for us."
- **Vercel enterprise support.** Interviewees described their relationship with Vercel as a partnership. They underscored the value of engaging with Vercel's support team, which includes skilled developers with deep knowledge of Next.js. Interviewees said that Vercel support helped them improve their workflows and quickly unstick projects.
    - The principal engineer in e-commerce said: "A really neat part of working with Vercel is [working with] the creators of Next.js, which you certainly wouldn't get from [another provider]. That's been awesome on the support front. … [Vercel's] customer support folks are dedicated to our account and are super-responsive. They are involved, and they know our business. Really, we can speak to them as teammates. [Their] actual responses have been tremendous."
    - The manager of engineering in e-commerce said: "From day one, [Vercel's support team has] been awesome. … They're big fans of ours and we're big fans of theirs. … We have a brilliant relationship with them. ... They're always there to answer our questions, whether it be framework problems or code issues. They're always there, and they almost feel like they're part of our team."
    - The CTO at the media organization said: "We have a weekly call with [Vercel's] customer success team… From day one, [their communication has] been very good and open. ... I can ping the customer team pretty much at any time, and I'll get a response back [quickly]."

"I don't have a relationship with any other vendor like I do with Vercel. I think we're really good partners. … They care about us, and we care about them. The support has been unrivaled. … I can't speak highly enough about them."

Manager of engineering, e-commerce

FLEXIBILITY

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

- **Innovating to remain on the cutting edge.** The ever-changing web development landscape demands businesses constantly adapt to be on the cutting edge and remain competitive in their industry. To that end, interviewees described how Vercel listened to their needs and worked to deliver new features that solved both existing and emerging problems.
    - The principal software engineer in e-commerce explained: "[Vercel] seem[s] to have an uncanny ability to read the market and roll out features that address needs. There have been times when we've needed things, waited a few weeks, and then Vercel [suddenly] came out with products to support [our needs]."
    - The managing engineer in e-commerce similarly described the value of Vercel's continuous feature development to address their specific needs and challenges: "Vercel keeps bringing out new features, [and] they always loop us into the beta processes because they know we love the new stuff. … They keep adding features to the platform, and we keep asking for features that we want for ourselves — for our needs — and they always listen and take our asks directly to the product team. They're great partners, and they do keep trying to better the platform. … It's just that level of commitment and care."
    - The director of infrastructure in financial services said: "There are people now who don't even remember the problems we had [before Vercel] because they're now so used to just writing code and shipping it. I think as new features drop and as Vercel matures — it's like a rocket ship. At the end of the day, we're going to be rewarded by Vercel's journey, not just our own."

> **"Phenomenally easy. [Vercel] is one of those amazing products that just works."**
>
> *CTO, media*

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	Implementation	$2,587,500	$0	$0	$0	$2,587,500	$2,587,500
Ftr	Next.js learning curve	$295,313	$0	$0	$0	$295,313	$295,313
Gtr	Ongoing management	$0	$47,174	$47,174	$47,174	$141,523	$117,316
Htr	Vercel subscription	$0	$242,660	$242,660	$242,660	$727,980	$603,460
	Total costs (risk-adjusted)	$2,882,813	$289,834	$289,834	$289,834	$3,752,316	$3,603,589

IMPLEMENTATION

Evidence and data. The biggest cost for the interviewees' organizations was the effort invested to migrate from their legacy websites to Next.js and Vercel. According to the interviewees, migrating to Next.js required the most effort. However, once the organizations were using Next.js, switching from another provider to Vercel was comparatively low-effort.

Most the interviewees' teams (four out of five) incrementally migrated their legacy websites to Next.js and Vercel over several months. These projects were substantial; they involved rewriting, refactoring, and rearchitecting legacy code. However, by leveraging Next.js and Vercel, the interviewees' teams migrated their legacy websites page-by-page without disrupting service to customers. For example, Vercel's intelligent traffic routing features made it possible for parts of the websites to use the new platforms while other parts of the websites continued to use legacy technologies. According to the interviewees, incrementally migrating to Next.js and Vercel not only reduced the burdens on their front-end teams but also allowed them to demonstrate the business value of their efforts early on. The interviewees said that business stakeholders recognized that the migration projects had long-term strategic value — both for developers and for customers.

> **"It felt like this was a golden opportunity. … We could rewrite the website to get it Vercel-friendly."**
>
> *Director of infrastructure, financial services*

The interviewees explained the costs of implementing Vercel as follows:

- The manager of engineering in e-commerce described an incremental migration: "The migration really went quite smoothly. … What was great was that we were able to run the old site and the new site in tandem. So ultimately, we were able to get our production version ready [on Vercel] at the same time we were still running in production on [our old provider]. … Our team could go in and test and make sure everything

looked good. … The [final] steps were really simple; the hookup was really easy."

The manager of engineering added: "When we moved to Vercel, we migrated all our existing apps over, and that was really very seamless. But that gave us the ability and the confidence to start building other apps. So when we went to Vercel, we [also] rebuilt the entire account experience in Next.js. … We delivered that within … months. That went really seamlessly."

- The senior engineer in financial services also described an incremental migration: "Next.js has a pattern for incremental migration for larger enterprise systems, and Vercel makes it easier by supporting a lot of out-of-the-box Next.js features at an infrastructure level. … Realistically, our site has thousands of pages, and rewriting and sending them all to Vercel initially — there's no way. And so the path for incremental adoption — the ability to split traffic and measure the difference and gain confidence that the new platform was working well and handling the volume of traffic and operating more performantly — was really nice. … It's now at the point that we just immediately [switch] new routes over to Vercel because we have a high degree of confidence that things will go well."
- The CTO in media described how their organization rebuilt the site: "We were experiencing slow performance, and new features were taking too long to develop, and so it was time for a technology refresh. … That is, a new CMS [content management system], a completely new front end, and a complete rethink about how we host the website. … Picking Next.js as the framework was the number one route for us. Once we chose the framework, we started at looking at different cloud providers to host the site. … After we had a contract with Vercel … we very quickly had a test environment set up."

The CTO continued: "The biggest challenge was the content migration. [The challenge] wasn't building the website … it was migrating the content [and rectifying] all the legacy things that get built up over 20-plus years of bad decisions. We could have had a basic website within a couple of months, but then we had to make it what we wanted and what the SEO team wanted."

> **"We really started to think about how users engage with our content and [realized] we could do a complete redesign. At the same time, we in engineering thought about a rebuild and the functional specs of what we wanted the site to do. So we came from two sides and met in the middle. … [Vercel] was the engineering solution and what the business needed to deliver the [user experience] it wanted."**
>
> *CTO, media*

The interviewees also quantified the following implementation costs:

- The director of infrastructure in financial services described a total implementation timeline of nine to 10 months to migrate to both Next.js and Vercel. For the first six to seven months, only 30% of the front-end team was involved. For three months, the entire front-end team contributed. The organization also used contractors.
- The manager of engineering in e-commerce said that migrating to Next.js took six to nine months

and involved 30% of the front-end team as well as other employees in IT and content.

- The principal engineer in e-commerce reported an eight-month implementation timeline with eight to 12 full-stack developers.
- The CTO in media said that migrating the legacy website to Next.js and Vercel took 13 months in total.
- The senior engineer in financial services said their team had been migrating the site Vercel incrementally over two years. Although implementation was still ongoing, the organization had already realized benefits from its investment.

Modeling and assumptions. For the composite organization, Forrester assumes:

- The implementation team consists of 15 developers (30% of the full front-end team). For several months, these developers focus on migrating the site full-time while the remaining developers continue with their regular tasks.
- The implementation team migrates the legacy website to Next.js in 10 months.
- Once the site is using Next.js, migrating to Vercel takes another two months (20% of the Next.js implementation timeline).
- In practice, the implementation team migrates the legacy website incrementally, and the composite organization begins accruing benefits before implementation is fully complete. However, for the purposes of modeling the economic impact of Vercel, Forrester has included the full implementation timeline in the initial year.

Risks. This cost is most likely to vary across organizations due to variability in Next.js implementation timelines. An organization's Next.js implementation timeline depends on both the state of its legacy environment and the resources it allocates to implementation. Also, organizations already using Next.js would not incur Next.js implementation costs, but they should still expect some costs associated with migrating from their legacy cloud providers to Vercel Frontend Cloud.

Results. To account for this risk, Forrester adjusted this cost upward by 15%, yielding a three-year, risk-adjusted total PV of $2.6 million.

Implementation

Ref.	Metric	Source	Initial	Year 1	Year 2	Year 3
E1	Front-end developers	Composite	50	50	50	50
E2	Average developer monthly salary (fully burdened)	A7/12	$12,500	$12,500	$12,500	$12,500
E3	Percentage of front-end team allocated to implementation	Interviews	30%	0%	0%	0%
E4	Front-end developers allocated to implementation	E1*E3	15	0	0	0
E5	Next.js implementation timeline (months)	Interviews	10	0	0	0
E6	Subtotal: Next.js implementation effort	E4*E5*E2	$1,875,000	$0	$0	$0
E7	Vercel implementation timeline (months)	Interviews	2	0	0	0
E8	Subtotal: Vercel implementation effort	E4*E7*E2	$375,000	$0	$0	$0
Et	Implementation	E6+E8	$2,250,000	$0	$0	$0
	Risk adjustment	↑15%				
Etr	Implementation (risk-adjusted)		$2,587,500	$0	$0	$0
	Three-year total: $2,587,500			**Three-year present value: $2,587,500**		

NEXT.JS LEARNING CURVE

Evidence and data. The interviewees said that their developers needed time to become proficient with Next.js, but they viewed the learning curve as typical for any new technology. According to the interviewees, the time to learn Vercel itself was negligible.

The interviewees explained the training costs as follows:

- The manager of engineering in e-commerce said: "If somebody already knows React [a JavaScript library for user interfaces], then Next.js has a little bit of a learning curve because it's different from your standard React app. But that's fine — most developers are fairly inquisitive and will read up as they start developing. … We brought in a handful of new people recently, and most of them got up to speed with Next.js pretty quickly. … Not a ton of developer time [was] needed there. … When it comes to [learning] Vercel, [the effort] has really been very minimal. We just give [developers] access and off they go."
- The senior engineer in finance said: "It's pretty straightforward. The docs are fantastic, and we'll often point people to those when they ask questions. Really, the only reason we have support [from Vercel] is because a lot of the developers using the platform aren't aware of some of the new features available to them. They're used to just shipping static assets, right? Now they have serverless functions, and so there's a bit of a knowledge gap — like how to diagnose and troubleshoot 500 errors. But that's not a platform issue. It's just knowledge and training as we grow people."
- The principal engineer in e-commerce said: "There's a little bit of a curve at the beginning, depending on where you're coming from. Understanding the difference between traditional server-rendered JavaScript applications and incremental static regeneration [ISR] can be a big hurdle. But on balance, it's been remarkably easy."
- The CTO in media said: "It all depends on where [developers] start on the learning curve. The more senior people picked up [Next.js] in weeks when we were building out stuff. … Some are struggling more than others — [they're] just at different stages in their careers. … We probably could have delivered the site far quicker if we had worked with an agency, but then we wouldn't have an engineering team to support it. So part of what we're trying to do is to make sure we've got a skilled team. … That takes time. You've got to invest in your team, and they've got to invest in themselves."

> **"[Next.js and Vercel are] remarkably easy. … We were able to ramp up rapidly. … It was not a tough experience. [Next.js] requires some JavaScript fundamentals but not much beyond that."**
>
> *Principal engineer, e-commerce*

The interviewees also quantified the following training costs:

- The CTO estimated that developers became fully proficient with Next.js and Vercel in three to six months, depending on their skill level.
- The manager of engineering in e-commerce shared that new hires typically learned the organization's full front-end stack in two months.
- According to the principal engineer in e-commerce, developers usually performed basic tasks (e.g., committing code) within one week

and more advanced tasks (e.g., leveraging ISR and APIs) after six to eight weeks.

Modeling and assumptions. For the composite organization, Forrester assumes:

- Fifty front-end developers become fully proficient with Next.js over three months.
- During those three months, the developers spend 15% of their time learning Next.js. This accounts for both dedicated effort to learn the new framework (e.g., reviewing documentation and tutorials) and reduced productivity while the framework is still unfamiliar.
- For the developers, the effort to learn Vercel is small enough to be negligible — most developers become fully proficient with Vercel in one to two days. Because the value of this time is so small, Forrester omits it from the model.

Risks. This cost is most likely to vary across organizations due to variability in the amounts of time developers need to learn Next.js. Developers may become fully proficient with Next.js in more or less time than modeled based their prior experiences, skills, etc.

Results. To account for this risk, Forrester adjusted this cost upward by 5%, yielding a three-year, risk-adjusted total PV of $295,000.

> **"It's pretty intuitive. It's pretty easy to get on board [Next.js and Vercel]. … We've deleted a ton of documentation for our legacy infrastructure. … [Next.js and Vercel] are vastly simpler, which means people have less cognitive overload."**
>
> *Director of infrastructure, financial services*

Next.js Learning Curve

Ref.	Metric	Source	Initial	Year 1	Year 2	Year 3
F1	Front-end developers	E1	50	50	50	50
F2	Average developer monthly salary (fully burdened)	E2	$12,500	$12,500	$12,500	$12,500
F3	Time to become fully proficient with Next.js (months)	Interviews	3	0	0	0
F4	Percentage of time spent becoming proficient with Next.js	Interviews	15%	0%	0%	0%
Ft	Next.js learning curve	F1*F3*F2*F4	$281,250	$0	$0	$0
	Risk adjustment	↑5%				
Ftr	Next.js learning curve (risk-adjusted)		$295,313	$0	$0	$0
	Three-year total: $295,313			**Three-year present value: $295,313**		

ONGOING MANAGEMENT

Evidence and data. The interviewees reported spending small amounts of time overseeing their organizations' Vercel instances — e.g., managing configurations, updates, and so on.

"We were looking for something that was highly maintained, that was battle-tested, that we were sure had legs. … [Vercel] basically came with batteries included, which was pretty important."

Senior engineer, financial services

The interviewees explained the ongoing management costs as follows:

- The principal engineer in e-commerce said: "[The effort is] minimal at this point. My team doesn't get too involved at all. I do [some work] as a matter of course, but it really doesn't extend much beyond me. And generally, I'm just keeping an eye on metrics and making sure nothing is wrong."

  The principal engineer acknowledged: "There have been a couple of times in the past when we have seen some infrastructure-related issues. Once, there was a bug in Next.js, and Vercel walked us through a patch for that in an hour. Another time was related to a downstream provider."

  They concluded: "So that's the sum total of infrastructure-related events! Over the course of the last three years, it's a pretty darn good track record. A lot better than we were doing on our own."

- The CTO in media said that Vercel required less overhead than the organization's prior solution: "We've repositioned people [who used to manage infrastructure] into roles they were better trained for. [And] we haven't had to hire anyone to manage the new infrastructure differently. [The work] is predominantly done by the SRE [site reliability engineering] team and me. They manage the releases; they manage the updates. We keep an eye on what Vercel releases to see what we should look at updating."

  The CTO continued: "With Vercel, we install a [code repository] integration and deploy code, and it just runs. The level of management from Vercel is a level above what we saw from other platforms. [Vercel] enabled us to go forward without any dedicated SRE assets. Vercel just kind of handles that for us."

The interviewees also quantified the following ongoing management costs:

- The senior engineer in financial services said that ongoing management of Vercel took up about 15% of both their time and the time of one other developer.

"In terms of administration in Vercel, I do next to nothing — which is great. … We're in a good place right now where the framework kind of looks after itself."

Manager of engineering, e-commerce

Modeling and assumptions. For the composite organization, Forrester assumes:

- Two front-end developers (4% of the team) are responsible for the ongoing management, maintenance, and administration of Vercel.
- These duties consume 15% of their time.

Risks. This cost is most likely to vary across organizations for idiosyncratic reasons. For example, the director of infrastructure in financial services reported an initial spike in management effort for reasons related to the organization's front-end architecture and an oversight during the Vercel implementation.

Results. To account for this risk, Forrester adjusted this cost upward by 5%, yielding a three-year, risk-adjusted total PV of $117,000.

Ongoing Management

Ref.	Metric	Source	Initial	Year 1	Year 2	Year 3
G1	Percentage of front-end team managing Vercel	Interviews	0%	4%	4%	4%
G2	Front-end developers managing Vercel	F1*G1	0	2	2	2
G3	Percentage of time spent managing Vercel	Interviews	0%	15%	15%	15%
G4	Time spent managing Vercel (hours)	G2*G3*2,080	0	624	624	624
G5	Developer hourly rate (fully burdened)	A8	$72	$72	$72	$72
Gt	Ongoing management	G4*G5	$0	$44,928	$44,928	$44,928
	Risk adjustment	↑5%				
Gtr	Ongoing management (risk-adjusted)		$0	$47,174	$47,174	$47,174
	Three-year total: $141,523			**Three-year present value: $117,316**		

VERCEL SUBSCRIPTION

Evidence and data. The interviewees believed that Vercel's subscription costs were appropriate for the premium services they received. For some of the interviewees, investing in Vercel had been a "build vs. buy" decision; these interviewees found that Vercel offered more features and greater capabilities than they could have ever built in-house. The interviewees consistently said that the value Vercel delivered was worth the price.

The interviewees explained the subscription costs for Vercel as follows:

- The senior engineer in financial services concluded: "[Building] even just the bare-bones features that Vercel provides would require at least two fully staffed teams for probably a year or more — and we'd probably have less uptime and worse performance. And so, from an economic standpoint, it makes sense to pay for that functionality because we get higher quality for less."
- The manager of engineering in e-commerce said: "[Vercel] is more expensive, without a doubt, but now we have a higher level of support. We have reliability. We have developer experience. So it's not apples to apples; it's apples to oranges. … I think it's a small price to pay. … [Vercel] is definitely more money … but we get a lot more value."
- The CTO in media also described how Vercel offered more value for the same price as the organization's prior provider: "We're spending about the same [with Vercel] as we were spending [with our old provider], but [now] we're getting all the compute on top. … We're getting extra features and all those other bits and pieces that we had to pay for before. … It is slightly more expensive than [another provider], but we need less resources to manage it."
- The principal engineer in e-commerce said: "We really appreciated the serverless implementation that Vercel offered. We could approximate that via [another cloud provider] ourselves, but the amount of effort and engineering time we would have had to put into platform-level considerations was just more than what we wanted to do. So that really swung the needle towards Vercel for us."

The interviewees' organizations had enterprise contracts with Vercel that included Vercel's highest level of support. Some interviewees also noted that they subscribed to extra Vercel user seats — more seats than the number of in-house developers — to accommodate contractors and others who needed part-time access.

> **"The pitch I'd give to a CFO or CEO is just this: You've got some of the brightest minds in the world working on one thing and doing that one thing really well. You should just pay them money to do that so you can focus on your business."**
>
> *Director of infrastructure, financial services*

Modeling and assumptions. For the composite organization, Forrester assumes:

- The composite organization purchases Vercel user seats for all 50 front-end developers.
- It purchases 13 more user seats (25% more user seats than the number of front-end developers) to reserve for contractors and other employees.
- The cost per Vercel user seat is $1,200 per year.

- The total cost for usage (e.g., bandwidth) and additional, optional Vercel platform features is $45,000 per year.
- The composite organization subscribes to Vercel's highest level of enterprise support for $100,000 per year.

Risks. This cost is most likely to vary across organizations based on:

- Number of Vercel user seats.
- Cost per Vercel user seat.
- Costs for additional Vercel platform usage.

Because enterprise contracts are highly variable, readers should contact Vercel for cost estimates tailored to their organizations.

Results. To account for these risks, Forrester adjusted this cost upward by 10%, yielding a three-year, risk-adjusted total PV of $603,000.

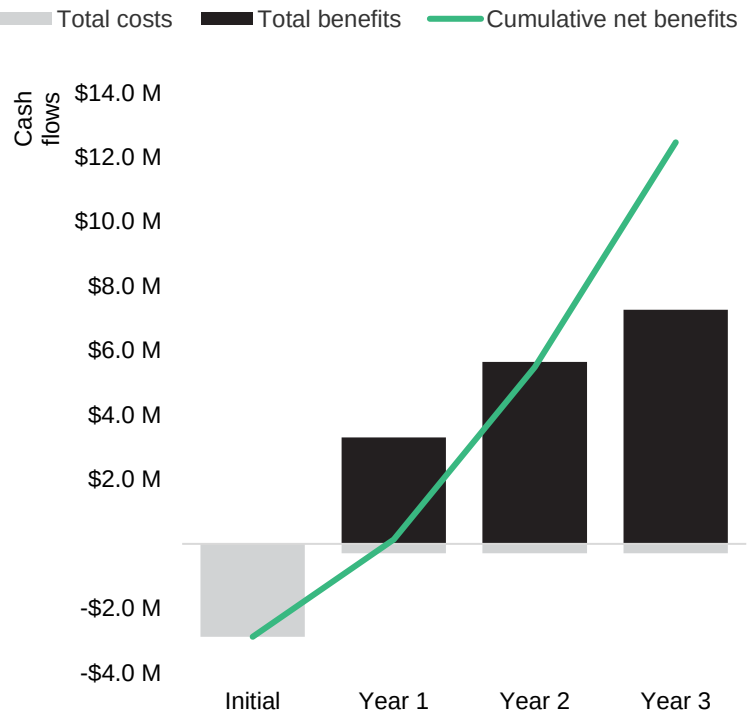
Vercel Subscription

Ref.	Metric	Source	Initial	Year 1	Year 2	Year 3
H1	Total developers	A1	0	50	50	50
H2	Vercel user seats	H1*125%	0	63	63	63
H3	Cost per Vercel user seat	Composite	$1,200	$1,200	$1,200	$1,200
H4	Vercel user seats	H2*H3	$0	$75,600	$75,600	$75,600
H5	Additional Vercel platform usage	Composite	$0	$45,000	$45,000	$45,000
H6	Vercel enterprise support	Composite	$0	$100,000	$100,000	$100,000
Ht	Vercel subscription	H4+H5+H6	$0	$220,600	$220,600	$220,600
	Risk adjustment	↑10%				
Htr	Vercel subscription (risk-adjusted)		$0	$242,660	$242,660	$242,660
	Three-year total: $727,980			**Three-year present value: $603,460**		

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, NPV, and payback period for the composite organization's investment. Forrester assumes a yearly discount rate of 10% for this analysis.

These risk-adjusted ROI, NPV, and payback period 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	($2,882,813)	($289,834)	($289,834)	($289,834)	($3,752,316)	($3,603,589)
Total benefits	$0	$3,306,206	$5,644,874	$7,265,081	$16,216,161	$13,129,189
Net benefits	($2,882,813)	$3,016,371	$5,355,040	$6,975,246	$12,463,845	$9,525,600
ROI						264%

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."

The initial investment column contains costs incurred at "time 0" or at the beginning of Year 1 that are not discounted. All other cash flows are discounted using the discount rate at the end of the year. PV calculations are calculated for each total cost and benefit estimate. NPV calculations in the summary tables are the sum of the initial investment and the discounted cash flows in each year. Sums and present value calculations of the Total Benefits, Total Costs, and Cash Flow tables may not exactly add up, as some rounding may occur.

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%.

PAYBACK PERIOD

The breakeven point for an investment. This is the point in time at which net benefits (benefits minus costs) equal initial investment or cost.

Appendix B: Potential Gross Profits By Industry

When technology solutions impact revenues, Forrester's Total Economic Impact methodology considers the benefits to organizations to be not top-line revenues but rather profits — i.e., revenues after subtracting associated costs. Forrester's standard approach is to calculate benefits based on operating margin; operating profits are defined as revenues minus both production costs and ongoing operating costs such as rent, salaries, marketing, etc. Forrester takes this approach because growing revenues typically involves scaling up operations, and Total Economic Impact benefit calculations try to isolate and reflect only economic impacts directly attributable to solutions.

However, some readers may wish to know the potential revenue impacts of Vercel on gross profits — i.e., on revenues minus production costs. The tables below show the gross profits for the composite organization based on profit margins that might be typical for industries of the interviewees' organizations. Forrester includes these tables for informational purposes only.

Gross Profits (Media)

Ref.	Metric	Source	Year 1	Year 2	Year 3
AA1	Total incremental revenue after Vercel	C9+D10	$50,572,500	$105,600,000	$143,722,500
AA2	Gross margin (media)	TEI standard	40%	40%	40%
AAt	Gross profits (media)	AA1*AA2	$20,229,000	$42,240,000	$57,489,000
	Risk adjustment	↓15%			
AAtr	Gross profits (media) (risk-adjusted)		$17,194,650	$35,904,000	$48,865,650
	Three-year total: $101,964,300			**Three-year present value: $82,017,713**	

Gross Profits (Financial Services)

Ref.	Metric	Source	Year 1	Year 2	Year 3
AB1	Total incremental revenue after Vercel	C9+D10	$50,572,500	$105,600,000	$143,722,500
AB2	Gross margin (financial services)	TEI standard	76%	76%	76%
ABt	Gross profits (financial services)	AB1*AB2	$38,435,100	$80,256,000	$109,229,100
	Risk adjustment	↓15%			
ABtr	Gross profits (financial services) (risk-adjusted)		$32,669,835	$68,217,600	$92,844,735
	Three-year total: $193,732,170			**Three-year present value: $155,833,655**	

Gross Profits (E-Commerce)					
Ref.	**Metric**	**Source**	**Year 1**	**Year 2**	**Year 3**
AC1	Total incremental revenue after Vercel	C9+D10	$50,572,500	$105,600,000	$143,722,500
AC2	Gross margin (e-commerce)	TEI standard	43%	43%	43%
ACt	Gross profits (e-commerce)	AC1*AC2	$21,746,175	$45,408,000	$61,800,675
	Risk adjustment	↓15%			
ACtr	Gross profits (e-commerce) (risk-adjusted)		$18,484,249	$38,596,800	$52,530,574
	Three-year total: $109,611,623			**Three-year present value: $88,169,042**	

Appendix C: Supplemental Material

Related Forrester Research

"Getting Started With Edge Development Platforms," Forrester Research, Inc., April 28, 2023

"Must-Have E-Commerce Features," Forrester Research, Inc., February 18, 2022

"The UX ROI For B2B Tech Vendors," Forrester Research, Inc., December 10, 2021

"Understand Developer Experience To Improve Business Outcomes," Forrester Research, Inc., June 21, 2023

Appendix D: Endnotes

[1] 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.

[2] Source: "Must-Have E-Commerce Features," Forrester Research, Inc., February 18, 2022.

[3] Source: "Margins by Sector (US)," New York University Leonard N. Stern School of Business, January 2023.

[4] Source: Ibid.

[5] According to Google, improving website performance above certain thresholds can reduce abandonment (when a visitor navigates away before a page loads) by up to 24%. Source: "The Science Behind Web Vitals," Google, May 21, 2020.


FORRESTER®